

Natural für Windows

Parameter-Referenz

Version 6.3.8 für Windows

Februar 2010

Dieses Dokument gilt für Natural ab Version 6.3.8 für Windows.

Hierin enthaltene Beschreibungen unterliegen Änderungen und Ergänzungen, die in nachfolgenden Release Notes oder Neuausgaben bekanntgegeben werden.

Copyright © 1992-2010 Software AG, Darmstadt, Deutschland und/oder Software AG USA, Inc., Reston, VA, Vereinigte Staaten von Amerika, und/oder ihre Lizenzgeber..

Der Name Software AG, webMethods und alle Software AG Produktnamen sind entweder Warenzeichen oder eingetragene Warenzeichen der Software AG und/oder der Software AG USA, Inc und/oder ihrer Lizenzgeber. Andere hier erwähnte Unternehmens- und Produktnamen können Warenzeichen ihrer jeweiligen Eigentümer sein.

Die Nutzung dieser Software unterliegt den Lizenzbedingungen der Software AG. Diese Bedingungen sind Bestandteil der Produktdokumentation und befinden sich unter <http://documentation.softwareag.com/legal/> und/oder im Wurzelverzeichnis des lizenzierten Produkts.

Diese Software kann Teile von Drittanbieterprodukten enthalten. Die Hinweise zu den Urheberrechten und Lizenzbedingungen der Drittanbieter entnehmen Sie bitte den "License Texts, Copyright Notices and Disclaimers of Third Party Products". Dieses Dokument

ist Bestandteil der Produktdokumentation und befindet sich unter <http://documentation.softwareag.com/legal/> und/oder im Wurzelverzeichnis des lizenzierten Produkts.

Inhaltsverzeichnis

1	Parameter-Referenz	1
2	Profilparameter — Einführung	3
	Benutzung von Profilparametern	4
	Gemeinsame Beschreibungen für Profilparameter und Session-Parameter	4
3	Session-Parameter — Einführung	5
	Benutzung von Session-Parametern	6
	Setzen von Session-Parametern	6
	Verarbeitung von Session-Parametern	8
4	ACIVERS - Define ACI Version for Use with EntireX Broker ACI	11
5	ACTPOLICY - Defines Default Activation Policy for DCOM Classes	13
6	AD - Attribute Definition	15
	AD-Parameter-Syntax	17
	Feldanzeige	17
	Ausrichtung der Feldwerte	18
	Feldeingabe/ausgabe-Charakteristika	18
	Auslegung alphanumerischer Felder	20
	Eingabezwang	20
	Mindestlänge der Eingabewerte	21
	Groß-/Kleinschreibung	21
	Füllzeichen	21
7	AL — Alphanumeric Length for Output	23
8	AUTO - Automatic Logon	25
9	AUTOREGISTER - Automatic Registry Update	27
10	AUTORPC - Automatic Natural RPC Execution	29
11	BATCH - Batch Mode Simulation	31
12	BATCHMODE - Batch Mode	33
13	BMBLANK - Display Trailing Blanks	35
14	BMCONTROL - Display Control Characters	37
15	BMFRAME - Window Frame Characters	39
16	BMSIM - Similar Batch Mode Output	41
17	BMTIME - Display Process Time	43
18	BMTITLE - Display Window Title	45
19	BMVERSION - Display Natural Version	47
20	BPID - Specify Buffer Pool ID	49
21	BPID2 - Specify Secondary Buffer Pool	51
22	BPNAME - Name of Natural Global Buffer Pool	53
23	BPSFI - Object Search First in Buffer Pool	55
24	CC - Error Processing in Batch Mode	57
25	CD - Color Definition	59
26	CDYNAM - Dynamic Loading of Non-Natural Programs	61
27	CF - Character for Terminal Commands	63
28	CLEAR - Processing of CLEAR Key in NEXT Mode	65
29	CM - Command Mode	67

30	CMOBJIN - Batch Input File for Natural INPUT Data	69
31	CMPRINT - Batch Output File	71
32	CMPRTnn - Additional Report	73
33	CMSYNIN - Batch Input File for Natural Commands and INPUT Data	75
34	CMWRKnn - Natural Work Files	77
35	COMPR - Set RPC Buffer Compression	79
36	COMSERVERID - Determine DCOM Server ID	81
37	CP - Default Code Page Name	83
38	CPCVERR - Code Page Conversion Error	85
39	CPOBJIN - Code Page of Batch Input File	87
40	CPPRINT - Code Page of Batch Output File	89
41	CPRPC - Define Code Page Name	91
42	CPSYNIN - Code Page of Batch Input File for Commands	93
43	CV - Attribute Control Variable	95
44	CVMIN - Control Variable Modified at Input	97
45	DBSHORT - Interpretation of Database Field Short Names	99
46	DBUPD - Database Updating	103
47	DC - Character for Decimal Point Notation	105
48	DD - Day Differential	107
49	DF - Date Format	109
50	DFOUT - Date Format for Output	111
51	DFS - Specify RPC Client's Default Server Address	113
52	DFSTACK - Date Format for Stack	115
53	DFTITLE - Output Format of Date in Standard Report Title	117
54	DL - Display Length for Output	119
55	DTFORM - Date Format	121
56	DU - Dump Generation	123
57	DY - Dynamic Attributes	125
	DY-Parametersyntax	126
	Beispiele	128
58	DYNPARM - Control Use of Dynamic Parameters	131
59	ECHO - Control Printing of Batch Input Data	133
60	EDTBPSIZE - Software AG Editor Buffer Pool Size	135
61	EDTLFILES - Number of Software AG Editor Logical Files	137
62	EJ - Page Eject	139
63	EM - Edit Mask	141
	Syntax	142
	Leerzeichen in Editiermasken	143
	Standard-Editiermasken	144
	Editiermasken für numerische Felder	144
	Editiermasken für alphanumerische Felder	147
	Editiermasken für binäre Felder – Format B	149
	Hexadezimale Editiermasken	149
	Editiermasken für Datums- und Zeitfelder (Formate D und T)	152
	Editiermasken für logische Felder (Format L)	156

64 EMFM - Edit Mask Free Mode	159
65 EMU - Unicode Edit Mask	161
66 ENDIAN - Endian Mode for Compiled Objects	163
67 ENDMMSG - Display Session-End Message	165
68 ES - Empty Line Suppression	167
69 ESCAPE - Ignore Terminal Commands %% and %	169
70 ESXDB - Database ID Used for Entire System Server DDMs	171
71 ET - Execution of END/BACKOUT TRANSACTION Statements	173
72 ETA - Error Transaction Program	175
73 ETDB - Database for Transaction Data	177
74 ETEOP - Issue END TRANSACTION at End of Program	179
75 ETID - Adabas User Identification	181
76 ETPDB - Databases Containing Entire Transaction Propagator Master Files	183
77 ETPSIZE - Size of Entire Transaction Propagator Buffer	185
78 FC - Filler Character for INPUT Statement	187
79 FC - Filler Character for DISPLAY Statement	189
80 FCDP - Filler Character for Dynamically Protected Input Fields	191
81 FDDM - Natural System File for DDMs	193
82 FDIC - Predict System File	195
83 FL - Floating Point Mantissa Length	197
84 FNAT - Natural System File for System Programs	199
85 FREEGDA - Release GDA in Utility Mode	201
86 FS - Default Format/Length Setting for User-Defined Variables	203
87 FSEC - Natural Security System File	205
88 FUSER - Natural System File for User Programs	207
89 GC - Filler Character for Group Headers	209
90 GFID - Global Format IDs	211
91 HC - Header Centering	213
92 HD - Header Definition	215
93 HE - Helproutine	217
HE-Parameter-Syntax	218
Ausführung von Helproutinen	220
Beispiele	220
94 HI - Help Character	223
95 HW - Heading Width	225
96 IA - Input Assign Character	227
97 IC - Insertion Character	231
98 ICU - Unicode Insertion Character	233
99 ID - Input Delimiter Character	235
100 IKEY - Processing of PA and PF Keys	237
101 IM - Input Mode	239
102 INIT-LIB - Library for Automatic Logon	241
103 IP - INPUT Prompting Text	243
104 IS - Identical Suppress	245
105 KC - Check for Statement Keywords	247

106	KCHECK - Check for Statement Keywords	249
107	KD - Key Definition	251
108	LC - Lower to Upper Case Translation	253
109	LC - Leading Characters	255
110	LCU - Unicode Leading Characters	257
111	LDB - Wait Time for Response of Local Adabas Database	259
112	LE - Reaction when Limit for Processing Loop Exceeded	261
113	LFILE - Logical System File Definition	263
	LFILE Parameter Syntax	265
	Example of LFILE Parameter	265
114	LOGONRQ - Logon for RPC Server Request Required	267
115	LS - Line Size	269
	Profilparameter LS	270
	Session-Parameter LS	270
	Spezifikation in Statements	271
116	LT - Limit for Processing Loops	273
117	MADIO - Maximum DBMS Calls between Screen I/O Operations	275
118	MAINPR - Override Default Output Report Number	277
119	MASKCME - MASK Compatible with MOVE EDITED	279
120	MAXBUFF - Maximum Buffer Size	281
121	MAXCL - Maximum Number of Program Calls	283
122	MAXYEAR - Maximum Year for Date/Time Values	285
123	MC - Multiple-Value Field Count	287
124	MFSET - Multi-Fetch Setting	289
125	ML - Position of Message Line	291
126	MP - Maximum Number of Pages of a Report	293
127	MS - Manual Skip	295
128	MSGSF - Display System Error Messages in Short/Full Format	297
129	NATLOG - Natural Log File	299
	Examples	300
130	NATVERS - Switching between Natural Environments	303
131	NC - Use of Natural System Commands	305
132	NCFVERS - NCF File Protocol Version	307
133	NL - Numeric Length for Output	309
134	NOAPPLERR - Suppress Message Number Prefix NAT	311
135	OPF - Overwriting of Protected Fields by Help routines	313
136	OPRB - Database Open/Close Processing	315
	OPRB String Syntax	317
137	PARM - Alternative Parameter File	319
138	PC - Periodic Group Count	321
139	PCHECK - Parameter Checking for Object Calling Statements	323
140	PERSIST - Flag a Parameter File as Persistent	325
141	PM - Print Mode	327
	Profilparameter PM	328
	Session-Parameter PM	328




142 PRGPAR - Data to be Passed to Program Receiving Control at Termination	331
143 PROGRAM - Non-Natural Program Receiving Control after Termination	333
144 PS - Page Size for Natural Reports	335
145 PSIGNF - Internal Representation of Positive Sign of Packed Numbers	337
146 RCFIND - Handling of Response Code 113 for FIND Statement	339
147 RCGET - Handling of Response Code 113 for GET Statement	341
148 RDACTIVE - Activate Remote Debugger	343
149 RDNODE - Remote Debugger Node Name	345
150 RDPORT - Remote Debugger Port	347
151 RDS - Define Remote Directory Server	349
152 RECAT - Dynamic Recataloging	351
153 REINP - Issue Internal REINPUT Statement for Invalid Data	353
154 ROSY - Read-Only Access to System Files	355
155 RPCSDIR - Library for Service Directory	357
156 RTINT - Allow Runtime Interrupt	359
157 SA - Sound Terminal Alarm	361
158 SB - Selection Box	363
Anmerkungen zur Syntax	364
Anmerkungen zur Laufzeitumgebung	365
159 SD - Time Delay between Two Screens	367
160 SERVER - Start Natural Session as an RPC Server Session	369
161 SF - Spacing Factor	371
162 SG - Sign Position	373
163 SM - Programming in Structured Mode	375
164 SNAT - Sound a Bell at Syntax Error	377
165 SORTSZE - Size of Sort Buffer	379
166 SPODDEBUGPORT - Debugger Port for Debugging in the Context of SPoD	381
167 SRETAIN - Retain Source Format	383
168 SRVCMIT - Server Commit Time	385
169 SRVNAME - Name of RPC Server	387
170 SRVNODE - Name of Node	389
171 SRVTERM - Server Termination Event	391
172 SRVUSER - User ID for RPC Server Registry	393
173 SRVWAIT - Wait Time of RPC Server	395
174 SSIZE - Size of Source Area Allocated by the Editors	397
175 STACK - Place Data/Commands on the Stack	399
176 STARTUP - Program Name for System Variable *STARTUP	401
177 STEPLIB - Initial Setting for *STEPLIB System Variable	403
178 SUBCHAR - Substitution Character for Default Code Page	405
179 UTF8 - UTF-8 Format for Sources	407
180 SYMGEN - Generate Symbol Table	409
181 SYNERR - Control of Syntax Errors	411
182 TC - Trailing Characters	413
183 TCU - Unicode Trailing Characters	415
184 TD - Time Differential	417

185 TF - Translation of Database ID/File Number	419
186 THSEP - Dynamic Thousands Separator	421
187 THSEPCH - Thousands Separator Character	423
188 TIMEOUT - Wait Time for RPC Server Response	425
189 TMPSTORTUNIQU - Alternate Algorithm for Generating Sort Work File Names	427
190 TQ - Translate Quotation Marks	429
191 TQMARK - Translate Quotation Marks	431
192 TRACE - Define Trace Level for Natural RPC Servers	433
193 TRANSP - Server Transport Protocol	435
194 TRYALT - Try Alternative Server Address	437
195 UC - Underlining Character	439
196 UDB - User Database ID	441
197 ULANG - User Language	443
198 USEDIC - Common Logical Name for Dictionary Servers	445
199 USER - User ID	447
200 USEREP - Repository Usage	449
201 USIZE - Size of User Buffer	451
202 WEBIO - Use Natural Web I/O Interface	453
203 WFOPFA - Opening of Work Files	455
204 WH - Wait for Record in Hold Status	457
205 WORK - Work-File Assignments	459
206 XADB - Database for Coordination of Transaction Processing	461
207 XREF - Creation of XRef Data for Natural	463
Extended XRef Data Generation (For Internal Use Only)	465
208 YSLW - Year Sliding or Fixed Window	467
209 ZD - Zero-Division Check	469
210 ZP - Zero Printing	471

1 Parameter-Referenz

Diese Dokumentation enthält ausführliche Informationen zu den Natural-Profil- und Session-Parametern.

Die Beschreibungen der Profilparameter stehen nur in englischer Sprache zur Verfügung.

	Profilparameter – Einführung	Enthält allgemeine Informationen zur Benutzung von Profilparametern sowie eine Liste mit Referenzen auf verwandte Themen.
	Session-Parameter – Einführung	Enthält allgemeine Informationen zur Benutzung und Verarbeitung von Session-Parametern.
	Parameter in alphabetischer Reihenfolge	Beschreibungen sämtlicher Profilparameter und Session-Parameter in alphabetischer Reihenfolge.

Grundsätzliche Informationen zur Benutzung von Parametern finden Sie in der *Operations*-Dokumentation, siehe *Profile Parameter Usage*.

2 Profilparameter — Einführung

- Benutzung von Profilparametern 4
- Gemeinsame Beschreibungen für Profilparameter und Session-Parameter 4

Benutzung von Profilparametern

Da die Benutzung der Natural-Profilparameter auf den von Natural unterstützten Plattformen unterschiedlich ist, befinden sich die Informationen zur grundsätzlichen Benutzung der Parameter in den entsprechenden plattformspezifischen Dokumentationen.

Siehe die folgende Dokumente:

- Profile Parameter Usage in der *Operations*-Dokumentation
- Creating a New Parameter File in der *Configuration Utility*-Dokumentation
- Overview of Profile Parameters in der *Configuration Utility*-Dokumentation

Gemeinsame Beschreibungen für Profilparameter und Session-Parameter

Wenn es einen Natural-Session-Parameter mit gleichem Namen und gleicher Funktionalität wie ein Profilparameter gibt, sind die Beschreibungen für diese Parameter in einem gemeinsamen Dokument zusammengefasst.

3 Session-Parameter — Einführung

- Benutzung von Session-Parametern 6
- Setzen von Session-Parametern 6
- Verarbeitung von Session-Parametern 8

Benutzung von Session-Parametern

In Natural werden Session-Parameter verwendet:

- um bestimmte Zeichen festzulegen,
- um Verarbeitungszeitgrenzen festzulegen,
- um Reaktionen auf bestimmte Bedingungen festzulegen,
- um bestimmte Maximal- bzw. Minimalgrößen festzulegen,
- um bestimmte Aspekte der Reportformatierung festzulegen.

Bei der Installation von Natural legt der Natural-Administrator für einige dieser Parameter bestimmte Standardwerte fest, die dann für alle Natural-Benutzer gelten.

Die für Ihre Natural-Session gültigen Parameterwerte können Sie sich ansehen, wenn Sie das Systemkommando `GLOBALS` eingeben. Nähere Einzelheiten hierzu finden Sie in der *Systemkommandos*-Dokumentation).

Setzen von Session-Parametern

Es gibt verschiedene Möglichkeiten, die Werte der Session-Parameter zu setzen:

- über die Standard-Parameterdatei (`NATPARM`), die bei der Installation von Natural angelegt wird;
- über dynamische Parameter beim Aufrufen einer Natural-Session (wie in der *Operations*-Dokumentation beschrieben);
- mit dem Systemkommando `GLOBALS`;
- mit dem **Session Parameters**-Befehl im **Tools**-Menü von Natural Studio;
- mit dem Statement `SET GLOBALS` (nur im Reporting Mode);
- mit dem Statement `FORMAT`;
- mit Statements wie `INPUT`, `DISPLAY`, `WRITE`, in denen bestimmte Parameter verarbeitet werden;
- über Terminalkommandos.

Statt der Parameterwerte `ON` und `OFF` können Sie wahlweise auch die Werte `T` (True = wahr) bzw. `F` (False = falsch) verwenden.

Ändern von Parameterwerten auf Session-Ebene — Das GLOBALS-Kommando

Einige der vom Natural-Administrator festgelegten Standardwerte können Sie für die Dauer einer Natural-Session ändern.

Sie können die Parameterwerte innerhalb einer Session ändern, indem Sie das folgende Systemkommando verwenden:

```
GLOBALS
```

Wenn Sie das GLOBALS-Kommando ausführen, erhalten Sie einen Schirm, auf dem die für Ihre laufende Session derzeit gültigen Parameterwerte angezeigt werden. Auf diesem Schirm können Sie auch die Werte ändern, die nicht Ihren Anforderungen entsprechen.

Die mit einem GLOBALS-Kommando festgelegten Parameterwerte gelten bis zum Ende der jeweiligen Natural-Session (und gelten für jedes im Laufe der Session gespeicherte Objekt), es sei denn, Sie ändern sie wieder durch ein anderes GLOBALS-Kommando.

Ändern von Parameterwerten auf Programm-Ebene — Das FORMAT-Statement

Einige Parameter-Standardwerte können Sie für die Dauer eines Programms (Reports) ändern. Dies geschieht mit einem FORMAT-Statement im jeweiligen Programm, welches die betreffenden session-weit gültigen Parameter überschreibt.

Beispiel für ein FORMAT-Statement:

```
FORMAT AL=10 HC=R
```

Die mit einem FORMAT-Statement angegebenen Parameterwerte gelten bis zum Ende des ausgeführten Programms, wenn sie nicht vorher im Programm durch ein anderes FORMAT-Statement geändert werden. Sie können allerdings nur einen Teil der Session-Parameter programmspezifisch umsetzen.

Andererseits gibt es eine Reihe von Parametern, die Sie nur auf Programm-Ebene setzen können, nicht aber auf Session-Ebene. Dies sind überwiegend Parameter, die sich auf die Formatierung eines Reports auswirken.

Ändern von Parameterwerten auf Statement-Ebene

Die meisten Parameter, die Sie mit einem `FORMAT`-Statement festlegen können, können Sie auch auf Statement-Ebene setzen, zum Beispiel bei einem `DISPLAY`-, `WRITE`-, `INPUT`- oder `REINPUT`-Statement, und zwar durch Einfügen des Parameters (in Klammern) hinter dem Statement-Namen.

Beispiel:

```
DISPLAY (SF=4) NAME JOB-TITLE CURR-CODE SALARY
```

Ein auf Statement-Ebene angegebener Parameterwert gilt nur für das jeweilige Statement, und zwar vor allen auf anderen Ebenen für diesen Parameter angegebenen Werten.

Ändern von Parameterwerten auf Feld-Ebene

Innerhalb eines `DISPLAY`-, `WRITE`-, `INPUT`- oder `REINPUT`-Statements können Sie bestimmte Parameterwerte noch feldspezifisch festlegen, indem Sie den Parameterwert für ein einzelnes Feld oder Ausgabe-Element unmittelbar hinter dem betreffenden Feld (in Klammern) angeben.

Beispiel:

```
DISPLAY NAME (AL=10) JOB-TITLE CURR-CODE SALARY
```

Dieser Wert gilt dann nur für dieses Feld, und zwar vor allen auf anderen Ebenen für diesen Parameter angegebenen Werten. Die feldspezifische Angabe von Parameterwerten ist allerdings nur für einen Teil der auf Statement-Ebene beeinflussbaren Parameter möglich.

Verarbeitung von Session-Parametern

Session-Parameter, die in den Statements `DISPLAY`, `FORMAT`, `PRINT`, `INPUT`, `REINPUT`, `WRITE`, `WRITE TITLE` oder `WRITE TRAILER` gesetzt werden, werden während der *Programmkompilierung* verarbeitet und sind daher in dem betreffenden Objektmodul enthalten.

Bei der Auswertung der Parameterwerte gilt folgende Hierarchie:

1. für einzelne Felder/Elemente gesetzte Parameter (höchste Priorität)
2. auf Statement-Ebene gesetzte Parameter
3. mit einem `FORMAT`-Statement gesetzte Parameter
4. die Standardwerte der Parameter (niedrigste Priorität)

Mit einem `SET GLOBALS`-Statement gesetzte Parameter bewirken eine Veränderung der Laufzeit-Umgebung. Sie gelten solange, bis sie mit einem weiteren `SET GLOBALS`-Statement (oder `GLOBALS`-Systemkommando) geändert werden.

4 ACIVERS - Define ACI Version for Use with EntireX Broker

ACI

This Natural profile parameter specifies the API version to be used with the EntireX Broker ACI. The broker stub in use as well as the called EntireX Broker must support the ACI version defined here. Please, refer to the EntireX documentation for the supported API versions. The setting of ACIVERS enables special features of the EntireX Broker, depending on the API version you are using. For details, see *Setting Up an EntireX Broker Access* in the *Natural Remote Procedure Call* documentation.

ACIVERS can be specified on both the client and the server side.

Possible settings	1 - 9	Single-digit number. The higher the version, the more features are available.
Default setting	2	API Version 2 is used.
Dynamic specification	yes	
Specification within session	no	

For additional information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

5 ACTPOLICY - Defines Default Activation Policy for DCOM

Classes

This Natural profile parameter defines the default activation policy for DCOM classes. It is evaluated if a Natural class is registered as a DCOM class with the REGISTER command.

Possible settings	ES	External single activation is performed.
	EM	External multiple activation is performed.
	IM	Internal multiple activation is performed.
Default setting	EM	
Dynamic specification	yes	
Specification within session	no	

6 AD - Attribute Definition

▪ AD-Parameter-Syntax	17
▪ Feldanzeige	17
▪ Ausrichtung der Feldwerte	18
▪ Feldeingabe/ausgabe-Charakteristika	18
▪ Auslegung alphanumerischer Felder	20
▪ Eingabezwang	20
▪ Mindestlänge der Eingabewerte	21
▪ Groß-/Kleinschreibung	21
▪ Füllzeichen	21

Attribut-Definition

Mit diesem Session-Parameter definieren Sie Feldattribute auf Feld/Element- oder Statement-Ebene.

Verwandter Session-Parameter: [CD](#) - Farbdefinition.

Mögliche Werte	Siehe unten.	Sie können mehrere Attribute in beliebiger Reihenfolge angeben.
Standard-Einstellung	Siehe unten.	
Gültige Statements	FORMAT	
	DISPLAY INPUT NEWPAGE WITH TITLE PRINT REINPUT WRITE WRITE TITLE WRITE TRAILER	Parameter können auf Statement-Ebene und/oder Element-Ebene angegeben werden.
	ASSIGN CALLNAT CALLDBPROC COMPUTE MOVE OPEN DIALOG PERFORM SEND EVENT SEND METHOD	Parameter können auf Element-Ebene angegeben werden, allerdings können nur die in der betreffenden Statement-Beschreibung angegebenen Attribute benutzt werden.
Gültiges Kommando	Keines.	



Anmerkung: Der AD-Parameter kann auch bei Function Calls angegeben werden. Es können jedoch nur die Attribute verwendet werden, die im Abschnitt *Function Call* aufgeführt sind.

Die Parameter-Syntax und die Bedeutung der Attribute und die einzelnen Werte werden in den folgenden Abschnitten erklärt.

AD-Parameter-Syntax

```
AD=[field-representation] [field-alignment] [field-i/o-characteristics]
[interpretation-of-alphanumeric-fields] [mandatory-input] [input-value-length]
[field-upper/lower-case] [filler-character]
```

Sie können mehrere Attribute in beliebiger Reihenfolge angeben. Mögliche Werte sind:

```
AD= { B } { C } { D } { I } { N } { U } { V } { Y }
    { L } { R } { Z }
    { A } { M } { O } { P }
    { E } { G } { T }
    { E } { H } { W }
    { 'c' }
```

Die Bedeutung der Attribute und die einzelnen Werte werden in den folgenden Abschnitten erklärt.

Beispiele:

```
DISPLAY #FIELDA (AD=R)
INPUT #FIELDB (AD=M)
INPUT (AD=IM) #FIELDA #FIELDB
```

Feldanzeige

Wert	Bedeutung	Statements	Erklärung
B	blinkend (*)	ASSIGN	Der Feldwert wird blinkend angezeigt.
C	kursiv (*)	COMPUTE	Der Feldwert wird kursiv angezeigt.
D	nicht intensiviert	MOVE	
I	intensiviert	DISPLAY	Der Feldwert wird normal angezeigt, d.h. weder intensiviert noch sonstwie hervorgehoben. Dies ist die Voreinstellung.
N	nicht sichtbar	FORMAT	Der Feldwert wird intensiviert, d.h. hell leuchtend angezeigt.
U	unterstrichen	INPUT	
V	invers (*)	PRINT	Ein eingegebener Feldwert wird nicht angezeigt.
		REINPUT	Der Feldwert wird unterstrichen angezeigt.
		WRITE	Der Feldwert wird invers angezeigt, d.h. in farblicher Umkehrung von Hintergrund und Feldwert.

Wert	Bedeutung	Statements	Erklärung
Y	dynamische Attribute	INPUT DISPLAY PRINT WRITE	Feldattribute werden dynamisch über eine Kontrollvariable (Format C) zugewiesen.

* Die mit einem Stern (*) markierten Feldanzeige-Attribute sind an entsprechende Hardware-Voraussetzungen gebunden und werden zur Laufzeit ignoriert, falls diese Voraussetzungen nicht gegeben sind.

Ausrichtung der Feldwerte

Wert	Bedeutung	Statements	Erklärung
L	linksbündig	DISPLAY FORMAT	Feldwerte werden linksbündig im Feld ausgegeben. Dies ist die Voreinstellung für alphanumerische Felder.
R	rechtsbündig	INPUT PRINT REINPUT	Feldwerte werden rechtsbündig im Feld ausgegeben. Dies ist die Voreinstellung für numerische Felder.
Z	vorangestellte Nullen	WRITE	Feld werte werden rechtsbündig im Feld ausgegeben, der Rest des Feldes wird mit Nullen aufgefüllt.

Feldeingabe/ausgabe-Charakteristika

Wert	Bedeutung	Statements	Erklärung
A	Eingabefeld, ungeschützt	INPUT FORMAT	Ein Feldwert wird in Antwort auf ein INPUT-Statement eingegeben. Dies ist die Voreinstellung.
	nur Eingabe	CALLNAT CALLDBPROC OPEN DIALOG PERFORM SEND EVENT SEND METHOD Function Call	<p>Markieren Sie einen Parameter mit AD=A, wird sein Wert nicht an das aufgerufene Objekt übergeben (Subprogramm, Stored Procedure, Subroutine, Dialog, Methode), sondern er erhält einen Wert von dem aufgerufenen Objekt.</p> <p>Bei einem mit BY VALUE in der Parameter Data Area des aufgerufenen Objekts definierten Feld kann das aufrufende Objekt keinen Wert erhalten. In diesem Fall bewirkt AD=A nur, dass das Feld auf den niedrigeren Wert von dem betreffenden Format zurückgesetzt wird (Leerzeichen für alphanumerisch, binäre Nullen für binär und Nullen für numerische Felder), bevor das Objekt aufgerufen wird.</p> <p>Bei einem CALLNAT-Statement kann AD=A für abgesetzt betriebene Subprogramme nützlich sein, die über Natural RPC in einer Client/Server -Umgebung ausgeführt werden, um die Belastung der gesendeten Daten zu reduzieren. Wenn ein Subprogramm lokal</p>

Wert	Bedeutung	Statements	Erklärung
			<p>ausgeführt wird, werden mit AD=A markierte Felder auf den niedrigen Wert des betreffenden Formats zurückgesetzt, bevor das Objekt aufgerufen wird.</p> <p>Ist bei einem SEND METHOD-Statement keine Methode in Natural implementiert, ist das Verhalten von der Methoden-Implementierung abhängig. Der Parameter wird dann als initialisierte Variante übergeben. Ob die externe Komponente dann einen Wert zurückgeben kann, ist in der Dokumentation der externen Komponente beschrieben. Sie kann auch im Natural Komponenten-Browser eingesehen werden.</p>
M	Ausgabefeld, änderbar	INPUT FORMAT	Der Wert des Feldes soll bei der Ausführung des INPUT-Statements angezeigt werden, und der ausgegebene Wert kann vom Benutzer überschrieben werden. Das Feld ist ein Ausgabefeld und kann geändert werden.
	änderbar	CALLNAT CALLDBPROC OPEN DIALOG PERFORM SEND EVENT SEND METHOD Function Call	<p>Standardmäßig kann der übergebene Wert eines Parameters im aufgerufenen Objekt (Subprogramm, Stored Procedure, Subroutine, Dialog, Methode) geändert werden, und der geänderte Wert wird wieder zurück an das aufrufende Objekt übergeben, wo es den ursprünglichen Wert überschreibt.</p> <p>Bei einem mit BY VALUE in der Parameter Data Area des aufgerufenen Objekts definierten Feld wird kein Wert zurückgegeben.</p> <p>Ist bei einem SEND METHOD-Statement keine Methode in Natural implementiert, ist das Verhalten von der Methoden-Implementierung abhängig. Der Parameter wird dann BY REFERENCE übergeben. Ob die externe Komponente einen By Reference- oder By Value-Parameter akzeptiert, ist in der Dokumentation der externen Komponente beschrieben. Dies kann auch im Natural Komponenten-Browser eingesehen werden.</p>
0	Ausgabefeld, geschützt	INPUT FORMAT	Der Wert des Feldes soll bei der Ausführung des INPUT-Statements angezeigt werden. Das Feld ist ein reines Ausgabefeld und kann nicht geändert werden.
	nicht änderbar	CALLNAT CALLDBPROC OPEN DIALOG PERFORM SEND EVENT SEND METHOD Function Call	<p>Wenn Sie einen Parameter mit AD=0 markieren, kann der übergebene Wert im aufgerufenen Objekt (Subprogramm, Stored Procedure, Subroutine, Dialog, Methode) geändert werden, aber der geänderte Wert kann nicht an das aufrufende Objekt zurückgegeben werden, d.h. das Feld im aufrufenden Objekt behält seinen ursprünglichen Wert bei.</p> <p>Intern wird AD=0 genauso verarbeitet wie BY VALUE (siehe den Abschnitt <i>Definition von Parameterdaten</i> in der Beschreibung des DEFINE DATA-Statements).</p> <p>Wenn bei SEND METHOD eine Methode in Natural implementiert wird, wird der Parameter behandelt, so wie er von BY VALUE in der Parameter Data Area der Methode definiert wurde (siehe den</p>

Wert	Bedeutung	Statements	Erklärung
			<p>Abschnitt <i>PARAMETER-Klausel</i> in der Beschreibung des <i>INTERFACE-Statements</i>).</p> <p>Wenn bei <i>SEND METHOD keine</i> Methode in Natural implementiert ist, ist das Verhalten abhängig von der Implementierung der Methode. Der Parameter wird dann <i>BY VALUE</i> übergeben. Ob die externe Komponente einen <i>By Reference-</i> oder <i>By Value-Parameter</i> akzeptiert, ist in der Dokumentation der externen Komponente beschrieben. Dies kann auch im Natural Komponenten-Browser eingesehen werden.</p>
P	vorübergehend geschützt (P = protected)	INPUT REINPUT	Wird in Verbindung mit einer Kontrollvariablen (Format C), dem <i>DY-Parameter</i> (dynamische Attribute) und dem <i>REINPUT-Statement</i> verwendet.



