

## Natural for Mainframes

ユーティリティ

バージョン 4.2.5

October 2009

This document applies to Natural バージョン 4.2.5 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © Software AG 1979-2009. All rights reserved.

The name Software AG™, webMethods™, Adabas™, Natural™, ApplinX™, EntireX™ and/or all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. Other company and product names mentioned herein may be trademarks of their respective owners.

# 目次

1 ユーティリティ	1
2 ユーティリティのアクティブ化	3
3 目的別にグループ化されたユーティリティ	5
4 ADACALL ユーティリティ- Adabas ダイレクトコールの発行	7
ADACALL の呼び出し	8
ADACALL パラメータ	9
ADACALL コマンドと PF キー	11
ユーザー出口 ADAEXIT	14
5 DBLOG ユーティリティ - データベースコールのロギング	15
6 DBLOG の実行	17
データベースロギングの基本原則	18
データ処理と格納	19
DBLOG のアクティブ化と非アクティブ化	20
選択的な DBLOG の使用	21
7 DBLOG メニュー	25
[DBLOG Menu] の機能	26
ロギング制限の指定	28
Adabas バッファの指定	28
8 DBLOG トレース画面	31
Adabas コマンド用の [DBLOG Trace] 画面	32
DL/I コール用の [DBLOG Trace] 画面	39
SQL ステートメント用の [DBLOG Trace] 画面	41
9 DBLOG スナップショット機能	45
Adabas コマンドのスナップショット機能	46
DL/I コール用のスナップショット機能	48
SQL ステートメント用のスナップショット機能	51
10 TEST DBLOG コマンド	53
構文図	54
キーワードの説明	55
11 INPL ユーティリティ	57
INPL ユーティリティの概要	58
ライブラリのみロード	63
DDM のみロード	64
エラーメッセージのみロード	64
すべてのオブジェクトをロード	65
製品インストールの置き換え	65
INPL ファイルのスキャン	66
Natural Security リカバー	67
12 NATPAGE ユーティリティ - 画面キャプチャ	69
13 NATRJE ユーティリティ- Natural リモートジョブエントリ	71
NATRJE の全般的な情報	73
Natural プログラムからの NATRJE の呼び出し	73
NATRJE リターンコード	78

UTM/TIAM に適用可能な NATRJE 機能 .....	80
14 オブジェクトハンドラ .....	83
15 オブジェクトハンドラの全般的な情報 .....	85
オブジェクト転送の原理 .....	86
オブジェクトハンドラの呼び出し .....	88
バッチまたはダイレクトコマンドの呼び出し .....	89
Natural プログラムからのオブジェクトハンドラコマンドの発行 .....	90
レポート、再スタート、およびトレース用のテキストメンバ .....	90
Natural Security .....	91
標準の PF キー .....	91
16 機能 .....	93
17 ウィザード .....	95
手順 1 - プロシージャの開始 .....	96
手順 2 - ワークファイルへのオブジェクトのアンロード/ワークファイルから のオブジェクトのロード/ワークファイルのオブジェクトのスキャン .....	97
手順 3 - パラメータの設定 .....	99
手順 4 - オブジェクトの選択 .....	100
手順 5 - 処理の実行 .....	101
手順 6 - 処理の続行 .....	102
18 上級ユーザー .....	103
上級ユーザーのアクティブ化 .....	104
オブジェクトの処理 .....	104
19 ロードの再スタート .....	107
20 表示 .....	109
Natural ライブラリオブジェクト .....	110
Natural システムエラーメッセージ .....	111
Natural コマンドプロセッサソース .....	112
FDT .....	113
Natural 関連オブジェクト .....	114
DDM .....	115
21 検索 .....	117
22 管理 .....	119
ワークプランライブラリ内の利用可能なワークプランのリストの作成 .....	120
新しいワークプランの作成 .....	122
ワークプランライブラリの変更 .....	124
23 システムファイルの選択 .....	127
24 ライブラリの選択 .....	129
25 システムエラーメッセージの選択 .....	131
列およびコマンド .....	132
26 オブジェクトの選択 .....	135
リスト画面の列およびコマンド .....	136
27 オブジェクト指定 .....	141
28 オブジェクト指定 - ワークファイルのすべてのオブジェクト .....	143
29 オブジェクト指定 - Natural ライブラリオブジェクト .....	145
Natural ライブラリオブジェクト .....	146

Natural ライブラリオブジェクトの詳細 .....	147
Natural ライブラリオブジェクトのプロパティ .....	150
Natural ライブラリオブジェクトの例外 .....	151
Natural ライブラリオブジェクトの例外のプロパティ .....	152
30 オブジェクト指定 - Natural システムエラーメッセージ .....	155
Natural システムエラーメッセージ .....	156
Natural システムエラーメッセージの詳細 .....	156
Natural システムエラーメッセージの例外 .....	157
31 オブジェクト指定 - Natural コマンドプロセッサ .....	159
Natural コマンドプロセッサ .....	160
Natural コマンドプロセッサソースの例外 .....	161
32 オブジェクト指定 - Natural 関連オブジェクト .....	163
Natural プロファイル .....	164
Natural デバッグ環境 .....	166
Natural DL/I サブファイル .....	167
33 オブジェクト指定 - DDM .....	171
DDM .....	172
DDM のプロパティ .....	173
DDM の例外 .....	174
34 オブジェクト指定 - FDT .....	175
35 SELECTION または LIST ワークプランの使用 .....	177
36 Settings .....	179
設定画面のフィールド .....	180
その他のオプションの設定 .....	182
グローバルパラメータの設定 .....	188
37 ワークプラン .....	193
ワークプランの作成、選択、および変更 .....	194
ワークプランの内容 .....	194
ワークプランの例 .....	195
ワークプランの参照 .....	196
38 名前と日時の指定 .....	199
名前 .....	200
日付 .....	201
時刻 .....	202
39 ワークファイル .....	205
ワークファイルの割り当て .....	206
ワークファイルフォーマット .....	207
40 ダイレクトコマンド .....	209
41 基本的なコマンド構文 .....	211
42 select-clause .....	215
select-clause の構文 .....	216
SELECTION または LIST ワークプラン .....	216
Natural ライブラリオブジェクトと DDM の選択 .....	217
Natural 関連デバッグ環境の選択 .....	224
Natural 関連プロファイルの選択 .....	225

Natural 関連 DL/I サブファイルの選択 .....	227
Natural システムエラーメッセージの選択 .....	228
Natural コマンドプロセッサソースの選択 .....	230
FDT の選択 .....	232
アプリケーションの選択 .....	233
削除命令のためのオブジェクト選択 .....	235
ヘルプテキストの選択 .....	238
43 オブジェクトリスト - LIST ワークプラン .....	239
object-type-and-location の構文 .....	240
object-name-description の構文 .....	242
オブジェクトリストの例 .....	243
44 parameter-setting .....	245
parameter-setting の構文 .....	246
parameter-setting のキーワードの説明 .....	247
45 option-setting .....	251
option-setting の構文 .....	252
option-setting のキーワードの説明 .....	254
46 ダイレクトコマンドの使用例 .....	261
同一プラットフォームのオブジェクトのアンロード .....	262
異なるプラットフォームのオブジェクトのアンロード .....	263
内部フォーマットでのオブジェクトのロード .....	264
転送フォーマットでのオブジェクトのロード .....	264
47 ナビゲーションと特別な機能のためのコマンド .....	265
48 バッチのコンディションコードとユーザー出口ルーチン .....	271
バッチで返されるコンディションコード .....	272
ユーザー出口ルーチンの適用 .....	272
使用可能なユーザー出口ルーチン .....	273
49 Tools .....	275
ステータス .....	276
最新の結果 .....	276
トレース .....	276
レポート .....	277
50 プロファイル設定 .....	279
PROFILES の編集 .....	280
パラメータ指定 .....	281
51 NATUNLD/NATLOAD および SYSTRANS からオブジェクトハンドラへの移行 .....	285
個々のコマンドの変換 .....	286
ユーザー出口ルーチンを使用したコマンドの処理 .....	288
OBJHAPI を使用した SYSTRANS コマンドの処理 .....	289
サポートされていない SYSTRANS オプション .....	289
52 レコーディングユーティリティ .....	291
記録の目的 .....	292
記録されるデータおよび機能 .....	292
セッションの記録 .....	293

記録の再生 .....	294
記録の操作 .....	296
53 SYSAPI ユーティリティ- Natural アドオン製品の API .....	297
前提条件 .....	298
SYSAPI の呼び出しと終了 .....	298
API プログラム例のリスト .....	299
キーワード検索の実行 .....	300
API プログラム例の使用 .....	301
54 SYSBPM ユーティリティ - バッファプールの管理 .....	303
55 SYSBPM の呼び出しと操作 .....	305
SYSBPM の呼び出し .....	306
オンラインヘルプ .....	307
SYSBPM メインメニュー - フィールド、機能、およびコマンド .....	308
Sysplex 環境での SYSBPM .....	311
56 オブジェクトのリスト .....	313
[List Objects] の呼び出し .....	314
列および選択オプション .....	315
PF キーとダイレクトコマンド .....	325
行コマンド .....	326
57 オブジェクトの削除 .....	327
58 ディレクトリ情報 .....	329
バッファプールオブジェクトのフィールド .....	331
バッファプールキャッシュオブジェクトのフィールド .....	332
PF キーとダイレクトコマンド .....	333
59 16 進表示 .....	335
PF キーとダイレクトコマンド .....	336
60 ワークファイルへの書き込み .....	337
61 ソート済み抽出の表示 .....	339
62 バッファプール統計 .....	341
一般バッファプール統計 .....	342
バッファプールのロード/ロケート統計 .....	345
バッファプールフラグメント .....	349
内部機能使用率 .....	351
バッファプールのハッシュテーブル統計 .....	351
パフォーマンスに関するヒント .....	354
PF キーとダイレクトコマンド .....	358
63 バッファプールキャッシュ統計 .....	361
一般バッファプールキャッシュ統計 .....	362
バッファプールキャッシュコール統計 .....	364
バッファプールキャッシュのハッシュテーブル統計 .....	366
パフォーマンスに関するヒント .....	367
PF キーとダイレクトコマンド .....	367
64 バッファプールの選択 .....	369
[Select Buffer Pool] の呼び出し .....	370
バッファプールの表示 .....	371

Reset Buffer Pool .....	371
65 ブラックリストの管理 .....	373
ブラックリストの管理 .....	374
オブジェクトセットのリスト .....	378
オブジェクトセットの編集 .....	379
ブラックリストへのオブジェクトセットの追加 .....	382
ブラックリストからのオブジェクトセットの削除 .....	382
オブジェクトセットソースオブジェクトの削除 .....	383
ユーティリティによる追加のオブジェクトセットの管理 .....	384
バッチモードでのブラックリストの管理 .....	384
66 プリロードリストの管理 .....	387
List Preload Lists .....	388
プリロードリストの編集 .....	389
バッファプールからのプリロードリストの生成 .....	392
プリロードリストの削除 .....	394
ユーティリティによる追加の管理機能 .....	394
67 パフォーマンスの考慮事項 .....	395
内部高速検索テーブル .....	396
Steplibs での検索 .....	397
オブジェクトの再利用と保持 .....	398
ローカルバッファプールとグローバルバッファプールの比較 .....	398
検索順序のモニタリング .....	399
68 SYSBPM ダイレクトコマンド .....	403
SORT .....	409
SORT BPC .....	409
69 バッチ処理 .....	411
関連トピック .....	412
70 アプリケーションプログラミングインターフェイス .....	415
71 SYSCP ユーティリティ - コードページの管理 .....	417
SYSCP の呼び出しと終了 .....	418
ソースのコードページの管理 .....	420
すべてのコードページ .....	432
Unicode プロパティ .....	437
72 SYSEDT ユーティリティ - エディタバッファプールサービス .....	439
Natural Security ライブラリプロファイルの定義 .....	440
SYSEDT の呼び出しと操作 .....	440
General Information .....	441
Generation Parameters .....	443
Users .....	444
Logical Files .....	445
Recovery Files .....	446
System Administration Facilities .....	447
73 SYSERR ユーティリティ .....	449
74 メッセージの全般的な情報 .....	451
メッセージタイプ .....	452



メッセージの言語 .....	453
メッセージの発行 .....	454
Natural システムショートメッセージの取得 .....	455
ユーザー定義ショートメッセージの取得 .....	455
メッセージ情報の取得 .....	456
75 SYSERR 呼び出し .....	457
76 機能 .....	459
メッセージの追加 .....	460
メッセージの削除 .....	462
メッセージの表示 .....	463
メッセージの変更 .....	464
メッセージの出力 .....	466
メッセージのスキャン .....	467
リストからのメッセージの選択 .....	469
メッセージの他の言語への翻訳 .....	471
システムファイルへのメッセージオンコピー - SYSMAIN ユーティリティ .....	473
77 パラメータ .....	475
Message Type .....	476
Library .....	476
Message Number .....	476
Language Codes .....	477
78 ダイレクトコマンド .....	479
79 大文字 - ERRUPPER .....	481
80 文字置換 - ERRCHAR .....	483
81 メッセージのアンロード - ERRULDUS .....	485
82 メッセージのロード - ERRLODUS .....	487
83 アプリケーション プログラミングインターフェイス USR0020P .....	489
84 SYSEXT ユーティリティ-Natural アプリケーションプログラミングインターフェイス .....	491
SYSEXT の基本機能 .....	492
SYSEXT の呼び出しと終了 .....	493
SYSEXT ユーティリティ画面 .....	495
SYSEXT ユーティリティ機能の実行 .....	495
Natural API の使用 .....	499
Natural API のリスト .....	500
85 SYSMAIN ユーティリティ - オブジェクトメンテナンス .....	505
86 SYSMAIN の全般的な情報 .....	507
SYSMAIN の基本的な機能 .....	508
オブジェクトタイプとストレージの場所 .....	509
機能の概要 .....	509
87 SYSMAIN の呼び出しと終了 .....	511
オンラインまたはバッチでの SYSMAIN の呼び出し .....	512
アプリケーションプログラミングインターフェイスでの SYSMAIN の呼び出し .....	513
SYSMAIN の終了 .....	514

88	メニュー機能とコマンドの使用 .....	517
	メニュー機能の実行 .....	518
	コマンドの実行 .....	521
	機能の説明 .....	523
	機能処理とレポート .....	530
	SYSMAIN オンラインヘルプ .....	535
89	プログラミングオブジェクト処理 .....	539
	[Programming Objects] メニューのフィールド .....	540
	プロファイルパラメータ RECAT の使用 .....	543
	プログラミングオブジェクトの選択リスト .....	543
	XRef について .....	548
	追加条件の指定 .....	550
	プログラミングオブジェクトのダイレクトコマンド構文 .....	551
90	デバッグ環境処理 .....	555
	デバッグ環境メニューのフィールド .....	556
	デバッグ環境の選択リスト .....	558
	デバッグ環境のダイレクトコマンド構文 .....	559
91	エラーメッセージ処理 .....	563
	エラーメッセージメニューのフィールド .....	564
	エラーメッセージの選択リスト .....	566
	エラーメッセージ番号の変更 .....	568
	言語の指定 .....	569
	エラーメッセージのダイレクトコマンド構文 .....	569
92	プロファイル処理 .....	573
	プロファイルメニューのフィールド .....	574
	プロファイルの選択リスト .....	576
	プロファイルのダイレクトコマンド構文 .....	577
93	ルール処理 .....	581
	ルールメニューのフィールド .....	582
	ルールの選択リスト .....	584
	ルールのダイレクトコマンド構文 .....	585
94	DL/I サブファイル処理 .....	589
	DL/I サブファイルメニューのフィールド .....	590
	DL/I サブファイルの選択リスト .....	591
	DL/I サブファイルのダイレクトコマンド構文 .....	592
95	DDM 処理 .....	595
	DDM メニューのフィールド .....	596
	DDM の選択リスト .....	598
	DDM のダイレクトコマンド構文 .....	599
96	ダイレクトコマンドのキーワードと変数 .....	603
	キーワードの説明 .....	604
	名前の範囲の指定 .....	615
97	SYSMAIN に発行される特殊コマンド .....	617
98	処理ステータスとエラー通知 .....	621
	オブジェクトの拒否と理由 .....	622

ステータスメッセージ .....	624
SYSMAIN エラー通知 .....	628
99 管理者向けの特別な考慮事項 .....	633
ファイルセキュリティ .....	634
Natural Security .....	635
ユーザー出口ルーチン .....	636
100 SYSNCP ユーティリティ .....	643
SYSNCP ユーティリティについて .....	644
SYSNCP の呼び出し .....	650
プロセッサの選択 .....	651
ヘッダーレコード .....	652
キーワードのメンテナンス .....	662
機能のメンテナンス .....	667
ランタイムアクション .....	672
プロセッサのカタログ化 .....	678
管理者サービス .....	678
セッションプロファイル .....	685
101 SYSPARM ユーティリティ .....	689
SYSPARM の呼び出し .....	690
List Profiles .....	691
Display Profile .....	692
Add New Profile .....	692
Modify Profile .....	693
プロファイルの編集 .....	693
Copy Profile .....	695
Delete Profile .....	696
ダイレクトコマンドとバッチ処理 .....	696
102 SYSRDC ユーティリティ .....	705
SYSRDC の機能コンポーネント .....	706
データ収集イベント .....	707
収集されるデータ .....	708
Natural Data Collector のアクティブ化 .....	710
トレースの記録 .....	711
外部モニタリング/計算用ユーザー出口 .....	712
CMRDC インターフェイスの呼び出し .....	713
103 SYSRPC ユーティリティ .....	721
104 SYSRPC の基本的な機能 .....	723
SYSRPC の呼び出し .....	724
SYSRPC の終了 .....	725
オンラインヘルプの起動 .....	725
105 サービスディレクトリメンテナンス .....	727
サービスディレクトリの概念 .....	728
Service Directory Maintenance の呼び出し .....	730
[Service Directory] 画面のフィールド .....	732
Service Directory Maintenance のコマンド .....	734

論理ノード名および論理サービスの定義 .....	737
106 スタブ生成 .....	739
107 パラメータ指定による単一スタブの生成 .....	741
Stub Generation 機能の使用 .....	742
パラメータの指定 .....	745
スタブ生成の例 .....	747
108 複数スタブの生成 .....	749
SYSRPC SGMASS コマンドの使用 .....	750
名前の指定と圧縮 .....	751
109 サイズ要件の計算 .....	753
SYSRPC CSMASS コマンドの使用 .....	754
名前の指定と圧縮 .....	756
110 パラメータメンテナンス .....	757
Parameter Maintenance の呼び出し .....	758
「NTRPC/RPC キーワードサブパラメータの指定」を参照してください。 .....	758
111 サーバーコマンドの実行 .....	761
Server Command Execution の使用 .....	762
サーバーに対する ping の実行 .....	763
サーバーの終了 .....	765
112 リモートディレクトリメンテナンス .....	769
Remote Directory Maintenance の使用 .....	770
リモートディレクトリのエディタ画面のフィールド .....	771
Remote Directory Maintenance のコマンド .....	772
113 SYSTP ユーティリティ .....	775
114 SYSTP の呼び出しと機能の実行 .....	777
115 SYSTP ユーティリティ画面の使用 .....	779
116 一般的な SYSTP 機能 .....	781
Natural モニタリング (SYSMON) .....	782
Natural 出力/ワークファイル - SYSFILE .....	785
Natural スワップ情報 .....	786
バッファ使用統計 (BUS) .....	790
Natural サブシステムおよびロールサーバー情報 .....	792
Natural スレッド使用統計 .....	793
Natural ライセンス情報 .....	796
117 CICS での SYSTP 機能 .....	797
Natural ユーザーセッション .....	798
Natural ロール機能 .....	803
Natural Thread Groups .....	804
Natural ストレージスレッド .....	805
NCI グローバルシステム情報 .....	806
NCI 生成オプション .....	808
Natural スレッドグループ定義 .....	808
固有の Natural ユーザーセッション .....	810
CICS タスク情報 .....	810
システム管理機能 .....	810












---











118 IMS/TM での SYSTP 機能 .....	813
ブロードキャスト .....	814
環境データの表示 .....	814
モニタリング .....	815
複数セッション .....	815
適用された NII ZAP .....	815
119 TIAM および UTM での SYSTP 機能 .....	817
P-Key ユーティリティ .....	818
共通メモリプールの表示 .....	822
120 バッチでの SYSTP (CICS セッション用) .....	823
バッチモードでの SYSTP の呼び出し .....	824
ログファイルの評価 .....	824
索引 .....	827



# 1 ユーティリティ

『ユーティリティ』ドキュメントでは、Naturalでユーティリティが呼び出される方法と、Naturalで使用可能なユーティリティについて説明します。

 ユーティリティのアクティブ化	Naturalでユーティリティが呼び出される方法について説明します。
 目的別にグループ化されたユーティリティ	すべてのユーティリティを目的別にグループ化してリストします。
 <b>ADACALL</b>	このユーティリティは、Adabasダイレクトコール（ネイティブコマンド）を直接 Adabas データベースに対して発行します。
 <b>DBLOG</b>	このユーティリティは、データベースコールをログに記録します。これにより、Naturalプログラムによって発行されたAdabasコマンド、DL/Iコール、またはSQLステートメントが示されます。
 <b>INPL</b>	このユーティリティは、Software AGにより提供されたNaturalオブジェクトをロードまたはスキャンします。
 <b>NATPAGE</b> 画面キャプチャ	このユーティリティは、Naturalセッション中の画面（マップおよびレポート）を取得します。
 <b>NATRJE</b>	このユーティリティは、スケジューリングおよび実行のためにJCLカードをNaturalプログラムからオペレーティングシステムに送信します。
 オブジェクトハンドラ	このユーティリティは、Natural環境での分散のためにNaturalオブジェクトとNatural以外のオブジェクトを処理します。これは、ソース環境のオブジェクトをワークファイルにアンロードし、それらをワークファイルからターゲット環境にロードすることによって実行されます。
 記録	このユーティリティは、Naturalセッション中に入力されたコマンドと入力データを記録します。記録したセッションを再実行します。
 <b>SYSAPI</b>	このユーティリティは、Naturalアドオン製品により提供されたアプリケーションプログラミングインターフェイス（API）を検索します。
 <b>SYSBPM</b>	このユーティリティは、Naturalバッファプールをモニタリングおよび制御します。

 <b>SYSCP</b>	このユーティリティは、コードページ情報を提供し、Naturalソースオブジェクトのコードページを管理するために使用できます。
 <b>SYSEDT</b>	このユーティリティは、エディタバッファプールのパラメータおよびランタイム情報を表示します。パラメータを変更し、論理的なワークファイルとリカバリファイルを削除します。
 <b>SYSERR</b>	このユーティリティは、アプリケーション固有のメッセージを作成します。さらに、既存の Natural システムメッセージのテキストを修正するために使用できます（推奨できません）。
 <b>SYSEXT</b>	このユーティリティは、Natural アプリケーションプログラミングインターフェイス (API) を検索します。
 <b>SYSMAIN</b>	このユーティリティは、コピー、移動、削除などのオブジェクト操作を Natural で実行します。
 <b>SYSNCP</b>	このユーティリティは、Natural アプリケーションに対するコマンド方式のナビゲーションシステムを定義します。
 <b>SYSPARM</b>	このユーティリティは、Natural パラメータプロファイルを作成し、メンテナンスします。
 <b>SYSRDC</b>	このユーティリティは、Natural アプリケーション内の処理フローに関するモニタリングデータと計算データを記録します。
 <b>SYSRPC</b>	このユーティリティは、Natural リモートプロシージャコール環境を確立し、メンテナンスします。
 <b>SYSTP</b>	このユーティリティは、Natural の TP モニタ固有の特性をモニタリングおよび制御します。



## 2 ユーティリティのアクティブ化

---

Natural では、FNAT システムファイル内の対応するユーティリティライブラリにログオンせずに Natural ユーティリティを呼び出します。この結果、Natural では、グローバルデータエリア (GDA) またはアプリケーションに依存しない変数 (AIV)、あるいはその両方が維持されます。現在のユーザーライブラリおよび設定は維持されます (GDA または AIV、あるいはその両方をリセットするには、『パラメータリファレンス』のプロファイルパラメータ FREEGDA を参照)。

アプリケーション環境の設定を維持するには、ユーティリティライブラリにログオンしないでください。代わりに、ユーティリティに対応する Natural システムコマンドを使用してユーティリティを呼び出します。

ユーティリティが終了すると、ユーティリティを呼び出したライブラリに戻ります。ただし、ユーティリティを呼び出す前にユーティリティライブラリに明示的にログオンした場合は、ユーティリティの終了後、その (ユーティリティ) ライブラリから移動しません。

例外：

オブジェクトソースはアクティブなライブラリ内でのみ編集できるため、SYSEXT ユーティリティおよび SYSEXV アプリケーションでは、対応するユーティリティライブラリへの暗黙的なログオンを実行します。

Natural Security で Natural ユーティリティの使用を制御する方法の詳細については、『*Natural Security*』ドキュメントの「*Protecting Utilities*」セクションを参照してください。

Natural Security がインストールされていない場合は、ユーザー出口ルーチン UTI-EX01 を使用して Natural ユーティリティの使用を制御できます。このユーザー出口ルーチンのプログラムソースは、ライブラリ SYSEXT でソースオブジェクト UTI-SX01 として提供されています。

### ▶手順 2.1. UTI-EX01 をアクティブにするには

- 1 ソースオブジェクト UTI-SX01 を UTI-EX01 の名前で CATALOG または STOW します。

ユーザー出口ルーチンのソースオブジェクト（要件に応じて変更可能）およびカタログ化オブジェクトが更新インストールで上書きされないように、別の名前が使用されます。

- 2 FNAT または FUSER システムファイル内のライブラリ SYSTEM に UTI-EX01 をコピーします。

ユーザー出口ルーチンの詳細については、ライブラリ SYSEXT の UTI-SX01 のソースオブジェクトを参照してください。

### 3 目的別にグループ化されたユーティリティ

---

次のリストは、すべての Natural ユーティリティを目的別に示しています。

管理	デバッグ	モニタリング	オブジェクト転送
SYSAPI	ADACALL	SYSBPM	INPL
SYSCP	DBLOG	SYSEDT	オブジェクトハンドラ
NATRJE	デバッグ	SYSRDC	SYSMAIN
SYSBPM	DUMP	SYSTP	
SYSEDT	NATPAGE 画面キャプチャ		
SYSERR	レコーディング		
SYSEXT	SYSRDC		
SYSNCP			
SYSPARM			
SYSRPC			
SYSTP			

---

# 4 ADACALL ユーティリティ - Adabas ダイレクトコール の発行

---

▪ ADACALL の呼び出し .....	8
▪ ADACALL パラメータ .....	9
▪ ADACALL コマンドと PF キー .....	11
▪ ユーザー出口 ADAEXIT .....	14

ユーティリティ ADACALL を使用して、学習、テスト、および問題分析のために Adabas データベースへ Adabas ダイレクトコール（ネイティブコマンド）を発行することができます。

ユーティリティ ADACALL は、ライブラリ SYSADA にあります。

## ADACALL の呼び出し

### ▶手順 4.1. ADACALL を呼び出すには

- 次のシステムコマンドを入力します。

```
SYSADA
```

次の例のような ADACALL メイン画面が表示されます。

```
15:53:32          ***** NATURAL ADACALL UTILITY *****          2006-12-14
User SAG          - ADABAS Direct Calls -
Mode Char
*** Control Block ***          First Byte 30          Call No. 45
  Cmd L3          Cmd ID SAG          File 316          Database 10
Resp 0          ISN 382          ISQ 0          ISL 0
  FBL 210          RBL 980          SBL 140          VBL 140          IBL 0
COP1          COP2          User Area          Cmd Time 4
Addition1          Addition2          Addition3          Addition4          Addition5
AA]?          227 48
*** Buffer Areas ***
Format AA,AC,AE.

Record 11111003ARTHUR          DENT

Search

Value

ISN
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Main Exit          Char Hex View Prnt Run Init Canc
```

ADACALL メイン画面で必要なパラメータ値を指定し、PF10（実行）を選択するか、またはコマンド行で ADACALL コマンド EXEC を入力して、Adabas コマンドを実行します。

上の画面例では、employees ファイルの論理読み込みのために Adabas コマンド L3 が実行されました。

バッファの一部のみが表示されます。ただし、コントロールブロックは完全に表示されます。いずれかのADACALLダイレクトコマンドまたは次のPFキーを使用することにより、バッファを完全に表示することができます。

## ADACALL パラメータ

ADACALL メイン画面で指定できるパラメータを次に示します。ADACALL オンラインヘルプ機能を使用すると、パラメータの概要を表示することができます。

### ▶手順 4.2. オンラインヘルプ機能を起動するには

- ヘルプが必要なフィールドにカーソルを置き、疑問符 (?) を入力するか、または PF1 を選択します。ただし、読み取り専用フィールドでは、PF1 のみが適用されます。

詳細については、Adabas のドキュメント『コマンドリファレンス』と『メッセージおよびコード』を参照してください。

パラメータ	説明	
<b>Mode</b>	バッファの内容の表示モードを示します。	
	Char	文字値。
	Hex	16 進値。
	モードを変更するには、ADACALL コマンド <b>CHAR</b> および <b>HEX</b> を参照してください。	
<b>Call No.</b>	セッションの開始以降に実行されたコマンドの数。	
<b>First Byte</b>	Adabas コントロールブロックの先頭バイト。 データベース ID (DBID) およびファイル番号 (FNR) が、1 バイトで表現されるか、または 2 バイトで表現されるかを示します。 H'00' = 1 バイトの DBID、FNR (ファイル番号 1~255) H'30' = 2 バイトの DBID、FNR (255 より大きいファイル番号)	
<b>Cmd</b>	Adabas コマンド。 <b>Adabas OP コマンド</b> を入力して実行し、次の関連するセクションで説明するパラメータを指定します。	
<b>Cmd ID</b>	コマンド ID。	
<b>File</b>	ファイル番号。 <b>First Byte</b> が H'00' に設定されている場合	

パラメータ	説明
	3桁のファイル番号であり、 <b>Database</b> は0に等しくありません。  <b>First Byte</b> が H'30' に設定されている場合 5桁のファイル番号。
<b>Database</b>	データベース ID (DBID)。デフォルトは、現在の Natural セッションの FUSER ファイルの DBID (上記「File」を参照)。  <b>First Byte</b> が H'30' に設定されている場合、データベース番号は、実行時に Adabas コントロールブロックのレスポンスコードフィールドに移動されます。
<b>Resp</b>	コマンドの実行後に返されるレスポンスコード。
<b>ISN</b>	内部シーケンス番号。
<b>ISQ</b>	ISN 数。
<b>ISL</b>	ISN リストの最小 ISN 値。
<b>FBL</b>	バイト単位のフォーマットバッファ長 (最大 210)。
<b>RBL</b>	バイト単位のレコードバッファ長 (最大 980)。
<b>SBL</b>	バイト単位のサーチバッファ長 (最大 140)。
<b>VBL</b>	バイト単位のバリューストックバッファ長 (最大 140)。
<b>IBL</b>	バイト単位の ISN バッファ長 (最大 200)。
<b>COP1</b>	コマンドオプション 1。
<b>COP2</b>	コマンドオプション 2。
<b>User Area</b>	コントロールブロックのユーザーエリア。
<b>Cmd Time</b>	コマンドの実行に要する時間。便宜上、100 分の 1 秒に変換されます。
<b>Addition1</b>	アディクション 1。
<b>Addition2</b>	アディクション 2。呼び出しが成功した場合、読み込み中のレコードの圧縮長およびフォーマットバッファを介して要求されたデータの非圧縮長が表示されます。ゼロ以外のレスポンスが返され、エラーの原因が無効なフォーマットバッファである場合、エラーが発生したフィールドおよびそのフィールドのフォーマットバッファ内へのオフセットが表示されません。
<b>Addition3</b>	アディクション 3。
<b>Addition4</b>	アディクション 4。VSAM ファイルが読み込まれる場合は、初期化されると、このフィールドは VSAM に設定されます。
<b>Addition5</b>	アディクション 5。
<b>Format</b>	フォーマットバッファ。(最後のピリオドは必須)
<b>Record</b>	レコードバッファ。
<b>Search</b>	サーチバッファ。(最後のピリオドは必須)
<b>Value</b>	バリューストックバッファ。
<b>ISN</b>	ISN バッファ。



## Adabas OP コマンド

Adabas コマンド OP（オープン）を実行すると、ADACALL ではウィンドウが表示され、次のパラメータを指定できます。

- 内部 ISN バッファに格納する最大 ISN
- ホールド状態で許可される最大レコード
- アクティブ状態を維持できる最大 CID（コマンド ID）
- Sx コマンドの実行に許可される最大時間

ウィンドウに関連する情報を入力し、Enter キーを押します。

パラメータおよび有効な値の詳細については、『Adabas コマンドリファレンスマニュアル』ドキュメントを参照してください。

## ADACALL コマンドと PF キー

次の ADACALL ダイレクトコマンドを使用すると、ADACALL パラメータ設定を変更したり、コマンド行にコマンドを入力するか、または対応する PF キーを選択して画面を切り替えたりできます。

ADACALL コマンド以外に、コマンド行から、Natural システムコマンドを発行することもできます。

次の表で、コマンドの下線部分は入力可能な省略形を表しています。

コマンド	PF キー	機能
	PF1	ADACALL のヘルプ機能を呼び出します。さまざまな ADACALL パラメータのうち、いずれかにカーソルを置いて PF1 キーを押すと、そのパラメータのヘルプ情報が表示されます。
	PF2	ADACALL メイン画面に戻ります。 <b>[Mode]</b> は CHAR に設定されます。
<u>BACK</u>	PF5	バッファが完全に表示されている場合に、ページが前のバッファに戻ります。  VIEW コマンドが適用された後でのみ有効です。つまり、コマンドは ADACALL メイン画面からは適用されません。
<u>CB</u>		コントロールブロックバッファが完全に表示されます。16 進モードでのみ有効です。
<u>CHAR</u>	PF6	現在のモードをキャラクタモード（EBCDIC）に変更します。
<u>D</u>		受け取ったレスポンスコードの拡張エラーメッセージテキストが表示されます。0（ゼロ）以外の Adabas レスポンスが返された場合、対応する短いエラーメッセージテキストがメッセージ行に表示されます。このコマンドを発行すると、拡張テキストを参照できます。
<u>EXEC</u>	PF10	指定したパラメータでダイレクトコマンドを実行します。

コマンド	PF キー	機能						
または RUN								
EXIT または STOP または Q または .	PF3 または PF12	終了します。ADACALL メイン画面で押すと、ADACALL が終了します。いずれかのバッファ画面が表示されている場合は、 <b>[Mode]</b> が変更されていない状態で、ADACALL メイン画面が表示されます。						
FB		フォーマットバッファが完全に表示されます。						
EWD	PF4	バッファが完全に表示されている場合に、ページを進めて次のバッファに移動します。  VIEW コマンドが適用された後でのみ有効です。つまり、コマンドは ADACALL メイン画面からは適用されません。						
Hex	PF7	現在のモードを 16 進に変更します。						
IB		ISN バッファが完全に表示されます。						
INIT	PF11	<p>バッファを初期化およびリセットします。ウィンドウが表示され、示されたバッファに対して次のいずれかの値を入力できます。</p> <table border="1" data-bbox="360 1119 1385 1507"> <tbody> <tr> <td>H</td> <td>対応するバッファをバイナリの 0 (H'00') で初期化します。</td> </tr> <tr> <td>H 以外のいずれかの文字または空白</td> <td>対応するバッファを空白 (H'40') で初期化します。</td> </tr> <tr> <td>空白文字</td> <td>対応するバッファを初期化しません。</td> </tr> </tbody> </table> <p>INIT ALL と入力すると、コントロールブロックを除くすべてのバッファが空白で初期化されます。または、コマンド INIT FB RB SB VB IB (すべてのバッファをリストする必要はなし) を指定して、リスト内のすべてのバッファを空白で初期化することができます。</p> <p><b>注意:</b> ISN バッファは、常にバイナリの 0 で初期化されます。</p>	H	対応するバッファをバイナリの 0 (H'00') で初期化します。	H 以外のいずれかの文字または空白	対応するバッファを空白 (H'40') で初期化します。	空白文字	対応するバッファを初期化しません。
H	対応するバッファをバイナリの 0 (H'00') で初期化します。							
H 以外のいずれかの文字または空白	対応するバッファを空白 (H'40') で初期化します。							
空白文字	対応するバッファを初期化しません。							
PRINT	PF9	<p>すべてのバッファのステータスに関するレポートが生成および表示されます。</p> <p>Natural 端末コマンドの %H を使用して、ハードコピーを取得することができます。</p>						
RB		レコードバッファが完全に表示されます。						

コマンド	PF キー	機能												
RUN		EXEC と同じです。												
SB		サーチバッファが完全に表示されます。												
VB		バリューバッファが完全に表示されます。												
VIEW	PF8	<p>すべてのバッファが完全に表示されます。最初に表示されるバッファは、レコードバッファです。FWD コマンドを使用して、他のバッファを確認することができます。</p> <p>16 進モードでレコードバッファに対して VIEW を実行すると、データは 4 ページに表示されます。</p> <p>次のページに進むには、コマンド FWD を入力するか、または PF4 を選択します。前のページに戻るには、コマンド BACK を入力するか、または PF5 を選択します。特定のページを表示するには、[Specify next page number] フィールドに 1~4 のページ番号を入力します。</p> <p>バッファを個別に表示するには、次のコマンドのいずれかを入力します。</p> <table border="1" data-bbox="456 856 1474 1423"> <tbody> <tr> <td>FB</td> <td>フォーマットバッファ</td> </tr> <tr> <td>RB</td> <td>レコードバッファ</td> </tr> <tr> <td>SB</td> <td>サーチバッファ</td> </tr> <tr> <td>VB</td> <td>バリューバッファ</td> </tr> <tr> <td>IB</td> <td>ISN バッファ</td> </tr> <tr> <td>CB</td> <td>コントロールブロック (デフォルト)。16 進モードでのみ有効です。VIEW を実行する前に HEX に変更します。</td> </tr> </tbody> </table>	FB	フォーマットバッファ	RB	レコードバッファ	SB	サーチバッファ	VB	バリューバッファ	IB	ISN バッファ	CB	コントロールブロック (デフォルト)。16 進モードでのみ有効です。VIEW を実行する前に HEX に変更します。
FB	フォーマットバッファ													
RB	レコードバッファ													
SB	サーチバッファ													
VB	バリューバッファ													
IB	ISN バッファ													
CB	コントロールブロック (デフォルト)。16 進モードでのみ有効です。VIEW を実行する前に HEX に変更します。													
VSAM		現在の Natural セッションに VSAM が定義されている場合、このダイレクトコマンドを発行して、VSAM ファイルをアクセスまたは更新できます。このコマンドを発行すると、VSAM ファイル名を求めるウィンドウが表示されます。コマンドが実行されると、コマンドは該当する VSAM ファイルに送られます。												

## ユーザー出口 ADAEXIT

---

ADACALL を使用すると、任意のデータベースにダイレクトコマンドを発行することができます。このため、セキュリティの手段として、ユーザー出口が提供されています。このユーザー出口は ADAEXIT と呼ばれ、ライブラリ SYSADA に含まれています。ADAEXIT は必要に応じて変更できます。Adabas コントロールブロックは、ADAEXIT にパラメータとして渡されます。コントロールブロックの内容を変更するために、ユーザー出口のソースコードを変更できます。単にデータベース ID またはファイル番号を変更するか、またはコマンドコードを XX に設定することによって、データベースコールが実行されるのを防ぐことができます。

# 5 DBLOG ユーティリティ・データベースコールのロギング

---

DBLOG ユーティリティは、Adabas コマンド、DL/I コールと SYNC/ROLB コール、または SQL ステートメントをログに記録するために使用します。ロギングは、アプリケーションのチューニング（データベースにアクセスするコマンドのフローの制御）や、データベースから返されるエラーコードの分析を行う場合に役立ちます。

- DBLOG の実行
- DBLOG メニュー
- DBLOG トレース画面
- DBLOG スナップショット機能
- TEST DBLOG コマンド



## 6 DBLOG の実行

---

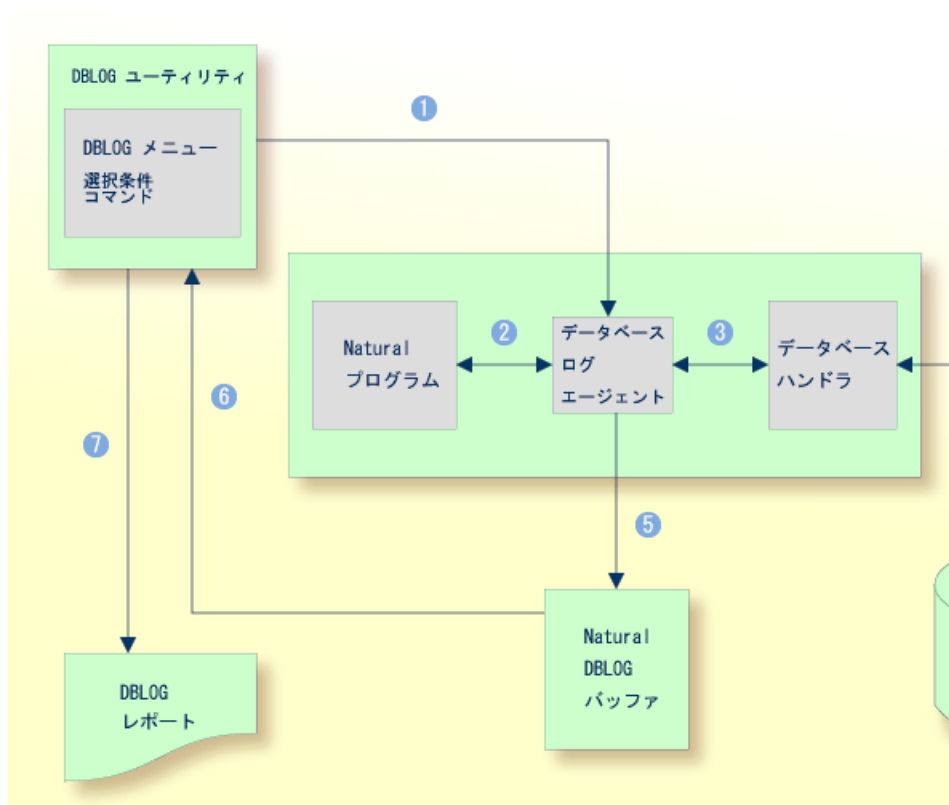
- データベースロギングの基本原則 ..... 18
- データ処理と格納 ..... 19
- DBLOG のアクティブ化と非アクティブ化 ..... 20
- 選択的な DBLOG の使用 ..... 21

DBLOG ユーティリティを使用すると、各 Adabas コマンド、DL/I コールと SYNC/ROLB コール、または SQL ステートメントがデータベースシステムにより処理された後、ログに記録されます。ロギングは、DBLOG をアクティブにして Natural プログラムを実行すると開始されます。

以下のトピックについて説明します。

## データベースロギングの基本原則

次の図は、DBLOG ユーティリティで Natural プログラムからデータベースコールがログに記録される時のプロセスフローを示しています。





## 凡例

- ① データベースコール（Adabas コマンド、DL/I コールと SYNC/ROLB コール、または SQL ステートメント）のロギングをアクティブにするには、対応する [DBLOG Menu] 機能または Natural システムコマンド TEST DBLOG を使用します。  
[DBLOG Menu] では、個々の選択条件を指定できます。例えば、ロギングを特定のデータベース ID やファイル番号に制限できます。
- ② Natural プログラムにより、データベースコールを開始するステートメント（例えば、FIND や READ など）が発行されます。
- ③ データベースログエージェントにより、データベースコールがデータベースハンドラに転送されます。
- ④ データベースハンドラにより、データベースコールが特定のデータベース（Adabas、DL/I、または SQL）に合わせて調整され、データベースコールにより要求されたデータが取得されて、このデータがデータベースログエージェントに返されます。
- ⑤ Natural DBLOG のデータベースログエージェントレコードにより、データベースハンドラにより返されたデータがバッファリングされ、このデータが Natural プログラムに転送されます。
- ⑥ DBLOG ユーティリティのレポート機能により、Natural DBLOG バッファに記録されたデータが読み取られ、① の [DBLOG Menu] で指定されている選択基準に従ってレコードが選択されます。
- ⑦ 選択されたデータレコードから、DBLOG ユーティリティのレポート機能によって、表示可能なレポートが生成され、ワークファイルに書き込まれるか、バッチ処理のために使用されます。

## データ処理と格納

DBLOG ユーティリティにより記録されたデータは、Natural DBLOG バッファに記録されます。バッファの初期サイズと最大サイズは、DSIZE プロファイルパラメータによって決まります（『パラメータリファレンス』ドキュメントを参照）。

DBLOG バッファにデータを保持する十分なスペースがない場合は、Natural によって、DSIZE プロファイルパラメータで指定されている最大サイズに達するまで、適切に（場合によっては何度も）DBLOG バッファサイズが拡大されます。最大値に達した後は、最新のログデータのみが Natural DBLOG バッファに保持されます。

ストレージ不足により、これ以上 DBLOG バッファサイズを拡大できない場合は、DBLOG バッファのスペースが不足していることを示す NAT7545 メッセージが発行されます。このようなストレージ不足の状況が発生する前にログに記録されたデータはすべて Natural DBLOG バッファに保持され、TEST DBLOG コマンドを使用することによって表示できます。

DBLOG は、オンラインモードでもバッチモードでも使用できます。DL/I コールおよび SYNC/ROLB コールは、CICS の下で、IMS/TM の下で、またはバッチモードでログに記録できます。バッチモード処理の詳細については、『オペレーション』ドキュメントで説明されている「バッチモードでの Natural」を参照してください。

記録されたログは、**[DBLOG Trace]** 画面に表示されます。

DBLOG ユーティリティには、データ記録のためのデフォルト設定があります。**[DBLOG Menu]** を使用するとき、ログに記録するコマンド、コール、またはステートメントの選択条件と、表示される情報を指定できます。**[DBLOG Menu]** には、ロギングをアクティブまたは非アクティブにするための機能も備わっています。また、Natural システムコマンド `TEST DBLOG` を使用して DBLOG 実行を制御することもできます。

**[DBLOG Trace]** 画面、**[DBLOG Menu]**、および Natural システムコマンド `TEST DBLOG` のフィールドについては、DBLOG ドキュメントの関連セクションで説明します。

## DBLOG のアクティブ化と非アクティブ化

---

次のセクションで、デフォルトの DBLOG ユーティリティ設定で DBLOG をアクティブまたは非アクティブにする場合に使用されるコマンドについて説明します。詳細については、「**TEST DBLOG コマンド**」を参照してください。

### ▶手順 6.1. Adabas の DBLOG をアクティブまたは非アクティブにするには

- 次の Natural システムコマンド（トグルコマンド）を入力します。

```
TEST DBLOG
```

または:

アクティブにするには、次のように入力します。

```
TEST DBLOG ON
```

非アクティブにするには、次のように入力します。

```
TEST DBLOG OFF
```

または:

**DBLOG** メニューで、アクティブにするにはファンクションコード「B」、非アクティブにするにはファンクションコード「E」を入力します。

### ▶手順 6.2. DL/I の DBLOG をアクティブまたは非アクティブにするには

- 次の Natural システムコマンド（トグルコマンド）を入力します。

```
TEST DBLOG D
```

または:

アクティブにするには、次のように入力します。

```
TEST DBLOG D ON
```

非アクティブにするには、次のように入力します。

```
TEST DBLOG D OFF
```

または:

**DBLOG** メニューで、アクティブにするにはファンクションコード「B」、非アクティブにするにはファンクションコード「E」を入力します。

### ▶手順 6.3. SQL の DBLOG をアクティブまたは非アクティブにするには

- 次の Natural システムコマンド（トグルコマンド）を入力します。

```
TEST DBLOG Q
```

または:

アクティブにするには、次のように入力します。

```
TEST DBLOG Q ON
```

非アクティブにするには、次のように入力します。

```
TEST DBLOG Q OFF
```

または:

**DBLOG** メニューで、アクティブにするにはファンクションコード「B」、非アクティブにするにはファンクションコード「E」を入力します。

## 選択的な DBLOG の使用

次の例は、**[DBLOG Menu]** で指定した選択条件を使用して Adabas コマンド、DL/I コール、または SQL ステートメントをログに記録する手順を示しています。

### ▶手順 6.4. 選択条件を使用して DBLOG を実行するには

- 1 次のいずれかの Natural システムコマンドを入力して、**[DBLOG Menu]** を呼び出します。

- Adabas の場合：

```
TEST DBLOG MENU
```

■ DL/I の場合：

```
TEST DBLOG D MENU
```

■ SQL の場合：

```
TEST DBLOG Q MENU
```

**DBLOG** メニューが表示されます。

- 2 [DBLOG Menu] で、ロギング制限を指定し、ロギングをアクティブにします。これには、入力フィールドに必要事項を入力し、ファンクションコード「B」を入力します。

「DBLOG started now」というメッセージが表示されます。

- 3 Adabas コマンド、DL/I コール、または SQL ステートメントが含まれる Natural プログラムを実行します。
- 4 次のいずれかの Natural システムコマンドを入力して、[DBLOG Trace] 画面を呼び出し、ロギングを非アクティブにします。

■ Adabas の場合：

```
TEST DBLOG
```

■ DL/I の場合：

```
TEST DBLOG D
```

■ SQL の場合：

```
TEST DBLOG Q
```

[DBLOG Trace] 画面が表示されます。

- 5 次のいずれかの Natural システムコマンドを入力して、Natural DBLOG バッファをクリアし、ロギングを非アクティブにします。

■ Adabas の場合：

```
TEST DBLOG OFF
```

■ DL/I の場合：

```
TEST DBLOG D OFF
```

■ SQL の場合：

```
TEST DBLOG Q OFF
```

---

DBLOG が終了し、NEXT プロンプトが表示されます。

詳細について、「[TEST DBLOG コマンド](#)」のセクションも参照してください。



# 7 DBLOG メニュー

---

▪ [DBLOG Menu] の機能 .....	26
▪ ログイン制限の指定 .....	28
▪ Adabas バッファの指定 .....	28

「DBLOG Menu」では、ロギングをアクティブまたは非アクティブにし、ログに記録する Adabas コマンド、DL/I コール、または SQL ステートメントを指定できます。

▶手順 7.1. 「DBLOG Menu」を呼び出すには

- 次のいずれかの Natural システムコマンドを入力します。

- Adabas の場合：

TEST DBLOG MENU

- DL/I の場合：

TEST DBLOG D MENU

- SQL の場合：

TEST DBLOG Q MENU

以下のトピックについて説明します。

## 「DBLOG Menu」の機能

次のセクションで、「DBLOG Menu」で提供される機能について説明します。機能を実行するには、「Code」フィールドに必要な機能に対応するコードを入力するか、必要な機能に対応する PF キーを押します。

コードまたは PF キー	機能	説明
B または PF4	<b>Begin Logging of Adabas Commands</b>	<p>選択条件に一致する Adabas コマンド、DL/I コール、または SQL ステートメントの DBLOG ロギングをアクティブにします。</p> <p>「<i>TEST DBLOG コマンド</i>」の代替コマンドも参照してください。</p> <p>コード <b>B</b> および <b>W</b> のオプションバッファに関する説明も参照してください。</p>
E または PF5	<b>End and Display Log Records</b>	<p>ロギングを非アクティブにし、Natural DBLOG バッファにデータがある場合は現在のログレコードの DBLOG トレース画面を表示します。現在のログデータは Natural DBLOG バッファに保持されます。</p> <p>「<i>TEST DBLOG コマンド</i>」の代替コマンドも参照してください。</p>



コードまたは PF キー	機能	説明
S または PF6	<b>Snapshot of Specific Adabas Commands</b>	<p>Adabas：指定された Adabas コマンドでプログラムを中断し、このコマンドのみの詳細情報を表示します。「<a href="#">Adabas コマンドのスナップショット機能</a>」を参照してください。</p> <p>DL/I および SQL：指定された DL/I コールまたは SQL ステートメントの詳細情報を収集します。「<a href="#">DL/I コールのスナップショット機能</a>」および「<a href="#">SQL ステートメントのスナップショット機能</a>」のスナップショット機能を参照してください。</p>
W または PF7	<b>Write Log Records to Work File</b>	<p>NaturalDBLOG バッファに含まれる記録済みのデータをワークファイルに書き込みます。ファイル構造（列およびログのシーケンス）は、「<a href="#">[DBLOG Trace] 画面</a>」で説明されている「<a href="#">[DBLOG Trace] 画面</a>」の構造に対応しています。</p> <p>この機能を選択すると、「<b>Work File Selection</b>」ウィンドウで次の情報を入力するように要求されます。</p> <ul style="list-style-type: none"> <li>■ ターゲットのワークファイルを指定します。 Natural Work File 1 にデータを出力する場合は、「N」（デフォルト設定の No）を入力します。</li> <li>または、 Entire Connection がインストールされている場合は、「Y」（Yes）を入力し、Natural Work File 7 を使用して PC テキストファイルにデータを出力します。</li> <li>■ Adabas コントロールブロックについて記録されたデータをワークファイルに書き込むかどうかを指定します。 データを含める場合は、「N」（デフォルト設定の No）を入力します。 データを除外する場合は、「Y」（Yes）を入力します。</li> </ul> <p><a href="#">コード B および W のオプションバッファに関する説明</a>も参照してください。</p>
	<b>Optional Buffers for Codes B and W</b>	<p>Adabas コマンドにのみ適用されます。</p> <p>ファンクションコード B または W の使用時に記録される追加の Adabas バッファを選択します。「<a href="#">Adabas バッファの指定</a>」を参照してください。</p>
PF3		「 <b>DBLOG Menu</b> 」を終了します。現在のログレコードは Natural DBLOG バッファに保持されます。
PF12		Natural DBLOG バッファをクリアして「 <b>DBLOG Menu</b> 」を終了し、NEXT プロンプトに戻ります。

## ロギング制限の指定

このセクションでは、ロギングを制限する選択条件を指定するために [DBLOG Menu] で提供されている入力フィールドについて説明します。

フィールド	説明
<b>Skip</b>	ファンクションコード S にのみ適用されます。 ロギングを開始する前にスキップするコマンド、コール、またはステートメントの数です。
<b>Program</b>	ロギングを、指定したプログラムにより発行されたコマンド、コール、またはステートメントに限定します。
<b>DBID</b>	Adabas コマンドにのみ適用されます。 ロギングを、指定したデータベース ID に対して発行されたコマンドに限定します。
<b>FNR</b>	Adabas コマンドにのみ適用されます。 ロギングを、指定したファイル番号に対して発行されたコマンドに限定します。
<b>Line from Line to</b>	ロギングを、指定したソース行番号の範囲内のコマンド、コール、またはステートメントに限定します。
<b>Low Resp High Resp</b>	Adabas コマンドにのみ適用されます。 ロギングを、指定した範囲内のレスポンスコードが返されたコマンドに限定します。
<b>Low Stat High Stat</b>	DL/I コールにのみ適用されます。 ロギングを、指定した範囲内のステータスコードが返されたコールに限定します。
<b>Low SQLC High SQLC</b>	SQL ステートメントにのみ適用されます。 ロギングを、指定した範囲内の SQL リターンコードが返されたステートメントに限定します。

## Adabas バッファの指定

Adabas コマンドにのみ適用されます。

デフォルトでは、Adabas コントロールブロックがログに記録されます。このコントロールブロックは、実行したコマンドに応じて、従来の Adabas コントロールブロック (ACB) か、拡張 Adabas コントロールブロック (ACBX) のいずれかになります。Adabas コントロールブロックの詳細については、Adabas のメインフレームドキュメントの「Adabas コントロールブロックの構造 (ACB および ACBX)」を参照してください。

コントロールブロック以外に、**[DBLOG Menu]** にリストされている 1 つ以上の Adabas バッファをログに記録できます。

<b>FB</b>	フォーマットバッファ
<b>RB</b>	レコードバッファ
<b>SB</b>	サーチバッファ
<b>VB</b>	バリュウバッファ
<b>IB</b>	ISN バッファ

これらのバッファのロギングを有効化し、ログに記録するバイト範囲を指定するには、バッファ名の下の入力フィールドを使用します。

#### ■ Bytes 0-79

目的のバッファの横に文字を入力して、ログに記録するバッファをマークします。**[From]** フィールドと **[To]** フィールドに値を入力しない場合、デフォルトで各バッファに対して最大 80 バイト (0~79) がログに記録されます。

#### ■ From

バッファをログに記録する開始バイトを指定する開始番号 (例えば、「100」) を入力できます。

バッファ全体 (最大 32KB) をログに記録する場合は、「X」または「\*」 (アスタリスク) を入力し、**[To]** フィールドは空白のままにします。

#### ■ To

バッファをログに記録する終了バイトを指定する終了番号 (例えば、「200」) を入力できます。入力できる最大値は、32767 (32 KB) です。

ログに記録するバッファの長さが全体の制限 2097151 KB (2 GB - 1 バイト) を超えた場合、Natural によって適切なメッセージが発行されます (「[データ処理と格納](#)」を参照)。

バッファのログは、**[DBLOG Trace]** 画面に表示できます (「[Adabas バッファの表示](#)」を参照)。



**注意:** デフォルトでは、スナップショット機能 (関連セクションを参照) を実行すると、すべての Adabas バッファがログに記録されます。したがって、この機能を実行する前に、オプションバッファはいずれもマークする必要はありません。



## 8 DBLOG トレース画面

---

- Adabas コマンド用の [DBLOG Trace] 画面 ..... 32
- DL/I コール用の [DBLOG Trace] 画面 ..... 39
- SQL ステートメント用の [DBLOG Trace] 画面 ..... 41

【DBLOG Trace】画面には、Natural DBLOG バッファに保持される Adabas コマンド、DL/I コールと SYNC/ROLB コール、または SQL ステートメントに関して記録されたログデータが表示されます。

以下のトピックについて説明します。

## Adabas コマンド用の 【DBLOG Trace】 画面

---

- Adabas コマンド用の 【DBLOG Trace】 の呼び出し
- 【DBLOG Trace】 画面の列およびコマンド
- Adabas バッファの表示
- マルチフェッチを使用する Adabas コマンドの表示

### Adabas コマンド用の 【DBLOG Trace】 の呼び出し

次の例は、Adabas コマンド用の 【DBLOG Trace】 画面を呼び出す手順を示しています。

1. 次の Natural プログラムを作成します。

```
DEFINE DATA LOCAL
1 EMP-VIEW VIEW OF EMPLOYEES
  2 NAME
END-DEFINE
READ (3) EMP-VIEW BY NAME
  DISPLAY NAME
END-READ
END
```

2. 次の Natural システムコマンドを入力します。

```
TEST DBLOG
```

「DBLOG started now」というメッセージが表示されます。

3. 次の Natural システムコマンドを入力します。

```
RUN
```

ソースエリアの Natural プログラムが実行されます。

4. 次のコマンドを再度入力します。

```
TEST DBLOG
```



列	PF キー	説明
	ダイレクトコマンド	
	PF3	〔DBLOG Trace〕画面を終了するか、バッファウィンドウを閉じます。現在のログレコードは Natural DBLOG バッファに保持されます。
	PF5	ログエントリを画面最上部に移動します。 列 [M] で、列 [No] にリストされた目的のコマンドとシーケンス番号の隣にカーソルを置いて PF5 を選択します。 ログの位置が変更され、選択したシーケンス番号が先頭になります。
	PF6 または --	リストの先頭、またはバッファウィンドウ内のデータの先頭までスクロールします。
	PF7 または -	リストまたはバッファウィンドウ内のデータを1ページ上方にスクロールします。
	PF8 または +	リストまたはバッファウィンドウ内のデータを1ページ下方にスクロールします。
	PF9 または ++	リストの末尾、またはバッファウィンドウ内のデータの末尾までスクロールします。
	PF10	複数のレコードバッファ/フォーマットバッファが含まれるバッファウィンドウでのみ使用できます。  前のレコードバッファ/フォーマットバッファを表示します。
	PF11	複数のレコードバッファ/フォーマットバッファが含まれるバッファウィンドウでのみ使用できます。  次のレコードバッファ/フォーマットバッファを表示します。
	PF12	Natural DBLOG バッファをクリアし、ロギングを非アクティブにします。



## Adabas バッファの表示

デフォルトでは、Adabas コントロールブロックが記録されます。1つ以上の Adabas バッファを記録する場合は、「[Adabas バッファの指定](#)」の説明に従ってロギング機能を実行する前に、**[DBLOG Menu]** で必要なバッファをマークする必要があります。例えば、**[DBLOG Menu]** でフォーマットバッファのロギングのみがマークされている場合、**[Format Buffer]** ウィンドウのみ表示できます。**[Record Buffer]** ウィンドウは表示できません。

### ▶手順 8.1. コントロールブロックまたはバッファの情報を表示するには

- 1 必要なコマンドの横にある入力フィールドに、必要なバッファに対応する行コマンドを入力し、Enter キーを押します。

行コマンド	必要なバッファ
C	コントロールブロック
F	フォーマットバッファ
R	レコードバッファ
S	サーチバッファ
V	バリューストック
I	ISN バッファ
.	ピリオド（「.」）を入力すると、 <b>[DBLOG Trace]</b> 画面が終了します。現在のログレコードは Natural DBLOG バッファに保持されます。

ウィンドウが開き、要求したコントロールブロックまたはバッファのログデータが表示されます。複数の行コマンドを入力した場合は、PF3 キーを押すと、次のコマンドのコントロールブロックまたはバッファを表示できます。

レコードバッファのデータが表示されたウィンドウの例を次に示します。

```

16:50:05          ***** NATURAL TEST UTILITIES *****          2008-08-01
User SAG          - DBLOG Trace -          Library SAG
M  No Cmd  DB   FNR  Rsp      ISN      ISQ  CID  CID(Hex)  OP Pgm      Line
_   1  RC   10
_   2  S1 20000  50          1      2232  ?  ?? 02000101  ADATEST 0200
R   3  L1 20000  50          1      2232  40404040  ADATEST 0200
+-----Page 1 of 1 (logged range:0x-0x4F)-----+ 200
!  _____ Seq No .. 3      Record Buffer1/13 (length:0x7A)      ! 340
! 0000 * C1C4D2C9 D5E2D6D5 40404040 40404040 * ADKINSON          * 0000 ! 350
! 0010 * 40404040 0000000B 00000001 40404040 *          ?  ?          * 0010 ! 350
! 0020 * 40404040 00000000 00000000 00000000 *          *          * 0020 ! 350
! 0030 * 00000000 00000000 00000000 00000000 *          *          * 0030 !
! 0040 * 00000000 F0F0F0F0 F0F0F0F0 40404040 *          00000000 * 0040 !
+-----+

```

```
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit      --      -      +      ++      <      >      Canc
```

次の表に、バッファウィンドウに表示されるフィールドとその説明を示します。

フィールド	説明
Page	現在のページの番号と、バッファに生成された合計ページ数（上の例では、1 of 1）
logged range	実際にログに記録されたバッファ長（16進形式）（上の例では、0x0-0x4F）
Seq No	コマンドのシーケンス番号。上の例では、このコマンドは3番目に実行されています（3）。
buffer-type	buffer-typeは、要求されたバッファのタイプです。
num-current /num-total	また、フォーマットバッファまたはレコードバッファの場合は、レコードバッファまたはフォーマットバッファの数も表示されます。
num-current	現在表示されているレコードバッファ／フォーマットバッファの数を表します。
num-total	<p>ログに記録されたレコードバッファ／フォーマットバッファの合計数を表します。</p> <p>拡張 Adabas コントロールブロック（ACBX）を使用するデータベースコールの場合、複数のフォーマットバッファ／レコードバッファがログに記録されます。上の例は、合計13レコードのうちの最初のレコード（1 / 13）を示しています。</p> <p>ACBXの詳細については、Adabasのメインフレームドキュメントの「Adabas コントロールブロックの構造（ACBおよびACBX）」を参照してください。</p>
length	16進形式の合計レコード長（上の例では、0x7A）
—	[SeqNo] の横の入力フィールドに、次のいずれかの行コマンドを入力できます。
C	コントロールブロックを表示します。
F または R	<p>フォーマットバッファ（F）またはレコードバッファ（R）を表示します。</p> <p>フォーマットバッファとレコードバッファのペアが存在する場合、レコードバッファに「F」を入力すると、一致するレコードバッファが表示され、フォーマットバッファに「R」を入力すると、一致するフォーマットバッファが表示されます。例えば、現在2番目のレコードバッファが表示されている場合に「F」を入</p>

フィールド	説明
	力すると、対応する 2 番目のフォーマットバッファが含まれるウィンドウが表示されます。
I	ISN バッファを表示します。
S	サーチバッファを表示します。
V	バリュースバッファを表示します。
<i>buffer-number</i>	表示するレコードバッファ／フォーマットバッファの数を入力できます。 下記の手順 2 も参照してください。
.	ピリオド（「.」）を入力すると、現在のバッファウィンドウが閉じます。

- 2 複数のレコードバッファ／フォーマットバッファが含まれるレコードバッファ／フォーマットバッファウィンドウでは、次のいずれかの方法を使用して、各レコードバッファ／フォーマットバッファを表示できます。

PF10 キーを押すと、前のレコードバッファ／フォーマットバッファが表示されます。

または:

PF11 キーを押すと、次のレコードバッファ／フォーマットバッファが表示されます。

または:

[ ] 入力フィールドに、表示するレコードバッファ／フォーマットバッファに対応する番号を入力します。

## マルチフェッチを使用する Adabas コマンドの表示

FIND、READ、または HISTOGRAM ステートメントで MULTI-FETCH 節が使用されている場合、一連のレコードを取得する Adabas コマンドのみが実際にデータベースにアクセスします。取得されたレコードは、そのレコードがデータベースループの実行中にフェッチされた、マルチフェッチバッファに移動されます。次のデータベースコールは、次の一連のレコードに対してのみ行われます。詳細については、『プログラミングガイド』の「MULTI-FETCH 節」を参照してください。

[DBLOG Trace] 画面に、データベースコールと非データベースコールの両方がリストされます。データベースコールは、[OP] 列の最初の位置に M でマークされ、マルチフェッチバッファの非データベースコールは小なり記号 (<) でマークされます。次の例を参照してください。

マルチフェッチを使用する **Adabas** コマンドの例

MFETCH という名前の次の Natural プログラムに対して DBLOG を実行します。

```

DEFINE DATA LOCAL
1 EMP-VIEW VIEW OF EMPLOYEES
  2 NAME
END-DEFINE
*
READ (5) MULTI-FETCH OF 3 EMP-VIEW BY NAME = 'ADKINSON'
  DISPLAY *COUNTER NAME
END-READ
END
    
```

次のような **[DBLOG Trace]** 画面が表示されます。

```

10:04:46          ***** NATURAL TEST UTILITIES *****          2008-07-24
User SAG          - DBLOG Trace -          Library SAG
M  No Cmd  DB   FNR  Rsp      ISN      ISQ  CID  CID(Hex)  OP  Pgm      Line
_   1  L3   10   316      2108      -?? 00600101 MA MFETCH  0060
_   2  L3   10   316      2109      -?? 00600101 <A MFETCH  0060
_   3  L3   10   316      2110      -?? 00600101 <A MFETCH  0060
_   4  L3   10   316      2111      -?? 00600101 MA MFETCH  0060
_   5  L3   10   316      2112      -?? 00600101 <A MFETCH  0060
_   6  RC   10   316          -?? 00600101 SI MFETCH  0060
_   7  RC   10          00000000 F  MFETCH  0090

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Print Exit          Posi  --  -  +  ++          Canc
    
```

シーケンス番号 1 および 4 として示された L3 コマンドは、データベースから一連のレコードを取得し（**[OP]** 列に M で示されています）、最初のレコードをプログラムに返します。これ以外のレコードは、マルチフェッチバッファにキャッシュされます。

シーケンス番号 2、3、および 5 として示された L3 コマンドは、マルチフェッチバッファからレコードを取得し（**[OP]** 列に < で示されています）、そのレコードをプログラムに返します。

マルチフェッチデータベースコールのレコードバッファの内容

マルチフェッチを使用するデータベースコールのレコードバッファには、データベースから取得されたすべてのレコードのデータが含まれます。これらは、処理された順序でリストされます。

Adabas では、一連のレコードをロードするとき、最初のバイトから、データベースから返されたレコードまでの範囲のレコードバッファが上書きされます。バッファに残っているスペースは消去されず、前のデータベースコール中にロードされた古いレコードのデータが含まれています。このため、例えば NAME(A20) として定義されたフィールドが読み込まれ、マルチフェッチ係数として 5 が使用された場合、レコードバッファの長さは 100 (20 \* 5) バイトとなります。

データベースから3つのレコードのみが返された場合は、レコードバッファにはこの最初の3レコード（バイト 1~60）のみが格納され、最後の2レコード（バイト 61~100）は変更されません。

## DL/I コール用の [DBLOG Trace] 画面

- DL/I コール用の [DBLOG Trace] の呼び出し
- DBLOG トレース画面の列

### DL/I コール用の [DBLOG Trace] の呼び出し

次の例は、DL/I コール用の [DBLOG Trace] 画面を呼び出す手順を示しています。

1. 次の Natural プログラムを作成します。

```
DEFINE DATA LOCAL
01 COURSE VIEW OF DNDL01-COURSE
  02 COURSEN (A3)
  02 TITLE (A33)
01 OFFERING VIEW OF DNDL01-OFFERING
  02 COURSEN-COURSE (A3)
  02 LOCATION (A31)
END-DEFINE
READ (5) COURSE BY COURSEN
  IF TITLE = 'NATURAL'
    FIND (1) OFFERING WITH COURSEN-COURSE = COURSEN
      MOVE 'DARMSTADT' TO LOCATION
      UPDATE
    END OF TRANSACTION
  END-FIND
END-IF
END-READ
END
```

2. 次の Natural システムコマンドを入力します。

```
TEST DBLOG D
```

「DBLOG started now」というメッセージが表示されます。

3. 次の Natural システムコマンドを入力します。

```
RUN
```

ソースエリアに含まれる Natural プログラムが実行されます。

4. 次のコマンドを再度入力します。

```
TEST DBLOG D
```

ロギングが非アクティブになり、DL/I コール用の [DBLOG Trace] 画面が表示されます。

```

User SAG                - DBLOG Trace -                Library SAG
No  Func PCB  NS  SC  DBD/PSB      First SSA (truncated)  IOA (trunc) Program  Line
-----
 1 PCB                PCNQA42                LOGDL1  0090
 2 GU   1   1   DNDL01  COURSE  *--(COURSEN =>      .      LOGDL1  0090
 3 GN   1   1   DNDL01  COURSE  *--(COURSEN =>      .Z01   LOGDL1  0090
 4 GN   1   1   DNDL01  COURSE  *--(COURSEN =>      .001   LOGDL1  0090
 5 GN   1   1   DNDL01  COURSE  *--(COURSEN =>      .004NATURA LOGDL1  0090
 6 GHNP 1   2   DNDL01  COURSE  *- (COURSEN =004    ?010791DAR LOGDL1  0110
 7 REPL 1                DNDL01                ?010791DAR LOGDL1  0130
 8 SYNC
 9 PCB                PCNQA42                LOGDL1  0110
10 GU   1   1   DNDL01  COURSE  *--(COURSEN = 004    .004NATURA LOGDL1  0110
11 GHNP 1   2   DNDL01  COURSE  *--(COURSEN = 004    ?010791DAR LOGDL1  0110
12 GN   1   1   DNDL01  COURSE  *--(COURSEN =>      +110   LOGDL1  0090
***** End of Log *****

NEXT                                                    LIB=SAG
    
```

## DBLOG トレース画面の列

次のセクションで、[DBLOG Trace] 画面で DL/I コール用に提供されているフィールドの列について説明します。

列	説明
No	シーケンス番号。コマンドは、実行された順序で表示されます。
Func	DL/I 機能。
PCB	PCB 番号。
NS	SSA の数。
SC	DL/I ステータスコード。
DBD/PSB	DB コールの場合、DBD 名。 スケジューリングコールの場合、PSB 名。
First SSA	最初の SSA の最初の 25 バイト。
IOA	最初の I/O エリアの最初の 13 バイト。
Program	Natural プログラム名。
Line	ソースコード行番号。

## SQL ステートメント用の [DBLOG Trace] 画面

- SQL ステートメント用の [DBLOG Trace] の呼び出し
- [DBLOG Trace] 画面の列およびコマンド

### SQL ステートメント用の [DBLOG Trace] の呼び出し

次の例は、SQL ステートメント用の [DBLOG Trace] 画面を呼び出す手順を示しています。

1. 次の Natural プログラムを作成します。

```
DEFINE DATA LOCAL
01 EMP VIEW OF DSN8810-EMP
  02 EMPNO
  02 FIRSTNME
  02 MIDINIT
  02 LASTNAME
  02 EDLEVEL
  02 SALARY
01 EMPPROJECT VIEW OF DSN8810-EMPPROJECT
  02 EMPNO
  02 PROJNO
  02 ACTNO
  02 EMPTIME
END-DEFINE
FIND (1) EMP WITH EMPNO > '000300'
  FIND (1) EMPPROJECT WITH EMPNO = EMPNO(0150)
  MOVE 0.75 TO EMPTIME
  UPDATE
  END-FIND
  ADD 1 TO EDLEVEL
  UPDATE
END-FIND
*
FIND (1) EMP WITH EMPNO > '000300'
  FIND (1) EMPPROJECT WITH EMPNO = EMPNO(0240)
  DISPLAY EMPPROJECT EMP.EDLEVEL
  END-FIND
END-FIND
ROLLBACK
END
```

2. 次の Natural システムコマンドを入力します。

```
TEST DBLOG Q
```

「DBLOG started now」というメッセージが表示されます。

3. 次の Natural システムコマンドを入力します。

RUN

ソースエリアの Natural プログラムが実行されます。

4. 次のコマンドを再度入力します。

TEST DBLOG Q

ロギングが非アクティブになり、下記の例のようなSQLステートメント用の [DBLOG Trace] 画面が表示されます。

```

11:28:58          ***** NATURAL Test Utilities *****                2008-07-28
User SAG          - DBLOG Trace -                                     Library SAG
M No   R SQL Statement (truncated)  CU SN SREF M Typ SQLC/W Program  Line LV
-
  1    SELECT EMPNO,FIRSTNME,MIDINIT 01 01 0150 D DB2          LOGSQL   0150 01
  2      FETCH CURSOR NEX              01 01 0150 D DB2          LOGSQL   0150 01
  3    SELECT EMPNO,PROJNO,ACTNO,EMP 02 02 0160 D DB2          LOGSQL   0160 01
  4      FETCH CURSOR NEX              02 02 0160 D DB2          LOGSQL   0160 01
  5    UPDATE DSN8810.EMPPROJACT SET 02 03 0160 D DB2          LOGSQL   0180 01
  6    CLOSE CURSOR                   02 02 0160 D DB2          LOGSQL   0160 01
  7    UPDATE DSN8810.EMP SET EDLEVE 01 04 0150 D DB2          LOGSQL   0210 01
  8    CLOSE CURSOR                   01 01 0150 D DB2          LOGSQL   0150 01
  9    SELECT EMPNO,FIRSTNME,MIDINIT 05 05 0240 D DB2          LOGSQL   0240 01
 10      FETCH CURSOR NEX              05 05 0240 D DB2          LOGSQL   0240 01
 11    SELECT EMPNO,PROJNO,ACTNO,EMP 06 06 0250 D DB2          LOGSQL   0250 01
 12      FETCH CURSOR NEX              06 06 0250 D DB2          LOGSQL   0250 01
 13    CLOSE CURSOR                   06 06 0250 D DB2          LOGSQL   0250 01
 14    CLOSE CURSOR                   05 05 0240 D DB2          LOGSQL   0240 01
 15    ROLLBACK                       00 00 0000 D DB2          LOGSQL   0290 01
-
-
Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Print Exit Top  Posi Bot  -   +                               Canc
    
```



## [DBLOG Trace] 画面の列およびコマンド

次のセクションで、[DBLOG Trace] 画面でSQLステートメント用に提供されているフィールドの列およびコマンドについて説明します。コマンドを実行するには、PF キーを押すか、またはコマンド行にダイレクトコマンドを入力します。

列	PF キー	説明
	ダイレクトコマンド	
M		行コマンドの入力オプション：
	E	EXPLAIN コマンドを実行します。このコマンドにより、DB2 または SQL/DS オプティマイザでの SQL ステートメントの実行方法の選択肢に関する情報が提供されます。  『 <i>Natural for DB2</i> 』ドキュメントの「DB2 EXPLAIN コマンド」および『 <i>Natural for SQL/DS</i> 』ドキュメントの「EXPLAIN コマンド」も参照してください。
L		LISTSQL コマンドを実行します。このコマンドにより、オブジェクトのソースコード内の Natural ステートメントと変換後の対応する SQL ステートメントがリストされます。SQL ステートメントは、Natural DBLOG バッファから取得されたライブラリ名、プログラム名、および行番号で識別されます。  『 <i>Natural for DB2</i> 』ドキュメントの「LISTSQL コマンド」および『 <i>Natural for SQL/DS</i> 』ドキュメントの「LISTSQL コマンド」も参照してください。
		<b>重要:</b> どちらのコマンドも Natural システムファイルから情報を取得するため、TEST DBLOG Q コマンドでロギング機能が実行された後で対応する Natural プログラムが再カタログ化された場合、望ましくない結果が発生することがあります。このような望ましくない結果は、ロギング後に変更されたステートメントによって引き起こされます。
No		シーケンス番号。ステートメントは、実行された順序で表示されます。
R		DB2 に Natural ファイルサーバーが使用されている場合にのみ適用されます。  対応するステートメントの前のアスタリスクによって、再選択されたことを示します。それ以外の場合は、列は空のままです。  『 <i>Natural for DB2</i> 』ドキュメントの「ファイルサーバーの概念」も参照してください。

列	PF キー	説明
	ダイレクト コマンド	
SQL ステートメント		ログに記録された SQL ステートメントの最初の 29 文字。
CU		カーソル番号。
SN		内部ステートメント番号。
SREF		ステートメント参照番号。
M		モード：ダイナミックは D、スタティックは S。
Typ		データベースタイプ：DB2 または /DS。
SQLC/W		SQLCA の SQLCODE フィールドの SQL リターンコード、または SQLCODE が 0 の場合は SQLCA の SQLWARN0 フィールドの警告。
Pgm		Natural プログラム名。
Line		ソースコード行番号。
LV		プログラムレベル。
	PF2	スクリーンショットのハードコピーを印刷します。
	PF3	「DBLOG Trace」を終了します。現在のログレコードは Natural DBLOG バッファに保持されます。
	PF4	リストの先頭までスクロールします。
	PF5	ログエントリを画面最上部に移動します。列 [M] で、列 [No] にリストされた目的のコマンドとシーケンス番号の隣にカーソルを置いて PF5 キーを選択します。ログの位置が変更され、選択したシーケンス番号が先頭になります。
	PF6	リストの末尾までスクロールします。
	PF7 または -	リストを 1 ページ上方にスクロールします。
	PF8 または +	リストを 1 ページ下方にスクロールします。
	PF12	Natural DBLOG バッファをクリアし、ロギングを非アクティブにします。

## 9 DBLOG スナップショット機能

---

- Adabas コマンドのスナップショット機能 ..... 46
- DL/I コール用のスナップショット機能 ..... 48
- SQL ステートメント用のスナップショット機能 ..... 51

スナップショット機能では、特定の Adabas コマンド、DL/I コール、または SQL ステートメントの詳細情報が提供されます。

以下のトピックについて説明します。

## Adabas コマンドのスナップショット機能

---

このスナップショット機能を使用すると、**[DBLOG Menu]** で指定した選択条件に一致する最初の Adabas コマンドの実行後に、プログラム実行が中断されます。プログラム中断後すぐに、指定した Adabas コマンドに対して生成された **[Snapshot Report]**（次のサンプル画面を参照）が表示されます。

スナップショット機能を実行すると、自動的にすべての Adabas バッファがログに記録されます。したがって、スナップショット機能の開始前に、**[DBLOG Menu]** でオプションバッファはいずれもマークする必要はありません。デフォルトの **[Snapshot Report]** には、コントロールブロック (CB) が表示されます。このコントロールブロックは、従来のコントロールブロック (ACB) か、拡張 Adabas コントロールブロック (ACBX) のいずれかとなります。

以下のトピックについて説明します。

- Adabas コマンド用の **[Snapshot Report]** の呼び出し
- **[Snapshot Report]** でのバッファの表示

### Adabas コマンド用の **[Snapshot Report]** の呼び出し

#### ▶手順 9.1. Adabas コマンド用の **[Snapshot Report]** 画面を呼び出すには

- 1 **[DBLOG Menu]** で、Adabas コマンドと追加の条件（必要な場合）を指定し、ファンクションコード「S」を入力します。

「DBLOG snapshot facility started now」というメッセージが表示されます。

- 2 **[DBLOG Menu]** で指定した Adabas コマンドが含まれる Natural プログラムを実行します。

プログラムの実行が停止され、下記の例のような **[Snapshot Report]** 画面が表示されます。

```

16:36:39          ***** NATURAL TEST UTILITIES *****                2006-12-12
                    - Snapshot Report -

Command Code : L3          Command ID   :   ??? 00200101 File Number   : 013C
Response Code:           0          ISN       :           1300
ISN Low Limit: 00000000    ISN Quantity :           0
FB Length   : 0009         RB Length   : 0014           SB Length   : 0008
VB Length   : 0014         IB Length   : 0000           Com. Option 1:
Com. Option 2: V          Additions 1  : AE]?           Additions 2  : ? ?
Additions 3  :           Additions 4  :
Global FID   : 0000000000000000 Command Time : 00000019 Pgm: SAGTEST Lin: 0020
Control Block
0000 * 30D5D3F3 00200101 013C0000 00000514 * ?NL3 ?????? ?? * 0000
0010 * 00000000 00000000 00090014 00080014 *      ? ? ? ? * 0010
0020 * 000000E5 C1C5BBCA 40404040 00120014 *      VAE]? ? ? * 0020
0030 * 00000000 00000000 00000000 00000000 *              * 0030
0040 * 00000000 00000000 00000019 00000000 *              ?   * 0040
0050 * 00000000 00000000 00000000 00000000 *              * 0050
0060 * 00000000 00000000 00000000 00000000 *              * 0060
0070 * 00000000 00000000 00000000 00000000 *              * 0070

Command ==>> CB
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help           Exit  CB    FB    RB    -    +    SB    VB    IB    Canc

```

### **[Snapshot Report]** でのバッファの表示

デフォルトの **[Snapshot Report]** 画面には、コントロールブロック (CB) が表示されます。レポートに別の Adabas バッファを表示する場合、またはレポートをスクロールする場合は、適切な PF キーを選択するか、このセクションで説明している同等のダイレクトコマンドをコマンド行で入力します。

PF キーが使用可能かどうかは、現在表示されているバッファによって決まります。バッファが画面に収まらない場合や、バッファに複数のフォーマットバッファ/レコードバッファが含まれている場合は、バッファのスクロールに必要な PF キーが画面に表示されます。

PF キー	ダイレクトコマンド	バッファ
PF4	CB	コントロールブロックを表示します。これがデフォルトです。
PF5	FB	フォーマットバッファを表示します。
PF6	RB または --	レコードバッファ (RB) を表示するか、長いバッファの先頭にまでスクロールします (-- )。
PF7	-	長いバッファを 1 ページ上方にスクロールします。
PF8	+	長いバッファを 1 ページ下方にスクロールします。
PF9	SB または ++	サーチバッファ (SB) を表示するか、長いバッファの末尾までスクロールします (++) 。
PF10	VB または <	バリューバッファ (VB) を表示します。複数のフォーマットバッファ/レコードバッファがある場合、前の (<) レコードバッファ/フォーマットバッファを表示します。
PF11	IB または >	ISN バッファ (IB) を表示します。複数のフォーマットバッファ/レコードバッファがある場合、次の (>) レコードバッファ/フォーマットバッファを表示します。

コントロールブロックまたはバッファ内に表示されるフィールドの詳細については、「[Adabas バッファの表示](#)」を参照してください。

## DL/I コール用のスナップショット機能

このスナップショット機能を使用すると、**[DBLOG Menu]** で指定した選択条件に一致する最初の DL/I コールの **[Snapshot Report]** (次のサンプル画面を参照) が生成されます。スナップショットによってプログラムフローが中断されることはありません。スナップショットデータは、ユーザーが適切な DBLOG コマンド (下記参照) を入力した場合にのみ、表示のために Natural DBLOG バッファに保持されます。

以下のトピックについて説明します。

- DL/I コール用の **[Snapshot Report]** の呼び出し



```

Zone : CDEDEC445644CDEDEC447FFF50000000000000000000000000000000000000000000000000000
Digit: 36492500C00D36492500E004D00000000000000000000000000000000000000000000000000000
SSA 2 : Char : OFFERING*F-
Zone : DCCDCDC5C64000000000000000000000000000000000000000000000000000000000000000000000000
Digit: 66659957C6000000000000000000000000000000000000000000000000000000000000000000000000
    
```

### DL/I コールの [Snapshot Report] 情報

DL/I コールの [Snapshot Report] 画面に、次の情報が表示されます。

- PSB（プログラム指定ブロック）の名前。
- PCB（プログラム通信ブロック）の番号。
- PCB マスク。次の要素で構成されます。
  - データベース名を含む DBD（データベース記述）
  - セグメントレベル番号
  - ステータスコード
  - 処理オプション
  - セグメント名
  - KFBA（キーフィールドバックエリア）の長さ
  - SENSG（センシティブセグメント）の数
  - KFBA：
    - キーフィールドバックエリア
  - SSA（セグメント検索指数）の数
- すべての SSA。
- I/O エリア。

キーフィールドバックエリア、すべての SSA（最大 15 の SSA が可能）、および I/O エリアの最初の 120 バイトが、10 進数形式と 16 進形式の両方で表示されます。

PCB の DBD 名は、Natural FDIC システムファイルから対応する NDB（Natural で DBD に相当するもの）を読み取るために使用されます。この NDB には、PCB で名前が指定されているセグメントが配置され、その最小／最大の長さとセグメントレベル番号が表示されます。セグメントレベル番号は、PCB での番号と一致している必要があります。この方法で、NDB と DL/I DBD の矛盾を検出することが可能です。

PSB 名は、Natural FDIC システムファイルから対応する NSB（Natural で PSB に相当するもの）を読み取るために使用されます。この NSB から、センシティブセグメントの数が表示されます。この数は、PCB での数と一致している必要があります。この方法で、Natural NSB と DL/I PSB の矛盾を検出することが可能です。



スナップショット機能を使用すると、DL/IDBD/PSB と Natural NDB/NSB で [Level Number] フィールドおよび [Number of SENSEGs] フィールドに同じ値が含まれているかどうかをチェックされます。ただし、同じ値であれば DL/IDBD/PSB と Natural NDB/NSB が完全に整合しているとは限りません。

上記の例では、DL/IDBD/PSBGEN プロシージャによって PSB が変更された後に Natural NATPSB プロシージャが実行されなかったため、[Number of SENSEGs] フィールドの値は異なっています。

## SQL ステートメント用のスナップショット機能

スナップショット機能を使用すると、[DBLOG Menu] で指定した選択条件に一致する最初の SQL ステートメントの [Snapshot Report] (次のサンプル画面を参照) が生成されます。スナップショットによってプログラムフローが中断されることはありません。

DBLOG トレース機能で表示されるステートメントとは異なり、スナップショットには完全なステートメントが表示されます (13 行に制限)。

スナップショットデータは、ユーザーが適切な DBLOG コマンド (下記参照) を入力した場合にのみ、表示のために Natural DBLOG バッファに保持されます。

以下のトピックについて説明します。

- SQL ステートメント用の [Snapshot Report] の呼び出し
- SQL ステートメントの [Snapshot Report] 情報

### SQL ステートメント用の [Snapshot Report] の呼び出し

#### ▶手順 9.3. SQL ステートメント用の [Snapshot Report] 画面を呼び出すには

- 1 [DBLOG Menu] で、SQL ステートメントと追加の条件 (必要な場合) を指定し、ファンクションコード「S」を入力します。

「DBLOG snapshot facility started now」というメッセージが表示されます。

- 2 [DBLOG Menu] で指定した SQL ステートメントが含まれる Natural プログラムを実行します。ログデータは Natural DBLOG バッファに書き込まれます。
- 3 スナップショットデータを表示するには、次のコマンドを入力します。

```
TEST DBLOG Q
```

または:

DBLOG メニューで、ファンクションコード「E」を入力します。

下記の例のような SQL ステートメントの [Snapshot Report] 画面が表示されます。

```

10:59:28          ***** NATURAL Test Utilities *****                2006-12-12
User SAG          - Snapshot Report -                                Library SAG

CU SN M Typ R SQLC/W      Library  Program  Store Clock Value   Line LV CID(Hex)
01 01 D DB2              SAG      SAGTEST  2002/04/03 14:23:06 0150 01 01500101

SQL Statement
SELECT EMPNO,FIRSTNME,MIDINIT,LASTNAME,EDLEVEL,SALARY FROM DSN8510.EMP WHERE EM
PNO > '000300' FOR UPDATE OF EDLEVEL

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Print Exit                                          Canc
    
```

### SQL ステートメントの [Snapshot Report] 情報

SQL ステートメントの [Snapshot Report] 画面に、次の情報が表示されます。

列	説明
CU	カーソル番号。
SN	内部ステートメント番号。
M	モード：ダイナミックは D、スタティックは S。
Typ	データベースタイプ：DB2 または SQL/DS。
R	DB2 に Natural ファイルサーバーが使用されている場合にのみ適用されます。  対応するステートメントの前のアスタリスクによって、再選択されたことを示します。それ以外の場合は、列は空のままです。  『Natural for DB2』ドキュメントの「ファイルサーバーの概念」も参照してください。
SQLC/W	SQLCA の SQLCODE フィールドの SQL リターンコード、または SQLCODE が 0 の場合は SQLCA の SQLWARN0 フィールドの警告。
Library	ログに記録されたステートメントが含まれる Natural プログラムがカタログ化されたライブラリ。
Program	ログに記録されたステートメントが含まれる Natural プログラムの名前。
Store Clock Value	ログに記録されたステートメントが含まれる Natural プログラムのタイムスタンプ。
Line	ログに記録されたステートメントのソースコード行番号。
LV	ログに記録されたステートメントが含まれる Natural プログラムのコールレベル。
CID (Hex)	ログに記録されたステートメントの 16 進形式のコマンド ID。

# 10 TEST DBLOG コマンド

---

- 構文図 ..... 54
- キーワードの説明 ..... 55

Natural システムコマンド TEST DBLOG は、DBLOG の実行、および Natural DBLOG バッファに現在保存されているログレコードの表示や削除を行う場合に使用されます。TEST DBLOG には、選択条件を指定するためのパラメータはありません。選択条件は、[\[DBLOG Menu\]](#) でのみ指定できます。

次の構文図および表で、コマンド TEST DBLOG に適用されるキーワードについて説明します。DBLOG のアクティブ化と非アクティブ化の両方（トグル効果）を行うために使用できるキーワードがあります。アクティブ化と非アクティブ化のいずれが行われるかは、「[キーワードの説明](#)」で説明しているように、Natural DBLOG バッファに保存されているデータが存在するかどうかによって決まります。

構文図に使用されている記号については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

以下のトピックについて説明します。

### 構文図

---

#### DBLOG のアクティブ化

```
TEST DBLOG [ A ] [ ON ]
             [ D ] [ CONT ]
             [ Q ] [ MENU ]
                [ ? ]
                [ * ]
```

#### DBLOG の非アクティブ化

```
TEST DBLOG [ A ] [ OFF ]
             [ D ] [ SHOW ]
             [ Q ] [ MENU ]
                [ ? ]
                [ * ]
```

## キーワードの説明

キーワード	機能
A	<p>デフォルト値です。</p> <p>トグル機能：</p> <p>Natural DBLOG バッファにデータが存在しない場合、Adabas コマンドのロギングをアクティブにします。</p> <p>Natural DBLOG バッファにデータが存在する場合、Adabas コマンドのロギングを非アクティブにし、現在のログレコードの <b>[DBLOG Trace]</b> 画面を表示します。</p>
D	<p>トグル機能：</p> <p>Natural DBLOG バッファにデータが存在しない場合、DL/I コールのロギングをアクティブにします。</p> <p>Natural DBLOG バッファにデータが存在する場合、DL/I コールのロギングを非アクティブにし、現在のログレコードの <b>[DBLOG Trace]</b> 画面を表示します。</p>
Q	<p>トグル機能：</p> <p>Natural DBLOG バッファにデータが存在しない場合、SQL ステートメントのロギングをアクティブにします。</p> <p>Natural DBLOG バッファにデータが存在する場合、SQL ステートメントのロギングを非アクティブにし、現在のログレコードの <b>[DBLOG Trace]</b> 画面を表示します。</p>
CONT	ロギングをアクティブ化または再アクティブ化（再開）します。再開すると、DBLOG 実行の停止後も、DBLOG は引き続き次のプログラムでロギングを実行し、前の記録から存在するデータに新しいログを追加します。
MENU	ロギングのアクティブ化または非アクティブ化や、ログに記録するコマンド、コール、またはステートメントの指定を行うためのオプションが提供されている <b>[DBLOG Menu]</b> を呼び出します。関連セクションを参照してください。
?	
*	
SHOW	ロギングを非アクティブにし、Natural DBLOG バッファにデータがある場合は現在のログレコードの <b>DBLOG</b> トレース画面を表示します。ログレコードデータは削除されず、Natural DBLOG バッファに保持されます。
ON	Natural DBLOG バッファをクリアし、ロギングをアクティブにします。
OFF	Natural DBLOG バッファをクリアし、ロギングを非アクティブにします。



# 11 INPL ユーティリティ

---

■ INPL ユーティリティの概要 .....	58
■ ライブラリのみロード .....	63
■ DDM のみロード .....	64
■ エラーメッセージのみロード .....	64
■ すべてのオブジェクトをロード .....	65
■ 製品インストールの置き換え .....	65
■ INPL ファイルのスキャン .....	66
■ Natural Security リカバー .....	67

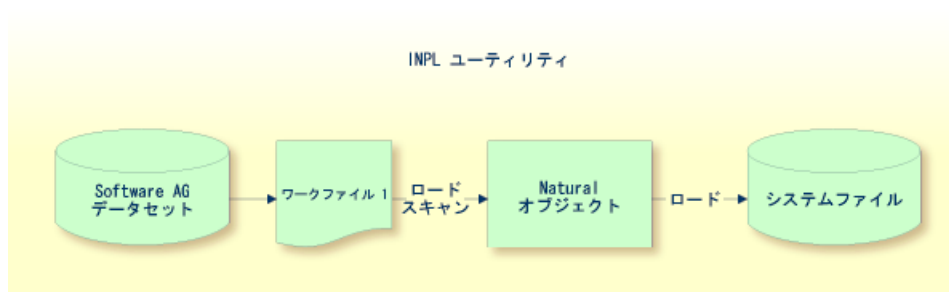
INPL ユーティリティ (Initial Natural Program Load) は、Software AG が提供するデータセットから Natural オブジェクトをロードまたはスキャンするために使用します。

## INPL ユーティリティの概要

---

INPL ユーティリティは、Software AG が提供する Natural オブジェクトを処理します。

次の図は、INPL 機能を簡単な図で示したものです。



Natural オブジェクトは、ワークファイル 1 に割り当てられたインストールまたは更新データセットとして提供されます。INPL ユーティリティは、ワークファイル 1 の Natural オブジェクトを Natural システムファイルにロードします。

Natural オブジェクトには、Natural システムファイル FNAT および FUSER、または DDM（データ定義モジュール）が保存されているシステムファイル FDIC 内のライブラリに含まれている、カタログ化オブジェクトおよびソースオブジェクトがあります。

INPL ユーティリティは、Natural オブジェクトのロード機能だけでなく、ワークファイル 1 に割り当てられたデータセットをチェックするスキャン機能、Natural Security 環境を初期化する **Natural Security** リカバリー機能も提供します。

INPL の実行中にエラーが発生すると、INPL は中断され、コンディションコード 40 で異常終了します。

このセクションでは、次のトピックについて説明します。

- 制限
- 特別な場合
- INPL の呼び出し
- 使用可能なオプション
- INPL レポート



## ■ チェックコマンド

### 制限

処理できるのは、「SAG システム INPL データセット」としてマークされたデータセットだけです。

### 特別な場合

次のいずれの場合も、Natural ダイナミックプロファイルパラメータ STACK を使用して INPL コマンドを指定する必要があります。

- INPL を空の FNAT システムファイルで実行する場合
- 新規システムメンテナンス更新 INPL を実行する場合
- 既存の製品インストールを置き換える場合
- INPL を Natural Security 環境で実行する場合

### INPL の呼び出し

#### ▶手順 11.1. INPL ユーティリティを呼び出すには

- 1 次の Natural システムコマンドを入力します。

```
INPL
```

次のような INPL メニューが表示されます。

```

18:32:19          ***** NATURAL INPL UTILITY *****          2005-04-21
User: SAG                                               Library: SYSTEM

                Code  Function
                L    Load Libraries Only
                D    Load DDMs Only
                E    Load Error Messages Only
                B    Load All Objects
                P    Replace Product Installation
                S    Scan INPL File
                R    Natural Security Recover
                ?    Help
                .    Exit

Code ..... B
Replace .... Y (Y/N/P/O)  Load Except . N (Y/N)
DDM Name ....

```

```

Library .....
Object Name .      Date .....      (YYYY-MM-DD)
Check Date .. N (Y/N)  Number ..... 0

```

2 対応するファンクションコードを INPL メニューの **[Code]** フィールドに入力することにより、次のいずれかの機能を選択できます。

- ライブラリのみロード
- **DDM** のみロード
- エラーメッセージのみロード
- すべてのオブジェクトをロード
- 製品インストールの置き換え
- **INPL** ファイルのスキャン
- **Natural Security** リカバー

これらの機能の詳細については、該当するセクションを参照してください。

3 「**使用可能なオプション**」の説明に従って、残りの入力フィールドを変更または完了してください。

4 Enter キーを押してエントリを確定します。

## 使用可能なオプション

次のセクションでは、INPL 機能のオブジェクト選択条件として1つまたは複数のパラメータを指定できる INPL メニュー上の入力フィールドについて説明します。この選択条件は、**[Code]** フィールドで指定します。使用するパラメータは、関連するドキュメントのセクションで説明するように、機能によって異なります。

フィールド	説明
<b>Replace</b>	<p>処理される Natural オブジェクトを、すでにシステムファイルに存在するものと置き換えるかどうかを指定します。</p> <p>可能な設定：</p> <p>Y すべての既存オブジェクトが置き換えられます。これはデフォルト設定です。</p> <p>N 既存のオブジェクトは置き換えられません。</p> <p>P すべての既存オブジェクトが置き換えられます。また、既存の Natural 製品または Natural アドオン製品のインストールも、INPL 処理時に実行される <b>チェックコマンド</b>（次を参照）に基づいて置き換えられます。</p> <p>0 指定したオブジェクトのすべてのオーナー情報をリセットします。 <b>Natural Security</b> リカバー機能にのみ適用されます。</p>

フィールド	説明
	<p>処理するオブジェクトより古いオブジェクトだけを置き換える場合は、「<b>Check Date</b>」も参照してください。</p> <p><b>Natural Security</b> リカバー機能を使用する場合、このフィールドに「0」オプションを入力すると、指定したオブジェクトのオーナー情報をリセットできます。</p>
<b>DDM Name</b>	<p>DDM の名前または名前の範囲。</p> <p>アスタリスク (*) で終わる値を入力すると、指定した値で始まる名前を持つ DDM が処理されます。アスタリスク (*) のみを入力した場合、またはこのフィールドが空の場合は、すべての DDM が処理されます。</p>
<b>Library</b>	<p>ライブラリの名前または名前の範囲。</p> <p>アスタリスク (*) で終わる値を入力すると、指定した値で始まる名前を持つ各ライブラリが処理されます。オブジェクト名を指定する場合は、ライブラリ名は必須です。</p>
<b>Object Name</b>	<p>Natural オブジェクト (DDM 以外) の名前または名前の範囲。</p> <p>値がアスタリスク (*) で終わる場合は、指定した値で始まる名前を持つ各オブジェクトが処理されます。</p> <p>このフィールドが空の場合は、<b>[Library]</b> フィールドに指定したライブラリに含まれるすべてのオブジェクトが処理されます。</p>
<b>Check Date</b>	<p>既存の Natural オブジェクトをタイムスタンプに基づいて置き換えるかどうかを指定します。</p> <p><b>[Replace]</b> を N に指定した場合、このパラメータは無効です。</p> <p>可能な設定：</p> <p>Y 同じ名前のオブジェクトより古いオブジェクトのみが置き換えられます。ロードされる前に保存またはカタログ化されたオブジェクトが古いオブジェクトです。</p> <p>N すべてのオブジェクトが置き換えられます。これはデフォルト設定です。</p>
<b>Load Except</b>	<p>Natural オブジェクトを処理から除外するかどうかを指定します。</p> <p>このパラメータはエラーメッセージには適用されません。</p> <p>可能な設定：</p> <p>Y <b>[DDM Name]</b>、<b>[Library]</b>、<b>[Object Name]</b> の各フィールドで指定したオブジェクトを除くすべてのオブジェクトが処理されます。</p> <p>N 除外なしの場合は、すべてのオブジェクトが処理されます。これはデフォルト設定です。</p> <p>ロード除外の例：</p> <p>ライブラリ ABC を除くすべてのライブラリがロードされます。</p> <p><b>Code = L</b> <b>Library = ABC</b></p> <p>接頭辞が XY 以外のすべての DDM がロードされます。</p>

フィールド	説明
	<b>Code = D</b> <b>DDM Name = XY*</b> 接頭辞が AB 以外のライブラリに含まれるすべてのオブジェクトと、接頭辞が CD 以外のすべての DDM がロードされます。 <b>Code = B</b> <b>Library = AB*</b> <b>DDM Name = CD*</b>
<b>Date</b>	このフィールドに入力した日付以降に保存されたか、またはカタログされた Natural オブジェクトだけを処理します。 日付は、YYYY-MM-DD (YYYY=年、MM=月、DD=日) の形式で入力する必要があります。
<b>Number</b>	Natural オブジェクトの処理を指定の数に制限します。INPL メニューで指定した選択条件に基づいてロードまたはスキャンされるすべてのオブジェクトがカウントされます。 処理されたオブジェクト数が <b>[Number]</b> フィールドに入力した値に達すると、処理は終了し、対応するメッセージが表示されます。

## INPL レポート

選択した INPL 機能が完了すると、対応する INPL レポートが画面に表示されます。

## チェックコマンド

INPL ユーティリティでは、INPL を実行すると自動的に実行される内部チェックコマンドが処理されます。チェックコマンドは、ロードプロセスまたはスキャンプロセスを制御して特定のイベントに対処するために使用します。INPL の実行時に実行されるチェックコマンドは、INPL レポートに書き込まれます。

チェックコマンドがイベントに対処するために使用するパラメータには、例えば、STOP、LOAD、CONTINUE、SKIP、USERLOAD、USERCONTINUE などがあります。INPL ファイルのスキャン機能を使用すると、すべてのパラメータを表示できます。例えば、これらのパラメータは、インストールする製品のバージョンがすでにインストールされている製品のバージョンより高いかどうかを確認するために使用します。

チェックコマンドは変更できませんが、製品インストールの置き換え機能（ファンクションコード P）を選択するか、または **[Replace]** オプションを P に設定することにより、パラメータ USERLOAD および USERCONTINUE の効果に影響を与えることができます。

USERLOAD または USERCONTINUE は、ファンクションコード P が選択されているか、または **[Replace]** オプションが P に設定されている場合にのみロードが実行または続行されることを示しています。

## USERLOAD および USERCONTINUE の例

次のチェックコマンド例では、INPL を実行する前に、現在インストールされている Natural のバージョンがチェックされます。

```
CHECK VERSION NAT 421 LT USERLOAD EQ LOAD GT USERLOAD
```

インストールされている Natural (NAT) のバージョンが 421 より低い (LT) か高い (GT) 場合は、ファンクションコード P が指定されているか **[Replace]** オプションが P に設定されているときにのみ、INPL 機能が実行されます。それ以外の場合、INPL ユーティリティは、処理を停止するかバッチでコンディションコード 40 で終了し、対応するメッセージが INPL レポートに書き込まれます。

バージョンが 421 の場合、ファンクションコード P が指定されているかどうか、または **[Replace]** オプションが P に設定されているかどうかに関係なく、INPL 機能は常に実行されます。

## ライブラリのみロード

INPL ユーティリティのこの機能は、Natural のカタログ化オブジェクトやソースオブジェクトを Natural システムファイル FNAT または FUSER の指定されたライブラリにロードするために使用します。

### ▶手順 11.2. ライブラリをロードするには

- 1 INPL メニューでファンクションコード「L」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Replace** (Y/N)
- **Load Except** (Y/N)
- **Library**
- **Object Name**
- **Date** (YYYY-MM-DD)
- **Check Date** (Y/N)
- **Number**

これらのパラメータの詳細については、「INPL ユーティリティの概要」セクションの「[使用可能なオプション](#)」を参照してください。

- 2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「INPL ユーティリティの概要」を参照）が出力されます。

## DDM のみロード

---

INPL ユーティリティのこの機能は、DDM をシステムファイル FDIC にロードするために使用します。

### ▶手順 11.3. DDM をロードするには

- 1 INPL メニューでファンクションコード「D」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Replace** (Y/N)
- **Load Except** (Y/N)
- **DDM Name**
- **Number**

これらのパラメータの詳細については、「INPL ユーティリティの概要」セクションの「[使用可能なオプション](#)」を参照してください。

- 2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「INPL ユーティリティの概要」を参照）が出力されます。

## エラーメッセージのみロード

---

INPL ユーティリティのこの機能は、ユーザー定義のエラーメッセージまたはシステムエラーメッセージをそれぞれ Natural システムファイル FUSER または FNAT にロードするために使用します。

### ▶手順 11.4. エラーメッセージをロードするには

- 1 INPL メニューでファンクションコード「E」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Replace** (Y/N)
- **Library**

これらのパラメータの詳細については、「INPL ユーティリティの概要」セクションの「[使用可能なオプション](#)」を参照してください。

- 2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「*INPL ユーティリティの概要*」を参照）が出力されます。

## すべてのオブジェクトをロード

INPL ユーティリティのこの機能は、エラーメッセージと DDM を含むすべての Natural オブジェクトをワークファイル 1 で指定されたライブラリにロードするために使用します。DDM は、システムファイル FDIC にロードされます。

### ▶手順 11.5. すべてのオブジェクトをロードするには

- 1 INPL メニューでファンクションコード「B」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Replace** (Y/N)
- **Load Except** (Y/N)
- **DDM Name**
- **Library**
- **Object Name**
- **Date** (YYYY-MM-DD)
- **Check Date** (Y/N)
- **Number**

これらのパラメータの詳細については、「*INPL ユーティリティの概要*」セクションの「[使用可能なオプション](#)」を参照してください。

- 2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「*INPL ユーティリティの概要*」を参照）が出力されます。

## 製品インストールの置き換え

すべてのオブジェクトをロード機能以外に、この機能では、現在のシステム環境にインストールされている既存のすべての Natural 製品または Natural アドオン製品が置き換えられます。既存の製品インストールの置き換えは、「*INPL ユーティリティの概要*」で説明している対応する **チェックコマンド**の実行に依存しています。

**▶手順 11.6. すべてのオブジェクトをロードして既存の製品インストールを置き換えるには**

1 INPL メニューで、ファンクションコード「P」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Replace** (Y/N)
- **Load Except** (Y/N)
- **DDM Name**
- **Library**
- **Object Name**
- **Date** (YYYY-MM-DD)
- **Check Date** (Y/N)
- **Number**

これらのパラメータの詳細については、「INPL ユーティリティの概要」セクションの「[使用可能なオプション](#)」を参照してください。

2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「INPL ユーティリティの概要」を参照）が出力されます。

## INPL ファイルのスキャン

---

INPL ユーティリティのこの機能は、ワークファイル1に割り当てられたデータセットの内容をスキャンするために使用します。

**▶手順 11.7. INPL ファイルをスキャンするには**

1 INPL メニューでファンクションコード「S」を入力します。この機能の実行時に有効になるパラメータを指定できます。

- **Load Except** (Y/N)
- **DDM Name**
- **Library**
- **Object Name**
- **Date** (YYYY-MM-DD)
- **Number**



これらのパラメータの詳細については、「INPL ユーティリティの概要」セクションの「[使用可能なオプション](#)」を参照してください。

2 エントリを確定します。

機能が完了すると、対応する **INPL レポート**（「INPL ユーティリティの概要」を参照）が出力されます。


## Natural Security リカバリー

INPL ユーティリティのこの機能は、Natural Security 環境を初期化するために使用します。

次のオプションを使用できます。

- **環境のリセット**
- **オーナーの削除**

### 環境のリセット

 **注意:** この機能を実行すると、ユーザープロファイル DBA、ライブラリプロファイル SYSSEC、初期インストール後のこの2つのオブジェクト間のリンクがリセットされます。ライブラリ SYSSEC へのそれ以外のリンクはすべてキャンセルされます。他の Natural Security プロファイルとリンクは変更されません。詳細については、Software AG 技術サポートに連絡してください。

#### ▶手順 11.8. 環境をリセットするには

- INPL メニューでファンクションコード「R」を入力します。

### オーナーの削除

#### ▶手順 11.9. オーナーを削除するには

- 指定したオブジェクトのオーナー情報をリセットするには、INPL メニューでファンクションコード「R」を入力し、**[Replace]** フィールドに「0」を入力します。



# 12 NATPAGE ユーティリティ・画面キャプチャ

---

NATPAGE ユーティリティは、Natural セッション中に画面に出力されるデータ（マップおよびレポート）を取得するときに使用します。ここで使用する画面という用語は、ページバッファの内容、つまり Natural によって出力される論理ページを表しています。

画面キャプチャは、『オペレーション』ドキュメントの該当するセクションで説明するように、Natural スクラッチパッドファイルに格納されます。

取得できる最大画面数は、セッションパラメータ PD によって決まります。『パラメータリファレンス』ドキュメントの説明を参照してください。

NATPAGE ユーティリティは、次の Natural 端末コマンドで構成されています。

コマンド	機能
%P	NATPAGE を有効にし、現在の画面および後続のすべての画面の内容を取得します。以前に取得した画面は削除されます。
%I	NATPAGE を有効にし、現在の画面の内容を取得します。
%O	NATPAGE を無効にします。
%S	NATPAGE を再開します。
%E	NATPAGE を使用して取得した画面を表示します。

これらの端末コマンドの詳細については、『端末コマンド』ドキュメントを参照してください。



# 13 NATRJE ユーティリティ-Natural リモートジョブエ ントリ

---

▪ NATRJE の一般的な情報 .....	73
▪ Natural プログラムからの NATRJE の呼び出し .....	73
▪ NATRJE リターンコード .....	78
▪ UTM/TIAM に適用可能な NATRJE 機能 .....	80

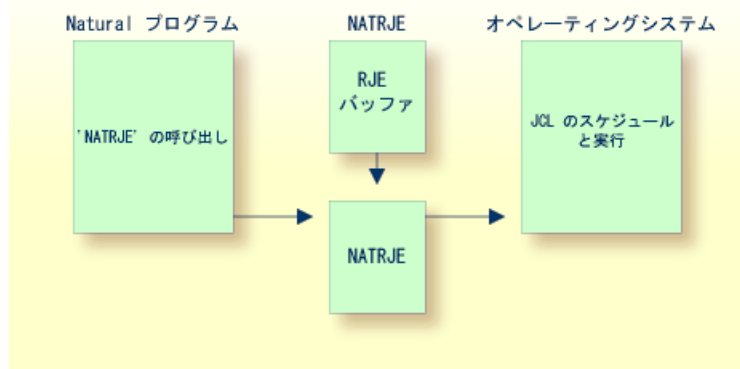
NATRJE ユーティリティ (Natural リモートジョブエントリ) を使用すると、スケジューリングおよび実行のために JCL カードを Natural プログラムからオペレーティングシステムに送信できます。例えば、NATRJE を使用して Natural バッチジョブを開始できます。

以下このドキュメントでは、*openUTM* を UTM と記載します。

関連トピック：

- *NREXPG - NATRJE のユーザー出口* - 『オペレーション』ドキュメント

## NATRJE の全般的な情報



1. Natural プログラムでは、オペレーティングシステムで実行される JCL カードを送信するために、NATRJE ユーティリティが呼び出されます。
2. NATRJE では、ジョブが完了したことを示す通知を Natural プログラムから受け取るまで、JCL カードが RJE バッファに収集されます。JCL カードは、送信されるまで RJE バッファに保持されます。RJE バッファの初期サイズは、RJESIZE プロファイルパラメータによって決まります (Natural の『パラメータリファレンス』ドキュメントを参照)。特定のジョブが RJE バッファに収容できない場合、バッファサイズは自動的に拡張されます。ジョブの最大サイズは、スレッドまたはリージョンのサイズによって決まります。
3. NATRJE では、オペレーティングシステムでスケジューリングおよび実行するために、JCL カードがオペレーティングシステム内部ジョブキューに転送されます。

### BS2000/OSD に関する注：

BS2000/OSD 環境では、ジョブの生成が完了すると、NATRJE によって BS2000/OSD データセットが生成され、JCL カードが転送されます。このデータセットは SAM ファイルであり、Enter キーを押すと BS2000/OSD オペレーティングシステムに送信されます。

## Natural プログラムからの NATRJE の呼び出し

以下に参考情報を示します。

- NATRJE の呼び出し

## ■ プログラム例

### NATRJE の呼び出し

#### ▶ 手順 13.1. NATRJE ユーティリティを呼び出すには

- Natural プログラムで CALL ステートメントを使用します。

CALL ステートメントの構文は次のとおりです。

```
CALL 'NATRJE' parm1 parm2 parm3 parm4
```

CALL ステートメントで指定するパラメータは次のとおりです。

パラメータ	説明
parm1	テーブルの最初の JCL カード。送信する 80 文字の JCL カードが 1 枚以上含まれています。
parm2	4 バイトのバイナリフィールド。送信する 80 文字の JCL カードの枚数が含まれています。
parm3	1 バイトの英数字フィールド。すべての JCL カードが送信されたかどうかを示すために使用します。
"	現在のジョブに対する最後の呼び出しではありません。次の CALL ステートメントに後続の JCL カードが続きます。JCL カードは RJE バッファに収集されます。
B	BS2000/OSD と z/OS のみ：現在のジョブに対する最後の呼び出し。  BS2000/OSD：ジョブが生成されデータセットに書き込まれますが、自動的に開始されません。  z/OS（バッチと TSO、IMS/TM と CICS）：ジョブは内部リーダーデータセットに書き込まれますが、送信されません。ファンクション L をこの後に呼び出すと、内部リーダーがクローズされてジョブがすべて送信されます。また、次の場合も内部リーダーがクローズされてジョブが送信されます。  画面 I/O（IMS/TM）、または セッション終了時（z/OS バッチ、TSO と IMS/TM）
C	現在のジョブをフラッシュします。ジョブはシステムに送信されません（BS2000/OSD の場合、データセットは作成されません）。
L	現在のジョブに対する最後の呼び出し。ジョブはシステムに送信されます。
	BS2000/OSD 環境：「 <a href="#">parm3 パラメータのその他の値</a> 」を参照してください。
parm4	NATRJE からレスポンスコードが返される 2 バイトのバイナリフィールド。



## プログラム例

各オペレーティングシステムで使用する Natural プログラムの例を次に示します。

- **z/OS**
- **z/VSE**
- **BS2000/OSD (例 1 および例 2)**

プログラム例 - **z/OS** :

次の例は、NATRJE を 1 回呼び出して 3 枚の JCL カードストリームを送信する Natural プログラムです。

```
DEFINE DATA LOCAL
  01 COUNT (B4)
  01 FLAG (A1)
  01 RETHEX (B2)
  01 CARDS (A240)
  01 REDEFINE CARDS
    02 CARD1 (A80)
    02 CARD2 (A80)
    02 CARD3 (A80)
END-DEFINE
MOVE '//JOBN JOB CLASS=G,MSGCLASS=X' TO CARD1
MOVE '//XXX EXEC PGM=IEFBRI4' TO CARD2
MOVE '//DD1 DD DSN=NATRJE.SOURCE,DISP=SHR' TO CARD3
MOVE 3 TO COUNT
MOVE 'L' TO FLAG
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
IF RETHEX = H'0000'
  WRITE 'JOB submitted successfully'
ELSE
  WRITE 'ERROR from NATRJE' RETHEX
END-IF
END
```

## プログラム例 - z/VSE :

次の例は、NATRJE を 3 回呼び出して 7 枚の JCL カードストリームを送信する Natural プログラムです。

```
DEFINE DATA LOCAL
  01 COUNT  (B4)
  01 FLAG   (A1)
  01 RETHEX (B2)
  01 CARDS  (A240)
  01 REDEFINE CARDS
    02 CARD1 (A80)
    02 CARD2 (A80)
    02 CARD3 (A80)
END-DEFINE
MOVE '* $$ JOB JNM=DSERV,CLASS=0,DISP=D' TO CARD1
MOVE '* $$ LST CLASS=A,DISP=D' TO CARD2
MOVE '// JOB DSERV TO DSERV SOURCE MEMBERS' TO CARD3
MOVE 3 TO COUNT
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
PERFORM RETCODE-CHECK
MOVE '// EXEC PROC=NATSPLP' TO CARD1
MOVE '// EXEC DSERV' TO CARD2
MOVE ' DSPLYS SD' TO CARD3
MOVE 3 TO COUNT
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
PERFORM RETCODE-CHECK
MOVE '/*' TO CARD1
MOVE '/&' TO CARD2
MOVE '* $$ EOJ' TO CARD3
MOVE 3 TO COUNT
MOVE 'L' TO FLAG
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
DEFINE SUBROUTINE RETCODE-CHECK
IF RETHEX NE H'0000'
  WRITE 'ERROR from NATRJE:' RETHEX
STOP
END-IF
END-SUBROUTINE
END
```

## プログラム例 1 - BS2000/OSD :

次の例は、NATRJE を 3 回呼び出して 9 枚の JCL カードストリームを送信する Natural プログラムです。

```
DEFINE DATA LOCAL
  01 COUNT  (B4)
  01 FLAG   (A1)
  01 RETHEX (B2)
  01 CARDS  (A240)
  01 REDEFINE CARDS
    02 CARD1 (A80)
    02 CARD2 (A80)
    02 CARD3 (A80)
END-DEFINE
MOVE '/LOGON' TO CARD1
MOVE '/SYSFILE SYSDTA=(SYSCMD)' TO CARD2
MOVE '/SYSFILE SYSIPT =IPT.PARM' TO CARD 3
MOVE 3 TO COUNT
  CALL 'NATRJE' CARDS COUNT FLAG RETHEX
  IF RETHEX NE H'0000' DO
    WRITE RETHEX (EM=HH)
  END-IF
MOVE '/SETSW ON=2' TO CARD1
MOVE '/EXEC NATB21' TO CARD2
MOVE 'LOGON APPLIC' TO CARD3
MOVE 3 TO COUNT
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
  IF RETHEX NE H'000' DO
    ...
    ...
  END-IF
MOVE 'RUNPGM' TO CARD1
MOVE 'FIN' TO CARD2
MOVE '/LOGOFF' TO CARD3
MOVE 3 TO COUNT
MOVE 'L' TO FLAG
CALL 'NATRJE' CARDS COUNT FLAG RETHEX
  ...
  ...
  ...
END
```

## プログラム例 2 - BS2000/OSD :

次の例は、NATRJE を 1 回呼び出して 9 枚の JCL カードストリームを送信する Natural プログラムです。

```
DEFINE DATA LOCAL
  01 COUNT  (B4)
  01 FLAG   (A1)
  01 RETHEX (B2)
  01 CARD1  (A80)
  01 CARD2  (A80)
  01 CARD3  (A80)
  01 CARD4  (A80)
  ...
  01 CARD9  (A80)
END-DEFINE
MOVE '/LOGON' TO CARD1
MOVE '/SYSFILE SYSDTA=(SYSCMD)' TO CARD2
...
MOVE '/LOGOFF' TO CARD9
MOVE 9 TO COUNT
MOVE 'L' TO FLAG
CALL 'NATRJE' CARD1 COUNT FLAG RETHEX
...
END
```

## NATRJE リターンコード

NATRJE モジュールを CALL で呼び出すと、CALL ステートメントの 4 番目のパラメータに次のリターンコードのいずれかが返されます。すべての環境に適用されるリターンコード、およびオペレーティングシステムによって異なる追加のリターンコードがあります。

- **すべての環境に共通のリターンコード**
- z/VSE の CICS およびバッチで使用する追加リターンコード
- z/OS の CICS で使用する追加リターンコード

■ BS2000/OSD で使用する追加リターンコード

すべての環境に共通のリターンコード

16進数	10進数	意味
00	00	正常に終了。
04	04	RJE ユーティリティは使用できません。
08	08	RJE ユーティリティは無効です。RJESIZE パラメータが 0 に設定されている可能性があります。
0C	12	JCL カードの枚数が正しくありません。
10	16	ファンクションコードが正しくありません。
14	20	RJE バッファスペースを使用できません。
18	24	パラメータの個数が正しくありません。
1C	28	送信中に I/O エラー。
20	32	ユーザー出口 NREXPG (『オペレーション』ドキュメントの「NREXPG - NATRJE のユーザー出口」を参照) によってジョブがフラッシュされました。

zVSE の CICS およびバッチで使用する追加リターンコード

<i>ffrr</i>	<i>ff</i> は XPCC 要求のファンクションコード、 <i>rr</i> は該当するリターンコードです。
-------------	-----------------------------------------------------------

zOS の CICS で使用する追加リターンコード

01 <i>nn</i>	CICS WRITEQ TD 失敗。 <i>nn</i> は 16 進形式の CICS レスポンスコードです。
01 <i>nn</i>	CICS CLOSE TD 失敗。 <i>nn</i> は 16 進形式の CICS レスポンスコードです。

BS2000/OSD で使用する追加リターンコード

9001	RJE バッファが見つかりません。
9002	バッファスペースを使用できません。
9003	LOGON コマンドが欠落しています。
9004	LOGON カードのみ生成されています。
9005	LOGON パラメータの数が多すぎます。
D010	ENTER マクロにエラーがあります。
Dxxx	オペレーティングシステムのエラー。 エラーメッセージはユーザープログラムに直接送信されます。 BS2000/OSD HELP コマンドによって追加の情報が提供されます。

## UTM/TIAM に適用可能な NATRJE 機能

このセクションでは、次のトピックについて説明します。

- SDF コマンド SET-LOGON-PARAMETER
- parm3 パラメータのその他の値
- BS2000/OSD データセットの名前

### SDF コマンド SET-LOGON-PARAMETER

実行対象のジョブの最初の JCL カード内で、ISP コマンド LOGON を SDF コマンド SET-LOGON-PARAMETER で置換できます。ただし、SDF コマンドを使用する際には次の制限があります。

- SET-LOGON-PARAMETER コマンドでは、次のコマンド省略形を使用する必要があります。

```
STLGP
```

- STLGP コマンドとともにユーザー ID、アカウント番号、またはパスワード（あるいはその組み合わせ）を指定する場合、ISP LOGON コマンドの構文を使用する必要があります。つまり、SDF キーワードオペランド USER-IDENTIFICATION、ACCOUNT、および PASSWORD は使用できません。これ以外のキーワードオペランドはいずれも、SDF コマンドで使用できます。

ISP および SDF コマンドの例：

ISP コマンド

```
/.job-id LOGON user-id,account-number,'password'
```

および対応する SDF コマンド：

```
/.job-id STGLP user-id,account-number,'password'
```

追加のキーワードオペランドを使用する SDF コマンド：

```
/.job-id STGLP user-id,account-number,'password', -  
/RESOURCES=*PARAMETERS(RUN-PRIORITY=220)
```

### parm3 パラメータのその他の値

値	説明
A	値 T と E の組み合わせ。
E	ジョブが生成され完了されます。オペレーティングシステム BS2000/OSD に送信する前に、パラメータ ERASE=YES が ENTER パラメータに追加されます。
T	ジョブが生成され完了されます。オペレーティングシステム BS2000/OSD に送信される前に、Natural MT パラメータを使用してタイムリミットが計算されます。MT を 0 に設定すると、タイムリミットは NTL になります。TIME=オペランドを指定すると、計算されたタイムリミットが ENTER パラメータに追加されます。

T、E または A を使用すると、ユーザーが作成した LOGON カード内にパラメータ TIME= または ERASE= が存在するかどうかはチェックされません。

### BS2000/OSD データセットの名前

JCL カード用に NATRJE が作成する BS2000/OSD データセットの名前は次のとおりです。

E.DDMMYY.HHMMSSSS.*program-name*.*user-id*

パラメータ	指定
DD	データセットの作成日
MM	データセットの作成月
YY	データセットの作成年
HH	データセットの作成時刻 (時間)
MM	データセットの作成時刻 (分)
SSSS	データセットの作成時刻 (秒およびミリ秒)
<i>program-name</i>	データセットを作成する Natural プログラムの名前
<i>user-id</i>	Natural ユーザー ID





# 14 オブジェクトハンドラ

---

オブジェクトハンドラは、Natural 環境での分散のために Natural および Natural 以外のオブジェクトを処理するように設計されています。これは、ソース環境のオブジェクトをワークファイルにアンロードし、それらをワークファイルからターゲット環境にロードすることによって実行されます。

- **オブジェクトハンドラの全般的な情報** バッチまたはオンラインモードでのオブジェクトハンドラの呼び出し、Natural Security の適用。
- **機能** オブジェクトハンドラメニュー機能（アンロード、ロード、ロードの再スタート、スキャン、表示、検索、および管理）の使用。
- **オブジェクト指定** オブジェクトハンドラメニュー機能で処理するオブジェクトの指定。
- **設定** オブジェクトハンドラメニュー機能のためのオプションとパラメータ設定の指定。
- **ワークプラン** オブジェクトハンドラ機能を実行するための標準プロシージャの使用。
- **名前と日時の指定** 名前、日付、時刻、および範囲の指定。
- **ワークファイル** オブジェクトハンドラに使用されるワークファイル。
- **ダイレクトコマンド** オブジェクトハンドラ機能を実行するためのダイレクトコマンドの使用。
- **バッチのコンディションコードとユーザー出口ルーチン** バッチモードで提供されるコンディションコードとユーザー出口ルーチン。
- **ツール** ステータス情報の表示およびトレースとレポートオプションの設定。
- **プロファイル設定** 個々のデフォルトプロシージャおよび標準プロシージャを定義するためのプロファイルの設定。
- **NATUNLD/NATLOAD および SYSTRANS からオブジェクトハンドラへの移行** ユーティリティ NATUNLD/NATLOAD および SYSTRANS からオブジェクトハンドラへの移行



# 15 オブジェクトハンドラの全般的な情報

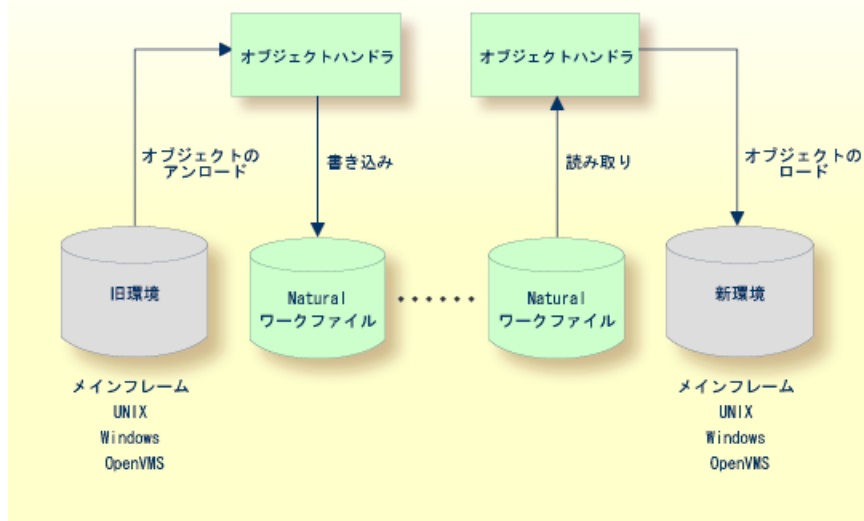
---

▪ オブジェクト転送の原理 .....	86
▪ オブジェクトハンドラの呼び出し .....	88
▪ バッチまたはダイレクトコマンドの呼び出し .....	89
▪ Natural プログラムからのオブジェクトハンドラコマンドの発行 .....	90
▪ レポート、再スタート、およびトレース用のテキストメンバ .....	90
▪ Natural Security .....	91
▪ 標準の PF キー .....	91

オブジェクトハンドラは、Natural システムライブラリ SYSOBJH 内にあるユーティリティ SYSOBJH、およびダイレクトコマンドインターフェイスで構成されています。また、Natural プログラムからオブジェクトハンドラ機能を実行するために、アプリケーションプログラミングインターフェイス OBJHAPI が用意されています。

## オブジェクト転送の原理

下の図に、オブジェクトをソース環境からワークファイルにアンロードし、ワークファイルからターゲット環境にロードすることによって、オブジェクトがオブジェクトハンドラでどのように転送されるかを示します。必要に応じて、ソース環境からターゲット環境へのワークファイルの転送に、FTP などのアプリケーションプロトコルを使用できます。



このセクションでは、次のトピックについて説明します。

- 転送環境とファイルセキュリティ
- オブジェクトハンドラによって処理されるオブジェクト

## ■ フォーマットオプション

### 転送環境とファイルセキュリティ

新旧の環境は、Adabas データベースに含まれている FNAT、FUSER、および FDIC の各システムファイル、またはメインフレーム上の VSAM ファイルシステム、または UNIX、OpenVMS、および Windows プラットフォーム上のファイルシステムです。次のセクションで説明するように、FNAT または FUSER システムファイルの Natural オブジェクトはライブラリに含めることができます。

ファイルセキュリティ（パスワードとサイファコード）は、Adabas または VSAM 環境のシステムファイルに対して定義されたセキュリティと関連付けられています。システムファイルにセキュリティが定義されている場合、オブジェクトハンドラ機能を実行する前に、必要なソース／ターゲットシステムファイルに対して、パスワード、サイファコード、および VSAM 名を指定する必要があります。これを行わなかった場合、Adabas または VSAM によって該当するエラーメッセージが発行されます。オブジェクトハンドラの開始時に Natural セッションに割り当てられているデフォルトのシステムファイルについては、セキュリティ情報を指定する必要はありません。

### オブジェクトハンドラによって処理されるオブジェクト

オブジェクトハンドラは、Natural ライブラリに含まれている Natural ソースオブジェクト（保存オブジェクトとも呼ばれる）およびカタログ化オブジェクト、Natural エラーメッセージ、Natural コマンドプロセッサソース、Natural DDM（データ定義モジュール）、Natural 関連オブジェクト、および Adabas FDT（フィールド定義テーブル）を転送します。

### フォーマットオプション

オブジェクトが処理されるソース環境およびターゲット環境に応じて、バイナリフォーマットまたはテキストフォーマットでデータを転送できます。

バイナリフォーマットは、ソースオブジェクトおよびカタログ化オブジェクト（DDMを含む）、エラーメッセージ、Natural コマンドプロセッサソース、Natural 関連オブジェクト、Adabas FDT に使用できます。

テキストフォーマットは、ソースオブジェクト、Natural コマンドプロセッサソース、エラーメッセージ、DDM、および Adabas FDT に適用されます。メインフレームおよび UNIX/OpenVMS/Windows プラットフォーム間では、テキストデータのみを転送できます。同一プラットフォーム間では、バイナリデータを転送できます。

## オブジェクトハンドラの呼び出し

---

このセクションでは、メニュー機能またはダイレクトコマンドのいずれかを使用してオブジェクトハンドラを呼び出す方法について説明します。

### ▶手順 15.1. 任意の Natural ライブラリからオンラインでオブジェクトハンドラを呼び出すには

- 1 次のシステムコマンドを入力します。

```
SYSOBJH
```

次のオプションを含む、オブジェクトハンドラのメインメニューが表示されます。

- Unload
- Load
- Scan
- View
- Administration

これらの機能の詳細と、上級ユーザーモードまたはウィザードでこれらの機能を実行する方法については、「[機能](#)」セクションを参照してください。

- 2 次のいずれかの方法で機能を選択します。

目的の機能に対応する項目の横にある入力フィールドに任意の文字を入力します。

または:

目的の機能に対応する PF キーを押します。

または:

コマンド行で、目的の機能に対応するオブジェクトハンドラコマンドを入力します。有効なコマンドの詳細については、「[ダイレクトコマンド](#)」セクションを参照してください。

### ▶手順 15.2. バッチモードまたはダイレクトコマンドオンラインモードでオブジェクトハンドラを呼び出すには

- 「[バッチまたはダイレクトコマンドの呼び出し](#)」および「[ダイレクトコマンド](#)」の説明に従って、システムコマンド SYSOBJH の後にダイレクトコマンドを入力します。

ダイレクトコマンドの実行後は、別のダイレクトコマンドを入力するか、ピリオド (.) を入力してオブジェクトハンドラを終了します。

## バッチまたはダイレクトコマンドの呼び出し

複数のコマンドをオンラインモードまたはバッチモードでオブジェクトハンドラに発行できます。コマンドシーケンスの最後のコマンドは、ピリオド (.)、STOP、END、QUIT または FIN にする必要があります。FIN にすると Natural セッションが終了します。

このセクションでは、次のトピックについて説明します。

- バッチモード
- オンラインモード

### バッチモード

オブジェクトハンドラへのコマンドは標準入力から読み込まれます。各コマンドは、キーワードまたはキーワードの後の入力デリミタ（セッションパラメータ ID）を入力することによって最大 20 のコマンドパーツ／文字列に分けることができます。各コマンドパーツ／文字列は 248 バイト以内でなければなりません。

コマンドが 1 行より長い場合、コマンドに属している各行（最後の行を除く）の末尾にセッションパラメータ CF で定義された文字（デフォルトは %）を入力します。これは次の行への継続を示します。ただし、これは、コマンド SYSOBJH を単独で 1 行に指定する場合に限り可能です。つまり、複数行コマンドが開始する同じ行に SYSOBJH を入力した場合は CF を使用できません。また、LS プロファイルパラメータを 250 に設定することをお勧めします。

例（ID が , に設定されているとみなす）：

```
SYSOBJH
UNLOAD PROG* LIBRARY MYLIB1, OBJTYPE N,%
WITH NEWLIBRARY MYLIB2%
WHERE REPORT TRANSFER
STOP
```

関連トピック：

- [ダイレクトコマンド](#)
- 『オペレーション』ドキュメントの「バッチモードでの *Natural*」

### オンラインモード

コマンド行に入力されるオブジェクトハンドラへのコマンドは、最高 20 のコマンドパーツで構成できます。

例：

```
SYSOBJH UNLOAD * LIB EXAMPLE WHERE TRANSFER
```

## Natural プログラムからのオブジェクトハンドラコマンドの発行

---

Natural システムライブラリ SYSOBJH 内にサブプログラムとして提供されている OBJHAPI アプリケーションプログラミングインターフェイスを使用して、Natural プログラムからオブジェクトハンドラにコマンドを発行できます。必要なパラメータおよび例については、ライブラリ SYSOBJH にある Natural プログラム DOC-API を参照してください。

## レポート、再スタート、およびトレース用のテキストメンバ

---

オブジェクトハンドラで作成されるレポート、再スタート、およびトレース用のデータは、Natural テキストメンバ（テキストタイプの Natural オブジェクト）としてワークプランライブラリに保存されます。オブジェクトハンドラでは、[Options] ウィンドウで明示的に指定されていないテキストメンバ名が生成されます。名前は、曜日と時刻を組み合わせで生成されます。例えば、21415568 という名前のメンバは、火曜日（週の 2 日目）の 14:15:56,8 に作成されています。

オブジェクトハンドラの Workplan-Library プロファイルオプション（「[プロファイルの設定](#)」を参照）を使用すると、テキストメンバが格納されるワークプランライブラリを指定できます。ROSY=ON プロファイルパラメータ（『[パラメータリファレンス](#)』を参照）で指定されている読み取り専用の FNAT または FUSER システムファイルが Natural セッションで使用されている場合、次のような状況においては、テキストメンバがスクラッチパッドファイルに格納されます（『[オペレーション](#)』ドキュメントを参照）。

- ワークプランライブラリに指定されているデータベース ID およびファイル番号が現在の FNAT または FUSER システムファイルと同じ場合
- ワークプランライブラリにデータベース ID およびファイル番号が指定されていない場合



## Natural Security

Natural Security 環境でオブジェクトハンドラを使用するには、オブジェクトハンドラに対してユーティリティプロファイルを Natural Security で定義する必要があります。少なくとも、デフォルトのプロファイルを定義する必要があります。ユーティリティプロファイルの詳細については、『Natural Security』ドキュメントの「Protecting Utilities」セクションを参照してください。

Natural Security がインストールされている場合、オブジェクトハンドラは、要求された機能が有効かどうかを確認するために、Natural Security の SYSOBJH ユーティリティプロファイルをチェックします。

ロード機能の実行中に Natural Security エラーが発生すると、次の操作が行われます。

- **[Write report]** オプションがオンラインモードで設定されていると、エラーメッセージがレポートファイルに出力され、現在のロードコマンドの処理は続行します。
- **[Write report]** オプションがバッチモードで設定されていると、エラーメッセージがレポートファイルに出力され、エラーが発生したロードコマンドの処理が完了した後、オブジェクトハンドラは終了します。
- **[Write report]** オプションが設定されていないと、エラーメッセージが発行され、ロードコマンドは終了します。

## 標準の PF キー

次の PF キーは、すべてのフルスクリーンマップで使用できます。

PF キー	説明
PF1	カーソルが置かれているフィールドのヘルプ機能呼び出します。
PF3	現在の画面を終了し、前の画面に戻ります。
PF6	リストの先頭にジャンプします。
PF7	リストを 1 ページ分スクロールアップします。 画面の場合は、1 画面 / ステップ戻ります。
PF8	リストを 1 ページ分スクロールダウンします。 ウィザード画面の場合は、次の画面 / ステップに進みます。
PF9	リストの最後にジャンプします。
PF10	ナビゲーションおよび特別な設定を行うためのコマンドを選択する <b>[Commands]</b> メニューが表示されます。「 <a href="#">ダイレクトコマンド</a> 」の「 <a href="#">ナビゲーションと特別な機能のためのコマンド</a> 」も参照してください。

## オブジェクトハンドラの全般的な情報

---

PF キー	説明
PF12	現在の機能をキャンセルします。
PF20	オブジェクトハンドラのアクティブなプログラムをすべてリスト表示します。技術的な問題を Software AG に連絡する際に役立つ情報です。

# 16 機能

---

このセクションでは、オブジェクトハンドラが提供する主な機能について説明します。


オブジェクトハンドラウィザードを利用して、アンロード、ロード、およびスキャン機能の実行に必要な手順を段階的に進めていくことができます。ウィザードはデフォルトで作動します。ウィザードの代わりに、熟練者用のアンロード、ロード、またはスキャンモードを使用する場合は、メインメニューで **[Advanced user]** の横にあるフィールドをマークします。また、オブジェクトハンドラの適切なプロファイルオプションを使用して、上級ユーザーモードをデフォルトとして設定することもできます。「[プロファイルの設定](#)」セクションも参照してください。

繰り返し設定とオブジェクト指定を定義する標準プロシージャを作成して、アンロード、ロード、またはスキャン機能の処理を自動化することができます。詳細については、「[ワークプラン](#)」セクションを参照してください。

このセクションでは、次のトピックについて説明します。

- [ウィザード](#)
- [上級ユーザー](#)
- [ロードの再スタート](#)
- [表示](#)
- [検索](#)
- [管理](#)
- [ワークプランライブラリの変更](#)
- [システムファイルの選択](#)
- [ライブラリの選択](#)

- ワークプランのリストと選択
- システムエラーメッセージの選択
- オブジェクトの選択

 **注意:**

1. 「ワークプランライブラリの変更」については、「管理」セクションで説明しています。
2. 「ワークプランのリストと選択」については、「管理」セクションの「ワークプランライブラリ内の利用可能なワークプランのリストの作成」で説明しています。

