

Adabas

ユーザー、ハイパーディスクリプタおよび照合ディスクリプタ出口

バージョン 8.1.3

June 2008

This document applies to Adabas Version 8.1.3 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 1971-2008. 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 ユーザー、ハイパーディスクリプタおよび照合ディスクリプタ出口	1
2 ユーザー出口 1 (一般処理)	3
3 ユーザー出口 2 (デュアルログ処理)	5
ユーザー出口 2 コールの処理フロー	7
入力パラメータ	8
出力パラメータ	10
BS2000 ユーザー出口 2 の起動オプション	10
4 ユーザー出口 3 (ユーザー定義のフォネティック処理)	13
入力パラメータ	14
5 ユーザー出口 4 (ユーザー生成のログデータ)	15
コマンドログフォーマット	16
6 ユーザー出口 5 (Adabas Review ハブイベントハンドラ)	19
入力パラメータ	20
出力パラメータ	21
7 ユーザー出口 6 (データ圧縮前のユーザー処理)	23
ADACMP のヘッダー処理	25
入力パラメータ	26
出力パラメータ	26
8 ユーザー出口 8 (オペレータインターフェイス)	27
入力パラメータ	29
9 ユーザー出口 9 (ADAULD)	31
10 ユーザー出口 11 (一般処理)	33
入出力パラメータ	35
11 ユーザー出口 12 (マルチデータセットログの処理)	37
ユーザー出口 12 コールの処理フロー	39
ユーザー出口インターフェイス	40
出力パラメータ	43
サンプルのユーザー出口のアクティブ化	43
12 ハイパーディスクリプタ出口 01 - 31	49
メインパラメータエリア	51
入力パラメータエリア (3 番目のパラメータアドレスのポイント先)	52
出力パラメータエリア	54
空値オプション	55
ハイパー出口初期化コール	56
ハイパー出口スタブ	56
13 照合ディスクリプタ出口 01~08	59
照合ディスクリプタ出口のインターフェイス	60
目次	63

1 ユーザー、ハイパーディスクリプタおよび照合ディスクリプタ出口

このdocumentでは、ADARUN パラメータの UEXn、HEXnn、および CDXnn を使用してアクティブにするユーザー出口について説明します（ADARUN パラメータについては『Adabas オペレーションマニュアル』を参照）。

このドキュメントで説明するユーザー出口は次のとおりです。

ユーザー出口	ADARUN	使用法
ユーザー出口 1	UEX1	コマンド処理（Adabas ニュークリアス）。この出口はサポートされなくなりました。
ユーザー出口 2	UEX2	デュアルログの処理
ユーザー出口 3	UEX3	ユーザー定義のフォネティック処理
ユーザー出口 4	UEX4	ユーザー生成のログデータ
ユーザー出口 5	UEX5	Adabas Review ハブイベントコントローラ
ユーザー出口 6	UEX6	データ圧縮（ADACMP）
ユーザー出口 8	UEX8	オペレータのインターフェイス
ユーザー出口 9	UEX9	データのアンロード（ADAULD）
ユーザー出口 11	UEX11	コマンド処理（Adabas ニュークリアス）。新しく追加されたユーザー出口です。
ユーザー出口 12	UEX12	マルチログの処理

ユーザー、ハイパーディスクリプタおよび照合ディスクリプタ出口

ハイパーディスクリプタ出口	ADARUN	使用法
1..31	HEX01 .. HEX31	ハイパーインデックス値作成に関するユーザー設定アルゴリズム

照合ディスクリプタ出口	ADARUN	使用法
1..8	CDX01 .. CDX08	対応する照合ディスクリプタの値のエンコードおよびデコードに関するユーザー設定アルゴリズム

Adabas その他の出口

次に示すその他のユーザー出口もサポートされています。

エントリ名	使用法
ADACDCUX	ADACDC ユーティリティ処理中に特定の時点で制御を取得できます。『Adabas ユーティリティマニュアル』を参照してください。
ADACSHUX	Adabas Caching Facility 処理中に特定の時点で制御を取得することができます。Adabas Caching Facility のドキュメントを参照してください。
ADALERTX	ハートビートレコードを調査したり、その調査に使うしきい値と間隔を変更したりできます。『Adabas DBA リファレンスマニュアル』の「Adabas Online System デモバージョン」を参照してください。
ADASMXIT	Adabas が提供するもの以外のリカバリアクションを実装できるように、PIN ルーチンのパラメータに値を指定するか、または検出された条件を PIN ルーチンが起動される前に検査します。『Adabas DBA リファレンスマニュアル』の「Adabas Online System デモバージョン」を参照してください。
DSFEX1	DLOG エリアがオーバーフローしないように、必要なジョブを自動的にサブミットします。『Adabas Delta Save Facility マニュアル』を参照してください。
UEXITA	Adabas 7 の ADALINK にリンクし、ターゲット、ルーター、または ADALINK 自体がコマンドを処理した後に、制御を受け取ります。『Adabas インストールマニュアル』を参照してください。
UEXITB	Adabas 7 の ADALINK にリンクし、コマンドがルーター 04 コールでターゲットに渡される前に制御を受け取ります。『Adabas インストールマニュアル』を参照してください。
UEXIT1	Adabas 8 の ADALINK にリンクし、コマンドがルーター 04 コールでターゲットに渡される前に制御を受け取ります。『Adabas インストールマニュアル』を参照してください。
UEXIT2	Adabas 8 の ADALINK にリンクし、ターゲット、ルーター、または ADALINK 自体がコマンドを処理した後に、制御を受け取ります。『Adabas インストールマニュアル』を参照してください。
UEXRAI	自動的に生成される ADARAI RECOVER JCL が DDJ CLOUT に書き込まれる前に変更できるようにします。『Adabas ユーティリティマニュアル』を参照してください。

2 ユーザー出口1 (一般処理)

ユーザー出口11の導入に伴い、ユーザー出口1のサポートはなくなりました。ただし、移行を簡単にするため、サンプルユーザー出口 UEX11UX1 が提供され、既存のユーザー出口1の前に挿入してユーザー出口11として起動させることができます。このサンプルは、ACB ダイレクトコールインターフェイスを使用して行われるダイレクトコールでのみ動作します。Adabas 8 ACBX ダイレクトコールインターフェイスを使用したダイレクトコールでは動作しません。

「[ユーザー出口11](#)」で説明しているように、このユーザー出口は、ユーザー出口11の制約を受けます。特に、変更可能なフィールドは、ファイル番号 (CQXFNR)、アディクション2 (ACBADD2)、アディクション3 (ACBADD3)、およびユーザーエリア (ACBUSER) の各フィールドに制限されます。他のACBフィールドを変更しても、CQXに他の変更を加えても、ニュークリアスからは無視されます。詳細は、サンプルユーザー出口内のコメントを参照してください。既存のユーザー出口1とのリンクの方法などが記述されています。Adabas ニュークリアスの、この移行支援サポートは将来の Adabas リリースでは廃止される予定です。

3 ユーザー出口2 (デュアルログ処理)

- ユーザー出口2 コールの処理フロー 7
- 入力パラメータ 8
- 出力パラメータ 10
- BS2000 ユーザー出口2 の起動オプション 10

ユーザー出口2 (デュアルログ処理)

このユーザー出口は、デュアルログの切り替え時に Adabas ニュークリアスから制御を受け取ります。これは、切り替える前のログを Adabas が再び使用する前に、そのログをコピーすることが目的です。この切り替えは、デュアルデータプロテクションログまたはデュアルコマンドログがセッションで有効な場合にのみ実行されます。

ADARES ユーティリティの該当する機能 (CLCOPY または PLCOPY) を実行するには、ユーザー出口ルーチンからプロシージャを起動する必要があります。

ユーザー出口2 は、次の場合に起動されます。

- PLOG/CLOG をコピーする必要がある場合に、Adabas ニュークリアスが起動されたとき
- 2つのログデータセットの間で、デュアルコマンドログとデュアルプロテクションログの切り替えが発生したとき
- Adabas ニュークリアスのシャットダウン中

ユーザー出口には、デュアルログデータセットのステータス情報が渡されます。

ユーザー出口は、次のうちいずれかの処理を実行できます。

- コールを無視する
- 格納スペースがなくなったログデータセットをコピーするためのジョブ (ADARES ユーティリティ) をサブミットする
- サブミットしたコピージョブが完了するまで待機する

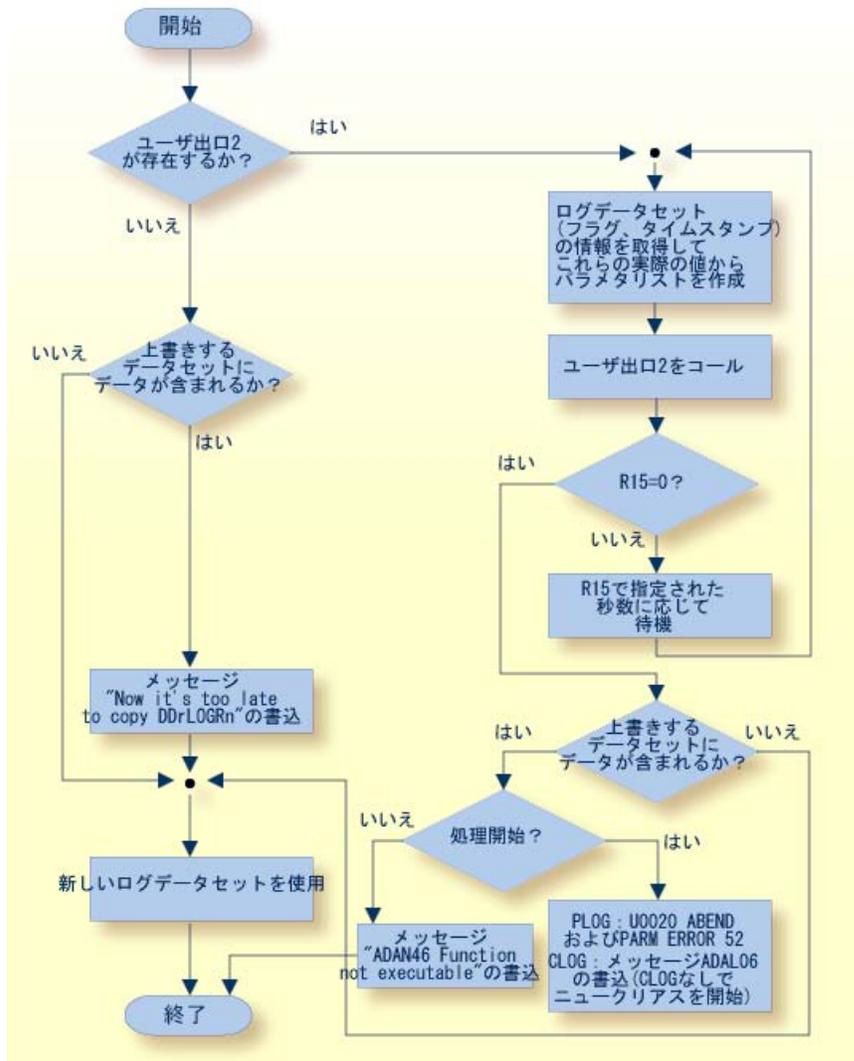
上書きするデータセットにデータが存在する場合は、"ADAN46 Function not executable" というコンソールメッセージが発行されます。