Anmerkung: Die Feldeingabe/ausgabe-Charakteristika A, M und O des AD-Parameters können auch in Function Calls angegeben werden.

Auslegung alphanumerischer Felder

Wert	Bedeutung	Statements	Erklärung
Q	Alphanumerisches Feld so anzeigen als sei es ein numerisches Feld	ASSIGN COMPUTE MOVE DISPLAY FORMAT INPUT PRINT REINPUT WRITE	<p>Dieses Attribut ist nur auf Großrechnern verfügbar. Es ist eine entsprechende Hardware-Funktion erforderlich.</p> <p>Ein alphanumerisches Feld wird interpretiert, als ob es ein numerisches Feld wäre. Wenn das Feld im Bereich des Profil- oder Session-Parameters <i>PM</i> angezeigt wird, wird der Wert des Feldes von links nach rechts anstatt von rechts nach links interpretiert.</p>

Eingabezwang

Wert	Bedeutung	Statements	Erklärung
E	Eingabe erforderlich	INPUT FORMAT	Als Antwort auf ein <i>INPUT-Statement</i> muss ein Feldwert eingegeben werden; andernfalls erscheint eine Fehlermeldung. Dies ist nur bei reinen Eingabefeldern (<i>AD=A</i>) relevant.
F	Wert optional	INPUT FORMAT	Als Antwort auf ein <i>INPUT-Statement</i> kann ein Feldwert eingegeben werden, muss aber nicht. Dies ist die Voreinstellung.

Mindestlänge der Eingabewerte

Wert	Bedeutung	Statements	Erklärung
G	Wertlänge	INPUT FORMAT	Ein als Antwort auf ein INPUT-Statement eingegebener Wert muss genauso lang sein wie das Feld, d.h. das Feld muss vollständig gefüllt werden. Dies ist nur bei reinen Eingabefeldern (AD=A) relevant.
H	Wertlänge	INPUT FORMAT	Ein als Antwort auf ein INPUT-Statement eingegebener Wert darf kürzer sein als das Feld, d.h. das Feld muss nicht vollständig gefüllt werden. Dies ist die Voreinstellung.

Groß-/Kleinschreibung

Wert	Bedeutung	Statements	Erklärung
T	Kleinbuchstaben werden umgesetzt	INPUT FORMAT	Ein eingegebener Wert wird automatisch in Großbuchstaben umgesetzt, d.h. es ist egal, ob ein Wert in Klein- oder Großbuchstaben eingegeben wird.
W	Kleinbuchstaben werden akzeptiert	INPUT FORMAT	Es erfolgt keine Umsetzung von Klein- in Großbuchstaben, d.h. ein Wert wird so interpretiert, wie er eingegeben wird. Um AD=W einzuschalten, müssen Sie den Wert ON für den Natural-Profilparameter LC angeben. Dies ist die Voreinstellung.


Füllzeichen

Wert	Bedeutung	Statements	Erklärung
'c'	filler character	INPUT FORMAT	Der unbeschriebene Teil eines Feldes (für reine Eingabefelder) wird bei der Anzeige mit dem Zeichen c gefüllt, wenn AD=A (reines Eingabefeld, ungeschützt) oder AD=M (Ausgabefeld, änderbar) angegeben wird.

Bevor der Wert für ein änderbares Feld (AD=M) angezeigt wird, werden nicht von dem Wert belegte Feldstellen mit dem angegebenen Füllzeichen wie folgt gefüllt:

- Vorangestellte oder nachfolgende Zeichen werden (abhängig von der Feldausrichtung) bei Feldern mit dem Format I, N und P gefüllt.
- Nachfolgende Zeichen werden für Felder des Formats A gefüllt.

Wenn der Benutzer einen Wert als Antwort auf ein `INPUT`-Statement eingibt, bevor der Wert dem Feld zugewiesen wurde,

- werden sowohl vorangestellte als auch nachfolgende Füllzeichen bei Feldern mit den Formaten I, N und P entfernt
 - werden nachfolgende Füllzeichen bei Feldern des Formats A entfernt.
-  **Vorsicht:** Füllzeichen, die auftreten können als Teil des Wertes entweder an vorangestellten oder nachfolgenden Positionen sollten vermieden werden, um ungewünschte Ergebnisse zu verhindern.

Wenn beispielsweise das Füllzeichen 0 (Null) für ein Feld des Formats N5 definiert ist und der Wert 00100 als Eingabedaten eingegeben wird, werden vorangestellte und nachfolgende Nullen entfernt, so dass nur der Wert 1 übrig bleibt und dem Feld zugewiesen wird. Aus demselben Grund sollte das Minuszeichen (-) als Füllzeichen für numerische Felder vermieden werden, wenn negative Werte eingegeben werden sollen.

7 AL — Alphanumeric Length for Output

Alphanumerische Länge der Ausgabe

Mit diesem Session-Parameter geben Sie die standardmäßige Ausgabelänge für ein alphanumerisches Feld an, d.h. wenn es kürzer als die Feldlänge ist, wird das Feld rechts abgeschnitten.

Mögliche Werte	1 bis n	n = Wert des LS -Parameters (Zeilenlänge) minus 1).
Standard-Einstellung	none	
Gültige Statements	FORMAT	Parameter können dynamisch mit dem <code>FORMAT</code> -Statement angegeben werden.
	DISPLAY INPUT PRINT WRITE	Parameter können auf Statement-Ebene und/oder Element-Ebene angegeben werden.
Gültiges Kommando	Keines.	



Anmerkungen:

1. Es empfiehlt sich nicht, den Session-Parameter `AL` für Eingabefelder ([Attributdefinition](#) `AD=A` oder `AD=M`) in `INPUT`-Statements zu verwenden.
2. Eine für ein Feld definierte Editiermaske (siehe Session-Parameter [EM](#)) setzt den Session-Parameter `AL` für dieses Feld außer Kraft.

Beispiel:

```
FORMAT AL=20
```

See also *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

8 AUTO - Automatic Logon

This Natural profile parameter causes an automatic logon to a specific library at the start of the Natural session.

Possible settings	ON	An automatic logon is executed at the start of the Natural session.
	OFF	No automatic logon is performed.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

The setting of the profile parameter `INIT-LIB` is used as library ID for the logon.



Anmerkung: If used with Natural Security, `AUTO=ON` disables logons with another user ID, and the `INIT-LIB` parameter is not evaluated (see the *Natural Security* documentation for further information).

9

AUTOREGISTER - Automatic Registry Update

This Natural profile parameter determines whether Natural classes are to be registered as DCOM classes each time they are cataloged. Alternatively, Natural classes can be registered manually with the system command REGISTER. This parameter is used by NaturalX.

Possible settings	ON	Registry will be updated automatically.
	OFF	Registry will not be updated automatically.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

10 AUTORPC - Automatic Natural RPC Execution

This Natural profile parameter determines whether or not Natural RPC will automatically try to execute a subprogram remotely (on the server side) which was not found locally (on the client side). For details see *Stubs and Automatic RPC* in the *Natural Remote Procedure Call (RPC)* documentation.

AUTORPC is specified on the client side only.

Possible settings	ON	Natural RPC will automatically try to execute it remotely.
	OFF	Natural RPC will not automatically try to execute it remotely. If AUTORPC=OFF, you can execute CALLNATs remotely using stubs.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	yes	At runtime, this value can be overwritten using the Parameter Maintenance function of the SYSRPC utility.

For additional information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

11 BATCH - Batch Mode Simulation

This Natural profile parameter sets the system variable *DEVICE to "BATCH" when Natural is started.

Possible settings	ON	When Natural is started with profile parameter BATCH set, error messages are not displayed, but written to a log file. The log file is named <i>natbatch.log</i> and is located in the Natural binary directory.
	OFF	Error messages are displayed but not written to a log file.
Default setting	OFF	
Dynamic specification	yes	This parameter can only be specified dynamically.
Specification within session	no	

To run Natural in batch mode, use the parameter `BATCHMODE` instead of `BATCH`.

12 BATCHMODE - Batch Mode

This Natural profile parameter applies to batch mode only. It enables batch mode and sets the system variable *DEVICE to "BATCH" when Natural is started.

Possible settings	ON	Natural will run in batch mode.
Default setting	OFF	Natural will run in interactive mode, or in batch mode simulation.
Dynamic specification	yes	The parameter can only be specified dynamically.
Specification within session	no	

13

BMBLANK - Display Trailing Blanks

This Natural profile parameter applies to batch mode only.

It is used to control the display of trailing blanks in the batch output file **CMPRINT**.

This parameter applies only if the parameter **BMSIM** is set to "MF". Trailing blanks are generated automatically if **BMSIM** is set to "MF". **BMBLANK** has no effect if **BMSIM** is set to "OS" or "VM".

Possible settings	ON	Trailing blanks are written to CMPRINT .
	OFF	No trailing blanks are written to CMPRINT .
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

14

BMCONTROL - Display Control Characters

This Natural profile parameter applies to batch mode only.

It controls the output of control characters (such as form feed and line feed) in the batch output file `CMPRINT`.

Possible settings	ON	Control characters will be written to CMPRINT.
	OFF	No control characters will be written to CMPRINT.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

15 BMFRAME - Window Frame Characters

This Natural profile parameter applies to batch mode only.

With this parameter you can define window frame characters that will be written to the batch output file [CMPRINT](#).

Specify a sequence of 6 characters.

Possible settings	6 characters	Character Position:	Displays:
		1	Horizontal bar
		2	Vertical bar
		3	Upper-left corner
		4	Upper-right corner
		5	Lower-left corner
		6	Lower-right corner
Default setting	- !++++		
Dynamic specification	yes		
Specification within session	no		

Example:

To define the following frame you have to specify `BMFRAME=123456`.

```
3111111111114  
2          2  
2          2  
5111111111116
```

16 BMSIM - Similar Batch Mode Output

This Natural profile parameter applies to batch mode only.

It is used for the general appearance description of the batch mode output file [CMPRINT](#).

Possible settings	MF	<p>Forces output similar to Natural for Mainframe Version 2.3. Each line in CMPRINT is filled with trailing blanks.</p> <p>A control character appears at the beginning of each line of CMPRINT. The control character codes are similar to the IBM control character option ASA.</p> <p>The following control character codes are used:</p> <table border="1"> <thead> <tr> <th>Control Code</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>blank</td> <td>Normal output line without control characters</td> </tr> <tr> <td>0</td> <td>Insert one empty line</td> </tr> <tr> <td>-</td> <td>Insert two empty lines</td> </tr> <tr> <td>+</td> <td>Print this line twice (bold printing)</td> </tr> <tr> <td>1</td> <td>Form feed before printing this line</td> </tr> </tbody> </table>	Control Code	Interpretation	blank	Normal output line without control characters	0	Insert one empty line	-	Insert two empty lines	+	Print this line twice (bold printing)	1	Form feed before printing this line
	Control Code	Interpretation												
	blank	Normal output line without control characters												
	0	Insert one empty line												
	-	Insert two empty lines												
	+	Print this line twice (bold printing)												
	1	Form feed before printing this line												
	VM	<p>Forces output similar to Natural for OpenVMS Version 2.1.</p> <p>The following control character codes are used:</p> <table border="1"> <thead> <tr> <th>Control Code</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>\n</td> <td>Line feed before printing this line</td> </tr> <tr> <td>\f</td> <td>Form feed before printing this line</td> </tr> </tbody> </table> <p>The FIN command will be generated if it is not specified in the batch command file CMSYNIN.</p>	Control Code	Interpretation	\n	Line feed before printing this line	\f	Form feed before printing this line						
	Control Code	Interpretation												
	\n	Line feed before printing this line												
\f	Form feed before printing this line													
OS	<p>Forces output similar to Natural for UNIX or OpenVMS.</p> <p>The following control character codes are used:</p>													

		Control Code	Interpretation
		\n	Line feed before printing this line
		\f	Form feed before printing this line
Default setting	0S		
Dynamic specification	yes		
Specification within session	no		

17 BMTIME - Display Process Time

This Natural profile parameter applies to batch mode only.

It is used to display the elapsed and used CPU time consumed by the Natural process. This output will be written to the end of the batch output file [CMPRINT](#).

Possible settings	ON	The elapsed and used CPU time is written to the end of the batch output file.
	OFF	The output is not written to the batch output file.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

The time format is as follows:

DDDxHH:II:SS.UU

where:

- *DDD* is the number of days (at maximum 999)
- *x* is blank if *DDD* is less or equal to 999,
or "+" (plus sign) if *DDD* is greater than 999
- *HH* is the number of hours
- *II* is the number of minutes
- *SS* is the number of seconds
- *UU* is the number of hundredths of seconds

Example:

```
Used CPU time:  0 00:00:00.56  
Elapsed time:  0 00:00:16.20
```

18

BMTITLE - Display Window Title

This Natural profile parameter applies to batch mode only.

It is used to control the displaying of window titles in the batch output file [CMPRINT](#).

Possible settings	ON	A window title will be displayed in CMPrint.
	OFF	No window title will be displayed in CMPrint.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

19

BMVERSION - Display Natural Version

This Natural profile parameter is used to control the display of the Natural version including the startup and termination time.

Possible settings	ON	The Natural version and startup time are written to the very first line of the batch output file <code>CMPRINT</code> , the termination time is written at the end of <code>CMPRINT</code> .
	OFF	The Natural version and startup time are not written to <code>CMPRINT</code> .
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

20

BPID - Specify Buffer Pool ID

This Natural profile parameter specifies the name (ID) of the Natural buffer pool.

Possible settings	1 - 8 characters	Name of the Natural buffer pool.
Default setting	NATBP	
Dynamic specification	yes	
Specification within session	no	



Anmerkung: Do not delete the default buffer pool „NATBP“, as it is possible that Natural may not function properly anymore.

21

BPID2 - Specify Secondary Buffer Pool

This Natural profile parameter specifies the name (ID) of a secondary buffer pool.

When Natural runs with a read-only buffer pool as the primary buffer pool, objects missing in the read-only buffer pool cannot be loaded. To avoid this, Natural can attach during execution to a secondary standard buffer pool (which allows read/write access) and activate the missing objects there. For further information, see *Secondary Read/Write Buffer Pool* in the *Operations* documentation.

Possible settings	1 - 8 characters	Name of the secondary buffer pool.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

22

BPNAME - Name of Natural Global Buffer Pool

This Natural profile parameter specifies the name of the Natural global buffer pool.

Possible settings	1 - 8 characters, or blank	Name of the Natural global buffer pool. If BPNAME= ' ' (blank) is set, a <i>local</i> Natural buffer pool is used.
Default setting	blank	
Dynamic specification	yes	This parameter can only be specified dynamically.
Specification within session	no	

23

BPSFI - Object Search First in Buffer Pool

This Natural profile parameter determines the sequence in which a requested object that is to be executed is searched for in the buffer pool and in the system file(s).

You can choose between two search sequences:

Possible settings	ON	<p>Search Sequence 1 is used (search buffer pool first for all libraries, then the system file(s)).</p> <p>Natural looks for the object in the following sequence until it is found:</p> <ol style="list-style-type: none">1. in the buffer pool, first in the current library, then in one steplib after another, then in the two SYSTEM libraries;2. in the system file(s), first in the current library, then in one steplib after another, then in the two SYSTEM libraries. <p>For performance reasons, it is recommended that you set BPSFI=ON in production environments.</p> <p>Vorsicht: If you set BPSFI=ON, make sure that object names are unique across all libraries that are involved in the search. If objects with the same name exist in different libraries being searched, unpredictable results may occur.</p>
	OFF	<p>Search Sequence 2 is used (alternating search in buffer pool and system file(s) for each library).</p> <p>Natural looks for the object in the following sequence until it is found:</p> <ol style="list-style-type: none">1. in the current library, first in the buffer pool, then in the system file(s);2. in one steplib after another, first in the buffer pool, then in the system file(s) for each steplib;3. in the two SYSTEM libraries, first in the buffer pool, then in the system file(s) for each library.

		BPSFI=OFF is recommended in development environments to always get the most current object from your own current library.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

For further information, see *Steplibs* and *Search Sequence for Object Execution* in the *Using Natural Studio* documentation.

24

CC - Error Processing in Batch Mode

Fehlerverarbeitung im Batch-Modus

Mit diesem Profilparameter bestimmen Sie, was geschehen soll, wenn bei der Kompilierung oder Ausführung eines Natural-Programms im Batch-Betrieb ein Fehler auftritt.

Er gilt nicht, wenn vom Benutzer geschriebene Fehlerbehandlungsroutinen benutzt werden.

Mögliche Werte	ON	Die Eingabedaten für die Batch-Eingabedateien CMSYNIN und CMOBJIN werden gelöscht, und zwar bis zu einer Zeile, die mit %% an den beiden ersten Stellen beginnt, oder einer „End-of-File“-Bedingung. Folgen weitere Eingabedaten, so liest Natural nach der %%-Zeile weiter.	
	OFF	Natural versucht dann, das nächste Programm (oder Kommando) im Eingabedatenstrom auszuführen. Wenn alle Eingabedaten verarbeitet wurden, wird Natural mit Return Code 61 beendet. Der Natural-Fehler 9987 (Error occurred during execution/compilation.) wird in die Batch-Ausgabedatei geschrieben (wenn der Profilparameter ENDMSG auf "ON" gesetzt ist).	
Standard-Einstellung	OFF		
Dynamische Spezifikation			
Spezifikation in Session	ja	Gültige Statements:	
		Gültiges Kommando:	
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

25 CD - Color Definition

Farbdefinition

Mit diesem Session-Parameter bestimmen Sie die Farbe, in der Felder angezeigt werden. Falls kein Farb-Bildschirm verwendet wird, wird dieser Parameter zur Laufzeit ignoriert.

Verwandter Session-Parameter: [AD](#) - Attribute Definition.

Mögliche Werte	BL	blau
	GR	grün
	NE	neutral
	PI	rosa
	RE	rot
	TU	türkis
	YE	gelb
Standard-Einstellung	NE	
Gültige Statements	FORMAT	
	DISPLAY INPUT PRINT WRITE	Parameter können auf Statement-Ebene und/oder Element-Ebene angegeben werden.
	ASSIGN MOVE REINPUT	Parameter können auf Statement-Ebene angegeben werden.
Gültiges Kommando	Keines	

Beispiel:

```
INPUT (CD=RE) #A #B
```

26

CDYNAM - Dynamic Loading of Non-Natural Programs

This Natural profile parameter determines whether or not non-Natural programs are to be loaded dynamically by Natural.

Possible settings	ON	Any number of non-Natural programs can be loaded dynamically during the execution of a Natural program.
	OFF	Dynamic loading of non-Natural programs is not performed by Natural.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

27

CF - Character for Terminal Commands

Steuerzeichen für Terminalkommandos

Mit diesem Profil- und Session-Parameter bestimmen Sie das Steuerzeichen für Natural-Terminalkommandos, d.h. das Zeichen, mit dem die Terminalkommandos beginnen.

Innerhalb einer Natural-Session kann der Profilparameter CF vom Session-Parameter CF überschrieben werden.

Mögliche Werte	beliebiges Sonderzeichen	Ein Terminalkommando muss mit dem angegebenen Zeichen anfangen. Das mit dem CF-Parameter angegebene Zeichen <ul style="list-style-type: none"> ■ darf nicht dasselbe sein wie das mit dem HI-Parameter (Hilfezeichen) oder IA-Parameter (Input-Zuweisungszeichen) angegebene Zeichen. ■ sollte nicht dasselbe sein wie das mit dem DC-Parameter (Dezimalzeichen) oder ID-Parameter (Input-Delimiterzeichen) angegebene Zeichen. ■ Im Map-Editor ist das Steuerzeichen für Terminalkommandos stets „%“ (um Konflikte mit Delimiterzeichen zu vermeiden, die in Maps benutzt werden), egal welches Zeichen mit dem CF-Parameter definiert ist. 	
	OFF	Es steht kein Steuerzeichen für Terminalkommandos zur Verfügung. Mit SET CONTROL angegebene Terminalkommandos können jedoch nach wie vor ausgeführt werden.	
Standard-Einstellung	%	Ein Terminalkommando muss mit dem Zeichen % anfangen.	
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS

Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities-Dokumentation</i> .
---	----------	--

Unter Natural Security: Die Einstellung dieses Parameters kann von der *Session Parameters*-Option des Library-Profiles überschrieben werden.

28

CLEAR - Processing of CLEAR Key in NEXT Mode

This Natural profile parameter causes Natural to execute a specific Natural terminal command whenever CLEAR is pressed during program execution in NEXT mode.

Possible settings	any character	The default action can be overridden by supplying a character which, when appended to the terminal-command control character (as specified with the CF parameter), forms a valid Natural terminal command. Example: <pre>CF=% CLEAR=R</pre> Natural executes the terminal command %R when the CLEAR key is pressed in NEXT mode.
Default setting	%	By default, when the CLEAR key is pressed, Natural responds as if the user had entered the terminal command %.
Dynamic specification	yes	
Specification within session	no	

Under Natural Security: The setting of this parameter can be overridden by the Session Parameters option of the Library Profile.

29

CM - Command Mode

This Natural profile parameter can be used to suppress Natural command mode (NEXT and MORE).

Possible settings	ON	NEXT and MORE are available for command input.
	OFF	The Natural session will be terminated whenever NEXT is encountered; the MORE line will be write-protected (no input possible).
Default setting	ON	
Dynamic specification	no	
Specification within session	no	

30

CMOBJIN - Batch Input File for Natural INPUT Data

This Natural profile parameter applies to batch mode only.

It is used for data intended to be read by Natural INPUT statements. These types of data can alternatively be placed in the CMSYNIN file immediately following the relevant RUN or EXECUTE command. The number of characters actually processed is restricted to 512 characters per line.



Anmerkung: If the file name or path assigned to this parameter contains special characters (e.g. backslash) or spaces, the entire string must be enclosed in double quotes, see example below.

Possible settings	any string	CMOBJIN="C:\tmp\data.txt"
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

If the setting for the profile parameter CMSYNIN is equal to the setting of CMOBJIN, Natural reads input from CMSYNIN.

If an error occurs, Natural reacts in accordance with the setting of the profile/session parameter CC.

31

CMPRINT - Batch Output File

This Natural profile parameter applies to batch mode only.

It is used to specify the batch output file for the output report resulting from `DISPLAY`, `PRINT` and `WRITE` statement in a Natural program. In addition, Natural commands from `CMSYNIN` and `INPUT` data from `CMOBJIN` are written to `CMPRINT`.



Anmerkung: If the file name or path assigned to this parameter contains special characters (e.g. backslash) or spaces, the entire string must be enclosed in double quotes, see example below.

Possible settings	any string	<code>CMPRINT="C:\tmp\out.txt"</code>
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

32 CMPRTnn - Additional Report

This Natural profile parameter applies to batch mode only.

It is used for additional reports referenced by any Natural program executed during the session. *nn* is a two digit decimal number in the range from 01 to 31 corresponding to the LPT device used by a report in a DISPLAY, PRINT and WRITE statement.

Possible settings	any string	If the file name or path assigned to this parameter contains special characters, e.g. backslash (\) or spaces, the entire string must be enclosed in double quotes. Example: CMPRT07="C:\tmp\report7.txt"
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

In order to allow the user to specify variable print file names, alpha-format system variables and numeric counter markers may be embedded in the file name specification for CMPRT*nn*.

The supported alpha-format system variables are:

- *APPLIC-ID
- *APPLIC-NAME
- *DEVICE
- *ETID
- *INIT-USER
- *LIBRARY-ID
- *NET-USER
- *PID
- *PROGRAM

*USER
*USER-NAME

If any of these strings (in upper case only) is encountered within the print file specification, it will be replaced at run-time with the contents of the appropriate system variable. Additionally, a counter marker (#) may be used. This will be replaced by a 2-digit counter which will automatically be incremented for each print file.

Example:

The specification `CMPRT01=abc_*PID_*ETID_*PROGRAM_#.dat` in a Natural session with process ID 1234, ETID XYZ running a program with the name PRINT which produces print file output to File 01 would produce print files with the following names (assuming the program runs 3 times):

```
abc_1234_XYZ_PRINT_01.dat  
abc_1234_XYZ_PRINT_02.dat  
abc_1234_XYZ_PRINT_03.dat
```

See also *Using Natural in Batch Mode* in the *Operations* documentation.

33 CMSYNIN - Batch Input File for Natural Commands and INPUT Data

This Natural profile parameter applies to batch mode only.

CMSYNIN is used for the batch input file. It contains Natural commands and data to be read by INPUT statements during execution of Natural programs (optionally). The number of characters actually processed is restricted to 512 characters per line.



Anmerkung: If the file name or path assigned to this parameter contains special characters, e.g. backslash (\) or spaces, the entire string must be enclosed in double quotes, see example below.

Possible settings	any string	CMSYNIN="C:\tmp\cmd.txt"
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

See also *Using Natural in Batch Mode* in the *Operations* documentation.

34 CMWRKnn - Natural Work Files

This Natural profile parameter applies to batch mode only.

CMWRKnn is used for Natural work files referenced by any Natural program executed during the session.

nn is a two digit decimal number in the range from 01 to 32 corresponding to the number used in a READ WORK FILE or WRITE WORK FILE statement.



Anmerkung: If the file name or path assigned to this parameter contains special characters, e.g. backslash (\) or spaces, the entire string must be enclosed in double quotes, see example below.

Possible settings	any string	CMWRK05="C:\tmp\workfile5.sag"
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

See also *Using Natural in Batch Mode* in the *Operations* documentation.

35

COMPR - Set RPC Buffer Compression

This Natural profile and session parameter can be used to set the RPC buffer compression. It is effective only, if the automatic Natural RPC execution is used (`AUTORPC=ON`) and the `CALLNAT` is executed without a stub. If a stub is used, the compression has already been set during stub generation. For details, see *Using Compression* in the *Natural Remote Procedure Call* documentation.

COMPR is specified on the client side only.

Possible settings	0	No compression will be performed.
	1	The send buffer contains modifiable fields and output fields and the format buffer. The reply buffer contains modifiable fields and input fields.
	2	Same as <code>COMPR=1</code> , additionally the reply buffer also contains the format buffer.
Default setting	1	
Dynamic specification	yes	
Specification within session	yes	The setting of this parameter can be changed using the <code>GLOBALS</code> system command.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

36

COMSERVERID - Determine DCOM Server ID

This Natural profile parameter can be used to determine the DCOM server name (used by NaturalX).

Possible settings	not specified, or 1-32 characters	DCOM server name.
Default setting	not specified	If COMSERVERID is not specified, the default server name "DEFAULT" is used.
Dynamic specification	yes	
Specification within session	no	

37 CP - Default Code Page Name

This Natural profile parameter defines the default code page for Natural data and Natural sources.

If no code page is specified for a code page sensitive operation such as data conversions to and from Unicode (for example, by means of a statement specific `ENCODED` option or by another profile parameter), the default code page applies.

For the current Natural session, it is assumed that all code page data, for example, Natural sources, contents of A-format fields, etc., are stored in this code page format.

See also *Profile Parameters* in the *Unicode and Code Page Support* documentation.

Possible settings	1 - 64 characters	The name of the desired code page.
	' ' (blank)	Reset to system code page.
Default setting	' ' (blank)	System code page.
Dynamic specification	yes	
Specification within session	no	

38

CPCVERR - Code Page Conversion Error

Codepage-Umsetzungsfehler

Mit diesem Profil- und Session-Parameter legen Sie fest, ob eine Fehlermeldung angezeigt wird, wenn ein Umsetzungsfehler bei folgenden Umsetzungen auftritt:

- von Unicode nach Codepage oder
- von Codepage nach Unicode oder
- von einer Codepage in eine andere Codepage

Unabhängig davon enthält der Zieloperand das Ergebnis der Umsetzung, wobei alle Zeichen, die nicht umgesetzt werden können, durch ein für die betreffende Codepage durch die International Components for Unicode (ICU) festgelegtes Ersatzzeichen ersetzt werden. Dieser Parameter wird nicht bei der Umsetzung von Natural-Sourcen berücksichtigt, wenn diese in den Editierbereich geladen oder katalogisiert werden.

Auf Großrechnern wird nicht berücksichtigt, ob ein Unicode-Feld vor einer Eingabe/Ausgabe über eine Terminalemulation in die Codepage umgesetzt wird. In diesem Fall wird das Ersatzzeichen durch das in `NATCONFIG` definierte Platzhalterzeichen ersetzt.

Siehe auch:

- *Profile Parameters* in der *Unicode and Code Page Support*-Dokumentation.
- *Using an Error Transaction Program* in the *Programming Guide*

Mögliche Werte	ON	Ein Natural-Fehler NAT3413 wird ausgegeben, wenn während der ICU-Umsetzung mindestens ein Code Point nicht korrekt umgesetzt werden konnte.	
	OFF	Es wird kein Fehler ausgegeben, wenn ein oder mehrere Code Points während der ICU-Umsetzung nicht korrekt umgesetzt werden konnten.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS

39

CPOBJIN - Code Page of Batch Input File

This Natural profile parameter specifies the code page of the batch input file which is defined by the Natural profile parameter [CMOBJIN](#).

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	1 -64 characters	ICU code page name (IANA name recommended).
	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Default setting	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Dynamic specification	yes	
Specification within session	no	

40

CPPRINT - Code Page of Batch Output File

This Natural profile parameter specifies the code page of the batch output file which is defined by the Natural profile parameter [CPPRINT](#).

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	1 - 64 characters	ICU code page name (IANA name recommended).
	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Default setting	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Dynamic specification	yes	
Specification within session	no	

41 CPRPC - Define Code Page Name

This Natural profile parameter specifies the name of the code page used by the EntireX Broker. Currently, it applies only to the Natural RPC facility when the transport protocol ACI (that is EntireX Broker) is used.

For more information about EntireX Broker refer to the section about Software AG's Internationalization in the EntireX Broker documentation.

See also *Unicode and Code Page Support, Configuration and Administration of the Unicode/Code Page Environment, Profile Parameters*.

CPRPC can be specified on both the client and the server side.

Possible settings	1 to 40 characters	Valid code page name of EntireX Broker.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

42

CPSYNIN - Code Page of Batch Input File for Commands

This Natural profile parameter specifies the code page of the batch input file for commands which is defined by the Natural profile parameter [CMSYNIN](#).

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	1 - 64 characters	ICU code page name (IANA name recommended).
	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Default setting	' ' (blank)	The code page resulting from the evaluation of the profile parameter CP is used.
Dynamic specification	yes	
Specification within session	no	

43 CV - Attribute Control Variable

Kontrollvariable

Mit diesem Session-Parameter wird eine Kontrollvariable referenziert. Eine Kontrollvariable wird mit Format C (siehe *Spezielle Formate* im *Leitfaden zur Programmierung*) definiert und dient folgenden Zwecken:

- Feldern dynamisch Attribute zuweisen und/oder
- den Status "modifiziert" eines Feldes in Verbindung mit einem INPUT- oder PROCESS PAGE-Statement prüfen; siehe auch *Logische Bedingungen*, *MODIFIED-Option* im *Leitfaden zur Programmierung*.