# 17 ウィザード

---

■ 手順 1 - プロシージャの開始 .....	96
■ 手順 2 - ワークファイルへのオブジェクトのアンロード／ワークファイルからのオブジェクトのロード／ワークファイルのオブジェクトのスキャン .....	97
■ 手順 3 - パラメータの設定 .....	99
■ 手順 4 - オブジェクトの選択 .....	100
■ 手順 5 - 処理の実行 .....	101
■ 手順 6 - 処理の続行 .....	102

オブジェクトハンドラでは、ウィザードを使用して次の処理の実行順序を指定できます。

- Natural システム環境から Natural ワークファイルへのデータのアンロード。
- ワークファイルから Natural システム環境へのデータのロード。
- Natural ワークファイルの内容のスキャン。

### ▶手順 17.1. ウィザードをアクティブにするには

- 必要に応じて、メインメニューで **[Advanced user]** フィールドを選択します（このフィールドはデフォルトでは未選択）。

画面（手順）間を移動するためのキー PF8 および PF7 がウィザードに表示されます。処理シーケンスをキャンセルするには、PF12 キーを使用します。

このセクションでは、アンロード、ロード、またはスキャンウィザードで実行される処理シーケンスについて説明します。

## 手順 1 - プロシージャの開始

---

### ▶手順 17.2. アンロード、ロード、またはスキャンプロシージャを開始するには

- 1 メインメニューで、目的の機能の横に任意の単一文字を入力するか、または対応する PF キーを使用して、**[Unload]**、**[Load]**、または **[Scan]** を選択します。

次のオプションを含む最初のウィザード画面が表示されます。

- **Unload/Load/Scan objects into/from Natural work file(s)**
- **Start Object Handler command procedure**

- 2 オブジェクトをワークファイルにアンロードする場合、オブジェクトをワークファイルからロードする場合、またはワークファイルのオブジェクトをスキャンする場合は、次の「[手順 2 - ワークファイルへのオブジェクトのアンロード／ワークファイルからのオブジェクトのロード／ワークファイルのオブジェクトのスキャン](#)」に進みます。

または:

オブジェクトのアンロード、ロード、またはスキャン用のコマンドプロシージャを使用する場合は、**[Start Object Handler command procedure]** を選択して次のように進みます。

1. 最初のウィザード画面で **[Start Object Handler command procedure]** を選択します。  
**[Procedure]** 画面が表示されます。

2. 次のいずれかのオプションを使用して、タイプ PROCEDURE のワークプラン名を [Name] フィールドに入力します。
  - トランザクションで使用するタイプ PROCEDURE のワークプラン名を入力します（「ワークプラン」も参照）。
  - [Select Workplan] を選択するか、または PF5 キーを押して、タイプ PROCEDURE の使用可能なワークプランのリストを表示します。選択するワークプランの横にある行で、コマンド「S」または「SE」を入力します。Enter キーを押してコマンドを実行し、[Procedure] 画面の [Name] フィールドに値を入力します。
3. 指定したワークプランを表示するには、[List Workplan] を選択するか、または PF4 キーを押します。

「管理」セクションの「ワークプランライブラリ内の利用可能なワークプランのリストの作成」も参照してください。

- 3 ENTER キーを押して続行します。
- 4 「手順5 - 処理の実行」に進みます。

## 手順2 - ワークファイルへのオブジェクトのアンロード／ワークファイルからのオブジェクトのロード／ワークファイルのオブジェクトのスキャン

▶手順17.3. Natural ワークファイルからオブジェクトをアンロードする／Natural ワークファイルにオブジェクトをロードする／Natural ワークファイルをスキャンするには

- 1 最初のウィザード画面で、[Unload/Load/Scan objects into/from Natural work file(s)] を選択します。
- 2 ENTER キーを押すか PF8（次へ）キーを押して続行します。ウィザードの [Options] 画面が開きます。この画面では、次のフィールド、コマンド、および代替用の PF キーを使用できます。

フィールド	PF キー	説明
Transfer format		<p>有効になるのは、[Use default options]（これがデフォルトです）を選択した場合に限られます。</p> <p>これを選択すると、ワークファイルにおける処理データの書き込み／読み取りが転送フォーマットで実行されます。「ワークファイル」の「ワークファイルフォーマット」も参照してください。</p> <p>アンロード機能：</p>

フィールド	PF キー	説明
		<p>アンロードするデータは、転送フォーマットでワークファイルに書き込まれます。以降のアンロードではこのフィールドの設定を変更する場合は、メインメニューに戻るか、コマンドGO UNLOAD END（「ダイレクトコマンド」の「ナビゲーションと特別な機能のためのコマンド」を参照）を入力してアンロード機能を再スタートする必要があります。</p> <p>ロード/スキャン機能：</p> <p>ロードまたはスキャンするデータは転送フォーマットにする必要があります。</p>
Unicode work file		<p>アンロード機能にのみ適用され、有効になるのは [Transfer format] が選択されている場合に限られます。</p> <p>このオプションが選択されている場合、すべてのオブジェクトソースが、ワークファイルに書き込まれる前に Unicode/UTF-8（Universal Transformation Format、8 ビット形式）に変換されます。</p>
Use PC File		<p>適用されるのは、Entire Connection がインストールされている場合に限られます。</p> <p>これを選択すると、処理データが Entire Connection ワークファイルに書き込まれます。</p>
PC File		<p>有効になるのは、 [Use PC File] を選択した場合に限られます。</p> <p>パス名および Entire Connection ワークファイルが使用されます。</p>
Use default options		<p>デフォルトオプションが使用されます（これがデフォルトです）。使用可能なオプションについては、「プロファイルの設定」および「設定」の「その他のオプションの設定」を参照してください。</p>
Set additional options	PF4	<p>有効になるのは、 [Use default options] を選択した場合に限られます。</p> <p>ウィザードの [Options] 画面を表示します。この画面で、デフォルト設定を変更したり、処理の順序を指定する追加オプションを入力したりすることができます。「設定」の「その他のオプションの設定」を参照してください。</p>
Use Option Workplan		<p>これが選択されている場合、タイプOPTIONのワークプランが使用されます（「ワークプラン」を参照）。</p>
Name		<p>有効になるのは、 [Use Option Workplan] を選択した場合に限られます。</p> <p>使用するタイプOPTIONのワークプランの名前。</p>
List Option Workplan	PF6	<p>有効になるのは、 [Use Option Workplan] を選択した場合に限られます。</p> <p>[Name] フィールドに指定されたワークプランの内容を表示します。</p>
Select Option Workplan	PF5	<p>有効になるのは、 [Use Option Workplan] を選択した場合に限られます。</p> <p>タイプOPTIONの使用可能なワークプランの選択リストが表示されます（「管理」の「ワークプランライブラリ内の利用可能なワークプランのリストの作成」も参照）。</p>



- 3 表示されたオプションを選択し、処理順序の指定に使用されるフィールドに（必要に応じて）値を入力します。
- 4 ENTER キーを押すか PF8（次へ）キーを押して続行します。

ウィザードの **[Parameters]** 画面が表示されます。

## 手順 3 - パラメータの設定

### ▶手順 17.4. 処理プロセスのパラメータを設定するには

- 1 **[Parameters]** 画面で次のオプションを選択し、（必要に応じて）処理シーケンスに使用されるフィールドに値を入力します。

フィールド	PF キー	説明
Do not use parameters		これが選択されている場合（デフォルト設定）、パラメータは設定されません。
Use global parameters		これが選択されている場合、グローバルパラメータが使用されます。「設定」の「 <a href="#">グローバルパラメータの設定</a> 」を参照してください。
Set global parameters	PF4	有効になるのは、 <b>[Use global parameters]</b> を選択した場合に限られます。  これが選択されている場合、 <b>[Parameters]</b> 画面が表示されます。キーワードと有効な入力値の詳細については、「設定」の「 <a href="#">グローバルパラメータの設定</a> 」および <i>parameter-setting</i> （ダイレクトコマンド）に関する項を参照してください。
Use Parameter Workplan		これが選択されている場合、タイプ PARAMETER のワークプランが使用されます（「 <a href="#">ワークプラン</a> 」を参照）。
Name		有効になるのは、 <b>[Use Parameter Workplan]</b> を選択した場合に限られます。  使用するタイプ PARAMETER のワークプランの名前。
List Parameter Workplan	PF6	有効になるのは、 <b>[Use Parameter Workplan]</b> を選択した場合に限られます。  これが選択されている場合、 <b>[Name]</b> フィールドで指定したワークプランの内容が表示されます。
Select Parameter Workplan	PF5	有効になるのは、 <b>[Use Parameter Workplan]</b> を選択した場合に限られます。  これが選択されている場合、タイプ PARAMETER の使用可能なワークプランの選択リストが表示されます（「 <a href="#">管理</a> 」の「 <a href="#">ワークプランライブラリ内の利用可能なワークプランのリストの作成</a> 」を参照）。

- 2 ENTER キーを押すか PF8 (次へ) キーを押して続行します。

[Select Unload/Load/Scan Type] 画面が表示されます。

## 手順 4 - オブジェクトの選択

---

### ▶手順 17.5. 処理するオブジェクトのタイプを選択するには

- 1 [Select Unload/Load/Scan Type] 画面で、以下の3つのオプションのいずれかを選択します。最初のオプションは、ロードおよびスキャン機能にのみ適用されることに注意してください。各オブジェクトタイプに適用されるキーワードと有効な値については、「[オブジェクト指定](#)」セクションの該当する記載を参照してください。

1. ワークファイルのすべてのオブジェクトを処理するには、[Load/Scan all objects] を選択します。
2. 特定タイプのオブジェクトを選択します。
  - **Natural** ライブラリオブジェクト
  - **Natural** システムエラーメッセージ
  - **Natural** コマンドプロセッサソース
  - **Natural** 関連オブジェクト
  - **DDM**
  - **FDT**

ENTER キーを押すか PF8 (次へ) キーを押して続行します。

選択したオブジェクトのタイプに応じて画面が表示され、処理するオブジェクトの選択条件を指定できます。

選択条件を入力し、必要に応じて [Details] を選択し (使用可能な場合) オブジェクトの詳細指定を行います。[Details] の詳細については、「[オブジェクト指定](#)」セクションの関連する説明を参照してください。

3. タイプ SELECTION または LIST のワークプランを使用するには、[Use Selection or List Workplan] を選択します。「[ワークプラン](#)」も参照してください。

ENTER キーを押すか PF8 (次へ) キーを押して続行します。

[Selection or List] 画面が表示されます。次のいずれかのオプションを使用して、タイプ SELECTION または LIST のワークプラン名を [Name] フィールドに入力します。

- ワークプランの名前を入力します。

または：

[**Select Workplan**] を選択するか、または PF5 (SelWP) キーを押して、使用可能なすべてのワークプランのリストを表示します。選択するワークプランの横にある行で、コマンド「S」または「SE」のいずれかを入力します。

Enter キーを押してコマンドを実行し、[**Selection or List**] 画面の [**Name**] フィールドに値を入力します。「管理」セクションの「[ワークプランライブラリ内の利用可能なワークプランのリストの作成](#)」も参照してください。

[**Name**] フィールドに入力したワークプランの内容を表示するには、[**List Workplan**] を選択するか、または PF4 (Li-WP) キーを押します。

- 2 Enter キーを押すか、または PF8 (Next) キーを押して続行します。

入力データから生成された処理コマンドがウィザードに表示されます。

コマンド SAVE を入力するか PF5 キー (保存) を押すと、タイプ PROCEDURE のワークプランとして表示されるコマンドを保存できます («[ワークプラン](#)」も参照)。

## 手順 5 - 処理の実行

### ▶ 手順 17.6. 処理プロシージャを実行するには

- 1 コマンド実行画面で Enter キーを押すか、または PF8 (Next) キーを押して設定を確認し、指定したオブジェクトを処理します。

必要に応じて PF7 (Back) キーを押して、コマンドの実行を確認する前に処理設定を修正します。

オブジェクトハンドラで機能が実行され、確認メッセージが表示されます。

- 2 ENTER キーを押して続行します。

処理されたオブジェクトのリストを示すレポート画面が表示されます。

- 3 PF3 (Exit) キーを押してレポート画面を終了するか、または PF12 (Canc) キーを押して機能を終了します。

ウィンドウが表示され、データの処理を続行するかどうかを選択できます。

- 4 [**No**] を選択してから Enter キーを押して機能を終了します。

または:

PF12 キーを押して機能を終了します。

メインメニューが表示されます。

## 手順 6 - 処理の続行

---

### ▶手順 17.7. 処理を続行するには

- 1 レポート画面で PF3 (Exit) キーを押します。

ウィンドウが表示され、次の処理手順に進むかどうかを選択できます。

- 2 [Yes] を選択します。

前の設定を再利用または変更するオプションを示す画面が表示されます。

# 18 上級ユーザー

---

▪ 上級ユーザーのアクティブ化 .....	104
▪ オブジェクトの処理 .....	104

このセクションでは、上級ユーザーモードを呼び出す方法、およびアンロード、ロード、スキャンの各機能を実行する方法について説明します。

## 上級ユーザーのアクティブ化

---

### ▶手順 18.1. 上級ユーザーモードをアクティブにするには

- メインメニューで **[Advanced user]** フィールドを選択します（このフィールドはデフォルトでは未選択）。

または:

オブジェクトハンドラプロファイルの **Advanced-Mode** パラメータを指定して、上級ユーザーモードをデフォルトとして設定します（「[プロファイルの設定](#)」を参照）。

## オブジェクトの処理

---

### ▶手順 18.2. 上級ユーザーモードでオブジェクトを処理するには

- 1 メインメニューで **[Advanced user]** フィールドを選択し、**[Unload]**、**[Load]**、または **[Scan]** を選択します。
- 2 ENTER キーを押して続行します。

**[Options]** および **[Parameters]** セクションを含む **[Unload/Load/Scan Settings]** 画面が表示されます。

- 3 「[設定](#)」セクションの説明に従って、オプションおよびパラメータを設定します。
- 4 ENTER キーを押して続行します。

**[Select Unload/Load/Scan Type]** 画面が表示されます。

- 5 処理するオブジェクトを選択します。「[オブジェクト指定](#)」セクションも参照してください。
- 6 **[Details]** を選択して、追加の選択条件を指定します。「[オブジェクト指定](#)」の関連するセクションを参照してください。
- 7 ENTER キーを押して続行します。

- オブジェクトハンドラプロファイルでパラメータ `Display-Cmd-in-Advanced-Mode` が `N` (`No`) に設定されている場合（これがデフォルトです）、またはそのようなプロファイルが存在しない場合は、入力データから生成されたコマンドが、選択データを指定した直後に実行されます。「[プロファイルの設定](#)」も参照してください。

[**Write report**] フィールド（これがデフォルトです）を選択した場合、[**Display Unload/Load/Scan Report**] 画面が開き、処理されたオブジェクトのリストが表示されます。「設定」セクションの「[ワークファイルオプション](#)」も参照してください。

- オブジェクトハンドラプロファイルでパラメータ `Display-Cmd-in-Advanced-Mode` が Y (Yes) に設定されている場合（「[プロファイルの設定](#)」を参照）、またはコマンド `SET ADVANCEDCMD ON`（「[ナビゲーションと特別な機能のためのコマンド](#)」を参照）が前に実行されている場合は、入力データから生成されたコマンドを示す画面が表示されます。

コマンド `SAVE` を入力するか `PF5` キー（保存）を押すと、タイプ `PROCEDURE` のワークプランとして表示されるコマンドを保存できます（「[ワークプラン](#)」も参照）。

`Enter` キーを押してコマンドの実行を確認するか、または `PF3` (`Exit`) キーを押してコマンドの実行を確認する前に処理設定を変更します。

[**Write report**] フィールド（これがデフォルトです）を選択した場合、[**Display Unload/Load/Scan Report**] 画面が開き、処理されたオブジェクトのリストが表示されます。「設定」セクションの「[ワークファイルオプション](#)」も参照してください。

---



# 19 ロードの再スタート

---

ロードの再スタート機能を使用すると、異常終了したロード機能を再スタートさせることができます。ワークファイルの処理が完了する前にロード機能が停止した場合、ロードの再スタート機能を使用して、停止した場所から処理を続行できます。

ロードの再スタートでは、ロードに関して指定された選択条件、オプション、およびパラメータ設定に従って、再スタート情報がテキストメンバ（テキストタイプのNaturalオブジェクト）に書き込まれている必要があります。

## ▶手順 19.1. ロード中に環境を設定するには

1 [Load Options] 画面で、次の操作を実行します。

- [Write restart information] オプションにマークを付けます。
- [Restart text member] フィールドに、再スタート情報が書き込まれているテキストメンバの名前を入力します。テキストメンバは、ワークプランライブラリに含まれている必要があります。

または：

[Restart text member] の横にある [Select text member] フィールドにマークを付け、ワークプランライブラリに存在するワークプランのリストからテキストメンバを選択します。

[Load Options] 画面については、「設定」セクションの「ワークファイルオプション／レポートオプション」を参照してください。

2 ロード機能を実行します。

## ▶手順 19.2. ロードの中断後にロードの再スタートを実行するには

- オブジェクトハンドラ画面で、コマンド行に次のコマンドを入力します。

### GO RESTART

[**Restart Options**] 画面が表示され、[**Restart text member**] フィールドに名前を入力するか、または [**Select text member**] にマークを付けてリストからテキストメンバを選択して、テキストメンバを指定できます。

または:

次のダイレクトコマンドを使用します。

### RESTART

RESTART の構文については、「[基本的なコマンド構文](#)」セクションを参照してください。

関連トピック:

「[管理](#)」セクションの「[ワークプランライブラリの変更](#)」

「[ナビゲーションと特別な機能のためのコマンド](#)」セクションの GO RESTART

## 20 表示

---

■ Natural ライブラリオブジェクト .....	110
■ Natural システムエラーメッセージ .....	111
■ Natural コマンドプロセッサソース .....	112
■ FDT .....	113
■ Natural 関連オブジェクト .....	114
■ DDM .....	115

この機能は、Naturalシステム環境に含まれているすべてのオブジェクトを表示するために使用します。選択するオブジェクトのタイプによっては、この機能を使用して必要に応じてオブジェクトを削除することもできます。

### ▶手順 20.1. ビュー機能を起動するには

- メインメニューで **[View]** を選択します。

または:

他の任意のオブジェクトハンドラ画面で、次のダイレクトコマンドを入力します。

```
GO VIEW
```

(「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」も参照してください。)

選択可能なすべてのオブジェクトタイプを示す **[Select View Type]** 画面が表示されます。

このセクションでは、**[Select View Type]** 画面にリストされるオブジェクトタイプの表示方法について説明します。

## Natural ライブラリオブジェクト

---

Natural ライブラリオブジェクトとは、プログラミングオブジェクトおよびユーザー定義エラーメッセージのことです。

### ▶手順 20.2. Natural ライブラリオブジェクトを表示するには

- 1 **[Select View Type]** 画面の **[Natural library objects]** を選択します。

現在のNatural環境で使用可能なすべてのシステムファイルのリストを示す **[View System Files]** 画面が表示されます。

画面の列の詳細については、「[システムファイルの選択](#)」で説明している **[Select System File]** 画面を参照してください。列の内容が同じです。

- 2 **[Cmd]** 列で、選択するシステムファイルの横に任意の単一文字を入力します。デフォルトでは、現在の FUSER または FNAT システムファイルが選択されます。

指定したシステムファイルで使用可能なすべてのライブラリのリストを示す **[View Libraries]** 画面が表示されます。

特定のライブラリからライブラリリストを開始したり、**[Library]** フィールドにライブラリ名またはライブラリ名の範囲を入力してオブジェクトにフィルタをかけたりすることがで

きます。有効な名前の範囲については、「名前と日時の指定」セクションの「名前」を参照してください。

3

- 4 [Cmd] 列で、選択するライブラリの横に次の行コマンドのいずれかを入力します。

L

LI

S

SE

指定したライブラリに含まれているすべてのオブジェクトのリストを示す [View Library Objects] 画面が表示されます。

この画面の詳細については、「オブジェクトの選択」で説明している [List] 画面に関する説明を参照してください。列の内容が同じです。

- 5 [Cmd] 列で、表示するオブジェクトの横に次の行コマンドのいずれかを入力します。

L

LI

または:

必要に応じて、削除するオブジェクトの横に次の行コマンドを入力します。

DE

入力したコマンドに応じて、選択したオブジェクトのソースコードが画面に表示されるか、または削除機能の実行に使用する確認ウィンドウが表示されます。

## Natural システムエラーメッセージ

### ▶手順 20.3. Natural システムエラーメッセージを表示するには

- 1 [Select View Type] 画面の [Natural system error messages] を選択します。  
[View System Error Messages] 画面が表示されます。
- 2 目的のシステムエラーメッセージが現在の FNAT または FUSER システムファイルに保存されていない場合は、[DBID] フィールドのデータベース ID および [FNR] フィールドのファイル番号を置き換えます。必要に応じて、システムファイルの Adabas パスワード ([Password]) とサイファコード ([Cipher]) を入力します。

必要に応じて、[Error number range]、[Languages]、および [Short/Long/All] フィールドで選択条件を指定します。これらのフィールドは、表示機能を実行してエラーメッセージのリストが表示される [View System Error Messages] 画面の [Number]、[Language]、[S/L] 列にそれぞれ対応しています。

この [View System Error Messages] 画面の詳細については、[List System Error Messages] 画面に関する説明を参照してください。列の内容が同じです。

- 3 選択したシステムエラーメッセージが画面に表示されたら、[Cmd] 列で表示するエラーメッセージの横に、次のいずれかの行コマンドを入力します。

L

LI

または:

[Cmd] 列で、目的のオブジェクトの横に次の行コマンドを入力すると、エラーメッセージを削除できます。

DE

入力したコマンドに応じて、選択したエラーメッセージのソースコードが画面に表示されるか、または削除機能の実行に使用する確認ウィンドウが表示されます。

## Natural コマンドプロセッサソース

▶手順 20.4. Adabas ファイルに保存されている Natural コマンドプロセッサソースを表示するには

- 1 [Select View Type] 画面の [Natural command processor sources] を選択します。  
[View Natural Command Processors] 画面が表示されます。
- 2 目的の Natural コマンドプロセッサソースが現在の FUSER システムファイルに保存されていない場合は、[DBID] フィールドに必要なデータベース ID を入力し、[FNR] フィールドにファイル番号を入力します。  
  
必要に応じて、[Password] フィールドに Adabas パスワードを入力し、[Cipher] フィールドにサイファコードを入力します。  
  
Natural コマンドプロセッサソースが保存されているすべてのライブラリのリストを示す [View Libraries] 画面が表示されます。
- 3 特定のライブラリからライブラリリストを開始したり、[Library] フィールドにライブラリ名またはライブラリ名の範囲を入力して Natural コマンドプロセッサソースにフィルタをかけたりすることができます。有効な名前の範囲については、「名前と日時の指定」セクションの「名前」を参照してください。

- 4 [Cmd] 列で、選択するライブラリの横に次の行コマンドのいずれかを入力します。

L

LI

S

SE

指定したライブラリに含まれているすべての Natural コマンドプロセッサソースのリストを示す [View Command Processors] 画面が表示されます。

この画面の詳細については、[List] 画面に関する説明を参照してください。列の内容が同じです。

- 5 [Cmd] 列で、目的のオブジェクトの横に次の行コマンドを入力すると、そのオブジェクトを削除することができます。

DE

削除機能の実行に使用する確認ウィンドウが表示されます。

## FDT

### ▶手順 20.5. Adabas データベースで使用可能な FDT を表示するには

- 1 [Select View Type] 画面の [FDTs] を選択します。

[View FDTs] 画面が表示されます。

- 2 目的のオブジェクトが現在の FNAT または FUSER システムファイルに保存されていない場合は、[DBID] フィールドのデータベース ID を置き換え、必要に応じて [FNR from] および [FNR to] フィールドに入力されているファイル番号の範囲を置き換えます。

指定したデータベースのファイル範囲にあるすべての FDT のリストを示す [View FDTs for DBID] 画面が表示されます。

## Natural 関連オブジェクト

### ▶手順 20.6. Natural 関連オブジェクトを表示するには

- 1 [Select View Type] 画面の [Natural-related objects] を選択します。

[Select Natural-Related Type] 画面が表示されます。

- 2 プロファイル、デバッグ環境、DL/I サブファイルなど、表示する Natural 関連オブジェクトタイプを選択します。

表示する Natural 関連オブジェクトの場所を指定できる [View] 画面が表示されます。

- 3 目的のオブジェクトが現在のシステムファイル（プロファイルではFNAT、デバッグ環境ではFUSER、DL/IサブファイルではFDIC）に保存されていない場合は、[DBID] フィールドに必要なデータベースIDを入力し、[FNR] フィールドにファイル番号を入力します。

必要に応じて、[Password] フィールドにAdabasパスワードを入力し、[Cipher] フィールドにサイファコードを入力します。

指定したタイプおよびシステムファイルのすべてのNatural関連オブジェクトのリストが画面に表示されます。

この画面の詳細については、[List] 画面に関する説明を参照してください。列の内容が同じです。

- 4 [Cmd] 列で、表示するオブジェクトの横に次の行コマンドのいずれかを入力します。

L

LI

または:

[Cmd] 列で、目的のオブジェクトの横に次の行コマンドを入力すると、そのオブジェクトを削除することができます。

DE

入力したコマンドに応じて、選択したNatural関連オブジェクトのソースコードが画面に表示されるか、削除機能の実行に使用する確認ウィンドウが表示されます。



## DDM

### ▶手順 20.7. Natural DDM を表示するには

- 1 [Select View Type] 画面の [DDMs] を選択します。

[View DDMs] 画面が表示されます。

- 2 目的のDDMが現在のFDICシステムファイルに保存されていない場合は、[DBID] フィールドに必要なデータベースIDを入力し、[FNR] フィールドにファイル番号を入力します。

必要に応じて、[Password] フィールドにAdabasパスワードを入力し、[Cipher] フィールドにサイファコードを入力します。

指定したシステムファイルに含まれているすべてのDDMのアルファベット順リストが画面に表示されます。

この [View DDMs] 画面の詳細については、[List] 画面に関する説明を参照してください。列の内容が同じです。

- 3 [Cmd] 列で、表示する DDM の横に次の行コマンドのいずれかを入力します。

L

LI

または:

[Cmd] 列で、目的の DDM の横に次の行コマンドを入力すると、その DDM を削除することができます。

DE

入力したコマンドに応じて、選択した DDM のソースコードが [List DDM] 画面に表示されるか、または削除機能の実行に使用する確認ウィンドウが表示されます。



# 21 検索

---

この機能は、Natural環境でオブジェクトを検索し、検出されたオブジェクトのレポートリストを生成するために使用します。

## ▶手順 21.1. 検索機能を起動するには

- オブジェクトハンドラの任意の画面で、コマンド行に次のコマンドを入力します。

```
GO FIND
```

検索機能で生成されたレポート画面に表示される列の詳細については、「[オブジェクト指定](#)」セクションを参照してください。GO FIND で提供されるサブコマンドについては、「[ダイレクトコマンド](#)」セクションの「[ナビゲーションと特別な機能のためのコマンド](#)」を参照してください。



## 22 管理

---

- ワークプランライブラリ内の利用可能なワークプランのリストの作成 ..... 120
- 新しいワークプランの作成 ..... 122
- ワークプランライブラリの変更 ..... 124

この機能は、オブジェクトハンドラのワークプランをメンテナンスするために使用します。

ワークプランと適用される構文の詳細については、「[ワークプラン](#)」セクションおよび「[ダイレクトコマンド](#)」セクションを参照してください。

このセクションでは、**[Administration]** 画面に表示されるオプションについて説明します。ワークプランを変更する手順については、「[ワークプランライブラリ内の利用可能なワークプランのリストの作成](#)」を参照してください。

## ワークプランライブラリ内の利用可能なワークプランのリストの作成

---

この機能は、ワークプランの編集や実行などの処理を行うため、ワークプランライブラリに含まれているすべてのワークプランをリストし、ワークプランを選択するために使用します。

### ▶手順 22.1. ワークプランをリストするには

- **[Administration]** 画面で、**[List the available Workplans in the Workplan library]** を選択するか、または PF4 (List) キーを押します。

ワークプランライブラリに含まれているすべてのワークプランのリストを示す **[List Workplans]** 画面が表示されます。

テキストタイプの Natural オブジェクトがワークプランの場合、ワークプランタイプおよびワークプランの説明の最初の 50 バイトが表示されます。PF5 キーを押して、追加情報を表示できます。

**[List Workplans]** 画面は、**[Unload/Load/Scan Settings]** 画面などに含まれている選択機能によっても表示されます。

**[List Workplans]** 画面に表示される列およびワークプランで実行できるコマンドについては、次のセクションで説明します。

- [List Workplans] 画面の列およびコマンド

### [List Workplans] 画面の列およびコマンド

次の表では、[List Workplans] 画面に表示される列およびコマンドについて説明します。

各列見出しの下にある入力フィールドを使用して、特定のワークプランからリストを開始したり、ワークプランにフィルタをかけたることができます。有効な入力値については、以下の表を参照してください。

列	PF キー	説明
Cmd		目的のワークプランの横にある入力フィールドで、次の行コマンドを入力できます。
	C または CH	構文をチェックします。 タイプ PROCEDURE、SELECTION、PARAMETER、および OPTION のワークプランにのみ適用されます。
	DE	ワークプランを削除します。
	ED	ワークプランを編集します。  「 <a href="#">ワークプランの保存</a> 」で説明している [Save Workplan] ウィンドウで、ワークプランの名前または説明を変更できます。
	EX	ワークプランを実行します。タイプ PROCEDURE のワークプランにのみ適用されます。
	L または LI	ワークプランをリストします。
	S または SE	現在の機能に使用するワークプランを選択します。 [Unload/Load/Scan Settings] 画面などの選択機能から [List Workplans] 画面を表示した場合にのみ適用されます。
Name		ワークプランの名前。  「名前と日時の指定」の「名前」の説明に従って、単一の名前または名前の範囲を入力できます。
Type		PROCEDURE などのワークプランタイプ。  有効な入力値は次のとおりです。  PROCEDURE または P SELECTION または S LIST または L PARAMETER または A OPTION または O

列	PF キー	説明
		TEXT または T  また、すべてのタイプとしてアスタリスク (*) を入力したり、SL などのようにショートタイプの組み合わせを入力したりすることもできます。
<b>Description</b>		ワークプランの説明。  「名前と日時の指定」の「名前」の説明に従って、単一の説明または説明の範囲を入力できます。
<b>User ID</b>		PF5 キーでのみ表示できます。  ワークプランを作成したユーザーの ID。  「名前」の説明に従って、単一のユーザー ID またはユーザー ID の範囲を入力できます。
<b>Date</b>		PF5 キーでのみ表示できます。  ワークプランの作成日。  「名前と日時の指定」の「日付」の説明に従って、単一の日付または日付の範囲を入力できます。
<b>Time</b>		PF5 キーでのみ表示できます。  ワークプランの作成時刻。  「名前と日時の指定」の「時刻」の説明に従って、単一の時刻または時刻の範囲を入力できます。
	PF4	追加情報表示 (PF5) から標準表示に切り替えます。
	PF5	追加情報 ( <b>User ID</b> 、 <b>Date</b> 、および <b>Time</b> ) を表示します。

## 新しいワークプランの作成

この機能によって [Create a new Workplan] 画面が表示され、この画面で、新しいワークプランのタイプおよびワークプランの編集に使用するフォーマットを指定できます。

[Free Format Editing] オプションを選択しない (フィールドはマークされず、これがデフォルト設定です) と、タイプ OPTION、PARAMETER、および SELECTION のワークプランでは画面に入力フィールドが表示されます。

[Free Format Editing] オプションを選択する (フィールドがマークされる) か、または別のタイプのワークプランを作成すると、ワークプランの内容を入力できる編集エリアを含むマップが表示されます。「ワークプラン」セクションの「ワークプランの内容」も参照してください。



フリーフォーマット編集のオン/オフの設定に使用できる代替のダイレクトコマンドについては、「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」にある SET コマンドを参照してください。

このセクションでは、次のトピックについて説明します。

- PROCEDURE ワークプランの作成
- LIST ワークプランの作成
- ワークプランの保存

## PROCEDURE ワークプランの作成

現在のオブジェクトハンドラ機能のために生成されたコマンドから、タイプ PROCEDURE のワークプランを作成できます。

### ▶手順 22.2. 生成されたコマンドから PROCEDURE ワークプランを作成するには

- 1 実行する機能用にコマンドが生成されて画面に表示されるまでは、オブジェクトハンドラウィザードを使用して、ワークプランに使用する機能を実行します。

または:

上級ユーザーモードでは、次のいずれかの方法を選択して、生成されたコマンドの表示をアクティブにします。

- 次のオブジェクトハンドラコマンドを入力します。

```
SET ADVANCEDCMD ON
```

または:

オブジェクトハンドラプロファイルで、パラメータ `Display-Cmd-in-Advanced-Mode` を Y (Yes) に設定します。詳細については、「[プロファイルの設定](#)」を参照してください。

- 実行する機能用にコマンドが生成されて画面に表示されるまでは、ワークプランに使用する機能を実行します。

- 2 PF5 キー（保存）を選択します。

[**Save Workplan**] ウィンドウが表示されます。

- 3 新しいワークプランの名前と説明を入力し、Enter キーを押します。

ワークプランが、PROCEDURE ワークプランとしてワークプランライブラリに保存されます。現在の機能のために生成されたコマンドが含まれます。

## LIST ワークプランの作成

タイプ LIST のワークプランの作成の詳細については、「[オブジェクトリスト-LIST ワークプラン](#)」セクションを参照してください。

## ワークプランの保存

### ▶手順 22.3. ワークプランを保存するには

- 1 ワークプランの編集が完了したら、コマンド行に次のコマンドを入力します。

```
SAVE
```

または:

PF5 キー（保存）を選択します。


[**Save Workplan**] ウィンドウが表示されます。

- 2 ワークプランの名前と説明を入力または変更し、Enter キーを押します。

ワークプランが、指定した名前で作成されたワークプランライブラリに保存されます。

## ワークプランライブラリの変更

この機能は、ワークプランライブラリを変更するときに使用します。すべてのワークプランはワークプランライブラリに保存する必要があります。

 **注意:** また、オブジェクトハンドラプロファイルで Workplan-Library パラメータを指定して、ワークプランのデフォルトライブラリを設定することもできます（「[プロファイルの設定](#)」を参照）。

[**Change Workplan Library**] 画面には、次のフィールドが表示されます。

フィールド	説明
<b>Library</b>	ワークプランライブラリの名前。デフォルトのライブラリは WORKPLAN です。
<b>Select library</b>	使用可能なすべてのワークプランライブラリのリストを表示します。「 <a href="#">ライブラリの選択</a> 」も参照してください。 同等の PF キー：PF4 (SeLib)
<b>DBID/FNR</b>	ワークプランライブラリが保存されているデータベース ID (DBID) とファイル番号 (FNR) を指定します。値が指定されていない場合、現在の FUSER または FNAT システムファイルが使用されます。

---

フィールド	説明
Passw./Ciph.	ワークプランライブラリが保存されている Adabas ファイルのパスワードとサイファコード。



## 23 システムファイルの選択

---

アンロード機能に使用するシステムファイルをリストから選択することができます。

アンロード機能を上級ユーザーモードで実行している場合、オブジェクト指定画面からこの選択リストを作成できます。

以下の手順は、Natural ライブラリオブジェクトをアンロードする場合の機能の使用例です。

### ▶手順 23.1. システムファイルをリストから選択するには

- 1 **[Unload Natural Library Objects]** 画面で、**[Select DBID/FNR]** を選択するか、または PF5 キー (DBID) を押します。

**[Select System File]** ウィンドウが表示され、現在の Natural 環境で使用可能なシステムファイルの名前 (**[System File]**)、データベース ID (**[DBID]**)、およびファイル番号 (**[FNR]**) がリスト表示されます。User defined は、ユーザーが指定したシステムファイルであることを表します。

- 2 目的のシステムファイルの横にある **[Sel]** 列に任意の単一文字を入力して、機能処理に使用するシステムファイルを選択します。デフォルトでは、FUSER システムファイルが選択されます。

または：

User-defined の横にある **[DBID]** および **[FNR]** フィールドに、選択するシステムファイルのデータベース ID とファイル番号をそれぞれ入力することができます。必要に応じて、**[Password/VSAM name]** および **[Cipher Code]** 列に、システムファイルの Adabas パスワードまたは VSAM 名および Adabas サイファコードを入力します。

選択したシステムファイルのデータベース ID とファイル番号が、**[Unload Natural Library Objects]** 画面の **[DBID]** / **[FNR]** フィールドに入力されます。デフォルトのシステムファイルを選択した場合、これらのフィールドは空のままになります。



# 24 ライブラリの選択

---

アンロード機能に使用するライブラリをリストから選択することができます。

上級ユーザーモードで実行した場合のアンロード機能のオブジェクト指定画面に表示される [Select library] (または [Select]) 機能には、選択リストが表示されます。

以下の手順は、Natural ライブラリオブジェクトをアンロードする場合の1つまたは複数のライブラリの選択例です。

## ▶手順 24.1. 1つのライブラリをリストから選択するには

- 1 [Unload Natural Library Objects] 画面で、[Select library] を選択するか PF4 キー (SeLib) を押します。

[Select Library] ウィンドウが開いて、すべてのライブラリ、データベースID (DBID)、およびライブラリが保存されているシステムファイルのファイル番号 (FNR) がリスト表示されます。

- 2 目的のライブラリの横にある [Cmd] 列に、任意の単一文字を入力します。
- 3 ENTER キーを押します。

[Unload Natural Library Objects] 画面の [Library] フィールドおよび [DBID] / [FNR] フィールドが、指定の名前および番号でそれぞれ埋められます。[DBID] / [FNR] フィールドに値が入力されていない場合、または「0」が入力されている場合、現在の FUSER または FNAT システムファイルが選択されます。

## ▶手順 24.2. 複数のライブラリをリストおよび選択するには

- 1 [Unload Natural Library Objects] 画面で、[Select library] を選択するか PF4 キー (SeLib) を押します。

[**Select Library**] ウィンドウが開いて、すべてのライブラリ、データベースID (**DBID**)、およびライブラリが保存されているシステムファイルのファイル番号 (**FNR**) がリスト表示されます。

- 2 [**Library**] フィールドに単一の名前または名前の範囲を入力して、選択するライブラリにフィルタをかけます。1つのライブラリ名を入力すると、このライブラリでリストが開始します。有効な名前の範囲については、「名前と日時の指定」セクションの「名前」を参照してください。

または:

[**DBID**] および [**FNR**] フィールドに、選択するライブラリが含まれているシステムファイルのデータベースIDとファイル番号をそれぞれ入力します。値が入力されていない場合、または「0」が入力されている場合、現在の FUSER または FNAT システムファイルが使用されます。

[**Select Library**] ウィンドウが [**Exceptions**] 画面から表示された場合、[**DBID**] および [**FNR**] は読み取り専用フィールドになります。

- 3 ENTER キーを押します。

[**Select Library**] ウィンドウに、指定した範囲のすべてのライブラリがリスト表示されま

- 4 PF4 キー (Se Rng) を押します。

[**Unload Natural Library Objects**] 画面の [**Library**] フィールドおよび [**DBID**] / [**FNR**] フィールドが、指定の名前 (または範囲) および番号でそれぞれ埋められます。  
[**DBID**] / [**FNR**] フィールドに値が入力されていない場合、または「0」が入力されている場合、現在の FUSER または FNAT システムファイルが選択されます。



# 25 システムエラーメッセージの選択

---

- 列およびコマンド ..... 132

アンロードする Natural システムエラーメッセージをリストから選択することができます。


アンロード機能を上級ユーザーモードで実行している場合、オブジェクト指定画面からこの選択リストを作成できます。

### ▶手順 25.1. Natural システムエラーメッセージを選択するには

- 必要に応じて、[Unload Natural System Error Messages] 画面で [Error number from/to] フィールドのメッセージ番号（デフォルトはすべての番号）を変更し、[Select system error messages] を選択します。

指定のシステムファイルに含まれているすべてのシステムエラーメッセージのリストを示す [List System Error Messages] 画面が表示されます。

この画面については、次のセクションで説明します。

 **注意:** ユーザー定義エラーメッセージの選択機能については、「[オブジェクトの選択](#)」セクションで説明します。

## 列およびコマンド

次の表では、[List System Error Messages] 画面に表示される列およびコマンドについて説明します。

各列見出しの下にある入力フィールドを使用して、特定のシステムエラーメッセージからリストを開始したり、メッセージにフィルタをかけたりすることができます。有効な入力値については、以下の表を参照してください。

列	PF キー	説明
Cmd		目的のシステムエラーメッセージの横にある入力フィールドに、次のいずれかの行コマンドを入力できます。
	L または LI	メッセージの短文テキストおよび長文テキストをリスト表示します。
	S または SE UL または U	以降のアンロード処理のためにメッセージを選択します。 注意： これらのコマンドは、以降の処理のために選択されたメッセージをマークするだけです。アンロード機能を

列	PF キー	説明						
		<p>実行するには、PF2 キー (Unloa) を押す必要があります (下記参照)。</p>						
	DE	メッセージを削除します。						
	DL	メッセージの長文テキストのみを削除します。						
<b>Number</b>		<p>システムエラーメッセージの番号。</p> <p>単一の番号または番号の範囲を入力できます。有効な範囲は次のとおりです。</p> <p><i>value</i>* <i>value</i> で開始する番号を持つすべてのメッセージ。</p> <p><i>value</i>&gt; <i>value</i> 以上の番号を持つすべてのメッセージ。 例：10&gt;</p> <p><i>value</i>&lt; <i>value</i> 以下の番号を持つすべてのメッセージ。 例：100&lt;</p>						
<b>S/L</b>		<p>システムエラーメッセージのテキストの種類：</p> <table border="1"> <tr> <td>S</td> <td>短文テキスト。</td> </tr> <tr> <td>L</td> <td>長文テキスト。</td> </tr> <tr> <td>A</td> <td>短文テキストと長文テキストの両方または片方。</td> </tr> </table>	S	短文テキスト。	L	長文テキスト。	A	短文テキストと長文テキストの両方または片方。
S	短文テキスト。							
L	長文テキスト。							
A	短文テキストと長文テキストの両方または片方。							
<b>Language</b>		<p>システムエラーメッセージの言語コード。</p> <p>選択するエラーメッセージに対して、8 個までの有効な言語コード (1 は英語など) を入力できます。</p> <p>アスタリスク (*) にすると、すべての言語コードが選択されます。</p>						
エラーメッセージのテキスト		システムエラーメッセージの短文テキスト。						
	PF2	<p>処理を行うため、選択されたシステムエラーメッセージのアンロードを開始します。</p> <p>別の方法としては、コマンド行に次のダイレクトコマンドのいずれかを入力します。</p> <p>UNLOAD または UNLD</p>						

## システムエラーメッセージの選択

---

列	PF キー	説明
	PF11	<p>PF2 キーを使用した以降のアンロード処理のため、リスト表示されたすべてのシステムエラーメッセージをマークします。</p> <p>別の方法としては、コマンド行に次のダイレクトコマンドのいずれかを入力します。</p> <p>SELECT ALL または SEL ALL</p>

## 26 オブジェクトの選択

---


- リスト画面の列およびコマンド ..... 136

## オブジェクトの選択

アンロードするオブジェクトをリストから選択することができます。この選択リストは、オブジェクトのソースのリスト表示やオブジェクトの削除などの他の目的にも使用できます。

上級ユーザーモードで実行した場合のアンロード機能のオブジェクト指定画面に表示される [Select objects] (または [Select]) 機能には、選択リストが表示されます。

選択リストは、次のセクションで説明する [List] 画面に表示されます。

 **注意:** Natural システムエラーメッセージの選択機能については、「[システムエラーメッセージの選択](#)」セクションで説明します。

## リスト画面の列およびコマンド

次の表では、[List] 画面に表示される列およびコマンドについて説明します。

[List] 画面での列の表示は、[Select Unload Type] メニューで選択したオブジェクトのタイプによって異なります。処理されたオブジェクトのタイプは、[List Library Objects] や [List Command Processors] などのように画面タイトルに示されます。

各列見出しの下にある入力フィールドを使用して、特定のオブジェクトからリストを開始したり、オブジェクトにフィルタをかけたりすることができます。有効な入力値については、以下の表を参照してください。

列	PF キー	説明
Cmd		目的のオブジェクトの横にある入力フィールドに、次のいずれかの行コマンドを入力できます。
	L または LI	オブジェクトのソースコードをリスト表示します (Natural コマンドプロセッサソースには適用されません)。  ユーザー定義エラーメッセージの場合: エラーメッセージの短文テキストおよび長文テキストがリスト表示されます。
	S または SE UL または U	以降のアンロード処理のためにオブジェクトを選択します。  注意: これらのコマンドは、以降の処理のために選択されたオブジェクトをマークするだけです。アンロード機能を実行するには、PF2 キー (Unloa) を押す必要があります (下記参照)。
	DE	オブジェクトを削除します。

列	PF キー	説明												
Name		<p>オブジェクト名です。</p> <p>単一の名前または名前の範囲を入力できます。「名前と日時の指定」の「名前」を参照してください。</p> <p>ユーザー定義エラーメッセージについては、メッセージ番号および言語コードが表示されます。例：10 (Lang =1) は、言語 1 (英語) のメッセージ番号 10 を表します。</p>												
Type		<p>Program など、Natural ライブラリオブジェクトのタイプです。</p> <p>有効な入力値は、1 つまたは複数のオブジェクトタイプコード (プログラムには P など) です。コードのリストについては、<i>select-clause</i> セクションの <b>NATTYPE</b> を参照してください。</p> <p>プロファイルまたは DL/I サブファイルタイプの Natural 関連オブジェクトについては、この列にプロファイルのタイプ (Editor など) または DL/I サブファイルのタイプ (NBD Subfile など) が表示されます。</p>												
S/C		<p>Natural ライブラリオブジェクトのタイプ。デフォルトでは、すべてのソース (S) オブジェクトまたはカタログされた (C) オブジェクトあるいはその両方が画面に表示されます。</p> <table border="1" data-bbox="414 934 1474 1402"> <tr> <td colspan="2">有効な入力値は、次の 1 つ以上のコードです。</td> </tr> <tr> <td>S</td> <td>ソースオブジェクトのみ。</td> </tr> <tr> <td>C</td> <td>カタログ化オブジェクトのみ。</td> </tr> <tr> <td>S/C</td> <td>ソースオブジェクトとカタログ化オブジェクトの両方 (両方ともある場合)。</td> </tr> <tr> <td>W</td> <td>STOW を実行された全オブジェクト：同じ日時のソースオブジェクトとカタログ化オブジェクト</td> </tr> <tr> <td>*</td> <td>ソースオブジェクトとカタログ化オブジェクトの両方または片方のすべて。</td> </tr> </table> <p>ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。</p>	有効な入力値は、次の 1 つ以上のコードです。		S	ソースオブジェクトのみ。	C	カタログ化オブジェクトのみ。	S/C	ソースオブジェクトとカタログ化オブジェクトの両方 (両方ともある場合)。	W	STOW を実行された全オブジェクト：同じ日時のソースオブジェクトとカタログ化オブジェクト	*	ソースオブジェクトとカタログ化オブジェクトの両方または片方のすべて。
有効な入力値は、次の 1 つ以上のコードです。														
S	ソースオブジェクトのみ。													
C	カタログ化オブジェクトのみ。													
S/C	ソースオブジェクトとカタログ化オブジェクトの両方 (両方ともある場合)。													
W	STOW を実行された全オブジェクト：同じ日時のソースオブジェクトとカタログ化オブジェクト													
*	ソースオブジェクトとカタログ化オブジェクトの両方または片方のすべて。													
M		<p>Natural ライブラリオブジェクトのプログラミングモード。デフォルトでは任意のモードが表示されます。</p> <p>有効な入力値は、次の 1 つ以上のコードです。</p> <p>S ストラクチャードモードのみ。</p> <p>R レポーティングモードのみ。</p> <p>* 任意のモード。ストラクチャードモードおよびレポーティングモードのいずれかあるいはその両方。</p>												

## オブジェクトの選択

列	PF キー	説明
		ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。
<b>Version</b>		<p>Natural ライブラリオブジェクトの保存またはカタログ化あるいはその両方を行った Natural のバージョン。</p> <p>V.R.S の形式でバージョンを入力できます。V は 1 桁のバージョン、R は 1 桁のリリース、S は 2 桁のシステムメンテナンスレベルです。バージョンの有効範囲については、「名前と日時の指定」の「名前」を参照してください。</p> <p>ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。</p>
<b>User ID</b>		<p>Natural ライブラリオブジェクトまたは DDM を保存またはカタログしたユーザーの ID。単一のユーザー ID またはユーザー ID の範囲を入力できます。「名前」を参照してください。</p> <p>ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。</p>
<b>Date</b>		<p>Natural ライブラリオブジェクトまたは DDM が保存またはカタログされた日付。単一の日付または日付の範囲を入力できます。「名前と日時の指定」の「日付」を参照してください。</p> <p>ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。</p>
<b>Time</b>		<p>Natural ライブラリオブジェクトまたは DDM が保存またはカタログされた時刻。単一の時刻または時刻の範囲を入力できます。「名前と日時の指定」の「時刻」を参照してください。</p> <p>ユーザー定義エラーメッセージについては、この列にエラーメッセージの短文テキストが表示されます。</p>
<b>DBID</b>		DDM または Natural 関連オブジェクトが保存されているシステムファイルのデータベース ID。
<b>FNR</b>		DDM または Natural 関連オブジェクトが保存されているシステムファイルのファイル番号。
	PF2	<p>処理を行うため、選択されたオブジェクトのアンロードを開始します。</p> <p>別の方法としては、コマンド行に次のダイレクトコマンドのいずれかを入力します。</p> <p>UNLOAD または UNLD</p>
	PF11	<p>PF2 キーによる後続のアンロード処理のために、リスト表示されたすべてのオブジェクトをマークします。</p> <p>別の方法としては、コマンド行に次のダイレクトコマンドのいずれかを入力します。</p> <p>SELECT ALL</p>



---

列	PF キー	説明
		または SEL ALL

---

# 27 オブジェクト指定

---

オブジェクトハンドラには [Select Unload/Load/Scan Type] 画面があり、この画面で、処理するオブジェクトのタイプを選択するか、タイプ SELECTION または LIST のワークプランを指定できます。

選択したオブジェクトのタイプごとに、個々のオブジェクト指定画面が表示されます。これらの画面を使用して、処理するオブジェクトの選択条件を指定します。

このセクションでは、各オブジェクト指定画面に表示されるオプションについて説明します。このセクションで説明するフィールドまたはファンクションキー（PF キー）が特定の機能や上級ユーザーモードでしか表示されない場合は、「上級ユーザーモードのアンロード機能にのみ適用されます。」などの適切な注釈によって示されます。

- ワークファイルのすべてのオブジェクト
- **Natural** ライブラリオブジェクト
- **Natural** システムエラーメッセージ
- **Natural** コマンドプロセッサソース
- **Natural** 関連オブジェクト
- **DDM**
- **FDT**
- **SELECTION** または **LIST** ワークプランの使用



# 28 オブジェクト指定・ワークファイルのすべてのオブジェクト

---

ロード機能またはスキャン機能にのみ適用されます。

【**Load/Scan All Objects on the Work File**】オプションは、ワークファイルで処理可能なすべてのオブジェクトを選択するために使用します。上級ユーザーモードでは、【**Load/Scan All Objects**】画面から【**Settings**】画面を表示して、この画面でオプションの指定およびパラメータの設定を行うことができます。「[設定](#)」セクションを参照してください。



# 29 オブジェクト指定 - Natural ライブラリオブジェクト

---

▪ Natural ライブラリオブジェクト .....	146
▪ Natural ライブラリオブジェクトの詳細 .....	147
▪ Natural ライブラリオブジェクトのプロパティ .....	150
▪ Natural ライブラリオブジェクトの例外 .....	151
▪ Natural ライブラリオブジェクトの例外のプロパティ .....	152

このセクションでは、Natural ライブラリオブジェクトを処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。Natural ライブラリオブジェクトとは、プログラミングオブジェクトおよびユーザー定義エラーメッセージのことです。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

## Natural ライブラリオブジェクト

「[Unload/Load/Scan Natural Library Objects]」画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
DBID/FNR		アンロード機能にのみ適用されます。  Natural ライブラリが保存されているシステムファイルのデータベース ID (DBID) とファイル番号 (FNR)。値が指定されていない場合、または 0 が指定されている場合、現在の FUSER または FNAT システムファイルが使用されます。
Select DBID/FNR	PF5  (上級ユーザーモード専用)	アンロード機能にのみ適用されます。  使用可能なシステムファイルのリストを表示します。
Password/Cipher		アンロード機能にのみ適用されます。  Natural ライブラリが保存されている Adabas ファイルのパスワードとサイファコード。
Library		ライブラリの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select library	PF4	使用可能なすべてのライブラリの選択リストが表示されます。「 <a href="#">ライブラリの選択</a> 」も参照してください。
Object name		Natural プログラミングオブジェクトの名前、あるいは名前の範囲。「 <a href="#">名前</a> 」を参照してください。  「[Natural Library Objects, Details]」画面で「[Natural programming objects]」フィールド (デフォルト設定) を選択した場合にのみ評価されます。「 <a href="#">Natural ライブラリオブジェクトの詳細</a> 」も参照してください。
Select objects		上級ユーザーモードのアンロード機能にのみ適用されます。  ライブラリの範囲が指定されていない場合、使用可能なすべての Natural オブジェクトの選択リストが表示されます (« <a href="#">オブジェクトの選択</a> 」も参照)。



フィールド	PF キー	説明
Error number from/to		最初と最後のメッセージ番号で定められたユーザー定義エラーメッセージの有効な範囲（1～9999）。  [Natural Library Objects, Details] 画面で [Error messages] フィールド（デフォルト設定）を選択した場合にのみ評価されます（「 <a href="#">Natural ライブラリオブジェクトの詳細</a> 」も参照）。
Details	PF6	[Natural Library Objects, Details] 画面が表示され、この画面でオブジェクトの詳細指定を入力できます。「 <a href="#">Natural ライブラリオブジェクトの詳細</a> 」を参照してください。
Settings	PF7	上級ユーザーモードで実行されている機能にのみ適用されます。  [Unload/Load/Scan Settings] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。

## Natural ライブラリオブジェクトの詳細

[Unload/Load/Scan Natural Library Objects, Details] 画面は、Natural ライブラリオブジェクトの選択条件を詳細指定するために使用します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[Unload/Load/Scan Natural Library Objects, Details] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
Library		ライブラリの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。  [Use Predict set] オプションを選択した場合、範囲指定はできません。
Select (ライブラリ)	PF4	使用可能なすべてのライブラリの選択リストが表示されます。「 <a href="#">ライブラリの選択</a> 」も参照してください。
DBID/FNR		上記の「Natural ライブラリオブジェクト」の「DBID/FNR」の項を参照してください。
Passw./Ciph.		アンロード機能にのみ適用されます。  Natural ライブラリが保存されている Adabas ファイルのパスワードとサイファコード。
Object Types:		Natural プログラミングオブジェクト。

オブジェクト指定 - Natural ライブラリオブジェクト

フィールド	PF キー	説明						
Natural programming objects								
Object Types: Error messages		ユーザー定義エラーメッセージ。						
オブジェクト名		上記の「Natural ライブラリオブジェクト」の「Object name」の項を参照してください。						
Use Predict set		<p>Predictがインストールされている場合に、アンロード機能および検索機能にのみ適用されます。</p> <p>このオプションは、処理するオブジェクトの名前を保有セットから読み取るために使用します。保有セットは、LIST XREF コマンドのセット保存オプションで作成されます。</p> <p>[Use Predict set] オプションを選択すると、次の制限が適用されます。</p> <ul style="list-style-type: none"> <li>■ [Object name] フィールドに、すべてのオブジェクトを示すアスタリスク (*) が含まれている必要があります。これはデフォルト設定です。</li> <li>■ [Library] フィールドに、1つのライブラリの名前が含まれている必要があります。名前の範囲は使用できません。</li> <li>■ [Set number] フィールドに、値が入力されている必要があります。</li> </ul> <p>Predict セットの詳細については、『Predict』ドキュメントを参照してください。</p>						
Set number		適用されるのは、[Use Predict set] が選択されている場合に限られます。 使用する保有セットを識別する 1 桁または 2 桁の数字。						
Set library		適用されるのは、[Use Predict set] が選択されている場合に限られます。 Predict セットに使用するために検索するライブラリの名前。名前を指定しない場合、[Library] フィールドに入力したライブラリがデフォルトで使用されます。						
Set user		適用されるのは、[Use Predict set] が選択されている場合に限られます。 保有セットを作成したユーザーの ID。ID が入力されていない場合、システム変数 *USER が指定された ID が使用されます（『システム変数』ドキュメントを参照）。						
Programming Object Options: S/C-Kind		<p>Natural プログラミングオブジェクトの種類：</p> <table border="1"> <tr> <td></td> <td></td> </tr> <tr> <td>S</td> <td>ソースオブジェクトのみ。</td> </tr> <tr> <td>C</td> <td>カタログ化オブジェクトのみ。</td> </tr> </table>			S	ソースオブジェクトのみ。	C	カタログ化オブジェクトのみ。
S	ソースオブジェクトのみ。							
C	カタログ化オブジェクトのみ。							

フィールド	PF キー	説明	
		A または *	ソースオブジェクトとカタログ化オブジェクトの両方または片方のすべて。これはデフォルト設定です。
		W	STOW を実行された全オブジェクト：同じ日時のソースオブジェクトとカタログ化オブジェクト
		B	ソースオブジェクトとカタログ化オブジェクトの両方（両方ともある場合）。
			<b>注意:</b> W と B は、アンロード機能でのみ有効です。W と B をロード機能またはスキャン機能で入力することもできますが、A として処理されます。
Programming Object Options: <b>Natural types</b>		Natural オブジェクトタイプのコード（プログラムには P など）。有効なコードのリストについては、 <i>select-clause</i> セクションの <b>NATTYPE</b> を参照してください。	
<b>Select Natural types</b>	PF6	1 つまたは複数の Natural オブジェクトタイプを選択できるウィンドウが表示されます。	
<b>Properties</b>	PF7	Natural プログラミングオブジェクトの追加プロパティを指定できる別の画面が表示されます。「 <a href="#">Natural ライブラリオブジェクトのプロパティ</a> 」を参照してください。	
Error Messages: <b>Error number from/to</b>		[ <b>Error number from/to</b> ] フィールドに入力したユーザー定義エラーメッセージの範囲（上記の「 <a href="#">Natural ライブラリオブジェクト</a> 」を参照）。	
Error Messages: <b>Language codes</b>		指定されたエラーメッセージの有効な言語コード（英語の場合はコード 1）。最大 8 つまで指定可能。  アスタリスク (*) にすると、すべての言語コードが選択されます。	
Error Messages: <b>S/L-Kind</b>		エラーメッセージのテキストの種類：	
		S	短文テキスト。
		L	長文テキスト。
		A	短文テキストと長文テキストの両方または片方。これがデフォルトです。
		B	両方とも存在する場合は短文テキストと長文テキスト（アンロード機能のみ）。
<b>Exceptions</b>	PF8	Natural プログラミングオブジェクトの選択の例外を指定できる別の画面が表示されます。「 <a href="#">Natural ライブラリオブジェクトの例外</a> 」を参照してください。	

## Natural ライブラリオブジェクトのプロパティ

[Unload/Load/Scan Library Objects, Properties] 画面は、処理を行うために選択した Natural ライブラリオブジェクトのプロパティを指定するために使用します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[Unload/Load/Scan Library Objects, Properties] 画面には、次のフィールドが表示されます。

フィールド	説明								
User ID	Natural プログラミングオブジェクトを保存またはカタログしたユーザーの ID。単一のユーザー ID またはユーザー ID の範囲を指定します。「名前と日時の指定」の「名前」を参照してください。								
Programming mode	Natural プログラミングオブジェクトのプログラミングモード： <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"></td> <td style="width: 50%;"></td> </tr> <tr> <td>R</td> <td>レポートモードのみ。</td> </tr> <tr> <td>S</td> <td>ストラクチャードモードのみ。</td> </tr> <tr> <td>A</td> <td>モードチェックは実行されません。これはデフォルト設定です。</td> </tr> </table>			R	レポートモードのみ。	S	ストラクチャードモードのみ。	A	モードチェックは実行されません。これはデフォルト設定です。
R	レポートモードのみ。								
S	ストラクチャードモードのみ。								
A	モードチェックは実行されません。これはデフォルト設定です。								
Natural version	Natural プログラミングオブジェクトの Natural バージョン。  有効なバージョンフォーマット：VRS（ここで、V は 1 桁のバージョン、R は 1 桁のリリース、S は 2 桁のシステムメンテナンスレベルです）  バージョンの範囲を指定することもできます。「名前」を参照してください。								
Object Date: Select all objects (no date check)	日付に関係なく、すべてのオブジェクトを選択します。								
Object Date: Select objects modified between/and	一定の範囲内にある保存またはカタログした日付または時刻（あるいはその両方）のすべてのオブジェクトを選択します。範囲を指定するには、該当するフィールドで開始日付または開始時刻（あるいはその両方）と終了日付または終了時刻（あるいはその両方）を指定します。  有効な入力値については、「名前と日時の指定」の「日付」および「時刻」を参照してください。日付には、次の特殊な入力値を使用できます：TODAY、YESTERDAY、MONTH、および YEAR。								
Object Date: Select objects modified on	正確な日付または時刻（あるいはその両方）を入力することによってこのフィールドで指定された日時に一致する、保存またはカタログした日付または時刻（あるいはその両方）のすべてのオブジェクトを選択します。  有効な入力値については、「日付」および「時刻」を参照してください。日付には、次の特殊な入力値を使用できます：TODAY および YESTERDAY。								

フィールド	説明
Object Size: <b>Select all objects (no size check)</b>	サイズに関係なく、すべてのオブジェクトを選択します。
Object Size: <b>Select objects with size between/and</b>	一定の範囲内にあるサイズのすべてのオブジェクトを選択します。範囲を指定するには、該当するフィールドで下限のサイズと上限のサイズを指定します。
Object Size: <b>Select objects with size</b>	このフィールドで指定されたサイズに一致するすべてのオブジェクトを選択します。

## Natural ライブラリオブジェクトの例外

「**Unload/Load/Scan Natural Library Objects, Exceptions**」画面は、Natural ライブラリオブジェクトの選択の例外を指定するために使用します。

「[Natural ライブラリオブジェクト](#)」、「[Natural ライブラリオブジェクトの詳細](#)」、および「[Natural ライブラリオブジェクトのプロパティ](#)」で指定した選択条件に一致するすべてのオブジェクトは、「**Unload/Load/Scan Library Objects, Exceptions**」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するオブジェクトは、処理から除外されます。

キーワードと有効な入力値の詳細については、「**ダイレクトコマンド**」セクションの *select-[clause](#)* に関する記載も参照してください。

「**Unload/Load/Scan Natural Library Objects, Exceptions**」画面は、「**Unload/Load/Scan Natural Library Objects, Details**」画面と基本的に同じです。下の表に示すフィールド、コマンド、および代替 PF キーの該当するセクションの説明を参照してください。「**Add/change properties for selection**」フィールドは、Natural プログラミングオブジェクトの例外に関して追加プロパティを指定するために使用します。「[Natural ライブラリオブジェクトの例外のプロパティ](#)」を参照してください。

フィールド	PF キー
<b>Library</b>	PF4
<b>Select</b>  (ライブラリ)	
Object Types: <b>Natural programming objects</b> <b>Error messages</b>	
<b>Object name</b>	

フィールド	PF キー
S/C-Kind	
Natural types	
Select Natural types	PF6
Properties	PF7
Error number	
S/L-Kind	
Languages	

## Natural ライブラリオブジェクトの例外のプロパティ

[Unload/Load/Scan Library Objects, Exceptions] 画面は、処理を行うために選択した Natural ライブラリオブジェクトのプロパティの例外を指定するために使用します。

画面には、次のフィールドが表示されます。

フィールド	説明
User ID	「Natural ライブラリオブジェクトのプロパティ」の「 <b>User ID</b> 」の項を参照してください。
Programming mode	「Natural ライブラリオブジェクトのプロパティ」の「 <b>Programming mode</b> 」の項を参照してください。
Natural version	「Natural ライブラリオブジェクトのプロパティ」の「 <b>Natural version</b> 」の項を参照してください。
Object Date: Ignore object date	日付チェックは実行されません。日付に関係なく、オブジェクトは処理されます。
Object Date: Exclude objects modified between/and	一定の範囲内にある保存またはカタログした日付または時刻（あるいは両方）のすべてのオブジェクトを処理から除外します。範囲を指定するには、該当するフィールドで開始日付または開始時刻（あるいは両方）と終了日付または終了時刻（あるいは両方）を指定します。  有効な入力値については、「名前と日時の指定」の「 <b>日付</b> 」および「 <b>時刻</b> 」を参照してください。日付には、次の特殊な入力値を使用できます：TODAY、YESTERDAY、MONTH、および YEAR。
Object Date: Exclude objects modified on	正確な日付または時刻（あるいはその両方）を入力することによってこのフィールドで指定された日時に一致する、保存またはカタログした日付または時刻（あるいはその両方）のすべてのオブジェクトを処理から除外します。  有効な入力値については、「 <b>日付</b> 」および「 <b>時刻</b> 」を参照してください。日付には、次の特殊な入力値を使用できます：TODAY および YESTERDAY。
Object Size:	サイズチェックは実行されません。サイズに関係なく、オブジェクトは処理されます。

フィールド	説明
<b>Ignore object size</b>	
Object Size: <b>Exclude objects with size between/and</b>	一定の範囲内にあるサイズのすべてのオブジェクトを処理から除外します。範囲を指定するには、該当するフィールドで下限のサイズと上限のサイズを指定します。
Object Size: <b>Exclude objects with size</b>	このフィールドで指定されたサイズに一致するすべてのオブジェクトを処理から除外します。





# 30 オブジェクト指定 - Natural システムエラーメッセージ

---

▪ Natural システムエラーメッセージ .....	156
▪ Natural システムエラーメッセージの詳細 .....	156
▪ Natural システムエラーメッセージの例外 .....	157

このセクションでは、指定のシステムファイルから Natural システムエラーメッセージを処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

## Natural システムエラーメッセージ

[Unload/Load/Scan Natural System Error Messages] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
Error number from/to		最初と最後のメッセージ番号で定められた Natural システムエラーメッセージの範囲。使用可能なすべてのシステムエラーメッセージを表示するには、[Select system error messages] を選択します。
Details	PF6	[Unload/Load/Scan Natural Library Objects, Details] 画面が表示され、この画面でオブジェクトの詳細指定を入力できます。「Natural システムエラーメッセージの詳細」を参照してください。
Settings	PF7	[Unload/Load/Scan Settings] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「設定」を参照してください。

## Natural システムエラーメッセージの詳細

[Unload/Load/Scan System Error Messages, Details] 画面は、Natural システムエラーメッセージの選択条件を詳細指定するために使用します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[Unload/Load/Scan System Error Messages, Details] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
DBID/FNR		アンロード機能にのみ適用されます。  Natural エラーメッセージが保存されている Adabas (FNR) ファイルのデータベース ID (DBID) とファイル番号。
Password/Cipher		アンロード機能にのみ適用されます。  Natural エラーメッセージソースが保存されている Adabas ファイルのパスワードとサイファコード。

フィールド	PFキー	説明
Error number from/to		上記の「Natural システムエラーメッセージ」の「Error number」の項を参照してください。
Language codes		「Natural ライブラリオブジェクトの詳細」の「Language codes」の項を参照してください。
S/L-Kind		「Natural ライブラリオブジェクトの詳細」の「S/L-Kind」の項を参照してください。
Exceptions	PF8	Natural システムエラーメッセージの選択の例外を指定できる別の画面が表示されます。「Natural システムエラーメッセージの例外」を参照してください。

## Natural システムエラーメッセージの例外

「Unload/Load/Scan System Error Messages, Exceptions」画面は、Natural システムエラーメッセージの選択の例外を指定するために使用します。

「Natural システムエラーメッセージ」および「Natural システムエラーメッセージの詳細」で指定した選択条件に一致するすべての Natural システムエラーメッセージは、「Unload/Load/Scan System Error Messages, Exceptions」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するエラーメッセージは、処理から除外されます。

例外画面に表示されるフィールドの説明については、上記の「Natural システムエラーメッセージの詳細」を参照してください。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。



# 31 オブジェクト指定 - Natural コマンドプロセッサ

---

▪ Natural コマンドプロセッサ .....	160
▪ Natural コマンドプロセッサソースの例外 .....	161

このセクションでは、Natural コマンドプロセッサソースを処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

## Natural コマンドプロセッサ

[**Unload/Load/Scan Natural Command Processors**] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
Library		Natural コマンドプロセッサライブラリの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select library	PF4	使用可能な Natural コマンドプロセッサライブラリの選択リストを表示します。「 <a href="#">ライブラリの選択</a> 」も参照してください。
DBID/FNR		アンロード機能にのみ適用されます。  Natural コマンドプロセッサソースが保存されている Adabas (FNR) ファイルのデータベース ID (DBID) とファイル番号。
Password/Cipher		アンロード機能にのみ適用されます。  Natural コマンドプロセッサソースが保存されている Adabas ファイルのパスワードとサイファコード。
Object name		Natural コマンドプロセッサソースの名前または名前の範囲。「名前」を参照してください。
Select objects		アンロード機能にのみ適用されます。  ライブラリの範囲が指定されていない場合にこのフィールドを選択すると、使用可能な Natural コマンドプロセッサソースの選択リストが表示されます（「 <a href="#">オブジェクトの選択</a> 」も参照）。
Exceptions	PF8	Natural コマンドプロセッサソースの選択の例外を指定できる別の画面が表示されます。「 <a href="#">Natural コマンドプロセッサソースの例外</a> 」を参照してください。
Settings	PF7	[ <b>Unload/Load/Scan Settings</b> ] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。

## Natural コマンドプロセッサソースの例外

---

「**Unload/Load/Scan Natural Command Processors, Exceptions**」画面は、Natural コマンドプロセッサソースの選択の例外を指定するために使用します。

「**Natural コマンドプロセッサソース**」で指定した選択条件に一致するすべてのオブジェクトは、「**Unload/Load/Scan Natural Command Processors, Exceptions**」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致する Natural コマンドプロセッサソースは、処理から除外されます。

例外ウィンドウに表示されるフィールドの説明については、上記の「**Natural コマンドプロセッサソース**」を参照してください。

キーワードと有効な入力値の詳細については、「**ダイレクトコマンド**」セクションの *select-clause* に関する記載も参照してください。

---



## 32 オブジェクト指定 - Natural 関連オブジェクト

---

■ Natural プロファイル .....	164
■ Natural デバッグ環境 .....	166
■ Natural DL/I サブファイル .....	167

このセクションでは、Natural 関連オブジェクトを処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。Natural 関連オブジェクトは、プロファイル、デバッグ環境、および DL/I サブファイルです。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

「[Unload/Load Type]」画面で Natural 関連オブジェクトを選択すると、「[Select Related Type]」画面が表示され、この画面で Natural 関連オブジェクトのタイプ（Natural プロファイル、デバッグ環境、または DL/I サブファイル）を指定できます。

## Natural プロファイル

「[Unload/Load/Scan Natural Profiles]」画面には、次のフィールドおよび PF キーが表示されません。

フィールド	PF キー	説明
DBID/FNR		アンロード機能にのみ適用されます。  Natural プロファイルが保存されている Adabas ファイルのデータベース ID (DBID) とファイル番号 (FNR)。  値が指定されていない場合、または「0」が指定されている場合、現在の FNAT システムファイルが使用されます。
Password/Cipher		アンロード機能にのみ適用されます。  Natural プロファイルが保存されている Adabas ファイルのパスワードとサイファコード。
Select (DBID/FNR)	PF5	アンロード機能にのみ適用されます。  「[Select System File]」ウィンドウが開き、Natural 環境で使用可能なすべてのシステムファイルがリスト表示されます。「システムファイルの選択」も参照してください。
Profile Types		処理するプロファイルのタイプ：  パラメータ、エディタ、マップ、および/またはデバイス。
Object name		プロファイルの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select (Object name)		アンロード機能にのみ適用されます。  使用可能なプロファイルの選択リストを示す「[List Profiles]」画面が表示されます（「オブジェクトの選択」も参照）。

フィールド	PF キー	説明
Exceptions	PF8	プロファイルの選択の例外を指定できる別の画面が表示されます。「 <a href="#">Natural プロファイルの例外</a> 」を参照してください。
Settings	PF7	「 <a href="#">Unload/Load/Scan Settings</a> 」画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。

キーワードと有効な入力値の詳細については、「[ダイレクトコマンド](#)」セクションの *select-clause* に関する記載も参照してください。

#### ■ Natural プロファイルの例外

### Natural プロファイルの例外

「[Unload/Load/Scan Natural Profiles, Exceptions](#)」画面は、Natural プロファイルの選択の例外を指定するために使用します。

「[Natural プロファイル](#)」で指定した選択条件に一致するすべてのオブジェクトは、「[Unload/Load/Scan Natural Profiles, Exceptions](#)」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するオブジェクトは、処理から除外されます。

キーワードと有効な入力値の詳細については、「[ダイレクトコマンド](#)」セクションの *select-clause* に関する記載も参照してください。

「[Unload/Load/Scan Natural Profiles, Exceptions](#)」画面には、次のフィールドが表示されます。

フィールド	説明
Object name	プロファイルの名前または名前の範囲。「 <a href="#">名前と日時の指定</a> 」の「 <a href="#">名前</a> 」を参照してください。
Profile Types	処理するプロファイルのタイプ： パラメータ、エディタ、マップ、および/またはデバイス。

## Natural デバッグ環境

[Unload/Load/Scan Natural Debug Environments] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
Library		ライブラリの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select library	PF4	使用可能なすべてのライブラリの選択リストが表示されます。「 <a href="#">ライブラリの選択</a> 」も参照してください。
DBID/FNR		アンロード機能にのみ適用されます。  Natural デバッグ環境が保存されている Adabas ファイルのデータベース ID (DBID) とファイル番号 (FNR)。  値が指定されていない場合、または「0」が指定されている場合、現在の FNAT システムファイルが使用されます。
Password/Cipher		アンロード機能にのみ適用されます。  Natural デバッグ環境が保存されている Adabas ファイルのパスワードとサイファコード。
Object name		デバッグ環境の名前または名前の範囲。「名前」を参照してください。
Select objects		アンロード機能にのみ適用されます。  使用可能なデバッグ環境の選択リストが表示されます（「 <a href="#">オブジェクトの選択</a> 」も参照）。
Exceptions	PF8	デバッグ環境の選択の例外を指定できる別の画面が表示されます。「 <a href="#">Natural デバッグ環境の例外</a> 」を参照してください。
Settings	PF7	[Unload/Load/Scan Settings] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの [select-clause](#) に関する記載も参照してください。

■ Natural デバッグ環境の例外

Natural デバッグ環境の例外

「**Unload/Load/Scan Debug Environments, Exceptions**」画面は、Natural デバッグ環境の選択の例外を指定するために使用します。

「**Natural デバッグ環境**」で指定した選択条件に一致するすべてのオブジェクトは、「**Unload/Load/Scan Debug Environments, Exceptions**」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するオブジェクトは、処理から除外されます。

キーワードと有効な入力値の詳細については、「**ダイレクトコマンド**」セクションの *select-clause* に関する記載も参照してください。

「**Unload/Load/Scan Debug Environments, Exceptions**」画面には、次のフィールドが表示されます。

フィールド	説明
Library	ライブラリの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select (ライブラリ)	使用可能なすべてのライブラリの選択リストが表示されます。「ライブラリの選択」も参照してください。
Object name	デバッグ環境の名前または名前の範囲。「名前」を参照してください。

Natural DL/I サブファイル

「**Unload/Load/Scan Natural DL/I Subfiles**」画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
DBID/FNR		アンロード機能にのみ適用されます。  Natural DL/I サブファイルが保存されている Adabas ファイルのデータベース ID (DBID) とファイル番号 (FNR)。  値が指定されていない場合、または「0」が指定されている場合、現在の FDIC システムファイルが使用されます。

フィールド	PF キー	説明
Password/Cipher		アンロード機能にのみ適用されます。  Natural DL/I サブファイルが保存されている Adabas ファイルのパスワードとサイファコード。
Select  (DBID/FNR)	PF5	アンロード機能にのみ適用されます。  [Select System File] ウィンドウが開き、Natural 環境で使用可能なすべてのシステムファイルがリスト表示されます。「システムファイルの選択」も参照してください。
Subfile Types		処理する DL/I サブファイルのタイプ：NSB および/または NDB。
Object name		DL/I サブファイルの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select  (Object name)		アンロード機能にのみ適用されます。  使用可能な DL/I サブファイルの選択リストが表示されます（「オブジェクトの選択」も参照）。
Exceptions	PF8	DL/I サブファイルの選択の例外を指定できる他の画面が表示されます。「Natural DL/I サブファイルの例外」を参照してください。
Settings	PF7	[Unload/Load/Scan Settings] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「設定」を参照してください。

■ Natural DL/I サブファイルの例外

Natural DL/I サブファイルの例外

[Unload/Load/Scan Natural DL/I Subfiles, Exceptions] 画面は、Natural DL/I サブファイルの選択の例外を指定するために使用します。

「Natural DL/I サブファイル」で指定した選択条件に一致するすべてのオブジェクトは、[Unload/Load/Scan Natural DL/I Subfiles, Exceptions] 画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するオブジェクトは、処理から除外されます。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[Unload/Load/Scan Natural DL/I Subfiles, Exceptions] 画面には、次のフィールドが表示されます。

フィールド	説明
Object name	DL/I サブファイルの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Subfile Types	処理する DL/I サブファイルのタイプ：NSB および／または NDB。

---



# 33 オブジェクト指定 - DDM

---

- DDM ..... 172
- DDM のプロパティ ..... 173
- DDM の例外 ..... 174

このセクションでは、Natural DDM（データ定義モジュール）を処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

## DDM

[Unload/Load/Scan DDMs] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
FDIC DBID/FNR		アンロード機能にのみ適用されます。  DDMが保存されている Adabas ファイルのデータベースID (DBID) とファイル番号 (FNR)。  値が指定されていない場合、または「0」が指定されている場合、現在の FDIC システムファイルが使用されます。
FDIC Password/Cipher		アンロード機能にのみ適用されます。  DDM が保存されている Adabas ファイルのパスワードとサイファコード。
DDM name		アンロード機能にのみ適用されます。  DDMの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
Select objects		アンロード機能にのみ適用されます。  使用可能なDDMの選択リストが表示されます（「オブジェクトの選択」も参照）。
Properties	PF7	DDM の追加プロパティを指定できる別の画面が表示されます。「 <a href="#">DDM のプロパティ</a> 」を参照してください。
Exceptions	PF8	DDMの選択の例外を指定できる別の画面が表示されます。「 <a href="#">DDMの例外</a> 」を参照してください。
Settings	PF7	[ <a href="#">Unload/Load/Scan Settings</a> ] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。

## DDM のプロパティ

[**Unload/Load/Scan DDMs, Properties**] 画面は、処理を行うために選択した DDM のプロパティを指定するために使用します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[**Unload/Load/Scan DDMs, Properties**] 画面には、次のフィールドが表示されます。

フィールド	説明
<b>User ID</b>	DDM を保存またはカタログしたユーザーの ID。  単一のユーザー ID またはユーザー ID の範囲を指定します。「名前と日時の指定」の「名前」を参照してください。
<b>DDM DBID</b>	DDM のデータベース ID (DBID)。  有効なエントリ：1~65535、または 0 (すべての DBID)
<b>DDM FNR</b>	DDM のファイル番号 (FNR) :  有効なエントリ：1~65535、または 0 (すべての FNR)
Object Date: <b>Select all objects (no date check)</b>	日付に関係なく、すべての DDM を選択します。
Object Date: <b>Select objects modified between/and</b>	「Natural ライブラリオブジェクトのプロパティ」の「Object Date」の項を参照してください。
Object Date: <b>Select objects modified on</b>	「Natural ライブラリオブジェクトのプロパティ」の「Object Date」の項を参照してください。
Object Size: <b>Select all objects (no size check)</b>	サイズに関係なく、すべての DDM を選択します。
Object Size: <b>Select objects with size between/and</b>	一定の範囲内にあるサイズのすべての DDM を選択します。範囲を指定するには、該当するフィールドで下限のサイズと上限のサイズを指定します。
Object Size: <b>Select objects with size</b>	このフィールドで指定されたサイズに一致するすべての DDM を選択します。

## DDM の例外

「**Unload/Load/Scan DDMs, Exceptions**」画面は、DDMの選択の例外を指定するために使用します。

「**DDM**」および「**DDMのプロパティ**」で指定した選択条件に一致するすべてのオブジェクトは、「**Unload/Load/Scan DDM, Exceptions**」画面で行った指定に対してチェックされます。例外として定義されたすべての指定項目に一致するオブジェクトは、処理から除外されます。

キーワードと有効な入力値の詳細については、「**ダイレクトコマンド**」セクションの *select-clause* に関する記載も参照してください。

「**Unload/Load/Scan DDMs, Exceptions**」画面には、次のフィールドが表示されます。

フィールド	説明
DDM name	DDMの名前または名前の範囲。「名前と日時の指定」の「名前」を参照してください。
DDM DBID	「DDMのプロパティ」の「 <b>DDM DBID</b> 」の項を参照してください。
DDM FNR	「DDMのプロパティ」の「 <b>DDM FNR</b> 」の項を参照してください。
User ID	「DDMのプロパティ」の「 <b>User ID</b> 」の項を参照してください。
Object Date: Ignore object date	日付チェックは実行されません。日付に関係なく、DDMは処理されます。
Object Date: Exclude objects modified between/and	「Natural ライブラリオブジェクトの例外のプロパティ」セクションの「 <b>Object Date</b> 」の項を参照してください。
Object Date: Exclude objects modified on	「Natural ライブラリオブジェクトの例外のプロパティ」セクションの「 <b>Object Date</b> 」の項を参照してください。
Object Size: Ignore object size	サイズチェックは実行されません。サイズ日付に関係なく、DDMは処理されます。
Object Size: Exclude objects with size between/and	一定の範囲内にあるサイズのすべてのDDMを処理から除外します。範囲を指定するには、該当するフィールドで下限のサイズと上限のサイズを指定します。
Object Size: Exclude objects with size	このフィールドで指定されたサイズに一致するすべてのDDMを処理から除外します。

# 34 オブジェクト指定 - FDT

このセクションでは、Adabas FDT（フィールド定義テーブル）を処理するために使用する、オブジェクト指定画面に表示されるオプションについて説明します。

キーワードと有効な入力値の詳細については、「ダイレクトコマンド」セクションの *select-clause* に関する記載も参照してください。

[Unload/Load/Scan FDTs] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
DBID		FDT が保存されているデータベースの ID。  ロードとスキャン： 有効な DBID、またはすべての DBID の場合は 0。
FNR		FDT が保存されているファイルの番号。  ロードとスキャン： 有効な FNR、またはすべての FDT の場合は 0。
Password/Cipher		アンロードおよびロード機能にのみ適用されます。  FDT が保存されている Adabas ファイルの Adabas パスワードとサイファコード。
Settings	PF7	[Unload/Load/Scan Settings] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。



# 35 SELECTION または LIST ワークプランの使用

このオプションは、タイプ SELECTION または LIST のワークプランを指定するために使用します。これらのワークプランが、処理するオブジェクトの選択条件を指定します。「[ワークプラン](#)」セクションも参照してください。

[**Unload/Load/Scan Selection or List**] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
<b>Name</b>	PF4	処理するワークプランの名前。
<b>List Workplan</b>		[ <b>Name</b> ] フィールドに指定されたワークプランの内容を表示します。
<b>Select Workplan</b>	PF5	使用可能なすべてのワークプランのリストを表示します。「 <a href="#">管理</a> 」「 <a href="#">ワークプランライブラリ内の利用可能なワークプランのリストの作成</a> 」も参照してください。
<b>Settings</b>	PF7	[ <b>Unload/Load/Scan Settings</b> ] 画面が表示され、この画面でオプションとパラメータ設定を指定できます。「 <a href="#">設定</a> 」を参照してください。





# 36 Settings

---

- 設定画面のフィールド ..... 180
- その他のオプションの設定 ..... 182
- グローバルパラメータの設定 ..... 188

設定オプションは、アンロード／ロード／検索／スキャン機能のオプション設定、またはアンロード／ロード機能のパラメータ設定を指定するために使用します。

### ▶手順 36.1. [Unload/Load/Scan Settings] 画面を表示するには

- アンロード、ロード、またはスキャンの任意の画面で、次の内部コマンドを入力します。

SETTINGS

「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」も参照してください。

または:

上級ユーザーモードをアクティブにし、機能を選択し Enter キーを押して、処理プロセスを開始します。

または:

上級ユーザー画面で PF7 (Setti) キーを押します。

デフォルトで選択されていない場合に、以下で説明する [Unload/Load/Scan Settings] 画面に表示されるオプションをアクティブにするには、対応する入力フィールドを任意の単一文字でマークします。

## 設定画面のフィールド

[Unload/Load/Scan Settings] 画面には、次のフィールドおよび PF キーが表示されます。

フィールド	PF キー	説明
Transfer format		<p>有効になるのは、<a href="#">[Use default options]</a>（これがデフォルトです）を選択した場合に限られます。</p> <p>これを選択すると、ワークファイルにおける処理データの書き込み／読み取りが転送フォーマットで実行されます。「ワークファイル」の「<a href="#">ワークファイルフォーマット</a>」も参照してください。</p> <p>アンロード機能： アンロードするデータは、転送フォーマットでワークファイルに書き込まれます。以降のアンロードではこのフィールドの設定を変更する場合は、メインメニューに戻るか、コマンド GO UNLOAD END（「ダイレクトコマンド」の「<a href="#">ナビゲーションと特別な機能のためのコマンド</a>」を参照）を入力してアンロード機能を再スタートする必要があります。</p> <p>ロードおよびスキャン機能：</p>

フィールド	PF キー	説明
		ロードまたはスキャンするデータは転送フォーマットにする必要があります。
Unicode work file		<p>アンロード機能にのみ適用され、有効になるのは [Transfer format] が選択されている場合に限られます。</p> <p>このオプションが選択されている場合、すべてのオブジェクトソースが、ワークファイルに書き込まれる前に Unicode/UTF-8 (Universal Transformation Format、8 ビット形式) に変換されます。</p> <p>Unicode ワークファイルが指定されている場合、転送オプション [Use conversion table]、[Substitute line references]、および [Incorporate free rules] は使用できません。</p>
Use PC File		<p>適用されるのは、Entire Connection がインストールされている場合に限られます。</p> <p>Entire Connection ワークファイルにデータを書き込みます。</p>
PC File		<p>適用されるのは、Entire Connection がインストールされている場合に限られます。</p> <p>Entire Connection ワークファイルへの完全パス名。現在のシステム環境でバックスラッシュ (\) セパレータを使用できない場合は、代わりにスラッシュ (/) を使用します。</p>
Use default options		デフォルトオプションが使用されます (これがデフォルトです)。「プロファイルの設定」および「その他のオプションの設定」も参照してください。
Set additional options	PF4	<p>有効になるのは、[Use default options] を選択した場合に限られます。</p> <p>処理シーケンスのデフォルト設定を修正し、追加オプションを入力できる [Options] 画面が表示されます。使用可能なオプションについては、「その他のオプションの設定」を参照してください。</p>
Use Option Workplan		タイプ OPTION のワークプランが使用されます。「ワークプラン」も参照してください。
Name ( [Use Option Workplan] の横)		<p>有効になるのは、[Use Option Workplan] を選択した場合に限られます。</p> <p>使用するタイプ OPTION のワークプランの名前。</p>
List Option Workplan	PF6	<p>有効になるのは、[Use Option Workplan] を選択した場合に限られます。</p> <p>指定したワークプランの内容が [Use Option Workplan] の横にある [Name] フィールドに表示されます。</p>
Select Option Workplan	PF5	<p>有効になるのは、[Use Option Workplan] を選択した場合に限られます。</p> <p>タイプ OPTION の使用可能なワークプランの選択リストが表示されます (「管理」の「ワークプランライブラリ内の利用可能なワークプランのリストの作成」も参照)。</p>
Do not use parameters		これが選択されている場合 (デフォルト設定)、パラメータは設定されません。

フィールド	PF キー	説明
Use global parameters		グローバルパラメータが使用されます。「 <a href="#">グローバルパラメータの設定</a> 」を参照してください。
Set global parameters	PF7	有効になるのは、「 <b>Use global parameters</b> 」を選択した場合に限られます。  [Parameters] 画面が表示されます。キーワードと有効な入力値の詳細については、「 <a href="#">グローバルパラメータの設定</a> 」および <i>parameter-setting</i> （「 <a href="#">ダイレクトコマンド</a> 」）を参照してください。
Use Parameter Workplan		タイプ PARAMETER のワークプランが使用されます。「 <a href="#">ワークプラン</a> 」も参照してください。
Name  （ [Use Parameter Workplan] の横）		有効になるのは、「 <b>Use Parameter Workplan</b> 」を選択した場合に限られます。  使用するタイプ PARAMETER のワークプランの名前。
List Parameter Workplan	PF9	有効になるのは、「 <b>Use Parameter Workplan</b> 」を選択した場合に限られます。  指定したワークプランの内容が [Use Parameter Workplan] の横にある [Name] フィールドに表示されます。
Select Parameter Workplan		有効になるのは、「 <b>Use Parameter Workplan</b> 」を選択した場合に限られます。  タイプ PARAMETER の使用可能なワークプランの選択リストが表示されます。「 <a href="#">管理</a> 」 「 <a href="#">ワークプランライブラリ内の利用可能なワークプランのリストの作成</a> 」も参照してください。

## その他のオプションの設定

[Options] 画面に表示されるセクションについて、以下で説明します。すべてのセクションが画面に表示されるわけではありません。これは、使用する機能、定義された設定、およびインストールされている製品によって、セクションが異なるからです。

キーワードと有効な入力値の詳細については、「[ダイレクトコマンド](#)」セクションの *option-setting* に関する記載も参照してください。

このセクションでは、次のトピックについて説明します。

- [ワークファイルオプションとレポートオプション](#)
- [XREF オプション](#)
- [バージョンチェック](#)
- [転送オプション](#)
- [置換オプション](#)
- [処理するオブジェクト数](#)
- [FDIC 設定](#)

## ■ FSEC 設定

### ワークファイルオプションとレポートオプション

次のセクションでは、ワークファイルおよびレポート用に表示されるオプションについて説明します。

フィールド	説明
Use PC File	適用されるのは、Entire Connection がインストールされている場合に限られます。 Entire Connection ワークファイルにデータを書き込みます。
PC File	適用されるのは、Entire Connection がインストールされている場合に限られます。 Entire Connection ワークファイルに割り当てられている完全パス名。現在のシステム環境でバックスラッシュ (\) セパレータを使用できない場合は、代わりにスラッシュ (/) を使用します。
Write report	処理するオブジェクトのレポートを <b>[Report text member]</b> フィールドで指定したレポートテキストメンバに書き込みます。  <b>[Write report]</b> オプションはデフォルトで選択されます。  レポートを表示するには、内部コマンド SHOW REPORT FILE を入力します（「ダイレクトコマンド」の「ナビゲーションと特別な機能のためのコマンド」を参照）。
Start new report	有効になるのは、 <b>[Write report]</b> を選択した場合に限られます。  新しいレポートが書き込まれる前に、レポートテキストメンバの内容を削除します。この指定がないと、新しいレポートが既存のレポートに付加されます。
Report text member	有効になるのは、 <b>[Write report]</b> を選択した場合に限られます。  レポートが書き込まれる、ワークプランライブラリに保存されたテキストメンバの名前。
Select text member	ワークプランライブラリに保存された全テキストメンバのリストを表示します。このリストから、 <b>[Report text member]</b> を選択できます。
Write restart information	ロード機能にのみ適用されます。  このオプションを設定すると、ロードの再スタート機能に対して再スタート情報が提供されます。  詳細については、「機能」の「ロードの再スタート」を参照してください。
Restart text member	<b>[Write restart information]</b> が選択された場合に、ロード機能にのみ適用されます。  再スタート情報が書き込まれる、ワークプランライブラリに保存されたテキストメンバの名前。  名前を指定しないと、オブジェクトハンドラで名前が生成され、テキストメンバに割り当てられます。

フィールド	説明
Select text member	ワークプランライブラリに保存された全テキストメンバのリストを表示します。このリストから、 <b>Restart text member</b> を選択できます。

## XREF オプション

XREF オプションは、データを内部フォーマットでアンロードまたはロードする場合（つまり、**Transfer format** フィールドが選択されていない場合）にのみ使用できます。XRef データを処理するには、Predict がインストールされている必要があります。

次のセクションでは、表示される XREF オプションおよびそれが適用される機能について説明します。

フィールド	説明	機能
Yes (unload XRef data) または Yes (load XRef data)	カタログ化オブジェクトとそのクロスリファレンスデータ（存在する場合）をアンロードします。クロスリファレンスがワークファイルに存在する場合は、カタログ化オブジェクトとそのクロスリファレンスデータをロードします。	アンロード ロード
No (ignore XRef data)	XRef データは処理されません。	アンロード ロード
Force	カタログ化オブジェクトとそのクロスリファレンスデータをロードします。ただしそれは、クロスリファレンスがワークファイル内に存在し、そのオブジェクトの Predict エントリが FDIC システムファイル内に存在する場合に限られます。	ロード
Doc	カタログ化オブジェクトをロードします。ただしそれは、そのオブジェクトの Predict エントリが FDIC システムファイル内に存在する場合に限られます。	ロード
Special	カタログ化オブジェクトとそのクロスリファレンスデータ（存在する場合）をロードします。	ロード

## バージョンチェック

**Version check** オプションは、データを内部フォーマットでロードする場合（つまり、**Transfer format** フィールドが選択されていない場合）にのみ使用できます。

**Version check** を選択すると、オブジェクトをカタログしワークファイルに書き込んだ Natural バージョンが現在の Natural バージョンと比較されます。現在よりも上位のバージョンの Natural でカタログされたオブジェクトは拒否されます。

## 転送オプション

転送オプションは、データを内部フォーマットで処理している場合（つまり、**[Transfer format]** フィールドが選択されている場合）にのみ使用できます。

次のセクションでは、表示される転送オプションおよびそれが適用される機能について説明します。

オプション	説明	機能
<b>Substitute line references</b>	適用されるのは、ソースコードの行番号がステートメント参照に使用されている場合に限られます。  行番号がソースコードの参照として使用されている場合、参照先の行の行番号と行番号参照はラベルで置き換えられます。ソースはデータベースでは変更されません。	アンロード
<b>Include line numbers</b>	このオプションを選択すると、行番号が転送されます。（デフォルトでは、Natural オブジェクトの行番号は転送されません。）	アンロード
<b>Incorporate free rules</b>	Predict がインストールされている場合、マップと関連付けられた Predict ルールがマップソースに組み込まれます。	アンロード
<b>Use conversion table</b>	注意： このオプションは、異なる文字セットを使用する環境間で非 FTP 転送を実行しており、転送ツールによって変換が行われないという特殊な場合にのみ使用します。  アンロード： 内部の Natural 変換テーブル（「 <b>System table</b> 」参照）またはユーザーによって定義された変換テーブル（「 <b>User table</b> 」参照）を使用して、データを ASCII フォーマットに変換します。  ロード： 内部の Natural 変換テーブル（「 <b>System table</b> 」参照）またはユーザーによって定義された変換テーブル（「 <b>User table</b> 」参照）を使用して、データを EBCDIC フォーマットに変換します。ワークファイル内のデータが ASCII フォーマットの場合、または変換プログラム（「 <b>User table</b> 」を参照）が指定されている場合にのみ適用されることに注意してください。	アンロード ロード
<b>System table</b>	有効になるのは、 <b>[Use conversion table]</b> を選択した場合に限られます。  アンロード： 内部的な Natural 変換テーブルを使用して、データを ASCII フォーマットに変換します。  ロード： 内部的な Natural 変換テーブルを使用して、データを EBCDIC フォーマットに変換します。	アンロード ロード

オプション	説明	機能										
User table	<p>有効になるのは、 <b>[Use conversion table]</b> を選択した場合に限られます。</p> <p>変換プログラム名をフィールドに入力すると、定義した変換プログラムを使用して、データが EBCDIC または ASCII フォーマットに変換されます。個々の変換プログラムを指定するには、ライブラリ SYSOBJH またはその steplib の 1 つにプログラムが保存されている必要があります。ライブラリ SYSOBJH にあるサブプログラム例 OTNCONAE および OTNCONEA を参照してください。</p> <p>変換プログラムが指定されていない場合、デフォルトで Natural ファイル NATCONV.INI 内のこれに相当する変換テーブルがアンロード ([ISO8859_1-&gt;EBCDIC]) 機能およびロード ([EBCDIC-&gt;ISO8859_1]) 機能で使用されます。</p>	アンロード ロード										
Use load code page	<p>このオプションを選択すると、ロード機能に使用するコードページ名を入力できるウィンドウが表示されます。</p> <p>このオプションを選択すると、UTF-8 でワークファイルにアンロードされたすべてのオブジェクトソースは、ワークファイルにロードされるときに指定のコードページで変換されます。 <b>「Unicode work file」</b> の項も参照してください。</p> <p>コードページ名として <b>「*CODEPAGE」</b> を入力すると、システム変数 *CODEPAGE に割り当てられている値が使用されます (『システム変数』ドキュメントを参照)。</p> <p>コードページ名を指定しないと、ソースオブジェクトは、アンロードするときに使用したコードページで変換されます。</p> <p><b>[Use load code page]</b> が指定されている場合、オプション <b>[Use conversion table]</b> および <b>[Translate to upper case]</b> は使用できません。</p>	ロード										
Translate to upper case	ロードするソースコードを大文字に変換します。	ロード										
Data area format	<p>データエリアにのみ適用されます。</p> <table border="1" data-bbox="310 1339 1235 1879"> <tr> <td></td> <td></td> </tr> <tr> <td></td> <td>データエリアソースをアンロードまたはロードするフォーマットを指定します。使用可能な入力値は次のとおりです。</td> </tr> <tr> <td></td> <td></td> </tr> <tr> <td>N</td> <td>データエリアを新しい内部データエリアフォーマットに変換します。</td> </tr> <tr> <td>0</td> <td>データエリアを以前の内部データエリアフォーマットに変換します。1 つ以上のデータエリアソースを以前の内部データエリアフォーマットに変換できない場合、アンロードの完了時にオブジェクトハンドラが該当するメッセージを発行します。さらに、アンロード機能によって生成されたアンロードレポートの <b>[Status]</b> 列 (影響を受けたデータエリアソース名の横) に、該当する注釈が表示されます。</td> </tr> </table>				データエリアソースをアンロードまたはロードするフォーマットを指定します。使用可能な入力値は次のとおりです。			N	データエリアを新しい内部データエリアフォーマットに変換します。	0	データエリアを以前の内部データエリアフォーマットに変換します。1 つ以上のデータエリアソースを以前の内部データエリアフォーマットに変換できない場合、アンロードの完了時にオブジェクトハンドラが該当するメッセージを発行します。さらに、アンロード機能によって生成されたアンロードレポートの <b>[Status]</b> 列 (影響を受けたデータエリアソース名の横) に、該当する注釈が表示されます。	アンロード ロード
	データエリアソースをアンロードまたはロードするフォーマットを指定します。使用可能な入力値は次のとおりです。											
N	データエリアを新しい内部データエリアフォーマットに変換します。											
0	データエリアを以前の内部データエリアフォーマットに変換します。1 つ以上のデータエリアソースを以前の内部データエリアフォーマットに変換できない場合、アンロードの完了時にオブジェクトハンドラが該当するメッセージを発行します。さらに、アンロード機能によって生成されたアンロードレポートの <b>[Status]</b> 列 (影響を受けたデータエリアソース名の横) に、該当する注釈が表示されます。											



オプション	説明	機能
	* データエリアを変換しません。これがデフォルトです。	
	詳細については、『エディタ』ドキュメントの「データエリアエディタ」を参照してください。	

## 置換オプション


次に説明する置換オプションは、ロード機能にのみ適用されます。

<b>Do not replace</b>	オブジェクトを置換しません。これがデフォルトです。
<b>Replace all</b>	すべてのオブジェクトを置換します。
<b>Replace obsolete</b>	ロードファイル内のオブジェクトの日付より前の日付になっているオブジェクトを置換します。
<b>Replace except newer</b>	ロードファイル内のオブジェクトの日付より新しい日付になっているオブジェクトを除いた、すべてのオブジェクトを置換します。

## 処理するオブジェクト数

処理するオブジェクト数は、ロード機能およびスキャン機能にのみ適用されます。

[**Number to process**] フィールドに、最大5桁の値を入力します。0より大きい値が指定された場合、指定した数のオブジェクトを処理した後、ロードまたはスキャン機能は停止します。

 **注意:** カタログされた Natural オブジェクトが同じ名前のソースオブジェクトの直後に処理された場合、両者で1つのオブジェクトと見なされます。

## FDIC 設定

FDIC 設定は、XRef データの処理に使用する Predict ファイル (FDIC) を指定する (Predict がインストールされている場合にのみ適用されます) か、または DDM をロードするために使用されます。

<b>DBID</b>	FDIC ファイルが保存されているデータベースの ID。
<b>ENR</b>	FDIC ファイルが保存されているファイルの番号。
<b>Password</b>	オプション。 FDIC ファイルが保存されている Adabas ファイルの Adabas パスワード。
<b>Cipher</b>	オプション。 FDIC ファイルが保存されている Adabas ファイルのサイファコード。

## FSEC 設定

FSEC 設定は、Natural Security がインストールされている場合にのみ適用されます。

FSEC 設定は、セキュリティチェックに使用する Natural Security データファイル (FSEC) を指定するために使用されます。

<b>DBID</b>	FSEC ファイルが保存されているデータベースの ID。
<b>FNR</b>	FSEC ファイルが保存されているファイルの番号。
<b>Password</b>	オプション。 FSEC ファイルが保存されている Adabas ファイルの Adabas パスワード。
<b>Cipher</b>	オプション。 FSEC ファイルが保存されている Adabas ファイルのサイファコード。

## グローバルパラメータの設定

ロード機能またはアンロード機能にのみ適用されます。

〔Parameters〕 画面に表示されるフィールドは、ロードまたはアンロード機能で処理するオブジェクトのグローバルパラメータの設定を変更したり、ロード機能のターゲット環境を変更したりするために使用できます。例えば、選択したオブジェクトをワークファイルにアンロードするために新しい名前（または名前の範囲）を指定したり、選択したオブジェクトをワークファイルからロードするために別のライブラリを指定したりすることができます。

アンロード機能でグローバルパラメータが指定されている場合、このパラメータ設定はワークファイルに書き込まれる前のオブジェクトに影響を与えます。ロード機能でグローバルパラメータが指定されている場合、このパラメータ設定はターゲット環境に書き込まれる前のオブジェクトに影響を与えます。

パラメータ設定を変更するために指定できる値は、目的のパラメータの横にある 〔Check Value〕 および 〔New Value〕 フィールドに入力されます。

〔Check Value〕 に値が入力されなかった場合、〔New Value〕 に入力された値は、特定のパラメータ設定が適用するすべてのオブジェクトに影響します。〔Check Value〕 に値が入力された場合、〔New Value〕 に入力された値は、特定のパラメータ設定と 〔Check Value〕 に入力された値が適用されるオブジェクトにのみ影響します。〔Check Value〕 または 〔New Value〕 が、処理するオブジェクトタイプに関連していない場合、どちらのフィールドに入力された値も無視されます。例えば、Natural システムエラーメッセージにライブラリ名は含まれていません。したがって、Natural システムエラーメッセージを処理するとき、〔Library〕 フィールドの 〔Check Value〕 または 〔New Value〕 に入力された値は無視されます。

[Check Value] および [New Value] は、パラメータ [Error number difference] および [Parameters] 画面の [System files for load] セクションに表示されるパラメータには適用されません。

有効なパラメータ設定については、「ダイレクトコマンド」セクションの「[parameter-setting](#)」も参照してください。

[Parameters] 画面には、次のフィールドが表示されます。

フィールド/セクション	説明
Object name	<p><b>Check Value/New Value:</b></p> <p>単一のオブジェクト名または名前の範囲。「名前と日時の指定」の「<a href="#">名前</a>」、および「<a href="#">新しい値のルール</a>」を参照してください。</p>
Library	<p><b>Check Value/New Value:</b></p> <p>単一のライブラリ名または名前の範囲。「<a href="#">名前</a>」および「<a href="#">新しい値のルール</a>」を参照してください。</p>
Date	<p><b>Check Value/New Value:</b></p> <p>単一の日付または日付の範囲。「名前と日時の指定」の「<a href="#">日付</a>」、および「<a href="#">新しい値のルール</a>」を参照してください。</p>
Time	<p><b>Check Value/New Value:</b></p> <p>単一の時刻または時刻の範囲。「名前と日時の指定」の「<a href="#">時刻</a>」、および「<a href="#">新しい値のルール</a>」を参照してください。</p>
User ID	<p><b>Check Value/New Value:</b></p> <p>単一のユーザー ID またはユーザー ID の範囲。「<a href="#">名前</a>」および「<a href="#">新しい値のルール</a>」を参照してください。</p>
Terminal ID	<p><b>Check Value/New Value:</b></p> <p>単一の端末 ID または端末 ID の範囲。「<a href="#">名前</a>」および「<a href="#">新しい値のルール</a>」を参照してください。</p>
Lang. codes	<p>適用されるのは、Natural システムエラーメッセージまたはユーザー定義エラーメッセージを処理する場合に限られます。</p> <p><b>Check Value/New Value:</b></p> <p>有効な言語コードは最大8つまでです（例えばスペイン語の場合はコード4）。複数の言語コードを指定する場合、[Check Value] に同じ数の言語コードが含まれている必要があります。この場合、[Check Value] の言語コードは、対応する [New Value] の言語コードで置き換えられます。</p> <p>注：英語（コード1）が唯一の有効な言語である長文テキストの Natural システムエラーメッセージには、[New Value] は適用されません。</p>

フィールド／セクション	説明
<b>Error number difference</b>	<p>適用されるのは、Natural システムエラーメッセージまたはユーザー定義エラーメッセージを処理する場合に限られます。</p> <p>エラーメッセージの新規の番号の範囲として使用される 4 桁の正または負の値 (+/-nnnn)。選択したエラーメッセージに新しい範囲を適用できるかどうかを確認するため、<b>[Error number from/to]</b> フィールド（「Natural ライブラリオブジェクト」を参照）に開始値と終了値を指定する必要があります。</p> <p>例：</p> <p><b>[Error number from/to]</b> でメッセージ番号 1～10 を選択し、<b>[Error number difference]</b> が 2000 に設定されていると、メッセージ番号が 2001～2010 に再設定されます。<b>[Error number difference]</b> の値を -1000 に設定すると、検証エラーが発生します。</p>
<b>FDT DBID/FNR</b>	<p><b>Check Value/New Value:</b></p> <p>Adabas FDT の有効なデータベース ID (DBID) およびファイル番号 (FNR) のいずれかあるいはその両方。</p>
ロード用のシステム ファイル：Load FNAT  <b>DBID</b> <b>FNR</b> <b>Password</b> <b>Cipher</b>	<p>ロード機能にのみ適用されます。</p> <p>ターゲット FNAT システムファイルのデータベース ID (DBID) とファイル番号 (FNR)。このシステムファイルは、ライブラリ名が SYSTEM ではなく SYS で始まるすべてのライブラリオブジェクトに対して使用されます。必要に応じて、システムファイルの Adabas パスワード (<b>[Password]</b>) とサイファコード (<b>[Cipher]</b>) を入力します。</p>
ロード用のシステム ファイル：Load FUSER  <b>DBID</b> <b>FNR</b> <b>Password</b> <b>Cipher</b>	<p>ロード機能にのみ適用されます。</p> <p>ターゲット FUSER システムファイルのデータベース ID (DBID) とファイル番号 (FNR)。このシステムファイルは、ライブラリ名が SYS で始まらないすべてのライブラリオブジェクトと、ライブラリ SYSTEM に対して使用されます。必要に応じて、システムファイルの Adabas パスワード (<b>[Password]</b>) とサイファコード (<b>[Cipher]</b>) を入力します。</p>
ロード用のシステム ファイル：Load FNAT/FUSER  <b>Select</b>	<p>ロード機能にのみ適用されます。</p> <p><b>[Select System File]</b> ウィンドウが表示され、Natural 環境で使用可能なすべてのシステムファイルのリストが表示されます。「<a href="#">システムファイルの選択</a>」を参照してください。</p>
ロード用のシステム ファイル：Load NCP  <b>DBID</b> <b>FNR</b> <b>Password</b>	<p>ロード機能にのみ適用されます。</p> <p>Natural コマンドプロセッサソースをロードするターゲット Adabas ファイルのデータベース ID (<b>[DBID]</b>) とファイル番号 (<b>[FNR]</b>)。必要に応じて、システムファイルの Adabas パスワード (<b>[Password]</b>) とサイファコード (<b>[Cipher]</b>) を入力します。</p>

フィールド／セクション	説明
Cipher	

このセクションでは、次のトピックについて説明します。

#### ■ 新しい値のルール

### 新しい値のルール

次のルールは、**[Object name]**、**[Library]**、**[Date/Time]**、**[User ID]**、および**[Terminal ID]**の**[New Value]**に適用されます。

**[New Value]** にアスタリスク (\*) 付きの範囲 (ABC\* など) が含まれている場合、アスタリスク (\*) の前の文字数によって、**[Check Value]** で置き換えられる文字数が決まります。これは、**[Check Value]** が **[New Value]** で指定した範囲より短い場合にも有効です (以下の「例」の例 2 を参照)。

例：

1. **[Object name]** が ABCDEFG で、**[New Value]** が ZYX\* に設定されると、結果のオブジェクト名は ZYXDEFG になります。
2. **[Object name]** が AB で、**[New Value]** が ZYX\* に設定されると、結果のオブジェクト名は ZYX になります。
3. **[Date/Time]** が 2005-03-26 で、**[New Value]** が 2006\* に設定されると、結果のオブジェクト日付は 2006-03-26 になります。



# 37 ワークプラン

---

■ ワークプランの作成、選択、および変更 .....	194
■ ワークプランの内容 .....	194
■ ワークプランの例 .....	195
■ ワークプランの参照 .....	196

ワークプランは、コマンド実行、オブジェクト選択、およびパラメータやオプション設定を行うための個々の標準プロシージャを定義します。ワークプランを使用することで、機能処理をさらに自動化することができます。

ワークプランはテキストタイプの Natural オブジェクトです。ワークプランは、デフォルトにより、現在の FUSER システムファイルの WORKPLAN ライブラリに保存されます。

## ワークプランの作成、選択、および変更

管理機能（関連するセクションを参照）を使用して、ワークプランの作成、リストからのワークプランの選択、ワークプランの変更、およびワークプランのデフォルトライブラリの変更を行うことができます。オブジェクトハンドラプロファイルで `Workplan-Library` パラメータを指定して、デフォルトライブラリを変更することもできます（「[プロファイルの設定](#)」を参照）。

## ワークプランの内容

ワークプランは、（オブジェクトハンドラによって生成された）ヘッダーおよび関連する命令やテキストの部分で構成されます。命令部分にはオブジェクトハンドラコマンドおよびパラメータやオプション設定が含まれます。テキスト部分には単純なテキストだけが含まれます。ヘッダーと命令またはテキスト部分には、コメント（例：ワークプランの簡単な説明）を含むことができます。コメントは必ずデリミタ文字 `/*` で始まり、1 行に制限されています。

ワークプランには、PROCEDURE、SELECTION、LIST、PARAMETER、OPTION、および TEXT の 6 タイプがあります。

下の表は、対応するワークプランタイプの有効なヘッダー（オブジェクトハンドラの外部でワークプランを作成する場合に入力される）のリストと命令またはテキスト部分の内容の説明です。また、オブジェクトハンドラのダイレクトコマンドを指定するときに適用される節へのクロスリファレンスも示します。ここで示すオブジェクトハンドラのダイレクトコマンドについては、「[ダイレクトコマンド](#)」セクションを参照してください。

有効なヘッダー	内容	「ダイレクトコマンド」の関連トピック
TYPE PROCEDURE	オブジェクトハンドラコマンドプロシージャの開始。  このワークプランには、PROCEDURE で使用できるオブジェクトハンドラコマンドの任意の組み合わせを含めることができます。セミコロン (;) で区切られた一連のコマンドを入力します。	<a href="#">基本的なコマンド構文</a>
TYPE SELECTION	オブジェクトの選択条件。	<a href="#">select-clause</a>



有効なヘッダー	内容	「ダイレクトコマンド」の関連トピック
	このワークプランは、オブジェクトハンドラのワークプランコマンドで使用できます。	
TYPE LIST	オブジェクトのリスト。  このワークプランは、オブジェクトハンドラのワークプランコマンドで使用できます。	<i>select-clause</i>  <i>オブジェクトリスト-LIST</i> ワークプラン
TYPE PARAMETER	アンロードまたはロード機能のパラメータ。  このワークプランは、オブジェクトがロードされる新しいターゲットライブラリの名前など、処理するオブジェクトの属性を変更するために使用できます。  TYPE PARAMETER は、オブジェクトハンドラのワークプランコマンドで使用できます。	<i>parameter-setting</i>
TYPE OPTION	アンロードまたはロード機能のオプション（レポート設定など）。  このワークプランは、オブジェクトハンドラのワークプランコマンドで使用できます。	<i>option-setting</i>
TYPE TEXT	ドキュメントのために使用できる任意のコメントテキスト。	該当なし

## ワークプランの例

次の表に、ワークプランに含まれる命令部分の例を示します。

ワークプランのタイプ	命令	説明
PROCEDURE	FINDLIB * LIB TEST	ライブラリ TEST の存在をチェックします。
PROCEDURE	UNLOAD A* LIB TEST	ライブラリ TEST から A で始まるすべての Natural プログラミングオブジェクト、およびすべてのユーザー定義エラーメッセージをワークファイル1にアンロードします。
SELECTION	* LIB TEST	ライブラリ TEST のすべてのオブジェクトを処理します。
TEXT	これはワークプランコメントです。	任意のテキスト。

このセクションでは、次のトピックについて説明します。

### ■ ワークプランの内容の例

#### ワークプランの内容の例

次に、UNLOAD コマンドを実行したときの PROCEDURE ワークプランの例を示します。

```
TYPE PROCEDURE /* VERSION=03.01 NATURAL VERSION=04.02.01 PL=0 AUTHOR=SAG
DATE=2006-07-20 09:40:12
/* unload from library TEST with target library PROD01
UNLOAD * LIB TEST OBJTYPE N
WITH NEWLIBRARY PROD01
WHERE REPORT MYREP01
```

#### ワークプランの参照

---

オブジェクトハンドラのメニュー機能またはダイレクトコマンド（「[ダイレクトコマンド](#)」セクションも参照）を使用して、ワークプランを参照することができます。

次の構文は、「[ダイレクトコマンド](#)」セクションで説明しているオブジェクトハンドラのダイレクトコマンドを使用してワークプランを参照する場合に適用されます。

```
( workplan-name
  [ LIBRARY library-name ]
  [ DBIDdbid [FNR fnr] ] [NAME vsam-name ]
  [CIPHER cipher ]
  [
    {
      PASSWORD
      PSW
    } password
  ]
)
```

構文オプションについては、次のセクションで説明します。

## ■ キーワードの説明

### キーワードの説明

下の表は、ワークプランを参照するための構文に適用されるキーワードと値の説明です。

キーワード	値	デフォルト値
<i>workplan-name</i>	ワークプランとして使用する、ワークプランライブラリに保存された Natural テキストメンバの名前。	デフォルトなし
LIBRARY	ワークプランが保存されているライブラリの名前。	WORKPLAN
DBID	ワークプランライブラリが保存されている Adabas データベースの ID。	0 (現在の FNAT/FUSER)
FNR	ワークプランライブラリが保存されている Adabas ファイルの番号。	0 (現在の FNAT/FUSER)
NAME	メインフレーム上のオブジェクトにのみ適用されます。 ワークプランライブラリが保存されている有効な VSAM ファイルの名前。	空白 (現在の FNAT/FUSER)
CIPHER	メインフレーム上のオブジェクトにのみ適用されます。 8 桁のサイファコード。	空白 (現在の FNAT/FUSER)
PASSWORD	メインフレーム上のオブジェクトにのみ適用されます。 8 文字の Adabas パスワード。	空白 (現在の FNAT/FUSER)

---

## 38 名前と日時の指定

---

■ 名前 .....	200
■ 日付 .....	201
■ 時刻 .....	202


単一の名前／日付／時刻、または名前／日付／時刻の範囲を使用して、Natural ライブラリオブジェクト、Natural コマンドプロセッサソース、Natural 関連オブジェクト、または Natural DDM（データ定義モジュール）を選択できます。

## 名前

単一の名前または名前の範囲を指定できます。

下記のオプションのリストで、*value* は 1 つ以上の文字の任意の組み合わせを表します。

	入力	選択される項目
	<i>value</i>	<i>value</i> に一致する名前を持つすべての項目。
	*	すべての項目。
	>	
	?	入力された疑問符 (?) の個数に相当する任意の文字数のすべての項目。
先頭文字	<i>value</i> *	<i>value</i> で始まる名前を持つすべての項目。  例：AB* 選択対象：AB、AB1、ABC、ABEZ 選択対象外：AA1、ACB
ワイルドカード	<i>value</i> ?	<i>value</i> で始まり、入力された疑問符 (?) の個数に相当する任意の文字数がその後続く名前の、すべての項目。  例：ABC? 選択対象：ABCA、ABCZ 選択対象外：AXC、ABCAA
	<i>value</i> ? <i>value</i> ? <i>value</i> * <i>value</i> ? * <i>value</i> ? <i>value</i> *	任意の順序でアスタリスク (*) と疑問符 (?) を指定した <i>value</i> に一致するすべての項目。  例：A?C*Z 選択対象：ABCZ、AXCBBBZ、ANCZ 選択対象外：ACBZ、ABDEZ、AXCBBBZA
開始値	<i>value</i> >	<i>value</i> 以降の名前を持つすべての項目。  例：AB> 選択対象：AB、AB1、BBB、ZZZZZZZ 選択対象外：AA1、AAB
終了値	<i>value</i> <	<i>value</i> 以前の名前を持つすべての項目。  例：AX< 選択対象：AB、AWW、AX 選択対象外：AXA、AY

 **注意:** 先頭文字（アスタリスク（\*）の指定）を利用できるのは、パラメータ指定オプション [New Value] に限られます。「設定」セクションの「グローバルパラメータの設定」にある「[新しい値のルール](#)」を参照してください。

## 日付

オブジェクトハンドラ内のすべての日付値は国際日付フォーマットで指定されます。

日付、日付の範囲、特殊日付、または特殊日付の範囲を指定できます。日付は、YYYY-MM-DD（YYYY=年、MM=月、DD=日）の形式で指定する必要があります。

以下のオプションのリストで、キーワードの下線の部分は有効な省略形を表し、*value*は1桁以上の数字の任意の組み合わせを表します。

	入力値	選択される項目
日付	YYYY-MM-DD	YYYY-MM-DD に等しい日付を持つすべての項目。 例：2003-02-15
先頭文字	<i>value</i> *	<i>value</i> で始まる日付を持つすべての項目。 例：2002* 選択対象：2002-01-01、2002-12-31 選択対象外：2001-12-31、2003-01-01
開始値	<i>value</i> >	<i>value</i> より大きい日付を持つすべての項目。 例：2002-05> 選択対象：2002-05-01、2002-12-31、2003-01-01、2003-12-31 選択対象外：2002-04-31、2001-12-31 <a href="#">特殊な日付の値</a> を <i>value</i> として使用することができます（下記参照）。
終了値	<i>value</i> <	<i>value</i> より小さい日付を持つすべての項目。 例：2003-02< 選択対象：2002-05-01、2002-12-31、2003-01-01、2003-01-31 選択対象外：2003-02-01、2003-05-18 <a href="#">特殊な日付の値</a> を <i>value</i> として使用することができます（下記参照）。
<b>特殊な日付の値</b>		
IQDAY (+/-nnnn)		当日の日付が含まれるすべての項目。  この後には、+nnnn または -nnnn を付け加えることができます。nnnn は最大 4 桁の数値です。  これを付け加えた場合、現在の日付に nnnn 日分を加算または減算した日付が対象として算出されます。

## 名前と日時の指定

	入力値	選択される項目
		例：当日の日付が 2003-03-01 の場合、TODAY +5 は 2003-03-06 になります。
YESTERDAY		当日の 1 日前の日付が含まれるすべての項目。
MONTH		当月内の日付の範囲が含まれるすべての項目。
		例：当月が 2003-02 選択対象：2003-02-01、2003-02-30 選択対象外：2003-03-01
		FMDATE：当月の最初の日で開始します。 TODATE：当月の最後の日で終了します。 FMDATE と TODATE の値が同じであれば、選択は 1 日に限定されます。
YEAR		当年内の日付の範囲が含まれるすべての項目。
		例：当年が 2003 選択対象：2003-01-01、2002-12-31 選択対象外：2002-31-12
		FMDATE：当年の最初の日で開始します。 TODATE：当年の最後の日で終了します。 FMDATE と TODATE の値が同じであれば、選択は 1 年に限定されます。



**注意:** 先頭文字（アスタリスク (\*) の指定）を利用できるのは、パラメータ指定オプション [New Value] に限られます。「設定」セクションの「グローバルパラメータの設定」にある「[新しい値のルール](#)」を参照してください。

## 時刻

単一の時刻または時刻の範囲を指定できます。時刻は、HH:II:SS（HH=時間、II=分、SS=秒）の形式で指定する必要があります。

下記のオプションのリストで、*value* は 1 桁以上の数字の任意の組み合わせを表します。

	入力値	選択される項目
時刻	HH:II:SS	HHIISS に等しい時刻を持つすべての項目。 例：14:15:16
先頭文字	<i>value</i> *	<i>value.</i> で始まる時刻を持つすべての項目。 例：13:* 選択対象：13:00:00、13:10:53、13:59:59



---

	入力値	選択される項目
		選択対象外：12:59:59、14:00:00



# 39      ワークファイル

---

■ ワークファイルの割り当て .....	206
■ ワークファイルフォーマット .....	207

このセクションでは、オブジェクトハンドラのアンロード/ロード/スキャン機能に適用されるワークファイルおよび有効なフォーマットについて説明します。

「設定」セクションの「[ワークファイルオプション](#)」も参照してください。

## ワークファイルの割り当て

下の表は、オブジェクトハンドラで使用されるワークファイルのリストです。

ファイル	説明
ワークファイル 1	アンロード、ロード、およびスキャン機能に使用されます。 アンロードされたデータが含まれます。
ワークファイル 7	使用されるのは、Entire Connection がインストールされていて、なおかつ、 <b>[Use PC File]</b> が <b>[Options]</b> 画面で選択されている場合に限られます（「 <a href="#">その他のオプションの設定</a> 」も参照）。  アンロード、ロードおよびスキャン機能で使用するには、ワークファイル 7 を Entire Connection ワークファイルとして定義する必要があります。  アンロードされたデータが含まれます。
ワークファイル 8	使用されるのは、Entire Connection がインストールされていて、なおかつ、 <b>[Use PC File]</b> が <b>[Options]</b> 画面で選択されている場合に限られます（「 <a href="#">その他のオプションの設定</a> 」も参照）。  アンロード、ロード、およびスキャンの機能で使用するには、ワークファイル 8 を Entire Connection ワークファイルとして定義する必要があります。  Entire Connection コマンドを処理する内部ファイルとして使用されます。  <b>注意:</b>  PROCEDURE タイプのワークプランを使用している場合、ユーザー出力ルーチン <b>OBJHEX03</b> （「 <a href="#">バッチのコンディションコードとユーザー出力ルーチン</a> 」を参照）またはオプション PCCOMMANDFILENUMBER、PCCOM、および PCCFN（ <a href="#">option-setting</a> を参照）を使用して、ワークファイルの番号を変更できます。

## ワークファイルフォーマット

ソース環境のオブジェクトをワークファイルにアンロードし、それらをワークファイルからターゲット環境にロードするために、2つのファイルフォーマット（内部フォーマットと転送フォーマット）が用意されています。バイナリデータを転送するには、ワークファイルを内部フォーマットにする必要があります。テキストデータを転送するには、ワークファイルを転送フォーマットにする必要があります。

このセクションでは、次のトピックについて説明します。

- 内部フォーマット
- 転送フォーマット

### 内部フォーマット

内部フォーマットとは、Natural ソースおよびカタログ化オブジェクト、エラーメッセージ、Natural コマンドプロセッサソース、Adabas FDT（フィールド定義テーブル）、Natural DDM（データ定義モジュール）、および Natural 関連オブジェクトを、1つの環境から別の環境に転送するために使用されるワークファイルの内部レコードレイアウトのことです。

同一プラットフォーム間でオブジェクトを転送するには、内部フォーマットのワークファイルを使用します。

内部フォーマットが有効な場合、Natural オブジェクトのソース環境からの読み取りと Natural ワークファイルへの書き込みは、オブジェクトハンドラのアンロード機能によって行われます。このワークファイルは、標準的なファイル転送サービスで別の環境に転送できます。ターゲット環境では、オブジェクトハンドラのロード機能を使用して、オブジェクトをワークファイルから読み取り、ローカルファイルまたはデータベースシステムにロードすることができます。

オブジェクトハンドラでは、可変長のレコードがワークファイルに書き込まれてワークファイルのサイズが削減されるため、RECFM=VB（可変長ブロック）を使用してワークファイル1を定義することをお勧めします。

RECFM=FB（固定長ブロック）を使用してワークファイル1を定義すると、アンロード機能でオプション FIXEDLENGTH の指定が必要になります。このオプションを指定しないと、ワークファイルに書き込まれるレコードにバイナリの0が埋め込まれます。



**注意:** ユーティリティ NATUNLD によってサーバーで作成されたワークファイルは、内部フォーマットで処理する必要があります。ワークファイルは、NATUNLD を適用したときと同じプラットフォームのサーバーで作成する必要があります。

### 転送フォーマット

「設定」セクションの「**Transfer format**」も参照してください。

転送フォーマットは、ワークファイルの一般的なレコードレイアウトであり、ロードデータまたはアンロードデータを含みます。このフォーマットはプラットフォームに依存せず、Natural オブジェクトのソース、Natural コマンドプロセッサソース、エラーメッセージ、DDM、および Adabas FDT を、1つのハードウェアプラットフォームから別のハードウェアプラットフォームに転送したり、UNIX、OpenVMS、メインフレームと Windows プラットフォーム間で転送したりするために使用できます。

「**Transfer format**」オプションが設定されていると、オブジェクトハンドラのアンロード機能によって、ハードウェアプラットフォームから Natural オブジェクトが読み取られた後、それらが再構築されます。

フォーマットされたレコードは Natural ワークファイルに書き込まれます。このワークファイルは、標準的なファイル転送サービスで別のプラットフォームに転送できます。ターゲットプラットフォームでは、オブジェクトハンドラのロード機能によってオブジェクトがワークファイルから読み取られ、ローカルファイルまたはデータベースシステムにロードされます。ワークファイルから読み取られたオブジェクトは、新しいハードウェアプラットフォームの構造に従って再構築されます。

### Unicode/UTF-8 でのソースの処理

転送フォーマットは、Unicode/UTF-8 (Universal Transformation Format、8 ビット形式) で Natural オブジェクトソースをアンロードまたはロードするためにも使用します。該当するアンロードオプションを指定した場合 (コマンドモードで **WORKFILETYPE** を UTF-8 に設定するか、メニューモードで **[Unicode work file]** を選択した場合)、すべてのオブジェクトソースは UTF-8 でワークファイルにアンロードされます。該当するロードオプションを指定した場合 (コマンドモードで **LOAD-CODE-PAGE** を指定するか、またはメニューモードで **[Use load code page]** を選択した場合)、UTF-8 のすべてのオブジェクトソースは、Natural システムファイルにロードされるときに指定のコードページで変換されます。

### SYSTRANS によるワークファイル

ユーティリティ SYSTRANS によって作成されたワークファイルは、転送フォーマットで処理されます。UTF-8 でエンコードされたオブジェクトソースを含むワークファイルは、SYSTRANS で処理できません。

# 40      ダイレクトコマンド

---

オブジェクトハンドラには、次の処理を行うためにダイレクトコマンドが用意されています。

- オブジェクトのアンロードやロードなど、オブジェクトハンドラ機能の実行
- ワークプラン（「[ワークプラン](#)」セクションも参照）の実行または参照
- ワークプラン内の命令としての使用
- 画面間の移動
- 特殊機能の実行

このセクションでは、基本的なコマンド構文と個々の節、これらのタスクの実行に使用できるパラメータやオプションの設定について説明します。また、ダイレクトコマンドの使用例も示します。

このセクションの構文図で使用する記号については、『システムコマンド』ドキュメントの「[システムコマンド構文](#)」を参照してください。

このセクションでは、次のトピックについて説明します。

- [基本的なコマンド構文](#)
- [select-clause](#)
- [オブジェクトリスト -LIST ワークプラン](#)
- [parameter-setting](#)
- [option-setting](#)
- [ダイレクトコマンドの使用例](#)
- [ナビゲーションと特別な機能のためのコマンド](#)





# 41 基本的なコマンド構文

---

このセクションでは、オブジェクトハンドラ機能およびタイプ PROCEDURE のワークプランを実行するために用意されているオブジェクトハンドラのダイレクトコマンドについて説明します。また、古いユーティリティ NATUNLD/NATLOAD および SYSTRANS からオブジェクトハンドラへの移行に使用するコマンドについても説明します。

このセクションの構文図で使用している変数値については、『オブジェクトハンドラ』ドキュメントの該当するセクションを参照してください。構文図で使用している記号については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

```
EXECUTE (procedure-workplan)
```

タイプ PROCEDURE のワークプランを実行します。「[ワークプラン](#)」セクションも参照してください。

```
UNLOAD select-clause [parameter-setting] [option-setting]
```

*parameter-setting* に定義されたパラメータおよび *option-setting* に定義されたオプションを使用して、*select-clause* に定義されたオブジェクトをアンロードします。

```
LOAD select-clause [parameter-setting] [option-setting]
```

*parameter-setting* に定義されたパラメータおよび *option-setting* に定義されたオプションを使用して、*select-clause* に定義されたオブジェクトをロードします。

```
LOADALL [parameter-setting] [option-setting]
```

*parameter-setting* に定義されたパラメータおよび *option-setting* に定義されたオプションを使用して、ワークファイルからすべてのオブジェクトをロードします。

```
SCAN select-clause [option-setting]
```

*option-setting* に定義されたオプションを使用して、*select-clause* に定義されたオブジェクトのワークファイルをスキャンします。

```
SCANALL [option-setting]
```

*option-setting* に定義されたオプションを使用して、すべてのオブジェクトのワークファイルをスキャンします。

```
FIND select-clause [option-setting]
```

*option-setting* に定義されたオプションを使用して *select-clause* に定義されたオブジェクトを検索し、検出されたオブジェクトのレポートをワークプランライブラリに保存されている Natural テキストメンバに書き込みます。また、検出されたオブジェクトのレポートは、ワークプランライブラリ内で Natural テキストメンバとして指定されたレポートファイルに書き込むことができます。

```
FINDLIB select-clause [option-setting]
```

*option-setting* に定義されたオプションを使用して *select-clause* に定義された Natural オブジェクトまたは Natural コマンドプロセッサソースのライブラリを検索し、検出されたオブジェクトのレポートをワークプランライブラリに保存されている Natural テキストメンバに書き込みます。また、検出されたオブジェクトのレポートは、ワークプランライブラリ内で Natural テキストメンバとして指定されたレポートファイルに書き込むことができます。

```
DELETE select-clause [option-setting]
```

*option-setting* に定義されたオプションを使用して、*select-clause* に定義されたオブジェクトを削除します。

制限：FDT は削除できません。

```
UNDELI select-clause [option-setting]
```

*option-setting* に定義されたオプションを使用して、*select-clause* に定義されたオブジェクトの削除命令をアンロードします。

```
RESTART [restart-text-member]
```

中断されたロード機能を継続します。これは、中断されたロード中に情報がワークプランライブラリに保存されている Natural テキストメンバに書き込まれた場合にのみ可能です。*option-setting* セクションの RESTART (「ダイレクトコマンド」) および「ロードの再スタート」を参照してください。

```
DISPLAY STATISTICS
```

処理されたオブジェクトの統計情報が表示されます。

```
NATUNLD natunld-direct-command
```

古いユーティリティ NATUNLD の構文でオブジェクトハンドラコマンドを実行します。

「[NATUNLD/NATLOAD](#) および [SYSTRANS](#) からオブジェクトハンドラへの移行」も参照してください。

```
NATLOAD natload-direct-command
```

古いユーティリティ NATLOAD の構文で発行されたオブジェクトハンドラダイレクトコマンドを実行します。「[NATUNLD/NATLOAD](#) および [SYSTRANS](#) からオブジェクトハンドラへの移行」も参照してください。

```
SYSTRANS systrans-direct-command
```

古いユーティリティ SYSTRANS の構文で発行されたオブジェクトハンドラダイレクトコマンドを実行します。「[NATUNLD/NATLOAD](#) および [SYSTRANS](#) からオブジェクトハンドラへの移行」も参照してください。



# 42

## select-clause

---

▪ select-clause の構文 .....	216
▪ SELECTION または LIST ワークプラン .....	216
▪ Natural ライブラリオブジェクトと DDM の選択 .....	217
▪ Natural 関連デバッグ環境の選択 .....	224
▪ Natural 関連プロファイルの選択 .....	225
▪ Natural 関連 DL/I サブファイルの選択 .....	227
▪ Natural システムエラーメッセージの選択 .....	228
▪ Natural コマンドプロセッサソースの選択 .....	230
▪ FDT の選択 .....	232
▪ アプリケーションの選択 .....	233
▪ 削除命令のためのオブジェクト選択 .....	235
▪ ヘルプテキストの選択 .....	238

*select-clause* は、タイプ SELECTION または LIST のワークプラン、あるいは処理するオブジェクト、FDT、またはアプリケーションの選択指定のいずれかで構成されます。

このセクションでは、*select-clause* に適用される構文について説明します。構文図で使用しているキーワードおよび変数値は、オブジェクトの選択条件の指定に使用できるパラメータを表します。必要な場合、変数値をキーワードと一緒に指定する必要があります。

## select-clause の構文

*select-clause* は、次のいずれかのオプションで構成されます。

```
{ (selection-workplan)
  (list-workplan)
  object-selection
  delete-instruction-selection
  help-text-selection }
```

*selection-workplan* および *list-workplan* オプションについては、以下の「SELECTION または LIST ワークプラン」で説明しています。

*object-selection* の使用法は、処理するオブジェクトタイプ、DDM、FDT、またはアプリケーションによって異なります。それぞれに適した構文およびキーワードについて、このセクションの残りで説明します。

*delete-instruction-selection* オプションについては、「[削除命令のためのオブジェクト選択](#)」で説明しています。

*help-text-selection* オプションについては、「[ヘルプテキストの選択](#)」で説明しています。

## SELECTION または LIST ワークプラン

タイプ SELECTION のワークプランには、ヘッダー (TYPE SELECTION) と次のいずれかのタイプのオブジェクトまたはファイルが含まれます：Natural ライブラリオブジェクト、Natural 関連オブジェクト、Natural システムエラーメッセージ、Natural コマンドプロセッサソース、DDM、または Adabas FDT (フィールド定義テーブル)。

タイプ LIST のワークプランには、ヘッダー (TYPE LIST) と「[オブジェクトリスト - LIST ワークプラン](#)」セクションで説明しているオブジェクトの選択リストが含まれます。このようなオブジェクトリストは、UNLOAD、LOAD、または FIND コマンドにのみ使用できます。

ワークプランの使用の詳細については、「[ワークプラン](#)」セクションを参照してください。

## Natural ライブラリオブジェクトと DDM の選択

この選択は、Natural DDM（データ定義モジュール）およびユーザー定義エラーメッセージなど、処理する Natural オブジェクトを選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural ライブラリオブジェクトと DDM の選択の構文

#### Natural ライブラリオブジェクトと DDM の選択の構文

```

object-name
LIBRARY library-name
[ DBIDdbid FNRfnr [NAME vsam-name] [CIPHER cipher] [ { PASSWORD password PSW } ] ]
[OBJTYPE group-type]
[ SETNOset-number [SETUSER set-user] [SETLIBRARY set-library] ]
[NATYPE object-type]
[SKIND object-kind]
[MODE object-mode]
[FMNUM error-number-from]
[TONUM error-number-to]
[SLKIND message-type]
[LANGUAGE languages]
[DDMDBIDddm-dbid] [DDMFNRddm-fnr]
[NATVERS natural-version]
[ DATEdate ]
[ [FMDATE date-from] [TODATE date-to] ]
[ [SIZE size] ]
[ [FMSIZE size-from] [TOSIZE size-to] ]
[USERID user-id]
[TID terminal-id]
[except-clause]


```

*except-clause*

```

EXCEPT
( object-name
[LIBRARY library-name]
[OBJTYPE group-type]
[SCKIND object-kind]
[NATTYPE object-type]
[MODE object-mode]
[SLKIND message-type]
[FMNUM error-number-from] [TONUM error-number-to]
[LANGUAGE languages]
[DDMDBID dgm-dbid] [DDMFNR dgm-fnr]
[NATVERS natural-version]
[
    DATE date
    [FMDATE date-from] [TODATE date-to]
]
[
    SIZE size
    [FMSIZE size-from] [TOSIZE size-to]
]
[USERID user-id]
[TID terminal-id]
)

```

 **注意:**

1. コマンド FINDLIB では、次のキーワードのみが処理されます：LIBRARY、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW。
2. Natural DDM を処理する場合、OBJTYPE を D に設定する必要があります。また、以下で説明しているように、一部のキーワードは DDM に適用されません。

**Natural** ライブラリオブジェクトと **DDM** の選択のキーワードの説明

次のセクションでは、処理するオブジェクトのキーワードおよび有効な値について説明します。

キーワード	有効な値	デフォルト値
<i>object-name</i>	有効なオブジェクト名または名前の範囲。 <i>object-name</i> に空白文字が含まれている場合、文字列を二重引用符 (" ") で囲む必要があります。  「名前と日時の指定」の「名前」も参照してください。	なし




キーワード	有効な値	デフォルト値
LIBRARY	<p>有効なライブラリ名または名前の範囲。</p> <p>OBJTYPE (下記参照) が D に設定されている場合、ライブラリ名は無視されます。</p> <p>SETNO が指定されている場合、名前の範囲は使用できません。</p> <p>「名前」も参照してください。</p>	なし
DBID	<p>メインフレーム上の DDM (OBJTYPE が D に設定。下記参照) には有効ではありません</p> <p>有効なデータベース ID。</p>	0 (現在の FNAT/FUSER)
FNR	<p>メインフレーム上の DDM (OBJTYPE が D に設定。下記参照) には有効ではありません</p> <p>有効なファイル番号。</p>	0 (現在の FNAT/FUSER)
NAME	<p>メインフレーム上のオブジェクトにのみ適用されます。メインフレーム上の DDM (OBJTYPE が D に設定。下記参照) には有効ではありません</p> <p>有効な VSAM 名。</p>	空白 (現在の FNAT/FUSER)
CIPHER	<p>メインフレーム上のオブジェクトにのみ適用されます。メインフレーム上の DDM (OBJTYPE が D に設定。下記参照) には有効ではありません</p> <p>オブジェクトが保存されている Adabas ファイルの 8 桁のサイファコード。</p>	空白 (現在の FNAT/FUSER)
PASSWORD または PSW	<p>メインフレーム上のオブジェクトにのみ適用されます。</p> <p>メインフレーム上の DDM (OBJTYPE が D に設定。下記参照) には有効ではありません</p> <p>8 文字の Adabas パスワード。</p>	空白 (現在の FNAT/FUSER)
OBJTYPE	<p>オブジェクトタイプは次のとおりです。</p> <p>D DDM</p> <p>E Natural エラーメッセージ</p> <p>N Natural プログラミングオブジェクト</p> <p>* アスタリスク (すべて)</p> <p>または有効な組み合わせ。</p> <p>例外： オブジェクトタイプ D は、他のどのタイプとも組み合わせることができません。</p>	*