ユーザー出口2 のサンプルは、Adabas のインストール手順を実行すると作成されます。詳細は、『*Adabas* インストールマニュアル』を参照してください。

ユーザー出口が、標準の BASR 14,15 アセンブラ命令を使用してコールされます。すべてのレジスタは、制御が受け取られときに保存され、Adabas に制御を返す直前にリストアされる必要があります。「[出力パラメータ](#)」で説明しているように、レジスタ 15 にはアクションコードが格納されます。

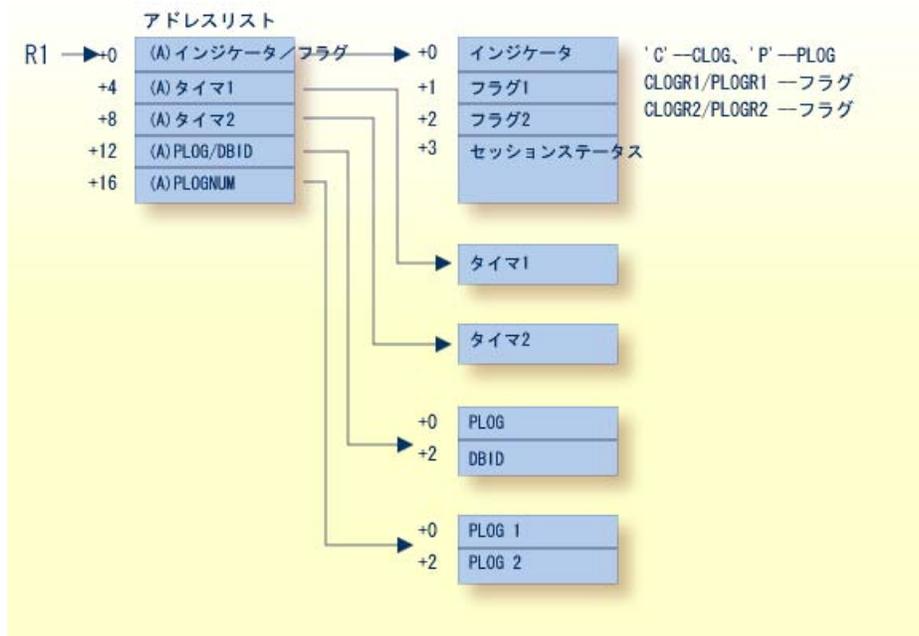
このchapterでは、次のトピックについて説明します。

ユーザー出口2 コールの処理フロー



デュアルログ処理フロー

入力パラメータ



デュアルログ処理ユーザー出口(2)パラメータ

アドレスリストの入力パラメータは次のとおりです。

パラメータ	フルワードアドレスの内容
0 (R1)	C/PLOG の識別情報およびフラグ 1/2。
4 (R1)	4 バイトのタイマ 1 フィールド。
8 (R1)	4 バイトのタイマ 2 フィールド。
12 (R1)	現行セッションの PLOG 番号と、それに続けてデータベース ID。
16 (R1)	4 バイト長のエリアで、最初の 2 バイトには PLOG1 の番号が格納され、次の 2 バイトには PLOG2 の番号が格納されます。

その他の入力パラメータを次の表に示します。

パラメータ	使用方法	
フラグ 1	DD/PLOGR1 および DD/CLOGR1 のステータスフラグ。	
フラグ 2	DD/PLOGR2 および DD/CLOGR2 のステータスフラグ。フラグの内容は次のとおり：	
	B'1...':	データセットはニュークリアスによる上書き処理中
	B'.1..':	ニュークリアスのデータセット上書き処理が完了
	B'.11.':	ADARES によるコピー処理中
	B'0000 0000':	データセットは空 (つまりコピー済み) であり、ニュークリアスによる再使用が可能な状態
	その他のフラグ 1/2 フィールド値は予約されています。DD/CLOGR1/2 については、CLOGLAYOUT=5 の場合に X'08' になります。フラグの 1 ビット目と 2 ビット目の設定は組み合わせることができ、例えば、X'40'、X'20' を組み合わせて X'60' と指定することができます。	
OPENOUT が指定されている場合、これらのフラグは、OPEN が出力データセットに発行された後に設定されます。OPENOUT が指定されていない場合、フラグの設定は、OPEN の発行前になります。		
セッションステータス	出口がコールされたときのニュークリアスのステータス情報を格納します。	
	X'S'	ニュークリアスセッションの起動時にコールされたことを示します。
	X'T'	ニュークリアスセッションの終了時にコールされたことを示します。
	X'W'	デュアルプロテクションログの切り替え時にコールされたことを示します。
TIMERn	ログデータセットの最初のブロックが書き込まれたときのタイムスタンプ (最长で4バイトの STCK 命令)。DD/PLOGR1 および DD/CLOGR1 の場合は TIMER1、DD/PLOGR2 および DD/CLOGR2 の場合は TIMER2 になります。	
PLOG	現行セッションのプロテクションログ番号 (2 バイト)。この値は PLOG の場合にだけ設定されます。CLOG の場合は X'00' がフィールドに格納されます。	
DBID	データベース ID (2 バイト)。	
PLOG1/2	PLOG1 と PLOG2 は、PLOG 番号 (それぞれ 2 バイト、合計で 4 バイト) を使用して指定することができます。前回のニュークリアスセッションが異常終了した場合、この 4 バイトには、そのセッションの PLOGNUM 値が格納されるため、この値を最初のユーザー出口2 コールで使用すれば、そのセッションの PLOG をコピーすることができます。後続のセッションでは、これらのバイトに現在の PLOGNUM 値が格納されます。先行のセッションが異常終了した場合には、ニュークリアス起動フェーズ中に、終了したセッションの PLOG 番号がこの 4 バイトに格納されます。この PLOG 情報は、正しい PLOG 番号をコピー先の PLOG エリアに割り当てるために起動フェーズで必要になります。その後、出口がコールされると、最新の PLOG 値がこれらのフィールドに格納されます。	

出力パラメータ

パラメータ	使用方法
R15 = 0	ニュークリアスは処理を続行します。
R15 > 0	R15の値は、ユーザー出口2を再度コールするまでの待機時間の秒数として扱われます。この秒数の間、ニュークリアスは割り込み禁止の待機状態になります。この待機時間の間は、どのコマンドも処理されません。

BS2000 ユーザー出口2の起動オプション

BS2000システムのユーザー出口2を使用している場合には、従来のリリースと異なり、ADARES CLCOPY または PLCOPY ジョブの名前はユーザー出口にハードコードされていません。したがって、次に説明するように、/ENTER ジョブを指定する方法が2つあります。

方法1

完全な /ENTER ジョブコマンドを含むジョブ変数を作成します。例としては、次のようなものがあります。

```
/DCLJV ENTER,ADARES,PLCOPY,LINK=*DDJBPLC
/SETJV *DDJBxLC,C'ENTER JOB,ENTER,ADARES,PLCOPY,ST=IMM'
```

/ENTER コマンドのすべてのオペランドを使用できます。/DCLJV ステートメントを指定すると、他の指定よりも優先されます。必要な場合は、ニュークリアスセッション中にジョブ変数の内容を変更できます。

リンク名 *DDJBPLC を使用して ADARES PLCOPY ジョブを割り当て、*DDJBCLC を使用して ADARES CLCOPY ジョブを割り当てます。

方法2

JV 指定を省略します。ユーザー出口2は、次のコマンドを発行するようになります。

```
'ENTER RES.E.xLCO'
```

 **Note:** BS2000 システムの場合は、ユーザー出口2は、ソース要素としてのみ供給されません。

4 ユーザー出口3 (ユーザー定義のフォネティック処理)

- 入力パラメータ 14

ユーザー出口3 (ユーザー定義のフォネティック処理)

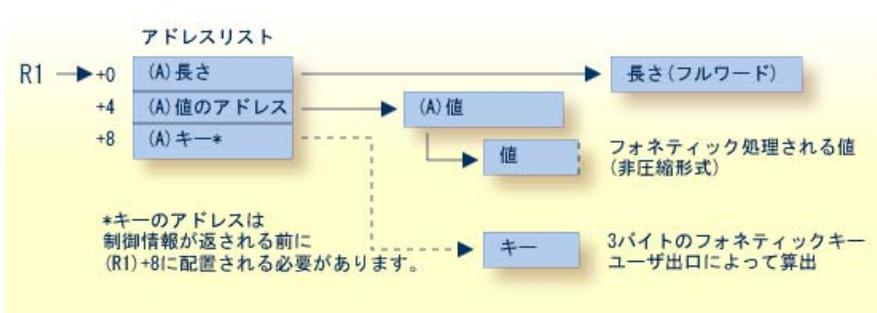
このユーザー出口は、ユーザー定義のフォネティック処理を実行するときに使用できます。このユーザー出口には、フォネティック処理が必要になるたびに、ADACMPユーティリティまたは Adabas ニュークリアスから制御が渡されます。

ユーザー出口では、指定した値から3バイトのフォネティックキーが生成されます。生成されたフォネティックキーのアドレスは、制御が戻ってくる前に8 (R1) に格納しておく必要があります。

このchapterでは、次のトピックについて説明します。

入力パラメータ

レジスタ1には、次のパラメータリストのアドレスが格納されます。



ユーザー定義フォネティックユーザ出口 (3) パラメータ

パラメータ	フルワードアドレスの内容
0 (R1)	フォネティック処理される値を格納する4バイト長の領域。
4 (R1)	フォネティック処理される値のアドレス。
8 (R1)	フォネティックキーを格納する3バイトの領域。このアドレスは、ユーザー出口の実行前にゼロに設定されます。実際のアドレスは、ユーザー出口の実行時に設定されます。

ユーザー出口が、標準のBASR 14,15 アセンブラ命令を使用してコールされます。すべてのレジスタは、制御が受け取られときに保存され、Adabasに制御を返す直前にリストアする必要があります。R15の内容は無視されます。

5 ユーザー出口4 (ユーザー生成のログデータ)

- コマンドログフォーマット 16

ユーザー出口4 (ユーザー生成のログデータ)

ユーザー出口4は、Adabas コマンドログレコードが書き込まれる直前にコールされます。このユーザー出口は、ユーザーログデータ (SMF レコード) に必要な専用の統計データを生成する場合、またはログレコードの書き込みを省略する場合に使用できます。

 **Note:** ユーザー出口4は、ADARUN LOGGING=NOが指定されていてもコールされます。ユーザー出口4を無効にする方法は、Adabas 実行から ADARUN UEX4 パラメータを削除する方法だけです。

このchapterでは、次のトピックについて説明します。

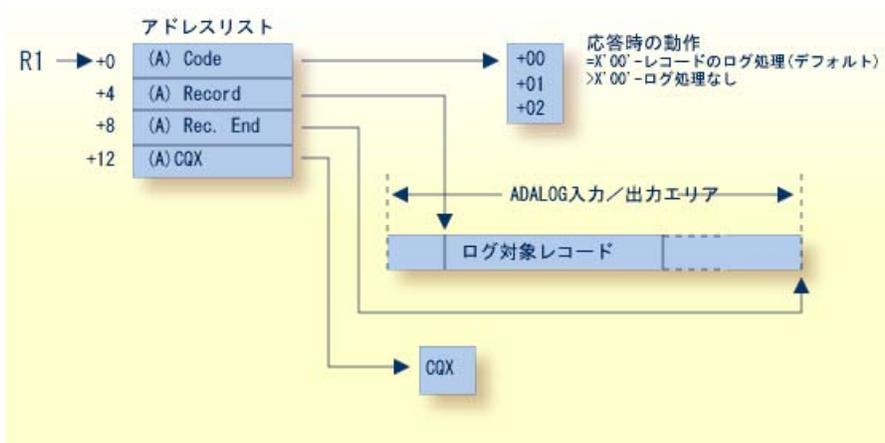
コマンドログフォーマット

Adabas では、2種類のコマンドログフォーマットがサポートされています。使用されるフォーマットは、ADARUN CLOGLAYOUT パラメータによって指定します。