Mögliche Werte	B, C, D, I, N, U, V	Felddarstellungsattribute (siehe Session-Parameter AD).
	P	Feldschutz (siehe Session-Parameter AD).
	BL, GR, NE, PI, RE, TU, YE	Farbe (eine Erläuterung der Farbcodes entnehmen Sie dem Session-Parameter CD).
Standard-Einstellung	Keine	
Gültige Statements	DISPLAY INPUT PRINT PROCESS PAGE WRITE	Parameter können auf Statement- und/oder Element-Ebene angegeben werden.
Gültiges Kommando	Keine	

Beispiel:

```
DEFINE DATA LOCAL
1 #ATTR(C)
1 #A (N5)
END-DEFINE
...
MOVE (AD=I CD=RE) TO #ATTR
INPUT #A (CV=#ATTR)
...
```

Mittels einer Kontrollvariablen und der MODIFIED-Option des IF-Statements kann auch überprüft werden, ob während der Ausführung eines INPUT- oder PROCESS PAGE-Statements Feldinhalte verändert worden sind.

```
IF #ATTR MODIFIED ...
```

Eine einzelne Attribut-Kontrollvariable kann auch zu mehreren Eingabefeldern zugewiesen werden, indem man sie einmal auf Statement-Ebene und mehrere Male auf Element-Ebene (Feldebene) angibt. In diesem Fall wird die Modified-Statusanzeige gesetzt, wenn eines der Felder, die die Kontrollvariable referenzieren, geändert worden ist.

Wenn Sie den CV-Parameter auf Statement-Ebene und auf Feldebene angeben und die Kontrollvariable für das betreffende Feld leer ist, wird die Attribut-Kontrollvariable für das Statement auch für das Feld genommen.

Die Attribut-Kontrollvariable kann auf maximal drei Dimensionen erweitert werden, zum Beispiel, CONTR(*), CONTR(*,*), CONTR(*,*,*), je nach Rang des entsprechenden Array.

44

CVMIN - Control Variable Modified at Input

This Natural profile parameter determines whether or not an attribute control variable is assigned the status `MODIFIED` when the setting of the field to which the attribute control variable is attached is overwritten by an *identical* setting. If an attribute control variable has been assigned the status `MODIFIED`, the `MODIFIED` option evaluates this as `TRUE`. This applies regardless of whether the input was entered manually, read from the stack or supplied in batch mode.

Possible settings	ON	If a field setting is overwritten by the same setting, the corresponding control variable will be assigned the status <code>MODIFIED</code> .
	OFF	If a field setting is overwritten by the same setting, the corresponding control variable will not be assigned the status <code>MODIFIED</code> .
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

45

DBSHORT - Interpretation of Database Field Short Names

This Natural profile and session parameter can be used to determine the interpretation of database field short names. It corresponds to the option `DBSHORT` of the Natural system command `COMPOPT`.

A database field defined in a DDM is described by two names:

- the short name with a length of 2 characters, used by Natural to communicate with the database (especially with Adabas);
- the long name with a length of 3-32 characters (1-32 characters, if the underlying database type accessed is DB2/SQL), which is supposed to be used to reference the field in the Natural programming code.

Under special conditions, you may reference a database field in a Natural program with its short name instead of the long name. This applies if running in Reporting Mode without Natural Security and if the database access statement contains a reference to a DDM instead of a view.

The decision if a field name is regarded as a short-name reference depends on the name length. When the field identifier consists of two characters, a short-name reference is assumed; a field name with another length is considered as a long-name reference. This standard interpretation rule for database fields can additionally be influenced and controlled by setting the compiler option `DBSHORT` to `ON` or `OFF`:

Possible settings	ON	The usage of a short name is allowed for referencing a database field. However, a data base short name is <i>not permitted</i> in general (even if <code>DBSHORT=ON</code>) <ul style="list-style-type: none">■ for the definition of a field when a view is created;■ when a view field is used in the programming code;■ when a <code>DEFINE DATA LOCAL</code> statement was previously used to defines variables;■ when running under Natural Security.
--------------------------	----	---

	OFF	<p>A database field may only be referenced via its long name. Every database field identifier is considered as a long-name reference, regardless of its length.</p> <p>If a two character name is supplied which can only be found as a short name but not as a long name, syntax error NAT0981 is raised at compile time.</p> <p>This makes it possible to use long names defined in a DDM with 2-byte identifier length. This option is essential if the underlying database you access with this DDM is SQL (DB2) and table columns with a two character name exist. For all other database types (for example, Adabas), however, any attempt to define a long field with a 2-byte name length will be rejected at DDM generation.</p> <p>Moreover, if no short-name references are used (what can be enforced via DBSHORT=OFF), the program becomes independent of being compiled without Natural Security.</p>	
Default setting	ON		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	Either GLOBALS or DBSHORT option of COMPOPT

Examples:

Assume the following data base field definition in the DDM EMPLOYEES:

Short Name	Long Name
AA	PERSONNEL - ID

Example 1:

```

OPTIONS DBSHORT=ON
READ EMPLOYEES
  DISPLAY AA      /* data base short name AA is allowed
END
    
```

Example 2:

```
OPTIONS DBSHORT=OFF
READ EMPLOYEES
  DISPLAY AA      /* syntax error NAT0981, because DBSHORT=OFF
END
```

Example 3:

```
OPTIONS DBSHORT=ON
DEFINE DATA LOCAL
1 V1 VIEW OF EMPLOYEES
  2 PERSONNEL-ID
END-DEFINE
READ V1 BY PERSONNEL-ID
  DISPLAY AA      /* syntax error NAT0981, because PERSONNEL-ID is defined in view;
                  /* (even if DBSHORT=ON)
END-READ
END
```


46

DBUPD - Database Updating

This Natural profile parameter indicates whether database updating is to be permitted during the Natural session.

Possible settings	ON	Database update is permitted.
	OFF	Database update is not permitted. A Natural statement which would cause a database update (STORE, UPDATE, DELETE) is not executed and an error message is generated.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.
	USR1042N *	* Recommended.

The DBUPD setting has no effect on the execution of Natural system commands.

47

DC - Character for Decimal Point Notation

Dezimaltrennzeichen

Mit diesem Profil- und Session-Parameter bestimmen Sie, welches Zeichen als Dezimaltrennzeichen (Komma bzw. Punkt) verwendet wird.

Innerhalb einer Natural-Session kann der Profilparameter `DC` vom Session-Parameter `DC` überschrieben werden.

Mögliche Werte	beliebiges Sonderzeichen	<p>Das mit dem <code>DC</code>-Parameter angegebene Zeichen gilt für alle Notationen, bei denen ein Dezimaltrennzeichen möglich ist; d.h. Variablen, Konstanten und Editiermasken. Das mit dem <code>DC</code>-Parameter angegebene Zeichen</p> <ul style="list-style-type: none"> ■ darf nicht dasselbe sein wie das mit dem Profil/Session-Parameter <code>IA</code> (Input-Zuweisungszeichen), oder auch dem Profil/Session-Parameter <code>ID</code> (Input-Delimiterzeichen) angegebene Zeichen. ■ sollte nicht dasselbe sein wie das mit dem Profil/Session-Parameter <code>CF</code> (Steuerzeichen für Terminalkommandos) oder Profilparameter <code>HI</code> (Hilfezeichen) angegebene Zeichen. 		
Standard-Einstellung	. (Punkt)			
Dynamische Spezifikation	ja			
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS	Parameter wird zur Laufzeit ausgewertet.
		Gültiges Kommando:	GLOBALS	Parameter kann dynamisch mit dem Systemkommando <code>GLOBALS</code> angegeben werden.

Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities-Dokumentation</i> .
---------------------------------------	----------	--



Anmerkung: Wenn Sie die DC-Einstellung in Ihrer Parameterdatei ändern, müssen Sie die DDM neu speichern, die Sie in Ihrem Natural-Programm verwenden, das eine neue .NSD-Datei auf der Platte speichert.

Unter Natural Security: Die Einstellung dieses Parameters kann von der Session Parameters-Option des Library-Profiles außer Kraft gesetzt werden.

48 DD - Day Differential

This Natural profile parameter is used to adjust the current machine date (as read by using the internal machine time) by adding/subtracting any number of days to/from it. This makes it possible to re-run an application that was to be run at a certain date but for some reason could not be run at that date.

The DD profile parameter is specified as follows:

DD=+nn

or

DD=-nn

where *nn* is the number of days.

Possible settings	-10953 to +10953	Machine date is adjusted. Specification of „+“ is optional.
	0	No adjustment is made.
Default setting	0	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See SYSEXT - <i>Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

See also the profile parameter [TD](#).

49 DF - Date Format

Datumsformat

Mit dem DF-Session-Parameter bestimmen Sie die Länge eines in alphanumerische Darstellung umgesetzten Datums, wenn hierfür keine Editiermaske angegeben ist. Der DF-Parameter wird zur Kompilierungszeit ausgewertet.

Die Reihenfolge der Tages-, Monats- und Jahreskomponenten sowie die Delimiterzeichen werden durch den DTFORM-Profilparameter bestimmt.

Mögliche Werte	S	8-Byte-Darstellung mit 2-stelliger Jahreskomponente und Delimitern (<i>yy-mm-dd</i>). Bei DF=S stehen nur 2 Stellen für die Jahres-Informationen zur Verfügung, d.h. dass wenn der Datumswert das Jahrhundert enthalten würde, gingen diese Informationen bei der Konvertierung verloren.
	I	8-Byte-Darstellung mit 4-stelliger Jahreskomponente und keinen Delimitern (<i>yyyymmd</i>). * Siehe Anmerkung.
	L	10-Byte-Darstellung mit 4-stelliger Jahreskomponente und Delimitern (<i>yyyy-mm-dd</i>). * Siehe Anmerkung.
Standard-Einstellung	S	
Gültige Statements	FORMAT	
	INPUT DISPLAY WRITE PRINT	Parameter kann dynamisch mit dem FORMAT-Statement angegeben werden.
	MOVE COMPRESS STACK RUN FETCH	Parameter kann auf Element-Ebene angegeben werden.

Gültiges Kommando	Keines
--------------------------	--------

Wenn der Wert eines Datumsfeldes in alphanumerisches Format umgesetzt wird (z.B. in einem MOVE-, DISPLAY-, WRITE- oder INPUT-Statement) und für die Umsetzung keine Editiermaske angegeben ist, wird das durch den Profilparameter `DTFORM` bestimmte Standarddatumsformat als Editiermaske genommen.

Dasselbe gilt bei der Eingabeauswertung einer in einem INPUT-Statement verwendeten Datumsvariablen: Wenn keine Editiermaske angegeben ist, wird die Eingabe entsprechend des durch den Profilparameter `DTFORM` bestimmten Datumsformats ausgewertet.



Anmerkung: Mit `DF=I` bzw. `DF=L` können Sie Ihre Anwendungen nach und nach auf 4-stellige Jahresdarstellung umstellen und dabei weiterhin die durch den `DTFORM`-Parameter gebotene Flexibilität ausnutzen.

Siehe auch den Abschnitt *Datumsformat für alphanumerische Darstellung – der DF-Parameter im Leitfaden zur Programmierung*.

50 DFOUT - Date Format for Output

Datumsformat für Ausgabe

Dieser Natural Profil- und Session-Parameter bestimmt die Form, in dem die Einstellungen der Datumsvariablen von INPUT-, DISPLAY-, PRINT- und WRITE-Statements angezeigt werden.

In einer Natural-Session kann der Profilparameter DFOUT durch den Session-Parameter DFOUT überschrieben werden.

Mögliche Werte	S	Datumsvariablen werden mit einer 2-stelligen Jahreskomponente und Delimitern angezeigt, wie vom Profilparameter DTFORM festgelegt. Beispiel: <i>yy-mm-dd</i>	
	I	Datumsvariablen werden mit einer 4-stelligen Jahreskomponente ohne Delimiter angezeigt. Beispiel: <i>yyyymmdd</i>	
Standard-Einstellung	S		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	



Anmerkungen:

1. Der *Profilparameter* DFOUT wird zur Laufzeit ausgewertet. Er gilt für Datumsfelder in INPUT-, DISPLAY-, PRINT- und WRITE-Statements, für die weder explizit Editiermasken angegeben sind noch der *Session-Parameter* DF gesetzt ist.
2. Bei beiden DFOUT-Einstellungen wird die Reihenfolge der Tages-, Monats- und Jahreskomponenten in den Datumswerten durch den Profilparameter DTFORM bestimmt.

Siehe auch *Datumsinformationen verarbeiten* im *Leitfaden zur Programmierung*.

51 DFS - Specify RPC Client's Default Server Address

This Natural profile parameter can be used to define an RPC default server address. It determines the server name, the server node, the logon indicator and the transport protocol. The default server address will be used only if no appropriate server is found in the service directory. For further information, see *Specifying RPC Server Addresses* in the *Natural Remote Procedure Call (RPC)* documentation.

To define a default server address, you specify up to 4 subparameters.

DFS is specified on the client side only.

Possible settings	<i>server-name</i>	(1 - 192 characters). There is no default, the value must be specified.		
	<i>server-node</i>	(1 - 192 characters). There is no default, the value must be specified.		
	<i>logon-indicator</i>	L	The client initiates a Natural logon to the server with the library name of the current library on the client.	
		(blank)	No server logon will be executed.	
	If nothing is specified, blank is the default. Note for Windows platforms: Instead of specifying L, check the selection box.			
	<i>transport-protocol-name</i>	The transport protocol to be used. ACI is the only possible value and the default.		

	<i>service-directory-indicator</i>	SERVDIR	A service directory must be present before the DFS profile parameter is evaluated.
		NOSERVDIR	No service directory is used before the DFS profile parameter is evaluated; that is, a service directory needs not be available on the client side.
		If nothing is specified, SERVDIR is the default.	
Default setting	none	Subparameter defaults, see above.	
Dynamic specification	yes	See below.	
Specification within session	yes	At runtime, this value can be overwritten using the Natural application programming interface USR2007N.	
Application Programming Interface	USR2007N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.	

For dynamic specification, the syntax is as follows:

```
DFS=(server-name,server-node,logon-indicator,transport-protocol-name,service-directory-indicator)
```

For the possible values for *server-name* and *server-node*, refer to [SRVNAME](#) and [SRVNODE](#).

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

52

DFSTACK - Date Format for Stack

Datumsformat für Stack

Dieser Natural Profil- und Session-Parameter bestimmt die Form, in der die Werte von Datumsvariablen mit einem STACK-, RUN- oder FETCH-Statement auf dem Stack abgelegt werden.

In einer Natural-Session kann der Profilparameter DFSTACK durch den Session-Parameter DFSTACK überschrieben werden.

Mögliche Werte	S	Datumsvariablen werden mit 2-stelliger Jahreskomponente und mit durch den Profilparameter DTFORM bestimmten Delimiterzeichen auf dem Stack abgelegt. Beispiel: <i>yy-mm-dd</i>
	C	Wie DFSTACK=S. Außerdem wird eine Änderung des Jahrhunderts (d.h. wenn das Jahrhundert, das genommen wird, wenn der Wert vom Stack gelesen wird, nicht dasselbe ist wie das Jahrhundert des ursprünglichen Datumswertes, bevor er auf dem Stack abgelegt wurde) zur Laufzeit abgefangen und ein Laufzeitfehler ausgegeben.
	I	Datumsvariablen werden mit vollständiger 4-stelliger Jahreskomponente und ohne Delimiterzeichen auf dem Stack abgelegt. Beispiel: <i>yyyymmdd</i>
Standard-Einstellung	S	
Dynamische Spezifikation	ja	
Spezifikation in Session	ja	Gültige Statements: SET GLOBALS

		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities-Dokumentation</i> .	

Der DFFSTACK-Parameter gilt nicht für STACK-, RUN- und FETCH-Statements, für die der Session-Parameter **DF** gesetzt ist.

Siehe auch *Datumsinformationen verarbeiten im Leitfaden zur Programmierung*.

53

DFTITLE - Output Format of Date in Standard Report Title

Datumsformat in Standard-Seitenüberschrift

Dieser Natural Profil- und Session-Parameter bestimmt die Form des Datums in einer Standard-Seitenüberschrift einer Reportseite (wie sie mit einem `DISPLAY-`, `WRITE-` oder `PRINT-Statement` ausgegeben wird).

In einer Natural-Session kann der Profilparameter `DFTITLE` durch den Session-Parameter `DFTITLE` überschrieben werden.

Mögliche Werte	S	Das Datum wird mit 2-stelliger Jahreskomponente und Delimiterzeichen ausgegeben. Beispiel: <i>yy-mm-dd</i>	
	L	Das Datum wird mit 4-stelliger Jahreskomponente und Delimiterzeichen ausgegeben. Beispiel: <i>yyyy-mm-dd</i>	
	I	Das Datum wird mit 4-stelliger Jahreskomponente ohne Delimiterzeichen ausgegeben. Beispiel: <i>yyyymmdd</i>	
Standard-Einstellung	S		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS

Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities-Dokumentation</i> .
---------------------------------------	----------	--



Anmerkungen:

1. Der DFTITLE-Parameter wird zur Laufzeit ausgewertet und bestimmt, ob das Datum mit einer 2-stelligen oder 4-stelligen Jahreskomponente mit oder ohne Delimiter angezeigt wird. Er hat keine Auswirkungen auf benutzerdefinierte Seitenüberschriften (wie sie mit einem WRITE TITLE-Statement angegeben werden).
2. Die Reihenfolge der Tages-, Monats- und Jahreskomponenten sowie die Delimiterzeichen werden durch den Profilparameter DTFORM bestimmt.

Siehe auch *Datumsinformationen verarbeiten* und *Datumsformat für Standard-Seitenüberschriften – Der DFTITLE-Parameter im Leitfaden zur Programmierung*.

54 DL - Display Length for Output

Ausgabelänge

Mit diesem Session-Parameter bestimmen Sie die Ausgabelänge eines Felds des Formats A oder U. Die Standardausgabelänge ist die Länge des Feldes.

Mögliche Werte	1 bis n	n = Wert des LS-Parameters (Line Size) minus 1
Standard-Einstellung	keine	
Gültige Statements:	FORMAT	
	DISPLAY INPUT PRINT WRITE	Parameter kann auf Statement- und/oder Element-Ebene angegeben werden.
Gültiges Kommando:	keines	

Beispiel:

```
FORMAT DL=20
```

Weitere Informationen und ein Beispiel für die Benutzung des Sessionparameters DL finden Sie in den folgenden Abschnitten des *Leitfadens zur Programmierung*:

- *Parameter zur Beeinflussung der Ausgabe von Feldern*
- *Ausgabelänge — der AL- und der NL-Parameter*
- *Ausgabelänge — der DL Parameter*

55

DTFORM - Date Format

This Natural profile parameter indicates the default format in which dates are to be provided automatically by Natural as part of the default title on Natural reports, as date constants and date input.

Possible settings	Value	Area	Date Format
	E	Europe	DD/MM/YYYY
	G	Germany	DD.MM.YYYY
	I	International	YYYY-MM-DD
	U	USA	MM/DD/YYYY
Default setting	I		
Dynamic specification	yes		
Specification within session	no		
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.	

The first day of a week is assumed to be Monday - unless DTFORM=U is specified, in which case Sunday is used.

For date constants, the year component (YYYY) consists of all four digits. Only the last two digits of the year component are used for reports, date input, the Natural system function VAL, and when the date is moved to an alphanumeric field.

The output format of the date in a default report page title is also specified by the profile parameter [DFTITLE](#).

See also *Processing of Date Information* and *Default Edit Mask for Date - DTFORM Parameter* in the *Programming Guide*.

56 DU - Dump Generation

Dump-Generierung

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, ob ein disassemblierter Objekt-Code-Dump generiert wird.

In einer Natural-Session kann der Profilparameter `DU` durch den Session-Parameter `DU` überschrieben werden.

Mögliche Werte	ON	<p>Wenn eines der Systemkommandos <code>CHECK</code>, <code>STOW</code>, <code>CATALOG</code>, <code>RUN</code> oder <code>CATALL</code> ausgeführt wird, wird ein disassemblierter Objekt-Code-Dump produziert.</p> <p>Diese Dump-Datei wird in das Verzeichnis geschrieben, das im Natural Configuration Utility mit der <code>Natural-TMP-Verzeichnisoption</code> definiert ist; siehe <i>Local Configuration File, Installation Assignments</i>.</p> <p>Der Name dieser Dump-Datei besteht aus dem Source-Dateiname und der Erweiterung <code>.DIA</code>. Wenn die Source-Datei nicht gespeichert wurde, ist der Name der Dump-Datei <code>GEN.DIA</code>. Wenn das Programm Datenbankzugriffs-Statements enthält, werden auch Dump-Dateien mit der Erweiterung <code>.ADA</code> (für Adabas) oder <code>.SQL</code> (für SQL-Datenbanken) erstellt. Wenn <code>XREF</code>-Daten generiert werden, wird eine Dump-Datei <code>.XRF</code> erstellt.</p> <p>Anmerkung: <code>DU=ON</code> kann eine große Dump-Datei erstellen (in Abhängigkeit von der Größe der Source-Datei), die eine erhebliche Verschlechterung der Systemperformanz verursachen kann.</p>	
	OFF	Keine Dump-Datei wird erstellt.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS

Unter Natural Security: Die Einstellung dieses Parameters kann durch die *Session Parameters*-Option des Library-Profiles überschrieben werden.

57

DY - Dynamic Attributes

- DY-Parametersyntax 126
- Beispiele 128

Dynamische Attribute

Mit diesem Session-Parameter werden Anzeigefeldern dynamisch Feldattribute zugewiesen

Mögliche Werte		Siehe unten.
Standard-Einstellung	Keine	
Gültige Statements	DISPLAY INPUT PRINT WRITE	Parameter kann auf Statement- und/oder Element-Ebene angegeben werden.
Gültiges Kommando	Keine	

Anfang und Ende einer Attribut-Definition werden mit besonderen Identifizierungszeichen (Escape-Zeichen) markiert.

Ein alphanumerisches Feld, das in einem INPUT-, DISPLAY-, WRITE- oder PRINT-Statement verarbeitet wird und Identifizierungszeichen enthält, wird an den Identifizierungszeichen getrennt und in Teilfelder aufgeteilt. Dann wird dem Teilfeld das entsprechende Attribut zugeordnet, und die Identifizierungszeichen werden durch Leerzeichen ersetzt.



Anmerkung: Wenn für einen Teil eines Feldes eine mit dem DY-Parameter gemachte Angabe wirksam ist, dann bleiben die aktuellen Felddarstellungs- und Farbattribute gültig, wenn in der DY-Parameterangabe keine neuen Einstellungen vorgenommen werden. Das bedeutet, dass die Feldfarbe nur dann durch ein DY-Parameterattribut geändert wird, wenn der DY-Parameter selbst eine neue neue Farbe festlegt. Das gilt ebenfalls für die **Feldanzeige**-Attribute, z. B. (AD=B, C, D, I, N, U, V) .

DY-Parametersyntax

```
DY={{escape-character1} [color-attribute] [i/o-characteristics]
[field-representation-attribute]} ... {escape-character2}
```

Die möglichen Werte werden im Folgenden erläutert.

escape-character1

Ein Zeichen, das den Anfang der Attribut-Definition markiert. Als Zeichen können Sie ein beliebiges Sonderzeichen (c) oder eine Hexadezimalzahl mit einem vorangestellten Apostroph ('xx) verwenden.

color-attribute

Das zuzuordnende Farbattribut. Erklärung der Farbcodes siehe Session-Parameter [CD](#).

BL	blau
GR	grün
NE	neutral
PI	rosa
RE	rot
TU	türkis
YE	gelb

i/o-characteristics

Wert	Bedeutung
P	Unterfeld soll schreibgeschützt sein.

Es kann ein P angegeben werden, damit das Unterfeld schreibgeschützt ist. Siehe auch Session-Parameter [AD](#) (Attribut-Definition).

field-representation-attribute

Zusätzliche Attribute können zugewiesen werden. Siehe auch Session-Parameter [AD](#) (Attribut-Definition).

Wert	Bedeutung
B	blinkend (*)
C	kursiv (*)
D	Standard normal
I	intensiviert
N	Nicht-Anzeige
U	unterstrichen
V	invers (*)

* Die mit einem Stern (*) markierten Feldanzeige-Attribute sind an entsprechende Hardware-Voraussetzungen gebunden und werden zur Laufzeit ignoriert, falls diese Voraussetzungen nicht gegeben sind.

escape-character2

Ein Identifizierungszeichen, das das Ende der Attribut-Definition markiert. Als Zeichen können Sie ein beliebiges Sonderzeichen (c) oder eine Hexadezimalzahl mit einem vorangestellten Apostroph ('xx) verwenden.

Sie können bis zu acht Attributsequenzen (Anfangsidentifizierungszeichen und Attribute) vor dem Zeichen, das das Ende der Attribut-Definitionen bestimmt, angeben.

Beispiele

Beispiel 1:

```
DY=<U>
```

Die Textkette:

```
THIS <i> UNDERLINED
```

wird ausgegeben als:

```
THIS i UNDERLINED
```

Beispiel 2:

```
DY=<BL|RE/GR>
```

Weist zu:

Blau zu <

Rot zu |

Grün zu /

> schaltet wieder zur ursprünglichen Farbe des Feldes zurück.

Beispiel 3:

```
DY=<P>;
```

Die Textkette:

```
Do not overwrite <this>
```

wird ausgegeben als:

```
Do not overwrite this
```

(wobei this geschützt ist)

58

DYNPARM - Control Use of Dynamic Parameters

This Natural profile parameter enables/disables the use of Natural dynamic parameters.

Possible settings	ON	Dynamic parameters supplied during Natural startup are processed.
	OFF	Dynamic parameters supplied during Natural startup are not processed. Anmerkung: If DYNPARM is set to OFF in the Natural default parameter file NATPARM, no alternative user-defined parameter files can be used when starting Natural.
Default setting	ON	
Dynamic specification	no	
Specification within session	no	

See *Dynamic Assignment of Parameter Values* for additional information on the use of dynamic parameters.

59

ECHO - Control Printing of Batch Input Data

This Natural profile parameter only applies in batch mode.

It is used to enable or disable the printing of input data provided to Natural during batch mode processing.

Possible settings	ON	Natural prints the input data provided during batch mode processing to the batch output file CMPRINT.
	OFF	Natural does <i>not</i> print input data provided during batch processing.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

It is also possible to suppress printing of a *single input line* by preceding it with a line containing the terminal command for record suppression %*.

Input read from CMSYNIN in command (NEXT) mode is echoed to the batch output file CMPRINT always.

60

EDTBPSIZE - Software AG Editor Buffer Pool Size

This Natural profile parameter is used to set the size of the Software AG Editor buffer pool.

Possible settings	0 - 4000	Size of the editor buffer pool in KB.
Default setting	400	
Dynamic specification	no	
Specification within session	no	

61

EDTLFILES - Number of Software AG Editor Logical Files

This Natural profile parameter is used to set the maximum number of the Software AG Editor sessions a user can open at a time.

Possible settings	10-999	Maximum number of Software AG Editor sessions.
Default setting	100	
Dynamic specification	no	
Specification within session	no	

62 EJ - Page Eject

Seitenvorschub

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, ob am Ende einer logischen Seite, beim Wechsel zwischen Programmeingabe und -ausgabe und nach der Meldung „Normal End“ ein Seitenvorschub erfolgen soll.

In einer Natural-Session kann der Profilparameter EJ durch den Session-Parameter EJ überschrieben werden. Die EJ-Einstellung kann wiederum durch ein EJECT-Statement überschrieben werden.

Mögliche Werte	ON	Ein Seitenvorschub wird ausgeführt.		
	OFF	Ein Seitenvorschub wird nicht ausgeführt. Bei Testläufen, bei denen Seitenumbrüche keine Rolle spielen, empfiehlt es sich, EJ=OFF zu setzen, um Papier zu sparen.		
Standard-Einstellung	ON			
Dynamische Spezifikation	ja			
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS	Parameter wird zur Laufzeit ausgewertet.
		Gültiges Kommando:	GLOBALS	Parameter kann mit dem GLOBALS-Systemkommando dynamisch angegeben werden.
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.		

Dieser Parameter gilt nur für den ersten auszugebenden Report (Report 0). Für weitere Reports ist das EJECT-Statement mit Report-Spezifikation (*rep*) zu verwenden.

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des Library-Profiles überschrieben werden.

63

EM - Edit Mask

▪ Syntax	142
▪ Leerzeichen in Editiermasken	143
▪ Standard-Editiermasken	144
▪ Editiermasken für numerische Felder	144
▪ Editiermasken für alphanumerische Felder	147
▪ Editiermasken für binäre Felder – Format B	149
▪ Hexadezimale Editiermasken	149
▪ Editiermasken für Datums- und Zeitfelder (Formate D und T)	152
▪ Editiermasken für logische Felder (Format L)	156

Editiermaske

Mit diesem Session-Parameter können Sie eine Editiermaske für ein Eingabe- und/oder Ausgabefeld definieren, welches in einem der Statements `DEFINE DATA`, `DISPLAY`, `INPUT`, `MOVE EDITED`, `PRINT`, `PROCESS PAGE/PROCESS PAGE UPDATE` oder `WRITE` verwendet wird.

Informationen zu Unicode-Editiermasken siehe Session-Parameter [EMU](#).

Der Session-Parameter `EM` kann auch bei Feldern mit Format `U` (Unicode) verwendet werden, siehe auch *Unicode and Code Page Support in the Natural Programming Language, Session Parameters*, `EMU`, `ICU`, `LCU`, `TCU` im Vergleich zu `EM`, `IC`, `LC`, `TC`.

Mögliche Werte		Siehe unten.
Standard-Einstellung	Keine	
Gültige Statements	<code>FORMAT</code>	Parameter kann dynamisch mit dem <code>FORMAT</code> -Statement angegeben werden.
	<code>DEFINE DATA</code> <code>DISPLAY</code> <code>INPUT</code> <code>PRINT</code> <code>PROCESS PAGE/PROCESS PAGE</code> <code>UPDATE</code> <code>WRITE</code>	Parameter kann auf Statement- und/oder Element-Ebene angegeben werden.
	<code>MOVE EDITED</code>	Parameter kann auf Element-Ebene angegeben werden.
Gültiges Kommando	Keines	

Die folgenden Themen werden nachfolgend erörtert:

Siehe auch *Codepage-Editiermasken – der EM-Parameter im Leitfaden zur Programmierung*.

Syntax

Bei Eingabefeldern muss ein Wert genau entsprechend der Editiermaske eingegeben werden. Um die Editiermaske für die Eingabe sichtbar zu machen, sollte ein Eingabefeld als modifizierbar (`AD=M`) definiert werden.

Für ein Datenbankfeld kann im DDM bereits eine Standard-Editiermaske definiert sein. Wenn Sie mit dem `EM`-Parameter eine Editiermaske für ein Datenbankfeld angeben, so gilt diese anstelle einer möglicherweise im DDM für das Feld definierten Standard-Editiermaske.

Geben Sie für ein Feld `EM=OFF` an, so wird für das Feld keine Editiermaske verwendet, auch keine möglicherweise im DDM definierte.

Auf Statement-Ebene eines DISPLAY-, FORMAT-, INPUT- oder WRITE-Statements kann keine Editiermaske definiert, sondern allenfalls EM=OFF gesetzt werden.

Wenn eine Editiermaske definiert ist, so überschreibt diese etwaige Einstellungen der Session-Parameter AL, NL and SG.

Diese Schreibweise gilt nur für die Zeichen 9, H, X und Z, mit denen bei numerischen (9, Z), hexadezimalen (H) und alphanumerischen (X) Editiermasken die signifikanten Stellen dargestellt werden. Der Unterschied zwischen 9 und Z ist im Abschnitt *Editiermasken für numerische Felder* weiter unten beschrieben.

Beispiele:

```
DISPLAY AA(EM=OFF) AB(EM=XX.XX)
WRITE SALARY (EM=ZZZ,ZZ9)
```

Lange Editiermasken können in Form einer Kurzschreibweise definiert werden. Die folgenden Beispiele zeigen die für numerische, hexadezimale und alphanumerische Editiermasken verwendbare Kurzschreibweise.

```
EM=9(4)-9(5) is equivalent to: EM=9999-99999
EM=H(10)     is equivalent to: EM=HHHHHHHHHHH
EM=X(6)..X(3) is equivalent to: EM=XXXXXX..XXX
```

Leerzeichen in Editiermasken

Leerzeichen innerhalb einer Editiermaske lassen sich durch das Zeichen auf Ihrer Tastatur darstellen, das in Hexadezimalcode H'20' (ASCII) bzw. H'5F' (EBCDIC) entspricht, d.h. das Zeichen ^ (oder ¬).