キーワード	有効な値	デフォルト値
SETNO	<p>Predictがインストールされている場合に、アンロード機能および検索機能にのみ適用されます。アプリケーションオブジェクトには適用されません（「<a href="#">アプリケーションオブジェクトの選択</a>」も参照）。</p> <p>処理するオブジェクトの名前に使用する保有セットを識別する 1 桁または 2 桁の数字。保有セットは、LIST XREF コマンドのセット保存オプションで作成されます。</p> <p>SETNO が指定されていると、<i>object-name</i> に指定した値は無視されます。</p> <p>Predict セットの詳細については、『Predict』ドキュメントを参照してください。</p>	なし
SETUSER	<p>Predictがインストールされている場合に、アンロード機能および検索機能にのみ適用されます。アプリケーションオブジェクトには適用されません（「<a href="#">アプリケーションオブジェクトの選択</a>」も参照）。</p> <p>Predict セットを作成したユーザーの ID。ID が指定されていないと、システム変数 *USER（『システム変数』ドキュメントも参照）の値が使用されます。</p>	*USER
SETLIBRARY	<p>Predictがインストールされている場合に、アンロード機能および検索機能にのみ適用されます。アプリケーションオブジェクトには適用されません（「<a href="#">アプリケーションオブジェクトの選択</a>」も参照）。</p> <p>Predict セットに使用するために検索するライブラリの名前。SETLIBRARY を指定しないと、LIBRARY で指定されたライブラリが代わりに使用されます。</p>	
NATTYPE	<p>OBJTYPE が D に設定されている場合、適用されません。</p> <p>Natural オブジェクトタイプの 1 つまたは複数の単一文字コード：</p> <p>P プログラム  N サブプログラム  S サブルーチン  C コピーコード  H ヘルプルーチン  T テキスト  7 ファンクション  8 アダプタ  G グローバルデータエリア  L ローカルデータエリア  A パラメータデータエリア  M マップ  4 クラス</p>	*

キーワード	有効な値	デフォルト値
	3 ダイアログ 5 Natural コマンドプロセッサ 9 リソース * すべてのオブジェクトタイプ	
SCKIND	<b>OBJTYPE</b> が D に設定されている場合、適用されません。  Natural プログラミングオブジェクトの種類。 有効な入力値は次のとおりです。  S ソースオブジェクト：ソース形式でのみ保存されるオブジェクト C カタログ化オブジェクト：カタログ形式でのみ保存されるオブジェクト A すべてのソースオブジェクトおよびカタログ化オブジェクト。 W STOW を実行された全オブジェクト：同じ日時のソースオブジェクトとカタログ化オブジェクト B ソースオブジェクトおよびカタログ化オブジェクト（両方ともある場合）。  <b>注意:</b> W と B は、UNLOAD および FIND コマンドにのみ有効です。LOAD および SCAN では、W と B は有効なエントリですが、A（すべてのオブジェクト）として処理されず、データが転送フォーマットで処理される場合、S（ソースオブジェクト）または A のみが適用されます。	A
MODE	<b>OBJTYPE</b> が D に設定されている場合、適用されません。  Natural プログラミングオブジェクトのプログラミングモード。 有効な入力値は次のとおりです。  A 任意。 R レポートモードのすべてのオブジェクト。 S ストラクチャードモードのすべてのオブジェクト。	
FMNUM	Natural エラーメッセージの開始番号。  有効な範囲：1～9999。	1
TONUM	Natural エラーメッセージの終了番号。  有効な範囲：1～9999。  値は、FMNUM の値（指定された場合）以上にする必要があります。	9999 または FMNUM の値 （指定されている場合）
SLKIND	Natural エラーメッセージのテキストのタイプ。 有効な入力値は次のとおりです。	A

キーワード	有効な値	デフォルト値
	<p>S 短文テキスト。 DELETE コマンドには適用できません（「<a href="#">基本的なコマンド構文</a>」を参照）。</p> <p>L 長文テキスト。 A 短文テキストと長文テキストの両方または片方。 B 短文テキストと長文テキスト（両方が存在する場合）。</p>	
LANGUAGE	ユーザー定義エラーメッセージの最大8つまでの有効な言語コード（英語の場合はコード1）。アスタリスク（*）にすると、すべての言語コードが選択されます。	*
DDMDBID	DDM の有効なデータベース ID（1～65535）。  UNLOAD、LOAD、および SCAN：0 はチェックが実行されないことを表します。データベース ID（DBID）に関係なく、DDM は処理されます。	0
DDMFNR	DDM の有効なファイル番号（1～65535）。  UNLOAD、LOAD、および SCAN：0 はチェックが実行されないことを表します。ファイル番号（FNR）に関係なく、DDM は処理されます。	0
NATVERS	Natural プログラミングオブジェクトの Natural バージョン。  有効なバージョンフォーマット：VRS （ここで、V は 1 桁のバージョン、R は 1 桁のリリース、S は 2 桁のシステムメンテナンスレベルです） バージョンの範囲を指定することもできます。「 <a href="#">名前</a> 」を参照してください。	空白 （チェックなし）
DATE	Natural プログラミングオブジェクトを保存またはカタログした日付。  時刻を追加するには、日付と時刻の間に空白を挿入します。有効なフォーマットと範囲については、「 <a href="#">名前と日時の指定</a> 」の「 <a href="#">日付</a> 」および「 <a href="#">時刻</a> 」を参照してください。  日付には、YESTERDAY および TODAY をいう特殊な表現を使用できます。「 <a href="#">日付</a> 」の「 <a href="#">特殊な日付の値</a> 」を参照してください。	空白 （チェックなし）
FMDATE	開始値：  Natural プログラミングオブジェクトがカタログまたは保存された日付は、この日付以降になります。フォーマットは DATE と同じです。「 <a href="#">日付</a> 」を参照してください。  YEAR、MONTH、YESTERDAY、および TODAY という特殊な表現を使用できます。「 <a href="#">日付</a> 」の「 <a href="#">特殊な日付の値</a> 」を参照してください。	空白 （チェックなし）

キーワード	有効な値	デフォルト値
TODATE	<p>終了値：</p> <p>Natural プログラミングオブジェクトがカタログまたは保存された日付は、この日付以前になります。フォーマットは DATE と同じです。「日付」を参照してください。</p> <p>YEAR、MONTH、YESTERDAY、および TODAY という特殊な表現を使用できます。「日付」の「特殊な日付の値」を参照してください。</p>	<p>空白</p> <p>(チェックなし)</p> <p>または上位の値</p> <p>(FMDATE が指定されている場合)</p>
SIZE	Natural プログラミングオブジェクトのサイズ (7桁以内)。	<p>0</p> <p>(チェックなし)</p>
FMSIZE	<p>開始値：</p> <p>Natural プログラミングオブジェクトの最小サイズ (7桁以内)。</p>	<p>0</p> <p>(チェックなし)</p>
TOSIZE	<p>終了値：</p> <p>Natural プログラミングオブジェクトの最大サイズ (7桁以内)。</p>	<p>0</p> <p>(チェックなし)</p> <p>または上位の値</p> <p>(FMSIZE が指定されている場合)</p>
USERID	<p>Natural プログラミングオブジェクトを保存またはカタログしたユーザーの ID。</p> <p>ユーザー ID の範囲を指定することもできます。「名前」を参照してください。</p>	<p>空白</p> <p>(チェックなし)</p>
TID	<p>OBJTYPE が D に設定されている場合、適用されません。</p> <p>Natural プログラミングオブジェクトが保存またはカタログされた端末の ID (Natural システム変数 *INIT-ID で指定される)。</p> <p>端末 ID の範囲を指定することもできます。「名前」を参照してください。</p>	<p>空白</p> <p>(チェックなし)</p>
EXCEPT	EXCEPT の前に入力された選択条件に一致しているすべての項目は、キーワード EXCEPT の後のカッコ内に含まれているすべてのパラメータに対してチェックされます。すべてのパラメータとも一致している場合、処理されません。	該当なし

 **注意:**

1. OBJTYPE に無関係なパラメータは無視されます。例：DATE、SIZE、および USERID は、Natural エラーメッセージに対して意味がありません。
2. LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります (「*parameter-setting* のキーワードの説明」の「LOADFNAT...」および「LOADFUSER...」の項を参照)。

## Natural 関連デバッグ環境の選択

この選択は、処理する Natural 関連デバッグ環境を選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural 関連デバッグ環境の選択の構文

#### Natural 関連デバッグ環境の選択の構文

```

object-name
NATPATH DEBUG
[LIBRARY library-name]
[
  DBIDdbid [FNRfnr] ]
[NAME vsam-name]
[CIPHER cipher]
[
  {
    PASSWORD
    PSW
  } password ]
[ EXCEPT
  (object-name
  [LIBRARY library-name]
  )]


```

#### Natural 関連デバッグ環境の選択のキーワードの説明

次のセクションでは、処理するデバッグ環境のキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値
<i>object-name</i>	有効なデバッグ環境名または名前の範囲。 「名前と日時の指定」の「名前」も参照してください。	なし
LIBRARY	有効なライブラリ名または名前の範囲。 「名前」も参照してください。	なし
DBID	有効なデータベース ID。	0 (現在の FUSER)
FNR	有効なファイル番号。	0 (現在の FUSER)

キーワード	有効な値	デフォルト値
NAME	有効な VSAM 名。	空白 (現在の FUSER)
CIPHER	デバッグ環境が保存されている Adabas ファイルの 8 桁のサイファコード。	空白 (現在の FUSER)
PASSWORD または PSW	8 文字の Adabas パスワード。	空白 (現在の FUSER)
EXCEPT	「 <i>Natural</i> ライブラリオブジェクトと DDM の選択」の「EXCEPT」の項を参照してください。	該当なし

 **注意:** LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります（「*parameter-setting* のキーワードの説明」の「LOADFNAT...」および「LOADFUSER...」の項を参照）。

## Natural 関連プロファイルの選択

この選択は、処理する Natural 関連プロファイルを選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural 関連プロファイルの選択の構文

#### Natural 関連プロファイルの選択の構文

```

object-name
NATPATH PROFILE
[OBJTYPE profile-type]
[ DBID dbid [FNR fnr] ]
[NAME vsam-name]
[CIPHER cipher]
[ { PASSWORD } password ]
[ EXCEPT
  (object-name
  [OBJTYPE profile-type]
  )]

```

## Natural 関連プロファイルの選択のキーワードの説明

次のセクションでは、処理するプロファイルのキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値														
<i>object-name</i>	有効なプロファイル名または名前 の範囲。 「名前と日時の指定」の「名前」も参照してください。	なし														
OBJTYPE	プロファイルのタイプ： <table border="1" data-bbox="332 598 1063 913"> <tr><td></td><td></td></tr> <tr><td>D</td><td>デバイスプロファイル</td></tr> <tr><td>E</td><td>エディタプロファイル</td></tr> <tr><td>M</td><td>マッププロファイル</td></tr> <tr><td>P</td><td>パラメータプロファイル</td></tr> <tr><td>*</td><td>アスタリスク (すべてのプロファイルタイプ)</td></tr> <tr><td></td><td></td></tr> </table> または任意の組み合わせ。			D	デバイスプロファイル	E	エディタプロファイル	M	マッププロファイル	P	パラメータプロファイル	*	アスタリスク (すべてのプロファイルタイプ)			*
D	デバイスプロファイル															
E	エディタプロファイル															
M	マッププロファイル															
P	パラメータプロファイル															
*	アスタリスク (すべてのプロファイルタイプ)															
DBID	有効なデータベース ID。	0 (現在の FNAT)														
FNR	有効なファイル番号。	0 (現在の FNAT)														
NAME	有効な VSAM 名。	空白 (現在の FNAT)														
CIPHER	プロファイルが保存されている Adabas ファイルの 8 桁のサイファコード。	空白 (現在の FNAT)														
PASSWORD または PSW	8 文字の Adabas パスワード。	空白 (現在の FNAT)														
EXCEPT	「Natural ライブラリオブジェクトと DDM の選択」の「EXCEPT」の項を参照してください。	該当なし														



**注意:** LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります（「*parameter-setting* のキーワードの説明」の「LOADFNAT...」および「LOADFUSER...」の項を参照）。



## Natural 関連 DL/I サブファイルの選択

この選択は、処理する Natural 関連 DL/I サブファイルを選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural 関連 DL/I サブファイルの選択の構文

#### Natural 関連 DL/I サブファイルの選択の構文

```

object-name
NATPATH SUBFILE
[OBJTYPE subfile-type]

[ DBID dbid [FNR fnr] ]
[NAME vsam-name]
[CIPHER cipher]
[ { PASSWORD } password
  PSW ]
[ EXCEPT
  ( object-name
    [OBJTYPE subfile-type]
  ) ]

```

#### Natural 関連 DL/I サブファイルの選択のキーワードの説明

次のセクションでは、処理する DL/I サブファイルのキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値								
<i>object-name</i>	有効な DL/I サブファイル名または名前の範囲。 「名前と日時の指定」の「名前」も参照してください。	なし								
OBJTYPE	DL/I サブファイルのタイプ： <table border="1" data-bbox="423 1711 1149 1896"> <tbody> <tr> <td></td> <td></td> </tr> <tr> <td>D</td> <td>NDB</td> </tr> <tr> <td>P</td> <td>NSB</td> </tr> <tr> <td>*</td> <td>アスタリスク（両方のサブファイルタイプ）</td> </tr> </tbody> </table>			D	NDB	P	NSB	*	アスタリスク（両方のサブファイルタイプ）	*
D	NDB									
P	NSB									
*	アスタリスク（両方のサブファイルタイプ）									

キーワード	有効な値	デフォルト値
DBID	有効なデータベース ID。	0 (現在の FDIC)
FNR	有効なファイル番号。	0 (現在の FDIC)
NAME	有効な VSAM 名。	空白 (現在の FDIC)
CIPHER	DL/Iサブファイルが保存されている Adabas ファイルの8桁のサイファコード。	空白 (現在の FDIC)
PASSWORD または PSW	8文字の Adabas パスワード。	空白 (現在の FDIC)
EXCEPT	「 <i>Natural</i> ライブラリオブジェクトと DDM の選択」の「EXCEPT」の項を参照してください。	該当なし



**注意:** LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります（「*parameter-setting* のキーワードの説明」の「LOADFNAT...」および「LOADFUSER...」の項を参照）。

## Natural システムエラーメッセージの選択

この選択は、処理する Natural システムエラーメッセージを選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural システムエラーメッセージの選択の構文

#### Natural システムエラーメッセージの選択の構文

```

ERROR NATERROR
[ DBID dbid FNR fnr
  [ NAME vsam-name [ { PASSWORD } password ]
  [ CIPHER cipher ] ] ]
[ FMNUM error-number-from ] [ TONUM error-number-to ]
[ SLKIND message-type ]
[ LANGUAGE languages ]
[ EXCEPT
  (
    [ FMNUM error-number-from ] [ TONUM error-number-to ]
  )
]

```


```
[SLKIND message-type]
[LANGUAGE languages]
)]
```

### Natural システムエラーメッセージの選択のキーワードの説明

次のセクションでは、処理する Natural システムエラーメッセージのキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値
DBID	メインフレーム上のシステムエラーメッセージにのみ適用されます。 有効なデータベース ID。	0 (現在の FNAT)
FNR	メインフレーム上のシステムエラーメッセージにのみ適用されます。 有効なファイル番号。	0 (現在の FNAT)
NAME	メインフレーム上のシステムエラーメッセージにのみ適用されます。 有効な VSAM 名。	空白 (現在の FNAT)
CIPHER	メインフレーム上のシステムエラーメッセージにのみ適用されます。 システムエラーメッセージが保存されている Adabas ファイルの 8 桁のサイファコード。	空白 (現在の FNAT)
PASSWORD または PSW	メインフレーム上のシステムエラーメッセージにのみ適用されます。 8 文字の Adabas パスワード。	空白 (現在の FNAT)
FMNUM	システムエラーメッセージの開始番号。 有効な範囲：1～9999。	1
TONUM	システムエラーメッセージの終了番号。 有効な範囲：1～9999。 値は、FMNUM の値（指定された場合）以上にする必要があります。	9999 または FMNUM の値 (指定されている場合)
SLKIND	「 <i>Natural</i> ライブラリオブジェクトと DDM の選択」の「 <a href="#">SLKIND</a> 」の項を参照してください。	A
LANGUAGE	システムエラーメッセージの最大 8 つまでの有効な言語コード（英語の場合はコード 1）。 アスタリスク (*) にすると、すべての言語コードが選択されます。	*

キーワード	有効な値	デフォルト値
EXCEPT	「Natural ライブラリオブジェクトとDDMの選択」の「EXCEPT」の項を参照してください。	

 **注意:** LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります（「*parameter-setting* のキーワードの説明」の「LOADFNAT...」の項を参照）。

## Natural コマンドプロセッサソースの選択

この選択は、処理する Natural コマンドプロセッサソースを選択するために使用します。

以下のセクションでは、適切な構文を示して説明します。

### ■ Natural コマンドプロセッサソースの選択の構文

#### Natural コマンドプロセッサソースの選択の構文

```

object-name PROCESSOR ncp-library-name
[
    DBID ncp-dbid FNR ncp-fnr [file-options]
]
[EXCEPT
    (object-name
    [LIBRARY ncp-library-name]
    )]


```

*file-options*

```

[NAME ncp-vsam-name]
[CIPHER ncp-cipher]
[ { PASSWORD
  PSW } ncp-password ]

```

 **注意:** コマンド FINDLIB では、次のキーワードのみが処理されます：PROCESSOR、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW。

## Natural コマンドプロセッサソースの選択のキーワードの説明

次のセクションでは、処理する Natural コマンドプロセッサソースのキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値
<i>object-name</i>	Natural コマンドプロセッサソース名または名前の範囲。 「名前と日時指定」の「名前」も参照してください。	なし
PROCESSOR	有効なライブラリ名または名前の範囲。 「名前」も参照してください。	なし
DBID	Natural コマンドプロセッサソースが保存されている Adabas ファイルの有効なデータベース ID。	0 (現在の FNAT/FUSER)
FNR	Natural コマンドプロセッサソースが保存されている Adabas ファイルの有効なファイル番号。	0 (現在の FNAT/FUSER)
NAME	メインフレーム上の Natural コマンドプロセッサソースにのみ適用されます。  有効な VSAM 名。	空白
CIPHER	Natural コマンドプロセッサソースが保存されている Adabas ファイルの 8 桁のサイファコード。	空白
PASSWORD または PSW	Natural コマンドプロセッサソースが保存されている Adabas ファイルの 8 文字の Adabas パスワード。	空白
EXCEPT	「Natural ライブラリオブジェクトと DDM の選択」の「EXCEPT」の項を参照してください。	



**注意:** LOAD または SCAN コマンドでは、DBID、FNR、NAME、CIPHER、および PASSWORD または PSW は無視されます。これらのパラメータは、代わりに *parameter-setting* 節で指定する必要があります（「*parameter-setting* のキーワードの説明」の「LOADNCP...」の項を参照）。

## FDT の選択

この選択は、処理する Adabas FDT（フィールド定義テーブル）を選択するために使用します。

**FDT** のロードの詳細については、「オブジェクト指定」セクションの「FDT」も参照してください。

以下のセクションでは、適切な構文を示して説明します。

### ■ FDT の選択の構文

#### FDT の選択の構文

```
FDT
DBID dbid
{ FNR fnr [CIPHER cipher] [ { PASSWORD } PSW password ] }
  FMFNR fnr-start TOFNR fnr-end
```

#### FDT の選択のキーワードの説明

次のセクションでは、処理する FDT のキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値
DBID	FDT のデータベース ID。	なし
FNR	FDT のファイル番号。	なし
CIPHER	FDT の 8 桁の Adabas サイファコード。	なし
PASSWORD または PSW	FDT の 8 文字の Adabas パスワード。	なし
FMFNR	FIND または UNLOAD コマンドにのみ適用されます。 開始値： FDT のファイル番号 (FNR)。	なし
TOFNR	FIND または UNLOAD コマンドにのみ適用されます。 終了値： FDT のファイル番号 (FNR)。	なし

## アプリケーションの選択

この選択は、Naturalスタジオのアプリケーションワークスペースで作成および管理されるアプリケーション、およびそのアプリケーションに属すライブラリまたはオブジェクトに適用されます。

以下のセクションでは、適切な構文を示して説明します。

- 基本アプリケーションと複合アプリケーションの選択
- アプリケーションライブラリの選択
- アプリケーションオブジェクトの選択

### 基本アプリケーションと複合アプリケーションの選択

この選択内容は検索機能にのみ適用されます。

構文

```
APPLICATION APNAME application-name
[APITYPE application-type]
[COMPAAPPLICATION compound-application-name]
[
    EXCEPT
    (APNAME application-name
    [APITYPE application-type]
    )]
```

### アプリケーションライブラリの選択

この選択内容は検索機能にのみ適用されます。

構文

```
APPLICATION APLIBRARY application-library-name
[BASEAPPLICATION base-application-name]
[ COMPAAPPLICATION compound-application-name ]
[
    DBIDdbid [FNRfnr] ]
[
    EXCEPT
    (APLIBRARY application-library-name
    [BASEAPPLICATION base-application-name]
    )]
```

## アプリケーションオブジェクトの選択

このセクションは、検索およびアンロード機能にのみ適用されます。

構文

```
APPLICATION APOBJECTS application-object-name
[BASEAPPLICATION base-application-name]
[COMPAPPLICATION compound-application-name]
[LIBRARY library-name]
[object-specification]
[
    EXCEPT
    (APOBJECT application-object-name
    [LIBRARY library-name]
    [BASEAPPLICATION base-application-name]
    [object-specification]
    )]
```

アプリケーションの選択のキーワードの説明

次のセクションでは、処理するアプリケーション、アプリケーションライブラリ、またはアプリケーションオブジェクトのキーワードおよび有効な入力値について説明します。

キーワード	有効な値	デフォルト値						
APNAME	Natural アプリケーションの有効な名前または名前の範囲。 「名前と日時の指定」の「名前」も参照してください。	*						
APTYPE	有効なアプリケーションタイプ： <table border="1" data-bbox="457 1377 1221 1591"> <tr> <td>B</td> <td>基本アプリケーション</td> </tr> <tr> <td>0</td> <td>複合アプリケーション</td> </tr> <tr> <td>*</td> <td>All：基本アプリケーションまたは複合アプリケーションあるいはその両方</td> </tr> </table>	B	基本アプリケーション	0	複合アプリケーション	*	All：基本アプリケーションまたは複合アプリケーションあるいはその両方	*
B	基本アプリケーション							
0	複合アプリケーション							
*	All：基本アプリケーションまたは複合アプリケーションあるいはその両方							
COMPAPPLICATION	APTTYPE が * または B に設定されている場合にのみ適用されます。 指定された基本アプリケーションが属す複合アプリケーションの名前または名前の範囲。 指定された複合アプリケーションに属す基本アプリケーションのみが選択され、複合アプリケーションに属さない基本アプリケーションは選択されません。	なし						



キーワード	有効な値	デフォルト値
EXCEPT	「 <i>Natural</i> ライブラリオブジェクトと DDM の選択」の「EXCEPT」の項を参照してください。	該当なし
APLIBRARY	Natural 基本アプリケーションまたは複合アプリケーションに属するライブラリの有効な名前または名前の範囲。  「名前と日時の指定」の「名前」も参照してください。	*
BASEAPPLICATION	アプリケーションライブラリまたはアプリケーションオブジェクトが属する Natural 基本アプリケーションの有効な名前。  「名前と日時の指定」の「名前」も参照してください。	*
DBID	アプリケーションライブラリの有効なデータベース ID。	0 (チェックなし)
FNR	アプリケーションライブラリの有効なファイル番号。	0 (チェックなし)
APOBJECT	基本アプリケーションまたは複合アプリケーションに属するアプリケーションオブジェクトの有効な名前または名前の範囲。  「名前と日時の指定」の「名前」も参照してください。	*
LIBRARY	有効なライブラリ名または名前の範囲。 OBJTYPE が D (「 <i>Natural</i> ライブラリオブジェクトと DDM の選択」を参照) に設定されている場合、ライブラリ名は無視されます。  「名前と日時の指定」の「名前」も参照してください。	*
<i>object-specification</i>	Natural ライブラリオブジェクトと DDM の構文図に示すように、アプリケーションオブジェクトに追加の選択条件を指定できることを示します。LIBRARY <i>library-name</i> の下に示しているすべての項目はアプリケーションオブジェクトにも適用できますが、EXCEPT 節の <i>object-name</i> はアプリケーションオブジェクトには無関係です。	該当なし

## 削除命令のためのオブジェクト選択

この選択は、Natural ライブラリオブジェクト、DDM、ユーザー定義エラーメッセージ、および Natural システムエラーメッセージに対する削除命令を指定する場合に使用します。この削除命令は、DELETEDALLOWED オプションを指定して、内部フォーマットのワークファイルがターゲット環境にロードされると実行されます。

次のセクションでは、適切な構文を示して説明します。

### ■ Natural ライブラリオブジェクトと DDM の削除命令の構文

- ユーザー定義エラーメッセージの削除命令の構文
- Natural システムエラーメッセージの削除命令の構文

## Natural ライブラリオブジェクトと DDM の削除命令の構文

```

object-name
LIBRARY library-name
[ OBJTYPE { N
           D } ]
[ NATTYPE { *
           V } ]
[SCKIND object-kind]

```

### Natural ライブラリオブジェクトと DDM の削除命令のキーワードの説明

次のセクションでは、処理するオブジェクトのキーワードおよび有効な値について説明します。

キーワード	有効な値	デフォルト値
<i>object-name</i>	有効なオブジェクト名、または名前範囲の開始値 ( <i>value</i> *) (例えば、ABC*)	なし
LIBRARY	有効なライブラリ名。範囲の指定はできません。  <i>OBJTYPE</i> (下記参照) が D に設定されている場合、ライブラリ名は無視されます。	なし
<i>OBJTYPE</i>	有効なオブジェクトタイプコード：  D DDM N Natural プログラミングオブジェクト  オブジェクトタイプ D をオブジェクトタイプ N と組み合わせることはできません。	*
<i>NATTYPE</i>	<i>OBJTYPE</i> が D に設定されている場合、適用されません。  Natural オブジェクトタイプ。有効な入力値は次のとおりです。  * すべてのオブジェクトタイプ V DDM (Windows、UNIX、および OpenVMS のみ)	*
<i>SCKIND</i>	<i>OBJTYPE</i> が D に設定されている場合、適用されません。  Natural プログラミングオブジェクトの種類。 有効な入力値は次のとおりです。	A

キーワード	有効な値	デフォルト値
	<p>S ソースオブジェクト。 <i>except-clause</i> で使用した場合（「<i>Natural</i> ライブラリオブジェクトと <i>DDM</i> の選択の構文」を参照）：ソース形式のみで保存されるオブジェクト。</p> <p>C カタログ化オブジェクト。 <i>except-clause</i> で使用した場合、カタログ形式のみで保存されるオブジェクト。</p> <p>A すべてのソースオブジェクトおよびカタログ化オブジェクト。</p>	

## ユーザー定義エラーメッセージの削除命令の構文

```
*
LIBRARY library-name
OBJTYPE E
FMNUM error-number-from
[TONUM error-number-to]
[SLKIND message-type]
[LANGUAGE languages]
```

*library-name* は、単一のライブラリの名前です。範囲の指定はできません。

この構文で使用される他の要素については、「*Natural* ライブラリオブジェクトと *DDM* の選択のキーワードの説明」を参照してください。

## Natural システムエラーメッセージの削除命令の構文

```
ERROR NATERROR
FMNUM error-number-from
[TONUM error-number-to]
[SLKIND message-type]
[LANGUAGE languages]
```

この構文で使用される要素については、「*Natural* ライブラリオブジェクトと *DDM* の選択のキーワードの説明」を参照してください。

## ヘルプテキストの選択

この選択は、ワークファイルに内部フォーマットが使用されている場合に、アンロード、ロード、スキャン、および検索の機能の実行中に Natural ヘルプテキストが処理されるように指定する場合に使用します。これらのヘルプテキストは、HELP システムコマンドで呼び出される Natural ヘルプユーティリティにより提供されるヘルプ情報と同じです（『システムコマンド』ドキュメントを参照）。

### ヘルプテキスト選択の構文

```
ERROR NATURAL HELP
[ DBID dbid FNR fnr [NAME vsam-name]
  [CIPHER cipher] [ { PASSWORD } password ] ]
```

この構文で使用される要素については、「[Natural ライブラリオブジェクトと DDM の選択のキーワードの説明](#)」を参照してください。

# 43 オブジェクトリスト - LIST ワークプラン

---

▪ object-type-and-location の構文 .....	240
▪ object-name-description の構文 .....	242
▪ オブジェクトリストの例 .....	243

オブジェクトリストとはタイプ LIST のワークプランのことであり、UNLOAD、LOAD、または FIND コマンドで処理するオブジェクトに対してオブジェクトの選択条件を指定します。オブジェクトリストは、*select-clause* および SELECTION ワークプランの代わりに使用できます。

オブジェクトリストに適用される構文は次のとおりです。

```
TYPE LIST
{ object-type-and-location (object-name-description ...) } ...
```

構文オプションについては、次のセクションで説明します。このセクションの構文図で使用しているキーワードおよび変数値は、オブジェクトの選択条件の指定に使用されるパラメータを表します。必要な場合、変数値をキーワードと一緒に指定する必要があります。各構文要素（カッコ内の構文要素を除く）は、新しい行で開始し、同じ行で終了する必要があります。

構文図で使用しているキーワードについては、「*select-clause*」セクションを参照してください。

## object-type-and-location の構文

---

次のセクションでは、*object-type-and-location* に適用される構文図を示します。

- Natural オブジェクトと DDM
- Natural システムエラーメッセージ
- Natural コマンドプロセッサソース
- Natural 関連デバッグ環境
- Natural 関連プロファイル
- Natural 関連 DLI サブファイル
- FDT

### Natural オブジェクトと DDM

```
LIBRARY library-name
[ DBID dbid FNR fnr [ NAME vsam-name ] [ CIPHER cipher ] [ { PASSWORD PSW } password ] ]
[ OBJTYPE group-type ]
```



#### 注意:

1. *library-name* には範囲は使用できません。

2. DDM では、OBJTYPE を D に設定する必要があります。

### Natural システムエラーメッセージ

```
ERROR NATERROR
[ DBID dbid FNR fnr [NAME vsam-name] [CIPHER cipher] [ { PASSWORD
  PSW } password ] ]
```

### Natural コマンドプロセッサソース

```
PROCESSOR ncp-library-name
[ DBID dbid FNR fnr [NAME vsam-name] [CIPHER [ { PASSWORD
  PSW } password ] ] ]
```

 **注意:** *ncp-library-name* には範囲は使用できません。

### Natural 関連デバッグ環境

```
NATPATH DEBUG
LIBRARY library-name
[ [ DBID dbid [FNR fnr] ] [NAME vsam-name] [CIPHER [ { PASSWORD
  PSW } password ] ] ]
```

### Natural 関連プロファイル

```
NATPATH PROFILE
[ [ DBID dbid [FNR fnr] ] [NAME vsam-name] [CIPHER [ { PASSWORD
  PSW } password ] ] ]
```

## Natural 関連 DLI サブファイル

NATPATH SUBFILE

```
[ [ DBID dbid
  [FNR fnr ] ] [NAME vsam-name] [CIPHER cipher] [ { PASSWORD } password ] ]
```

## FDT

FDT

## object-name-description の構文

---

次のセクションでは、*object-name-description* に適用される構文図を示します。

- Natural オブジェクト
- DDM
- Natural システムエラーメッセージ
- Natural コマンドプロセッサソース
- Natural 関連デバッグ環境
- Natural 関連プロファイル
- Natural 関連 DLI サブファイル
- FDT

## Natural オブジェクト

```
object-name [SCKIND object-kind]
{
  error-number [SLKIND message-type] [LANGUAGE languages]
  FMNUM error-number-from TONUM error-number-to [SLKIND message-type] [LANGUAGE
  languages]
}
```

## DDM

*object-name*



## Natural システムエラーメッセージ

```

error-number [SLKIND message-type] [LANGUAGE languages]
{ FMNUM error-number-from TONUM error-number-to [SLKIND message-type] [LANGUAGE languages] }

```

## Natural コマンドプロセッサソース

```
object-name
```

## Natural 関連デバッグ環境

```
object-name
```

## Natural 関連プロファイル

```
object-name [OBJTYPE profile-type]
```

## Natural 関連 DLI サブファイル

```
object-name [OBJTYPE subfile-type]
```

## FDT

```
DBID dbidFNR fnr [CIPHER cipher] [ { PASSWORD } password ]
      [ { PSW } ]
```

## オブジェクトリストの例

次に、タイプ LIST のワークプランの例を示します。

```

TYPE LIST
LIBRARY LIB-1 OBJTYPE N      /* process Natural objects from library 'LIB-1'
( A* SCKIND S                /* all sources objects whose names start with 'A'
B1                            /* source and/or cataloged object of 'B1'
CDE> SCKIND C )             /* all cataloged objects with names greater than/equal
to 'CDE'
/*                            /* comment line
LIBRARY LIB-2                /* process Natural objects from library 'LIB-2'
/*                            /* including error messages and shared resources

```

## オブジェクトリスト - LIST ワークプラン

---

```
( *                               /* all source and/or cataloged objects
                                /* including shared resources
FMNUM 1 TONUM 100                /* error messages from 1 to 100
)
```

# 44 parameter-setting

---

- parameter-setting の構文 ..... 246
- parameter-setting のキーワードの説明 ..... 247

*parameter-setting* 節は、オブジェクトを処理する LOAD または UNLOAD コマンドの属性を変更し、LOAD コマンドの出力宛先（FNAT など）を定義するために使用します。

以下のセクションでは、*parameter-setting* 節に適用される構文を示して説明します。構文図で使用しているキーワードおよび変数値は、属性の変更に使用されるパラメータを表します。必要な場合、変数値をキーワードと一緒に指定する必要があります。

## parameter-setting の構文

```
WITH
{
  (parameter-workplan)
  parameter-clause
}
```

*parameter-workplan* に適用される構文については、「ワークプラン」セクションの「[ワークプランの参照](#)」を参照してください。

以下のセクションでは、*parameter-clause* の構文を示します。

### parameter-clause の構文

#### parameter-clause の構文

```
[ [NAME old-name] NEWNAME new-name ]
[ [LIBRARY old-library-name]
  NEWLIBRARY new-library-name ]
[ LOADFNATDBID fnat-dbid LOADFNATFNR fnat-fnr
  [LOADFNATNAME vsam-name]
  [LOADFNATCIPHER fnat-cipher]
  [ { LOADFNATPASSWORD
      LOADFNATPSW } fnat-password ] ]
[ LOADFUSERDBID fuser-dbid LOADFUSERFNR fuser-fnr
  [LOADFUSERNAME fuser-vsam-name]
  [LOADFUSERCIPHER fuser-cipher]
  [ { LOADFUSERPASSWORD
      LOADFUSERPSW } fuser-password ] ]
[ LOADNCPDBID ncp-file-dbid LOADNCPFNR ncp-file-fnr
  [LOADNCPNAME ncp-file-vsam-name]
  [LOADNCPCIPHER ncp-file-cipher]
  [ { LOADNCPPASSWORD
      LOADNCPPSW } ncp-file-password ] ]
```

[	[ <a href="#">FDTDBID</a> <i>old-fdt-dbid</i> <a href="#">FDTFNR</a> <i>old-fdt-fnr</i> ] <a href="#">NEWFDTDBID</a> <i>new-fdt-dbid</i> <a href="#">NEWFDTFNR</a> <i>new-fdt-fnr</i>	]
[	<a href="#">ERRNUMDIFF</a> <i>modification-of-error-message-range</i>	
[	[ <a href="#">LANGUAGE</a> <i>old-language</i> ]	]
[	<a href="#">NEWLANGUAGE</a> <i>new-language</i>	]
[	[ <a href="#">DATE</a> <i>old-date</i> ] <a href="#">NEWDATE</a> <i>new-date</i>	]
[	[ <a href="#">USERID</a> <i>old-userid</i> ] <a href="#">NEWUSERID</a> <i>new-userid</i>	]
[	[ <a href="#">TID</a> <i>old-terminal-id</i> ] <a href="#">NEWTID</a> <i>new-terminal-id</i>	]
[	[ <a href="#">PATH</a> <i>old-external-path-name</i> ] <a href="#">NEWPATH</a> <i>new-external-path-name</i>	]

## parameter-setting のキーワードの説明

次のセクションでは、*parameter-setting* のキーワードおよび変数値（該当する場合）について説明します。

キーワード	値	利用可能なコマンド
NAME	NEWNAME が指定されている場合にチェックされるオブジェクト名。	
NEWNAME	新しいオブジェクト名。	
LIBRARY	NEWLIBRARY が指定されている場合にチェックされるライブラリ名。	
NEWLIBRARY	新しいライブラリ名。  <b>LOAD</b> 機能に関する注意事項：  NEWLIBRARY は、 <a href="#">DELETEALLOWED</a> オプションを使用して処理されたワークファイルの削除命令で使用されるライブラリ名には影響を及ぼしません。	
LOADFNATDBID	FNAT ライブラリのデータベース ID (DBID)。	LOAD
LOADFNATFNR	FNAT ライブラリのファイル番号 (FNR)。	LOAD
LOADFNATNAME	メインフレーム上のオブジェクトにのみ適用されます。 FNAT の VSAM ファイル名。	LOAD
LOADFNATCIPHER	メインフレーム上のオブジェクトにのみ適用されます。 FNAT のサイファコード。	LOAD
LOADFNATPASSWORD	メインフレーム上のオブジェクトにのみ適用されます。 FNAT の Adabas パスワード。	LOAD
LOADFUSERDBID	FUSER ライブラリの DBID。	LOAD

キーワード	値	利用可能なコマンド
LOADFUSERFNR	FUSER ライブラリの FNR。	LOAD
LOADFUSERNAME	メインフレーム上のオブジェクトにのみ適用されます。 FUSER の VSAM ファイル名。	LOAD
LOADFUSERCIPHER	メインフレーム上のオブジェクトにのみ適用されます。 FUSER のサイファコード。	LOAD
LOADFUSERPASSWORD	メインフレーム上のオブジェクトにのみ適用されます。 FUSER の Adabas パスワード。	LOAD
LOADNCPDBID	Natural コマンドプロセッサソース用 Adabas ファイルの DBID。	LOAD
LOADNCPFNR	Natural コマンドプロセッサソース用 Adabas ファイルの FNR。	LOAD
LOADNCPNAME	メインフレーム上のオブジェクトにのみ適用されます。 Natural コマンドプロセッサソース用 Adabas ファイルの VSAM 名。	LOAD
LOADNCPCIPHER	メインフレーム上のオブジェクトにのみ適用されます。 Natural コマンドプロセッサソース用 Adabas ファイルのサイファコード。	LOAD
LOADNCPPASSWORD	メインフレーム上のオブジェクトにのみ適用されます。 Natural コマンドプロセッサソース用 Adabas ファイルの Adabas パスワード。	LOAD
FDTDBID	NEWFDTDBID が指定されている場合にチェックされる Adabas FDT (フィールド定義テーブル) の DBID。	
NEWFDTDBID	FDT の新しい DBID。	
FDTFNR	NEWFDTFNR が指定されている場合にチェックされる FDT の DBID。	
NEWFDTFNR	FDT の新しい FNR。	
ERRNUMDIFF	UNLOAD または LOAD コマンドの実行中に Natural エラーメッセージに追加される数字 (正または負)。  選択条件として FMNUM および TONUM (「select-clause」を参照) が指定された場合にのみ、ERRNUMDIFF を指定できます。指定されていない場合は、有効な結果をチェックできません。	
LANGUAGE	NEWLANGUAGE (下記参照) が指定されている場合にチェックされる Natural エラーメッセージの最大 8 つまでの有効な言語コード (英語の場合はコード 1)。  <i>language</i> に複数の言語コードが含まれている場合、 <i>new-language</i> には同じ数の言語コードが含まれている必要があります。各 <i>language</i> 言語コードは、対応する位置の <i>new-language</i> で置き換えられます。  <i>language</i> が指定されていない場合、 <i>new-language</i> に複数の言語コードを含めることはできません。	
NEWLANGUAGE	新しいユーザー定義エラーメッセージの最大 8 つまでの有効な言語コード (スペイン語の場合はコード 4)。このオプションは、英語 (言語	

キーワード	値	利用可能なコマンド
	コード 1) が唯一の有効な言語である長文テキストの Natural システム エラーメッセージには適用されません。  上記の「LANGUAGE」も参照してください。	
DATE	オブジェクト日付。  時刻を追加するには、日付と時刻の間に空白を挿入します。有効なフォーマットと範囲については、「名前と日時の指定」の「日付」および「時刻」を参照してください。	
NEWDATE	新しいオブジェクト日付。  NEWDATE には、日付の後に時刻を続けることができます。時刻を追加するには、日付と時刻の間に空白を挿入します。「名前と日時の指定」の「日付」および「時刻」も参照してください。	
USERID	NEWUSERID が指定されている場合にチェックされるユーザー ID。	
NEWUSERID	新しいユーザー ID。	
TID	メインフレーム上のオブジェクトにのみ適用されます。 NEWTID が指定されている場合にチェックされる端末 ID。	
NEWTID	メインフレーム上のオブジェクトにのみ適用されます。 新しい端末 ID。	
PATH	NEWPATH が指定されている場合にチェックされるパス名。	
NEWPATH	新しいパス名。	



#### 注意:

1. 処理される選択条件に適用できないパラメータは無視されます。
2. LOADFNAT...、LOADFUSER...、および LOADNCP... は、LOAD コマンドにのみ使用され、他のコマンドでは無視されます。
3. LOADFNAT... は、SYS で始まるライブラリ (SYSTEM を除く) に対して使用されます。
4. LOADFUSER... は、SYS で始まるライブラリを除くライブラリ (SYSTEM は含まれる) に対して使用されます。
5. LOADNCP... は、Natural コマンドプロセッサソースに対して使用されます。





# 45 option-setting

---

- option-setting の構文 ..... 252
- option-setting のキーワードの説明 ..... 254

*option-setting* 節は、オブジェクトハンドラコマンドオプションのデフォルト値を変更するために使用します。

以下のセクションでは、*option-setting* 節に適用される構文を示して説明します。構文図で使用しているキーワードおよび変数値は、デフォルト値の指定に使用されるパラメータを表します。必要な場合、変数値をキーワードと一緒に指定する必要があります。

## option-setting の構文

```
WHERE
{
  (option-workplan)
  option-clause
}
```

*option-workplan* に適用される構文図については、「ワークプラン」セクションの「[ワークプランの参照](#)」を参照してください。

以下のセクションでは、*option-clause* の構文を示します。

### ■ option-clause の構文

#### option-clause の構文

```
[ REPLACE { ALL
             OBSOLETE
             EXCEPT } ]
[ transfer-options
  internal-format-options ]
[ NOREPORT
  NEWREPORT [file-name]
  REPORT [file-name]
  BATCHREPORT ]
[ NORESTART
  RESTART [restart-text-member] ]
[ NUMBERPROCESS number ]
[ FIXEDLENGTH ]
[ FDIC (dbid,fnr,password,cipher) ]
[ FSEC (dbid,fnr,password,cipher) ]
[ { WORKFILETYPE
     WETYPE } { DEFAULT
                UTF-8 } ]
```

[ { PC NEWPC } ]	[file-name]	]
[ { PCCOMMANDFILENUMBER PCCOM PCCFN } ]	command-file-number	]

### セパレータ

FDICおよびFSECキーワードに続く各値の間に、または値がない場合に、セパレータとしてコマを使用する必要があります。例えば、FDIC (10,21,,2a)のように使用します。

セッションパラメータ ID (『パラメータリファレンス』ドキュメントの「ID-INPUT 区切り文字」を参照) がコマに設定されている場合、値を区切るセパレータとしてスラッシュ (/) を使用します。

### transfer-options

<u>TRANSFER</u>			
[	<u>CONVERSION-TABLE</u> {	<u>SYSTEM-TABLE</u> <u>USER-TABLE</u> [conversion-program]	}]
<u>[SUBSTITUTE]</u>			
<u>[INCLUDE-LINE-NUMBERS]</u>			
<u>[UPPERCASE-TRANSLATION]</u>			
<u>[INCORPORATE-FREE-RULES]</u>			
<u>[LOAD-CODE-PAGE</u> code-page-name]			
<u>[DA-FORMAT</u> data-area-format]			

### internal-format-options

[	XREF {	ON OFF DOC FORCE SPECIAL	}]
<u>[DELETEALLOWED]</u>			
<u>[NOSYMBOLTABLE]</u>			
<u>[VERSIONCHECK]</u>			

## option-setting のキーワードの説明

次のセクションでは、*option-setting* のキーワードおよび変数値（該当する場合）について説明します。

オプション	説明	利用可能なコマンド
REPLACE	指定されたオプションに従って、既存のオブジェクトを置き換えます。  ALL       すべてのオブジェクト（デフォルト設定）。 OBSOLETE   ロードファイル内のオブジェクトの日付より古い日付になっているすべてのオブジェクト。 EXCEPT   ロードファイル内のオブジェクトの日付より新しい日付になっているオブジェクトを除いた、すべてのオブジェクト。	LOAD LOADALL
TRANSFER	転送モードを設定します。  データは、転送フォーマットで読み込みおよび書き込みが行われます。有効なオプションについては、「 <a href="#">transfer-options のキーワードの説明</a> 」を参照してください。	UNLOAD LOAD SCAN
NOREPORT	以下のレポートファイル設定を指定します。  データはレポートファイルに記録されません。これは、FIND および FINDLIB コマンドに対するデフォルト設定です。	
NEWREPORT	以下のレポートファイル設定を指定します。  レポートデータは記録され、ワークプランライブラリに保存されている Natural テキストメンバに書き込まれます。既存のファイルは上書きされます。	
REPORT	以下のレポートファイル設定を指定します。  レポートデータは記録され、ワークプランライブラリに保存されている Natural テキストメンバに書き込まれます。これは、UNLOAD、LOAD、LOADALL、SCAN、SCANALL、および DELETE コマンドに対するデフォルト設定です。	
BATCHREPORT	バッチ処理時または OBJHAPI アプリケーションプログラミングインターフェイスの使用時のレポート設定を指定します。  レポートデータは、それぞれ SYSOUT に書き込まれるか、または画面に出力されます（レポートデータはファイルに書き込まれません）。	
NORESTART	再起動情報はファイルに書き込まれません。	LOAD

オプション	説明	利用可能な コマンド
RESTART	再スタート情報は、ワークプランライブラリに保存されている Natural テキストメンバに書き込まれます。	LOAD
NUMBERPROCESS	処理するオブジェクトの数を指定します。  指定された数を処理した後、LOAD または SCAN コマンドは停止します。	LOAD SCAN
FIXEDLENGTH	アンロードワークファイルのフォーマットを固定サイズの最大レコード長に設定します。  内部フォーマットで書き込まれる場合、各データレコードには 256 バイトが含まれ、転送フォーマットでは 100 バイトが含まれます。	UNLOAD
FDIC	処理に使用するシステムファイル FDIC を指定します。  Adabas ファイルのデータベース ID ( <i>dbid</i> )、ファイル番号 ( <i>fnr</i> )、パスワード ( <i>password</i> )、およびサイファコード ( <i>cipher</i> )。  値が指定されていない場合、または「0」が指定されている場合、現在の FDIC システムファイルが使用されます。	UNLOAD LOAD DELETE
FSEC	処理に使用するシステムファイル FSEC を指定します。  Adabas ファイルのデータベース ID ( <i>dbid</i> )、ファイル番号 ( <i>fnr</i> )、パスワード ( <i>password</i> )、およびサイファコード ( <i>cipher</i> )。  値が指定されていない場合、または 0 が指定されている場合、現在の FSEC システムファイルが使用されます。	UNLOAD LOAD DELETE
WORKFILETYPE または WFTYPE	データがワークファイルに対して読み込みおよび書き込みされる ときの Natural ワークファイル 1 のワークファイルタイプ。  DEFAULT デフォルトのバイナリワークファイル。 UTF-8 Unicode/UTF-F8 でエンコードされたバイナリワーク ファイル。  UTF-8 は、TRANSFER が指定されている場合にアンロー ド機能にのみ適用されます。  UTF-8 が指定されている場合、CONVERSION-TABLE、 SUBSTITUTE、および INCORPORATE-FREE-RULES オプ ションは使用できません。  WORKFILETYPE が指定されていない場合、現在のタイプが使用され ます。	UNLOAD

オプション	説明	利用可能な コマンド
PC NEWPC	<p>適用されるのは、Entire Connectionがインストールされている場合に 限られます。</p> <p>Entire Connection ワークファイルに対してデータの書き込みまたは 読み込みを行います。 <i>file-name</i> は、Entire Connection ワーク ファイルに割り当てられている完全パス名を表します。現在のシス テム環境でバックスラッシュ (\) セパレータを使用できない場合 は、代わりにスラッシュ (/) を使用します。 <i>file-name</i> を指定し ないと、ワークファイル名の入力を求めるプロンプトが Entire Connection から表示されます。</p> <p>NEWPCが指定されると、既存のワークファイルの内容がアンロード したデータで上書きされるか、または新しいワークファイルが先頭 から埋められます。これに該当しない場合、データが追加されま す。</p> <p>「ワークファイル」の「<a href="#">ワークファイルの割り当て</a>」も参照してく ださい。</p>	UNLOAD LOAD SCAN
PCCOMMANDFILENUMBER または PCCOM または PCCFN	<p>適用されるのは、Entire Connectionがインストールされている場合に 限られます。</p> <p>Entire Connection コマンドの処理に使用されるワークファイルの番 号を指定します。</p> <p>デフォルト値はワークファイル8の8であり、このファイルを Entire Connection ワークファイルとして定義する必要があります。</p> <p>「ワークファイル」の「<a href="#">ワークファイルの割り当て</a>」も参照してく ださい。</p>	UNLOAD LOAD SCAN

次のセクションでは、*transfer-options* と *internal-format-options* のキーワードおよび変数  
値（該当する場合）について説明します。

- [transfer-options のキーワードの説明](#)
- [internal-format-options のキーワードの説明](#)

## transfer-options のキーワードの説明

TRANSFER キーワードを使用する場合、次のオプションを指定できます。

オプション	説明	利用可能なコマンド
CONVERSION-TABLE	次のいずれかの変換テーブルを使用して、転送フォーマットで処理されたデータを変換します。  SYSTEM-TABLE :  内部的な Natural 変換テーブル。  USER-TABLE :  <i>conversion-program</i> が指定されている場合のユーザー定義変換テーブル。このプログラムは、ライブラリ SYSOBJH またはその <i>steplib</i> の1つに保存されている必要があります。ライブラリ SYSOBJH にあるプログラム例 OTNCONAE および OTNCONEA を参照してください。	UNLOAD LOAD SCAN
SUBSTITUTE	転送フォーマットでのアンロード中に行参照をラベルで置き換えます。  このオプションは、ソースコードの行番号がステートメント参照に使用されている場合にのみ適用されます。その場合、参照された行の行番号および行番号参照はラベルで置き換えられます。ソースはデータベースでは変更されません。	UNLOAD
INCLUDE-LINE-NUMBERS	転送フォーマットでのアンロード中に行番号を転送します。デフォルトでは、Natural オブジェクトの行番号はアンロードされません。	UNLOAD
UPPERCASE-TRANSLATION	転送フォーマットでのロード中にソースコードを大文字に変換します。デフォルトでは、Natural オブジェクトのソースコードは変換されません。	LOAD
INCORPORATE-FREE-RULES	Predictがインストールされている場合、転送フォーマットでのアンロード中に、マップに連結されている Predict フリールールのソーステキストをマップソースに組み込みます。	UNLOAD
LOAD-CODE-PAGE	Unicode/UTF-8 (Universal Transformation Format、8ビット形式) でエンコードされたオブジェクトソースの変換に使用されるコードページを指定します。  このオプションを使用すると、UTF-8 でワークファイルにアンロードされたすべてのオブジェクトソースは、ワークファイルにロードされるときに指定のコードページで変換されます。  <i>code-page-name</i> として *CODEPAGE を指定するか、または <i>code-page-name</i> が指定されていないと、システム変数	LOAD LOADALL

オプション	説明	利用可能なコマンド
	*CODEPAGEに割り当てられている値が使用されます（『システム変数』ドキュメントを参照）。  LOAD-CODE-PAGEが指定されている場合、CONVERSION-TABLEおよびUPPERCASE-TRANSLATIONオプションは使用できません。	
DA-FORMAT	データエリアソースのフォーマット変換を指定します。「設定」の「転送オプション」にある「 <b>Data area format</b> 」を参照してください。	UNLOAD LOAD

### internal-format-options のキーワードの説明

*internal-format-options* を使用する場合、次のオプションを指定できます。

オプション	説明	利用可能なコマンド
XREF	適用されるのは、Predict がインストールされている場合に限られます。  カタログされた Natural オブジェクトの XRef データをロードまたはアンロードします。次の値のいずれかを指定できます。	LOAD UNLOAD
ON	UNLOAD : カタログ化オブジェクトとそのクロスリファレンスデータ（存在する場合）をアンロードします。  LOAD : クロスリファレンスがワークファイルに存在する場合は、カタログ化オブジェクトとそのクロスリファレンスデータをロードします。	
OFF	XRef データは処理されません。これがデフォルトです。	
DOC	LOAD にのみ適用されます。  カタログ化オブジェクトをロードします。ただしそれは、そのオブジェクトの Predict エントリが FDIC システムファイル内に存在する場合に限られます。	
FORCE	LOAD にのみ適用されます。  カタログ化オブジェクトとそのクロスリファレンスデータをロードします。ただしそれは、クロスリファレンス	



オプション	説明		利用可能なコマンド
		がワークファイル内に存在し、そのオブジェクトの Predict エントリが FDIC システムファイル内に存在する場合には限られます。	
	SPECIAL	LOAD にのみ適用されます。 カタログ化オブジェクトとそのクロスリファレンスデータ（存在する場合）をロードします。	
DELETEALLOWED	オブジェクトを内部フォーマットでロードするときに、ユーティリティ NATUNLD によって作成されたワークファイルから削除命令を処理します。		LOAD
NOSYMBOLTABLE	<p>カタログされた Natural ライブラリオブジェクトを、対応する内部的な Natural シンボルテーブルなしでアンロードします。</p> <p>これにより、必要なディスクストレージの量が削減されます。ただし、これは実稼動環境においてのみ役立ちます。実稼動環境では、シンボルテーブルを必要とする複数のアプリケーション開発機能が使用できず、またプロファイルパラメータ RECAT=ON（『パラメータリファレンス』ドキュメントを参照）が適用されないからです。</p>		UNLOAD
VERSIONCHECK	<p>ロードするカタログ化オブジェクトの Natural バージョンをチェックします。オブジェクトをカタログしワークファイルに書き込んだ Natural バージョンが現在の Natural バージョンと比較されます。現在よりも上位のバージョンの Natural でカタログされたオブジェクトは拒否されます。</p> <p>VERSIONCHECK は、データを内部フォーマットでロードする場合（つまり、TRANSFER オプションが指定されていない場合）にのみ使用できます。</p>		LOAD



## 46      ダイレクトコマンドの使用例

---

- 同一プラットフォームのオブジェクトのアンロード ..... 262
- 異なるプラットフォームのオブジェクトのアンロード ..... 263
- 内部フォーマットでのオブジェクトのロード ..... 264
- 転送フォーマットでのオブジェクトのロード ..... 264

このセクションでは、オブジェクトハンドラのダイレクトコマンドの使用例を示します。



**ヒント:** 追加例については、オブジェクトハンドラ機能用に生成されたコマンドを表示できます。このコマンドは、ウィザードを使用するときに自動的に表示されます。上級ユーザーモードでは、オブジェクトハンドラコマンド SET ADVANCEDCMD ONを入力するか、オブジェクトハンドラプロファイルでパラメータ Display-Cmd-in-Advanced-Mode を Y (Yes) に設定する（「[プロファイルの設定](#)」も参照）ことによって、コマンド表示を有効にすることができます。

## 同一プラットフォームのオブジェクトのアンロード

---

このセクションでは、ローカルメインフレーム、UNIX、OpenVMS、または Windows のいずれかの環境で、オブジェクトを同一プラットフォームでロードできるようにするために、オブジェクトを内部フォーマットでワークファイルにアンロードする例を示します。

- ライブラリ ABC からすべての Natural プログラミングオブジェクト（ソースオブジェクトのみ）をアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N SCKIND S
```

- ライブラリ ABC からすべての Natural プログラミングオブジェクト（カタログ化オブジェクトのみ）をアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N SCKIND C
```

- ライブラリ ABC からすべての Natural プログラミングオブジェクト（カタログ化オブジェクトとソースオブジェクト）をアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N SCKIND A
```

- ライブラリ ABC からすべての Natural プログラミングオブジェクト（ソースオブジェクトのみ）をアンロードし、ライブラリ ABCNEW にロードします。

```
UNLOAD * LIB ABC OBJTYPE N SCKIND S WITH NEWLIBRARY ABCNEW
```

- メインフレームで、名前が EMP で始まりデータベース 88 をポイントしているすべての DDM をアンロードします。

```
UNLOAD EMP* LIB * OBJTYPE D DDMBID 88
```

- UNIX、OpenVMS、または Windows で、名前が EMP で始まりデータベース 88 をポイントしているすべての DDM をアンロードします。

```
UNLOAD EMP* LIB * OBJTYPE N NATTYPE V DDMBID 88
```

- UNIX、OpenVMS、または Windows で、名前が EMP で始まるすべての DDM をライブラリ VLIB からアンロードし、ライブラリ VLIBNEW にロードします。

```
UNLOAD EMP* LIB VLIB OBJTYPE N NATTYPE V WITH NEWLIBRARY VLIBNEW
```

- すべてのユーザー定義エラーメッセージをライブラリ ERRLIB からアンロードし、ライブラリ NEWERR にロードします。

```
UNLOAD * LIB ERRLIB OBJTYPE E SLKIND A WITH NEWLIBRARY NEWERR
```

- Windows で、すべての Natural プログラミングオブジェクト（カタログ化オブジェクトとソースオブジェクト）をライブラリ ABC から PC 上のポータブルワークファイルにアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N WHERE WORKFILE C:\WF1.SAG WORKFILETYPE PORTABLE
```

または

```
UNLOAD * LIB ABC OBJTYPE N WHERE WORK C:\WF1.SAG WFT P
```

## 異なるプラットフォームのオブジェクトのアンロード

このセクションでは、メインフレームでアンロードして UNIX、OpenVMS、または Windows 環境でロードするなど、オブジェクトを異なるプラットフォームでロードできるようにするために、オブジェクトを転送フォーマットでワークファイルにアンロードするコマンド例を示します。

- ライブラリ ABC からすべての Natural プログラミングオブジェクト（ソースオブジェクトのみ）をアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N WHERE TRANSFER
```

- すべての Natural プログラミングオブジェクト（ソースオブジェクトのみ）とユーザー定義エラーメッセージをライブラリ ABC からアンロードします。

```
UNLOAD * LIB ABC WHERE TRANSFER
```

- すべての Natural プログラミングオブジェクト（ソースオブジェクトのみ）を固定レコード長のライブラリ ABC からアンロードします。

```
UNLOAD * LIB ABC OBJTYPE N WHERE TRANSFER FIXEDLENGTH
```

## 内部フォーマットでのオブジェクトのロード

---

このセクションでは、オブジェクトを内部フォーマットでワークファイルからロードするコマンド例を示します。

- すべてのオブジェクトをライブラリ LIBNEW にロードして、すでに存在しているオブジェクトと置き換えます。

```
LOADALL WITH NEWL LIBNEW WHERE REPLACE ALL
```

- ターゲットライブラリ TGTLIB のすべてのオブジェクトを新しいターゲットライブラリ NEWTGT にロードします。

```
LOAD * LIB TGTLIB WITH NEWLIBRARY NEWTGT
```

- ユーザー定義エラーメッセージ 1000~1500 のみをライブラリ ERRLIB からロードします。

```
LOAD * LIB ERRLIB OBJTYPE E FMNUM 1000 TONUM 1500
```

## 転送フォーマットでのオブジェクトのロード

---

このセクションでは、オブジェクトを転送フォーマットでワークファイルからロードするコマンド例を示します。

- すべてのオブジェクトをライブラリ LIBNEW にロードして、すでに存在しているオブジェクトと置き換えます。

```
LOADALL WITH NEWL LIBNEW WHERE TRANSFER REPLACE ALL
```

- ターゲットライブラリ TGTLIB のすべてのオブジェクトを新しいターゲットライブラリ NEWTGT にロードします。

```
LOAD * LIB TGTLIB WITH NEWLIBRARY NEWTGT WHERE TRANSFER
```

# 47 ナビゲーションと特別な機能のためのコマンド

CUI（キャラクタユーザーインターフェイス）環境におけるオブジェクトハンドラコマンドは、ナビゲーション、トレースファイルの指定などの特別な機能設定を行うことを主な目的としています。

オブジェクトハンドラコマンドは、任意のオブジェクトハンドラ画面のコマンド行に入力されます。オブジェクトハンドラ画面から Natural システムコマンドを実行するには、コマンドの前に2つのスラッシュ（//）を入力します。Natural システムコマンドによってオブジェクトハンドラが終了することに注意してください。

## ▶手順 47.1. オブジェクトハンドラの [Commands] メニューを表示するには

- PF10（Cmds）キーを押します。

または:

オブジェクトハンドラの任意の画面で、コマンド行に次のコマンドを入力します。

```
CMDS
```

オブジェクトハンドラコマンドを以下に示します。キーワードの下線部分は、入力可能な省略形を表します。Sub はサブコマンドを示します。

コマンド	サブコマンド1	サブコマンド2	説明
CANCEL			現在の機能をキャンセルし、オブジェクトハンドラのメインメニューを表示します。
<u>C</u> HANGE	<u>W</u> ORKPLAN	<u>L</u> IBRARY	管理機能を呼び出し、ワークプランライブラリを変更できる画面を表示します。
<u>C</u> LLEAR			マップ内の入力フィールドの現在の内容をデフォルト値にリセットします。
<u>C</u> MDS			[Commands] 画面を表示します。

## ナビゲーションと特別な機能のためのコマンド

コマンド	サブコマンド 1	サブコマンド 2	説明
または COMMANDS			
BYE			オブジェクトハンドラを終了します。
EXIT			
QUIT			
.			
FIN			オブジェクトハンドラを終了し、Natural セッションを終了します。
GO	HOME		オブジェクトハンドラのメインメニューを表示します。
GO	UNLOAD		アンロード機能呼び出します。
GO	UNLOAD	END	現在のアンロード機能を終了します。
		ERROR	Natural システムエラーメッセージのアンロード機能呼び出します。
		DDM	DDM のアンロード機能呼び出します。
		FDT	FDT のアンロード機能呼び出します。
		LIBRARY	Natural ライブラリオブジェクトのアンロード機能呼び出します。
		NCP	Natural コマンドプロセッサソースのアンロード機能呼び出します。
		RELATED	Natural 関連オブジェクトのアンロード機能呼び出します。
		SELECTION または LIST	アンロード機能で使用するために SELECTION または LIST ワークプランを入力または選択できる画面を表示します。
GO	LOAD		ロード機能呼び出します。
GO	LOAD	ALL	ワークファイルに含まれているすべてのオブジェクトのロード機能呼び出します。
		END	現在のロード機能を終了します。
		ERROR	Natural システムエラーメッセージのロード機能呼び出します。
		DDM	DDM のロード機能呼び出します。
		FDT	FDT のロード機能呼び出します。
		LIBRARY	Natural ライブラリオブジェクトのロード機能呼び出します。
		NCP	Natural コマンドプロセッサソースのロード機能呼び出します。
RELATED	Natural 関連オブジェクトのロード機能呼び出します。		



コマンド	サブコマンド 1	サブコマンド 2	説明
		<u>S</u> LECTION または <u>L</u> IST	ロード機能で使用するために SELECTION または LIST ワークプランを入力または選択できる画面を表示します。
<u>G</u> O	<u>R</u> ESTART		ロードの再スタート機能で使用するためにテキストメンバを指定できる画面を表示します。
<u>G</u> O	<u>S</u> CAN		スキャン機能呼び出します。
<u>G</u> O	<u>S</u> CAN	<u>A</u> LL	ワークファイルに含まれているすべてのオブジェクトのスキャン機能呼び出します。
		<u>E</u> ND	現在のスキャン機能を終了します。
		<u>E</u> RROR	Natural システムエラーメッセージのスキャン機能呼び出します。
		<u>D</u> DM	DDM のスキャン機能呼び出します。
		<u>F</u> DT	FDT のスキャン機能呼び出します。
		<u>L</u> IBRARY	Natural ライブラリオブジェクトのスキャン機能呼び出します。
		<u>N</u> CP	Natural コマンドプロセッサソースのスキャン機能呼び出します。
		<u>R</u> ELATED	Natural 関連オブジェクトのスキャン機能呼び出します。
		<u>S</u> LECTION または <u>L</u> IST	スキャン機能で使用するために SELECTION または LIST ワークプランを入力または選択できる画面を表示します。
<u>G</u> O	<u>A</u> DMIN		管理機能呼び出します。
<u>G</u> O	<u>A</u> DMIN	<u>C</u> HANGE	ワークプランライブラリを変更できる画面を表示します。
		<u>C</u> REATE	ワークプランを作成できるメニューを開きます。
		<u>L</u> IST	ワークプランライブラリで使用可能なワークプランのリストを生成します。
<u>G</u> O	<u>V</u> IEW		表示機能呼び出します。
<u>G</u> O	<u>V</u> IEW	<u>D</u> DM	DDM の表示機能呼び出します。
		<u>E</u> RROR	Natural システムエラーメッセージの表示機能呼び出します。
		<u>F</u> DT	FDT の表示機能呼び出します。
		<u>L</u> IBRARY	Natural ライブラリオブジェクトの表示機能呼び出します。
		<u>N</u> CP	Natural コマンドプロセッサソースの表示機能呼び出します。
		<u>R</u> ELATED	Natural 関連オブジェクトの表示機能呼び出します。
<u>G</u> O	<u>F</u> IND		検索機能呼び出します。
<u>G</u> O	<u>F</u> IND	<u>D</u> DM	DDM の検索機能呼び出します。

## ナビゲーションと特別な機能のためのコマンド

コマンド	サブコマンド1	サブコマンド2	説明
		<u>E</u> RROR	Natural システムエラーメッセージの検索機能呼び出します。
		<u>F</u> DT	FDT の検索機能呼び出します。
		<u>L</u> IBRARY	Natural ライブラリオブジェクトの検索機能呼び出します。
		<u>N</u> CP	Natural コマンドプロセッサソースの検索機能呼び出します。
		<u>R</u> ELATED	Natural 関連オブジェクトの検索機能呼び出します。
		<u>S</u> ELECTION または <u>L</u> IST	検索機能で使用するために SELECTION または LIST ワークプランを入力または選択できる画面を表示します。
HELP			オブジェクトハンドラのヘルプ機能呼び出します。
INIT			オブジェクトハンドラユーティリティを再初期化します。
READ	<u>P</u> ROFILE		テキストメンバPROFILEに定義されているように、オブジェクトハンドラの設定を更新します（「 <a href="#">プロファイルの設定</a> 」も参照）。
SET	<u>A</u> DVANCEDCMD	ON	オブジェクトハンドラで生成されたコマンドの上級ユーザーモードでの表示を有効にします。
		OFF	オブジェクトハンドラで生成されたコマンドの上級ユーザーモードでの表示を無効にします。
	<u>E</u> XECUTIONMSG	ON	処理ステータスが表示されるウィンドウを有効にします。
		OFF	処理ステータスが表示されるウィンドウを無効にします。
	<u>F</u> REE	ON	フリーフォーマット編集を有効にします。
		OFF	フリーフォーマット編集を無効にします。
	<u>T</u> RACE	ON	トレースモードを有効にします。オブジェクトハンドラの各操作が画面に表示されます。
		OFF	トレースモードを無効にします。
<u>W</u> ORKFILE		トレースモードを有効にします。オブジェクトハンドラの各操作がワークプランライブラリに保存されている Natural テキストメンバに書き込まれます。	
<u>T</u> RACEFILE		トレースに使用するワークプランライブラリの Natural テキストメンバを指定できる画面を表示します。	
SETTINGS			アンロード、ロードまたはスキャン設定を指定できる画面を表示します。
SHOW または <u>D</u> ISPLAY	<u>L</u> AST	<u>M</u> ESSAGE	オブジェクトハンドラの処理インターフェイスによって発行された最後のインターフェイスのリターンコードとメッセージが表示されます。
		<u>R</u> ESULT	オブジェクトハンドラの処理インターフェイスによって発行された最新の結果が表示されます。
	<u>P</u> ROFILE		テキストメンバPROFILES（存在する場合）が表示されます。

コマンド	サブコマンド1	サブコマンド2	説明
	REPORT		最後に作成されたレポートが表示されます。
	STATISTICS		処理されたオブジェクトの統計情報が表示されます。
	STATUS		オブジェクトハンドラの現在のステータス（グローバル変数の内容）が表示されます。
	TRACE	FILE	ワークプランライブラリに保存されている、トレースを含む Natural テキストメンバが表示されます。



# 48 バッチのコンディションコードとユーザー出口 ルーチン

---

- バッチで返されるコンディションコード ..... 272
- ユーザー出口ルーチンの適用 ..... 272
- 使用可能なユーザー出口ルーチン ..... 273

このセクションでは、バッチモードのオブジェクトハンドラ機能に対して返されるコンディションコードおよび機能処理に使用できるユーザー出口ルーチンについて説明します。

### バッチで返されるコンディションコード

---

バッチモードのオブジェクトハンドラ処理は、次のいずれかのコンディションコードで終了します。

コンディションコード	説明
0	オブジェクトハンドラ処理は正常終了しました。
30	内部的なオブジェクトハンドラエラーが発生しました。
40	オブジェクトハンドラコマンドでエラーが検出されました。
50	オブジェクトハンドラ処理中にエラーが発生しました。
60	オブジェクトハンドラ処理中に Natural Security エラーが発生しました。
99	オブジェクトハンドラ処理中に Natural エラーが発生しました。

### ユーザー出口ルーチンの適用

---

オブジェクトハンドラのユーザー出口ルーチンは、Natural システムライブラリ SYSOBJH にソースオブジェクトとして提供されます。これらのソースオブジェクトには SRC-EX $nn$  という名前が付いており、 $nn$  はユーザー出口ルーチンの番号を表します。

#### ▶手順 48.1. ユーザー出口ルーチンを起動するには

- Natural システムライブラリ SYSOBJH 内の  $nn$  OBJHEX $nn$  という名前で、ソースオブジェクト SRC-EX に対し CATALOG または STOW を実行します。

ユーザー出口ルーチンのソースオブジェクト（要件に応じて変更可能）およびカタログ化オブジェクトが更新インストールで上書きされないように、別の名前が使用されます。

ユーザー出口ルーチンの詳細については、ライブラリ SYSOBJH にある SRC-EX $nn$  のソースオブジェクトを参照してください。

## 使用可能なユーザー出口ルーチン

次のユーザー出口ルーチンを使用することができます。

- 障害処理のための OBJHEX01
- オブジェクト拒否のための OBJHEX02
- デフォルトオプション値のための OBJHEX03

### 障害処理のための OBJHEX01

コンディションコードがバッチモードで0（ゼロ）より大きい値に設定されると、オブジェクトハンドラが処理を停止する前に、ユーザー出口ルーチン OBJHEX01（使用可能な場合）が呼び出されます。このユーザー出口ルーチンで、オブジェクトハンドラ処理の続行または終了を指定できます。終了する場合には、コンディションコードを変更できます。詳細については、Natural システムライブラリ SYSOBJH のユーザー出口ルーチン SRC-EX01 のソースを参照してください。

### オブジェクト拒否のための OBJHEX02

オブジェクトハンドラのロード機能がバッチモードで正常に実行されても（コンディションコード0）、1つまたは複数のオブジェクトがロード中に拒否された（例：置換されなかった）場合、オブジェクトハンドラが処理を停止する前に、ユーザー出口ルーチン OBJHEX02（使用可能な場合）が呼び出されます。OBJHEX02 で、オブジェクトハンドラ処理の続行または終了を指定できます。終了する場合には、コンディションコードを変更できます。詳細については、Natural システムライブラリ SYSOBJH のユーザー出口ルーチン SRC-EX02 のソースを参照してください。

### デフォルトオプション値のための OBJHEX03

ユーザー出口ルーチン OBJHEX03 を適用して、オブジェクトハンドラコマンドの処理に対してデフォルトオプションを設定できます。このユーザー出口ルーチンは、オブジェクトハンドラコマンドが処理される前に呼び出されます。詳細については、Natural システムライブラリ SYSOBJH のユーザー出口ルーチン SRC-EX03 のソースを参照してください。





# 49 Tools

---

■ ステータス .....	276
■ 最新の結果 .....	276
■ トレース .....	276
■ レポート .....	277

オブジェクトハンドラには、ステータス情報とレポートを表示し、トレース設定のチェックと修正を行うための特殊機能が用意されています。

## ステータス

---

現在使用されているオブジェクトハンドラ機能、ユーザー環境、ワークプランライブラリ、および以下で説明しているトレースオプションの設定を表示します。

### ▶手順 49.1. ステータスを表示するには

- 任意のオブジェクトハンドラ画面で、コマンド行に次のコマンドを入力します。

```
SHOW STATUS
```

「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」にある **SHOW** コマンドの説明も参照してください。

## 最新の結果

---

オブジェクトハンドラの処理インターフェイスによって発行された最後の内部コマンドおよび有効なターンコードとメッセージを表示します。

### ▶手順 49.2. 最新の結果を表示するには

- 任意のオブジェクトハンドラ画面で、コマンド行に次のコマンドを入力します。

```
SHOW LAST RESULT
```

「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」にある **SHOW** コマンドの説明も参照してください。

## トレース

---

トレース機能を有効または無効にします。トレース機能により内部のオブジェクトハンドラのプログラムフローが記録され、エラー診断用の制御情報が提供されます。トレースオプションはデフォルトでオフに設定されています。

### ▶手順 49.3. 設定を変更するには

- 「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」の説明に従って、コマンド `SET TRACE` を使用します。

## レポート

ロード、アンロード、またはスキャンされたオブジェクトをリスト表示し、処理を中断する可能性のあるエラーを記録します。「設定」セクションの「ワークファイルオプション」も参照してください。レポートオプションはデフォルトで設定されており、アンロード、ロード、またはスキャン機能の実行後に表示されます。

### ▶手順 49.4. 最新レポートファイルの内容を表示するには

- 任意のオブジェクトハンドラ画面で、コマンド行に次のコマンドを入力します。

```
SHOW REPORT
```

「ダイレクトコマンド」セクションの「ナビゲーションと特別な機能のためのコマンド」にある `SHOW` コマンドの説明も参照してください。



# 50 プロファイル設定

---

■ PROFILES の編集 .....	280
■ パラメータ指定 .....	281

Naturalには、現在のオブジェクトハンドラユーティリティ環境のデフォルト設定をカスタマイズするオプションが用意されています。このために、Natural システムライブラリ SYSOBJH にはテキストメンバPROFILEがあります。PROFILEは、フラグやオプションの環境固有のデフォルト値を指定するために使用します。これらは、対応するオブジェクトハンドラ画面を入力するときに表示されます。

### ▶手順 50.1. 個々のプロフィール設定を有効にするには

- 1 テキストオブジェクト PROFILE を PROFILES という名前でライブラリ SYSOBJH に保存します。
- 2 テキストオブジェクト PROFILES で必要なパラメータを設定し、変更内容を保存します。
- 3 オブジェクトハンドラを起動して、PROFILES を有効にします。

有効になると、「Defaults from SYSOBJH profile (PROFILES) read.」という Natural 確認メッセージが表示されます。

### ▶手順 50.2. 個々のプロフィール設定を無効にするには

- テキストオブジェクト PROFILES をライブラリ SYSOBJH から削除します。

## PROFILES の編集

---

次のセクションでは、テキストオブジェクト PROFILES の編集に関する一般的な手順を示します。

アスタリスク (\*) または /\* で始まる空行はコメントとみなされるため、無視されます。

各行の長さは 90 バイト以内に制限されています。

このセクションでは、次のトピックについて説明します。

### ■ 一般的な設定およびユーザー固有の設定

#### 一般的な設定およびユーザー固有の設定

すべてのユーザーに適用される一般的なプロフィール設定を指定したり、単一ユーザーにのみ適用される個々のプロフィール設定を指定したりできます。

一般設定に使用するパラメータは、[General-Start] および [General-End] で区切られたブロック内に入力します。

個々の設定に使用するパラメータは、任意の個数のユーザー固有ブロック内に入力します。各ユーザー固有ブロックは、[User-Start UID-EXAM] および [User-End UID-EXAM] で区切られ、

UID-EXAMはNaturalシステム変数\*USERによって割り当てられるユーザーIDを表します（『システム変数』ドキュメントを参照）。各ユーザー固有ブロックは、ブロックの区切り内で入力されたIDを持つユーザーに対してのみ適用されます。

ユーザー固有ブロックで行ったパラメータ設定は、一般ブロックで行ったパラメータ設定よりも優先されます。

#### ユーザー固有ブロックの例

```
[User-Start SAG]
  Advanced-Mode           Y
  Option-Write-Report     N
[User-End SAG]
```

上記の例では、一般ブロックで行った対応する設定に関係なく、ユーザーSAGに対しては、メインメニューで「**Advanced mode**」フィールドが有効である（選択されている）として表示され、オブジェクトハンドラの「**Options**」画面では「**Write report**」フィールドが無効である（選択されていない）として表示されます。

## パラメータ指定

下の表に、PROFILEに含まれるパラメータ、入力可能な値、およびパラメータが適用されるオブジェクトハンドラ機能を示します。また、パラメータの簡単な説明または『オブジェクトハンドラ』ドキュメントの該当する参照セクションも示します。デフォルトのパラメータ値は、下線付きで示しています。

パラメータ	設定可能値	機能	説明/ ドキュメントのセクション
Advanced-Mode	<u>N</u> (No) または Y (Yes)	アンロード ロード スキャン	「機能」の「 <a href="#">上級ユーザー</a> 」
Display-Cmd-in-Advanced-Mode	<u>N</u> または Y	アンロード ロード スキャン	上級ユーザーモードで実行される機能用に生成されたオブジェクトハンドラコマンドを表示します。
Display-ExecutionMsg	<u>N</u> または Y	アンロード ロード スキャン	処理ステータスが表示されるウィンドウを有効にします。
Workplan-Library	<u>WORKPLAN</u> または 他の任意のワークプランライブラリ	アンロード ロード スキャン 管理	「 <a href="#">ワークプラン</a> 」および「 <a href="#">ワークプランライブラリの変更</a> 」（「 <a href="#">管理</a> 」）

プロフィール設定

パラメータ	設定可能値	機能	説明/ ドキュメントのセクション
Workplan-Library-DBID	Q (現在の FNAT/FUSER) または 他の任意の Adabas データベース ID (DBID)	アンロード ロード スキャン 管理	「ワークプランライブラリの変更」
Workplan-Library-FNR	Q (現在の FNAT/FUSER) または 他の任意の Adabas ファイル番号 (FNR)	アンロード ロード スキャン 管理	「ワークプランライブラリの変更」
TRACE	N または Y	アンロード ロード スキャン	「ツール」の「トレース」
TRACE-TARGET	S (Screen) または W (Work file)	アンロード ロード スキャン	「トレース」
Option-Replace	N または Y または O (Obsolete) E (Except)	ロード	「設定」の「置換オプション」
Option-TRANSFER-FORMAT	N または Y	アンロード ロード スキャン	「ワークファイル」の「ワークファイルフォーマット」
Option-TR-INCLUDE-LINE-NUMBERS	N または Y	アンロード	「転送オプション」の「Include line numbers」
Option-TR-SUBSTITUTE	N または Y	アンロード	「転送オプション」の「Substitute line numbers」
Option-TR-TRANSLATE-TO-UPPER	N または Y	ロード	「その他のオプションの設定」の「Translate to upper case」
Option-TR-USE-CONVERSION-TABLE	N または S (System table) または U (User table)	アンロード ロード	「転送オプション」の「Use conversion table」
Option-TR-CONV-TABLE-NAME-LOAD	OTNCONEA または ユーザー記述サブプログラム	ロード	「Use conversion table」



パラメータ	設定可能値	機能	説明/ ドキュメントのセクション
Option-TR-CONV-TABLE-NAME-UNLD	<u>OTNCONAE</u> または ユーザー記述サブプログラム	アンロード	「 <a href="#">Use conversion table</a> 」
Option-Write-Report	N または Y	アンロード ロード スキャン	「ツール」の「 <a href="#">レポート</a> 」 「ワークファイルオプション/レポートオプション」の「 <a href="#">Write report</a> 」
Option-Write-Restart-Info	N または Y	ロード	「ワークファイルオプション/レポートオプション」の「 <a href="#">Write restart information</a> 」
USE-OPTION-WORKPLAN	N または Y	アンロード ロード スキャン	「 <a href="#">ワークプラン</a> 」
OPTION-WORKPLAN-Name	<u>OPTIONWP</u> または タイプOPTIONの他の 任意のワークプラン	アンロード ロード スキャン	「 <a href="#">ワークプラン</a> 」
USE-PARAMETER-WORKPLAN	N または Y	アンロード ロード	「 <a href="#">ワークプラン</a> 」
PARAMETER-WORKPLAN-Name	<u>PARAWPLN</u> または タイプPARAMETERの 他の任意のワークプラン	アンロード ロード	「 <a href="#">ワークプラン</a> 」
WORK-FILE-1-Name	PCファイルに使用されるワークファイル（ワークファイル7）。	アンロード ロード スキャン	「 <a href="#">ワークファイル</a> 」 適用されるのは、Entire Connectionがインストールされている場合に限り限られます。
Report-File-Name	レポートに使用される、ワークプランライブラリに保存されたテキストメンバの名前。	アンロード ロード スキャン	「ワークファイルオプション/レポートオプション」の「 <a href="#">Write report</a> 」
Restart-File-Name	再スタート情報に使用される、ワークプランライブラリに保存されたテキストメンバの名前。	ロード	「ワークファイルオプション/レポートオプション」の「 <a href="#">Write restart information</a> 」

## プロファイル設定

---

パラメータ	設定可能値	機能	説明/ ドキュメントのセクション
Trace-File-Name	トレースに使用される、ワークプランライブラリに保存されたテキストメンバの名前。	すべての機能	「ツール」の「 <a href="#">トレース</a> 」

# 51 NATUNLD/NATLOAD および SYSTRANS からオブジェ クトハンドラへの移行

---

- 個々のコマンドの変換 ..... 286
- ユーザー出力ルーチンを使用したコマンドの処理 ..... 288
- OBJHAPI を使用した SYSTRANS コマンドの処理 ..... 289
- サポートされていない SYSTRANS オプション ..... 289

古いユーティリティ NATUNLD/NATLOAD および SYSTRANS からオブジェクトハンドラに移行するには、このセクションで説明する 2 つの方法を使用できます。

## 個々のコマンドの変換

---

移行用に提供されているオブジェクトハンドラコマンドを使用することにより、NATUNLD/NATLOAD または SYSTRANS ダイレクトコマンドを対応するオブジェクトハンドラコマンドに変換できます。これらの移行コマンドにより、古いユーティリティにより使用されるコマンド構文が、オブジェクトハンドラにより使用されるコマンド構文に自動的に変換されます。

### ▶手順 51.1. 単一のコマンドを変換するには

- 1 次のいずれかのオブジェクトハンドラダイレクトコマンドを使用します。

NATUNLD

続いて NATUNLD ダイレクトコマンドを入力します。

または:

NATLOAD

続いて NATLOAD ダイレクトコマンドを入力します。

または:

SYSTRANS

続いて SYSTRANS ダイレクトコマンドを入力します。

指定したユーティリティコマンドが、対応するオブジェクトハンドラコマンドに変換されます。

- 2 オブジェクトハンドラの後続のコマンドを、それぞれユーティリティ NATUNLD、NATLOAD、または SYSTRANS に適用される構文で指定します。

このユーティリティの構文は、オブジェクトハンドラセッションの間中は有効です。

**NATUNLD** コマンドの例

以下に、2つの連続する NATUNLD ユーティリティコマンドと、それに対応するオブジェクトハンドラコマンドの例を示します。

古い NATUNLD コマンド：	NATUNLD ALL * FM LIB1 TO LIB2
	ALL PG* FM LIB2
新しいオブジェクトハンドラコマンド：	SYSOBJH NATUNLD ALL * FM LIB1 TO LIB2
NATUNLD 構文での後続のオブジェクトハンドラコマンド：	ALL PG* FM LIB2

**SYSTRANS** コマンドの例：

以下に、2つの連続する SYSTRANS ユーティリティコマンドと、それに対応するオブジェクトハンドラコマンドの例を示します。

古い SYSTRANS コマンド：	TRANSCMD EXECUTE UNLOAD N FROM LIB1 NAME ETID
	END
新しいオブジェクトハンドラコマンド：	SYSOBJH SYSTRANS EXECUTE UNLOAD N FROM LIB1 NAME ETID END
SYSTRANS 構文での後続のオブジェクトハンドラコマンド：	END

バッチ処理での **SYSTRANS** の例：

以下に、マップ入力データおよび対応するオブジェクトハンドラコマンドと入力データを使用してバッチで SYSTRANS ユーティリティコマンドを処理する例を示します。

古い SYSTRANS バッチシーケンス：

```
SYSTRANS
U
N,N,N,Y,N,N,N,N
N
SRCLIB1,PGM1,*,TGTLIB1
```

新しいオブジェクトハンドラバッチシーケンス：

```
SYSOBJH SYSTRANS
U
N,N,N,Y,N,N,N,N
N
SRCLIB1,PGM1,*,TGTLIB1
```

## ユーザー出口ルーチンを使用したコマンドの処理

ユーティリティコマンドを1つずつ定義しなおさなくても、ユーティリティコマンドにより呼び出されるユーザー出口ルーチン内に、機能実行のための SYSOBJH を指定することができます。このルーチンにより、NATUNLD/NATLOAD または SYSTRANS に発行されたダイレクトコマンドが SYSOBJH コマンドとしてオブジェクトハンドラに転送されるかどうかが決まります。

### ▶手順 51.2. ユーザー出口ルーチンをアクティブにするには

#### 1 NATUNLD/NATLOAD の場合：

ソースオブジェクト U-S-EX03 に対し、UNLDEX03 という名前で Natural システムライブラリ SYSUNLD に SAVE を実行します。ソースオブジェクトが SYSUNLD 内に提供されません。

SYSTRANS の場合：

ソースオブジェクト TRA-E2-S に対し、TRA-EX-2 という名前で Natural システムライブラリ SYSTRANS に SAVE を実行します。ソースオブジェクトが SYSTRANS 内に提供されません。

#### 2 UNLDEX03 または TRA-EX-2 のソースをそれぞれ開き、USE-SYSOBJH を Y に設定します（デフォルトとして入力されている値です）。N に設定すると、ユーザー出口ルーチンが無効になります。

どのような場合に古いユーティリティではなくオブジェクトハンドラを使用するか、または誰がオブジェクトハンドラの使用を許可されるかなど、使用の条件を定義することができます。

#### 3 ソースオブジェクトに対し、CATALOG または STOW を実行します。

## OBJHAPI を使用した SYSTRANS コマンドの処理

---

(Natural システムライブラリ SYSOBJH 内に提供されている) OBJHAPI アプリケーションプログラミングインターフェイスを使用すると、SYSTRANS ユーティリティの構文でオブジェクトハンドラコマンドを実行できます。

この目的で OBJHAPI を使用する場合、OBJHAPI を呼び出すプログラム内でパラメータ P-EXTENSIONS-EXEC-SYSTRANS-CMD を指定する必要があります。詳細については、ライブラリ SYSOBJH 内に提供されているプログラム例 DOC-API を参照してください。

## サポートされていない SYSTRANS オプション

---

オブジェクトハンドラでは、SYSTRANS ダイレクトコマンドオプションの WORK-FILE-INPUT、SPECIAL-CONVERSION、RULE-LOAD、および UNLOAD-RULES はサポートされていません。





# 52 レコーディングユーティリティ

---

▪ 記録の目的 .....	292
▪ 記録されるデータおよび機能 .....	292
▪ セッションの記録 .....	293
▪ 記録の再生 .....	294
▪ 記録の操作 .....	296

レコーディングユーティリティを使用すると、Naturalセッションを記録し、記録したセッションを後で再生できます。

関連ドキュメント：

端末コマンド

## 記録の目的

---

レコーディングユーティリティは次の目的に使用できます。

- デモンストレーション  
入力データなどの複数のコマンドを入力する代わりに、記録したキーボードアクションシーケンスを再生して標準手順を実行できます。
- アプリケーション開発  
例えば、複数のオブジェクト（プログラムやマップなど）に同じ変更を適用する場合、レコーディングを使用すると、必要な作業量を減らすことができ、関連するすべてのオブジェクトに同じ変更を加えることができます。
- テスト  
記録を再生するだけで、標準のテスト手順を実行できます。
- 品質管理  
アプリケーションを変更する前と後に、記録を再生して2つの実行結果を比較することで、変更により特定の影響を受けなかったことを確認できます。
- ユーザー訓練  
記録の再生をユーザー訓練プログラムに組み込むことによって、ユーザーに特定の手順を示すことができます。あるセッションにおけるユーザーのキーボードアクションを記録しておき、誤入力や効率的な入力方法をユーザーに知らせることもできます。ユーザーアクションの記録は、アプリケーションのユーザーインターフェイスに関する欠陥を検出する場合にも役立ちます。

## 記録されるデータおよび機能

---

レコーディングユーティリティでは、次の情報が記録されます。

- 画面で入力するすべての入力データおよびコマンド（端末コマンドなど）
- 使用するすべてのファンクションキー（PF キー）
- システム変数\*CURSOR（『システム変数』ドキュメントを参照）に含まれている現在のカーソル位置

## セッションの記録

このセクションでは、記録を有効および無効にする手順について説明します。

- ライブラリの指定
- 記録の有効化
- 記録の無効化

### ライブラリの指定

#### ▶手順 52.1. 以降のすべての記録を保存するライブラリを指定するには

- 次の端末コマンドを入力します。

```
%B=library-name
```

*library-name* を指定しないで記録プロセスを有効にすると、記録を保存するライブラリの名前は、記録プロセスを有効にしたときのシステム変数 \*INIT-USER (『システム変数』ドキュメントを参照) の値と同じになります。

セッションを記録中に別のライブラリにログオンしても、記録が保存されるライブラリは同じままです。つまり、%B= または \*INIT-USER ライブラリで指定したライブラリのいずれかです。これは、1回の記録で、複数のアプリケーション間のキーボードアクションを記録できることを意味します。

### 記録の有効化

#### ▶手順 52.2. 記録を有効にするには


- 次の端末コマンドを入力します。

```
%Bname
```

以降のすべてのキーボードアクションが記録されます。

*name* は、記録されたデータが Recording タイプの Natural オブジェクトとしてソース形式で保存されるときの名前を指定します。このソースは、他の Natural ソースと同様に扱うことができます (削除やコピーなどが可能)。ただし、編集はしないでください。記録にはバイナリデータが含まれており、エディタによって破壊されるためです。

*name* は 1 回だけ指定できます。記録先として指定したライブラリ内に同じ名前の記録オブジェクトがすでに存在している場合、「RECORDING 処理内エラーがあります」というメッセージが返されます。

-  **注意:** 記録中に、非アクティビティタイムアウトなどによりトランザクションのバックアウトまたはロールバックが発生すると、記録の一部が削除されるため、記録全体が使用できなくなります。

記録中に入力される端末コマンド `%Aname` の後には端末コマンド `%B` を続けて入力する必要があります。次の「[%A の記録](#)」を参照してください。

### 記録の無効化

#### ▶手順 52.3. 記録を無効にするには

- 次の端末コマンドを入力します。

```
%B
```

記録が終了します。

### 記録の再生

---

記録を再生すると、例えばコマンドおよびファンクションキーのシーケンスが実際に再実行されます。

記録は端末タイプとは無関係です。つまり、ある端末で記録されたセッションを別のタイプの端末で再生できます。記録をバッチモードで再生することもできます。記録されたオンラインセッションは、バッチモードで再生すると異なる動作をします。

このセクションでは、次のトピックについて説明します。

- [ステップモードおよびバックグラウンドモード](#)
- [再生の有効化](#)
- [再生の中断](#)

#### ステップモードおよびバックグラウンドモード

記録は、バックグラウンドモードおよびステップモードのいずれかで再生できます。

バックグラウンドモードの場合、記録全体が非表示で再生されます。つまり、記録されているすべてのキーボードアクションが実行されている間、端末画面には何が実行されているか表示されません。バックグラウンドモードで再生すると、記録に端末コマンド `%R`（「[記録の操作](#)」を参照）が含まれている場合を除いて、記録の再生を中断できません。

ステップモードの場合、記録は1ステップずつ再生され、すべてのキーボードアクションが画面に表示されます。任意のファンクションキーを選択すると、次のステップに進むことができま

す。ステップモードでは、CLEAR キーを押すと記録の再生を中断することもできます。「再生の中断」を参照してください。

デフォルトでは、記録はバックグラウンドモードで再生されます。

#### ▶手順 52.4. モードを設定するには

- 1 ステップモードを有効にするには、次の端末コマンドを入力します。

```
%GON
```

- 2 ステップモードを無効にしてバックグラウンドモードを有効にするには、次の端末コマンドを入力します。

```
%GOFF
```

- 3 ステップモードとバックグラウンドモードを交互に切り替えるには、次の端末コマンドを入力します。

```
%G
```

## 再生の有効化

#### ▶手順 52.5. 記録を再生するには

- 次の端末コマンドを入力します。

```
%Aname
```

指定した名前で保存された記録が再度実行されます。

#### %Aname の記録

セッションの記録中にコマンド %Aname を発行すると、%Aname で指定された記録は実行されず、記録対象のオブジェクトソースにコマンド %Aname が含まれます。このため、ある記録内で別の記録を実行し、複数の記録を連結できます。ただし、記録はネストできません。%Aname コマンドが含まれている記録は、このコマンドより後の記録の実行が停止され、name が終了しても残りの実行は再開されません。したがって、%Aname の後に続く記録データは再生されません。これを回避するには、記録中に %Aname および %B を続けて入力する必要があります。

### 再生の中断

#### ▶手順 52.6. ステップモードで再生中の記録を中断するには

- CLEAR キーを押します。

記録を中断したら、次のことを行うことができます。

- 記録の再生を停止した場所からセッションの再生を正常に続行できます。
- 追加のキーボードアクションを記録に挿入できます。CLEAR キーを押した後にコマンド %B を入力すると、%B を再度入力するまで、実行するすべてのアクションが記録ソースに挿入されます。その後、記録が再開されます。
- 記録の次のステップを変更できます。CLEAR キーを押した後、コマンド %R を入力してから、次のステップとして入力データを新たに指定します。記録されたソース内にあるこのステップの入力データは、新たに指定した入力データで上書きされます。Enter キーを押すと、新しい入力データが記録されたこのステップが実行され、記録が再開されます。
- 任意のヘルプルーチンを実行できます。CLEAR キーを押した後、コマンド %J を入力し、続いて希望するヘルプルーチンの名前を入力します。ヘルプルーチンが呼び出され、ヘルプルーチンの実行が終了するとすぐに記録が続行されます。

### 記録の操作

---

端末コマンド %R を記録することにより、再生時に記録の単一のステップを操作できます。これは、ステップモードとバックグラウンドモードの両方に適用されます。バックグラウンドモードの場合、再生中の記録を操作する唯一の方法は %R です。例えば、記録時には認識されないパスワードなどの機密データを入力できるようにする操作が必要な場合があります。

最後の画面を再表示する端末コマンド %R が記録されている場合、記録の再生中に、ユーザー入力用に後続の画面が表示されます。つまり、この画面に表示される入力データは、記録から取得されるのではなく、ユーザーによる入力から取得されます。その後、記録が続行されます。

# 53      SYSAPIユーティリティ-Naturalアドオン製品のAPI

---

▪ 前提条件 .....	298
▪ SYSAPIの呼び出しと終了 .....	298
▪ APIプログラム例のリスト .....	299
▪ キーワード検索の実行 .....	300
▪ APIプログラム例の使用 .....	301

SYSAPI ユーティリティは、Entire Output Management などの Natural アドオン製品のアプリケーションプログラミングインターフェイス (API) を検索し、テストするために使用します。

Natural アドオン製品の API は、データへのアクセスやデータの変更、またはアドオン製品やサブコンポーネントに固有のサービスの実行に使用される Natural サブプログラム (カタログ化オブジェクト) です。

Natural アドオン製品の API は、特定の Natural アドオン製品に固有のオブジェクト用に提供される Natural ライブラリやシステムファイル内に用意されています。Natural アドオン製品の API の使用方法については、各アドオン製品のドキュメントを参照してください。

SYSAPI ユーティリティでは、Natural アドオン製品の API ごとに、機能説明、1つのプログラム例、API 固有のキーワードが提供されています。

関連トピック：

- 「Natural ユーザー出口」 - 『オペレーション』ドキュメント
- 「[SYSEXT - Natural アプリケーションプログラミングインターフェイス](#)」 - 『ユーティリティ』ドキュメント

## 前提条件

---

- 適切な Natural アドオン製品がサイトにインストールされている必要があります。
- インストールされている Natural アドオン製品のバージョンで SYSAPI ユーティリティ機能がサポートされている必要があります。

## SYSAPI の呼び出しと終了

---

このセクションでは、SYSAPI ユーティリティの呼び出しと終了の手順について説明します。

### ▶手順 53.1. SYSAPI ユーティリティを呼び出すには

- 次のシステムコマンドを入力します。

```
SYSAPI
```

1つまたは複数の Natural アドオン製品がリストされている SYSAPI メニューが表示されます。製品ごとに、関連する API グループが表示されます。

各 API グループは、Natural アドオン製品に用意されている特定の API 機能を表しています。API グループには、この機能に関連するすべてのプログラム例が含まれています。



APIグループを選択すると、このグループに含まれているすべてのプログラム例が表示されます（「[API プログラム例のリスト](#)」を参照）。



**注意:** SYSAPI ユーティリティ画面のコマンド行に、Natural システムコマンドを入力できます。

#### ▶手順 53.2. SYSAPI を終了するには

- PF3 キーまたは PF12 キーを押します。

## API プログラム例のリスト

選択した API グループで使用可能なすべてのプログラム例のリストを表示できます。API プログラム例の名前または名前の範囲を指定すると、このリストを短くできます。

#### ▶手順 53.3. すべての API プログラム例をリストするには

- SYSAPI メニューで、目的の API グループの横にある入力フィールドに任意の文字を入力するか、カーソルを置いて選択します。

選択した API グループの SYSAPI リスト画面が表示されます。画面の **[Example]** 列に API プログラム例の名前が、**[Function]** 列に該当する API の簡単な機能説明が表示されます。

このリストは、プログラム例の名前に基づいてアルファベット順にソートされます。必要に応じて、PF8 キーを押してリストを 1 ページ下方にスクロールします。リストの先頭に移動するには、PF6 キーを押します。

#### ▶手順 53.4. プログラム例を 1 つまたは特定の範囲のみをリストするには

- 1 SYSAPI リスト画面の **[Example]** 入力フィールドに、次の表の入力値を入力します。 *value* は、1 つ以上の文字の任意の組み合わせを表します。

入力値	選択するプログラム例
<i>value</i>	<i>value</i> に一致するプログラム例。
*	すべてのプログラム例。これはデフォルト設定です。
<i>value</i> * <i>value</i> ?	任意の位置に任意の順序で入力された、1 つ以上のアスタリスク (*) や 1 つ以上の疑問符 (?) と組み合わせられた <i>value</i> に一致するすべてのプログラム例。 アスタリスク (*) は、任意の文字列を表します。 疑問符 (?) は、1 文字を表します。
例	

入力値	選択するプログラム例
	ABC*
	A?C*Z

2 Enter キーを押します。

SYSAPI リスト画面に、指定したプログラム例が表示されます。

## キーワード検索の実行

キーワードは、現在のタスクに関連する API を検索する場合に役立ちます。キーワード検索機能を使用すると、選択した API グループに関連するすべてのキーワードを表示したり、キーワード別に API プログラム例を表示したりできます。

### ▶手順 53.5. 特定のAPI グループに関連するすべてのキーワードを表示するには

■ コマンド行に「KEYWORDS」と入力します。

または:

PF5 キーを押します。

「**List Keywords**」ウィンドウが表示されます。その API グループに固有のすべてのキーワードがアルファベット順にリストされます。

必要に応じて、PF8 キーを押してキーワードのリストを下方にスクロールしたり、PF6 キーを押してリストの先頭に戻ったりできます。

### ▶手順 53.6. キーワードでプログラム例をリストするには

1 SYSAPI リスト画面の「**Keywords**」の横にある入力フィールドに、正しいキーワードを入力します。必要に応じて、PF10 キーを押してすべての「**Keywords**」フィールドの内容をクリアします。

または:

「**List Keywords**」ウィンドウで最大 3 つまでキーワードを選択します。

■ 目的のキーワードの横に、選択するキーワードであることを示す任意の文字を入力して、Enter キーを押します。

選択したキーワードが、「**List Keywords**」ウィンドウの「**Selected keywords**」の下に表示されます。

必要に応じて、PF10 キーを押して [Selected keywords] にリストされたキーワードをクリアし、キーワードの選択を繰り返します。

- PF3 キーを押して、キーワードの選択を確定します。

SYSAPI リスト画面の [Keywords] フィールドに、選択したキーワードが表示されます。また、指定したキーワードが含まれているプログラム例が画面にリストされます。

- 2 必要に応じて、[And/Or] フィールドに「A」または「0」（デフォルト設定は「A」）を入力し、論理条件に基づいてキーワードを組み合わせます。
- 3 Enter キーを押して、キーワード検索を実行します。

指定したキーワードが含まれているプログラム例が画面にリストされます。

## API プログラム例の使用

SYSAPI ユーティリティには、API プログラム例の表示、変更、または実行に使用できる行コマンドがあります。SYSAPI リスト画面の目的のプログラム例の横にある [Cmd] 列に行コマンドを入力します。この列に疑問符 (?) を入力すると、有効な行コマンドがリストされます。

SYSAPI リスト画面では次の行コマンドを使用できます。

行コマンド	機能
L	ソースコードをリストします。
E	ソースコードを編集します。
X	プログラムを実行します。



# 54 SYSBPM ユーティリティ・バッファプールの管理

---

ユーティリティ SYSBPM は、Natural、DL/I または SORT タイプのバッファプールの管理に使用します。

SYSBPM では、バッファプールキャッシュを含むバッファプールの現在のステータスに関する統計情報、およびバッファプールと BP キャッシュにロードされた Natural オブジェクトに関する情報が提供されています。SYSBPM は、管理機能も備えています。

Natural バッファプールの全般的な説明については、『オペレーション』ドキュメントを参照してください。

バッファプールは、Natural パラメータモジュールのマクロ NTBPI、または対応するダイナミックプロファイルパラメータ BPI で定義されます。これらの詳細については、『パラメータリファレンス』ドキュメントを参照してください。バッファプールのタイプは、「バッファプールの初期化」で説明されているように、NTBPI マクロの TYPE サブパラメータによって判断することができます。

- **SYSBPM の呼び出しと操作**

- オブジェクトのリスト

- オブジェクトの削除

- ディレクトリ情報


- 16 進表示

- ワークファイルへの書き込み

- ソート済み抽出の表示

- バッファプール統計

- バッファプールキャッシュ統計
- バッファプールの選択
- ブラックリストの管理
- プリロードリストの管理
- パフォーマンスの考慮事項
- **SYSBPM** ダイレクトコマンド
- バッチ処理
- アプリケーションプログラミングインターフェイス

 **注意:** このドキュメントでは、バッファプールは「BP」とも記載されています。

# 55

## SYSBPM の呼び出しと操作

---

▪ SYSBPM の呼び出し .....	306
▪ オンラインヘルプ .....	307
▪ SYSBPM メインメニュー - フィールド、機能、およびコマンド .....	308
▪ Sysplex 環境での SYSBPM .....	311

SYSBPM ユーティリティの機能は、常に Natural バッファプールおよびバッファプールキャッシュのいずれか、またはその両方に現在ロードされている Natural オブジェクトを参照します。バッファプールまたはバッファプールキャッシュには、格納またはカタログされた実行済みの Natural オブジェクトのみが含まれることに注意してください。RUN コマンドで実行されたオブジェクトは、バッファプールまたはバッファプールキャッシュにロードされません。

SYSBPM メニューのファンクションコードまたは PF キーを選択して、SYSBPM 機能を実行できます。または、関連するセクションの説明に従って、**SYSBPM ダイレクトコマンド**を使用できます。

このセクションでは、SYSBPM ユーティリティを呼び出す方法、オンラインヘルプテキストを表示する方法、および SYSBPM メインメニューを使用して機能を選択する方法について説明します。各機能の詳細は、SYSBPM ドキュメントの関連するセクションに記載されています。

また、Sysplex 環境での SYSBPM の使用方法も記載されています。

## SYSBPM の呼び出し

### ▶手順 55.1. SYSBPM ユーティリティを呼び出すには

- 次の Natural システムコマンドを入力します。

```
SYSBPM
```

次のような SYSBPM メインメニューが表示されます。

```

16:12:23          ***** NATURAL SYSBPM UTILITY *****          2002-08-27
BPNAME QA41GBP          - Main Menu -          Type Global Nat
BPPROP OFF          Loc DAEF QA41          Preload QA41GBPL

          Object Functions          Object Pool Statistics

          L List Objects          A Buffer Pool
          D Delete Object          C BP Cache

          I Directory Information          Other Functions
          H Hexadecimal Display
          W Write to work file          S Select Buffer Pool
          X Display sorted extract          B Blacklist Maintenance
          ? Help          P Preload List Maintenance
          . Exit

Code .. _   Library ... *_____
Object .... *_____
DBID ..... 0_____ FNR .. 0_____ Object Pool ... B (B,C,*)
    
```



```
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit  Last      Flip      Canc
```

- SYSBPM メインメニューで、次のいずれかのオプションを選択して、バッファプールまたはバッファプールキャッシュにロードされた実行済みオブジェクトを指定します。
- 「**SYSBPM** メインメニュー - フィールド、機能、およびコマンド」の説明に従って、入力フィールドに入力します。

または、  
コマンド行で、「**SYSBPM** ダイレクトコマンド」の説明に従って SYSBPM ダイレクトコマンドを入力します。

## オンラインヘルプ

SYSBPM のオンラインヘルプ機能では、**SYSBPM** ダイレクトコマンド（関連のセクションを参照）や SYSBPM 画面に表示されるフィールドの有効な入力値に関する情報が提供されています。

### ▶手順 55.2. SYSBPM ダイレクトコマンドのオンラインヘルプ機能呼び出すには

- 任意の SYSBPM 画面でカーソルをコマンド行に置き、PF1 キーを押すか、疑問符 (?) を入力します。

[**Help**] ウィンドウが開き、使用可能なすべての SYSBPM ダイレクトコマンドのリストが表示されます。

### ▶手順 55.3. SYSBPM 入力フィールドのオンラインヘルプ機能呼び出すには

- 任意の SYSBPM の画面で、カーソルを任意の入力フィールドに置き、PF1 キーを押すか、または疑問符 (?) を入力します。

関連するフィールドの [**Help**] ウィンドウが開き、すべての有効な入力値のリストが表示されます。

## SYSBPM メインメニュー・フィールド、機能、およびコマンド

このセクションでは、次のトピックについて説明します。

- フィールド
- 機能
- PF キーとダイレクトコマンド

### フィールド

次の表に、SYSBPM メインメニューで表示されるフィールドとその説明を示します。

フィールド	説明
<b>BPNAME</b>	プロファイルパラメータ BPNAME で指定されたグローバルバッファプールの名前。ローカルバッファプールでは、名前はありませんが、BPNAME について空白のフィールドが表示されます。  『パラメータリファレンス』ドキュメントの「BPNAME - Natural グローバルバッファプールの名前」も参照してください。
<b>BPPROP</b>	バッファプール内のオブジェクトへの変更の伝播を制御するためのプロファイルパラメータ BPPROP の設定。  『パラメータリファレンス』ドキュメントの「BPPROP - グローバルバッファプールの伝播」も参照してください。
<b>Type</b>	Global Nat、Local Nat、Global Sort、または Global DL/I などのバッファプールのタイプ。
<b>Loc</b>	場所。ホスト ID（上の例の DAEF）およびサブシステム ID（上の例の QA41）が表示されます。
<b>Preload</b>	ロードされている場合は、プリロードリストの名前。  「 <a href="#">プリロードリストの管理</a> 」も参照してください。
<b>Library</b>	実行済みオブジェクトが格納されるライブラリの名前。 名前を指定するかアスタリスク (*) 表記を使用することができます。  デフォルトのアスタリスク (*) では、すべてのライブラリが選択されます。
<b>Object</b>	バッファプールにロードされた実行済みオブジェクトの名前。 名前を指定するかアスタリスク (*) 表記を使用することができます。  デフォルトのアスタリスク (*) では、すべてのオブジェクトが選択されます。
<b>DBID</b>	実行済みオブジェクトが格納され、ロード元となるシステムファイル FNAT または FUSER のデータベース ID (DBID)。  DBID に 0 (ゼロ。デフォルト) を指定した場合、DBID に関係なく、指定したオブジェクトが選択されます。0 以外の値は、特定の DBID 指定を表します。

フィールド	説明						
FNR	<p>実行済みオブジェクトが格納され、ロード元となるシステムファイル FNAT または FUSER のファイル番号 (FNR)。</p> <p>FNR に 0 (ゼロ。デフォルト) を指定した場合、FNR に関係なく、指定したオブジェクトが選択されます。0 以外の値は、特定の FNR 指定を表します。</p>						
Object Pool	<p>次のセクションで説明する [List Objects]、[Directory Information]、[Write to Work File]、および [Display Sorted Extract] 機能に適用されます。</p> <p>これらの機能に使用するオブジェクトプールのタイプを選択します。</p> <table border="1"> <tbody> <tr> <td>B</td> <td>バッファプール。</td> </tr> <tr> <td>C</td> <td>バッファプールキャッシュ。</td> </tr> <tr> <td>*</td> <td>バッファプールとバッファプールキャッシュ。これはデフォルトです。</td> </tr> </tbody> </table> <p>[List Objects] 機能では、バッファプールまたはバッファプールキャッシュにロードされたすべてのオブジェクトのリストを生成します。バッファプールにロードされたオブジェクトが最初に表示され、次に、バッファプールキャッシュ内のオブジェクトが表示されます。</p> <p>入力したオブジェクトプールのタイプによってリストの開始地点が決まります。例えば、「C」と入力すると、リストはバッファプールキャッシュにロードされたオブジェクトで始まります。リストを上方にスクロールすると、バッファプールにロードされたオブジェクトを表示できます。バッファプールキャッシュ内にオブジェクトがない場合、リストにはバッファプールのオブジェクトが含まれています。</p>	B	バッファプール。	C	バッファプールキャッシュ。	*	バッファプールとバッファプールキャッシュ。これはデフォルトです。
B	バッファプール。						
C	バッファプールキャッシュ。						
*	バッファプールとバッファプールキャッシュ。これはデフォルトです。						

## 機能

SYSBPM メインメニューで提供される機能は、次の 3 つのセクションで構成されています。

- [Object Functions] セクションには、バッファプールまたはバッファプールキャッシュ内のオブジェクトを表示または操作する機能があります。
- [Object Pool Statistics] セクションには、ハッシュテーブルを含むバッファプールまたはバッファプールキャッシュのオブジェクトに依存しない統計データを取得するための機能があります。オブジェクトに依存しないデータには、オブジェクト名、サイズ、アドレスなどのオブジェクトに関する個々の情報は含まれていません。
- [Other Functions] セクションには、バッファプールを選択するための機能、およびバッファプールまたはバッファプールキャッシュにロードするオブジェクトを指定するための機能があります。

次に個々の機能を示します。機能呼び出しには、**[List Objects]** の「L」など、必要な機能に対応する 1 文字のコードを **[Code]** フィールドに入力します。

コード	機能	説明
L	オブジェクトのリスト	バッファプールおよびバッファプールキャッシュ（使用している場合）のいずれかあるいはその両方にロードされたオブジェクトに関する情報を表示します。各リスト項目に個別にアクセスし、各オブジェクトに対してさまざまな機能を実行できます。
D	オブジェクトの削除	バッファプールおよびバッファプールキャッシュから 1 つ以上のオブジェクトを削除します。
I	ディレクトリ情報	バッファプールまたはバッファプールキャッシュにロードされた指定のオブジェクトの全ディレクトリ情報を表示します。
H	16 進表示	バッファプールにロードされた指定のオブジェクトを 16 進形式で表示します。
W	ワークファイルへの書き込み	ローカルファイルまたは PC テキストファイルに、バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方にあるオブジェクトディレクトリ情報を書き込みます。
X	ソート済み抽出の表示	バッファプールまたはバッファプールキャッシュにある 50 のオブジェクトディレクトリのソート済みリストを表示します。リスト項目は、指定したいずれかのソート条件を使用して並べることができます。
A	バッファプール	<b>[Buffer Pool Statistics]</b> メニューを呼び出します。このメニューから、ハッシュテーブル統計を含むバッファプールのオブジェクトに依存しない統計機能呼び出すことができます。
C	バッファプールキャッシュ	必須のバッファプールキャッシュ。  <b>[BP Cache Statistics]</b> メニューを呼び出します。このメニューから、ハッシュテーブル統計を含むバッファプールキャッシュのオブジェクトに依存しない統計機能呼び出すことができます。
S	バッファプールの選択	使用可能なすべてのバッファプールの選択リストを表示します。
B	ブラックリストの管理	実行しないオブジェクトのブラックリストを管理する場合に使用する <b>[Blacklist Maintenance]</b> メニューを呼び出します。
P	プリロードリストの管理	<b>[Preload List Maintenance]</b> メニューを呼び出します。プリロードリストでは、バッファプールの初期化時にバッファプールにロードするオブジェクトの名前を指定できます。

## PF キーとダイレクトコマンド

SYSBPM メインメニューでは、次の表に示す PF キーや SYSBPM ダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、「[SYSBPM ダイレクトコマンド](#)」を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ヘルプ情報を表示します。 <a href="#">オンラインヘルプ</a> も参照してください。
PF3	EXIT	現在の機能／画面を終了して、前の画面を表示します。
PF4	LAST	最後に入力した SYSBPM ダイレクトコマンドを表示します。
PF6	FLIP	PF キー行を切り替えます。PF1～PF12 キーと PF13～PF24 キーの間で表示を切り替えます。
PF12	CANCEL	EXIT と同じです。
PF15	MENU	SYSBPM メインメニューを呼び出します。

## Sysplex 環境での SYSBPM

Natural で別のオペレーティングシステムイメージ（ホスト）に切り替えると、常にバッファプールも切り替わります。バッファプールの切り替えは、異なるホスト ID によって示されます。このホスト ID は、SYSBPM 画面の [Loc] フィールドに表示されます。

切り替えは、各端末 I/O の後、つまり、いずれかのファンクションキーを選択した後、または Enter キーを選択することによって実行できます。バッファプールの切り替え後、参照コマンドおよび位置調整コマンドの TOP、BOTTOM、+、-、LEFT、RIGHT は実行されません。代わりに、リストは新しいバッファプールの先頭から始まります。

BPPROP プロファイルパラメータ（『パラメータリファレンス』ドキュメントの「BPPROP - グローバルバッファプールの伝播」を参照）が PLEX または GPLEX に設定されている場合、ブラケットリストの操作、オブジェクトの削除、またはバッファプールの初期化を行う SYSBPM コマンドは、まず、通常どおり実行され、次に同じサブシステムで使用可能な他のバッファプールに伝播します。バッファプールの切り替えによって機能が中断したり、伝播したりした場合は、該当するメッセージが表示されます。Natural が別のホストに正常に切り替えられ、バッファプールが変更された場合にも、該当するメッセージが表示されます。



## 56 オブジェクトのリスト

---

▪ [List Objects] の呼び出し .....	314
▪ 列および選択オプション .....	315
▪ PF キーとダイレクトコマンド .....	325
▪ 行コマンド .....	326

この機能では、**[List Objects]** 画面を呼び出します。この画面で、バッファプールまたはバッファプールキャッシュ（使用している場合）に現在ロードされている Natural オブジェクトのディレクトリに関する統計データを取得することができます。

このセクションでは、**[List Objects]** 画面に表示される統計データ、およびオブジェクトやオブジェクトの範囲の選択、それらのオブジェクトの現在のステータスの操作、**[List Objects]** 画面での移動に使用できるコマンドと機能について説明します。

オブジェクトのソート済み表示については、「**Display Sorted Extract 機能**」も参照してください。

## **[List Objects]** の呼び出し

### ▶手順 56.1. **[List Objects]** 画面を呼び出すには

- SYSBPM メインメニューの **[Code]** フィールドに、ファンクションコード「L」を入力し、オブジェクトを指定します。「SYSBPM の呼び出しと操作」セクションで説明している有効な **フィールド入力値** を参照してください。

または:

次の SYSBPM ダイレクトコマンドを入力して、バッファプール内のオブジェクトのリストに直接移動します。

```
DISPLAY LIST library-name object-name dbid fnr
```

または:

次の SYSBPM ダイレクトコマンドを入力して、バッファプールキャッシュ内のオブジェクトのリストに直接移動します。

```
DISPLAY CLIST library-name object-name dbid fnr
```

次のような **[List Objects]** 画面が表示されます。

```
17:13:17          ***** NATURAL SYSBPM UTILITY *****          2002-09-16
BPNAME QA41GBP          - List Objects -          Type Global Nat
BPPROP OFF          Loc DAEF QA41
C Library Object      DBID  FNR Loc RLD Use Max  Reuse      TotalUC ObjSize Sto
*  *
__ SYSBPM  BPMCALL    10 1640 B          1 1          5  8,516 12
__ SYSBPM  BPMNSC     10  410 B          1          4  3,380  4
__ SYSDLINP PCNDL02   255 253 B  R          1          19   292  4
__ SYSLIBS NAT00017   10  410 B          1          1  5,000  8
```



___	SYSLIB	ATEST	10	1640	B	R	1	340	16,148	16
___	SYSLIB	CATALL10	10	410	B		1	1	4,256	8
___	SYSBPM	BPM141-M	10	1640	B		1	1	5,944	8
___	SYSDLINS	U246005	255	253	B	R	1	14	52	4
___	SYSBPM	MENU	10	1640	B		1	5	10,392	12
___	SYSLIBS	NAT00040	10	410	B		1	1	2,816	4
___	SYSLIBS	NAT00034	10	410	B		1	1	2,672	4
___	SYSDLIND	DNDL01	255	253	B	R	3	42	552	4
___	SYSLIB	ACATALL	10	410	B		1	3	55,728	56
___	SYSDLINS	U246004	255	253	B	R	2	28	172	4

Top of List

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---

Help Exit Last Cache -- - + ++ > Canc

【List Objects】画面には、次の個々のオブジェクトがすべて表示されます。

1. 現在バッファプールにロードされているオブジェクト（表示の最初の部分）
2. 現在バッファプールキャッシュにロードされているオブジェクト（表示の2番目の部分）

表示される統計は、バッファプールの内容のスナップショットであり、Enterキーを押すたびに更新されます。

バッファプールにロードされた **GDA** オブジェクトに関する注：

【List Objects】画面には、GDA（グローバルデータエリア）について2つのエントリが表示される場合があります。1つのエントリには、GDA 自体の中のデータが含まれ、もう1つのエントリには、そのGDAの内部 Natural 記号テーブルが含まれています。この現象は、GDAを参照するプログラムがカタログされている場合に発生することがあります。

## 列および選択オプション

このセクションでは、【List Objects】画面に表示される列と統計データ、およびオブジェクトの選択条件を指定するための入力フィールドと値について説明します。

### ▶手順 56.2. 表示するオブジェクトの選択条件を指定するには

- 列のタイトルの下にある入力フィールドに、該当するフィールドについて示されているように、有効な値または範囲を入力します。

デフォルト値は、空白文字またはすべてのオブジェクトが選択されるアスタリスク (\*) です。

## オブジェクトのリスト

---

列	説明
C	この列には、オブジェクトの機能を実行する行コマンドを入力できます。「 <a href="#">行コマンド</a> 」も参照してください。
Library	オブジェクトのロード元のライブラリ。 選択条件を指定するには、「 <a href="#">名前と範囲の指定</a> 」を参照してください。
Object	オブジェクトの名前です。 選択条件を指定するには、「 <a href="#">名前と範囲の指定</a> 」を参照してください。
DBID	オブジェクトのロード元の Natural システムファイルのデータベース ID。 特定のデータベースのオブジェクトを選択するには、有効な数値を入力します。
FNR	オブジェクトのロード元の Natural システムファイルのファイル番号。 特定のファイルのオブジェクトを選択するには、有効な数値を入力します。

列	説明	
Loc	オブジェクトの場所。	
	B	バッファプール。
	B/C	バッファプールとバッファプールキャッシュ。
	C	バッファプールキャッシュ。
	C/B	バッファプールキャッシュとバッファプール。
		最初の位置に B が表示された場合、統計データはバッファプールから取得されます。最初に C が表示された場合、データはバッファプールキャッシュから取得されます。また、この位置に応じて、統計画面のフィールドに異なる行コマンドが適用されます（「 <a href="#">行コマンド</a> 」も参照）。
		オブジェクトの場所を指定するには、次のいずれかの値を入力します。
	B	バッファプールのみロードされたすべてのオブジェクトを選択します。
	B/C	バッファプールとバッファプールキャッシュにロードされたすべてのオブジェクトを選択します。
	B *	バッファプールまたはバッファプールとバッファプールキャッシュの両方にロードされたすべてのオブジェクトを表示します (B* = B + B/C)。
	C	バッファプールキャッシュのみロードされたすべてのオブジェクトを選択します。
C/B	バッファプールキャッシュとバッファプールにロードされたすべてのオブジェクトを選択します。	
C*	バッファプールキャッシュまたはバッファプールキャッシュとバッファプールの両方にロードされたすべてのオブジェクトを選択します (C* = C + C/B)。	

## オブジェクトのリスト

列	説明										
<b>RLD</b>	<p>バッファプールまたはバッファプールキャッシュ内のオブジェクトの現在のステータスです。バッファプールキャッシュのステータスは、オブジェクトのロックのみを表します。したがって、<b>[RLD]</b> 列の <b>[L]</b>（ロック状態）の下にのみ示されます。</p> <p>バッファプール：</p> <table border="1" data-bbox="342 352 1373 808"> <tr> <td data-bbox="342 493 716 640">R</td> <td data-bbox="721 493 1373 640">常駐としてマークされます。常駐とは、<b>[Use]</b> 列の関連する値が 0（ゼロ）に変更され、オブジェクトが使用されないことを示す場合にも、オブジェクトがバッファプールから削除されないことを意味します。</td> </tr> <tr> <td data-bbox="342 646 716 682">L</td> <td data-bbox="721 646 1373 682">ロード機能の進行中でロックされている。</td> </tr> <tr> <td data-bbox="342 688 716 808">D</td> <td data-bbox="721 688 1373 808">オブジェクトの削除コールは保留中。<b>[Use]</b> 列の値が 0（ゼロ）に変更されるとすぐに、オブジェクトは削除されます。</td> </tr> </table> <p>バッファプールキャッシュ：</p> <table border="1" data-bbox="342 934 1373 1039"> <tr> <td data-bbox="342 940 716 976">L</td> <td data-bbox="721 940 1373 976">ロード機能の進行中でロックされている。</td> </tr> <tr> <td data-bbox="342 982 716 1018">D</td> <td data-bbox="721 982 1373 1018">削除のためにロックされている。</td> </tr> </table> <p>特定のステータスのオブジェクトをすべて選択するには、上記で説明しているように、コード R、L、または D を入力します。</p>	R	常駐としてマークされます。常駐とは、 <b>[Use]</b> 列の関連する値が 0（ゼロ）に変更され、オブジェクトが使用されないことを示す場合にも、オブジェクトがバッファプールから削除されないことを意味します。	L	ロード機能の進行中でロックされている。	D	オブジェクトの削除コールは保留中。 <b>[Use]</b> 列の値が 0（ゼロ）に変更されるとすぐに、オブジェクトは削除されます。	L	ロード機能の進行中でロックされている。	D	削除のためにロックされている。
R	常駐としてマークされます。常駐とは、 <b>[Use]</b> 列の関連する値が 0（ゼロ）に変更され、オブジェクトが使用されないことを示す場合にも、オブジェクトがバッファプールから削除されないことを意味します。										
L	ロード機能の進行中でロックされている。										
D	オブジェクトの削除コールは保留中。 <b>[Use]</b> 列の値が 0（ゼロ）に変更されるとすぐに、オブジェクトは削除されます。										
L	ロード機能の進行中でロックされている。										
D	削除のためにロックされている。										

列	説明	
<b>Use</b>	バッファプールのみ。	
	現在オブジェクトを実行している Natural アプリケーションの数。	
	オブジェクトを選択するには、次のいずれかを指定します。	
	<i>value</i>	数値。 この番号を持つすべてのオブジェクトを選択します。 例：10 選択対象：10
	> <i>value</i>	開始数値 (>)。 <i>value</i> より大きい番号を持つすべてのオブジェクトを選択します。 例：>10 選択対象：11、21 選択対象外：10
< <i>value</i>	終了数値 (<)。 <i>value</i> より小さい番号を持つすべてのオブジェクトを選択します。 例：<10 選択対象：9、8 選択対象外：10	
<b>Max</b>	バッファプールのみ。  オブジェクトがバッファプールにロードされて以来、そのオブジェクトを実行したアプリケーションの最大数。  オブジェクトを選択するには、上記の「 <b>Use</b> 」にある有効な入力値を参照してください。	
<b>Reuse</b>	バッファプールキャッシュのみ。  オブジェクトがバッファプールキャッシュからバッファプールにロード（再利用）された回数を示します。  オブジェクトを選択するには、上記の「 <b>Use</b> 」にある有効な入力値を参照してください。	

## オブジェクトのリスト

列	説明	
<b>TotalUC</b>	総使用数。オブジェクトがバッファプールにロードされて以来のオブジェクトの検索コールの総数。	
	バッファプールキャッシュを使用している場合は、オブジェクトがバッファプールから削除されてバッファプールキャッシュに保存されても、この値は失われません。そのため、この値は、オブジェクトがシステムファイルからロードされて以来使用された回数を示します。	
	バッファプールオブジェクトについては、この値は定期的に更新されます。バッファプールキャッシュオブジェクトについては、この値はオブジェクトがバッファプールから削除され、バッファプールキャッシュに保存された後のみ更新されます。	
	オブジェクトを選択するには、次のいずれかを指定します。	
	<i>value</i> または > <i>value</i>	数値または開始数値 (>)。 <i>value</i> . 以上の番号を持つすべてのオブジェクトを選択します。 例：>10 選択対象：10、11、21 選択対象外：9
	< <i>value</i>	終了数値 (<)。 <i>value</i> 以下の番号を持つすべてのオブジェクトを選択します。 例：<10 選択対象：10、9、8 選択対象外：11
	<b>ObjSize</b>	オブジェクトのサイズ。  オブジェクトを選択するには、上記の「 <b>TotalUC</b> 」にある有効な入力値を参照してください。
	<b>Sto</b>	バッファプールまたはバッファプールキャッシュ内のオブジェクトに割り当てる必要があるストレージ。バッファプールのテキストレコードのサイズは、バッファプールの初期化時に定義します。  オブジェクトを選択するには、上記の「 <b>Use</b> 」にある有効な入力値を参照してください。
<b>[BP Load Time]</b> *	オブジェクトが最初にバッファプールにロードされた日時。  この日時は、オブジェクトがバッファプールとバッファプールキャッシュから削除されるまで保持されます。バッファプールキャッシュから削除しただけでは日時の表示は削除されません。  オブジェクトを選択するには、「 <b>日付の指定</b> 」と「 <b>時刻の指定</b> 」を参照してください。	

列	説明
[BP Last Action] *	バッファプールのみ。 オブジェクトがアプリケーションによって最後に使用された日時。 オブジェクトを選択するには、「日付の指定」と「時刻の指定」を参照してください。
[BPC Load Time] *	バッファプールキャッシュ (BPC) のみ。 オブジェクトが最初にバッファプールキャッシュにロードされた日時。 この日時は、オブジェクトがバッファプールキャッシュから削除されるまで保持されます。 オブジェクトを選択するには、「日付の指定」と「時刻の指定」を参照してください。
[BPC Last Get] *	バッファプールキャッシュ (BPC) のみ。 オブジェクトがバッファプールからバッファプールキャッシュに最後にスワップされた日時。このタイムスタンプは、オブジェクトがバッファプールキャッシュ内ですでに使用可能であったため、バッファプールキャッシュに再度書き込まれなかった場合も更新されます。 オブジェクトを選択するには、「日付の指定」と「時刻の指定」を参照してください。
[BPC Last Put] *	バッファプールキャッシュ (BPC) のみ。 オブジェクトがバッファプールキャッシュからバッファプールに最後にロードされた日時。 オブジェクトを選択するには、「日付の指定」と「時刻の指定」を参照してください。
[1.BPperiod] *	バッファプールキャッシュ (BPC) のみ。 オブジェクトが最初にロードされた時間で始まり、オブジェクトがバッファプールからバッファプールキャッシュに最初にスワップされた時間で終了する、オブジェクトがバッファプールで使用可能な期間。24 時間後、時間枠の表示はキャンセルされ、**:**:**.* の記号で置き換えられます。

\*これらの列を表示するには、PF11 キーを押します（「PF キーとダイレクトコマンド」を参照）。

このセクションでは、次のトピックについて説明します。

- 名前と範囲の指定
- 日付の指定

■ 時刻の指定

名前と範囲の指定

〔Library〕 および 〔Object〕 のいずれかあるいはその両方の入力フィールドに名前または名前の範囲を入力して、〔List Objects〕 画面に表示されるオブジェクトのリストを短くすることができます。

下記のオプションのリストで、*value* は 1 つ以上の文字の任意の組み合わせを表します。

入力値	選択されるライブラリ／オブジェクト
*	すべてのライブラリ／オブジェクト。 これはデフォルトです。
<i>value</i>	<i>value</i> と同じ名前を持つすべてのライブラリ／オブジェクト。
<i>value</i> *	<i>value</i> で始まる名前を持つすべてのライブラリ／オブジェクト。 例：AB* 選択対象：AB、AB1、ABC、ABEZ 選択対象外：AA1、ACB
<i>value</i> ?	<i>value</i> で始まり、入力された疑問符 (?) の個数に相当する任意の文字数がある後に続く名前を持つすべてのライブラリ／オブジェクト。 例：ABC? 選択対象：ABCA、ABCZ 選択対象外：AXC、ABCAA
<i>value</i> ? <i>value</i> ?	任意の順序でアスタリスク (*) と疑問符 (?) を指定した <i>value</i> に一致するすべての項目。
<i>value</i> * <i>value</i> ?	
* <i>value</i> ? <i>value</i> *	例：A?C*Z 選択対象：ABCZ、AXCBBBZ、ANCZ 選択対象外：ACBZ、ABDEZ、AXCBBBZA
<i>value</i> >	<i>value</i> 以降の名前を持つすべてのライブラリ／オブジェクト。 例：AB> 選択対象：AB、AB1、BBB、ZZZZZZZ 選択対象外：AA1、AAB
<i>value</i> <	<i>value</i> . 以前の名前を持つすべてのライブラリ／オブジェクト。 例：AX< 選択対象：AB、AWW、AX 選択対象外：AXA、AY



## 日付の指定

日付の入力フィールドに、日付、日付の範囲、特殊な日付、または特殊な日付の範囲を入力して、**[List Objects]** 画面に表示されるオブジェクトのリストを短くすることができます。

日付は、YYYYMMDD (YYYY=年、MM=月、DD=日) の形式で指定する必要があります。

下記のオプションのリストで、*value* は 1 桁以上の数字の任意の組み合わせを表します。

入力値	選択されるオブジェクト	
YYYYMMDD	YYYYMMDD と同じ日付のすべてのオブジェクト。 例：20070630	
<i>value</i> *	<i>value</i> で始まる日付のすべてのオブジェクト。 例：2007* 選択対象：20070101～20071231 選択対象外：20061231、20080101	
<i>value</i> >	<i>value</i> 以降の日付のすべてのオブジェクト。 例：2007> 選択対象：20070101～20070101 選択対象外：20061231	
<i>value</i> <	<i>value</i> より前の日付のすべてのオブジェクト。 例：2007< 選択対象：20060101～20061231 選択対象外：20070101、20071231	
特殊な日付の値		
TODAY または T0+/-n	現在の日付、あるいは現在の1日前または1日後の日付のすべてのオブジェクト。	
	TODAY	現在の日付のすべてのオブジェクト。
	T0+/-n	現在の日付に <i>n</i> 日分を加算または減算した日付のすべてのオブジェクト。  例：現在の日付は 2007 年 6 月 30 日です。 T0-5 では、20070625 が選択されます。
YESTERDAY	現在の日付の 1 日前の日付のすべてのオブジェクト。  例：現在の日付は 2007 年 6 月 30 日です。 YESTERDAY では、20070629 が選択されます。	
MONTH	当月の日付のすべてのオブジェクト。  例：当月は 2007 年 6 月です。 MONTH では、20070601～20070630 が選択されます。	

## オブジェクトのリスト

---

入力値	選択されるオブジェクト
YEAR	当年の日付のすべてのオブジェクト。  例：当年は 2007 年です。 YEAR では、20070101～20071231 が選択されます。

### 時刻の指定

時刻の入力フィールドに時刻または時刻の範囲を入力して、「**List Objects**」画面に表示されるオブジェクトのリストを短くすることができます。

時刻は、*HH:II:SS* (*HH*=時間、*II*=分、*SS*=秒) の形式で指定する必要があります。

次のオプションのリストで、*value* は 1 桁以上の数字の任意の組み合わせを指定できます。

入力値	選択されるオブジェクト
<i>HH:II:SS</i>	<i>HH:II:SS</i> と同じ時刻のすべてのオブジェクト。  例：14:15:16
<i>value*</i>	<i>value.</i> で始まる時刻のすべてのオブジェクト。  例：13* 選択対象：13:00:00、13:10:53、13:59:59 選択対象外：12:59:59、14:00:00
<i>value&gt;</i>	<i>value</i> 以降の時刻のすべてのオブジェクト。  例：12:30> 選択対象：12:30:00、12:30:01、16:34:01 選択対象外：12:29:59
<i>value&lt;</i>	<i>value</i> より前の時刻のすべてのオブジェクト。  例：12:30< 選択対象：12:29:59 選択対象外：12:30:00

## PF キーとダイレクトコマンド

[List Objects] 画面で、次の表に示す PF キーや SYSBPM ダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、「[SYSBPM ダイレクトコマンド](#)」を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ヘルプ情報を表示します。 [C] 列上のカーソルで指定した場合、 使用可能なすべてのコマンドと機能が表示されます。  列タイトルの下にある入力フィールド上のカーソルで指定した場合、オブジェクトの 選択に使用可能なすべての入力値が表示されます。
PF3	<u>EXIT</u>	現在の機能／画面を終了して、前の画面を表示します。
PF4	LAST	最後に入力した SYSBPM ダイレクトコマンドを表示します。
PF5	<u>CACHE</u>	バッファプールキャッシュデータが存在する場合にのみ適用されます。  バッファプールキャッシュオブジェクトの統計データが含まれるリストの上部にスク ロールします。
PF6	-	バッファプールオブジェクトの統計データが含まれるリストの上部にスクロールしま す。
PF7	-	リストを 1 ページ上方にスクロールします。
PF8	+	リストを 1 ページ下方にスクロールします。
PF9	++	リストの末尾までスクロールします。
PF10	<	リストを左にスクロールします。
	LEFT	右にスクロールするには、PF11 キーを押します。
PF11	>	リストを右にスクロールし、[BP Load Time]、[BP Last Action]、[BPC Load Time]、[BPC Last Get]、[BPC Last Put] および [1.BPperiod] の追加の画面列 を表示します。  左にスクロールするには、PF10 キーを押します。
	RIGHT	
PF12	<u>CANCEL</u>	EXIT と同じです。
PF15	MENU	SYSBPM メインメニューを呼び出します。

## 行コマンド

【List Objects】画面の【C】列で、表示される各オブジェクトに対して、次に示すいずれかの行コマンドを入力できます。

コマンド	機能
CL	バッファプールのみ。 常駐としてマークされたオブジェクトのマークを解除します。
DE	バッファプールまたはバッファプールキャッシュから削除するオブジェクトをマークします。 オブジェクトは、関連する【Use】カウントが0（オブジェクトは使用されない）に変更されるとすぐに削除されます。バッファプールオブジェクトに対して発行した場合、オブジェクトはバッファプールとバッファプールキャッシュの両方から削除されます。バッファプールキャッシュオブジェクトに対して発行した場合、オブジェクトはバッファプールキャッシュからのみ削除されます。
HD	バッファプールのみ。 オブジェクトのディレクトリ情報を16進形式で表示します。
HE	バッファプールのみ。 関連するセクションで説明しているように、【Hexadecimal Display】機能に対応します。
F0	バッファプールのみ。 関連する【Use】カウントに関係なく、バッファプールからオブジェクトを直ちに削除します。
LD	関連するセクションで説明しているように、【Hexadecimal Display】機能に対応します。
RE	バッファプールのみ。 オブジェクトを常駐としてマークします。
Z0	【Object】、【Use】、【Max】、【Reuse】、【TotalUC】、【ObjSize】および【Sto】フィールドにズームインし、完全な長さで表示します。 ズームアウトするには、Enter キーを押します。

入力したコマンドごとに、画面に表示される行の行上書きテキストに関する確認メッセージが表示されます。表示されるメッセージ：

- Failed（正常に実行されなかったすべての機能に対する応答）
- Deleted（DE または F0 コマンドに対する応答）
- Released（CL コマンドに対する応答）
- Resident（RE コマンドに対する応答）

# 57 オブジェクトの削除

---

この機能では、バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方から1つまたは複数のオブジェクトを削除します。オブジェクトを削除するオブジェクトプールを指定するには、「SYSBPM ダイレクトコマンド」の説明に従って、**DELETE** コマンドを使用します。

**[Current Use Count]**（「ディレクトリ情報」を参照）が0（ゼロ）のオブジェクトは直ちに削除されます。0は、そのようなオブジェクトが使用されないことを表します。**[Current Use Count]** が0より大きいオブジェクトは、削除対象としてマークされ、**[Current Use Count]** が0に変更されるとすぐに削除されます。

## ▶手順 57.1. [Delete Objects] を呼び出すには

- SYSBPMメインメニューで、ファンクションコード「D」を入力し、削除するオブジェクトを指定します。有効な**フィールド入力値**の詳細については、「SYSBPMの呼び出しと操作」を参照してください。

または:

次のSYSBPM ダイレクトコマンドを入力します。