- CLOGLAYOUT=5 (デフォルト) は、Adabas 5.2 以上の場合にサポートされます。
- CLOGLAYOUT=8 を指定すると新しいフォーマットが使用されますが、このフォーマットは Adabas 8 以上の場合にのみサポートされます。

どちらのフォーマットも、コマンドログフォーマットの説明に記載されています。

ユーザー出口とコマンドログの評価プログラムが、使用中のフォーマットを認識していることを確認してから、フォーマットを切り替えるようにしてください。



ユーザー生成ログデータ ユーザー出口 (4) パラメータ

パラメータ	アドレスの内容
0 (R1)	<p>ログ記録のアクションコードを格納する1バイト。このバイトのコードは次のとおりです。</p> <ul style="list-style-type: none"> ■ +00 -- コールごとにレコードを記録するためのアクションコード。ゼロ以外の値に変更すると、このレコードは DDLOG に書き込まれません。 ■ +01 -- 予約済み (将来的に使用される予定) ■ +02 -- 2 バイトのデータベース ID。
4 (R1)	記録対象のレコード。このアドレスは、ニュークリアスセッションの最後に出口がコールされた場合は 0 です。
8 (R1)	Adabas I/O エリアの終点。このアドレスは、ニュークリアスセッションの最後に出口がコールされた場合は 0 です。
12 (R1)	コマンドキューエレメント (CQX)。このアドレスは、ニュークリアスセッションの最後に出口がコールされた場合は 0 です。

記録対象のレコードは、ユーザー出口で変更することができます。また、4 (R1) のレコードのアドレスも変更することができます。ログ記録のアクションコードは、必ず Adabas ニュークリアスに制御が戻る前に指定するようにしてください。

 **Caution:** レコードを変更する場合、8 (R1) に格納されている ADALOG I/O エリアの終了アドレスを超えないようにしてください。

6 ユーザー出口5 (Adabas Review ハブイベントハンドラ)

- 入力パラメータ 20
- 出力パラメータ 21

ユーザー出口 5 は、Adabas Review ハブでイベントが発生したときに、Adabas ニュークリアスからコールされます。

ここで言うイベントとは、次のものを指します。

- Adabas セッションの開始時に Adabas Review ハブとの接続が確立された
- Adabas セッションの終了時に Adabas Review ハブとの接続が切断された
- コマンドログレコードの送信処理からゼロ以外のリターンコードが返された

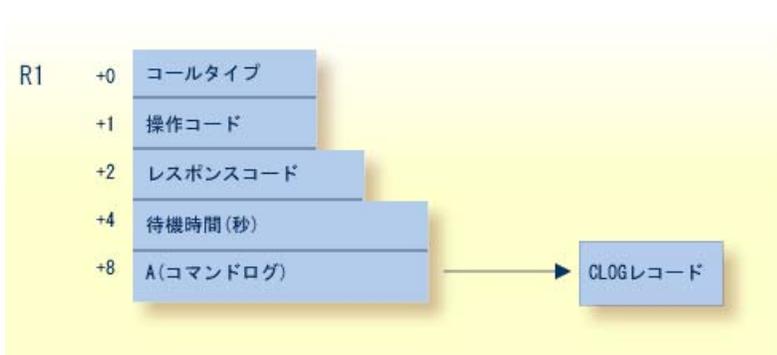
出口は、AMODE=31 の状態で起動された場合、同じ状態で制御を返す必要があります。

ロギングエラーが発生したときに何らかの対処を行う必要があるため、出口はエラー発生時にどのように対応するかを判断します。記録されなかったレコードと、Adabas Review ハブのロギング要求から返されたレスポンスコードが、エラーの対応方法を決めるうえでの判断材料となります。

このchapterでは、次のトピックについて説明します。

入力パラメータ

入力パラメータとして、レジスタ 1 は、次のパラメータリストをポイントします。



パラメータ	使用方法						
0 (R1)	<p>出口コールの状態を表します。このバイトの値は次のとおりです。</p> <table border="1"> <tr> <td>O</td> <td>Adabas Review ハブとの接続開始</td> </tr> <tr> <td>C</td> <td>Adabas Review ハブとの接続終了</td> </tr> <tr> <td>L</td> <td>Adabas Review ハブにロギングエラーを送信</td> </tr> </table>	O	Adabas Review ハブとの接続開始	C	Adabas Review ハブとの接続終了	L	Adabas Review ハブにロギングエラーを送信
O	Adabas Review ハブとの接続開始						
C	Adabas Review ハブとの接続終了						
L	Adabas Review ハブにロギングエラーを送信						
1 (R1)	<p>ロギングエラーに対処するためのアクション（開始時と終了時には無視される）。パラメータリストのこのフィールドには、ロギングエラーの対処として、出口から次の値のいずれかを渡す必要があります。</p>						

パラメータ	使用方法	
	W	指定した時間だけ待機してから再試行する
	R	ロギング処理をすぐに再試行する
	I	ロギングエラーを無視し、その結果に関わらず処理を継続する
2 (R1)	ロギングエラーのレスポンスコード。このレスポンスコードは、『Adabas メッセージおよびコードマニュアル』に記載されている Adabas レスポンスコードと同じです。	
4 (R1)	ロギングエラー発生時にロギングを再試行するまでの待機時間（秒単位）を格納します。このフィールドはフルワードです。	
8 (R1)	Adabas ニュークリアスが Adabas Review ハブに送信しようとしたコマンドログレコードのアドレス。	

入力パラメータとしての他のレジスタ値

R13	Adabas ニュークリアスルーチンをコールするためのセーブエリア
R14	Adabas ニュークリアス内のリターンアドレス
R15	出口のエントリポイントアドレス

出力パラメータ

- ロギングエラー発生時の値として、出口によって operation フィールドに値が設定される必要があります。待機の値 (W) を選択した場合、ゼロ以外の時刻の値も必要です。
- レジスタ 15 はゼロに設定する必要があります。また、その他のレジスタは元の値に戻す必要があります。

7 ユーザー出口 6 (データ圧縮前のユーザー処理)

■ ADACMP のヘッダー処理	25
■ 入力パラメータ	26
■ 出力パラメータ	26

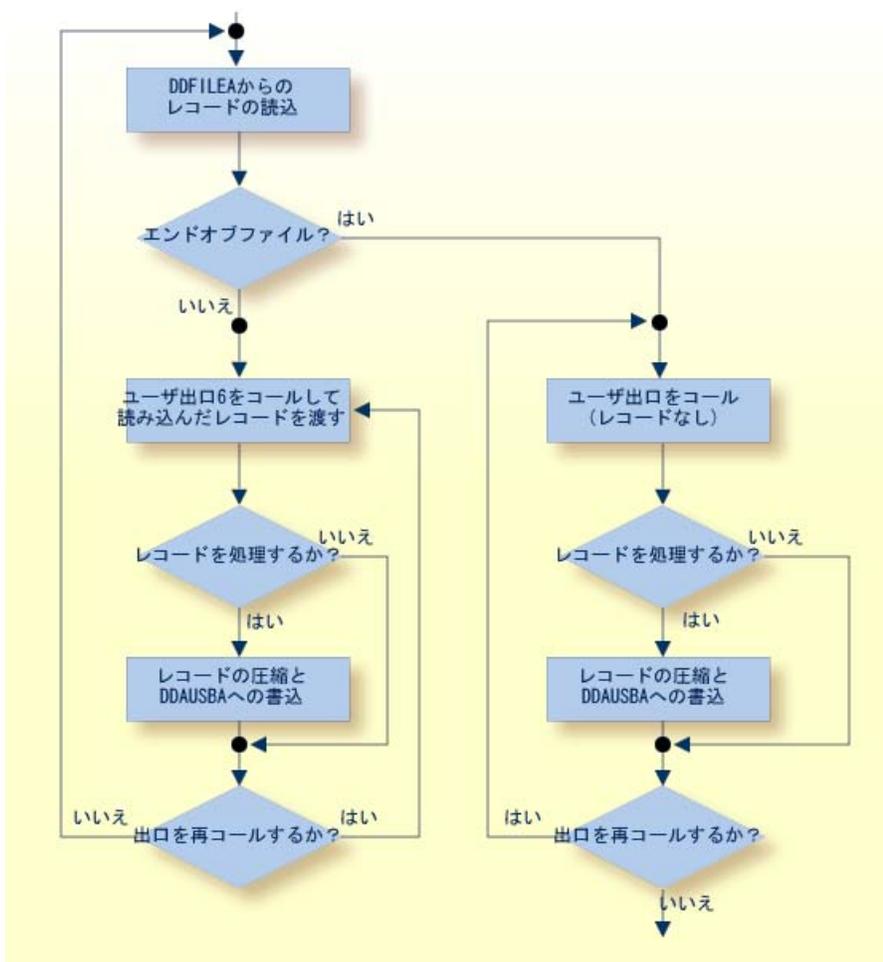
ユーザー出口6（データ圧縮前のユーザー処理）

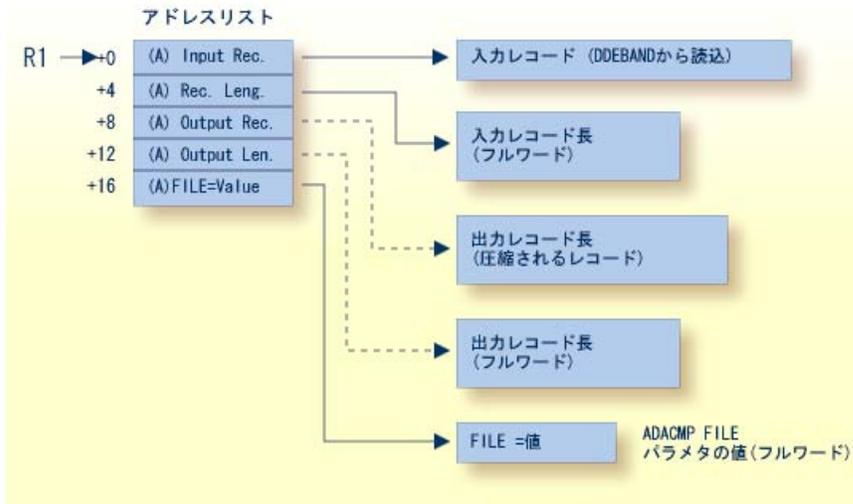
このユーザー出口を使用すると、ADACMP COMPRESS ユーティリティがレコードを処理する前に、そのレコードに対するユーザー処理を実行することができます。これは、ADACMP DECOMPRESS ユーティリティから出力された非圧縮レコードの順序および内容を制御するためにも使用できます。このように使用すると、ユーザー出口により、どの非圧縮レコードがADACMPによりDDAUSBAデータセットに書き込まれるかが制御されます。

このユーザー出口プログラムは、アランブラまたはCOBOLで記述することができます。ADACMP ユーティリティジョブ側で、ADARUN UEX6=*program parameter* を指定する必要があります。*program* はユーザープログラムの名前で、*parameter* はそのプログラムに渡すパラメータです。

ユーザー出口6は、次のイベントのいずれかが発生した直後に、ADACMP COMPRESS ユーティリティ機能からコールされます。これにより、コール時に入力にレコードを追加することができます。