Standard-Editiermasken

Wenn Sie für ein Feld keine Editiermaske angeben, erhält das Feld entsprechend seinem Format eine Standard-Editiermaske:

Feldformat	Standard-Editiermaske
A	X
B	H
N, P, I	Z9
F	wissenschaftliche Darstellung
D	abhängig vom Standard-Datumsformat (wie mit dem Profilparameter DTFORM gesetzt)
T	HH:II:SS
L	Leerzeichen / X

Editiermasken für numerische Felder

Eine für Felder mit Format N, P, I oder F definierte Editiermaske muss mindestens eine 9 oder ein Z enthalten.

Enthält sie mehr „9er“ und „Zs“ als der Feldwert lang ist, wird die Anzahl der Ausgabestellen in der Editiermaske der Anzahl der für den Feldwert definierten Stellen angepasst.

Hat die Editiermaske weniger signifikante Stellen als der Feldwert, wird der Feldwert um die entsprechende Anzahl der Stellen vor bzw. nach dem Dezimaltrennzeichen (Punkt oder Komma) verkürzt ausgegeben.

Die folgenden Themen werden nachfolgend erörtert:

- [Zeichen zur Definition numerischer Editiermasken](#)
- [Vorzeichen](#)
- [Führende Literale](#)
- [Literale Einschubzeichen und nachgestellte Zeichen](#)
- [Nachfolgende Vorzeichen](#)

- Beispiele für numerische Editiermasken

Zeichen zur Definition numerischer Editiermasken

Zeichen	Funktion
9	Auszugebende Stelle (eine Stelle des Feldwertes).
.	<p>(period) Ein Punkt, als erstes Zeichen verwendet, stellt ein Dezimaltrennzeichen (Komma oder Punkt) dar und ist signifikant. Nachfolgende Punkte werden als Literale behandelt.</p> <p>Anmerkung: An dieser Stelle stellt der Punkt das zurzeit als Dezimaltrennzeichen festgelegte Zeichen dar. Falls mit dem Session- oder Profilparameter DC ein anderes Zeichen gewählt wird (zum Beispiel ein Komma), dann ist anstelle des Punktes dieses Zeichen zu verwenden.</p>
Z	Nullunterdrückung bei vorangestellten Nullen. Dies gilt standardmäßig für numerische Felder. Zur gleitenden Nullunterdrückung kann das Z mehrmals angegeben werden. Rechts vom Dezimaltrennzeichen darf kein Z stehen. Ein Nullwert kann unter Einbeziehung aller Zs in der Editiermaske als lauter Leerzeichen ausgegeben werden (siehe auch Session-Parameter ZP).

Vor den Neunern oder Zs können eins oder mehrere andere Zeichen stehen.

Vorzeichen

Wenn das erste Zeichen vor den Neunern oder Zs ein +, -, S oder N ist, kann ein Vorzeichen angezeigt werden:

Zeichen	Funktion
+	Ein gleitendes Vorzeichen, das vor/nach der Zahl ausgegeben werden soll. Das Zeichen wird je nach Wert der Zahl als + oder - generiert.
-	Ein gleitendes Minus-Vorzeichen, das vor/nach der Zahl ausgegeben werden soll, wenn die Zahl negativ ist.
S	Ein Vorzeichen, das vor dem Feld ausgegeben werden soll. Das Vorzeichen wird je nach Wert der Zahl als + oder - generiert.
N	Ein Minus-Vorzeichen, das vor dem Feld ausgegeben werden soll, wenn der Feldwert negativ ist.

Führende Literale

Eine beliebige Anzahl von führenden Literalen kann vor der ersten anzeigbaren Stelle erscheinen (wie durch Z oder 9 angezeigt). Diese müssen auf ein Vorzeichen folgen. Wenn kein Vorzeichen vorhanden ist, und das erste führende Literal ist +, -, S oder N, muss es in Apostrophen stehen. Wenn ein führendes Literal H, X, Z oder 9 ist, muss es in Apostrophen stehen.

Das zuerst angegebene führende Literal erscheint nur in der Ausgabe, wenn der Wert führende Nullen enthält und die Editiermaske mit Z definiert ist (führende Nullwertunterdrückung). Dieses

Zeichen wird dann als Füllzeichen benutzt, das anstatt eines Leerzeichens für führende Nullen angezeigt wird. Nachfolgende führende Literale werden so angezeigt, wie sie eingegeben werden.

Literale Einschubzeichen und nachgestellte Zeichen

Es können auch literale Einschubzeichen und nachgestellte Zeichen benutzt werden. Das Symbol (^) kann zur Darstellung eines vorangestellten, eingefügten oder nachgestellten Leerzeichens verwendet werden. Durch Setzen von signifikanten Zeichen (9, H, Z, X) in Apostrophen ist es möglich, vorangestellte, eingefügte und nachgestellte Zeichenketten zu haben.

Nicht signifikante Editiermasken-Zeichen müssen nicht in Apostrophen stehen. Innerhalb derselben Editiermasken-Notation ist es möglich, Gruppen von vorangestellten Zeichenketten, Einschubzeichen und/oder nachfolgenden Zeichen zu haben, von denen einige in Apostrophen stehen und andere nicht.

Nachfolgende Vorzeichen

Ein nachfolgendes Vorzeichen wird durch ein + oder - als letztes Zeichen der Editiermaske angegeben. Ein + bewirkt, dass das Vorzeichen je nach Feldwert entweder als + oder - ausgegeben wird; ein - bewirkt, dass bei einem positiven Feldwert ein Leerzeichen und bei einem negativen Feldwert ein - ausgegeben wird. Ist für eine Editiermaske ein vorangestelltes und ein nachgestelltes Vorzeichen definiert, werden beide ausgegeben.

Beispiele für numerische Editiermasken

Die folgende Tabelle zeigt in der oberen Zeile die Werte numerischer Felder (Format N), wie sie ohne Editiermaske ausgegeben würden, und darunter die unter Verwendung der verschiedenen Editiermasken ausgegebene Form:

Wert	0000.03 (N4.2)	-0054 (N4)	+0087 (N4)	0962 (N4)	1830 (N4)
Editiermaske					
EM=9.9	0.0	4.	7.	2.	0.
EM=99	00	54	87	62	30
EM=S99	+00	-54	+87	+62	+30
EM=+Z9	+0	-54	+87	+62	+30
EM=-9.99	0.03	-4.	7.	2.	0.
EM=N9	0	-4	7	2	0
EM=*9.99	0.03	4.	7.	2.	0.
EM=Z99	00	54	87	962	830
EM=*EURZZ9.9	EUR**0.0	EUR*54.	EUR*87.	EUR962.	EUR830.
EM=999+	000+	054-	087+	962+	830+
EM=999-	000	054-	087	962	830

IC=\$ EM=ZZZ.99	\$.03	\$54.	\$87.	\$962.	\$830.
EM=H(6)					
- ASCII:	303030303033	30303574	30303837	30393632	31383330
- EBCDIC:	F0F0F0F0F0F3	F0F0F5D4	F0F0F8F7	F0F9F6F2	F1F8F3F0

Durch Kombination von Editiermasken mit den Parametern IC und TC ist es bei einem DISPLAY-Statement möglich, negative Zahlen in verschiedenen Formen auszugeben.

Editiermasken für alphanumerische Felder

Für mit Format A definierte Felder kann eine alphanumerische Editiermaske definiert werden; sie muss mindestens ein X enthalten; jedes X steht für ein auszugebendes Zeichen. Ein H als erstes Zeichen kennzeichnet eine **hexadezimale Editiermaske**. Ein Leerzeichen wird durch ein Circumflex (^) dargestellt.

Alle anderen Zeichen — außer Klammern — können als vorangestellte, eingeschobene oder nachgestellte Zeichen verwendet werden, wobei diese Zeichen wahlweise durch Apostrophe eingegrenzt werden können oder nicht. Sollen die Zeichen %, eine abschließende runde Klammer () oder ein Anführungszeichen (") als nichtsignifikante Einschubzeichen verwendet werden, müssen sie in Apostrophen angegeben werden.

Werden dem ersten signifikanten X Zeichen vorangestellt, wird das erste dieser Zeichen nicht ausgegeben, sondern als Füllzeichen benutzt, und es tritt an die Stelle aller führenden Leerzeichen im alphanumerischen Ausgabefeld.

Beispiel:

```
DEFINE DATA LOCAL
1 #X (A4) INIT <' 34'>
END-DEFINE
WRITE #X (EM=*A:X:)
    6X #X (EM=*A:XX:)
    6X #X (EM=*A:XXX:)
    6X #X (EM=*A:XXXX:)
    6X #X (EM=1234XXXX5678)
END
```

Ausgabe:

```
A:*:      A:**:      A:**3:      A:**34:      23411345678
```

Zeichen, die unmittelbar auf das letzte signifikante X folgen, werden ausgegeben.

Ist die Editiermaske kürzer als das Feld, wird die Anzahl der ausgegebenen Stellen auf die Länge der Editiermaske gekürzt.

Ist die Editiermaske länger als das Feld, wird die Anzahl der ausgegebenen Stellen bei der ersten überstehenden Stelle abgeschnitten.

Beispiel:

```
DEFINE DATA LOCAL
  1 #TEXT (A4) INIT <'BLUE'>
END-DEFINE
WRITE #TEXT (EM=X-X-X)      displays as 'B-L-U'      .. only three bytes of field
                             displayed
WRITE #TEXT (EM=X-X-X-X-X) displays as 'B-L-U-E-' .. mask definition was
                             truncated to (EM=X-X-X-X-)
```

Beispiel für alphanumerische Editiermasken

Das folgende Programm definiert Editiermasken für ein Feld mit Format/Länge A4, das den Wert BLUE enthält:

```
** Example 'EMMASK1': Edit mask
*****
DEFINE DATA LOCAL
  1 #TEXT (A4)
END-DEFINE
*
ASSIGN #TEXT = 'BLUE'
WRITE NOTITLE 'MASK 1:' 5X #TEXT (EM=X.X.X.X)
              / 'MASK 2:' 5X #TEXT (EM=X^X^X^X)
              / 'MASK 3:' 5X #TEXT (EM=X--X--X)
              / 'MASK 4:' 5X #TEXT (EM=X-X-X-X-X-X)
              / 'MASK 5:' 5X #TEXT (EM=X' 'X' 'X' 'X)
              / 'MASK 6:' 5X #TEXT (EM=XX...XXX)
              / 'MASK 7:' 5X #TEXT (EM=1234XXXX)
END
```

Ausgabe des Programms EMMASK1:

```

MASK 1:   B.L.U.E
MASK 2:   B L U E
MASK 3:   B - L - - U
MASK 4:   B - L - U - E -
MASK 5:   B L U E
MASK 6:   BL...UE
MASK 7:   234BLUE

```

Editiermasken für binäre Felder – Format B

Editiermasken für binäre Felder können mittels der Notation X oder H definiert werden. Für binäre Felder wird die Notation X unterstützt, als ob H anstelle von X angegeben worden wäre.

Hexadezimale Editiermasken

Wird als erstes Zeichen einer Editiermaske ein H angegeben, so wird der Wert eines alphanumerischen oder numerischen Feldes in hexadezimaler Form ausgegeben. Jedes H steht für zwei Hexadezimalstellen, die jeweils einem numerischen/alphanumerischen Byte im Source-Feld entsprechen.

Alle anderen Zeichen können als Einschubzeichen oder nachgestellte Zeichen verwendet werden. Ist die Editiermaske kürzer als das Feld, wird der Feldwert entsprechend verkürzt ausgegeben. Ist das Feld kürzer als die Editiermaske, wird die Editiermaske der Feldlänge entsprechend verkürzt ausgegeben.

Einschubzeichen oder nachgestellte Zeichen können wahlweise durch Apostrophe eingegrenzt werden.

Alle mit einer hexadezimalen Editiermaske angezeigten Felder werden als alphanumerische Felder behandelt. Ist die Editiermaske kürzer als das Feld, werden daher alle numerischen oder alphanumerischen Stellen von links nach rechts ohne Berücksichtigung von Dezimalstellen ausgegeben.

Wenn eine hexadezimale Editiermaske als eine Eingabeeditiermaske benutzt wird, werden alle Zeichen 0-9, a-f, , A-F, Leerzeichen und die hexadezimale Null als eine hexadezimale Ziffer akzeptiert.



Anmerkung: Leerzeichen und die hexadezimale Null werden als 0 und ein Kleinbuchstabe (a-f) als Großbuchstabe angesehen.

Editiermasken-Beispiele für hexadezimale Felder

Die folgenden Tabellen zeigen hexadezimale Editiermasken mit Ergebnissen aus den ursprünglichen Feldern und über jeder Spalte angezeigten Werten. Alle numerischen Werte (-10, +10, 01), für die Editiermasken gelten, stammen aus im Format N2 definierten Feldern. Der alphanumerische Wert AB stammt aus einem mit Format/Länge A2 definierten Feld.

ASCII:

Wert =>	AB	-10	+10	01
EM=HH	4142	3170	3130	3031
EM=H^H	41 42	31 70	31 30	30 31
EM=HH^H	4142	3170	3130	3031
EM=H-H	41-42	31-70	31-30	30-31
EM=H	41	31	31	30

EBCDIC:

Wert =>	AB	-10	+10	01
EM=HH	C1C2	F1D0	F1F0	F0F1
EM=H:H	C1 C2	F1 D0	F1 F0	F0 F1
EM=HH:H	C1C2	F1D0	F1F0	F0F1
EM=H-H	C1-C2	F1-D0	F1-F0	F0-F1

EM=H	C1	F1	F1	F0
------	----	----	----	----

Beispielprogramm mit hexadezimalen Editiermasken:

```

** Example 'EMMASK2': Edit mask
*****
DEFINE DATA LOCAL
1 #TEXT1 (A2)
1 #TEXT2 (N2)
END-DEFINE
*
ASSIGN #TEXT1 = 'AB'
ASSIGN #TEXT2 = 10
*
WRITE NOTITLE
    'MASK (EM=HH) :' 18T #TEXT1 (EM=HH)      30T #TEXT2 (EM=HH)
  / 'MASK (EM=H^H) :' 18T #TEXT1 (EM=H^H)    30T #TEXT2 (EM=H^H)
  / 'MASK (EM=HH^H) :' 18T #TEXT1 (EM=HH^H)  30T #TEXT2 (EM=HH^H)
  / 'MASK (EM=H-H) :' 18T #TEXT1 (EM=H-H)    30T #TEXT2 (EM=H-H)
  / 'MASK (EM=H)   :' 18T #TEXT1 (EM=H)      30T #TEXT2 (EM=H)
END

```

Ausgabe des Programms EMMASK2 (ASCII):

```

MASK (EM=HH) : 4142      3130
MASK (EM=H^H) : 41 42    31 30
MASK (EM=HH^H) : 4142    3130
MASK (EM=H-H) : 41-42    31-30
MASK (EM=H)   : 41       31

```

Ausgabe des Programms EMMASK2 (EBCDIC):

```

MASK (EM=HH) : C1C2      F1F0
MASK (EM=H^H) : C1 C2    F1 F0
MASK (EM=HH^H) : C1C2    F1F0
MASK (EM=H-H) : C1-C2    F1-F0
MASK (EM=H)   : C1       F1

```

Editiermasken für Datums- und Zeitfelder (Formate D und T)

Zur Definition von Editiermasken für Felder, die mit dem Format D (Datumsfeld) oder T (= Time; Zeitfeld) definiert sind, können die in den folgenden Abschnitten beschriebenen Zeichen verwendet werden:

- Datums- und Zeitfelder (Formate D und T)
- Syntaktische Einschränkungen für Datumszeichen
- Hinweise für Eingabe-Editiermaske
- Hinweise für Wochenanzeige (WW oder ZW) in Ausgabe-Editiermaske
- Nur für Zeitfelder (Format T):
- Beispiele für Datums- und Zeit-Editiermasken

Datums- und Zeitfelder (Formate D und T)

Zeichen	Bedeutung
DD	Tag (Day).
ZD	Tag mit Nullwertunterdrückung.
MM	Monat.
ZM	Monat mit Nullwertunterdrückung.
YYYY	Jahr (Year), vierstellig (siehe Abschnitt Hinweise für Eingabe-Editiermaske).
YY	Jahr, zweistellig (siehe Abschnitt Hinweise für Eingabe-Editiermaske).
Y	Jahr, einstellig. Darf nicht für Eingabefelder verwendet werden.
WW	Woche (siehe die Abschnitte Hinweise für Eingabe-Editiermaske und Hinweise für Wochenanzeige in Ausgabe-Editiermaske).
ZW	Woche mit Nullwertunterdrückung (siehe die Abschnitte Hinweise für Eingabe-Editiermaske und Hinweise für Wochenanzeige in Ausgabe-Editiermaske).
JJJ	Julianischer Tag.
ZZJ	Julianischer Tag mit Nullwertunterdrückung.
NN . . . oder N(<i>n</i>)	Name des Wochentages (sprachabhängig). Die Maximallänge wird durch die Anzahl der Ns bzw. durch <i>n</i> bestimmt. Ist der Name länger als die Maximallänge, wird er abgeschnitten; ist er kürzer, wird seine tatsächliche Länge genommen.
0	<p>Nummer des Wochentags.</p> <p>Ob Montag oder Sonntag als erster Wochentag genommen wird, hängt vom Profilparameter DTFORM ab).</p> <p>Ist DTFORM=U, dann ist Sonntag = 1, Montag = 2 usw.).</p> <p>Ist DTFORM=<i>sonstige</i>, dann ist Montag = 1, Dienstag = 2 usw.).</p>

Zeichen	Bedeutung
LL . . . oder L(<i>n</i>)	Name des Monats (sprachabhängig). Die Maximallänge wird durch die Anzahl der Ls bzw. durch <i>n</i> bestimmt. Ist der Name länger als die Maximallänge, wird er abgeschnitten; ist er kürzer, wird seine tatsächliche Länge genommen.
R	Jahr in römischen Ziffern (maximal 13 Stellen). Darf für Eingabefelder nicht benutzt werden.

Syntaktische Einschränkungen für Datumszeichen

Für *Eingabe-/Ausgabe*-Editiermasken dürfen Sie die folgenden Zeichen *nicht* verwenden:

Text			Zeichen		
Monat	mit	Monatsnamen	MM oder ZM	mit	LL oder L(<i>n</i>)
Tagesname	mit	Wochentag	NN oder N(<i>n</i>)	mit	0

Für *Eingabe*-Editiermasken dürfen Sie die folgenden Zeichen *nicht* verwenden:

Text			Zeichen		
1-stelliges Jahr	und auch nicht	ein Jahr in römischen Ziffern	Y	und auch nicht	R
Day	ohne	Monat oder Monatsnamen	DD oder ZD	ohne	MM oder ZM oder LL oder L(<i>n</i>)
Woche	ohne	Jahr	WW oder ZW	ohne	YYYY oder YY
Monat	ohne	Jahr	MM oder ZM	ohne	YYYY oder YY
Julian. Tag	ohne	Jahr	JJJ oder ZZJ	ohne	YYYY oder YY
Tagesname	ohne	Woche	NN oder N(<i>n</i>)	ohne	WW oder ZW
Wochentag	ohne	Woche	0	ohne	WW oder ZW
Julian. Tag	mit	Monat	JJJ oder ZZJ	mit	MM oder ZM
Julian. Tag	mit	Woche	JJJ oder ZZJ	mit	WW oder ZW
Monat	mit	Woche	MM oder ZM	mit	WW oder ZW

Hinweise für Eingabe-Editiermaske

Die gültigen Jahreswerte (YYYY) sind 1582 - 2699.

Wird in einer Eingabe-Editiermaske nur das Jahr (YY oder YYYY) angegeben, aber nicht Monat und Tag, werden die Werte für Monat und Tag jeweils auf 01 gesetzt. Werden in einer Eingabe-Editiermaske nur Jahr (YY oder YYYY) und Monat (MM) angegeben, aber kein Tag, wird der Wert für Tag auf 01 gesetzt.

Bei einer 2-stelligen Jahresangabe (YY) ist das zum Füllen der Jahresdarstellung benutzte Jahrhundert standardmäßig das aktuelle Jahrhundert. Dies gilt aber nicht, wenn ein Sliding Window oder Fixed Window definiert ist. Weitere Einzelheiten entnehmen Sie dem Profilparameter [YSLW](#).

Wird eine Woche (WW oder ZW) aber kein Wochentag (0) oder Tagesname (NN . . .) angegeben, geht man vom ersten Wochentag aus.

Hinweise für Wochenanzeige (WW oder ZW) in Ausgabe-Editiermaske

Wenn `DTFORM=U` (USA-Format) gesetzt ist, beginnt die Woche am Sonntag, wohingegen bei allen anderen `DTFORM`-Einstellungen der erste Wochentag der Montag ist. Ob eine Woche die 52./53. Woche des alten oder die 1. Woche des neuen Jahres ist, hängt davon ab, welches Jahr mehr Wochentage enthält. Mit anderen Worten, wenn der Donnerstag (Mittwoch bei `DTFORM=U`) dieser Woche im alten Jahr liegt, gehört die Woche zum alten Jahr; liegt er im neuen Jahr, gehört die Woche zum neuen Jahr.

Wenn sich eine Darstellung der Woche (WW oder ZW) und des Jahres (YYYY oder YY oder Y) auf derselben Editiermaske befinden, entspricht die Anzeige des Jahres stets der Woche, ungeachtet des Jahres in dem zugrundeliegenden Datumsfeld.

Beispiel:

```
DEFINE DATA LOCAL
1 D (D)
END-DEFINE
MOVE EDITED '31-12-2003' TO D(EM=DD-MM-YYYY)
DISPLAY D(EM=DD-MM-YYYY_N(10)) D(EM=DD-MM-YYYY/WW)
```

Obwohl das zugrundeliegende Datum der 31. Dezember 2003 ist, wenn die Woche WW in der Editiermaske enthalten ist, wird es wie folgt angezeigt:

```

          D          D
-----
31-12-2003_Wednesday 31-12-2004/01

```

Nur für Zeitfelder (Format T):

Zeichen	Bedeutung
T	Zehntelsekunden (Tenths of a second).
SS	Sekunden.
ZS	Sekunden mit Nullwertunterdrückung.
II	Minuten.
ZI	Minuten mit Nullwertunterdrückung.
HH	Stunden.
ZH	Stunden mit Nullwertunterdrückung.
AP	AM/PM-Element (englische Zeitangabe: AM = vormittags, PM = nachmittags).

Beispiele für Datums- und Zeit-Editiermasken

```

** Example 'EMDATI': Edit mask for date and time variables
*****
*
WRITE NOTITLE
  'DATE INTERNAL :' *DATX (DF=L) /
  '                :' *DATX (EM=N(9)' 'ZW.'WEEK 'YYYY) /
  '                :' *DATX (EM=ZZJ'.DAY 'YYYY) /
  '    ROMAN      :' *DATX (EM=R) /
  '    AMERICAN  :' *DATX (EM=MM/DD/YYYY)      12X 'OR ' *DAT4U /
  '    JULIAN    :' *DATX (EM=YYYYJJJ)        15X 'OR ' *DAT4J /
  '    GREGORIAN:' *DATX (EM=ZD.' 'L(10)''YYYY) 5X 'OR ' *DATG ///
*
  'TIME INTERNAL :' *TIMX                      14X 'OR ' *TIME /
  '                :' *TIMX (EM=HH.II.SS.T) /
  '                :' *TIMX (EM=HH.II.SS' 'AP) /
  '                :' *TIMX (EM=HH)
END

```

Ausgabe des Programms EMDATI:

```

DATE INTERNAL : 2005-01-12
               : Wednesday 2.WEEK 2005
               : 12.DAY 2005
  ROMAN       : MMV
  AMERICAN    : 01/12/2005          OR 01/12/2005
  JULIAN      : 2005012             OR 2005012
  GREGORIAN   : 12.January2005     OR 12January 2005

TIME INTERNAL : 16:04:14           OR 16:04:14.8
               : 16.04.14.8
               : 04.04.14 PM
               : 16

```

Editiermasken für logische Felder (Format L)

Editiermasken für Felder, die das Format L haben (logische Felder), können wie folgt definiert werden:

```
(EM=[false-string]/true-string)
```

wobei *false-string* für die für „falsch“ auszugebende Zeichenkette und *true-string* für die für „wahr“ auszugebende Zeichenkette steht.

Beispiel für Editiermasken für logisches Feld

```

** Example 'EMLOGV': Edit mask for logical variables
*****
DEFINE DATA LOCAL
1 #SWITCH (L) INIT <true>
1 #INDEX (I1)
END-DEFINE
*
FOR #INDEX 1 5
  WRITE NOTITLE #SWITCH (EM=FALSE/TRUE) 5X 'INDEX =' #INDEX
  WRITE NOTITLE #SWITCH (EM=OFF/ON) 7X 'INDEX =' #INDEX
  IF #SWITCH
    MOVE FALSE TO #SWITCH
  ELSE
    MOVE TRUE TO #SWITCH
  END-IF
/*
  SKIP 1
END-FOR
END

```

Ausgabe des Programms EMLOGV:

```
TRUE      INDEX = 1
ON        INDEX = 1

FALSE     INDEX = 2
OFF       INDEX = 2

TRUE      INDEX = 3
ON        INDEX = 3

FALSE     INDEX = 4
OFF       INDEX = 4

TRUE      INDEX = 5
ON        INDEX = 5
```


64

EMFM - Edit Mask Free Mode

This Natural profile parameter is used to activate/deactivate the Edit Mask Free mode at session startup. This mode allows you to omit literals during input into a field with a numeric edit mask.

Possible settings	ON	Edit Mask Free Mode is activated.
	OFF	Edit Mask Free Mode is deactivated.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	Within a running Natural session, you may override this setting with the terminal control command %FM+ or %FM-.

For additional information, see *Numeric Edit Mask Free Mode* in the INPUT statement description in the *Statements* documentation.

65

EMU - Unicode Edit Mask

Unicode-Editiermaske

Mit diesem Session-Parameter können Sie für ein Eingabe- und/oder Ausgabefeld, das in einem DEFINE DATA-, DISPLAY-, INPUT-, MOVE EDITED-, PRINT-, PROCESS PAGE- oder WRITE-Statement verwendet wird, eine Unicode-Editiermaske definieren.

Mögliche Werte		Siehe unten.
Standard-Einstellung	Keine	
Gültige Statements		
	DEFINE DATA DISPLAY INPUT PRINT WRITE	Parameter kann auf Element-Ebene angegeben werden.
	MOVE EDITED	Parameter kann auf Element-Ebene angegeben werden.
Gültiges Kommando	none	

Die Syntax des Session-Parameters EMU ist identisch mit der Syntax des Session-Parameters EM (siehe EM [-Syntax](#)).

Siehe auch *Unicode-Editiermasken* — EMU-Parameter im *Leitfaden zur Programmierung*.

Editiermasken, die mit EMU definiert sind, werden im Unicode-Format gehalten, so dass der Inhalt unabhängig von der installierten System-Codepage ist.

Weitere Informationen und ein Beispiel finden Sie in *Unicode and Code Page Support in the Natural Programming Language, Session Parameters*, Abschnitt EMU, ICU, LCU, TCU versus EM, IC, LC, TC.

66

ENDIAN - Endian Mode for Compiled Objects

Endian-Modus für kompilierte Objekte

Mit diesem Natural-Profil- und Session-Parameter können Sie die Architektur angeben, für die der Compiler GPs generieren soll. Siehe auch *Portierbare generierte Natural-Programme im Leitfaden zur Programmierung*.

In einer Natural-Session kann der Profilparameter ENDIAN durch den Session-Parameter ENDIAN überschrieben werden.

Mögliche Werte	DEFAULT	Endian-Modus wird von der aktuell benutzten Architektur abgeleitet.	
	LITTLE	Der Compiler generiert GP für „Little Endian“-Modus.	
	BIG	Der Compiler generiert GP für „Big Endian“-Modus.	
Standard-Einstellung	DEFAULT		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	keine
		Gültiges Kommando:	GLOBALS

67

ENDMSG - Display Session-End Message

This Natural profile parameter is used to suppress the display the default message NAT9995 that is displayed at the end of the Natural session to indicate that the Natural session has been ended normally.

Possible settings	ON	Message NAT9995, NAT9978 or NAT9987 is written to the batch output file CMPRINT at the end of the session.
	OFF	Message NAT9995, NAT9978 or NAT9987 does not appear in CMPRINT .
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

- If Natural terminates with a startup error, then Natural message NAT9987 will be used instead of NAT9995.
- If Natural terminates with a runtime error, then Natural message NAT9978 will be used instead of NAT9995.

68

ES - Empty Line Suppression

Leerzeilenunterdrückung

Mit diesem Session-Parameter können Sie die Ausgabe der von einem DISPLAY- oder WRITE-Statement erzeugten Leerzeilen unterdrücken.

Mögliche Werte	ON	Eine Zeile aus einem DISPLAY- oder WRITE-Statement, die alle Leerwerte enthält, wird nicht ausgegeben. Diese Einstellung ist besonders nützlich, wenn Arrays angezeigt werden (z.B. multiple Felder oder Felder in Periodengruppen), um nicht überflüssig viele Leerwerte auszugeben.
	OFF	Leerzeilenunterdrückung ist ausgeschaltet.
Standard-Einstellung	OFF	
Spezifikation in Session	ja	
Gültige Statements	DISPLAY FORMAT WRITE	
Gültiges Kommando	Keines	

Um die Leerwertunterdrückung auch für numerische Werte zu erhalten, muss für die betreffenden Felder neben `ES=ON` auch der Parameter `ZP=OFF` gesetzt werden, was bewirkt, dass Nullwerte in Leerwerte umgesetzt und dann ebenfalls nicht ausgegeben werden. Siehe auch Session-Parameter [IS](#) und [ZP](#).