```
DELETE library-name object-name dbid fnr
```



## 58 ディレクトリ情報

---

- バッファプールオブジェクトのフィールド ..... 331
- バッファプールキャッシュオブジェクトのフィールド ..... 332
- PF キーとダイレクトコマンド ..... 333

この機能は、バッファプールまたはバッファプールキャッシュに現在ロードされている Natural オブジェクトの完全なディレクトリを表示する場合に使用します。

### ▶手順 58.1. [Directory Information] を呼び出すには

- SYSBPM メインメニューで、次のように指定します。

ファンクションコード「I」を入力します。

[Object]、[Library]、[DBID]、および [FNR] の各フィールドで、ディレクトリを表示するオブジェクトを指定します。有効な入力値の詳細については、「SYSBPM の呼び出しと操作」を参照してください。

[Object Pool] フィールドに、オブジェクトディレクトリ情報の読み取り元オブジェクトプールのコードを入力します。

B または * (アスタリスク)	バッファプール
C	バッファプールキャッシュ

または:

次のいずれかの SYSBPM ダイレクトコマンドを使用します。

- `DISPLAY DIRECTORY library-name object-name dbid fnr`

(バッファプールにロードされた指定のオブジェクトのディレクトリに対して)

- `DISPLAY CDIRECTORY library-name object-name dbid fnr`

(バッファプールキャッシュにロードされた指定のオブジェクトのディレクトリに対して)

[Directory Information] 画面が表示されます。選択したオブジェクトプール (バッファプールまたはバッファプールキャッシュ) に応じて、画面には異なるフィールドが表示されます。詳細については、次のセクションを参照してください。



## バッファプールオブジェクトのフィールド

バッファプールにロードされたオブジェクトのディレクトリの場合、**[Directory Information]** 画面には指定したオブジェクトに関する次のフィールドと情報が表示されます。

フィールド	説明	
<b>Directory of</b>	オブジェクトのタイプ（マップなど）と名前を入力します。	
<b>Loaded</b>	<b>from Library</b>	バッファプールにロードされたオブジェクトのロード元のライブラリの名前。
	<b>on DBID/FNR</b>	バッファプールにロードされたオブジェクトのロード元の FNAT または FUSER システムファイルのデータベース ID (DBID) とファイル番号 (FNR)。
	<b>on</b>	オブジェクトがバッファプールにロードされた日時。
	<b>by User</b>	オブジェクトを実行したユーザーの ID。
<b>Last Action on</b>	オブジェクトがアプリケーションによって最後に使用された日時。	
<b>BP Directory at Address</b>	バッファプール内のオブジェクトのディレクトリのアドレス。	
<b>Object at Address</b>	バッファプール内のオブジェクトのアドレス。	
<b>Allocated Size (KB)</b>	オブジェクトのバッファプールに割り当てられたサイズ。	
<b>Object Size</b>	オブジェクトのサイズ。	
<b>Status (RLD)</b>	オブジェクトのステータス：	
	R	オブジェクトはバッファプール内に常駐。 常駐とは、 <b>[Current Use Count]</b> （次を参照）が 0（ゼロ）に変更された場合にも、オブジェクトがバッファプールから削除されないことを意味します。
	L	オブジェクトは現在ロック状態。
	D	オブジェクトの削除コールは保留中。オブジェクトは、 <b>[Current Use Count]</b> （次を参照）が 0（ゼロ）に変更されるとすぐにバッファプールから削除されます。
<b>Current Use Count</b>	現在オブジェクトを実行しているアプリケーションの数。  0（ゼロ）の値は、オブジェクトが使用されないことを表します。	
<b>Maximum Use Count</b>	オブジェクトがバッファプールにロードされて以来、そのオブジェクトを実行したアプリケーションの最大数。	

フィールド	説明
BP Total Use	オブジェクトがシステムファイルからバッファプールにロードされて以来実行された合計回数。  バッファプールキャッシュを使用している場合は、オブジェクトがバッファプールから削除されてバッファプールキャッシュに保存されても、この値は失われません。そのため、この値は、オブジェクトがシステムファイルからロードされて以来使用された回数を示します。
Cataloged	[ <b>Directory Information</b> ] 画面の [ <b>Cataloged</b> ] セクションに表示される情報は、『システムコマンド』ドキュメントで説明している Natural システムコマンド LIST DIRECTORY で表示される情報と同じです。

## バッファプールキャッシュオブジェクトのフィールド

バッファプールキャッシュにロードされたオブジェクトのディレクトリの場合、 [**Directory Information**] 画面には指定したオブジェクトに関する次のフィールドと情報が表示されます。

フィールド	説明
<i>object-type</i>	オブジェクトのタイプ（マップなど）と名前を入力します。
Library	バッファプールにロードされたオブジェクトのロード元のライブラリの名前。
DBID	オブジェクトがバッファプールに最初にロードされたロード元の FNAT または FUSER システムファイルのデータベース ID。
FNR	オブジェクトがバッファプールに最初にロードされたロード元の FNAT または FUSER システムファイルのファイル番号。
Last Put	オブジェクトがバッファプールキャッシュからバッファプールに最後にロードされた日時。
BP Load Time	オブジェクトが最初にバッファプールにロードされた日時。
Cache Load Time	オブジェクトが最初にバッファプールキャッシュにロードされた日時。
Last Get	オブジェクトがバッファプールからバッファプールキャッシュに最後にスワップされた日時。
Position Index	バッファプールキャッシュ内のオブジェクトの連続的に番号付けられた内部 Natural 位置インデックス。
First Data Block Offset	バッファプールキャッシュ内のオブジェクトのディレクトリのアドレス。
Allocated Size (KB)	オブジェクトのバッファプールキャッシュに割り当てられたサイズ。
Object Size	オブジェクトのサイズ。
Status	オブジェクトのステータス：  L                      ロード機能の進行中でロックされている。

フィールド	説明
	D   削除のためにロックされている。
Reuse	オブジェクトがバッファプールキャッシュからバッファプールに返された回数を示します。
BP Total Use	オブジェクトが、システムファイルから最初にバッファプールにロードされ、次にバッファプールキャッシュにロードされて以来実行された合計回数。
Cataloged	『 <b>Directory Information</b> 』画面の『 <b>Cataloged</b> 』セクションに表示される情報は、『システムコマンド』ドキュメントで説明している Natural システムコマンド LIST DIRECTORY で表示される情報と同じです。

## PF キーとダイレクトコマンド

『**Directory Information**』画面で、次の表に示す PF キーまたは SYSBPM ダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、『**SYSBPM ダイレクトコマンド**』を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ダイレクトコマンドに関するヘルプ情報を表示します。
PF2	NEXT	オブジェクトの範囲を選択した場合にのみ適用されます。オブジェクトを順に表示してから、NEXT コマンドが入力された画面を再表示します。
PF3	EXIT	現在の機能／画面を終了して、前の画面を表示します。
PF4	LAST	最後に入力した SYSBPM ダイレクトコマンドを表示します。
PF6	FLIP	PF キー行を切り替えます。PF1～PF12 キーと PF13～PF24 キーの間で表示を切り替えます。
PF12	CANCEL	EXIT と同じです。
PF15	MENU	SYSBPM メインメニューを呼び出します。
	FDELETE	バッファプールのみ。  [Use] カウントに関係なく、バッファプールからオブジェクトを直ちに削除します。
	RESIDENT	バッファプールのみ。  オブジェクトを常駐としてマークします。常駐とは、[Use] カウントが 0 (オブジェクトは使用されない) の場合にも、オブジェクトがバッファプールから削除されないことを意味します。
	CLEAR	バッファプールのみ。  常駐としてマークされたオブジェクトのマークを解除します。
	DELETE	オブジェクトに削除のマークを付けます。前のセクションで説明した <b>バッファプール</b> または <b>バッファプールキャッシュ</b> の「ステータス D」を参照してください。



# 59

## 16進表示

---

- PF キーとダイレクトコマンド ..... 336

この機能は、バッファプールに現在ロードされている Natural オブジェクトのコードを 16 進形式で表示する場合に使用します。

#### ▶手順 59.1. [Hexadecimal Display] を呼び出すには

- SYSBPM メインメニューで、ファンクションコード「H」を入力し、オブジェクトを指定します。有効なフィールド入力値の詳細については、「SYSBPM の呼び出しと操作」セクションを参照してください。

または:

次の SYSPBM ダイレクトコマンドを入力します。

```
DISPLAY HEX library-name object-name dbid fnr
```

[Hexadecimal Display] 画面が開き、オブジェクトコードが 16 進形式で表示されます。

## PF キーとダイレクトコマンド

画面に表示されたオブジェクト内で、現在の場所に関連する絶対 16 進アドレスまたは 16 進オフセットのいずれかを入力して、特定の位置に移動することができます。

[Hexadecimal Display] 画面で、次の表に示す PF キーまたは SYSBPM ダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、「SYSBPM ダイレクトコマンド」を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ダイレクトコマンドに関するヘルプ情報を表示します。
PF2	NEXT	オブジェクトの範囲を選択した場合にのみ適用されます。 オブジェクトを順に表示してから、NEXT コマンドが入力された画面を再表示します。
PF3	EXIT	現在の機能/画面を終了して、前の画面を表示します。
PF4	LAST	最後に入力した SYSBPM ダイレクトコマンドを表示します。
PF6	-	表示の上部にスクロールします。
PF7	-	1 ページ上方にスクロールします。
PF8	+	1 ページ下方にスクロールします。
PF9	++	表示の最後までスクロールします。
PF12	CANCEL	EXIT と同じです。
PF15	MENU	SYSBPM メインメニューを呼び出します。
	NEXT	オブジェクトの範囲を選択した場合にのみ適用されます。 オブジェクトを順に表示してから、NEXT コマンドが入力された画面を再表示します。

# 60 ワークファイルへの書き込み

---

この機能では、現在バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方にロードされている Natural オブジェクトのディレクトリ情報をワークファイルへ書き込みます。

## ▶手順 60.1. [Write to Work File] を呼び出すには

- 1 SYSBPM メインメニューで、次の情報を入力します。
  - [Code] フィールドに、ファンクションコード「W」を入力します。
  - [Object Pool] フィールドに、オブジェクトディレクトリ情報の読み取り元オブジェクトプールのコードを入力します。

B	バッファプール。
C	バッファプールキャッシュ。
* (アスタリスク)	バッファプールとバッファプールキャッシュ。

または:

次のいずれかの SYSBPM ダイレクトコマンドを入力します。

### ■ WRITE ALL

(バッファプールとバッファプールキャッシュのデータを保存)

### ■ WRITE BP

(バッファプールのデータのみ保存)

### ■ WRITE BPC

(バッファプールのデータのみ保存)

[**Work File Selection**] ウィンドウが表示されます。

2 次の情報を指定します。

- 対象のワークファイルを選択します。

「N」（No。デフォルト設定）を入力して、Natural Work File 1 にデータを出力します。

または、

Entire Connection がインストールされている場合は、「Y」（Yes）を入力し、Natural Work File 7 を使用して PC テキストファイルにデータを出力します。

- ワークファイル内の列を区切るために使用する、セミコロン (;) などのデリミタ文字を入力します。デフォルトは空白文字です。

ワークファイルに書き込まれる統計データは、[**List Objects**] 機能によって生成されたリストのスナップショットです。列の詳細については、「[オブジェクトのリスト](#)」を参照してください。

PC テキストファイルは、スプレッドシート計算のベースとして使用できます。



# 61 ソート済み抽出の表示

この機能では、現在バッファプールにロードされている Natural オブジェクトの 50 のディレクトリエントリのソート済み抽出を生成します。このリストは関連するセクションで説明しているように、常駐としてマークするオブジェクトやプリロードリストに含めるオブジェクトを決める場合など、評価の目的で使用できます。

## ▶手順 61.1. [Display Sorted Extract] を呼び出すには

- SYSBPM メインメニューで、ファンクションコード「X」を入力し、[Object Pool] フィールドで、「B」（バッファプール）または「C」（バッファプールキャッシュ）を選択して、プールのタイプを指定します。

[Specify Sort Criteria] ウィンドウが表示されます。入力フィールドに、次のいずれかのコードを入力して、列および統計列をソートする順序を選択します。

ソート値	0	[ObjSize] = Object Size. これはデフォルトです。
	T	TotalUC=Total Use Count. 詳細については、「ディレクトリ情報」の「BP Total Use」を参照してください。
	L	[BP Last Action] (バッファプールにのみ適用されます)。オブジェクトがアプリケーションによって最後に使用された日時。
ソート順	D	降順。これはデフォルトです。
	A	昇順。

または:

SYSBPM ダイレクトコマンド SORT または SORT BPC を使用します。詳細については、「SYSBPM ダイレクトコマンド」を参照してください。

[BP Extract] 画面が表示され、指定したソート条件が示されます。

[BP Extract] 画面は、[List Objects] 画面に類似しています。列および適用されるコマンドの詳細については、「[オブジェクトのリスト](#)」を参照してください。

## 62 バッファプール統計

---

■ 一般バッファプール統計 .....	342
■ バッファプールのロード／ロケート統計 .....	345
■ バッファプールフラグメント .....	349
■ 内部機能使用率 .....	351
■ バッファプールのハッシュテーブル統計 .....	351
■ パフォーマンスに関するヒント .....	354
■ PF キーとダイレクトコマンド .....	358

【**Buffer Pool**】機能では、【**Buffer Pool Statistics**】メニューを呼び出します。このメニューは、Naturalオブジェクトとは無関係のバッファプール関連の統計（ハッシュテーブル統計を含む）を取得する場合に使用します。

### ▶手順 62.1. 【**Buffer Pools Statistics**】を呼び出すには

- SYSBPM メインメニューで、次のファンクションコードを入力します。

```
A
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY STATISTICS
```

【**Buffer Pool Statistics**】メニューが表示されます。

このセクションでは、【**Buffer Pool Statistics**】メニューで使用可能な機能、およびそれらの機能により呼び出される画面に表示されるコマンドについて説明します。

## 一般バッファプール統計

---

この機能は、バッファプールのパフォーマンスをモニタし、バッファプールの動作に関する統計を表示する場合に使用します。

### ▶手順 62.2. 【**General Buffer Pool Statistics**】を呼び出すには

- 【**Buffer Pool Statistics**】メニューで、次のファンクションコードを入力します。

```
G
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY GENERAL
```

【**General Buffer Pool Statistics**】画面が表示されます。

[General Buffer Pool Statistics] 画面に表示される統計は、バッファプールのスナップショットであり、Enter キーを押すたびに更新されます。次の情報が表示されます。

フィールド	説明
Buffer Pool Address	バッファプールのアドレス。
Directory Section	<p>バッファプールの先頭に対するバッファプールディレクトリセクションのアドレス</p> <p>バッファプールにロードされた各オブジェクトには、そのオブジェクトに関する情報を含むディレクトリエントリが必要です。これらのディレクトリエントリのためのスペースは、バッファプールに割り当てられます。</p>
Text Record Section	<p>バッファプールの先頭に対するテキストレコードのアドレス。</p> <p>ディレクトリエントリによって使用されるスペースが割り当てられた後、残りのスペースは、テキストレコード（サイズはデフォルトで4KB）と呼ばれるブロックに分割されます。オブジェクトは、サイズに応じて1つまたは複数のテキストレコードを占有できます。</p>
Dataspace attached	バッファプールにアタッチされたデータスペース（バッファプールキャッシュ）の名前。
Buffer Pool Size (MB)	<p>バッファプール全体のサイズ（MB 単位）。</p> <p>バッファプールのサイズは、パラメータモジュールのNTBPIマクロ、または『パラメータリファレンス』ドキュメントで説明されているBPIプロファイルパラメータで指定できます。</p>
Directory Entry Size	ディレクトリエントリのサイズ（バイト単位）。
Text Record Size (KB)	<p>テキストレコードのサイズ（KB 単位）。テキストレコードのサイズは、パラメータモジュールのNTBPIマクロ、または『パラメータリファレンス』ドキュメントで説明されているBPIプロファイルパラメータで指定できます。</p> <p>INITIALIZE コマンドを使用してバッファプールを再初期化すると、既存のバッファプールのテキストレコードのサイズを変更することができます。</p> <p>デフォルトのテキストレコードのサイズは、4KBに設定されています。ただし、多くの小さいオブジェクトで構成されるアプリケーションを使用する場合は、2KBに減らすことをお勧めします。これにより、バッファプール内の未使用のスペースの比率が減少しますが、アルゴリズム2（『オペレーション』ドキュメントの「METHOD=S」を参照）がより頻繁に呼び出されるようになります。</p>
Buffer Pool Start	バッファプールが最初に開始された日時。
Last Initialization	<p>バッファプールが最後に初期化された日時、および初期化を実行したユーザーの ID。</p> <p>バッファプールは次の場合に初期化されます。</p> <ul style="list-style-type: none"> <li>■ バッファプールの最初の開始、</li> <li>■ INITIALIZE SYSBPM ダイレクトコマンドの実行、または</li> </ul>

フィールド	説明
	<p>■ GBPオペレーティングプログラムのREFRESHコマンドの実行。詳細については、『オペレーション』ドキュメントの「グローバルバッファプールのオペレーティング機能」も参照してください。</p>
Text Records - Total	テキストレコードの総数。
Text Records - Used	現在使用されているテキストレコードの数。
Text Records - Used in %	現在使用されているテキストレコードの比率。
Text Records - Max Used	使用されるテキストレコードの最大数。
Text Records - Total Size	<p>使用されるすべてのテキストレコードが使用するスペースの合計。 [Text Records - Used] に1つのレコードのサイズを掛けたもの。</p> <p>テキストレコードサイズの合計とオブジェクトサイズの合計との差は、テキストレコードセクションの未使用のサイズを表し、システム管理者がテキストレコードのサイズを変更するかどうかの指標とすることもできます。</p>
Text Records - Avg Usage %	<p>使用されるすべてのテキストレコードの平均使用率。 [Objects - Total Size] を [Text Records - Total Size] で割ったもの。</p> <p>この値は、75%を著しく下回らないようにしてください。バッファプールがほとんどいっぱいの場合、75%を超える値は、バッファプールが適切に使用されていることを示します。使用率が75%を著しく下回る場合は、テキストレコードのサイズを縮小する必要があります。</p>
Space Used %	<p>テキストレコードセクションの実際の使用率。 [Objects - Total Size] を [Text Records] セクションの合計サイズで割ったもの。</p> <p><b>ヒント:</b> バッファプールがほとんどいっぱい、つまり、 [Text Records - Used] フィールドの値がほぼ100%の場合、75%を超える値は、バッファプールが適切に使用されていることを示します。使用率が75%を著しく下回る場合は、テキストレコードのサイズを縮小する必要があります。</p>
Objects - Loaded	現在バッファプールにロードされているオブジェクトの数。
Objects - Max Loaded	バッファプールの開始以降、バッファプールにこれまで同時にロードされたオブジェクトの最大数。
Objects - Total Size	現在ロードされているオブジェクトの合計サイズ (バイト単位)。
Objects - Avg TR Used	1つのオブジェクトが使用するテキストレコードの平均数。
Objects - SumOfUseCounts	<p>現在バッファプールにロードされているすべてのオブジェクトの使用回数の合計。</p> <p>使用回数では、現在オブジェクトを実行しているすべてのアプリケーションがカウントされます。オブジェクトが現在使用されていない場合、そのオブジェクトの使用回数は0 (ゼロ) に変わります。</p>
Objects - AvgLifetimeUsed(min)	現在バッファプールにロードされているオブジェクトの平均存続時間 (分単位)。
Objects - AvgLifetimeReplace(min)	バッファプールですでに置き換えられているオブジェクトの平均存続時間 (分単位)。

## バッファプールのロード／ロケート統計

この機能では、バッファプールへのオブジェクトのロードおよびバッファプールでのオブジェクトの検索に関する統計情報が提供されています。この情報は、バッファプールのパフォーマンスの指標として使用することもできます。

### ▶手順 62.3. [Buffer Pool Load/Locate Statistics] を呼び出すには

- [Buffer Pool Statistics] メニューで、次のファンクションコードを入力します。

```
L
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY LOAD
```

[Buffer Pool Load/Locate Statistics] 画面が表示されます。

[Buffer Pool Load/Locate Statistics] 画面に表示される統計は、バッファプールのスナップショットであり、Enter キーを押すたびに更新されます。

画面には次の情報が表示されます。

フィールド	説明
Total Locate Calls	オブジェクト検索コールの合計数。つまり、Natural バッファプールマネージャがバッファプールでオブジェクトを検索するよう要求された合計回数。  検索が正常に終了した場合、オブジェクトはバッファプールまたはバッファプールキャッシュ内で検出され、Natural システムファイルからのロードは不要となり、コールおよび I/O が回避されます。
Total Locate Calls - successful	正常に終了した検索コールの絶対数としての合計数。
Total Locate Calls - failed	失敗した検索コールの合計数。
Quick Locate Calls	高速検索コールの合計数。  高速検索コールでは、内部高速検索テーブルを使用して、バッファプール内のオブジェクトを検索します。詳細については、「 <a href="#">内部高速検索テーブル</a> 」を参照してください。  この場合、ライブラリ（名前、データベース ID、ファイル番号）、および前回正常に終了した検索コールのバッファプール内のオブジェクトの場所が、オブジェクトの検索に再度使用されます。  これは、オブジェクトを検索するための最も効果的な方法です。

フィールド	説明
Quick Locate Calls - successful	正常に実行された高速検索コールの数。
Quick Locate Calls - failed	失敗した高速検索コールの数。  失敗した高速検索コールは、内部高速検索テーブルを使用して、オブジェクトがバッファプール内の前回の位置に見つからなかったことを示します（「パフォーマンスの考慮事項」の関連するセクションを参照）。  これは、オブジェクトがバッファプールから削除されたか、または別の位置に移動された場合に発生します。高速検索コールの失敗により、通常検索コールが発生しますが、Natural ランタイムではオブジェクトを含むライブラリが記憶されているため、steplib 検索は実行されません。
Normal after Quick	[Quick Locate Calls - failed] が原因で発生した通常検索コールの数。  [Normal after Quick] の値は、常に [Quick Locate Calls - failed] の数と同じです。
Normal after Quick - successful	[Quick Locate Calls - failed] が原因で発生し、正常に終了した通常検索コールの数。  高速後の通常コールは、要求されたオブジェクトがまだバッファプール内にあるが、別の位置にある場合に正常に終了します。
Normal after Quick - failed	[Quick Locate Calls - failed] が原因で発生し、失敗した通常検索コールの数。  高速後の通常コールの失敗は、要求されたオブジェクトがバッファプールで使用できなくなったことを示します。その結果、オブジェクトは、内部高速検索テーブル内のライブラリエントリを使用して、システムファイルから再ロードされます（「パフォーマンスの考慮事項」の関連するセクションを参照）。
Normal Locate Calls	通常検索コールの合計数。  通常検索コールでは、内部高速検索テーブルを使用せずに、バッファプール内のオブジェクトが検索されます（「パフォーマンスの考慮事項」の関連するセクションを参照）。  通常検索コールは、LOGON システムコマンドが実行された後に、Natural セッション内で最初に参照された場合に常に発生します。高速検索コールと比較して、Natural ランタイムでは、オブジェクトが存在するライブラリはまだ認識されません。
Normal Locate Calls - successful	バッファプール内の必要なオブジェクトの検索が正常に終了した通常検索コールの数。
Normal Locate Calls - failed	失敗した通常検索コールの数。  失敗した通常検索コールは、名前および存在するライブラリによって識別されたオブジェクトが、バッファプール内に見つからなかったことを示します。



フィールド	説明
	失敗した通常検索は、バッファプールキャッシュまたはシステムファイルからのロード、あるいは <code>steplib</code> チェーン内の次のライブラリの通常検索コールによって発生します。
<b>STEPLIB Searches</b>	<p><code>steplib</code> ライブラリでオブジェクトの検出に失敗したために発生した <b>[Normal Locate Calls]</b> の数。</p> <p><code>steplib</code> 検索が正常に実行されなかった通常検索コールでは、バッファプールキャッシュまたはシステムファイルからオブジェクトはロードされません。</p> <p><b>[STEPLIB Searches]</b> では、現在のライブラリにも、どの <code>steplib</code> やシステムファイルにも含まれていないオブジェクトの検索コールはカウントされません（エラーメッセージ「不正なコマンドです。またはライブラリにプログラムがありません」）。不正なプログラミングのために失敗した検索コールは、<b>[Normal Locate Calls - failed]</b> のカウンタに追加されます。</p> <p><b>[STEPLIB Searches]</b> の数は、次の公式を使用して計算されます。</p> <p><b>[Normal Locate Calls - failed]</b> - ( <b>[Number Loads into BP]</b> - <b>[Normal after Quick - failed]</b> )</p> <p><b>[STEPLIB Searches]</b> の数が少ないほど、バッファプールのパフォーマンスは向上します。</p> <p>「<a href="#">Steplibs での検索</a>」も参照してください。</p>
<b>Number Loads into BP</b>	<p>バッファプールへのロードが正常に実行された回数。</p> <p>バッファプールへのロード（ストレージ割当要求）は、データベースからのロードまたはバッファプールキャッシュからのロードによってトリガすることができます。</p>
<b>Loads from Cache</b>	<p>バッファプールキャッシュ内に存在するオブジェクトの正常に完了した検索コールの合計数。この情報は、前の検索コール（<b>[Normal after Quick - failed]</b> または <b>[Normal Locate Calls - failed]</b>）が失敗した場合にのみカウントされます。保存されたデータベースのロードの数を表します。つまり、バッファプールキャッシュがない場合、オブジェクトはデータベースからロードする必要があります。</p>
<b>Loads from DB</b>	<p>オブジェクトが Natural システムファイルからバッファプールにロードされた回数。</p> <p>1つのオブジェクトをロードするために複数のロードコールが必要な場合があるため、この値は、最新のバッファプール更新以降に実行されたオブジェクトのロードの実際の数となります。</p> <p>オブジェクトをロードする場合、バッファプールマネージャでは異なる検索アルゴリズムが使用されます。『オペレーション』ドキュメントの「<code>METHOD=S</code>」および「<code>METHOD=N</code>」を参照してください。</p>

フィールド	説明
<p><b>Loads from DB - finished</b></p>	<p>正常に終了したオブジェクトのロードの数。</p> <p>次のいずれかの理由によってロード操作がキャンセルされた場合、オブジェクトのロードは終了できません。</p> <ul style="list-style-type: none"> <li>■ オブジェクトの同時ロードが発生した。 [<b>Loads from DB - concurrent</b>] を参照してください。</li> <li>■ オブジェクトのロード中、Adabas レスポンスコードが返される。</li> <li>■ オブジェクトのロード中、SYSBPM 削除操作がそのオブジェクトに対して実行される。</li> </ul>
<p><b>Loads from DB - concurrent</b></p>	<p>同じオブジェクトに対して同時に実行されたオブジェクトのロードの数。</p> <p>オブジェクトの同時ロードは、同時に実行されている複数の Natural セッションが同じオブジェクトを要求した場合に発生します。オブジェクトが1つのセッションによってロードされている間、他のセッションが同じオブジェクトを要求し、ロードが完了する前にオブジェクトのロードを開始します。この場合、同じオブジェクトが複数回ロードされます。</p> <p>オブジェクトのロードを終了する最初のセッションは、他のセッションのオブジェクトをバッファプールからの削除対象としてマークします。その場合、他のセッションはオブジェクトのロードを中止し、削除対象としてマークされたオブジェクトをバッファプールから削除して、最初のセッションによって正常にロードされたオブジェクトを使用します。</p> <p>カウンタ [<b>Loads from DB - finished</b>] および [<b>Loads from DB - concurrent</b>] によって計算されたオブジェクトの数は、通常同じです。この数は、両方のセッションがロードを終了した後に初めて同時ロードが検出された場合にのみ異なります。</p>
<p><b>Load Calls</b></p>	<p>バッファプールが更新されて以降に実行されたロードコールの合計数。ロードコールは、オブジェクトの読み込み元のシステムファイルへのアクセスに関連付けられます。</p> <p>システムファイルアクセスの数は、次のように計算されます。</p> <ul style="list-style-type: none"> <li>■ Adabas システムファイル： [<b>Load Calls</b>] の数 + [<b>Loads from DB</b>]（上記参照）。合計数には、Adabas RC コールは含まれません。</li> <li>■ VSAM システムファイル： [<b>Load Calls</b>] の数。</li> </ul>
<p><b>Number Loads BP 2nd</b></p>	<p>このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=S（選択プロセス）を使用している場合に表示されます。</p> <p>このフィールドには、ストレージ割り当て要求がアルゴリズム2の検索条件を満たした回数が表示されます。詳細については、『オペレーション』ドキュメントの「METHOD=S」を参照してください。</p>

フィールド	説明
Number Load Cycles	このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=N（次に使用可能）を使用している場合に表示されます。『オペレーション』ドキュメントを参照してください。  このフィールドは、検索がバッファプールの先頭から実行された回数を示します。その数は、バッファプール内のサイクルの頻度の見積りをラップアラウンド方式で示します。
Last Cycle Start	このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=N（次に使用可能）を使用している場合に表示されます。『オペレーション』ドキュメントを参照してください。  [Number Load Cycles] が最後に増加した日時。
Number Lock Retries	このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=N（次に使用可能）を使用している場合に表示されます。『オペレーション』ドキュメントを参照してください。  このフィールドは、ロックされたバッファプールエントリが割り当て要求を満たせなかったために、それらのエントリのチェーンを解除する必要があった回数を示します。
Largest Alloc (TR)	テキストレコードの数で指定された、これまでに要求された最大の単一割り当てサイズ。
Number Load Failure	1つのオブジェクトのロードが失敗した合計回数。失敗の原因として、ロードの要求時にすべてのディレクトリエントリが使用されていたか、またはロードを実行するために十分なストレージがテキストレコードセクションで使用可能でないことを挙げることができます。
Number Load Failure - Sizes failing last	直近3回の失敗したストレージ割り当て要求によって求められたテキストレコードの数。

バッファプールでのスペースの割り当てに使用する検索メソッドの詳細については、『オペレーション』ドキュメントの「バッファプールの検索メソッド」を参照してください。

## バッファプールフラグメント

この機能では、バッファプールフラグメントの概要を示します。つまり、いくつの異なる Natural オブジェクトがどれだけの数のテキストレコードを占有しているかについて、およびオブジェクトの場所がバッファプールに分散する方法についての概要を示します。

### ▶手順 62.4. [Buffer Pool Fragmentation] を呼び出すには

- [Buffer Pool Statistics] メニューで、次のファンクションコードを入力します。

F

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY FRAGMENTATION
```

「**Buffer Pool Fragmentation**」画面が表示されます。

「**Buffer Pool Fragmentation**」画面に表示される一部のフィールドは、「**一般バッファプール統計**」で説明した項目と同じです。

**Buffer Pool Size**

**Buffer Pool Address**

**Text Record Section**

**Text Record Size**

**Number of Text Records** (「**Text Records - Total**」に対応)

また、画面には、いくつかの異なる個々のオブジェクトがどれだけのテキストレコードサイズを専有しているかを示す図が表示されます。

例:

```

          1----+-----10----+-----20----+-----30----+-----40----+-----50
005F0480  .._.+***_____++  ....**_.+**_.++  *..+***+__++++XX
    
```

図の各記号は1つのテキストレコードを表し、等号記号の各順序は、1つ以上のテキストレコードを専有する異なる個々のオブジェクトを表します。記号の意味は、次のとおりです。

_ および .	「 <b>Current Use Count</b> 」(「ディレクトリ情報」を参照)が0(ゼロ)のオブジェクト。
+ および *	「 <b>Current Use Count</b> 」が0(ゼロ)より大きいオブジェクト。
空白文字	未使用のテキストレコード。
XX	バッファプールの最後。使用できるテキストレコードがこれ以上ないことを意味します。

上の例では、バッファプールに48のテキストレコードが含まれています。それらのテキストレコードのうち3つは使用されていません。残りは24の異なるオブジェクトによって専有されています。そのうち12は「**Current Use Count**」が0(ゼロ)で、12は「**Current Use Count**」が0を超えています。

## 内部機能使用率

この機能では、Natural バッファプールマネージャに対するコールに関する統計情報が提供されています。

### ▶手順 62.5. [Internal Function Usage] を呼び出すには

- [Buffer Pool Statistics] メニューで、次のファンクションコードを入力します。

```
I
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY FUNCTION
```

[Internal Function Usage] 画面が表示されます。

[Internal Function Usage] 画面に表示される統計は、バッファプールのスナップショットであり、Enter キーを押すたびに更新されます。

[Total Calls] フィールドには、バッファプールマネージャのすべての内部コールの総数が示されます。

内部的に、バッファプールマネージャはさまざまな機能のために呼び出すことができます。機能ごとに、バッファプールマネージャが呼び出された回数が、絶対数と比率の両方で表示されます。さらに、これらの数が水平棒グラフで表されます。

## バッファプールのハッシュテーブル統計

この機能は、Natural タイプのバッファプールにのみ適用されます。

[Buffer Pool Hash Table Statistics] には、ハッシュテーブルのスロットおよびスロットごとの競合に関する統計が表示されます。統計を使用して、使用されているハッシュアルゴリズムの効率を調べます。

ハッシュテーブルの詳細については、『オペレーション』ドキュメントの「バッファプールのハッシュテーブル」を参照してください。

統計は、主として、Software AG の担当者によって内部でのみ使用されます。

▶手順 62.6. [Buffer Pool Hash Table Statistics] を呼び出すには

- [Buffer Pool Statistics] メニューで、次のファンクションコードを入力します。

H

または:

次の SYSBPM ダイレクトコマンドを入力します。

DISPLAY HASH

[Hash Table Collisions] 画面が表示されます。

[Hash Table Collisions] 画面に表示される統計は、ハッシュテーブルのスナップショットであり、Enter キーを押すたびに取得されます。次の情報が表示されます。

フィールド	説明
<b>Total Number of Slots</b>	ハッシュテーブルのスロットの合計数、つまり、オブジェクト名をオブジェクトの場所に関連付けるハッシュテーブル内の使用可能なエントリの合計。  スロットの数、つまりハッシュテーブルのサイズは、テキストレコードの数に基づいて内部で計算されます。
<b>Number of Slots used</b>	少なくとも1つのオブジェクト名がマップされているハッシュテーブル内のスロットの数。
<b>Number of Slots free</b>	オブジェクト名がマップされていないハッシュテーブル内のスロットの数。
<b>Max. Collisions per Slot</b>	スロットの競合の最大数。競合の最大数は、オブジェクトの最大可能検索パスです。  競合は、ハッシュアルゴリズムによって2つの異なるオブジェクトの名前が同じスロットにマップされた場合に発生します。その場合、別のスロットを見つけるために競合の解決が使用されます。
<b>Collisions</b>	現在の競合の数。発生した競合に応じて、次のようにテーブルには最大10行が含まれています。
0	競合なし。
1	1 競合。
2	2 競合。
3	3 競合。
4	4 競合。
5	5 競合。
6 - 10	6~10 競合。
11 - 15	11~15 競合。

フィールド	説明	
	16 - 20	16~20 競合。
	21	21 競合以上。
	競合がないとは、スロットごとに1つのオブジェクト名のみがマップされていることを意味します。このオブジェクトを検索するには、ハッシュテーブルに1回だけアクセスする必要があります。	
	競合の数が0（ゼロ）より大きい場合、例えば、 $x$ の場合、 $x+1$ オブジェクト名が同じスロットにマップされています。これらのオブジェクトの1つを検索するには、ハッシュテーブルに最大で $x+1$ . 回アクセスする必要があります。	
<b>Number of Slots</b>	競合の数に関連したスロットの数。 また、これらのスロットと使用されたすべてのスロットとの比率も表示されます。	
<b>Number of Slots Totalled</b>	<b>Number of Slots</b> と同じ値ですが、合計された値です。	

## ハッシュテーブル統計の例

```

14:36:26          ***** NATURAL SYSBPM UTILITY *****          2003-08-13
BPNAME NATGBP      - Buffer Pool Hash Table Statistics -          Type Global Nat
BPPROP OFF                                               Loc DAEF QA41

Total Number of Slots ..          523
Number of Slots used ..          475 ( 90.8 %)      Max. Collisions
Number of Slots free ..          48 ( 9.1 %)          per Slot ..... 7

Collisions          Number of Slots      Number of Slots Totalled
  0                   0 ( 0.0 %)          0 ( 0.0 %)
  1                  164 ( 34.5 %)          164 ( 34.5 %)
  2                  194 ( 40.8 %)          358 ( 75.3 %)
  3                   96 ( 20.2 %)          454 ( 95.5 %)
  4                   16 ( 3.3 %)          470 ( 98.9 %)
  5                    4 ( 0.8 %)          474 ( 99.7 %)
  6 - 10              1 ( 0.2 %)          475 ( 100.0 %)

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit  Last      Flip      Canc

```

## パフォーマンスに関するヒント

この機能は、Nat (Natural) タイプのバッファプールとバッファプールキャッシュにのみ適用されます。

[Performance Hints] 機能では、全体的なパフォーマンスの評価を含む、Natural バッファプールとバッファプールキャッシュの負荷に関する統計情報が提供されています。

関連トピック：

### ■ パフォーマンスの考慮事項

このセクションでは、次のトピックについて説明します。

- [Performance Hints] の呼び出し
- パフォーマンスに関するヒントの評価

### [Performance Hints] の呼び出し

このセクションでは、[Performance Hints] 機能および [Performance Hints] 画面に表示される統計フィールドを呼び出す方法について説明します。

#### ▶ 手順 62.7. [Performance Hints] を呼び出すには

- [Buffer Pool Statistics] メニューで、次のファンクションコードを入力します。

```
P
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY PERFORMANCE
```

次のような [Performance Hints] 画面が表示されます。

```
13:27:16          ***** NATURAL SYSBPM UTILITY *****          2005-05-19
BPNAME QA41GBP          - Performance Hints -          Type Global Nat
BPPROP OFF              Loc DAEF QA41
                               Preload QA41GBPL

                               Rating
                               (1=best - 6=worst)
```



```

Buffer Pool

Locates / Loads Ratio .....      162.85      3

Wrap Time Last (hh:mm:ss) ..      00:06:29      4

Wrap Time Avg (hh:mm:ss) ...      00:01:22      5

BP Cache

Object Reuse Factor .....          3.11         4

Wrap Time Last (hh:mm:ss) ..      02:02:29      1

Wrap Time Avg (hh:mm:ss) ...      00:32:06      3

Get / Search % .....              74.48 %

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10---PF11---PF12---
      Help       Exit  Last       Flip                               Canc

```

[Performance Hints] 画面のフィールドには、次の情報が表示されます。

フィールド	説明
Buffer Pool - Locates / Loads Ratio	<p>[Loads from DB] に対する [Total Locate Calls - successful] の比率。1 より大きい値は、Natural で、システムファイルからロードされたオブジェクトよりも多くのオブジェクトがバッファプールに保存されていることを示します。</p> <p>この比率は、バッファプールの効率の指標として使用できます。この数が多いほど、バッファプールのパフォーマンスは向上します。この比率は、1 つのバッファプールセッションから次のセッションへのパフォーマンスの主要な指標です。</p>

フィールド	説明
<b>Buffer Pool - Wrap Time Last</b>	<p>このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=N (次に使用可能) を使用している場合に表示されます。『オペレーション』ドキュメントを参照してください。</p> <p>バッファプールが最後に完全に再利用されてからの時、分、秒 (hh:mm:ss) 単位の時間 (ラップアラウンド)。</p> <p>オブジェクトはバッファプールに次々に (順番に) ロードされ、最初にバッファプールの先頭を埋め、最後に末尾を埋めます。バッファプールの末尾に到達すると、バッファの末尾はラップアラウンドされ、バッファプールの先頭で次のオブジェクトが再びロードされます。</p> <p>バッファプールが初めて完全にいっぱいになると、前のラップアラウンドサイクルでロードされ、現在ロックされていない (常駐または使用中としてマークされている) オブジェクトが、バッファにロードされた新しいオブジェクトで上書きされます。</p> <p>バッファプールにロードされたすべてのオブジェクトに、オブジェクトの名前、オブジェクトが保存されているライブラリ、バッファプールへのロードの <b>[BP Load Time]</b> タイムスタンプなどの情報を含むディレクトリエントリが割り当てられます。</p> <p><b>[Wrap Time Last]</b> は、オブジェクトがバッファプールにロードされるたびに評価されません。<b>[Wrap Time Last]</b> は、オブジェクトの最後のロード (<b>[BP Load Time]</b>) とオブジェクトの最後の上書きとの間の期間を示します。</p> <p>ラップアラウンドサイクルの期間が長くなるほど、バッファプールのパフォーマンスは向上します。この期間は、ユーザートラフィックが通常の営業時間より少ない夜間や週末には大幅に異なります。</p>
<b>Buffer Pool - Wrap Time Avg</b>	<p>このフィールドは、ストレージ割り当てのための検索メソッドとして METHOD=N (次に使用可能) を使用している場合に表示されます。『オペレーション』ドキュメントを参照してください。</p> <p>バッファプールが初期化または更新されてからの 1 回のラップアラウンドサイクルの時、分、秒 (hh:mm:ss) 単位の平均時間。</p> <p><b>[Wrap Time Avg]</b> は、バッファプールの存続時間をラップアラウンドサイクルの数で除算して計算されます。</p> <p><b>[Wrap Time Avg]</b> と比較した <b>[Wrap Time Last]</b> では、バッファプールが現在平均よりも頻繁に使用されているかどうかを示されます。</p>
<b>BP Cache - Object Reuse Factor</b>	<p>バッファプールキャッシュからスワップされたオブジェクト (<b>Get コール</b>) と、バッファプールキャッシュにスワップされたオブジェクト (<b>Put コール</b>) の比率。</p> <p>この値は、正常に終了した Get コールを正常に終了した Put コールで除算して計算されます。この値は、再利用ファクタ全体、つまり、バッファプールキャッシュに 1 回ロードされたオブジェクトをバッファプールに正常に再ロードできる回数を示します。値が大きいほど、バッファプールキャッシュの効率はよくなります。</p> <p><b>[Object Reuse Factor]</b> の例：                      5.70 の比率は、バッファプールキャッシュにロードされたオブジェクトがバッファプールに平均 5.7 回スワップされたことを示します。</p>

フィールド	説明
BP Cache - Wrap Time Last	<p>バッファプールキャッシュが最後に完全に再利用されてからの時、分、秒 (<i>hh:mm:ss</i>) 単位の時間 (ラップアラウンド)。</p> <p>オブジェクトはバッファプールキャッシュに次々に (順番に) ロードされ、最初にバッファプールキャッシュの先頭を埋め、最後に末尾を埋めます。バッファプールキャッシュの末尾に到達すると、バッファプールキャッシュの末尾はラップアラウンドされ、バッファプールキャッシュの先頭で次のオブジェクトが再びロードされます。バッファプールキャッシュが初めて完全にいっぱいになると、前のラップアラウンドサイクルでロードされたオブジェクトが、バッファにロードされた新しいオブジェクトで上書きされます。</p> <p>バッファプールキャッシュにロードされた各オブジェクトに、オブジェクトの名前、オブジェクトが保存されているライブラリ、バッファプールキャッシュへロードされたときの <b>[BPC Load Time]</b> タイムスタンプなどの情報を含むディレクトリエントリが割り当てられます。</p> <p><b>[Wrap Time Last]</b> は、オブジェクトがバッファプールキャッシュにロードされるたびに評価されます。 <b>[Wrap Time Last]</b> は、オブジェクトの最後のロード ( <b>[BPC Load Time]</b> ) とオブジェクトの最後の上書きとの間の期間を示します。</p> <p>ラップアラウンドサイクルの期間が長くなるほど、バッファプールキャッシュのパフォーマンスは向上します。この期間は、ユーザートラフィックが通常の営業時間より少ない夜間や週末には大幅に異なります。</p>
BP Cache - Wrap Time Avg	<p>バッファプールキャッシュが開始されてからの1回のラップアラウンドサイクルの時、分、秒 (<i>hh:mm:ss</i>) 単位の平均時間。</p> <p><b>[Wrap Time Avg]</b> は、バッファプールキャッシュの存続時間をラップアラウンドサイクルの数で除算して計算されます。</p> <p><b>[Wrap Time Avg]</b> と比較した <b>[Wrap Time Last]</b> では、バッファプールキャッシュが現在平均よりも頻繁に使用されているかどうかを示されます。</p>
BP Cache - Get / Search %	<p>バッファプールがバッファプールキャッシュに送信した検索要求 (Search コール) の合計数と比較した、バッファプールキャッシュから正常に返されたオブジェクト (Get コール) の比率</p> <p>この値は、バッファプールが、Natural システムファイルの FNAT または FUSER ではなく、バッファプールキャッシュからロードできたオブジェクトの比率を表します。値が大きいほど、キャッシュの効率はよくなります。</p> <p>steplibs のチェーン内のオブジェクトの検索により、バッファプールがバッファプールキャッシュに送信する Search コールの数が増加する場合があります。ただし、オブジェクトが steplib で見つからない場合があるため、これらのコールでは、正常な Get コールまたは Natural システムファイルからのロードは実行されません。 <b>[Get/Search %]</b> では、頻繁に使用される steplibs の長いチェーンの検索は考慮されません。全体的なパフォーマンスを向上させるため、使用する steplibs はできるだけ少なくすることをお勧めします。</p> <p><b>[Get / Search %]</b> 値の例： 70% の値は、バッファプールにロードされたすべてのオブジェクトの 70% がバッファプールキャッシュから取得され、30% がシステムファイルからロードされたことを示します。</p>

### パフォーマンスに関するヒントの評価

[Performance Hints] 画面に表示される統計値は、パフォーマンス評価システムのベースです。この評価システムでは、1は最良（最高）を表し、6は最悪（最低）を表します。

この評価により、バッファプールまたはバッファプールキャッシュが、指定したサイズでどのように動作するかを知ることができます。評価値は、ほとんどのシステム環境の要件に適合する必要があります。

バッファプールの5倍のバッファプールキャッシュを使用すると効果的であることが判明している環境があります。評価が悪くなる傾向がある場合は、バッファプールまたはバッファプールキャッシュのサイズを大きくする必要があります。ただし、評価が悪い場合でも、バッファプールまたはバッファプールキャッシュのサイズによって、優れたパフォーマンスを保証できます。バッファプールキャッシュの評価がよい場合には、バッファプールの評価はあまり重要ではありません。ただし、負荷が非常に大きい環境では、評価は、バッファプールまたはバッファプールキャッシュのサイズを変更するタイミングの有効な指標となることがあります。

[Performance Hints] 画面に表示される統計値は、Natural バッファプールおよびバッファプールキャッシュのスナップショットであり、ラップアラウンドが発生するたびに更新されます。統計値が意味のある結果を導き出すようになるには、バッファプールおよびバッファプールキャッシュをしばらくの間実行する必要があります。例えば、バッファプールのサイズが非常に大きいためにバッファプールキャッシュコールがほとんど必要ない場合には、バッファプールキャッシュに関する統計は重要ではありません。

統計を評価する場合、システム環境のタイプ（例えば、実稼働環境とテスト環境）、バッファプールを使用するアプリケーションのタイプ（バッチ、オンライン、ユーザー定義、またはシステム）、ユーザートラフィック（ピーク時または通常時）、および特異な操作要因も考慮する必要があります。

## PF キーとダイレクトコマンド

バッファプール統計の画面で、次の表に示す PF キーまたは SYSBPM ダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、「[SYSBPM ダイレクトコマンド](#)」を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ヘルプ情報を表示します。 <a href="#">オンラインヘルプ</a> も参照してください。
PF3	<u>EXIT</u>	現在の機能／画面を終了して、前の画面を表示します。
PF4	<u>LAST</u>	最後に入力した SYSBPM ダイレクトコマンドを表示します。
PF6	<u>FLIP</u>	PF キー行を切り替えます。PF1～PF12 キーと PF13～PF24 キーの間で表示を切り替えます。

PF キー	コマンド	機能
PF8 (Load)	<u>D</u> ISPLAY <u>L</u> OAD	[ <b>General Buffer Pool Statistics</b> ] 画面にのみ適用されます。 [ <b>Buffer Pool Load/Locate Statistics</b> ] 画面が表示されます。
PF8 (Gen)	<u>D</u> ISPLAY <u>G</u> ENERAL	[ <b>Buffer Pool Load/Locate Statistics</b> ] 画面にのみ適用されます。 [ <b>General Buffer Pool Statistics</b> ] 画面が表示されます。
PF12	<u>C</u> ANCEL	<b>EXIT</b> と同じです。
PF15	MENU	SYSBPM メインメニューを呼び出します。



## 63 バッファプールキャッシュ統計

---

■ 一般バッファプールキャッシュ統計 .....	362
■ バッファプールキャッシュコール統計 .....	364
■ バッファプールキャッシュのハッシュテーブル統計 .....	366
■ パフォーマンスに関するヒント .....	367
■ PF キーとダイレクトコマンド .....	367

[BP Cache] 機能は、Natural タイプのバッファプールにのみ適用されます。

この機能では、バッファプールキャッシュに関する統計情報を取得する場合に使用する [BP Cache Statistics] メニューを呼び出します。

[BP Cache] 機能は、グローバルバッファプールの初期化時にバッファプールキャッシュがインストールされている場合にのみ実行できることに注意してください（ローカルバッファプールに対するバッファプールキャッシュのサポートなし）。

### ▶手順 63.1. [BP Cache Statistics] を呼び出すには

- SYSBPM メインメニューで、次のファンクションコードを入力します。

```
C
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY CSTATISTICS
```

[BP Cache Statistics] メニューが表示されます。

このセクションでは、[BP Cache Statistics] メニューで使用可能な機能、およびそれらの機能により呼び出される画面に表示されるコマンドについて説明します。

## 一般バッファプールキャッシュ統計

---

この機能では、バッファプールキャッシュの動作に関するアドレスおよび統計を表示します。

### ▶手順 63.2. [General BP Cache Statistics] を呼び出すには

- [BP Cache Statistics] メニューで、次のファンクションコードを入力します。

```
G
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY CGENERAL
```

[General BP Cache Statistics] 画面が表示されます。



「**General BP Cache Statistics**」画面に表示される統計は、バッファプールのスナップショットであり、Enter キーを押すたびに更新されます。次の情報が表示されます。

フィールド	説明
<b>Dataspace - Name</b>	バッファプールキャッシュが存在するデータスペースの名前。
<b>Dataspace - SToken</b>	SToken (Space Token) という用語では、データスペースが識別されます。
<b>Dataspace - ALET</b>	ALET (Address List Entry Token) という用語では、データスペースにアクセスするためのインデックスが識別されます。
<b>Dataspace - Size (MB)</b>	バッファプールキャッシュのサイズ (MB 単位)。
<b>Dataspace - Current state</b>	次のバッファプールキャッシュのステータス。  not initialized locked for init closed free for operation undefined
<b>Dataspace - Initialization</b>	バッファプールキャッシュが初期化された日時。
<b>Internal buffer offsets - Header buffer</b>	一般バッファプールキャッシュ情報を含むバッファプールキャッシュのヘッダー。
<b>Internal buffer offsets - Hash buffer</b>	ハッシュテーブルが含まれています (「 <a href="#">バッファプールキャッシュのハッシュテーブル統計</a> 」も参照)。
<b>Internal buffer offsets - Directory buffer</b>	バッファプールキャッシュの先頭に関連する、バッファプールキャッシュディレクトリセクションのアドレス。  バッファプールキャッシュにロードされた各 Natural オブジェクトには、そのオブジェクトに関する情報を含むディレクトリエントリが必要です。これらのディレクトリエントリのスペースは、バッファプールキャッシュ自体から取得されます。
<b>Internal buffer offsets - Text buffer</b>	バッファプールキャッシュの先頭に関連する、テキストバッファのアドレス。  他のすべてのバッファのスペースを割り当てた後、残りのスペースは、4KBのサイズでテキストレコードに分割されます。  オブジェクトは、サイズに応じて1つまたは複数のテキストレコードを占有できます。
<b>Tot. Text Records</b>	バッファプールキャッシュ内のテキストレコードの総数。  テキストレコードの数は、バッファプールキャッシュのサイズによって異なります。バッファプールキャッシュのテキストレコードのサイズは、4KBです。
<b>Insert position</b>	次のオブジェクトが挿入されるテキストレコードのインデックス番号。  オブジェクトは、バッファプールから削除する必要がある場合にバッファプールキャッシュに挿入されます。

フィールド	説明
Reuse cycles	バッファプールキャッシュが完全に再利用された回数。  バッファプールキャッシュがいっぱいになるたびに、バッファプールキャッシュマネージャによってバッファプールキャッシュが先頭から再利用され、オブジェクトが上書きされます。オブジェクトは、バッファプールキャッシュが再び使用されるまでバッファプールキャッシュ内に維持されます。
Objects - Max Loaded	現在バッファプールキャッシュにロードされているオブジェクトの最大数。
Objects - Loaded	現在バッファプールキャッシュにロードされているオブジェクトの数。

## バッファプールキャッシュコール統計

この機能では、バッファプールキャッシュへのオブジェクトのロード (put)、バッファプールキャッシュからのオブジェクトの取得 (get) および削除に関する統計情報が提供されています。この情報は、バッファプールキャッシュのパフォーマンスの指標として使用することもできます。

### ▶手順 63.3. [BP Cache Call Statistics] を呼び出すには

- [BP Cache Statistics] メニューで、次のファンクションコードを入力します。

```
L
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY CLOAD
```

[BP Cache Call Statistics] 画面が表示されます。

[BP Cache Call Statistics] 画面に表示される統計は、バッファプールのスナップショットであり、Enter キーを押すたびに更新されます。次の情報が表示されます。

フィールド	説明
Search calls	バッファプールキャッシュでのオブジェクトの検索中に、バッファプールがバッファプールキャッシュに送信した Search コールの数。  オブジェクトが見つかった場合、Search コールによって Get コールが実行されます。
Get calls (from BP cache)	バッファプールキャッシュからバッファプールへのオブジェクトのロードを試行しているときに、バッファプールがバッファプールキャッシュに送信した Get コールの数。

フィールド	説明
Get calls - successful	<p>バッファプールキャッシュによる正常に終了したGetコールの数。つまり、バッファプールキャッシュがバッファプールにスワップしたオブジェクトの数。</p> <p>バッファプールがロードを試行したオブジェクトがバッファプールキャッシュからバッファプールに実際にロードされた場合、Get コールは正常に終了します。</p> <p>例えば、オブジェクトがSearch コールによって検出された後に削除された場合、Get コールは失敗します。</p>
Put calls (to BP cache)	<p>バッファプールからバッファプールキャッシュへのオブジェクトのスワップを試行しているときに、バッファプールがバッファプールキャッシュに送信したPutコールの数。</p>
Put calls - successful	<p>オブジェクトがバッファプールからバッファプールキャッシュにスワップされたPutコールの数。</p>
Put calls - obj. already cached	<p>すでにバッファプールキャッシュにロードされているオブジェクトについて、バッファプールがバッファプールキャッシュに送信したPut コールの数。</p>
Delete calls	<p>バッファプールキャッシュからのオブジェクトの削除を試行しているときに、バッファプールがバッファプールキャッシュに送信したDelete コールの数。</p> <p>Delete コールでは、単一のオブジェクトまたはオブジェクトの範囲が要求されます（「<a href="#">オブジェクトの削除</a>」セクションも参照）。</p>
Delete calls - successful	<p>バッファプールがバッファプールキャッシュに送信した、正常に終了したDeleteコールの数。</p> <p>少なくとも1つのオブジェクトがバッファプールキャッシュから実際に削除された場合、Delete コールは正常に終了します。</p> <p>要求されたオブジェクトがバッファプールキャッシュにロードされていないために削除できなかった場合、Delete コールは失敗します。</p> <p>Delete コールの総数と比較して、正常に終了したDelete コール数が非常に少なくなることがあります。このような状況は、例えば、複数のNatural オブジェクトがCATALLコマンドでカタログされた場合に発生します。その場合、正常にカタログされたすべてのオブジェクトについて、NaturalではバッファプールキャッシュにDelete コールが送信されます。ただし、通常、カタログ化オブジェクトのほとんどは、Delete コールの送信時にバッファプールキャッシュにロードされておらず、削除試行は失敗します。</p>
Initialization	<p>バッファプールキャッシュが初期化された日時。</p>
Last reuse cycle	<p>最後に上書きされたオブジェクトがロードされた日時。</p> <p>別のオブジェクトをロードするためにバッファプールキャッシュのスペースを再利用する必要がある場合、オブジェクトはバッファプールキャッシュ内で上書きされます。バッファプールキャッシュに最初にロードされたオブジェクトは最初にスワップされます。つまり、バッファプールキャッシュに最も長い間存在しているオブジェクトのロード日時は、<b>[Last reuse cycle]</b> の日時に対応します。</p>
Last access	<p>バッファプールがバッファプールキャッシュに最後にアクセスした日時。</p>

フィールド	説明
Last Put (to BP cache)	バッファプールがバッファプールキャッシュに最後に Put コールを送信した日時。
Last Get (from BP cache)	バッファプールがバッファプールキャッシュに最後に Get コールを送信した日時。
Last Delete	バッファプールがバッファプールキャッシュに最後に Delete コールを送信した日時。

## バッファプールキャッシュのハッシュテーブル統計

この機能では、ハッシュテーブルのスロットおよびスロットごとの競合に関する統計を表示します。統計を使用して、使用されているハッシュアルゴリズムの効率を調べます。

### ▶手順 63.4. [BP Cache Hash Table Statistics] を呼び出すには

- [BP Cache Statistics] メニューで、次のファンクションコードを入力します。

```
H
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY CHASH
```

[Cache Hash Table Collisions] 画面が表示されます。

[Cache Hash Table Collisions] 画面に表示される統計は、ハッシュテーブルのスナップショットであり、Enter キーを押すたびに更新されます。次の情報が表示されます。

フィールド	説明
Total Number of Slots	ハッシュテーブルのスロットの合計数、つまり、オブジェクト名をオブジェクトの場所に関連付ける使用可能なエントリの合計。  スロットの数、つまりハッシュテーブルのサイズは、テキストレコードの数に基づいて内部で計算されます。
Number of Slots used	1つ以上のエントリがあるスロットの数。
Number of Slots free	エントリがないスロットの数。
Max. Collisions per Slot	すべてのスロットの競合の最大数。  競合の最大数は、オブジェクトの最大可能検索パスです。
Collisions	可能性がある競合の数。  0 (ゼロ) は、競合がない、または1つのエントリを示します。5以上の競合がある場合、競合の数は範囲 (6 - 10 など) で指定されます。

フィールド	説明
Number of Slots	競合の数ごとにグループ化されたスロットの数。  例えば、競合の数が3の場合、オブジェクトを検出するために検索アルゴリズムが最大で3回避される必要があります。また、これらのスロットと使用されたすべてのスロットとの比率も表示されます。
Number of Slots Totalled	Number of Slots と同じ値ですが、合計された値です。

## パフォーマンスに関するヒント

「バッファプール統計」の「パフォーマンスに関するヒント」セクションを参照してください。

## PF キーとダイレクトコマンド

バッファプールキャッシュ統計の画面で、次の表に示すPF キーまたはSYSBPMダイレクトコマンドを使用できます。コマンドの下線部分は最短の省略形を表します。コマンドの詳細については、「[SYSBPMダイレクトコマンド](#)」を参照してください。

PF キー	コマンド	機能
PF1		SYSBPM ヘルプ情報を表示します。 <a href="#">オンラインヘルプ</a> も参照してください。
PF3	<u>EXIT</u>	現在の機能／画面を終了して、前の画面を表示します。
PF4	<u>LAST</u>	最後に入力したSYSBPMダイレクトコマンドを表示します。
PF6	<u>FLIP</u>	PF キー行を切り替えます。PF1～PF12 キーとPF13～PF24 キーの間で表示を切り替えます。
PF8 (CLoad)	<u>DISPLAY CLOAD</u>	[General BP Cache Statistics] 画面にのみ適用されます。 [BP Cache Call Statistics] 画面が表示されます。
PF8 (CGen)	<u>DISPLAY CGENERAL</u>	[BP Cache Call Statistics] 画面にのみ適用されます。 [General BP Cache Statistics] 画面が表示されます。
PF12	<u>CANCEL</u>	<a href="#">EXIT</a> と同じです。
PF15	<u>MENU</u>	SYSBPM メインメニューを呼び出します。



# 64 バッファプールの選択

---

- [Select Buffer Pool] の呼び出し ..... 370
- バッファプールの表示 ..... 371
- Reset Buffer Pool ..... 371

SYSBPMには、起動時に、Naturalシステム環境用に定義されたバッファプールを表示し、現在のNaturalセッションで使用しているバッファプール以外のバッファプールを選択する機能があります。

このセクションでは、**[Select Buffer Pool]** 機能、およびバッファプールの表示やリセットを実行するためのSYSBPMダイレクトコマンドについて説明します。

### **[Select Buffer Pool]** の呼び出し

#### ▶手順 64.1. **[Select Buffer Pool]** を呼び出すには

- 1 SYSBPM メインメニューで、次のファンクションコードを入力します。

```
S
```

または:

次のSYSBPMダイレクトコマンドを入力します。

```
SELECT BP
```

**[Select Buffer Pool]** ウィンドウが開き、現在のバッファプール（グローバルまたはローカル）、および現在Naturalシステム環境で使用可能なその他のすべてのグローバルバッファプールに関する情報が表示されます。

<b>BPNAME</b>	バッファプールの名前。
<b>タイプ</b>	Global Nat、Local Nat、Global Sort、Global DL/I、Edit（エディタ）、Mon（モニタ）などのバッファプールのタイプ。
<b>ステータス</b>	現在のステータス。
<b>Preload</b>	プリロードリストの名前（ロードされている場合）。
<b>Address</b>	バッファプールのアドレス。
<b>Loc</b>	<b>[HostID]</b> および <b>[SubsID]</b> （サブシステムID）によって示されたバッファプールの場所。

このフィールドの詳細については、「[SYSBPM メインメニュー - フィールド、機能、およびコマンド](#)」を参照してください。

- 2 **[C]** 列に任意の文字を入力して、Natural、DL/I、またはSortタイプ（他のすべてのバッファプールは表示のみ可能）のバッファプールを選択し、Enterキーを押します。指定したバッファプールは現在のセッションで定義され、それに応じて、SYSBPMメインメニューの上部に表示されるバッファプール関連のフィールド（上記参照）の内容が変わります。



「**Select Buffer Pool**」ウィンドウでバッファプールを選択すると、そのバッファプールにすべてのSYSBPM機能が適用されます。ただし、Naturalセッション自体は、起動時のバッファプールで実行されます。

## バッファプールの表示

▶手順 64.2. Natural システムで使用可能なバッファプールを表示するには

- 次のSYSBPMダイレクトコマンドを入力します。

```
DISPLAY BUFFERPOOL
```

「**Display Buffer Pools**」ウィンドウが開き、上記の「**Select Buffer Pool**」ウィンドウで説明したのと同じ情報が表示されます。

## Reset Buffer Pool

▶手順 64.3. バッファプールをリセットするには

- 次のSYSBPMダイレクトコマンドを入力します。

```
RESET BUFFERPOOL
```

SYSBPMでは、最初に現在のNaturalセッションに定義されたバッファプールに切り替わり、それに応じて、SYSBPMメインメニューの上部に表示されるバッファプール関連のフィールドの内容（「**Select Buffer Pool**」の呼び出し」を参照）が変わります。

または:

「**Select Buffer Pool**」機能（上記参照）を呼び出し、バッファプールの起動を選択します。



# 65      ブラックリストの管理

---

■ ブラックリストの管理 .....	374
■ オブジェクトセットのリスト .....	378
■ オブジェクトセットの編集 .....	379
■ ブラックリストへのオブジェクトセットの追加 .....	382
■ ブラックリストからのオブジェクトセットの削除 .....	382
■ オブジェクトセットソースオブジェクトの削除 .....	383
■ ユーティリティによる追加のオブジェクトセットの管理 .....	384
■ バッチモードでのブラックリストの管理 .....	384

## ブラックリストの管理

---

この機能は、Natural オブジェクトを管理する場合に使用します。ブラックリストでは、バッファプールにロードしない Natural オブジェクトを指定できます。その場合、すでにバッファプールにロードされているオブジェクトは削除されます。バッファプールキャッシュが有効になっている場合、その Natural オブジェクトはバッファプールキャッシュからも削除されます。ブラックリストは、常に現在有効なバッファプールに適用されます。

ブラックリストでは、個々の Natural オブジェクトまたはライブラリ（あるいはその両方）、および複数の Natural オブジェクトを含むオブジェクトセットを管理することができます。オブジェクトセットでは、実行されないオブジェクトを指定し、複数の個々のオブジェクトではなく単一のセットをブラックリストに追加することができます。また、オブジェクトの個別管理とセットごとの管理を組み合わせることもできます。

オブジェクトセットのソースコードは、Text タイプの Natural ソースオブジェクトとして、現在の Natural ライブラリとシステムファイルに格納されます。オブジェクトセットソースの最初の行には、コメント「\*\*BBL\*\*」（バッファプールブラックリスト）が含まれています。

ブラックリストの詳細については、『オペレーション』ドキュメントの「Natural バッファプール」の関連するセクションを参照してください。

### ▶手順 65.1. [Blacklist Maintenance] を呼び出すには

- SYSBPM メインメニューで、次のファンクションコードを入力します。

```
B
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
BLACKLIST
```

[Blacklist Maintenance] メニューが表示されます。

[Blacklist Maintenance] メニューで提供される機能については、次のセクションで説明します。このセクションでは、追加のブラックリストの管理機能についても説明します。

## ブラックリストの管理

---

この機能では、[Maintain Blacklist] 画面を呼び出します。この画面で、現在ブラックリストで使用可能なすべての Natural オブジェクトを表示し、管理することができます。

### ▶手順 65.2. [Maintain Blacklist] 画面を呼び出します。

- [Blacklist Maintenance] メニューで、次のファンクションコードを入力します。

M

または:

次の SYSBPM ダイレクトコマンドを入力します。

DISPLAY BLACKLIST

[**Maintain Blacklist**] 画面が開き、現在のブラックリストが表示されます。PF7 キーと PF8 キーを使用して、1 ページずつ上下にスクロールします。

SYSBPM セッション中に [**Maintain Blacklist**] 機能を以前呼び出したときのモード設定に応じて、[**Maintain Blacklist**] 画面が [**Display Mode**] (SYSBPM の初期化時デフォルト) または [**Add Mode**] で表示されます。1 つのモードから他のモードに切り替えるには、PF9 キーを使用します。

このセクションでは、次のトピックについて説明します。

- オブジェクトの追加
- オブジェクトの変更
- オブジェクトの削除

## オブジェクトの追加

### ▶手順 65.3. ブラックリストにオブジェクトを追加するには

- 1 [**Blacklist Maintenance**] メニューで、次のファンクションコードを入力します。

M

または:

次の SYSBPM ダイレクトコマンドを入力します。

DISPLAY BLACKLIST

[**Blacklist Maintenance**] 画面が表示されます。

- 2 必要に応じて、PF9 キーを押して [**Add Mode**] に切り替えます。

空の入力フィールドが表示された画面が開きます。

- 3 該当する入力フィールドに、オブジェクトが保存されているライブラリの名前、オブジェクトの名前、および対応するデータベース ID (DBID) とファイル番号 (FNR) を入力します。DBID と FNR を入力しなかった場合は、これらの ID と番号は、名前が SYS で始まるライブラリ (SYSTEM ライブラリを除く) 内の現在のシステムファイル FUSER または FNAT から取得されます。

[**Add Mode**] 画面をクリアする必要がある場合は、コマンド行に次のいずれかを入力します。

CLE

または

CLEAR

- 4 PF5 キーを押して追加を確認します。

または:

コマンド行に、次のいずれかを入力します。

UP

または

UPDATE

該当するメッセージが表示されます。

### オブジェクトの変更

#### ▶手順 65.4. ブラックリストのオブジェクトを変更するには

- 1 **[Blacklist Maintenance]** メニューで、次のファンクションコードを入力します。

M

または:

次の SYSBPM ダイレクトコマンドを入力します。

DISPLAY BLACKLIST

**[Blacklist Maintenance]** 画面が表示されます。

- 2 必要に応じて、PF9 キーを押して **[Display Mode]** に切り替え、現在ブラックリストに含まれているすべてのオブジェクトのリストを取得します。
- 3 該当する入力フィールドの既存のエントリを新しい値に置き換えます。
- 4 PF5 キーを押して変更を確認します。

または:

コマンド行に、次のいずれかを入力します。

UP

または

UPDATE

該当するメッセージが表示されます。

## オブジェクトの削除

### ▶手順 65.5. ブラックリストから個々のオブジェクトを削除するには

- 1 **[Blacklist Maintenance]** メニューで、次のファンクションコードを入力します。

```
M
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY BLACKLIST
```

**[Blacklist Maintenance]** 画面が表示されます。

- 2 必要に応じて、PF9 キーを押して **[Display Mode]** に切り替え、現在ブラックリストに含まれているすべてのオブジェクトのリストを取得します。
- 3 **[C]** 列の必須オブジェクトの横に、次の行コマンドを入力します。

```
DE
```

- 4 Enter キーを押して削除を確認します。

該当するメッセージが表示されます。

### ▶手順 65.6. ブラックリストからすべてのオブジェクトを削除するには

- 1 **[Blacklist Maintenance]** メニューで、次のファンクションコードを入力します。

```
M
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DISPLAY BLACKLIST
```

**[Blacklist Maintenance]** 画面が表示されます。

- 2 PF2 キーを押します。  
**[Confirm Delete]** ウィンドウが表示されます。
- 3 「Y」 (Yes) を入力して削除を確認します。

または:

「N」(No。デフォルト)を入力するか、ウィンドウに何も入力しないで PF3 キーを押して、削除をキャンセルします。

- 4 Enter キーを押して操作を確認します。

該当するメッセージが表示されます。

## オブジェクトセットのリスト

---

この機能では、既存のすべてのオブジェクトセットのリストが表示される **[List Object Sets]** 画面を呼び出します。

### ▶手順 65.7. **[List Object Sets]** 画面を呼び出すには

- **[Blacklist Maintenance]** メニューで、ファンクションコード「L」、ライブラリ名、およびオブジェクトセット名を入力します。オブジェクトセット名として、アスタリスク (\*) 表記も使用できます。

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
LIST SET library-name set-name
```

*set-name* として、アスタリスク (\*) 表記も使用できます。

**[List Object Sets]** 画面が開き、指定したセットが表示されます。

提供されているいずれかの行コマンドを使用してセットを変更し、ブラックリストに追加したり削除したりして、**[List Object Sets]** 画面でオブジェクトセットを操作できます。画面の左端の列(接頭辞情報)に疑問符(?)を入力すると、使用可能な行コマンドがリストされます。

コマンド行に疑問符(?)を入力して **[Help]** ウィンドウを呼び出すと、**[List Object Sets]** 画面のコマンド行に入力できるコマンドがリストされます。必要に応じて、PF7 キーと PF8 キーを使用してウィンドウ内を上下にスクロールします。



## オブジェクトセットの編集

この機能では、[Edit Object Set] 画面を呼び出します。この画面で、新しいオブジェクトセットの作成、既存のセットへのオブジェクトの追加、オブジェクトの変更、およびセットからのオブジェクトの削除を実行できます。

[Edit Object Set] 画面に表示される編集機能は、Software AG Editor（『エディタ』ドキュメントを参照）で利用できる機能のサブセットです。

画面の左端の列（接頭辞情報）に疑問符 (?) を入力すると、使用可能な行コマンドがリストされます。

コマンド行に疑問符 (?) を入力して [Help] ウィンドウを呼び出すと、[Edit Object Set] 画面のコマンド行に入力できるコマンドがリストされます。必要に応じて、PF7 キーと PF8 キーを使用してウィンドウ内を上下にスクロールします。

このセクションでは、次のトピックについて説明します。

- オブジェクトセットの作成
- オブジェクトセットの変更

### オブジェクトセットの作成

#### ▶手順 65.8. オブジェクトセットを作成するには

- 1 [Blacklist Maintenance] メニューに、ファンクションコード「E」とライブラリ名を入力します。オブジェクトセットの名前は入力しないで、対応するフィールド（ある場合）の内容をクリアします。

[Edit Object Set] 画面が表示されます。

- 2 該当する入力フィールドに、オブジェクトが保存されているライブラリの名前、オブジェクトの名前、および対応するデータベース ID (DBID) とファイル番号 (FNR) を入力します。DBID と FNR を入力しなかった場合は、これらの ID と番号は、名前が SYS で始まるライブラリ (SYSTEM ライブラリを除く) 内の現在のシステムファイル FUSER または FNAT から取得されます。
- 3 オブジェクトセットを Text タイプのソースオブジェクトとして格納するには、[Edit Object Set] 画面のコマンド行に、次のコマンドを入力します。

```
SA set-name
```

### オブジェクトセットの変更

このセクションでは、オブジェクトセットへのオブジェクトの追加、既存のオブジェクトの変更、およびオブジェクトセットからのオブジェクトの削除の手順について説明します。これらのオブジェクトセットのいずれかを変更しても、現在のブラックリストは更新されないことに注意してください。

#### ▶手順 65.9. オブジェクトセットに新しいオブジェクトを追加するには

- 1 **「Blacklist Maintenance」** メニューで、ファンクションコード「E」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

**「List Object Sets」** 画面で、左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT SET library-name set-name
```

**「Edit Object Set」** 画面が開き、指定したオブジェクトセットが表示されます。

- 2 該当する入力フィールドに、オブジェクトが保存されているライブラリの名前、オブジェクトの名前、および対応するデータベース ID (DBID) とファイル番号 (FNR) を入力します。DBID と FNR を入力しなかった場合は、これらの ID と番号は、名前が SYS で始まるライブラリ (SYSTEM ライブラリを除く) 内の現在のシステムファイル FUSER または FNAT から取得されます。
- 3 変更を保存するには、**「Edit Object Set」** 画面のコマンド行に次のコマンドを入力します。

```
SA
```

#### ▶手順 65.10. オブジェクトセットのオブジェクトを変更するには

- 1 **「Blacklist Maintenance」** メニューで、ファンクションコード「E」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

**「List Object Sets」** 画面で、左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT SET library-name set-name
```

〔**Edit Object Set**〕画面が開き、指定したオブジェクトセットが表示されます。

- 2 該当する入力フィールドの既存のエントリを新しい値に置き換えます。
- 3 変更を保存するには、〔**Edit Object Set**〕画面のコマンド行に次のコマンドを入力します。

```
SA
```

#### ▶手順 65.11. オブジェクトセットからオブジェクトを削除するには

- 1 〔**Blacklist Maintenance**〕メニューで、ファンクションコード「E」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

〔**List Object Sets**〕画面で、左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT SET library-name set-name
```

〔**Edit Object Set**〕画面が開き、指定したオブジェクトセットが表示されます。

- 2 左端の列の必須オブジェクトの横に、次の行コマンドを入力します。

```
D
```

指定したオブジェクトがオブジェクトセットから削除されます。

- 3 変更を保存するには、〔**Edit Object Set**〕画面のコマンド行に次のコマンドを入力します。

```
SA
```

## ブラックリストへのオブジェクトセットの追加

---

この機能は、オブジェクトセットのすべてのオブジェクトをブラックリストに追加する場合に使用します。

### ▶手順 65.12. ブラックリストにオブジェクトセットを追加するには

- **[Blacklist Maintenance]** メニューで、ファンクションコード「A」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

**[List Object Sets]** 画面で、左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

```
AC
```

または:

**[Edit Object Set]** 画面で、コマンド行に次のコマンドを入力します。

```
AC
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
ADD SET library-name set-name
```

オブジェクトセットがブラックリストに追加されたことを確認するメッセージが表示されません。

 **注意:** コマンド AC は、**[Add Object Set to Blacklist]** と同等の ACTIVATE を表します。

## ブラックリストからのオブジェクトセットの削除

---

この機能は、オブジェクトセットのすべてのオブジェクトをブラックリストから削除する場合に使用します。**[Delete Object Set]** 機能では、オブジェクトセットはソースオブジェクトとして削除されないことに注意してください。オブジェクトセットのオブジェクトは、上記のように、いつでも再びブラックリストに追加することができます。「[オブジェクトセットソースオブジェクトの削除](#)」も参照してください。

▶手順 65.13. ブラックリストからオブジェクトセットを削除するには

- **「Blacklist Maintenance」** メニューで、ファンクションコード「D」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

**「List Object Sets」** 画面で、左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

```
DA
```

または:

**「Edit Object Set」** 画面で、コマンド行に次のコマンドを入力します。

```
DA
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
DELETE SET library-name set-name
```

オブジェクトセットがブラックリストから削除されたことを確認するメッセージが表示されます。



**注意:** コマンド DA は、**「Delete Object Set from Blacklist」** と同等の DEACTIVATE を表します。

## オブジェクトセットソースオブジェクトの削除

▶手順 65.14. オブジェクトセットのソースオブジェクトを削除するには

- 1 **「Blacklist Maintenance」** メニューで、ファンクションコード「L」、ライブラリ名、およびオブジェクトセット名を入力します。

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
LIST SET library-name list-name
```

**「List Object Sets」** 画面が表示されます。

- 2 左端の列の必須オブジェクトセットの横に、次の行コマンドを入力します。

D

[DELETE] ウィンドウが表示されます。

- 3 オブジェクトセットの名前を入力して、削除されたことを確認します。

確認メッセージが表示されます。



**注意:** オブジェクトセットのソースオブジェクトを削除しても、現在のブラックリストは更新されません。

## ユーティリティによる追加のオブジェクトセットの管理

---

Natural ユーティリティの SYSMAIN と Object Handler には、オブジェクトセットを管理するための追加機能があります。機能には、異なる Natural ライブラリとシステムファイルやさまざまなメインフレームプラットフォームとの間のオブジェクトセットの転送、および異なる環境のオブジェクトセットの削除または検索が含まれています。

Natural ユーティリティを使用する場合、オブジェクトセットは Text タイプのその他のソースオブジェクトと同じように処理されます。

詳細については、『ユーティリティ』ドキュメントの関連するセクションを参照してください。

## バッチモードでのブラックリストの管理

---

「[バッチ処理](#)」で説明しているように、SYSBPM のブラックリスト管理機能は、バッチモードで実行することもできます。

また、Natural システムライブラリ SYSBPM では、ブラックリスト管理機能（すべてのオブジェクトの追加および削除）のバッチ処理をさらに容易にする BPMBLBAT コマンドを使用できます。

ライブラリ SYSBPM にログオンし、次の図に示す構文を使用して BPMBLBAT コマンドを実行します。図で使用している記号については、『システムコマンド』ドキュメントの「システムコマンド構文」セクションを参照してください。

```
BPMBLBAT { FUNC=LOCK, BPNAME=bp-name, (object-specification)  
          { FUNC=RLS, BPNAME=bp-name } }
```

このセクションでは、次のトピックについて説明します。

- [構文の説明](#)

■ 入力の例

構文の説明

上記の BPMBLBAT 構文図に示したキーワードと *object-specification* 句の詳細については、次のセクションを参照してください。 [入力の例](#)では、キーワードの使用方法を示します。

構文項目	説明	
FUNC=LOCK	ブラックリストにオブジェクト名を追加します。詳細については、「ブラックリストの管理」セクションの「 <a href="#">オブジェクトの追加</a> 」を参照してください。	
FUNC=RLS	リストに含まれたオブジェクト名とともにブラックリスト全体を削除します。この機能は、「ブラックリストの管理」セクションの「 <a href="#">ブラックリストからすべてのオブジェクトを削除するには</a> 」で説明している削除機能に対応します（バッチモードでは、ブラックリストから単一のオブジェクト名を削除することはできません）。	
BPNAME	ブラックリストがロードされたバッファプールの名前。	
<i>object-specification</i>	<i>object-specification</i> 句に適用されるキーワード（次の各構文を参照）は次のとおりです。	
	LIB	オブジェクトが格納されるライブラリの名前。
	DBID FNR	オブジェクトの格納先のデータベースID（DBID）とファイル番号（FNR）。DBIDとFNRを入力しなかった場合は、これらのIDと番号は、名前がSYSで始まるライブラリ（SYSTEMライブラリを除く）内の現在のシステムファイルFUSERまたはFNATから取得されます。
	<i>object-name</i>	ブラックリストに追加するオブジェクトの名前。 1~8の位置の個別の行にそれぞれの名前を入力します。 入力の最後を示すには、個別の行にピリオド（.）を入力します。

*object-specification* の構文

*object-specification* に適用される構文を以下に示して説明します。

```
LIB=lib-name, DBIDdbid, FNRfnr [object-name]...
```

### 入力の例

#### 例 1 - ブラックリストへのオブジェクトの追加

次の入力例は、バッファプール NATGBP 内のオブジェクト A、B、および C をブラックリストに追加する方法を示しています。

```
/*Job
.
.
.
*/Job

LOGON SYSBPM
BPMBLBAT
FUNC=LOCK,BPNAME=NATGBP,LIB=SAGTEST,DBID=10,FNR=32
A
B
C
.
FIN
```

#### 例 2 - ブラックリストの削除

次の入力例は、バッファプール NATGBP 内のブラックリストを削除する方法を示しています。

```
/*Job
.
.
.
*/Job
LOGON SYSBPM
BPMBLBAT
FUNC=RLS,BPNAME=NATGBP
.
FIN
```



## 66 プリロードリストの管理

---

▪ List Preload Lists .....	388
▪ プリロードリストの編集 .....	389
▪ バッファプールからのプリロードリストの生成 .....	392
▪ プリロードリストの削除 .....	394
▪ ユーティリティによる追加の管理機能 .....	394

この機能は、Natural タイプのバッファプールにのみ適用されます。

【Preload List Maintenance】は、プリロードリストを管理する場合に使用します。プリロードリストでは、バッファプールの初期化時にバッファプールにロードされる Natural オブジェクトの名前を指定できます。

プリロードリストのソースコードは、Natural システムライブラリ SYSBPM に Text タイプの Natural ソースオブジェクトとして格納されます。プリロードリストソースの最初の行には、コメント「\*\*BPL\*\*」（バッファプールプリロードリスト）が含まれています。

プリロードリストの詳細については、『オペレーション』ドキュメントの「Natural バッファプール」の関連のセクションを参照してください。

### ▶手順 66.1. 【Preload List Maintenance】を呼び出すには

- SYSBPM メインメニューで、次のファンクションコードを入力します。

```
P
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
PRELOADLIST
```

【Preload List Maintenance】メニューが表示されます。

【Preload List Maintenance】メニューで提供される機能については、次のセクションで説明します。このセクションでは、プリロードリストの追加管理機能についても説明します。

## List Preload Lists

---

この機能では、既存のすべてのプリロードリストのリストが表示される【List Preload Lists】画面を呼び出します。

### ▶手順 66.2. 【List Preload Lists】を呼び出すには

- 【Preload List Maintenance】メニューで、ファンクションコード「L」とプリロードリスト名を入力します。プリロードリスト名として、アスタリスク（\*）表記も使用できます。

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
LIST PRELOADLIST list-name
```

*list-name* として、アスタリスク (\*) 表記も使用できます。

[List Preload Lists] 画面が表示されます。

画面の左端の列（接頭辞情報）に疑問符 (?) を入力すると、使用可能な行コマンドがリストされます。

コマンド行に疑問符 (?) を入力して [Help] ウィンドウを呼び出すと、[List Preload Lists] 画面のコマンド行に入力できるコマンドがリストされます。必要に応じて、PF7 キーと PF8 キーを使用してウィンドウ内を上下にスクロールします。

## プリロードリストの編集

この機能では、プリロードリストの新規作成、既存のリストへのオブジェクトの追加または削除ができる [Edit Preload List] 画面を呼び出します。



**重要:** [Edit Preload List] 画面に表示される編集機能は、Software AG Editor（『エディタ』ドキュメントを参照）で使用できる機能のサブセットです。そのため、プリロードリストを編集する Natural セッションを開始する前に、Natural プロファイルパラメータ EDPSIZE を 0（ゼロ）より大きい値に設定します。『パラメータリファレンス』ドキュメントの「プロファイルパラメータ」も参照してください。EDPSIZE は少なくとも 100 に設定することをお勧めします。

画面の左端の列（接頭辞情報）に疑問符 (?) を入力すると、使用可能な行コマンドがリストされます。

コマンド行に疑問符 (?) を入力して [Help] ウィンドウを呼び出すと、[Edit Preload List] 画面のコマンド行に入力できるコマンドがリストされます。必要に応じて、PF7 キーと PF8 キーを使用してウィンドウ内を上下にスクロールします。

このセクションでは、次のトピックについて説明します。

### ■ プリロードリストの作成

### ■ プリロードリストの変更

## プリロードリストの作成

### ▶手順 66.3. プリロードリストを作成するには

- 1 **[Preload List Maintenance]** メニューで、次のように指定します。
  - ファンクションコード「E」を入力します。
  - **[Preload List Name]** フィールドの内容をクリアします。つまり、プリロードリストの名前を入力しないでください。
  - **[Library]** および **[Objects]** フィールドは、デフォルトのアスタリスク (\*) のままにします。

**[Edit Preload List]** 画面が表示されます。

- 2 該当する入力フィールドに、オブジェクトが保存されているライブラリの名前、オブジェクトの名前、および対応するデータベース ID (DBID) とファイル番号 (FNR) を入力します。

DBID と FNR を入力しなかった場合は、これらの ID と番号は、名前が SYS で始まるライブラリ (SYSTEM ライブラリを除く) 内の現在のシステムファイル FUSER または FNAT から取得されます。

値を入力しない場合、編集画面の **[R]** 列で常駐フラグが「Y」 (Yes) に設定されます。常駐とは、**[Use Count]** が 0 (ゼロ) に変更された場合にも、オブジェクトがバッファプールから削除されないことを意味します (**[Use Count]** は、「ディレクトリ情報」で説明している **[Current Use Count]** に対応します)。

- 3 プリロードリストを Text タイプのソースオブジェクトとしてライブラリ SYSBPM に格納するには、**[Edit Preload List]** 画面のコマンド行に次のコマンドを入力します。

```
SA set-name
```

「[バッファプールからのプリロードリストの生成](#)」も参照してください。

## プリロードリストの変更

### ▶手順 66.4. プリロードリストに新しいオブジェクトを追加するには

- 1 **[Preload List Maintenance]** メニューで、ファンクションコード「E」とプリロードリスト名を入力します。

または:

〔List Preload Lists〕画面で、左端の列の必須プリロードリストの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT PRELOADLIST list-name
```

〔Edit Preload List〕画面が開き、指定したプリロードリストが表示されます。

- 2 該当する入力フィールドに、オブジェクトが保存されているライブラリの名前、オブジェクトの名前、および対応するデータベース ID (DBID) とファイル番号 (FNR) を入力します。

DBID と FNR を入力しなかった場合は、これらの ID と番号は、名前が SYS で始まるライブラリ (SYSTEM ライブラリを除く) 内の現在のシステムファイル FUSER または FNAT から取得されます。

- 3 変更を保存するには、〔Edit Preload List〕画面のコマンド行に次のコマンドを入力します。

```
SA
```

#### ▶手順 66.5. プリロードリストのオブジェクトを変更するには

- 1 〔Preload List Maintenance〕メニューで、ファンクションコード「E」、ライブラリ名、およびプリロードリスト名を入力します。

または:

〔List Preload Lists〕画面で、左端の列の必須プリロードリストの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT PRELOADLIST list-name
```

〔Edit Preload List〕画面が開き、指定したプリロードリストが表示されます。

- 2 該当する入力フィールドの既存のエントリを新しい値に置き換えます。
- 3 変更を保存するには、〔Edit Preload List〕画面のコマンド行に次のコマンドを入力します。

```
SA
```

### ▶手順 66.6. プリロードリストからオブジェクトを削除するには

- 1 **「Preload List Maintenance」** メニューで、ファンクションコード「E」、ライブラリ名、およびプリロードリスト名を入力します。

または:

**「List Preload Lists」** 画面で、左端の列の必須プリロードリストの横に、次の行コマンドを入力します。

```
E
```

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
EDIT PRELOADLIST list-name
```

**「Edit Preload List」** 画面が開き、指定したプリロードリストが表示されます。

- 2 左端の列の必須オブジェクトの横に、次の行コマンドを入力します。

```
D
```

指定したオブジェクトがプリロードリストから削除されます。

- 3 変更を保存するには、**「Edit Preload List」** 画面のコマンド行に次のコマンドを入力します。

```
SA
```

## バッファプールからのプリロードリストの生成

---

この機能は、現在バッファプールにロードされているオブジェクトの名前を使用して新しいプリロードリストを生成する場合に使用します。現在バッファプール内にあるオブジェクトから、プリロードリストに含めるオブジェクトを選択することができます。

### ▶手順 66.7. プリロードリストを生成するには、次のいずれかのコマンドを使用します。

- 1 **「Preload List Maintenance」** メニューで、ファンクションコード「G」とプリロードリスト名を入力します。**「Library」**、**「Objects」**、**「Resident」**、**「Use Count」**、および**「Total Use Count」** フィールドで、リストに含めるオブジェクトを指定します。

- 現在バッファプール内にあるすべてのオブジェクトを含めるには、**[Library]**、**[Objects]**、および **[Resident]** フィールドにアスタリスク (\*) を入力し、**[Use Count]** および **[Total Use Count]** フィールドを空白のままにします。
- または、  
バッファプール内の指定したオブジェクトを含めるには、次に示すフィールドに次の値を入力します。

<b>Library</b>	単一名またはアスタリスク (*) 表記。
<b>Objects</b>	単一名またはアスタリスク (*) 表記。
<b>Resident</b>	すべてのオブジェクトの場合はアスタリスク (*)。または現在バッファプール内に常駐としてマークされているオブジェクトの場合は「Y」 (Yes)。
使用回数	>10のように、 <i>value</i> (>) で始まる数字。 <b>[Use Count]</b> が <i>value</i> 以上のすべてのオブジェクトを選択します。  このフィールドは、「ディレクトリ情報」で説明している <b>[Current Use Count]</b> に対応します。
<b>Total Use Count</b>	>10のように、 <i>value</i> (>) で始まる数字。 <b>[Total Use Count]</b> が <i>value</i> 以上のすべてのオブジェクトを選択します。  このフィールドは、「ディレクトリ情報」で説明している <b>[BP Total Use]</b> に対応します。

- 2 次のいずれかの SYSBPM ダイレクトコマンドを入力します。

```
GENERATE PRELOADLIST list-name
```

または

```
GENERATE PRELOADLIST list-name gen-library
```

(上記のフィールド値の説明も参照してください)

バッファプールからプリロードリストが生成されたことを確認するメッセージが表示されません。

すべてのプリロードリストオブジェクトは、デフォルトで常駐（**[R]** 列のエントリ Y）として生成されます。リストから削除するオブジェクトを手動で選択します。

ライブラリ SYSBPM のオブジェクトは、生成されたプリロードリストには含まれません。それらのオブジェクトは、この機能を実行するためにのみバッファプールにロードされたオブジェクトであるとみなされるためです。

## プリロードリストの削除

---

### ▶手順 66.8. プリロードリストを削除するには

- 1 **[Preload List Maintenance]** メニューで、ファンクションコード「L」とプリロードリスト名を入力します。

または:

次の SYSBPM ダイレクトコマンドを入力します。

```
LIST PRELOADLIST list-name
```

**[List Preload Lists]** 画面が表示されます。

- 2 左端の列の必須オブジェクトの横に、次の行コマンドを入力します。

```
D
```

**[DELETE]** ウィンドウが表示されます。

- 3 プリロードリストの名前を入力して、削除されたことを確認します。

確認メッセージが表示されます。

## ユーティリティによる追加の管理機能

---

Natural ユーティリティの SYSMAIN と Object Handler には、プリロードリストを管理するための追加機能があります。機能には、異なる Natural ライブラリとシステムファイルやさまざまなメインフレームプラットフォームとの間のプリロードリストの転送、および異なる環境のプリロードリストの削除または検索が含まれています。

Natural ユーティリティを使用する場合、プリロードリストは Text タイプのその他のソースオブジェクトと同じように処理されます。

詳細については、『ユーティリティ』ドキュメントの関連するセクションを参照してください。



## 67 パフォーマンスの考慮事項

---

▪ 内部高速検索テーブル .....	396
▪ Steplibs での検索 .....	397
▪ オブジェクトの再利用と保持 .....	398
▪ ローカルバッファプールとグローバルバッファプールの比較 .....	398
▪ 検索順序のモニタリング .....	399

このセクションでは、バッファプールとバッファプールキャッシュのパフォーマンス関連の問題についての一般的なアドバイスを提供します。

このセクションで説明する統計項目の詳細については、「[バッファプールのロード/ロケート統計](#)」を参照してください。

関連トピック：

### ■ [パフォーマンスに関するヒント](#)

## 内部高速検索テーブル

---

Natural セッション内で Natural オブジェクトが初めて参照されると、バッファプールマネージャによってそのオブジェクトのディレクトリエントリが作成されます。ディレクトリエントリは、オブジェクトを識別するために使用され、オブジェクトの名前、オブジェクトが存在するライブラリ（名前、データベース ID、ファイル番号）、バッファプール内のオブジェクトのアドレス（位置）などの情報が含まれています。

Natural ランタイムシステムでは、最近実行されたオブジェクト、それらのオブジェクトが存在するライブラリ（名前、データベース ID、ファイル番号）、および対応するバッファプールディレクトリエントリのアドレスが、Natural セッションの間、内部高速検索テーブルに記憶されます。

ユーザーが以前 Natural セッションで使用されたオブジェクトを呼び出すと、Natural ランタイムシステムによって、内部高速検索テーブルの情報がバッファプールマネージャに渡されます。これにより、バッファプールマネージャでは、通常検索手順ではなく、時間のかからない高速検索コールを実行できます（「[高速検索コール](#)」を参照）。これは、オブジェクトを検索するための最も効果的な方法です。

バッファプール内のオブジェクトの位置が変更された場合、バッファプールマネージャでは、自動的に通常検索コールがスケジュールされます。バッファプールから削除されたオブジェクトや別のオブジェクトによって上書きされたオブジェクトがバッファプールに再ロードされた場合、位置は通常変更されます。

内部高速検索テーブルには、最大128のエントリが含まれています。高速検索テーブルは LOGON システムコマンドによってリセットし、通常検索コールを使用して再び埋める必要があります。そのため、[LOGON] を実行するアプリケーションではパフォーマンスが失われます。

[[Quick Locate Calls](#)] に対する [[Normal Locate Calls](#)] の比率の高さは、各 LOGON での Natural アプリケーションの LOGON コマンドの使用によって内部高速検索テーブルがリセットされたことを示します。

## Steplib での検索

steplib ライブラリの長いチェーンを介して Natural オブジェクトを検索すると、パフォーマンスに悪影響を与えます。

Natural ランタイムでは、要求されたオブジェクトが見つかるまで、各 steplib に対してバッファプールマネージャへのコールが発行されます。ただし、通常、要求されたオブジェクトを含まない steplib ライブラリへの不要なコールは回避されます。Natural ユーティリティ DBLOG を使用して、バッファプールでの検索順序をモニタリングできます。詳細については、「[検索順序のモニタリング](#)」を参照してください。

BPSFI プロファイルパラメータ（『パラメータリファレンス』ドキュメントを参照）の設定に応じて、追加のデータベースコールが必要となる場合があります。

steplib チェーンの長さは、**[Normal Locate Calls]**（steplib 検索を含まず）に対する **[Normal Locate Calls]**（steplib 検索を含む）の比率の高さによって示され、次のように計算されます。

**Normal Locate Calls** : ( **[Normal Locate Calls]** - **[STEPLIB Searches]** )

### Steplib 検索の例

デフォルトの steplib チェーン（FUSER のライブラリ SYSTEM、FNAT のライブラリ SYSTEM）を検索する場合、SYSTEM（FNAT）からオブジェクトをロードしようとするたびに、次のような結果になります。

3 **[Normal Locate Calls]** と 2 **[STEPLIB Searches]**

説明：

3 件の通常検索コールは、現在のライブラリ、ライブラリ SYSTEM（FUSER）、およびライブラリ SYSTEM（FNAT）の検索が原因で発生します。オブジェクトは現在のライブラリにもライブラリ SYSTEM（FUSER）にも格納されないため、（少なくとも）2 件の通常検索コールは失敗します。上記の式を使用すると、比率は 3 対 1 になります。

オブジェクトが現在のライブラリにある場合、結果は次のようになります。

1 通常検索コールと 0（ゼロ） **[STEPLIB Searches]**。上記の式を使用すると、比率は 1 対 1 になります。

## オブジェクトの再利用と保持

---

多くのNaturalオブジェクトを含み、各オブジェクトがこれまでほとんど実行されていないアプリケーションは、バッファプールのパフォーマンスに大きな影響を与えます。いずれのオブジェクトもバッファプールでの存続期間が短く、多くのオブジェクトをシステムファイルからロードする必要があります。パフォーマンス上の理由から、アプリケーションでは、複数のオブジェクトに含まれている同一のソースコードを1つのオブジェクトに移動するなどして、できる限りオブジェクトを再利用する必要があります。

[List Objects] 機能を使用して、オブジェクトの使用を確認することができます（「オブジェクトのリスト」セクションを参照）。例えば、[Max] 列には、オブジェクトを実行したアプリケーションの最大数に関する情報が表示され、[TotalUC] 列には、バッファプールにロードされたオブジェクトの検索コールの合計数が表示されます。

オブジェクトは、[List Objects] 機能を使用するか、プリロードリストで指定して、個別に常駐にすることができます（「プリロードリストの管理」を参照）。

## ローカルバッファプールとグローバルバッファプールの比較

---

アプリケーションごとに要件が異なるため、ローカルバッファプールとグローバルバッファプールの使い分けに関する一般的な推奨事項はありません。ただし、経験則から、次のトピックについて一般的なアドバイスを提供することはできます。

このセクションでは、次のトピックについて説明します。

- ローカルバッファプールの使用
- グローバルバッファプールの使用

### ローカルバッファプールの使用

異なるアプリケーション環境にローカルバッファプールを割り当てることができる場合は、単一の大きいバッファプールではなく、複数の小さいローカルバッファプールを使用して、パフォーマンスを向上させることができます。

例えば、CICS 環境で各 AOR (Application Operating Region) にローカルバッファプールを使用すると、通常はパフォーマンスが向上します。

## グローバルバッファプールの使用

同じ Natural オブジェクトを参照する異なるバッチアプリケーションでは、個々のアプリケーションにローカルバッファプールを使用するのではなく、1つの共通のグローバルバッファプールを使用すると、パフォーマンスが向上する場合があります。その場合、各アプリケーションによって要求されるオブジェクトが、他のいずれかのアプリケーションによってグローバルバッファプールにすでにロードされている可能性が高くなります。

## 検索順序のモニタリング

このセクションでは、Natural ユーティリティの DBLOG を使用してバッファプールでの検索順序をモニタリングする方法について説明します。

### ▶手順 67.1. バッファプールでの検索順序をモニタリングするには

- 1 次のパラメータを設定して、Natural セッションを開始します。

```
LOG=ON DSIZE=(2 - 256) BPSFI=OFF
```

データベースロギングに十分なメモリを使用できる値に DSIZE を設定します。

- 2 次のシステムコマンドを入力します。

```
TEST DBLOG MENU
```

**DBLOG** メニューが表示されます。

- 3 **[VB]** の横にあるフィールドにマークを付けてバリュバッチオプションを選択し、ファンクションコード「B」を入力して DBLOG を有効にします。
- 4 プログラム例を実行します（次の例の PGM01）。
- 5 次のコマンドを再度入力します。

```
TEST DBLOG
```

次のような **[DBLOG Trace]** 画面が表示されます。

```
16:00:30          ***** NATURAL TEST UTILITIES *****          2005-05-17
User SAG                - DBLOG Trace -                          Library TEST

M  No Cmd  DB   FNR  Rsp      ISN      ISQ  CID  CID(Hex)  OP  Pgm   Line
-   1 S1   10  1640
-   2 RC   10
-   3 S1   10   32      00000000  ATEST  5400
-   4 S1   10   32      00000000  F  ATEST  7110
-   5 S1   10   32      00000000      0000
```



- 8 バリュースタックでは、オブジェクトが検索されたライブラリの名前（ここでは TEST）とオブジェクトの名前（ここでは PGM01）が表示されます。

スタックは、選択した追加の各 S1 コマンドについて、同じオブジェクト名であるが異なるライブラリ名を表示します。このことは、オブジェクトが現在のライブラリに見つからなかったこと、およびオブジェクトを見つけるために `steplibs` をスキャンする必要があったことを示します。





# 68

## SYSBPM ダイレクトコマンド

---

▪ SORT .....	409
▪ SORT BPC .....	409

このセクションで説明する SYSBPM ダイレクトコマンドを使用して、SYSBPM ユーティリティの機能を直接実行したり、SYSBPM 画面にオンラインまたはバッチモードで移動したりすることができます。特定の画面にのみ適用される追加の SYSBPM ダイレクトコマンドの詳細については、個々の SYSBPM 機能について説明しているセクションを参照してください。

バッファプールキャッシュまたはバッファプールハッシュテーブルを参照する SYSBPM ダイレクトコマンドは、Natural タイプのバッファプールにのみ適用されます。

次の表に、使用できるすべての SYSBPM ダイレクトコマンド（サブコマンドを含む）、コマンドとともに使用できるパラメータ、および同等の SYSBPM メニュー機能を示します。

次に示す SYSBPM ダイレクトコマンドは、任意の SYSBPM 画面のコマンド行に入力できます。他のいずれかのコマンドプロンプトから、またはバッチモードで SYSBPM ダイレクトコマンドを実行するには、ダイレクトコマンドの先頭に SYSBPM などのキーワードを付ける必要があります。

SYSBPM ADD BLACKLIST

SYSBPM コマンドの下線部分は最短の省略形を表します。コマンドで必要なパラメータ値は、斜体文字で表されています。

コマンド	パラメータ	機能
<u>+</u>	なし	リストを1ページ下方にスクロールします。
<u>-</u>	なし	リストを1ページ上方にスクロールします。
<u>ADD</u> <u>BLACKLIST</u>	なし	<b>[Maintain Blacklist]</b> 画面を呼び出します（「ブラックリストの管理」を参照）。
<u>ADD</u> <u>SET</u>	<i>library-name</i> <i>set-name</i>	指定したオブジェクトセットのすべてのオブジェクトをブラックリストに追加します（「ブラックリストへのオブジェクトセットの追加」を参照）。
<u>BLACKLIST</u>	なし	<b>[Blacklist Maintenance]</b> メニューを呼び出します。
<u>BOTTOM</u>	なし	リストの末尾までスクロールします。
<u>CANCEL</u>	なし	<b>EXIT</b> と同じです。
<u>CHECK</u> <u>HASH</u> または <u>CHECK</u> <u>HT</u>	なし	バッファプールハッシュテーブルの一貫性をチェックし、検出した矛盾の数を返します。  <b>[REBUILD HASH]</b> も参照してください。
<u>CLOSE</u> <u>BPC</u>	なし	必須のバッファプールキャッシュ。  バッファプールキャッシュを閉じます。これ以降、バッファプールはバッファプールキャッシュなしで実行されます。バッファプールキャッシュは INITIALIZE BPC コマンドを使用して再スタートできます。

コマンド	パラメータ	機能
DELETE	なし	バッファプールとバッファプールキャッシュからすべてのオブジェクトを削除します。  [Directory Information] 画面で入力した場合（「PF キーとダイレクトコマンド」の「DELETE」を参照）。
DELETE	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	指定したオブジェクトをバッファプールとバッファプールキャッシュから削除します（「オブジェクトの削除」を参照）。
DELETE ALL	なし	すべてのオブジェクトをブラックリストから削除します（「ブラックリストからのオブジェクトセットの削除」を参照）。
DELETE BUFFERPOOL または DELETE BP	なし <i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	すべてのオブジェクトをバッファプールからのみ削除します。 指定したオブジェクトをバッファプールからのみ削除します。
DELETE BPC	なし <i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	必須のバッファプールキャッシュ。 すべてのオブジェクトをバッファプールキャッシュからのみ削除します。 必須のバッファプールキャッシュ。 指定したオブジェクトをバッファプールキャッシュからのみ削除します。
DELETE BLACKLIST	なし	ブラックリストエントリを削除できる [Maintain Blacklist] 画面を呼び出します（「ブラックリストの管理」を参照）。
DELETE SET	<i>library-name</i> <i>set-name</i>	指定したオブジェクトセットのすべてのオブジェクトをブラックリストから削除します（「ブラックリストからのオブジェクトセットの削除」を参照）。
DISPLAY ALL	なし	DISPLAY LIST と同じです。
DISPLAY BUFFERPOOL または DISPLAY BP	なし	「バッファプールの選択」の「バッファプールの表示」を参照してください。
DISPLAY BLACKLIST	なし	[Maintain Blacklist] 画面を呼び出します（「ブラックリストの管理」を参照）。
DISPLAY CDIRECTORY	なし	必須のバッファプールキャッシュ。 [Directory Information] 画面を呼び出します。
DISPLAY CGENERAL	なし	必須のバッファプールキャッシュ。 [General BP Cache Statistics] 画面を呼び出します（「一般バッファプールキャッシュ統計」を参照）。

コマンド	パラメータ	機能
DISPLAY CHASH	なし	<b>[BP Cache Hash Table Statistics]</b> 機能（「バッファプールキャッシュのハッシュテーブル統計」を参照）を呼び出し、 <b>[Cache Hash Table Collisions]</b> 画面を表示します。
DISPLAY CLIST	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	必須のバッファプールキャッシュ。 <b>[List Objects]</b> 画面を呼び出します。  DISPLAY LIST コマンドとは異なり、このコマンドでは、リストの先頭にバッファプールキャッシュオブジェクトに関するデータを表示する統計レポートが生成されます。
DISPLAY CLOAD	なし	必須のバッファプールキャッシュ。  <b>[BP Cache Call Statistics]</b> 画面を呼び出します（「バッファプールキャッシュ統計」を参照）。
DISPLAY CSTATISTICS	なし	必須のバッファプールキャッシュ。  <b>[BP Cache Statistics]</b> メニューを呼び出します（「バッファプールキャッシュ統計」を参照）。
DISPLAY DIRECTORY	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	<b>[Directory Information]</b> 画面を呼び出します。
DISPLAY ERAGMENTATION	なし	<b>[Buffer Pool Fragmentation]</b> 画面を呼び出します（「バッファプール統計」を参照）。
DISPLAY FUNCTION	なし	<b>[Internal Function Usage]</b> 画面を呼び出します（「バッファプール統計」を参照）。
DISPLAY GENERAL	なし	<b>[General Buffer Pool Statistics]</b> 画面を呼び出します（「バッファプール統計」を参照）。
DISPLAY HASH または DISPLAY HT	なし	<b>[Buffer Pool Hash Table Statistics]</b> 機能を呼び出し、 <b>[Hash Table Collisions]</b> 画面を表示します。
DISPLAY HDIRECTORY	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	オブジェクトのディレクトリ情報が 16 進形式で表示される <b>[Directory Information Hex]</b> 画面を呼び出します。
DISPLAY HEX	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	オブジェクトのソースが 16 進形式で表示される <b>[Hexadecimal Display]</b> を呼び出します。
DISPLAY LIST	<i>library-name</i> <i>object-name</i> <i>dbid fnr</i>	<b>[List Objects]</b> 画面を呼び出します。  DISPLAY CLIST コマンドとは異なり、このコマンドでは、リストの先頭にバッファプールオブジェクトに関するデータを表示する統計レポートが生成されます。
DISPLAY LOAD	なし	<b>[Buffer Pool Load/Locate Statistics]</b> 画面を呼び出します（「バッファプール統計」を参照）。

コマンド	パラメータ	機能
DISPLAY PERFORMANCE	なし	Naturalバッファプールとバッファプールキャッシュのパフォーマンス関連の統計が表示される <b>[Performance Hints]</b> 画面を呼び出します（「バッファプール統計」および「バッファプールキャッシュ統計」を参照）。
DISPLAY STATISTICS	なし	<b>[Buffer Pool Statistics]</b> メニューを呼び出します（「バッファプール統計」を参照）。
EDIT PRELOADLIST	<i>list-name</i>	<b>[Edit Preload List]</b> 画面を呼び出します（「プリロードリストの管理」を参照）。
EDIT SET	<i>library-name</i> <i>set-name</i>	<b>[Edit Object Set]</b> 画面を呼び出します（「ブラックリストの管理」を参照）。
EXIT	なし	現在の機能／画面を終了して、前の画面を表示します。
FLIP	なし	PFキー行を切り替えます。PF1～PF12キーとPF13～PF24キーの間で表示を切り替えます。
GENERATE PRELOADLIST	<i>list-name</i> <i>gen-library</i>	<b>[Generate Preload List from Buffer Pool]</b> 機能を呼び出します。
INITIALIZE	なし  または 1、2、4、8、12、16	バッファプールとバッファプールキャッシュを再初期化します。テキストレコードサイズを指定しない場合、現在のテキストレコードサイズが使用されます。  <b>[Current Use Count]</b> （「ディレクトリ情報」を参照）が0の場合（次の警告を参照）、またはバッファプールが破壊されている場合にのみこの機能を使用してください。  <b>注意:</b> バッファプール内のオブジェクトがアクティブなセッションで実行されている間にそのバッファプールを再初期化しようとすると、（カウントするSYSBPMユーザー自身ではなく）そのバッファプールの <b>[Current Use Count]</b> が表示された <b>[Confirm Initialization]</b> ウィンドウが開きます。 <b>[Current Use Count]</b> が0（ゼロ）でないときに「Y」を入力してバッファの再初期化を確定すると、予測できない結果がアクティブなセッションで発生し、Naturalが異常終了する可能性があります。
INITIALIZE BP	なし  または 1、2、4、8、12、16	バッファプールのみを再初期化します。テキストレコードサイズを指定しない場合、現在のテキストレコードサイズが使用されます。  上記の「警告」も参照してください。
INITIALIZE BPC	なし	必須のバッファプールキャッシュ。  バッファプールキャッシュのみを再初期化します。バッファプールキャッシュのテキストレコードサイズは固定されます（4KB）。
LAST	なし	最後に入力したSYSBPMダイレクトコマンドを表示します。
LIST PRELOADLIST	<i>list-name</i>	指定したオブジェクトの <b>[List Preload Lists]</b> 画面を呼び出します（「プリロードリストの管理」を参照）。

コマンド	パラメータ	機能
LIST SET	library-name set-name	指定したライブラリまたはオブジェクトの <b>[List Object Sets]</b> 画面を呼び出します（「ブラックリストの管理」を参照）。  set-name として、アスタリスク (*) も使用できます。
MENU	なし	SYSBPM メインメニューを呼び出します（「 <b>SYSBPM の呼び出しと操作</b> 」を参照）。
PRELOADLIST	なし	<b>[Preload List Maintenance]</b> メニューを呼び出します（「プリロードリストの管理」を参照）。
QUIT	なし	EXIT と同じです。
REBUILD HASH または REBUILD HT	なし	CHECK HASH で矛盾が検出された場合、ハッシュテーブルを再作成します。  REBUILD HASH では、現在のハッシュテーブルを削除し、現在のバッファプールの内容から新しいハッシュテーブルを再作成します。
RESET BUFFERPOOL または RESET BP	なし	バッファプールをリセットします（「 <b>バッファプールのリセット</b> 」を参照）。
SELECT BUFFERPOOL または SELECT BP	なし	Natural、DL/I、および Sort タイプのバッファプールにのみ適用されます。  バッファプールの選択リストを呼び出します（「 <b>バッファプールの選択</b> 」を参照）。
SQRT BPC	(下記の構文)	バッファプールキャッシュをソートします（「 <b>ソート済み抽出の表示</b> 」を参照）。
SQRT BUFFERPOOL	(下記の構文)	バッファプールをソートします（「 <b>ソート済み抽出の表示</b> 」を参照）。
STOP	なし	SYSBPM ユーティリティを終了します。
TOP	なし	リストの先頭までスクロールします。
WRITE BP または WRITE BPC または WRITE ALL		オブジェクトのディレクトリデータをローカルファイルまたは PC テキストファイルに書き込みます。  「 <b>ワークファイルへの書き込み</b> 」も参照してください。

## SORT

---

```
SORT [ { BUFFERPOOL } ] [ { OBJECT } ] [ { D } ]  
      [ BP ] [ TOTAL ] [ A ]  
            [ LAST ]
```

## SORT BPC

---

```
SORT BPC { OBJECT } [ { D } ]  
          { TOTAL }  
          { LAST }
```





# 69 バッチ処理

---

- 関連トピック ..... 412

SYSBPM ユーティリティで利用できる機能は、バッチモードでも実行できます。

そのためには、SYSBPM アプリケーションプログラミングインターフェイス USR4340N または USR0340N を使用することをお勧めします（「[アプリケーションプログラミングインターフェイス](#)」を参照）。

オンライン入力コマンドシーケンスをシミュレートして、SYSBPM ユーティリティをバッチモードで使用することもできます。

SYSBPM では Software AG Editor の機能が使用されているため、バッチジョブを作成する場合は、次の事項に注意してください。

- プロファイルパラメータ EDPSIZE を設定します。または、プロファイルパラメータ BPI を TYPE=EDIT を使用して Software AG Editor バッファプール用に設定し、エディタワークファイルをバッチジョブで定義します。
- [Code] フィールドは、一部の SYSBPM 画面では使用できないことに注意してください。
- ダイレクトコマンドを入力するためにコマンド行の入力フィールドにカーソルを置く場合、入力フィールドを省略する必要があることに注意してください。
- SYSBPM ユーティリティをナビゲートするには、端末コマンド %K を使用して PF キーをシミュレートします。例えば、%K3 を使用して [List Objects] 画面を終了します。
- マップのために1行以上入力するには、セッションパラメータ CF で定義した継続文字（デフォルトは %）を使用します。

以下に、上記の事項について説明したトピックと Natural ドキュメントソースを示します。

## 関連トピック

---

トピック	ドキュメント
<a href="#">バッチモードでのブラックリストの管理</a>	SYSBPM ユーティリティ
<a href="#">SYSBPM ダイレクトコマンド</a>	SYSBPM ユーティリティ
<a href="#">EDPSIZE - Software AG Editor 補助バッファプールのサイズ</a>	パラメータリファレンス
<a href="#">BPI - バッファプールの初期化</a>	パラメータリファレンス
<a href="#">Software AG Editor のインストール</a>	インストール
<a href="#">%K と %KP - PF キーと PA キーのシミュレート</a>	端末コマンド
<a href="#">EDBP - Software AG Editor バッファプール定義</a>	パラメータリファレンス
<a href="#">エディタバッファプール</a>	オペレーション
<a href="#">エディタワークファイル</a>	オペレーション
<a href="#">バッチモードでの Natural</a>	オペレーション
<a href="#">非スクリーンモードでの INPUT ステートメントの使用</a>	ステートメント

トピック	ドキュメント
バッチモードでの <i>INPUT</i> ステートメントの使用	ステートメント



# 70 アプリケーションプログラミングインターフェイス

このセクションでは、アプリケーションプログラミングインターフェイス (API) のUSR0340N、USR0341N、およびUSR4340Nについて説明します。これらのAPIは、現在バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方にロードされているNaturalオブジェクトを処理する場合に使用します。APIは、NaturalシステムライブラリSYSEXTで提供されています。

これらのAPIの詳細については、次を参照してください。

- タイプTextの関連するNaturalソースオブジェクトおよびNaturalシステムライブラリSYSEXTのプログラム例。
- 次の表に示すSYSBPM機能。

関連トピック：

- [SYSBPM バッチ処理](#)

API	機能
USR0340N	<ul style="list-style-type: none"><li>■ バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方からオブジェクトを削除します。</li><li>■ オブジェクトを常駐としてマークします。</li><li>■ オブジェクトから常駐フラグを削除します。</li><li>■ オブジェクトのディレクトリ情報を読み取ります。</li><li>■ 一般バッファプール統計およびバッファプールのロード/ロケート統計を取得します。</li></ul> 対応するSYSBPM機能： <a href="#">オブジェクトの削除</a> <a href="#">オブジェクトのリスト</a> <a href="#">ディレクトリ情報</a>

API	機能
	<p>一般バッファプール統計                      バッファプールのロード/ロケート統計</p>
USR0341N	<p>不要になったオブジェクトを削除してバッファプールをクリーンアップするために、ガーベッジを収集します。</p> <p>指定されるオブジェクトの選択条件は、オブジェクトの相対経過時間です。相対経過時間は、オブジェクトがバッファプール内にロードされていた時間です。[BP Last Action] 日付から計算されます。最少経過時間は 30 分です。</p> <p>SYSBPM 機能「<a href="#">List Objects</a>」も参照してください。</p>
USR4340N	<p>SYSBPM ユーティリティをバッチで使用する代わりに、このインターフェイスをバッチ処理に使用することをお勧めします。</p> <p>バッファプールおよびバッファプールキャッシュのいずれかあるいはその両方にロードされたオブジェクトのリストを [Object Size] (ObjSize)、[Total Use Count] (TotalUC)、または [BP Last Action] でソートして表示します。[BP Last Action] は、バッファプールにのみ適用されます。</p> <p>対応する SYSBPM 機能：</p> <p><a href="#">ソート済み抽出の表示</a>  <a href="#">オブジェクトのリスト</a></p>
USR4341N	<p>実行されず、バッファプールにロードされない Natural オブジェクトのブラックリストを管理します。</p> <p>SYSBPM 機能「<a href="#">Blacklist Maintenance</a>」も参照してください。</p>

# 71 SYSCP ユーティリティ・コードページの管理


---

▪ SYSCP の呼び出しと終了 .....	418
▪ ソースのコードページの管理 .....	420
▪ すべてのコードページ .....	432
▪ Unicode プロパティ .....	437

SYSCP ユーティリティを使用すると、現在の Natural for Windows 環境で使用可能なコードページおよび ICU (International Components for Unicode) データファイルに関する情報を表示できます。また、SYSCP ユーティリティを使用して、ソースオブジェクトのコードページ割り当てを変更したり、ソースオブジェクト用にコードページを変換したりすることもできます。

これにより、Natural でコードページが定義または有効化されていない場合や、ソースオブジェクトが不正なコードページまたは Unicode フォーマットに変換されている場合に発生する問題を回避できます。

Natural での Unicode、コードページおよび Unicode 固有の項目のサポートの詳細については、SYSEXV アプリケーションの説明とプレゼンテーション、および次の「関連トピック」を参照してください。

 **注意:** SYSCP ユーティリティの使用は、Natural Security によって制御されます。詳細については、『*Natural Security*』ドキュメントの「SYSCP - Code Page Administration - Utility Profiles」を参照してください。

関連トピック：

- 「Unicode およびコードページのサポート」：Natural ドキュメント
- Unicode：Unicode Consortium (Web サイト <http://www.unicode.org/>)
- ICU：IBM ICU Documentation (Web サイト <http://www-306.ibm.com/software/globalization/icu/documents.jsp>)
- IBM Converter Explorer のドキュメント (Web サイト <http://demo.icu-project.org/icu-bin/convexp>)

## SYSCP の呼び出しと終了

---

SYSCP ユーティリティの呼び出しおよび終了方法と機能の実行方法については、次のセクションで説明します。

### ▶ 手順 71.1. SYSCP ユーティリティを呼び出すには

- 次のシステムコマンドを入力します。

```
SYSCP
```



次の例のような [SYSCP] メニューが表示されます。

```
11:19:07          ***** NATURAL SYSCP UTILITY *****          2007-06-13
User SAG          - Menu -          ICU Version 3.6
                                     Unicode Version 5.0

                                     Function

                                     _ Code Page Maintenance of Sources
                                     _ All Code Pages
                                     _ Unicode Properties
                                     _ Help
                                     _ Exit

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
          Help          Exit                                     Canc
```

画面の上部に ICU および Unicode の現在のバージョンが表示されます。

このメニューに含まれている機能については、以降で説明します。

### ▶手順 71.2. SYSCP 機能を実行するには

- [SYSCP] メニューで、必要な機能の横にある入力フィールドにカーソルを置き、Enter キーを押します。

または:

[SYSCP] メニューで、必要な機能の横にある入力フィールドに任意の文字を入力し、Enter キーを押します。



**注意:** SYSCP ユーティリティ画面のコマンド行には、任意の Natural システムコマンドを入力できます。システムコマンドによって、SYSCP ユーティリティが終了します。

### ▶手順 71.3. SYSCP を終了するには

- PF3 キーまたは PF12 キーを押します。

または:

[SYSCP] メニューの [Exit] を選択します。

## ソースのコードページの管理

---

[Code Page Maintenance of Sources] 機能を使用すると、Natural ライブラリに含まれているソースオブジェクトのコードページ情報のリスト、ソースオブジェクトのコードページ割り当ての変更、およびソースオブジェクトのコードページの変換を行うことができます。

コードページのすべてのメンテナンス機能は標準 IANA 名を参照します（「All Code Pages」の「Cmd」も参照）。コードページのメンテナンス機能を使用する場合は、IANA 以外のコードページ名を使用することはできません。

コードページのメンテナンス機能の結果は、レポート画面に出力されます。詳細については、「機能の結果レポート」を参照してください。

**Code Page Maintenance of Sources** を呼び出すと、次の例のようなメンテナンスメニューが表示されます。

```

15:59:12          ***** NATURAL SYSCP UTILITY *****          2006-10-19
User SAG          - Code Page Maintenance of Sources -

                Code  Function

                L   List Code Page Information of Sources
                C   Check Conversion of Unassigned Sources
                A   Assign Code Page Information to Sources
                K   Check Conversion of Assigned Sources
                T   Convert to Different Code Page
                ?   Help
                .   Exit

Code ..... _

Library ... SAGTEST_      DBID ..... _____ FNR ..... _____
                        Password ..          Cipher ..

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit                                  Canc

```

【Code Page Maintenance of Sources】メニューのフィールドと機能、およびこのファンクションによって提供されるオプションと処理については、次のセクションで説明します。

- [【Code Page Maintenance of Sources】メニュー](#)
- [ソースのコードページ情報のリスト](#)
- [Check Conversion of Unassigned Sources](#)
- [Assign Code Page Information to Sources](#)
- [Check Conversion of Assigned Sources](#)
- [Convert to Different Code Page](#)
- [名前の指定](#)
- [オブジェクト選択リスト](#)

## ■ 機能の結果レポート

### [Code Page Maintenance of Sources] メニュー

次の表に、[Code Page Maintenance of Sources] メニューで表示されるフィールドとその説明を示します。

フィールド	説明
Code	実行する機能の入力コード。例えば、ソースのコードページ情報をリストするには、Lを入力します。
Library	コードページのメンテナンス機能の実行対象となるソースオブジェクトが含まれている Natural ライブラリの名前。  入力されるデフォルト名は、システム変数*LIBRARY-IDで指定されている現在のライブラリの名前です（『システム変数』ドキュメントを参照）。
DBID	指定したライブラリが保存されている Natural システムファイルのデータベース ID (DBID)。  値を指定しない場合、または値が 0 の場合は、現在の FUSER または FNAT システムファイルが使用されます。
FNR	指定した Natural ライブラリが保存されている Natural システムファイルのファイル番号 (FNR)。  値が指定されていないか、または 0 が指定されている場合、現在の FUSER システムファイルか FNAT システムファイルのファイル番号が使用されます。
Password	指定したシステムファイルがパスワードで保護されている場合は、適切な Adabas パスワードを 8 文字で入力する必要があります。
Cipher	指定したシステムファイルが暗号化されている場合は、適切な Adabas サイファコードを 8 桁で入力する必要があります。

### ソースのコードページ情報のリスト

この機能を使用すると、次の例に示すように、Natural ライブラリに含まれているすべてのソースオブジェクトのコードページ情報がリストされます。

```

14:50:26          ***** NATURAL SYSCP UTILITY *****          2006-10-19
User SAG          - List Code Page Information of Sources -
                                     Listed Library SAGTEST

Cmd  Name      Code Page      Type
---  *        *              *
___  LDA1      IBM01147      Local
___  LDA2      IBM01147      Local
___  LDA3      IBM037        Local
___  LDA4      IBM01147      Local
___  MAP1      IBM01147      Map

```

—	MAP2	IBM037	Map
—	MAP3	IBM01147	Map
—	MAP4		Map
—	PGM1		Program
—	PGM2	IBM01147	Program
—	PGM3	IBM01147	Program
—	PROG1	IBM01147	Program
—	PROG2		Program
—	PROG3	IBM037	Program
—	PROG4	IBM01147	Program

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---

Help Exit -- - + ++ Canc

次の表に、[List Code Page Information of Sources] 画面に表示されるフィールドおよび列とその説明を示します。

フィールド/列	説明
Listed Library	「 [Code Page Maintenance of Sources] メニュー」の「Library」を参照してください。
Cmd	選択したソースオブジェクトに対して実行される次の行コマンドの入力フィールド。  LD オブジェクトディレクトリ情報を表示します。  この行コマンドは、『システムコマンド』ドキュメントの「ディレクトリ情報の表示」で説明されているコマンド LIST DIRECTORY <i>object-name</i> に対応しています。
Name	ソースオブジェクトの名前。
Code Page	ソースオブジェクトのコードページ情報 (IANA 名)。ソースオブジェクトがコードページに割り当てられていない場合、この列は空になります。
Type	プログラムやマップなど、Natural オブジェクトのタイプ。

オブジェクトのフィルタ

選択条件を指定すると、[List Code Page Information of Sources] 画面に表示されるオブジェクトのリストを短くすることができます。

#### ▶手順 71.4. 選択条件を指定するには

- 1 列見出し [Name] および [Code Page] の下の入力フィールドで、デフォルトのアスタリスク (\*) を「[名前の指定](#)」にリストされている任意の入力値で置き換えます。

- 2 列見出し [Type] の下の入力フィールドで、デフォルトのアスタリスク (\*) を次のいずれか1つ以上のタイプコード (11個まで) で置き換えます。セパレータ文字は使用しません。

コード	オブジェクトタイプ	コード	オブジェクトタイプ
P	プログラム	A	パラメータデータエリア
N	サブプログラム	G	グローバルデータエリア
S	サブルーチン	L	グローバルデータエリア
M	マップ	C	コピーコード
H	ヘルプルーチン	T	テキスト
M	マクロ	R	レポート
7	ファンクション	Z	記録
3	ダイアログ	4	クラス
5	プロセッサ		
*	すべてのタイプ		

### Check Conversion of Unassigned Sources

この機能を使用すると、未割り当てソースオブジェクトをコードページに変換できるかどうかを確認できます。

未割り当てソースオブジェクトは、コードページ情報を持たないオブジェクトで、元々コードページ情報がサポートされていない Natural バージョンで保存されたオブジェクトです。コードページ情報が指定されていないため、変換できるかどうかをチェックするソースオブジェクトに対して指定するコードページを選択する必要があります。このコードページは、ソースで使用されている文字セットによって異なります。

[Check Conversion of Unassigned Objects] 機能を呼び出すと、次の例のような画面が表示されます。

```

14:56:51          ***** NATURAL SYSCP UTILITY *****          2006-10-19
User SAG          - Check Conversion of Unassigned Sources -

Check if source objects that have no code page information can be
converted from a given code page to a target code page.

Use selection list .. Y

Source code page .... IBM01147_____
Target code page .... IBM01140_____

Object name ..... *_____

```

```

Library ..... SAGTEST_      DBID ..... 10____      FNR ..... 32____
                                Password ..                Cipher ..