- DDEBAND からレコードが読み込まれたとき。
- DDEBAND でエンドオブファイル条件が発生したとき。





ADACMP ユーザー処理 ユーザ出口 (6) パラメータ

このchapterでは、次のトピックについて説明します。

ADACMP のヘッダー処理

パラメータ HEADER=YES が指定された状態で ADACMP が実行される場合、ADACMP COMPRESS の入力レコードと ADACMP DECOMPRESS の出力レコードの先頭に、32 バイトの ADAH および ADAC ヘッダーが付加されます。このヘッダーには、複数の物理レコードのグループが論理レコードとして定義され、この場合には論理レコードの長さは 32 KB を超えることも可能です。ADAH および ADAC ヘッダーの DSECT は、付属の Adabas 8 SRCE ライブラリのメンバ ADAH および ADAC 内にあります。これらのヘッダーには、非圧縮データを含む 1 つの論理レコードがどのような物理レコードで構成されているのかが定義されています。

HEADER=YES が指定された状態で、ADACMP とユーザー出口 6 が両方とも実行されている場合、ADACMP はユーザー出口 6 に各物理レコードを渡します。したがって、ユーザー出口アプリケーション側で、これらのヘッダーを使用して、物理レコードと論理レコードの関係を判断する必要があります。

ADACMP ヘッダーとレコードのセグメント化の詳細については、「セグメント化されたレコードに関する考慮事項」を参照してください。

入力パラメータ

パラメータ	使用方法
0 (R1)	入力レコードのアドレス。可変長レコードの前にある長さフィールドは無視されます。このアドレスはフルワードで、ADACMPがDD/EBANDでエンドオブファイルを検出した後にユーザー出口がコールされる場合には-1 (X'FFFF FFFF')が格納されます。
4 (R1)	入力レコード長を格納するフィールドのアドレス。固定長レコードの場合、この値は論理レコード長になります。可変長レコードの場合、この値は実際のデータのみ長さ（長さフィールド自体を除いた長さ）になります。このアドレスのポイント先はフルワードで、ADACMPがDD/EBANDでエンドオブファイルを検出した後にユーザー出口がコールされる場合には-1 (X'FFFFFFFF')が格納されます。
8 (R1)	ユーザー出口へのエントリにバイナリの0が含まれます（「出力パラメータ」を参照）。
12 (R1)	ユーザー出口へのエントリにバイナリの0が含まれます（「出力パラメータ」を参照）。
16 (R1)	ADACMP COMPRESS ユーティリティジョブから渡される FILE パラメータ値のアドレス。このアドレスは、右端（下位）2バイトに格納されます。このアドレスはフルワードで、その場所と内容がユーザー出口の処理中に変わらないようにする必要があります。ADACMP COMPRESSから FILE パラメータ値が渡されない場合、このフルワードはX'00000000'です。

出力パラメータ

パラメータ	使用方法
8 (R1)	ユーザー出口の出力レコードのアドレス。このレコードは、ADACMP圧縮アルゴリズムの入力として使用されます。このレコードのアドレスは、ユーザー出口がコールされたときに、常に8 (R1)に格納する必要があります。リターン時に、このフィールドにバイナリのゼロが格納されると、ADACMPは、入力レコードを無視して処理を続行します。
12 (R1)	返すレコードの長さを格納するための4バイトフィールドのアドレス。このフィールドのアドレスは、出口がコールされたときに、常に12 (R1)に格納する必要があります。リターン時に、このフィールドにバイナリのゼロが格納されると、ADACMPは、レコードを無視して処理を継続します。12 (R1)がポイントする長さフィールド自体の長さは4バイトですが、実際に使用される部分は、下位／右端（3～4バイト目）のハーフワードです。リターン時にバイト2にX'01'が格納されると、次のレコードがDDEBANDから読み込まれる前に出口が再びコールされます。したがって、DD/EBANDからレコードが読み込まれるごとに、レコードを複数回にわたってADACMPに返すことができます。

8 ユーザー出口8 (オペレーターインターフェイス)

- 入力パラメータ 29

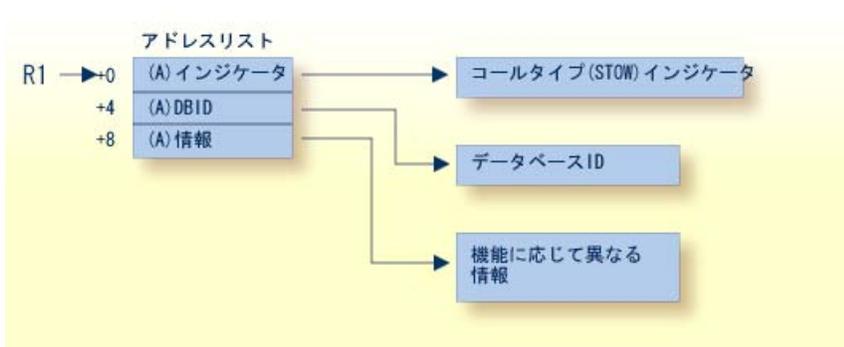
ユーザー出口 8 (オペレータインターフェイス)

このユーザー出口は Adabas ニュークリアスから制御を受け取ります。受け取るタイミングは、ニュークリアスの起動時または終了時、ニュークリアスまたは Adabas ユーティリティがオペレータからメッセージを受け取ったとき、またはメッセージをオペレータに送信したときです。次の場合に、ユーザー出口 8 を使用して、オペレータに特定の指示を送ることができます。

- ニュークリアスが処理を開始したとき、正常に処理を終了したとき
- Adabas がコンソールメッセージをオペレータに送信するときに、補足情報を付加する場合
- オペレータが入力したコマンドを確認する場合

ユーザー出口 8 は、次の場合に起動されます。

- (MODE=MULTI の場合のみ) Adabas が起動した後。ニュークリアスがユーザープログラムからのコールに反応できる状態になった直後。この時点では、ニュークリアスはすでにアクティブになっています。
- Adabas ニュークリアスまたはユーティリティがコンソールオペレータメッセージを発行した直後。ユーザー出口コールは標準のメッセージ処理に対する追加として機能します。メッセージ自体をユーザー出口の実行中に変更することはできません。
- Adabas ニュークリアスまたはユーティリティがオペレータコマンドを受信した後。出口は、コマンドが実際に処理される前にコールされます。コール時には、コマンドの実行を拒否したり、別のコマンドに置き換えたりすることもできます。ただし、コール元のエリアで、コマンドを変更することはできません。
- Adabas ニュークリアスが正常に終了する前。この時点では、ニュークリアスはすでにアクティブではないため、ニュークリアスをコールすると、レスポンスコード 148 が返されます。また、ニュークリアスが異常終了した場合には、この出口はコールされません。



オペレータ インターフェイス ユーザー出口 (8) パラメータ

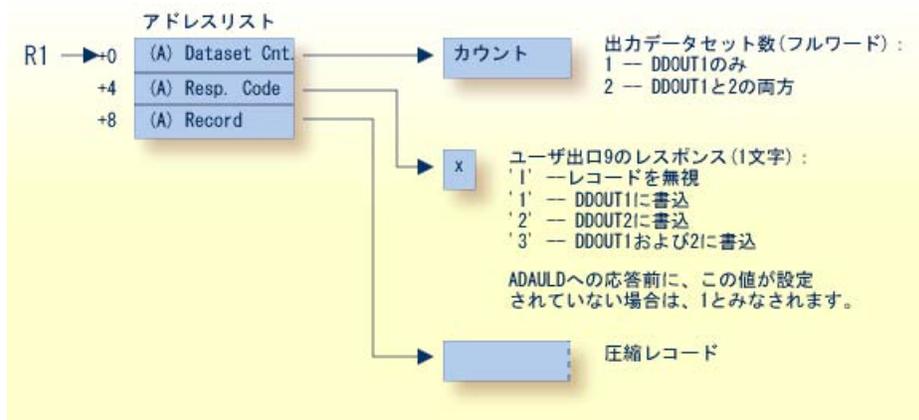
このchapterでは、次のトピックについて説明します。

入力パラメータ

パラメータ	使用方法	
0 (R1)	コールタイプを格納するバイトのアドレス。コールタイプは、次に示すように S、T、O、W の値です。	
	S	ニュークリアス起動時にコールする
	T	ニュークリアス終了時にコールする
	O	ニュークリアス/ユーティリティに対するオペレータメッセージ付きでコールする
	W	オペレータに対するニュークリアス/ユーティリティのメッセージ付きでコールする
4 (R1)	データベース ID を格納するフルワードのアドレス。	
8 (R1)	コールタイプが O、W のときの可変長メッセージ関連情報のアドレス。このアドレスの情報は、コールタイプに応じて次の形式になります。	
	O	1 バイトのメッセージ長の後に、英数字のメッセージが続きます。この長さには、長さバイト自体の長さは含まれません。メッセージを変更する場合は、8 (R1) のポイント先が、リターン時の新しいメッセージを示している必要があります。このメッセージの構成は、前述のとおりです。メッセージを出力しない場合は、8 (R1) のポイント先が、X'0140' という値の 2 バイトのフィールドになっている必要があります。
	W	8(R1) points to the message, which has the following structure: DC H'message-length+4' DC H'0' DC C'message-text'

9 ユーザー出口9 (ADAULD)

ユーザー出口9は、圧縮レコードが書き込み可能な状態になったときに、ADAULD からコールされます。ユーザー出口では、レコードを DD/OUT1 と DD/OUT2 のいずれか、または両方に書き込むか、あるいはどちらにも書き込まないかを判定します。



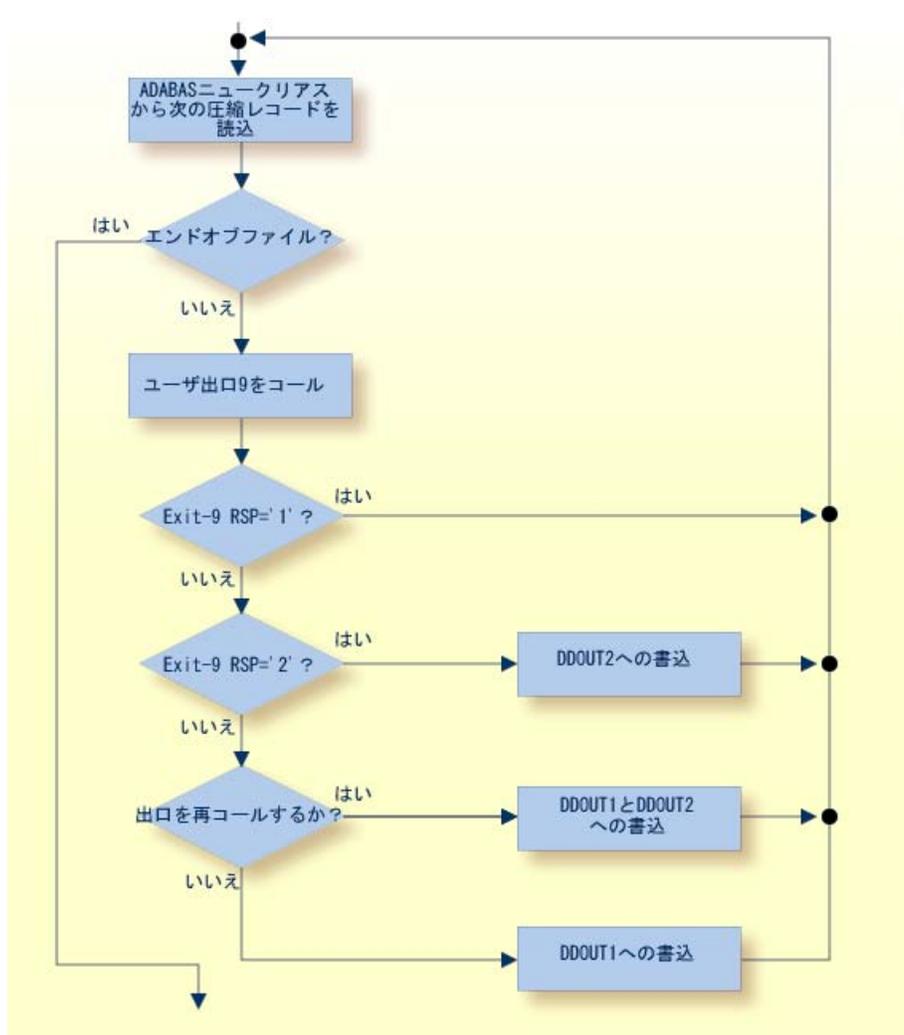
ADAULD ユーザー出口 (9) パラメータ

Notes:

1. DD/OUT1 および DD/OUT2 のブロックサイズが同じでない場合は、ADAULD でエラーが発生します。

3 番目のアドレスがポイントする圧縮レコードの構造は、次のとおりです。

```
AL2 (L1) total length (inclusive)
AL2 (L2) record length (inclusive)
AL4 (ISN)
XL (L2 - 8) '...compressed fields...'
XL (L1 - L2 - 2) '...DVT entries...'
```



ユーザー出口 9 出力制御フロー

ユーザー出口が、標準の BASR 14,15 アセンブラ命令を使用してコールされます。すべてのレジスタは、制御を受け取るときに保存され、ADAULD に制御を返す直前にリストアされる必要があります。R15 の内容は無視されます。

10 ユーザー出口 11 (一般処理)

- 入出力パラメータ 35

このユーザー出口は、Adabas ニュークリアスがコマンドを受け取った直後に Adabas から制御を受け取ります。コマンドのタイプ (シンプルアクセス、コンプレックスアクセス、更新) の判別には使用されますが、コマンド自体は処理されません。

このユーザー出口の最も一般的な使用法としては、ACBX へのセキュリティパスワードやサイファコードの挿入があります。

このユーザー出口の機能は、従来のユーザー出口 1 とほぼ同じですが、CQX 構造と ACBX データ構造のコピーがユーザー出口 11 の処理中に使用される点が異なります。ユーザー出口 1 で使用される実際の構造は、ユーザー出口 11 では使用されません。また、Adabas 8 以降ではユーザー出口 1 はサポートされません。

ACBX の特定フィールドのみが出口により変更される場合があります。この特定フィールドとしては、ACBXFNR (ファイル番号)、ACBXADD2 (アディクション 2)、ACBXADD3 (アディクション 3) および ACBXUSER (ユーザーエリア) が該当します。他の ACBX フィールドを変更しても、CQX を変更しても、ニュークリアスからは無視されます。DSECT EX11PARM は、ユーザー出口 11 のパラメータリストをマッピングします。

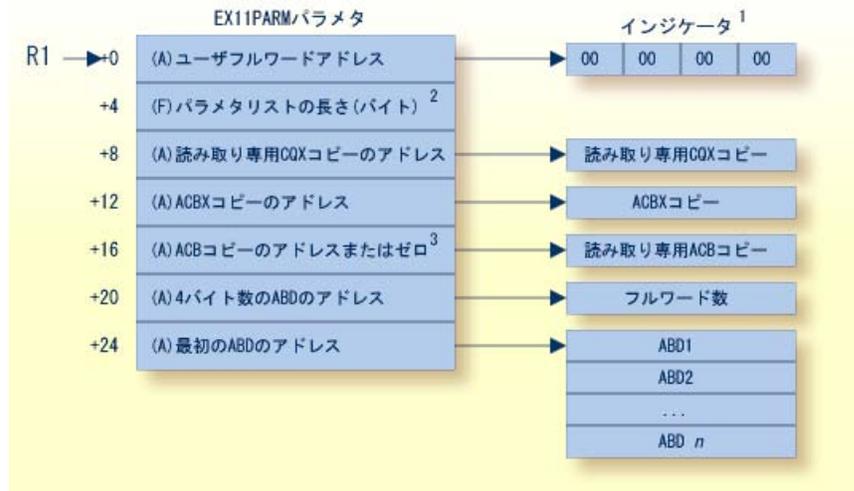
ユーザー出口が、標準の BASR 14,15 アセンブラ命令を使用してコールされます。レジスタ 1 には、パラメータリストのアドレスが格納されます。すべてのレジスタは、制御を受け取られときに保存され、Adabas に制御を返す直前にリストアされる必要があります。



Notes:

1. すべてのユーザー出口は、ユーザー出口のコール時にアクティブだったものと同じプログラム状況ワード (PSW) フィールドをコール元プログラムに返す必要があります。この要件は、アドレッシングモード (AMODE)、プログラムマスク、問題状況のフラグ、PSW キー、およびアドレススペース制御設定において、特に重要になります。コンディションコードを保持する必要はありません。これらの PSW フィールドのいずれかをユーザー出口で変更する場合に、変更前の値を確実に返す方法としては、コードのうち変更部分を BAKR 命令と PR 命令で挟み込む方法があります。
2. コール時に元の ACBX から指定されたファイル番号を変更することはできません。ユーザー出口でファイル番号を変更する必要がある場合は、その出口に対して作成された ACBX のコピー内の ACBXFNR フィールドを変更してください。
3. Adabas コントロールブロックのコマンドコードフィールドを変更することはできません。変更しようとする、レスポンスコードが返されます。
4. Adabas バッファ記述 (ABD) の Adabas バッファの長さは、コールで使用されるため、変更することはできません。

入出力パラメータ



汎用処理 ユーザー出口 (1) パラメータ

¹インジケータ：ユーザー出口11のコール前に、フルワードのインジケータエリアがゼロに設定されます。ユーザー出口11コールの出力結果として、最初のインジケータバイトがゼロになった場合、コマンドが実行可能であることを表します。また、ゼロ以外になった場合は、コマンドが実行されないことを示し、レスポンスコード 22 が返されます。

²パラメータ長：EX11PARM パラメタリストの長さは 28 バイト以上です。

³ACB コピーのアドレス：コマンドが ACBX インターフェイスダイレクトコールを使用する場合は、このアドレスをゼロに設定する必要があります。

11 ユーザー出口 12 (マルチデータセットログの処理)

■ ユーザー出口 12 コールの処理フロー	39
■ ユーザー出口インターフェイス	40
■ 出力パラメータ	43
■ サンプルのユーザー出口のアクティブ化	43

 **Note:** UEX2 および UEX12 は、Adabas ニュークリアスセッションでは相互に排他的であり、指定できるのはどちらか一方だけです。

このユーザー出口は、マルチログデータセットの切り替え時に、Adabas ニュークリアスから制御を受け取ります。これは、切り替える前のログデータセットを Adabas が再び使用する前に、そのログデータセットをコピーすることが目的です。この切り替えは、マルチデータセットのデータプロテクションログと、マルチデータセットのコマンドログのいずれか、または両方がセッションで有効な場合にのみ実行されます。

ユーザー出口ルーチンの役割は、ADARES ユーティリティの該当機能 (CLCOPY または PLCOPY) を実行するためのプロシージャを起動することです。

ユーザー出口 12 は、次の場合に起動されます。

- マルチ PLOG/CLOG データセットをコピーする必要がある場合に、Adabas ニュークリアスが起動されたとき
- ログデータセットが切り替えられたとき
- Adabas ニュークリアスのシャットダウン中

ユーザー出口には、ログのタイプ (PLOG または CLOG) と、マルチログデータセットのステータスに関する情報が渡されます。

ユーザー出口は、次のうちいずれかの処理を実行できます。

- コールを無視し、Adabas が処理を継続できるように制御を返す
- 格納スペースがなくなったログデータセットをコピーするジョブをサブミットし、コピー元のログデータセットを空のデータセットとして扱う (ADARES ユーティリティ)
- 指定時間だけ待機し、PLOG/CLOG データセットのステータス情報を更新して、ユーザー出口を再度コールするように Adabas に指示する 待機状態の間、処理対象のログタイプに応じたログレコードを生成する可能性があるコマンドは、処理できません。

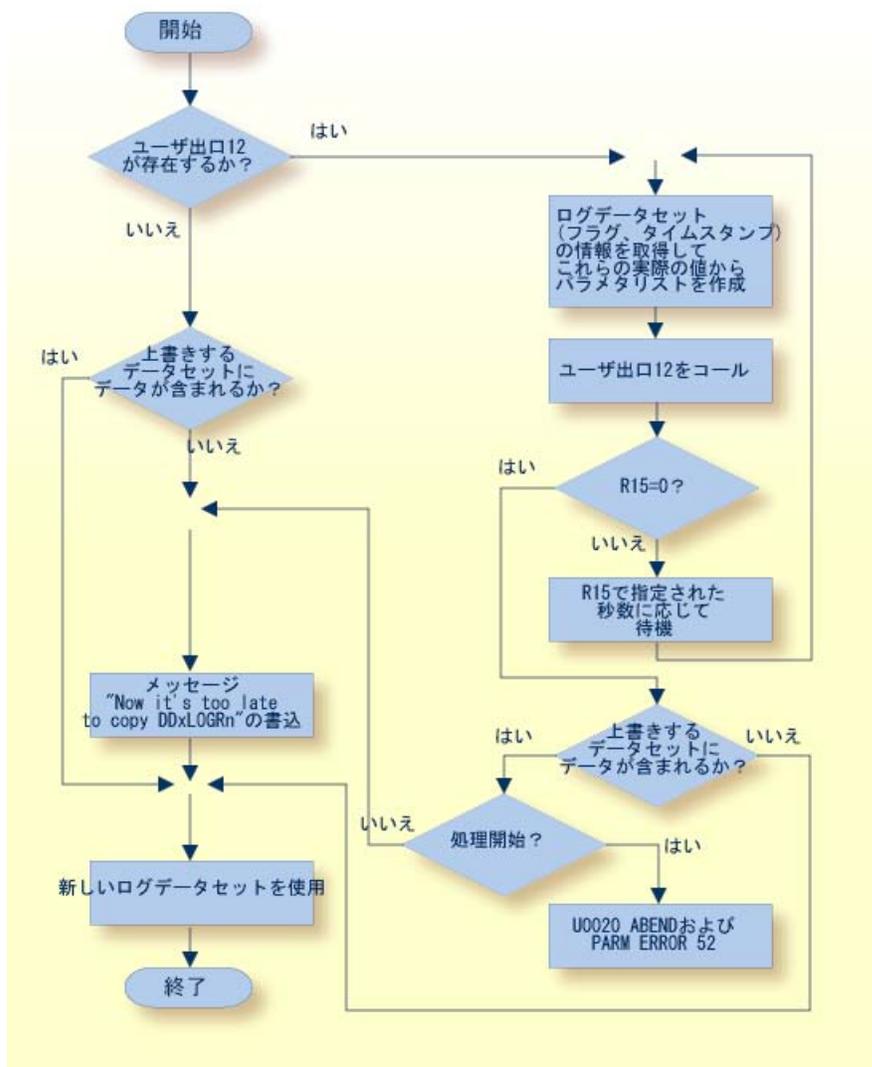
ユーザー出口 12 のサンプルは、Adabas のインストール手順を実行すると作成されます。詳細は、『*Adabas* インストールマニュアル』を参照してください。