Beispiel:

```
DISPLAY (ES=ON) NAME CITY
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

69

ESCAPE - Ignore Terminal Commands %% and %.

This Natural profile parameter can be used to disable the terminal commands %% and %..

Possible settings	ON	Enables the use of terminal commands %% and %..
	OFF	The terminal commands %% and %.. will be ignored; that is, it will not be possible to leave the currently active Natural program or the Natural session respectively by entering %% or %..
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

70

ESXDB - Database ID Used for Entire System Server DDMs

This Natural profile parameter applies to the Entire System Server Interface.

It specifies the database ID used for Entire System Server's DDMs.

Possible settings	1 - 254	Database ID. To activate this parameter, a database ID in the range of 1 to 254 must be specified. Anmerkung: Entire System Server's DDMs are cataloged with DBID=148. If you are using an Adabas database with this DBID, specify a different number for ESXDB. For information on how to do this, please refer to <i>Setting up the Entire System Server Interface</i> in the <i>Operations</i> documentation.
	0	With ESXDB=0, the Entire System Server Interface is not active.
Default setting	0	
Dynamic specification	no	
Specification within session	no	

71 ET - Execution of END/BACKOUT TRANSACTION

Statements

This Natural profile parameter specifies for which databases END TRANSACTION and BACKOUT TRANSACTION statements are to be executed.

Possible settings	ON	END TRANSACTION and BACKOUT TRANSACTION statements are executed for all databases which have been referenced since the beginning of the Natural session or since the last execution of an END TRANSACTION and BACKOUT TRANSACTION statement.
	OFF	END TRANSACTION and BACKOUT TRANSACTION statements are executed only for the databases affected by the transaction (and - if applicable - for the database to which transaction data are written).
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	



Anmerkung: Any updates to a database which are not executed under the control of Natural (that is, by native invocation of the database link routines) do not affect the Natural transaction logic.

72

ETA - Error Transaction Program

This Natural profile parameter provides the name of the program which receives control if an error condition is detected during Natural program execution.

Possible settings	1 to 8 characters	Program name for error transaction.
	blank	With ETA=' ' (blank), no error transaction program is called.
Default setting	blank	
Dynamic specification	yes	
Specification within session	yes	
Application Programming Interface	USR1041N	USR1041N is a sample error transaction program delivered in source form. See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

The setting of this parameter can be modified by a user program by way of assignment to the system variable *ERROR-TA or, if Natural Security is installed, within the Natural Security library profile; see *Components of a Library Profile* in the *Natural Security* documentation.

For further information, see *Using an Error Transaction Program* in the *Programming Guide*.

73

ETDB - Database for Transaction Data

This Natural profile parameter specifies the database in which transaction data, as supplied with an `END TRANSACTION` statement is to be stored.

Possible settings	1 - 65535, except 255	Database ID. Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
	0	The transaction data is written to the database specified with the profile parameter UDB .
Default setting	0	
Dynamic specification	yes	
Specification within session	no	

Other transaction processing related parameters: [ET](#) | [ETEOP](#)

74

ETEOP - Issue END TRANSACTION at End of Program

This Natural profile parameter determines whether or not an implicit `END TRANSACTION` statement is to be issued at the end of a Natural program (that is, before `NEXT` mode is reached).

Possible settings	ON	Natural will issue an implicit <code>END TRANSACTION</code> statement at the end of a Natural program.
	OFF	Natural will not issue any implicit <code>END TRANSACTION</code> statement at the end of a Natural program.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

75

ETID - Adabas User Identification

This Natural profile parameter is used as an identifier for Adabas-related information; for example, for identification of data stored as a result of an `END TRANSACTION` statement.

Possible settings	1 to 8 characters	The setting is used as the user ID setting in an Adabas open call. The Adabas user ID has its own syntax. Please consult your Adabas Command Reference Documentation if you want to use special characters for the setting of ETID.
	OFF	The ETID is set to ' ' (blanks), but Natural Security is allowed to set the ETID.
	' ' (blank)	The ETID is set to ' ' (blanks). This value is passed to Adabas on an open call without being modified by Natural Security.
	\$\$	The ETID is replaced by the process ID.
Default setting	' ' (blank)	
Dynamic specification	yes	
Specification within session	no	

If the ETID is not specified neither in NATPARM nor dynamically, Natural uses the setting of *INIT-USER to fill the ETID.

\$\$ will be evaluated during the start-up of Natural.

The ETID might also be used by Natural for work-file name generation. This might cause problems when running multiple instances on one machine or again, if special characters are used for the setting of ETID. Please refer to the Natural parameter [TMPSORTUNIQ](#) for a solution.

76 ETPDB - Databases Containing Entire Transaction

Propagator Master Files

This Natural profile parameter only applies if Software AG's Entire Transaction Propagator is installed.

ETPDB specifies the databases that contain Entire Transaction Propagator master files.

Possible settings	1 - 255	Database IDs; for details, see the Entire Transaction Propagator documentation. The entries in the ETPDB field must be separated by blanks.
	0	Entire Transaction Propagator is not to be used.
	' ' (blank)	
Default setting	' ' (blank)	
Dynamic specification	no	
Specification within session	no	

77

ETPSIZE - Size of Entire Transaction Propagator Buffer

This Natural profile parameter only applies if Entire Transaction Propagator is installed. It determines the size of the Entire Transaction Propagator buffer.

Possible settings	10 - 63	Size of the Entire Transaction Propagator buffer in KB. If Entire Transaction Propagator is to be used, an appropriate value has to be specified for this parameter; see the Entire Transaction Propagator documentation. If the requested space is not available, the Entire Transaction Propagator cannot be used.
	0	Entire Transaction Propagator is not to be used.
Default setting	0	
Dynamic specification	no	
Specification within session	no	

78

FC - Filler Character for INPUT Statement

This Natural profile parameter is used to specify the default filler character to be used for fields displayed by an INPUT statement.

Possible settings	any character	Default filler character. It is used to pre-fill fields non-protected input fields (field attribute specification AD=A) when fields are written to a terminal by an INPUT statement. For modifiable input fields (field attribute specification AD=M), it is used to fill the rest of the field.
Default setting	blank	
Dynamic specification	yes	
Specification within session	no	

79

FC - Filler Character for DISPLAY Statement

Füllzeichen für DISPLAY-Statement

Mit diesem Session-Parameter bestimmen Sie das Füllzeichen, mit dem bei Spaltenüberschriften, die über ein DISPLAY-Statement erzeugt werden, der Platz rechts und links der Überschrift aufgefüllt wird.

Im Gegensatz zum Parameter `GC`, der für Überschriften über einen Reihe von Spalten hinweg gilt, gilt der Parameter `FC` für einzelne Spalten.

Mögliche Werte	ein beliebiges Zeichen	Füllzeichen für einzelne Überschriften. FC gilt nur, wenn die Spaltenbreite von der Feldlänge und nicht von der Länge der Überschrift bestimmt wird. Siehe auch Session-Parameter HW); sonst wird der FC-Parameter ignoriert.	
Standard-Einstellung	leer		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY (FC=*)
```

80

FCDP - Filler Character for Dynamically Protected Input

Fields

Füllzeichen für dynamisch geschützte Felder

Mit diesem Natural Profil- und Session-Parameter können Sie die Anzeige von Füllzeichen für Eingabefelder unterdrücken, die dynamisch schreibgeschützt wurden (d.h. denen das Attribut `AD=P` mittels einer Kontrollvariablen zugewiesen wurde).

Je nach dem Wert dieses Parameters werden dynamisch geschützte Eingabefelder entweder mit Leerzeichen oder mit den definierten Füllzeichen angezeigt.

In einer Natural-Session kann der Profilparameter `FCDP` durch den Session-Parameter `FCDP` überschrieben werden.

Mögliche Werte	ON	Dynamisch geschützte Eingabefelder werden mit Füllzeichen gefüllt angezeigt. Dies kann bei Benutzern den Eindruck erwecken, sie könnten in diese Felder etwas eingeben.	
	OFF	Dynamisch geschützte Eingabefelder werden mit Leerzeichen gefüllt angezeigt.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Beispiel:

```

DEFINE DATA LOCAL
1 #FIELD1 (A5)
1 #FIELD2 (A5)
1 #CVAR1 (C) INIT <(AD=P)>
1 #CVAR2 (C)
END-DEFINE
*
INPUT #FIELD1 (AD=Y'_' CV=#CVAR1) /* field is protected
      #FIELD2 (AD=Y'_' CV=#CVAR2) /* field is not protected
...
END

```

Die Ausführung des vorstehenden Programms führt zu folgender Anzeige:

FCDP=ON:

#FIELD1 _____ #FIELD2 _____

FCDP=OFF:

#FIELD1 #FIELD2 _____

81

FDDM - Natural System File for DDMs

This Natural profile parameter is used to specify five subparameters for the Natural system file for DDMs.

If this system file is defined, all DDMs are stored on the specified path. DDMs stored in libraries will no longer be accessible from Natural. This is similar to Natural on mainframes, where all DDMs are stored in the system file `FDIC`.

If the FDDM system file is undefined (*database-ID* and *file-number* = 0), the DDMs are stored in the libraries as supplied before. The system file FDDM is displayed as an inactive environment.

For information on system files, refer to: System Files in the Natural Operations documentation.

Possible settings	<i>database-ID</i>	1 - 254	Database ID 255 is reserved for logical system files for Software AG products, see Natural profile parameter LFILE .
	<i>file-number</i>	1 - 255	File number.
	<i>password</i>	1 - 8 characters	The password is only required if the Natural user-program system file has been password-protected using the Adabas security feature. The password is reserved for future use; currently, it is ignored.
	<i>cipher key</i>	1 - 8 numeric digits	The cipher key is only required if the Natural user-program system file has been ciphered using the Adabas security feature. The cipher key is reserved for future use; currently, it is ignored.
	RO		Indicates that the Natural user-program system file is „read-only“ and is only specified if modifications on the file are to be disabled.

Default setting	none
Dynamic specification	yes
Specification within session	no

The syntax of this parameter is:

```
FDDM=(database-ID,file-number,password,cipher-key,RO)
```

Examples:

```
FDDM=(22,5)
FDDM=(22,5,,12345)
FDDM=(22,5,,,RO)
```

82

FDIC - Predict System File

This Natural profile parameter defines four subparameters for the Predict system file which Predict uses to retrieve and/or store data.



Anmerkung: In a remote development environment, a Development Server File is used instead, see the *SPoD - Natural's Single Point of Development* and the *Natural Development Server* documentation.

For specific information on system files, refer to:

System Files in the *Operations* documentation.

Possible settings	<i>database-ID</i>	1 - 65535, except 255	Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE . If no FDIC is available, do not enter anything in the DBID field.
	<i>file-number</i>	:1 - 5000	File number. If no FDIC is available, do not enter anything in the file number field.
	<i>password</i>	1 to 8 characters	The password is only required if the Predict system file has been password-protected using the Adabas security feature.
	<i>cipher-key</i>	1 to 8 numeric digits	Cipher key for the Predict system file. It is only required if the Predict system file has been ciphered using the Adabas security feature.
		RO	Read only option - not supported on this platform.
Default setting	none		
Dynamic specification	yes		
Specification within session	no		

The syntax for this parameter is:

```
FDIC=(database-ID,file-number,password,cipher-key,RO)
```

Examples:

```
FDIC=(10,5,PASSW1,12345678)
FDIC=(1,200,,12345678)
FDIC=(1,5)
FDIC=(,5)
```

83 FL - Floating Point Mantissa Length

Fließpunkt-Mantissenlänge

Mit diesem Session-Parameter bestimmen Sie die Mantissen-Länge einer Gleitkomma-Variablen während der Ein- oder Ausgabe.

Mögliche Werte	1 - 16	Mantissen-Länge. Die Gesamtlänge für Vorzeichen, Exponent und Dezimalkomma ist FL + 6.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT INPUT PRINT WRITE
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY FL=5 -> +1.2345E+03
```


84

FNAT - Natural System File for System Programs

This Natural profile parameter is used to specify the database ID, file number, password and cipher key and read-only flag for the Natural system file for Natural system programs.

The Natural system file is the file from which all Natural system programs are retrieved and upon which all system commands operate. Error texts and Natural help information related to the Natural system libraries are also contained in this system file.

For specific information on system files, refer to *System Files* in the *Operations* documentation.

Possible settings	<i>database-ID</i>	1-254	Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
	<i>file-number</i>	1-255	File number.
	<i>password</i>	1 to 8 characters	The password is only required if the Natural system file has been password-protected using the Adabas security feature. The password is reserved for future use; currently, it is ignored.
	<i>cipher key</i>	1 to 8 numeric digits	The cipher key is only required if the Natural system file has been ciphered using the Adabas security feature. The cipher key is reserved for future use; currently, it is ignored.
	R0		Read-only option. R0 indicates that the Natural system file is „read-only“ and is only specified if modifications on the file are to be disabled.
Default setting	none		
Dynamic specification	yes		

Specification within session	no
-------------------------------------	----

The syntax of this parameter is:

FNAT=(*database-ID,file-number,password,cipher-key,RO*)

Examples:

FNAT=(,102)
FNAT=(99,102, , ,RO)
FNAT=(99,102,PASSW2)

85

FREEGDA - Release GDA in Utility Mode

This Natural profile parameter controls whether current user global data area (GDA) and application-independent variables (AIV) are to be reset or not when a utility is invoked in utility mode (see *Utility Activation* in the *Utilities* documentation), that is, by using the direct command that corresponds to the utility's name.

Possible settings	ON	The current user GDA and AIV variables are reset before a utility is started. This behavior corresponds to the previous situation when the utility was invoked using the system command LOGON <library>.
	OFF	The current user GDA and AIV variables are preserved when a utility is started. Note that this will increase the data size correspondingly and may lead to thread problems under certain operating systems.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

86

FS - Default Format/Length Setting for User-Defined

Variables

Format-Spezifikation für Benutzervariablen

Dieser Natural Profil- und Session-Parameter gilt nur für den Reporting Mode; er hat keine Auswirkungen im Structured Mode.

Mit diesem Session-Parameter bestimmen Sie, ob für die Definition von Benutzervariablen im Reporting Mode Standardformat und -länge gelten sollen (siehe auch *Format und Länge von Benutzervariablen im Leitfaden zur Programmierung*).

In einer Natural-Session kann der Profilparameter FS durch den Session-Parameter FS überschrieben werden.

Mögliche Werte	ON	Einer neuen Benutzervariablen wird im Reporting Mode von Natural kein/e Standardformat/-länge zugeordnet; Sie müssen Format und Länge explizit definieren.	
	OFF	Eine Benutzervariable in einem Natural-Programm, für die Format und Länge nicht explizit definiert sind, erhält Standardformat/-länge N7.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Applicable Statements:	SET GLOBALS
		Applicable Command:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session-Parameter-Option des Library-Profiles überschrieben werden.

87

FSEC - Natural Security System File

This Natural profile parameter only applies if Natural Security is used.

This parameter defines four subparameters for the Natural Security system file which is used by Natural Security to retrieve/store its security information.

For information on system files, refer to *System Files* in the *Operations* documentation.

Possible settings	<i>database-ID</i>	1-65535, except 255	Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
		0	DBID=0 sets FSEC inactive. This is mandatory for a non-security environment. If no FSEC system file is available, do not enter anything in the file number field.
	<i>file-number</i>	1-5000	File number for the Natural Security system file.
		0	FNR=0 sets FSEC inactive. This is mandatory for a non-security environment. If no FSEC system file is available, do not enter anything in the file number field.
	<i>password</i>	1 to 8 characters	The password is only required if the Natural Security system file has been password-protected using the Adabas security feature. The password is reserved for future use; currently, it is ignored.
	<i>cipher-key</i>	1 to 8 numeric digits	The cipher key is only required if the Natural Security system file has been ciphered using the Adabas security feature. The cipher key is reserved for future use; currently, it is ignored.

	RO	Read only option - not supported on this platform.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

The syntax of this parameter is:

```
FSEC=(database-ID,file-number,password,cipher-key,RO)
```

Examples:

```
FSEC=(10,8)
FSEC=10,5,PASSW1,12345678
FSEC=1,200,,12345678
```


88

FUSER - Natural System File for User Programs

This Natural profile parameter defines five subparameters for the Natural user-program system file. This system file is the database file from which all user-written Natural programs are retrieved.

For information on system files, refer to: *System Files* in the *Operations* documentation.

Possible settings	<i>database-ID</i>	1-254	Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
	<i>file-number</i>	1-255	File number.
	<i>password</i>	1 to 8 characters	The password is only required if the Natural user-program system file has been password-protected using the Adabas security feature. The password is reserved for future use; currently, it is ignored.
	<i>cipher-key</i>	1 to 8 numeric digits	The cipher key is only required if the Natural user-program system file has been ciphered using the Adabas security feature. The cipher key is reserved for future use; currently, it is ignored.
	RO		Read-only option. RO indicates that the Natural user-program system file is „read-only“ and is only specified if modifications on the file are to be disabled.
Default setting	none		
Dynamic specification	yes		
Specification within session	no		

The syntax of this parameter is:

```
FUSER=(database-ID,file-number,password,cipher-key,RO)
```

Examples:

```
FUSER=( 22 , 5 )  
FUSER=( 22 , 5 , , , RO )  
FUSER=( 22 , 5 , PASSW2 )
```

89

GC - Filler Character for Group Headers

Füllzeichen für Gruppenüberschriften

Mit diesem Session-Parameter definieren Sie das Füllzeichen, mit dem bei Überschriften, die über ein `DISPLAY`-Statement erzeugt werden, der Platz rechts und links der Überschrift aufgefüllt wird.

Im Gegensatz zum Session-Parameter `FC` gilt dieses Zeichen für Überschriften, die über mehrere Spalten, also über eine Gruppe von Feldern, gehen.

Mögliche Werte	beliebiges Zeichen	Füllzeichen für Gruppenüberschriften.	
Standard-Einstellung	leer		
Spezifikation in Session	ja	Gültige Statements:	<code>DISPLAY</code> <code>FORMAT</code>
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY (GC=*)
```


90

GFID - Global Format IDs

This Natural profile and session parameter is used to control Natural's internal generation of global format IDs so as to influence Adabas's performance concerning the re-usability of format buffer translations.

Possible settings	ON	Global format IDs are generated for all views.	
	OFF	Global format IDs are not generated.	
	VID	Global format IDs are generated only for views in local/global data areas, but not for views defined within programs.	
Default setting	ON		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Commands:	Either GLOBALS or GFID option of COMPOPT

For details on global format IDs, see the Adabas documentation.

91 HC - Header Centering

Überschriften-Zentrierung

Mit diesem Session-Parameter bestimmen Sie die Ausrichtung von Spaltenüberschriften.

Mögliche Werte	C	Spaltenüberschriften werden zentriert ausgegeben.	
	L	Spaltenüberschriften werden linksbündig ausgegeben.	
	R	Spaltenüberschriften werden rechtsbündig ausgegeben.	
Standard-Einstellung	C		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY (HC=L)
```

92 HD - Header Definition

Festlegung der Standard-Spaltenüberschrift

Mit diesem Session-Parameter können Sie festlegen, welche Standard-Spaltenüberschrift verwendet werden soll, wenn

- das Feld mit einem `DISPLAY`-Statement ausgegeben wird;
- ein Gleichheitszeichen (=) unmittelbar vor das Feld in einem `WRITE`- oder `INPUT`-Statement platziert wird.

Mögliche Werte	<code>'text'</code>	Maximal 120 alphanumerische or Unicode Zeichen.
Standard-Einstellung	keine	
Gültiges Statement:	<code>DEFINE DATA</code>	Parameter kann auf Feld-/Elementebene angegeben werden.
Gültiges Kommando:	keines	

93

HE - Helproutine

- HE-Parameter-Syntax 218
- Ausführung von Helproutinen 220
- Beispiele 220

Mit diesem Session-Parameter können Sie den Namen einer Helproutine angeben, die einem Feld zugewiesen werden soll.

Mögliche Werte		Siehe <i>HE-Parameter-Syntax</i> weiter unten.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	INPUT
		Gültiges Kommando:	keines

Helproutinen können mit dem Natural-Programm-Editor erstellt werden, Help Maps mit dem Natural-Map-Editor.

Die zugewiesene Helproutine kann dann bei der Verarbeitung des betreffenden INPUT-Statements oder der betreffenden Map aufgerufen werden, indem der Benutzer ein Fragezeichen (?) (das ist die Voreinstellung) in das Feld eingibt oder den Cursor in das Feld plaziert und die mit dem SET KEY-Statement definierte Hilfe-Funktionstaste drückt.

Wenn ein Wert von einer Helproutine an ein Eingabefeld übergeben werden soll, muss das Feld als änderbar (AD=M) definiert sein.

Die folgenden Themen werden nachfolgend erörtert:

HE-Parameter-Syntax

Der Parameter hat die folgende Syntax:

$$HE=\text{operand1} \left[, \left\{ \begin{array}{l} \text{operand2} \\ = \\ nX \end{array} \right\} \right] \dots 20$$

Operanden-Definitionstabelle:

Operand	Mögliche Struktur	Mögliche Formate	Referenzierung erlaubt	Dynam. Definition
<i>operand1</i>	C S	A	nein	nein
<i>operand2</i>	C S A	A U N P I F B D T L C G O	nein	nein

Syntax-Beschreibung:

<i>operand1</i>	<i>operand1</i> ist der Name der Helproutine oder Map, die aufgerufen werden soll. Der Name kann eine 1 bis 8 Zeichen umfassende alphanumerische Konstante oder Benutzervariable sein. Wenn eine Variable verwendet wird, muss sie vorher definiert worden sein. Der Name kann ein Kaufmännisches Und (&) enthalten; zur Ausführungszeit wird dieses Zeichen durch den aus einem Zeichen bestehenden Code ersetzt, der dem aktuellen Wert der Systemvariablen *LANGUAGE entspricht. Diese Funktion ermöglicht die Verwendung von mehrsprachigen Helproutinen oder Maps.
<i>operand2</i>	Nach dem Helproutinen-Namen können Sie 1 bis 20 Parameter (<i>operand2</i>) angeben, die an die Helproutine übergeben werden. Diese können Konstanten oder Benutzervariablen sein, die die Parameterwerte enthalten.
=	Geben Sie ein Gleichheitszeichen (=) als Parameter an, wird der Name des Feldes (unter dem es in der Map definiert ist) an die Helproutine übergeben. Ist die Helproutine nicht einem Feld, sondern einer Map zugeordnet, wird mit = der Map-Name übergeben. <ul style="list-style-type: none"> ■ Wird das Gleichheitszeichen bei HE= auf Statementebene angegeben, wird der Name des zurzeit ausgeführten Objekts (so wie er in der Systemvariablen *PROGRAM enthalten ist) an die Helproutine oder Help-Map übergeben. In Beispiel 3 ist der übergebene Objektname PROGRAM1. ■ Wird das Gleichheitszeichen bei HE= auf Elementebene (Feldebene) angegeben, wird der Name des Feldes an die Helproutine oder Help-Map übergeben. In Beispiel 3 ist der übergebene Feldname #PARM1. <p>Wird mit = ein Feld- bzw. Map-Name übergeben, so muss der entsprechende Parameter in der Helproutine mit Format/Länge A65 definiert werden.</p>
<i>nX</i>	Die Notation <i>nX</i> können Sie verwenden, um wegzulassende Parameter anzugeben, das heisst, Parameter, für die keine Werte übergeben werden sollen. Die entsprechenden empfangsseitigen Parameter im DEFINE DATA PARAMETER-Statement der aufgerufenen Helproutine müssen als OPTIONAL definiert werden.

**Anmerkungen:**

1. Die Operanden müssen entweder mit dem Input-Delimiterzeichen (wie mit dem Session-Parameter ID definiert) oder mit einem Komma voneinander getrennt werden. Ein Komma darf hierzu allerdings nicht verwendet werden, falls das Komma als Dezimalkomma (mit dem Session-Parameter DC) definiert ist.
2. Wenn Parameter angegeben werden, so muss die Helproutine mit einem DEFINE DATA PARAMETER-Statement beginnen, in dem Felder definiert werden, die in Format und Länge den übergebenen Parametern entsprechen.
3. Der Wert des Feldes, dem die Helproutine zugeordnet ist, kann in der Helproutine referenziert werden. Hierzu muss im DEFINE DATA PARAMETER-Statement der Helproutine ein Feld definiert werden, das in Format und Länge dem ursprünglichen Feld entspricht. Werden in dem DEFINE DATA PARAMETER-Statement noch andere Felder definiert, so muss dieses Feld immer als letztes definiert werden.

4. Ist das Feld, für das die Helproutine angegeben wird, ein Element eines Arrays, so können die Ausprägungen dieses Feldes von der Helproutine referenziert werden; hierzu müssen Sie Index-Parameter mit Format I (Integer), N (numerisch ungepackt) oder P (gepackt numerisch) am Schluss des `DEFINE DATA PARAMETER`-Statements definieren. Entsprechend der Array-Dimensionen können Sie bis zu drei Index-Parameter angeben.

Ausführung von Helproutinen

Wenn eine Helproutine — durch Eingabe eines Fragezeichens (?) in das Feld oder durch Drücken der (mit einem `SET KEY`-Statement definierten) Hilfetaste, oder über ein `REINPUT USING HELP`-Statement — aufgerufen wird, werden alle in andere Felder eingegebenen Werte erst verarbeitet, nachdem die Ausführung der Helproutine beendet ist.



Anmerkung: Pro `INPUT`-Statement ist jeweils nur eine Hilfe-Anforderung möglich. Wenn für mehrere Felder gleichzeitig Hilfe angefordert wird (z.B. durch Eingabe von Fragezeichen in mehrere Felder), wird nur die erste Hilfe-Anforderung ausgeführt.

Beispiele

Beispiel 1:

```
/* MAIN PROGRAM
DEFINE DATA
1 #A(A20/1:3)
END-DEFINE
...
SET KEY PF1=HELP
...
INPUT #A (2) (HE='HELPA',=)
...
END
```

Beispiel 2:

```
/* HELP-ROUTINE 'HELPA'
DEFINE DATA PARAMETER
1 #VARNAME (A65)
1 #PARM1 (A20)
1 #VARINDEX (I2)
END-DEFINE
...
```

Beispiel 3:

```
* Program 'PROGRAM1'
*
DEFINE DATA LOCAL
1 #PARM1 (A65) INIT <'valueparm1'>
END-DEFINE
SET KEY PF1 = HELP
FORMAT KD=ON
*
INPUT (AD=M HE='HELP1',=)
  'Enter ? for name of executed object:'
  / #PARM1
*
INPUT (AD=M)
  'Enter ? for field name:'
  / #PARM1 (HE='HELP1',=)
*
END
```

Parameter Data Area in Beispiel-Helproutine HELP1:

```
* Helproutine 'HELP1'
*
DEFINE DATA PARAMETER
1 #FLD1 (A65)
END-DEFINE
...
```


94

HI - Help Character

This Natural profile parameter defines the character which is to be used to invoke a field-specific helproutine or a map helproutine (if defined for a given map).

Possible settings	any special character	The character which is to be used to invoke a field-specific helproutine or a map helproutine. The character specified with the profile parameter HI must not be the same as the one specified with the profile/session parameter CF (control character for mainframe terminal commands); it should not be the same as the one specified with the profile/session parameter DC (decimal character), profile/session parameter IA (input assign character) or profile/session parameter ID (input delimiter character).
	blank	When HI=' ' is set, a help key must be defined in the Natural application, using the SETKEY statement correspondingly; otherwise it is not possible to invoke a helproutine for any field.
Default setting	?	Question mark.
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR0350N	See SYSEXT - <i>Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

95 HW - Heading Width

Überschriftenbreite

Mit diesem Session-Parameter bestimmen Sie die Breite einer mit einem `DISPLAY`-Statement erzeugten Ausgabespalte.

Mögliche Werte	ON	Die Breite einer <code>DISPLAY</code> -Spalte wird entweder durch die Länge des Überschriftentextes oder die Länge des Feldes bestimmt, je nachdem was länger ist. Dies gilt auch, wenn kein Überschriftentext ausgegeben wird, entweder weil das <code>DISPLAY</code> -Statement das Schlüsselwort <code>NOHDR</code> enthält, oder das <code>DISPLAY</code> -Statement ein sekundäres <code>DISPLAY</code> ist (siehe auch <code>DISPLAY</code> -Statement).	
	OFF	Die Breite einer <code>DISPLAY</code> -Spalte wird durch die Länge des Feldes bestimmt. <code>HW=OFF</code> gilt nur für <code>DISPLAY</code> -Statements, die keine Überschriften erzeugen (d.h. entweder ein erstes <code>DISPLAY</code> -Statement mit der <code>NOHDR</code> -Option oder ein sekundäres <code>DISPLAY</code> -Statement).	
Standard-Einstellung	ON		
Spezifikation in Session	ja	Gültige Statements:	<code>DISPLAY</code> <code>FORMAT</code>
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY (HW=OFF)
```


96

IA - Input Assign Character

Input-Zuweisungszeichen

Das mit diesem Session-Parameter definierte Zeichen gilt als Zuweisungszeichen für Eingabeparameter bei der Verarbeitung von `INPUT`-Statements im Keyword/Delimiter-Modus oder bei der Verarbeitung von Daten aus dem Natural-Stack.

In einer Natural-Session kann der Profilparameter `IA` durch den Session-Parameter `IA` überschrieben werden.

Mögliche Werte	jedes Sonderzeichen	Zuweisungszeichen für die Verarbeitung des Eingabeparameters in <code>INPUT</code> -Statements. Das mit dem <code>IA</code> -Parameter angegebene Zeichen <ul style="list-style-type: none"> ■ muss ein anderes sein als das mit dem Profil/Session-Parameter <code>CF</code> angegebene (Steuerzeichen für Terminalkommandos auf Großrechnern), <code>DC</code> (Dezimalzeichen) oder <code>ID</code> (<code>INPUT</code>-Delimiterzeichen); ■ sollte ein anderes sein als das mit dem <code>HI</code>-Profilparameter (Hilfezeichen) definierte. 	
Standard-Einstellung	=	Gleichheitszeichen.	
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Beispiel:

In dem folgenden Beispiel wird davon ausgegangen, dass anfänglich das standardmäßige INPUT-Zuweisungszeichen (=) gilt.

```
** Example 'IACHAR': Input Assign character
*****
DEFINE DATA LOCAL
1 #A (A1)
1 #B (A1)
END-DEFINE
*
INPUT #A #B
*
WRITE 'Field #A:' #A / 'Field #B:' #B
*
END
```

1. Geben Sie das folgende Kommando ein:

```
IACHAR #A=Y,#B=X
```

Das Programm erzeugt die folgende Ausgabe:

```
Page      1                                05-01-19  11:05:51
Field #A: Y
Field #B: X
```

2. Geben Sie das folgende Kommando ein:

```
GLOBALS IA=:
```

Das INPUT-Zuweisungszeichen wird auf Doppelpunkt (:) gesetzt.

3. Geben Sie dann das folgende Kommando ein:

```
IACHAR #B:X,#A:Y
```

Das Programm erzeugt die folgende Ausgabe:

Page 1

06-11-13 12:12:24

Field #A: Y

Field #B: X

Unter Natural Security: Die Einstellung dieses Parameters kann durch die *Session-Parameter-Option* des Library-Profiles überschrieben werden.

97 IC - Insertion Character

Einfügungszeichen

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein `DISPLAY`-Statement ausgegeben wird, unmittelbar vor dem Feldwert ausgegeben. Die Ausgabelänge des Feldes vergrößert sich dadurch entsprechend.



Anmerkung: Der Unterschied zwischen den Session-Parametern `LC`, `LCU` und `IC`, `ICU` wird ersichtlich, wenn das entsprechende Feld rechtsbündig ausgegeben wird (`AD=R`). Das Einfügungszeichen wird zwischen den führenden Leerzeichen und dem Feldwert eingefügt, wohingegen das vorangestellte Zeichen vor dem führenden Leerzeichen ausgegeben wird.

Bei numerischen Werten werden die Einfügungszeichen vor der ersten signifikanten Stelle ausgegeben.

Die Parameter `IC` und `LC` schließen einander aus.

Der Parameter `IC` kann auch bei Feldern mit Format `U` verwendet werden. Informationen zum Unicode-Format siehe *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC* in der *Unicode and Code Page Support*-Dokumentation.