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help           Exit                                Canc

```

次の表に、**[Check Conversion of Unassigned Objects]** 画面に表示されるフィールドとその説明を示します。

フィールド	説明
<b>Use selection list</b>	指定したソースオブジェクトに選択処理と自動処理のどちらを使用するかを指定します。「 <a href="#">オブジェクト選択リスト</a> 」を参照してください。
<b>Source code page</b>	指定したソースオブジェクト（コードページはまだ割り当てられていません）を、このコードページから <b>[Target code page]</b> フィールドに入力されているコードページに変換できるかどうかを確認するために使用されるコードページの名前。この変換チェックが成功すると、 <b>[Assign Code Page Information to Sources]</b> 機能で、 <b>[Target code page]</b> に指定されているコードページを使用できます。  入力されるデフォルト名は、*CODEPAGE システム変数によって返される IANA 名です（『システム変数』ドキュメントを参照）。
<b>Target code page</b>	指定した未割り当てソースオブジェクトの変換をチェックするコードページの名前。  入力されるデフォルト名は、*CODEPAGE システム変数によって返される IANA 名です（『システム変数』ドキュメントを参照）。
<b>Object name</b>	処理される 1 つのソースオブジェクトの名前または名前の範囲。有効な入力値については、「 <a href="#">名前の指定</a> 」を参照してください。
<b>Library</b>	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <b>Library</b> 」を参照してください。
<b>DBID</b>	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <b>DBID</b> 」を参照してください。
<b>FNR</b>	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <b>FNR</b> 」を参照してください。
<b>Password</b>	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <b>Password</b> 」を参照してください。
<b>Cipher</b>	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <b>Cipher</b> 」を参照してください。

## Assign Code Page Information to Sources

この機能を使用すると、未割り当てソースオブジェクトにコードページを割り当てることができます。このオブジェクトのソースコードは、特定のコードページに変換されていません。

また、この機能を使用して、コードページが割り当てられているソースオブジェクトのコードページ情報を変更することもできます。この場合、コードページ名 (IANA 名) のみが変更され、このオブジェクトのソースコードは変換されません。

次の表に、[Assign Code Page Information to Sources] 画面に表示されるフィールドとその説明を示します。

フィールド	説明
Use selection list	指定したソースオブジェクトに選択処理と自動処理のどちらを使用するかを指定します。「 <a href="#">オブジェクト選択リスト</a> 」を参照してください。
Forced assignment	コードページ情報を持つソースオブジェクトと持たないソースオブジェクトのいずれかを処理するかを指定します。  可能な値は次のとおりです。  Y 「Yes」を示します。 強制割り当てが有効になります。コードページ情報は、コードページ情報を持つオブジェクトの特定のコードページに変更されます。  N 「No」を示します。 強制割り当てが無効になります (デフォルト設定)。指定したコードページは、コードページ情報を持たないオブジェクトにのみ割り当てられます。
Code page	指定したソースオブジェクトに割り当てられるコードページの名前。  入力されるデフォルト名は、*CODEPAGE システム変数によって返される IANA 名です (『システム変数』ドキュメントを参照)。
Object name	処理される1つのソースオブジェクトの名前または名前の範囲。有効な入力値については、「 <a href="#">名前の指定</a> 」を参照してください。
Library	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Library</a> 」を参照してください。
DBID	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">DBID</a> 」を参照してください。
FNR	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">FNR</a> 」を参照してください。
Password	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Password</a> 」を参照してください。
Cipher	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Cipher</a> 」を参照してください。



## Check Conversion of Assigned Sources

この機能を使用すると、割り当て済みソースオブジェクトを現在のコードページ（オブジェクトディレクトリ情報として入力されている）から別のコードページに変換できるかどうかをテストできます。

割り当て済みソースオブジェクトは、コードページ情報を持つオブジェクトです。

次の表に、[**Check Conversion of Assigned Sources**] 画面に表示されるフィールドとその説明を示します。

フィールド	説明
<b>Use selection list</b>	選択したソースオブジェクトに選択処理と自動処理のいずれを使用するかを指定します。「 <b>オブジェクト選択リスト</b> 」を参照してください。
<b>Current code page</b>	オブジェクト選択条件として使用されるコードページの名前または名前の範囲。有効な入力値については、「 <b>名前の指定</b> 」を参照してください。  デフォルト設定は、すべてのコードページを意味するアスタリスク (*) です。
<b>New code page</b>	指定した割り当て済みソースオブジェクトの変換をチェックするコードページの名前。  入力されるデフォルト名は、*CODEPAGE システム変数によって返される IANA 名です（『システム変数』ドキュメントを参照）。
<b>Object name</b>	処理される1つのソースオブジェクトの名前または名前の範囲。有効な入力値については、「 <b>名前の指定</b> 」を参照してください。
<b>Library</b>	「 [Code Page Maintenance of Sources] メニュー」の「 <b>Library</b> 」を参照してください。
<b>DBID</b>	「 [Code Page Maintenance of Sources] メニュー」の「 <b>DBID</b> 」を参照してください。
<b>FNR</b>	「 [Code Page Maintenance of Sources] メニュー」の「 <b>FNR</b> 」を参照してください。
<b>Password</b>	「 [Code Page Maintenance of Sources] メニュー」の「 <b>Password</b> 」を参照してください。
<b>Cipher</b>	「 [Code Page Maintenance of Sources] メニュー」の「 <b>Cipher</b> 」を参照してください。

## Convert to Different Code Page

この機能を使用すると、割り当て済みソースオブジェクトを現在のコードページ（オブジェクトディレクトリ情報として入力されている）から別のコードページに変換できます。未割り当てソースオブジェクトを変換することはできません。

次の表に、**[Convert to Different Code Page]** 画面に表示されるフィールドとその説明を示します。

フィールド	説明
Use selection list	指定したソースオブジェクトに選択処理と自動処理のどちらを使用するかを指定します。「 <a href="#">オブジェクト選択リスト</a> 」を参照してください。
Current code page	オブジェクト選択条件として使用されるコードページの名前または名前の範囲。有効な入力値については、「 <a href="#">名前の指定</a> 」を参照してください。  デフォルト設定は、すべてのコードページを意味するアスタリスク (*) です。
New code page	指定したソースオブジェクトの変換先となるコードページの名前。  入力されるデフォルト名は、*CODEPAGE システム変数によって返される IANA 名です (『システム変数』ドキュメントを参照)。
Object name	処理される1つのソースオブジェクトの名前または名前の範囲。有効な入力値については、「 <a href="#">名前の指定</a> 」を参照してください。
Library	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Library</a> 」を参照してください。
DBID	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">DBID</a> 」を参照してください。
FNR	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">FNR</a> 」を参照してください。
Password	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Password</a> 」を参照してください。
Cipher	「 <a href="#">[Code Page Maintenance of Sources]</a> メニュー」の「 <a href="#">Cipher</a> 」を参照してください。

## 名前の指定

選択条件として、名前または名前の範囲を指定できます。

下記のオプションのリストで、*value* は1つ以上の文字の任意の組み合わせを表します。

	入力値	選択される項目
	<i>value</i>	<i>value</i> に一致する名前を持つすべての項目。
	*	すべての項目。
	?	入力された疑問符 (?) の個数に相当する任意の文字数のすべての項目。
先頭文字	<i>value</i> *	<i>value</i> で始まる名前を持つすべての項目。  例：AB* 選択対象：AB、AB1、ABC、ABEZ 選択対象外：AA1、ACB
ワイルドカード	<i>value</i> ?	ワイルドカード。  <i>value</i> で始まり、入力された疑問符 (?) の個数に相当する任意の文字数がその後続く名前の、すべての項目。  例：ABC?

	入力値	選択される項目
		選択対象：ABCA、ABCZ 選択対象外：AXC、ABCAA
	<i>value?</i> <i>value?</i> <i>value*</i> <i>value?</i> <i>*value?value*</i>	任意の順序でアスタリスク (*) と疑問符 (?) を指定した <i>value</i> に一致するすべての項目。 例：A?C*Z 選択対象：ABCZ、AXCBBBZ、ANCZ 選択対象外：ACBZ、ABDEZ、AXCBBBZA
開始値	<i>value</i> >	<i>value</i> 以降の名前を持つすべての項目。 例：AB> 選択対象：AB、AB1、BBB、ZZZZZZZ 選択対象外：AA1、AAB
終了値	<i>value</i> <	終了値： <i>value</i> . 以前の名前を持つすべての項目。 例：AX< 選択対象：AB、AWW、AX 選択対象外：AXA、AY

## オブジェクト選択リスト

[Use selection list] オプションを設定すると、メンテナンス機能に対して選択処理と自動処理のいずれを使用するかを指定できます。選択処理を使用する場合は、機能の実行前に、選択したオブジェクトの選択リストが選択画面に表示されます。

[Use selection list] オプションは、[List Code Page Information of Sources] 機能には適用されません。

[Use selection list] の使用可能な設定は、次のとおりです。

- Y "Yes" を示します。  
選択処理が有効になります（デフォルト設定）。指定した選択条件に一致するすべてのソースオブジェクトの選択リストが表示されます。このリストから、処理するオブジェクトを選択できます。
- N "No" を示します。  
選択処理が無効になり、指定した選択条件に一致するすべてのソースオブジェクトに対して、機能が即時に実行されます。

オブジェクト選択リストは、次の例のようになります。

```

16:28:43          ***** NATURAL SYSCP UTILITY *****          2006-10-19
User SAG          - Check Conversion of Assigned Sources -
  Target code page IBM01140
Cmd  Name          Code Page          Message
-----
___  LDA1          IBM01147
___  LDA2          IBM01147
___  LDA3          IBM01147
___  LDA4          IBM01147
___  MAP1          IBM01147
___  MAP2          IBM01147
___  MAP3          IBM01147
___  PGM2          IBM01147
___  PGM3          IBM01147
___  PROG1         IBM01147
___  PROG3         IBM01147
___  PROG4         IBM01147

Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help          Exit          All X          Canc
    
```

次の表に、オブジェクト選択画面に表示されるフィールドおよび列とその説明を示します。

フィールド／列	説明	
Target code page	ソースオブジェクトの割り当てまたは変換のチェックや実行に使用されるコードページ。	
Cmd	選択したソースオブジェクトに対して実行される次のいずれかの行コマンドの入力フィールド。	
	EX または X	メンテナンス機能を実行します。 すべてのオブジェクトに対して行コマンドを一度に発行するには、PF5 キーを押します。
	LD	オブジェクトディレクトリ情報を表示します。 この行コマンドは、『システムコマンド』ドキュメントの「ディレクトリ情報の表示」で説明されているコマンド LIST DIRECTORY <i>object-name</i> に対応しています。
Name	指定した選択条件に一致するソースオブジェクトの名前。	
Code Page	ソースオブジェクトの現在のコードページ情報。	

フィールド／列	説明
Message	この列には、メンテナンス機能の実行の終了時にテキストのみが表示されます。この場合、この列には、ソースオブジェクトの処理ステータスを示すメッセージが表示されます。「機能の結果レポート」も参照してください。

## 機能の結果レポート

メンテナンス機能の実行が終了すると、レポート画面に処理結果が表示されます。レポート画面は、「オブジェクト選択リスト」で示されているオブジェクト選択画面の例と似ています。

次の表に、結果レポート画面に表示されるフィールドおよび列とその説明を示します。

フィールド／列	説明												
Target code page	ソースオブジェクトの割り当てまたは変換のチェックや実行に使用されるコードページ。												
Cmd	入力できません。												
Name	指定した選択条件に一致するソースオブジェクトの名前。												
Code Page	ソースオブジェクトの現在のコードページ情報。												
Message	この列には、処理対象として選択したソースオブジェクトの処理ステータスを示すメッセージが表示されます。このメッセージは、機能の実行の成功または可能性のあるエラー理由を示します。  表示されるメッセージ：  <table border="0"> <tr> <td>Assignment possible</td> <td>ソースオブジェクトを指定したコードページに割り当てることができます。</td> </tr> <tr> <td>Conversion error, at least one code point not translated.</td> <td>ソースオブジェクトを指定したコードページに割り当てたり、変換したりすることができません。</td> </tr> <tr> <td>Code page assigned</td> <td>ソースオブジェクトが指定したコードページに割り当てられています。</td> </tr> <tr> <td>Conversion possible</td> <td>ソースオブジェクトを指定したコードページに変換できます。</td> </tr> <tr> <td>Code page converted</td> <td>ソースオブジェクトが現在のコードページから別のコードページに変換されています。</td> </tr> <tr> <td>Not converted</td> <td>ソースオブジェクトが指定したコードページに変換されていません。これは、このソースオブジェクトが指定したコードページにすでにエンコードされているためです。</td> </tr> </table>	Assignment possible	ソースオブジェクトを指定したコードページに割り当てることができます。	Conversion error, at least one code point not translated.	ソースオブジェクトを指定したコードページに割り当てたり、変換したりすることができません。	Code page assigned	ソースオブジェクトが指定したコードページに割り当てられています。	Conversion possible	ソースオブジェクトを指定したコードページに変換できます。	Code page converted	ソースオブジェクトが現在のコードページから別のコードページに変換されています。	Not converted	ソースオブジェクトが指定したコードページに変換されていません。これは、このソースオブジェクトが指定したコードページにすでにエンコードされているためです。
Assignment possible	ソースオブジェクトを指定したコードページに割り当てることができます。												
Conversion error, at least one code point not translated.	ソースオブジェクトを指定したコードページに割り当てたり、変換したりすることができません。												
Code page assigned	ソースオブジェクトが指定したコードページに割り当てられています。												
Conversion possible	ソースオブジェクトを指定したコードページに変換できます。												
Code page converted	ソースオブジェクトが現在のコードページから別のコードページに変換されています。												
Not converted	ソースオブジェクトが指定したコードページに変換されていません。これは、このソースオブジェクトが指定したコードページにすでにエンコードされているためです。												

## すべてのコードページ

この機能を使用すると、次の例に示すように、現在のNatural環境で使用可能なすべてのコードページがリストされます。

```

17:21:36          ***** NATURAL SYSCP UTILITY *****          2007-08-02
User SAG              - All Code Pages -

Cmd Stat Name                                           Units
-----
_   D   UTF-8                                           1 - 3
_   D   UTF-16                                          2 - 2
_   D   UTF-16BE                                         2 - 2
_   D   UTF-16LE                                         2 - 2
_   D   UTF-32                                           4 - 4
_   D   UTF-32BE                                         4 - 4
_   D   UTF-32LE                                         4 - 4
_   D   UTF16_PlatformEndian                             2 - 2
_   D   UTF16_OppositeEndian                             2 - 2
_   D   UTF32_PlatformEndian                             4 - 4
_   D   UTF32_OppositeEndian                             4 - 4
_   D   UTF-7                                           1 - 4
_   D   IMAP-mailbox-name                               1 - 4
_   D   SCSU                                           1 - 3

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit      Sort      -      +      Canc

```

次の PF キーを使用できます。

- PF8 キー（または Enter キー）。リストを 1 ページ下方にスクロールします。
- PF7 キー。リストを 1 ページ上方にスクロールします。
- PF5 キー。リストのコードページ名を昇順でソートします。リストのサイズによっては、『パラメタリファレンス』の「SORT-ソートプログラムの制御」で説明されているように、SORT プロファイルパラメータを使用してソートバッファのサイズを増やす必要がある場合があります。

次の表に、**[All Code Pages]** 画面に表示される列とその説明を示します。

列	説明	
<b>Cmd</b>	選択したコードページに対して実行される次のいずれかの行コマンドの入力フィールド。	
	N	コードページに使用されている名前をすべて表示します。  IANA (Internet Assigned Numbers Authority) 名は、明確な標準コードページ名です。Natural では、Unicode との相互変換のために、IANA 名をデフォルトコードページ名として使用します (『パラメータリファレンス』ドキュメントに記載されている CP プロファイルパラメータを参照)。IANA 名は、*CODEPAGE システム変数によって返されます (『システム変数』ドキュメントを参照)。  CCSID (Coded Character Set Identifier) は、IBM で識別される文字セットです。  エイリアス名: コードページの 1 つ以上の代替名。
	C	選択したコードページのすべてのコードポイントを表示します。次の「 <a href="#">コードポイントリスト</a> 」を参照してください。
	T	ウィンドウを呼び出して、Unicode との間のコードポイント変換をテストします。次の「 <a href="#">変換テスト</a> 」を参照してください。
<b>Stat</b>	Natural セッション中に使用されるすべてのコードページは、NATCONFIG モジュールで事前定義され、有効になっている必要があります。  この列には、コードページの NATCONFIG ステータスが表示されます。	
	E	コードページは NATCONFIG モジュールで定義され、有効になっています。
	D	コードページは NATCONFIG モジュールで定義されていますが、無効になっています。
	N	コードページは NATCONFIG モジュールで定義されていません。
	NATCONFIG モジュールの詳細については、『オペレーション』ドキュメントの「Natural コンフィグレーションテーブル」を参照してください。	
	<b>Name</b>	内部 ICU 名。
<b>Units</b>	コードポイントに割り当てられているコード単位 (最大および最小バイト数)。	

このセクションでは、次のトピックについて説明します。

- [Code Point List](#)
- [変換テスト](#)

## Code Point List

この機能を使用すると、次の例に示すように、選択したコードページのすべてのコードポイントがリストされます。

```

13:38:33          ***** NATURAL SYSCP UTILITY *****          2007-08-06
+----- Code Points of UTF-8 -----+
! CP: 00000000  U: 0000      NULL                      !
! CP: 00000001  U: 0001 ?    START OF HEADING        !
! CP: 00000002  U: 0002 ?    START OF TEXT          !
! CP: 00000003  U: 0003 ?    END OF TEXT            !
! CP: 00000004  U: 0004 ?    END OF TRANSMISSION    !
! CP: 00000005  U: 0005 ?    ENQUIRY                    !
! CP: 00000006  U: 0006 ?    ACKNOWLEDGE             !
! CP: 00000007  U: 0007 ?    BELL                          !
! CP: 00000008  U: 0008 ?    BACKSPACE                 !
! CP: 00000009  U: 0009 ?    CHARACTER TABULATION   !
! CP: 0000000A  U: 000A ?    LINE FEED (LF)         !
! CP: 0000000B  U: 000B ?    LINE TABULATION        !
! CP: 0000000C  U: 000C ?    FORM FEED (FF)        !
! CP: 0000000D  U: 000D ?    CARRIAGE RETURN (CR)  !
! CP: 0000000E  U: 000E ?    SHIFT OUT              !
! CP: 0000000F  U: 000F ?    SHIFT IN                !
+-----+
_  D   ibm-912_P100-1995          1 - 1
_  D   ibm-913_P100-2000         1 - 1
_  N   ISCII,version=0           1 - 4
_  N   ISCII,version=1           1 - 4

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit  LByte Prop  --  -  +      <<  >      Canc

```

このリストには、次の情報が表示されます。

- コードページのコードポイントのバイトシーケンス (CP)
- 対応する Unicode コードポイントのバイトシーケンス (U)
- Unicode 文字 この文字が現在の端末エミュレーションによって解釈されない場合、置換文字 (コードページで定義されます。ここでは (?) が表示されます。
- Unicode 文字の標準名



次の表に、各コードポイントリストに使用できる PF キーとその説明を示します。

PF キー	機能
PF4	<p>最大単位が 1 バイトのコードページには適用できません。</p> <p><b>[Leading Bytes of Code Point]</b> ウィンドウを開き（関連セクションを参照）、表示するバイト範囲を入力します。</p> <p>現在の先頭バイト数を確定してこのウィンドウを閉じるには、PF3 キーまたは Enter キーを押します。</p> <p>現在の入力をキャンセルしてこのウィンドウを閉じるには、PF12 キーを押します。</p>
PF5	カーソルが配置されたリスト項目について、 <b>[Unicode Properties]</b> 画面（関連するセクションを参照）を起動します。
PF6	バイト範囲の最初の（先頭ではない）バイトを 16 進値 0x00 にリセットします。
PF7	<p>選択したバイト範囲を 1 ページ上方にスクロールします（「<b>Specifying Leading Bytes</b>」も参照）。</p> <p>UTF-16 または UTF-32 コードページでは、すべてのバイト範囲をスクロールできます。</p>
PF8 (または Enter キー)	<p>選択したバイト範囲を 1 ページ下方にスクロールします（「<b>Specifying Leading Bytes</b>」も参照）。</p> <p>UTF-16 または UTF-32 コードページでは、すべてのバイト範囲をスクロールできます。</p>
PF10	画面の左端に移動します。
PF11	画面の右に移動します。

### Specifying Leading Bytes

この機能は、最大 1 バイト単位のコードページには適用できません。

コードポイントの特定の先頭バイトのバイト範囲（16 進値 0x00~0xFF）を表示するには、**[Leading Byte of Code Point]** ウィンドウを使用します。

次の UTF-8 コードページの例では、先頭バイトとして 16 進値 0x22 および 0x32 が入力されています。

```
+----- Leading Bytes of Code Point -----+
!                                     !
!   Maximum number of bytes .. 3      !
!                                     !
!                                     !
!   Enter leading bytes ..... 00 2232 00 !
!                                     !
!                                     !
!                                     !
+-----+
```

PF3 キー（または Enter キー）を押すと、コードポイントリストに 16 進値 0x00223200~0x002232FF のバイトが表示されます。



**注意:** UTF-16LE や UTF-32LE など、バイトスワップされたコードページの場合、バイトは逆のバイト順で読み取られ、表示されます。

## 変換テスト

選択したコードページから CP プロファイルパラメータで定義されているデフォルトのコードページ (\*CODEPAGE の値) へのコードポイント変換をテストできます。

- 英数文字列から Unicode コードポイントへの変換または逆の変換をテストできます。
- また、16 進値から Unicode コードポイントへの変換または逆の変換もテストできます。

次の例に示すコードページ（ここでは ibm-1140\_P100-1997）の変換ウィンドウには、次の情報が含まれています。

- コードポイントに割り当てられているバイト単位数（最大および最小バイト数）
- 英数文字列およびそれと等価の 16 進値
- 対応する Unicode コードポイント

```

+----- Test Conversion of ibm-1140_P100-1997 -----+
!
! Code page units .. 1 - 1   (minimum, maximum of bytes)
!
!
! Code Page Characters
! Alphanumeric .. ABC
! Hexadecimal ... C1 C2 C3 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40
!                   40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40 40
!
! Unicode
! Code points ... 0041 0042 0043 0020 0020 0020 0020 0020 0020 0020
!                   0020 0020 0020 0020 0020 0020 0020 0020 0020 0020
!                   0020 0020 0020 0020 0020 0020 0020 0020 0020 0020
!                   0020 0020 0020 0020 0020 0020 0020 0020 0020 0020
!
!
!
+-----+

```

### ▶手順 71.5. 文字またはコードポイントを変換するには

- 1 変換されるリテラルの文字列またはコード単位シーケンスを入力するフィールドをアクティブにします。

[**Alphanumeric**] フィールド（デフォルトの入力フィールド）にリテラル文字列を入力するには、PF6 キーを押します。

または:

[**Hexadecimal**] フィールドに 16 進値を入力するには、PF7 キーを押します。

または:

[**Unicode**] フィールドに Unicode コードポイントを入力するには、PF8 キーを押します。

- 2 Enter キーを押します。

フィールドに入力された値が、同等のコードポイントまたはリテラルの文字列に変換されます。

## Unicode プロパティ

この機能を使用すると、デフォルトのコードページ（\*CODEPAGE の値）に含まれている文字（次の例では、コードページ IBM01140 の文字 A）に対して、Unicode 文字のプロパティが TRUE（yes）または FALSE（no）のいずれであるかが表示されます。

```

14:43:19          ***** NATURAL SYSCP UTILITY *****          2008-09-23
User SAG          - Unicode Properties -

Default code page ... IBM01140

Alpha character ..... A      C1      hexadecimal      Substitution .. ? 3F
Unicode code point .. 0041

Unicode char. name .. LATIN CAPITAL LETTER A

Alphabetic ..... yes          Control ..... no
Alphanumeric ..... yes        Space ..... no
Lower case ..... no           Whitespace ..... no
Upper case ..... yes          Blank ..... no
Digit ..... no                Punctuation .... no
Hexadecimal ..... yes         Combining ..... no
Graphic ..... yes             Surrogate ..... no
Printable ..... yes           Right to left .. no

```

```
Command ===>
```

```
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---  
      Help      Exit      Uni      Canc
```

[**Alpha character**] フィールドに、プロパティを確認する文字を入力できます。Unicode コードポイントを入力する場合は、PF5 キーを押します。

画面に表示される Unicode 文字のプロパティの詳細については、Web サイト <http://www.unicode.org/Public/4.1.0/ucd/UCD.html> にある Unicode Consortium のドキュメント『Unicode Character Database』を参照してください。

# 72 SYSEDIT ユーティリティ・エディタバッファプ ールサービス

---

▪ Natural Security ライブラリプロファイルの定義 .....	440
▪ SYSEDIT の呼び出しと操作 .....	440
▪ General Information .....	441
▪ Generation Parameters .....	443
▪ Users .....	444
▪ Logical Files .....	445
▪ Recovery Files .....	446
▪ System Administration Facilities .....	447

エディタバッファプールサービスユーティリティ SYSEDT は、Natural 管理者向けのものであり、次の目的で使用されます。

- エディタバッファプールのパラメータおよびランタイム情報の表示
- パラメータの変更
- 論理的なワークファイルおよびリカバリファイルの削除

## Natural Security ライブラリプロファイルの定義

---

Natural Security がインストールされている場合、SYSEDT ユーティリティのライブラリセキュリティプロファイルを作成する必要があります。

詳細については、『Natural Security』ドキュメントの「*Library Maintenance*」を参照してください。

## SYSEDT の呼び出しと操作

---

以下に参考情報を示します。

- [SYSEDT ユーティリティの呼び出し](#)
- [SYSEDT 機能の呼び出し](#)
- [ダイレクトコマンドヘルプの使用](#)

### SYSEDT ユーティリティの呼び出し

#### ▶ 手順 72.1. SYSEDT ユーティリティを呼び出すには

- Natural システムコマンド「SYSEDT」を入力します。

次の機能を含む SYSEDT メインメニューが表示されます。

- [General Information](#)
- [Generation Parameters](#)
- [Users](#)
- [Logical Files](#)
- [Recovery Files](#)
- [System Administration Facilities](#)

このセクションの残りの部分で、SYSEDT 機能について説明します。

## SYSEDT 機能の呼び出し

### ▶手順 72.2. SYSEDT 機能呼び出すには

- SYSEDT メインメニューで、対応するファンクションコードを入力します。

または:

SYSEDT メインメニューで、適切な PF キーを押します。

または:

システムコマンド「SYSEDT」を入力した後、SYSEDT メインメニューで使用可能ないずれかのファンクションコードを入力することにより、SYSEDT メインメニューをスキップして目的の機能に直接アクセスします。

例：コマンド SYSEDT G を使用すると、SYSEDT ユーティリティの [General Information] 画面が呼び出されます。

次のセクションで、各機能について説明します。

### ダイレクトコマンドヘルプの使用

コマンド行で疑問符 (?) を入力すると、SYSEDT ユーティリティ内部で使用可能なすべてのダイレクトコマンドがアルファベット順に表示されます。

## General Information

---

### ▶手順 72.3. [General Information] 機能呼び出すには

- SYSEDT メインメニューで、ファンクションコード「G」を入力します。

または:

SYSEDT メインメニューで、PF10 (GInfo) キーを押します。

[General Information] 画面が表示され、エディタバッファプールの現在のステータスの概要が表示されます。

項目	表示される内容
Usage Statistics	後に続く項目の、現在使用可能な合計数、現在使用されている数、および使用可能な数に対する現在使用中の数のパーセント。
Buffer Pool Blocks	エディタバッファプール内のブロック数。
Work File Records	エディタワークファイル内のレコードの数。
Control	制御レコードの数。常に1つです。
Work	ワークレコードの数。
Recovery Records	リカバリレコードの数。
Logical Files	論理ファイルの数。
Requests	読み取り要求と書き込み要求の合計数、バッファプールブロックの読み取り要求と書き込み要求の数（[Pool] 列）、およびワークファイルまたはリカバリファイルの読み取り要求と書き込み要求の数（[File] 列）。[Copy] 列には、（ロックされた読み取り要求とは異なり）対応するバッファプールブロックから最終的に削除される読み取り要求が表示されます。
Read Work	論理ファイルレコードの読み取り要求の数。論理ファイルレコードは、バッファプール内（[Pool] 列）またはワークファイル内（[File] 列）にあります。論理ファイルレコードは、ロックされた要求またはコピー要求によって読み取ることができます。ロックされた要求とは、レコードが一定期間バッファプールに保持される要求であり、コピー要求とは、読み取られた後にバッファプールから削除される要求です。
Write Work	論理ファイルレコードの書き込み要求の数。使用可能な空きブロックがない場合、レコードをバッファプール（[Pool]）に書き込むか、またはワークファイル（[File]）に移動することができます。
Read Recovery	エディタワークファイル内のリカバリレコードの読み取り要求の数。
Write Recovery	エディタワークファイル内のリカバリレコードの書き込み要求の数。
Timeout Values	項目を秒単位のタイムアウト値とともに表示します。これらのタイムアウト値は、PF5 (Updat) キーを押した後に変更できます。再び PF5 (Save) キーを押すと、動的に設定されます。バッファプールの再起動中、変更済みの値は保持されません。代わりにワークファイル制御レコードからの値が使用されます。
Logical Files	この時間の経過後、この時間中にアクセスされていなかった論理ファイルは削除されます。
Files Delete Check	この時間の経過後、すべての論理ファイルが削除可能かどうかを調べるチェックが定期的に実行されます。
Changed Blocks	この時間の経過後、変更済みのブロックを、ワークファイルに書き込むことによって解放できます。
Unchanged Blocks	この時間の経過後、変更されていないブロックを、ワークファイルに書き込むことによって解放できます。
Locked Blocks	この時間の経過後、ロックされた要求で読み取られたブロックを、ワークファイルに書き込むことによって解放できます。



## Generation Parameters

### ▶手順 72.4. [Generation Parameters] 機能呼び出すには

- SYSEDT メインメニューで、ファンクションコード「P」を入力します。

または:

SYSEDT メインメニューで、PF11 (Parms) キーを押します。

[Generation Parameters] 画面が表示されます。

次の表で、[Generation Parameters] 画面に表示される各パラメータをアルファベット順に説明します。

パラメータ	説明
CTOUT	変更済みのバッファプールブロックのタイムアウト値 (秒単位)。
DDNAME	JCL 定義のエディタワークファイルの名前。
DSNAME	ワークファイルデータセットの名前。
DTOUT	論理ファイルが削除可能かをチェックする時間間隔 (秒単位)。
FMODE	ワークファイル名のモード (A1~Z9)。 このパラメータは、CMS にのみ適用されます。それ以外の環境では表示されません。
FTOUT	アクセスされていない論理ファイルが削除されるまでのタイムアウト値 (秒単位)。
IMSG	オペレータコンソールで発行される初期化メッセージ。
ITOUT	バッファプールブロック初期化のタイムアウト値 (秒単位)。
LRECL	ワークファイルレコードの長さ。
LTOUT	ロックされたバッファプールブロックのタイムアウト値 (秒単位)。
MAXLF	エディタバッファプール内の論理ファイルの最大数。
PWORK	ワークレコードとして使用されるワークファイルレコードのパーセント。
RECNUM	ワークファイルレコードの合計数。
RWORK	標準の論理ファイルのワークレコードのパーセント。
UTOUT	変更されていないバッファプールブロックのタイムアウト値 (秒単位)。

これらのパラメータの詳細については、『オペレーション』ドキュメントの「Software AG Editor の使用」セクションを参照してください。

[Start] 列は、バッファプールの再起動を示しています。上記のパラメータに対して、次の開始値を表示できます。

値	説明
L	対応するパラメータの値は、エディタパラメータモジュールまたはワークファイル定義から取得されます。
C	対応するパラメータ値を変更すると、バッファプールコールドスタートが強制的に実行されます。リカバリレコードは失われます。
W	対応するパラメータ値を変更すると、バッファプールのウォームスタートが実行されます。リカバリレコードは保持されます。

## パラメータ値の変更

### ▶手順 72.5. パラメータ値を変更するには

- 1 PF5 (Updat) キーを押します。
- 2 PF5 キーを再び押して、新しいパラメータ値をエディタワークファイル制御レコード内に保存します。

新しい値は、バッファプールが再び起動されるとアクティブになります。

## Users

### ▶手順 72.6. [Users] 機能を起動するには

- SYSEDIT メインメニューで、ファンクションコード「U」を入力します。

または:

SYSEDIT メインメニューで、PF7 (Users) キーを押します。

[Users] 画面が表示され、次の情報が示されます。

項目	表示される内容
User ID	Natural ユーザー ID。
Logical Files	各ユーザーに定義されている論理ファイルの数。
Pool Blocks	各ユーザーのバッファプールブロックの数。
Work Records	各ユーザーのワークレコードの数。
Recovery Files	各ユーザーのリカバリファイルの数。
Recovery Records	各ユーザーのリカバリレコードの数。

リストされた各ユーザーに対して、次のいずれかの行コマンドを実行できます。行コマンドは、目的のユーザーの横の [C] 列に入力します。

行コマンド	機能
?	ウィンドウの入力フィールドに入力できるすべての有効な行コマンドのリストを含むウィンドウを開きます。
.	現在の画面を終了し、前の画面に戻ります。
/ または P	行を画面上部に配置します。
F	このユーザーの論理ファイルを選択します。
R	このユーザーのリカバリファイルを選択します。
D	このユーザーのすべての論理ファイル/リカバリファイルを削除します。

## Logical Files

### ▶手順 72.7. [Logical Files] 機能呼び出すには

- SYSEDТ メインメニューで、ファンクションコード「F」を入力します。

または:

SYSEDТ メインメニューで、PF6 (Files) キーを押します。

[Logical Files] 画面が表示され、次の情報が示されます。

項目	表示される内容
File No.	論理ファイル番号。
User ID	Natural ユーザー ID。
Type	論理ファイルタイプ。
Pool Blks	各論理ファイルに現在使用されているバッファプールブロックの数。
File Recs	各論理ファイルに現在割り当てられているワークファイルレコードの数。
Last Access	各論理ファイルの最後の読み取り要求または書き込み要求の日時。

リストされた各論理ファイルに対して、次のいずれかの行コマンドを実行できます。行コマンドは、目的のユーザーの横の [C] 列に入力します。

行コマンド	機能
?	ウィンドウの入力フィールドに入力できるすべての有効な行コマンドのリストを含むウィンドウを開きます。
.	現在の画面を終了し、前の画面に戻ります。
/ または P	行を画面上部に配置します。
S	このユーザーの論理ファイルを選択します。
D	論理ファイルを削除します。

## Recovery Files

### ▶手順 72.8. [Recovery Files] 機能呼び出すには

- SYSEDT メインメニューで、ファンクションコード「R」を入力します。

または:

SYSEDT メインメニューで、PF8 (Recov) キーを押します。

[Recovery Files] 画面が表示され、次の情報が示されます。

項目	表示される内容
User ID	Natural ユーザー ID。
Member	ライブラリメンバ名。
Library	ライブラリ名です。
Type	ライブラリタイプ。
Recs	各リカバリファイルのリカバリレコードの数。
Creation Date/Time	リカバリファイルの作成日時。

リストされた各リカバリファイルに対して、次のいずれかの行コマンドを実行できます。行コマンドは、目的のユーザーの横の [C] 列に入力します。

行コマンド	機能
?	ウィンドウの入力フィールドに入力できるすべての有効な行コマンドのリストを含むウィンドウを開きます。
.	現在の画面を終了し、前の画面に戻ります。
/ または P	行を画面上部に配置します。
S	このユーザーのリカバリファイルを選択します。
D	リカバリファイルを削除します。

## System Administration Facilities

### ▶手順 72.9. [Administration Facilities] 画面を呼び出すには

- SYSEDT メインメニューで、ファンクションコード「A」を入力します。

または:

SYSEDT メインメニューで、PF5 (Admin) キーを押します。

[Administration Facilities] 画面が表示され、エディタバッファプールを終了するか SYSEDT ユーティリティを終了するかを選択できる機能が提供されます。

バッファプールを終了するように選択した場合 (ファンクションコード T)、確認を求めるウィンドウが表示されます。

確認のために「Yes」を入力した場合、新しいウィンドウが表示され、エディタバッファプールをすぐに再起動するかどうかを尋ねられます。

再び「Yes」を入力すると、バッファプールがすぐに再起動されます。これにより、変更した生成パラメータをすぐにアクティブ化できます。










「No」を指定した場合は、SYSEDT が終了し、TP 環境外部で、例えば、エディタワークファイルのサイズ変更などのアクションを実行できます。『オペレーション』ドキュメントの「エディタワークファイル」および「エディタバッファプール」も参照してください。



# 73 SYSERR ユーティリティ

Natural アプリケーションを開発する場合、エラーメッセージや情報メッセージと Natural コードを区別して、別々に管理できます。これにより、メッセージの標準化、さまざまなメッセージのメッセージ範囲の事前定義、メッセージの他の言語への翻訳、メッセージの詳細を説明する長文テキストのメッセージへの付加などを簡単に行うことができます。

SYSERR ユーティリティを使用して、独自のアプリケーション固有メッセージを作成できます。また、既存の Natural システムメッセージをカスタマイズすることもできます。

	メッセージの全般的な情報
	<b>SYSERR</b> 呼び出し
	機能
	パラメータ
	ダイレクトコマンド
	大文字 - <b>ERRUPPER</b>
	文字置換 - <b>ERRCHAR</b>
	メッセージのアンロード - <b>ERRULDUS</b>
	メッセージのロード - <b>ERRLODUS</b>
	アプリケーション プログラミングインターフェイス <b>USR0020P</b>





## 74 メッセージの全般的な情報

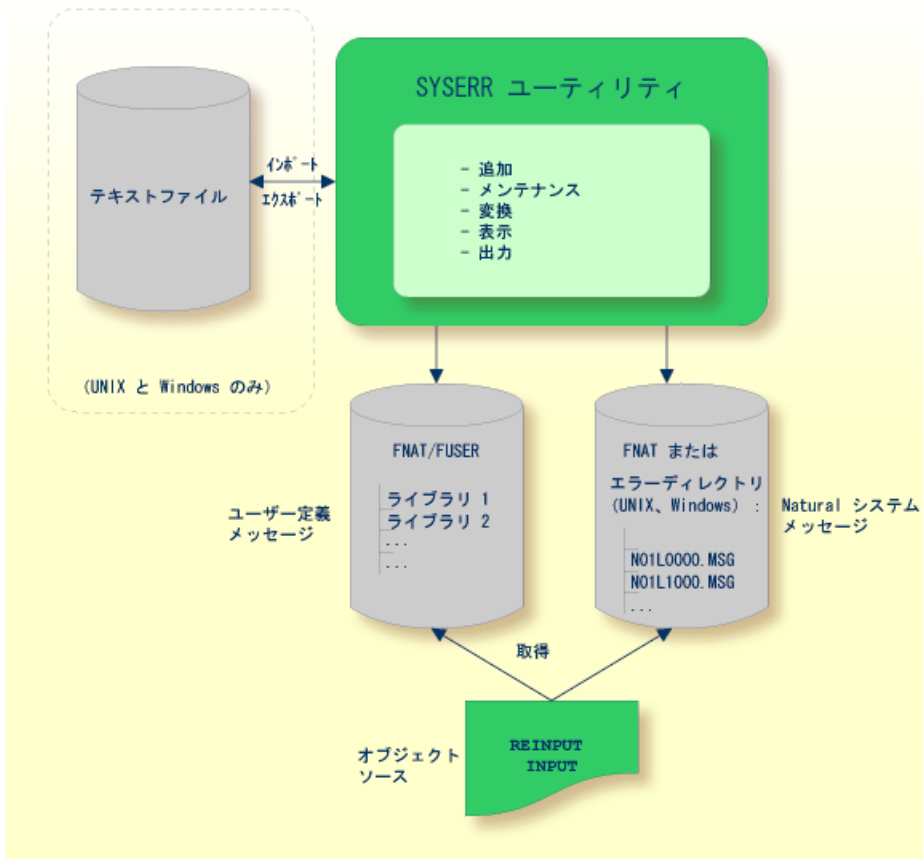
---

■ メッセージタイプ .....	452
■ メッセージの言語 .....	453
■ メッセージの発行 .....	454
■ Natural システムショートメッセージの取得 .....	455
■ ユーザー定義ショートメッセージの取得 .....	455
■ メッセージ情報の取得 .....	456

## メッセージの全般的な情報

このセクションでは、SYSERRユーティリティを使用して管理できるメッセージとメッセージ言語のタイプ、およびNaturalシステム環境でのメッセージの発行および取得方法について説明します。

次の図に、SYSERRユーティリティの機能とNaturalでのメッセージの処理を示します。



## メッセージタイプ

メッセージタイプには、Natural（システム）メッセージとユーザー定義メッセージの2つがあります。

Natural システムメッセージは、Natural ニュークリアスおよび Natural ユーティリティによって発行されます。Natural システムメッセージは Software AG によって提供され、Natural システムファイル FNAT にメッセージファイルとして保存されます。Natural システムメッセージは、NAT0230 のように、NAT で始まり 4 桁の数字が続きます。

ユーザー定義メッセージは、ユーザーが作成したアプリケーションによって発行されます。ユーザー定義メッセージは、システムファイル FUSER または FNAT のライブラリ (SYS- ライブラリを含む) にメッセージファイルとして保存されます。


メッセージは、他の言語に翻訳できます。各言語は、個別のメッセージファイルに保存されます。各ライブラリおよびメッセージファイルには、最大 9999 のメッセージを保存できます。

メッセージテキストには、次の 4 つのタイプがあります。

- Natural システムショートメッセージ
- Natural システムロングメッセージ
- ユーザー定義ショートメッセージ
- ユーザー定義ロングメッセージ

ショートメッセージは 1 行のメッセージで、対応するエラー状況が発生すると、メッセージ行に表示されます。

ロングメッセージは対応するショートメッセージの詳細な説明で、問題の解決方法も含まれています。

 **注意:** Natural システムメッセージを変更すると、誤ったメッセージになったり、Natural の新しいバージョンのリリース時に変更が失われたりする可能性があります。

## メッセージの言語

メッセージは、最大 60 の言語で作成できます。『システム変数』ドキュメントのシステム変数「\*LANGUAGE」を参照してください。

次のルールおよび制限が適用されます。

- Natural システムショートメッセージは、最初に英語で入力する必要があります。その後、他の言語に翻訳できます。
- Natural システムロングメッセージは、英語で入力できますが、他の言語に翻訳できません。
- ユーザー定義ショートメッセージは、任意の言語で入力でき、その後、他の言語に翻訳できます。
- ユーザー定義ロングメッセージは、任意の言語で入力できますが、同じ言語の対応するショートメッセージがある必要があります。

## メッセージの発行

---

このセクションでは、Natural プログラムで Natural システムショートメッセージまたはユーザー定義ショートメッセージを発行するために使用する Natural ステートメント INPUT および REINPUT について説明します。

### ▶手順 74.1. プログラムで Natural システムショートメッセージを発行するには

#### ■ Natural ステートメントの入力

```
INPUT WITH TEXT *-nnnn
```

または

```
REINPUT WITH TEXT *-nnnn
```

*nnnn* は、要求されたメッセージの番号です（先頭のゼロは省略可能）。

### ▶手順 74.2. プログラムでユーザー定義ショートメッセージを発行するには

#### ■ Natural ステートメントの入力

```
INPUT WITH TEXT *nnnn
```

または

```
REINPUT WITH TEXT *nnnn
```

*nnnn* は、要求されたメッセージの番号です（先頭のゼロは省略可能）。

## メッセージテキストのダイナミック置換

メッセージテキストには、表記 *:n* で識別される変数部分を含めることができます。*n* はオカレンス 1~7 を表します。変数部分は、ランタイムの値に置き換えられます。

詳細については、「INPUT 構文1- ダイナミック画面レイアウトの指定」の「*operand3*」または『ステートメント』ドキュメントの「REINPUT」の「*operand3*」を参照してください。

## Natural システムショートメッセージの取得

プログラムでNaturalシステムショートメッセージが参照されている場合、Naturalでは、Naturalシステムファイル FNAT 内の要求されたメッセージ番号が次の順序で検索されます。

1. システム変数 \*LANGUAGE で設定されている現在の言語コード
2. 言語コード 1 (英語)

上記のいずれも見つからない場合は、存在しないメッセージがプログラムで参照されています。表示されるのは、NAT0230 など、接頭辞が NAT のメッセージ番号のみです。

## ユーザー定義ショートメッセージの取得

プログラムでユーザー定義ショートメッセージが参照されている場合、Naturalでは、最初に、システム変数 \*LANGUAGE で設定されている現在の言語コードで要求されたメッセージ番号 *nnnn* が検索されます (『システム変数』ドキュメントを参照)。メッセージがない場合は、言語コード 1 (英語) で要求されたメッセージ番号 *nnnn* が検索されます。ここにもメッセージがない場合は、言語コード 1 で要求されたメッセージ番号 *n000* (*n*は要求されたメッセージ番号の最初の数字) が検索されます。

この3段階の検索は、最初に現在のライブラリで実行されます。ここで何も見つからなかった場合は、対応するメッセージが見つかるまで、他のライブラリが同様に検索されます。

検索されるライブラリの順序は、次のとおりです。

1. システム変数 \*LIBRARY-ID で設定されている現在のライブラリ。
2. *steplib*。Natural Security がインストールされている場合は、現在のライブラリの Natural Security プロファイルで *steplib* が指定される順序。
3. システム変数 \*STEPLIB で設定されているデフォルトの *steplib*。
4. システムファイル FUSER (\*) のライブラリ SYSTEM。
5. システムファイル FNAT (\*) のライブラリ SYSTEM。

(\*) 現在のライブラリのファイル名が SYS で始まる場合は、SYSTEM FUSER の前に SYSTEM FNAT が検索されます。

## メッセージ情報の取得

---

ショートメッセージが表示された場合、問題状況についての追加情報を検索できます。

- システムコマンド HELP を使用すると、Natural システムログメッセージまたはユーザー定義ログメッセージを表示できます。
- システムコマンド LASTMSG を使用すると、最後に表示されたメッセージのショートテストおよびエラー状況に関する追加情報をリストできます。表示される情報には、最後のメッセージの前に表示された可能性がある関連エラーメッセージが含まれます。

これらのコマンドについては、『システムコマンド』ドキュメントを参照してください。

# 75 SYSERR 呼び出し

## ▶手順 75.1. SYSERR ユーティリティを呼び出すには

- システムコマンド SYSERR を入力します。

SYSERR ユーティリティのメインメニューが表示されます。

```
16:15:35          ***** NATURAL SYSERR UTILITY *****          2008-09-18
                        - Menu -

      Code  Function
      ----  -
      AD    Add new messages
      DE    Delete messages
      DI    Display messages
      MO    Modify messages
      PR    Print messages
      SC    Scan in messages
      SE    Select messages from a list
      TR    Translate messages into another language
      ?    Help
      .    Exit
      ----  -

Code .. __  Message type .... US
           Library ..... SYSTEM__
           Message number .. 1__ - 9999
           Language codes .. 1_____

Command ==>
```

```
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---  
      Help           Exit                                           Canc
```

SYSERR ユーティリティのメインメニューから、メッセージを追加および管理するすべての機能を実行できます。各機能については、「[機能](#)」を参照してください。機能とともに適用されるパラメータの概要については、「[パラメータ](#)」を参照してください。パラメータの使用に適用される制限については、「[機能](#)」の各機能の説明を参照してください。

SYSERR ユーティリティでは、詳細なオンラインヘルプを利用できます。フィールド固有のヘルプ情報を表示するには、関連するフィールドに疑問符を入力して Enter キーを押すか、カーソルを目的のフィールドに置いて PF1 キーを押します。



# 76 機能

---

■ メッセージの追加 .....	460
■ メッセージの削除 .....	462
■ メッセージの表示 .....	463
■ メッセージの変更 .....	464
■ メッセージの出力 .....	466
■ メッセージのスキャン .....	467
■ リストからのメッセージの選択 .....	469
■ メッセージの他の言語への翻訳 .....	471
■ システムファイルへのメッセージオンコピー - SYSMAIN ユーティリティ .....	473

SYSERRユーティリティ機能を呼び出すには、SYSERRメインメニューの入力フィールドに、必要な機能に対応するコードおよび1つ以上のパラメータを入力します。このセクションでは、メニューに表示される機能と、各機能に対して指定できるパラメータについて説明します。パラメータの一般的な使用方法については、「[パラメータ](#)」を参照してください。

## メッセージの追加

### ▶手順 76.1. 新しいメッセージを追加するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	AD
Message Type	NS Natural システムショートメッセージ NL Natural システムロングメッセージ US ユーザー定義ショートメッセージ UL ユーザー定義ロングメッセージ  ロングメッセージは、ショートメッセージの説明であるため、対応するショートメッセージがある場合にのみ追加できます。
Library	任意の既存 Natural ライブラリ。
Message Number	追加するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。メッセージを1つのみ追加する場合は、左側の [Message Number] フィールドに新しいメッセージの番号を入力して右側のフィールドをクリアするか、または両方のフィールドに番号を入力します。
Language Codes	メッセージを追加する言語のコード。メッセージタイプが NS または NL の場合、言語コードは英語の <b>1</b> である必要があります。その他のメッセージタイプの場合は、フィールドに入力した最初の言語コードが使用され、他の言語は無視されます。

- 2 Enter キーを押します。

[Add Short Message] 画面が表示されます。

```

11:20:23          ***** NATURAL SYSERR UTILITY *****          2003-09-16
                    - Add Short Message -

Number           Short Message
-----
SYSERR1004
                .....1.....2.....3.....4.....5.....+..

Sample ..... Message sample number 0000

```

- 3 メッセージ番号の横の入力行にショートメッセージテキストを入力します。サンプルメッセージの使用（上の例で示されている）については、次の「[サンプルメッセージのコピー](#)」を参照してください。
- 4 Enter キーを押して、新しいショートメッセージを保存します。
- 5 対応するロングメッセージテキストを追加するには、PF9/Long キーを押します。

[Add Long Message] 画面が表示されます。

```

11:21:59          - Add Long Message SYSERR1004 Language 1 -          2003-09-16
1 Tx. Message sample number 1004
2      .
3      .
4 Ex. .
5      .
6      .
7      .
8      .
9      .
10     .
11     .
12     .
13     .
14     .
15     .
16     .
17     .
18 Ac. .
19     .
20     .

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Add      Exit      -      +      Copy      Canc

```

- 6 3つの入力エリア Tx. (テキスト)、Ex. (説明)、および Ac. (アクション) にテキストを入力します。
- 7 Enter キーを押して、ロングメッセージを保存します。

- 8 PF9/Short キーを押して、ショートメッセージに戻るか、またはメッセージ番号の範囲を選択している場合は次のショートメッセージを昇順で追加します。
- 9 PF3 キーまたは PF12 キーを押して、SYSERR メインメニューに戻ります。

または:

メッセージ番号の範囲を選択している場合は、PF8 キーまたは PF7 キーを押して、次のショートメッセージを昇順または降順で追加します。

### サンプルメッセージのコピー

メッセージ入力行の下に、Sample というラベルの行が表示されています。この行に含まれているテキストをメッセージ入力行にコピーするには、空のメッセージ入力行に .C と入力します。

サンプルメッセージに文字列 0000 が含まれている場合、この文字列 0000 は、次の例に示すように、サンプルがコピーされたときにメッセージ番号で置き換えられます。

```

11:21:13          ***** NATURAL SYSERR UTILITY *****          2003-09-16
                  - Add Short Message -

Number           Short Message
-----
SYSERR1004      Message sample number 1004
                .....1.....2.....3.....4.....5.....+..
Sample ..... Message sample number 0000
    
```

サンプルメッセージの作成方法については、「SYSERR - ダイレクトコマンド」の **SAMPLE** コマンドの説明を参照してください。

## メッセージの削除

### ▶手順 76.2. メッセージを削除するには

- SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	DE
Message Type	NS Natural システムショートメッセージ NL Natural システムロングメッセージ US ユーザー定義ショートメッセージ

フィールド	入力値
	UL ユーザー定義ロングメッセージ  対応するショートメッセージを削除しないでロングメッセージを削除することはできませんが、対応するロングメッセージを削除しないでショートメッセージを削除することはできません。ロングメッセージが存在するショートメッセージを削除すると、両方削除することを確認するメッセージが表示されます。
Library	任意の既存 Natural ライブラリ。
Message Number	削除するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。
Language Codes	メッセージを削除する言語のコード。指定したメッセージが、使用可能なすべての言語で削除されることを示すには、アスタリスク (*) を入力します。

## メッセージの表示

### ▶手順 76.3. メッセージを表示するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	DI
Message Type	NS Natural システムショートメッセージ NL Natural システムロングメッセージ US ユーザー定義ショートメッセージ UL ユーザー定義ロングメッセージ
Library	任意の既存 Natural ライブラリ。
Message Number	表示するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。
Language Codes	メッセージを表示する言語のコード。指定可能な言語コードは1つのみです。複数の言語コードを指定した場合は、最初のコードのみが使用され、他のコードはすべて無視されます。

- 2 Enter キーを押します。

ショートメッセージの場合、[Display Short Messages] 画面が表示されます。

```

15:41:11          ***** NATURAL SYSERR UTILITY *****          2003-09-17
                  - Display Short Messages -

Number           Short Message (English)
-----
NAT0001         Missing/invalid syntax; undefined variable name/keyword.
NAT0002         No file is available with specified name or number.
NAT0003         Invalid character string for file name or file number.
NAT0004         DEFINE DATA must be the first statement if present.
NAT0005         Closing parenthesis missing in arithm/logical expression.
NAT0006         ESCAPE statement used when no processing loop active.
NAT0007         Invalid THRU or TO clause in READ LOGICAL or HISTOGRAM.

```

次のページを表示するには、PF8 キーを押します。

ロングメッセージの場合、[Display Long Message] 画面が表示されます。PF8 キーを押して次のページを表示するか、または PF7 キーを押して前のページを表示すると、メッセージが1つずつ表示されます。[Display Long Messages] 画面は、次の「メッセージの変更」で示す [\[Modify Long Message\]](#) 画面に似ています。

## メッセージの変更

### ▶手順 76.4. メッセージを変更するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	MO
Message Type	NS Natural システムショートメッセージ NL Natural システムロングメッセージ US ユーザー定義ショートメッセージ UL ユーザー定義ロングメッセージ
Library	任意の既存 Natural ライブラリ。
Message Number	変更するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。
Language Codes	メッセージを変更する言語のコード。指定可能な言語コードは1つのみです。複数の言語コードを指定した場合は、最初のコードのみが使用され、他のコードはすべて無視されます。

- 2 Enter キーを押します。

[Modify Short Message] 画面が表示されます。

```

18:52:33          ***** NATURAL SYSERR UTILITY *****          2003-09-16
                    - Modify Short Message -

Number           Short Message (English)
-----
SYSERR1004      Message sample number 1004
                  ....+....1....+....2....+....3....+....4....+....5....+..

 1 Tx. Input missing.
 2      .
 3      .
 4 Ex. Input value missing in field XYZ.
 5      Enter an alphanumeric value.
 6      .
 7      .
 8      .
18 Ac. Enter value in field XYZ.
19      .
20      .

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Mod           Exit           -       +           Copy           Canc

```

参照用に、ロングメッセージが画面の下半分に表示されます。

ロングメッセージを変更する場合は、[Modify Long Message] 画面が表示されます。

```

18:54:02          - Modify Long Message SYSERR1004 (English) -          2003-09-16
 1 Tx. Input missing.
 2      .
 3      .
 4 Ex. Input value missing in field XYZ.
 5      Enter an alphanumeric value.
 6      .
 7      .
 8      .
 9      .
10     .
11     .
12     .
13     .
14     .
15     .
16     .
17     .
18 Ac. Enter value in field XYZ.
19     .

```

```

20      .
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Mod      Exit      -      +      Copy      Canc

```

- 3 Enter キーを押して、変更を保存します。
- 4 メッセージ番号の範囲を選択している場合は、PF8 キーまたは PF7 キーを押して、次のメッセージを昇順または降順で変更します。

## メッセージの出力

### ▶手順 76.5. メッセージを出力するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	PR
Message Type	NS Natural システムショートメッセージ NL Natural システムロングメッセージ US ユーザー定義ショートメッセージ UL ユーザー定義ロングメッセージ
Library	任意の既存 Natural ライブラリ。
Message Number	出力するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。
Language Codes	メッセージを出力する言語のコード。指定可能な言語コードは1つのみです。複数の言語コードを指定した場合は、最初のコードのみが使用され、他のコードはすべて無視されます。

- 2 ENTER キーを押します。

出力ウィンドウが表示されます。

```

+-----Print Natural System Messages-----+
!                                     !
!   Language code .... 1               !
!                                     !
!   Long texts, too .. N               !
!   Message number ... 1___ - 25      !
!   Lines per page ... 60_            !
!   Left margin ..... 10              !
!   Top margin ..... 0_               !

```



```

! Bottom margin .... 0_          !
! Printer ID ..... PRT1_____ !
!                               !
!                               !
+-----+

```

- 3 プリントウィンドウに表示されているオプションおよび論理プリンタ名を指定します。

論理プリンタ名の詳細については、『*Natural* ステートメント』のステートメント DEFINE PRINTER を参照してください。

- 4 Enter キーを押して、選択したメッセージをプリンタに出力します。

#### ▶手順 76.6. すべての *Natural* システムメッセージを出力するには

- SYSERR メインメニューのフィールドに、次の値を入力します。

- ファンクションコード PR
- メッセージタイプ NS または NL
- メッセージ番号 1~9999
- 言語コード 1 (英語) または 2 (ドイツ語)

ライブラリ ID は必要なく、入力しても無視されます。

## メッセージのスキャン

この機能を使用すると、メッセージをスキャンして特定の文字列を検索できます。スキャンできるのは、ショートメッセージのみです。

#### ▶手順 76.7. メッセージをスキャンするには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	SC
Message Type	NS Natural システムショートメッセージ US ユーザー定義ショートメッセージ
Library	任意の既存 <i>Natural</i> ライブラリ。
Message Number	スキャンするメッセージの範囲を示す、最初と最後のメッセージの番号 (最大 4 桁)。

フィールド	入力値
Language Codes	1～9、A～Z、および a～y を使用して 9 文字以内の言語コードを指定するか、またはアスタリスク (*) を指定します。

2 Enter キーを押します。

スキャンウィンドウが表示されます（次の「例」を参照）。

3 次の情報を指定します。

Scan Value(s)	4 つの空のフィールドに、検索する用語を 4 つまで入力します。スキャンでは、大文字および小文字の該当する用語が検索されます。
OR/AND/NOT	検索用語ごとに異なる条件オペレータ OR、AND、または NOT を使用できます。  次の「例」では、「buffer」および「pool」という単語が含まれているすべてのショートメッセージが検索されます。  オペレータ NOT が指定されている場合は、「buffer」および「pool」が含まれていないすべてのショートメッセージが検索されます。
Absolute	このフィールドをマークすると、メッセージテキストの長い文字列に含まれている文字列も検索されます。  例えば、値「meter」を探してスキャンすると、「parameter」や「millimeter」などの単語も検索されます。
Immediate	このフィールドをマークすると、個別のメッセージが順に表示されます。マークしない場合は、検索の終了後にメッセージのリストが表示されます。  [Language Codes] フィールドで複数の言語またはアスタリスク (*) を指定した場合は、[Immediate] をマークする必要があります。

例：

```

+-----+
!  Scan value(s)      Or/And/Not      !
!  -----          -
!  BUFFER_____     AND              !
!  POOL_____       !
!  _____       !
!  _____       !
!  _____       !
!  Absolute ..... X  !
!  Immediate ..... _ !
!  _____       !
+-----+

```

4 Enter キーを押します。

指定したスキャン条件が適用されるすべてのメッセージが表示されます。検索文字列が含まれる単語は強調表示されます。次に、スキャンの出力例を示します。

```
11:32:27          ***** NATURAL SYSERR UTILITY *****          2000-07-04
                  - Scan in Short Messages -

Number           Short Message (English)
-----
NAT0777         Buffer pool full.
```

この画面で PF10 キーを押すと、現在のスキャンの検索条件を表示できます。

## リストからのメッセージの選択

この機能を使用すると、メッセージの範囲を表示し、さらに処理する個別のメッセージを選択できます。表示できるのは、ショートメッセージのみです。

### ▶手順 76.8. メッセージを選択するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	SE
Message Type	NS Natural システムショートメッセージ US ユーザー定義ショートメッセージ
Library	任意の既存 Natural ライブラリ。 ライブラリ ID にアスタリスク (*) が追加されている場合は、使用可能なすべてのライブラリのリストが選択用に表示されます。
Message Number	選択用に表示するメッセージの範囲を示す、最初と最後のメッセージの番号（最大 4 桁）。
Language Codes	メッセージを表示する言語のコード。 複数の言語コードを指定すると、最初のコードのショートメッセージのみが表示されます。各メッセージの使用可能な言語を表示するには、アスタリスク (*) を入力します。

- 2 Enter キーを押します。

[Select Messages] 画面が表示されます。

```

11:33:21          ***** NATURAL SYSERR UTILITY *****          2000-07-04
                    - Select Messages -

Se Number          Short Message (English)          Languages
                    short          long
-----
__ NAT0001          Missing/invalid syntax; undefined variable name/ 1          1
__ NAT0002          No file is available with specified name or numb 1          1
__ NAT0003          Invalid character string for file name or file n 1          1
__ NAT0004          DEFINE DATA must be the first statement if prese 1          1
__ NAT0005          Closing parenthesis missing in arithm/logical ex 1          1
__ NAT0006          ESCAPE statement used when no processing loop ac 1          1
__ NAT0007          Invalid THRU clause in READ LOGICAL/HISTOGRAM st 1          1

```

3 列 [Se] で、次の行コマンドのいずれかを入力します。

コマンド	機能
DE	メッセージを削除します。
DI	[Select Messages] 画面を、上記の「メッセージの表示」に示されている <b>[Display Short Messages]</b> 画面に変換します。また、このコマンドを使用して選択したメッセージをリストの先頭に配置し、次の「.X」で説明されているように表示されるメッセージの数を減らします。
LA	メッセージの翻訳先の言語を表示します。
MO	メッセージを変更します。
PR	メッセージをプリンタに出力します。
SH	ショートメッセージを表示します。 このコマンドは、SYSERR メインメニューの [Language Code] フィールドにアスタリスク (*) が入力されている場合にのみ使用できます。
TR	メッセージを別の言語に翻訳します。
.X	選択したメッセージをリストの先頭に配置して、より狭いメッセージ範囲を定義します。これにより、表示されるメッセージの数が削減されます。  このコマンドを使用して選択したメッセージがリストの先頭に配置され、このメッセージの上にリストされていたメッセージは表示から削除されます。これにより、SYSERR メインメニューのメッセージ範囲がリセットされ、[Select Messages] 画面はここで選択したメッセージで始まります。
.Y	選択したメッセージ以前のメッセージのみをリストして、より狭いメッセージ範囲を定義します。  このコマンドを使用して選択したメッセージ以降にリストされていたすべてのメッセージは、表示から削除されます。これにより、SYSERR メインメニューのメッセージ範囲がリセットされ、[Select Messages] 画面はここで選択したメッセージで始まります。

4 Enter キーを押して続行します。

## メッセージの他の言語への翻訳

この機能を使用すると、ある言語のショートメッセージを1つ以上の他の言語に翻訳できます。ロングメッセージを他の言語に翻訳するには、「[メッセージの追加](#)」で説明されている [Add Long Messages] 機能を使用します。

### 手順 76.9. メッセージを翻訳するには

- 1 SYSERR メインメニューのフィールドに、次の値を入力します。

フィールド	入力値
Code	TR
Message Type	NS Natural システムショートメッセージ US ユーザー定義ショートメッセージ
Library	任意の既存 Natural ライブラリ。
Message Number	選択用に表示するメッセージの範囲を示す、最初と最後のメッセージの番号（最大4桁）。
Language Codes	言語コードを9つまで指定します。言語コードは、1~9、A~Z、および a~y の単一英数字です。

- 2 Enter キーを押します。

[Translate Short Message] 画面が表示されます。

```

11:42:21          ***** NATURAL SYSERR UTILITY *****          2003-07-04
                    - Translate Short Message -

Number ..... SYSERR0002

Languages ... 123...7.9ABCD.....

----- . . . + . . . 1 . . . + . . . 2 . . . + . . . 3 . . . + . . . 4 . . . + . . . 5 . . . + . .
English   Short message number 2 _____
German    Short message number 2 (German) _____
French    Short message number 2 (French) _____
Spanish   _____
Italian   _____
Dutch     _____
Turkish   Short message number 2 (Turkish) _____
Danish    _____
Norwegian Short message number 2 (Norwegian) _____
----- . . . + . . . 1 . . . + . . . 2 . . . + . . . 3 . . . + . . . 4 . . . + . . . 5 . . . + . .

```

```

1 Short message number 2
4 Line 4 of message number 2 (long text, English)
18 Line 18 of message number 2 (long text, English)

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Mod   Help           Exit           -           +           Opts           Canc

```

[Number] フィールドおよび [Language] フィールドには、すでにメッセージが存在するメッセージ番号と言語コードが表示されます（画面例では、1、2、3、7、9、A、B、C、および D）。

[Number] および [Language] の下のセクションには、言語およびその対応する翻訳がリストされるか、またはこの機能が呼び出されたときに SYSERR メインメニューの [Language Code] フィールドで指定した新しい言語への翻訳を入力するための空の入力行が表示されます（画面例では、1、2、3、4、5、6、7、8、および 9。4、5、6、および 8 は新しい言語）。

参照用に、画面下部には、言語/ショートメッセージのセクションで最初にリストされている言語（画面例では英語）に対応しているロングメッセージが 3 行表示されます。行 1、4、および 18 は、デフォルトで表示されます。ロングメッセージの他の行を表示するには、3 つの行番号（1、4、または 18）を別の行番号で上書きし、Enter キーを押します。

- 3 指定した新しい言語の横の入力行に翻訳を入力します。
- 4 Enter キーを押します。

#### ▶手順 76.10. ショートメッセージの翻訳を変更するには

- 1 [Translate Short Message] 画面で PF10 キーを押します。

[Options] ウィンドウが表示されます。

```

+----- Options -----+
!
! Modification of all fields allowed ..... N          !
!
! Currently recognized language codes ..... 123456789  !
!
+-----+

```

- 2 [Modification] フィールドに Y を入力して、デフォルトを上書きします。このウィンドウでは、翻訳する言語コードを 9 つまで指定することもできます。

#### ▶手順 76.11. 翻訳を空の入力行にコピーするには

- 1 [Translate Short Message] 画面で、空の入力行の先頭から 2 つの入力位置に .C と入力します。

- 2 別の言語ですでに存在しているショートメッセージの任意の場所にカーソルを合わせます（コピーできるのは表示モードで表示されているテキストのみです）。
- 3 Enter キーを押します。

## システムファイルへのメッセージョンコピー -SYSMAIN ユーティリティ

メッセージやメッセージの範囲は、ライブラリからライブラリに、またはシステムファイルからシステムファイルにコピーできます。

▶手順 76.12. ライブラリまたはファイルのメッセージを別のライブラリまたはファイルにコピーするには

- SYSMAIN ユーティリティの [Copy Messages] 機能を使用します。

または:

アンロード機能 **ERRULDUS** およびロード機能 **ERRLODUS** を使用します。





# 77 パラメータ

---

▪ Message Type .....	476
▪ Library .....	476
▪ Message Number .....	476
▪ Language Codes .....	477

このセクションでは、SYSERRメインメニューの機能に指定できるパラメータについて説明します。特定の機能でのパラメータの使用に適用される制限については、「[機能](#)」で説明しています。

### Message Type

---

処理するメッセージのタイプを指定します。次のテーブルに、使用可能なメッセージタイプを示します。

タイプ	説明
NS	Natural システムショートメッセージ
NL	Natural システムロングメッセージ
US	ユーザー定義ショートメッセージ
UL	ユーザー定義ロングメッセージ

### Library

---

メッセージが作成されるか、または管理されるライブラリを指定します。Natural システムメッセージ（メッセージタイプ NS および NL）にアクセスする場合は、ライブラリを指定する必要があります。[**Library**] フィールドの入力値は無視されます。

### Message Number

---

メッセージ範囲の最初の番号と最後の番号を指定します。ライブラリおよび言語の最大メッセージ番号は 9999 です。メッセージ番号 0000 は使用できません。メッセージ番号を 1 つのみ指定するには、左側の [**Message Number**] フィールドにメッセージ番号を入力するか、または両方のフィールドに番号を入力します。

---

## Language Codes

---

使用可能な言語コードを9~60個指定します。言語コードは、1~9、A~Z、およびa~yの単一英数字です。言語コードを表示または選択するには、[Language Codes] フィールドの先頭に疑問符(?)を入力して、Enterキーを押します。詳細については、『システム変数』ドキュメントのシステム変数「\*LANGUAGE」を参照してください。



# 78      ダイレクトコマンド

SYSERR メインメニューでは、次のコマンドをコマンド行に入力して実行できます。

コマンド	機能
LAYOUT	有効なメッセージ範囲を指定して、メッセージを分類します。範囲を重複させることができます。新しいメッセージを追加できるのは、その番号がレイアウトで指定した範囲内にある場合のみです。
NEXT	指定されているメッセージ番号範囲内で、次のフリーメッセージ番号を検索します。フリーとは、このメッセージ番号が使用可能で、どの言語のメッセージファイルにも割り当てられていないことを意味します。
NEXTTAB	NEXT と同じですが、特定の番号を選択するための番号のリストを返します。
RESTART	ユーティリティを終了しないで、SYSERR（およびそのデフォルト値）を再初期化します。
SAMPLE	<p>[EditSAMPLE message] ウィンドウを呼び出します。このウィンドウでは、新しいショートメッセージの作成時にマスタとして使用するサンプルメッセージを作成したり、変更したりします。</p> <p>サンプルメッセージを作成または変更するには、次の手順に従います。</p> <ul style="list-style-type: none"><li>■ [EditSAMPLE message] ウィンドウのエディタエリアで、必要なメッセージテキストを入力するか、または既存のテキストを変更します。文字列 <b>0000</b>（テキストと組み合わせるまたは組み合わせずに）を入力すると、この文字列 <b>0000</b> は、新しいメッセージのコピー時にそのメッセージの番号で置き換えられます。「機能」セクションの「<a href="#">サンプルメッセージのコピー</a>」も参照してください。</li><li>■ [Read or Write sample] フィールドに <b>W</b> を入力して、エントリを保存します。</li><li>■ [Library] フィールドに、サンプルメッセージを使用するライブラリの名前を入力します。[Library] フィールドに入力しない場合、サンプルメッセージは Natural システムメッセージに適用されます。</li><li>■ PF3 キーを押して、[Edit SAMPLE message] ウィンドウを終了します。</li></ul> <p>各言語およびライブラリにサンプルメッセージを 1 つ定義できます。</p>

## ダイレクトコマンド

---

コマンド	機能
SECURITY	セキュリティウィンドウを呼び出します。このウィンドウでは、セキュリティで保護されている Adabas ファイルおよび VSAM ファイルにアクセスするためのパスワードおよびサイファを入力できます。
SHIFT	有効にすると、新しいメッセージの変更または追加を確定したときに、ショートメッセージのテキストが左側の余白に自動的にシフトされます。
TRACE	データベースアクセス数をカウントします。指定したメッセージ番号に達すると、ウィンドウが表示されます。デフォルトの番号は 900 です。0 に設定すると、トレース機能が終了します。コマンド TRACE ON および TRACE OFF は、コマンド行に直接入力できます。TRACE ON により、アクセスカウンタが 900 に設定され、TRACE OFF により、アクセスカウンタが 0 に設定されます。
USEREXIT	Natural システムライブラリ SYSERR でプログラム USEREXIT を呼び出します。

# 79 大文字 - ERRUPPER

---

Naturalシステムメッセージは小文字で表示されます。使用している端末で小文字が適切に表示されない場合は、Naturalシステムライブラリ SYSERR でプログラム ERRUPPER を実行して、メッセージを小文字から大文字に変換します。

ただし、大文字に変換したメッセージを再度小文字に変換することはできません。小文字メッセージを復元するには、次の2つの方法があります。

- ERRLODUS またはオブジェクトハンドラを使用して、メッセージを再ロードします。
- 変換前に ERRULDUS またはオブジェクトハンドラを使用して、フリー言語コードへの小文字メッセージをアンロードします。これにより、常にバックアップが存在します。

『[オブジェクトハンドラ](#)』ドキュメントも参照してください。





# 80 文字置換 - ERRCHAR

使用している端末で特定の文字が適切に表示されない場合は、その文字を検索して、任意の新しい文字で置換できます。これには、NaturalシステムライブラリSYSERRでプログラムERRCHARを実行します。ただし、置換できるのはNaturalシステムショートメッセージ内の文字のみです。ERRCHARを使用して、特定の文字をスキャンし、この文字を表す16進コードを別の16進コードで置き換えます。

プログラムERRCHARを実行すると、次の機能を持つERRCHARメニューが表示されます。

- 指定された文字のスキャン
- 文字のスキャンと置換
- メッセージの16進形式での表示
- 使用している端末用のEBCDIC文字テーブル
- 文字セットERRCSETを使用した翻訳

ERRCHARメニューには、次の入力フィールドがあります。

フィールド	説明
Error Number	検索または検索／置換操作に含まれるメッセージの範囲。
Language Code	検索または検索／置換操作に含まれるNaturalシステムショートメッセージの言語コード。
Scan Value	スキャン対象となる16進値。
Replace Value	検索されたすべてのスキャン対象値を置き換える16進値。使用している端末のEBCDIC文字テーブルを使用して、端末で表示できる文字を判別します。



# 81 メッセージのアンロード - ERRULDUS

プログラム ERRULDUS を使用すると、Natural でサポートされているすべてのメッセージタイプ（SYSERR を使用して作成されたユーザー定義ログメッセージとショートメッセージ、および Natural システムメッセージ）がアンロードされます。メッセージは、FNAT システムファイルまたは FUSER システムファイルから読み込まれ、ワークファイル 2 に書き込まれます。

## ▶手順 81.1. ERRULDUS を呼び出すには

- SYSERR メインメニューのコマンド行で、「ERRULDUS」と入力します。

ERRULDUS メニューが表示されます。

```
16:11:13          ***** NATURAL SYSERR UTILITY *****          08-09-18
- ERRULDUS (Unload Texts to Work File 2) -

Code  Function
-----
US    User supplied short error texts
UL    User supplied long  error texts
U     User supplied short AND long texts
NS    NATURAL short error texts
NL    NATURAL long  error texts
H     NATURAL help texts
.     Exit
-----

Code ..... _
Source Library .. SYSERR__      Source Language Code .. 01
Target Library .. SYSERR__      Target Language Code .. 01
Error Number .... 1__ - 9999
Replace ..... N
Report ..... ON_

Please enter valid code.
Command ==>
```

## メッセージのアンロード - ERRULDUS

```
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Exec  Help           Exit                               Canc
```

ERRULDUS メニューには、次の入力フィールドが表示されます。

フィールド	説明
Code	アンロードされるメッセージのタイプ。有効なタイプは次のとおりです。  US ユーザー定義ショートメッセージ UL ユーザー定義ロングメッセージ U ユーザー定義ショートメッセージおよびロングメッセージ NS Natural システムショートメッセージ NL Natural システムロングメッセージ H Natural ヘルプテキスト . 処理の終了
Source Library	メッセージのアンロード元のライブラリの名前。名前は、アスタリスク (*) を使用して切り捨てることができます。アスタリスク (*) のみを入力すると、すべてのライブラリのメッセージがアンロードされます。メッセージタイプ NS または NL の場合、[Source Library] は無視されます。
Source Language Code	アンロードされるメッセージの言語コード。
Target Library	メッセージのロード先となるライブラリの名前。メッセージタイプ NS または NL の場合、[Target Library] は無視されます。
Target Language Code	メッセージのロード先となる言語コード。
Starting Error Number	最初にアンロードされるメッセージの番号。開始番号が指定されていない場合は、1 が使用されます。
Ending Error Number	最初にアンロードされるメッセージの番号。
Replace	ターゲットライブラリを上書きするには、Y を入力します。デフォルトは N ("No" を示します) です。
Report	ON を指定すると、アンロードの完了時に、アンロードされたすべてのメッセージのリストが表示されます。OFF を指定すると、リストは表示されません。

ERRULDUS では、[Code] フィールドのピリオド (.) が検出されるまで入力が読み込まれます。アンロード後に、統計のリストが表示されます。


## 82 メッセージのロード - ERRLODUS

---

プログラム ERRLODUS を使用すると、プログラム ERRULDUS を使用して事前にアンロードされたメッセージがロードされます。

メッセージは、ワークファイル 2 から読み込まれ、FNAT システムファイルまたは FUSER システムファイルに書き込まれます。アンロード時に [Replace] に Y が指定されていた場合は、ERRLODUS によってシステムファイルの既存のメッセージが上書きされます。

ERRLODUS に ON を指定すると、ロードの完了時に、ロード（追加または置換）されたすべてのメッセージのリストが表示されます。

 **注意:** Natural Security では、ライブラリのセキュリティプロパティの [Utilities] オプションの設定にかかわらず、管理者のみが ERRLODUS をオンラインで使用できます。『Natural Security』ドキュメントの「Library Maintenance」にある「Utilities」も参照してください。



# 83 アプリケーションプログラミングインターフェイス

## USR0020P

---

Natural システムライブラリ SYSEXT のアプリケーションプログラミングインターフェイス USR0020P を使用すると、FNAT システムファイルまたは FUSER システムファイルからメッセージを読み取ることができます。これにより、例えば、Natural システムライブラリ SYSERR を使用せずにロングメッセージをアプリケーションに表示できます（独自のユーザー定義ヘルプシステムとして）。

Natural システムライブラリ SYSEXT にログオンし、コマンド行にコマンド MENU を入力します。表示されるリストで、疑問符 (?) を使用してプログラム USR0020P をマークします。プログラムで実行する機能を選択できるウィンドウが表示されます。I を入力すると、USR0020P の使用についての詳細情報が表示されます。





# 84 SYSEXT ユーティリティ-Natural アプリケーション プログラミングインターフェイス

---

▪ SYSEXT の基本機能 .....	492
▪ SYSEXT の呼び出しと終了 .....	493
▪ SYSEXT ユーティリティ画面 .....	495
▪ SYSEXT ユーティリティ機能の実行 .....	495
▪ Natural API の使用 .....	499
▪ Natural API のリスト .....	500

## SYSEXT ユーティリティ - Natural アプリケーションプログラミングインターフェイス

SYSEXT ユーティリティを使用すると、現在のシステムライブラリ SYSEXT に含まれている Natural アプリケーションプログラミングインターフェイス (API) を検索し、テストできます。

Natural API は Natural サブプログラム (カタログ化オブジェクト) で、データにアクセスして必要に応じて変更したり、Natural に固有のサービス、サブコンポーネント、またはサブ製品を実行したりする場合に使用します。Natural API の目的は、情報を取得または変更したり、Natural ステートメントではアクセスできないサービスを使用したりすることです。

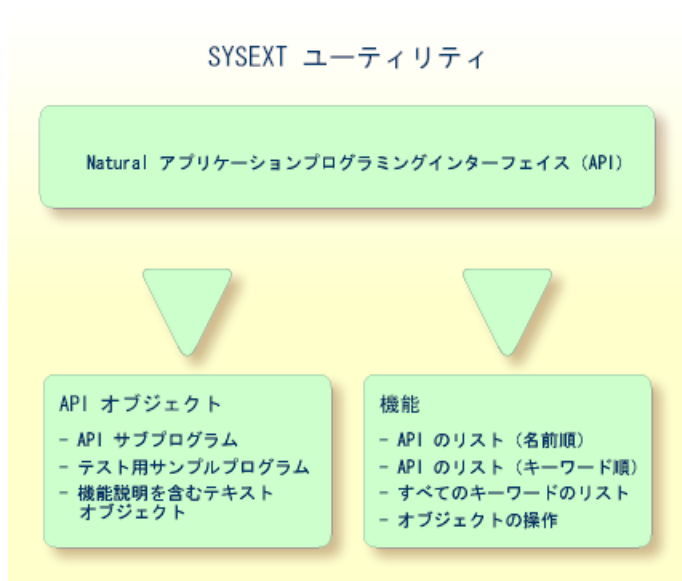
関連トピック：

- 「Natural ユーザー出口」 - 『オペレーション』ドキュメント
- 「Application Programming Interfaces」 - 『Natural Security』ドキュメント
- 「Application Programming Interfaces」 - 『Natural SAF Security』ドキュメント
- 「[SYSAPI - Natural アドオン製品の API](#)」 - 『ユーティリティ』ドキュメント

## SYSEXT の基本機能

SYSEXT ユーティリティには、各 Natural API について、機能説明、1つのプログラム例、および API に固有のキーワードがあります。

次の図に、SYSEXT で提供されている、API をテストし、実装するための Natural オブジェクトおよび主な機能の概要を示します。



## Natural API に提供されているオブジェクト

各 Natural API に提供されている一般的な Natural オブジェクトのタイプは、次のセクションでリストされています。特定の API で必要となる追加オブジェクトは、リストされていません。

API のすべての関連オブジェクトは、システムファイル FNAT のライブラリ SYSEXT に含まれています。

次のテーブルで、*nnnn* は、API に割り当てられている 4 桁の数字です。相互に関連しているすべての API オブジェクトには、識別番号が付いています。

オブジェクト名	説明
USR <i>nnnn</i> N	指定された機能を実行する API サブプログラム（カタログ化オブジェクト）。
USR <i>nnnn</i> P	API の効果のテストに使用できるプログラム例（ソースオブジェクト）。 プログラム例によって、対応するサブプログラム USR <i>nnnn</i> N が呼び出されます。
USR <i>nnnn</i> T	対応する API の説明が含まれているテキストオブジェクト。API と API に関連するキーワードの目的、機能、および呼び出し規則について説明します。  テキストオブジェクトを表示するには、行コマンド D を使用します。「SYSEXT ユーティリティ機能の実行」の「 <a href="#">行コマンド</a> 」を参照してください。  テキストオブジェクトを検索するには、キーワードを指定します。「SYSEXT ユーティリティ機能の実行」の「 <a href="#">キーワード検索</a> 」を参照してください。

## SYSEXT の呼び出しと終了

このセクションでは、SYSEXT ユーティリティの呼び出しおよび終了手順について説明します。

### ▶手順 84.1. SYSEXT を呼び出すには

- 次のシステムコマンドを入力します。

```
SYSEXT
```

## SYSEXT ユーティリティ-Natural アプリケーションプログラミングインターフェイス

次の例のようなSYSEXT ユーティリティ画面が表示され、使用可能な Natural API がすべてリストされます。

```
13:11:42          ***** NATURAL SYSEXT UTILITY *****          2005-08-01
User SAG              - Menu -                               Library SYSEXT

Cmd  Source      Interface  Comment
Prod
_   USR0010P     USR0010N  Get 'SYSPROF' Information
NAT
_   USR0011P     USR0011N  Information about logical file
NAT
_   USR0020P     USR0020N  Read any error text from FNAT / FUSER
NAT
_   USR0040P     USR0040N  Get type of last error
NAT
_   USR0050P     USR0050N  Get 'SYSPROD' Information
NAT
_   USR0060P     USR0060N  Copy LFILE definition from 'FNAT' to 'FUSER'
NAT
_   USR0070P     USR0070N  Default Editor Profile 'SYSTEM'
NAT
_   USR0080P     USR0080N  Handle Type/Name of Editor Contents
NAT
_   USR0100P     USR0100N  Control LRECL
NVS
_   USR0120P     USR0120N  Read Natural Short Error Message
NAT
_   USR0210P     USR0210N  Save, cat or stow NATURAL object
NAT
_   USR0220P     USR0220N  Read Natural Long Error Message
NAT
_   USR0320P     USR0320N  Read User Short Error Message from FNAT or FUSER
NAT
_   USR0330P     USR0330N  Read Natural Object Directory
NAT

Keyword .. _____

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit      --      +      Canc
```

このリストは、プログラム例の名前（USR $nnnn$ P）によってソートされます。必要に応じて、PF8 キーを押してリストを下方にスクロールしたり、PF7 キーを押してリストの先頭に戻ったりできます。

▶ **手順 84.2. SYSEXT を終了するには**

- SYSEXT ユーティリティ画面で PF3 キーまたは PF12 キーを押します。
- または:
- コマンド行にピリオド（.）または EXIT を入力します。

## SYSEXT ユーティリティ画面

次の表に、SYSEXT ユーティリティ画面に表示されるフィールドおよび列とその説明を示します。

列	説明
<b>Cmd</b>	テキストオブジェクトまたはプログラム例で実行される行コマンドの入力フィールド。「SYSEXT ユーティリティ機能の実行」の「 <a href="#">行コマンド</a> 」を参照してください。
<b>Source</b>	API の呼び出しに使用できるプログラム例の名前（ソースオブジェクト USR $nnnn$ P）。
<b>Interface</b>	API サブプログラム USR $nnnn$ N の名前。
<b>Comment</b>	API の目的の概要。
<b>Prod</b>	API の影響を受ける Natural (NAT) または Natural アドオン製品の製品コード。  例：NAT = Natural、NDB = Natural for DB2、NVS = Natural for VSAM、PRD = Predict、RPC = Natural リモートプロシージャコール。
<b>Keyword</b>	APIに関連したキーワードの入力フィールド。「SYSEXT ユーティリティ機能の実行」の「 <a href="#">キーワード検索</a> 」を参照してください。

## SYSEXT ユーティリティ機能の実行

SYSEXT ユーティリティの機能を使用すると、テキストオブジェクトに対する操作、および SYSEXT ユーティリティ画面の API リストで参照されるプログラム例を実行できます。また、キーワード検索機能を使用して、API のリストを短くしたり、API に対して使用可能なすべてのキーワードをリストしたりすることもできます。

以下のセクションでは、次のトピックについて説明します。

- [行コマンド](#)

■ キーワード検索

行コマンド

行コマンドを使用すると、オブジェクト操作を実行できます。行コマンドは、目的の API の横の [Cmd] 列に入力します。この列に疑問符 (?) を入力すると、有効な行コマンドがリストされます。

SYSEXT ユーティリティ画面では、次の行コマンドを使用できます。

行コマンド	機能
E	プログラム例 USRnnnnP を編集します。
L	プログラム例 USRnnnnP をリストします。
R	プログラム例 USRnnnnP を実行します。
X	プログラム例 USRnnnnP を実行します。
D	対応する API の説明のテキストオブジェクト USRnnnnT をリストします。 API と API に関連するキーワードの目的、機能、および呼び出し規則について説明します。
K	指定されている API に関連するキーワードをリストします。
.	SYSEXT ユーティリティを終了します。

キーワード検索

キーワードは、現在のタスクに関連した Natural API の検索に役立ちます。キーワード検索機能を使用すると、キーワードで API をリストしたり、特定の API に関連したキーワードをリストしたりできます。

▶ 手順 84.3. キーワードで API をリストするには

■ 次のいずれかの方法を選択します。

■ キーワードを 1 つ入力します。

SYSEXT ユーティリティ画面の [Keyword] フィールドにキーワードを入力し、Enter キーを押します。

指定したキーワードを適用するすべての API が表示されます。

■ 有効なキーワードのリストからキーワードを 1 つ選択します。

1. SYSEXT ユーティリティ画面の [Keyword] フィールドにアスタリスク (\*) を入力し、Enter キーを押します。

次の例に示すように、[List Keywords] ウィンドウに、すべての API に使用可能なキーワードがすべてリストされます。

必要に応じて、PF8 キーを押してキーワードのリストを下方にスクロールしたり、PF7 キーを押してリストの先頭に戻ったりできます。

2. 目的のキーワードの横にある [M] 列に任意の文字を入力して、Enter キーを押します。  
または [Direct] フィールドに目的のキーワードを入力して、Enter キーを押します。

指定したキーワードを適用するすべての API が表示されます。

■ 有効なキーワードのリストから複数のキーワードを選択します。

1. SYSEXT ユーティリティ画面の [Keyword] フィールドにアスタリスク (\*) を入力し、Enter キーを押します。

次の例に示すように、[List Keywords] ウィンドウに、すべての API に使用可能なキーワードがすべてリストされます。

必要に応じて、PF8 キーを押してキーワードのリストを下方にスクロールしたり、PF7 キーを押してリストの先頭に戻ったりできます。

2. 目的のキーワードの横にある [M] 列にアスタリスク (\*) を入力して、Enter キーを押します。

次の例のような [Keyword] ウィンドウが表示され、選択した最初のキーワード（この例では \*LANGUAGE）が適用されるすべての API がリストされます。

```

11:08:58                ***** NATURAL SYSEXT UTILITY *****
2005-11-10
User SAG                - Menu -                Library
SYSEXT

Cm +-----List Keywords-----+ +-----Keyword *LANGUAGE-----+
Prod
_ !                               ! !                               !
  NAT
_ ! M Keyword                     ! ! Object                       !
  NAT
_ ! _ *DATX                        ! ! USR0020P                       !
  NAT
_ ! _ *ERROR-NR                    ! ! USR0120P                       !
  NAT
_ ! _ *ERROR-TA                    ! ! USR0320P                       !
  NAT
_ ! * *LANGUAGE                     ! ! USR1027P                       !
  NAT
_ ! _ *LEVEL                        ! ! USR1030P                       !
  NAT
_ ! _ *STEPLIB                      ! ! USR2034P                       !
  NAT

```

```

_ ! * *TIMESTMP                ! ! USR3320P                !
  NVS
_ ! _ *TIMX                      ! !                      !
  NAT
_ ! _ ABEND                      ! !                      !
  NAT
_ !                               ! !                      !
  NAT
_ ! Direct ... _____      ! !                      !
R NAT
_ !                               ! !                      !
  NAT
+-----+ +-----+
Keyword .. *_____

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help           Exit                --      +
Canc

```

3. Enter キーを押して、選択した次のキーワード（この例では \*TIMESTMP）の [Keyword] ウィンドウを開きます。
4. Enter キーまたは PF3 キーを押して、[List Keywords] ウィンドウに戻ります。
5. PF3 キーを押してすべてのウィンドウを終了し、SYSEXT ユーティリティ画面に戻ります。

▶手順 84.4. 特定のAPIに関連するキーワードをリストするには

- 目的の API の横に行コマンド K を入力します。次の例に示すように、指定した API の [Keywords] ウィンドウが表示され、この APIに関連するキーワードがすべてリストされます。

```

11:24:27          ***** NATURAL SYSEXT UTILITY *****          2005-11-10
User SAG              - Menu -                                Library SYSEXT

Cmd  Source      Interface  Comment                                     Prod
K   USR0010P    USR0010  +-Keywords for USR0010P--+                NAT
_   USR0011P    USR0011  !                                     ! file                NAT
_   USR0020P    USR0020  ! DBID                                  ! FNAT / FUSER        NAT
_   USR0040P    USR0040  ! ENVIRONMENT                            !                     NAT
_   USR0050P    USR0050  ! FILE                                    !                     NAT
_   USR0060P    USR0060  ! FNAT                                    ! m 'FNAT' to 'FUSER' NAT

```



```

_   USR0070P  USR0070 !   FNR                ! YSTEM'                NAT
_   USR0080P  USR0080 !   FUSER              ! r Contents            NAT
_   USR0100P  USR0100 !   SYSPROF           !                        NVS
_   USR0120P  USR0120 !                        ! Message              NAT
_   USR0210P  USR0210 !                        ! object               NAT
_   USR0220P  USR0220 !                        ! essage              NAT
_   USR0320P  USR0320 +-----+ sage from FNAT or FUSER NAT
_   USR0330P  USR0330N  Read Natural Object Directory    NAT

Keyword .. _____

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---;

```

必要に応じて、PF8 キーを押してリストを 1 ページ下方にスクロールします。リストの先頭に移動するには、PF7 キーを押します。

## Natural API の使用

ライブラリ SYSEXT に含まれている API を 1 つを使用する場合、次の選択肢があります。

1. ライブラリ SYSEXT をアプリケーションの `steplib` として定義します (推奨)。
2. 選択したサブプログラムをシステムファイル FNAT のライブラリ SYSTEM にコピーします (推奨)。
3. 選択したサブプログラムをアプリケーションの任意の `steplib` にコピーします。
4. 選択したサブプログラムをアプリケーションライブラリにコピーします。

選択肢 1 には、Natural インストールのアップグレード時に API に固有のアクティビティの必要がないという利点があります。

選択肢 2 には (選択肢 3 および 4 と比べて)、すべての API が 1 つのライブラリにあるという利点があります。Natural インストールをアップグレードする場合は、アップグレードする必要がある API のライブラリを 1 つ確認するだけで済みます。

## Natural API のリスト

次の SYSEXT ユーティリティで使用できる Natural アプリケーションプログラミングインターフェイス (API) のリストには、全般的な情報のみが示されています。API の現在のステータスについては、システムコマンド SYSEXT を実行します。

リストの列は、SYSEXT ユーティリティ画面に表示される列と対応しています。列の詳細については、「[SYSEXT ユーティリティ画面](#)」を参照してください。

Natural リモートプロシージャコール (RPC) に適用される API の使用の詳細については、『Natural リモートプロシージャコール』ドキュメントの「Natural RPC で使用するアプリケーションプログラミングインターフェイス」を参照してください。

インターフェイス	コメント	製品
USR0010N	SYSPROF 情報の取得	NAT
USR0011N	論理ファイルに関する情報	NAT
USR0020N	FNAT / FUSER からエラーテキストを読み取り	NAT
USR0040N	最後のエラーのタイプを取得	NAT
USR0050N	SYSPROD 情報の取得	NAT
USR0060N	LFILE 定義を FNAT から FUSER へコピー	NAT
USR0070P	デフォルトエディタプロファイル SYSTEM	NAT
USR0080N	エディタの内容のタイプ/名前を処理	NAT
USR0100N	LRECL の制御	NVS
USR0120N	Natural ショートエラーメッセージの読み取り	NAT
USR0210N	Natural オブジェクトの保存、カタログ、STOW	NAT
USR0220N	Natural ロングエラーメッセージの読み取り	NAT
USR0320N	FNAT または FUSER からユーザーショートエラーメッセージを読み取り	NAT
USR0330N	Natural オブジェクトディレクトリの読み取り	NAT
USR0340N	Natural バッファプールインターフェイス	NAT
USR0341N	Natural バッファプールインターフェイス	NAT
USR0350N	現在記録中のフラグの読み取り	NAT
USR0360N	ユーザーショートエラーメッセージの変更	NAT
USR0400N	検索された UPDATE の影響を受ける行数	NDB
USR0420N	FUSER からユーザーロングエラーメッセージを読み取り	NAT
USR0421N	FUSER 上のユーザーロングエラーメッセージの更新	NAT
USR0500N	ウィンドウのタイトルバーの文字列を表示	NAT
USR0600N	プログラムレベル情報の表示	NAT
USR0610N	DB エラー情報の表示	NAT

SYSEXT ユーティリティ - Natural アプリケーションプログラミングインター  
フェイス

インターフェイス	コメント	製品
USR0620N	文字列の変換	NAT
USR0622N	エラーカウンターのリセット	NAT
USR1002N	Natural 環境パラメータの保存と復元	NAT
USR1005N	一部の Natural システムパラメータに関する情報	NAT
USR1006N	スキップシーケンシャル処理のサポート	NVS
USR1007N	ワークファイル/出力ファイル割り当ての表示	NAT
USR1009N	*TIMESTMP を数値変数に変換	NAT
USR1011N	ワイルドカード/アスタリスクのチェック (ショート)	NAT
USR1012N	エラーのダイナミック部分 :1: の読み取り	NAT
USR1013N	現在の文字セットの表示	NAT
USR1014N	*** 行の計算機能 ***	NAT
USR1016N	コピーコードのエラーレベルの表示	NAT
USR1017N	CATALL コールを CATALL コントロールリストに追加	NAT
USR1018N	ダイナミック OPEN	NVS
USR1019N	SYSBUS 情報の取得	NAT
USR1020N	ユーザーショートエラーメッセージを FUSER に追加	NAT
USR1021N	ワイルドカード/アスタリスクのチェック (ロング)	NAT
USR1022N	データベースのタイプ	NAT
USR1023N	日時変数の変換	NAT
USR1024N	CATALL の結果の読み取り	NAT
USR1025N	複数の steplib の処理	NAT
USR1026N	RETURN 情報の表示	NAT
USR1027N	ユーザーショートエラーメッセージの検索	NAT
USR1028N	ビット/バイトの変換	NAT
USR1029N	Natural オブジェクトのタイプの取得	NAT
USR1030N	言語コードの変換	NAT
USR1031N	オブジェクト名のチェック	NAT
USR1032N	カタログされた Natural オブジェクトのリスト (タイプを併記)	NAT
USR1033N	カタログされた DDM の DBID/FNR の検索	NAT
USR1034N	NTTF ファイルテーブルの表示	NAT
USR1035N	Software AG エディタエンジンを介したオブジェクトのメンテナンス	NAT
USR1036N	Software AG エディタのユーザープロファイルのメンテナンス	NAT
USR1037N	Natural ABEND データに関する情報	NAT
USR1038N	現在のプラットフォームの特性の取得	NAT
USR1040N	UDB パラメータの取得または設定	NAT
USR1041N	エラートランザクション (*ERROR-TA) のインストール	NAT

SYSEXT ユーティリティ-Natural アプリケーションプログラミングインターフェイス

インターフェイス	コメント	製品
USR1042N	UPDATE コマンドの値の取得または設定	NAT
USR1043N	Adabas ダイレクトコールの実行	NAT
USR1047N	ファイル名をダイナミックに切り換え	NVS
USR1048N	PF キーラベルの変更	NAT
USR1050N	ワークファイル名の取得または設定	NAT
USR1051N	Predict の各種データ用インターフェイス	PRD
USR1054N	ライブラリのリスト	NAT
USR1055N	ライブラリ内のオブジェクトのリスト	NAT
USR1056N	FDIC ファイル上またはライブラリ内の DDM のリスト	NAT
USR1057N	Natural ソースを配列に読み取り	NAT
USR1058N	DDM ソースを配列に読み取り	NAT
USR1066N	Natural "Executing ..." メッセージの表示	NAT
USR1067N	ライブラリ名のチェック	NAT
USR1068N	DBMS コールの値の取得またはリセット	NAT
USR1070N	オペレータコマンドを Entire Net-Work に発行	NAT
USR1071N	RPC : TOKEN データの維持	RPC
USR1072N	保有セットのコマンド ID の取得	NAT
USR2001N	最後のエラーに関する情報の読み取り	NAT
USR2002P	ヘルプ機能のデフォルトテキスト文字列	NAT
USR2003P	MAINMENU のデフォルト設定	NAT
USR2004N	論理ファイルに関する情報	NAT
USR2005N	内部ファイル変換テーブルへのアクセス	NAT
USR2006N	詳細なメッセージ情報の取得	NAT
USR2007N	RPC デフォルトサーバー情報の設定/取得	RPC
USR2008N	VSAM/ISAM データセットのダイナミック OPEN	NVS
USR2009N	エラーのダイナミック部分 :1: の読み取り	NAT
USR2010N	DB エラー情報の表示	NAT
USR2011N	ワークファイル名の取得または設定	NAT
USR2012N	システム変数 *NET-USER の値の取得	NAT
USR2013N	SYSPROF 情報の取得	NAT
USR2014N	Software AG エディタエンジンを介したオブジェクトのメンテナンス	NAT
USR2015N	Natural RPC 用の EBCDIC または ASCII 変換テーブル	NAT
USR2016N	マッププロファイルの FNAT から FUSER へのコピー	NAT
USR2017N	FUSER からのマッププロファイル処理の有効化	NAT
USR2018N	Natural オブジェクトディレクトリの読み取り	NAT
USR2019N	Natural ソースのソースエリアへの保存/ソースエリアからの読み取り	NAT

SYSEXT ユーティリティ - Natural アプリケーションプログラミングインター  
フェイス

インターフェイス	コメント	製品
USR2020N	Adabas ダイレクトコールの実行	NAT
USR2021N	データセットのダイナミックアロケーション	NAT
USR2022N	保存データエリアへの GUID の挿入	NAT
USR2023N	データベースのタイプ (2 バイト)	NAT
USR2026N	TECH 情報の取得	NAT
USR2027N	セッション待機時間の定義	NAT
USR2028N	Natural バージョンの出力	NAT
USR2029N	ファイルのダイナミックアロケーション (BS2000/OSD)	NAT
USR2030N	エラーのダイナミック部分 :1;... の読み取り	NAT
USR2031N	SYSPROD 情報の取得	NAT
USR2032N	CLOSE CONVERSATION のコミットのサポート	RPC
USR2033N	PRD List Xref セットに関する情報	PRD
USR2034N	エラーテキストの読み取り (全体が対象)	NAT
USR2035N	SSL のサポート	RPC
USR2036N	*TIMESTAMP を数値変数に変換	NAT
USR2071N	クライアント側の EntireX Security のサポート	RPC
USR2072N	サーバー側の EntireX Security のサポート	RPC
USR2073N	RPC サーバーの ping または終了	RPC
USR2075N	EntireX Broker サービスの終了	RPC
USR3001N	ロールサーバーのディレクトリエントリのリスト	NAT
USR3002N	ロールサーバーのディレクトリエントリの削除	NAT
USR3005N	ドキュメンテーションオブジェクトの処理	PRD
USR3013N	SYSPROF 情報の取得	NAT
USR3025N	複数の steplib の処理	NAT
USR3320N	FNAT または FUSER からユーザーショートエラーメッセージを検索	NAT
USR4001N	Natural プロファイルパラメータ PROGRAM の設定  (USR6002N に対応していますが、メインフレームでのみ使用可能です)	NAT
USR4002N	現在のシステムの変数の取得	NAT
USR4003N	Natural スタック情報の取得 (英数字)	NAT
USR4004N	ダイナミック Natural プロファイルパラメータの取得	NAT
USR4005N	現在のキー設定をすべて読み取り	NAT
USR4006N	CICS 環境での Natural セッションの選択 (およびキャンセル)	NCI
USR4007N	プロファイルパラメータ SYNERR の現在の値の取得/設定	NAT
USR4008N	RPC 実行のライブラリの設定	RPC
USR4009N	EntireX のパラメータの設定	RPC
USR4010N	サーバーのランタイム設定の取得	RPC

SYSEXT ユーティリティ - Natural アプリケーションプログラミングインターフェイス

インターフェイス	コメント	製品
USR4011N	可変入力の A20 ハッシュ値の作成	NAT
USR4201N	データエリアソースの処理	NAT
USR4202N	コードページのテーブル情報の取得	NAT
USR4203N	Natural オブジェクトのロック、ロックのチェック、アンロック	NAT
USR4204N	CICS コンテナ名の設定または再設定	NCI
USR4206N	ライブラリ内のオブジェクトのリスト、およびディレクトリ情報を戻す	NAT
USR4207N	マシン特性を NATQVS が提供する方式で戻す	NAT
USR4208N	Natural リソースの読み取り／書き込み	NAT
USR4209N	サブルーチンのショートネームを戻す	NAT
USR4210N	Base64 の英数字／バイナリのバイト変換	NAT
USR4211N	DBCS 文字の取得	NAT
USR4212N	データエリアの分析	NAT
USR4213N	DBCS サポートのための文字列処理	NAT
USR4340N	バッファプールコンテンツまたはバッファプールキャッシュのリスト	NAT
USR4341N	Natural バッファプールのブラックリストの管理	NAT
USR6002N	一部の内部カウンタの現在の値を取得	NAT
USR6204N	Natural プロファイルパラメータ PROGRAM の設定  (USR4001N に対応していますが、すべてのプラットフォームで使用可能です)	NAT
USR6303N	Natural スタック情報の取得 (Unicode)	NAT
USR6304N	RPC 実行のための信頼できる状態の設定／取得	RPC
USR6305N	Reliable RPC メッセージのコミット／ロールバック	RPC
USR6306N	現在の EntireX Broker ユーザーの UOW のステータス	RPC

# 85 SYSMAIN ユーティリティ・オブジェクトメンテナンス

---

SYSMAIN ユーティリティは、コピー、移動、置換、削除などのオブジェクトメンテナンス機能を実行するために使用します。

- SYSMAIN の全般的な情報
- SYSMAIN の呼び出しと終了
- メニュー機能とコマンドの使用
- プログラミングオブジェクト処理
- デバッグ環境処理
- エラーメッセージ処理
- プロファイル処理
- ルール処理
- DL/I サブファイル処理
- DDM 処理
- ダイレクトコマンドのキーワードと変数
- SYSMAIN に発行される特殊コマンド
- 処理ステータスとエラー通知
- 管理者向けの特別な考慮事項





## 86 SYSMAIN の全般的な情報

---

■ SYSMAIN の基本的な機能 .....	508
■ オブジェクトタイプとストレージの場所 .....	509
■ 機能の概要 .....	509

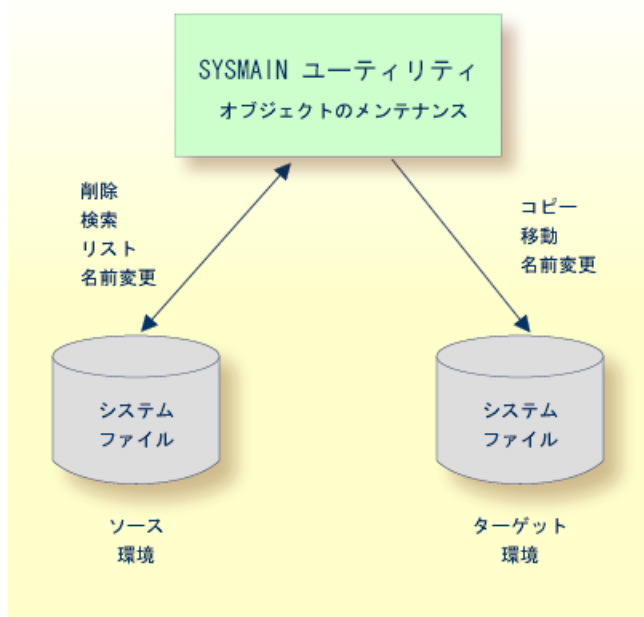
SYSMAIN ユーティリティは、Natural 環境または異なる環境で Natural オブジェクトをメンテナンスするために使用します。

このセクションでは、オブジェクトメンテナンス機能および SYSMAIN で処理できる Natural オブジェクトのタイプについて基本的な情報を提供します。

## SYSMAIN の基本的な機能

---

次の図は、SYSMAIN 機能を簡単な図で示したものです。



SYSMAIN ユーティリティでは、Natural オブジェクトをソース環境からターゲット環境にコピーまたは移動し、ソース環境で削除するなどのオブジェクト操作を実行します。名前の変更機能は、ソース環境とターゲット環境の両方で実行できます。機能の概要と機能が適用される Natural オブジェクトについては、次のセクションで説明します。SYSMAIN ユーティリティの機能は、オンラインおよびバッチモードで使用できます。

ソース環境またはターゲット環境は、Adabas データベース、または VSAM ファイルシステムに含まれる FNAT、FUSER、および FDIC システムファイルです。次のセクションで説明するように、FNAT または FUSER システムファイルの Natural オブジェクトはライブラリに含めることができます。

## オブジェクトタイプとストレージの場所

次の表に、SYSMAIN でメンテナンスできる Natural オブジェクトのタイプを示します。次の表に示すように、Natural オブジェクトの場所はオブジェクトタイプによって異なります。

オブジェクト/サブファイル	説明	場所
プログラミングオブジェクト	デバッグ環境とエラーメッセージを除く Natural ライブラリに格納されているすべての Natural オブジェクトタイプ。	FUSER および FNAT のライブラリ
デバッグ環境	オンラインプログラムテスト用のデバッグ環境。	FUSER のライブラリ
エラーメッセージ	ショートおよび拡張（ロング）Natural システムエラーメッセージとユーザー定義のエラーメッセージ。	FUSER および FNAT のライブラリ
プロファイル	エディタプロファイル、マッププロファイル、デバイスプロファイル、およびパラメータプロファイル。	FNAT
ルール	自動ルールおよびフリールール。	FDIC
DDM	データ定義モジュール。	FDIC
DL/I サブファイル	Natural NSB、NDB、および UDF。	FDIC

## 機能の概要

次の表に、SYSMAIN で使用できる機能を示します。表には、Natural オブジェクトの各タイプにどの機能が有効であるかも示します。各機能の詳細については、「メニュー機能とコマンドの使用」の「[機能の説明](#)」を参照してください。

機能	プログラミングオブジェクト	デバッグ環境	エラーメッセージ	プロファイル	ルール	DDM	DL/I サブファイル
<b>Copy</b> オブジェクトを1つのシステムファイルから他のシステムファイルにコピーします。	x	x	x	x	x	x	x
<b>Delete</b> オブジェクトをシステムファイルから削除します。	x	x	x	x	x	x	x
<b>Find</b> システムファイル内のオブジェクトを検索します。	x		x				
<b>List</b>	x	x	x	x	x	x	x

機能	プログラミングオブジェクト	デバッグ環境	エラーメッセージ	プロファイル	ルール	DDM	DL/サブファイル
システムファイル内のオブジェクトの範囲を表示します。							
<b>Move</b> オブジェクトを1つのシステムファイルから他のシステムファイルに転送します。	X	X	X	X	X	X	X
<b>Rename</b> オブジェクトに新しい名前を付け、任意で、新しいシステムファイルに転送します。	X	X	X	X	X		

## 87 SYSMAIN の呼び出しと終了

---

- オンラインまたはバッチでの SYSMAIN の呼び出し ..... 512
- アプリケーションプログラミングインターフェイスでの SYSMAIN の呼び出し ..... 513
- SYSMAIN の終了 ..... 514

このセクションでは、SYSMAINユーティリティをオンラインまたはバッチで、あるいはアプリケーションプログラミングインターフェイスを使用して呼び出し、終了する方法について説明します。

## オンラインまたはバッチでの SYSMAIN の呼び出し

次に、ダイレクトコマンド（オンラインまたはバッチ）あるいはメニュー機能を使用して SYSMAIN ユーティリティを呼び出す方法について説明します。

### ▶手順 87.1. SYSMAIN をオンラインで呼び出すには

- 任意のライブラリから、次の Natural システムコマンドを入力します。

```
SYSMAIN
```

または:

Natural メインメニューの [Maintenance and Transfer Utilities] メニューを呼び出し、[Transfer Objects to Other Libraries] を選択します。

次のような SYSMAIN ユーティリティのメインメニューが表示されます。

```

10:39:24          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG                - Main Menu -

      Code  Object                                Code  Function
      A    Programming Objects                    C    Copy
      D    Debug Environments                     D    Delete
      E    Error Message Texts                    F    Find
      P    Profiles                                L    List
      R    Rules                                    M    Move
      S    DL/I Subfiles                           R    Rename
      V    DDMS                                     ?    Help
      ?    Help                                    .    Exit
      .    Exit

Object Code .. A          Function Code .. _

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del Find List Move Ren
    
```

システム変数 \*LIBRARY-ID の現在の設定が SYSMAIN に渡され、プログラミングオブジェクトとデバッグ環境を処理するためのデフォルトのソースライブラリとして使用されます。

#### ▶手順 87.2. SYSMAIN をバッチで呼び出すには

- 次のダイレクトコマンドを使用します。

```
SYSMAIN
```

この後に 1 つ以上のコマンド文字列が続きます。「[ダイレクトコマンドの発行](#)」も参照してください。

## アプリケーションプログラミングインターフェイスでの SYSMAIN の呼び出し

MAINUSER は、一種のアプリケーションプログラミングインターフェイス (API) です。MAINUSER を使用すると、SYSMAIN を呼び出す通常の手順を実行することなく、任意のユーザー記述オブジェクト (サブルーチン、プログラム、サブプログラムなど) から直接 SYSMAIN 機能を実行できます。

SYSMAIN 機能の処理完了時にユーティリティは終了し、制御は、要求を発行したプログラム、サブプログラム、またはサブルーチンに戻されます。

MAINUSER は、SYSMAIN システムライブラリ内のサブプログラムタイプのカatalog化オブジェクトとして提供されます。MAINUSER は、オンラインモードでもバッチモードでも使用することができます。



**注意:** MAINUSER をユーザーライブラリに置くことはできません。このため、MAINUSER を FNAT または FUSER システムファイルの SYSTEM ライブラリ、あるいはアプリケーションの steplib である任意の SYS 接頭辞を持つライブラリにコピーする必要があります。

### MAINUSER の呼び出し

MAINUSER は、CALLNAT ステートメントおよびその関連するパラメータを使用して呼び出します (「*Natural* ステートメント」の「CALLNAT」も参照)。MAINUSER を SYSMAIN ライブラリ内から呼び出すことはできません。

#### ▶手順 87.3. MAINUSER を呼び出すには

- 次の構文要素を指定して CALLNAT ステートメントを発行します。

```
CALLNAT 'MAINUSER' command error message library
```

ここでの各変数値は次のパラメータを表します。

パラメータ	Natural データフォーマット/データ長	説明
<i>command</i>	A250	SYSMAIN で実行するダイレクトコマンド文字列。
<i>error</i>	N4	処理の最後に SYSMAIN によって発行される、処理の正常終了またはエラーを示すリターンコード。
<i>message</i>	A72	オンラインでのエラーに対応したメッセージ。
<i>library</i>	A8	SYSMAIN を含むライブラリ。指定しない場合、デフォルトは SYSMAIN です。

呼び出し可能なルーチンの例として、SYSMAIN ライブラリの MAINCALL プログラムがあります。

## SYSMAIN の終了

---

### ▶手順 87.4. SYSMAIN を終了するには

- 任意の SYSMAIN 画面のコマンド行に、次のいずれかのダイレクトコマンドを入力します。

```
.
```

(ピリオド)

または

```
END
```

または

```
QUIT
```

または:

必要に応じて、PF3 キー (終了) を繰り返し押します。

または:

バッチモードで、次のいずれかのダイレクトコマンドを使用します。

```
END
```

または

```
QUIT
```



「メニュー機能とコマンドの使用」の「[ダイレクトコマンドの発行](#)」も参照してください。



**重要:** 環境が正しくリセットされない場合があるため、SYSMAIN ユーティリティは端末コマンド %% で終了しないでください。



## 88      メニュー機能とコマンドの使用

---

■ メニュー機能の実行 .....	518
■ コマンドの実行 .....	521
■ 機能の説明 .....	523
■ 機能処理とレポート .....	530
■ SYSMAN オンラインヘルプ .....	535

SYSMAIN 機能は、メニューモードまたはコマンドモードで実行できます。

このセクションでは、SYSMAIN 機能を実行する場合に SYSMAIN メニュー、ダイレクトコマンド、および表示されるオプションを使用する方法について説明します。

## メニュー機能の実行

メニューモードでは、処理するオブジェクトタイプと実行する機能のコードを入力して、SYSMAIN 機能を実行します。ファンクションコードを入力する代わりに、対応する PF キーを使用できます。

### ▶手順 88.1. SYSMAIN メニュー機能を実行するには

- **SYSMAIN** ユーティリティのメインメニューの **[Object Code]** フィールドに、必要なオブジェクトに対応する 1 文字のコード（次の例では、**[Programming Objects]** の「A」）を入力し、**[Function Code]** フィールドに、必要な機能に対応する 1 文字のコード（次の例では、**[Copy]** の「C」）を入力します。ファンクションコードを入力する代わりに、機能に対応する PF キー（ここでは PF4）を使用できます。**PF キー**のリストも参照してください。

```
10:55:14          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG              - Main Menu -

      Code  Object                                Code  Function

      A    Programming ObjectsC    Copy
      D    Debug Environments        D    Delete
      E    Error Message Texts      F    Find
      P    Profiles                  L    List
      R    Rules                      M    Move
      S    DL/I Subfiles             R    Rename
      V    DDMs                      ?    Help
      ?    Help                      .    Exit
      .    Exit

Object Code .. A          Function Code .. C

Command ==>
Enter-PF1---PF2---PF3-PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del Find List Move Ren
```

次に示す [Copy Programming Objects] メニューのように、該当するオブジェクトタイプ固有のメニューが表示されます。

```

17:28:52          ***** NATURAL SYSMAN UTILITY *****          2005-08-10
User SAG          - Copy Programming Objects -

Code  Function

      A  Copy All/Individual Objects
      C  Copy only Cataloged Objects
      S  Copy only Saved Objects
      W  Copy only Stowed Objects
      ?  Help
      .  Exit

      Code ..... A          Sel. List ... Y
Object Name ..... *_____ Type ..... _____
                                Set Number .. __   XREF .. N
Source Library ... OLDLIB__ Database .... 10__   File .. 50__
Target Library ... NEWLIB__ Database .... 10__   File .. 60__
Options Replace ... N          Criteria .... N

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del Find List Move Ren Fsec Fdic Fnac

```

オブジェクトタイプ固有のメニューに表示されるフィールドは、実行する SYSMAIN 機能によって異なります。特定の機能に適用されないフィールドは、それぞれの画面に表示されません。

フィールドを使用して、オブジェクト選択条件と処理オプションを指定します。フィールドの詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連のセクションを参照してください。

### PF キーの使用

PF キーを使用して、SYSMAIN メニュー機能を実行できます。メニュー内で無効な PF キーは、そのメニューで表示されません。次の表に、PF キーの概要を示します。

PF キー	名前	機能
PF1	Help	現在のカーソル位置に応じてオンラインヘルプを表示します。  カーソルが <b>[Object Code]</b> フィールドまたは <b>[Function Code]</b> フィールドに置かれている場合、SYSMAIN 一般ヘルプが表示されます。  カーソルが別のフィールドにある場合、フィールド固有のヘルプが表示されます。 「 <a href="#">SYSMAIN オンラインヘルプ</a> 」も参照してください。
PF2	Menu	<b>SYSMAIN</b> ユーティリティのメインメニューを表示します。
PF3	Exit	前の画面に戻ります。 <b>SYSMAIN</b> ユーティリティのメインメニューで PF3 キーを押すと、SYSMAIN は終了します。
PF4	Copy	指定したオブジェクトの <b>[Copy]</b> 機能を実行します。
PF5	Del	指定したオブジェクトの <b>[Delete]</b> 機能を実行します。
PF6	Find	プログラミングオブジェクト、エラーメッセージ、およびルールにのみ適用されます。  指定したオブジェクトの <b>[Find]</b> 機能を実行します。
PF7	List	指定したオブジェクトの <b>[List]</b> 機能を実行します。
PF8	Move	指定したオブジェクトの <b>[Move]</b> 機能を実行します。
PF9	Ren	DDM および DL/I サブファイルには適用されません。  指定したオブジェクトの <b>[Rename]</b> 機能を実行します。
PF10	Fsec	Natural Security がインストールされている場合、FSEC システムファイルの Adabas セキュリティ情報を指定するためのセキュリティ画面を呼び出します。
PF11	Fdic	デバッグ環境、エラーメッセージ、およびプロファイルには適用されません。  プログラミングオブジェクト、ルール、および DDM にのみ適用されます。  FDIC システムファイルの Adabas セキュリティ情報を指定するためのセキュリティ画面を呼び出します。Predict がインストールされている場合、プログラミングオブジェクトに <b>[Copy]</b> 、 <b>[Delete]</b> 、 <b>[Move]</b> 、または <b>[Rename]</b> 機能を実行するときに Predict セットユーザーを指定することもできます。

PF キー	名前	機能
PF12	Fnat	ルール、DL/I サブファイル、および DDM には適用されません。  FNAT または FUSER システムファイル、あるいはその両方のシステムファイル情報を指定するためのセキュリティ画面を呼び出します。

## コマンドの実行

このセクションでは、ダイレクトコマンドを使用するか、または SYSMAIN ユーティリティ内からシステムコマンドを発行して、SYSMAIN 機能を実行する方法について説明します。

ダイレクトコマンドは、オンラインモードまたはバッチモードで SYSMAIN 機能を実行するために使用します。

バッチモードでは、処理されたオブジェクトのステータスを示すレポートが自動的に表示されます。このレポートはオンラインモードでも表示できます（「[オンラインレポートモード](#)」を参照）。

- [ダイレクトコマンドの発行](#)
- [SYSMAIN コマンド行の使用](#)

### ダイレクトコマンドの発行

ダイレクトコマンドは、パラメータを表すキーワードの文字列で構成されます。ピリオド (.) は、コマンドの終わりを示します。この文字がコマンド文字列内で検出されると、後続のデータはすべて無視されます。

ダイレクトコマンドの発行時に適用される構文の詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションを参照してください。各キーワードと変数の詳細については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。ダイレクトコマンドの例は、オブジェクトタイプ関連セクション、SYSMAIN オンラインヘルプ、および [Selection] 画面（「[選択処理](#)」の[画面例](#)を参照）に示されます。

#### ▶ 手順 88.2. ダイレクトコマンドをオンラインで発行するには

- 任意の Natural コマンドプロンプトで、次のコマンドを入力します。

```
SYSMAIN
```

続けてコマンド文字列を入力します。コマンド文字列の各キーワードは、デリミタの代わりに空白文字で区切ることができます。

または:

任意の SYSMAIN メニューのコマンド行に、ダイレクトコマンド文字列を入力します。コマンド文字列の各キーワードは、デリミタの代わりに空白文字で区切ることができます。

または:

Natural オブジェクト内から、ダイレクトコマンド文字列をパラメータとして MAINUSER サブプログラムを呼び出します。コマンド文字列の各キーワードは、デリミタの代わりに空白文字で区切ることができます。

### ▶手順 88.3. ダイレクトコマンドをバッチで呼び出すには

- 1 SYSMAIN コマンドを使用して、次のいずれかの方法でコマンド文字列を指定します。
  1. SYSMAIN コマンドの同じ入力行にコマンド文字列を続けます。コマンド文字列の各キーワードは、デリミタの代わりに空白文字で区切ることができます。
  2. SYSMAIN コマンドの次の入力行にコマンド文字列を続けます。コマンド文字列の各キーワードは、デリミタで区切る必要があります。空白文字で区切ることはできません。

ダイレクトコマンド文字列が1行よりも長い場合は、文字 CF (『パラメータリファレンス』ドキュメントの「CF - 端末コマンドの文字」に記載されている CF セッションパラメータも参照) を行の末尾に置き、次の行にダイレクトコマンドを続ける必要があります。

- 2 SYSMAIN コマンドの後に他の Natural コマンドを実行する場合は、まずダイレクトコマンド END または QUIT を使用して、SYSMAIN コマンドを終了する必要があります。

### SYSMAIN コマンド行の使用

任意の SYSMAIN メニューのコマンド行に、次のいずれかのコマンドを入力できます。

- SYSMAIN 機能を実行するためのダイレクトコマンド。
- SYSMAIN ユーティリティに対する特殊コマンド (「[SYSMAIN に発行される特殊コマンド](#)」を参照)。
- システムコマンド。コマンドがシステムコマンドとして一意に識別されない場合、先頭に2つのスラッシュ (//) を置く必要があります。「MAINEX05 - ダイレクトコマンド検証用のユーザー出口ルーチン」の「[注意](#)」も参照してください。



## 機能の説明

SYSMAIN メニューで提供される機能またはダイレクトコマンドとして提供される機能については、次のセクションで説明します。メニューで提供される機能ごとに、同じ名前の対応するダイレクトコマンドがあります。例外：[**Help**]

ダイレクトコマンドの使用時に適用される構文の詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションを参照してください。

SYSMAIN ユーティリティに発行できる特殊コマンドの詳細については、「[SYSMAIN に発行される特殊コマンド](#)」を参照してください。

機能/ コマ ンド	説明
Copy	<p>Natural オブジェクトをソース環境からターゲット環境にコピーします。オブジェクトはソース環境で変更されないまま維持されます。</p> <p>コピーするオブジェクトと同じ名前のオブジェクト（エラーメッセージの場合は同じ番号）がターゲット環境にすでに含まれている場合、指定したオブジェクトはコピーされません。</p> <p>置換オプション（「<a href="#">置換オプションの使用</a>」を参照）を使用して、ターゲット環境内のオブジェクトを上書きできます。</p>
Delete	<p>Natural オブジェクトをソース環境から削除します。オンライン自動処理中は（「<a href="#">機能処理とレポート</a>」を参照）、確認ウィンドウが開き、機能を続行または終了するオプションが表示されます。</p> <p>エラーメッセージでは、Natural Security がインストールされている場合、削除機能をシステムエラーメッセージに対して使用できません。システムエラーメッセージを削除しようとすると、「4897 : エラーアプリケーションの指定に誤りがあります。」という Natural システムエラーメッセージが出力されます。</p>
Find	<p>プログラミングオブジェクトとエラーメッセージにのみ適用されます。</p> <p>ソース環境で1つ以上のプログラミングオブジェクトまたはエラーメッセージを検索します。</p> <p>オンライン処理中、現在スキャンされているライブラリを示すウィンドウが表示されます。</p> <p>メニューモードの検索機能には、ライブラリをリストして選択したり、選択リストに表示されるオブジェクトの数を減らしたりするオプションもあります。それぞれ「<a href="#">ライブラリのリストと選択</a>」または「<a href="#">選択リストを短くするには</a>」（「<a href="#">選択リストの使用</a>」）を参照してください。</p>
List	<p>ソース環境内の Natural オブジェクトの範囲を表示します。</p> <p>プログラミングオブジェクトの場合、デバッグ環境とエラーメッセージです。メニューモードのリスト機能には、ライブラリをリストして選択したり、選択リストに表示されるオブジェクトの数を減らしたりするオプションもあります。それぞれ「<a href="#">ライブラリのリストと選択</a>」または「<a href="#">選択リストを短くするには</a>」（「<a href="#">選択リストの使用</a>」）を参照してください。</p>

機能/ コマン ド	説明
	<p>プログラミングオブジェクトの場合、バッチモードでは、ダイレクトコマンド LISTLIB を使用してライブラリ名のリストを取得できます。</p>
<b>Move</b>	<p>Natural オブジェクトをソース環境からターゲット環境へ転送します。オブジェクトはソース環境から削除され、ターゲット環境に追加されます。移動するオブジェクトと同じ名前のオブジェクト（エラーメッセージの場合は同じ番号）がターゲット環境にすでに含まれている場合、指定したオブジェクトは移動されません。</p> <p>オンライン自動処理中は、確認ウィンドウが開き、機能を続行または終了するオプションが表示されます。</p> <p>置換オプション（「<a href="#">置換オプションの使用</a>」を参照）を使用して、ターゲット環境内のオブジェクトを上書きできます。</p>
<b>Rename</b>	<p>DL/I サブファイルおよび DDM には適用されません。</p> <p>次のいずれかのオプションを使用して、オブジェクトに新しい名前（エラーメッセージの場合は、新しい番号）を付けます。</p> <ol style="list-style-type: none"> <li>1. ソース環境内のオブジェクトの名前を変更します。</li> <li>2. オブジェクトの名前を変更し、別の環境、つまり、ターゲット環境に転送します。ソース環境とターゲット環境の両方で同時にオブジェクトの名前を変更することはできません。名前の変更機能は、環境ごとに個別に実行する必要があります。</li> </ol> <p>名前の変更機能では、ソース環境内の元のオブジェクトが削除されます。このため、元のオブジェクトを保持するオプションを含むプロンプトが表示されます。元のオブジェクトを保持することを選択すると、オブジェクトは削除されません。</p> <p>名前を変更するオブジェクトと同じ名前のオブジェクト（エラーメッセージの場合は同じ番号）がターゲット環境にすでに含まれている場合、指定したオブジェクトの名前は変更されません。置換オプション（「<a href="#">置換オプションの使用</a>」を参照）を使用して、ターゲット環境内のオブジェクトを上書きできます。</p> <p>自動処理を使用して名前を変更できるのは、1つのプログラミングオブジェクト、環境、プロファイル、またはルールのみです。プログラミングオブジェクト、環境、プロファイル、またはルールの範囲の名前を変更する場合は、<a href="#">選択処理</a>（下記参照）を使用する必要があります。</p> <p>エラーメッセージの場合、1つの（ソース）ライブラリ内のエラーメッセージ範囲の番号を変更するときは、範囲の値が重複しないようにしてください。例えば、エラー番号1～6を新しいエラー番号5～10として変更することはできません。エラーメッセージの範囲は、<a href="#">自動処理</a>（下記参照）を使用して変更できます。広範囲のエラーメッセージを処理する場合、メッセージの処理に大きなリソースが必要となることがあります。このような場合には、バッチモード処理をお勧めします。</p>
<b>Help</b>	<p>SYSMAIN に関するヘルプ情報を表示します（「<a href="#">SYSMAIN オンラインヘルプ</a>」を参照）。</p> <p>ヘルプ機能は、SYSMAIN メニューでのみ使用できます。対応するダイレクトコマンドはありません。</p>

機能/ コマンド	説明
Exit	SYSMAIN ユーティリティを終了します。  PF3 キー (Exit) をSYSMAIN ユーティリティのメインメニューで押した場合も、SYSMAIN が終了します。

このセクションでは、次のトピックについて説明します。

- 置換オプションの使用
- ライブラリのリストと選択

### 置換オプションの使用

コピー、移動、または名前を変更するオブジェクトと同じ名前のオブジェクトがターゲット環境にすでに含まれている場合、指定したオブジェクトは処理されず、処理は次のオブジェクトに進みます。置換オプションを使用して、このデフォルトの機能を上書きし、ターゲット環境のオブジェクトを上書きすることができます。

プログラミングオブジェクトを置き換えた場合、そのプログラミングオブジェクトは Natural バッファプールからも削除されます。Predict がインストールされている場合は、既存のクロスリファレンスレコードもすべて削除されます。

#### ▶手順 88.4. 置換オプションをコマンドモードでアクティブにするには

- コマンド文字列で、キーワード REPLACE を指定します。

#### ▶手順 88.5. 置換オプションをメニューモードでアクティブにするには

- 1 SYSMAIN メニューの [Replace] フィールドで、「Y」を入力します。
- 2 機能を実行します。[Sel. List] (選択リスト) オプションが N に設定されている場合、ウィンドウが表示され、実行する前にすべての置換オプションを確認するかどうかを選択できます。各置き換えを確認するには「Y」を入力し、確認しないで処理を続行するには Enter キーを押します。
- 3 「Y」を入力して置き換えを確認する場合、置き換えるオブジェクトごとにウィンドウが表示され、次のいずれかの文字を入力できます。
  - Y の場合、ウィンドウに示されたオブジェクトが置き換えられます。
  - N の場合、ウィンドウに示されたオブジェクトは置き換えられません (デフォルト設定)。
  - ピリオド (.) によって、機能処理が終了します。または、PF3 キーを使用することもできます。

## ライブラリのリストと選択

このオプションは、プログラミングオブジェクト、デバッグ環境、およびエラーメッセージにのみ適用されます。

リスト機能を使用する場合、指定したオブジェクトを含むライブラリの選択リストを呼び出すことができます。指定したオブジェクトが複数のライブラリに含まれている場合、選択リストの呼び出しは検索機能にも適用されます。

### ▶手順 88.6. プログラミングオブジェクトのライブラリのリストを呼び出すには

- **[List]** メニューの **[Library]** フィールドにライブラリ名の範囲を入力し（「**名前の範囲の指定**」も参照）、**[Name]** フィールドにオブジェクト名または名前の範囲を入力します。

**[Library]** フィールドにアスタリスク (\*) を入力した場合（リスト機能のみ）、指定したシステムファイル内で使用可能なすべてのライブラリのリストが表示されます。

次に示す **[Library Selection]** 画面の例では、LIB で始まる名前を持つすべてのライブラリが選択されます。

```

21:23:03          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          -      Library Selection      -

LIST ALL * IN LIB* WHERE DBID 10 FNR 32

  C Library  S/C    C Library  S/C    C Library  S/C    C Library  S/C
  - - - - -  - - -  - - - - -  - - -  - - - - -  - - - - -  - - -
  _ LIB1     S/C    _ LIB2     S/C    _ LIB3     C      _ LIB4     S/C
  _ LIB5     S      _ LIB6     S/C    _ LIB7     S/C    _ LIB8     S/C
  _ LIB9     S/C    _ LIB10    S/C    _ LIB11    S/C    _ LIB12    S/C
  _ LIB13    S      _ LIB14    S/C    _ LIB15    C      _ LIB16    S/C
  _ LIB17    S/C    _ LIB18    S/C    _ LIB19    S/C    _ LIB20    S/C

                                           Object Start Value: *

Enter options (above), or '?' (Help) or '.' (Exit): _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del Find List Move Ren Canc
    
```

〔Library〕列には、指定した範囲に一致するすべてのライブラリが表示されます。〔S/C〕列は、保存（ソース）オブジェクトまたはカタログ化オブジェクト、あるいはその両方がライブラリに含まれるかどうかを示します。

〔Library Selection〕画面で使用可能な行コマンドのリストの詳細については、「[プログラミングオブジェクトの選択リスト](#)」を参照してください。

検索機能を使用するとき、指定したオブジェクトが複数のライブラリにある場合は、〔Library Selection〕画面が表示されます。

〔Library Selection〕画面は、PF3 キー (Quit) を押して〔Find Selection〕または〔List Selection〕画面を終了した場合にも表示されます。

#### ▶手順 88.7. デバッグ環境のライブラリのリストを呼び出すには

- 〔Debug Environments〕メニューでの操作：

〔Code〕フィールドに「L」を入力し、〔Environment Name〕フィールドにデバッグ環境の名前または名前の範囲を入力し、〔Source Library〕フィールドにライブラリ名の範囲を入力します（「[名前の範囲の指定](#)」を参照）。

〔Environment Name〕と〔Source Library〕にアスタリスク（\*）を入力した場合、指定したシステムファイル内のデバッグ環境を含むすべてのライブラリのリストが表示されます。

次に示す〔Library Selection〕画面の例では、L で始まる名前を持つすべてのライブラリが選択されます。

```

09:55:08          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          -   Library Selection   -
LIST DEBUG * IN LIB* FROM DBID 10 FNR 32

C Library          C Library          C Library          C Library
- - - - -          - - - - -          - - - - -          - - - - -
_ LIB1             _ LIB5             _ LIB9             _ LIB11
_ LIB20

Object Start Value ... *

```

```

Enter options (above), or '?' (Help) or '.' (Exit): _
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Menu  Exit  Copy  Del   Find  List  Move  Ren                Canc
    
```

【Library】列には、指定したデバッグ環境を含むすべてのライブラリが表示されます。  
 【Library Selection】画面で使用可能な行コマンドのリストの詳細については、「[デバッグ環境の選択リスト](#)」を参照してください。

▶手順 88.8. エラーメッセージのライブラリのリストを呼び出すには

■ 次のいずれかの方法を選択します。

1. 【List】メニューの【No. From】フィールドにエラーメッセージ番号（次の例では1）を入力するか、または番号の範囲を指定（開始番号を【No. From】フィールドに、終了番号を【No. To】フィールドに入力）します。

【Library】フィールドに、ライブラリ名の範囲を入力します（「[名前](#)の範囲の指定」も参照）。

次のような【Library Help】ウィンドウが表示されます。

```

14:02:47          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          - List Error Message Texts -

                +-----+
                ! --- Library Help --- !
Code  Fun !      - Source -      !
                !                      !
      A  Lis !    1  system messages  !
      E  Lis !    2  LIB1              !
      S  Lis !    3  LIB2              !
      ?  Hel !    4  LIB3              !
      .  Exi !    5  LIB4              !
                !    6  LIB5              !
Code ..... A    !    7  LIB6              !
    