ユーザー出口が、標準の BASSMR14,R15 アセンブラ命令を使用してコールされます。すべてのレジスタは、制御が受け取られときに保存され、Adabas に制御を返す直前にリストアされる必要があります。他のセクションで説明しているように、レジスタ 15 にはアクションコードが格納されます。

 **Note:** ユーザー出口 12 は、ユーザー出口のコール時にアクティブだったものと同じ AMODE 値をコール元プログラムに返す必要があります。値を返すアセンブラ命令としては、BSM 0,R14 が適しています。

このchapterでは、次のトピックについて説明します。

ユーザー出口 12 コールの処理フロー



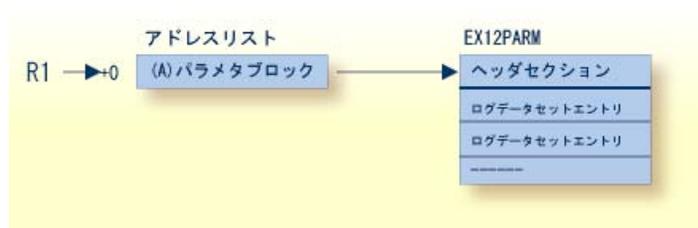
マルチ ログ処理フロー

ユーザー出口インターフェイス

必要に応じて、ユーザー出口は、それ自体の処理を初期化することができます。また、ユーザー出口は、そのリソースの状態を記録するために、EX12PARM パラメータブロックのフィールド EX12USER に、任意の値を格納することもできます。このフィールドは、ユーザー出口が専用で使用するフィールドとして扱われますが、その出口の後続の処理でも使用することができます。出口が初めてコールされたときに、このフィールドはゼロに設定されますが、それ以降は変更されることはありません。

終了時は、必要な後処理を行うために、ユーザー出口が再度コールされます。

パラメータ



ユーザー出口 12 パラメータ

EX12PARM パラメータブロックの DSECT

```

MACRO
EX12PARM
*****
.*
.* Name          Ex12Parm                               *
.*
.* Component     ADABAS User Exit                       *
.*
.* Function      Parameter list for User Exit 12        *
.*               (replacement for User Exit 2 for use when there are *
.*               more than 2 PLOGs or CLOGs)             *
.*
.* Parameters    None                                    *
.*
.* Restrictions  None                                    *
.*
.* Notes        None                                    *
.*
*****
.*
EX12PARM DSECT ,          User Exit 12 Parameter List
*
EX12HDR  DS      0F          Common header section
*
EX12USER DS      F          Reserved for the user. This field +
                             is initialized to zero before the +
                             exit is called the first time, and +
                             will not be altered by ADABAS after +
                             that. It can be used to maintain +
                             information across invocations.
*
EX12LOGT DS      X          Log type
EX12PLOG EQU     C'P'       PLOG
EX12CLOG EQU     C'C'       CLOG
*
EX12TYPE DS      X          Call type
  
```

ユーザー出口 12 (マルチデータセットログの処理)

```

EX12TBEG EQU      C'S'          Nucleus start
EX12TSW  EQU      C'W'          Log switch
EX12TEND EQU      C'T'          Nucleus termination
          DS      XL2           Reserved
*
EX12NLOG DS        F            Number of logs
EX12DBID DS        F            Database ID
EX12NUCI DS        F            Nucleus ID
*
EX12PLGN DS        F            Current session PLOG number          +
                                (zero for CLOGs)
EX12NCMP DS        F            Log just completed
EX12STAT DS        X            Flags of next log in sequence
EX12WNUC EQU      X'80'        Being written by the nucleus
EX12FULL EQU      X'40'        Completed by the nucleus
EX12RES  EQU      X'20'        Being copied by ADARES
EX12CL5  EQU      X'08'        CLOGLAYOUT=5
EX12UNUS EQU      X'00'        Unused and/or copied
          DS      XL3           Reserved
          DS      4F           Reserved
*
EX12HDRL EQU      *-EX12HDR     Length of header section
*
EX12ENT  DSECT    ,
EX12LOG  DS        0F           Start of individual log data set          +
                                entries. This section is repeated          +
                                for the number of logs specified in          +
                                field EX12NLOG
*
EX12LTIM DS        XL8         Time stamp of write to log data set
EX12LNUM DS        F            Number of log data set
EX12LFLG DS        X            Flags (mapped as in EX12STAT)
          DS      XL3           Reserved
          DS      4F           Reserved
*
EX12LOGL EQU      *-EX12LOG     Length of a log data set entry
*
*
          MEND

```

出力パラメータ

パラメータ	使用方法
R15 = 0	ニュークリアスは処理を続行します。
R15 > 0	R15 の値は、ユーザー出口 12 を再度コールするまでの待機時間の秒数として扱われ、その間、すべてのログデータセットのステータスが最新のまま維持されます。待機状態の間は、ログエントリを作成する可能性があるコマンドは、処理できません。

サンプルのユーザー出口のアクティブ化

サンプルのユーザー出口は、アセンブラ言語で記述されています。このサンプルの機能は次のとおりです。

- コールされたログの用途とタイプを示すメッセージを発行します。
- 空ではないすべてのログデータセットのステータスとタイムスタンプを示すメッセージを発行します。
- 格納スペースのないログデータセットが存在し、出口が最後にコールされたときとステータスが異なるログデータセットが1つでもある場合は、出口は 80 バイトのレコードを入力ファイルから読み込み、出力ファイルに書き込みます。? という文字のオカレンスはすべて、PLOG イベントと CLOG イベントのどちらを処理するために出口が起動されたのかによって、P または C に置き換えられます。したがって、ログタイプに応じたイベントに基づいて、入力ファイルは正しく処理されます。通常、入力ファイルにはジョブ制御ステートメントが存在し、出力ファイルの出力先はジョブ実行キューになります。
- 格納スペースが残っているログデータセットが1つでもあれば、出口はコール元に戻り、R15 をゼロに設定するので、Adabas は処理を継続します。
- どのログデータセットにも格納スペースがない場合、出口はコール元に戻りますが、R15 をゼロ以外に設定するので、Adabas は R15 の秒数だけ待機し、待機時間が終わると、ログデータセットのステータスを最新のまま維持した状態で出口を再びコールします。デフォルトの待機時間は 30 秒です。

z/OS

z/OS では、付属のサンプルユーザー出口 UX12SAMP の形式は、ソースとロードモジュールです。どちらも変更や再アSEMBルせずに、そのまま使用することができます。

ユーザー出口のソースとロードモジュールは、それぞれ Adabas ソースとロードライブラリに収録されています。ユーザー出口 UX12ASML をアSEMBルするジョブは、Adabas ジョブライブラリに収録されています。また、ジョブライブラリには、サンプルジョブ UX12CJOB も収録されており、カスタマイズしたり、ADARES ユーティリティの PLCOPY または CLCOPY 機能を起動するユーザー出口からサブミットしたりできます。

サンプルユーザー出口を次のようにしてアクティブにします。

1. ADARUN NxLOG={2-8} の他に、Adabas ニュークリアスに ADARUN UEX12=UX12SAMP を指定します。
2. ユーザー出口から DDNAME COPYJOB 配下のジョブをサブミットできるように、ジョブ制御モデルを設定してください。

DD ステートメントは次のように指定します。

```
//INTRDR2 DD SYSOUT=(*,INTRDR)
```



Note: PLCOPY または CLCOPY ジョブをサブミットするときに使用する DD ステートメントは、サンプルのユーザー出口 2 でもユーザー出口 12 でも同じです。

VSE

VSE では、付属のサンプルユーザー出口 UX12SAMP の形式はソースとフェーズです。どちらも変更や再アSEMBルせずに、そのまま使用することができます。

ソース A.book とフェーズは、Adabas VSE サブライブラリに収録されています。ユーザー出口 UX12ASML.X をアSEMBルするジョブは、Adabas サブライブラリにも収録されています。また、サンプルジョブ UX12CJOB.X も収録されており、カスタマイズしたり、ADARES ユーティリティの PLCOPY または CLCOPY 機能を起動するユーザー出口からサブミットしたりできます。

サンプルユーザー出口を次のようにしてアクティブにします。

1. ADARUN NxLOG={2-8} の他に、Adabas ニュークリアスに ADARUN UEX12=UX12SAMP を指定します。
2. ユーザー出口から DLBL COPYJOB 配下のジョブをサブミットできるように、ジョブ制御モデルを設定してください。

BS2000

BS2000 では、付属のサンプルユーザー出口 UX12SAMP の形式は、ソースとオブジェクトモジュールです。どちらも変更や再アセンブルせずに、そのまま使用することができます。

ユーザー出口のソースは、Adabas ソースライブラリに収録されています。ユーザー出口 UX12ASML をアセンブルするジョブは、Adabas ジョブライブラリに収録されています。また、ジョブライブラリには、サンプルジョブ UX12CJOB も収録されており、ADARES ユーティリティの PLCOPY または CLCOPY 機能を起動する UX12SAMP からサブミットすることができます。

サンプルユーザー出口を次のようにしてアクティブにします。

1. ADARUN NxLOG={2-8} の他に、Adabas ニュークリアスに ADARUN UEX12=UX12SAMP を指定します。
2. マシンがジョブ変数を処理できる場合は、Adabas ニュークリアスジョブに次のステートメントを入力してください。

```
/SET-JV-LINK LINK-NAME=*DDJBDSF, -
/          JV-NAME=#JV.DSF
/MODIFY-JV  JV-ID=JV-NAME(JV-NAME=#JV.DSF), -
/          VALUE='ENTER-JOB FROM-FILE=*LIB-ELEM(, -
/LIB=library, ELEM=dsfjob), -
/JOB-CLASS=JCDSF'
```

ここでは次の内容を表しています。

library ライブラリ

dsfjob 手順2 でカスタマイズしたジョブが存在するメンバ

マシンがジョブ変数を処理できない場合は、次のジョブがサブミットされます。

```
(ADABAS.SRC,UX12CJBP,J)
```

PLCOPY ジョブの場合は上記のとおりです。

```
(ADABAS.SRC,UX12CJBC,J)
```

CLCOPY ジョブの場合は上記のとおりです。

データベース ID をパラメータとして指定するサンプルジョブは、Adabas ソースライブラリ ADAvrs.SRC に収録されています。

このジョブでは目的を果たせない場合は、付属のソース UX12SAMP を修正してください。

BS2000 システムのユーザー出口 12 を使用している場合には、従来のリリースと異なり、ADARES CLCOPY または PLCOPY ジョブの名前はユーザー出口にハードコードされていません。したがって、次に説明するように、/ENTER ジョブを指定する方法が 2 つあります。

方法 1

完全な /ENTER ジョブコマンドを含むジョブ変数を作成します。例としては、次のようなものがあります。

```
/DCLJV ENTER.ADARES.PLCOPY, LINK=*DDJBPLC  
/SETJV *DDJBx12,C'ENTER-PROC(srclib,ux12cjob),(DB=dbid),J-N=X12COPY'
```

ここでは次の内容を表しています。

srclib UX12CJBx(J) が収録されているライブラリです。x は、PLCOPY ジョブの場合には P、CLCOPY ジョブの場合には C にそれぞれ置き換わります。

ux12cjob それぞれのプロシージャジョブの J 要素名です。

dbid このニュークリアスのデータベース ID です。

/ENTER コマンドのすべてのオペランドを使用できます。/DCLJV ステートメントを指定すると、他の指定よりも優先されます。必要な場合は、ニュークリアスセッション中にジョブ変数の内容を変更できます。

リンク名 *DDJBPLC を使用して ADARES PLCOPY ジョブを割り当て、*DDJBCLC を使用して ADARES CLCOPY ジョブを割り当てます。