Mögliche Werte	beliebige Zeichen	Einzufügende Zeichenkette. Sie können eine Zeichenkette von eins bis zehn Zeichen eingeben. Sie können die Zeichenkette wahlweise in Apostrophen (') angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen (") oder eine schließende Klammer enthält, muss in Apostrophen stehen. Ein Leerzeichen in einer nicht durch Apostrophe eingegrenzten Zeichenkette wird durch ein Zirkumflex (^) dargestellt.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT

	Gültiges Kommando:	Keines
--	--------------------	--------

Beispiel:

```
DISPLAY AA(IC=*)  
DISPLAY SALARY(IC='$')
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

98

ICU - Unicode Insertion Character

Unicode-Einfügsungszeichen

Der Session-Parameter `ICU` ist identisch mit dem Session-Parameter `IC`. Der Unterschied ist, dass das Einfügsungszeichen immer im Unicode-Format gespeichert wird. Dadurch können Sie die Einfügsungszeichen mit gemischten Zeichen aus unterschiedlichen Codepages eingeben. Es ist sichergestellt, dass immer das korrekte Zeichen angezeigt wird, unabhängig von der installierten System-Codepage.

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein `DISPLAY`-Statement ausgegeben wird, unmittelbar vor dem Feldwert ausgegeben. Die Ausgabelänge des Feldes vergrößert sich dadurch entsprechend.

Bei numerischen Werten werden die Einfügsungszeichen vor der ersten signifikanten Stelle ausgegeben.

Die Parameter `ICU` und `LCU` schließen einander aus.

Mögliche Werte	beliebige Zeichen	Einzufügende Zeichenkette. Sie können eine Zeichenkette von eins bis zehn Zeichen eingeben. Sie können die Zeichenkette wahlweise in Apostrophen (') angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen (") oder eine schließende Klammer enthält, muss in Apostrophen stehen. Ein Leerzeichen in einer nicht durch Apostrophe eingegrenzten Zeichenkette wird durch ein Zirkumflex (^) dargestellt.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	<code>DISPLAY</code> <code>FORMAT</code>
		Gültiges Kommando:	Keines

Siehe auch:

- *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*
- *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC in der Unicode and Code Page Support-Dokumentation.*

99

ID - Input Delimiter Character

Input-Delimiterzeichen

Das mit dem Natural Profil- und Session-Parameter definierte Zeichen gilt als Delimiterzeichen zum Abgrenzen von Werten bei `INPUT`-Statements im Keyword/Delimiter-Modus.

In einer Natural-Session kann der Profilparameter `ID` durch den Session-Parameter `ID` überschrieben werden.

Mögliche Werte	jedes Sonderzeichen	<p>INPUT-Delimiterzeichen. Das mit diesem Parameter angegebene Zeichen</p> <ul style="list-style-type: none">■ darf nicht dasselbe sein wie das mit dem Profil/Session-Parameter <code>DC</code> (Dezimalzeichen) oder <code>IA</code> (INPUT-Zuweisungszeichen) angegebene Zeichen;■ sollte nicht dasselbe sein wie das mit dem <code>CF</code>-Parameter (Steuerzeichen für Großrechner-Terminalkommandos) oder <code>HI</code>-Parameter (Hilfezeichen) angegebene Zeichen. <p>Der Punkt (.) sollte nicht als INPUT-Delimiter benutzt werden, weil dies zu Situationen führen könnte, in denen ein Programmende-Punkt fälschlicherweise als INPUT-Delimiterzeichen interpretiert wird. Ein Stern (*) sollte auch nicht verwendet werden.</p>
-----------------------	---------------------	--

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des Library-Profiles überschrieben werden.

100

IKEY - Processing of PA and PF Keys

This Natural profile parameter specifies the action to be taken when a video-terminal program-attention key (PA key) or program-function key (PF key) is used to enter data, and the key has not been defined to the Natural program with the `SET KEY` statement.

Possible settings	ON	Natural reacts as if ENTER had been pressed.
	OFF	A REINPUT message is generated, prompting the user to press a valid key.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

101 IM - Input Mode

Input-Modus

Dieser Natural Profil- und Session-Parameter bestimmt den Standardmodus für Video-Terminals.

In einer Natural-Session kann die Einstellung des Profilparameters IM durch den Session-Parameter IM überschrieben werden.

Mögliche Werte	F	Forms-Modus.	
	D	Delimiter-Modus.	
Standard-Einstellung	F		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Weitere Informationen zum Delimiter-Modus und Forms-Modus, entnehmen Sie dem INPUT-Statement.

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des Library-Profiles überschrieben werden.

102

INIT-LIB - Library for Automatic Logon

This Natural profile parameter specifies the name of the library to be used for an automatic logon (see the profile parameter [AUTO](#)) when Natural is started.

Possible settings	1-8 characters	Valid library name.
Default setting	none	
Dynamic specification	no	
Specification within session	no	

If Natural Security is installed, `INIT-LIB` is not evaluated; the library to be used for automatic logon is read from the `FSEC` system file (see the *Natural Security* documentation for further information).

103

IP - INPUT Prompting Text

Eingabeaufforderungstext

Mit diesem Session-Parameter wird bei INPUT-Statements der Text, der zur Eingabe auffordert, gesteuert.

Mögliche Werte	ON	Eingabe-/Ausgabefeldern eines INPUT-Statements, denen kein Text-element vorangestellt ist, wird der betreffende Feldname vorangestellt.	
	OFF	Es wird kein Eingabeaufforderungstext in Form von Feldnamen generiert; nur wenn einem Feld explizit ein Textelement vorangestellt ist, wird dieser Text als Eingabeaufforderung ausgegeben.	
Standard-Einstellung	ON		
Spezifikation in Session	ja	Gültige Statements:	FORMAT INPUT
		Gültiges Kommando:	Keines

Beispiel:

```
FORMAT IP=OFF
```


104 IS - Identical Suppress

Unterdrückung identischer Werte

Mit diesem Session-Parameter können Sie die mehrfache Ausgabe identischer Feldwerte in aufeinander folgenden Zeilen bei einem `WRITE-` oder `DISPLAY-`Statement unterdrücken.

Mit dem Statement `SUSPEND IDENTICAL SUPPRESS` können Sie die Wirkung von `IS=ON` für einen einzelnen Datensatz unterdrücken.

Der `IS`-Parameter kann in Verbindung mit den Parametern `ES` und `ZP` zur Unterdrückung der Ausgabe von Leerzeilen eingesetzt werden.

Mögliche Werte	ON	Wenn <code>IS=ON</code> gesetzt ist, wird der Wert eines Feldes nicht angezeigt, falls er mit dem vorherigen Wert des Feldes identisch ist. Erzeugt ein <code>DISPLAY-</code> oder <code>WRITE-</code> Statement unter Verwendung der <code>VERT-</code> Option oder der Schrägstrich-Notation (<code>/</code>) mehrzeilige Ausgaben, so gilt <code>IS=ON</code> nur für die jeweils erste Zeile.	
	OFF	Es wird keine automatische Unterdrückung angewendet.	
Standard-Einstellung	OFF		
Spezifikation in Session	ja	Gültige Statements:	<code>DISPLAY</code> <code>FORMAT</code> <code>WRITE</code>
		Gültiges Kommando:	Keines

Beispiel:

```
FORMAT IS=ON
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

105

KC - Check for Statement Keywords

This parameter corresponds to the Natural profile parameter [KCHECK](#).

106

KCHECK - Check for Statement Keywords

This profile parameter checks field declarations in a programming object against a set of critical Natural keywords. If a variable name defined matches one of these keywords, a syntax error is reported when the programming object is checked or cataloged.

Possible settings	ON	The check for keywords is performed.	
	OFF	No check for keywords is performed.	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	KCHECK option of COMPOPT

The section *Performing a Check for Natural Reserved Keywords* contains a list of the keywords that are checked by the KCHECK parameter.

The document *Natural Reserved Keywords* in the *Programming Guide* contains an overview of all Natural keywords and reserved words.

107

KD - Key Definition

PF-Tasten-Anzeige

Dieser Session-Parameter dient dazu, die PF-Tasten zugewiesenen Namen (siehe SET KEY-Statement) anzeigen zu lassen.

Wenn `KD=ON` gesetzt ist, werden diese Informationen automatisch bei jeder mit `INPUT`, `WRITE`, `DISPLAY` und `PRINT` erzeugten Ausgabe am unteren Bildschirmrand angezeigt.

Da diese Anzeige zwei Zeilen in Anspruch nimmt, muss die logische Seitenlänge (siehe Session-Parameter `PS`) entsprechend um zwei Zeilen reduziert werden.

Mögliche Werte	ON	Die den PF-Tasten zugewiesenen Namen werden angezeigt.	
	OFF	Die den PF-Tasten zugewiesenen Namen werden nicht angezeigt.	
Standard-Einstellung	OFF		
Spezifikation in Session	ja	Gültige Statements:	FORMAT
		Gültiges Kommando:	Keines

Hinweis für grafische Benutzeroberflächen:

Falls PF-Tasten definiert sind, werden sie, unabhängig von der Einstellung dieses Parameters, immer angezeigt. Sind keine PF-Tasten definiert, kann der Parameter benutzt werden, um die Anzeige der `ENTER`-Taste ein- bzw. auszuschalten.

Beispiel:

```
FORMAT KD=ON
```

108

LC - Lower to Upper Case Translation

This Natural profile parameter controls lower-case to upper-case translation of input characters.



Anmerkung: This parameter does not apply to Natural stack data which was placed on the Natural stack by the `STACK` statement.

Possible settings	ON	No translation of lower-case characters to upper case is performed.
	OFF	All lower-case characters, except input from the Natural stack which was place there by the <code>STACK</code> statement, is translated to upper case by Natural.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	To disable or enable lower-case to upper-case translation dynamically within the active Natural session, you should use the terminal commands <code>%L</code> or <code>%U</code>
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

109 LC - Leading Characters

Vorangestellte Zeichen

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein `DISPLAY`-Statement ausgegeben wird, unmittelbar vor dem Feld ausgegeben. Die Breite der Ausgabespalte vergrößert sich dadurch entsprechend.

Die Session-Parameter `LC` und `IC` schließen einander aus.

Der Parameter `LC` kann auch mit Felder des Formats `U` benutzt werden. Informationen zu Unicode-Format entnehmen Sie dem Dokument *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC*.

Mögliche Werte	beliebige Zeichen	<p>Sie können eine Zeichenkette von 1 bis 10 Zeichen definieren.</p> <p>Sie können die Zeichenkette wahlweise in Apostrophen (') angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen (") oder eine schließende Klammer enthält, muss in Apostrophen stehen. Ein Leerzeichen in einer nicht durch Apostrophe eingegrenzten Zeichenkette wird durch ein Zirkumflex (^) dargestellt.</p>	
Standard-Einstellung	Keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY {LC=*}
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

110

LCU - Unicode Leading Characters

Vorangestellte Unicode-Zeichen

Der Session-Parameter `LCU` ist identisch mit dem Session-Parameter `LC`. Der Unterschied ist, dass die vorangestellten Zeichen immer im Unicode-Format gespeichert werden. Dadurch können Sie die vorangestellten Zeichen mit gemischten Zeichen aus unterschiedlichen Codepages eingeben. Es ist sichergestellt, dass immer das korrekte Zeichen angezeigt wird, unabhängig von der installierten System-Codepage.

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein `DISPLAY`-Statement ausgegeben wird, unmittelbar vor dem Feld ausgegeben. Die Breite der Ausgabespalte vergrößert sich dadurch entsprechend.

Die Session-Parameter `LCU` und `ICU` schließen einander aus.

Mögliche Werte	beliebige Zeichen	Sie können eine Zeichenkette von 1 bis 10 Zeichen definieren. Sie können die Zeichenkette wahlweise in Apostrophen (') angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen (") oder eine schließende Klammer enthält, muss in Apostrophen stehen. Ein Leerzeichen in einer nicht durch Apostrophe eingegrenzten Zeichenkette wird durch ein Zirkumflex (^) dargestellt.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keines

Siehe auch:

- *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*

- *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC in der Unicode and Code Page Support-Dokumentation.*

111

LDB - Wait Time for Response of Local Adabas Database

This Natural profile parameter specifies the time limit Natural is to wait for a response of the local Adabas database.

Possible settings	1 - 999	Time limit in seconds. If it is exceeded, an appropriate error message is issued.
	0	No time limit will be in effect.
Default setting	30	
Dynamic specification	no	
Specification within session	no	

If Entire Net-Work is installed, the specified time limit also affects Entire Net-Work's timeout processing.

112

LE - Reaction when Limit for Processing Loop Exceeded

Reaktion auf Limit-Überschreitung bei Verarbeitungsschleifen

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, was geschehen soll, wenn bei der Ausführung einer Verarbeitungsschleife das angegebene Limit (d.h. die maximale Anzahl der Schleifendurchläufe) bei einem Statement `READ`, `FIND` oder `HISTOGRAM` erreicht wird.

Das Limit kann entweder ein (mit dem `LIMIT`-Statement angegebenes) globales Limit oder ein schleifenspezifisch festgesetztes Limit sein.

In einer Natural-Session kann der Profilparameter `LE` durch den Session-Parameter `LE` überschrieben werden.

Mögliche Werte	ON	Die Schleife wird beendet, wenn das Limit erreicht ist. Das Programm wird dann mit dem auf die beendete Schleife folgenden Statement fortgesetzt. Nach Beendigung der Ausführung des Natural-Objekts, wird der Fehler NAT0957 (Database loop limit reached with 'LE=ON'.) ausgegeben. LE=ON gilt nur für Programme, die von einer in der Systemdatei <code>FUSER</code> abgelegten Library geladen werden, d.h. der Library <code>SYSTEM</code> , oder einer Library mit einem Namen, der nicht mit dem Präfix <code>SYS</code> anfängt.	
	OFF	Die Schleife wird beendet, wenn das Limit erreicht ist. Das Programm wird dann mit dem auf die beendete Schleife folgenden Statement fortgesetzt. Nach Beendigung der Ausführung des Natural-Objekts, wird keine Fehlermeldung ausgegeben.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS

Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities-Dokumentation</i> .
---------------------------------------	----------	--

Beispiel:

```
DEFINE DATA LOCAL
1 EMPL-VIEW VIEW OF EMPLOYEES
  2 NAME
END-DEFINE
READ (10) EMPL-VIEW BY NAME
  WRITE NAME
END-READ
END
```

LE=OFF: nach 10 Datensätzen wird die Schleife ohne eine Meldung beendet.

LE=ON: nach 10 Datensätzen wird die Schleife mit einer Fehlermeldung NAT0957 (Database loop limit reached with 'LE=ON') beendet.

113

LFILE - Logical System File Definition

- LFILE Parameter Syntax 265
- Example of LFILE Parameter 265

This Natural profile parameter specifies information concerning the physical database file to be associated with a logical system file for Software AG products.

It can be used for Software AG products which have their own system files (for example, Connect and Natural Elite) to specify where such a system file is to be located. Such products use Database ID 255 and a logical file number (FNR) in their data definition modules (DDMs). With the LFILE parameter, you specify which physical database ID (DBID) and file number (and, if applicable, password and cipher key) are associated with that logical file number.

Natural records the physical file information and uses it for any database calls to Database ID=255 and File number=*logical-ID*.

Possible settings	<i>logical-FNR</i>	1 - 251	Logical file number (LFL). This parameter is mandatory.
	<i>physical-DBID</i>	0 - 65535, except 255	Physical database ID (DBID). Database ID 255 is reserved for logical system files for Software AG products.
	<i>physical-FNR</i>	1 - 5000	Physical file number (FNR).
	<i>password</i>		Must be a setting of 1 to 8 characters. *
	<i>cipher-key</i>		Must be a setting of 1 - 8 numeric digits. *
	<i>options</i>	RO	For read-only access. The RO flag is not supported on this platform.
Default setting	none		
Dynamic specification	yes		
Specification within session	no		
Application Programming Interface	USR0011N	See SYSEXT - <i>Natural Application Programming Interfaces</i> in the Utilities documentation. * Recommended.	
	USR2004N *		

* Password and cipher key are only required if the database file has been password-protected and/or ciphered using the Adabas security feature. With FDDM, FNAT and FUSER, the password and the cipher key are reserved for future use; currently they are ignored.

To define different logical files, the LFILE parameter must be specified several times.

LFILE Parameter Syntax

The LFILE parameter is specified as follows:

```
LFILE=(logical-FNR,physical-DBID,physical-FNR,password,cipher-key,RO)
```

Example of LFILE Parameter

```
LFILE=(180,73,10),LFILE=(251,40,9,TEST99)
```


114

LOGONRQ - Logon for RPC Server Request Required

This Natural profile parameter determines whether or not logon data are required for an RPC server request.

LOGONRQ is specified on the server side only.

Possible settings	ON	A logon is required; that is, the server only accepts requests from clients which include logon data in the RPC server request. For conversational requests, the logon data is only necessary when the conversation is opened.
	OFF	A logon is <i>not required</i> . Logon data is nevertheless processed.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

For Natural clients the logon data can either be requested:

- by setting the LOGON option of the SYSRPC Service Directory Maintenance;
- using the logon indicator of the profile parameter [DFS](#).

You are strongly recommended to set LOGONRQ=ON if the Natural RPC server runs under Natural Security. For further information, see *Using Natural RPC with Natural Security* in the *Natural Remote Procedure Call (RPC)* documentation.

For additional information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

115

LS - Line Size

▪ Profilparameter LS	270
▪ Session-Parameter LS	270
▪ Spezifikation in Statements	271

Zeilenlänge

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, wieviele Stellen eine von einem DISPLAY-, INPUT- oder WRITE-Statement erzeugte Zeile höchstens lang sein darf.

Die folgenden Themen werden behandelt:

Profilparameter LS

Wenn LS als Profilparameter benutzt wird, kommt LS im Batch-Betrieb zum Einsatz und definiert die physische Zeilenlänge. Im Online-Betrieb ist die Zeilenlänge immer auf die physische Bildschirmbreite gesetzt.

Mögliche Werte	35 - 250	Höchstwert der pro Zeile zulässigen Zeichen.
	0	Physische Zeilenlänge benutzen (meistens 132).
Standard-Einstellung	0	
Dynamische Spezifikation	ja	

Session-Parameter LS

Mögliche Werte	2 - 250	Höchstwert der pro Zeile zulässigen Zeichen.
Standard-Einstellung	0	Physische Zeilenlänge.
Gültiges Kommando	GLOBALS	
Gültige Statements	FORMAT SET GLOBALS	
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.



Anmerkung: Beim Logon in eine Library wird LS auf die physische Zeilenlänge zurückgesetzt.

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des Library-Profiles überschrieben werden.

Spezifikation in Statements

Wenn er in einem Statement angegeben wird, wird der LS-Parameter bei der Kompilierung ausgewertet.

Gültige Statements	DISPLAY INPUT WRITE	Der Parameter kann auf Statement-Ebene angegeben werden.
---------------------------	---------------------------	--

Limit für Verarbeitungsschleifen

Mit diesem Natural Profil- und Session-Parameter können Sie das allgemeine Limit für Verarbeitungsschleifen in Natural-Programmen bestimmen, d.h. wieviele Datensätze eine Verarbeitungsschleife in einer Natural-Anwendung maximal verarbeiten darf.

Dieses Limit gilt für alle Statements, mit denen Datensätze aus der Datenbank gelesen werden, d.h.:

- Statements, die eine Verarbeitungsschleife auslösen, wie zum Beispiel `READ`, `FIND`, `HISTOGRAM` oder `SELECT` und
- Statements, mit denen nur ein einzelner Datensatz gelesen wird, wie zum Beispiel `FIND UNIQUE`, `FIND NUMBER`, `FIND FIRST`, `GET (SAME)` und `SELECT SINGLE`.

Alle gelesenen Datensätze werden gezählt und das Ergebnis dieser Zählung wird mit dem im LT-Parameter gesetzten Wert verglichen. Hierbei werden alle gelesenen Datensätze mitgezählt, auch solche, die aufgrund einer `WHERE`-Klausel eines `FIND`-, `READ`- oder `HISTOGRAM`-Statements zurückgewiesen und nicht weiterverarbeitet werden. Das mit dem LT-Parameter gesetzte Limit hat keine Auswirkung auf die Statements `STORE`, `UPDATE`, `DELETE`, `END TRANSACTION` und `BACKOUT TRANSACTION`.

Wird ein Datensatz von der Datenbank gelesen, wird die Anzahl der gelesenen Datensätze erhöht, bevor sie mit dem Wert des LT-Parameters verglichen wird. Übersteigt der erhöhte Zählwert den aktuellen LT-Wert, wird der Natural-Fehler NAT1003 (Global limit for database calls reached) ausgegeben. Der Wert der gezählten gelesenen Datensätze wird immer dann auf Null gesetzt, wenn ein Natural-Programm auf Level 1 gestartet wird. Der Wert wird allerdings nicht zurückgesetzt, wenn das Programm auf Level 1 ein anderes Natural-Objekt aufruft (weitere Informationen hierzu siehe *Mehrere Stufen (Levels) aufgerufener Objekte* im Leitfaden zur Programmierung). Deshalb wird mit dem LT-Parameter die Anzahl der Datensätze begrenzt, die von einem Level 1-Programm und von den Objekten gelesenen wurden, die von diesem Programm auf einem anderen Level als Level 1 aufgerufen wurden.

Wird der Wert des LT-Parameters dynamisch in einem Programm mittels einem SET GLOBALS LT=*n*-Statement geändert, wird der neue Limit-Wert beim nächsten Statement wirksam, das einen Datensatz von der Datenbank liest.

In einer Natural-Session kann der Profilparameter LT durch den Session-Parameter LT überschrieben werden.

Mögliche Werte	0 - 2147483647 In einer Session: 0 bis <i>n</i> (<i>n</i> = Wert des Profilparameters LT beim Start der Session)	Maximale Anzahl der Datensätze, die in einer gegebenen Verarbeitungsschleife gelesen werden können. Hierbei werden alle gelesenen Datensätze mitgezählt, auch solche, die aufgrund einer WHERE-Klausel zurückgewiesen und nicht weiterverarbeitet werden. LT=0 legt fest, dass für Verarbeitungsschleifen kein Limit gelten soll.	
Standard-Einstellung	99999999		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS, siehe Anmerkung.
		Gültiges Kommando:	GLOBALS, siehe Anmerkung.
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	



Anmerkung: Wird der LT-Parameter mit einem SET GLOBALS-Statement oder einem SET GLOBALS-Systemkommando benutzt, kann der einstellbare Limit-Wert nicht größer als der LT-Wert sein, der im in der Natural Parameterdatei NATPARM gesetzt wurde.

117 MADIO - Maximum DBMS Calls between Screen I/O

Operations

This Natural profile parameter is used to specify the maximum number of DBMS calls permitted between two screen I/O operations (also in batch mode).

Possible settings	30 - 32767	Maximum number of DBMS calls.
	0	MADIO=0 indicates that no limit is to be in effect.
Default setting	512	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation. * Recommended.
	USR1068N *	

If the specified limit is exceeded, the Natural program is interrupted and the user is notified with an appropriate Natural error message.

118

MAINPR - Override Default Output Report Number

This Natural profile parameter is used to override the default output report number for all Natural reports. It must be set to a valid printer number (0 - 31).

Possible settings	0 - 31	Valid printer number.
Default setting	0	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR6002N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

Specifying a MAINPR setting is the same as changing all of the DISPLAY, PRINT, WRITE or INPUT statements' printer references from the default setting (0) to the desired printer number.

A physical printer corresponding to the report number specified must be defined to Natural as described in the *Configuration Utility* documentation, section *Device/Report Assignments*.

119

MASKCME - MASK Compatible with MOVE EDITED

This Natural profile parameter is used to control Natural's compiler.

Possible settings	ON	The range of valid year values that match the YYYY mask characters is 1582 - 2699 to make the MASK option compatible to MOVE EDITED.	
	OFF	The range of valid year values that match the YYYY mask characters is 0000 - 2699.	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	MASKCME option of COMPOPT

120

MAXBUFF - Maximum Buffer Size

This Natural profile parameter can be specified on both the client and the server side.

On the server side, it determines the size of the buffer provided by the server to receive the client request including data and to send back the result. The buffer must be large enough to hold the largest of the following two data areas for all client requests:

- the request received by the client,
- the result send back to the client.

If the size of the buffer is too small for a request, a temporary buffer with the required size is allocated and used for this request.

On the client side, it determines the size of the buffer provided for the automatic execution of Natural RPC calls. This buffer is used to build the client request including data and to receive the result from the server. The buffer must be large enough to hold the largest of the following two data areas for all requests sent by the client:

- the request send to the server,
- the result received from the server.

If the size of the buffer is too small for a request, a temporary buffer with the required size is allocated and used for this request.

For further information, see *Stubs and Automatic RPC Execution* in the *Natural Remote Procedure Call (RPC)* documentation.

The size of the data exchanged between the client and server is provided by the stub generation function of the `SYSRPC` utility. To calculate the size for automatic RPC execution, you may use the `SYSRPC CSMAS` command; see *Calculating Size Requirements* in the *SYSRPC Utility* documentation.

Possible settings	1 - 2097147	Maximum buffer size in KB.
	0	No buffer is allocated.
Default setting	0	
Dynamic specification	yes	
Specification within session	no	

Dependency on ACIVERS Settings

In case of an EntireX Broker node, special considerations apply if you are using Entire Net-Work as a transport layer. With Entire Net-Work, the receive buffer length passed to the EntireX Broker stub is restricted by the startup parameter IUBL and must not exceed 32 KB. Depending on the setting of the Natural profile parameter ACIVERS, the receive buffer length is set as follows:

- ACIVERS=1: 32000
- ACIVERS=2: 30K
- ACIVERS>2: the value specified with MAXBUFF

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

121

MAXCL - Maximum Number of Program Calls

This Natural profile parameter is used to specify the maximum number of program calls permitted between two screen I/O operations.

If the specified limit is exceeded, the Natural program is interrupted and the user is notified with an appropriate Natural error message.

Possible settings	10 - 32767	Maximum number of program calls.
	0	MAXCL=0 indicates that no limit is to be in effect.
Default setting	50	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation. * Recommended.
	USR1068N *	

122

MAXYEAR - Maximum Year for Date/Time Values

This Natural profile parameter sets the maximum value for the year part of date and time values that can be entered as constants or as terminal input.

Possible settings	2699	The maximum year that can be entered is 2699; that is, the maximum date value that can be entered is 2699-12-31.
	9999	The maximum year that can be entered is 9999; that is, the maximum date value that can be entered is 9999-12-31.
Default setting	2699	
Dynamic specification	yes	
Specification within session	no	

MAXYEAR=9999 changes the maximum date value that can be entered from 2699-12-31 to 9999-12-31.



Anmerkung: Before setting the value for MAXYEAR to "9999", you should carefully check your application for arithmetic operations or assignments of date or time values to fields that have data formats other than date or time, and perform the necessary changes. Otherwise, unexpected overflows leading to Natural errors at execution time may occur. For example, you should check for

- redefinitions of date/time fields with P6/P12 fields
- assignments of date/time values to non-date/time fields such as `P6 := D`
- arithmetic operations with date/time values where the result is assigned to a non-date/time field, for example: `P6 := D + 7`

- input of date/time fields that is used in arithmetic operations with non-date/time fields later on, for example:

```
INPUT D(D)
P6 := D + 1
```

The use of the Natural Engineer is recommended to check your application.

The setting of MAXYEAR affects

- checking of date/time constants by the compiler, for example: `P6 := D'2699-12-31'`
- INPUT statements with input or modifiable date/time fields
- MOVE EDITED statements with source or target date/time fields
- IS (D) option in logical condition criteria
- MASK option in logical condition criteria with four-digit year check (YYYY)
- VAL system function with date field as target operand

You should ensure that the MAXYEAR settings are the same for

- cataloging and executing a Natural application
- Natural RPC servers and Natural RPC clients

See also:

- *Formats D - Date, and T - Time in the Programming Guide*
- *Date and Time Constants in the Programming Guide*
- Session parameter [EM](#) in the *Parameter Reference* documentation

123 MC - Multiple-Value Field Count

Anzahl multipler Feldwerte



Anmerkung: Dieser Parameter darf nur im Reporting Mode verwendet werden.

Mit diesem Session-Parameter geben Sie an, wieviele Werte eines multiplen Feldes standardmäßig ausgegeben werden sollen, wenn das Feld ohne Index in einem `DISPLAY-` oder `WRITE-Statement` angegeben ist.

Mögliche Werte	1 - 191	Anzahl der Werte.	
Standard-Einstellung	1		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT INPUT PRINT WRITE
		Gültiges Kommando:	Keines.

Beispiel:

```
FORMAT MC=5
```


124

MFSET - Multi-Fetch Setting

This Natural profile parameter specifies whether multi-fetch is used to retrieve records from Adabas databases.

Possible settings	NEVER	Always use single-fetch.
	OFF	Use single-fetch as default. This can be overwritten on statement level.
	ON	Use multi-fetch as default. This can be overwritten on statement level.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

125

ML - Position of Message Line

Meldungszeilen-Position

Dieser Profilparameter und Session-Parameter gibt die Zeile an, die benutzt werden soll für die Anzeige von Anwendungen, die die Meldungszeilen-Position nicht explizit mittels des `SET CONTROL 'M'`-Statements setzen. Informationen zum Operanden 'M' siehe auch Natural Terminalkommando %M.



Anmerkung: Der Profilparameter ML ist in der Configuration Utility nicht vorhanden. Verwenden Sie stattdessen den Session-Parameter ML.

Mögliche Werte	B	Natural-Meldungen werden am unteren Rand des Bildschirms angezeigt.	
	T	Natural-Meldungen werden am oberen Rand des Bildschirms angezeigt.	
Standard-Einstellung	B		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET CONTROL 'M'
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

126

MP - Maximum Number of Pages of a Report

Maximale Seitenzahl eines Reports

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, wieviele Seiten ein Report höchstens erzeugen darf.

In einer Natural-Session kann die Einstellung des Profilparameters MP vom `FORMAT`-Statement verkleinert aber nicht vergrößert werden. Der mit dem Session-Parameter MP angegebene Wert gilt nur für den angegebenen Report.

Mögliche Werte	1 - 99999	Die angegebene Zahl bezieht sich auf die Anzahl der physischen Seiten und ist unabhängig von der Seitennummer der Startseite. Wird die maximale Seitenzahl überschritten, so wird das Programm mit einer Fehlermeldung abgebrochen.	
	0	Es ist keine maximale Seitenzahl definiert.	
Standard-Einstellung	0		
Dynamische Spezifikation	ja		
Spezifikation in Session	nein	Gültige Statements:	DISPLAY FORMAT PRINT WRITE
		Gültiges Kommando:	Keines.

127 MS - Manual Skip

Manuelle Cursor-Positionierung

Mit diesem Session-Parameter steuern Sie die Positionierung des Cursors bei der Verarbeitung eines INPUT-Statements.

Mögliche Werte	ON	Siehe Beispiel weiter unten. Anmerkung: Die Einstellung MS=ON wird unter BS2000/OSD nicht unterstützt.	
	OFF	Ist MS=OFF gesetzt, so wird der Cursor ins nächste Eingabefeld plaziert, sobald der Wert des aktuellen Feldes vollständig eingegeben ist.	
Standard-Einstellung	OFF		
Spezifikation in Session	ja	Gültige Statements:	FORMAT INPUT
		Gültiges Kommando:	Keines.

Beispiel:

```
INPUT (MS=ON) #A #B
```


128

MSGSF - Display System Error Messages in Short/Full

Format

This Natural profile parameter can be used to avoid truncation of Natural system error messages.

Possible settings	ON	System error messages will be displayed in full; that is, program name, line number and actual message text.
	OFF	System error messages will be displayed in short form; that is, only the actual message text will be displayed (but not the program name and line number).
Default setting	ON	
Dynamic specification	yes	
Specification within session	yes	

By default, a Natural system error message consists of the following:

- the name of the program,
- the number of the line that caused the error,
- the actual text of the message.

Depending on the size of the window in which the message is displayed, the text may be truncated. With this parameter, you can avoid such truncation.

129

NATLOG - Natural Log File

▪ Examples	300
------------------	-----

This Natural profile parameter is used to log messages that will not (or could not) be written to the standard output in interactive mode or to the output file `CMPRINT` in batch mode.

Possible settings	OFF	Disables the log mechanism.
	ERR	Logs error messages.
	INF	Logs information and success messages.
	WRN	Logs warning messages.
	ALL	Logs all types of messages.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

The location of the NATLOG file is the `TEMP` directory of Natural (specified in the local configuration file `NATURAL.INI`). If this path is unknown, Natural creates the file in the current directory.

Natural tries to create the following file name, if the `user-ID/et-ID` information could be retrieved:

`NATURAL_<user-ID>_<et-ID>.LOG`

If `user-ID` and `et-ID` could not be retrieved, then the following file name is used: `NATURAL.LOG`.

Example File Names:

A Natural batch process is running with `user-ID=SYSTEM` and `et-ID=14`, then the resulting file name is `NATURAL_SYSTEM_14.LOG`.

If the `user-ID` could not be retrieved (which is the case if an error occurs during the initialization phase of Natural), then the resulting file name is `NATURAL.LOG`.

Examples

The following example shows the contents of a Natural log file. At top of the file, there is a header with some environment information, for example the Natural Version, the parameter file currently in use and so on. Two entries follow. The first one an entry which displays the I/O channels needed for batch mode. The second entry shows an error message. Both messages are counted in the statistics summary.

Example NATLOG Output Contents for Windows

```

# #####
#           N a t u r a l   L o g   F i l e
# #####
#
# Logging started at : 06-Mar-2007 08:10:12.044
#
# Natural Version      : V v . r . s SAG 2003
# Server Type         : (none)
# Device              : BATCH (real)
# Parameter File      : NATPARM
#
# User ID             : NATURAL
# ET ID               : TEST
# Network User ID     : MYDOMAIN\NATURAL
#
# Host Name           : PCNAT
# Machine Class       : PC
# Operating System    : WNT-X86 4.0 (1381)
#
# Process ID          : 274
#
# NATLOG Option       : ALL
#
# #####
# -----
# 08:10:13.003 NATURAL      INFORMATIONAL  STATISTICS:  INF=1  WRN=0  ERR=0
# -----
# setting of parameter CMSYNIN (command file)
# D:\TEMP\syn37437.tmp
# setting of parameter CMOBJIN (input file)
#
# setting of parameter CMPRINT (output file)
# D:\TEMP\out37437.tmp
# -----
# 08:10:15.020 NATURAL      ERROR          STATISTICS:  INF=1  WRN=0  ERR=1
# -----
#           NATURAL Startup Error:  42
#           Batch mode driver error.
#           Parameter CMOBJIN not set.