```

```

!      8  LIB7      !
Error  No. From .. 1__ !
Source Library ... LIB*_ !      ! 0__ FNR .. 32__
!      Enter selection or      !
!      '.' to Exit: 8_      !
+-----+
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del  Find List Move Ren  Fsec      Fnat

```

ウィンドウで、必要なライブラリに対応する番号を入力してライブラリを選択します。上記の例では、「8」を入力して LIB7 を選択しています。正しい FNAT システムファイルを指定した場合、システムエラーメッセージも選択できます。必要に応じて、Enter キーを押してリストの最後までスクロールします。

2. **[Find]** メニューの **[Number]** フィールドにエラーメッセージ番号を入力し、**[Library]** フィールドにライブラリ名の範囲を指定します（「[名前](#)の範囲の指定」も参照）。

指定したライブラリの範囲（ここでは「LIB\*」）について次のような **[Find Selection]** 画面が表示されます。

```

19:09:34          ***** NATURAL SYSMAN UTILITY *****          2005-08-10
User SAG          - Find Selection -

FIND ERR 1 TYPE A FROM LIB* LANG * WHERE DBID 10 FNR 32

C  Library      Error Message Text(s) for Error number: 1      Type
-  - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
_  LIB1         Short message of Error 1: wrong input value.      S
_  LIB2         Short message of Error 1: incorrect syntax.      S
_  LIB3         Short message of Error 1: undefined variable.  S

```

```

_ LIB4      Short message of Error 1: timeout error.          S
_ LIB5      Short message of Error 1: initialization failed.
S/E
_ LIB6      Short message of Error 1: invalid format.
S/E
_ LIB7      Short message of Error 1: wrong password.
S/E
_ LIB8      Short message of Error 1: input missing.
S/E
_ LIB9      Short message of Error 1: undefined keyword.
S/E
_ LIB10     Short message of Error 1: invalid command.        S

Enter options (above), or '?' (Help) or '.' (Exit): __

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del          List Move Ren          Canc
```

[Library] 列には、指定したエラーメッセージ（ここでは 1）を含むすべてのライブラリが表示されます。[Type] 列は、指定したオブジェクトにショート (S) エラーメッセージおよび拡張 (E) エラーメッセージがあるかどうかを示します。

[Find Selection] 画面で使用可能な行コマンドの詳細については、「[エラーメッセージの選択リスト](#)」を参照してください。

## 機能処理とレポート

SYSMAIN 機能を実行する場合、選択処理と自動処理の 2 種類の処理を使用できます。

このセクションでは、2種類の機能処理、および機能処理に関するステータスレポートを取得する方法について説明します。

- [自動処理](#)
- [選択処理](#)
- [選択リストの使用](#)



## ■ オンラインレポートモード

### 自動処理

自動処理は、コマンドモードでオンライン操作する場合のデフォルトタイプの処理です。バッチモードでは、自動処理は標準タイプの処理です。

自動処理は、中間の選択リストを表示せずにオブジェクトを処理するオンラインまたはバッチの機能です。この場合、機能の選択後に端末 I/O はほとんど、またはまったく必要ありません。

自動処理をオンラインで使用した場合、個々のオブジェクトのステータスは表示されませんが、処理の完了時に該当するメッセージが表示されます。次のいずれかの Natural システムエラーメッセージが表示された場合、指定したオブジェクトの一部またはすべてが処理されていません。

4867：この選択条件では何も見つかりません。

4810：選択条件により全データが拒否されました。

4893：処理が正常終了しましたが、一部のデータは処理されませんでした。

エラー NAT4810 の拡張（ロング）メッセージテキストでは、オブジェクトが処理されなかった理由が表示されます。オブジェクトが処理されない理由については、「[処理ステータスとエラー通知](#)」セクションでも説明しています。

バッチモードまたは選択処理は、各オブジェクトの処理後にステータスを確認する必要がある場合に使用します。

オンラインで処理する場合、必要に応じて、すべてのオブジェクトの処理ステータスを含むバッチレポートを取得できます（「[オンラインレポートモード](#)」を参照）。

### 選択処理

選択処理は、メニューモードで操作する場合のデフォルトタイプの処理です。

選択処理は、指定した選択条件に一致するすべてのオブジェクトのリストを表示するオンライン機能です。選択リストには次のオプションがあります。

- さらに処理する 1 つまたは複数のオブジェクトを選択します。
- オブジェクトを処理する前に、追加機能（ソースコードの表示など）を実行します（「[選択リストの使用](#)」を参照）。
- 各オブジェクトの処理ステータスに関する個々のメッセージを取得します（Moved や Copied など）。
- SYSMAN メニューに入力した入力値に対応するダイレクトコマンドを表示します（次の[選択画面の例](#)を参照）。このオプションは、メニュー機能の代わりにダイレクトコマンドを使用する場合に役立つことがあります。

## 選択リストの使用

このセクションでは、1つまたは複数のオブジェクトの処理を続行するために選択リストを呼び出す方法について説明します。また、このセクションでは、リスト機能または検索機能を使用する場合に選択リストを短くする方法についても説明します。このオプションは、大きいライブラリで表示するオブジェクトの数を制限する場合に役立ちます。

### ▶手順 88.9. 選択リストをオンラインで使用するには

- 1 メニューモードの場合、SYSMAIN メニューの **[Sel.List]** フィールドに「Y」（デフォルト設定）を入力して、選択処理をアクティブにします。「N」を入力すると、選択処理は非アクティブになります。

または:

コマンドモードの場合、ダイレクトコマンドの *with-clause* にキーワード **HELP** を含めるか、またはオブジェクト名の直後に疑問符 (?) を入力します (『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションの構文図も参照)。

選択処理をアクティブにすると、次のような **[Selection]** 画面が表示されます。

```

11:05:22          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          -      Copy      Selection      -

COPY ALL * WITH XREF N FROM OLDLIB WHERE DBID 10 FNR 50 TO NEWLIB WHERE DBID
10 FNR 60

C  Name      Type   S/C  Message      C  Name      Type   S/C  Message
-  - - - - -  - - - - -  - - - - -  - - - - -  -  - - - - -  - - - - -  - - - - -
_  AA        Copycd S/C          _  G0000001 Global S
_  G0000002 Global S/C          _  G0000003 Global S/C
_  L0000001 Local S/C          _  L0000002 Local S/C
_  L0000003 Local S          _  M0000001 Map S
_  M0000002 Map S/C          _  M0000003 Map S/C
_  P0000001 Progrm S          _  P0000002 Progrm C
_  P0000003 Progrm S/C       _  P0000004 Progrm S/C
_  P0000005 Progrm S        _  P0000006 Progrm S/C
_  P0000007 Progrm S/C     _  S0000001 Subpgm S/C
_  S0000002 Subpgm S/C     _  S0000003 Subpgm S
_  T0000001 Text S          _  T0000002 Text

                                           Listed Library: OLDLIB

Enter options (above), or '?' (Help) or '.' (Exit) _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Menu Exit Copy Del Find List Move Ren Canc
    
```

リストには指定したオブジェクトの名前とタイプが含まれ、保存オブジェクト (s) およびカタログ化オブジェクト (c) が存在するかどうかを示されます。

上記の例で強調表示されている選択リスト上部の行には、メニューフィールドに入力した入力値に対応する SYSMAIN ダイレクトコマンドが表示されます。

このコマンドは、コマンドモードで作業する場合に適用されるコマンド構文に対応します。ただし、SYSMAIN ドキュメントのオブジェクトタイプ関連セクションの構文図に示されているように、一部のキーワードはオプションです。

- 2 処理を続行する 1 つ以上のオブジェクトを選択します。目的のオブジェクトの横の [C] (コマンド) 列に、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションに記載されている行コマンドのいずれかを入力します。
- 3 Enter キーを押して、1 つ以上の行コマンドを実行します。

行コマンドは、指定したオブジェクト名のアルファベット順に実行されますが、削除などの SYSMAIN メンテナンス機能を実行するコマンドは、常に最後に実行されます。

ソースコードの表示などの SYSMAIN メンテナンス機能がいずれのオブジェクトでも実行されなかった場合は、任意のオブジェクトに行コマンドを再度入力できます。ただし、SYSMAIN メンテナンス機能がいずれかのオブジェクトに実行されると、[C] 列のフィールドは入力できなくなります。

行コマンドが実行されると、次の例に示すように、処理されたオブジェクトのステータスが [Message] 列に表示されます。

```

11:05:22          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          -      Copy      Selection      -

COPY ALL * WITH XREF N FROM OLDLIB WHERE DBID 10 FNR 50 TO NEWLIB WHERE DBID
10 FNR 60

C  Name      Type  S/C  Message      C  Name      Type  S/C  Message
-  - - - - -  - - - - -  - - - - -  - - - - -  -  - - - - -  - - - - -  - - - - -
_  AA        Copycd S/C  Copied      _  G0000001  Global S   Copied
_  G0000002  Global S/C  Copied      _  G0000003  Global S/C  Copied
_  L0000001  Local  S/C  Copied      _  L0000002  Local  S/C  Copied
_  L0000003  Local  S    Replaced    _  M0000001  Map    S    Replaced
_  M0000002  Map    S/C  Replaced    _  M0000003  Map    S/C  Replaced
_  P0000001  Progrm S
_  P0000003  Progrm S/C
_  P0000005  Progrm S
_  P0000007  Progrm S/C
_  S0000002  Subpgm S/C
_  T0000001  Text   S
_                                     _  T0000002  Text

Listed Library: OLDLIB

```

```
Enter options (above), or '?' (Help) or '.' (Exit) _
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Help Menu Exit Copy Del Find List Move Ren Canc
```

表示されるステータスメッセージのリストについては、「[処理ステータスとエラー通知](#)」セクションの「[ステータスメッセージ](#)」を参照してください。

### ▶手順 88.10. 選択リストを短くするには

- **[List Selection]** 画面または **[Find Selection]** 画面の **[Reposition to]** フィールドに、オブジェクトの名前を入力するか、または名前の範囲を指定して（「[名前の範囲の指定](#)」を参照）、Enter キーを押します。

名前を指定した場合、リストは指定した名前から始まります。名前の範囲を指定した場合、リストにはその範囲内のオブジェクトのみが表示されます。

**[Reposition to]** オプションは、オンラインモードでのみ有効です。このオプションはリスト機能の選択条件ではありません。

## オンラインレポートモード

オンラインレポートモードを使用すると、オンラインで選択リストではなく、SYSMAIN バッチレポートを取得できます。オンラインバッチレポートには、SYSMAIN 機能によって影響を受けるすべてのオブジェクトが表示され、それらの各オブジェクトに対して実行されるアクションが示されます。

### ▶手順 88.11. バッチレポートをオンラインで使用するには

- 1 SYSMAIN メニューの **[Sel. List]**（選択リスト）フィールドに「N」を入力して自動処理をアクティブにします。
- 2 コマンド行で、次のいずれかのダイレクトコマンドを入力します。

```
BATCH
```

または

```
BAT
```

バッチモードがアクティブにされたことを確認するメッセージが表示されます。

- 3 必要に応じて、次の端末コマンドを入力して、レポートのハードコピーを取得できます

```
%H
```

（『[端末コマンド](#)』ドキュメントの「[%H-ハードコピー出力](#)」も参照）。

- 4 SYSMAIN 機能を実行します。SYSMAIN では、バッチモードの場合と同様に機能が処理されます。このため、次の例に示すように、各アクションの結果のみがレポートタイプ形式で表示されます。

```

10:50:30          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG          - Copy      Function -

COPY ALL * WITH XREF N FROM TESTLIB1 WHERE DBID 10 FNR 30 TO TESTLIB2 WHERE
DBID 10 FNR 40

Saved      Progrm TEST1      has now been Copied
Saved      Progrm TEST2      has now been Copied
Saved      Progrm TEST3      has now been Copied
Stowed     Progrm TEST4      has now been Copied
Stowed     Progrm TEST5      has now been Copied
Stowed     Progrm TEST6      has now been Copied
Stowed     Progrm TEST7      has now been Copied
Saved      Record TEST8      has now been Copied
Saved      Text   TEST9      has now been Copied
Cataloged  Progrm TEST10     has now been Copied
Saved      Progrm TEST11     has now been Copied
Stowed     Progrm TEST12     has now been Copied
MORE
Stowed     Progrm TEST13     has now been Copied
Cataloged  Progrm TEST14     has now been Copied
Stowed     Progrm TEST15     has now been Copied

```

- 5 必要に応じて、MORE プロンプトでシステムコマンドか端末コマンドを入力して、機能処理を中断できます。中断しない場合、リストの最後に達するまで Enter キーを押して SYSMAIN メニューに戻り、機能が正常に実行されたことを示すメッセージを確認します。

## SYSMAIN オンラインヘルプ

SYSMAIN オンラインヘルプ機能では、ダイレクトコマンド構文の詳細な説明とダイレクトコマンドの例を含む、SYSMAIN で利用できるすべての機能に関する情報が提供されます。

ヘルプ機能に加えて、SYSMAIN では、SYSMAIN 画面で使用できるすべての入力フィールドに関する個々の情報が提供されます。

▶手順 88.12. SYSMAIN ヘルプトピックを呼び出すには

- 1 SYSMAIN ユーティリティのメインメニューで、 [Object Code] フィールドまたは [Function Code] フィールドにカーソルを置き、PF1 キー (Help) を押すか、または疑問符 (?) を入力します。

次のようなヘルプトピックのリストが示されたSYSMAIN ユーティリティの [Help Menu] が表示されます。

```

15:54:28          ***** NATURAL SYSMAIN UTILITY *****          2005-08-10
User SAG              - Help Menu -
                                     Help Name SHT-0001

Code  Topic
   A  SYSMAIN General Overview
   C  Command Mode
   E  Environment Definition
   F  Functions / Commands
   S  Security Environment
   .  Exit

Code ... _
Select a function code.
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Menu  Exit                                     Canc
    
```

- 2 [Code] フィールドに、目的のヘルプトピックに対応する 1 文字のコードを入力します。
- 3 Enter キーを押します。

選択したヘルプトピックに関する情報が示された結果画面が表示されるか、または検索の絞り込みに役立つより詳細なヘルプトピックが示された別のメニューが表示されます。

▶手順 88.13. 入力フィールドでヘルプを呼び出すには

- 問題のフィールドにカーソルを置き、PF1 キー (Help) を押すか、または疑問符 (?) を入力し、Enter キーを押します。

フィールド固有の説明が表示されたウィンドウが開き、該当する場合は、有効な入力値のリストが表示されます。





# 89      プログラミングオブジェクト処理

---

▪ [Programming Objects] メニューのフィールド .....	540
▪ プロファイルパラメータ RECAT の使用 .....	543
▪ プログラミングオブジェクトの選択リスト .....	543
▪ XRef について .....	548
▪ 追加条件の指定 .....	550
▪ プログラミングオブジェクトのダイレクトコマンド構文 .....	551

すべての SYSMAIN 機能をプログラミングオブジェクトに対して実行できます。SYSMAIN でメンテナンスできるプログラミングオブジェクトには、次のタイプの Natural オブジェクトがあります。プログラム、サブプログラム、サブルーチン、コピーコード、ヘルプルーチン、マップ、ローカルデータエリア、グローバルデータエリア、パラメータデータエリア、クラス、テキスト、記録、Natural コマンドプロセッサ、ダイアログ、ファンクション、ISPF マクロ、レポート、アダプタ、およびリソース。

プログラミングオブジェクトは、含まれているライブラリの名前によってシステムファイルに格納されます。ライブラリが SYS で始まる場合（ライブラリ SYSTEM を除く）、オブジェクトは FNAT システムファイルに格納されます。それ以外のすべてのライブラリでは、オブジェクトは FUSER システムファイルに格納されます。

このセクションでは、プログラミングオブジェクトに SYSMAIN 機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

## [Programming Objects] メニューのフィールド

[Programming Objects] メニューには、プログラミングオブジェクトの処理に必要なすべての SYSMAIN 機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明
Recat: ON	プロファイルパラメータ RECAT が ON に設定されていることを示します。「 <a href="#">プロファイルパラメータ RECAT の使用</a> 」を参照してください。
Code	保存（ソース）オブジェクトおよびカタログ化オブジェクトを処理するかどうかを指定します。
A	保存オブジェクトまたはカタログ化オブジェクト、あるいはその両方として存在するオブジェクトが処理されます。
C	カタログ化オブジェクトとして存在するすべてのオブジェクトが処理されます。
S	保存オブジェクトとして存在するすべてのオブジェクトが処理されます。
W	保存オブジェクトおよびカタログ化オブジェクトの両方として存在するオブジェクトのみが処理されます。この例外は、コピーコード、テキスト、および記録です。ただし、このオプションを指定すると処理対象になります。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。

フィールド	説明	
	Y	Yes。 選択処理がアクティブになります。オブジェクトの処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列、フィールド、および行コマンドの詳細については、「 <a href="#">プログラミングオブジェクトの選択リスト</a> 」を参照してください。
	N	No。 選択処理が非アクティブになります。
<b>Name</b>	処理するオブジェクトの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」も参照）。 デフォルト設定はアスタリスク（*）で、すべての名前が選択されます。	
<b>New Name</b>	名前の変更機能で名前を変更するときに、オブジェクトに付ける名前。	
<b>Type</b>	プログラムのPやマップのMなど、処理するオブジェクトタイプに対応するコード（「 <a href="#">TYPEの指定 - プログラミングオブジェクト</a> 」を参照）。 任意の順序で1つ以上のコードを入力できます。例えば、「PAM」と入力すると、パラメータデータエリアとマップが処理されます。	
<b>Set Number</b>	LIST XREF コマンドの Predict XRef セット保存オプションを使用して作成された保有セットの数。このセットに含まれるオブジェクトにすべての SYSMAIN 処理機能を適用できます。 有効な番号を指定すると、SYSMAIN で Predict セットであると見なされます。番号を指定しないと、通常のオブジェクト処理と見なされます。 [Additional Criteria] ウィンドウの [Set Library] フィールドおよび [Set User] フィールドを使用して、Predict セットのライブラリとユーザー ID を指定できます（「 <a href="#">追加条件の指定</a> 」を参照）。	
<b>XREF</b>	Predict システムファイルに保存された XRef（クロスリファレンス）データが処理されるかどうかを示します。	
	N	No。 削除機能を使用する場合を除いて、XRef データは処理されません。カタログ化オブジェクトを削除または置換すると、このオブジェクトの既存の XRef データはすべて SYSMAIN により必ず削除されます。N はデフォルト設定です。
	Y	Yes。 すべての XRef データが処理されます。
	S	Special。 クロスリファレンスデータがあるかどうかに関係なく、指定したオブジェクトが処理されます。すべての既存の XRef データが処理されます。
	F	Force。 すべての XRef データが処理されます。オブジェクトは Predict に記録される必要があります。

フィールド	説明								
	<p>詳細については、「<a href="#">XRefの考慮事項</a>」を参照してください。</p>								
Library	<p>ソースライブラリまたはターゲットライブラリの名前</p> <p>ソースライブラリは、処理対象のオブジェクトを含みます。ターゲットライブラリは、オブジェクトのコピー先または移動先のライブラリです。オブジェクト名はここで変更されることがあります。</p> <p>「<a href="#">ライブラリのリストと選択</a>」も参照してください。</p>								
Database	<p>ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。</p> <p>ソースデータベースは、処理対象のオブジェクトが保存されているライブラリおよびシステムファイルを含みます。ターゲットデータベースは、オブジェクトのコピー先または移動先のライブラリおよびシステムファイルを含みます。オブジェクト名はここで変更されることがあります。</p> <p>有効なデータベース ID は 1~65535 です。</p>								
File	<p>ソースシステムファイルまたはターゲットシステムファイル (FNAT または FUSER) のファイル番号 (FNR)。</p> <p>有効なファイル番号は 1~65535 です。</p> <p>ソースファイルは、処理対象のオブジェクトが保存されているライブラリを含みます。ターゲットファイルは、オブジェクトのコピー先または移動先のライブラリを含みます。オブジェクト名はここで変更されることがあります。</p>								
Replace	<p>移動、コピー、または名前変更機能の使用時にオブジェクトを置き換えるかどうかを指定します。</p> <table border="1" data-bbox="282 1157 1378 1478"> <tr> <td data-bbox="282 1157 662 1205"></td> <td data-bbox="662 1157 1378 1205"></td> </tr> <tr> <td data-bbox="282 1205 662 1318">Y</td> <td data-bbox="662 1205 1378 1318"> <p>Yes。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えます。</p> </td> </tr> <tr> <td data-bbox="282 1318 662 1432">N</td> <td data-bbox="662 1318 1378 1432"> <p>No。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えません。これはデフォルト設定です。</p> </td> </tr> <tr> <td data-bbox="282 1432 662 1478"></td> <td data-bbox="662 1432 1378 1478"></td> </tr> </table> <p>「<a href="#">置換オプションの使用</a>」も参照してください。</p>			Y	<p>Yes。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えます。</p>	N	<p>No。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えません。これはデフォルト設定です。</p>		
Y	<p>Yes。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えます。</p>								
N	<p>No。 ターゲット環境に存在する名前と同じ名前を持つオブジェクトを置き換えません。これはデフォルト設定です。</p>								
Criteria	<p>追加のオブジェクト選択条件を指定できる [Additional Criteria] ウィンドウを呼び出します。可能な値は次のとおりです。</p> <table border="1" data-bbox="282 1619 1378 1856"> <tr> <td data-bbox="282 1619 703 1665"></td> <td data-bbox="703 1619 1378 1665"></td> </tr> <tr> <td data-bbox="282 1665 703 1778">N</td> <td data-bbox="703 1665 1378 1778"> <p>No。 [Additional Criteria] ウィンドウを呼び出しません。これはデフォルト設定です。</p> </td> </tr> <tr> <td data-bbox="282 1778 703 1856">Y</td> <td data-bbox="703 1778 1378 1856"> <p>Yes。 [Additional Criteria] ウィンドウを呼び出します。</p> </td> </tr> <tr> <td data-bbox="282 1856 703 1900"></td> <td data-bbox="703 1856 1378 1900"></td> </tr> </table>			N	<p>No。 [Additional Criteria] ウィンドウを呼び出しません。これはデフォルト設定です。</p>	Y	<p>Yes。 [Additional Criteria] ウィンドウを呼び出します。</p>		
N	<p>No。 [Additional Criteria] ウィンドウを呼び出しません。これはデフォルト設定です。</p>								
Y	<p>Yes。 [Additional Criteria] ウィンドウを呼び出します。</p>								

フィールド	説明
	「 <a href="#">追加条件の指定</a> 」も参照してください。

## プロファイルパラメータ RECAT の使用

セッションの開始時にプロファイルパラメータ RECAT が ON に設定されている場合、この設定は、**[Programming Objects]** メニューと **[Selection]** 画面に示されます。

RECAT=ON で SYSMAIN 機能を使用すると、標準のダイナミックな再カタログルールが適用されます（『パラメータリファレンス』ドキュメントの「RECAT - ダイナミックな再カタログ」を参照）。これは次のことを意味します。

- オブジェクトが保存オブジェクトとカタログ化オブジェクトの両方として存在する場合、保存オブジェクトとカタログ化オブジェクトのいずれも独立して処理することはできません。
- オブジェクトがカタログ化オブジェクトとしてのみ存在する場合は、カタログ化オブジェクトのみを処理する SYSMAIN 機能を実行できない（「ダイナミックな再カタログの要求が不正です。」というエラーメッセージが表示されます）か、または **[Selection]** 画面からカタログ化オブジェクトを選択できません。

自動処理を使用する場合、これらのルールに従わないすべてのオブジェクトは無視され、処理は次のオブジェクトに移ります。

## プログラミングオブジェクトの選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべてのプログラミングオブジェクトの選択リストが **[Selection]** 画面に表示されます。

このセクションでは、**[Selection]** 画面に表示される列とフィールド、およびプログラミングオブジェクトをさらに処理するための行コマンドについて説明します。

### ■ 列とフィールド

■ 行コマンド

列とフィールド

次の列とフィールドが [Selection] 画面に表示されます。

列/フィールド	説明
C	行コマンド用の入力フィールド（下記参照）。
Name	指定した選択条件に一致するプログラミングオブジェクトの名前。
Type	「 <b>TYPE 指定 - プログラミングオブジェクト</b> 」でリストしているオブジェクトのタイプに対応するコード。
S/C	プログラミングオブジェクトに存在するオブジェクト。保存/ソース (S) オブジェクトまたはカタログ化 (C) オブジェクト、あるいはその両方。
Message	プログラミングオブジェクトの処理ステータスを示すメッセージ。表示されるメッセージについては、「 <b>ステータスメッセージ</b> 」を参照してください。
Listed Library	選択したプログラミングオブジェクトを含むライブラリの名前。

行コマンド

次のいずれかの行コマンドを、[Selection] 画面の [C]（コマンド）列に入力できます。

行コマンド	機能
A	<p>[S/C]（保存/カタログ化）列に表示されたすべてのオブジェクト、つまり、保存/ソース (S) オブジェクトまたはカタログ化 (C) オブジェクトあるいはその両方を処理します。</p> <p>このコマンドは、検索またはリスト機能には適用されません。</p>
B	<p>カタログ化オブジェクトを Natural バッファプールから削除します。オブジェクトを指定して Enter キーを押すと表示されるウィンドウに「DELETE」を入力して、指定したオブジェクトの削除を確認する必要があります。</p>
C	<p>対応する保存オブジェクトがある場合にも、カタログ化オブジェクトのみを処理します。保存オブジェクトとしてのみ存在するオブジェクトに C を指定した場合、エラーが発生します。</p> <p>このコマンドは、検索またはリスト機能には適用されません。</p>
D	<p>[Library Selection] 画面にのみ適用されます（「<b>ライブラリのリストと選択</b>」も参照）。</p> <p>指定したライブラリに含まれるオブジェクトの短いリストを表示します。リストに含まれる情報（名前、タイプ、ソース/カタログ化オブジェクト）は、「<b>メニュー機能とコマンドの使用</b>」の「<b>選択処理</b>」で示した [Selection] 画面に表示される情報と同じです。</p>
H	<p>保存（ソース）オブジェクトのハードコピーを生成します。</p> <p>指定したオブジェクトのソースコードが出力され、画面に表示されます。</p>

行 コマ ンド	機能
I	<p>オブジェクトのディレクトリ情報を表示します。</p> <p>このコマンドは、システムコマンド <code>LIST DIR object-name</code> に類似しています。表示されるディレクトリ情報の詳細については、『システムコマンド』ドキュメントの「ディレクトリ情報の表示」を参照してください。</p>
L	<p>保存（ソース）オブジェクトのソースコードを表示します。</p> <p>このコマンドは、システムコマンド <code>LIST object-name</code> に対応します。ソースコード画面から実行できるコマンドの詳細については、『システムコマンド』ドキュメントの「ソースのリスト」を参照してください。例外：EXPAND コマンドは使用できません。表示するソースオブジェクトが <code>steplib</code> ライブラリに含まれている場合、ZOOM コマンドは使用できません。</p> <p><b>[Library Selection]</b> 画面上（「<a href="#">ライブラリのリストと選択</a>」も参照）：</p> <p>ライブラリに含まれるオブジェクトの拡張リストを表示します。行コマンド D によって表示される情報に加えて、拡張リストには、プログラミングモード、Natural バージョン、ユーザー ID、保存／カタログ化日時などのオブジェクトディレクトリ情報が表示されます。</p>
R	<p>リソースタイプの Natural オブジェクトのロングネームを表示します。</p> <p><b>[Library Selection]</b> 画面上（「<a href="#">ライブラリのリストと選択</a>」も参照）：</p> <p>外部サブルーチンの使用を確認します。指定したライブラリとオブジェクトの範囲について、カタログ化オブジェクトが存在するオブジェクトを表示し、オブジェクトが外部サブルーチンを参照するかどうかを示します。</p> <p>外部サブルーチンを参照するオブジェクトごとに、外部サブルーチンの名前、および可能な場合は、そのサブルーチンに存在するカタログ化オブジェクトの名前を表示します。サブルーチンタイプのオブジェクトの場合、同等の別名のロングネームが表示されます。</p>
S	<p>対応するカタログ化オブジェクトがある場合にも、保存（ソース）オブジェクトのみを処理します。カタログ化オブジェクトとしてのみ存在するオブジェクトに S を指定した場合、エラーが発生します。</p> <p>このコマンドは、検索またはリスト機能には適用されません。</p> <p><b>[Library Selection]</b> 画面上（「<a href="#">ライブラリのリストと選択</a>」も参照）：</p> <p>別名のロングネームまたは同等のカタログ化オブジェクトのショートネーム、あるいはその両方によって、指定したライブラリに含まれているすべての外部サブルーチンをリストします。</p>
X	<p>Natural Connection と Entire Connection がインストールされている場合にのみ適用されます。</p> <p>保存（ソース）オブジェクトを PC にダウンロードします（「<a href="#">PC へのソースオブジェクトのダウンロード</a>」も参照）。</p>
Z	<p>サイズの計算：DATSIZE、ESIZE、MCG サイズなどの、保存（ソース）オブジェクトおよびカタログ化オブジェクトのサイズを確認します。</p>





```

! Specify the relevant PC options                ! S/C
!
! Drive ..... C                                ! S/C
! Path ..... SOURCES/SUBPROGRAMS                ! S/C
! Extension .. NS*                              ! S
!
!
!
!
! Warning: This will overwrite any existing objects with !
!
! the same path, name and extension.            ! Exit) _
+-----+ -PF10--PF11--PF12---
      Help  Menu  Exit  Copy  Del   Find  List  Move  Ren          Canc

```

〔**Drive**〕フィールドに、オブジェクトのダウンロード先のPCドライブの名前を入力します。デフォルト設定はCです。

〔**Path**〕フィールドに、オブジェクトのダウンロード先のPCディレクトリ／サブディレクトリの名前を入力します（上記の例では、*SOURCES*ディレクトリの*SUBPROGRAMS*サブディレクトリ）。ディレクトリとサブディレクトリ間、および複数のサブディレクトリ間のセパレータとしてスラッシュ (/) を入力します。指定したディレクトリ／サブディレクトリが存在しない場合は、該当するエラーメッセージが返されます。

〔**Extension**〕フィールドに、オブジェクトのソースコードのロード先のテキストファイルの拡張子を入力します。NS\* と指定すると、アスタリスクが指定したオブジェクトタイプで置き換えられます（「**TYPE の指定 - プログラミングオブジェクト**」を参照）。

例：SUB1という名前のサブプログラムがSUB1.NSNという名前のファイルにロードされます。

- 4 ENTER キーを押して、オブジェクトをダウンロードします。

メッセージ行に、「4824 : 要求したオプションの処理が正常終了しました。」というメッセージが表示されます。

また、画面に〔**Message**〕列が表示された場合、ダウンロードしたオブジェクトの横にExported というメッセージが表示されます。

- 5 必要に応じて、この選択リストから他のオブジェクトのダウンロードを続けます。

現在の〔**Selection**〕画面からの追加のダウンロードに対して、〔**PC Download Options**〕ウィンドウは再表示されません。このウィンドウは、〔**Selection**〕画面の呼び出し後、ま

たはダイレクトコマンド SET PC の実行後に、行コマンド X を最初に発行したときのみ表示されます。

〔PC Download Options〕ウィンドウの設定は、SYSMAIN を終了するまでアクティブのままです。

## XRef について

---

Predict システムファイルに格納されたすべてのクロスリファレンス (XRef) データは、SYSMAIN で処理できます。XREF オプションでは、XRef データを SYSMAIN で処理する必要があるかどうかを示されます。カタログ化オブジェクトに削除または置き換え機能が実行される場合、XRef データは常に削除されます。

Predict がインストールされていない場合、XREF オプションを N に設定すると、Predict ファイルの検証は実行されません。XREF オプションを Y に設定したときに、使用中の FDIC ファイルが有効な Predict ファイルでない場合、エラーメッセージが返されます。

XREF オプションを設定する場合のルールは、Natural Security によるルールと同じです。ただし、非セキュリティ環境では、制限はありません。

このセクションでは、次のトピックについて説明します。

- XREF を N に設定
- XREF を Y または F に設定
- XREF を F に設定
- XREF を S に設定
- XREF エラー

### XREF を N に設定

XREF オプションを N に設定した場合、XRef データは処理されませんが、カタログ化プログラミングオブジェクトが削除されるか置き換えられると、SYSMAIN では XRef が削除されます。ターゲット Predict システムファイルは、ソースまたはターゲットの FDIC システムファイルの現在の設定に従って決定されます。デフォルトは、Natural セッションの開始時に FDIC プロファイルパラメータ割り当てられた値です (『パラメータリファレンス』ドキュメントの「FDIC - Predict システムファイル」を参照)。

## XREF を Y または F に設定

XREF オプションを Y または F に設定した場合、処理中に次のアクションが適用されます。

- SYSMAIN では、XRef データが Predict システムソースファイルにすでに存在するかどうかを検証されます。
- 置き換えオプションがアクティブ (Y に設定) で、プログラミングオブジェクトがターゲット環境から削除される場合、XRef データは Predict システムターゲットファイルから削除されます。
- プログラミングオブジェクトを新しい環境にコピーすると、プログラミングオブジェクトの XRef データが Predict システムソースファイルから Predict システムターゲットファイルにコピーされます。それに応じてライブラリ名を名前の変更機能を使用して変更すると、オブジェクト名も変更されます。
- 移動機能が要求されると、プログラミングオブジェクトの XRef データが Predict システムソースファイルから削除されます。

## XREF を F に設定

XREF オプションを F に設定した場合、SYSMAIN では、プログラミングオブジェクト (プログラム、サブルーチン、サブプログラム、マップ、またはヘルプルーチンのみ) に、Predict システムターゲットファイルで定義された Predict プログラムエントリがあるかどうかさらにチェックされます。ない場合、オブジェクトの処理は終了します。

## XREF を S に設定

XREF オプションを S に設定した場合、特殊ケースが適用され、指定したオブジェクトの範囲は、すべてのオブジェクトにクロスリファレンスデータがあるかどうかに関係なく、対応する XRef データとともに処理されます。クロスリファレンスデータがあるオブジェクトはクロスリファレンスデータとともに処理され、クロスリファレンスデータがないオブジェクトも処理されます。

## XREF エラー

XRef データの SYSMAIN 処理中に次の矛盾が発生すると、オブジェクトまたは機能のすべての処理が終了し、エラーメッセージが表示されます。

- Natural Security の XREF オプションの値は F または Y で、それぞれ Y または N の値を指定しました。
- XREF オプションは F に設定され、SYSMAIN では、処理中のオブジェクトの記録されたプログラムエントリが Predict に検出されませんでした。
- 無効な Predict ファイルを指定しました。
- Natural Security の XREF オプションの値は F または Y で、S 値を指定しました。

## 追加条件の指定

[Programming Object] メニューの入力フィールドで指定した選択条件に加えて、オブジェクトの保存またはカタログ化に関連する日付/時刻、ユーザー ID、および端末 ID によってオブジェクトを選択できます。

また、Predict セットのユーザー ID とライブラリを指定することもできます。このオプションは、リスト機能および検索機能には適用されません。

例えば、特定の日の 8 時～12 時の間に特定のユーザーによって特定の端末でカタログ化されたオブジェクトのみを選択できます。つまり、選択条件に従ったオブジェクトの処理は、各条件ではなく、すべての選択された条件全体に基づきます。

### ▶手順 89.2. 追加の選択条件を指定するには

- 1 [Programming Objects] メニューの [Criteria] フィールドで、N (デフォルト) を Y で置き換えます。

次のような [Additional Criteria] ウィンドウが表示されます。

```

15:24:55          ***** NATURAL SYSMAIN UTILITY *****          2006-09-15
User MM0          - Copy Programming Objects -

+-----+
!           --- Additional Criteria ---           !
!
! Object Type ..... PN_____ !
! Date/Time From .. 2006-01-01 _____ !
! Date/Time To .... _____ !
! User ID ..... SAG_____ !
! Terminal ID ..... _____ !
! Set Library ..... _____ !
! Set User ..... _____ !
!
Code .. ! Command ==> !
Object Name .. !
+-----+ N
Source Library ... SYSTEM__ Database .... 10__ File .. 32__
Target Library ... _____ Database .... 10__ File .. 32__
Options Replace ... N          + Criteria .... Y

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
```

Help Menu Exit Copy Del Find List Move Ren Fsec Fdic Fnat

必要な選択条件を入力します。[Additional Criteria] ウィンドウの [Object Type] フィールドに1つ以上のオブジェクトタイプコードを入力した場合、[Programming Objects] メニューの [Type] フィールドは同じオブジェクトタイプコードで事前設定されます。

- 2 [Programming Objects] メニューに戻ると、[Criteria] フィールドの前のプラス記号 (+) は、追加のオブジェクト選択条件が指定されていることを示します。上記の例では、ユーザー ID SAG です。

[Object Type] フィールドのみにエントリが含まれている場合、このエントリはすでに [Programming Objects] メニューの [Type] フィールドに表示されるため、プラス記号 (+) は表示されません。

## プログラミングオブジェクトのダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、プログラミングオブジェクトに SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- FIND、LIST、および LISTLIB
- RENAME
- *where-clause*

- with-clause

## COPY と MOVE

$\left\{ \begin{array}{l} \text{COPY} \\ \text{MOVE} \end{array} \right\}$	$\left[ \left[ \begin{array}{l} \text{ALL} \\ \text{CATALOGED} \\ \text{SAVED} \\ \text{STOWED} \end{array} \right] \right]$	<i>name</i> [ FM [LIBRARY] ]	] [where-clause]
		TO [LIBRARY]	] [with-clause]

例：

```
COPY PROG1 FM TESTORD TO ORDERS DBID 1 FNR 6 REP
```

```
C PGM* WITH REP TYPE PNS FM PRODLIB TO TESTLIB
```

```
M PROG1 FM OLDLIB TO NEWLIB
```

```
MOVE STOWED * TO NEWLIB WHERE DBID 100 FNR 160 FMDATE 2007-01-01 FM OLDLIB WITH XREF Y
```

## DELETE

$\text{DELETE}$	$\left[ \left[ \begin{array}{l} \text{ALL} \\ \text{CATALOGED} \\ \text{SAVED} \\ \text{STOWED} \end{array} \right] \right]$	<i>name</i> [ IN [LIBRARY] <i>lib-name</i> ]	] [where-clause] [with-clause]

例：

```
DELETE C M> IN LIB ORDERS
```


```
D * IN TESTLIB DBID 1 FNR 5 NAME SYSNAT
```

```
D SA * IN LIBTEST TYPE GLA
```

```
D * TYPE PM IN TESTORD FMDATE 2007-01-01 TODATE 2007-04-30
```

## FIND、LIST、および LISTLIB

$\left\{ \begin{array}{l} \text{FIND} \\ \text{LIST} \\ \text{LISTLIB} \end{array} \right\}$	$\left[ \left[ \begin{array}{l} \text{ALL} \\ \text{CATALOGED} \\ \text{SAVED} \\ \text{STOWED} \end{array} \right] \right]$	<i>name</i> [ IN [LIBRARY] <i>lib-name</i> ]	] [where-clause] [with-clause]

 **注意:** ダイレクトコマンド LISTLIB は、バッチモードでのみ使用可能で、ライブラリ名のリストの取得に使用します。

例：

```
FIND SAVED MENU IN TESTLIB
```

```
FIND STOWED MAINMENU IN SYS* WHERE DBID 1 FNR 5
```

```
F ALL PROG2 IN PROD* FNR 27 DBID 1
```

```
LIST * IN TESTLIB
```

```
LIST DT* IN TESTLIB
```

```
L SAVED TEST* IN TESTLIB TYPE PNS FNR 6
```

```
L SA TEST* TYPE PM IN TESTLIB FNR 6 DBID 2 FMDATE 2007-01-01
```

```
LISTLIB ALL MENU IN SYS* DBID 10 FNR 44
```

## RENAME

```
RENAME [ [ { ALL  
          CATALOGED  
          SAVED  
          STOWED } ] ] name AS new-name [with-clause]  
          [ FM [LIBRARY] lib-name ] [where-clause]  
          [ TO [LIBRARY] lib-name [where-clause] ]
```

例：

```
RENAME PGM1 AS PROG1 FM TESTLIB
```

```
R PGM1 AS PROG1 FM TESTLIB DBID 1 FNR 5 TO PRODLIB DBID 2 FNR 6
```

```
R PGM* TYPE PS RCOP FM TESTLIB TO PRODLIB
```

## where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name] [CIPHER cipher]  
          [ { PASSWORD } password ]  
          [ { PSW } password ]  
          [DIC (dbid,fnr,password,cipher)]  
          [SEC (dbid,fnr,password,cipher)]
```

## セパレータ

DICおよびSECキーワードに続く各値の間に、または値がない場合に、セパレータとしてコンマを使用する必要があります。例えば、DIC (10,,secret,2a)のように使用します。

セッションパラメータ ID (『パラメータリファレンス』ドキュメントの「ID-INPUT 区切り文字」を参照) がコンマに設定されている場合、値を区切るセパレータとしてスラッシュ (/) を使用します。

**with-clause**

[WITH]	[TYPE <i>type</i> ]	[FMDATE <i>date-from</i> ]	[TODATE <i>date-to</i> ]	[FMTIME <i>time-from</i> ]
	[TOTIME <i>time-to</i> ]	[USER <i>user-id</i> ]	[TID <i>terminal-id</i> ]	[XREF <i>xref</i> ] [HELP]
[REPLACE]	[RCOP]	[EXTEND]	[	PROMPT NOPROMPT ] [
				MON NOMON ]
[SETUSER <i>set-user</i> ]	[SETNO <i>set-number</i> ]	[SETLIBRARY <i>set-library</i> ]		



# 90 デバッグ環境処理

---

- デバッグ環境メニューのフィールド ..... 556
- デバッグ環境の選択リスト ..... 558
- デバッグ環境のダイレクトコマンド構文 ..... 559

検索機能を除いて、すべての SYSMAIN 機能をデバッグ環境で実行できます。

デバッグ環境の指定は、関連する FUSER システムファイルのデータベース ID (DBID) とファイル番号 (FNR) に常に対応している必要があります。

このセクションでは、デバッグ環境で SYSMAIN 機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

関連トピック：

「*Debug Environment Maintenance*」 - 『デバッグ』ドキュメント

## デバッグ環境メニューのフィールド

[**Debug Environments**] メニューには、デバッグ環境の処理に必要なすべての SYSMAIN 機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	実行される機能を指定します（「 <a href="#">機能の説明</a> 」を参照）。	
	C	デバッグ環境をコピーします。
	D	デバッグ環境を削除します。
	L	デバッグ環境をリストします。
	M	デバッグ環境を移動します。
	R	デバッグ環境の名前を変更します。
<b>注意:</b> デバッグ環境が1つのライブラリから他のライブラリに移動またはコピーされた場合、ブレイクポイントとウォッチポイントは新しいライブラリに適合される必要があります。詳細については、『デバッグ』ドキュメントの「 <i>Maintain Debug Environments in Different Libraries</i> 」を参照してください。		
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	
	Y	Yes。 選択処理がアクティブになります。デバッグ環境の処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列、フィールド、および行コマンドの詳細については、「 <a href="#">デバッグ環境の選択リスト</a> 」を参照してください。
N	No。 選択処理が非アクティブになります。	

フィールド	説明	
名前	<p>処理するデバッグ環境の名前または名前の範囲（「<a href="#">名前の範囲の指定</a>」も参照）。</p> <p>デフォルト設定はアスタリスク（*）で、すべての名前が選択されます。</p>	
New Name	<p>名前の変更機能で名前を変更するときに、デバッグ環境に付ける名前。</p>	
Library	<p>ソースライブラリまたはターゲットライブラリの名前</p> <p>ソースライブラリは、処理対象のデバッグ環境を含みます。ターゲットライブラリは、デバッグ環境のコピー先または移動先のライブラリです。デバッグ環境名はここで変更されることがあります。</p> <p>「<a href="#">ライブラリのリストと選択</a>」も参照してください。</p>	
Database	<p>ソースデータベースまたはターゲットデータベースのデータベース ID（DBID）。</p> <p>ソースデータベースは、処理対象のデバッグ環境が保存されているライブラリとシステムファイルを含みます。ターゲットデータベースは、デバッグ環境のコピー先または移動先のライブラリおよびシステムファイルを含みます。デバッグ環境名はここで変更されることがあります。</p> <p>有効なデータベース ID は 1～65535 です。</p>	
File	<p>ソースシステムファイルまたはターゲットシステムファイル（FNATまたはFUSER）のファイル番号（FNR）。</p> <p>有効なファイル番号は 1～65535 です。</p> <p>ソースファイルは、処理対象のデバッグ環境が保存されているライブラリを含みます。ターゲットファイルは、デバッグ環境のコピー先または移動先のライブラリを含みます。デバッグ環境名は、ここで名前が変更されることがあります。</p>	
Replace	<p>移動、コピー、または名前変更機能の使用時にデバッグ環境を置き換えるかどうかを指定します。</p>	
	Y	<p>Yes。 ターゲット環境に存在する名前と同じ名前を持つデバッグ環境を置き換えます。</p>
	N	<p>No。 ターゲット環境に存在する名前と同じ名前を持つデバッグ環境を置き換えません。これはデフォルト設定です。</p>
	<p>「<a href="#">置換オプションの使用</a>」も参照してください。</p>	

## デバッグ環境の選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべてのデバッグ環境の選択リストが **[Selection]** 画面に表示されます。

このセクションでは、**[Selection]** 画面に表示される列とフィールド、およびデバッグ環境をさらに処理するための行コマンドについて説明します。

- 列とフィールド
- 行コマンド

### 列とフィールド

次の列とフィールドが **[Selection]** 画面に表示されます。

列／フィールド	説明
C	行コマンド用の入力フィールド（下記参照）。
Environment	指定した選択条件に一致するデバッグ環境の名前。
Message	デバッグ環境の処理ステータスを示すメッセージ。表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。
Listed Library	選択したデバッグ環境を含むライブラリの名前。

### 行コマンド

次のいずれかの行コマンドを、**[Selection]** 画面の **[C]**（コマンド）列に入力できます。

行コマンド	機能
A	デバッグ環境を処理します。  この行コマンドは、 <b>[List Selection]</b> 画面では使用できません。 <b>[List Selection]</b> 画面では、ピリオド（.）を入力できるのは画面を終了する場合のみです。
L	指定したライブラリに含まれるデバッグ環境のリストを表示します。  <b>[Library Selection]</b> 画面にのみ適用されます（「 <a href="#">ライブラリのリストと選択</a> 」も参照）。

## デバッグ環境のダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用してデバッグ環境に SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- LIST
- RENAME
- *where-clause*
- *with-clause*

### COPY と MOVE

```
{ COPY } DEBUG name FM [LIBRARY] lib-name [where-clause]
{ MOVE } TO [LIBRARY] lib-name [where-clause] [with-clause]
```

例：

```
COPY D ENV FM TESTLIB WHERE DBID 1 FNR 5 TO PRODLIB WHERE DBID 2 FNR 5 WITH REP
```

```
C DEBUG ENV FM TESTLIB FNR 6 TO PRODLIB FNR 7 REP
```

```
MOVE DEBUG ENV FM OLDLIB WHERE DBID 1 FNR 5 TO NEWLIB WHERE DBID 2 FNR 5
```

```
M DEBUG ENV FM OLDLIB FNR 6 TO NEWLIB FNR 7 REP
```

## DELETE

```
DELETE DEBUG name [ IN [LIBRARY] lib-name ] [where-clause] [with-clause]
```

例：

```
DELETE DEBUG U* IN TESTLIB FNR 150
```

```
D DEBUG TEST* IN TESTLIB IN DBID 177 FNR 205
```

## LIST

```
LIST DEBUG name [ IN [LIBRARY] lib-name ] [where-clause] [with-clause]
```

例：

```
LIST DEBUG ENV* IN TESTLIB DBID 1 FNR 5
```

```
L D DT* IN TESTLIB DBID 10
```

## RENAME

```
RENAME DEBUG name ASnew-name [with-clause]  
          IN [LIBRARY] lib-name [where-clause]  
          TO [LIBRARY] lib-name [where-clause]
```

例：

```
RENAME D OLDENV AS NEWENV IN TESTLIB RCOP
```

```
R DEBUG OLDENV AS NEWENV IN TESTLIB DBID 1 FNR 4 TO PRODLIB DBID 1 FNR 5
```

```
R DEBUG OLDENV AS NEWENV IN TESTLIB FNR 4 TO PRODLIB FNR 5 REPLACE RCOP
```

### where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name ]  
[CIPHER cipher] [ { PASSWORD } password ]  
                  [ PSW ]
```

**with-clause**

[WITH] [REPLACE] [RCOP] [ PROMPT NOPROMPT ] [ MON NOMON ] [HELP]
------------------------------------------------------------------------





# 91 エラーメッセージ処理

---

■ エラーメッセージメニューのフィールド .....	564
■ エラーメッセージの選択リスト .....	566
■ エラーメッセージ番号の変更 .....	568
■ 言語の指定 .....	569
■ エラーメッセージのダイレクトコマンド構文 .....	569

すべてのSYSMAIN機能をユーザー定義のエラーメッセージおよびNaturalシステムエラーメッセージに実行できます。エラーメッセージは、1つのライブラリから他のライブラリに移動できます。または、各エラーメッセージの言語テキストはコピー、移動、または置き換えが可能です。また、名前の変更機能を使用して、1つのエラーメッセージまたはエラーメッセージの範囲の番号を変更できます。

Natural Security がインストールされている場合は、権限を持つユーザーのみが Natural システムエラーメッセージを処理できます。

エラーメッセージは、タイプに応じてシステムファイルに格納されます。Natural システムエラーメッセージは FNAT システムファイルに格納され、ユーザー定義のエラーメッセージは FUSER または FNAT システムファイルに格納されます。

このセクションでは、プログラミングオブジェクトにSYSMAIN機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

関連トピック：

[『SYSERR Utility』ドキュメント](#)

## エラーメッセージメニューのフィールド

[エラーメッセージテキスト] メニューには、エラーメッセージの処理に必要なすべてのSYSMAIN機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	ショートまたは拡張（ロング）エラーメッセージ、あるいはその両方を処理するかどうかを指定します。可能な値は次のとおりです。	
	A	ショートまたは拡張（ロング）メッセージ、あるいはその両方が存在するすべてのエラーメッセージを処理します。
	E	拡張（ロング）エラーメッセージが存在するすべてのエラーメッセージを処理します。対応するショートエラーメッセージがある拡張エラーメッセージのみを処理します。  対応するショートエラーメッセージがターゲット環境にない場合、拡張エラーメッセージはターゲット環境に転送できません。
	S	ショートエラーメッセージが存在するすべてのエラーメッセージを処理します。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	

フィールド	説明
	<p>Y</p> <p>Yes。 選択処理がアクティブになります。エラーメッセージの処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列と行コマンドの詳細については、「<a href="#">エラーメッセージの選択リスト</a>」を参照してください。</p> <p>N</p> <p>No。 選択処理が非アクティブになります。</p>
<b>[No. From]</b> または <b>[Number]</b>	処理するエラーメッセージの数、または終了番号を <b>[No. To]</b> フィールドに入力した場合は、番号の範囲の開始番号。「 <a href="#">エラーメッセージ番号の変更</a> 」も参照してください。
<b>No. To</b>	開始番号を <b>[No. From]</b> フィールドに入力した場合は、処理するエラーメッセージ番号の範囲の終了番号。「 <a href="#">エラーメッセージ番号の変更</a> 」も参照してください。
<b>New From</b>	名前の変更機能に適用されます。  既存のエラーメッセージに割り当てる新しい番号、または既存のエラーメッセージの範囲に割り当てる新しい番号の範囲の開始番号。「 <a href="#">エラーメッセージ番号の変更</a> 」も参照してください。
<b>New To</b>	名前の変更機能に適用されます。  既存のエラーメッセージの範囲に割り当てる新しいエラーメッセージ番号の範囲の終了番号。「 <a href="#">エラーメッセージ番号の変更</a> 」も参照してください。
<b>Library</b>	ソースライブラリまたはターゲットライブラリの名前または名前の範囲。Natural システムエラーメッセージを処理する場合は、 <b>[Library]</b> フィールドを空のままにします。  ソースライブラリには、処理対象のエラーメッセージが含まれています。ターゲットライブラリは、エラーメッセージのコピー先または移動先のライブラリです。エラーメッセージは、ここで番号が変更されることがあります。  <a href="#">「ライブラリのリストと選択」</a> も参照してください。
<b>Lang.</b>	エラーメッセージを使用できる言語コード。  言語コードを任意に組み合わせて、言語を指定できます。言語に割り当てられている言語コードについては、『システム変数』ドキュメントの「*LANGUAGE」で「 <a href="#">言語コードの割り当て</a> 」を参照してください。Natural システムの拡張（ロング）エラーメッセージでは、言語コード 1 のみを使用できます。すべての既存の言語のエラーメッセージを選択するには、アスタリスク（*）を入力します。  <a href="#">「言語の指定」</a> も参照してください。
<b>Database</b>	ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。  ソースデータベースは、処理対象のエラーメッセージが保存されているライブラリおよびシステムファイルを含みます。ターゲットデータベースは、エラーメッセージのコピー先または移動先のライブラリおよびシステムファイルを含みます。エラーメッセージは、ここで番号が変更されることがあります。

フィールド	説明				
	有効なデータベース ID は 1~65535 です。				
<b>File</b>	<p>ソースシステムファイルまたはターゲットシステムファイルのファイル番号 (FNR)。</p> <p>有効なファイル番号は 1~65535 です。</p> <p>ソースファイルは、処理対象のエラーメッセージが保存されているライブラリを含みます。ターゲットファイルは、エラーメッセージのコピー先または移動先のライブラリを含みます。エラーメッセージは、ここで番号が変更されることがあります。</p>				
<b>Replace</b>	<p>移動、コピー、または名前変更機能の使用時にエラーメッセージを置き換えるかどうかを指定します。</p> <table border="1"> <tr> <td>Y</td> <td>Yes。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えます。</td> </tr> <tr> <td>N</td> <td>No。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えません。これはデフォルト設定です。</td> </tr> </table> <p>「置換オプションの使用」も参照してください。</p>	Y	Yes。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えます。	N	No。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えません。これはデフォルト設定です。
Y	Yes。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えます。				
N	No。 ターゲット環境に存在する番号と同じ番号を持つエラーメッセージを置き換えません。これはデフォルト設定です。				

## エラーメッセージの選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべてのエラーメッセージの選択リストが **[Selection]** 画面に表示されます。

このセクションでは、**[Selection]** 画面に表示される列、およびエラーメッセージをさらに処理するための行コマンドについて説明します。

- 列
- 行コマンド

## 列

[Selection] 画面に次の列が表示されます。

列	説明
C	行コマンド用の入力フィールド（下記参照）。
Error	指定した選択条件に一致するエラーメッセージの名前。
Error Message Text	ショートエラーメッセージのテキスト。
Type	エラーメッセージのタイプ：S（ショート）およびE（拡張/ロング）。
Lang	エラーメッセージについて存在する、順番の、またはアルファベット順の第1言語コード。エラーメッセージについて存在するすべての言語を表示する場合は、[Selection] 画面で、行コマンド「L」を入力するか、またはリスト機能を使用します。
Message	エラーメッセージの処理ステータスを示すメッセージ。  この列は、エラーメッセージで行コマンドが実行された後にのみ表示されます。  表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。

## 行コマンド

次のいずれかの行コマンドを、[Selection] 画面の [C]（コマンド）列に入力できます。

行コマンド	機能
A	[Type] 列に表示されたすべてのタイプのメッセージを処理します。指定できるタイプは、ショート（S）、拡張/ロング（E）、またはショートおよび拡張（S/E）です。
E	ターゲット環境に対応するショートメッセージが存在する場合、拡張（ロング）メッセージを処理します。  ショートメッセージとしてのみ存在するメッセージにEを指定した場合、エラーが返されません。
S	すべてのショートメッセージを処理します。
L	処理する前にエラーメッセージを確認します。ショートまたは拡張（ロング）エラーメッセージ、あるいはその両方は、以前に [Error Message Texts] メニューで指定した機能に応じて、すべての既存の言語について表示されます。  この行コマンドは、[Library Selection] 画面でも使用できます（「 <a href="#">ライブラリのリストと選択</a> 」も参照）。

## エラーメッセージ番号の変更

---

1つのエラーメッセージの番号またはエラーメッセージの範囲の番号を変更できます。

### ▶手順 91.1. 1つのエラーメッセージの番号を変更するには

- 1 [No. From] フィールドに、変更するエラーメッセージの番号を入力し、[New From] フィールドに、新しいエラーメッセージ番号を入力します。
- 2 [Source Library] フィールドに、番号を変更するエラーメッセージが含まれているライブラリの名前を入力します。

番号を変更したエラーメッセージを異なるライブラリに配置する場合は、[Target Library] フィールドに名前を入力します。

### ▶手順 91.2. エラーメッセージの範囲の番号を変更するには

- 1 [No. From] フィールドに、番号を変更するエラーメッセージの範囲の開始番号を入力し、[No. To] フィールドに、終了番号を入力します。

[New From] フィールドに、エラーメッセージの新しい範囲の開始番号を入力し、[New To] フィールドに、終了番号を入力します。

ソースライブラリの [No. From] および [No. To] フィールドの範囲で指定されたエラーメッセージの番号は、ターゲットライブラリの [New From] および [New To] フィールドの範囲の番号と同じである必要があります。例えば、エラーメッセージ番号 1~6 を新しいエラーメッセージ番号 7~10 として変更することはできません。

1つのライブラリ内のエラーメッセージの範囲を変更する場合、範囲の値が重複しないようにしてください。例えば、エラーメッセージ番号 1~6 を新しいエラーメッセージ番号 5~10 として変更することはできません。

- 2 [Source Library] フィールドに、番号を変更するエラーメッセージが含まれているライブラリの名前を入力します。

番号を変更したエラーメッセージを異なるライブラリに配置する場合は、[Target Library] フィールドに名前を入力します。

「エラーメッセージのダイレクトコマンド構文」の「**RENAME**」の範囲指定の例も参照してください。

## 言語の指定

言語を指定する場合は、次の点を考慮してください。

- リスト機能を実行し、エラーメッセージについて指定した言語コードが存在しない場合、SYSMAINでは、システム変数\*LANGUAGE（『システム変数』ドキュメントの「\*LANGUAGE」を参照）で指定したデフォルトの言語が使用されます。
- ソースエラーメッセージにアスタリスク（\*）を指定した場合、このエラーメッセージに定義されたすべての言語コードによって、ターゲットエラーメッセージに定義されたすべての言語コードが上書きされます。

例えば、ソースエラーメッセージは言語1、2、および3のみで存在し、ターゲットエラーメッセージには言語1、4、および6のみが定義されている場合、コピー機能の実行後、生成されるターゲットエラーメッセージは言語1、2、および3のみで存在します。

- 言語を個々のコードとして指定した場合、言語コードの各オカレンスは個別に処理されます。

例えば、ソースエラーメッセージに言語1、2、および3が含まれている場合に言語コードを123に設定し、ターゲットエラーメッセージに言語1、4、および6が含まれている場合に言語コードを同じように123に設定すると、コピー機能の実行後、生成されるターゲットエラーメッセージには、言語1、2、3、4、および6が含まれます。ただし、英語のターゲットエラーメッセージ（言語1）のみは、ソースエラーメッセージの英語のテキストで上書きされます。


- ソースエラーメッセージに1つの言語コードを指定し、ターゲットエラーメッセージに複数の言語コードを指定した場合、コピー機能の実行後、生成されるターゲットエラーメッセージは、ターゲットに指定した最初の言語になります。

## エラーメッセージのダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、エラーメッセージにSYSMAIN機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

-  **重要:** システムエラーメッセージの場合は、*lib-name* として「NATURAL-SYSTEM」または「NATURAL-SYS」を指定します。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- FIND
- LIST
- RENAME
- where-clause
- with-clause

### COPY と MOVE

```
{ COPY } ERROR number [THRU number]  
{ MOVE } FM [LIBRARY] lib-name [where-clause]  
TO [LIBRARY] lib-name [where-clause] [with-clause]
```

例：

```
COPY ERROR 1 FM ACCOUNTS TO ACCOUNTS1 REP WITH TYPE A
```

```
C ERROR 1 THRU 50 FM ACCT WHERE DBID 1 FNR 10 LANG 123456 TO ACCT WHERE DBID 5 FNR  
26 LANG 234567 WITH REP HELP
```

```
MOVE E 200 THRU 210 FM ACCT FNR 10 LANG 123 TO ACCT LANG 123 TYPE S
```

```
M E 376 TYPE E FM ACCT LANG E TO ACCT LANG G
```

### DELETE

```
DELETE ERROR number [THRU number]  
[ IN [LIBRARY] lib-name ] [where-clause] [with-clause]
```

例：

```
DELETE ERROR 1 THRU 10 IN LIBRARY ACCT WHERE DBID 1 FNR 2 PSW GUESS CIPH 137561  
WITH TYPE E MON HELP
```

```
D E 100 IN ACCT
```



**FIND**

```
FIND ERROR number [ IN [LIBRARY] lib-name ] [where-clause] [with-clause]
```

例：

```
FIND E 4280 IN A* MON
```

```
F ERROR 10 IN LIB ACCT WHERE DBID 1 FNR 3 WITH TYPE E
```

**LIST**

```
LIST ERROR number [THRU number] [ IN [LIBRARY] lib-name ] [where-clause] [with-clause]
```

例：

```
LIST E 1 THRU 10 IN ACCT
```

```
L ERROR 100 THRU 150 IN LIB ACCT WHERE DBID 12 FNR 5
```

**RENAME**

1つの（ソース）ライブラリ内のエラーメッセージの範囲の番号を変更する場合、範囲の値が重複しないようにしてください（「[無効な番号範囲の例](#)」を参照）。

```
RENAME ERROR number [THRU number] AS new-number  
[THRU new-number] [with-clause]  
IN [LIBRARY] lib-name [where-clause]  
TO [LIBRARY] lib-name [where-clause]
```

例：

```
RENAME ERR 1 AS 101 IN ACCT
```

```
R ERROR 1 THRU 100 AS 101 THRU 200 IN CLAIMS
```

```
R ERROR 101 THRU 200 AS 1 THRU 100 IN CLAIMS
```

```
RENAME ERROR 1 THRU 50 AS 11 THRU 60 WITH TYPE A REP HELP MON RCOP IN LIBRARY ACCT  
WHERE DBID 1 FNR 2 TO LIB ACCOUNT WHERE FNR 3
```

**無効な番号範囲の例**

番号の範囲が重複しているため、次の例は無効です。

```
R ERROR 1 THRU 100 AS 51 THRU 150 IN CLAIMS
```

R ERROR 101 THRU 200 AS 51 THRU 150 IN CLAIMS

### where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name] [CIPHER cipher]  
[ { PASSWORD } password ] [LANGUAGE language] [SEC  
  PSW ] ( dbid,fnr,password,cipher )]
```

### セパレータ

SEC キーワードに続く各値の間に、または値がない場合に、セパレータとしてコンマを使用する必要があります。例えば、SEC (10,,secret,2a) のように使用します。セッションパラメータ ID (『パラメータリファレンス』ドキュメントの「ID-INPUT 区切り文字」を参照) がコンマに設定されている場合、値を区切るセパレータとしてスラッシュ (/) を使用します。

### with-clause

```
[WITH] [TYPE type] [REPLACE] [RCOP] [ PROMPT ] [ MON ] [HELP]  
[ NOPROMPT ] [ NOMON ]
```

## 92 プロファイル処理

---

- プロファイルメニューのフィールド ..... 574
- プロファイルの選択リスト ..... 576
- プロファイルのダイレクトコマンド構文 ..... 577

検索機能以外のすべてのSYSMAIN機能をデバイスプロフィール、エディタプロフィール、マッププロフィール、およびパラメータプロフィールの各プロフィールのタイプに実行できます。デバイス、エディタ、およびマップの各プロフィールは、プログラムエディタまたはマップエディタで作成されます。パラメータプロフィールは、SYSPARMユーティリティで作成されません。

このセクションでは、プログラミングオブジェクトにSYSMAIN機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

関連トピック：

『エディタ』ドキュメントおよび『[SYSPARM ユーティリティ](#)』ドキュメント

## プロフィールメニューのフィールド

[Profiles] メニューには、プロフィールの処理に必要なすべてのSYSMAIN機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	実行される機能を指定します（「 <a href="#">機能の説明</a> 」を参照）。	
	C	プロフィールをコピーします。
	D	プロフィールを削除します。
	L	プロフィールをリストします。
	M	プロフィールを移動します。
	R	プロフィールの名前を変更します。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	
	Y	Yes。 選択処理がアクティブになります。プロフィールの処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列と行コマンドの詳細については、「 <a href="#">プロフィールの選択リスト</a> 」を参照してください。
	N	No。 選択処理が非アクティブになります。
Name	処理するプロフィールの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」も参照）。	
New Name	名前の変更機能で名前を変更するときに、プロフィールに付ける名前。	
Type	処理するプロフィールのタイプ：	

フィールド	説明	
	D	デバイスプロファイルのみを処理します。
	E	エディタプロファイルのみを処理します。
	M	マッププロファイルのみを処理します。
	P	パラメータプロファイルのみを処理します。
	* または 空フィールド	すべてのプロファイルを処理します。これはデフォルト設定です。
	D、E、M、および P タイプは、任意に組み合わせて使用できます。	
<b>Database</b>	<p>ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。</p> <p>ソースデータベースは、処理対象のプロファイルが保存されているシステムファイルを含みます。ターゲットデータベースは、プロファイルのコピー先または移動先のシステムファイルを含みます。プロファイル名はここで変更されることがあります。</p> <p>有効なデータベース ID は 1~65535 です。</p>	
<b>File</b>	<p>ソースシステムファイルまたはターゲットシステムファイルのファイル番号 (FNR)。</p> <p>有効なファイル番号は 1~65535 です。</p> <p>ソースファイルは、処理対象のプロファイルを含みます。ターゲットファイルは、プロファイルのコピー先または移動先のファイルです。プロファイル名はここで変更されることがあります。</p>	
<b>Name</b> ( [New Name] の下)	<p>VSAM ファイルにのみ適用されます。</p> <p>ソースファイル番号またはターゲットファイル番号の DDNAME/FCT エントリ。</p>	
<b>Replace</b>	<p>移動、コピー、または名前変更機能の使用時にプロファイルを置き換えるかどうかを指定します。</p>	
	Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つプロファイルを置き換えます。
	N	No。 ターゲット環境に存在する名前と同じ名前を持つプロファイルを置き換えません。これはデフォルト設定です。
	「置換オプションの使用」も参照してください。	

## プロファイルの選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべてのプロファイルの選択リストが **[Selection]** 画面に表示されます。

このセクションでは、**[Selection]** 画面に表示される列、およびプロファイルをさらに処理するための行コマンドについて説明します。

- 列
- 行コマンド

### 列

**[Selection]** 画面に次の列が表示されます。

列	説明										
C	行コマンド用の入力フィールド（下記参照）。										
Profile	指定した選択条件に一致するプロファイルの名前。										
Type	プロファイルのタイプ： <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;"></td> <td style="width: 90%;"></td> </tr> <tr> <td>D</td> <td>デバイスプロファイル</td> </tr> <tr> <td>E</td> <td>エディタプロファイル</td> </tr> <tr> <td>M</td> <td>マッププロファイル</td> </tr> <tr> <td>P</td> <td>パラメータプロファイル</td> </tr> </table>			D	デバイスプロファイル	E	エディタプロファイル	M	マッププロファイル	P	パラメータプロファイル
D	デバイスプロファイル										
E	エディタプロファイル										
M	マッププロファイル										
P	パラメータプロファイル										
Message	プロファイルの現在の処理ステータスを示すメッセージ。表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。										

### 行コマンド

次の表に、**[Selection]** 画面の **[C]**（Command）列に入力できる行コマンドを示します。行コマンド E、M、D、および P は、任意の組み合わせで入力できます。

行コマンド	機能
A	デバイスプロファイル、エディタプロファイル、マッププロファイル、またはパラメータプロファイル、あるいはそのすべてを処理します。
D	エディタプロファイル、マッププロファイル、およびパラメータプロファイルが表示されている場合でも、デバイスプロファイルのみを処理します。デバイスプロファイルとして存在しないプロファイルに D を指定した場合、エラーが発生します。

行 コマンド	機能
E	デバイスプロフィール、マッププロフィール、およびパラメータプロフィールが表示されている場合でも、エディタプロフィールのみを処理します。エディタプロフィールとして存在しないプロフィールに E を指定した場合、エラーが発生します。
M	デバイスプロフィール、エディタプロフィール、およびパラメータプロフィールが表示されている場合でも、マッププロフィールのみを処理します。マッププロフィールとして存在しないプロフィールに M を指定した場合、エラーが発生します。
L	パラメータプロフィールの内容を表示します。
P	デバイスプロフィール、エディタプロフィール、およびマッププロフィールが表示されている場合でも、パラメータプロフィールのみを処理します。パラメータプロフィールとして存在しないプロフィールに P を指定した場合、エラーが発生します。

## プロフィールのダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、プロフィールに SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- LIST
- RENAME
- where-clause

- `with-clause`

## COPY と MOVE

```
{ COPY } PROFILE name [with-clause] FM [where-clause] TO [where-clause]  
{ MOVE }
```

例：

```
COPY PRO USER1 TYPE E FM DBID 1 FNR 5 TO DBID 2 FNR 5
```

```
C PROFILE USER TYPE MED REP FM FNR 6 TO FNR 7
```

```
COPY PRO USER1 TYPE P FM DBID 10 FNR 44 TO DBID 3 FNR 7
```

```
MOVE PRO USER1 TYPE E FM DBID 1 FNR 5 TO DBID 2 FNR 5
```

```
M PROFILE USER1 TYPE MED REP FM FNR 6 TO FNR 7
```

## DELETE

```
DELETE PROFILE name [ IN [where-clause] ] [with-clause]
```

例：

```
DELETE PRO U* TYPE DM
```

```
D PRO TEST* IN DBID 177 FNR 205
```

## LIST

```
LIST PROFILE name [ IN [where-clause] ] [with-clause]
```

例：

```
LIST PRO USER* IN DBID 1 FNR 5
```

```
L PRO DT* TYPE E
```



**RENAME**

```

RENAME PROFILE name AS new-name
                IN [where-clause]
                TO [where-clause] [with-clause]

```

例：

```
RENAME PROFILE USER1 AS USER2 RCOP
```

```
R PRO USER1 AS USER2 DBID 1 FNR 4 TO DBID 1 FNR 5
```

```
R PRO USER1 AS NEWUSER IN FNR 4 TO FNR 5 REPLACE RCOP
```

**where-clause**

```

[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name]
[CIPHER cipher] [ { PASSWORD } password ]
                  { PSW }

```

**with-clause**

```

[WITH] [TYPE type] [REPLACE] [RCOP] [ PROMPT ] [ MON ]
      [ NOPROMPT ] [ NOMON ] [HELP]

```



# 93      ルール処理

---

■ ルールメニューのフィールド .....	582
■ ルールの選択リスト .....	584
■ ルールのダイレクトコマンド構文 .....	585

検索機能以外のすべての SYSMAIN 機能を自動ルールおよびフリールールに実行できます。

Predict がインストールされている場合は、ルールの処理に SYSMAIN ではなく Predict を使用することをお勧めします。Natural では、Predict によって提供されたルールに関する一部の情報が処理されません（フォーマットタイプや整合性チェックタイプなど）。

ルール環境の指定は、関連する FDIC システムファイルのデータベース ID (DBID) とファイル番号 (FNR) に常に対応している必要があります。

このセクションでは、プログラミングオブジェクトに SYSMAIN 機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

## ルールメニューのフィールド

[Rules] メニューには、ルールの処理に必要なすべての SYSMAIN 機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	実行される機能を指定します（「 <a href="#">機能の説明</a> 」を参照）。	
	C	ルールをコピーします。
	D	ルールを削除します。
	L	ルールを一覧表示します。
	M	ルールを移動します。
	R	ルールの名前を変更します。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	
	Y	Yes。 選択処理がアクティブになります。ルールの処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列と行コマンドの詳細については、「 <a href="#">ルールの選択リスト</a> 」を参照してください。
	N	No。 選択処理が非アクティブになります。
Name	処理するルールの名前または名前の範囲（「 <a href="#">名前範囲の指定</a> 」も参照）。	
New Name	名前の変更機能で名前を変更するときに、ルールに付ける名前。	
Type	処理するルールのタイプ：	

フィールド	説明									
	A	自動ルールのみを処理します。								
	F	フリールールのみを処理します。								
	AF * または 空フィールド	すべてのルールを処理します。自動ルールまたはフリールール、あるいはその両方 これはデフォルト設定です。								
<b>Database</b>	<p>ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。</p> <p>ソースデータベースは、処理対象のルールが保存されている FDIC システムファイルを含みます。ターゲットデータベースは、ルールのコピー先または移動先の FDIC システムファイルを含みます。ルール名はここで変更されることがあります。</p> <p>有効なデータベース ID は 1~65535 です。</p>									
<b>File</b>	<p>ソースまたはターゲットの FDIC システムファイルのファイル番号 (FNR)。</p> <p>有効なファイル番号は 1~65535 です。</p> <p>ソースファイルは、処理対象のルールを含みます。ターゲットファイルは、ルールのコピー先または移動先のファイルです。ルール名はここで変更されることがあります。</p>									
<b>Name</b> ( [File] の横)	<p>VSAM ファイルにのみ適用されます。</p> <p>ソースファイル番号またはターゲットファイル番号の DDNAME/FCT エントリ。</p>									
<b>Replace</b>	<p>移動、コピー、または名前変更機能の使用時にルールを置き換えるかどうかを指定します。</p> <table border="1" data-bbox="427 1125 1472 1440"> <tr> <td data-bbox="427 1125 792 1167"></td> <td data-bbox="792 1125 1472 1167"></td> </tr> <tr> <td data-bbox="427 1167 792 1283">Y</td> <td data-bbox="792 1167 1472 1283">Yes。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えます。</td> </tr> <tr> <td data-bbox="427 1283 792 1398">N</td> <td data-bbox="792 1283 1472 1398">No。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えません。これはデフォルト設定です。</td> </tr> <tr> <td data-bbox="427 1398 792 1440"></td> <td data-bbox="792 1398 1472 1440"></td> </tr> </table> <p>「置換オプションの使用」も参照してください。</p>				Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えます。	N	No。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えません。これはデフォルト設定です。		
Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えます。									
N	No。 ターゲット環境に存在する名前と同じ名前を持つルールを置き換えません。これはデフォルト設定です。									

## ルールの選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべてのルールの選択リストが [Selection] 画面に表示されます。

このセクションでは、[Selection] 画面に表示される列、およびルールをさらに処理するための行コマンドについて説明します。

- 列
- 行コマンド

### 列

[Selection] 画面に次の列が表示されます。

列	説明
C	行コマンド用の入力フィールド（下記参照）。
Rule Name	指定した選択条件に一致するルールの名前。
Type	ルールのタイプ。Free ルールまたは Auto（自動）ルール。
Ver. Type	整合性チェックタイプ。 Predict で定義された整合性チェックから直接取得されたルールの属性。値「Unknown」は、Predict に使用可能な対応する整合性チェック情報がないことを示します。有効な値の詳細については、『Predict』ドキュメントを参照してください。
Format	フォーマットタイプ。 Predict で定義された整合性チェックから直接取得されたルールの属性。値「Unknown」は、Predict に使用可能な対応する整合性チェック情報がないことを示します。有効な値の詳細については、関連の『Predict』ドキュメントを参照してください。
Message	ルールの現在の処理ステータスを示すメッセージ。表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。

### 行コマンド

次のいずれかの行コマンドを、[Selection] 画面の [C]（コマンド）列に入力できます。

行コマンド	機能
A	指定したルールを処理します。
F	自動ルールにのみ適用されます。  指定したルールが割り当てられたすべての DDM とフィールドが表示されます。

行 コマンド	機能
H	ルールの保存（ソース）オブジェクトのハードコピーを生成します。 指定したルールのソースコードが出力され、画面に表示されます。
I	ルールのディレクトリ情報を表示します。 このコマンドは、システムコマンド LIST DIR と類似しています（『システムコマンド』ドキュメントの「ディレクトリ情報の表示」を参照）。
L	ルールのソースコードを表示します。

## ルールのダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、ルールに SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- LIST
- RENAME
- *where-clause*
- *with-clause*

### COPY と MOVE

```
{ COPY } RULE name [ FM where-clause ] TO where-clause [with-clause]
```

例：

```
C RULE TESTRULE FM FNR 20 TO FNR 24 REPLACE
```

```
COPY R C< FM FNR 20 TO FNR 24
```

```
M RULE TESTRULE FM FNR 20 TO FNR 24 REPLACE
```

```
MOVE RULE C< FM FNR 20 TO FNR 24
```

### DELETE

```
DELETE RULE name [with-clause] [ IN where-clause ]
```

例：

```
D RULE DEMORULE IN DBID 12 FNR 27
```

### LIST

```
LIST RULE name [ IN where-clause ] [with-clause]
```

例：

```
L RULE * DBID 1 FNR 5
```

### RENAME

```
RENAME RULE name AS new-name [ FM where-clause ] [ TO where-clause ] [with-clause]
```

例：

```
RENAME R OLDRULE AS NEWRULE FM DBID 1 FNR 4 TO DBID 1 FNR 5
```

```
R RULE OLDRULE AS NEWRULE FM FNR 4 TO FNR 5 REPLACE RCOP
```

### where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name] [CIPHER cipher]
```

```
[ { PASSWORD } password ]  
[ PSW ]
```

```
[DIC (dbid,fnr,password,cipher)]
```

```
[SEC (dbid,fnr,password,cipher)]
```

### セパレータ

DICおよびSECキーワードに続く各値の間に、または値がない場合に、セパレータとしてコンマを使用する必要があります。例えば、DIC (10,,secret,2a)のように使用します。セッションパラメータ ID (『パラメータリファレンス』ドキュメントの「ID-INPUT 区切り文字」を参



照) がコンマに設定されている場合、値を区切るセパレータとしてスラッシュ (/) を使用します。

### with-clause

```
[WITH] [TYPE type] [REPLACE] [RCOP] [ PROMPT ] [ MON ] [HELP]  
[NOPROMPT] [NOMON]
```



# 94 DL/I サブファイル処理

---

- DL/I サブファイルメニューのフィールド ..... 590
- DL/I サブファイルの選択リスト ..... 591
- DL/I サブファイルのダイレクトコマンド構文 ..... 592

DL/I 用の Natural がインストールされている場合、検索機能と名前変更機能以外のすべての SYSMAIN 機能を DL/I サブファイルに実行できます。

DL/I サブファイルには以下のものが含まれています。

- NDB = Natural に定義された DL/I DBD（データベース記述）。
- NSB = Natural に定義された DL/I PSB（プログラム指定ブロック）。
- UDF = Natural に定義された DL/I ユーザー定義フィールド。

NDB、NSB、DBD、PSB、および UDF の詳細については、『DL/I用のNatural』ドキュメントを参照してください。

このセクションでは、プログラミングオブジェクトに SYSMAIN 機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

## DL/I サブファイルメニューのフィールド

「DL/I Subfiles」メニューには、DL/I サブファイルの処理に必要なすべての SYSMAIN 機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	実行される機能を指定します（「 <a href="#">機能の説明</a> 」を参照）。	
	C	サブファイルをコピーします。
	D	サブファイルを削除します。
	L	サブファイルを一覧表示します。
	M	サブファイルを移動します。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	
	Y	Yes。 選択処理がアクティブになります。DL/I サブファイルの処理時に選択リストが表示されます。これはデフォルト設定です。選択リストに表示される列、フィールド、および行コマンドの詳細については、「 <a href="#">DL/I サブファイルの選択リスト</a> 」を参照してください。
	N	No。 選択処理が非アクティブになります。
Name	処理する DL/I サブファイルの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」を参照）。	
Type	処理する DL/I サブファイルのタイプ。次のいずれかのタイプを指定する必要があります。	

フィールド	説明	
	D	NDB と UDF のみを処理します。これはデフォルト設定です。
	P	NSB のみを処理します。
<b>Database</b>	<p>ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。</p> <p>ソースデータベースは、処理対象の DL/I サブファイルが保存されている FDIC システムファイルを含みます。ターゲットデータベースは、DL/I サブファイルのコピー先または移動先の FDIC システムファイルを含みます。</p> <p>有効なデータベース ID は 1~65535 です。</p>	
<b>File</b>	<p>ソースまたはターゲットの FDIC システムファイルのファイル番号 (FNR)。</p> <p>有効なファイル番号は 1~65535 です。</p> <p>ソースファイルは、処理対象の DL/I サブファイルを含みます。ターゲットファイルは、DL/I サブファイルのコピー先または移動先のファイルです。</p>	
<b>Name</b> ( [File] の横)	<p>VSAM ファイルにのみ適用されます。</p> <p>ソースファイル番号またはターゲットファイル番号の DDNAME/FCT エントリ。</p>	
<b>Replace</b>	<p>移動機能またはコピー機能の使用時に DL/I サブファイルを置き換えるかどうかを指定します。</p>	
	Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つ DL/I サブファイルを置き換えます。
	N	No。 ターゲット環境に存在する名前と同じ名前を持つ DL/I サブファイルを置き換えません。これはデフォルト設定です。
	「置換オプションの使用」も参照してください。	

## DL/I サブファイルの選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべての DL/I サブファイルの選択リストが [Selection] 画面に表示されます。

このセクションでは、[Selection] 画面に表示される列とフィールド、および DL/I サブファイルをさらに処理するための行コマンドについて説明します。

### ■ 列とフィールド

## ■ 行コマンド

### 列とフィールド

次の列とフィールドが [Selection] 画面に表示されます。

列／フィールド	説明
C	行コマンド用の入力フィールド（下記参照）。
Subfile	指定した選択条件に一致する DL/I サブファイルの名前。
Message	DL/I サブファイルの処理ステータスを示すメッセージ。表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。
Listed Subfiles	DL/I サブファイルのタイプ。NDB（UDF を含む）または NSB。

### 行コマンド

次のいずれかの行コマンドを、[Selection] 画面の [C]（コマンド）列に入力できます。

行コマンド	機能
A	NSB サブファイルまたは NDB サブファイルと UDF サブファイルを処理します。 行コマンド A は、リスト機能では使用できません。
L	NSB サブファイルまたは NDB サブファイルと UDF サブファイルをリストします。

## DL/I サブファイルのダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、DL/I サブファイルに SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE と LIST
- where-clause

- `with-clause`

## COPY と MOVE

```
{ COPY } DL1 name [with-clause] [ FM where-clause ] TO where-clause
{ MOVE }
```

例：

```
COPY DL1 SUBFILE1 TYPE D FM DBID 1 FNR 5 TO DBID 2 FNR 5
```

```
C DL1 SUBFILE REP FM FNR 6 TO FNR 7 TYPE D
```

```
COPY DL1 SUBFILE1 TYPE P TO DBID 3 FNR 7
```

```
MOVE DL1 SUBFILE1 TYPE D FM DBID 1 FNR 5 TO DBID 2 FNR 5
```

```
M DL1 SUBFILE1 REP FM FNR 6 TO FNR 7 TYPE D
```

## DELETE と LIST

```
{ DELETE } DL1 name [ IN where-clause ] [with-clause]
{ LIST }
```

例：

```
DELETE DL1 S* TYPE D
```

```
D DL1 TEST* IN DBID 177 FNR 205 TYPE D
```

```
LIST DL1 SUBFILE* IN DBID 1 FNR 5 TYPE D
```

```
L DL1 SF* TYPE P
```

## where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name]
[CIPHER cipher] [ { PASSWORD } password ]
                  [ PSW ]
```

**with-clause**

[WITH] [TYPE *type*] [REPLACE] [RCOP] [ PROMPT  
NOPROMPT ] [ MON  
NOMON ] [HELP]



# 95 DDM 処理

---

■ DDM メニューのフィールド .....	596
■ DDM の選択リスト .....	598
■ DDM のダイレクトコマンド構文 .....	599

検索機能と名前の変更機能以外のすべての SYSMAIN 機能を DDM に実行できます。

DDM 環境の指定は、関連する FDIC システムファイルのデータベース ID (DBID) とファイル番号 (FNR) に常に対応している必要があります。

このセクションでは、プログラミングオブジェクトに SYSMAIN 機能を実行するためのメニュー機能と選択リストオプション、およびダイレクトコマンドの使用時に適用される構文について説明します。

## DDM メニューのフィールド

[DDM] メニューには、DDM の処理に必要なすべての SYSMAIN 機能が含まれています。次の表に、メニューで表示されるフィールドとその説明を示します。

フィールド	説明	
Code	実行される機能を指定します (「 <a href="#">機能の説明</a> 」を参照)。	
	C	DDM のコピー。
	D	DDM の削除。
	M	DDM を移動します。
Sel. List	選択処理と自動処理のどちらを使用するかを指定します。	
	Y	Yes。 選択処理がアクティブになります。DDM の処理時に選択リストが表示されます。これはデフォルト設定です。  選択リストに表示される列と行コマンドの詳細については、「 <a href="#">DDM の選択リスト</a> 」を参照してください。
N	No。 選択処理が非アクティブになります。	
Name	処理する DDM の名前または名前の範囲 (「 <a href="#">名前の範囲の指定</a> 」も参照)。	
DDM DBID	このフィールドを使用して、指定したデータベース ID (DBID) でカタログ化された DDM のみを選択できます。  有効なデータベース ID は 1~65535 です。  このフィールドを空にするか、または 0 (ゼロ) に設定した場合、DBID は選択条件として使用されません。	

フィールド	説明				
<b>DDM FNR</b>	<p>このフィールドを使用して、指定したファイル番号（FNR）でカタログ化された DDM のみを選択できます。</p> <p>有効なファイル番号は 1～65535 です。</p> <p>このフィールドを空にするか、または 0（ゼロ）に設定した場合、FNR は選択条件として使用されません。</p>				
<b>Database</b>	<p>ソースデータベースまたはターゲットデータベースのデータベース ID（DBID）。</p> <p>ソースデータベースは、処理対象の DDM が保存されている FDIC システムファイルを含みます。ターゲットデータベースは、DDM のコピー先または移動先の FDIC システムファイルを含みます。</p> <p>有効なデータベース ID は 1～65535 です。</p>				
<b>File</b>	<p>ソースまたはターゲットの FDIC システムファイルのファイル番号（FNR）。</p> <p>有効なファイル番号は 1～65535 です。</p> <p>ソースファイルは、処理対象の DDM を含みます。ターゲットファイルは、DDM のコピー先または移動先のファイルです。</p>				
<b>Name</b> ( [File] の横)	<p>VSAM ファイルにのみ適用されます。</p> <p>ソースファイル番号またはターゲットファイル番号の DDNAME/FCT エントリ。</p>				
<b>Replace</b>	<p>移動機能またはコピー機能の使用時に DDM を置き換えるかどうかを指定します。</p> <table border="1"> <tr> <td>Y</td> <td>Yes。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えます。</td> </tr> <tr> <td>N</td> <td>No。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えません。これはデフォルト設定です。</td> </tr> </table> <p>「置換オプションの使用」も参照してください。</p>	Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えます。	N	No。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えません。これはデフォルト設定です。
Y	Yes。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えます。				
N	No。 ターゲット環境に存在する名前と同じ名前を持つ DDM を置き換えません。これはデフォルト設定です。				
<b>Del.NSC-Def.</b>	<p>このフィールドは、Natural Security 環境でのみ表示されます。</p> <p>Natural Security 定義の削除を示します。</p> <p>DDM をソース環境から削除したり、新しい環境に移動して異なる FSEC システムファイルを指定した場合、このフィールドを使用して、ソース FSEC の DDM 定義を削除するかどうかを指定できます。</p> <p>可能な値は次のとおりです。</p> <table border="1"> <tr> <td>Y</td> <td>Yes。 ソース FSEC の DDM 定義を削除します。</td> </tr> </table>	Y	Yes。 ソース FSEC の DDM 定義を削除します。		
Y	Yes。 ソース FSEC の DDM 定義を削除します。				

フィールド	説明
	N No。 ソース FSEC の DDM 定義を保持します。これはデフォルト設定です。

## DDM の選択リスト

選択処理をアクティブにした場合、指定した選択条件に一致するすべての DDM の選択リストが [Selection] 画面に表示されます。

このセクションでは、[Selection] 画面に表示される列、および DDM をさらに処理するための行コマンドについて説明します。

- 列
- 行コマンド

### 列

[Selection] 画面に次の列が表示されます。

列	説明
C	行コマンド用の入力フィールド（下記参照）。
DDM Name	指定した選択条件に一致する DDM の名前。
DBID	DDM がカタログ化されたデータベース ID (DBID)。  DBID が表示されないか、または 0 (ゼロ) が表示される場合、DDM は現在の DBID でカタログ化されました。
FNR	DDM がカタログ化されたファイル番号 (FNR)。
Cat Date	DDM がカタログ化された日付。
Time	DDM がカタログ化された時刻。
User ID	DDM をカタログ化したユーザーの ID。
Message	DDM の処理ステータスを示すメッセージ。  この列は、DDM に行コマンド A を実行した後にのみ表示されます。  表示されるメッセージについては、「 <a href="#">ステータスメッセージ</a> 」を参照してください。

このタイプの情報を提供しない古いバージョンの Natural で DDM がカタログ化された場合、[Cat Date]、[Time]、および [User ID] 列は空となります。

## 行コマンド

次のいずれかの行コマンドを、[Selection] 画面の [C] (コマンド) 列に入力できます。


行 コマンド	機能
A	指定した DDM を処理します。  この行コマンドは、リスト機能では使用できません。
L	指定した DDM を一覧表示します。  表示されるフィールド定義の詳細については、『SYSDDM ユーティリティ』ドキュメントの「フィールド属性列」を参照してください。
R	指定した DDM にリンクされたすべての自動ルールを一覧表示します。  行コマンド R では、[Rule Usage for DDM/Fields] 画面を呼び出します。この画面には、DDM でルールを使用するかどうかを示されます。ルールを使用する場合、ルールの名前および参照される DDM フィールドが画面に表示されます。
X	Natural Connection と Entire Connection がインストールされている場合にのみ適用されます。  DDM を PC にダウンロードします。ダウンロード方法の詳細については、「 <a href="#">PC へのソースオブジェクトのダウンロード</a> 」を参照してください。

## DDM のダイレクトコマンド構文

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用して、DDM に SYSMAIN 機能を実行する場合に適用される構文について説明します。ダイレクトコマンドの一般的な使用方法については、「[コマンドの実行](#)」を参照してください。

次の構文図で使用されているキーワードと変数値については、「[ダイレクトコマンドのキーワードと変数](#)」を参照してください。構文図の記号は、システムコマンドで使用される構文記号に対応します。これらの記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

*where-clause* と *with-clause* の構文は、どのコマンドでも同じです。

 **注意:** 互換性の理由から、DDM のダイレクトコマンドでは、キーワード DDM の代わりにキーワード VIEW (または短縮形の V) を使用できます。

このセクションでは、次のトピックについて説明します。

- COPY と MOVE
- DELETE
- LIST

- where-clause
- with-clause

## COPY と MOVE

```
{ COPY } DDM name [ FM where-clause ] TO where-clause [with-clause]
{ MOVE }
```

例：

```
C DDM PERSONNEL FM FNR 20 TO FNR 24 REPLACE
```

```
COPY DDM C< FM FNR 20 TO FNR 24
```

```
M DDM PERSONNEL FM FNR 20 TO FNR 24 REPLACE
```

```
MOVE DDM C< FM FNR 20 TO FNR 24
```

## DELETE

```
DELETE DDM name [with-clause] [ IN where-clause ]
```

例：

```
D DDM FINANCE IN DBID 12 FNR 27
```

## LIST

```
LIST DDM name [ IN where-clause ] [with-clause]
```

例：

```
L DDM * IN DBID 1 FNR 5
```

## where-clause

```
[WHERE] [DBID dbid] [FNR fnr] [NAME vsam-name] [CIPHER cipher]
[ { PASSWORD } password ]
[ { PSW } ]
[DIC (dbid,fnr,password,cipher)]
[SEC (dbid,fnr,password,cipher)]
```

## セパレータ

DIC および SEC キーワードに続く各値の間に、または値がない場合に、セパレータとしてコンマを使用する必要があります。例えば、DIC (10,,secret,2a) のように使用します。セッションパラメータ ID (『パラメータリファレンス』ドキュメントの「ID-INPUT 区切り文字」を参照) がコンマに設定されている場合、値を区切るセパレータとしてスラッシュ (/) を使用します。

**with-clause**

```
[WITH] [ { DDMDBID } DDM-dbid ] [ { DDMFNR } DDM-fnr ] [REPLACE] [ PROMPT ] [ MON ]
[ DDBID ] [ DFNR ] [ NOPROMPT ] [ NOMON ] [HELP]
```





# 96      ダイレクトコマンドのキーワードと変数

---

- キーワードの説明 ..... 604
- 名前の範囲の指定 ..... 615

このセクションでは、オンラインモードまたはバッチモードでダイレクトコマンドを使用する場合に関連するすべてのキーワードと変数について説明します。各キーワードは、オブジェクトの選択条件を指定するため、または実行中のコマンドのオプションを設定するために使用するパラメータを表します。必要な場合、変数値をキーワードと一緒に指定する必要があります。

次に示す構文図で使用されている記号は、システムコマンドで使用される構文記号に対応します。これらの構文記号の詳細については、『システムコマンド』ドキュメントの「システムコマンド構文」を参照してください。

このセクションで説明する *where-clause* と *with-clause* の詳細を含めて、キーワードが参照するダイレクトコマンド構文の詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションを参照してください。

## キーワードの説明

このセクションでは、ダイレクトコマンドで使用するキーワードと、必要に応じて、対応する変数値について説明します。

キーワードはアルファベット順に示しています。斜体の文字はキーワードで指定する必要がある変数値です。変数値ごとに Natural データのフォーマットと長さが示されています。

キーワード	値	Natural データ フォー マツト ／長さ	説明
ALL または A	<i>name</i>	A9	プログラミングオブジェクトにのみ適用されます。 処理するオブジェクトの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」も参照）。保存（ソース）オブジェクトおよびカタログ化オブジェクトのいずれかまたは両方が処理されます。
AS	<i>new-name</i>	A8 または A12	DL/I サブファイルおよび DDM には適用されません。 <b>RENAME</b> コマンドで名前を変更するときに、オブジェクトに付ける新しい名前。 フォーマット／長さ A12 は、デバッグ環境にのみ適用されます。
	<i>new-number</i>	N4	エラーメッセージの場合 エラーメッセージに割り当てる新しい番号、または <b>RENAME</b> コマンドを使用する場合に既存のエラーメッセージの範囲に割り当てる新しい番号の範囲の開始番号。

キーワード	値	Natural データ フォー マット ／長さ	説明
CATALOGED	<i>name</i>	A9	プログラミングオブジェクトにのみ適用されます。  処理するカタログ化オブジェクトの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」を参照）。
CIPHER	<i>cipher</i>	A8	<i>where-clause</i> で使用されるソースファイルまたはターゲットシステムファイル、あるいはその両方の Adabas サイファコード。  ルールおよび DDM の場合、CIPHER の代わりに対応する <b>DIC</b> 指定を使用できます。 <i>cipher</i> を 2 回指定した場合は、最後に指定したものが使用されます。
DBID	<i>dbid</i>	N5	ソースデータベースまたはターゲットデータベースのデータベース ID (DBID)。  ソースデータベースは、処理対象のオブジェクトが保存されているシステムファイルを含みます。ターゲットデータベースは、オブジェクトのコピー先または移動先のシステムファイルを含みます。必要に応じて、オブジェクト名はここで変更されます。エラーメッセージの場合は、番号が変更されます。  有効な DBID は 1～65535 です。  DBID またはファイル番号 ( <b>FNR</b> ) を指定せず、SYSMAIN をシステムコマンド SYSMAIN またはサブプログラム MAINUSER で呼び出した場合（「 <a href="#">アプリケーションプログラミングインターフェイスでのSYSMAINの呼び出し</a> 」も参照）、次の処理が適用されます。SYSMAIN を呼び出したシステムファイルの DBID と FNR が常に使用されます。例：FUSER システムファイルに含まれているライブラリから SYSMAIN を入力した場合は、このシステムファイルの DBID と FNR が使用されます。  ルールおよび DDM の場合、DBID の代わりに対応する <b>DIC</b> 指定を使用できます。 <i>dbid</i> を 2 回指定した場合は、最後に指定したものが使用されます。
DDM または VIEW	<i>name</i>	A32	DDM にのみ適用されます。  処理する DDM の名前または名前の範囲。「 <a href="#">名前の範囲の指定</a> 」も参照してください。
DDMDBID または DDBID	<i>dgm-dbid</i>	N5	DDM にのみ適用されます。  DDM データベース ID (DBID)：指定した DBID でカタログ化されたすべての DDM が処理されます。  有効な DDM DBID は 0～65535 です。値を指定しない、または 0（ゼロ）を指定した場合、DDM DBID はチェックされません。

## ダイレクトコマンドのキーワードと変数

キーワード	値	Natural データ フォー マット ／長さ	説明
DDMFNR または DFNR	<i>dmd-fnr</i>	N5	DDM にのみ適用されます。  DDM ファイル番号 (FNR) : 指定した FNR でカタログ化されたすべての DDM が処理されます。  有効な DDM FNR は 0~65535 です。値を指定しない、または 0 (ゼロ) を指定した場合、DDM FNR はチェックされません。
DEBUG	<i>name</i>	A12	デバッグ環境にのみ適用されます。  処理するデバッグ環境の名前または名前の範囲。「 <a href="#">名前の範囲の指定</a> 」も参照してください。
DIC	<i>dbid</i> <i>fnr</i> <i>password</i> <i>cipher</i>	A80	エラーメッセージ、プロファイル、および DL/I サブファイルには適用されません。  データベース ID ( <i>dbid</i> )、ファイル番号 ( <i>fnr</i> )、Adabas パスワード ( <i>password</i> )、Adabas サイファコード ( <i>cipher</i> ) など、FDIC ソースシステムファイルとターゲットシステムファイルのいずれかまたは両方の環境を指定します。  ルールおよび DDM の場合、対応する DIC 指定の代わりに DBID、FNR、CIPHER、および PASSWORD 指定を使用できます。またはその反対を使用できます。項目を 2 回指定した場合は、最後に指定したものが使用されます。
DL1 または SUBFILES または S	<i>name</i>	A8	DL/I サブファイルにのみ適用されます。  処理する DL/I サブファイルの名前または名前の範囲。「 <a href="#">名前の範囲の指定</a> 」も参照してください。
ERROR	<i>number</i>	N4	エラーメッセージにのみ適用されます。  処理するエラーメッセージの番号、または <a href="#">THRU</a> を指定した場合は、番号の範囲の開始番号。
EXTEND	—	—	バッチモードで使用している場合、プログラミングオブジェクトおよび <a href="#">LIST</a> または <a href="#">FIND</a> コマンドにのみ適用されます。  EXTEND を指定しない場合、指定したライブラリに含まれるオブジェクトの短いリストが表示されます。短いリストには、オブジェクトの名前とタイプが含まれ、ソースオブジェクトまたはカタログ化オブジェクト、あるいはその両方が存在するかどうかを示されます。  EXTEND を指定した場合、指定したライブラリに含まれるオブジェクトの拡張リストが表示されます。EXTEND を指定しない場合に表示される情報に加えて、拡張リストには、オブジェクトディレクト

キーワード	値	Natural データ フォー マット ／長さ	説明
			りからの次の情報が表示されます。プログラミングモード、Naturalバージョン、ユーザーID、保存／カタログ化日時、およびオブジェクトのソース（存在する場合）。
FROM または FM または IN	<i>lib-name</i>	A8	プログラミングオブジェクト、デバッグ環境、およびエラーメッセージの場合、ソースライブラリを指定します。  プロファイル、ルール、DDM、および DL/I サブファイルの場合は、 <i>where-clause</i> を導入します。
FMDATE または FMDD	<i>date-from</i>	A10	プログラミングオブジェクトにのみ適用されます。  期間の開始日。指定した日付以降に保存またはカタログ化されたすべてのオブジェクトが処理されます。  終了日を <b>TODATE</b> で指定しない場合、指定した日付以降のすべてのオブジェクトが処理対象として選択されます。  SYSMAIN メニュー画面の右上に示されているように、日付はプロファイルパラメータ DTFORM の設定に従って指定する必要があります（『パラメータリファレンス』ドキュメントの「DTFORM - 日付フォーマット」を参照）。デフォルト設定は、2005-08-20 のように、国際標準フォーマットの YYYY-MM-DD（YYYY=年、MM=月、DD=日）です。
FMTIME または FMTM または FMTT	<i>time-from</i>	A5	<b>FMDATE</b> を指定した場合、プログラミングオブジェクトにのみ適用されます。  開始日を指定します。指定した日時以降に保存またはカタログ化されたすべてのオブジェクトが処理されます。  時刻は、11:33 のように、HH:II（HH=時間、II=分）の形式で指定する必要があります。
FNR	<i>fnr</i>	N5	ソースシステムファイルまたはターゲットシステムファイル（FNAT、FDIC、または FUSER）のファイル番号（FNR）。  ソースシステムファイルは、処理対象のオブジェクトを含みます。ターゲットシステムファイルは、オブジェクトのコピー先または移動先のシステムファイルです。必要に応じてオブジェクト名はここで変更されます。エラーメッセージの場合は番号が変更されます。  有効な FNR は 1～65535 です。  データベース ID（DBID）または FNR を指定せず、システムコマンド SYSMAIN またはサブプログラム MAINUSER で SYSMAIN を呼び出した場合（「 <a href="#">アプリケーションプログラミングインターフェイスでのSYSMAINの呼び出し</a> 」も参照）、次の処理が適用されま

## ダイレクトコマンドのキーワードと変数

キーワード	値	Natural データ フォー マット /長さ	説明
			<p>す。SYSMAIN を呼び出したシステムファイルの DBID と FNR が常に使用されます。例：FUSER システムファイルに含まれているライブラリから SYSMAIN を入力した場合は、このシステムファイルの DBID と FNR が使用されます。</p> <p>ルールおよび DDM の場合、FNR の代わりに対応する DIC 指定を使用できます。fnr を 2 回指定した場合は、最後に指定したものが使用されます。</p>
HELP または ?	—	—	<p>オンライン選択処理をアクティブにします。</p> <p>キーワード HELP を <i>with-clause</i> に含めるか、または疑問符 (?) をオブジェクト名の最後の文字として入力できます。</p>
LANGUAGE	<i>language</i>	A9	<p>エラーメッセージにのみ適用されます。</p> <p>処理するエラーメッセージの言語のコード。</p> <p>言語コードを任意に組み合わせて、言語を指定できます。言語に割り当てられている言語コードについては、『システム変数』ドキュメントの「*LANGUAGE」で「言語コードの割り当て」を参照してください。「言語の指定」も参照してください。</p> <p>アスタリスク (*) を使用して、エラーメッセージのすべての既存の言語を処理対象に選択できます。</p>
LIBRARY または APPLIC または APL	<i>lib-name</i>	A8	<p>プログラミングオブジェクト、デバッグ環境、およびエラーメッセージにのみ適用されます。</p> <p>ソースライブラリまたはターゲットライブラリの名前を示すオプションキーワード。キーワードと各値を省略すると、SYSMAIN を呼び出す前にログオンしたライブラリが処理に使用されます。</p> <p>ソースライブラリは、処理対象のオブジェクトを含みます。ターゲットライブラリは、オブジェクトのコピー先または移動先のライブラリです。オブジェクト名はここで変更されることがあります。エラーメッセージの場合は、番号が変更されます。</p> <p>システムエラーメッセージの場合は、<i>lib-name</i> として「NATURAL-SYSTEM」または「NATURAL-SYS」を指定します。</p> <p><i>lib-name</i> は、FROM および TO キーワードの直後に指定する必要があります。オプションのキーワード LIBRARY を使用する場合は、FROM または TO と <i>lib-name</i> の間に入力する必要があります。</p>
MON または NOMON	—	—	<p>バッチモードでは適用されません。</p> <p>SYSMAIN での現在の動作の追跡をアクティブ (MON) あるいは非アクティブ (NOMON または MONOFF) にします。処理中は、読み取</p>

キーワード	値	Natural データ フォー マット ／長さ	説明
または MONOFF			り、削除、更新、追加を実行中のオブジェクトに関する情報、およびエラーが発生したかどうかユーザーに通知されます。プログラミングオブジェクトでは、XRef データでの処理に関する情報も通知されます。この機能は、非会話型モードで実行できる TP 環境でのみ有効です。
NAME	<i>vsam-name</i>	A8	ソースファイル番号またはターゲットファイル番号の DDNAME/FCT エントリ。
PROMPT または NOPROMPT	—	—	バッチモードでは適用されません。  SYSMAIN プロンプトを有効 (PROMPT) または無効 (NOPROMPT) にします。NOPROMPT の場合は、確認画面は表示されません。例えば、何かを削除する前に、SYSMAIN はユーザーに確認を求めます。
PASSWORD ま たは PSW	<i>password</i>	A8	<i>where-clause</i> で使用されるソースファイルまたはターゲットシステムファイル、あるいはその両方の Adabas パスワード。  ルールおよび DDM の場合、PASSWORD の代わりに対応する DIC 指定を使用できます。 <i>password</i> を 2 回指定した場合は、最後に指定したものが使用されます。
PROFILE	<i>name</i>	A8	プロファイルにのみ適用されます。  処理するプロファイルの名前または名前の範囲。「 <a href="#">名前の範囲の指定</a> 」も参照してください。
RCOP	—	—	名前を変更するオブジェクトのコピーを作成することを指定します。
REPLACE	—	—	<i>with-clause</i> で使用されている置換オプションをアクティブ化します。  ターゲット環境内の同じ名前のオブジェクトが、処理されるオブジェクトで置き換えられます。  <b>注意:</b> プログラミングオブジェクトを置き換えた場合、そのプログラミングオブジェクトは Natural バッファプールからも削除されます。Predict がインストールされている場合は、既存のクロスリファレンスレコードもすべて削除されます。
RULE	<i>name</i>	A32	ルールにのみ適用されます。  処理するルールの名前または名前の範囲。「 <a href="#">名前の範囲の指定</a> 」も参照してください。

## ダイレクトコマンドのキーワードと変数

キーワード	値	Natural データ フォー マット /長さ	説明
SAVED	<i>name</i>	A9	プログラミングオブジェクトにのみ適用されます。  処理する保存（ソース）オブジェクトの名前または名前の範囲。 「 <a href="#">名前の範囲の指定</a> 」も参照してください。
SEC	<i>dbid</i> <i>fnr</i> <i>password</i> <i>cipher</i>	A80	プロファイルおよび DL/I サブファイルには適用されません。  データベース ID ( <i>dbid</i> )、ファイル番号 ( <i>fnr</i> )、Adabas パスワード ( <i>password</i> )、Adabas サイファコード ( <i>cipher</i> ) など、FSEC ソースシステムファイルとターゲットシステムファイルのいずれかまたは両方の環境を指定します。
SETNO	<i>set-number</i>	N2	LIST XREF コマンドの Predict XRef セット保存オプションを使用して作成された保有 Predict セットの数。このセットに含まれるオブジェクトにすべての SYSMAIN 処理機能を適用できます。  有効な番号を指定すると、SYSMAIN で Predict セットであると見なされます。番号を指定しないと、通常のオブジェクト処理と見なされます。
SETLIBRARY	<i>set-library</i>	A8	Predict ファイルのセキュリティの一部として Predict セットのライブラリ指定を上書きするオプションをアクティブにします。  SETLIBRARY は、SETNO に有効な番号が指定された場合にのみ評価されます。
SETUSER	<i>set-user</i>	A8	Predict ファイルのセキュリティの一部として Predict セットのユーザー ID 指定を上書きするオプションをアクティブにします。  SETUSER は、SETNO に有効な番号が指定された場合にのみ評価されます。
STOWED または BOTH	<i>name</i>	A9	プログラミングオブジェクトにのみ適用されます。  保存（ソース）オブジェクトおよびカタログ化オブジェクトが処理されるオブジェクトの名前または名前の範囲（「 <a href="#">名前の範囲の指定</a> 」も参照）。保存オブジェクトおよびカタログ化オブジェクトの両方として存在するオブジェクトのみが処理されます。  この例外はコピーコード、テキスト、および記録で、いずれもカタログ化できません。ただし、このオプションを指定すると処理対象になります。
THRU	<i>number</i> または <i>new-number</i>	N4	エラーメッセージにのみ適用されます。  開始番号を AS で指定した場合は、処理するエラーメッセージ番号の範囲の終了番号。



キーワード	値	Natural データ フォー マット ／長さ	説明
TID	<i>terminal-ID</i>	A8	プログラミングオブジェクトにのみ適用されます。  端末 ID。指定した端末上で保存またはカタログ化されたすべてのオブジェクトが処理されます。
T0	<i>lib-name</i>	A8	プログラミングオブジェクト、デバッグ環境、およびエラーメッセージの場合、 ターゲットライブラリを指定します。  プロファイル、ルール、DDM、および DL/I サブファイルの場合は、 <i>where-clause</i> を導入します。
TODATE または TODD	<i>date-to</i>	A10	プログラミングオブジェクトにのみ適用されます。  期間の終了日。指定した日付以前に保存またはカタログ化されたすべてのオブジェクトが処理されます。開始日は <b>FMDATE</b> で指定できます。  SYSMAIN メニュー画面の右上に示されているように、日付はプロファイルパラメータ DTFORM の設定に従って指定する必要があります（『パラメータリファレンス』ドキュメントの「DTFORM - 日付フォーマット」を参照）。デフォルト設定は、2005-08-20 のように、国際標準フォーマットの YYYY-MM-DD（YYYY=年、MM=月、DD=日）です。
TOTIME または TOTT または TOTM	<i>time-to</i>	A5	<b>TODATE</b> を指定した場合、プログラミングオブジェクトにのみ適用されます。  期間の終了時間。指定した日時以前にカタログ化または保存されたすべてのオブジェクトが処理されます。  時刻は、11:33 のように、HH:II（HH=時間、II=分）の形式で指定する必要があります。
TYPE	<i>type</i>	—	次の「 <b>TYPE の指定</b> 」で示すように、処理対象のプログラミングオブジェクト、エラーメッセージ、プロファイル、ルール、または <b>DL/I サブファイル</b> のタイプ。
USER または USR	<i>user-id</i>	A8	プログラミングオブジェクトにのみ適用されます。  ユーザー ID。指定したユーザーによって保存またはカタログ化されたすべてのオブジェクトが処理されます。
WHERE	<i>where-clause</i>	—	<i>where-clause</i> の開始を示すオプションキーワード。  <i>where-clause</i> は常に、FROM または T0 キーワードおよび該当する場合は <i>lib-name</i> の後に入力する必要があります。節内のキーワードと値は任意の順序で指定できます。

キーワード	値	Natural データ フォー マット /長さ	説明	
			詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションでダイレクトコマンド構文を参照してください。	
WITH	<i>with-clause</i>	—	<p><i>with-clause</i> の開始を示すオプションキーワード。</p> <p><i>with-clause</i> のキーワードと値は任意の順序で指定でき、<i>with-clause</i> はダイレクトコマンド文字列内の最初の 3 つの位置を除く任意の位置に配置できます。</p> <p>詳細については、『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションでダイレクトコマンド構文を参照してください。</p>	
XREF	F または N または S または Y	A1	プログラミングオブジェクトにのみ適用されます。	
			Predict システムファイルに保存されたクロスリファレンス (XRef) データが処理されることを示します。	
			次の値のいずれかを指定できます。	
			F	すべての XRef データが処理されます。オブジェクトは Predict に記録される必要があります。
			N	DELETE コマンドを使用する場合を除いて、XRef データは処理されません。カタログ化オブジェクトを削除または置換すると、このオブジェクトの既存の XRef データはすべて SYSMAIN により必ず削除されます。
			S	XRef データがあるかどうかに関係なく、指定したオブジェクトが処理されます。
			Y	すべての XRef データが処理されます。
			詳細については、「 <a href="#">XRef の考慮事項</a> 」を参照してください。	

## TYPE 指定 - プログラミングオブジェクト

**Natural** データフォーマット / 長さ : **A20**

次の表に、プログラミングオブジェクトの有効なオブジェクトタイプコードをすべて示します。

コード	オブジェクトタイプ
P	プログラム
N	サブプログラム
S	サブルーチン
M	マップ
H	ヘルプルーチン
0	ISPF マクロ
3	ダイアログ
5	プロセッサ
A	パラメータデータエリア
G	グローバルデータエリア
L	ローカルデータエリア
C	コピーコード
T	テキスト
R	レポート
Z	記録
4	クラス
7	ファンクション
8	アダプタ
9	リソース
*	すべてのプログラミングオブジェクトタイプ

### TYPE 指定 - エラーメッセージ

**Natural** データフォーマット／長さ：A1

次の表に、エラーメッセージの有効なタイプコードをすべて示します。

コード	タイプ
S	ショートエラーメッセージ
E	拡張（ロング）エラーメッセージ
A	すべてのエラーメッセージタイプ。ショートメッセージまたは拡張メッセージ、あるいはその両方

### TYPE 指定 - プロファイル

**Natural** データフォーマット／長さ：A3

次の表に、プロファイルの有効なタイプコードをすべて示します。

コード	タイプ
E	エディタプロファイル
D	デバイスプロファイル
M	マッププロファイル
P	パラメータプロファイル
*	すべてのプロファイルタイプ。

### TYPE 指定 - ルール

**Natural** データフォーマット／長さ：A2

次の表に、ルールの有効なタイプコードをすべて示します。

コード	タイプ
A	自動ルール
F	フリールール
AF	すべてのルールタイプ。自動ルールまたはフリールール、あるいはその両方

## TYPE 指定 - DL/I サブファイル

**Natural** データフォーマット／長さ：A1

次の表に DL/I サブファイルの有効なタイプコードをすべて示します。

コード	タイプ
D	NDB と UDF
P	NSB

## 名前の範囲の指定

すべての SYSMAIN 機能には、処理されるオブジェクトについて名前または名前の範囲のいずれかを指定するオプションが用意されています。また、メニューモードでは、[Find Selection] 画面または [List Selection] 画面で、名前や名前の範囲を指定して、表示されるオブジェクトの数を制限できます。「選択リストの使用」の「[選択リストを短くするには](#)」も参照してください。

プログラミングオブジェクトで検索機能またはリスト機能を使用する場合は、ライブラリ名の範囲も指定できます。同様のことが、デバッグ環境でリスト機能を使用する場合やエラーメッセージで検索機能を使用する場合に適用されます。ただし、ライブラリ範囲を指定すると、選択条件が満たされる回数によっては、レスポンスタイムに悪影響を与えることがあります。

次の表に、名前の範囲として有効な表記を示します。value は 1 文字以上の任意の組み合わせを示します。

入力	選択される項目
*	すべての項目。
value*	value で始まる名前を持つすべての項目。 例：AB* 選択対象：AB、AB1、ABC、ABEZ 選択対象外：AA1、ACB
value>	value 以降の名前を持つすべての項目。 例：AB> 選択対象：AB、AB1、BBB、ZZZZZZZ 選択対象外：AA1、AAB
value<	value 以前の名前を持つすべての項目。 例：AX< 選択対象：AB、AWW、AX 選択対象外：AXA、AY



# 97

## SYSMAIN に発行される特殊コマンド

SYSMAIN ユーティリティにコマンドを発行して、ユーティリティ自体の操作に関連する機能を実行したり、Natural システムファイルのセキュリティを定義したりできます。

コマンド	機能
ADAON または NOADA または ADAOFF	エラートラッピングをアクティブ (ADAON) あるいは非アクティブ (NOADA または ADAOFF) にします。  デバッグの目的で、異常データベースエラーをトラップします (オンラインでプログラミングオブジェクトにのみ適用)。
BATCH または NOBATCH	SYSMAIN ユーティリティをバッチモード (BATCH) に切り替えると、すべての処理は SYSMAIN がバッチで実行されているように行われます。オンライン自動処理を使用すると、バッチレポートが表示されます。「 <a href="#">オンラインレポートモード</a> 」も参照してください。  NOBATCH では、SYSMAIN ユーティリティはオンラインモードに切り替えられます。
CLEAR	現在のワークエリアをクリアします。この機能は、ラージオブジェクトのソースコードがワークエリアに含まれ、SYSMAIN ユーティリティにより大きい ESIZE が必要な場合に役立つことがあります。
DISPLAY	発生したエラーの拡張 (ロング) メッセージテキストを表示します。
FINDFIRST	指定したオブジェクトを含む最初のオブジェクトが検出された場合、プログラミングオブジェクトの FIND コマンドを停止します。
FINDALL	プログラミングオブジェクトの FIND コマンドですべてのオブジェクトを検索します。
MON または NOMON または MONOFF	SYSMAIN での現在の動作の追跡をアクティブ (MON) あるいは非アクティブ (NOMON または MONOFF) にします。「 <a href="#">ダイレクトコマンドのキーワードと変数</a> 」の「 <a href="#">MON</a> 」も参照してください。
PROMPT または NOPROMPT	SYSMAIN プロンプトを有効 (PROMPT) または無効 (NOPROMPT) にします。「 <a href="#">ダイレクトコマンドのキーワードと変数</a> 」の「 <a href="#">PROMPT</a> 」も参照してください。

## SYSMAIN に発行される特殊コマンド

コマンド	機能
SET	すべての特殊 SYSMAIN コマンドについて説明している [Command Help] ウィンドウを呼び出します。
SET FDIC	Predict システムファイルの Adabas セキュリティ情報を指定するウィンドウを呼び出します。このウィンドウでは、FDIC プロファイルパラメータを参照します（『パラメータリファレンス』ドキュメントの「FDIC - Predict システムファイル」を参照）。バッチモードでは、DIC キーワードを <i>where-clause</i> で使用して、セキュリティ情報を指定できます（『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションのダイレクトコマンド構文を参照）。  「 <a href="#">管理者向けの特別な考慮事項</a> 」も参照してください。
SET FNAT	SYSMAIN ソースシステムファイルとターゲットシステムファイルの Adabas セキュリティ情報を指定するウィンドウを呼び出します。バッチモードでは、SEC キーワードを <i>where-clause</i> で使用して、セキュリティ情報を指定できます（『SYSMAIN ユーティリティ』ドキュメントのオブジェクトタイプ関連セクションのダイレクトコマンド構文を参照）。  「 <a href="#">管理者向けの特別な考慮事項</a> 」も参照してください。
SET FSEC	Natural Security がインストールされている場合、FSEC システムファイルの Adabas セキュリティ情報を指定するウィンドウを呼び出します。このウィンドウでは、FSEC プロファイルパラメータを参照します（『パラメータリファレンス』ドキュメントの「FSEC - Natural Security システムファイル」を参照）。  「 <a href="#">管理者向けの特別な考慮事項</a> 」も参照してください。
SET PC	Natural Connection と Entire Connection がインストールされている場合にのみ適用されます。  PC 接続をアクティブ化します。この設定は、%+ および %- 端末コマンドで断続的に変更できます（『端末コマンド』ドキュメントの「Natural Connection の使用の有効化および無効化」も参照）。SET PC により、SYSMAIN では PC 接続のステータスが再確認されます。
STATUS	Software AG 技術サポートにとって重要な SYSMAIN 変数の現在の値を表示します。
TOTAL	最後に実行された SYSMAIN 機能の処理を確認する [Results of Function] ウィンドウを呼び出します。保存（ソース）オブジェクトとカタログ化オブジェクトについて、次の情報が表示されます。
<b>Read</b>	オブジェクト名の指定に基づいて実際に読み込まれたオブジェクトの合計数。
<b>Rejected</b>	読み込まれた後に、指定した選択条件に基づいて拒否されたオブジェクトの合計数。「 <a href="#">オブジェクトの拒否と理由</a> 」も参照してください。
<b>Processed</b>	処理されたオブジェクトの合計数。
<b>Added</b>	ターゲット環境に追加された新規オブジェクトの合計数。
<b>Updated</b>	更新された既存のオブジェクトの合計数（可能な場合、SYSMAIN では既存のオブジェクトを削除して新規オブジェクトを追加するのではなく、既存のオブジェクトを更新しようとします）。



コマンド	機能	
	<b>Deleted</b>	機能および置き換えオプションの設定に応じてソース環境またはターゲット環境から削除されたオブジェクトの合計数。
	<b>Replaced</b>	ターゲット環境で置き換えられたオブジェクトの合計数。
	<b>Not Repl.</b>	ターゲット環境で置き換えられなかったオブジェクトの合計数。
	<b>Recs.Read:</b>	読み込まれたレコードの合計数。
. または END または QUIT	SYSMAIN を終了します。	



# 98 処理ステータスとエラー通知

---

■ オブジェクトの拒否と理由 .....	622
■ ステータスメッセージ .....	624
■ SYSMAIN エラー通知 .....	628

このセクションでは、機能処理時のオブジェクト拒否の考えられる理由、処理後に表示されるステータスメッセージ、および処理中のエラー通知について説明します。

## オブジェクトの拒否と理由

---

SYSMAIN 機能の実行中に、指定した選択条件を満たす 1 つ以上のオブジェクトが検出されたが、それらのオブジェクトの一部またはすべてがその後の処理を拒否された場合、次のいずれかの Natural システムエラーが発生します。

4867：この選択条件では何も見つかりません。

4810：選択条件により全データが拒否されました。

4893：処理が正常終了しましたが、一部のデータは処理されませんでした。

SYSMAIN コマンド **TOTAL**（「SYSMAIN に発行される特殊コマンド」を参照）を使用して要求の特定のステータスを確認できます。

オブジェクト拒否の考えられる理由については、次のセクションで説明します。

- 無効なオブジェクトタイプ、日付／時刻、ユーザー、または端末
- 同一のターゲット名
- XREF とユーザー出口
- ショートエラーメッセージまたは拡張エラーメッセージがない
- ライブラリ制限

### 無効なオブジェクトタイプ、日付／時刻、ユーザー、または端末

- オブジェクトを選択し、拒否されました。オブジェクトタイプが指定した処理のタイプに無効であったためです。例えば、プログラミングオブジェクトを処理するとき、プログラムタイプまたはサブルーチンタイプのオブジェクトに処理が制限されている場合、マップはすべて拒否されます。
- オブジェクトを選択し、拒否されました。指定したオブジェクトが保存またはカタログ化された日付または時刻が、指定した日付または時刻の範囲外であったためです。
- オブジェクトを選択し、拒否されました。指定したオブジェクトが保存またはカタログ化されたユーザー ID または端末 ID が、指定したユーザー ID または端末 ID に一致しなかったためです。
- カタログ化オブジェクトを選択し、拒否されました。プロファイルパラメータ **RECAT** が ON に設定され、カタログ化オブジェクトに対応する保存（ソース）オブジェクトが存在しなかったためです。「[プロファイルパラメータ RECAT の使用](#)」も参照してください。
- 保存（のみの）オブジェクトを選択し、拒否されました。プロファイルパラメータ **RECAT** が ON に設定され、ターゲット環境に同じ名前のカタログ化オブジェクトがすでに含まれていたためです。「[プロファイルパラメータ RECAT の使用](#)」も参照してください。

## 同一のターゲット名

- オブジェクトを選択し、拒否されました。ターゲット環境に同じ名前でも識別されるオブジェクトがすでに含まれており、置き換えオプションがアクティブ化されていなかった（**[Replace]** フィールドが N に設定されているか、またはキーワード REPLACE が指定されていない）ためです。
- S (サブルーチン) タイプのカatalog化プログラミングオブジェクトを選択し、拒否されました。外部サブルーチンの名前がターゲットライブラリの別のサブルーチンの名前と同じであったためです。
- 4 (クラス) タイプのカatalog化プログラミングオブジェクトを選択し、拒否されました。外部クラスの名前がターゲットライブラリの別のクラスの名前と同じであったためです。

## XREF とユーザー出口

- XREF オプションがアクティブ化され（**[XREF]** フィールドが N に設定されていない、またはキーワード XREF が指定されている）、指定したプログラミングオブジェクトの XRef データがありませんでした。
- ユーザー出口ルーチンがアクティブであり、オブジェクトの処理中にゼロ以外のリターンコードが返されました。

## ショートエラーメッセージまたは拡張エラーメッセージがない

- 拡張（ロング）エラーメッセージを選択し、拒否されました。ソースライブラリに対応するショートエラーメッセージがなかったためです。
- 拡張（ロング）エラーメッセージを選択しましたが、処理されませんでした。ターゲット環境に対応するショートエラーメッセージがなかったためです。
- 移動、削除、または番号変更するためにショートエラーメッセージを選択しましたが、処理できませんでした。対応する拡張エラーテキストが選択条件に含まれていなかったためです。拡張エラーメッセージには、対応するショートエラーメッセージが常に必要です。

## ライブラリ制限

- Predict Application Control/Predict Application Audit で制御されるライブラリを指定しました。オブジェクトは SYSMAIN では処理できません。

## ステータスメッセージ

次の表に、[Selection] 画面の [Message] 列に表示されるステータスメッセージ、またはオブジェクトに SYSMAIN 機能を実行した後にバッチレポート（「[オンラインレポートモード](#)」も参照）に表示されるステータスメッセージを示します。示されている行コマンドの詳細については、『SYSMAIN』ドキュメントのオブジェクトタイプ関連セクションで「[選択リスト](#)」を参照してください。

メッセージ	説明	機能は正常に終了したか
Class Exists	プログラミングオブジェクトにのみ適用されます。  クラスのカatalog化オブジェクトの外部クラス名は、ターゲット環境のクラスの他のカatalog化オブジェクトにすでに使用されています。	×
Copied	次のいずれかの行コマンドが [Copy] / [Rename Selection] 画面から実行されました。 A または プログラミングオブジェクトの A、C、S または エラーメッセージの A、E、S または プロファイルの A、D、E、M、P	○
DB Error: nnn	処理中にオブジェクトについてデータベースエラーが返されました。	×
Deleted	次のいずれかの行コマンドが [Delete Selection] 画面から実行されました。 A または プログラミングオブジェクトの A、C、S または エラーメッセージの A、E、S または プロファイルの A、D、E、M、P	○
Dev exists	プロファイルにのみ適用されます。  置き換えオプションが N に設定され、行コマンド A または D が [Copy] / [Move] / [Rename Selection] 画面から実行されました。ターゲット環境内の名前と同じ名前を持つデバイスプロファイルは置き換えられませんでした。	×
Directory	行コマンド I（ディレクトリの表示）が [Selection] 画面から実行されました。	○
Edt exists	プロファイルにのみ適用されます。  置き換えオプションが N に設定され、行コマンド A または E が [Copy] / [Move] / [Rename Selection] 画面から実行されました。ターゲット環境内の名前と同じ名前を持つエディタプロファイルは置き換えられませんでした。	×

メッセージ	説明	機能は正常に終了したか
Err: NAT2999	プログラミングオブジェクトにのみ適用されます。  カタログ化オブジェクトが XREF オプションを F に設定して処理されていました。指定した FDIC システムファイルにオブジェクトの Predict エントリがありません。	×
Err: NAT4852	Natural Security 違反が発生しました。	×
Exit: nnn	ユーザー出口ルーチンがアクティブであり、ゼロ以外のリターンコードが出口によって返されました (nnn=リターンコード)。「ユーザー出口ルーチン」も参照してください。	×
Exported	プログラミングオブジェクトと DDM にのみ適用されます。  行コマンド X (PC ダウンロード) が [Selection] 画面からソースオブジェクトに実行されました。	○
Ext Exists	エラーメッセージにのみ適用されます。  拡張 (ロング) エラーメッセージが存在するエラーメッセージのショートエラーメッセージのみを削除しようとしてしました。これにより、対応するショートエラーメッセージが存在しない拡張 (ロング) エラーメッセージが発生します。これは Natural では許可されていません。	×
File Listed	ルールにのみ適用されます。  行コマンド F (DDM/フィールドの表示) が [Selection] 画面から自動ルールで実行されました。	○
In Use	オブジェクトの Natural UPDATE/READ 処理中に、Adabas レスポンスコード 145 が返されました。	×
Invalid	デバッグ環境にのみ適用されます。  [Selection] 画面に表示されるデバッグ環境の1つに無効な行コマンドが入力されました。	×
Listed	DDM とデバッグ環境には適用されません。  行コマンド L (ソースコードの表示) が [Selection] 画面から実行されました。	○
Map exists	プロファイルにのみ適用されます。  置き換えオプションが N に設定され、行コマンド A または M が [Copy] / [Move] / [Rename Selection] 画面から実行されました。ターゲット環境内の名前と同じ名前を持つマッププロファイルは置き換えられませんでした。	×
Moved	次のいずれかの行コマンドが [Move Selection] 画面から実行されました。 A または プログラミングオブジェクトの A、C、S または エラーメッセージの A、E、S または プロファイルの A、D、E、M、P	○

## 処理ステータスとエラー通知

メッセージ	説明	機能は正常に終了したか
Must be Auto	ルールにのみ適用されます。 行コマンド F (DDM/フィールドの表示) が [Selection] 画面からフリールールで実行されました。行コマンド F は自動ルールでのみ実行できます。	×
Name Erro	デバッグ環境にのみ適用されます。 名前の変更機能を使用しましたが、指定した新しい名前が無効であることが検出されました。選択に新しい名前を指定しなかったか、指定した名前に無効な特殊文字が含まれていたか、または開始文字がアルファベット文字ではありませんでした。	×
NBP Deleted または Ignored	プログラミングオブジェクトにのみ適用されます。 行コマンド B (バッファプールからのオブジェクトの削除) が [Selection] 画面から実行されました。指定したオブジェクトのバッファプールからの削除が確認されたかどうかによって、処理の完了時に返されるメッセージは、NBP Deleted または Ignored のどちらかとなります。	Yes/No
No Lang 1	エラーメッセージにのみ適用されます。 Natural システム拡張 (ロング) エラーメッセージには、言語コード 1 (英語) のみを使用できます。Natural システム拡張エラーメッセージのコピーが試みられましたが、言語として言語コード 1 が指定されていませんでした。	×
no Short Err	エラーメッセージにのみ適用されます。 追加の処理を行うために拡張 (ロング) エラーメッセージが選択されましたが、ターゲットエラーメッセージ番号に対応するショートエラーメッセージがありませんでした。	×
No Xref	プログラミングオブジェクトにのみ適用されます。 カタログ化オブジェクトの処理中に、XREF オプションが Y または F に設定されました。指定した FDIC システムファイルにオブジェクトの XRef データがありません。	×
Not Found	エラーメッセージとプロファイルのみに適用されます。 処理中に更新ロジックのエラーが発生し、要求されたエラーメッセージまたはプロファイルを検出できませんでした。このことは、指定したエラーメッセージまたはプロファイルが選択と更新の間に削除されたことを意味します。	×
Not Replaced または Not Repld	置き換えオプションが N に設定され、次のいずれかの行コマンドが [Copy] / [Move] / [Rename Selection] 画面から実行されました。 A または プログラミングオブジェクトの A、C、S または エラーメッセージの A、E、S または プロファイルの A、D、E、M、P	×



メッセージ	説明	機能は正常に終了したか
	ターゲット環境内の名前と同じ名前を持つオブジェクトは置き換えられませんでした。	
Parm exists	プロファイルにのみ適用されます。  置き換えオプションがNに設定され、行コマンドAまたはPが <b>[Copy]</b> / <b>[Move]</b> / <b>[Rename Selection]</b> 画面から実行されました。ターゲット環境内の名前と同じ名前を持つパラメータプロファイルは置き換えられませんでした。	×
Printed	プログラミングオブジェクトとルールにのみ適用されます。  行コマンドH（ハードコピーの生成）が <b>[Selection]</b> 画面から実行されました。	○
Renamed  または  Copied	デバッグ環境とエラーメッセージにのみ適用されます。  次のいずれかの行コマンドが <b>[Rename Selection]</b> 画面から実行されました。デバッグ環境のA、あるいはエラーメッセージのA、E、またはS。  元のオブジェクトを保持するオプションが指定されているかどうかによって、処理の完了時に返されるメッセージがRenamedまたはCopiedとなります。	○
Renamed as  または  Copied as	プログラミングオブジェクトにのみ適用されます。  行コマンドA、C、またはSが <b>[Rename Selection]</b> 画面から実行されました。元のオブジェクトを保持するオプションが指定されているかどうかによって、処理の完了時に返されるメッセージがRenamed asまたはCopied asとなります。	○
Replaced	置き換えオプションがYに設定され、次のいずれかの行コマンドが <b>[Copy]</b> / <b>[Move]</b> / <b>[Rename Selection]</b> 画面から実行されました。  Aまたは プログラミングオブジェクトの A、C、Sまたは エラーメッセージの A、E、Sまたは プロファイルの A、D、E、M、P  ターゲット環境内の名前と同じ名前を持つオブジェクトが置き換えられました。	○
Sized	プログラミングオブジェクトにのみ適用されます。  行コマンドZ（サイズの計算）が <b>[Selection]</b> 画面から実行されました。	○
Subrtn Exists	プログラミングオブジェクトにのみ適用されます。  サブルーチンのカタログ化オブジェクトの外部サブルーチン名は、ターゲット環境の他のカタログ化サブルーチンによってすでに使用されています。	×

メッセージ	説明	機能は正常に終了したか
Updated	エラーメッセージにのみ適用されます。  選択したエラーメッセージに指定した言語のテキストは、以前は存在していませんでした。SYSMAIN では、エラーメッセージが新しい言語テキストで更新されました。	○
Src locked	プログラミングオブジェクトにのみ適用されます。  行コマンド A または S が <b>[Copy]</b> / <b>[Move]</b> / <b>[Rename]</b> / <b>[Delete Selection]</b> 画面から実行されました。  各オブジェクトのソースがロックされていたため、オブジェクトは処理されませんでした。	×

## SYSMAIN エラー通知

SYSMAIN では、ランタイムエラーが発生した場合、常に処理中に回復が試みられます。この機能は、自動的にアクティブ化され、システム変数 \*ERROR-TA が使用されます（『システム変数』ドキュメントの「\*ERROR-TA」も参照）。この機能は、SYSMAIN が正常終了すると非アクティブ化されます。

SYSMAIN の終了に端末コマンド（『端末コマンド』ドキュメントの「%% および%-現在のオペレーションの中断」も参照）を使用すると、異常終了とみなされ、\*ERROR-TA システム変数がリセットされません。このシステム変数は、SYSMAIN を再度呼び出し、正常に終了することでリセットされます。\*ERROR-TA システム変数を設定した場合、SYSMAIN では終了時に、以前に割り当てられた値にリセットされます。

選択条件に対して無効なデータを指定すると、メッセージ行にエラーメッセージが表示されます。ショートエラーメッセージの意味について不確かな場合は、特殊コマンド **DISPLAY**（「SYSMAIN に発行される特殊コマンド」も参照）を入力して、対応する拡張（ロング）エラーメッセージテキストの表示をアクティブ化できます。

このセクションでは、次のトピックについて説明します。

- データ入力エラー

■ エラー処理

データ入力エラー

選択条件に対して無効なデータを指定すると、メッセージ行にエラーメッセージが表示されます。場合によっては、特定のエントリのオンラインヘルプ機能が呼び出されます。この機能では、エラーに関するより詳細な情報が提供されます。

バッチモードでエラーが発生した場合、エラーメッセージおよび対応するエラー番号が出力され、SYSMAIN ユーティリティは終了します。

エラー処理

要求によって処理エラーが発生した場合、SYSMAIN では次の例に示すようなエラーレポートが発行されます。

```

16:51:08          *** SYSMAIN Error Report ***          2005-08-10

The following internal error occurred while processing the
SYSMAIN function xxxxxx (cc):

    Error in field specification for IF SELECTION statement.

Error Number .. eeee
Program ..... pppppppp
Status Code ... s          Status ..... tttttttt
Line ..... 1111          Level ..... vv
Device ..... dddddddd
User ID ..... uuuuuuuu   User Name ... nn...nn
    
```

ウィンドウに表示される情報は、エラー原因の分析に役立ちます。

上記のウィンドウ内の値は、次のとおりです。

フィールド	値	説明
SYSMAIN FUNCTION	xxxxxx	実行中の SYSMAIN 機能。
	cc	Software AG 技術サポートにとって役立つ内部ステータスコード。次のコードが表示されます。  A 自動処理。 D XRef データは削除中。 E 処理中のエラー (SYSMAIN のフラグ)。 F XRef データ処理中のステータス設定。

フィールド	値	説明
		G XRef データ処理中のステータス設定。 H 選択リスト処理。 I オプションは処理中。 S 単一オブジェクト処理。 T コマンドプロセッサによる SYSMAIN の終了。 V XRef データ処理中のステータス設定。 X コマンドプロセッサによる SYSMAIN の終了。 Y 整合性チェックエラーが発生、再表示。 Z 整合性チェックエラーが発生、再表示。
エラー番号	eeee	システム変数 *ERROR-NR に対応します (『システム変数』ドキュメントを参照)。
Program	pppppppp	システム変数 *PROGRAM に対応します (『システム変数』ドキュメントを参照)。
Status Code	s	エラーの種類。表示されるコードは次のとおりです。  C コマンド処理エラーです。 L ログオンエラーです。 O オブジェクト時間エラーです。 S 修正不可能な構文エラーです。
Line	1111	システム変数 *ERROR-LINE に対応します (『システム変数』ドキュメントを参照)。
Device	dddddddd	システム変数 *DEVICE に対応します (『システム変数』ドキュメントを参照)。
User ID	uuuuuuuu	システム変数 *USER に対応します (『システム変数』ドキュメントを参照)。
Status	tttttttt	システム変数 *ERROR-TA に対応します (『システム変数』ドキュメントを参照)。
Level	vv	システム変数 *LEVEL に対応します (『システム変数』ドキュメントを参照)。
User Name	nn...nn	システム変数 *USER-NAME に対応します (『システム変数』ドキュメントを参照)。

処理エラーが発生した場合、ウィンドウ内の情報に注意し、Enter キーを押します。SYSMAIN ユーティリティでは、入力フィールドのデータ値を維持したまま、最後のアクティブなメニュー画面を回復しようとしています。

ウィンドウに特殊コマンド **DISPLAY** を入力すると (「SYSMAIN に発行される特殊コマンド」も参照)、発生したエラーの拡張 (ロング) メッセージテキストが表示されます。

バッチ処理中に処理エラーが発生した場合、SYSMAINユーティリティでは関連するエラーメッセージを出力し、終了します。

特定のユーザーエラーによってウィンドウが表示される場合もあります。SYSMAINでは評価時にすべてのエラーをトラップしようとはしますが、一部をトラップできない場合もあります。例えば、ユーザーがDDMをある環境から別の環境へコピーするよう要求したが、無効なデータベースID (DBID) を指定した場合、SYSMAINではそのデータベースにアクセス使用とします。Adabas レスポンスコード 148 が返され、SYSMAIN ERROR トランザクションが呼び出されてウィンドウが表示されます。同様に、無効なファイルによって、データベースから多数のエラーが送信される場合があります。

Adabas レスポンスコード 9 が返された場合、SYSMAIN ではユーザーにエラーを知らせるメッセージを書き込み、最後の機能またはメニューから処理を再開します。特定の要求が完了していない場合は、レスポンスコード 9 によって BACKOUT TRANSACTION が最後の未完了トランザクションとなったと考えられます。



# 99 管理者向けの特別な考慮事項

---

■ ファイルセキュリティ .....	634
■ Natural Security .....	635
■ ユーザー出口ルーチン .....	636

このセクションでは、SYSMAINユーティリティのセキュリティ面およびSYSMAINに提供されるユーザー出口ルーチンについて説明します。

## ファイルセキュリティ

ファイルセキュリティ（パスワードとサイファコード）は、Adabas または VSAM 環境のシステムファイルに対して定義されたセキュリティと関連付けられています。システムファイルにファイルセキュリティが定義されている場合、SYSMAIN機能を実行する前に、必要なソースシステムファイルまたはターゲットシステムファイル（あるいはその両方）に関して、パスワード、サイファコード、またはVSAM名（あるいはそのすべて）を指定する必要があります。これを行わなかった場合、Adabas または VSAM によって該当するエラーメッセージが発行されます。SYSMAIN ユーティリティの開始時に Natural セッションに割り当てられるデフォルトシステムファイルに対してセキュリティ情報を指定する必要はありません。

### ▶手順 99.1. パスワードおよびサイファコードを指定するには

- 1 任意のSYSMAINユーティリティメニューから、次の表に示すPFキーまたは特殊コマンドを使用して、必要なシステムファイルのセキュリティウィンドウを呼び出します。

システムファイル	コマンド	PF キー	影響を受けるオブジェクト/データ
FUSER、FNAT	SET FNAT	PF12	- プログラミングオブジェクト - デバッグ環境 - エラーメッセージ - プロファイル
FDIC	SET FDIC	PF11	- ルール - DL/I サブファイル - DDM - XRef 情報
FSEC	SET FSEC	PF10	- Natural Security プロファイル

指定したシステムファイルについて表示されるセキュリティウィンドウは、次のFUSERシステムファイルおよびFNATシステムファイルの例に類似しています。

```

+-----+
!   --- Security for the Natural System Files ---   !
!                                                    !
! Specify the password(s), cipher(s) and VSAM FCT !
! name(s) for the source/target file(s) below:    !
!                                                    !
!   - Source -                               - Target -   !
! Library .... OLDLIB           Library .... NEWLIB    !
! Database ... 10                Database ... 10       !