方法 2

JV 指定を省略します。ユーザー出口 12 は、次のコマンドを発行するようになります。

```
'ENTER-PROC RES.EPdbid.xLCO,(DB=dbid),J-N=xL12<dbid>'
```

データベースを起動する前に、データベースのジョブデッキからジョブ要素を抽出する場合には、次のコードを使用します。

```
/DELETE-FILE RES.EPdbid.CLCO
/SET-JOB-STEP
/DELETE-FILE RES.EPdbid.PLCO
/SET-JOB-STEP
/ASS-SYSDTA *SYSCMD
/STA-LMS
//O srclib
//EXT-E (,ux12cjbc,J),RES.EPdbid.CLCO
//EXT-E (,ux12cjbpc,J),RES.EPdbid.PLCO
//END
/ASS-SYSDTA *P
```

ここでは次の内容を表しています。

dbid P/CLCOPY データベースのデータベース ID です。この ID は、プロシージャでは &DBID. になります。

srclib テンプレートのコピープロシージャが収録されているソースライブラリです。

ux12cjbc 出口 12 の ADARES CLCOPY ジョブを実行するプロシージャです。

ux12cjbpc 出口 12 の ADARES PLCOPY ジョブを実行するプロシージャです。

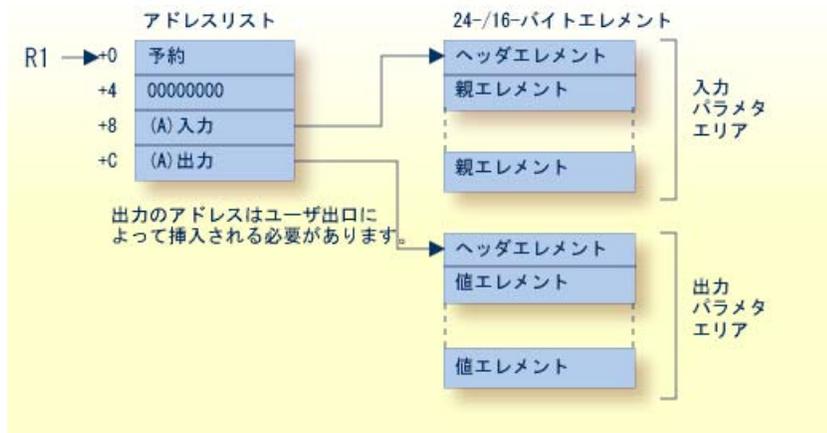


Note: BS2000 システムの場合、ユーザー出口 12 の供給形態は、ソース要素およびオブジェクト要素の両方です。

12 ハイパーディスクリプタ出口 01 - 31

■ メインパラメータエリア	51
■ 入力パラメータエリア (3 番目のパラメータアドレスのポイント先)	52
■ 出力パラメータエリア	54
■ 空値オプション	55
■ ハイパー出口初期化コール	56
■ ハイパー出口スタブ	56

ハイパーディスクリプタ出口（ハイパー出口）1～31（HEX01...HEX31）には、ユーザーが指定するディスクリプタ値のアルゴリズムを定義する必要があります（『Adabas ユーティリティマニュアル』を参照）。ハイパー出口は、ハイパーディスクリプタ値が生成されるたびに、ADACMP ユーティリティまたは Adabas ニュークリアスによりコールされます。ADACMP は常に、それ自体の ADARUN ステートメントで指定されたハイパーディスクリプタ出口を使用します。ADAINV ユーティリティによりハイパーディスクリプタ出口が指定されると、Adabas ニュークリアスの ADARUN ステートメントで指定された出口が使用されます。



ハイパーディスクリプタ出口パラメータ

 **Notes:**

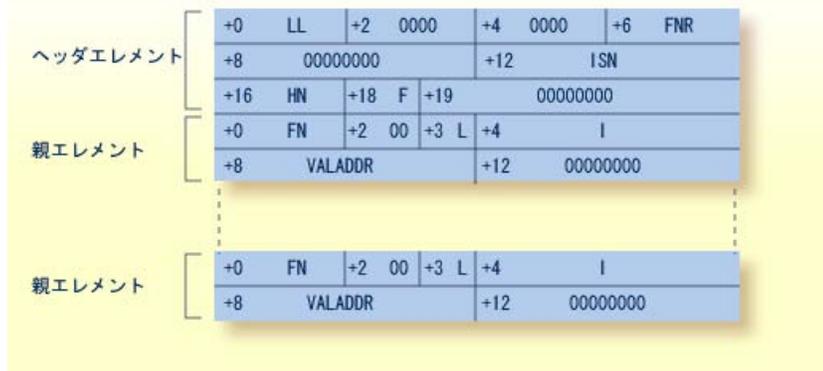
1. ハイパー出口は、ハイパー出口のコール時にアクティブだったものと同じ AMODE 値をコール元プログラムに返す必要があります。
2. Adabas 8 以降がインストールされていて、ハイパー出口が新しいパラメタリストを使用できるように更新されていない場合、必要なパラメタリストの変更と入力パラメタの変換を行うハイパー出口に、ハイパー出口スタブをリンクすることができます。ハイパー出口スタブでは、MU または PE はサポートされていません。詳細は、「[ハイパー出口スタブ](#)」を参照してください。
3. Adabas ニュークリアスまたは ADACMP の起動時に、ハイパー出口がロードされるごとに初期化がコールされます。詳細は、「[ハイパー出口初期化コール](#)」を参照してください。

このchapterでは、次のトピックについて説明します。

メインパラメータエリア

パラメータ	内容
0 (R1)	予約済み (変更不可)。
4 (R1)	フルワードのゼロ (変更不可)。
8 (R1)	入力パラメータエリアの開始アドレス。
12 (R1)	出力パラメータエリアの開始アドレス。このアドレスは、ユーザー作成のプログラムにより挿入される必要があります。出力パラメータエリアは、常にユーザーハイパー出口から返す必要があります。値を返さない場合、アドレスのポイント先は、値エレメントが存在しないことを示すように全体の長さが設定されたヘッダーエレメントになります。

入力パラメータエリア（3番目のパラメータアドレスのポイント先）



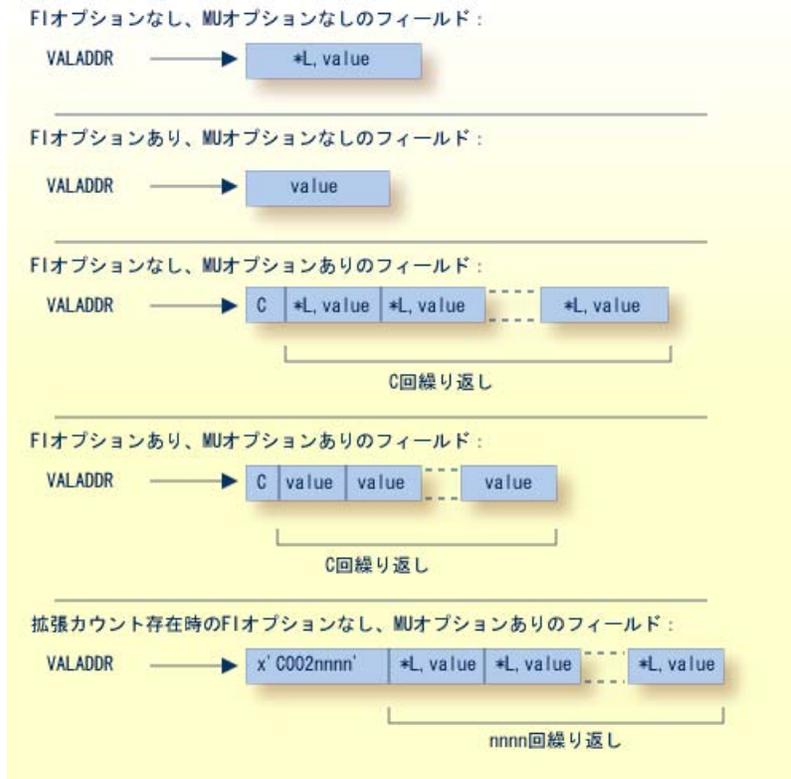
ヘッダーエレメントフィールド

LL	入力パラメータエリア全体の長さ（この長さフィールドも含む）
FNR	ファイル番号
ISN	レコードに割り当てられた ISN
HN	ハイパーディスクリプタの名前
F	フラグバイト： <ul style="list-style-type: none"> ■ X'02' の場合は、ファイルに拡張 MU または PE フィールドが存在することを表します。 ■ X'80' の場合は、初期化コールを表します。

親エレメントフィールド

FN	親フィールドの名前。
L	親フィールドが FI オプション付きで定義されている場合、VALADDR がポイントする値の長さになります。
I	親フィールドのピリオディックグループのインデックス（4バイト長）。親フィールドが PE グループに含まれていない場合、このバイトの値はゼロです。
VALADDR	親フィールドの値のアドレス。値のフォーマットは、フィールドのオプションにより決定されます。親フィールドが NU（空値省略）オプションで定義され、このフィールドの値が省略された場合は、入力パラメータエレメントは作成されません。

次の例は、VALADDR により、FI（固定ストレージ）および MU（マルチプルバリュー）オプションの組み合わせで親フィールドにポイントされた値のフォーマットを示します。



上記の意味は次に示すとおりです。

C MU カウントを表す 1 バイト値。MY 値の対象が、拡張 MU または PE フィールド付きで定義されたファイルである場合は、拡張カウントが存在する可能性があります。拡張カウントの詳細については、ADACMP ドキュメントの「圧縮レコード内の 191 を超える MU および PE オカレンスの特定」を参照してください。

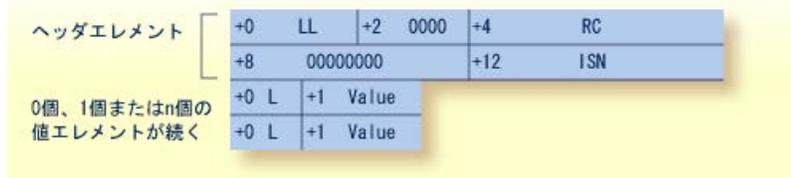
*L 16 進値の長さです。この長さには、このフィールド自体の長さ値（1 バイトまたは 2 バイト）も含まれます。長さが 1~127 の場合に必要なバイト数は 1 です。長さが 128~255 の場合に必要なバイト数は 2 です。最初のバイトは X'80' に、2 番目のバイトは実際の長さの値に設定されます（次の例の表を参照）。

長さ	バイト 1	バイト 2
L=127:	x'7F'	(x'80')
L=128:	(x'80')	(x'80')
L=255:	(x'80')	x'FF'

出力パラメータエリア

このエリアを割り当てて、ハイパーディスクリプタユーザー出口の領域全体に適用する必要があります。このエリアのアドレスは、メインパラメータエリアの2番目の位置に配置する必要があります。

このエリアは、8バイトヘッダーおよび、その後続く、生成された圧縮形式のハイパーディスクリプタ値で構成されます。



ヘッダー要素

LL	出力パラメータエリア全体の長さ（この長さフィールドも含む）。値を返さない場合、全体の長さは、ヘッダー要素の長さに設定されます。
00	予約済みのスペース。これは、0に設定する必要があります。
RC	リターンコード。ハイパー出口でゼロ以外の値を設定すると、コールを拒否することを示します。この場合に設定する値としては、“16”が推奨です。このフィールドがゼロ以外の場合、コールはエラーになり、レスポンスコード 79 が返されます。
ISN	ディスクリプタ値に割り当てられた ISN。元の ISN を変更する場合は、新しい ISN をここに挿入する必要があります。Adabas ニュークリアスへのリターン時にこれらの4バイトに0が含まれる場合は、元の ISN が使用されます。これは4バイトのバイナリ値です。