```


130

NATVERS - Switching between Natural Environments

This Natural profile parameter enables you to access various Natural environments, that is, various *err*, *txt*, *bin*, *etc*, *samples* and *tmp* subdirectories.

Possible settings	any character string	Determines the Natural version.
Default setting	<i>v.r.s</i>	Anmerkung: A DEFAULT-VERSION entry in <i>SAG.INI</i> is no longer necessary.
Dynamic specification	yes	This parameter can only be specified dynamically.
Specification within session	no	

Where *v.r.s* is the current version of Natural. In the following sample programs, the characters *v.r.s* stand for the *version, release, system maintenance level*.

The NATVERS value you enter is used by Natural to determine the *SAG.INI* section you want to use with your Natural session. By default, the Natural installation creates a section in the *SAG.INI* file that points to the installed Natural environment.

The Natural installation automatically identifies the Natural version and creates the section [NATURAL=*v.r.s*] in your *SAG.INI* file that points to the installed Natural environment. This might look as in the following example:

```
[NATURAL-v.r.s]  
PROD_API=C:\Program Files\Software AG\v.r.s\Bin\natprd32.dll  
NATGUI_BMP=C:\Program Files\Software AG\Natapps\Pic  
NATDIR=C:\Program Files\Software AG  
NATINI=C:\Program Files\Software AG\v.r.s\Etc  
[NATURAL-v.r.s-END]
```

If you want to access environments other than the one that was installed automatically, create a new section for each environment as shown in the following example:

```
[NATURAL-Server v.r.s]
NATVERS=v.r.s
PROD_API=\\Server\Program Files\Software AG\v.r.s\Bin\natprd32.dll
NATGUI_BMP=\\Server\Program Files\Software AG\Natapps\Pic
NATDIR=\\Server\Program Files\Software AG
NATINI=\\Server\Program Files\Software AG\v.r.s\Etc
[NATURAL-Server v.r.s-END]
```

Replaced settings are still contained in the *SAG.INI* file and can be used by specifying them with NATVERS to switch to the corresponding Natural environments.

131

NC - Use of Natural System Commands

Verwendung von Systemkommandos

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, ob Natural-Systemkommandos während der Natural- Session verwendet werden können.

In einer Natural-Session kann der Profilparameter NC durch den Session-Parameter NC überschrieben werden.

Mögliche Werte	ON	Systemkommandos können nicht verwendet werden – außer FIN, LAST, LOGOFF, LOGON, RENUMBER, RETURN, SETUP und TECH. Ist Natural Security installiert, sind ungeachtet der Einstellung des NC-Profilparameters alle von Ihnen mit Natural Security gesetzten Systemkommando-Beschränkungen (Command Restrictions) gültig. In einer Natural Development Server-Umgebung auf Großrechnern wird der Wert OFF für den Natural Development Server angenommen, auch wenn NC=ON angegeben wurde. Wurde NC=ON client-seitig angegeben, werden clientseitig abgesetzte Systemkommandos im Anschluss daran zurückgewiesen (siehe oben).	
	OFF	Alle Systemkommandos können verwendet werden.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	



Anmerkung: Natural-Terminalkommandos und Benutzerkommandos (Objektmodul-Namen) sind vom NC-Parameter nicht betroffen.

132

NCFVERS - NCF File Protocol Version

This Natural profile parameter enables downward compatibility with Natural Versions lower than Version 6.1. It specifies the protocol version of the Entire Connection format file (.NCF) to be used. This Entire Connection format is generated when work files of type ENTIRE CONNECTION or DEFAULT work files with the file extension .NCD are written.

Possible settings	0	A format file of Entire Connection Protocol Version 0 is written. The format files created are compatible with those of Natural Versions lower than Version 6.1.
	2	A format file of Entire Connection Protocol Version 2 is written, which is created by Natural Version 6.1 or 6.2.
	3	A format file of Entire Connection Protocol Version 3 is written, which is created by Natural Version 6.3.
Default setting	3	
Dynamic specification	no	
Specification within session	no	

For information on the work file types ENTIRE CONNECTION and the Entire Connection format, refer to ENTIRECONNECTION in the DEFINE WORKFILE statement documentation and *Work File Formats* in the *Operations* documentation.

133

NL - Numeric Length for Output

Numerische Länge der Ausgabe

Mit diesem Session-Parameter bestimmen Sie die Standard-Eingabe-/Ausgabelänge eines numerischen Feldes, das in einem `DISPLAY-`, `INPUT-`, `PRINT-` oder `WRITE-`Statement verwendet wird.

Der Parameter `NL` darf nicht für Gruppen angegeben werden.

Eine für ein Feld definierte Editiermaske setzt den `NL`-Parameter für dieses Feld außer Kraft.

Mögliche Werte	<i>nn.m</i>	<p>Die Länge wird in der Form <i>nn.m</i> angegeben, wobei <i>nn</i> für die Stellen vor dem Dezimaltrennzeichen (Punkt oder Komma) und <i>m</i> für die Stellen nach dem Dezimaltrennzeichen steht.</p> <p>Die Angabe von Stellen nach dem Dezimaltrennzeichen ist nicht erforderlich. <i>m</i> darf nicht größer als 7 sein. Insgesamt darf die Länge von <i>nn</i> und <i>m</i> zusammen 29 Stellen nicht überschreiten.</p> <p>Anmerkung:</p> <ol style="list-style-type: none"> 1. Ist die <code>NL</code>-Länge kleiner als die Feldlänge, werden die ausgegebenen Werte entsprechend abgeschnitten, ohne dass dies zu einer Fehlermeldung führt. 2. Ist die <code>NL</code>-Länge größer als die Feldlänge, werden die freien Stellen mit Leerzeichen aufgefüllt, und es führt zu keinem Fehler, wenn ein Eingabefeld abgeschnitten wird. 	
Standard-Einstellung	Keine		
Spezifikation in Session	ja	Gültige Statements:	<code>DISPLAY</code> <code>FORMAT</code> <code>INPUT</code> <code>PRINT</code> <code>WRITE</code>
		Gültiges Kommando:	Keines

Beispiel:

```
DISPLAY #AA(NL=20) #AB(NL=3.2)
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

134

NOAPPLERR - Suppress Message Number Prefix NAT

This Natural profile parameter is used to suppress the message number prefix „NAT“ with user-supplied error messages.

Possible settings	ON	The prefix „NAT“ is not displayed in error messages.
	OFF	The prefix „NAT“ is displayed in error messages.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

Überschreiben geschützter Felder durch Help routines

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, ob der Inhalt eines schreibgeschützten Feldes (Attribut `AD=P`) durch eine Help routine, die dem Feld zugewiesen ist, überschrieben werden kann.

In einer Natural-Session kann der Profilparameter `OPF` durch den Session-Parameter `OPF` überschrieben werden.

Mögliche Werte	ON	Eine einem Feld zugewiesene Help routine kann den Inhalt des Feldes Werte überschreiben, selbst wenn das Feld schreibgeschützt ist.	
	OFF	Die Inhalte schreibgeschützter Felder können nicht durch Help routines überschrieben werden.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	



Anmerkungen:

1. Der Profilparameter `OPF` gilt nur für das Feld, für das eine Help routine aufgerufen wird; er gilt nicht für Parameter, die explizit an die Help routine übergeben werden. Das bedeutet, dass der `OPF`-Parameter wirkungslos bleibt, falls Sie das Feld, für das Hilfe aufgerufen wird, auch noch explizit als an die Help routine zu übergebenden Parameter angegeben haben.
2. Des weiteren können Sie im Reporting Mode die `OPF`-Einstellung mittels des Statements `SET GLOBALS` ändern.

136

OPRB - Database Open/Close Processing

- OPRB String Syntax 317

This Natural profile parameter controls the use of the Adabas C `Open/Close` commands during a Natural session.

Possible settings	OFF	If the OPRB parameter is set to OFF, a Natural session starts with an Adabas OP command requesting UPD (access/update) to the Natural system file. Natural also issues RELEASE CID (Adabas RC) commands to release all ISN lists (ISN lists specified in a RETAIN clause of a Natural FIND statement are not released).
	OPRB=(<i>string</i>)	You can specify an Open request in accordance with the syntax described below. See also the examples listed.
Default setting	OFF	
Dynamic specification	no	
Specification within session	no	

It is required if any of the following conditions are true:

- An explicit list of Adabas files to be accessed/updated is to be provided. This is necessary, for example, if Adabas cluster updating or exclusive file control is to be requested.
- The Adabas record buffer to be used with the initial Adabas OP command can be explicitly provided. To access databases you have to specify the DBIDs and file numbers with their corresponding access rights at the OPRB string.
- The character set for Adabas format W is to be provided. To access databases you have to specify the DBIDs and the required encoding name.

Entries may not contain blanks, must be enclosed in parenthesis and must follow the rules defined in the relevant Adabas documentation.

In addition to the Adabas syntax, internal file numbers can be specified by using the *x-y* notation (that is, all numbers between *x* and *y*).

OPRB String Syntax

DBID=(*x*) specifies the database for the following access right entries:

ACC=(<i>file-list</i>)	Specifies access permission (read) for the files in the file list.
UPD=(<i>file-list</i>)	Specifies update permission (read/write) for the files in the file list.
EXU=(<i>file-list</i>)	Specifies exclusive update permission (exclusive read/write) for the files in the file list.
WCODE= <i>encoding</i>	Specifies the encoding for W fields in the Adabas user session (Adabas for Mainframes). Required encoding code for Adabas on Mainframes is 4095.
WCHARSET= <i>charset</i>	Specifies the default character set used for W fields in record and value buffers in the Adabas user session (Adabas for UNIX, OpenVMS and Windows). Required encoding names for Adabas on UNIX, OpenVMS and Windows are UTF-16LE or UTF-16BE.

The trailing record buffer dot (.) can be omitted in the OPRB string because it is appended automatically.

DBID=0 specifies the default record buffer entry and can be omitted if it is the first DBID listed in the OPRB string. This default record buffer is taken if there is no specific entry for the requested database.

Combinations of the keywords ACC, UPD, EXU and WCODE or WCHARSET must follow the rules as defined in the relevant Adabas documentation. Natural issues an OP command at the start of a Natural session and a CL command at the end of the session. At the end of a Natural program, only the required RC commands are issued to release held ISN lists.

Example 1:

```
(ACC=2,3,4,DBID=15,UPD=3,4,ACC=5)
```

The following entries were defined:

```
'UPD=3,4,ACC=5.' for DB 15  
'ACC=2,3,4.' for other databases (DB 0)
```

Example 2:

```
(DBID=15,ACC=2-7)
```

The following entry was defined:

```
'ACC=2,3,4,5,6,7.' for DB 15; access to other databases is not permitted.
```

Example 3:

```
(DBID=0,ACC=2,3,4,5.)
```

The following entry was defined:

```
'ACC=2,3,4,5.' for all databases (DB 0).
```



Anmerkung: If you have Natural Security installed, open/close processing works the same way as without Natural Security; the OPRB parameter in the security profile is provided for future use only.

Example 4:

```
(DBID=0,ACC=2,3,4,5,DBID=12,WCHARSET='UTF-16LE',UPD=3-10)
```

The following entry was defined:

```
'ACC=2,3,4,5.' for all databases (DB 0).
```

```
WCHARSET='UTF-16LE',UPD=3,4,5,6,7,8,9,10. for DB 12.
```


137

PARM - Alternative Parameter File

This Natural profile parameter can be used for Natural startup (Studio/Runtime/Server) in order to specify an alternative parameter file Natural is to run with. If no PARM parameter is specified, Natural will start using the default NATPARM parameter file containing all default settings for each single parameter. Use the Natural Configuration Utility in order to create your own new parameter file. See *Creating a New Parameter File* in the *Configuration Utility* documentation.

Possible settings	1-8 characters	Any valid file name.
Default setting	none	
Dynamic specification	yes	This parameter can only be specified dynamically.
Specification within session	no	

138

PC - Periodic Group Count

Anzahl der Periodengruppen-Ausprägungen

Dieser Session-Parameter kann nur im Reporting Mode verwendet werden. Er dient zur Festlegung der Anzahl der Periodengruppen-Ausprägungen, die standardmäßig ausgegeben werden, wenn eine Periodengruppe (oder ein in einer Periodengruppe enthaltenes Feld) in einem DISPLAY- oder WRITE-Statement ohne Index angegeben wird.

Mögliche Werte	1 - 191	Anzahl der Werte.
Standard-Einstellung	1	
Gültige Statements	FORMAT	
	INPUT DISPLAY WRITE PRINT	Parameter kann auf Statement- und/oder Elementebene (Feldebene) angegeben werden.
Gültiges Kommando:	Keines	

Beispiel:

```
FORMAT PC=5
```


139

PCHECK - Parameter Checking for Object Calling

Statements

This Natural profile parameter is used to control Natural's compiler.

Possible settings	<p>ON</p> <p>The compiler checks the number, format, length and array index bounds of the parameters that are specified in an object calling statement, such as <code>CALLNAT</code>, <code>PERFORM</code>, <code>INPUT USING MAP</code>, <code>PROCESS PAGE USING</code>, <code>OPEN DIALOG</code>, <code>SEND EVENT</code>, helproutine calls. Also, the OPTIONAL feature of the <code>DEFINE DATA PARAMETER</code> statement is considered in the parameter check.</p> <p>The parameter check is based on a comparison of the parameters of the calling statement with the <code>DEFINE DATA PARAMETER</code> definitions for the object to be called.</p> <p>It requires that</p> <ul style="list-style-type: none">■ the name of the subprogram to be invoked is defined as an alphanumeric constant (not as an alphanumeric variable),■ the subprogram to be invoked is available as a cataloged object. <p>Otherwise, <code>PCHECK=ON</code> will have no effect.</p> <p>Problems in Using the CATAL Command with PCHECK=ON</p> <p>When a <code>CATAL</code> command is used in conjunction with <code>PCHECK=ON</code>, you should consider the following:</p> <p>If a <code>CATAL</code> process is invoked, the order in which the programming objects are compiled depends primarily on the type of the object and secondarily on the alphabetical name of the object. The object type sequence used is: <code>DDMs</code>, <code>GDA</code>s, <code>LDA</code>s, <code>PDA</code>s, classes, maps, dialogs, external subroutines, subprograms, functions, adapters, helproutines, programs. Within objects of the same type, the alphabetical order of the name determines the sequence in which they are cataloged.</p>
--------------------------	---

		<p>As mentioned above, the parameters of the object calling statement are checked against the compiled form of the called object. If the calling object (the one which is being compiled and includes the object calling statement) is cataloged before the invoked object, the PCHECK result may be wrong if the parameters in the object calling statement and in the called object were changed. In this case, the new object image of the called object has not yet been produced by the CATALL command.</p> <p>This causes the <i>new</i> parameter layout in the object calling statement to be compared with the <i>old</i> parameter layout of the DEFINE DATA PARAMETER statement of the called subprogram.</p> <p>Solution:</p> <ul style="list-style-type: none"> ■ Set compiler option PCHECK to OFF. ■ Perform a general compile with CATALL on the complete library, or if just one or a few objects were changed, perform a separate compile on these objects. ■ Set compiler option PCHECK=ON. ■ On the complete library, perform a general compile with the function CHECK. of system command CATALL enabled. 	
	OFF	No parameter check is performed.	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	PCHECK option of COMPOPT

140

PERSIST - Flag a Parameter File as Persistent



Vorsicht: This Natural profile parameter is intended for easier handling of parameter settings during development. Its use should be restricted to parameter files which are used by a single person or for a specific task. Multiple concurrent accesses to the same persistent parameter file might cause unexpected interferences.

PERSIST can only be set in the Configuration Utility.

Possible settings	ON	Setting the PERSIST parameter to "ON" marks the parameter file as persistent. This means that all settings which are changed during a Natural Studio session are saved to the currently used parameter file when the session is ended. The next time Natural Studio is started with that parameter file, it will resume with the same parameter settings. The saved settings include those which were made at session startup (dynamic parameters issued at the operating system command prompt when Natural was invoked) and those which were changed during the session, e.g. with the Natural system commands GLOBALS, COMPOPT.
	OFF	The parameter file is not flagged as persistent.
Default setting	OFF	
Dynamic specification	no	
Specification within session	no	

141

PM - Print Mode

- Profilparameter PM 328
- Session-Parameter PM 328

Druck-Modus

Folgende Themen werden behandelt:

Profilparameter PM

Mit diesem Profilparameter bestimmen Sie, wie Felder gedruckt oder angezeigt werden sollen.

PM=I betrifft alle vom System gesteuerten Ausgabeschirmbestandteile, das heisst, Systemvariablen und PF-Tastenzeilen. Darüber hinaus sind von dieser Einstellung alle nicht-alphanumerischen Felder, z.B. numerische Felder und Datumsfelder, betroffen. Außerdem wird bei Natural Web I/O Interface-Terminals die Anzeigerichtung in den Feldern von links nach rechts in rechts nach links geändert. Die Routine für die Feldinvertierung wird in der Natural-Source-Library als Assembler-Modul NATPM ausgeliefert und kann bei Bedarf geändert werden.

Ausführliche Informationen über die Verwendung der Einstellung PM=I siehe *Bidirectional Language Support* in der *Unicode and Code Page Support*-Dokumentation.

Mögliche Werte	I	Die Standardbildschirmrichtung beim Ausführen von Programmen ist von rechts nach links. Das RTL-Attribut für neue Dialoge wird geprüft.
	R	Die Standardbildschirmrichtung beim Ausführen von Programmen ist von links nach rechts. Das RTL-Attribut für neue Dialoge wird nicht geprüft.
Standard-Einstellung	R	
Dynamische Spezifikation	ja	
Spezifikation in Session	ja	
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.

Session-Parameter PM

Mit diesem Session-Parameter bestimmen Sie, wie Felder angezeigt werden sollen.

Mögliche Werte	PM=I	Umgekehrte Feldrichtung.
	PM=N	Das Feld wird bei der Hardcopy-Ausgabe ignoriert (d.h. nicht gedruckt).
Standard-Einstellung	Keine	Die Standardfeldrichtung wird verwendet und wird bei der Hardcopy-Ausgabe berücksichtigt.
Gültige Statements	COMPRESS DEFINE DATA DISPLAY FORMAT INPUT MOVE PRINT WRITE	

Ausführliche Informationen zur Benutzung von PM=I finden Sie unter *Bidirectional Language Support* in der *Unicode and Code Page Support*-Dokumentation.

142 PRGPAR - Data to be Passed to Program Receiving

Control at Termination

This Natural profile parameter specifies data to be passed to the program that receives control when Natural terminates (see also profile parameter [PROGRAM](#)).

Possible settings	any valid character string	Data to be passed.
Default setting	blank	No data is passed.
Dynamic specification	no	
Specification within session	no	

143 PROGRAM - Non-Natural Program Receiving Control after Termination

This Natural profile parameter specifies the non-Natural program which is to receive control after the termination of the Natural session.

Possible settings	1-12 characters	Non-Natural program
Default setting	none	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR6204N (for all platforms)	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

Data for the program specified with the profile parameter PROGRAM can be supplied with the TERMINATE statement.

144

PS - Page Size for Natural Reports

Länge einer Reportseite

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, wieviele Zeilen eine von einem DISPLAY- oder WRITE-Statement erzeugte Reportseite höchstens haben darf.

Wenn er als Profilparameter benutzt wird, kommt der PS-Parameter im Batch-Betrieb zum Einsatz und definiert die physische Seitenlänge. Im Online-Betrieb wird die physische Seitenlänge stets auf die physische Bildschirmhöhe gesetzt.

Mögliche Werte	1 - 250	Maximale Anzahl der Zeilen pro Seite.	
	0	<p>Die physische Seitenlänge soll benutzt werden.</p> <p>Wenn PS=0 für den ersten auszugebenden Report (Report 0) angegeben wird, wird die physische Seitenlänge des verwendeten Geräts minus 1 verwendet.</p> <p>Wenn PS=0 für die Reports 1 – 31 angegeben wird, führt dies dazu, dass die automatische Verarbeitung einer neuen Seite (Newpage) unterdrückt wird, d.h. es wird keine automatische Seitenvorschub-Verarbeitung ausgeführt.</p>	
Standard-Einstellung	0		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT INPUT SET GLOBALS WRITE
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Siehe auch *Seitengröße – der PS-Parameter im Leitfaden zur Programmierung*.

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des Library- Profils überschrieben werden.

145 PSIGNF - Internal Representation of Positive Sign of Packed Numbers

This Natural profile parameter can be used to define the internal representation of the positive sign of packed numbers.

Possible settings	ON	The positive sign of a packed number is represented internally as H ' F ' .	
	OFF	The positive sign of a packed number is represented internally as H ' C ' .	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	PSIGNF option of COMPOPT

146

RCFIND - Handling of Response Code 113 for FIND

Statement

This Natural profile parameter specifies the action to be taken if Adabas Response Code 113 (requested ISN not found) is returned during the execution of a `FIND` statement processing loop.

Possible settings	ON	Response Code 113 causes the program to be terminated.
	OFF	Response Code 113 will be ignored, and processing of the <code>FIND</code> loop will continue by reading the next record.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

147 RCGET - Handling of Response Code 113 for GET

Statement

This Natural profile parameter specifies the action to be taken if Adabas Response Code 113 (requested ISN not found) is returned during the execution of a GET statement.

Possible settings	ON	Response Code 113 causes the program to be terminated.
	OFF	Response Code 113 will be ignored, the system variable *ISN will be set to "0", and processing will continue.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

148

RDACTIVE - Activate Remote Debugger

This Natural profile parameter specifies whether a remote debugger on a Windows computer is to be used or not. `RDACTIVE` is only relevant if the system command `DEBUG` has been used. It is used in combination with the profile parameters `RDNODE` (where the relevant node name is specified) and `RDPORT` (where the port number is specified).

Possible settings	ON	Remote debugging is enabled. For DCOM (Windows only) or RPC servers, the remote debugging session is opened automatically.
	OFF	Remote debugging is not possible. It is possible, however, to use the debugger integrated in Natural Studio.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	



Anmerkung: See the *Debugger* documentation for further details.

149

RDNODE - Remote Debugger Node Name

This Natural profile parameter specifies the node name of the Windows computer on which a remote debugger is to be called.

Possible settings	Character string	Node name.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	



Anmerkung: See the *Debugger* documentation for further details.

150

RDPORT - Remote Debugger Port

This Natural profile parameter specifies the port number on which a remote debugger on a Windows computer has been installed.

Possible settings	0 - 9999	Port number. By default, the debugger service is installed on the Windows computer's Port 2600. You can therefore leave the UNIX computer's RDPORT setting at the default. If, however, Port 2600 on Windows has been reserved for another service, and a different port number has been specified, you must change RDPORT accordingly.
Default setting	2600	
Dynamic specification	yes	
Specification within session	no	



Anmerkung: See the *Debugger* documentation for further details.

151

RDS - Define Remote Directory Server

This Natural profile parameter allows you to define up to 10 remote directory servers. For each remote directory server, you specify up to 5 subparameters.

RDS is specified on the client side only.

Possible settings	<i>server-name</i>	The server name (1 - 8 characters). There is no default, the value must be specified.		
	<i>server-node</i>	The server node (1 - 8 characters). There is no default, the value must be specified.		
	<i>subprogram</i>	The name of the subprogram titled CALLNAT, (1 - 8 characters) to be used as interface (default is RDSSCDIR).		
	<i>logon-indicator</i>	A logon indicator. If nothing is specified, blank is the default.		
		L	The client initiates a Natural logon to the server with the library name of the current library on the client.	
		blank	No server logon will be executed.	
	Note for Windows platforms: instead of "L", check the selection box.			
	<i>transport-protocol-name</i>	The transport protocol to be used. "ACI" is the only possible value and the default.		
Default setting	none	Subparameter defaults, see above.		
Dynamic specification	yes	See below.		
Specification within session	no			

For dynamic specification the syntax is as follows.

Using 1 server:

```
RDS=(server-name,server-node-name,subprogram,logon-indicator,transport-protocol-name)
```

Using 2-10 servers:

```
RDS=((server-name,server-node  
name,subprogram,logon-indicator,transport-protocol-name)(server-name,server-node  
name,subprogram,logon-indicator,transport-protocol-name)...(server-name,server-node  
name,subprogram,logon-indicator,transport-protocol-name))
```

For additional information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

152

RECAT - Dynamic Recataloging

This Natural profile parameter specifies the action to be taken if Natural detects an inconsistency in the global data area definition as defined in the program currently being executed; that is, the global data area in the program does not correspond to the definition of the global data area currently in use.

Possible settings	ON	<p>Wichtig: This profile parameter only applies to Natural objects of Version 2.3 and above.</p> <p>Natural issues an error message and disables the system commands CATALOG, PURGE and SAVE.</p> <p>If the RECAT parameter has been set to "ON", and an object exists in both source and cataloged form, then the source and the cataloged object cannot be processed independently. In order to ensure consistency between the source and the cataloged object, Natural disables the system command CATALOG (also when invoked via CATALL). In addition, the system commands PURGE and SAVE are disabled for a source for which a corresponding cataloged object exists.</p> <p>Anmerkung: Only objects satisfying the criteria for a particular command (such as PURGE) will be displayed in the corresponding selection box.</p>
	OFF	Natural issues an error message.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See SYSEXT - <i>Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

153

REINP - Issue Internal REINPUT Statement for Invalid

Data

Interner REINPUT bei ungültigen Daten

Dieser Natural Profil- und Session-Parameter kann benutzt werden, um einen internen REINPUT für ungültige Daten zu verhindern.

Standardmäßig führt Natural automatisch ein internes REINPUT-Statement aus, wenn auf ein INPUT-Statement hin ungültige Daten eingegeben werden. Mit diesem Session-Parameter können Sie diesen Automatismus ausschalten. Dadurch haben Sie die Möglichkeit, solche Eingabefehler in Ihrer Anwendung selbst zu verarbeiten.

In einer Natural-Session kann der Profilparameter REINP durch den Session-Parameter REINP überschrieben werden.

Mögliche Werte	ON	Bei Eingabe ungültiger Daten wird ein internes REINPUT-Statement ausgeführt.	
	OFF	Bei Eingabe ungültiger Daten wird kein internes REINPUT-Statement ausgeführt.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

154

ROSY - Read-Only Access to System Files

This Natural profile parameter disables modifications on the Natural system files [FDDM](#), [FNAT](#), [FUSER](#), [FDIC*](#) and [FSEC*](#).

* Not supported on this platform.

Possible settings	ON	No data can be written to, modified on or deleted from the system files. Natural issues an error message instead of performing any action that would modify any of these system files.
	OFF	Data can be written to, modified on and deleted from the system files.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

155

RPCSDIR - Library for Service Directory

This Natural profile parameter specifies the name of the Natural library (or one of its steplibs) used by the client at runtime. This parameter is evaluated by the `SYSRPC` utility functions Service Directory Maintenance and Server Command Execution.

`RPCSDIR` is specified on the client side only.

Possible settings	1 - 8 characters	Valid Natural library name.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

For further information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

156

RTINT - Allow Runtime Interrupt

This Natural profile parameter determines whether it should be allowed to interrupt a running Natural application that does not respond anymore by using the interrupt key combination of the operating system (CTRL+BREAK).

Possible settings	ON	Interrupts are allowed.
	OFF	Interrupts are not allowed.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

For further information, see *Interrupting a Running Natural Application* in the *Programming Guide*.

157

SA - Sound Terminal Alarm

Terminal-Warnton

Dieser Natural Profil- und Session-Parameter gibt an, ob die Terminal-Alarmfunktion benutzt werden soll.

In einer Natural-Session kann der Profilparameter SA durch den Session-Parameter SA überschrieben werden.

Mögliche Werte	ON	Der Warnton wird jedesmal ausgelöst, wenn der Benutzer von Natural zu einer Eingabe aufgefordert wird..	
	OFF	Der Warnton wird zur Eingabeaufforderung nicht ausgelöst. Allerdings kann der Warnton nach wie vor über die SOUND ALARM-Klausel eines REINPUT-Statements ausgelöst werden.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

158

SB - Selection Box

- Anmerkungen zur Syntax 364
- Anmerkungen zur Laufzeitumgebung 365

Auswahlfeld

Auswahlfelder stehen in einem INPUT-Statement ausschließlich auf Großrechnern zur Verfügung. Für andere Plattformen können Auswahlfelder nur im Map-Editor definiert werden.

Auswahlfelder können an Eingabefelder angehängt werden. Sie bilden eine komfortable Alternative zu an Feldern angehängte Helprountinen, da Sie ein Auswahlfeld unmittelbar in Ihrem Programm kodieren können. Sie brauchen kein zusätzliches Programm wie bei Helprountinen.

Sie können eine Auswahlfeld-Klausel für jede INPUT-Variable vom Typ Alpha definieren, ungeachtet der Tatsache, ob dieses Feld ein Eingabe- oder Ausgabefeld oder beides ist.

Hierzu sieht die Syntax wie folgt aus:

```
SB=operand1 [,operand1]...
```

wobei *operand1* einen Wert-Operanden darstellt, der zum Auffüllen des Auswahlfeldes mit Elementen benutzt wird.

Operand	Mögliche Struktur	Mögliche Formate	Referenzierung erlaubt	Dynam. Definition
<i>operand1</i>	C S A	A	yes	no

Bei SB geben Sie die anzuzeigenden Werte innerhalb des Auswahlfeldes ein.

Um ein Auswahlfeld einem Feld zuzuweisen

geben Sie das Attribut SB für ein INPUT-Alphafeld in Ihrem Natural-Programm mittels der folgenden Beispiel-Syntax ein:

```
INPUT #FLD (SB='value1', #ITEM1, #ITEM2(1:3), #ITEM3(*))
```

Folgende Themen werden behandelt:

Anmerkungen zur Syntax

Es ist möglich, einem Feld sowohl ein Auswahlfeld als auch eine Helprountine zuzuweisen.

Auswahlfelder können für jedes Variablenfeld in einem INPUT-Statement definiert werden. Dazu gibt es folgende Ausnahmen:

Systemvariablen	Zum Beispiel: *PROGRAM, *COM
Benannte Konstanten (nur auf Großrechnern)	Definiert mit einer CONST-Klausel in DEFINE DATA-Statement.

Außer dem SB-Attribut können auch andere Attribute definiert werden, z.B. AD oder CD.

Das Auswahlfeld muss nicht änderbar sein, wie dies bei AD=A oder AD=M der Fall ist. Mit anderen Worten ist es möglich, auch für ein schreibgeschütztes Ausgabefeld wie AD=0 ein Auswahlfeld zur Verfügung zu stellen (und Werte auszuwählen). Wenn Sie AD=0 benutzen, ist der Benutzer dazu gezwungen, aus einer Reihe von vordefinierten Werten zu wählen, die ihrerseits in einem Auswahlfeld erscheinen.

Anmerkungen zur Laufzeitumgebung

Auswahlfeld-Position

Wenn ein ein Auswahlfeld enthaltendes Programm ausgeführt wird, wird das Auswahlfeld nach demselben Positionierungsalgorithmus auf dem Bildschirm positioniert, der für Hilfefenster verwendet wird, d.h. die Länge und Position des Auswahlfeldes wird automatisch festgelegt, *in der Nähe* des Feldes.

Auswahlfeld-Attribute

Die Farbe und die intensivierten Attribute, die dem Feld zugewiesen sind, werden auch für die in dem entsprechenden Auswahlfeld angezeigten Werte benutzt.

Editiermasken in Auswahlfeldern

Wenn für das Feld eine Editiermaske definiert worden ist, wird die Editiermaske für alle Auswahlfeld-Werte verwendet.

Um eine Editiermaske für ein Feld zu definieren, benutzen Sie das `INPUT`-Statement, damit Sie eine Editiermaske für ein Feld definieren können. Dies ist in dem folgenden Code-Beispiel veranschaulicht.