```




```

! File ..... 32          File ..... 32          !
!                               !
! Password ...           Password ...           !
! Cipher .....          Cipher .....          !
! VSAM Name .. _____ VSAM Name .. _____ !
+-----+

```

- 2 ウィンドウで、必要なソースシステムファイルまたはターゲットシステムファイル（あるいはその両方）に関して、適切なパスワード、サイファコード、または VSAM 名（あるいはそのすべて）を入力します。

 **注意:** [Library] フィールドは、プログラミングオブジェクト、デバッグ環境、またはエラーメッセージを処理する場合にのみ適用されます。

ファイルセキュリティを定義すると、SYSMAIN ユーティリティでは、そのセキュリティ情報がすべての後続処理に使用されます。セッションの初期化時に取得したデフォルトのセキュリティ情報を使用する必要がある場合は、対応するセキュリティウィンドウを再度呼び出し、パスワード、サイファコード、または VSAM 名の各フィールド、あるいはそのすべてをクリアする必要があります。パスワードとサイファコードは非表示であるため、フィールドが空であるかのように見える場合にも、再度クリアする必要があります。

## Natural Security

Natural Security 環境内で SYSMAIN ユーティリティを使用する場合は、次の 2 つの点について考慮する必要があります。

- [Natural Security 環境の定義](#)
- [Natural Security 環境での SYSMAIN の使用制限](#)

### Natural Security 環境の定義

ソースライブラリとターゲットライブラリは、1 つの Natural Security 環境内または異なる 2 つの Natural Security 環境内に配置できます。これらの環境は SYSMAIN ユーティリティに対して定義されている必要があります。

使用する Natural Security 環境の定義は、特殊コマンド SET FSEC で指定します。

デフォルトでは、Natural セッションの開始時に割り当てられた現在の FSEC 設定が使用されます。これらの設定は、[Security for Natural Security (FSEC) Files] ウィンドウで変更した場合、次の SET FSEC プロセスで変更するまで有効です。バッチモードまたはダイレクトコマンドモードでは、SEC キーワードを使用して、ファイルセキュリティと要求の割り当てを指定する必要があります。

ソース環境とターゲット環境を決定すると、SYSMAIN では、Natural Security を使用してソースライブラリとターゲットライブラリの両方が確認されます（ソースまたはターゲットのデータベースとファイル、あるいはその両方は、ライブラリセキュリティプロファイルで指定したデータベースID（DBID）およびファイル番号（FNR）に対応している必要があります。これらの値を指定しない場合、セキュリティプロファイルからデフォルト値が取得されます）。

### Natural Security 環境での SYSMAIN の使用制限


SYSMAIN ユーティリティ自体の使用を制限することも、SYSMAIN ユーティリティで処理されるソースライブラリおよびターゲットライブラリの使用を制限することもできます。SYSMAIN 機能の使用は、アプリケーションプログラミングインターフェイス MAINUSER を使用して呼び出した場合、個別に制御できます。詳細については、『Natural Security』ドキュメントの「Protecting Utilities」を参照してください。

## ユーザー出口ルーチン

---

SYSMAIN ユーティリティのユーザー出口ルーチンは、処理中の各オブジェクトに関する情報を提供するため、または機能処理を制御するために使用されます。ユーザー出口ルーチンは Natural サブプログラムであり、CALLNAT ステートメントを使用して呼び出されます。

サブプログラムのソースコードとそれらのソースコードが使用するデータエリアは、SM-UX-*nn*（*nn*=01~11）の名前でソースオブジェクトとしてライブラリ SYSMAIN に格納されます。ユーザー出口ルーチンを使用可能にするには、対応するソースオブジェクトをライブラリ SYSMAIN またはその *steplib* のいずれかに、MAINEX*nn* の名前でカタログ化します。

 **注意:** ユーザー出口ルーチンのソースオブジェクトの名前とカタログ化オブジェクトの名前は、更新インストールによるソースオブジェクトの上書きがカタログ化オブジェクトに影響することがないように、異なるものとなっています。

必要に応じて、任意のユーザー出口ルーチンを変更したり、展開したりできます。

コード論理によっては、これらの出口ルーチンを使用することにより、SYSMAIN ユーティリティに追加のオーバーヘッドが発生します。ただし、出口処理が完了したら、常に SYSMAIN に制御を返す必要があります。

SYSMAIN ユーティリティでは Adabas ファイルに対して ET ロジックが使用されるため、ユーザー出口ルーチンの使用により、トランザクションタイムリミット（Adabas パラメータ TT）が延長されることがあります。また、Adabas トランザクションの定義は変更しないでください。つまり、ET/BT コマンドや END/BACKOUT TRANSACTION ステートメントは発行しないでください。END TRANSACTION ステートメントの発行は、すべて SYSMAIN によって実行されます。ただし、ユーザーがユーザー出口ルーチンを使用して SYSMAIN 機能を正常終了する場合、このルールは適用されません。その場合は、終了する前に BACKOUT TRANSACTION を発行する必要があります。

リターンコードをゼロ以外の値に設定すると、SYSMAINから受け取ったエラーよりも優先されます。出口からエラーを受け取った場合、エラーはメッセージフィールドに配置され、必要に応じて表示または出力されます。ただし、自動処理は、処理が最少の端末I/Oで完了するため、例外となります。

次のセクションでは、個々のユーザー出口ルーチンについて説明します。

- MAINEX01 - オブジェクト検証用の第1ユーザー出口ルーチン
- MAINEX02 - オブジェクト検証用の第2ユーザー出口ルーチン
- MAINEX03 - 要求検証用のユーザー出口ルーチン
- MAINEX04 - ファイル割り当ての変更用のユーザー出口ルーチン
- MAINEX05 - ダイレクトコマンド検証用のユーザー出口ルーチン
- MAINEX06 - SYSMAIN 初期化用のユーザー出口ルーチン。
- MAINEX07 - SYSMAIN 終了用のユーザー出口ルーチン。
- MAINEX08 - バッチモードで何も検出されない場合のユーザー出口ルーチン
- MAINEX09 - バッチモードでの異常終了用のユーザー出口ルーチン
- MAINEX10 - バッチモードでのコマンドエラー用のユーザー出口ルーチン
- MAINEX11 - SYSMAIN に特殊フラグを設定するためのユーザー出口ルーチン

### MAINEX01 - オブジェクト検証用の第1ユーザー出口ルーチン

機能	オブジェクトがSYSMAINによって処理される前に、オブジェクトに関連付けられたデータ要素の現在の値設定を検証します。
注意	RESP-CODE パラメータをゼロ以外の値に設定することにより、MAINEX01 に渡されるすべてのオブジェクトを拒否できます。  いずれかの追加ロジックを実行する場合、トランザクションはトランザクション終了ステータスにできません。このため、END TRANSACTION ステートメントやBACKOUT TRANSACTION ステートメントは発行しないでください。  制御をSYSMAINに戻す必要があります。
パラメータ	PARM-AREA1 (A250) SYSMAIN パラメータエリア (固定値)。 PARM-AREA2 (A250) SYSMAIN パラメータエリア (変数値)。 RESP-CODE (B1) SYSMAIN に返されるレスポンスコード。  <b>注意:</b> 変更できるのは RESP-CODE パラメータのみです。
ローカルデータエリア	SM-UX-L

### MAINEX02 - オブジェクト検証用の第2ユーザー出口ルーチン

機能	オブジェクトが SYSMAIN によって処理された後に、オブジェクトに関連付けられたデータ要素の現在の値設定を検証します。
注意	RESP-CODE パラメータをゼロ以外の値に設定することにより、MAINEX02 に渡されるすべてのオブジェクトを拒否できます。  いずれかの追加ロジックを実行する場合、トランザクションはトランザクション終了ステータスにできません。このため、END TRANSACTION ステートメントや BACKOUT TRANSACTION ステートメントは発行しないでください。  制御を SYSMAIN に戻す必要があります。
パラメータ	PARM-AREA1 (A250) SYSMAIN パラメータエリア (固定値)。 PARM-AREA2 (A250) SYSMAIN パラメータエリア (変数値)。 RESP-CODE (B1) SYSMAIN に返されるレスポンスコード。  <b>注意:</b> 変更できるのは RESP-CODE パラメータのみです。
ローカルデータエリア	SM-UX-L

### MAINEX03 - 要求検証用のユーザー出口ルーチン

機能	ダイレクトコマンドまたはメニューモードでオンライン入力された情報に関して SYSMAIN に対して行われたすべての要求を検証します。MAINEX03 では、SYSMAIN がコマンドを処理する前に制御を取得します。
注意	RESP-CODE パラメータをゼロ以外の値に設定することにより、MAINEX03 に渡されるすべてのコマンドを拒否できます。  他のロジックを追加できますが、必要な END TRANSACTION 要求はユーザーがデータベースに発行する必要があります。  制御を SYSMAIN に戻す必要があります。
パラメータ	PARM-AREA (A250) コマンド文字列。 RESP-CODE (B1) SYSMAIN に返されるレスポンスコード。  <b>注意:</b> 変更できるのは RESP-CODE パラメータのみです。

## MAINEX04 - ファイル割り当ての変更用のユーザー出口ルーチン

機能	Natural システムファイルのデータベース、ファイル、パスワード、およびサイファコードを上書きします。
注意	MAINEX04 は、いずれかの要求が SYSMAIN によって処理または検証される前に呼び出されます。制御が MAINEX04 に渡されると、ステータスはトランザクション終了となります。このため、要求を拒否する場合は、RESP-CODE パラメータをゼロ以外の値に設定する必要があります。  制御を SYSMAIN に戻す必要があります。
パラメータ	PARM-AREA (A250) SYSMAIN パラメータエリア RESP-CODE (B1) SYSMAIN に返されるレスポンスコード。
ローカルデータエリア	SM-UX-L4

## MAINEX05 - ダイレクトコマンド検証用のユーザー出口ルーチン

機能	SYSMAIN のオンライン処理中に入力されたすべてのダイレクトコマンドを検証します。また、システムコマンドを示すために使用された特殊コマンドを上書きできます。
注意	MAINEX05 は、SYSMAIN 内で発行されたいずれかのダイレクトコマンドが処理される前に呼び出されます。例えば、MAINEX05 を使用すると、いずれかの <b>SET コマンド</b> (「SYSMAIN に発行される特殊コマンド」を参照) を検証できます。また、それらのコマンドが発行されるのを防ぐこともできます。これらのコマンドを検証し、RESP-CODE パラメータにゼロ以外の値を返して拒否できます。制御が MAINEX05 に渡されると、ステータスはトランザクション終了となります。  SYSMAIN 内で入力するコマンドは、先頭に 2 つの斜杠 (/) を付ける必要があります (「 <b>SYSMAIN コマンド行の使用</b> 」も参照)。MAINEX05 では、この目的で他の 2 つの特殊文字を定義できます。そのためには、希望する文字を CMD-DEL パラメータに割り当てます。CMD-DEL を空白に設定した場合、SYSMAIN ではデフォルト値の 2 つの斜杠 (/) が使用されます。制御を SYSMAIN に戻す必要があります。
パラメータ	COMMAND (A68) SYSMAIN に発行された現在のコマンド。 CMD-DEL (A3) システムコマンド用の特殊文字。 RESP-CODE (B1) SYSMAIN に返されるレスポンスコード。

**MAINEX06 - SYSMAIN 初期化用のユーザー出口ルーチン。**

機能	SYSMAIN セッションの初期化時に制御を取得します。
注意	MAINEX06はSYSMAINセッションの開始時に呼び出され、オブジェクトの削除、移動、置き換えのような要求を確認するためのプロンプトなど、一部のSYSMAINデフォルト設定を上書きできます。  すべてのパラメータが検証されます。パラメータが無効な場合は、ソフトウェア設定が使用されます。  制御をSYSMAINに戻す必要があります。
パラメータ データエリア	SM-UX-L6

**MAINEX07 - SYSMAIN 終了用のユーザー出口ルーチン。**

機能	SYSMAIN セッションの終了時に制御を取得します。
注意	MAINEX07は、SYSMAINセッションの終了時に、制御をSYSMAINで保持するかどうかを決定するために呼び出されます。
パラメータ	USER-AREA (A50) 自由に使用するエリア。

**MAINEX08 - バッチモードで何も検出されない場合のユーザー出口ルーチン**

機能	バッチモードでコマンドについてオブジェクトが検出されない場合、追加の処理を決定します。
注意	MAINEX08は、バッチモードで実行された特定のコマンドについて、指定した条件に一致するオブジェクトが検出されなかった場合に呼び出されます。この場合は、制御をSYSMAINに戻す必要はありませんが、戻してもかまいません。制御がSYSMAINに戻ると、SYSMAINは次のコマンドの処理を続行します。
パラメータ	CMD (A250) コマンド文字列。

**MAINEX09 - バッチモードでの異常終了用のユーザー出口ルーチン**

機能	バッチモードでエラーが発生した場合に実行するアクションを決定します。
注意	MAINEX09は、バッチモードのSYSMAIN処理でエラーが発生した場合に呼び出されます。この場合は、制御をSYSMAINに戻す必要はありませんが、戻してもかまいません。制御をSYSMAINに戻した場合、SYSMAINはコンディションコード45で終了します。  <b>注意:</b> エラー NAT4810、NAT4818、NAT4867、NAT4868、および NAT4893 は、このユーザー出口ルーチンでは処理できません。

パラメータ	CMD (A250)          コマンド文字列。 ERROR-CODE (N4) 終了の原因となったエラーの番号
-------	-----------------------------------------------------------------

### MAINEX10 - バッチモードでのコマンドエラー用のユーザー出口ルーチン

機能	バッチモードでコマンドエラーが発生した場合に実行するアクションを決定します。
注意	MAINEX10 は、バッチモードで SYSMAIN コマンドにエラーが検出された場合に呼び出されます。この場合は、制御を SYSMAIN に戻す必要はありませんが、戻してもかまいません。制御が SYSMAIN に戻ると、SYSMAIN は次のコマンドの処理を続行します。
パラメータ	CMD (A250)          コマンド文字列。 ERROR-CODE (N4) 終了の原因となったエラーの番号

### MAINEX11 - SYSMAIN に特殊フラグを設定するためのユーザー出口ルーチン

機能	特殊設定ユーザー出口ルーチン。
注意	MAINEX11 は、SYSMAIN セッションの開始時に呼び出され、バッチでの MAINUSER メッセージの表示など、一部の特殊 SYSMAIN フラグを設定できます。  使用可能なフラグの詳細については、ユーザー出口ルーチンのソースオブジェクト (SM-UX-11) を参照してください。  制御を SYSMAIN に戻す必要があります。
パラメータ	FLAGS (A250) フラグ文字列 (再定義)。





# 100

## SYSNCP ユーティリティ

---

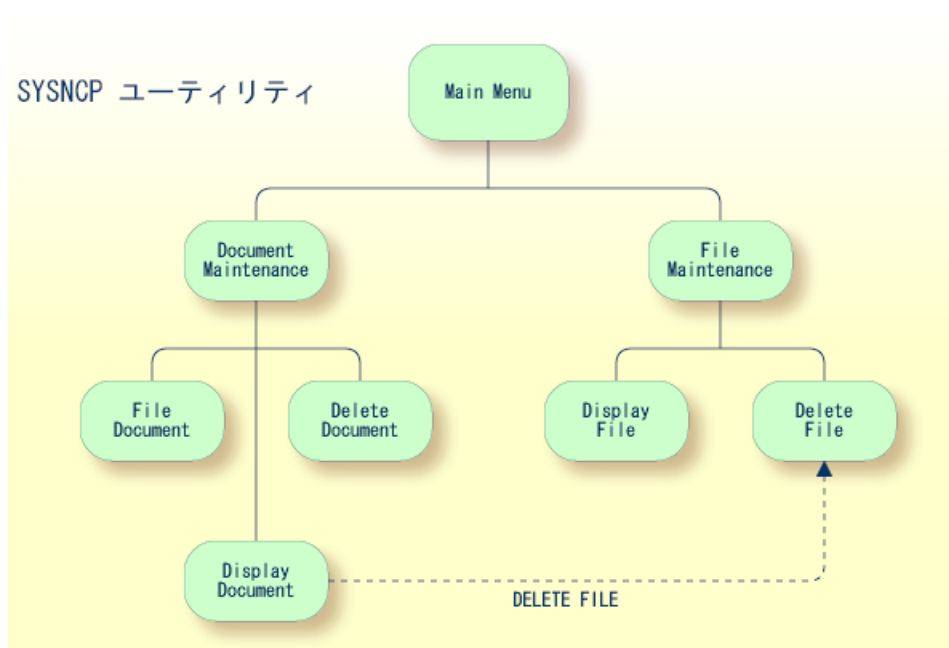
▪ SYSNCP ユーティリティについて .....	644
▪ SYSNCP の呼び出し .....	650
▪ プロセッサの選択 .....	651
▪ ヘッダーレコード .....	652
▪ キーワードのメンテナンス .....	662
▪ 機能のメンテナンス .....	667
▪ ランタイムアクション .....	672
▪ プロセッサのカタログ化 .....	678
▪ 管理者サービス .....	678
▪ セッションプロファイル .....	685

SYSNCPユーティリティは、Naturalアプリケーションに対してコマンド方式のナビゲーションシステムを定義するために使用します。

Naturalコマンドプロセッサ（NCP）は、メンテナンスとランタイムの2つのコンポーネントで構成されています。SYSNCPユーティリティはメンテナンス部分に相当し、アプリケーション内のナビゲーションを定義および制御するためのすべて機能を備えています。PROCESS COMMAND ステートメント（『ステートメント』ドキュメントを参照）は、Naturalプログラムを呼び出すために使用されるランタイム部分です。

## SYSNCP ユーティリティについて

ユーザーがダイレクトコマンドを使用してあるアクティビティから別のアクティビティに移動できるアプリケーションは、ユーザーがメニュー階層を経由して目的のアクティビティに移動するアプリケーションよりも、有用性の点ではるかに上回っています。



上の図は、ダイレクトコマンドを使用することの利点を示しています。メニュー階層がナビゲーションの基盤となっているアプリケーションでは、Display Document（ドキュメント表示）機能から Delete File（ファイル削除）機能へ移動するユーザーは、ドキュメント関連のブランチを経由して Main Menu（メインメニュー）に戻ってから、ファイル関連のブランチに入る必要があります。これは、Display Document（ドキュメント表示）機能から Delete File（ファイル削除）機能に直接アクセスする場合と比べて、明らかに非効率的です。

以下に参考情報を示します。

- オブジェクト指向のデータ処理

- コマンドプロセッサの機能
- コマンドプロセッサの構成
- コマンドとは
- コマンドプロセッサの作成

## オブジェクト指向のデータ処理

Naturalコマンドプロセッサを使用して、アプリケーション内のナビゲーションを定義および制御します。例えば、コマンドプロセッサを使用して、Display Document（ドキュメント表示）機能に直接アクセスできる DISPLAY DOCUMENT コマンドを定義します。このコマンドを使用できる画面のコマンド行にこのコマンド文字列を入力すると、Naturalコマンドプロセッサによって入力処理され、コマンドに割り当てられているアクションが実行されます。

メニュー方式のアプリケーションとは対照的に、Naturalコマンドプロセッサで実装されるコマンド方式のアプリケーションは、オブジェクト指向のデータ処理に向かって大きな一歩を踏み出しています。このアプローチには次の利点があります。

- アプリケーションの設計は、ある結果に達するまでの過程に依存する必要がなく、求める結果自体にのみ依存します。そのため、アプリケーション設計は構成要素内の処理フローに影響されなくなります。
- アプリケーションの処理ユニットが互いに独立するようになるため、簡単、迅速、しかもより効率的に、アプリケーションをメンテナンスすることができます。
- アプリケーションは、独立した処理ユニットを追加することで簡単に拡張できます。このため、エンドユーザーにとって利用しやすいだけでなく、プログラマにとっても作成しやすいアプリケーションになります。

Natural コマンドプロセッサには、さらに次の利点があります。

- プログラムコーディングの縮小  
今までは、コマンド処理を操作するために、長くて同じような構造化ステートメントブロックを繰り返す必要がありましたが、コマンドプロセッサを呼び出す PROCESS COMMAND ステートメントを指定するだけで済むようになりました。実際のコマンド操作をソースコードに指定する必要がなくなりました。これにより、必要なコーディングの量が大幅に削減されます。
- コマンド操作の効率化  
標準的な方法で一元集中的にコマンド操作を定義するため、アプリケーションのコマンド処理部分の作成およびメンテナンスに伴う作業をより迅速かつ効率的に行うことができます。
- パフォーマンスの改善  
Naturalコマンドプロセッサは、特にパフォーマンス面を考慮して設計されています。そのため、できる限り迅速にコマンドが処理され、Naturalアプリケーションのパフォーマンス改善に役立ちます。

## コマンドプロセッサの機能

Naturalコマンドプロセッサには、効率的で使いやすいコマンド操作のための機能が数多く備えられています。

- コマンドの柔軟な操作  
頻繁に使用するコマンドに対して、別名（キーワードのシノニム）および省略形を定義できます。
- 省略されたキーワードの一意性の自動チェック  
コマンドプロセッサにより、SYSNCPに指定したキーワードごとに他のすべてのキーワードとの比較が自動的に行われ、キーワードを一意に識別するために必要な各キーワードの最小文字数が決定されます。つまり、アプリケーションでコマンドを入力する場合、ユーザーは各キーワードを、コマンドプロセッサが他のキーワードと区別するために必要とする最小限の長さに短縮できます。
- コマンドのローカルおよびグローバルな有効性  
特定のコマンドに対応して実行するアクションを、すべての条件で同じにするか、または状況に依存するかをSYSNCPで指定できます。例えば、前に発行されたプログラムに応じてアクションを決定することができます。さらに、ある条件下では有効で、別の条件下では無効になるようにコマンドを定義できます。
- 不正コマンドに対するエラー処理  
ユーザー独自のエラー処理ルーチンをコマンドに結び付けることや、エラー入力をNaturalで処理することができます。
- 機能のセキュリティ  
Natural Securityを使用すると、ライブラリ固有およびユーザー固有の使用条件をSYSNCPで生成したテーブルに定義できます。このようにして、Naturalアプリケーションの特定の機能またはキーワードを、特定のユーザーに対して許可または禁止することができます。これを、機能のセキュリティと呼びます。『Natural Security』ドキュメントの「Functional Security」も参照してください。
- ヘルプテキスト  
SYSNCPでは、キーワードまたはコマンドにヘルプテキストを結び付けることができます。その後でPROCESS COMMAND ACTION TEXTステートメントを指定することにより、コマンド固有のヘルプテキストをプログラムに返すことができます。
- コマンド処理のオンラインテスト  
コマンドの実行結果が意図したものにならなかった場合は、PROCESS COMMANDステートメント（『ステートメント』ドキュメントを参照）とライブラリSYSNCPに提供されているソース形式のサンプルテストプログラムEXAM\*を使用して、コマンドが正しく処理されなかった理由を調べることができます。EXAM-\*プログラム名の末尾が、関連するアクションウィンドウの上の境界線に、省略形として表示されます。例えば、EXAM-Cの場合はCと表示されます。

**▶手順 100.1. コマンドプロセッサをランタイムでテストするには**

- 1 ダイレクトコマンド「EXAM」を入力して、すべてのテストプログラムをリストします。  
「**Demonstrate PROCESS COMMAND Statement**」ウィンドウが表示されます。
- 2 ファンクションコード「O」を入力して、プロセッサを開きます。
- 3 プロセッサの名前を入力します。
- 4 リストされたファンクションコードのいずれかを選択して、コマンドアクションを適用します。例えば、CHECK の場合は C を選択します。
- 5 ファンクションコード「Q」を入力して、プロセッサを閉じます。

**コマンドプロセッサの構成**

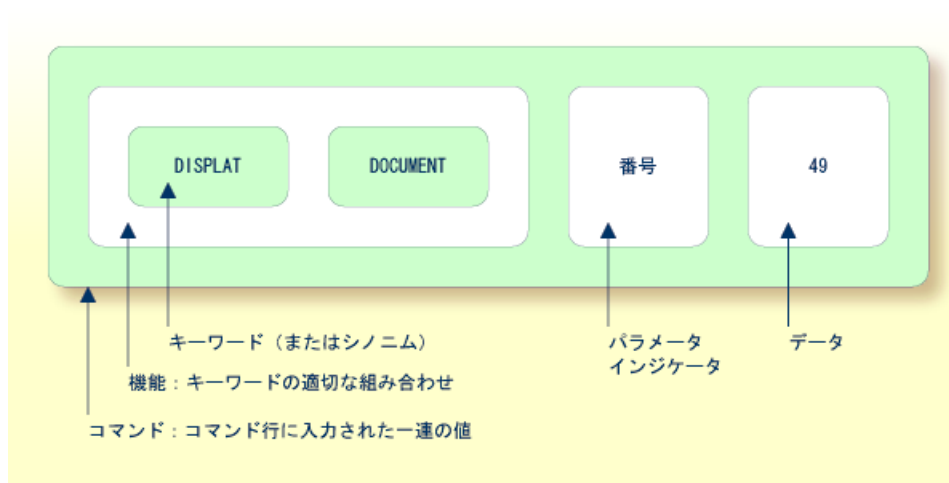
Natural コマンドプロセッサは、開発とランタイムの2つの部分で構成されます。

- 開発部分は SYSNCP ユーティリティであり、このセクションで説明しています。SYSNCP ユーティリティを使用して、コマンド（下記参照）およびコマンドの実行に対応して実行されるアクションを定義します。SYSNCP は、管理者が設定した定義により、コマンド入力時に実行する処理を判断するデシジョンテーブルを生成します。これらのテーブルは、プロセッサタイプの Natural メンバに含まれます。
- ランタイム部分は PROCESS COMMAND ステートメントであり、『ステートメント』ドキュメントで説明しています。PROCESS COMMAND ステートメントを使用して、Natural プログラム内でコマンドプロセッサを呼び出します。PROCESS COMMAND ステートメントでは、ユーザーによるコマンド入力を処理するためにこの時点で使用するプロセッサの名前を指定します。

**コマンドとは**

コマンドとは、コマンド行に入力される一連の値であり、アプリケーションによって認識および処理されます。コマンドは、最大3つの要素を含むことができます。

- 機能：  
1つ以上の有効なキーワード。MENU または DISPLAY DOCUMENT など。
- パラメータインジケータ：  
オプション。コマンドデータを導くキーワード。
- コマンドデータ：  
機能に送られる情報。コマンドデータは、英数字または数字で指定できます。例えば、表示されるファイルの名前または番号を指定します。



コマンドは、常にアプリケーション内のある状態から実行されます。この状態に達した位置をロケーションと呼びます。コマンドによって、ユーザーはあるロケーションから別のロケーションに導かれます。そのため、各コマンドはベクトルとみなすことができます。

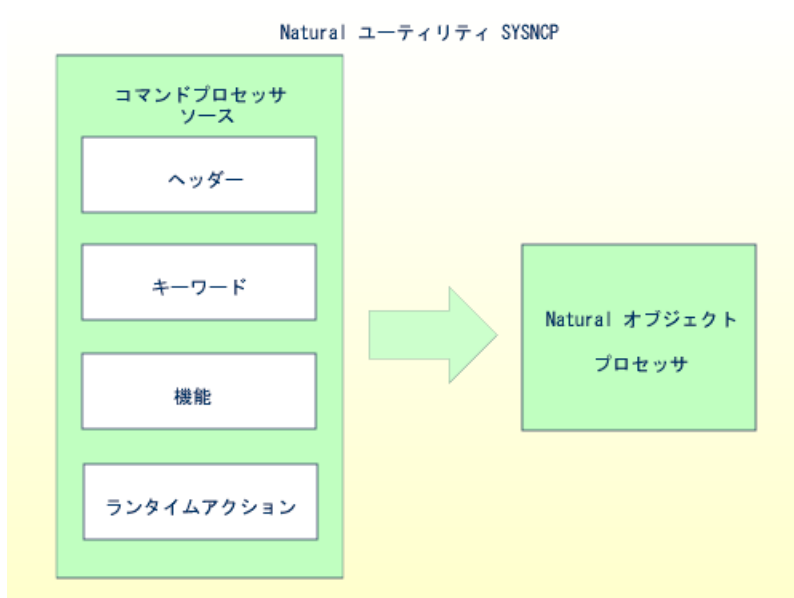


特定のコマンドを発行できるロケーションを、システム全体またはユーザーベースで制限することができます。例えば、システム全体を基準とした場合、コマンド内に指定する機能をローカルまたはグローバルにすることができます。グローバル機能はあらゆるロケーションから発行できるのに対し、ローカル機能は指定されたロケーションからのみ発行できます。ただし、Natural Security がアクティブになっている環境では、キーワードおよび機能に制限を設けることができます。

### コマンドプロセッサの作成

SYSNCP ユーティリティを使用して、コマンドプロセッサを作成およびメンテナンスします。コマンドプロセッサには、ユーザーが有効なコマンドを入力したときの処理を決定するデシジョンテーブルが含まれています。

コマンドプロセッサの作成は累積的な作業であり、プロセッサの全般的な設定を構築するヘッダ定義から、キーワード定義、機能定義、および機能に対するアクションの関連付けまでのさまざまな手順を伴います。SYSNCPには、キーワード、機能、およびアクションを指定するための特別なエディタが用意されています。



コマンドプロセッサ開発の最終的な成果物は複合的なコマンドプロセッサソースであり、このソースをカタログすると、プロセッサタイプの Natural オブジェクトが生成されます。このオブジェクトを Natural の PROCESS COMMAND ステートメントで参照するたびに、Natural コマンドプロセッサのランタイムシステムが起動されます。

コマンドプロセッサの作成に必要な手順の概要は、以下のとおりです。

#### ▶手順 100.2. コマンドプロセッサを作成するには

- 1 セッションプロファイルを確認および修正します。  
SYSNCP 自体が使用するセッションプロファイルには、SYSNCP による特定のアクションの実行方法および情報の表示方法を制御するさまざまなパラメータが含まれています。必要な修正を行った後、そのプロファイルを指定のユーザー ID で保存することができます。「[セッションプロファイル](#)」を参照してください。
- 2 コマンドプロセッサを初期化します。  
コマンドプロセッサの名前およびコマンドプロセッサを保存するライブラリを指定します。
- 3 グローバル設定（ヘッダー）を定義します。  
コマンドプロセッサに対するさまざまなグローバル設定を定義します。例えば、キーワード編集時の記述テキスト、キーワードの最小文字数と最大文字数、ランタイム時に処理するキーワードの順番、ランタイムエラー処理、およびランタイム時の機能呼び出しに PF キーを使用できるかどうかを設定できます。「[ヘッダーレコード](#)」を参照してください。
- 4 キーワードを定義します。

コマンドプロセッサで処理するキーワードはそれぞれ、コマンドの1~3番目のどのエントリとして入力するかに関する指定とともに定義します。キーワードのシノニムは、パラメータインジケータとしても定義できます。キーワードごとにユーザーテキストを定義できます。このテキストは、その後、PROCESS COMMAND ACTION TEXT ステートメントを使用して、ランタイム時に読み込むことができます。「[キーワードのメンテナンス](#)」を参照してください。

- 5 機能を定義します。  
機能は、キーワード結合を有効にすることで定義します。機能は、ローカル（アプリケーション内の特定のロケーションからのみ呼び出し可能）またはグローバル（アプリケーション内のどこからでも呼び出し可能）あるいはその両方として定義できます。「[機能のメンテナンス](#)」を参照してください。
- 6 ランタイムアクションを定義します。  
ランタイム時にコマンドが発行されたときに、コマンドプロセッサによって実行されるアクションを定義します。例えば、Natural プログラムを取り出す、コマンドを Natural スタックの先頭に置く、データを Natural スタックの先頭に置く、およびコマンド行の内容を変更するアクションを定義します。「[ランタイムアクション](#)」を参照してください。
- 7 コマンドプロセッサをカタログします。  
作成したソースを Natural オブジェクト（プロセッサタイプ）として、指定した Natural ライブラリにカタログします。これで、PROCESS COMMAND ステートメントを使用して、Natural プログラムからこのコマンドプロセッサを呼び出すことができます。「[プロセッサのカタログ化](#)」を参照してください。

## SYSNCP の呼び出し

### ▶手順 100.3. SYSNCP ユーティリティを呼び出すには

- システムコマンド SYSNCP を入力します。

[Processor Source Maintenance] メニューが表示されます。

```

18:22:53          ***** NATURAL SYSNCP UTILITY *****          2000-05-22
User SAG          - Processor Source Maintenance -

Code  Function
S     Select Processor
N     Create New Processor
H     Modify Header
K     Define Keywords
F     Define Functions
R     Define Runtime Actions
C     Catalog Processor

```



```


      A   Administrator Services
      ?   Help
      .   Exit

      Code .. _   Name .. SAGTEST_   Library .. SYSNCP__

Logon to SYSNCP accepted.
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Cmd   Exit Last List Flip                               Canc

```

このメニューから、コマンドプロセッサの作成およびメンテナンスに必要なすべての機能呼び出すことができます。機能呼び出すには、[Code] フィールドにコード文字を入力します。

 **注意:** SYSNCP ユーティリティを呼び出すか、または SYSNCP を再起動すると、ダイナミックなカスタマイズを目的としてユーザー出口NCP-USR1が呼び出されます。Naturalシステムライブラリ SYSNCP で提供されているプログラム NCP-USR1 を参照してください。

## ヘルプ

SYSNCPの各入力フィールドおよび一部の出力フィールドのヘルプを表示するには、そのフィールドにカーソルを置いて PF1 キーを押します。

## プロセッサの選択


プロセッサ選択機能では、存在するすべてのコマンドプロセッサソースを関連情報とともにリストします。Natural Security がインストールされている場合、ログオンを許可されているライブラリに対してカタログ可能なソースのみがリストされます。これらの制限は、管理者ステータスを持つユーザーには適用されません。

### ▶手順 100.4. プロセッサ選択機能呼び出すには

- 1 [Processor Source Maintenance] メニューで、ファンクションコード「S」を入力します。
- 2 Enter キーを押します。

プロセッサごとに次の情報が表示されます。

Name	コマンドプロセッサの名前。
Library	プロセッサを作成する Natural ライブラリの名前。プロセッサをカタログすると、このライブラリに保存されます。
User ID	プロセッサを作成したユーザーの ID。
Date	プロセッサが作成された日付。
Status	プロセッサ開発の段階。有効なステータスの値については、「ヘッダーレコード」の <i>Current Status</i> を参照してください。
Cat	プロセッサがカタログされているかどうかを示します。

 **注意:** Natural システムライブラリ SYSNCP で提供されているユーザー出口 NCP-SELX を使用すると、表示内容を特定のプロセッサに制限することができます。

3 [Ac] フィールドに任意の文字を入力して、プロセッサを選択します。

[Processor Source Maintenance] メニューが開き、選択したプロセッサの名前が自動的に [Name] フィールドに表示されます。

[Ac] フィールドに「?」（疑問符）を入力すると、ウィンドウが開き、その他の有効なオプションがリストされます。

コマンドプロセッサの名前およびライブラリ名の長さは、1~8文字です。英大文字（A~Z）、数字（0~9）、および特殊文字（"-","/","\$","&","#","+、および"\_"）で構成できます。

## ヘッダーレコード

ヘッダーメンテナンス機能では、コマンドプロセッサに対するさまざまなグローバル設定を定義します。これらの定義をまとめてヘッダーと呼びます。ヘッダーを作成および修正するために、7つのヘッダーメンテナンス画面が用意されています。コマンドプロセッサのヘッダー設定は、開発の任意の段階で更新できます（以下のセクションを参照）。設定を修正した後、コマンドプロセッサのステータスは常に Header に設定されます（*Current Status* も参照）。

以下に参考情報を示します。

- コマンドプロセッサの新規作成
- ヘッダー修正 - 全般的な説明
- キーワードランタイムオプション - ヘッダー 1
- キーワードエディタオプション - ヘッダー 2
- その他のオプション - ヘッダー 3
- コマンドデータ処理 - ヘッダー 4
- ランタイムエラー処理 - ヘッダー 5
- 統計 - ヘッダー 6

## ■ ステータス - ヘッダー 7

### コマンドプロセッサの新規作成

#### ▶手順 100.5. 新しいコマンドプロセッサを作成するには

- 1 [Processor Source Maintenance] メニューで、ファンクションコード「N」（Create New Processor）、  
作成するコマンドプロセッサの名前、および  
後でコマンドプロセッサをカタログする Natural ライブラリの名前を入力します。
- 2 Enter キーを押します。  
  
最初のヘッダーメンテナンス画面が表示されます。

最初のヘッダーメンテナンス画面および後続の画面には、編集可能なデフォルト値が入力されています。

### ヘッダー修正 - 全般的な説明

ヘッダー修正機能を使用して既存のヘッダーをメンテナンスします。つまり、指定したコマンドプロセッサのさまざまなヘッダー設定を修正します。

#### ▶手順 100.6. 既存のヘッダーを修正するには

- 1 [Processor Source Maintenance] メニューで、ファンクションコード「H」（Modify Header）、  
該当するコマンドプロセッサの名前、および  
このコマンドプロセッサがカタログされているライブラリの名前を入力します。
- 2 Enter キーを押します。  
  
最初のヘッダーメンテナンス画面が表示されます。
- 3 以下に示すヘッダーメンテナンス画面の入力フィールドを修正します。
- 4 Enter キーを押して修正を確認します。

プロセッサヘッダーの定義とメンテナンスには、7種類の画面を使用できます。ヘッダーの定義については前のセクションを参照してください。

#### ▶手順 100.7. ヘッダーメンテナンス画面間を移動するには

- PF8 キー（進む）または PF7 キー（戻る）を使用します。

各画面には、次の情報が表示されます。

Name	コマンドプロセッサの名前。
Library	作成したコマンドプロセッサオブジェクトを、カタログ後に保存するライブラリの名前。
DBID、FNR	指定したライブラリが存在するデータベースの ID およびファイル。
Created by	このコマンドプロセッサを初期化した Natural ユーザーのユーザー ID。
Date	コマンドプロセッサが新規作成された日付。
Current Status	<p>コマンドプロセッサのステータス。</p> <p>Init      コマンドプロセッサが初期化されました。</p> <p>Header    コマンドプロセッサのヘッダーが作成または修正されました。</p> <p>Keysave   キーワードが定義および保存されました。</p> <p>Keystow   キーワードがチェックおよび格納されました。</p> <p>Function   キーワード結合が定義されました。</p> <p>Action    ランタイムアクションが定義されました。</p> <p>Object    コマンドプロセッサのオブジェクト形式が作成されました。</p> <p>Frozen    コマンドプロセッサがフリーズされました。</p> <p>Copied    コマンドプロセッサがコピーされました。</p> <p>Error     エラーが検出されました。</p>

### キーワードランタイムオプション・ヘッダー 1

前述のとおりヘッダー修正機能を選択すると、[Processor Header Maintenance 1] 画面が表示されます。

```

16:40:19          ***** NATURAL SYSNCP UTILITY *****          2000-05-04
User SAG          - Processor Header Maintenance 1 -

Modify Processor          Name SAGTEST Library SYSNCP DBID 10 FNR 32
Created by SAG          Date 2000-04-29          Current Status Init

Keyword Runtime Options:
-----
First Entry used as ..... Action_____
Second Entry used as ..... Object_____
Third Entry used as ..... Addition_____

Minimum Length ..... _1
Maximum Length ..... 16
Dynamic Length Adjustment .. -
    
```

```

Keyword Sequence ..... 123_____
Alternative Sequence ..... _____
Local/Global Sequence ..... LG_____

Processor Header with name SAGTEST for library SYSNCP has been added.
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Cmd  Exit Last List Flip -      +                               Canc
    
```

コマンドプロセッサに定義するキーワードに適用されるさまざまな属性をこの画面に入力します。

フィールド	説明
First Entry used as	<p>キーワードシーケンスの定義時に、1番目のエントリ（エントリタイプ1）として入力したすべてのキーワードと関連付けられる記述テキスト。</p> <p>例えば、キーワードシーケンスの1番目のキーワードが実行するアクション（DISPLAY、DELETE など）を示す場合、このフィールドに記述テキスト「Action」を入力します。</p> <p>「キーワードのメンテナンス」セクションで説明されているように、このフィールドに入力したテキストの最初の4文字は、キーワードエディタの <b>[Use]</b> というヘッダ名の一列に表示されます。</p>
Second Entry used as	<p>キーワードシーケンスの定義時に、2番目のエントリ（エントリタイプ2）として入力したすべてのキーワードと関連付けられる記述テキスト。</p> <p>例えば、キーワードシーケンスの2番目のキーワードが使用するオブジェクト（DOCUMENT、FILE など）を示す場合、このフィールドに記述テキスト「Object」を入力します。</p> <p>「キーワードのメンテナンス」セクションで説明されているように、このフィールドに入力したテキストの最初の4文字は、キーワードエディタの <b>[Use]</b> というヘッダ名の一列に表示されます。</p>
Third Entry used as	<p>キーワードシーケンスの定義時に、3番目のエントリ（エントリタイプ3）として入力したすべてのキーワードと関連付けられる記述テキスト（TITLE、PARAGRAPH など）。</p> <p>「キーワードのメンテナンス」セクションで説明されているように、このフィールドに入力したテキストの最初の4文字は、キーワードエディタの <b>[Use]</b> というヘッダ名の一列に表示されます。</p>
Minimum Length	<p>キーワードを定義するときに指定可能な最小文字数。有効な値は1~16文字です。デフォルトは1文字です。</p>
Maximum Length	<p>キーワードを定義するときに指定可能な最大文字数。有効な値は1~16文字です。デフォルトは16文字です。</p>
Dynamic Length Adjustment	<p>可能な値は次のとおりです。</p>

フィールド	説明
	<ul style="list-style-type: none"> <li>+ ランタイム時に、各キーワードを完全な形で入力する必要があります。</li> <li>- ランタイム時に、各キーワードの省略形を指定できます。ただし、他のキーワードに対して一意性を保てる場合に限りです。</li> </ul> <p>S 指定したキーワードに必要な入力文字数は、「キーワードのメンテナンス」で説明されているように、キーワードの定義時にキーワードエディタの <b>[ML] フィールド</b> で指定します。</p>
Keyword Sequence	ランタイム時に処理される、キーワードエントリの順番。指定できる値は1、2、3、およびP（パラメタインジケータ用）です。デフォルトは12で、最初が1番目のキーワードエントリ、次が2番目のキーワードエントリという順番になります。「キーワードのメンテナンス」の <b>[E] フィールド</b> も参照してください。
Alternative Sequence	デフォルトの順番（上記で指定）がランタイム中にエラーになった場合に、ランタイムで処理されるキーワードの代替順序。
Local/Global Sequence	ランタイム時にコマンドを有効にする順序を指定します。可能な値は次のとおりです。 <ul style="list-style-type: none"> <li>L コマンドをローカルコマンドとして有効にします。</li> <li>G コマンドをグローバルコマンドとして有効にします。</li> </ul> <p>デフォルトの有効順序はLGです。この指定は、コマンドを最初にローカルコマンドとして有効にし、次に、必要に応じて、グローバルコマンドとして有効にします。</p>

## キーワードエディタオプション・ヘッダー 2

**[Processor Header Maintenance 2]** 画面では、追加のキーワード属性を入力できます。

フィールド	説明
Header 1 for User Text	この2つのフィールドには記述テキストを入力します。このテキストは、キーワードエディタの、ユーザーテキスト用に確保されている列の上に表示されます。またこのテキストは、『ステートメント』ドキュメントの説明に従って PROCESS COMMAND ステートメントで TEXT オプションを指定すると、ランタイム中に出力されます。
Header 2 for User Text	
Prefix Character 1	このフィールドと次の3つのフィールドを使用して、キーワードに16進接頭辞を付けます。これにより、通常のキーボードでは表せない内部キーワードの処理が可能になります。コマンドプロセッサがカタログされると、キーワードのすべての接頭文字が、指定された16進値で置き換えられます。 <p>[Prefix Character] フィールドのいずれかに空白以外の文字を入力した場合、指定された文字は [Hexadecimal Replacement] フィールドに指定した16進値で置き換えられます。</p>
Hex. Replacement 1	このフィールドに指定した値が [Prefix Character] フィールドに指定した文字に代わって、ランタイム時にキーワードの接頭辞として使用されます。
Prefix Character 2	上記の Prefix Character 1 を参照してください。
Hex. Replacement 2	上記の Hex. Replacement 1 を参照してください。

フィールド	説明
Keywords in Upper Case	<p>キーワードエディタおよびアプリケーション内で、キーワードを大文字に変換するかどうかを指定します。</p> <p>Y キーワードエディタで入力されたキーワードは、自動的に大文字に変換されます。アプリケーションでは、エンドユーザーはキーワードを大文字でも小文字でも入力できます。</p> <p>N キーワードエディタで入力されたキーワードは、大文字に変換されません。アプリケーションでは、エンドユーザーはキーワードエディタに実際に表示されるとおりにキーワードを入力する必要があります。</p>
Unique Keywords	<p>プロセッサ内のキーワードを一意にする必要があるかどうかを指定します。</p> <p>Y 各キーワードは、そのタイプに関係なくプロセッサ内で一意に定義する必要があります。</p> <p>N 各キーワードは、指定のキーワードタイプ (1、2、3、またはP) ごとに一意に定義する必要があります。</p>

### その他のオプション・ヘッダー 3

[Processor Header Maintenance 3] 画面では、その他のオプションを入力できます。

フィールド	説明
Invoke Action Editor	<p>機能エディタからランタイムアクションエディタを呼び出すかどうかを指定します。「<a href="#">ランタイムアクションエディタ</a>」および「<a href="#">機能定義</a>」を参照してください。</p> <p>Y 有効なキーワード結合を機能エディタで定義すると、必ずランタイムアクションエディタが呼び出されます。</p> <p>N 機能エディタでは、ランタイムアクションエディタは抑制されます。</p> <p><b>注意:</b> Natural システムライブラリ SYSNCP で提供されているユーザー出口 NCP-REDM を使用する場合、このオプションは Y に設定する必要があります。Y に設定しないと、不正なランタイムアクションの値をこの時点で検出できず、ランタイムエラーを引き起こす可能性があります。</p>
Catalog User Texts	<p>ユーザーテキストをコマンドプロセッサとともにカタログするかどうかを指定します。</p> <p>Y コマンドプロセッサをカタログするときに、編集行 (キーワードエディタ) (「<a href="#">キーワード定義</a>」を参照) のテキスト部分、およびアクション行 (ランタイムアクションエディタ) のユーザーテキスト部分が、関連するキーワードまたは機能に結合されます。その後、このテキストは、PROCESS COMMAND ステートメントの TEXT オプションを使用して、ランタイム時に読み込むことができます。</p> <p>N テキストはコマンドプロセッサとともにカタログされず、ランタイム時に読み込むこともできません。</p>

フィールド	説明
Security Prefetch	<p>セキュリティチェックを、ランタイム中最初にコマンドプロセッサが呼び出されたときに実行するか、または各コマンドの評価時に実行するかを指定します。</p> <p>Y Natural Security がインストールされている場合、セキュリティチェックは、コマンドプロセッサが呼び出されたときにすべてのキーワードに対して実行されます。</p> <p>N Natural Security がインストールされている場合、セキュリティチェックは、各キーワードの評価と一緒に実行されます。</p> <p>オプションYを選択した場合、セキュリティチェックは、コマンドプロセッサが呼び出されたときにすべてのキーワードに対して1回のみ実行されます。チェック処理は時間がかかるため、ランタイム時の最初のコマンドの評価は比較的時間がかかりますが、残りすべてのコマンドの評価は比較的速く処理されます。逆に、オプションNを選択した場合、セキュリティはコマンドが評価される前にキーワードごとに個別にチェックされるため、各コマンドの評価時間は常に同じとなります。</p>
Command Log Size	<p>ランタイム時に処理されたコマンドは、コマンドプロセッサによってコマンドログエリアに保存されます。コマンドログに割り当てるストレージスペースをKB数で入力フィールドに指定します。</p> <p>0 コマンドログにストレージスペースを割り当てません。コマンドログは作動しません。</p> <p>1 コマンドログに1KBのストレージスペースを割り当てます。コマンドログは作動します。</p>
Implicit Keyword Entry	<p>後続するすべてのコマンドに対する暗黙的なキーワードとして、タイプ1のキーワードを保持するかどうかを指定します。</p> <p>1 タイプ2のキーワードのみを含むコマンドを入力した場合、コマンドプロセッサによって、最後に入力したタイプ1のキーワードが暗黙的なキーワードとみなされます。</p> <p>N オプションは無効です。</p>
Command Delimiter	<p>コマンド行に複数のコマンドを指定する場合は、コマンドを区切るために使用する文字を指定します。ランタイム時には、先頭のコマンドのみが実行されます。</p> <p>例：</p> <p>DISPLAY CUSTOMER; MODIFY CUSTOMER; PRINT</p>
PF-Key may be Command	<p>PF キーにコマンドを割り当てることができるかどうかを指定します。ランタイム時にコマンドプロセッサですべて空白のコマンド行を受け取ると、ユーザーによってPFキーが押されたかどうかチェックされます。</p> <p>可能な値は次のとおりです。</p> <p>A このPFキーのID（システム変数 *PF-NAME）をコマンドとして使用します。</p> <p>K *PF-KEY システム変数の内容をコマンドとして使用します。</p> <p>Y *PF-NAME が空の場合、*PF-KEY システム変数の内容を代わりに使用します。</p>



フィールド	説明
	<p>N PFキーはコマンドとして使用できません。"コマンド行を受け入れられません"というメッセージとともに Natural エラー NAT6913 が発行されます。</p> <p>システム変数 *PF-NAME および *PF-KEY の詳細については、『システム変数』ドキュメントを参照してください。</p>

#### コマンドデータ処理 - ヘッダー 4

【Processor Header Maintenance 4】画面に入力する属性で、機能に対するコマンドデータの処理方法を指定します。コマンドデータはオプションです。

オプションは以下のとおりです。

フィールド	説明
Data Delimiter	<p>データの前に付ける文字を指定します。デフォルトのデータデリミタは "#" です。</p> <p>例：ADD CUSTOMER #123</p>
Data Allowed	<p>ランタイム時にデータ入力を認めるかどうかを指定します。</p> <p>N データが見つかり、ランタイムエラーが発生します。</p> <p>D データが存在する場合には取り除かれます。</p> <p>S データは Natural スタックの最上部に表示されます。チェックは実行されません。</p> <p>Y データはチェックされ、タイプ P (パラメータインジケータ) のキーワードエントリが評価されます。</p> <p>Y を指定した場合の例：DISPLAY CUSTOMER NAME=SMITH</p>
More than one Item Allowed	<p>適用されるのは、[Data Allowed] オプションが Y に設定されている場合に限られます。複数のデータ文字列を許可するかどうかを指定します。</p> <p>N 複数のデータ文字列が見つかり、ランタイムエラーが発生します。</p> <p>D 先頭のデータ文字列より後の全データが取り除かれます。</p> <p>Y 複数のデータ文字列を許可します。</p> <p>例：ADD ARTICLE #111 #222</p> <p>一意性が保証される場合に限り、データデリミタを省略できます。</p> <p>例：ADD ARTICLE 123</p>
Maximum Length of one Item	<p>適用されるのは、[Data Allowed] オプションが Y に設定されている場合に限られます。</p> <p>データ文字列に許可される最大文字数を指定します。指定の最大文字数を超過すると、ランタイムエラーが発生します。有効範囲は 1~99 です。</p>

フィールド	説明
Item Must be Numeric	適用されるのは、[Data Allowed] オプションがYに設定されている場合に限りま す。各データ値を整数値にする必要があるかどうかを指定します。  Y データ入力は正の整数値にする必要があります。これに反すると、ランタイムエ ラーが発生します。 N データはどのようなタイプでも許可されます。
Put to Top of Stack	適用されるのは、[Data Allowed] オプションがYに設定されている場合に限りま す。データを配置する場所を指定します。  Y データは Natural スタックの最上部に表示されます。 1-9 データは DDM フィールド RESULT-FIELD の <i>n</i> 番目のオカレンスに配置されま す。ランタイムアクションの結果としてオカレンスにすでに値が入っていた場 合は、上書きされます。
If Error, Drop all Data	適用されるのは、[Data Allowed] オプションがYまたはNに設定されている場合 に限ります。データ評価エラーに対する処理を指定します。  Y データの評価時にエラーが発生した場合、データは破棄され、処理は続行します。 N データの評価時にエラーが発生した場合、下記の説明のようにエラーハンドラに 制御が渡ります。

## ランタイムエラー処理 - ヘッダー 5

[Processor Header Maintenance 5] 画面に入力する属性で、ランタイムエラーの処理方法を指  
定します。

フィールド	説明
General Error Program	ランタイム処理中にコマンドプロセッサによってエラーが検出された場合に、 制御を受け取るプログラムの名前。このプログラムが呼び出される時、Natural スタックには次の情報が格納されています。  エラー番号 (N4) 行番号 (N4) ステータス (A1) プログラム名 (A8) レベル (N2)  エラープログラムも特定のエラー処理も指定しなかった場合 (下記参照)、 Natural システム変数 *ERROR-TA に含まれている名前のプログラムが呼び出さ れます。それ以外の場合には、Natural システムエラーメッセージが発行されま す。
Keyword not found	キーワードを見つけられなかった場合に実行するアクションが指定されている かどうかを示します。

フィールド	説明
Keyword missing	キーワードタイプがなかった場合に実行するアクションが指定されているかどうかを示します。
Keyword Sequence Error	キーワードシーケンスエラーの場合に実行するアクションが指定されているかどうかを示します。
Command not defined	未定義コマンドの場合に実行するアクションが指定されているかどうかを示します。
Data disallowed	禁止されたデータの場合に実行するアクションが指定されているかどうかを示します。
Data Format/Length Error	フォーマット/長さエラーの場合に実行するアクションが指定されているかどうかを示します。
General Security Error	全体的なセキュリティチェック中にエラーが検出された場合に実行するアクションが指定されているかどうかを示します。
Keyword Security Error	キーワードのセキュリティチェック中にエラーが検出された場合に実行するアクションが指定されているかどうかを示します。
Command Security Error	コマンドのセキュリティチェック中にエラーが検出された場合に実行するアクションが指定されているかどうかを示します。

## 統計 - ヘッダー 6

[**Processor Header Maintenance 6**] 画面には、コマンドプロセッサに指定したキーワードに関する統計データをレポートする出力フィールドのみが表示されます。

次の統計情報が表示されます。

フィールド	説明
Entry <i>n</i> Keywords	コマンドプロセッサに定義されたタイプ <i>n</i> のキーワードの数。シノニムは含まれません。
Entry <i>n</i> Keywords + Synonyms	タイプ <i>n</i> のキーワードとそのシノニムの合計。
Highest IKN for Entry <i>n</i>	キーワードタイプ <i>n</i> の最大内部キーワード番号。
Possible Combinations	定義されたキーワードの可能な結合の数。
Cataloged Functions	現在カタログされているキーワード結合の数。

## ステータス・ヘッダー 7

「Processor Header Maintenance 7」画面には、コマンドプロセッサの各部分が実行または修正されたときの日付と時刻をレポートする出力フィールドのみが表示されます。

## キーワードのメンテナンス

キーワードは、機能を定義するための基本的なコンポーネントです。キーワードを定義する前に、ヘッダーメンテナンスレコードを作成する必要があります。「[ヘッダーレコード](#)」を参照してください。

- [キーワード定義](#)
- [エディタコマンド](#)
- [位置調整コマンド](#)
- [行コマンド](#)

### キーワード定義

コマンドで使用するキーワードは、キーワード定義機能とキーワードエディタで作成します。キーワードエディタは、エディタ行が別々のフィールドに分割されていること以外は、既存の Natural エディタと類似しています。Natural プログラムエディタで使用する [エディタコマンド](#)（関連するセクションを参照）および [行コマンド](#)（関連するセクションを参照）の大部分は、キーワードエディタでも使用できます。

#### ▶手順 100.8. キーワードエディタを呼び出すには

- 1 「Processor Source Maintenance」メニューで、ファンクションコード「**K**」（Define Keywords）を入力します。
- 2 Enter キーを押します。

キーワードエディタ画面が表示されます。

以下のキーワードエディタ画面が表示されます。このセクションの例として使用するキーワードがすでにいくつか定義されています。

```

09:42:39                - SYSNCP Keyword Editor -                2000-05-04
Modify Keywords          Name SAGTEST   Library SYSNCP   DBID 10   FNR 32

I Line E Use  Keyword                IKN   ML Comment
-----
1 1 Acti MENU                1004   1


```

```

2 1 Acti DISPLAY          1002  2
3 S Syno SHOW            1002  1
4 1 Acti DELETE          1001  2
5 S Syno PURGE           1001  1
6 S Syno ERASE           1001  1
7 1 Acti FILE            1003  4
8 P Parm NAME            4002  2
9 2 Obje FILE            2001  4
10 P Parm NUMBER         4001  2
11 2 Obje DOCUMENT       2003  2
12 1 Acti INFORMATION    1005  1
13
14
----- All -----
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Cmd  Exit  Last  List  Flip  -1   +1   Top  Bot  Info  Canc

```

コマンド言語に必要なすべてのキーワードをキーワードエディタに入力します。キーワードは任意の順番で入力できます。ただし、シノニムは関連するキーワードの直後に続ける必要があります。各キーワードに対して、そのキーワードがコマンド構文のどの部分に属するかを指定するタイプを割り当てます。コマンドプロセッサのコマンド構文規則は、プロセッサヘッダーで指定します。「ヘッダーレコード」の「[キーワードランタイムオプション-ヘッダー-1](#)」を参照してください。例えば、キーワードを、タイプ1（コマンドの先頭に入力）、タイプ2、タイプ3、別のキーワードのシノニム、またはパラメータインジケータのいずれにするかを指定できます。

 **注意:** 現在のところ、どのようなコンピュータにも意味論を理解する能力はないため、コマンド言語には厳密な構文が必要です。そのため、言葉による入力が、コマンド言語でやり取りする唯一の実用的な方法となっています。

上記の例では、キーワード DELETE と DISPLAY がタイプ1のキーワードとして定義されています。プロセッサヘッダーで指定したとおり、これらのキーワードはアクションを示します。キーワード DOCUMENT はタイプ2のキーワードとして定義されており、オブジェクトを示します。ただし、キーワード FILE はタイプ1と2の両方として定義されているため、コマンド内の位置に応じてアクションまたはオブジェクトを示します。2つのキーワードタイプを結び付けて DELETE FILE や FILE DOCUMENT などのコマンドを作ることができます。

コマンド行から SAVE または STOW コマンドを発行して、入力したキーワードを保存することができます。STOW コマンドは、キーワード定義をソース形式で保存することに加えて、それらの整合性チェックも実行します。キーワードが正常に格納されると、ランタイム時にコマンドを評価するために使用される内部キーワード番号 (IKN) がキーワードに与えられます。シノニムは常にマスタキーワードと関連付けられ、マスタの IKN を取得します。

キーワードエディタの各行には以下のフィールドが含まれています。

フィールド	説明
I	出力フィールド。この情報フィールドには次の値が表示されます。  E 定義エラーが検出されたことを示します。 X 行が X でマークされています。 Y 行が Y でマークされています。 Z 行が X と Y の両方でマークされています。 S この行にスキャン対象値が見つかりました。
Line	出力フィールド。エディタの行番号が表示されます。
E	キーワードのエントリタイプ、つまり、キーワードをコマンドに入力する位置を指定します。エントリタイプは、1 番目、2 番目、3 番目、シノニム、またはパラメータインジケータのいずれかです。  例えば、上記の <b>キーワードエディタ画面</b> の例では、キーワード DELETE はエントリタイプ 1、DOCUMENT はエントリタイプ 2 になっています。これらのキーワードを使用すると、DELETE DOCUMENT コマンドを定義できます。  このフィールドに入力できる文字は以下のとおりです。  1 この行に定義したキーワードは、コマンドシーケンスの 1 番目の項目として使用されます。 2 この行に定義したキーワードは、コマンドシーケンスの 2 番目の項目として使用されます。 3 この行に定義したキーワードは、コマンドシーケンスの 3 番目の項目として使用されます。 S この行に定義したキーワードは、前の行に指定したタイプ 1、2、3、または P のキーワードのシノニムとして使用されます。 P この行に定義したキーワードは、コマンドシーケンスでパラメータインジケータとして使用されます。 * この行にはキーワードを定義しません。代わりに、この行はコメント行としてのみ使用されます。 ? この記号は、キーワード指定が誤りであることを示す表示値です。
Use	出力フィールド。このフィールドに表示される値は、[E] フィールドに入力した値で決まります。  1~3 1~3 番目の各キーワードエントリの場合は、プロセッサヘッダーに指定したユーザーテキストの先頭 4 文字がそれぞれ表示されます。「ヘッダーレコード」の「 <b>キーワードエディタオプション-ヘッダー-2</b> 」も参照してください。 S シノニムの省略形である SYNO が表示されます。 P パラメータインジケータの省略形である PARM が表示されます。
Keyword	定義するキーワードを入力します。空白で埋めることはできません。プロセッサヘッダーで、キーワードに大文字のみ使用できるように指定した場合、キーワードは、どのように入力され

フィールド	説明
	<p>たかに関係なく、常に大文字に変換されます。それ以外の場合、大文字／小文字は入力されたままとなります。</p> <p>キーワードの最大文字数と最小文字数は、ヘッダーに指定した設定で決まります。デフォルトは1~16文字です。キーワードは、ヘッダーにその他の指定がない限り、一意にする必要があります。「ヘッダーレコード」の「<a href="#">キーワードエディタオプション-ヘッダー2</a>」で説明されているように、キーワード接頭辞を使用できます。</p>
IKN	<p>出力フィールド。内部キーワード番号 (IKN) は、有効な各キーワードに割り当てられる識別子です。IKN は、テストおよびデバッグに役立ちます。IKN は、キーワードが正常に格納されたときのみ割り当てられます。「エディタコマンド」の <a href="#">STOW</a> コマンドも参照してください。各キーワードに一意の IKN が割り当てられます。ただし、シノニムにはマスタの IKN が割り当てられます。上記の <a href="#">キーワードエディタ画面</a> の例の DISPLAY および SHOW を参照してください。</p>
ML	<p>入出力フィールド。キーワードの最小文字数を示します。「ヘッダーレコード」の「<a href="#">キーワードランタイムオプション-ヘッダー1</a>」で説明されているように、プロセッサヘッダーの <a href="#">[Dynamic Length Adjustment]</a> フィールドに S を指定していると、このフィールドは入力フィールドになります。この場合、キーワードに必要な入力文字数を指定する必要があります。S 以外の場合は、他のキーワードとの混同を避けるために、ユーザーが指定する必要があります。キーワードの最小文字数が、このフィールドに表示されます。</p> <p>例えば、上記の <a href="#">キーワードエディタ画面</a> の例の場合、キーワード MENU に必要な入力は <b>M</b> のみですが、キーワード DISPLAY は、キーワード DELETE との混同を避けるために、「<b>DI</b>」と入力する必要があります。</p>
Comment	<p>キーワードに関するテキストを自由に入力できます。入力に対する制限はありません。「ヘッダーレコード」の「<a href="#">その他のオプション-ヘッダー3</a>」で説明されているように、ヘッダー定義の <a href="#">[Catalog User Texts]</a> フィールドを Y に設定した場合、このユーザーテキストはカタログされたコマンドプロセッサに組み込まれます。このユーザーテキストは、PROCESS COMMAND ステートメントの TEXT オプションを使用して、ランタイム時に読み込むことができます。この列の先頭に表示されるヘッダーテキストは、ヘッダー定義の <a href="#">[Header for User Text 1]</a> および <a href="#">[Header for User Text 2]</a> フィールドで制御されます。</p>

## エディタコマンド

キーワードエディタのコマンド行に入力できるコマンドは以下のとおりです。

コマンド	機能
ADD	エディタの最後の部分に空行を 10 行追加します。
CANCEL	プロセッサメンテナンスメニューに戻ります。
CHECK	キーワードソースの整合性をテストします。
EXIT	プロセッサメンテナンスメニューに戻ります。
HELP	有効なエスケープ文字および他の有効なプロセッサ設定を表示します。
INFO	カーソルが位置付けられているキーワードに関する情報を表示します。

コマンド	機能
LET	最後に Enter キーを押して以降、現在の画面に行ったすべての修正を無効にします。
POINT	行コマンド .N が入力された行を現在の画面の先頭に位置付けます。
RECOVER	最後の SAVE または STOW の前に存在したキーワードソースに戻します。
RESET	現在の X および Y 行マークを削除します。
SAVE	キーワードソースを保存します。
SCAN	スキャンを実行して、スキャン値の次のオカレンスを探します。
STOW	キーワードソースが格納され、有効なキーワードに対して内部キーワード番号 (IKN) が生成されます。

### 位置調整コマンド

エディタの位置付けコマンドは、Naturalプログラムエディタの位置付けコマンドと同じです。詳細については、『エディタ』ドキュメントの「プログラムエディタ」を参照してください。

エディタの最後の行には、エディタ内の表示位置を示す出力フィールドが含まれています。以下の出力値が表示されます。

Top	エディタは現在キーワードソースの先頭に位置付けられています。
Mid	エディタは現在キーワードソースの中央に位置付けられています。
Bot	エディタは現在キーワードソースの最後に位置付けられています。
Emp	エディタは現在空白です。
All	ソース全体が現在の画面に含まれています。

### 行コマンド

キーワードエディタの行コマンドは、コマンド J および S を使用できないこと以外は、Naturalプログラムエディタと同じです。


各コマンドは [E] フィールドから入力します。コマンドの残りの部分は、次の画面のように、[Keyword] フィールドに入力します。

```

09:42:39                - SYSNCP Keyword Editor -                2000-05-04
Modify Keywords          Name SAGTEST   Library SYSNCP   DBID 10   FNR 32

I Line E Use  Keyword                IKN   ML Comment
-----
  1 1 Acti MENU                1004   1
  2 1 Acti DISPLAY            1002   2
  3 S Syno SHOW                1002   1
  4 . Acti i(3)TE            1001   2
    
```



 **注意:** 行を移動 (.M) またはコピー (.C) する場合は、個々のキーワードを常にシノニムと一緒に移動またはコピーするようにします。

行を削除 (.D) する場合、対応するキーワードおよびこれらのキーワードを含むすべての機能は、STOW エディタコマンドを発行するまで、データベースから削除されません。STOW コマンドを発行しない限り、これらの機能は機能エディタに表示され続けます。

## 機能のメンテナンス

機能は、キーワードエディタに入力したキーワードで構成されます。機能を定義する前に、キーワードが正常に格納されている必要があります。「[キーワードのメンテナンス](#)」を参照してください。

- [機能定義](#)
- [エディタコマンド](#)
- [ダイレクトコマンド QUICK-EDIT](#)
- [ローカル機能とグローバル機能](#)
- [機能を有効にする手順](#)

### 機能定義

機能定義機能および機能エディタを使用して機能を指定し、特定のロケーションからアクセスできる有効なコマンドを構成します。

#### ▶手順 100.9. 機能エディタを呼び出すには

- 1 [Processor Source Maintenance] メニューで、ファンクションコード「F」 (Define Functions) を入力します。
- 2 Enter キーを押します。

機能エディタ画面が表示されます。

機能エディタには、キーワードエディタで格納したキーワードの有効な結合がすべて表示されます。

次の画面は機能エディタを示しています。この画面には、「キーワードのメンテナンス」のキーワードエディタ画面で例として使用されているキーワードが含まれています。

```

09:45:53          ***** NATURAL SYSNCP UTILITY *****          2000-05-04
User SAG          - Function Editor -
Edit Global Combinations      Name SAGTEST  Library SYSNCP      DBID 10      FNR 32

Global

I Ac   Action          Object          Addition          Global Local Any Loc
-----
      DELETE
      DELETE          DOCUMENT          Yes
      DELETE          FILE              Yes
      DISPLAY
      DISPLAY          DOCUMENT          Yes
      DISPLAY          FILE              Yes
      FILE
      FILE             DOCUMENT          Yes
      FILE             FILE              Yes
      INFORMATION          Yes
      INFORMATION          DOCUMENT
      INFORMATION          FILE

Repos: _____

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Cmd  Exit Last List Flip      +      Top   Loc  Loc+  Canc
    
```

アプリケーション内の有効な機能として指定するキーワード結合を、それぞれ有効にする必要があります。キーワード結合は、グローバル機能、ローカル機能、またはその両方として有効にできます。グローバル機能はアプリケーション内のどこからでも呼び出すことができますが、ローカル機能はアプリケーション内の特定のロケーションからのみ呼び出すことができます。

機能エディタ画面の左上隅に表示されている2つのフィールドは、現在有効なモード（ローカルまたはグローバル）および現在キーワード結合を有効にできるロケーションを示しています。上の画面の "Edit Global Combinations" というテキストは、グローバルモードが有効になっていることを示しています。ローカルモードが有効になっている場合は、ここに "Edit Local Combinations" というテキストが表示されます。上の画面では、このテキストの下に "Global" というテキストが表示されています。これは、リストされているすべての結合をグローバルとして有効にできることを示しています。ローカルモードでは、このフィールドに、ローカルとして有効にできるロケーションの名前が表示されます（例："Local DISPLAY FILE"）。

機能エディタには次の列が含まれています。

列	説明
I	出力フィールド。機能編集の結果として次の値が出力されます。  E ランタイムアクションが編集されました。 D 参照されるロケーションが表示されました。 V 有効性が発行されました。 R 有効性が削除されました。
Ac	実行するアクション。次の値を入力できます。  VG グローバル機能として有効にします。 VL ローカル機能として有効にします。 RG グローバル機能としての有効性を削除します。 RL ローカル機能としての有効性を削除します。 DL 指定した機能をローカル機能として参照するすべての機能を表示します。 EG グローバル機能のランタイムアクションエディタを呼び出します。「ランタイムアクション」の「ランタイムアクションエディタ」を参照してください。 EL ローカル機能のランタイムアクションエディタを呼び出します。「ランタイムアクション」の「ランタイムアクションエディタ」を参照してください。 +G 任意のグローバル機能をメンテナンスできるように、グローバルモードを呼び出します。 +L 現在の行のローカル機能をメンテナンスできるように、この行のローカルモードを呼び出します。 IN この行にキーワードの情報が表示されます。
Action	この3つの列を使用して、現在定義されているキーワードの有効な結合をすべて表示します。
Object	各キーワード列の上部に表示されるテキストは、プロセッサヘッダーで指定した <b>[First Entry used as]</b> 、 <b>[Second Entry used as]</b> 、および <b>[Third Entry used as]</b> の各フィールドで制御されます。「ヘッダーレコード」の「 <b>キーワードランタイムオプション-ヘッダー1</b> 」を参照してください。
Addition	
Global	機能がグローバルコマンドとして定義されている場合、このフィールドに Yes が表示されます。
Local	機能がローカルコマンドとして定義されている場合、このフィールドには現在のロケーションに対する Yes が表示されます。ただし、ローカルモードの場合のみです。
Any Loc	任意のロケーション。機能がプロセッサ内の別のどこかでローカルコマンドとして定義されている場合、このフィールドには他のいずれかのロケーションに対する Yes が表示されます。

## エディタコマンド

機能エディタのコマンド行に入力できるコマンドは以下のとおりです。

コマンド	機能
ANY ON	[Any Loc] 列を使用できるようにします。
ANY OFF	[Any Loc] 列を使用できないようにします。列は疑問符で埋められます。これにより、機能エディタ内のスクロールが速くなります。さらに、3番目の再位置付けフィールドを使用できます。進行中の処理情報ウィンドウは表示されません。
FIELD	キーワード特定の結合を表示します。
GLOBAL	グローバルモードを有効にします。
LOC	次のロケーショングループに位置付けます。
LOC+	1 ロケーション先に位置付けます。
SINGLE ON	単一ワード機能のみを表示します。
SINGLE OFF	使用可能なすべての結合を表示します。
TOP	リストの先頭に位置付けます。

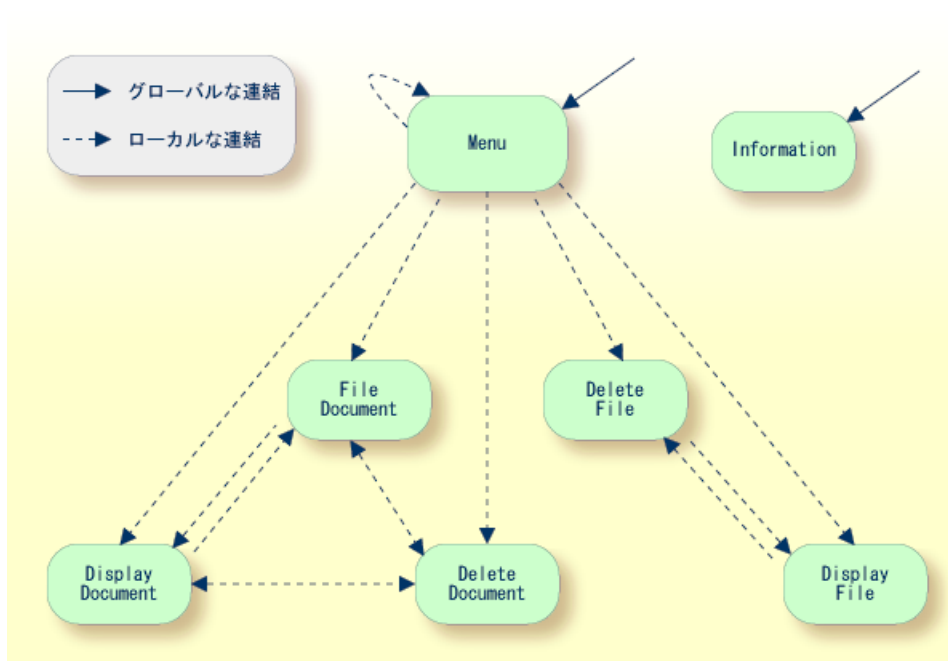
## ダイレクトコマンド QUICK-EDIT

ダイレクトコマンド QUICK-EDIT を使用すると、キーワードまたは IKN を直接入力することで、ローカル機能およびグローバル機能と、対応するランタイムアクションをすばやく定義することができます。これは、非常に大きいコマンドプロセッサに役立ちます。ただし、コマンドを発行できるロケーションが検証されないため、ランタイム時にナビゲーションが正しく機能しない場合があることに注意してください。

## ローカル機能とグローバル機能

ローカル機能とグローバル機能の概念を理解するには、有効な各キーワードの結合をアプリケーション内のロケーションとして図に示す必要があります。例えば、Display File という名前のロケーションとして示します。機能エディタでは、このロケーションから発行できるコマンドと、DISPLAYFILE コマンドを使用することでこのロケーションに到達できるロケーションを指定します。

サンプルアプリケーション内のローカルおよびグローバルな連結：



上記のサンプルアプリケーションでは、Menu ロケーションと Information ロケーションのみが、グローバルとして指定されています。そのため、これらのロケーションは、サンプルアプリケーション内の残りのすべてのロケーションから直接アクセスできます。Information を除く全ロケーションは、Menu ロケーションに対してローカルとして指定されています。Display File ロケーションから Display Document ロケーションに到達するには、Menu を経由する方法しかありません。

### 機能を有効にする手順

機能エディタは、グローバルモードとローカルモードの2つのモードで動作します。グローバルモードではグローバル機能を有効にし、ローカルモードではグローバル機能とローカル機能を有効にすることができます。グローバルモードがデフォルトのモードです。エディタの [I] フィールドの上にある出力フィールドで、エディタがグローバルモードであるかローカルモードであるかを確認できます。エディタがグローバルモードの場合は、Global と表示されます。エディタがローカルモードの場合には、ローカル機能が有効になるロケーションが表示されます。アプリケーションに対してグローバル機能とローカル機能を有効にする一般的な手順は、次のとおりです。

#### ▶手順 100.10. グローバル機能とローカル機能を有効にするには

- 1 グローバルモードの機能エディタで、該当するアクションの隣の [Ac] フィールドに「VG」（グローバルを有効にする）を入力して、すべてのグローバル機能を有効にします。

Enter キーを押します。

[Runtime Action Definition] 画面が表示されます。

- 2 PF3 キーを押して、機能エディタに戻ります。

ヘッダー名 [Global] 列の下、検証された機能の横に [Yes] が表示されます。

- 3 前の手順で有効にした各グローバル機能の [Ac] フィールドに「+L」を入力して、ローカルモードに切り換えます。

Enter キーを押します。

- 4 このグローバル機能のロケーションとして使用する各機能の [Ac] フィールドに、「VL」（ローカルを有効にする）を入力します。

Enter キーを押します。

[Runtime Action Definition] 画面が表示されます。

- 5 PF3 キーを押して、機能エディタに戻ります。

ヘッダー名 [Local] 列の下、検証された機能の横に [Yes] が表示されます。

- 6 ローカルロケーションに対してローカル機能を有効にするには、前の手順で有効にした各ロケーションの [Ac] フィールドに「+L」（ローカルモードを呼び出す）を入力して、このロケーションから使用するすべてのローカル機能を有効にします。

ENTER キーを押します。

- 7 現在のロケーションに対するローカル機能として使用する各機能の [Ac] フィールドに、「VL」（ローカルを有効にする）を入力します。

- 8 PF3 キーを押して、機能エディタに戻ります。

ヘッダー名 [Local] 列の下、検証された機能の横に [Yes] が表示されます。



**注意:** コマンドプロセッサヘッダー（ [Processor Header Maintenance 3] ）の [Invoke Action Editor] フィールドを Y に設定していると、 [Runtime Action Definition] ウィンドウがアクションごとに表示されます。「ランタイムアクション」の「ランタイムアクションエディタ」を参照してください。

## ランタイムアクション

機能エディタで有効なキーワード結合をローカル機能またはグローバル機能として特定した後、各機能を1つ以上のランタイムアクションと関連付けることができます。ランタイムアクションは、機能が発行されると常に実行される1つ以上の手順で構成されます。

以下に参考情報を示します。

- [ランタイムアクション定義](#)

## ■ ランタイムアクションエディタ

### ランタイムアクション定義

SYSNCP には、ランタイムアクションを定義できる場所が2つあります。機能エディタ（「機能のメンテナンス」を参照）と結果エディタです。このセクションでは、結果エディタについて、機能に対するランタイムアクションの指定方法を説明します。

#### ▶手順 100.11. 結果エディタを呼び出すには

- 1 **[Processor Source Maintenance]** メニューで、ファンクションコード「R」（Define Runtime Actions）を入力します。
- 2 Enter キーを押します。

結果エディタ画面が表示されます。

```

09:47:03          ***** NATURAL SYSNCP UTILITY *****
2000-05-04
User SAG          - Result Editor -

List defined combinations   Name SAGTEST   Library SYSNCP   DBID 10   FNR
32

I Ac Location          Command          Result
-----
< Global >            MENU            KR
< Global >            INFORMATION     SF
DELETE FILE           DISPLAY FILE     SF
DELETE DOCUMENT      DISPLAY DOCUMENT SF
DISPLAY FILE         DELETE FILE     SF
DISPLAY DOCUMENT    DELETE DOCUMENT SF
DISPLAY DOCUMENT    FILE DOCUMENT   SF
FILE DOCUMENT       DELETE DOCUMENT SF
FILE DOCUMENT       DISPLAY DOCUMENT SF
MENU                DELETE FILE     KCS

```

```

MENU                                DELETE DOCUMENT                    KCCS
MENU                                DISPLAY FILE                        KRCS

Repo _____
-----
Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Cmd  Exit Last List Flip      +      Top  Loc-- Loc+  Canc
    
```

結果エディタには、機能エディタで指定したすべてのローカル機能とグローバル機能が表示されます。エディタの各行には、コマンドを発行できるロケーション（[Location] フィールド）、コマンド（[Command] フィールド）、およびコマンドが発行されたときに実行されるアクションの短縮形（[Result] フィールド）が表示されます。

次の表は画面フィールドの説明です。

フィールド	説明
I	出力フィールド。この行で最後に実行されたアクションについての情報。
Ac	実行するアクション。次の値を入力できます。  DI この機能に対するランタイムアクション定義を表示します。 ED この機能に対するランタイムアクション定義を編集します。 PU この機能を削除します。
Location	出力フィールド。  コマンド（下記の [Command] フィールドを参照）を発行できるアプリケーション内のロケーション。機能がグローバルの場合、このフィールドに <Global> が表示されます。コマンドは任意のロケーションから発行できます。
Command	出力フィールド。コマンド。  長いキーワードを使用しているときには、[Location] および [Command] フィールドの内容が切り捨てられる場合もあります。
Result	出力フィールド。  コマンドが発行されたときに実行されるアクションの省略形が表示されます。先頭文字は <b>Keep Location</b> 情報（下記参照）を示します。他の文字については、下記の <b>ランタイムアクション定義</b> の表を参照してください。



## ランタイムアクションエディタ

ランタイムアクションエディタを使用して、特定のロケーションからコマンドが発行されたときに実行するアクションを定義します。ランタイムアクションエディタは、グローバルまたはローカル機能として定義された機能に対してのみ呼び出されます。このエディタは、機能エディタまたは結果エディタから呼び出すことができます。

### ▶手順 100.12. 機能エディタからランタイムアクションエディタを呼び出すには

- 1 グローバル機能の場合は、[Ac] フィールドに「EG」（グローバル編集）を入力します。

または:

ローカル機能の場合は、[Ac] フィールドに「EL」（ローカル編集）を入力します。

- 2 Enter キーを押します。

### ▶手順 100.13. 結果エディタからランタイムアクションエディタを呼び出すには

- 1 [Ac] フィールドに、「ED」を入力します。

- 2 Enter キーを押します。

[Runtime Action Definition] ウィンドウが表示されます。

```

Runtime Action Definition

Location .... DISPLAY DOCUMENT
Command ..... DELETE DOCUMENT

Keep Location .... S
Data allowed ..... Y   More than one .... N   Max. Length ..... 99
Numeric ..... N   TOP of STACK ..... Y   Error: Drop ..... Y

A Runtime Action Definition
- -----
F DE-PGM_____
- _____
- _____
- _____
- _____
- _____
- _____

```

アクションは常に発生元と宛先に関連付けられます。発生元はコマンドが発行されるロケーションであり、宛先はコマンド自体です。このように、コマンドが使用される関係を基準にして、1つのコマンドに異なるアクションを関連付けることができます。

ランタイムアクションエディタでは、アクションが実行された後ロケーションをそのままにするか、またはコマンド自体を新たな現行ロケーションにするかも指定します。

アクションは、エディタの左端の列に1文字コードを入力して指定します。コードの隣のフィールドに、アクションに付随するすべてのパラメータを入力します。このフィールドに「\*」を入力すると、後続の入力はすべてコメントとみなされます。必要なパラメータを省略すると、入力を求めるメッセージが表示されます。

ランタイム時に実行されるアクションの順番は、エディタのエントリ順序（上から下へ）で決定されます。そのため、FETCHが指定されると、その下に指定されたいずれのアクションも実行されません。

ランタイムアクションエディタのフィールドは次のとおりです。

フィールド	説明
Location	出力フィールド。コマンドを発行するロケーション。機能がグローバルとして定義されている場合、<Global>が表示されます。
Command	出力フィールド。このコマンドに対するアクションを指定します。
Keep Location	<p>アクションが実行された後、現在のロケーションをアクティブにするか、または新しいロケーションをアクティブにするかを指定します。このフィールドの値は、EXECオプションを指定した場合にのみコマンドに影響します。可能な値は次のとおりです。</p> <p>K 現在のロケーションを保持します。実行するアクションは、現在のロケーションにのみ影響します。</p> <p>S 新しいグローバルまたはローカルロケーションを設定します。アクションを実行すると、コマンドプロセッサはそのコマンドを新たな現行ロケーションにします。この後に入力するコマンドはすべて、この新ロケーションのローカルコマンドまたはグローバルコマンドのいずれかである必要があります。</p> <p><b>注意:</b> 定義したアクション自体は、ロケーションに影響を与えません。つまり、実行するアクションは現在のロケーションを変更しません。</p>
Other Options	<p>その他のオプションはすべて、このコマンドシーケンスで提供されるパラメータの処理と関連します。詳細については、「ヘッダーレコード」の「<a href="#">コマンドデータ処理-ヘッダー4</a>」を参照してください。</p> <p>これらのオプションのヘッダーデフォルトを有効にするには、「*」（アスタリスク）を入力します。</p>

#### ▶手順 100.14. ランタイムアクションを定義するには

- 1 前述のように、「**Runtime Action Definition**」ウィンドウを呼び出します。

- 2 フィールド [A] にアクションコードを入力し、対応するフィールドにアクションを入力します。

コード	ランタイムアクション定義
V	デフォルト値です。ランタイムアクションを指定しません。
T	PROCESS COMMAND ステートメントの TEXT または GET オプションを使用して、ランタイム時に読み込めるテキストです。
M	コマンド行を修正します。このデータはコマンド行に置かれます。
C	コマンド。このコマンドは Natural スタックの先頭に置かれます。アスタリスク (*) を指定した場合、この PROCESS COMMAND ステートメントを発行したプログラムの名前がスタックの先頭に置かれます (STACK TOP COMMAND *PROGRAM)。(*)
D	データ。このデータは Natural スタックの先頭に置かれます。(*)
F	Natural プログラム名。このプログラムに対して FETCH ステートメントが実行されます。(*)
S	Natural の STOP ステートメント。ランタイム時にこのステートメントが実行されます。(*)
E	この行に指定した値は、即座に Natural システム変数 *ERROR-NR に転送されます。
R	DDM フィールド RETURN-CODE にリターンコードが入力されます。『ステートメント』ドキュメントの PROCESS COMMAND を参照してください。
1~9	テキスト文字列。この値は複数の DDM フィールド RESULT-FIELD に入力されます。『ステートメント』ドキュメントの PROCESS COMMAND を参照してください。
*	コメント行。

\* これらのアクションは、PROCESS COMMAND ステートメントの EXEC オプションでのみ実行されます。

- 3 PF3 キーを押して、[Runtime Action Definition] ウィンドウを閉じます。




**注意:** ユーザー出口 NCP-REAM を使用すると、上記のコードの一部またはすべてを使用できます。ユーザー出口 NCP-REEM を使用すると、ランタイムアクション定義テーブルのヘッダーの下の行を修正できます。ユーザー出口 NCP-REDM を使用すると、ランタイムアクション定義のデフォルト値を定義できます。このユーザー出口を使用する場合は、「ヘッダーレコード」の *Invoke Action Editor* も参照してください。上記のユーザー出口はすべて Natural システムライブラリ SYSNCP で提供されています。

## プロセッサのカタログ化

コマンドプロセッサで使用するすべての機能にランタイムアクションを指定した後、コマンドプロセッサをカタログする必要があります。コマンドプロセッサをカタログすると、プロセッサタイプの Natural オブジェクトが生成されます。

### ▶手順 100.15. コマンドプロセッサをカタログするには

- 1 [Processor Maintenance] メニューで、ファンクションコード「C」（Catalog Processor）、カタログするコマンドプロセッサの名前、およびコマンドプロセッサをカタログする Natural ライブラリの名前を入力します。
- 2 Enter キーを押します。

 **注意:** Natural Security がインストールされている場合、コマンドプロセッサの使用を許可する必要があります。『Natural Security』ドキュメントの「Functional Security」を参照してください。

## 管理者サービス

SYSNCP には、コマンドプロセッサの管理機能が用意されています。Natural Security で定義されたシステム管理者のみが、これらのサービスにアクセスする権限を持ちます。

### ▶手順 100.16. 管理者サービスにアクセスするには

- 1 [Processor Source Maintenance] メニューで、ファンクションコード「A」（Administrator Services）を入力します。
- 2 Enter キーを押します。

[Administrator Services] 画面が表示されます。

```
09:49:11          ***** NATURAL SYSNCP UTILITY *****          2000-05-04
User SAG          - Administrator Services -
```

```
Code  Function
S     Select Processor
C     Copy Processor Source
D     Delete Processor Source
P     Print Source/Object/NCP-Buffer
```

```


U      Unload Processor to Work File 3
L      Load Processor from Work File 3
F      Freeze Processor Source
R      References from Natural Security
?      Help
.      Exit

Code .. _      Name .. SAGTEST_      Library .. SYSNCP__

Command ==>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help Cmd   Exit Last List Flip                                     Canc

```

 **注意:** Natural Security がインストールされていない場合、すべてのユーザーが管理者ステータスになることに注意してください。

以下に参考情報を示します。

- [プロセッサ選択](#)
- [プロセッサソースのコピー](#)
- [プロセッサソースの削除](#)
- [ソース/オブジェクト/NCP バッファのプリント](#)
- [プロセッサのアンロード](#)
- [プロセッサのロード](#)
- [プロセッサソースのフリーズ](#)
- [Natural Security からの参照](#)

## プロセッサ選択

「[プロセッサの選択](#)」を参照してください。

## プロセッサソースのコピー

プロセッサソースのコピーでは、プロセッサ全体をコピーするか、または選択したソース（ヘッダー、キーワード、機能、ランタイムアクション定義）のみをコピーするかを選択できます。

### ▶手順 100.17. コマンドプロセッサをコピーするには

- 1 [Administrator Services] メニューで、ファンクションコード「C」を入力します。
- 2 Enter キーを押します。

[Copy Processor Source] ウィンドウが開き、ソースおよびターゲット情報が表示されま  
す。

```

Copy Processor Source

Source          Target
Name ..... SAGTEST_
Library ..... SYSNCP__
DBID ..... 10__
FNR ..... 32__
Password ....
Cipher Key ..

Replace ..... NO_
    
```

- [Source] 側のフィールドに、コピーするプロセッサの名前、プロセッサが保存されているライブラリ、データベース ID (DBID)、およびファイル番号 (FNR) を入力します。デフォルト値は [Administrator Services] メニューで指定したプロセッサです。

[Target] 側のフィールドには、コピー先のプロセッサの名前、コピーしたプロセッサを保存するライブラリ、データベース ID (DBID)、およびファイル番号 (FNR) を入力します。

ソースファイルやターゲットファイルがパスワードとサイファキーの両方またはいずれかで保護されている場合には、[Cipher Key] フィールドに適切なパスワードとサイファキーの両方またはいずれかを入力します。

ターゲット環境のプロセッサを上書きする場合は、[Replace] フィールドに「YES」を入力します。このフィールドのデフォルトは NO です。

- Enter キーを押します。

ソースを選択するための次のようなウィンドウが表示されます。

```

Copy Processor Source

Mark  Copy          Source  Target
-----
_    Header .....   yes    no
_    Keywords .....  yes    no

_    Functions .....  yes    no
     Runtime Action Definitions .. no     no
    
```

```

Source Name SAGTEST      Library SYSNCP      DBID 10      FNR 32
Target Name TEST2       Library SYSNCP      DBID 10      FNR 32

Replace ... NO

```

- 5 該当する **[Mark]** フィールドに任意の文字を入力して、コピーするソースを選択します。
- 6 Enter キーを押します。

## プロセッサソースの削除

この機能は、プロセッサソースを削除するために使用します。

### ▶手順 100.18. コマンドプロセッサを削除するには

- 1 **[Administrator Services]** メニューで、ファンクションコード「**D**」を入力します。
  - 2 Enter キーを押します。
- [Delete Processor Source]** ウィンドウが表示されます。
- 3 削除するプロセッサの名前、プロセッサが保存されているライブラリ、データベース ID、およびファイル番号を指定します。ファイルがパスワードとサイファキーの両方またはいずれかによって保護されている場合、適切なパスワードとサイファキーの両方またはいずれかを入力する必要があります。
  - 4 Enter キーを押します。

削除するソースを選択するための次のようなウィンドウが表示されます。

```

Delete Processor Source

Mark  Delete                                     Available
-----
_     Header .....                               yes
_     Keywords .....                             yes
_     Functions .....                            yes
_     Runtime Action Definitions ..             yes

Name SAGTEST      Library SYSNCP      DBID 10      FNR 32

```

各プロセッサソース（ヘッダー、キーワード、機能、ランタイムアクション定義）の右側にあるフィールドは、ソースが存在するかどうかを示しています。コマンドプロセッサの作成は累積的な処理であるため、あるソースを削除する場合、そのソースをベースにしている

ソースもすべて削除する必要があります。例えば、上の画面で、機能のソースを削除する場合は、ランタイムアクション定義のソースも削除する必要があります。

- 5 [Available] と示されている各ソースを選択するには、該当する [Mark] フィールドに任意の文字を入力します。
- 6 Enter キーを押します。

### ソース/オブジェクト/NCP バッファのプリント

プロセッサソースの他に、プロセッサオブジェクトおよびNCPバッファもプリントできます。

#### ▶手順 100.19. コマンドプロセッサ項目をプリントするには

- 1 [Administrator Services] メニューで、ファンクションコード「P」を入力します。
- 2 ENTER キーを押します。  
[Print Source/Object/NCP-Buffer] ウィンドウが表示されます。
- 3 プリントするプロセッサの名前、プロセッサが保存されているライブラリ、データベースID、およびファイル番号を指定します。ファイルがパスワードとサイファキーの両方またはいずれかによって保護されている場合、適切なパスワードとサイファキーの両方またはいずれかを入力する必要があります。
- 4 Enter キーを押します。
- 5 プリントする項目を選択するための次のようなウィンドウが表示されます。

```

Print Source/Object/NCP-Buffer

Mark  Print                                     Available
-----
-     Header .....                               yes
-     Keywords .....                             yes

-     Functions .....                            yes
-     Runtime Action Definitions ..             yes

-     Processor Object .....                    yes
-     NCP-Buffer .....                          no

Printer ..... _____

Name SAGTEST   Library SYSNCP   DBID 10       FNR 32
    
```

各プロセッサソース（ヘッダー、キーワード、機能、ランタイムアクション定義）の右側にあるフィールドは、項目が存在するかどうかを示しています。



[Printer] フィールドに入力できる値は、論理プリンタ ID、VIDEO、または SOURCE です。『ステートメント』ドキュメントの DEFINE PRINTER も参照してください。

- 6 該当する [Mark] フィールドに任意の文字を入力してプリントする項目を選択し、[Printer] フィールドに論理プリンタ名、あるいは「VIDEO」または「SOURCE」を入力します。
- 7 Enter キーを押します。

## プロセッサのアンロード

### ▶手順 100.20. コマンドプロセッサをアンロードするには

- 1 [Administrator Services] メニューで、ファンクションコード「U」を入力します。
- 2 Enter キーを押します。 [Unload Processor to Work File 3] ウィンドウが表示されます。


```

                                Unload Processor to Work File 3
                                Source                Target
Name ..... SAGTEST_
Library ..... SYSNCP__          SYSNCP__
DBID ..... 10___
FNR ..... 32___
Password ....
Cipher Key ..
Report ..... NO_

```

- 3 [Source] 側のフィールドに、アンロードするプロセッサの名前、プロセッサが存在するライブラリ、データベース ID、およびファイル番号を入力します。デフォルト値は [Administrator Services] メニューで指定したプロセッサです。ファイルがパスワードとサイファキーの両方またはいずれかで保護されている場合は、適切なパスワードとサイファキーの両方またはいずれかを入力します。
- 4 レポートを出力する場合は、[Report] フィールドに「YES」を入力します。デフォルトは NO です。ファイル拡張子を使用する必要はありません。拡張子を使用する場合は、ファイル拡張子 ".sag" を使用する必要があります。
- 5 Enter キーを押します。

プロセッサをアンロードすると、すべてのプロセッサソース（ヘッダー、キーワード、機能、ランタイムアクション定義）がワークファイル 3 に書き込まれます。

 **注意:** あるハードウェアプラットフォームから別のハードウェアプラットフォームにコマンドプロセッサを転送するには、[オブジェクトハンドラ](#)を使用します。

## プロセッサのロード

### ▶手順 100.21. コマンドプロセッサをロードするには

- 1 [Administrator Services] メニューで、ファンクションコード「L」を入力します。
- 2 Enter キーを押します。

プロセッサをワークファイル3から Natural ライブラリにロードするための [Load Processor from Work File 3] ウィンドウが表示されます。

```
Load Processor from Work File 3

Replace existing processors .. N
Produce load report ..... NO_
```

- 3 ロードするプロセッサで同じ名前を持つ既存のプロセッサを置き換えるかどうかを、[Replace existing processors] フィールドに「Y」または「N」（デフォルトはN）を入力して指定します。
- 4 レポートを出力する場合は、[Produce load report] フィールドに「YES」（デフォルトはNO）を入力します。
- 5 Enter キーを押します。



**注意:** プロセッサ名とプロセッサのロード先のライブラリはワークファイルから取得されます。

## プロセッサソースのフリーズ

プロセッサを現在の状態でフリーズして、ユーザーがこれ以上修正できないようにすることができます。

### ▶手順 100.22. コマンドプロセッサをフリーズするには

- 1 [Administrator Services] メニューで、ファンクションコード「F」を入力します。
- 2 Enter キーを押します。 [Freeze Processor Source] ウィンドウが表示されます。
- 3 フリーズするプロセッサの名前、プロセッサが保存されているライブラリ、データベースID、およびファイル番号を指定します。 ファイルがパスワードとサイファキーの両方またはいずれかによって保護されている場合、適切なパスワードとサイファキーの両方またはいずれかを入力する必要があります。
- 4 Enter キーを押します。

- 次に表示されるウィンドウで、プロセッサソースの修正を許可するかまたは禁止するかを **Y** または **N** で指定します。デフォルトは **Y** です。
- Enter** キーを押します。

### Natural Security からの参照

この機能は、Natural Security が稼働している環境でのみ有効です。この機能を使用して、Natural Security から機能のセキュリティの参照を削除します。

Natural Security でプロセッサに対する機能のセキュリティを定義すると、自動的に参照が作成されます。これらの参照はプロセッサソースとともに、FSEC ではなく、FNAT/FUSER システムファイルに保存されます。

#### ▶手順 100.23. Natural Security 機能から参照を呼び出すには

- [**Administrator Services**] メニューで、ファンクションコード「**R**」を入力します。
- Enter** キーを押します。  
**[Delete References]** ウィンドウが表示されます。
- プロセッサの名前、プロセッサが保存されているライブラリ、データベースID、およびファイル番号を指定します。ファイルがパスワードとサイファキーの両方またはいずれかによって保護されている場合、適切なパスワードとサイファキーの両方またはいずれかを入力する必要があります。
- Enter** キーを押します。
- 次に表示されるウィンドウで、主参照、機能参照、および補助参照を削除できます。

コマンドプロセッサの機能のセキュリティの詳細については、『*Natural Security*』ドキュメントの「*Functional Security*」を参照してください。

## セッションプロファイル

セッションプロファイルとは、ユーザー定義が可能なデフォルトの集合のことです。セッションプロファイルによって、SYSNCP 画面の表示方法または入力に対する SYSNCP の応答方法が決定されます。セッションプロファイルでは、例えば、セッションに対するデフォルトのコマンドプロセッサ、または画面属性に割り当てる色を定義できます。SYSNCP には、すべての新規ユーザーに発行される STANDARD という標準セッションプロファイルがあります。必要に応じて、複数の異なるセッションプロファイルを作成し、有効にすることができます。

SYSNCP の管理者は、SYSNCP 内のどのセッションプロファイルでもアクセスおよび修正できます。その他のユーザーもすべてのセッションプロファイルにアクセスできますが、修正できるのは自分のユーザー ID で作成されたセッションプロファイル、または自分のユーザー ID と同じ名前をもつセッションプロファイルのみです。

## ▶手順 100.24. セッションプロファイルを定義または修正するには

- **[Processor Source Maintenance]** メニューのコマンド行から PROFILE コマンドを発行します。

3つのセッションプロファイルメンテナンス画面のうちの最初の画面が表示されます。

以下に参考情報を示します。

- セッションプロファイル名
- セッションパラメータ - プロファイル 1
- カラー属性 - プロファイル 2
- その他の属性 - プロファイル 3

### セッションプロファイル名

標準プロファイル STANDARD またはシステム変数 \*USER の値が、デフォルトのプロファイル名として使用されます。

新規セッションプロファイルを定義するとき、パラメータおよび属性はデフォルト値になっています。必要に応じてこれらのデフォルト値を修正し、新しい名前を入力して PF5 キーを押すことで、これらの値を保存できます。

各プロファイル画面の [Session Profile Name] フィールドは、入出力フィールドになっています。そのため、[Profile Name] フィールドに別のプロファイル名を入力して PF5 キーまたは PF4 キーをそれぞれ押すことで、これらのどの画面からでもそのプロファイルを定義、読み込み、または保存することができます。

### セッションパラメータ - プロファイル 1

最初のプロファイルメンテナンス画面では、以下のフィールドを修正できます。

フィールド	説明
Apply Terminal Control 1	起動時に SYSNCP によって発行される SET CONTROL ステートメントのパラメータを入力します。  例えば、このフィールドのいずれかに「Z」を入力すると、SYSNCP により SET CONTROL 'Z' ステートメントが発行されます。
Apply Terminal Control 2	
Default Processor Name	このセッションで使用するデフォルトのコマンドプロセッサ名。
Default Processor Library	コマンドプロセッサを保存するために使用する Natural ライブラリ。
Cancel Reaction	要求された修正を完了しないで CANCEL コマンドを発行した場合に必ず警告を発行するかどうかを指定します。  W 警告を発行します。

フィールド	説明
	B 警告を発行しないでバックアウトし、キャンセルします。
Clear Key Allowed	クリアキーの使用を許可するかどうかを指定します。  N クリアキーの使用を禁止します。 Y クリアキーは有効で、CANCEL と同じ効果を持ちます。
Default Cursor Position	カーソル位置を指定します。  1 カーソルを画面の最初のフィールドに位置付けます。 C カーソルをコマンド行に位置付けます。
Exec/Display Last Command	LAST コマンドで実行される処理を指定します。  E コマンド行で発行した最後のコマンドを実行します。 D コマンド行で発行した最後のコマンドを表示します。

## カラー属性・プロファイル 2

2番目のプロファイルメンテナンス画面では、さまざまな画面属性に色を割り当てることや、既存の色の割り当てを上書きすることができます。

以下のカラーコードを指定して、色を割り当てます。

コード	色
BL	青
GR	緑
NE	デフォルト色
PI	ピンク
RE	赤
TU	空色
YE	黄色

画面属性に対する色の割り当てについては、『端末コマンド』ドキュメントの端末コマンド %= も参照してください。

### その他の属性・プロファイル 3

3 番目のプロファイルメンテナンス画面で指定できる属性は次のとおりです。

フィールド	説明
Message Line Position	メッセージを表示する行。推奨値は 21 です。詳細については、『端末コマンド』ドキュメントの端末コマンド %M も参照してください。
Text for PF5 Key	PF5 ファンクションキーは、グローバルな用途（セッション単位）のために予約されています。PF キー行に表示される PF5 キーのテキストをこのフィールドに入力できます。
Command for PF5 Key	PF5 ファンクションキーは、グローバルな用途（セッション単位）のために予約されています。PF5 キーを押したときに実行されるコマンドを、このフィールドに入力できます。

また、画面には、このプロファイルを最後に修正した日付およびユーザーも表示されます。

# 101 SYSPARM ユーティリティ

---

▪ SYSPARM の呼び出し .....	690
▪ List Profiles .....	691
▪ Display Profile .....	692
▪ Add New Profile .....	692
▪ Modify Profile .....	693
▪ プロファイルの編集 .....	693
▪ Copy Profile .....	695
▪ Delete Profile .....	696
▪ ダイレクトコマンドとバッチ処理 .....	696

SYSPARM ユーティリティは、プロファイル名を付けて保存する Natural プロファイルパラメータのセットを作成および管理するために使用します。

ダイナミックプロファイルパラメータを使用して Natural を呼び出す場合は、Natural を呼び出すたびに個々のパラメータを指定できます。ただし、より簡単な方法として、SYSPARM にパラメータのセットを指定し、そのセットにプロファイル名を付けて保存し、1つのダイナミックパラメータ `PROFILE=profile-name` のみを使用して Natural を呼び出すことができます。このプロファイルに定義したパラメータは、次にダイナミックパラメータとして Natural に渡されるため、PROFILE の構文に準拠している必要があります（『Natural パラメータリファレンス』ドキュメントを参照）。

プロファイルに指定できる個々のプロファイルパラメータの詳細については、『Natural パラメータリファレンス』ドキュメントの「プロファイルパラメータ」を参照してください。

このプロファイルは、Natural システムファイルの FNAT または FUSER に格納されます。現在の FNAT システムファイルのデータベース ID (DBID) とファイル番号 (FNR) は、[SYSPARM Menu] に表示されます。

プロファイルの使用を特定のユーザーに限定するには、プロファイルパラメータ USER を使用します（『Natural パラメータリファレンス』ドキュメントを参照）。

## SYSPARM の呼び出し

### ▶手順 101.1. SYSPARM ユーティリティを呼び出すには

- Natural システムコマンド SYSPARM を入力します。

[Natural SYSPARM Utility Menu] が表示されます。

[SYSPARM Menu] には、次のフィールドと機能があります。

フィールド／機能	説明
List Profiles	すべてのプロファイルのリストを表示します。リストから、表示、変更、または削除するプロファイルを選択できます。 「 <a href="#">List Profiles</a> 」を参照してください。
Display Profile	特定のプロファイルを表示します。 「 <a href="#">Display Profile</a> 」を参照してください。
Add New Profile	新規プロファイルを作成します。 「 <a href="#">Add New Profile</a> 」および「 <a href="#">Editing Profiles</a> 」を参照してください。
Modify Profile	既存のプロファイルを変更します。 「 <a href="#">Modify Profile</a> 」および「 <a href="#">Editing Profiles</a> 」を参照してください。
Copy Profile	既存のプロファイルをコピーして新規プロファイルを作成します。 「 <a href="#">Copy Profile</a> 」を参照してください。



フィールド／機能	説明
Delete Profile	既存のプロファイルを削除します。 「 <i>Delete Profile</i> 」を参照してください。
Profile	指定したプロファイルを選択します。有効なプロファイル名を入力するか、アスタリスク (*) 表記を使用してプロファイルの範囲を選択します。使用可能なすべてのプロファイルを選択するには、アスタリスク (*) または空白フィールドを使用します。
Copy to	指定したプロファイルをコピーします。
DBID/FNR	プロファイルを格納する Natural システムファイルのデータベース ID (DBID) とファイル番号 (FNR) を選択します。デフォルトは現在の FNAT です。
Password/Cipher	Adabas ファイルのパスワード (8 文字) とサイファコード (8 桁) を指定します。

このドキュメントでは、これ以降上記の SYSPARM 機能について説明します。

## List Profiles

この機能では、既存のすべてのプロファイルのリストを表示します。

### ▶手順 101.2. List Profiles 機能呼び出すには

- [SYSPARM Menu] でファンクションコード「L」を入力し、[Profile] フィールドにプロファイルの名前を入力するか、名前の範囲を指定します。
  - すべてのプロファイルを表示するには、アスタリスク (\*) を入力するか、またはフィールドを空白のままにします。
  - 指定した値で始まる名前を持つすべてのプロファイルを表示するには、例えば ABC\* のように、アスタリスク (\*) 表記を使用します。
  - 指定した値以降の名前を持つすべてのプロファイルを表示するには、例えば ABC> のように、大なり記号 (>) を使用します。
  - 指定した値以前の名前を持つすべてのプロファイルを表示するには、例えば ABC< のように、小なり記号 (<) を使用します。

[List Profiles] 画面が表示されます。

[List Profiles] 画面で PF7 キーまたは PF8 キーを押して、リストを上下にスクロールします。

次に、表示、変更、または削除するプロファイルを選択する場合に使用できる PF キーと行コマンドを示します。プロファイルをカーソルでマークし、適切な PF キーを押すか、プロファイルの横の [Sel.] 列に行コマンドを入力し、Enter キーを押します。

キー	行 コマンド	機能
PF4	D	指定したプロファイルの Display Profile 機能呼び出します。
PF5	M	指定したプロファイルの Modify Profile 機能呼び出します。
PF6	X	指定したプロファイルの Delete Profile 機能呼び出します。

## Display Profile

この機能は、既存のプロファイルを表示するために使用します。

### ▶手順 101.3. Display Profile 機能呼び出すには

- [SYSPARM Menu] で、ファンクションコード **D** とプロファイルの名前を入力します。

または:

[List Profiles] 画面で、該当するセクションの説明に従ってプロファイルを選択します。

[Display Profile] 画面が開き、指定したプロファイルが表示されます。

プロファイルを変更するには、Modify Profile 機能呼び出す PF5 キーを押します（「[Modify Profile](#)」を参照）。

## Add New Profile

この機能は、新規プロファイルを作成するために使用します。

### ▶手順 101.4. Add New Profile 機能呼び出すには

- [SYSPARM Menu] で、ファンクションコード **A** とプロファイルの名前を入力します。

表示された画面で、「[Editing Profiles](#)」の説明に従って、プロファイルに含めるプロファイルパラメータを入力します。

## Modify Profile

---

この機能は、既存のプロファイルを変更するために使用します。

### ▶手順 101.5. Modify Profile 機能呼び出すには

- [SYSPARM Menu] で、ファンクションコード **M** とプロファイルの名前を入力します。

または:

[**Display Profile**] 画面で、PF5 キーを押します。

または:

[**List Profiles**] 画面で、該当するセクションの説明に従ってプロファイルを選択します。

新規プロファイルを追加する画面に類似した画面が表示され、選択したプロファイルを変更できます（次の「プロファイルの編集」を参照）。

## プロファイルの編集

---

プロファイルの追加または変更用の画面には、それぞれ最大長が72文字の18行の入力行があります。入力する個々のパラメータは、1つ以上の空白またはコンマで区切る必要があります。パラメータの指定は、任意の数の行にまたがって記述できます。

編集用の追加のスペースが必要な場合、または異なるプロファイルやグループプロファイルをカテゴリごとに組み合わせる場合は、対象のプロファイルの最後のエントリとして PROFILE パラメータを入力して、複数のプロファイルを連結します。これにより、最初のプロファイルでは、2番目のプロファイルに指定したパラメータ文字列が最初のプロファイルの文字列の一部として評価されます。

例:

プロファイル P1 の末尾に「PROFILE=P2」を入力し、プロファイル P2 の末尾に「PROFILE=P3」を入力した場合、まず P1 が呼び出され、その後 P2、P3 の順に呼び出されます。

コメントのテキストを各エディタ行に入力したり、コメントを任意の数の行にまたがって記述できます。コメントの先頭には「/＊」を付け、末尾には「＊/」を付ける必要があります。

以下に参考情報を示します。

- 機能

- Help with Parameters

## 機能

プロファイルの編集用に表示される画面では、次の機能を使用できます。

キー	機能
PF4	プロファイル内のパラメータ指定の構文が正しいかどうかをチェックします。
PF5	プロファイルを格納します。
PF9	カーソルのある行の下に空行を 1 行挿入します。
PF10	カーソルのある行を削除します。
PF11	カーソルのある行をコピーします。

## Help with Parameters

### ▶手順 101.6. 特定のプロファイルパラメータに関するヘルプ情報を呼び出すには

1 [Help with Parameters] フィールドで、次の操作を行います。

- パラメータの名前を入力します。
- または、アスタリスク (\*) 表記を含む文字列を入力します。  
その文字列で始まる名前を持つすべてのパラメータが表示されます。  
任意の文字でマークを付けて、パラメータを選択します。
- またはアスタリスク (\*) を入力します。  
すべてのパラメータのリストが表示されます。  
任意の文字でマークを付けて、パラメータを選択します。

2 Enter キーを押します。

または:

カーソルを特定のエディタ行に置き、Enter キーを押します (次の「[特殊な位置指定](#)」を参照)。

次の 2 つのセクションがある画面が表示されます。

- 上部のセクションには、指定したパラメータに関するヘルプテキストが表示されます。テキストが現在の画面を超える場合は、[More Help] フィールドにプラス (+) 記号を入力して、次の画面を表示します。前の画面に戻るにはマイナス (-) 記号を入力します。または、Help with Parameters 機能を終了するには、ピリオド (.) を入力します。

- 下部のセクションは、 [Enter your parameter specification] です。
- 3 [Enter your parameter specification] では、パラメータ指定を入力します。
  - 4 PF4 キーを押して、パラメータ指定をチェックします。
  - 5 PF3 キーを押して、**Help with Parameters** 機能を終了し、プロファイルの下部、または以前にマークを付けた位置にパラメータ指定を追加します（次の「特殊な位置指定」を参照）。

#### 特殊な位置指定

プロファイルの特定の位置に新規パラメータ指定を追加するには、次の手順に従います。

- [Help with Parameters] フィールドに、値（またはアスタリスク）を入力し、Enter キーを押しません。
- 新規パラメータ指定を配置する行の上のエディタ行にカーソルを置きます。
- Enter キーを押して Help with Parameters 機能を呼び出し、パラメータ指定を入力します。
- PF3 を押します。

以前にマークを付けたエディタ行の下に、新規パラメータ指定が追加されます。

## Copy Profile

この機能は、既存のプロファイルをコピーして新規プロファイルを作成するために使用します。

この機能では、同じ Natural システムファイル内でのみプロファイルのコピーできます。1つの Natural システムファイルから他の Natural システムファイルにプロファイルのコピーするには、該当するドキュメントの説明に従って **SYSMAIN** ユーティリティを使用します。

#### ▶手順 101.7. Copy Profile 機能を呼び出すには

- [SYSPARM Menu] で次の操作を行います。
  - ファンクションコード「C」を入力します。
  - [Profile] フィールドに、既存のプロファイルの名前を入力します。
  - [Copy To] フィールドに、新規プロファイルの名前を入力します。

## Delete Profile

---

この機能は、既存のプロファイルを削除するために使用します。

### ▶手順 101.8. Delete Profile 機能呼び出すには

1 [SYSPARM Menu] で、ファンクションコード「X」を入力します。

または:

[List Profiles] 画面で、該当するセクションの説明に従ってプロファイルを選択します。

[Delete a Profile] ウィンドウが表示されます。

2 [Delete a Profile] ウィンドウで次の操作を行います。

- プロファイルの名前を入力して削除を確認し、Enter キーを押します。
- または、何も入力しないで Enter キーを押して操作をキャンセルします。

## ダイレクトコマンドとバッチ処理

---

SYSPARM ユーティリティ機能は、ダイレクトコマンドモードまたはバッチモードでも実行できます。

SYSPARM ユーティリティメニューで提供される機能に加えて、コメントのテキスト、入力デリミタ、および INPUT 割り当て文字を指定するためのオプションがバッチモードで提供されます。

このセクションでは、次のトピックについて説明します。

- [バッチ処理](#)
- [SYSPARM の構文](#)
- [where-clause](#)
- [with-clause](#)

## ■ バッチでの SYSPARM の例

### バッチ処理

次に、SYSPARM をバッチモードで実行する場合の一般的な使用方法について説明します。

- SYSPARM を終了するには、個別の行にピリオド (.) または「FIN」を入力します。「FIN」では Natural セッションが終了します。
- 1 つ以上の SYSPARM 機能を実行するには、各機能を個別の行に指定します。
- 機能を 2 行にまたがって記述するには、セッションパラメータ CF (デフォルトは %) で定義した文字を最初の行の任意の場所に入力します。この文字は、次の行へ続くことを示します。

「バッチでの SYSPARM の例」も参照してください。

関連トピック：

- 「バッチモードでの Natural」 (Natural の『オペレーション』ドキュメント)
- 「非スクリーンモードでの INPUT ステートメントの使用」 (Natural の『ステートメント』ドキュメント)
- 「バッチモードでの INPUT ステートメントの使用」 (Natural の『ステートメント』ドキュメント)

### SYSPARM の構文

SYSPARM ダイレクトコマンドの処理時またはバッチモードで適用される構文を次に示します。ダイレクトコマンドの使用時に適用される制限の詳細については、次の「キーワードの説明」を参照してください。

図で使用している記号については、『Natural システムコマンドリファレンス』ドキュメントの「システムコマンド構文」セクションを参照してください。

[SYSPARM]	{	{	ADD	} profile-name	[where-clause] [with-clause]	}
			MODIFY			
	DISPLAY	} profile-name	[where-clause]	}		
	LIST					
	DELETE   X	} profile-name1	TO	} profile-name2 [where-clause]		
COPY						
	{	ASSIGN	} [character]	}		
	{	DELIMITER				
	{	COMMENT	{ ON	} [comment]	}	
	{		OFF			
					...	

キーワードの説明

次のセクションでは、上記のSYSPARM構文図に示したキーワードと節について簡単に説明し、必要に応じて、特別な使用方法を示します。コマンドを表すキーワードの詳細については、このセクションで説明した対応する SYSPARM 機能を参照してください。

また、「[バッチでの SYSPARM の例](#)」では、キーワードと節の使用方法を示します。

キーワード	説明
SYSPARM	SYSPARM ユーティリティを呼び出します。 このキーワードは、NEXT プロンプトまたは MORE プロンプトから SYSPARM コマンドを実行する場合にのみ必要となります。
ADD	<b>Add Profile</b> 機能を実行します。  コマンド ADD とプロファイルの内容に含まれる後続の入力行を明確に区切るには、キーワード <b>WITH</b> を使用します。  例：  ADD <i>profile-name</i> WITH AUTO=ON ADD コマンドを終了するには、個別の行にピリオド (.) を入力します。  「 <a href="#">プロファイルの編集</a> 」も参照してください。
MODIFY	<b>Modify Profile</b> 機能を実行します。  MODIFY コマンドを終了するには、次の例に示すように、個別の行にピリオド (.) を入力します。  プロファイルにコメントのテキストを挿入するには、文字 /* と */ の間にコメントを配置します。
DISPLAY	<b>Display Profile</b> 機能を実行します。
LIST	<b>List Profiles</b> 機能を実行します。
DELETE または X	<b>Delete Profile</b> 機能を実行します。
COPY	<b>Copy Profile</b> 機能を実行します。
ASSIGN	INPUT 割り当て文字を指定または表示します。現在のセッションに定義された INPUT 割り当て文字を含むデータの処理を開始する前に、INPUT 割り当て文字を変更する必要があります。さもないと、Natural エラー NAT1105（「入力のためのキーワード ... が定義されていません」）が表示されます。  値が指定されていない場合、出力には現在のセッションに対して定義された文字が含まれます。
DELIMITER	INPUT 区切り文字を指定または表示します。現在のセッションに定義された INPUT 区切り文字を含むデータの処理を開始する前に、INPUT 区切り文字を変更する必要があります。さもないと、Natural エラー NAT1105（「入力のためのキーワード ... が定義されていません」）が表示されます。



キーワード	説明
	値が指定されていない場合、出力には現在のセッションに対して定義された文字が含まれます。
COMMENT	<p>REPLACE コマンドには適用されません。</p> <p>コメントオプションをアクティブ (ON) にし、変更された各パラメータの後にコメントを書き込む、またはコメントオプションを非アクティブ (OFF) にします。</p> <p>アクティブにして個々の <i>comment</i> を指定しない場合 (下記参照)、各パラメータに挿入されたコメントに次のデータが含まれます。</p> <ul style="list-style-type: none"> <li>■ Natural システム変数 *USER によって生成された、プロフィールを最後に変更したユーザーの ID および</li> <li>■ Natural システム変数 *DATV および *TIMX によって生成された、プロフィールが最後に変更された日時 (『Natural システム変数』ドキュメントの該当するセクションも参照)。</li> </ul> <p>コメントを変更するには、<b>REPLACE</b> コマンドを使用します (「キーワードの説明-WITH 節」を参照)。</p>
<i>character</i>	任意の特殊文字。Natural の『パラメータリファレンス』ドキュメントに記載されている Natural セッションパラメータ ID (INPUT 区切り文字) および IA (INPUT 割り当て文字) を参照してください。
<i>comment</i>	文字 /* と */ の間に配置するコメントのテキスト。
<i>profile-name</i>	<p>プロフィールの名前または名前の範囲。</p> <p>プロフィールの範囲は、LIST コマンドでのみ指定できます。バッチモードでは、LIST コマンドまたは DISPLAY コマンドでのみ指定できます。</p>
<i>profile-name1</i> <i>profile-name2</i>	<p>COPY コマンドにのみ適用されます。</p> <p>新規プロフィール (<i>profile-name1</i>) の作成元となるソースプロフィール (<i>profile-name1</i>)、およびデータのコピー先となるターゲットプロフィール (<i>profile-name2</i>)</p>
WHERE	<i>where-clause</i> の開始を示すオプションキーワード。次の「 <b>WHERE 節</b> 」を参照してください。
WITH	<p>ADD、DISPLAY、MODIFY、および LIST コマンドにのみ適用されます。</p> <p>ADD とともに使用する場合、<i>with-clause</i> は、キーワード WITH とその後に続くプロフィールの内容で構成されます。</p> <p><i>with-clause</i> の開始を示すオプションキーワード。次の「<b>WITH 節</b>」を参照してください。</p>
...	<p>バッチモードでのみ適用されます。</p> <p>各機能を個別の行に配置して、1 つ以上の SYSPARM 機能を指定できることを示します。</p>

**where-clause**

```
[WHERE] [DBIDdbid] [FNRfnr] [PASSWORDpassword] [CIPHERcipher]
```

ここからは、*where-clause* で示されるキーワードと適用される値について説明します。

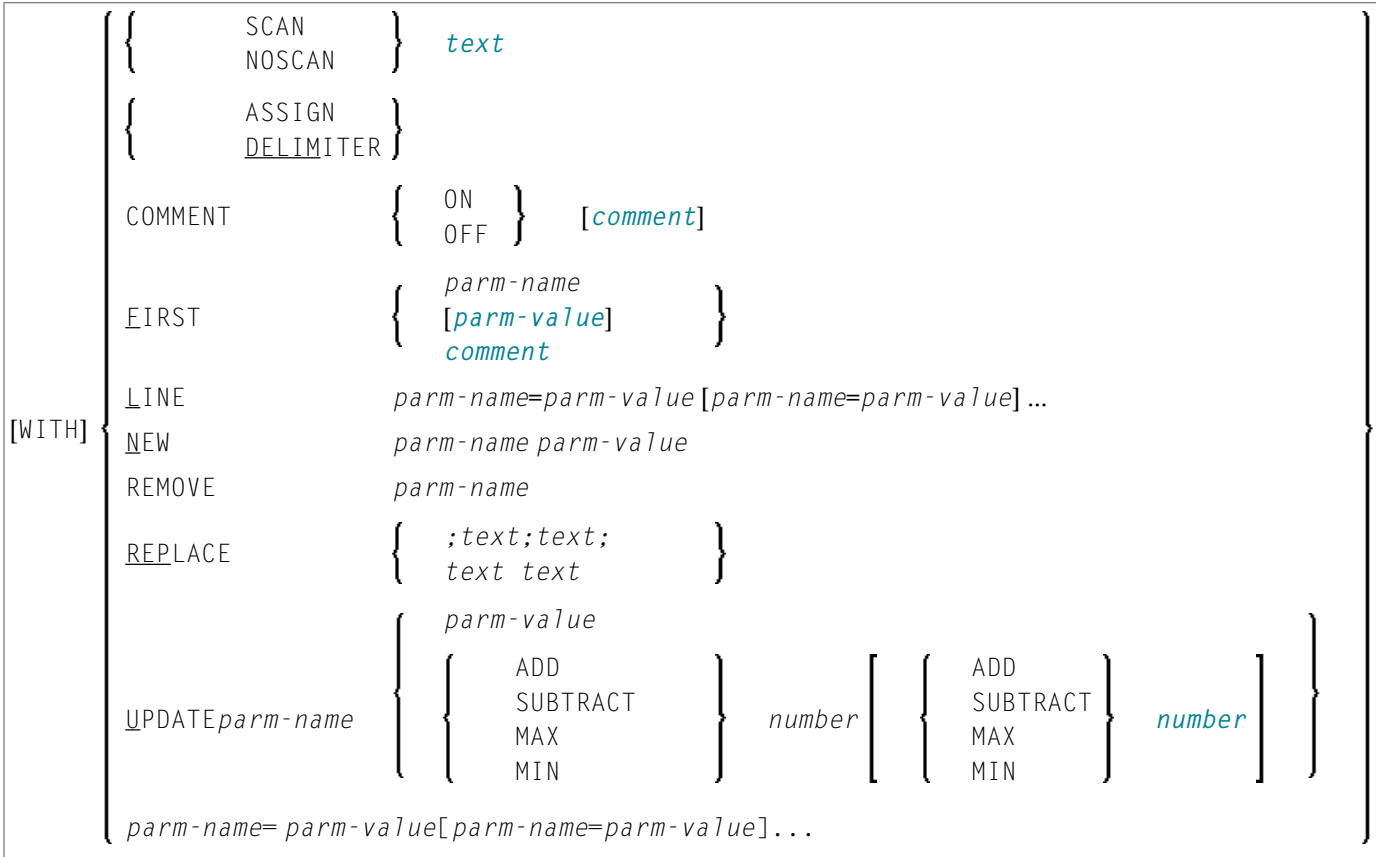
キーワードの説明 - *where-clause*


*where-clause* は任意指定です。構文は次のとおりです。

キーワード	有効な値
DBID	プロファイルが保存されている Adabas ファイルのデータベース ID (DBID)。
FNR	プロファイルが保存されている Adabas ファイルのファイル番号 (FNR)。
PASSWORD	プロファイルが保存されている Adabas ファイルの 8 文字の Adabas パスワード。
CIPHER	プロファイルが保存されている Adabas ファイルの 8 桁のサイファコード。

**with-clause**

*with-clause* は任意指定で、ADD、DISPLAY、MODIFY、および LIST コマンドにのみ適用されます。構文は次のとおりです。



 **注意:** 上記の構文図では、次の構文は ADD コマンドにのみ適用されます。[WITH]  
 parm-name=parm-value [parm-name=parm-value]...

キーワードの説明 - *with-clause*

ここからは、*with-clause* で示されるキーワードと適用される値について説明します。

キーワード	有効な値
SCAN	コマンド LIST および DISPLAY にのみ適用されます。  text 文字列の、現在の Natural システムファイルで使用可能な 1 つ以上のプロファイル (パラメータやパラメータ値など) をスキャンし、指定した text 文字列を含むプロファイルのリストを提供します。
NOSCAN	コマンド LIST および DISPLAY にのみ適用されます。  現在の Natural システムファイルで使用可能な 1 つ以上のプロファイルのスキャンから text 文字列を除外し、指定したテキスト文字列を含まないプロファイルのリストを提供します。
ASSIGN	「キーワードの説明」の「 <b>ASSIGN</b> 」を参照してください。
DELIMITER	「キーワードの説明」の「 <b>DELIMITER</b> 」を参照してください。
COMMENT	「キーワードの説明」の「 <b>COMMENT</b> 」を参照してください。

キーワード	有効な値
<u>F</u> IRST	<p>パラメータとその値またはコメントを、プロファイルの最初の位置に移動します。この操作は、例えば、プロファイルパラメータ PARM を指定する場合に必要なことがあります。</p> <p>プロファイルの最初の位置に含まれるコメントは、その位置にとどまります。パラメータとその値は、コメントが最初の位置を占めていない場合にのみ、最初の位置に移動されます。コメントが最初の位置を占めている場合、パラメータはコメントの後に追加されます。FIRST で挿入されるコメントは、常に、当初最初の位置を占めていたすべてのコメントの前の位置に配置されます。</p>
<u>L</u> INE	LINE の後に指定されたパラメータをプロファイルの最後の新しい行に配置します。
<u>N</u> EW	パラメータをプロファイルの最後に付加しますが、必ずしも新しい行に付加するわけではありません。パラメータを新しい行に配置するには、LINE パラメータを使用します。
<u>R</u> EMOVE	パラメータおよび割り当てられた値 (カッコを含む) をプロファイルから削除します。削除するパラメータは、プロファイルパラメータ RPC のサブパラメータであるプロファイルパラメータ DFS など、サブパラメータも指定できます。
<u>R</u> EPLACE	プロファイルに含まれる古い <i>text</i> 文字列を新しい <i>text</i> 文字列で置き換えます。「キーワードの説明」のオプション <b>COMMENT</b> も参照してください。
<u>U</u> PDATE	パラメータに割り当てられた値を更新します。 指定したパラメータがまだ存在しない場合は、プロファイルに付加されます。
<i>parm-name</i>	パラメータの完全な名前。
<i>comment</i>	「キーワードの説明」の「 <i>comment</i> 」を参照してください。
<i>parm-value</i>	パラメータに割り当てられた値。
<i>text</i>	<p>プロファイルに含まれる任意のテキスト文字列。</p> <p>SCAN および NOSCAN コマンドには次の要件が適用されます。テキスト文字列に空白文字を含めることはできません。</p> <p>REPLACE コマンドには次の要件が適用されます。</p> <p>古い <i>text</i> と新しい <i>text</i> のいずれにも空白文字が含まれていない場合、新旧の <i>text</i> の間に空白文字を配置します。古い <i>text</i> または新しい <i>text</i> に空白文字が含まれている場合、指定した INPUT 区切り文字を古い <i>text</i> と新しい <i>text</i> の間、および新旧の <i>text</i> 文字列全体を囲んで配置します。</p> <p>例：</p> <pre>;This comment is old;This comment is new;</pre> <p>指定したデリミタ文字の代わりに、次のいずれかの文字を選択できます。</p> <ul style="list-style-type: none"> <li>' 一重引用符</li> <li>, コンマ</li> <li>. ピリオド</li> <li>; セミコロン</li> <li>/ スラッシュ</li> </ul>

キーワード	有効な値
	\ バックスラッシュ   縦棒
ADD	指定された数を加算して、パラメータ値のサイズを増やします。MAX または MIN を指定して、加算を最大値または最小値に制限できます（下記参照）。
SUBTRACT	指定された数を減算して、パラメータ値のサイズを減らします。MAX または MIN を指定して、減算を最大値または最小値に制限できます（下記参照）。
MAX	パラメータ値の最大値を指定します。
MIN	パラメータ値の最小値を指定します。
<i>number</i>	数値。

## バッチでの SYSPARM の例

次のサンプルプロファイルと入力例では、SYSPARM コマンドの使用およびバッチで実行されたプロファイル変更の結果を示します。

- サンプルプロファイル TESTPROF - ジョブサブミット前
- 入力例
- サンプルプロファイル TESTPROF - ジョブ実行後
- サンプルプロファイル TESTPRO1 - ジョブ実行後

### サンプルプロファイル TESTPROF - ジョブサブミット前

```
/* This is a test profile. */
AUTO=ON FNAT = (102,110,PASSWORD) FUSER=(1099,1100,PASSWORD,12345678)
RPC=(RPCSIZE=80,SRVNAME=MYSERV,SERVER=ON,DFS=(SRV2,NODE1,,ACI))
PRINT=((2,12,18),AM=STD,DEST='PRINT**',OPEN=INITOBJ,CLOSE=CMD)
PRINT=((1,3,6-11,15),AM=NAF)
ESIZE=90
```

### 入力例

```
/*JCL
.
.
.
*/
SYSPARM
COPY TEST1 TO TESTPROF WHERE DBID 10 FNR 32
DISP TESTPROF
COMMENT ON /* TESTUSER 29.Jul.2003 */
MODIFY TESTPROF with REM DFS
    REPLACE ;test profile;test profile for SYSPARM in batch;
    NEW BPSIZE 4096
```

```
UPDATE ESIZE ADD 20 MAX 100
FIRST PARM INHOUSE
DELIMITER $
UPDATE FUSER (,6)
ASSIGN :
LINE IM=D, INTENS=1, AUTO=T, MT=0, MADIO=0
.
DISP TESTPROF
ADD TESTPRO1 WITH /* 106,210 */
    FNAT=(106,210,PASSWORD),FUSER=(,211)
.
DISPLAY TESTPRO1
DELETE TESTPRO2 WHERE DBID 10 FNR 32
X TESTPRO3 WHERE DBID 10 FNR 32
.
FIN
```

### サンプルプロファイル TESTPROF - ジョブ実行後

```
/* This is a test profile for SYSPARM in batch. */
PARM=INHOUSE /* TESTUSER 29.Jul.2003 */
AUTO=ON FNAT = (102,110,PASSWORD) FUSER=(,6) /* TESTUSER 29.Jul.2003 */
RPC=(RPCSIZE=80,SRVNAME=MYSERV,SERVER=ON, )
PRINT=((2,12,18),AM=STD,DEST='PRINT**',OPEN=INITOBJ,CLOSE=CMD)
PRINT=((1,3,6-11,15),AM=NAF)
ESIZE=100 /* TESTUSER 29.Jul.2003 */ BPSIZE=4096 /* TESTUSER 29.Jul.2003
*/
IM=D, INTENS=1, AUTO=T, MT=0, MADIO=0 /* TESTUSER 29.Jul.2003 */
```

### サンプルプロファイル TESTPRO1 - ジョブ実行後

```
/* 106,210 */
FNAT=(106,210,PASSWORD),FUSER=(,211)
```

# 102      SYSRDC ユーティリティ

---

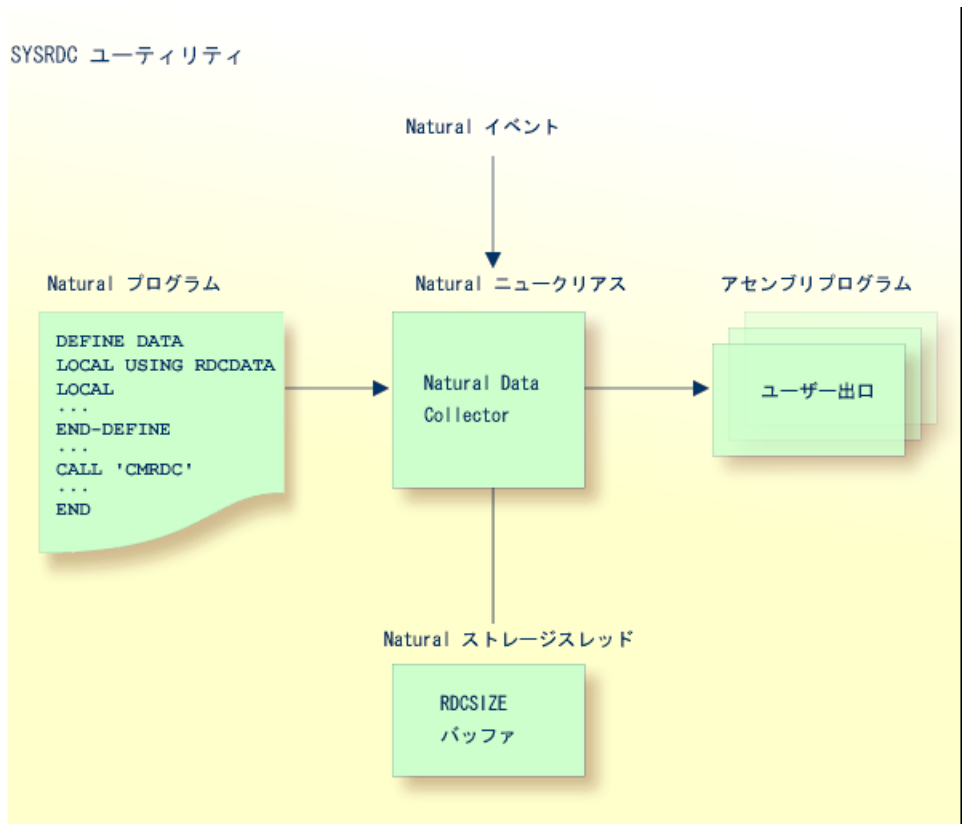
■ SYSRDC の機能コンポーネント .....	706
■ データ収集イベント .....	707
■ 収集されるデータ .....	708
■ Natural Data Collector のアクティブ化 .....	710
■ トレースの記録 .....	711
■ 外部モニタリング／計算用ユーザー出口 .....	712
■ CMRDC インターフェイスの呼び出し .....	713

ユーティリティ SYSRDC は、Natural アプリケーション内の内部プロセスフローに関するデータのモニタリングおよび計算を記録するために使用します。このデータは、外部プログラムまたは Natural プログラムの Natural セッション活動を評価する場合に使用できます。データは、Natural 内で実行されたイベントで収集されます。

## SYSRDC の機能コンポーネント

SYSRDC ユーティリティには、データを収集してその後の処理用に提供するためのコンポーネントが用意されています。次のセクションでは、これらのコンポーネントをリストして説明します。

- **Natural** ニュークリアスの一部としての **Natural Data Collector** :  
RDCSIZE バッファでデータを収集し、データ記録を制御します。
- ユーザー出口 :  
データを外部モニタリングプログラムおよび計算プログラムに渡します。
- **CMRDC** インターフェイス :  
Natural プログラム内の現在の Natural セッション用のデータを評価します。





## データ収集イベント

Natural 内のアクティビティを指定するイベント。次の表に、Natural 内でデータを収集するイベントおよび使用可能なイベントのタイプを示します。

イベントの各タイプには、1文字か2文字のイベントコードが割り当てられています。最初の文字はイベントのタイプを表し、2番目の文字はサブタイプを表します。

例えば、イベントコード **PL** で、**P** はイベントタイプ "プログラム" を表し、**L** はサブタイプ "ロード" を表します。

イベント	イベントコード
セッションの初期化時。	SI
セッションの終了時。	ST
プログラムのロード時。	PL
プログラムの起動時。	PS
プログラムの終了時。	PT
データベースコールの前。	DB
データベースコールの後。	DA
端末 I/O の前。	IB
端末 I/O の後。	IA
Natural 以外のプログラムのコールの前。	CB
Natural 以外のプログラムのコールの後。	CA
ランタイムエラー時。	E
内部トレースコール時。	N
内部トレースによって提供される情報は、Software AG の担当者によるデバッグの目的でのみ使用されます。 次の「内部トレースコールイベントをアクティブにするには」も参照してください。	
Natural ステートメント時。	NS
アウトバウンド RPC メッセージ時。	RO
インバウンド RPC メッセージ時。	RI
RPC 要求の実行開始時。	RS
ユーザー定義イベント時。	U

### ▶手順 102.1. 内部トレースコールイベントをアクティブにするには

- 1 Natural プロファイルパラメータ ITRACE=ON を指定します。
- 2 次の方法によって、内部トレースコールを発行する Natural コンポーネントを定義します。

- Natural パラメータモジュールの NTTRACE マクロを使用。

または、ダイナミック Natural プロファイルパラメータ TRACE を使用。

NTTRACE マクロおよび ITRACE パラメータと TRACE パラメータの詳細については、『パラメータリファレンス』ドキュメントを参照してください。

## 収集されるデータ

Natural Data Collector がイベントで収集するデータについては、次のセクションで説明します。

データのレイアウトの詳細については、Natural ソースデータセット NAMRDC、または Natural システムライブラリ SYSRDC に提供されているローカルデータエリア RDCDATA を参照してください。

収集されるデータは、一般データとイベント固有データの2つのカテゴリに分類されます。

- 一般データ
- イベント固有データ

### 一般データ

次の一般データは、すべてのイベントで収集されます。

- Software AG 製品名
- 製品バージョン
- オペレーティングシステム
- TP モニタ
- Natural ニュークリアスとバッファプールのアドレッシングモードや常駐モードなどの実行モード。
- TP ユーザーまたはバッチジョブ名
- TP 端末 ID
- 現在の Natural ユーザー ID
- 現在の Natural Security ユーザーグループ ID
- 現在の Natural ライブラリ
- 現在の Natural プログラム、
- 現在のプログラムレベル、
- 現在実行されているプログラムステートメントの行番号。

## イベント固有データ

次のデータは、次のイベントでのみ収集されます。

イベント	データ要素
セッションの初期化	なし
セッションの終了	終了リターンコード。 Natural 終了メッセージコード NAT99nn。 バックエンドプログラムの名前。
プログラムのロード	ロードされるプログラムの名前。 ロードライブラリの名前。
プログラムの開始/終了	プログラムタイプ。 プログラム名。 プログラムライブラリ名。 プログラムライブラリのデータベース ID。
データベースコール	データベースタイプ。 コマンドコード。 コマンド ID。 データベース ID。 レスポンスコード。 パラメータアドレスをリストするポインタ（ユーザー出口にのみ有用）。
端末 I/O	送信されるバイト数。 読み込まれるバイト数。 割り当てられたセッションストレージの合計。 圧縮セッションストレージ長。
Natural 以外のプログラムのコール	呼び出されたプログラムの名前。 ダイナミックモードやスタティックモードなどのコールモード。 プログラムリンクの場所。 パラメータタイプ。 パラメータアドレス。 プログラムのエントリアドレス。 レスポンスコード。
ランタイムエラー	Natural システムエラーメッセージコード。 外部アベンドコード。 Natural プロファイルパラメータ ETA で指定されたエラー処理プログラムの名前（『パラメータリファレンス』ドキュメントの該当するセクションを参照）。
内部トレース	Natural ニュークリアスコンポーネントに関する最大 250 バイトの情報。 提供される情報は、Software AG の担当者によるデバッグの目的でのみ使用されます。
RPC 要求	RPC-specific 情報：  環境：C = クライアント、S = サーバー。  RDC サブタイプ S：

イベント	データ要素
	サブプログラム名、 Adabas ユーザー ID (ETID)、 会話ステータス、 ログオンインジケータ (Y=ログオン実行)、 RPC 要求の偽装インジケータ (Y=偽装実行)。  RDC サブタイプ O および I: トランスポートプロトコル、 RPC 機能、 クライアントユーザー ID のタイプ、 メッセージの長さ、 トランスポート層のリターンコード、 外部会話 ID、 クライアントユーザー ID、 サーバーノード (クライアント専用アウトバウンドメッセージ)、 サーバー名 (クライアント専用アウトバウンドメッセージ)。
ユーザー定義イベント	最大 250 バイトのユーザー定義情報。

## Natural Data Collector のアクティブ化

Natural Data Collector は、Natural プロファイルパラメータ RDCSIZE によってアクティブ化され、制御されます (『パラメータリファレンス』ドキュメントも参照)。

このパラメータでは、Natural Data Collector の RDCSIZE バッファのサイズ (KB 単位) が決定され、[トレースの記録](#)が制御されます (該当するセクションを参照)。

デフォルトでは、パラメータは RDCSIZE=0 に設定されます。この場合、Natural Data Collector は非アクティブになります。

### ▶手順 102.2. Natural Data Collector をアクティブにするには

- RDCSIZE=2 に設定します。

Natural Data Collector はアクティブになり、収集されたデータがユーザー出口に供給されます。データは RDCSIZE バッファに記録されません。

データを RDCSIZE バッファに記録する方法については、次の「[トレースの記録](#)」を参照してください。

## トレースの記録

収集されたイベントデータは、常にユーザー出口に供給され、外部モニタリング／計算プログラムで使用されます（該当するセクションを参照）。現在の Natural セッションのイベントデータは、ユーザー出口を使用せずに RDCSIZE バッファに記録できます。この機能は、テストの目的に使用すると便利です。このセクションでは、RDCSIZE バッファへのデータの記録をトレースの記録と呼びます。

トレースの記録は、セッションの開始時に開始され、セッションの終了時に終了します。RDCSIZE バッファに記録されたデータには、同じセッション内の任意の Natural プログラムからアクセスできます。

RDCSIZE バッファはラップアラウンドモードでデータが記録されます。つまり、バッファがいっぱいになると、最も古い記録が上書きされます。セッションの終了時に、バッファの内容は削除されます。

トレースの記録は、プロファイルパラメータ **RDCSIZE** でアクティブにされます（『パラメータリファレンス』ドキュメントの該当するセクションも参照）。このパラメータでは、RDCSIZE バッファのサイズも決定されます。

### ▶手順 102.3. トレースの記録をアクティブにし、使用するには

- 1 Natural プロファイルパラメータ **RDCSIZE** を 2 より大きい値に設定します。

次の式を使用して、RDCSIZE バッファに収まるレコードの数を計算します。

$$11 * (RDCSIZE - 2) = \text{レコードの数}$$

現在の Natural セッションについて收拾されたすべてのイベントデータは、RDCSIZE バッファに記録されます。

- 2 「**CMRDC インターフェイスの呼び出し**」の説明に従って、CMRDC インターフェイスを使用し、RDCSIZE バッファの内容を取得します。
- 3 必要に応じて、CMRDC インターフェイスを使用して、停止または再開、および RDCSIZE バッファでのトレースの記録を実行し、データを記録するイベントを指定したり、「**CMRDC インターフェイスの呼び出し**」で説明されている追加機能を実行します。

## 外部モニタリング／計算用ユーザー出口


イベントデータは、Naturalセッションでの活動を評価するために外部モニタリング／計算プログラムに渡すことができます。この処理は、ユーザー出口および提供されているユーザー出口プログラムの例によって実行されます。ユーザー出口プログラムは、アセンブリ言語で作成されます。

Natural Data Collector では、任意の数のユーザー出口がサポートされます。ユーザー出口は、Natural プロファイルパラメータ RDCEXIT で定義できます（『パラメータリファレンス』ドキュメントの該当するセクションを参照）。外部モニタリング／計算プログラムは、各ユーザー出口にアタッチできます。

RDCEX1、RDCEX2、および RDCEX3 の3つの出口名が事前定義されています。Natural ニュークリアスにリンクされる出口のエントリポイントとしてこれらの名前の1つを使用した場合、Natural プロファイルパラメータ RDCEXIT は必要ありません。

「[データ収集イベント](#)」で示したすべてのイベントで、ユーザー出口は次の表に示す標準リネージコール会話を使用して、制御を引き継ぎます。

登録	内容
1	2つのアドレスで構成されるパラメータアドレスリストをポイントします。1つのアドレスは一般データをポイントし、もう1つのアドレスはイベント固有データをポイントします。これらのエリアのレイアウトは、それぞれ DSECT RDCGDATA および RDCLDATA によってマップされます。どちらの DSECT も、Natural マクロ NAMRDC にソース形式で提供されます。
13	72 バイトの標準セーブエリアをポイントします。
14	リターンアドレスを含みます。
15	エントリポイントアドレスまたはユーザー出口のリターンコードのいずれかが含まれます。

 **注意:** ユーザー出口は CMRDC インターフェイスを使用しないで呼び出されます。

ユーザー出口プログラムは、Natural と同じ属性である必要があります。つまり、同じアドレスリングモードを持ち、リエントラントである必要があります。ユーザー出口プログラムは、静的にリンクされた Natural 以外のプログラムの規則に従って、Natural ニュークリアスとリンクしている必要があります。『パラメータリファレンス』ドキュメントで説明しているプロファイルパラメータ「CSTATIC」および「RCA」も参照してください。

デフォルトでは、400バイトの排他的ワークエリア（セッション当たり）が各ユーザー出口に提供されます（RDCGWRKA フィールド）。ユーザー出口により大きいワークエリアが必要な場合は、Natural プロファイルパラメータ RDCEXIT のユーザー出口名の後に指定できます（『パラメータリファレンス』ドキュメントの該当するセクションを参照）。ワークエリアの長さは RDCGWRKL フィールドのユーザー出口に渡され、整合性チェックに使用されます。このワークエリアの場所は、Natural の再配置によってセッション中に変更される場合がありますが、内容は維持されます。

TP モニタ環境では、CICS での CSA アドレスなどの TP アンカーアドレスが提供されます (RDCGANCH フィールド)。このアドレスを使用して、システム情報にアクセスできます。

ユーザー出口プログラムの実行中にプログラムチェックが発生した場合、アベンド状況の繰り返しを避けるため、残りのセッションに対する以降のデータ収集は無効になります。

以降のセクションでは、次の項目について説明します。

- リターンコード
- ユーザー出口プログラムの例

## リターンコード

ゼロ以外のリターンコードは、次の 2 つのイベントにのみサポートされます。

- データベースコールの前。ここでは、コントロールブロックに格納された Adabas レスポンスコードをレジスタ 15 に格納できます。Adabas コールは実行されません。
- プログラムの起動時。ここでは、Natural エラーメッセージコードをレジスタ 15 に格納できます。プログラムは実行されませんが、指定した数でエラー条件が発生します。

## ユーザー出口プログラムの例

次のユーザー出口プログラムの例が、Natural ソースデータセットに含まれています。

プログラム	データセット	用途
NAMRDC	NAT $nnn$ .SRCE	一般データおよびイベント固有データ用の DSECT マクロ。
XNATRDC1	NAT $nnn$ .SRCE	z/OS および z/VSE オペレーティングシステム環境の Natural TSO インターフェイスと Natural バッチインターフェイス。
XNCFRDC1	NAT $nnn$ .SRCE	Com-plete。
XNCIRDC1	NCI $nnn$ .SRCE	Natural CICS インターフェイス。

## CMRDC インターフェイスの呼び出し

CMRDC アプリケーションプログラミングインターフェイスは、RDCSIZE バッファに記録されたトレースデータを取得および制御するために使用されます。この機能は、テストの目的に使用すると便利です。

### ▶ 手順 102.4. CMRDC インターフェイスを呼び出すには

- Natural プログラムで、CALL ステートメントを発行します。

CMRDC インターフェイスで使用できる機能および対応する Natural CALL ステートメントに適用される構文については、次のセクションで説明します。

また、Natural システムライブラリ SYSRDC には、プログラム例および該当するローカルデータエリア RDCDATA が含まれています。

以降のセクションでは、次の項目について説明します。

- **トレースレコードの取得**
- **トレースの記録の停止と再開**
- **トレースの記録のイベントタイプの選択**
- **User-Defined イベント**
- **トレースのステータスの取得**
- **CMRDC リターンコード**
- **ライブラリ SYSRDC のプログラム例**

### トレースレコードの取得

RDCSIZE バッファからデータを読み込むには、次の Natural ステートメントを使用して CMRDC を呼び出します。

```
CALL 'CMRDC' function event-time gen-data event-data seq.-number
```

次のパラメータが渡されます。

パラメータ	フォーマット/長さ	説明
<i>function</i>	A1	使用可能な関数：  F 最初のトレースレコードを取得します。 G 次のトレースレコードを取得します。 N 指定したシーケンス番号のレコードを取得します。
<i>event-time</i>	N10 または N12	イベントの時刻。  時刻は次のいずれかの形式で返されます。  <i>HHMMSSXXXX</i> HH=時間、MM=分、SS=秒、XXXX=秒の小数部分  <i>HHMMSSXXXXXXXX</i> HH=時間、MM=分、SS=秒、XXXXXX=秒の小数部分
<i>gen-data</i>	A252	一般データ
<i>event-data</i>	A252	イベント固有データ



パラメータ	フォーマット/長さ	説明
<i>seq. -number</i>	I4	レコードのシーケンス番号。 関数 <b>N</b> にのみ適用されます（上記参照）。

取得機能は、RDCSIZEバッファでのデータ記録を暗黙的に停止します。記録を再開するには、次に示す関数 **S** を使用します。

### トレースの記録の停止と再開

RDCSIZE バッファでデータのトレースの記録を停止または再開するには、次のステートメントを使用して CMRDC を呼び出します。


```
CALL 'CMRDC' function
```

次のパラメータが渡されます。

パラメータ	フォーマット/長さ	説明
<i>function</i>	A1	使用可能な関数：  S RDCSIZE バッファをクリアし、トレースの記録を再開します。 P トレースの記録を停止します。

### トレースの記録のイベントタイプの選択

デフォルトでは、トレースの記録にすべてのイベントが選択されます。特定のイベントのみを記録する場合は、この関数を使用します。

 **注意:** この関数では、RDCSIZE バッファにデータが記録されるイベントのみが選択されます。ユーザー出口に渡されるデータには影響を与えません。また、トレースの記録のステータス（開始/停止）にも影響を与えません。

記録するイベントのタイプを選択するには、次のステートメントを使用して CMRDC を呼び出します。

```
CALL 'CMRDC' function type...
```

次のパラメータが渡されます。

パラメータ	フォーマット／長さ	説明
<i>function</i>	A1	使用可能な関数：  T トレース記録のイベントを選択します。
<i>type</i>	A1、A2、A3	「データ収集イベント」で示した、記録するイベントのタイプの1文字または2文字のイベントコード。  必要なイベントコードの任意の数のパラメータを指定します。  または、イベントの範囲を選択することも、イベントを選択しないこともできます。  ALL すべてのイベント。 <i>value*</i> <i>value</i> で始まるイベントコードを持つすべてのイベント。 例えば、 <b>P*</b> では、プログラムタイプのすべてのイベントPL、PS、およびPTが選択されます。  空白 空白文字ではイベントは選択されません。

### User-Defined イベント

ユーザー定義イベントを指定するには、次のステートメントを使用してCMRDCを呼び出します。

```
CALL 'CMRDC' function record
```

次のパラメータが渡されます。

パラメータ	フォーマット／長さ	説明
<i>function</i>	A1	使用可能な関数：  U ユーザー定義トレースイベント。
<i>record</i>	Annn	最大250バイトの長さ ( <i>nnn</i> ) のトレース記録

## トレースのステータスの取得

トレースのステータスを取得するには、次のステートメントを使用して CMRDC を呼び出します。

```
CALL 'CMRDC' function status types
```

次のパラメータが渡されます。

パラメータ	フォーマット/ 長さ	説明
<i>function</i>	A1	使用可能な関数：  C トレースのステータスを取得します。
<i>status</i>	A1	現在のトレースのステータス：  S トレースはアクティブです。 P トレースは非アクティブです。
<i>types</i>	Ann	現在設定されているイベントタイプの2文字のイベントコード（「 <a href="#">トレースの記録のイベントタイプの選択</a> 」を参照）。  すべてのイベントタイプに十分なスペースを提供するには、 <i>types</i> 変数に64バイトの最小長 ( <i>nn</i> ) が必要です。  このパラメータは任意指定です。

## CMRDC リターンコード

コード	意味
0	機能は正常に実行されました。
4	最後のトレース記録 関数 <b>F</b> および <b>G</b> にのみ適用されます（上記参照）。
8	この関数のパラメータが少なすぎます。
12	ファンクションコードが正しくありません。
16	Natural Data Collector は非アクティブです。例えば、RDCSIZE=0 です。
20	エラーの後、Natural Data Collector は無効になっています。
24	トレースの記録に使用できるバッファスペースがありません（RDCSIZE=2 以下）。
28	パラメータ値が正しくありません。 関数 <b>T</b> および <b>N</b> にのみ適用されます（上記参照）。

## ライブラリ SYSRDC のプログラム例

Natural システムライブラリ SYSRDC には、次のプログラム例が含まれています。

プログラム	機能
RDCDISP	RDCSIZE バッファ内のすべてのレコードを表示し、プログラムで指定されたフィールドを示します。次の「プログラム RDCDISP の出力例」も参照してください。
RDCSTART	トレースの記録を再開します。
RDCSTOP	トレースの記録を停止します。
RDCSET	トレース記録のイベントを選択します。
RDCUSER	ユーザー定義イベント。
RDCSTAT	トレースのステータスを取得します。

### プログラム RDCDISP の出力例

次の画面例は、プログラム例 RDCDISP によって生成された出力レポートの抽出を示しています。

ETIME1	TY	GGUID	LV	GPGM	T	GCAPL	PRLIB	PRNAM	CO	DB	FN	RC
0.0000	SI	SAG				SYSTEM						
0.0002	DB	SAG				SYSTEM			OP	10		
0.0002	DA	SAG				SYSTEM			OP	10		
0.0000	DB	SAG				SYSTEM			S1	10	1640	
0.0005	DA	SAG				SYSTEM			S1	10	1640	
0.0001	PL	SAG				SYSTEM	SYSLIB	MAINMENU				
0.0001	PS	SAG	1	MAINMENU	F	SYSTEM	SYSLIB	MAINMENU		10	1640	
0.0001	PL	SAG	1	MAINMENU		SYSTEM	SYSLIB	NAT00029				
0.0000	DB	SAG	1	MAINMENU		SYSTEM			S1	10	1640	
0.0002	DA	SAG	1	MAINMENU		SYSTEM			S1	10	1640	
0.0000	PS	SAG	2	NAT00029	N	SYSTEM	SYSLIBS	NAT00029		10	1640	
0.0000	CB	SAG	2	NAT00029		SYSTEM		CMMPP	S			
0.0000	CA	SAG	2	NAT00029		SYSTEM		CMMPP	S			
0.0000	PT	SAG	2	NAT00029	N	SYSTEM	SYSLIBS	NAT00029				
0.0000	PL	SAG	1	MAINMENU		SYSTEM	SYSLIB	USR2003P				
0.0000	PS	SAG	2	USR2003P	N	SYSTEM	SYSLIB	USR2003P		10	1640	
0.0001	PT	SAG	2	USR2003P	N	SYSTEM	SYSLIB	USR2003P				
0.0000	CB	SAG	1	MAINMENU		SYSTEM		CMUB	S			

次の表で、画面例に表示される列およびそれらの列で参照される変数について説明します。詳細については、プログラムソースのコメントおよびローカルデータエリア RDCDATA を参照してください。示されている Natural システム変数の詳細については、『システム変数』ドキュメントを参照してください。

列	説明
ETIME1	現在のイベントと前のイベントの実行の間隔（秒単位）。
TY	「 <a href="#">データ収集イベント</a> 」で示したイベントのタイプ。
GCAUID	Natural システム変数 *USER によって割り当てられた現在の Natural ユーザー ID。
LV	プログラムレベル。
GPGM	Natural システム変数 *PROGRAM によって割り当てられた現在のプログラムの名前。
T	プログラムのタイプ。
GCAPL	Natural システム変数 *APPLIC-ID によって割り当てられた現在のアプリケーションライブラリの ID。
PRLIB	プログラムが格納されるライブラリの ID。 例えば、イベントコード PL などのプログラムタイプのイベントに適用されます。
PRNAM	イベントのタイプについてロードされるプログラムの名前。
CO	データベースコマンド。
DB	データベース ID。
FN	データベースのファイル番号。
RC	データベースのレスポンスコード。



# 103

## SYSRPC ユーティリティ

---

ユーティリティ SYSRPC は、リモートプロシージャコールをメンテナンスするために使用します。

- **SYSRPC の基本的な機能**
- **Service Directory Maintenance**
- **Stub Generation**
- サイズ要件の計算
- **Parameter Maintenance**
- **Server Command Execution**
- **Remote Directory Maintenance**