Note: ハイパー出口が、ヘッダー要素の ISN フィールドで ISN を返す場合は、ファイルが USERISN=YES で定義され、ファイルが後でリロードされるときに ISN が割り当て直されるのを防ぎます。

値要素

L	この次の値の長さ（この長さバイトも含む）。最大長は、ハイパーディスクリプタに使用されているフォーマットにより決定されます。
値	インデックスに挿入されるディスクリプタ値。値は、このハイパーディスクリプタに割り当てられるフォーマットに応じた規則に従っている必要があります。ハイパーディスクリプタが PE オプションで定義されている場合は、1バイト PE インデックスを含む1つのバイトが値の直後にあり、長さLに含まれている必要があります。PEオプションで定義されているハイパーディスクリプタの対象が、拡張 MU または PE フィールド付きで定義されたファイルである場合は、2バイトの PE インデックス

	を含む2つのバイトが値の直後に続き、長さLに含まれている必要があります。ニュークリアスは、パック形式または数値形式の値が正しいかどうかをチェックします。パックフィールドの正しい符号は、A、C、E、F（正）およびB、D（負）です。ニュークリアスはすべての符合をFまたはDに変更します。
--	---

例

L	値	注
04	R E D	
06	B L U E02	X'02' は PE インデックス
03	123F	パック 10 進数の 123
04	123F01	インデックス 1 を持つ PE グループのパック 10 進数の 123
07	B L U E0002	X'0002' は、拡張 MU または PE フィールド付きで定義されたファイルの PE インデックス
05	123F010A	インデックス 266 を持つ拡張 PE グループのパック 10 進数の 123

空値オプション

NU（空値）オプションは、ハイパーディスクリプタまたは親フィールドに対して指定できます。起こりうる組み合わせは次のとおりです。

- ハイパーディスクリプタが NU ではない場合
 - 親フィールドは NU ではなく、値は空値です。ハイパー出口がコールされ、空値が渡されます。
 - 親フィールドは NU で、値は空値です。ハイパー出口がコールされ、この親フィールドには入力パラメータ要素が作成されません。
 - すべての親フィールドは NU で、すべての値は空値です。ハイパー出口がコールされ、どの親フィールドにも入力パラメータ要素が作成されません。
- ハイパーディスクリプタが NU である場合
 - 親フィールドは NU ではなく、値は空値です。ハイパー出口がコールされ、空値が渡されます。
 - 親フィールドは NU で、その値は空値です。ハイパー出口がコールされ、この親フィールドには入力パラメータ要素が作成されません。
 - すべての親フィールドは NU で、すべての値は空値です。ハイパー出口はコールされません。

ハイパー出口初期化コール

Adabas ニュークリアスまたは ADACMP の起動時に、ハイパー出口がロードされるたびに初期化コールが実行されます。主要なパラメータエリアは、説明に従って使用する必要があります。3番目のパラメータアドレスのポイント先は、ヘッダー長によって後続の値がないことを示す入力パラメータエリアです。フラグバイトは、初期化コールを表す x'80' に設定されます。リターン時には、値が返されていないことを示すヘッダー長とともに、4番目のパラメータアドレスが出力パラメータエリアに設定される必要があります。

ハイパー出口スタブ

ハイパー出口スタブの役割は、以前のハイパー出口を変更しなくても Adabas 8 パラメータリストを使用できるようにすることです。ハイパー出口スタブは、新しいパラメータエリアを使用するために既存のハイパー出口をすぐに更新したくないお客様に向けた一時的な解決方法として用意されています。拡張 MU または PE フィールドが定義されたファイルが対象の場合には、ハイパー出口スタブは機能しません。このようなファイルに対してハイパー出口スタブがコールされた場合、レスポンスコードが返されます。ハイパー出口スタブにリンクしたハイパー出口が、以前のバージョンの Adabas で使用されることはありますが、ハイパー出口スタブ自体が Adabas 8 パラメータを使用するハイパー出口で使用されることはありません。

JOBS データセットのサンプルジョブ LNKHEX8 には、ハイパー出口スタブをハイパー出口にリンクするときの例が収録されています。

z/VM 環境では、次の手順に従って、ハイパー出口スタブにリンクするハイパー出口を作成してください。

1. 付属のテープに収録されているファイル INPUT TEXT を編集します。3 行目 (NAME HEXnnMOD) を、作成するハイパー出口の名前に変更します。ADARUN HEXnn パラメータに同じ名前を設定してください。他の行は変更しないようにしてください。ファイルを保存します。

```
INCLUDE LIBDEF (HEXV87,HEXOLD)
INCLUDE TXTDEF
NAME HEXnnMOD
```

2. 次のコマンドを使用してハイパー出口を生成します。

- FILEDEF LIBDEF DISK ADAV813 TXTLIB x (x は ADAV813 TXTLIB が収録されているファイルモード)
- FILEDEF TXTDEF DISK HEXnn TEXT x (HEXnn は作成するハイパー出口、x はそのハイパー出口が収録されているファイルモード)
- LKED INPUT (LIBE HEXnn) (HEXnn はハイパー出口が収録される新しい LOADLIB の名前)

3. Adabas ニュークリアスを起動する前に、GLOBAL LOADLIB HEXnn を指定してください。

13 照合ディスクリプタ出口 01～08

- 照合ディスクリプタ出口のインターフェイス 60

照合ディスクリプタ出口 1~8 (CDX01~CDX08) は、該当する照合ディスクリプタの値をエンコードしたり、デコードしたりするとき 사용합니다。

照合ディスクリプタは、英数字フィールドまたはワイド文字フィールドに定義することができます。定義した値は、インデックスには保存されますが、レコード自体には保存されません。値の変換に使用する照合ディスクリプタ出口の数と、照合ディスクリプタの数は、対応関係にあります。

サンプルとして、照合ディスクリプタ出口 CDXE2A が Adabas ソースデータセットに収録されています。エンコード機能では EBCDIC から ASCII に変換され、デコード機能では ASCII から EBCDIC に変換されます。

照合出口には3つの機能エントリポイントがあり、これらは次のイベントの発生時にコールされます。

INITIALIZE 機能

- ニュークリアスセッションの開始
- 照合出口が定義されたときのユーティリティの初期化 (ADARUN パラメータ)

ENCODE 機能

- 親の値の更新/挿入/削除 (ニュークリアス)
- 照合ディスクリプタを検索値に指定して検索 (ニュークリアス)
- レコードの圧縮 (ADACMP)

DECODE 機能

- 出口が DECODE 機能をサポートする場合のみ、照合 DE によるインデックス (L9) の読み込み (ニュークリアス)

このchapterでは、次のトピックについて説明します。

照合ディスクリプタ出口のインターフェイス

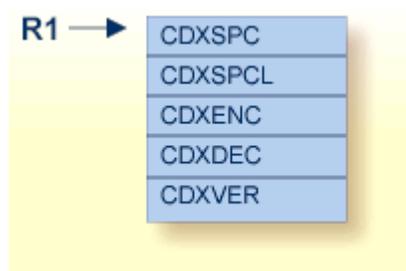
照合ディスクリプタ出口のインターフェイスは、Adabas ソースデータセットの CDXPARM DSECT で定義します。このインターフェイスには、次の3つの機能があります。

- 初期化
- エンコード
- デコード (オプション)

初期化パラメータ

R1のポイント先は、コール元の5つのストレージエリアをポイントするアドレスのリストです。照合ディスクリプタ出口によって、次の5つの設定が行われる必要があります。

CDXSPC	デフォルトのスペース文字（最大4バイト）
CDXSPCL	スペース文字のサイズを格納するフルワード
CDXENC	エンコード機能のアドレス
CDXDEC	デコード機能のアドレス。返されたアドレスがゼロの場合、デコードはサポートされていません。また、L9処理には照合ディスクリプタを使用できません。
CDXVER	ゼロバイトのバージョン区切り文字列のアドレス



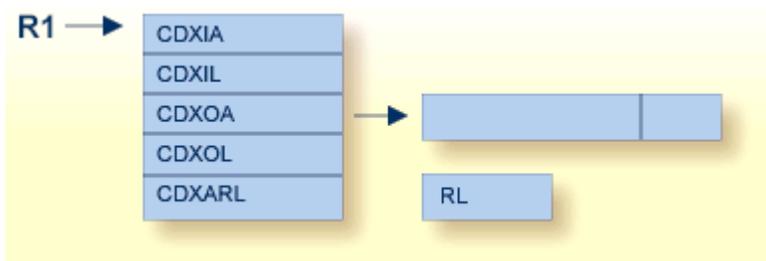
エンコード／デコードのパラメータ

R1は、次に示すように、5つのパラメータ（フルワード）のリストをポイントします。

CDXIA	入力文字列のアドレス
CDXIL	入力文字列の長さ
CDXOA	出力文字列のアドレス
CDXOL	出力エリアのサイズ
CDXARL	返される出力文字列の長さのアドレス

照合ディスクリプタ出口が実行する格納処理は次のとおりです。

- CDXOA が示すアドレスのエリアに出力文字列を格納する
- CDXARL が示すアドレスのフルワードに入力文字列の長さを格納する



索引

A

- Adabas Review
 - ハイイベントハンドラのユーザー出口 5, 19
- ADAULD ユーティリティ
 - ユーザー出口 9 の処理, 31

B

- BS2000
 - ユーザー出口 2 のオプション, 10
 - ユーザー出口 2 の起動オプション, 46

お

- オペレータ
 - 指示を送るプログラム, 28
- オペレータインターフェイスのユーザー出口, 27

こ

- コマンドログ
 - デュアルデータセットユーザー出口, 6
- コマンドログ (CLOG)
 - フォーマット, 16
 - 複数データセットユーザー出口, 37
 - ユーザー出口 4 の処理, 15

さ

- サイファコード
 - ACBX に挿入するプログラム, 34

て

- 出口 1, 3
- 出口 11, 33
- 出口 12, 37
- 出口 2, 5
- 出口 3, 13
- 出口 4, 15
- 出口 5, 19
- 出口 6, 23
- 出口 8, 27
- 出口 9, 31
- デュアルログ処理, 5
- データ圧縮
 - ユーザー出口 6 の前処理, 23

は

- ハイパーディスクリプタ
 - ユーザー出口, 49
 - 説明, 1
- パスワード
 - ACBX に挿入するプログラム, 34

ふ

- フォネティック処理
 - ユーザー出口 3 の処理, 13
- プロテクションログ
 - デュアルデータセットユーザー出口, 6
- プロテクションログ (PLOG)
 - 複数データセットユーザー出口, 37

ゆ

- ユーザー
 - ログデータ処理, 16
- ユーザー出口
 - 2
 - デュアルログ処理、BS2000 オプション, 46
 - BS2000 ユーザー出口 2 のオプション, 10
 - 概要, 1
 - 説明, 1
 - 出口 1, 3
 - 出口 11, 33
 - 出口 12, 37, 40
 - 出口 2, 5
 - 出口 3, 13
 - 出口 4, 15
 - 出口 5, 19
 - 出口 6, 23
 - 出口 8, 27
 - 出口 9, 31
- ユーザー出口を処理するコマンド, 3, 33

ろ

- ログ
 - デュアルデータセットユーザー出口 2 の処理, 5
 - 複数データセットユーザー出口, 37