関連トピック：

- SYSRPC ユーティリティ機能を適用してサーバーとクライアントシステム間の通信フレームワークを確立する方法の詳細については、『*Natural* リモートプロシージャコール (RPC)』ドキュメントを参照してください。
- SYSRPC ユーティリティに関連する表記の説明は、『*Natural* リモートプロシージャコール (RPC)』ドキュメントの「*Natural* RPC の用語」を参照してください。
- SYSRPC の使用は、*Natural Security* によって制御できます。『*Natural Security*』ドキュメントの「*Protecting Utilities*」を参照してください。
- リモートプロシージャコールをメンテナンスするために提供されたアプリケーションプログラミングインターフェイスの詳細については、『*Natural* リモートプロシージャコール (RPC)』ドキュメントの「*Natural* RPC で使用するアプリケーションプログラミングインターフェイス」を参照してください。





# 104      SYSRPC の基本的な機能

---

■ SYSRPC の呼び出し .....	724
■ SYSRPC の終了 .....	725
■ オンラインヘルプの起動 .....	725

このセクションでは、SYSRPCユーティリティを起動、終了する手順、およびヘルプ機能呼び出す手順について説明します。

## SYSRPC の呼び出し

---

システムコマンドまたはメニュー機能を使用して、SYSRPCユーティリティを呼び出すことができます。

### ▶手順 104.1. システムコマンドを使用して SYSRPC を呼び出すには

- コマンド行に次のコマンドを入力します。

```
SYSRPC
```

Enter キーを押します。

SYSRPC ユーティリティの **[Client Maintenance]** メニューが表示されます。

### ▶手順 104.2. メニュー機能を使用して SYSRPC を呼び出すには

- 1 Natural の **[Main Menu]** から、**[Maintenance and Transfer Utilities]** を選択し、Enter キーを押します。

**[Maintenance and Transfer Utilities]** メニューが表示されます。

- 2 **[Maintain Remote Procedure Calls]** を選択し、Enter キーを押します。

SYSRPC ユーティリティの **[Client Maintenance]** メニューが表示されます。

**[Client Maintenance]** メニューから、RPC (リモートプロシージャコール) のメンテナンスに使用可能な次のすべての機能呼び出すことができます。

- **Service Directory Maintenance**
- **Stub Generation**
- **Parameter Maintenance**
- **Server Command Execution**
- **Remote Directory Maintenance**

これらの機能の詳細については、関連するセクションを参照してください。

---

## SYSRPC の終了

---

▶手順 104.3. SYSRPC ユーティリティを終了するには

- [Client Maintenance] メニューの [Code] フィールドに、ピリオド (.) を入力します。  
または:  
PF3 キー (終了) を押します。

---

## オンラインヘルプの起動

---

▶手順 104.4. オンラインヘルプ機能を起動するには

- PF1 (Help) キーを押します。



# 105 サービスディレクトリメンテナンス

---

■ サービスディレクトリの概念 .....	728
■ Service Directory Maintenance の呼び出し .....	730
■ [Service Directory] 画面のフィールド .....	732
■ Service Directory Maintenance のコマンド .....	734
■ 論理ノード名および論理サービスの定義 .....	737

**Service Directory Maintenance** 機能は、クライアントの呼び出し元プログラムをサーバーのサブプログラムに接続するためにサービスディレクトリをメンテナンスする場合に使用します。

サービスディレクトリ情報は、NTRPC/RPC キーワードサブパラメータ RPCSDIR で定義されているライブラリ内のNATCLTGSサブプログラムに格納されます（『パラメータリファレンス』ドキュメントを参照）。RPCSDIR が設定されている場合、サービスディレクトリメンテナンス機能は、RPCSDIR で指定されているライブラリを参照します。RPCSDIR が設定されていない場合（デフォルト）、ユーザーがログオンしているライブラリが参照されます。この場合は、サービスディレクトリメンテナンス機能を実行する前に、実行時にクライアントにより使用されるライブラリ（またはその steplib の 1 つ）にログオンします。

サービスディレクトリメンテナンス用に参照されるライブラリの名前は、**[Service Directory]** 画面の右上隅に表示されます（「*Service Directory Maintenance* の呼び出し」を参照）。RPCSDIR が設定されている場合、画面タイトルに **[Central]** が含まれます。これは、画面に表示されているライブラリが、現在ユーザーがログオンしているライブラリではなく、RPCSDIR で指定されている中央ライブラリであることを示しています。

注意：

NATCLTGS が Natural システムライブラリ SYSRPC に保存されている場合、NATCLTGS をアプリケーションライブラリまたはその steplib のいずれかに移動することを強くお勧めします。

**Service Directory Maintenance** 機能を適用する方法の詳細については、『*Natural* リモートプロシージャコール (RPC)』ドキュメントの「*Natural* RPC 環境の運用」で「RPC サーバーアドレスの指定」を参照してください。

## サービスディレクトリの概念

---

サービスディレクトリには、下位のフィールドを上位のフィールドに割り当てるためのカスケードリストを含む階層構造があります。最上位の階層レベルはノードで、最下位の階層レベルはプログラムです。ノード、サーバー、ライブラリ、およびプログラムを同じ行に入力することはできません。同じ行に入力した場合、該当するエラーメッセージが表示されます。下位のフィールドの値は、上位のフィールドの下に行に入力する必要があります。複数のサーバーを 1 つのノードに、複数のライブラリを 1 台のサーバーに、複数のプログラムを 1 つのライブラリにそれぞれ割り当てることができます。

サービスディレクトリに指定されたノードおよびサーバー名は、物理名または論理名と論理サービスです。

- **物理ノードおよびサーバー**

## ■ ロケーショントランスペアレンシ

### 物理ノードおよびサーバー

物理ノードとサーバー名は、実際のノード（有効な TCP/IP または Entire Net-Work アドレス）とサーバーの名前を表します。

「例1 - *[Service Directory]* の標準ビュー」では、2 台のサーバーが1つのノードに対して定義されます。両方のサーバーが同じノード ETB045 に接続されます。サブプログラム SUB1 に対するリモート CALLNAT がサーバー NRPC001 上で実行されます。一方、サブプログラム SUB2 および SUB3 はサーバー NRPC002 上で実行されます。

ここで指定するサーバー名は、NTRPC/RPC キーワードサブパラメータ SRVNAME（『パラメータリファレンス』ドキュメントを参照）で指定したサーバー名と同一である必要があります。同じように、サービスディレクトリのノード名は、NTRPC/RPC キーワードサブパラメータ SRVNODE（『パラメータリファレンス』ドキュメントを参照）でサーバーに指定したノード名と同一である必要があります。

### ロケーショントランスペアレンシ

ロケーショントランスペアレンシとは、物理ノード名を論理ノード名で置き換えることができ、物理ノードとサーバー名の組み合わせを論理サービスで置き換えることができるという概念です。

論理ノード名と論理サービスは EntireX で定義され、Natural ランタイム時に物理ノードとサーバー名に対して割り当てられます。

「例1 - *[Service Directory]* の標準ビュー」で、**[Node]** フィールドの \*LOCTRAN は、**[Server]** フィールドに論理サービス NRPC001-LOGICAL が含まれることを示します。**[Node]** フィールドの LOGBROKER=NODE は、論理ノード名を示します。

関連トピック：

- [論理ノード名および論理サービスの定義](#)。
- 『Natural リモートプロシージャコール (RPC)』ドキュメントの「*Natural RPC 環境の運用*」セクションの「*ロケーショントランスペアレンシの使用*」。
- EntireX ドキュメントの関連セクション。

## Service Directory Maintenance の呼び出し

注意：

**Service Directory Maintenance** 機能では、Natural エディタを呼び出します。その結果、**Service Directory Maintenance** の呼び出し時に、ソースワークエリアに保存されたデータが失われる可能性があります。該当するメッセージで、不用意に既存のエントリを削除しないよう警告されます。PF12 キーを押して機能をキャンセルするか、または Enter キーを押して操作を確認し、ソースワークエリアをクリアします。

### ▶手順 105.1. Service Directory Maintenance 機能呼び出すには

- 1 **[Client Maintenance]** メニューの **[Code]** フィールドに、次のコマンドを入力します。

```
SM
```

- 2 ENTER キーを押します。

- サービスディレクトリにすでにサービス定義が含まれている場合、ウィンドウが開いて次のメッセージが表示されます。

```
Existing service definitions found
```

ウィンドウの **[Code]** フィールドに「A」（デフォルト）を入力し、古い定義を保持して新しい定義を追加し、Enter キーを押します。

または

ウィンドウの **[Code]** フィールドに「I」を入力し、既存のすべての定義を無視してサービスディレクトリから削除し、Enter キーを押します。

次に示すように、**[Service Directory]** 画面の標準ビューが表示されます。

#### 例 1 - **[Service Directory]** の標準ビュー

15:32:25		*** NATURAL Remote Procedure Call ***				2004-04-14
		Service Directory				SYSRPC
	Node	Tr.	Server	Logon	Library	Program
1	ETB045_____	B	_____	—	_____	_____
2	_____	—	NRPC001_____	N	_____	_____
3	_____	—	_____	—	SYSTEM__	_____
4	_____	—	_____	—	_____	SUB1_____
5	_____	—	NRPC002_____	Y	_____	_____
6	_____	—	_____	—	SYSTEM__	_____
7	_____	—	_____	—	_____	SUB2_____



```

8      _____  -  _____  -  _____  SUB3_____
9      *LOCTRAN_____  -  _____  -  _____  _____
10     _____  B  NRPC001-LOGICAL_  N  _____  _____
11     _____  -  _____  -  _____  SYSTEM_____
12     _____  -  _____  -  _____  SUB1_____
13     LOGBROKER=NODE  B  _____  N  _____  _____
14     _____  -  NRPC002_____  N  _____  _____
15     _____  -  _____  -  _____  SYSTEM_____
16     _____  -  _____  -  _____  S?B*_____

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  LocTr Exit  Find  -H   +H   -P   +P   Top  Bot  <   Canc
  
```

【Service Directory】画面には、最大 500 行入力できます。

- PF11 キーを押すか、コマンド行に小なり (<) 記号を入力すると、次の例のような【Service Directory】画面の拡張ノード／サーバービューが表示されます。

例 2 - 【Service Directory】の拡張ノード／サーバービュー

```

14:48:33          *** NATURAL Remote Procedure Call ***          2004-04-14
                  Service Directory                               SYSRPC

      Node                Tr.                Server                Logon
1  ETB045_____  B  _____  _____  _____
2  _____  -  NRPC001_____  N  _____  _____
3  _____  -  _____  -  _____  _____
4  _____  -  _____  -  _____  _____
5  _____  -  NRPC002_____  Y  _____  _____
6  _____  -  _____  -  _____  _____
7  _____  -  _____  -  _____  _____
8  _____  -  _____  -  _____  _____
9  *LOCTRAN_____  -  _____  -  _____  _____
10 _____  B  NRPC001-LOGICAL_____  N  _____  _____
11 _____  -  _____  -  _____  _____
12 _____  -  _____  -  _____  _____
13 LOGBROKER=NODE_____  B  _____  _____  N  _____  _____
14 _____  -  NRPC002_____  N  _____  _____
15 _____  -  _____  -  _____  _____
16 _____  -  _____  -  _____  _____

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  LocTr Exit  Find  -H   +H   -P   +P   Top  Bot  >   Canc
  
```

PF11 キーを押すか、コマンド行に大なり (>) 記号を入力すると、「例1 - [Service Directory] の標準ビュー」に示すような [Service Directory] 画面の標準ビューが表示されます。

## [Service Directory] 画面のフィールド

[Service Directory] 画面には、次の入力フィールドがあります（1行当たり1エントリ）。

フィールド	説明		
Node	<p>リモート CALLNAT の送信先であるノードの名前。『Natural リモートプロシージャコール (RPC)』ドキュメントの「Natural RPC の用語」も参照してください。</p> <p>入力の最大長は次のとおりです。</p> <p>[Service Directory] 画面の標準ビュー： 16 文字                      [Service Directory] 画面の拡張ノード/サーバービュー： 32 文字                      [Location Transparency] ウィンドウの使用（「ダイレクトコマンドと PF キー」の「PF2」の項を参照）： 192 文字</p>		
Tr.	<p>トランスポートプロトコル。</p> <p>B は EntireX Broker ACI プロトコルを示します。</p>		
Server	<p>リモート CALLNAT の送信先であるサーバーの名前。『Natural リモートプロシージャコール (RPC)』ドキュメントの「Natural RPC の用語」も参照してください。</p> <p>入力の最大長は次のとおりです。</p> <p>[Service Directory] 画面の標準ビュー： 16 文字                      [Service Directory] 画面の拡張ノード/サーバービュー： 32 文字                      [Location Transparency] ウィンドウの使用（「ダイレクトコマンドと PF キー」の「PF2」の項を参照）： 192 文字</p>		
Logon	<p>サーバーへの Natural ログオンを開始します。</p> <hr/> <p>ログオンの開始はサーバーまたはノードレベルで可能で、下位レベルの階層で作成されたすべての定義に適用されます。Logon オプションを特定のサーバーに設定した場合、すべての関連ライブラリとサブプログラムの定義に適用されます。</p> <hr/> <p>設定可能な値は次のとおりです。</p> <hr/> <table border="1"> <tr> <td>Y</td> <td>Y (Yes) に設定した場合、各非会話型 CALLNAT 要求または会話の開始ごとに、クライアントでは、[Server]</td> </tr> </table>	Y	Y (Yes) に設定した場合、各非会話型 CALLNAT 要求または会話の開始ごとに、クライアントでは、[Server]
Y	Y (Yes) に設定した場合、各非会話型 CALLNAT 要求または会話の開始ごとに、クライアントでは、[Server]		

フィールド	説明					
		フィールドに属する下位の [Library] 列のライブラリに関係なく、クライアントの現在のライブラリ名を使用して、サーバーへの Natural ログオンが開始されます。アプリケーションプログラミングインターフェイス USR4008N を使用して、異なるライブラリを指定できます（「ログオンオプションの使用」の「異なるライブラリへのログオン」を参照）。				
	N または <i>blank</i>	N (No) に設定するか、または値を入力しない場合、ログオンは開始されません。				
	リモート CALLNAT が実行された（正常または失敗）後、または会話の終了時に、サーバーライブラリは前の状態にリセットされます。詳細については、『Natural リモートプロシージャコール (RPC) 』ドキュメントの「ログオンオプションの使用」を参照してください。					
	「 <a href="#">サーバーコマンドの実行</a> 」も参照してください。					
<b>Library</b>	SYSTEM、またはリモート CALLNAT の実行時にクライアントアプリケーションがログオンするライブラリの名前。					
<b>Program</b>	クライアントからアクセスするリモートサブプログラムの名前。  名前または名前の範囲を入力できます。有効な名前は、1つ以上のアスタリスク (*) または1つ以上の疑問符 (?)、あるいはその両方を含む1文字以上の英数字の任意の組み合わせです。  アスタリスク (*) は任意の文字列を示し、疑問符 (?) は1文字を示します。  無効な組み合わせは次のとおりです。  <table border="1" data-bbox="358 1365 1474 1570"> <tr> <td data-bbox="358 1365 792 1486">*?</td> <td data-bbox="792 1365 1474 1486">アスタリスクの後に疑問符を入力すると、?*に変換されます。</td> </tr> <tr> <td data-bbox="358 1486 792 1570">**</td> <td data-bbox="792 1486 1474 1570">2つ以上の連続したアスタリスクは1つのアスタリスクに変換されます。</td> </tr> </table>		*?	アスタリスクの後に疑問符を入力すると、?*に変換されます。	**	2つ以上の連続したアスタリスクは1つのアスタリスクに変換されます。
*?	アスタリスクの後に疑問符を入力すると、?*に変換されます。					
**	2つ以上の連続したアスタリスクは1つのアスタリスクに変換されます。					

### ノードとサーバーの選択条件

Natural ランタイム時、ノードおよびサーバーの選択は、[Program] フィールドと [Library] フィールドの値によって異なります。次の条件に従います。

非会話型 CALLNAT

1. [Library] フィールドに、現在のアプリケーションライブラリの名前または SYSTEM が含まれている必要があります。
2. CALLNAT ステートメントに指定するサブプログラムの名前は、ポイント (1) の [Library] フィールドに属する [Program] フィールドに含まれている必要があります。

会話型 CALLNAT

1. [Library] フィールドに、現在のアプリケーションライブラリの名前または SYSTEM が含まれている必要があります。
2. OPEN CONVERSATION ステートメントに指定するサブプログラムは、ポイント (1) の [Library] フィールドに属する [Program] フィールドに含まれている必要があります。Library

非会話型または会話型 CALLNAT に使用されるノードとサーバーは、ポイント (1) の [Library] フィールドの上位 [Node] フィールドおよび [Server] フィールドから取得されます。

## Service Directory Maintenance のコマンド

---

このセクションでは、[Service Directory] 画面で使用できるコマンドに関する情報について説明します。

- 行コマンド
- ダイレクトコマンドと PF キー

### 行コマンド

[Service Directory] 画面で使用できる行コマンドは、フィールド値を含む1行または複数の行をコピー、移動、または削除する場合に使用します。

行コマンドは行の先頭に入力します。つまり、シーケンス番号を上書きします。それから ENTER キーを押します。

「行ブロックをコピーまたは移動するには」およびダイレクトコマンド **RESET** の項も参照してください。

行コマンド	機能
A	CC または MM でマークされた行ブロックを、コマンドが入力された行の下にコピーまたは移動します。
CC	コピーする行ブロックをマークします。
D	マークされた行を削除します。
DD	行ブロックをマークして削除します。  ブロックの最初の行と最後の行にこのコマンドを入力して行ブロックをマークし、Enter キーを押してコマンドを実行します。
I	コマンドが入力された行の下に空行を 5 行挿入します。
MM	移動する行ブロックをマークします。
P	CC または MM でマークされた行ブロックを、コマンドが入力された行の上にコピーまたは移動します。

#### ▶手順 105.2. 行のブロックをコピーまたは移動するには

- 1 ブロックが開始する行の先頭で、次のいずれかの行コマンドでシーケンス番号を上書きします。

CC

これがブロックをコピーする場合のコマンドです。

MM

これがブロックを移動する場合のコマンドです。

- 2 ブロックが開始する行の末尾で、次のいずれかの行コマンドでシーケンス番号を上書きします。

CC

これがブロックをコピーする場合のコマンドです。

MM

これがブロックを移動する場合のコマンドです。

- 3 ENTER キーを押します。

行コマンドが非表示になり、シーケンス番号が再度表示され、行ブロックがマークされています。

- 4 マークされた行ブロックを下または上の行に配置する行の先頭に、次のいずれかの行コマンドを入力します。

A

これがブロックを指定した行の下にコピーまたは移動する場合のコマンドです

P

これがブロックを指定した行の上にコピーまたは移動する場合のコマンドです

- 5 A または P を実行できるのは、少なくとも 1 つのフィールドが埋められている行のみであることに注意してください。
- 6 ENTER キーを押します。

行ブロックが指定された行の下または上にコピーされるか、または移動します。

### ダイレクトコマンドと PF キー

[Service Directory] 画面では、次のダイレクトコマンドと PF キーを使用できます。

ダイレクトコマンド	PF キー	機能
EXPIRATION		<p>リモートディレクトリデータはランタイム時にロードされます。このデータが有効とされる期間は設定された有効時間（秒単位）によって決定します。有効時間の設定後にディレクトリデータが要求された場合、データは自動的に再ロードされます。有効時間が 0 に設定されていると、リモートディレクトリデータは再ロードされません。</p> <p>ダイレクトコマンド EXPIRATION では、例えば、「EXPIRATION 86400」のように、秒単位の有効時間を入力できます。最大値は 8 桁の数です。</p> <p>コマンドとともにパラメータを入力しない場合、[Expiration Time] ウィンドウが表示され、現在の時刻を表示または変更できます。</p>
RESET		<p>「行コマンド」で説明したように、行コマンドで設定した行マークを削除します。</p> <p>行が正しくマークされていない場合、該当するメッセージが表示され、RESET を入力する前に、誤った行コマンドを削除する必要があることに注意してください。</p>
	PF1	エディタオンラインヘルプを呼び出します。
	PF2	論理ノード名または論理サービスを定義できる [Location Transparency] ウィンドウを呼び出します「 <a href="#">論理ノード名および論理サービスの定義</a> 」を参照）。
	PF3	終了します。変更内容を保存し、[Service Directory] 画面を終了するように促されます。
FIND	PF4	<p>名前を検索できる [Find Item] ウィンドウを呼び出します。</p> <p><b>Find what</b>                    32 文字以内で検索する英数字の文字列を入力します。                      大文字／小文字の区別 大文字と小文字を区別するには、デフォルト設定の N (No) を Y (Yes) で置き換えます。</p>

ダイレクト コマンド	PF キー	機能
		<p><b>Whole words only</b> 完全に一致する検索文字列のみを検索するには、デフォルト設定の N (No) を Y (Yes) で置き換えます。</p> <p>Enter キーを押して検索を開始し、一致から次の一致（ある場合）へ移動します。最初から検索を再開するには、PF4 キーを押します。</p> <p>一致はカーソルでマークされます。</p>
-H	PF5	半ページ後／前へスクロールします。
+H	PF6	
-P	PF7	1 ページ後／前にスクロールします。
+P	PF8	
TOP	PF9	リストの先頭までスクロールします。
BOT	PF10	リストの末尾までスクロールします。
	PF11	<b>[Service Directory]</b> 画面の標準ビュー（「例1 - <b>[Service Directory]</b> の標準ビュー」を参照）と <b>[Node]</b> フィールドおよび <b>[Server]</b> フィールドの拡張ビュー（「例2 - <b>[Service Directory]</b> の拡張ノード／サーバービュー」を参照）を切り替えます。
>	PF11	<b>[Node]</b> フィールドと <b>[Server]</b> フィールドの拡張ビューを表示します。拡張ノード／サーバービューには、 <b>[Library]</b> フィールドおよび <b>[Program]</b> フィールドは表示されません（「例2 - <b>[Service Directory]</b> の拡張ノード／サーバービュー」を参照）。
<	PF11	<b>[Service Directory]</b> 画面の標準ビューを表示します（「例1 - <b>[Service Directory]</b> の標準ビュー」を参照）。
CANCEL	PF12	変更内容を保存せずに <b>[Service Directory]</b> 画面を終了します。

## 論理ノード名および論理サービスの定義

論理ノード名または論理サービスは、すでにいずれかの値を含むノードフィールドまたはサーバーフィールドにのみ定義できます。

論理サービスを定義すると、元の（論理）ノード名は \*LOCTRAN で置き換えられ、論理ノード名または論理サービスを自動的に元に戻すことはできないことに注意してください。論理名およびサービスを削除する方法については、「[論理ノード名または論理サービスを削除するには](#)」を参照してください。

### ▶手順 105.3. 論理サービスを定義するには

- 1 カーソルを **[Server]** フィールドに置き、PF2 キー (LocTr) を押します。

[**Location Transparency - Logical Service**] ウィンドウが表示されます。

- 必要に応じて既存の値を変更し、Enter キーを押します。

追加のウィンドウとして [**Server Type Conversion**] ウィンドウが表示されます。次のいずれかを選択します。

- 「Y」 (Yes) を入力し、Enter キーを押して、変換を確認し、実行します。

指定したサーバーに関連する [**Node**] フィールドの値が、文字列 \*LOCTRAN で置き換えられます。この文字列は、ノード/サーバーの組み合わせが論理サービスに変換されたことを示します。

- 変換をキャンセルする場合は、「Y」以外のいずれかの文字を入力するか、または値を入力しません。

物理ノード名とサーバー名は保持されます。

### ▶手順 105.4. 論理ノード名を定義するには

- カーソルを [**Node**] フィールドに置き、PF2 キー (LocTr) を押します。

[**Location Transparency - Logical Node Name**] ウィンドウが開き、プリセット値の LOGBROKER= が表示されます。 *namename* は、論理 EntireX Broker 名を示します。

必要に応じて、*name* を変更しますが、文字列 LOGBROKER= は変更しないでください。

- Enter キーを押して変換を確認し、実行します。

物理ノード名が論理名に変換されます。

または:

PF12 キー (Canc) を押して変換をキャンセルします。

物理ノード名は保持されます。

### ▶手順 105.5. 論理ノード名または論理サービスを削除するには

- 論理ノード名：  
[**Node**] フィールドで、文字列 LOGBROKER= を削除します。

論理サービス：

論理サービスを削除し、**行コマンド** D および I を使用して物理サーバーを挿入します（「**行コマンド**」を参照）。



# 106 スタブ生成

---


スタブサブプログラムは、クライアントの呼び出し元プログラムをサーバーのサブプログラムに接続するために使用する Natural サブプログラムです。

次に示す 1 つの重要な例外を除き、Natural RPC（リモートプロシージャコール）自動実行が使用される場合、クライアントスタブサブプログラムは実際には必要ありません。ただし、『*Natural* リモートプロシージャコール (RPC)』ドキュメントの「*Natural* RPC 環境の運用」セクションの「スタブおよび RPC 自動実行」で説明しているように、スタブサブプログラムを生成することが有効な場合があります。

**注意：EntireX RPC サーバー**

EntireX RPC サーバーで呼び出すサブプログラムの IDL (Interface Definition Language) 定義にグループ構造が含まれている場合、スタブサブプログラムが必要です。この場合、このセクションで説明する適切な SYSRPC スタブ生成機能を使用して、同じグループ構造をスタブサブプログラムに定義する必要があります。

スタブサブプログラムは、新しいパラメータ定義、あるいはサブプログラムまたはパラメータデータエリア (PDA) の既存の定義から生成できます。

 **注意:** スタブサブプログラムの生成に使用したサブプログラムまたは PDA は、クライアント側のローカル環境で参照できなくなります。スタブ生成機能ではサブプログラムのソースが完全に変更されるため、サブプログラムはローカルプログラムコールに使用できなくなります。

次のセクションでは、単一または複数のスタブサブプログラムを生成する場合に使用する機能とコマンドについて説明します。

- [パラメータ指定による単一スタブの生成](#)
- [複数スタブの生成](#)



# 107      パラメータ指定による単一スタブの生成

---

■ Stub Generation 機能の使用 .....	742
■ パラメータの指定 .....	745
■ スタブ生成の例 .....	747


**Stub Generation**機能には、必要なパラメータ定義を入力するか、必要なパラメータ定義を既存のサブプログラムまたはパラメータデータエリア (PDA) から読み込むスタブ生成の画面を使用して、単一スタブサブプログラムをオンラインで生成するオプションがあります。

スタブサブプログラムを PDA から生成する場合、スタブサブプログラムを生成する前にサブプログラムを作成し、内部パラメータデータエリアを定義する手間を省くことができます。

## Stub Generation 機能の使用

---

スタブサブプログラムは、現在のシステムファイルの現在の Natural ライブラリに作成されます。このため、リモート CALLNAT の実行時にクライアントで使用されるアプリケーションライブラリまたはその steplib のいずれかにログオンすることを強くお勧めします。

 **重要:** スタブ生成機能では、ソースワークエリアに含まれるすべてのデータが上書きされます。スタブ生成機能呼び出す場合、該当するメッセージで、不用意に既存のデータを削除しないよう警告されます。PF12 キーを押して機能をキャンセルするか、Enter キーを押して操作を確認し、ソースワークエリアの内容を上書きします。

### ▶手順 107.1. 単一スタブサブプログラムを生成するには

- 1 SYSRPC ユーティリティを呼び出す前に、次の点を考慮してください。
  - スタブサブプログラムを生成するライブラリにログオンします。
  - PDA からスタブサブプログラムを生成するときは、スタブサブプログラムの生成元の PDA を参照するオブジェクトがクライアント側にある場合、その PDA の名前を変更するか、またはコピーします。PDA の新しい名前は、リモート CALLNAT プログラムの名前と一致する必要があります。

- 2 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

```
SG
```

- 3 ENTER キーを押します。

[Generate Client Stub Routine] ウィンドウが表示されます。

- 4 [Program Name] フィールドに、生成するスタブサブプログラムの名前を入力します。

スタブサブプログラムの名前は、リモート CALLNAT プログラムの名前と一致する必要があります。

[Library] フィールドは現在のライブラリの名前で事前設定されており、変更できません。

【**Compression**】フィールドに、圧縮タイプ0、1、または2（デフォルトは1）を入力します。『*Natural* リモートプロシージャコール (RPC)』ドキュメントの「*Natural* RPC 環境の運用」にある「圧縮の使用」の記載を参照してください。

5 ENTER キーを押します。

- 【**Program Name**】フィールドに入力した名前が、割り当てられたライブラリにすでに存在するオブジェクトの名前に一致した場合、ウィンドウが開いて該当するメッセージが表示されます。

操作をキャンセルする場合は、「N」（No）を入力して Enter キーを押します。【**Client Maintenance**】メニューに戻ります。

または：

スタブの生成を続行する場合は、「Y」（はい）を入力して ENTER キーを押します。

指定した名前が、サブプログラムタイプのカタログ化オブジェクトまたは PDA と同一の場合、それぞれのサブプログラムのパラメータ定義または PDA が【**Stub Generation**】画面に表示されます。

指定した名前が、ソースオブジェクトも存在するスタブサブプログラムと同一の場合、以前のスタブ生成の全フィールド属性（「[パラメータの指定](#)」も参照）が継承されます。これに該当しない場合、すべてのフィールド属性は M（変更可能）に設定されます。

- 【**Program Name**】フィールドに入力した名前が、割り当てられたライブラリにすでに存在するオブジェクトの名前に一致しない場合、空の【**Stub Generation**】画面が表示されます。

6 【**Stub Generation**】画面において、スタブサブプログラムで使用するパラメータの追加または変更を行います。詳細については、「[パラメータの指定](#)」を参照してください。

【**Stub Generation**】画面で使用できるコマンドは、「*Service Directory Maintenance*」セクションの[コマンドと PF キーに関する項目](#)で説明されているコマンドに対応します。

例外：

ダイレクトコマンド	機能
EXPIRATION	スタブ生成には適用できません。
LIMIT 32000 または LIMIT 1GB または LIMIT (サイズなし)	サイズの上限を 32000 バイトまたは 1 GB に設定します。  サイズ表示のない LIMIT では、LIMIT 32000 または LIMIT 1GB で設定されたサイズ制限が削除されます。

7 ENTER キーを押してスタブサブプログラムを生成し、手順を終了します。スタブサブプログラムは割り当てられたライブラリに生成されます。

[**SYSRPC - Information**] ウィンドウが表示され、クライアントとサーバー間でデータを送受信するためにスタブサブプログラムで必要となるサイズが示されます。このサイズには、スタブサブプログラムで使用される内部RPC情報が含まれます。サイズがわかっていると、EntireX Broker 使用時の Broker 属性ファイルなど、ミドルウェア層を構成する際に役立ちます。

サンプルサブプログラム TESTS5（下記の「例1」を参照）からスタブサブプログラムを生成すると、[**SYSRPC - Information**] ウィンドウに次のメッセージが表示されます。

```
Stub TESTS5 is generated in library SAGTEST (99,49).
It requires:
    Send length: 2249 bytes
    Receive length: 2221 bytes
```

ダイナミックパラメータの X-array または X-group 配列を使用すると、このメッセージで示されるのは最小長の要件のみとなります。実際の長さの要件はプログラムの実行時に初めて決まりますが、呼び出しごとに長さが異なる可能性があります。[Send length] または [Receive length] の値が Entire Net-Work の上限である 32000 バイトを超えている場合、ウィンドウが開いて警告が表示されます。

「Y」（はい）を入力して続行するか、「N」（いいえ）を入力して生成をキャンセルします。「Y」を入力した場合、この設定は SYSRPC セッション全体で有効になります。つまり、これ以降は警告が表示されることなく、スタブサブプログラムの生成を続けることができます。

送受信されるデータの合計（内部 RPC 情報を含まない）が 1073739357 バイト（1 GB から内部 RPC 情報の 2467 バイトを引いた数値）を超えた場合、SYSRPC は処理を停止してエラーメッセージを発行します。このエラーメッセージには、小計計算の基準になるフィールドで転送できたデータの量が、小計としてバイト単位で表示されます。次に、対応するフィールドがマークされます。この場合は、データの量を減らしてスタブサブプログラムの生成を続行します。

スタブサブプログラムが Natural システムライブラリ SYSRPC で生成された場合、Natural 転送ユーティリティの SYSMAN またはオブジェクトハンドラを使用して、スタブサブプログラムをアプリケーションライブラリまたは steplib に移動する必要があります。状況によっては、ターゲット環境でスタブサブプログラムのソースを再カタログする必要がありますので注意してください。

## パラメータの指定

【Stub Generation】画面に表示される入力フィールドに、スタブサブプログラムで使用されるパラメータ定義を入力できます。最大5000パラメータ指定できます。次の表に示されていない場合、フィールドへの入力必須です。

フィールド	説明
Level	<p>フィールドのレベル。</p> <p>レベルには、01（最高レベル）～99（最低レベル）の範囲の数を指定できます。先頭の0は任意です。</p> <p>グループ定義の例については、「<a href="#">グループの定義</a>」および「<a href="#">例2</a>」を参照してください。</p>
Attr	<p>パラメータを指定する属性。</p> <p>M（可変フィールド）、O（出力フィールド - OUT）、または I（入力フィールド）。</p> <p>2以上のレベル番号を割り当てられたパラメータは、グループの一部であるとみなされます。グループ内のパラメータは、1レベル上を割り当てられた直前のグループと同じ属性である必要があります。ネスト構造のグループでは、この属性は最高レベルのグループの属性です。グループ定義の例については、「<a href="#">例2</a>」を参照してください。</p> <p>スタブサブプログラムがサブプログラムまたは PDA から生成された場合、属性はデフォルトの M で、変更が必要な場合があります。</p> <p>スタブサブプログラムが別のスタブサブプログラムから生成された場合、元のスタブサブプログラムに指定された属性値が維持されます。</p> <p>生成されたスタブサブプログラムには、パラメータに指定された属性の IN、OUT、または INOUT を示すコメントが含まれます。</p>
Type	<p>N（数値）と G（グループ）、または K（漢字）などの Natural データフォーマット。Natural データフォーマット C（属性制御）および Handle は使用できません。</p> <p>Natural データフォーマットの詳細については、『プログラミングガイド』の「ユーザー定義変数」セクションの「ユーザー定義変数のフォーマットおよび長さ」および「特別なフォーマット」を参照してください。</p>
Length	<p>パラメータまたは DYNAMIC の長さ。</p> <p>このフィールドは、D（日付）、G（グループ）、L（論理）、および T（時間）の Natural データフォーマットには適用されません。</p> <p>Natural データフォーマット A は 1073739357 バイトに制限されており、Natural データフォーマット B は 536869678 バイトに制限されています。</p> <p>DYNAMIC はダイナミックパラメータを示し、Natural データフォーマット A および B に適用されます。</p>
Prec	<p>Natural データフォーマット N（数値）および P（パック）のみに適用されます。任意です。</p>

フィールド	説明
	パラメータの精度、つまり、小数点の後の桁数。
Dimension 1/2/3	配列のみに適用されます。任意です。 パラメータの1次元、2次元、および3次元。 X-array または X-Group 配列は、次元にアスタリスク (*) を入力して指定します。 「 <a href="#">X-array および X-Group 配列の定義</a> 」も参照してください。

以降のセクションでは、次の項目について説明します。

- [グループの定義](#)
- [X-array および X-Group 配列の定義](#)

## グループの定義

EntireX RPC サーバー上の Natural 以外のオブジェクトを呼び出すクライアント Natural オブジェクトのグループ構造のみを定義する必要があります。グループ構造は、EntireX の IDL 定義に対応している必要があります。Natural RPC サーバー上のサブプログラムを呼び出すクライアント Natural オブジェクトには、グループ構造は必要ありません。


クライアント Natural オブジェクトからスタブサブプログラムに渡されるグループ配列および X-Group 配列は連続している必要があります。このため、常に全次元用のアスタリスク (\*) 表記を使用して、スタブサブプログラムに完全な配列を渡すことを強くお勧めします。また、クライアント Natural プログラム、スタブサブプログラム、およびサーバープログラムに同一のデータ定義を使用することを強くお勧めします。

グループ定義の例については、「[例2](#)」も参照してください。

## X-array および X-Group 配列の定義

パラメータのいずれかの次元が拡張可能な場合、そのパラメータの他のすべての次元も拡張可能です。サブプログラムでパラメータに拡張可能な固定次元を定義すると、スタブ生成機能によって警告が発行され、自動的に固定次元が拡張可能な次元に変更されます（「[例3](#)」を参照）。グループ構造では、レベルごとに拡張可能な次元または固定次元のいずれかを定義できます。レベル間で固定次元が拡張可能な次元に自動的に変更されることはありません。

Natural RPC では、拡張可能な上限のみがサポートされます。このため、スタブサブプログラムの生成された DEFINE DATA PARAMETER エリアのすべての X-array と X-Group 配列は、(1:\*) として定義されます。

 **注意:** 拡張可能な下限を持つ X-array または X-group array を含むサブプログラムまたは PDA からスタブサブプログラムを生成した場合、拡張可能な下限は拡張可能な上限に変換されます。

拡張可能な次元を持つグループの詳細については、「[例3](#)」を参照してください。



## スタブ生成の例

このセクションでは、Naturalサブプログラムおよびそれらのサブプログラムから生成されるスタブサブプログラムの例を示します。

次に示すパラメータ定義は、Naturalシステムライブラリ SYSRPC で提供されるサンプルサブプログラムから抽出されています。

### 例 1

次の DEFINE DATA PARAMETER エリア（サンプルサブプログラム TESTS5）は、**[Stub Generation]** 画面の 4 つの変更可能なパラメータおよび対応するパラメータ定義を示しています。

```
DEFINE DATA
PARAMETER
  01 #IDENTIFIER (A10)
  01 #N-OF-ID (I4)
  01 #FREQ (P5.2)
  01 #A100 (A100/5,4)
```

スタブ生成								
	Level	Attr	Type	Length	Prec	次元 1	次元 2	次元 3
1	01	M	A	10				
2	01	M	I	4				
3	01	M	P	5	2			
4	01	M	A	100		5	4	

### 例 2

次の DEFINE DATA PARAMETER エリア（サンプルサブプログラム TESTS6）は、**[Stub Generation]** 画面のネストされたグループ構造および対応するパラメータ定義を示しています。

```
DEFINE DATA
PARAMETER
  01 GROUP-1(10)
    02 A (A20)
    02 B (A20)
  02 GROUP-2(20)
    03 C (A10/5)
    03 D (A10)
  01 LINE (A) DYNAMIC
```

## パラメータ指定による単一スタブの生成

スタブ生成								
	Level	Attr	Type	Length	Prec	次元 1	次元 2	次元 3
1	01	M	G			10		
2	02	M	A	20				
3	02	M	A	20				
4	02	M	G			20		
5	03	M	A	10		5		
6	03	M	A	10				
7	01	M	A	DYNAMIC				

### 例 3

次の DEFINE DATA PARAMETER エリア（サンプルサブプログラム TESTS7）は、**[Stub Generation]** 画面の拡張可能な次元があるネストされたグループ構造および対応するパラメータ定義を示しています。

```
DEFINE DATA
PARAMETER
  01 GROUP-1(10)
    02 A (A20)
    02 B (A20)
    02 GROUP-2(0:*)
      03 C (A10/5)
      03 D (A10)
  01 LINE (A) DYNAMIC
```

スタブ生成								
	Level	Attr	Type	Length	Prec	次元 1	次元 2	次元 3
1	01	M	G			10		
2	02	M	A	20				
3	02	M	A	20				
4	02	M	G			*		
5	03	M	A	10		5		
6	03	M	A	10				
7	01	M	A	DYNAMIC				

# 108 複数スタブの生成

---

- SYSRPC SGMASS コマンドの使用 ..... 750
- 名前の指定と圧縮 ..... 751

コマンド `SYSRPC SGMASS` を使用して、単一または複数のスタブサブプログラムをオンラインモードまたはバッチモードで生成できます。

スタブサブプログラムは、サブプログラムまたはパラメータデータエリア (PDA) から作成します。

## SYSRPC SGMASS コマンドの使用

---

スタブサブプログラムをオンラインで生成するためのいずれかの `NEXT` または `MORE` コマンドプロンプトに、`SYSRPC SGMASS` コマンドを入力できます。

以降のセクションでは、次の項目について説明します。

- [SYSRPC SGMASS の構文](#)
- [SYSRPC SGMASS レポート](#)

### SYSRPC SGMASS の構文

次の構文図に、`SYSRPC SGMASS` に適用される構文を示します。

```
SYSRPC SGMASS [name] [compression]
```

構文項目の *name* および *compression* については、「[名前の指定と圧縮](#)」セクションで説明します。

### SYSRPC SGMASS レポート

次に示すように、`SYSRPC SGMASS` コマンドでは、コマンドによって生成されたスタブサブプログラム (オブジェクト) をリストするレポートが生成されます。

```
Page          1                               2006-05-24 16:09:17
```

```
SYSRPC - Stub Generation in Library SAGTEST
```

```
Generation Criteria:
```

```
Object name or range: RPC*  
Compression: 1
```

```
Generation Results:
```

```
Number of objects found:      8
```

```

Maximum send length: 200228
Maximum receive length: 1024192

```

Object	Type	Send Length	Receive Length	Message
RPCCALL1	N	209	202	
RPCCALL2	N	219	240	Compression=2
RPCCALL3	N	204	193	
MORE				

このレポートは3つのセクションで構成されており、それぞれに次の情報が表示されます。

■ **Generation Criteria:**

スタブサブプログラムが生成されたときに適用された条件。単一のオブジェクト名か名前の範囲（ここではRPC\*）と、圧縮タイプ（ここでは1）が表示されます。

■ **Generation Results:**

スタブ生成対象として選択されたオブジェクト数。

クライアントとの間でデータの送受信を行うために、生成されたすべてのスタブサブプログラムで必要となる最大バッファサイズ。

■ **Object List:**

生成された各スタブサブプログラムの名前およびタイプ（ここではタイプのN）。クライアントとの間でデータの送信（[Send Length]）や受信（[Receive Length]）を行うために各オブジェクトが必要とするバッファサイズ。[Message]列に、各スタブ生成に関する可能性があるコメント。上記の例では、Compression=2は、コマンドで要求されたようにCompression 1でオブジェクトRPCCALL2が生成されなかったことを示します。オブジェクトリストは、オブジェクト名のアルファベット順にソートされます。

MOREプロンプトが表示された場合は、ENTERキーを押してレポートの最後までスクロールします。

単一または複数のオブジェクトのスタブ生成が失敗した場合、影響を受けたオブジェクトの数および該当するエラーメッセージがレポートで示されます。

## 名前の指定と圧縮


スタブ生成用に選択するオブジェクト（サブプログラムまたはPDA）および使用する圧縮のタイプを指定できます。

- **名前**

### ■ 圧縮

#### 名前

オブジェクト名または名前の範囲を指定できます。オブジェクトの名前または名前の範囲の指定は任意です。

 **注意:** オブジェクトの名前または名前の範囲を指定しない場合、まれに例外はありますが（下記参照）、現在のライブラリのすべてのサブプログラムまたは PDA がスタブサブプログラムに変換されます。

以下は、有効な名前の指定について説明したものです。ここで *value* は、任意の組み合わせによる 1 文字以上の英数字を表します。

入力	選択されるオブジェクト
*	すべてのサブプログラムまたは PDA。 これはデフォルト設定です。
<i>value</i>	<i>value</i> と同じ名前を持つサブプログラムまたは PDA。
<i>value</i> *	<i>value</i> で始まる名前を持つすべてのサブプログラムまたは PDA。
<i>value</i>	<i>value</i> 以前の名前を持つすべてのサブプログラムまたは PDA。
<i>value</i>	<i>value</i> 以降の名前を持つすべてのサブプログラムまたは PDA。

#### 名前の例外

Natural システムライブラリ SYSRPC では、SYSRPC SGMASS は RDS、RPC、NAT、NAD、または NSC のいずれかの接頭辞で始まる名前を持つすべてのサブプログラムをスタブ生成から除外します。

ユーザーライブラリでは、SYSRPC SGMASS はサブプログラム NATCLTGS をスタブ生成から除外します。

#### 圧縮

0、1、2 のいずれかの圧縮タイプを選択できます。圧縮の指定は任意です。スタブ生成に使用するデフォルトタイプは 1 です。

『Natural リモートプロシージャコール (RPC) 』ドキュメントの「Natural RPC 環境の運用」にある「圧縮の使用」の記載も参照してください。

# 109 サイズ要件の計算

---

■ SYSRPC CSMASS コマンドの使用 .....	754
■ 名前の指定と圧縮 .....	756

SYSRPC CSMASS コマンドは、クライアントからサーバーへ（またはその逆へ）データを送信するためにスタブなしのRPCコールに必要なバッファサイズを計算する場合に使用します。サイズがわかっていると、EntireX Broker 使用時の Broker 属性ファイルなど、ミドルウェア層を構成する際に役立ちます。

スタブサブプログラムの生成時にすでにサイズが計算されている場合でも、必要があれば、スタブサブプログラムのサイズ計算を実行できます。

SYSRPC CSMASS は、オンラインモードでもバッチモードでも使用できます。

## SYSRPC CSMASS コマンドの使用

---

サイズ要件をオンラインで計算するためのいずれかの NEXT または MORE コマンドプロンプトに、SYSRPC CSMASS コマンドを入力できます。

以降のセクションでは、次の項目について説明します。

- [SYSRPC CSMASS の構文](#)
- [SYSRPC CSMASS レポート](#)

### SYSRPC CSMASS の構文

次の構文図に、SYSRPC CSMASS コマンドに適用される構文を示します。

```
SYSRPC CSMASS [name] [compression]
```

構文項目の *name* および *compression* については、「[名前の指定と圧縮](#)」セクションで説明します。

### SYSRPC CSMASS レポート

次に示すように、SYSRPC CSMASS コマンドでは、指定したサブプログラム（オブジェクト）の送信長／受信長の要件を示すレポートが生成されます。

```
Page          1                               2006-05-24 15:54:12
```

```
SYSRPC - Calculation of Buffer Sizes for Stubless RPC in Library SAGTEST
```

```
Calculation Criteria:
```

```
Object name or range: RPC*  
Compression: 1
```



## Calculation Results:

```

Number of objects found:      8
  Maximum send length: 200228
  Maximum receive length: 1024192

```

Object	Type	Send Length	Receive Length	Message
RPCCALL1	N	209	202	
RPCCALL2	N	219	240	Compression=2
RPCCALL3	N	204	193	
MORE				

このレポートは3つのセクションで構成されており、それぞれに次の情報が表示されます。

■ **Calculation Criteria:**

計算を実行したときに適用された条件。単一のオブジェクト名か名前の範囲（ここではRPC\*）と、圧縮タイプ（ここでは1）が表示されます。

■ **Calculation Results:**

サイズ計算対象として選択されたオブジェクト数。

クライアントとの間でデータの送受信を行うために、選択されたすべてのオブジェクトで必要となる最大バッファサイズ。

■ **Object List:**

計算対象として選択された各オブジェクトの名前およびタイプ（ここではサブプログラムタイプのN）。クライアントとの間でデータの送信（[Send Length]）や受信（[Receive Length]）を行うために各オブジェクトが必要とするバッファサイズ。[Message]列に、各オブジェクト計算に関する可能性があるコメント。上記の例では、Compression=2は、コマンドで要求されたようにCompression 1でオブジェクトRPCCALL2が計算されなかったことを示します。オブジェクトリストは、オブジェクト名のアルファベット順にソートされます。

MOREプロンプトが表示された場合は、ENTERキーを押してレポートの最後までスクロールします。

単一または複数のオブジェクトのサイズ計算が失敗した場合、影響を受けたオブジェクトの数および該当するエラーメッセージがレポートで示されます。

## 名前の指定と圧縮

サイズ計算用に選択するオブジェクト（サブプログラムまたは PDA）および使用する圧縮のタイプを指定できます。

- [名前](#)
- [圧縮](#)

### 名前

オブジェクト名または名前の範囲を指定できます。名前または名前の範囲を指定しない場合、現在のライブラリに含まれるすべてのサブプログラムのサイズまたは PDA が計算されます。

以下は、有効な名前の指定について説明したものです。ここで *value* は、任意の組み合わせによる 1 文字以上の英数字を表します。

入力	選択されるオブジェクト
*	すべてのサブプログラムまたは PDA。 これはデフォルト設定です。
<i>value</i>	<i>value</i> と同じ名前を持つサブプログラムまたは PDA。
<i>value</i> *	<i>value</i> で始まる名前を持つすべてのサブプログラムまたは PDA。
<i>value</i> <	<i>value</i> 以前の名前を持つすべてのサブプログラムまたは PDA。
<i>value</i> >	<i>value</i> 以降の名前を持つすべてのサブプログラムまたは PDA。

### 圧縮

0、1、2 のいずれかの圧縮タイプを選択できます。圧縮の指定は任意です。スタブ生成に使用するデフォルトタイプは 1 です。

『*Natural* リモートプロシージャコール (RPC) 』ドキュメントの「*Natural* RPC 環境の運用」にある「*圧縮の使用*」の記載も参照してください。


# 110      パラメータメンテナンス

---

- Parameter Maintenance の呼び出し ..... 758
- 「NTRPC/RPC キーワードサブパラメータの指定」を参照してください。 ..... 758

クライアントセッションのみに適用されます。

パラメータメンテナンス機能は、RPC プロファイルパラメータまたは NTRPC マクロのキーワードサブパラメータをセッション内でダイナミックに変更するために使用します。

 **注意:** パラメータ変更はユーザーセッションがアクティブである場合にのみ保持されません。セッションが終了すると削除されます。

## Parameter Maintenance の呼び出し

---

### ▶手順 110.1. Parameter Maintenance 機能呼び出しには

- 1 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

PM

[Client Parameter Maintenance] 画面が表示されます。

- 2 入力フィールドの値を変更します。  
「[NTRPC/RPC キーワードサブパラメータの指定](#)」を参照してください。
- 3 すべての変更を保存して [Client Parameter Maintenance] 画面を終了するには、PF3 キー (Exit) を押します。

または:

パラメータ変更を保存せずにウィンドウを終了するには、PF12 キー (Canc) を押します。

[Client Maintenance] メニューが表示されます。

## 「NTRPC/RPC キーワードサブパラメータの指定」を参照してください。

---

[Client Parameter Maintenance] 画面に表示される入力フィールドで、次の表に示す NTRPC マクロまたは RPC プロファイルパラメータのキーワードサブパラメータの設定を変更できます。

フィールド	説明
<b>Timeout</b>	<p>クライアントが RPC サーバーの応答を待機する秒数を指定します。</p> <p>『パラメータリファレンス』ドキュメントで説明しているNTRPC/RPC キーワードサブパラメータ TIMEOUT も参照してください。</p>
<b>Try alternative servers</b>	<p>RPC クライアントが代替サーバー上でサービスの実行を試行する (ON) か、またはしない (OFF) かを指定します。『Natural リモートプロシージャコール (RPC)』ドキュメントの「代替サーバーの使用」も参照してください。</p> <p>『パラメータリファレンス』ドキュメントで説明しているNTRPC/RPC キーワードサブパラメータ TRYALT も参照してください。</p>
<b>Compression for auto remote RPC</b>	<p>自動的に生成された RPC コールの圧縮タイプを指定します (『Natural プロシージャコール (RPC)』ドキュメントの「圧縮の使用」を参照)。</p> <p>『パラメータリファレンス』ドキュメントで説明しているNTRPC/RPC キーワードサブパラメータ COMPR も参照してください。</p> <p>RPC 自動実行の詳細については、『Natural リモートプロシージャコール (RPC)』ドキュメントの「Natural RPC 自動実行の操作」を参照してください。</p>

パラメータ設定の詳細については、『パラメータリファレンス』ドキュメントの「RPC- リモートプロシージャコール設定」セクションの「キーワードサブパラメータ」を参照してください。



# 111      サーバーコマンドの実行

---

■ Server Command Execution の使用 .....	762
■ サーバーに対する ping の実行 .....	763
■ サーバーの終了 .....	765

SYSRPC ユーティリティは、サーバー実行コマンド ping および terminate を提供します。これらのコマンドは、サービスディレクトリに定義されたアクティブなサーバーを制御するために使用します。ping コマンドでは、サーバー接続を確認するためにサーバーに内部メッセージを送信します。terminate では、単一のサーバータスクの終了を要求する内部メッセージをサーバーに送信するか、または EntireX Broker サービスに関連付けられたすべてのサーバータスクの終了を要求するコマンドを EntireX Broker サービスに発行します。

サーバー実行コマンドは、NTRPC/RPC キーワードサブパラメータ RPCSDIR で定義されているライブラリ内のサービスディレクトリを参照します（『パラメータリファレンス』ドキュメントを参照）。RPCSDIR が設定されていない場合（デフォルト）、現在ログオンしているライブラリが使用されます。ライブラリの名前は、次のセクションに示す [Server Command Execution] 画面の右上隅に表示されます。

## Server Command Execution の使用

### ▶手順 111.1. Server Command Execution を使用するには

- 1 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

```
XC
```

- 2 ENTER キーを押します。

次のような [Server Command Execution] 画面の標準ビューが表示されます。

```
14:09:38          *** NATURAL Remote Procedure Call ***          2006-01-27
                               Server Command Execution                               SYSRPC

Node                Server                Message
1  ETB045
2                    NRPC001                Natural
3                    NRPC002
4  *LOCTRAN
5                    NRPC001-LOGICAL
6  LOGBROKER=NODE
7                    NRPC002

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help ERR  Exit <   -H   +H   -P   +P   TOP  BOT  >   Canc
```



標準ビューには、**[Node]**、**[Server]**、および**[Message]**の各列が示されます。  
**[Message]**列の下のフィールドは一部が表示され、最大8文字が示されます。

- PF4キーを押すか、または画面下部のコマンド行に小なり (<) 記号を入力すると、次の例のような**[Server Command Execution]**画面の拡張メッセージビューが表示されます。

```

14:14:08          *** NATURAL Remote Procedure Call ***          2006-01-27
                   Server Command Execution                      SYSRPC

      Server      Message
1
2  NRPC001      Natural RPC Server 6.3.1.0 on WNT-x86
3  NRPC002
4
5  NRPC001-LOGICAL
6
7  NRPC002

-----
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  ERR   Exit  <    -H   +H   -P   +P   TOP  BOT   >   Canc
  
```

拡張ビューでは、**[Message]**列の最大50文字のメッセージテキストを確認できます。このビューでは、**[Node]**列は表示されず、**[Server]**列の下のフィールドは一部が表示され、最大16文字が示されます。標準ビューでは30文字が表示されます。

PF11キーを押すか、またはコマンド行に大なり (>) 記号を入力すると、「標準ビューの例」に示すような**[Server Command Execution]**画面の標準ビューが表示されます。

## サーバーに対する ping の実行

**[Server Command Execution]**画面の標準または拡張メッセージビューから、サーバーに ping を実行できます。

次に、標準ビューからサーバーに ping を実行する手順の例を示します。

RPCサーバーに ping を実行する別の方法については、『Natural リモートプロシージャコール (RPC)』ドキュメントで説明しているアプリケーションプログラミングインターフェイス「USR2073N」を参照してください。

▶手順 111.2. サーバーに対して ping を実行するには

- 1 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

```
XC
```

- 2 ENTER キーを押します。

[Server Command Execution] 画面の標準ビューが表示されます。

- 3 シーケンス番号と [Node] 列の間にある空の列で、ping を実行するサーバーに属する行に次のコマンドを入力します。

```
PI
```

次に例を示します。

```
09:49:41          *** NATURAL Remote Procedure Call ***          2006-01-27
                  Server Command Execution                      SYSRPC

      Node                Server                Message
1      ETB045
2      PI                  NRPC001
3                        NRPC002
4      *LOCTRAN
5      PI                  NRPC001-LOGICAL
6      LOGBROKER=NODE
7                        NRPC002
```

- 4 ENTER キーを押します。サーバーから次のメッセージが返されます。

```
ServerV.R.S.PL on operating system
```

ここでは次の内容を表示しています。

Server は、サーバーのタイプを示します。V.R.S.PL は、サーバーの 1 桁のバージョン V、1 桁のリリース R、1 桁のシステムメンテナンスレベル S、および 1~2 桁のサーバーのパッチレベル PL を示します。operating system は、サーバーが実行されているオペレーティングシステムを示します。

サンプルメッセージ：

Natural RPC Server 6.3.1.0 on WNT-x86

サーバーに対する ping が失敗してエラーが発生した場合は、PF2 キー (ERR) を押して RPC 関連の Natural および EntireX Broker メッセージを表示できます (「RPCERR プログラムの使用」 (「RPC セッションのステータスのモニタ」、『Natural リモートプロシージャコール (RPC)』ドキュメント) を参照)。

- 5 [Server Command Execution] 画面の標準ビュー (「標準ビューの例」を参照) に一部が表示されるメッセージテキストをさらに表示するには、次の手順を実行します。

PF4 キーを押します。

または:

コマンド行に、小なり (<) 記号を入力します。

## サーバーの終了

SYSRPC ユーティリティでは、サーバーを終了するために TE (サーバーの終了) と TS (EntireX Broker サービスの終了) の 2 つのコマンドを使用できます。

TE では、サーバーに内部メッセージを送信して、単一のサーバータスクを終了します。サーバーが、メインフレームプラットフォーム上のレプリカを含む複数のサーバータスクに関連付けられている場合、TE コマンドを使用して各サーバータスクを個別に終了するか、または TS コマンドを使用してすべてのサーバータスクを一度に終了することができます。

TS では、EntireX Broker's Command and Information Services (ETBCIS。詳細については、EntireX のドキュメントを参照) を呼び出して、EntireX Broker サービスに関連付けられたすべてのサーバータスクを終了します。ここで使用する サービス という用語は、同一のプラットフォーム上または異なるプラットフォーム上の同じサーバー名で実行されるすべてのサーバータスクを表します。

[Server Command Execution] 画面の標準または拡張メッセージビューから、サーバータスクを終了できます。

次のセクションでは、標準ビューから単一のサーバータスクまたは EntireX Broker サービスを終了する手順を示します。

サーバーを停止する別の方法については、『Natural リモートプロシージャコール (RPC)』ドキュメントの「Natural RPC サーバーの終了」とアプリケーションプログラミングインターフェイス USR2073N に関する項、ならびに「EntireX Broker サービスの終了」とアプリケーションプログラミングインターフェイス USR2075N に関する項を参照してください。

### ▶手順 111.3. 単一のサーバータスクを終了するには

- 1 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

```
XC
```

[Server Command Execution] 画面の標準ビューが表示されます。

- 2 シーケンス番号と [Node] 列の間にある空の列で、終了するサーバーに属する行を見つけ、次のコマンドを入力します。

```
TE
```

(サーバーの ping の実行例で示した、コマンド PI の入力とほぼ同じです。)

- 3 ENTER キーを押します。

サーバーから次のメッセージが返されます。

```
Terminating ServerV.R.S.PL on operating system
```

ここでは次の内容を表示しています。

*Server* は、サーバーのタイプを示します。*V.R.S.PL* は、サーバーの 1 桁のバージョン *V*、1 桁のリリース *R*、1 桁のシステムメンテナンスレベル *S*、および 1~2 桁のサーバーのパッチレベル *PL* を示します。*operating system* は、サーバーが実行されているオペレーティングシステムを示します。

サンプルメッセージ：

```
Terminating Natural RPC Server 6.3.1.0 on WNT-x86
```

サーバーの終了が失敗してエラーが発生した場合は、PF2 キー (ERR) を押して RPC 関連の Natural および EntireX Broker メッセージを表示できます (「RPCERR プログラムの使用」(「RPC セッションのステータスのモニタ」、『Natural リモートプロシージャコール (RPC)』ドキュメント) を参照)。

[Server Command Execution] 画面の標準ビューに一部が表示されるメッセージテキストをさらに表示するには、次の手順を実行します。

PF4 キーを押します。

または

コマンド行に、小なり (<) 記号を入力します。

- 4 サービスディレクトリで [Logon] オプションが設定されている場合、リモート CALLNAT 実行の場合と同じように、TE コマンドによってログオンデータ (ユーザー ID、パスワード、およびライブラリ名) がサーバーに送信されます。クライアント側に Natural Security がインストールされておらず、アプリケーションプログラミングインターフェイス USR1071N

で現在の Natural セッションにログオンデータが設定されていない場合、ユーザー ID とパスワードの入力を要求する **[Security Token Data]** ウィンドウが表示されます。「セキュリティの使用」に記載されている「USR1071N」、『Natural リモートプロシージャコール (RPC)』ドキュメントの「Natural Security での Natural RPC の使用」も参照してください。

サーバー側で LOGONRQ=ON (『Natural リモートプロシージャコール (RPC)』ドキュメントの「セキュリティの使用」も参照) が設定されている場合、TE コマンドでログオンデータがクライアントから送信される必要があります。

Natural Security がサーバーにインストールされている場合、転送されるログオンデータによって Natural システムライブラリ SYSRPC へのログオンが有効になる必要があります。

#### ▶手順 111.4. EntireX Broker サービスを終了するには

- 1 **[Client Maintenance]** メニューの **[Code]** フィールドに、次のコマンドを入力します。

```
XC
```

**[Server Command Execution]** 画面の標準ビューが表示されます。

- 2 シーケンス番号と **[Node]** 列の間にある空の列で、終了するサーバーに属する行を見つけて次のコマンドを入力します。

```
TS
```

(サーバーの ping の実行例で示した、コマンド PI の入力とほぼ同じです。)

- 3 ENTER キーを押します。

**[SYSRPC - Terminating EntireX Broker Service]** ウィンドウが表示されます。

- 4 ログオンに必要な場合は、EntireX Broker の適切なユーザー ID とパスワードを入力します。

会話に関連するサーバータスクを終了する場合は、**[Terminate immediately]** フィールドに「Y」を入力して即時終了を要求します。「N」(デフォルト設定)を入力すると、会話に関連するすべてのサーバータスクは実行可能なままとなります。

現在の SYSRPC セッション中にこのウィンドウが繰り返し表示されないようにするには、**[Do not show this window again]** を選択します。

- 5 Enter キーを押して、EntireX Broker サービスを終了します。



# 112      リモートディレクトリメンテナンス

---

■ Remote Directory Maintenance の使用 .....	770
■ リモートディレクトリのエディタ画面のフィールド .....	771
■ Remote Directory Maintenance のコマンド .....	772

**Remote Directory Maintenance** 機能は、クライアントの呼び出し元プログラムをサーバーのサブプログラムに接続するためにリモートディレクトリをメンテナンスする場合に使用します。

**Remote Directory Maintenance** 機能を適用する方法の詳細については、『*Natural* リモートプロシージャコール (RPC) 』ドキュメントの「RPC サーバーアドレスの指定」（「*Natural* RPC 環境の運用」）、「リモートディレクトリサーバー (RDS) の使用」、および「*Natural* RPC の用語」を参照してください。

## Remote Directory Maintenance の使用

---



**注意:** コード「C」（次の説明を参照）を入力して新規リモートディレクトリを作成すると、既存のディレクトリのエントリは上書きされます。

### ▶手順 112.1. Remote Directory Maintenance 機能を使用するには

- 1 [Client Maintenance] メニューの [Code] フィールドに、次のコマンドを入力します。

RD

- 2 ENTER キーを押します。

ウィンドウが表示されます。

- 3 入力フィールドに、次のいずれかのコマンドを入力します。

C

これがディレクトリを作成する場合のコマンドです。

M

これがディレクトリを変更する場合のコマンドです。

- 4 ENTER キーを押します。

追加のウィンドウが表示されます。

- 5 秒単位の期限時間を入力し（次の「[期限時間](#)」も参照）、Enter キーを押します。



以下の例のようなエディタ画面が表示されます。

```


-----S 01-----Columns 001 072
====>          SCROLL====>  CSR
***** NODE      L T      SERVER      L T      LIBRARY      L T      PROGRAM      L T
***** ***** top of data *****
000001 ETB01      Y B      NRPC2301      SYSTEM      SUB1
000002                                     SUB2
000003                                     NRCP2301      SYSTEM      SUB3
000004                                     NRPC2302      SYSTEM      SUB4
000005                                     SUB5
000006                                     SUB6
000007 ETB01      NRPC2301  Y      SYSTEM      SUB7
000008                                     SUB1          Y
000009                                     SUB2
000010                                     NRCP2301      SYSTEM      SUB3
000011                                     NRPC2302      SYSTEM      SUB4
000012                                     SUB5
000013 ETB01      Y      NRPC2301      SYSTEM      SUB6
000014                                     SUB1          Y
000015                                     SUB2

000016 ETB01      Y      NRPC2301      SYSTEM      SUB3
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Quit  Save  Find  Chng  Up    Down          Canc

```

### 有効時間

リモートディレクトリデータはランタイム時にロードされます。このデータが有効とされる期間は設定された有効時間（秒単位）によって決定します。有効時間の設定後にディレクトリデータが要求された場合、データは自動的に再ロードされます。有効時間が0に設定されていると、リモートディレクトリデータは再ロードされません。

 **注意:** 新規ディレクトリを作成する場合は（コード C）、top of data 行の先頭に行コマンド I を入力して、入力行を呼び出します。次の「[行コマンド](#)」も参照してください。

## リモートディレクトリのエディタ画面のフィールド

**[Remote Directory Service]** メンテナンスのエディタ画面に含まれるフィールドは、「*Service Directory Maintenance*」セクションの「[\[Service Directory\] 画面のフィールド](#)」で説明したフィールドと同一です。[L] フィールドは、[Logon] フィールドに相当します。

**[Service Directory Maintenance]** 画面に表示されるフィールドに加えて、[T]（Transport）フィールドが表示されます。EntireX Broker の「B」を入力します。

## Remote Directory Maintenance のコマンド

このセクションでは、[Remote Directory Service] メンテナンスのエディタ画面で使用できるコマンドについて説明します。

- 行コマンド
- ダイレクトコマンドと PF キー

### 行コマンド

エディタ画面で使用できる行コマンドは、フィールド値を含む1行または複数の行をコピー、移動、または削除する場合に使用します。

行コマンドは行の先頭に入力します。つまり、シーケンス番号を上書きします。それから ENTER キーを押します。

「行ブロックをコピーまたは移動するには」およびダイレクトコマンド RESET の項も参照してください。

行コマンド	機能
A	C、CC、M、または MM でマークされた行を、コマンドが入力された行の下にコピーまたは移動します。
B	C、CC、M、または MM でマークされた行を、コマンドが入力された行の上にコピーまたは移動します。
C	コピーする 1 行をマークします。
CC	コピーする行ブロックをマークします。
D (n)	コマンドが入力された行で始まる 1 行または n 行を削除します。n には、1~9 の範囲で数値を指定できます。
DD	行ブロックをマークして削除します。  ブロックの最初の行と最後の行にコマンドを入力して、行ブロックをマークし、Enter キーを押してコマンドを実行します。
I (n)	コマンドが入力された行の下に 1 行または n 行の空行を挿入します。n には、1~9 の範囲で数値を指定できます。
M	コマンドが入力された行の下に 1 行移動します。
MM	移動する行ブロックをマークします。

#### ▶手順 112.2. 行のブロックをコピーまたは移動するには

- 1 ブロックが開始する行の先頭に、次のいずれかの行コマンドを入力します。

CC

これがブロックをコピーする場合のコマンドです。

MM

これがブロックを移動する場合のコマンドです。

- 2 ブロックが終了する行の先頭に、次のいずれかの行コマンドを入力します。

CC

これがブロックをコピーする場合のコマンドです。

MM

これがブロックを移動する場合のコマンドです。

- 3 ENTER キーを押します。

行ブロックがマークされ、そのことが「Block is pending」というメッセージで示されます。

- 4 ブロックを下または上の行に配置する行の先頭に、次のいずれかの行コマンドを入力します。

A

これがブロックを指定した行の*below*にコピーまたは移動する場合のコマンドです

B

これがブロックを指定した行の*above*にコピーまたは移動する場合のコマンドです

- 5 ENTER キーを押します。

行ブロックが指定された行の下または上にコピーされるか、または移動します。

## ダイレクトコマンドと PF キー

エディタ画面では、次のダイレクトコマンドと PF キーを使用できます。

ダイレクトコマンド	PF キー	機能
RESET		行コマンド（「 <a href="#">行コマンド</a> 」を参照）またはダイレクトコマンド CHANGE（下記参照）で設定された行マークを削除します。
TOP		リストの先頭までスクロールします。
BOT		リストの末尾までスクロールします。
FIND <i>string</i>		エディタをスキャンして文字 <i>string</i> を検索します。例：FIND ETB1  PF5 キー（Find）を押して次を検索します。

ダイレクトコマンド	PF キー	機能
CHANGE <i>string1</i> <i>string2</i>		文字 <i>string1</i> を <i>string2</i> で置き換えます。例：CHANGE ETB1 ETB2 PF6 キー (Chng) を押して次を置き換えます。
	PF1	Help。オンラインヘルプを呼び出します。
	PF3	Quit。すべての変更を保存し、エディタ画面を終了します。
	PF4	すべての変更を保存します。
	PF5	Find。ダイレクトコマンド FIND (上記参照) を使用して、指定された文字列をスキャンします。
	PF6	Change。ダイレクトコマンド CHANGE (上記参照) を使用して、指定された文字列を置き換えます。
	PF7	Up。1 ページ前にスクロールします。
	PF8	Down。1 ページ後にスクロールします。
	PF12	Cancel。変更を保存しないでエディタ画面を終了します。

# 113 SYSTP ユーティリティ

---

SYSTP ユーティリティは、TP モニタに固有の Natural の特性をモニタリングおよび制御するために使用します。

SYSTP ユーティリティは、ほとんどの TP モニタのもとで、ほとんどの環境で使用可能な機能を提供します。これらの機能については、「[一般的な SYSTP 機能](#)」で説明しています。

SYSTP ユーティリティでは、次の TP モニタに機能を追加します。

- CICS
- IMS/TM
- TIAM および *open*UTM

このような環境依存の機能については、「[SYSTP ユーティリティ](#)」の関連セクションを参照してください。



**注意:** このドキュメントでは、*open*UTM は UTM と記載されています。

- [● SYSTP の呼び出しと機能の実行](#)
- [● SYSTP ユーティリティ画面の使用](#)
- [● 一般的な SYSTP 機能](#)
- [● CICS での SYSTP 機能](#)
- [● IMS/TM での SYSTP 機能](#)
- [● TIAM および UTM での SYSTP 機能](#)
- [● バッチでの SYSTP \(CICS セッション用\)](#)



# 114 SYSTP の呼び出しと機能の実行

このセクションでは、SYSTP ユーティリティを呼び出して、SYSTP ユーティリティ機能を実行する手順について説明します。SYSTP ユーティリティ機能を実行するには、SYSTP ユーティリティメニューまたは SYSTP ダイレクトコマンドを使用します。

## ▶手順 114.1. SYSTP を呼び出してメニュー機能を実行するには

- 1 コマンドプロンプトで、次のシステムコマンドを入力します。

```
SYSTP
```

SYSTP ユーティリティの **[Main Menu]** が表示されます。

- 2 必要な機能に対応する文字コードを **[Code]** フィールドに入力してから Enter キーを押すことにより、SYSTP 機能を実行します。

「E」を入力すると、環境依存の TP モニタ機能のメニューが表示されます。この機能は、Com-plete および TSO では使用できません。

## ▶手順 114.2. SYSTP ダイレクトコマンドを発行するには

- コマンドプロンプトで、またはバッチモードで、SYSTP コマンドを使用します。このコマンドの後には、必要な SYSTP メニュー機能に対応するファンクションコードを指定します。複数のファンクションコードは、メニューから実行される順序で入力できます。複数のファンクションコードを入力する場合は、ピリオド (.) で区切ります。

例：

```
■ SYSTP M.A
```

**[Natural Monitor Menu]** を呼び出し (SYSTP **[Main Menu]** のファンクションコード **M**)、Natural モニタを有効にします ( **[Natural Monitor Menu]** のファンクションコード **A** )。

### ■ SYSTP S.A.C

次の処理の後に、**[Slot Size Calculation]** 画面を呼び出します。  
SYSTP **[Main Menu]** から **S** が実行された後  
**[Natural Swap Main Menu]** から **A** が実行された後  
**[Natural Swap Administration Menu]** から **C** が実行された後

「[バッチでの SYSTP \(CICS セッション用\)](#)」も参照してください。



# 115

## SYSTP ユーティリティ画面の使用

SYSTP ユーティリティメニューで提供されている機能呼び出すには、ファンクションコードを入力するか、必要な機能に対応する PF キー（選択可能な場合）を押します。

次の表で、多くの SYSTP ユーティリティ画面で使用可能な行コマンドおよび PF キー（または対応するダイレクトコマンド）について説明します。行コマンドは、画面上の必要なリスト項目の横の [C] 列に入力します。ダイレクトコマンドはコマンド行に入力します。

PF キーまたはダイレクトコマンド	行コマンド	機能
? または *	? または *	[C] 列に入力した場合、現在の SYSTP 画面で使用可能なすべての行コマンドが表示されます。  コマンド行に入力した場合、SYSTP ユーティリティで使用可能なすべてのダイレクトコマンドが表示されます。
PF1 または HELP	n/a	現在使用されている SYSTP 機能のヘルプ情報を表示します。
PF3	.	現在の画面を終了し、前のレベル/画面に戻ります。
PF5	/ または P	行コマンドが入力された（またはカーソルが置かれている）行を現在の画面の最上部に配置します。
PF4	S または U	行コマンドが入力された（またはカーソルが置かれている）行に含まれている項目の詳細情報を表示します。
PF6	n/a	リストの先頭までスクロールします。

PF キーまたは ダイレクトコマンド	行コマンド	機能
または --		
PF7 または -	n/a	リストを1ページ前にスクロールします。
PF8 または +	n/a	リストを1ページ先にスクロールします。
PF9 または ++	n/a	リストの末尾までスクロールします。

すべての SYSTP 統計画面のヘッダーには、次の情報が含まれます。

- Natural システム変数 \*USER によって割り当てられた現在のユーザーの ID が示された [User] フィールド (『システム変数』ドキュメントを参照)
- Natural システム変数 \*INIT-ID によって現在のユーザーに割り当てられている端末 ID が示された [TID] フィールド (『システム変数』ドキュメントを参照)

# 116 一般的な SYSTP 機能

---

▪ Natural モニタリング (SYSMON) .....	782
▪ Natural 出力／ワークファイル - SYSFILE .....	785
▪ Natural スワップ情報 .....	786
▪ バッファ使用統計 (BUS) .....	790
▪ Natural サブシステムおよびロールサーバー情報 .....	792
▪ Natural スレッド使用統計 .....	793
▪ Natural ライセンス情報 .....	796

## Natural モニタリング (SYSMON)


---

Natural モニタリング機能を使用すると、現在の Natural セッション中に使用された Natural プログラムおよび端末の統計を表示できます。

Natural モニタの範囲は、統計データが収集されるエリアによって決まります。

- グローバルまたはローカルなモニタバッファプールが (BPI プロファイルパラメータまたは NATPARM パラメータモジュールの NTBPI マクロによって割り当てられたとおりに) 使用されている場合は (『パラメータリファレンス』ドキュメントを参照)、このバッファプールを共有するすべての Natural セッションのプログラムおよび端末の統計が収集されます。
- モニタバッファが (MONSIZE プロファイルパラメータにより指定されているとおりに) スレッド内で使用されている場合 (『パラメータリファレンス』ドキュメントを参照)、現在の Natural セッションのプログラム統計および端末情報のみが収集されます。モニタバッファの場合、最小値 64 KB を使用することをお勧めします。

Natural モニタの範囲を定義する以外に、RDCSIZE プロファイルパラメータを設定する必要があります。このパラメータでは Natural Data Collector を有効化します。

 **注意:** アクティブになっている場合、モニタリング機能がシステム全体のパフォーマンスに影響を及ぼすことがあります。

[Natural Monitoring (SYSMON)] 機能呼び出すと、次の機能を使用できる [Natural Monitor Menu] が表示されます。

- モニタの有効化／無効化
- モニタステータス情報
- プログラム／端末統計の表示

### モニタの有効化／無効化

これらの機能では、モニタ機能を有効化または無効化できます。

モニタ機能がアクティブになっている場合は、現在のセッションの統計情報の収集が開始されます。モニタ機能が非アクティブになっている場合は、統計サマリがシステムログファイルに書き込まれます。

## モニタステータス情報

この機能を使用すると、モニタデータがモニタバッファプールか MONSIZE パラメータで割り当てられたスペースのどちらに（または両方に）収集されるかに関する統計情報が示され、割り当てられたスペースのサイズとアドレスが表示されます。

## プログラム／端末統計の表示

モニタの起動後に実行されたすべての Natural プログラム、およびモニタの起動後にアクティブになった端末に関する統計情報を表示できます。

### ▶手順 116.1. プログラムまたは端末の統計機能を実行するには

- 1 [Natural Monitor Menu] の [Code] フィールドに、「P」（プログラム統計の場合）または「T」（端末統計の場合）を入力します。

表示するプログラム／端末またはライブラリ（あるいはその両方）の選択基準を指定します。

[Name of LTERM or Program] フィールドまたは [Name of Library] フィールド（あるいはその両方）に、必要な項目の名前を入力するか、アスタリスク (\*) またはワイルドカード (?)（あるいはその組み合わせ）を使用して名前の範囲を指定します。フィールドを空白のままにした場合、またはアスタリスク (\*) のみを入力した場合は、すべてのプログラム／端末およびライブラリが選択されます。

名前範囲の例：

「\*CD」と入力した場合、ABCD、ACD が選択されます。

「AB\*」と入力した場合、AB1、ABC、ABEZ が選択されます。

「ABC?」と入力した場合、ABCA、ABCZ が選択されます。

「A?C\*Z」と入力した場合、ABCZ、AXCBBBZ、および ANCZ が選択されます。

- 2 選択基準を指定した後で Enter キーを押すと、次の例のような統計概要画面が表示されます。

```

13:44:35          ***** NATURAL SYSTP UTILITY *****          2008-08-29
User SAG          - Natural Monitor Program Statistics -          TID TCK8

C Program  Library      NAT- ADA- Ext- Mean-  Screen I/O  User   Sys  Fetch
*          *          time time time  time     No  KB   Acc  Acc
-----
_ MONMNU1M SYSTP          0   0   0   0.0    18   9    6    0    19
_ SYMAPOM  SYSTP          0   0   0   0.0     0   0    0    0    52
_ S2SCOM01 SYSTP          0   0   0   0.0     0   0    0    0    36
* MONMNU1P SYSTP          0   0   0   0.0     1   0   28    0    6
_ MONLST1P SYSTP          0   0   0   0.0     3   0   21    0   12
_ SYMAP1M  SYSTP          0   0   0   0.0    19  13    2    0   20
_ NAT00059 SYSTP          0   0   0   0.0     0   0    0    0   346
_ STPMNU1P SYSTP          0   0   0   0.0     4   0   30    4    8
_ NAT42004 SYSTP          0   0   0   0.0     0   0    0    0    9

```

```

_ STPMMM1M SYSTP      0  0  0  0.0    7  5  4  0  9
_ SYSTPRET SYSTP      0  0  0  0.0    0  0 13  0  4
_ NAT00030 SYSTP      0  0  0  0.0    0  0  0  0  4
_ NSCC0002 SYSTP      0  0  0  0.0    0  0 14  0  7
_ LOGON     SYSTP      0  0  0  0.0    0  0  0  0  1
  Top of List                                     1 of 3
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Cont Help Menu Exit Sel      --  -  +  ++  Term      Canc

```

この画面には、現在の Natural セッションでアクティブになっているすべてのプログラム／端末およびライブラリがリストされます。

PF4 キーを押すと、**[Selection]** ウィンドウが表示されます。このウィンドウを使用すると、画面に表示される項目リストを縮小するための選択基準を指定できます（手順 1 を参照）。

PF10 キーを使用すると、プログラム統計画面と端末統計画面を切り替えることができます。

次の表に、統計概要画面に表示される列、および詳細統計画面の対応するフィールド名（下記の列 **[C]** を参照）について説明します。

列	対応するフィールド	説明
<b>C</b>	n/a	統計概要画面にのみ適用されます。  この入力フィールドを使用すると、選択したプログラム／端末の詳細な統計画面を表示できます。  詳細な統計情報を表示するリスト項目の隣に、任意の文字を入力して Enter キーを押します。  <b>注意:</b> アクティブなプログラム／端末の統計概要が繰り返し表示される場合は、最後の繰り返し以降に最もアクティブだったプログラム／端末の横の <b>[C]</b> 列にアスタリスク (*) が表示されます。
<b>Program</b>	プログラムの名前	プログラム統計にのみ適用されます。  アクティブなプログラムの名前。
<b>Terminal</b>	端末の名前	端末統計にのみ適用されます。  アクティブな端末の名前 (ID) 。
<b>Current Program</b>	現在のプログラム／ライブラリ	端末統計にのみ適用されます。  実行されたプログラムの名前、およびこのプログラムが含まれるライブラリの名前。
<b>Library</b>	ライブラリの名前	プログラム統計にのみ適用されます。  実行されたプログラムが含まれるライブラリの名前。

列	対応するフィールド	説明
NAT-time	Time in Natural	Natural ニュークリアスおよびインターフェイスでの時間。
ADA-time	Time in Adabas	Adabas からの応答を待機していた時間。
Ext-time	Time in external program	ユーザーが作成したモジュールによって必要とされた時間。
Mean-time	Mean evaluation time	1 つの Natural 画面トランザクションの経過時間。
Screen I/O No	Number of Screen I/Os	画面 I/O の数。
Screen I/O KB	Amount of data transmitted	画面との間で転送されたデータ量。
n/a	Evaluation time > 3 sec	端末の詳細統計画面にのみ適用されます。 3 秒より長い評価時間の割合。
n/a	Evaluation time > 6 sec	端末の詳細統計画面にのみ適用されます。 6 秒より長い評価時間の割合。
User Acc	Number of user file accesses	Adabas ユーザーファイルへのアクセスのカウンタ。
Sys Acc	Number of system file accesses	フェッチを含む、Natural システムファイルへのアクセスのカウンタ。
Fetch	Number of fetches	フェッチの合計回数のカウンタ。

## Natural 出力／ワークファイル - SYSTP

この機能を使用すると、使用可能なワークファイルおよび出力ファイルに関する情報が表示されます。

この機能は、システムコマンド SYSTP（『システムコマンド』ドキュメントを参照）で呼び出すこともできます。

この機能は、CICS セッションに対してはバッチモードで使用することもできます。

この機能呼び出すと、[Work File Information] 画面に、定義されているすべてのワークファイルと出力ファイルのリストが表示されます。各ファイルに対して次の情報が提供されます。

列	説明
No.	ワークファイル／出力ファイルの数
Type	割り当てのタイプ（ワークファイル／出力ファイルが割り当てられているオペレーティングシステム、TP モニタまたは Natural 製品ファイル）
Name	ワークファイル／出力ファイルの名前
Recfm	ワークファイル／出力ファイルのレコードフォーマット
Lrecl	ワークファイル／出力ファイルの論理レコード長（該当する場合）
Blksz	ワークファイル／出力ファイルのブロックサイズ
Status	次のいずれかのステータス：  Available for Input/Output  または  Open for Input/Output

z/VSE では、論理ユニットの割り当ても表示されます。

### Natural 出力ファイル／ワークファイルのコマンド

「**Work File Information**」画面では、「**SYSTP ユーティリティ画面の使用**」で説明しているコマンドの他に、次の PF キーと行コマンドを使用できます。

PF キー	行コマンド	機能
PF10	n/a	出力ファイルのリストにスクロールします。
PF11	n/a	ワークファイルのリストにスクロールします。
n/a	D	対応する Natural コントロールブロック（ワークファイルエリア）をダンプ形式で表示します（Software AG 技術サポートが内部的に使用するため）。

## Natural スワップ情報

この機能は、CICS および UTM でのみ使用できます。

スワッププールマネージャによって、Natural スワッププールのオンラインモニタリングと制御が可能になります。このセクションでは、スワッププールの動作方法ではなく、スワッププールマネージャの使用方法について説明します。Natural スワッププールの操作の詳細については、『オペレーション』ドキュメントの「Natural スワッププール」を参照してください。

この機能呼び出すと、次の機能を選択できる「**Natural Swap Main Menu**」が表示されます。

- **管理**
- **デバッグ機能**



- 情報
- メンテナンスサービス
- ステータス情報

## 管理

- スロットサイズの計算
- スワッププールステータスの変更
- 再構成制御データの更新

### スロットサイズの計算

この機能は、現在の使用状況に基づいて、スワッププールのレイアウトの最適値を表示します。

これらの値を格納して、後で初期化／再編成に使用できます（格納された後は、[メンテナンスサービス](#)機能を使用してメンテナンスすることもできます）。

これらの値を使用して、スワッププール認識を開始することもできます。

詳細については、この機能のオンラインヘルプを参照してください。

### スワッププールステータスの変更

この機能は、Natural スワッププールの有効化または無効化に使用します。また、スワッププール同期の待機時間および待機回数を変更できます。

詳細については、この機能のオンラインヘルプを参照してください。

### 再構成制御データの更新

この再構成機能を使用すると、スワッププール管理において最も重要なパラメータを変更できます。値を変更するには、有効なパスワードを入力する必要があります。

詳細については、この機能のオンラインヘルプを参照してください。

## デバッグ機能

この機能は、UTM でのみ使用できます。

 **注意:** この機能は、必ず事前に Software AG 技術サポートに相談したうえで使用してください。

この機能を使用すると、内部画面デバッグバッファを有効または無効にすることができます。画面デバッグバッファの有効化は、端末 I/O の矛盾が発生した場合に、その矛盾を特定するために使用します。この機能を使用すると、最後の 3 つの端末 I/O シーケンスに関する情報が記録されます。バッファのサイズは 3KB であり、ラップアラウンドプロシージャで使用されます。

また、Natural ロールファイルへの非同期書き込み操作のトレース機能を有効化／無効化することもできます。

詳細については、この機能のオンラインヘルプを参照してください。

### 情報

- アドレスの表示
- バッファ使用率の概要の表示
- スワッププール情報の表示
- 論理スワッププールの表示
- 再構成制御データの表示
- スワッププール使用率の表示
- 統計リストの作成

#### アドレスの表示

この機能は、さまざまなプールのアドレスを表示します。

#### バッファ使用率の概要の表示

この機能は、さまざまなNatural バッファおよびNatural ユーザースレッドのサイズ (MAXSIZE) を最適化するために使用します (『オペレーション』ドキュメントの「MAXSIZE」を参照)。Natural バッファ使用率の概要を有効化、無効化および表示します。

バッファ統計の有効化と無効化は、有効なパスワードでのみ実行できます。バッファ統計を表示するには、パスワードは不要です。

表示されるバッファは、**バッファ使用統計 (BUS)** 機能によって表示されるバッファと同じです。

#### スワッププール情報の表示

この機能を使用すると、制御／統計データやメモリサイズなど、現在使用中のスワッププールに関する情報が表示されます。

表示される情報の各項目については、この機能のオンラインヘルプを参照してください。

## 論理スワッププールの表示

この機能は、論理スワッププールの現在のテーブルを表示します。

テーブルでは、特定の論理スワッププールを任意の文字でマークして、その論理スワッププールに関する追加情報を取得できます。

表示される情報の各項目については、この機能のオンラインヘルプを参照してください。

## 再構成制御データの表示

この機能は、スワッププール認識に関連するすべての情報を表示します。

画面の左半分に、スワッププール再構成テーブルが表示されます。テーブルには、圧縮された Natural ユーザースレッドと標準スロットサイズ間の比較サイズに関する累積統計が含まれます。テーブルは、スワッププールの認識のたびにクリアされます。テーブルの左半分には、ユーザースレッドが標準スロットサイズよりも大きかった頻度と、その程度が表示されます。テーブルの右半分には、ユーザースレッドが標準スロットサイズよりも小さかった頻度と、その程度が表示されます。テーブルのこの半分のサイズは、スワッププールマネージャで指定された係数に依存する単位で表現されます。

**n** というラベルの付いた行では、標準スロットサイズよりも 9 ページ / 単位を超えて長い / 短いユーザースレッドがカウントされます。これらのユーザースレッドの平均の長さは、**Av.+n** というラベルの付いた行に表示されます。

表示される情報の各項目については、この機能のオンラインヘルプを参照してください。

## スワッププール使用率の表示

この機能を使用すると、初期化または最後の認識の後のスワッププールの使用率に関する情報が表示されます。

表示される情報の各項目については、この機能のオンラインヘルプを参照してください。

## 統計リストの作成

この機能は、現在のスワッププール使用率の統計のリストを作成するために使用します。

- UTM では、スワッププール統計は SYSLIST に書き込まれます。
- CICS 環境では、スワッププール統計は NCMPRM マクロの MSGDEST パラメータで指定された CICS の宛先に書き込まれます（『TP モニタインターフェイス』ドキュメントを参照）。

### メンテナンスサービス

- パラメータメンテナンス
- パスワードメンテナンス

#### パラメータメンテナンス

この機能は、スワッププールの初期化または認識のパラメータをオンラインで変更するために使用します。

サブ機能に加えて、変更できる個々の項目が、この機能のオンラインヘルプで説明されます。

この機能の使用は、パスワードで保護されます（下記を参照）。

#### パスワードメンテナンス

この機能は、[パラメータメンテナンス](#)機能で使用されるパスワードを変更または回復するために使用します。

初期パスワードは SYSTP です。

#### ステータス情報

この機能では、Natural スワッププール、バッファ使用率の概要および UTM 画面デバッグの現在のステータスを表示できます。

### バッファ使用統計 (BUS)

---

この機能を使用すると、Natural バッファの使用率に関する統計情報（現在の Natural セッションに割り当てられているバッファ、および使用されているバッファ容量）が表示されます。

統計リストの最後にある [Total] の数字によって、バッファ圧縮の効率に関する結論を引き出すことができます。

この機能は、**SYSTP** メニューから、またはシステムコマンド BUS を使用して呼び出すことができます。

この機能呼び出すと、現在の Natural セッションで実際に使用されているすべてのバッファを示すリストが表示されます。

これらの各バッファについて、次の情報が [Buffer Usage Statistics] 画面に表示されます。

列	説明
<b>No.</b>	バッファには割り当て順に番号が付けられます。
<b>Name</b>	バッファの名前。現在のセッションで実際に要求されたバッファだけがリストに表示されます。
<b>Type</b>	Vは、可変バッファを示します。可変バッファのサイズは、(Natural スレッドの外部で割り当てられた場合でも) 必要に応じて自動的に増やされます。スレッドの外部で割り当てられている場合は、端末I/O時にスレッドにコピーされます。スレッドに収まらない場合は、実際に使用されている長さに切り捨てられます。
<b>Size</b>	バッファのサイズ (バイト単位)。
<b>Used</b>	現在使用されているバイト数。この値は、スレッドを使用している環境 (例えば、CICS や UTM など) でバッファ圧縮に使用されます。
<b>Perc. (Used)</b>	現在使用されている割合。つまり、[Size] 列の値との相対での [Used] 列の値です。
<b>MaxUsed</b>	現在のセッションでこれまでに使用された最大バイト数 (現在使用されているサイズではありません)。
<b>Perc. (MaxUsed)</b>	現在のセッション使用の割合。つまり、[Size] 列の値との相対での [MaxUsed] 列の値です。
<b>MaxSize</b>	現在のセッションでこれまでにバッファに割り当てられた最大サイズ (バイト単位) (可変バッファにのみ適用されます)。
<b>Perc. (MaxSize)</b>	現在のサイズ ([Size] 列の値) との相対での、これまでに割り当てられた最大サイズ ([MaxSize] 列の値) (可変バッファにのみ適用されます)。1000 以上の割合は、高輝度で表示された 999.9 で示されます。
リストの最後には、次の情報が表示されます。	
<b>ThrdSize</b>	Natural スレッドの現在のサイズ (KB 単位)。
<b>Total</b>	すべてのバッファサイズの合計 (バイトとKBの両方) および使用された/割り当てられた割合。これらの合計は、PF10 キーでも表示できます (下記を参照)。  [MaxSize] については、合計は、セッションでこれまでに必要とされたスレッドサイズの最大追加量を示します。

## バッファ使用統計のコマンド

「**Buffer Usage Statistics**」画面では、「**SYSTP ユーティリティ画面の使用**」で説明しているコマンドの他に、次の PF キーと行コマンドを使用できます。

PF キー	行コマンド	機能
PF4	D	カーソル/コマンドでマークされたバッファの内容をダンプ形式で表示します (Software AG 技術サポートが内部的に使用するため)。
PF10	n/a	[ <b>Total</b> ] バッファ使用量の数値を表示します。
PF11	n/a	バッファの相対アドレス、つまり、入力/出力制御バッファ (IOCB) との相対でのアドレス表示します。

## Natural サブシステムおよびロールサーバー情報

この機能は、環境で使用可能な Natural サブシステムのリストと、サブシステムに関連付けられている承認済みサービスマネージャおよびロールサーバーの現在のステータス (アクティブまたは非アクティブ) を表示します。

「**Natural Sub-Systems**」画面では、「**SYSTP ユーティリティ画面の使用**」で説明しているコマンドの他に、次の行コマンドを使用できます。

行コマンド	機能	オペレーティングシステム
B	名前、タイプ、サイズなど、バッファプールおよびバッファプールキャッシュ (使用可能な場合) に関する情報を表示します。	z/OS および z/VSE
D	セッション情報プール (SIP) のサーバーディレクトリエントリをダンプ形式で表示します (Software AG 技術サポートが内部的に使用するため)。  SIPの詳細については、『オペレーション』ドキュメントの「z/OS での認可サービスマネージャ」を参照してください。	z/OS
L	ロールサーバーファイルディレクトリのエントリを表示およびリセットします。	z/OS
R	ロールサーバーロットやロールサーバーファイルの数、ロールアウトアクティビティとロールインアクティビティ、ロールファイル I/O など、ロールサーバーの統計情報を表示します。  この情報は、『オペレーション』ドキュメントの「ロールサーバーのパフォーマンスチューニング」で説明されているロールサーバーの調整に役立ちます。例えば、統計データを使用して、Natural アプリケーションに対して最適なスレッドサイズやロールファイルサイズを判断できます。	z/OS
S	認可サービスマネージャに適用される ZAP のリストを表示します。	z/OS

行コマンド	機能	オペレーティングシステム
U	セッション情報プール (SIP) のサーバーディレクトリエントリを表示します。  SIPの詳細については、『オペレーション』ドキュメントの「z/OSでの認可サーバスマネージャ」を参照してください。	z/OS
Z	ロールサーバーに適用される ZAP のリストを表示します。	z/OS

## Natural スレッド使用統計

この機能は、CICS、Com-plete、IMS/TM、および UTM でのみ使用できます。Sysplex 環境では使用できません。

この機能を使用すると、Natural アプリケーションに対して最適なスレッドサイズやロールファイルサイズを判断できます。

この機能は、Natural バッファプールの領域を占有するため、必要な場合にのみアクティブにする必要があります。最適なスレッドサイズを判断した後で非アクティブにする必要があります。非アクティブにすると、バッファプール内の領域が再び使用可能になります。

### ▶手順 116.2. Natural スレッド使用統計機能を実行するには

- 1 Natural アプリケーションに対して、512~1024KB の範囲で大きなサイズのスレッドを定義します。使用されている Natural アドオン製品の数を考慮します。
- 2 Natural アプリケーションを稼働モードまたはテストモードで起動します。
- 3 SYSTP の [Main Menu] の [Code] フィールドに「T」を入力して、SYSTP ユーティリティを呼び出し、[Natural Thread Usage Statistics] 機能を選択します。

[Thread Usage Statistics] 画面が表示されます。

次の表で、この画面に表示される列について説明します。

- 4 機能を有効化するには、[Code] フィールドに「A」を入力します。

スレッド使用統計機能が有効になったことを示すメッセージが表示されます。

- 5 通常の稼働条件で Natural アプリケーションを使用します。

スレッド使用統計機能がバックグラウンドで実行され、使用されたバッファサイズがログに記録されます。

- 6 統計を再び表示するには、[Thread Usage Statistics] 画面を呼び出し、[Code] フィールドに次のいずれかを入力します。

S

統計を表示する場合

P

統計を出力する場合

G

GETMAIN 統計を表示する場合（「物理 GETMAIN 統計の表示」を参照）

D

統計を出力して機能を無効化する場合

機能の完了後は、ファンクションコード D を使用してバッファプールスペースを解放することをお勧めします。

[Natural Thread Usage Statistics] 画面には、次の情報が表示されます。

列	説明
No.	バッファには割り当て順に番号が付けられます。
Ext. Buffer	これらのバッファのサイズは外部で定義されます（NATPARM パラメータモジュール内）。
Defined Size	NATPARM パラメータモジュールで定義されたバッファサイズ。
Max. Allocated Size	割り当てられた最大バッファサイズ。  内部 BB 領域では、ESIZE プロファイルパラメータの値に 14368 バイトが追加されることに注意してください（『パラメータリファレンス』ドキュメントを参照）。
Max. Used Size	使用される最大バッファサイズ。
Sum of external buffer sizes	NATPARM パラメータモジュールで定義されたすべてのバッファサイズの合計。
Sum of internal buffer sizes	Natural で内部的に要求されたすべてのバッファサイズの合計。
Sum of physical GETMAINs	Natural ワークプール、および Natural ユーザースレッドの外部にある可変 Natural バッファのすべての物理 GETMAIN の合計。



列	説明
<b>Max. used thread length</b>	Naturalによって使用される最大スレッド長。この長さは、最小の（最適な）Natural スレッド長として定義します。  2 で割り切れる次の KB 数値に切り上げます。
<b>Max. compressed thread length</b>	Natural ロールファイルに書き込まれた、圧縮された Natural スレッドの最大の長さ。  この長さは、最小の（最適な）Natural ロールファイル長として定義します。

## 物理 GETMAIN 統計の表示

物理 GETMAIN 統計は、Natural ワークプール、および Natural ユーザースレッドの外部にある可変 Natural バッファに関連するすべての物理 GETMAIN に関する情報を提供します。これらは、Natural セッションの起動時の元のバッファサイズ（**[Org. Size]**）、物理 GETMAIN の数（**[No.]**）、物理 GETMAIN のバッファ長（**[Bytes]**）、およびバッファ位置（**[P]**）（16-MB 境界より上または下）を示します。

統計データは、常に、端末 I/O 内で Natural アプリケーションのすべてのユーザーに対して要求された最大の長さのバッファを表します。統計は、各バッファに対して最大 6 つのエントリを提供します。これらのエントリは、ラップアラウンド手順によって上書きされることがあります。最大数は、考慮される各バッファについて、端末 I/O 内の物理 GETMAIN の最大数と同じです。

統計の最初の 2 つのエントリは、それぞれ 16-MB 境界より上（**[WRKPOOLA]**）または下（**[WRKPOOLB]**）の Natural ワークプール（使用可能な場合）を表します。ここで、物理 GETMAIN の最大数は、端末 I/O 中に同時に使用可能なワークプールの量を表します。ワークプールの長さの合計が、端末 I/O 内のワークプールの合計ストレージ要件になります。

後続のすべての統計エントリは、可変 Natural バッファの物理 GETMAIN を表します。これは、スペースが不十分なため Natural ユーザースレッドで定義できなかったか、または Natural ユーザースレッドの外部で増加したものです。これらのバッファについて、物理 GETMAIN の最大数は、端末 I/O 内の各バッファの最大スペース要件を示します。前に要求された合計ストレージスペースは、次の各物理 GETMAIN の前に解放されています。つまり、最大数のすべての物理 GETMAIN の合計は、Natural アプリケーションのすべてのユーザーについて、端末 I/O 中に Natural ユーザースレッドの外部の可変バッファの最大ストレージ要件を示します。

## Natural ライセンス情報

---

この機能は、ライセンスキーファイルの内容を XML 形式で表示する画面を呼び出します。

# 117 CICS での SYSTP 機能

---


▪ Natural ユーザーセッション .....	798
▪ Natural ロール機能 .....	803
▪ Natural Thread Groups .....	804
▪ Natural ストレージスレッド .....	805
▪ NCI グローバルシステム情報 .....	806
▪ NCI 生成オプション .....	808
▪ Natural スレッドグループ定義 .....	808
▪ 固有の Natural ユーザーセッション .....	810
▪ CICS タスク情報 .....	810
▪ システム管理機能 .....	810

SYSTP ユーティリティは、CICS に固有の機能を提供します。

### ▶手順 117.1. CICS で特定の SYSTP 機能呼び出すには

- SYSTP の [Main Menu] の [Code] フィールドに、[Environment-Dependent Functions] に対応する「E」を入力します。

続いて表示されるメニューから、このセクションで説明している機能を選択できます。

 **注意:** このセクションの以降の部分では、Natural CICS インターフェイスは NCI とも記載されています。

## Natural ユーザーセッション

この機能は、Natural 環境のアクティブなユーザーセッションのリストを表示するために使用します。

この機能呼び出すと、次の情報が示される [Natural User Sessions] 画面が表示されます。

列	説明
Term ID	Natural セッションに関連付けられている、CICS 内で一意の端末 ID。
User ID	Natural セッションの Natural ユーザー ID。
Tran	Natural セッションの実行に現在使用されている CICS トランザクション ID。 擬似会話型セッションの場合、これは擬似会話型再起動トランザクション ID です。
Start Date / Time	Natural セッションの開始日時。
Last Act	最後の画面出力の時刻。
Stat	セッションステータス: 「 <i>Operational Status</i> 」を参照してください。
Program	現在アクティブな Natural プログラム。
Library	ユーザーが現在作業している Natural ライブラリ。

PF10 キーを押した場合は、セッション日時の表示が次のセッションリソースデータで置換されません。

列	説明
Thrd Grp	ユーザーが割り当てられているスレッドグループ。
Thread	最後に使用されたスレッドの名前。
Roll Fac	割り当てられているロール機能。

このセクションでは、次のトピックについて説明します。

- Natural ユーザーセッションのコマンド
- Natural ユーザーセッション統計

## Natural ユーザーセッションのコマンド

[Natural User Sessions] 画面に表示された各項目に対して、「SYSTP ユーティリティ画面の使用」で説明しているコマンド以外に、次のいずれかの行コマンドを実行できます。

行コマンド	機能
C	<p>セッションをキャンセルします。</p> <p>確認ウィンドウが表示されます。このウィンドウで「YES」と入力してセッションの終了を指定できます。選択されたセッションは、[UserID] 列に表示される数字 (#) 記号で示されます。また、対象のセッションについて、キャンセルコマンドを実行したユーザー（管理者）の [Natural User Session Statistics] 画面にオペレーションステータス「Purged by Admin」が表示されません。</p> <p>セッションは、終了のマークが付けられたセッションのオーナーが次の端末 I/O、Adabas コールまたは外部プログラムコールを実行したときに実際に終了します。セッションオーナーは、対応する終了通知を受信します。</p>
F	<p>セッションをフラッシュします。</p> <p>確認ウィンドウが表示されます。このウィンドウで「YES」と入力してセッションを即時に終了できます。終了されたセッションは、[UserID] 列に表示される数字 (#) 記号で示されます。セッションを終了したユーザー（管理者）は、終了はしたが [Natural User Session] 画面のリストにまだ表示されているセッションに対して [Natural User Session Statistics] 画面を呼び出そうとすると、終了メッセージを受信します。セッションオーナーは、対応する終了通知を受信します。</p>
R	<p>セッションを再びアクティブにします。</p> <p>前に説明した C（キャンセル）コマンドを元に戻します。R コマンドは、セッションに対して設定されている終了フラグを削除し、セッションステータスをアクティブにリセットします。前に説明した F（フラッシュ）コマンドで終了したセッションを再びアクティブにすることはできません。</p>
W	<p>セッションを起動します。</p> <p>MAXROLL プロファイルパラメータでゼロ以外の待機間隔が指定された CMROLL コールで中断されていたセッションを即時に再びアクティブにします（『パラメータリファレンス』ドキュメントを参照）。Natural システムライブラリ SYSEXTP で提供されているプログラム例 SUSPEND も参照してください。</p>

### ▶手順 117.2. 選択条件を指定してリスト項目の数を制限するには

- [Natural User Sessions] 画面で、PF4 キーを押します。

[**Selection for User Sessions**] ウィンドウが表示されます。このウィンドウで、非アクティビティ日時、非アクティビティ時間間隔、Natural サーバー ID、CICS システム ID、端末 ID、ユーザー ID または トランザクション ID によってユーザーセッションを選択できます。

非アクティビティ日時および非アクティビティ時間間隔では、指定された日時より前または指定された時間間隔より前にアクティブでなかったすべてのセッションが表示されます。

日付は YYYY-MM-DD (YYYY = year, MM = month, DD = day) の形式で指定する必要があります。時刻は、HH:II:SS (HH = hours, II = minutes, SS = seconds) の形式で指定する必要があります。

ID の範囲を指定するには、アスタリスク (\*) を先頭または末尾の文字として使用するか、疑問符 (?) をワイルドカード文字として使用します。「[名前範囲の例](#)」も参照してください。

## Natural ユーザーセッション統計

上記の [**Natural User Sessions**] 画面に表示される各セッションについて、行コマンド s または u を使用して [**Natural User Session Statistics**] 画面を呼び出すことによって、追加情報を表示できます。次のセクションでは、この画面に表示される情報について説明します。

[**Natural User Session Statistics**] 画面上のすべてのサイズは、次のフィールド説明で指示されていない限り KB 単位です。

フィールド	説明
Started	セッションが開始したときの曜日、日付および時刻。
Last Actions	ユーザーが最後にアクティブだった日付と時刻。
User	Natural システム変数 *USER によって割り当てられた Natural ユーザー ID (『システム変数』ドキュメントを参照)。
at Terminal	Natural システム変数 *INIT-ID によって割り当てられた、Natural セッションに関連付けられている端末の ID (『システム変数』ドキュメントを参照)。
Transid	Natural の実行に使用されている擬似会話型トランザクション ID。
Task # in	CICS によって割り当てられたタスク番号に続いて、CICS リージョンの ID。
Cur Strg Used	このセッションで使用されている現在のストレージ容量。
Max Strg Used	このセッションでこれまでに使用された最大ストレージ容量。
Thread Size	このスレッドのサイズ。
Thread Name	最後に使用されたスレッドの名前。GETMAIN を使用して割り当てられたスレッドの場合、スレッドの名前は接頭辞 NSCP とそれに続く端末 ID から構成されます。
Thread Group	関連付けられているスレッドグループの名前 (トランザクション ID の開始によってトリガされます)。
of Type	スレッドグループに使用されているスレッドの種類。

フィールド	説明
	SHR 常駐ストレージスレッドです。
	GETM GETMAIN を使用して割り当てられるストレージスレッドです。
	NONE スレッドは使用されていません。すべての Natural ストレージ要求が CICS に渡されます。
Natural Library	Natural システム変数 *LIBRARY-ID によって割り当てられた Natural ライブラリ ID (『システム変数』ドキュメントを参照)。
Natural Program	Natural システム変数 *PROGRAM によって割り当てられた、セッションによって現在使用されている Natural プログラムの名前 (『システム変数』ドキュメントを参照)。
Line No.	セッションで現在使用されている Natural プログラム内の行番号。
Operational Status	次のセクションの「 <i>Operational Status</i> 」を参照してください。
Roll Facility	関連付けられているロール機能の名前。
Roll Recs (Last)	最後のロールアウトについてロール機能に書き込まれた記録数。
Roll Recs (Max)	ロールアウト中に書き込まれた最大記録数。
Roll Record Size	このロール機能の記録サイズ。
Slot Size	スレッドを完全にロールアウトするために必要な記録数。
Restart Rec. No.	<p>ロールアウト制御情報を含む記録の番号。この記録が最初にロールインされる必要があります。</p> <p><b>VSAM</b> ロールファイル：</p> <p>次の情報は、VSAM ロールファイルにのみ適用されます。</p> <p>再起動記録番号 (RecNum)、スロット番号 (SN) およびスロットサイズ (SZ) の関係は次のとおりです。</p> $\text{RecNum} = (\text{SN}-1) * \text{SZ} + 2$ <p>または</p> $\text{SN} = (\text{RecNum}-2) / \text{SZ} + 1$
Slot Number	このセッションに属する VSAM ロールファイル内のスロットの番号 (VSAM のみ)。上記の [Restart Rec. No.] の「 <b>VSAM ロールファイル</b> 」も参照してください。
Compressed Length	現在スワップ/ロールアウトされている関連ストレージの量。
Session Resumes	セッション再開の合計数。
Swap-Ins	スワッププールからのスワップインによるセッション再開の数。
Thread Switches	セッションが以前に実行されていたスレッドとは異なるスレッドへのスワップ/ロールインによるセッション再開の数。

フィールド	説明
Roll-Ins	ロール機能からのロールインによるセッション再開の数。
Region Switches	CICS リージョンスイッチの数。
OpSys Switches	Sysplex 環境のオペレーティングシステムイメージスイッチの数。

### Operational Status

このフィールドは、次のいずれかのオペレーションステータスを示します。

ステータス	省略形	説明
Active	Act	現在アクティブです。
Inactive	Ina	非アクティブですが、まだスレッド内にあります。
Swapped	Swp	スワップされ、スワッププール内にあります。
Rolled out	Rld	ロールアウトされ、ロール機能内にあります。
Wait (Init)	WtI	セッション初期化でスレッドを待機しています。
Wait (Resume)	WtR	セッション再開でスレッドを待機しています。
Initializing	Int	セッションを初期化しています。
Resuming	Res	セッションを再開中で、スレッド内にありますが、まだアクティブではありません。
Suspending	Sus	セッションを中断しています。
Terminating	Trm	セッションを終了しています。
Swapping out	Swo	セッションをスワップアウトしています。
Swapping in	Swi	セッションをスワップインしています。
Rolling out	Out	スレッドまたはスワッププールからロールアウトしています。
Rolling in	In	ロール機能からロールインしています。

[Operational Status] には、次の追加情報を表示できます。

ステータス	説明
Conversational	擬似会話型／トランザクション指向のセッションとは対照的なダイアログ指向のセッション (PSEUDO=OFF) です。  『パラメータリファレンス』ドキュメントで説明している PSEUDO プロファイルパラメータも参照してください。
Forced Conversational	PSEUDO=ON セッションの最後の画面 I/O が会話型でした。  『パラメータリファレンス』ドキュメントで説明している PSEUDO プロファイルパラメータも参照してください。
No-Roll	セッションはロールを許可されていません。
Compressed	セッションはスワッププールまたはロール機能内で圧縮されています。



ステータス	説明
Thread Switched	現在使用されているスレッドは、以前に使用されていたスレッドと同じではありません。
Thread Locked	セッションでスレッドが切り替えられないようにします（例えば、RELO=OFF）。No-Roll/Conversational ステータスも強制的に設定されます。  『パラメータリファレンス』ドキュメントで説明している RELO プロファイルパラメータも参照してください。
Purged by Admin	セッションが管理者によってキャンセルされました（フラグが設定されました）。
Spool Task	タスクはスプール/プリントタスクです。
Asynchronous Task	タスクは非同期タスクであり、端末にバインドされていません。

## Natural ロール機能

この機能は、ユーザーワークエリアのロールアウトにどのスワップファイルが使用可能であるかを表示して、アクティブなユーザーのためにスワッププール内の領域を空けるために使用されます。これらのスワップファイルをロール機能と呼びます。

この機能呼び出すと、現在の CICS リージョンの **[Natural Roll Facilities]** 画面が表示されます（画面タイトルの CICS ID で示されます）。各ロール機能について、次の情報が表示されます。

列	説明
<b>Facility Name</b>	TEMPSTOR は補助的な一時ストレージに使用され、MAINSTOR は主要な一時ストレージに使用されます。その他のファイル名は、CICS ファイル制御テーブル (FCT) で定義された VSAM ロールファイルです。none は、ロール機能が何も使用されていないことを示します。
<b>Record Size</b>	このロール機能のレコードサイズ。
<b>Slot Size</b>	スレッドを完全にロールアウトするために必要なレコード数（最大スレッドサイズをレコードサイズで割って切り上げた数値）。
<b>No. of Slots</b>	このロールファイルに収まるセッション数（ファイルレコードの数をスロットサイズで割って切り捨てた数値）。VSAM ロールファイルにのみ適用されます。
<b>Facility Users Cur / Max</b>	このロール機能に割り当てられているユーザーセッションの現在の数（ <b>[Cur]</b> ）および最大数（ <b>[Max]</b> ）。
<b>Roll Counts Out / In</b>	このロール機能に対するセッションロール操作の数。
<b>Status</b>	機能のユーザー数が使用可能なスロット数と等しい場合は、Full を示します。

[Natural Roll Facilities] 画面では、「[SYSTP ユーティリティ画面の使用](#)」で説明しているコマンドの他に、次の PF キーおよび対応する行コマンドを使用できます。

PF キー	行コマンド	機能
PF4	U	マークされたロール機能に対する <a href="#">Natural User Sessions</a> 画面（関連するセクションを参照）を呼び出します。この画面には、この機能を使用するすべての Natural ユーザーセッションが表示されます。

## Natural Thread Groups

この機能は、Natural で使用可能なスレッドグループを表示するために使用します。

この機能呼び出すと、現在の CICS リージョンの [Natural Thread Groups](#) 画面が表示されます（画面タイトルの CICS ID で示されます）。各スレッドグループについて、次の情報がこの画面に表示されます。

列	説明
Group Name	スレッドグループ名。
Group Users Cur / Max	このスレッドグループに割り当てられているユーザーの現在の数（[Cur]）および最大数（[Max]）。
Thread Type	使用されているスレッドの種類。 <a href="#">Natural User Session Statistics</a> を参照してください。
TCBs	同時にアクティブになるセッションの最大数。
Thread Size	スレッドグループの共通スレッドサイズ。
Strg Used	このグループの任意のスレッドに割り当てられている最大ストレージ。
Queue Sizes Cur / Max / AtMax	スレッドグループの中央待機キューの現在のサイズ（[Cur]）および最大サイズ（Max）と、最大値に達した回数（[AtMax]）。  このスレッドグループに対してパラメータ THREADS がゼロより大きく定義されている場合にのみ適用されます。『TP モニタインターフェイス』ドキュメントの「THREADS - スレッドグループごとのスレッドまたはタスクの数」も参照してください。
VSAM Aux / Main	グループに対して定義されているロール機能。CICS 一時ストレージ（補助または主要）は、VSAM ロールファイルが使用可能でないかいっぱいの場合に、必ず VSAM をバックアップします。

## Natural スレッドグループのコマンド

[Natural Thread Groups] 画面では、「[SYSTP ユーティリティ画面の使用](#)」で説明しているコマンドの他に、次の PF キーおよび対応する行コマンドを使用できます。

PF キー	行コマンド	機能
PF4	U	カーソル/コマンドでマークされたスレッドグループを使用するすべての <b>Natural ユーザーセッション</b> (関連セクションを参照) を表示します。
PF10	T	カーソル/コマンドでマークされたスレッドグループに対する <b>Natural ストレージスレッド</b> (下記を参照) を表示します。
PF11	D	カーソル/コマンドでマークされたスレッドグループに対する <b>Natural スレッドグループ定義</b> (下記を参照) を表示します。

## Natural ストレージスレッド

この機能は、Natural 環境のストレージスレッドに関する情報を表示するために使用します。

この機能を呼び出すと、現在の CICS リージョンの [**Natural Storage Threads**] 画面が表示されます (画面タイトルの CICS ID で示されます)。画面には、次の情報が表示されます。

列	説明				
<b>Thread Name</b>	スレッドの名前。				
<b>Grp No.</b>	このスレッドが属するグループの番号。				
<b>Thrd Size</b>	使用可能なスレッドサイズ。				
<b>Strg Used</b>	このスレッドで使用されたストレージの最大容量。				
<b>Use Count</b>	このスレッドの処理が選択された回数。				
<b>Roll-Ins LOG / Phys.</b>	ロールイン操作の回数： <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"><b>Log.</b></td> <td>セッション再開</td> </tr> <tr> <td><b>Phys.</b></td> <td>ロール機能からのロールイン</td> </tr> </table>	<b>Log.</b>	セッション再開	<b>Phys.</b>	ロール機能からのロールイン
<b>Log.</b>	セッション再開				
<b>Phys.</b>	ロール機能からのロールイン				
<b>Queue Sizes Cur / Max / AtMax</b>	キューで待機しているユーザー数： <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"><b>Cur</b></td> <td>スレッドでキューに入っているユーザーの現在の数。この数 <math>n</math> が 1 より大きい場合、<math>n-1</math> のユーザーが待機しています。</td> </tr> <tr> <td><b>Max</b></td> <td>このスレッドの最大キューカウント。</td> </tr> </table>	<b>Cur</b>	スレッドでキューに入っているユーザーの現在の数。この数 $n$ が 1 より大きい場合、 $n-1$ のユーザーが待機しています。	<b>Max</b>	このスレッドの最大キューカウント。
<b>Cur</b>	スレッドでキューに入っているユーザーの現在の数。この数 $n$ が 1 より大きい場合、 $n-1$ のユーザーが待機しています。				
<b>Max</b>	このスレッドの最大キューカウント。				

列	説明
	<b>AtMax</b>   最大回数。
<b>Term ID</b>	データがこのスレッド内にある、Natural セッションに属している端末 ID。
<b>Task No.</b>	このスレッドで現在アクティブな CICS タスクの ID。ID が表示されない場合、このスレッドでアクティブなセッションはありません。

## Natural ストレージスレッドのコマンド

[Natural Storage Threads] 画面では、「[SYSTP ユーティリティ画面の使用](#)」で説明しているコマンドの他に、次の行コマンドと PF キーを使用できます。

行コマンド	PF キー	機能
C	n/a	「Natural ユーザーセッション」の「 <a href="#">セッションのキャンセル</a> 」を参照してください。
D	PF11	<b>Natural スレッドグループ定義</b> を表示します（関連セクションを参照）。
F	n/a	「Natural ユーザーセッション」の「 <a href="#">セッションのフラッシュ</a> 」を参照してください。
G	PF10	<b>Natural スレッドグループ</b> を表示します（関連セクションを参照）。
R	n/a	「Natural ユーザーセッション」の「 <a href="#">セッションの再アクティブ化</a> 」を参照してください。
U	n/a	スレッドで現在アクティブな Natural ユーザーセッションに関する統計を表示します。「 <a href="#">Natural ユーザーセッション統計</a> 」も参照してください。

## NCI グローバルシステム情報

この機能は、システムディレクトリに関するデータを表示するために使用します。

この機能呼び出すと、現在の CICS リージョンの「**Global System Information**」画面が表示されます（画面タイトルの CICS ID で示されます）。画面には、次の情報が表示されます。

フィールド	説明
<b>Natural User Sessions</b>	システム内の Natural セッションの現在の数（ <b>[Cur]</b> ）および最大数（ <b>[Max]</b> ）。
<b>Concurrent SCP Active</b>	同時のシステム制御プログラム（SCP）要求の現在の数（ <b>[Cur]</b> ）および最大数（ <b>[Max]</b> ）。  SCP 要求は、セッション初期化、セッション中断、セッション再開、およびセッション終了です。
<b>SIR Block Extensions</b>	ローカル SIR ブロックエクステンションの現在の数（ <b>[Cur]</b> ）および最大数（ <b>[Max]</b> ）。

フィールド	説明
Slots in 1st SIR Block	プライマリユーザーコントロールブロック (NCMDIR マクロの最初の USERS サブパラメータ) に収まるユーザーセッションの数 (「USERS-セッション情報レコード」を参照)。
Slots in SIR Block Extns	セカンダリユーザーコントロールブロック (NCMDIR マクロの2番目の USERS サブパラメータ) に収まるユーザーセッションの数 (「USERS-セッション情報レコード」を参照)。
VSAM Roll File Slots	チェックする VSAM ロールファイルの数 (ROLLFLS)。
Possible Roll Facilities	VSAM ロールファイルの数に CICS (MAINSTOR および TEMPSTOR) 用の 2 を加えた数。
Thread Groups	システム起動時にすべての NCMTGD マクロ指定を評価することで判断される スレッドグループの数。「NCMTGD マクロパラメータ」*も参照してください。
System Recoveries	統計カウントまたはコントロールブロックチェーン、あるいはその両方の修正の数。
Size of DIR Extension (B)	スレッドコントロールブロックと VSAM ロールファイルオンラインディレクトリについてシステム起動時に使用されたバイト数。
Operating System Host ID	オペレーティングシステムイメージの数。
Assembled Last	システムディレクトリソースモジュールが最後にアセンブルされた日時。
CICS System ID	CICS リージョンの ID。
使用可能なリソース： スワッププール ローカルバッファプール ソートバッファプール DL/I バッファプール 編集バッファプール モニタプール RNM バッファプール	現在の NCI システム環境で使用可能なリソース：スワッププール、Natural バッファプール、モニタバッファプール、および RNM バッファプール。  サポートされているすべてのバッファプールのタイプ、サイズ (KB)、および場所 (16 MB 境界より下または上)。
Max Thread Size	すべての有効なスレッドグループでの最大スレッドサイズ。
VSAM Roll Files	VSAM ロールファイルの数。
Main/Aux TempStor	CICS 主要または補助一時ストレージが Natural/CICS ロール機能に対して使用可能かどうかを示します。
Session Logging	Natural/CICS ログの保存先が CICS DCT (保存先制御テーブル) で定義されているかどうか、また、ログの保存先が使用可能かどうかを示します。セッションのログ保存先は、NCMPRM マクロの LOGDEST パラメータで定義されます。
Message Logging	Natural/CICS エラーメッセージログの保存先が CICS DCT で定義されているかどうか、また、ログの保存先が使用可能かどうかを示します。  メッセージのログ保存先は、NCMPRM マクロの MSGDEST パラメータで定義されます。
Message Switching	メッセージ切り替えトランザクション ID が CICS で定義されているかどうか、また、トランザクション ID が使用可能かどうかを示します。メッセー

フィールド	説明
	<p>ジ切り替えのトランザクションは、NCMPRM マクロの MSGTRAN パラメータで定義されます。</p> <p>このトランザクション ID が使用可能でない場合、SYSTP セッションフラッシュ（「Natural ユーザーセッション」の「セッションのフラッシュ」を参照）は使用できません。</p>
Console Terminal	Natural/CICS 用の CICS コンソールが使用可能かどうかを示します。コンソール端末 ID は、NCMPRM マクロの CONSOLE* パラメータで定義されます。

\* 『TP モニタインターフェイス』ドキュメントを参照してください。

## NCI 生成オプション

この機能は、CICS で稼働している Natural の生成パラメータ設定を表示するために使用します。これらのパラメータの値は、インストール時に作成される NCIPARM パラメータモジュールの一部であるマクロ NCMPRM 内で決定されます。

この機能呼び出すと、現在の CICS リージョンの [**Generation Options**] 画面が表示されます（画面タイトルの CICS ID で示されます）。この画面には、Natural の生成オプション設定の概要が表示されます。

[**Generation Options**] 画面の各パラメータ設定の背後には、NCMPRM マクロのパラメータがあります。これらのパラメータ名は、PF10 キーを押すことで表示できます。パラメータ名の画面とパラメータ説明の画面を切り替えるには、PF10 キーを押します。

関連トピック：

- 『インストール』ドキュメントの「Natural CICS インターフェイスのインストール」
- 『TP モニタインターフェイス』ドキュメントの「NCMPRM マクロパラメータ」

## Natural スレッドグループ定義

この機能は、Natural スレッドグループ定義を表示するために使用します。

この機能呼び出すと、現在の CICS リージョンの [Natural Thread Group Definitions] 画面が表示されます（画面タイトルの CICS ID で示されます）。この画面には、次の情報が表示されます。

列	説明								
Grp No.	スレッドグループ番号。								
Group Type	グループ定義のタイプ： <table border="1" data-bbox="500 535 1472 926"> <tr> <td>SHR</td> <td>スレッドグループに使用する常駐ストレージスレッド。</td> </tr> <tr> <td>GETM</td> <td>GETMAIN を使用して割り当てられるストレージスレッドです。</td> </tr> <tr> <td>なし</td> <td>使用するスレッドはありません。すべての Natural ストレージ要求が CICS に渡されます。</td> </tr> <tr> <td>Alias</td> <td>トランザクション ID / タスク要求キーによってトリガされる他のプライマリロール機能を割り当てるためのスレッドグループ再定義。</td> </tr> </table>	SHR	スレッドグループに使用する常駐ストレージスレッド。	GETM	GETMAIN を使用して割り当てられるストレージスレッドです。	なし	使用するスレッドはありません。すべての Natural ストレージ要求が CICS に渡されます。	Alias	トランザクション ID / タスク要求キーによってトリガされる他のプライマリロール機能を割り当てるためのスレッドグループ再定義。
SHR	スレッドグループに使用する常駐ストレージスレッド。								
GETM	GETMAIN を使用して割り当てられるストレージスレッドです。								
なし	使用するスレッドはありません。すべての Natural ストレージ要求が CICS に渡されます。								
Alias	トランザクション ID / タスク要求キーによってトリガされる他のプライマリロール機能を割り当てるためのスレッドグループ再定義。								
Roll Fac.	割り当てられたプライマリロール機能： VSAM、Aux（補助の一時ストレージ）、Main（メインの一時ストレージ）、または none（割り当てられたロール機能なし）。								
Thread Size	スレッドストレージ GETMAIN サイズ（スレッドグループタイプ GETM および SHR の場合）。								
TCBs	このスレッドグループで同時にアクティブになる Natural セッションの最大数。								
Transaction IDs / Task Request Keys	Natural の CICS トランザクション定義内に定義されています。								

## Natural スレッドグループ定義のコマンド

[Natural Thread Group Definitions] 画面では、「[SYSTP ユーティリティ画面の使用](#)」で説明しているコマンドの他に、次の PF キーおよび対応する行コマンドを使用できます。

PF キー	行コマンド	機能
PF4	S	カーソル / コマンドでマークされたスレッドグループに対するスレッドグループ定義を表示します。
PF10	G	カーソル / コマンドでマークされたスレッドグループに関連付けられた <b>Natural ストレージスレッド</b> （関連セクションを参照）を表示します。

## 固有の Natural ユーザーセッション

---

この機能は、「[Natural ユーザーセッション統計](#)」で説明されている [Natural User Session Statistics] 画面を呼び出します。

## CICS タスク情報

---

この機能は、CICS 環境での Natural タスクに関するステータス情報を表示する [SYSTP Task Information] ウィンドウを呼び出します。

## システム管理機能

---

この機能は、デバッグとトレースの機能にアクセスするために使用します。

この機能呼び出すと、次の機能を選択できるメニューが表示されます。

- [トレース機能](#)
- [デバッグ機能](#)
- [ロギング用のシステムスナップショット](#)
- [システムハイウォーターマークのリセット](#)
- [共通ダイナミックパラメータ制御情報](#)
- [適用された NCI ソース変更](#)
- [適用された NCI ZAP](#)

### トレース機能

この機能は、Software AG の担当者の内部使用のために予約されています。

### デバッグ機能

この機能は、Software AG の担当者の内部使用のために予約されています。



## ロギング用のシステムスナップショット

この機能は、（使用されたかどうかに関係なく）すべての SCP 機能に関する情報を示す完全な SYSTP バッチレポート（「**バッチでの SYSTP (CICS セッション用)**」も参照）を提供します。このような機能を次に示します。

- スレッドグループ
- TYPE=SHR スレッド
- ロール機能

このすべての情報が、Natural/CICS ログファイルに記録されます（使用可能な場合）。

## システムハイウォーターマークのリセット

この機能は、前に説明したシステムスナップショット機能を構成します。また、すべてのシステムハイウォーターマークをリセットできます。次に例を示します。

- ユーザーセッションの数
- すべてのスレッドグループおよびロール機能
- UCB ブロックエクステンションの数
- ストレージの量
- すべてのスレッドグループと TYPE=SHR スレッド
- すべての待機キューの値とカウント
- すべてのロール機能のロールカウント

## 共通ダイナミックパラメータ制御情報

この機能は、PRMDEST 宛先から取得した共通ダイナミックプロファイルパラメータを表示します（使用可能な場合）。『TP モニタインターフェイス』ドキュメントの「PRMDEST - Natural CICS プロファイルパラメータの入力先の名前」も参照してください。

## 適用された NCI ソース変更

この機能は、現在の CICS リージョンの [Applied NCI Source Changes] 画面を呼び出します（画面タイトルの CICS ID で示されます）。この画面には、現在の Natural TP 環境に適用されたすべてのソース変更の数が表示されます。

## 適用された NCI ZAP

この機能は、現在の CICS リージョンの [Applied NCI Zaps] 画面を呼び出します（画面タイトルの CICS ID で示されます）。この画面には、現在の Natural TP 環境に適用されたすべての ZAP の数が表示されます。

# 118

## IMS/TM での SYSTP 機能

---

■ ブロードキャスト .....	814
■ 環境データの表示 .....	814
■ モニタリング .....	815
■ 複数セッション .....	815
■ 適用された NII ZAP .....	815

SYSTP ユーティリティは、IMS/TM に固有の機能を提供します。

### ▶手順 118.1. IMS/TM で特定の SYSTP 機能呼び出すには

- SYSTP の [Main Menu] の [Code] フィールドに、[Environment-Dependent Functions] に対応する「E」を入力します。

続いて表示される [NII Menu] メニューから、このセクションで説明している機能を選択できます。

- 📄 **注意:** このセクションの以降の部分では、Natural IMS/TM インターフェイスは NII とも記載されています。

## ブロードキャスト

---

この機能は、IMS 環境の特定のユーザーグループにメッセージをブロードキャストするために使用します。

この機能呼び出すと、次の機能を選択できる [Broadcasting Menu] が表示されます。

- ブロードキャストメッセージの作成
- すべてのブロードキャストメッセージの一覧表示

ブロードキャスト機能の詳細については、『TP モニタインターフェイス』ドキュメントの「IMS/TM 環境の Natural - 特別な機能」を参照してください。

## 環境データの表示

---

この機能は、Natural IMS/TM に関する環境データを表示するために使用します。

この機能呼び出すと、現在の Natural セッションで使用される環境テーブルに関する [Environment Table] 画面が表示されます。画面には、Natural IMS/TM インターフェイスの現在のパラメータ設定が表示されます。

パラメータは更新できません。IMS パラメータの詳細については、『TP モニタインターフェイス』ドキュメントの「IMS/TM 環境での Natural」を参照してください

---

## モニタリング

---

この機能は、同じ Natural サブシステムで稼働する Natural ユーザーセッションに関するモニタリングデータを表示するために使用します。

この機能呼び出すと、ユーザーセッションに関するモニタリングデータを表示するために次の機能を選択できる **[Monitoring]** 画面が表示されます。

- **Active Sessions**

IMS/TM で稼働するすべてのアクティブな Natural セッションが表示されます。

- **Suspended Sessions**

IMS/TM で現在中断されているすべての Natural セッションが表示されます。

- **User Selection**

特定の Natural ユーザーセッションのみ表示するために選択条件を指定できるウィンドウが呼び出されます。

---

## 複数セッション

---

この機能は、複数の Natural セッションを表示または作成するために使用します。

この機能呼び出すと、すべてのアクティブなセッションが一覧表示される **[IMS Multi Session]** 画面が表示されます。また、作成キーと再開キーの設定が表示されます。『TP モニタインターフェイス』ドキュメントの「マルチセッション機能の働き」を参照してください。

---

## 適用された NII ZAP

---

この機能は、現在の Natural TP 環境に適用されているすべての ZAP の数を表示する **[Applied NII Zaps]** 画面を呼び出します。



# 119

## TIAM および UTM での SYSTP 機能

---

- P-Key ユーティリティ ..... 818
- 共通メモリアンプールの表示 ..... 822

SYSTP ユーティリティは、TIAM および UTM に固有の機能を提供します。

▶手順 119.1. TIAM および UTM で特定の SYSTP 機能呼び出すには

- SYSTPの [Main Menu] の [Code] フィールドに、 [Environment-Dependent Functions] に対応する「E」を入力します。

続いて表示されるメニューから、このセクションで説明している機能を選択できます。

## P-Key ユーティリティ

この機能は、TP モニタ TIAM および UTM のもとで Siemens 975X 端末のプログラム可能な P キーのロードをサポートします。

標準 Natural キー設定（ファンクションキーモード KN、K0、または KS）をキー P1~P20 にロードするか、ユーザー定義値を個々のキーにロードできます。『オペレーション』ドキュメントの「9750 デバイスのファンクションキーサポート」も参照してください。

この機能は、次のメニューを選択できる [P-Key Utility] メニューを呼び出します。

```

15:54:05          ***** NATURAL SYSTP UTILITY *****          1998-03-25
User VR000001          - P-Key Utility -          TID 0709

          Code      Function          Parameter

          KU      Load User Values    A,H
          KS      Set KS Mode          L,N
          KN      Set KN Mode          L,N
          K0      Set K0 Mode          L,N
          KF      Load F1 - F20
          ?      Help
          .      Exit

          Code .. __          Parameter A

Select function.
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
          Help Menu Exit KU   KS   KSN  KN   KNN  K0   KON  KF   Canc
    
```





```

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Menu  Exit  Load          -          +          Canc

```

【Mode】フィールドは、機能呼び出したときに指定したパラメータに応じて HEX または ALPHA に設定されます。モードを切り替えるには、現在の値を A (ALPHA の場合) または H (HEX の場合) で置き換えます。


- ALPHA モードでは、画面の左半分を使用して、ロードするキーの横に英数字値を入力できます。
- HEX モードでは、画面の右半分に 16 進形式でキーに値を割り当てることもできます。

各 P キーについて、空の入力フィールドに英数字値を入力するか、その下の行に 16 進値を入力します (【Parameter】値が A の場合、16 進数フィールドは入力がブロックされます)。

キーに対して値が指定されていない場合は、標準の Natural キー設定 (ファンクションキーモード KN、KO または KS) がこのキーに適用されます。したがって、混在する P キーの使用が可能です。つまり、一部のキーがユーザー定義機能を持ち、その他のキーが標準の Natural 機能を持つことができます。

PF4 キーを押すか 【Function】フィールドに「L」と入力して、値をロードします。

PF8 キーを押すか 【Function】フィールドにプラス (+) 記号を入力して、追加の P キーに画面を移動します。

 **注意:** Natural は、H'40'~H'6F' (= 疑問符) より小さいすべてのバイナリ値を自動的に変換します。したがって、H'40' より小さいバイナリ値をロードできるようにするには、その前に、マクロ NTTAB (変換テーブル) を変更してこの自動変換を回避する必要があります。この操作は、H'27' (= ESCAPE) および H'19' (= Endemarke) で特に重要です。詳細については、『パラメータリファレンス』ドキュメントの「TAB - 標準出力文字変換」を参照してください。

## LPFSUP01 インターフェイスを使用したユーザー値のロード

ユーザー値のロード機能は、アプリケーションプログラミングインターフェイス (API) としてユーザーアプリケーションでも使用できます。API は、キーのロードを実行する Natural サブプログラム LPFSUP01 で構成されます。LPFSUP01 は、システムライブラリ SYSEXTP で提供され、ユーザーライブラリまたは steplib にコピーできます。

## ▶手順 119.2. LPFSUP01 を呼び出すには

- 次のステートメントを発行します。

```
CALLNAT 'LPFSUP01' P-VALUE(*)
```

ここで、P-VALUE は配列 (A24/20) として定義する必要があります。

例：

```
DEFINE DATA LOCAL
  1 P-VALUE (A24/20)
END-DEFINE
* LOAD '/STA L EM DUE1' TO P1, '/STA P EM DUE1' TO P4
COMPRESS '/STA L' h'192786' INTO P-VALUE(1)
COMPRESS '/STA P' h'192786' INTO P-VALUE(4)
CALLNAT 'LPFSUP01' P-VALUE(*)
END
```

システムライブラリ SYSEXTP にあるプログラム例 LPFEXAM1 も参照してください。

## キー割り当てモードの設定

次の機能は、Siemens 端末でキー割り当てを設定するために使用します。

モード	機能
KS モードの設定	端末コマンド %KS * を実行します。PF5 キーを押すか、[P-Key Utility] メニューでファンクションコード「S」を入力することで呼び出されます。
KN モードの設定	端末コマンド %KN * を実行します。PF7 キーを押すか、[P-Key Utility] メニューでファンクションコード「N」を入力することで呼び出されます。
KO モードの設定	端末コマンド %K0 * を実行します。PF9 キーを押すか、[P-Key Utility] メニューでファンクションコード「0」を入力することで呼び出されます。

\* 『端末コマンド』ドキュメントを参照してください。

キー割り当てモードの詳細については、『オペレーション』ドキュメントの「BS2000/OSD 環境での Natural」を参照してください。

## P キーへの送信キーコードのロード

F1～F20のロード機能は、特定の送信キー（F）コード F1～F20 をキー P1～P20 にロードするために使用します。この機能は、F コードを個別に選択できることを除き、キー割り当てモード KN と同様です。

この機能が呼び出されると、次の画面が表示されます。

```

15:56:34          ***** NATURAL SYSTP UTILITY *****          1998-06-25
User VR000001          - Load F-Codes -          TID VR000001

P01  _           P02  _           P03  _           P04  _           P05  _

P06  _           P07  _           P08  _           P09  _           P10  _

P11  _           P12  _           P13  _           P14  _           P15  _

P16  _           P17  _           P18  _           P19  _           P20  _

Mark P-Key to be loaded with F-Code
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Load          Menu  Exit                                     Canc

```

PキーをFコードとともにロードするには、適切なキーをマークし、Enterキーを押します。マークされているキーだけがFコードとともに呼び出されます。その他のPキーは、元の値を保持します。

## 共通メモリアールの表示

この機能には、Natural で使用されるすべての共通メモリアールのリストが表示されます。

各共通メモリアールについて表示される情報の各項目については、この機能のオンラインヘルプを参照してください。

# 120 バッチでの SYSTP (CICS セッション用)

---

- バッチモードでの SYSTP の呼び出し ..... 824
- ログファイルの評価 ..... 824

また、SYSTP ユーティリティは、バッチモードでの Natural/CICS セッションに関する統計データの入手にも使用できます。

Natural/CICS セッションに関する統計データが書き込まれる Natural ログファイルは、Natural バッチジョブにワークファイル 1 として（つまり、CMWKF01 経由で）割り当てする必要があります。また、このファイルは、オンラインシステム、つまり、CICSDCT（送信先制御テーブル）内にも定義する必要があります。『TP モニタインターフェイス』ドキュメントの NCMPRM マクロの LOGDEST パラメータを参照してください。

## バッチモードでの SYSTP の呼び出し

---

### ▶手順 120.1. SYSTP ユーティリティをバッチモードで呼び出すには

- バッチジョブで、次のいずれかのコマンドを指定します。

```
SYSTP xxx
```

または

```
LOGON SYSTP  
SYSBATCH xxx
```

xxx は、処理するデータの種類を示します。例えば、xxx=nci は、データが Natural/CICS オンラインシステムによって収集されることを指定します。

## ログファイルの評価

---

データは、Natural が初期化またはリセットされたとき、および Natural セッションが終了したときに Natural ログファイルに書き込まれます。

Natural CICS インターフェイスは、次のレコードを Natural ログファイルに書き込みます。

- Natural 環境が初期化またはリセットされたときの開始ログレコード
- Natural セッションが終了したときのセッションログレコード


Natural 環境が初期化されると、システム ID がシステムコントロールブロックに書き込まれます。このシステム ID は、すべてのログレコードにも属します。したがって、Natural ログファイルは、複数の Natural/CICS オンライン環境で共有できます。

ログに記録された情報は、Natural/CICS オンライン環境の使用状況の追跡に使用されます。したがって、ほとんどの情報は、Natural 環境の機能を表します。ログファイルは、CICS の機能を参照する計算またはモニタリングツールにすることは意図していません。

システム ID に基づいて、Natural セッションに関連するデータとともに複数のレポートが作成されます。

- 時間順に表示されているログファイルデータ。これは、セッションログレコードがセッション終了日時でソートされることを意味します。
- Natural 環境の設定および使用方法に関する統計
- スレッドグループに関する統計（使用されている場合）
- プログラムストレージスレッドに関する統計（使用されている場合）
- ロール機能に関する統計（使用されている場合）

すべての Natural 環境に対して、このレポートのセットが Natural ログファイルの Natural/CICS セッションに関するレコードとともに作成されます。

 **注意:** セッション終了ログレコードは、当然、対応するセッションで使用されていたリソースのみを反映します。したがって、これらのレコードは、完全な Natural 環境を反映しない場合があります。完全な Natural 環境のレポートは、[システム管理機能](#)（関連セクションを参照）を使用して Natural 環境全体のスナップショットを作成することで取得できます。





# 索引

---

## ゆ

ユーティリティ  
概要, 1