```
DEFINE DATA  
LOCAL  
1 A(A4)  
END-DEFINE  
MOVE 'ABCD' TO A  
*  
SET KEY PF1 = HELP  
FORMAT KD=ON  
*  
INPUT A (AD=M EM=X.X.X.X SB='1234','WXYZ')  
WRITE A  
END
```

Zeilenlänge für Auswahlfelder

Die Zeilenlänge des Auswahlfeldes stimmt mit der Feldlänge überein, der das Auswahlfeld entspricht.

Wenn ein für das Auswahlfeld vorgesehener Wert die Zeilenlänge des Auswahlfeldes überschreitet, wird der Wert abgeschnitten.

Reihenfolge der Auswahlfeld-Werte

Auswahlfeld-Werte werden in der Reihenfolge angezeigt, in der sie im `SB`-Attribut erscheinen.

159

SD - Time Delay between Two Screens

This Natural profile parameter can be used to delay the time related to screen output display. This is the time delay between two screens during a non-conversational write operation (see the Natural terminal command %N).

Possible settings	0 - 100	The unit for the specified setting is a tenth of a second, which means that SD=10 leads to a delay of one second.
Default setting	0	
Dynamic specification	no	
Specification within session	no	

160

SERVER - Start Natural Session as an RPC Server

Session

This Natural profile parameter specifies whether or not the Natural session will be started as an RPC server session.

SERVER can be specified on both the client and the server side.

Possible settings	ON	The Natural session will be started as an RPC server session.
	OFF	The Natural session will not be started as an RPC server session.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

161 SF - Spacing Factor

Spaltenabstand

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, wieviele Leerstellen zwischen zwei Ausgabespalten eines mit einem `DISPLAY`-Statement erzeugten Natural- Reports standardmäßig eingefügt werden sollen.

In einer Natural-Session kann der Profilparameter `SF` durch den Session-Parameter `SF` überschrieben werden.

Mögliche Werte	1 - 30	Anzahl der Leerstellen. Anmerkung: Der <code>SF</code> -Parameter kann nicht auf Null (0) gesetzt werden, d.h. es muss wenigstens ein Leerzeichen zwischen Report-Spalten stehen.	
Standard-Einstellung	1		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Unter Natural Security: Die Einstellung dieses Parameters kann durch die *Session Parameters*-Option des Library-Profiles überschrieben werden.

Siehe auch *Column Spacing - SF Parameter and nX Notation* im Leitfaden zur Programmierung.

162 SG - Sign Position

Vorzeichen-Stelle

Mit diesem Session-Parameter bestimmen Sie, ob einem numerischen Feld eine zusätzliche Stelle zur Anzeige des Vorzeichens vorangestellt werden soll.

Wenn der EM-Parameter (Editiermodus) angegeben wird, überschreibt er den SG-Parameter.

Mögliche Werte	ON	Eine Vorzeichen-Stelle wird belegt.	
	OFF	Keine Vorzeichen-Stelle wird belegt. Wenn SG=OFF gesetzt ist, werden negative Werte ohne das Minuszeichen (-) ausgegeben. SG=OFF verhindert nicht die Eingabe von negativen Werten in Eingabefeldern.	
Standard-Einstellung	ON		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT INPUT PRINT WRITE
		Gültiges Kommando:	Keines

Beispiel:

```
FORMAT SG=OFF
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

163

SM - Programming in Structured Mode

Programmierung im Structured Mode

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, ob im Structured Mode programmiert werden muss oder nicht.

In einer Natural-Session kann der Profilparameter `SM=OFF` durch den Session-Parameter `SM=ON` überschrieben werden.

Mögliche Werte	ON	Die Programmierung darf nur in Structured Mode-Syntax erfolgen. Anmerkung: Falls mit dem Profilparameter <code>SM</code> festgelegt wird, dass im Structured Mode programmiert werden muss <code>SM=ON</code> , wird ein Versuch, diese Einstellung während der Session per Systemkommando <code>GLOBALS</code> und Session-Parameter <code>SM</code> zu ändern mit einer Fehlermeldung zurückgewiesen (<code>Reporting mode not permitted</code>).	
	OFF	Die Programmierung kann entweder in Structured Mode oder in Reporting Mode erfolgen.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	Keine
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe <i>SYSEXT - Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Falls Natural Security installiert ist:

- legt die Einstellung der Modus-Option im Security-Profil der Library fest, ob der Profilparameter `SM` benutzt werden kann (siehe auch *Programming mode* in der *Natural Security*-Dokumentation;
- kann es sein, dass dieser Parameter im Security-Profil einer Library deaktiviert ist; ist dies der Fall, so gilt für die Library unabänderlich Structured Mode.

164

SNAT - Sound a Bell at Syntax Error

This Natural profile parameter is used to sound a bell when the compiler detects a syntax error in a Natural program.

Possible settings	ON	A bell will sound when a syntax error is encountered.
	OFF	No bell will sound in the case of syntax errors.
Default setting	OFF	
Dynamic specification	no	
Specification within session	no	

165

SORTSIZE - Size of Sort Buffer

This Natural profile parameter specifies the amount of storage to be reserved for use by the SORT program. This sort buffer is only allocated when executing a Natural program which contains a SORT statement.

Possible settings	500 - 2048	Buffer size in KB. Increasing the setting leads to faster SORT processing, in particular when all data to be sorted fit into the sort buffer.
Default setting	500	
Dynamic specification	yes	
Specification within session	no	

166 SPODDEBUGPORT - Debugger Port for Debugging in the Context of SPoD

This Natural profile parameter is only relevant if you intend to use the old Natural debugger in conjunction with an earlier version (NDV2.1) of the Natural Development Server.

It specifies the port number which will be used when debugging applications in a mapped environment.

Possible settings	0 - 65535	Port Number. By default the port number is set to "0". Natural Studio then evaluates a free port on the PC. The evaluated port will be used as the communication channel between the Natural Studio Debugger and the remote application. The host where the Natural Development Server (NDV) is running, must be prepared to work with any port. If a value not equal to zero is specified, Natural Studio will use that specified port as the debug port for the communication channel between the Natural Studio Debugger and the remote application. Use this option, if only a limited number of ports is allowed on the host where the NDV server is running.
Default setting	0	
Dynamic specification	yes	
Specification within session	no	

167

SRETAIN - Retain Source Format

This Natural profile parameter specifies that all existing sources have to be saved in their original encoding format.

When new sources are created, they will be saved either in the current code page format or in UTF-8 format, depending on the setting of profile parameter `SUTF8`. This is independent of the setting of `SRETAIN`.

If a source can not be saved in the target code page format, because this code page does not define all characters contained in the source, a message is displayed which allows the user to choose whether he/she wants to remove the problematic characters or cancel the save process.

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	ON	The original code page of an existing Natural source is retained. If the profile parameter <code>SUTF8</code> is defined as well, new sources will be saved in UTF-8 format.
	OFF	For existing Natural sources with format UTF-8 the encoding will not be changed. Existing sources with other encodings will be saved using the current code page.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

168

SRVCMIT - Server Commit Time

This Natural profile parameter specifies the time at which a Natural RPC server automatically commits an RPC conversation or a non-conversational RPC request. This parameter is only evaluated if the profile parameter `ETEOP` is set to `ON`.

SRVCMIT is specified on the server side only.

Possible settings	B	The Natural RPC server automatically commits a database transaction before the reply is sent to the client. If the reply fails, the database transaction is already committed.
	A	The Natural RPC server automatically commits a database transaction after the reply has been successfully sent to the client. If the reply fails, the database transaction is rolled back.
Default setting	B	
Dynamic specification	yes	
Specification within session	no	

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

169

SRVNAME - Name of RPC Server

This Natural profile parameter specifies the name of the RPC server, with which it registers on the node specified with the profile parameter [SRVNODE](#).

SRVNAME is specified on the server side only.

Possible settings	1 - 192 characters	Valid server name.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

You may either specify a physical server name of up to 32 characters or a logical service name of up to 192 characters. In case of a logical service name, an asterisk (*) must be specified with the [SRVNODE](#) parameter(intentionally left empty).

In case of an EntireX Broker node, the value of SRVNAME corresponds to the value of the SERVER attribute of a service entry in the broker attribute file, as below:

```
CLASS=RPC, SERVICE=CALLNAT, SERVER=srvname
```

Example:

```
SRVNAME='PRODUCTION_SERVER'           /* physical server name */  
SRVNAME='MY_LOGICAL_SERVICE,MY_SET'    /* logical server name */
```

For more details about Location Transparency and logical service names, refer to the EntireX documentation.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

170 SRVNODE - Name of Node

This Natural profile parameter specifies the name of the node upon which an RPC server registers.

SRVNODE is specified on the server side only.

Possible settings	1 - 192 characters	Node name.
Default setting	none	
Dynamic specification	yes	
Specification within session	no	

You may either specify a physical node name of up to 32 characters, a logical node name of up to 192 characters or an asterisk (*) (intentionally left empty) to indicate that `SRVNAME` contains a logical service name.

In case of an EntireX Broker node, a physical node name may refer to an Entire Net-Work node or to a TCP/IP address. Note that the broker stub in use must support the naming notation. For details about the structure of node names and their support by the broker stubs, refer to the EntireX documentation.

The examples below are based on the EntireX notation.

```
SRVNODE=ETB001 /* Entire Net-Work node */
SRVNODE=PCBROKER /* host name for a TCP/IP address */
SRVNODE='157.189.160.95:1958:TCP' /* TCP/IP address with port number */
SRVNODE='tcpip://host.com:1958' /* host name for a TCP/IP address
with port number */
SRVNODE='LOGBROKER=MY_LOGICAL_NODE,MY_SET' /* logical node name */
SRVNODE='*' /* logical service name in SRVNAME */
```

If a host name is used for the TCP/IP address, the name must either be known to your DNS server or it must be defined in the hosts file of your TCP/IP configuration.

If the port number is omitted, either a default port number is used by the EntireX broker stub or a host name must be used, and the host name must be known to your DNS server or must be defined in the services file of your TCP/IP configuration.

For more details about Location Transparency and logical node names, refer to the EntireX documentation.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

171

SRVTERM - Server Termination Event

This Natural profile parameter specifies the event at which a Natural RPC server is automatically terminated.

SRVTERM is specified on the server side only.

Possible settings	NEVER	A Natural RPC server is never automatically terminated. To terminate a Natural RPC server refer to <i>Terminating a Natural RPC Server</i> and <i>Terminating an EntireX Broker Service</i> in the <i>Natural Remote Procedure Call (RPC)</i> documentation.
	TIMEOUT	A Natural RPC server is automatically terminated if the wait time for the next client request outside of an RPC conversation is exceeded. TIMEOUT should only be set if you use an Attach Manager to dynamically start Natural RPC servers on request.
Default setting	NEVER	
Dynamic specification	yes	
Specification within session	no	

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

172

SRVUSER - User ID for RPC Server Registry

This Natural profile parameter specifies the user ID needed to register an RPC server on the node specified with the profile parameter [SVRNODE](#).

In case of an EntireX Broker node, `SRVUSER` is also used to logon to the EntireX Broker. A password is either taken from Natural Security (see `*NSC` below) or specified via the application programming interface `USR2072N`.

`SRVUSER` is specified on the server side only.

Possible settings	<i>user-ID</i>	Valid user ID, *USER or *NSC. 1 to16 characters.
	*USER	If <code>SRVUSER</code> is set to *USER, the Natural server uses the current Natural user ID (see system variable *USER) to logon to the node.
	*NSC	If <code>SRVUSER</code> is set to *NSC and Natural Security is installed, the Natural server uses the current Natural user ID (see system variable *USER) and the password defined for this user ID in Natural Security to logon to the node.
Default setting	<i>timestamp</i>	If the user ID is omitted, the timestamp will be used.
Dynamic specification	yes	
Specification within session	no	

For additional information on Natural RPC, see the *Natural Remote Procedure Call (RPC)* documentation.

173

SRVWAIT - Wait Time of RPC Server

This Natural profile parameter specifies the number of seconds the server is to wait for an RPC client request. If this time is exceeded, the RPC server is informed by the transport layer. The RPC server writes a corresponding message to the Natural RPC server trace file and continues to wait for an RPC client request.

SRVWAIT is specified on the server side only.

Possible settings	0 - 32767	Wait time in seconds.
Default setting	0	Unlimited wait time.
Dynamic specification	yes	
Specification within session	no	

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

174

SSIZE - Size of Source Area Allocated by the Editors

This Natural profile parameter determines the maximum size of the Natural source area, which will be dynamically allocated by the Natural editors.

Possible settings	1 - 100	Maximum size of the Natural source area in MB.
Default setting	1	
Dynamic specification	no	
Specification within session	no	



Anmerkung: The maximum size for one Natural source member is 1 MB (independent of SSIZE).

175

STACK - Place Data/Commands on the Stack

This Natural profile parameter is used to place data/commands on the Natural stack.



Anmerkung: If `STACK` is used, a colon (`:`) must be specified with the profile (or session) parameters `DC`, `HI`, `IA` and `ID`.

Possible settings	any character string
Default setting	none
Dynamic specification	yes
Specification within session	no

The profile parameter `STACK` is used to place data/commands on the Natural command stack. The amount of data to be passed with this parameter is limited to 512 bytes. If this limit is exceeded, a corresponding error message is returned.

The stack can contain a sequence of Natural commands and/or user-specified commands, together with their data, for execution at the beginning of the Natural session. The command stack is processed before the user is prompted for input on the screen.

If an `INPUT` statement is encountered during stack processing, the corresponding input screen is generated only if the required input data were not supplied with the command when the stack was created. Any reports generated during stack processing are displayed as usual.

Each system or user-defined command can be optionally followed by data which are used to satisfy requests for information required during the processing of the command. If the command is a user command (that is the name of a user program), any data provided resolve the data requirements of `INPUT` statements within the user program.

Multiple settings for one `INPUT` statement are separated by a comma. Data for multiple `INPUT` statements are separated by a colon (:). A semicolon (;) is used to delimit multiple commands, for example:

```
LOGON:USER1;UCMD1 A,B;UCMD2 C,D:E;FIN
```

Logs on to the library `USER1`, executes the commands `UCMD1` and `UCMD2` providing the corresponding input data, and ends the Natural session.

```
CMD DATA:DATA;CMD
```

Places commands and data on stack. Since some commands (for example, `GLOBALS`) do not read parameters by `INPUT`, a blank character should be used rather than a colon to delimit a command from the first parameter data element.

If specified dynamically, the character string provided as data for the `STACK` parameter must be enclosed in apostrophes; if the character string contains multiple commands, it must also be enclosed in parentheses, for example:

```
STACK="(LOGON SYSTEM;UCMND)"
```

Logs on to the library `SYSTEM` and executes the command `UCMND`.

176

STARTUP - Program Name for System Variable

*STARTUP

This Natural profile parameter specifies a program name for the Natural system variable *STARTUP. The program whose name is contained in *STARTUP is executed each time the Natural command line is invoked. In a Natural program, you can assign another program name to *STARTUP.

Possible settings	1 - 8 characters	Valid program name.
Default setting	none	
Dynamic specification	no	
Specification within session	no	



Anmerkung: If you have Natural Security installed, STARTUP is not evaluated; the startup program to be used is read from the library profile defined in Natural Security.

177

STEPLIB - Initial Setting for *STEPLIB System Variable

This Natural profile parameter specifies the initial setting for the system variable *STEPLIB.

Possible settings	1-8 characters	Any valid library name.
Default setting	SYSTEM	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

The contents of the system variable *STEPLIB influences the order in which objects are searched in the system file.

Using the Configuration Utility, you can define additional steplib's that can be searched for objects which cannot be found in the current library. See *Steplibs* in the *Configuration Utility* documentation.

See also *Steplibs* and *Search Sequence for Object Execution* in the *Using Natural Studio* documentation.

178

SUBCHAR - Substitution Character for Default Code

Page

This Natural profile parameter allows you to specify the substitution character for the default code page. The substitution character is automatically inserted whenever the conversion of a Unicode character into the current default code page (see profile parameter [CP](#)) fails and the profile parameter [CPCVERR](#) is set to "OFF".

Possible settings	<i>n</i>	Substitution character.
	OFF	If "OFF" is specified, the ICU default substitution character is used; for further information, see also <i>Profile Parameters in the Unicode and Code Page Support</i> documentation.
Default setting	OFF	
Dynamic specification	no	
Specification within session	no	

179

SUTF8 - UTF-8 Format for Sources

This Natural profile parameter specifies the default format to be used when Natural sources are saved.

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	ON	The default format for saving Natural sources is UTF-8. All sources will be saved in UTF-8 format, which assures that the source content does not depend on the installed system code page. If the profile parameter SRETAIN is also set to "ON", only newly created sources will be saved in UTF-8 format. Existing sources will then be saved in the original encoding, if possible.
	OFF	The default format for saving Natural sources is „code page“.
Default setting	OFF	
Dynamic specification	no	
Specification within session	no	

180

SYMGEN - Generate Symbol Table

Symboltabelle generieren

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie ob eine Symboltabelle generiert werden soll.

Die Symboltabelle enthält alle in einem Natural-Programm verwendeten Symbole (z.B. Variablen-namen). Sie ist Bestandteil des generierten Programms und wird z.B. für den Natural Debugger und den Dialogeditor verwendet.

In einer Natural-Session kann der Profilparameter SYMGEN durch den Session-Parameter SYMGEN überschrieben werden.

Mögliche Werte	ON	Eine Symboltabelle wird generiert.	
	OFF	Keine Symboltabelle wird generiert.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	Keine
		Gültiges Kommando:	GLOBALS

181

SYNERR - Control of Syntax Errors

This Natural profile parameter specifies whether or not syntax errors will be passed to the error transaction program.

Possible settings	ON	Syntax errors are passed to the error transaction program.
	OFF	Syntax errors are not passed to the error transaction program.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR4007N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

The error transaction program is defined either with the profile parameter [ETA](#) or by a user program by way of assignment to the system variable *ERROR-TA or, if Natural Security is installed, within the Natural Security library profile.

For further information, see *Using an Error Transaction Program* in the *Programming Guide*.

182 TC - Trailing Characters

Nachgestellte Zeichen

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein DISPLAY-Statement ausgegeben wird, unmittelbar hinter dem Feld angezeigt. Die Breite der Ausgabespalte vergrößert sich dadurch entsprechend.

Der Parameter TC kann auch für Felder des Formats U benutzt werden. Weitere Informationen zum Unicode-Format entnehmen Sie dem Abschnitt *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC*.

Mögliche Werte	beliebige(s) Zeichen	Sie können eine Zeichenkette von 1 bis 10 Zeichen definieren. Sie können die Zeichenkette wahlweise in Apostrophen angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen oder ein schließendes Klammerzeichen enthält, muss in Apostrophen stehen.	
Standard-Einstellung	Keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keines

Beispiele:

```
FORMAT TC=*  
DISPLAY (TC='*B*')
```

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

183

TCU - Unicode Trailing Characters

Nachgestellte Unicode Zeichen

Der Session-Parameter `TCU` ist identisch mit dem Session-Parameter `TC`. Der Unterschied ist, dass die nachgestellten Zeichen immer im Unicode-Format gespeichert werden. Dadurch können Sie die nachgestellten Zeichen mit gemischten Zeichen aus unterschiedlichen Codepages eingeben. Es ist sichergestellt, dass immer das korrekte Zeichen angezeigt wird, unabhängig von der installierten System-Codepage.

Die mit diesem Session-Parameter angegebene Zeichenkette wird bei einem Feld, das über ein `DISPLAY`-Statement ausgegeben wird, unmittelbar hinter dem Feld angezeigt. Die Breite der Ausgabespalte vergrößert sich dadurch entsprechend.

Mögliche Werte	beliebige(s) Zeichen	Sie können eine Zeichenkette von 1 bis 10 Zeichen definieren. Sie können die Zeichenkette wahlweise in Apostrophen angeben; in diesem Fall darf die Zeichenkette jedes beliebige Zeichen enthalten. Eine Zeichenkette, die Anführungszeichen oder ein schließendes Klammerzeichen enthält, muss in Apostrophen stehen.	
Standard-Einstellung	Keine		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT
		Gültiges Kommando:	Keine

Siehe auch:

- *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*
- *Unicode and Code Page Support in the Natural Programming Language, Session Parameters, EMU, ICU, LCU, TCU versus EM, IC, LC, TC*

184 TD - Time Differential

This Natural profile parameter specifies a time differential to be applied to the Natural time/date setting to ensure that the current local time/date is used, rather than the computer center time/date. This parameter is applicable in an environment in which remote nodes are being used in a computer network.

Possible settings	-23,59 to 23,59	The notation <i>hours, minutes</i> is used to add/subtract the specified time to/from the physical machine time to set the time/date to be used by Natural; <i>minutes</i> (if specified) must be 00-59.
Default setting	0,0	
Dynamic specification	yes	
Specification within session	no	

Examples:

```
TD=6           (6 hours ahead)
TD=-11        (11 hours behind)
TD=(5,30)     (5 hours and 30 minutes ahead)
TD=(-6,30)    (6 hours and 30 minutes behind)
```


185

TF - Translation of Database ID/File Number

 **Vorsicht:** This parameter applies to user files only. It does not apply to system files.

This Natural profile parameter is used to translate the database ID/file number of a production database into the database ID/file number of a test database.

The translation of file number is relevant when developing an application in a production environment; it enables you to develop an application in a test database and then transfer the finished application to the production database without having to change or re-compile the application.

The transfer is done by a translation of the application's database identification (DBID) and file number (FNR): with the `TF` parameter, you specify the production DBID/FNR and the test DBID/FNR. The Natural objects are then cataloged with the production DBID/FNR, but whenever a database access is executed, the production DBID/FNR is translated into the test DBID/FNR; that is, the test database is used. This means that testing can take place in the actual production environment, but not with production data.

Possible settings	<i>production-DBID</i>	0-65535, except 255 or can be an asterisk (*) which stands for all DBIDs. DBID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
	<i>production-FNR</i>	1-5000, or an asterisk (*) which stands for all FNRs.
	<i>test-DBID</i>	0-65535, except 255.
	<i>test-FNR</i>	1-5000
Default setting	none	
Dynamic specification	yes	This parameter can be specified dynamically and in the Natural parameter file NATPARM.
Specification within session	no	

The asterisk (*) notation for *production-DBID* and *production-FNR* is mutually exclusive.

The parameter is specified as follows:

```
TF=(production-DBID,production-FNR,test-DBID,test-FNR)
```

The TF parameter can be specified several times. Existing specifications are displayed in a corresponding list box.



Anmerkungen:

1. Production and test databases must be of the same type (Adabas/Adabas, for example).
2. If the database type is not specified in NATPARM, the Adabas is used as default type; that is, SQL and XML databases must be specified explicitly.
3. For SQL and XML databases, the file number must always be set to 1. The DBID must be in the range of 0-254.
4. The profile parameter UDB (User Database ID) is evaluated before the TF parameter is evaluated.

This Natural profile and session parameter is used to enable or disable the use of dynamic thousands separators in edit masks at compilation time. At runtime the dynamic thousands separators are replaced by the value (thousands separator character) of the profile and session parameter `THSEPCH`.

Possible settings	ON	Dynamic thousands separators are used. Every dynamic thousands separator that is not part of a string literal is replaced at runtime with the thousands separator character defined with <code>THSEPCH</code> .	
	OFF	Dynamic thousands separators are not used, that is, thousands separators in the edit mask are treated as literal and displayed unchanged at runtime. This is the compatibility setting.	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	THSEP option of COMPOPT



Anmerkungen:

1. In the Natural source, the dynamic thousands separator is either a comma (,) or a period (.), depending on the current setting of the profile and session parameter `DC` (decimal character). If a comma is specified, then the dynamic thousands separator is a period, otherwise it is a comma.
2. Fields in Unicode format should not be redefined as alphanumeric (A) or numeric (N) fields.

See also:

- Profile parameter `THSEPCH` in the *Parameter Reference*.
- *Customizing Separator Character Displays* in the *Programming Guide*.

187

THSEPCH - Thousands Separator Character

Tausender-Trennzeichen

Mit diesem Profil- und Session-Parameter definieren Sie das zur Laufzeit als Tausender-Trennzeichen (Thousands Separator) zu verwendende Zeichen. Das Tausender-Trennzeichen dient dann zum Ersetzen der dynamischen Tausender-Trennzeichen (Dynamic Thousands Separators) in Editiermasken.



Anmerkung: In der Natural-Source wird das dynamische Tausender-Trennzeichen immer durch ein Komma (,) oder einen Punkt (.) dargestellt.

Mögliche Werte	beliebiges Zeichen	Zur Laufzeit wird das dynamische Tausender-Trennzeichen durch dieses Zeichen ersetzt. Soll das Tausender-Trennzeichen durch ein Komma ersetzt werden, muss das Komma in Anführungszeichen gesetzt werden, d.h. THSEPCH=' , ', wenn die dynamische Parameterfunktion benutzt wird, um einzelne Parameter von einander abzutrennen. Wenn das Tausender-Trennzeichen ein Anführungszeichen (') sein soll, dann muss es als zwei Anführungszeichen angegeben werden, die wiederum in Anführungszeichen gesetzt sein müssen, d.h. THSEPCH=' '' '.	
Standard-Einstellung	, (Komma)	Standardmäßig wird als Tausender-Trennzeichen ein Komma verwendet.	
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	keine
		Gültiges Kommando:	GLOBALS

Siehe auch:

- Profilparameter [THSEP](#) in der *Parameter-Referenz-Dokumentation*.

- Option THSEP des Systemkommandos COMPOPT in der *Systemkommandos-Dokumentation*.
- *Trennzeichen-Angaben an lokale Standards anpassen im Leitfaden zur Programmierung.*

188

TIMEOUT - Wait Time for RPC Server Response

This Natural profile and session parameter specifies the number of seconds the client is to wait for an RPC server response. If this time is exceeded, the remote procedure call will be terminated with a corresponding error message.

TIMEOUT is specified on the client side only.

Possible settings	0-32767	Seconds.
Default setting	55	
Dynamic specification	yes	
Specification within session	yes	The setting of this parameter can be changed using the GLOBALS system command.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

189

TMPSORTUNIQ - Alternate Algorithm for Generating

Sort Work File Names

If this profile parameter is specified on the Natural startup command, Natural will use a different algorithm from normal when generating work file names for sort operations. It forces Natural to generate a unique file name for any required work files without embedding the values of the `ETID` parameter and user ID into the file name.

The normal, default algorithm creates work file names with the values of the `ETID` setting and the user ID embedded. Whereas this algorithm usually results in file names which are unique to a particular Natural session, this may cause problems in environments where the `ETID` contains characters which are invalid within a file name, or where multiple Natural sessions are running which use the same user ID and no `ETID` specification (thus possibly resulting in work file names for sort operations being created which are not unique).

Possible settings	specified or not specified	If specified, Natural will use a different algorithm from normal when generating work file names for sort operations.
Default setting	not specified	
Dynamic specification	yes	
Specification within session	no	

190

TQ - Translate Quotation Marks

This parameter corresponds to the Natural profile parameter [TQMARK](#).

191

TQMARK - Translate Quotation Marks

This Natural profile parameter controls the translation of a quotation mark (") within a Natural text constant. It takes effect at compilation time only.

Possible settings	ON	Each quotation mark within a text constant is output as a single apostrophe.	
	OFF	Quotation marks within text constants are not translated, they are output as quotation marks.	
Default setting	ON		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	OPTIONS
		Applicable Command:	TQMARK option of COMPOPT



Anmerkung: Do not confuse quotation mark (") with double apostrophes (' '). Double apostrophes within a text constant are always output as a single apostrophe ('), regardless of the setting of the TQMARK parameter.

Example 1 (TQMARK=ON):

```
WRITE 'THERE" S A QUOTATION MARK'
```

is displayed as: THERE'S A QUOTATION MARK

Example 2 (TQMARK=OFF):

```
WRITE 'THERE"S A QUOTATION MARK'
```

is displayed as: THERE"S A QUOTATION MARK

Example 3 (TQMARK=ON or OFF):

```
WRITE 'DOUBLE APOSTROPHES'' OUTPUT IS A SINGLE APOSTROPHE'
```

is displayed as: DOUBLE APOSTROPHES' OUTPUT IS A SINGLE APOSTROPHE

192

TRACE - Define Trace Level for Natural RPC Servers

This Natural profile parameter activates the RPC trace facility and determines the trace level *n* to be used. For further information, see *Using the Server Trace Facility* p.p. in the *Natural Remote Procedure Call (RPC)* documentation.

TRACE is specified on the server side only.

Possible settings	0	Nothing is traced.
	1	Only messages (inclusive Natural errors) are traced.
	(1, E)	Messages are traced in the event of an error only.
	2	All messages and data from/to client are traced.
	(2, E)	Messages and data from/to client are traced in the event of an error only.
Default setting	0	
Dynamic specification	yes	
Specification within session	no	

The values 3 - 9 displayed in the selection box of the Configuration Utility are also accepted. These values are for future use and behave like TRACE=2.

For further information see the *Natural Remote Procedure Call (RPC)* documentation.

193

TRANSP - Server Transport Protocol

This Natural profile parameter determines which server transport protocol is used. If ACI is used, you can additionally specify the transport method.

TRANSP is specified on the server side only.

Possible settings	ACI	ACI is used. The transport method is defined by the EntireX Broker.
	(ACI , TCP)	ACI is used with TCP/IP.
	(ACI , NET)	ACI is used with Entire Net-work, i.e. using the Adabas protocol.
	(ACI , TCP - NET)	Trying to use ACI with TCP. If not available, ACI is used with NET.
	(ACI , NET - TCP)	Trying to use ACI with NET. If not available, ACI is used with TCP.
Default setting	ACI	
Dynamic specification	yes	
Specification within session	no	

The use of TRANSP is no longer required as you may now specify the full node name with [SRVNODE](#). It is still supported for compatibility reasons.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

194

TRYALT - Try Alternative Server Address

This Natural profile and session parameter determines whether an RPC client should try to execute an RPC request on an alternative server (ON) or not (OFF). For further information, see *Specifying RPC Server Addresses* in the *Natural Remote Procedure Call (RPC)* documentation.

TRYALT is specified on the client side only.

Possible settings	ON	If a request could not be executed on the node you specified, the RPC client tries to find an alternative server address to send that request to.
	OFF	No such attempt will be made.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	yes	The setting of this parameter can be changed using the system command GLOBALS.

For further information, see the *Natural Remote Procedure Call (RPC)* documentation.

195 UC - Underlining Character

Unterstreichungszeichen

Das mit diesem Session-Parameter angegebene Zeichen wird als Unterstreichungszeichen verwendet für:

- Spaltenüberschriften, die von `DISPLAY`-Statements generiert werden;
- Seitenüberschriften/-unterschriften, die über `WRITE TITLE`-/`WRITE TRAILER`-Statements mit `UNDERLINED`-Option erzeugt werden.

Mögliche Werte	beliebiges Zeichen	Siehe auch die folgende Anmerkung.	
	OFF		
Standard-Einstellung	- (Bindestrich)		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT WRITE TITLE WRITE TRAILER
		Gültiges Kommando:	Keines



Anmerkung: Falls Sie keine Unterstreichung von Spaltenüberschriften wünschen, haben Sie folgende Möglichkeiten:

- `UC=` — Statt einer Unterstreichung wird eine Leerzeile ausgegeben.
- `UC=OFF` — Die Feldwerte werden unmittelbar unter der Überschrift, ohne Leerzeile dazwischen, ausgegeben.

Sie können `UC=OFF` nur auf `Statement`-Ebene eines `DISPLAY`-Statements angeben; in diesem Fall können Sie für einzelne Felder in dem Statement keine anderen `UC`-Angaben machen.

Beispiele:

```
FORMAT UC=*  
DISPLAY (UC= ) NAME AGE (UC=+)
```

Siehe auch *Unterstreichungszeichen für Überschriften – UC-Parameter* im Leitfaden zur Programmierung.

This Natural profile parameter specifies the DBID to be used for a database access.

Possible settings	0 - 65535, except 255	Valid database ID. Database ID 255 is reserved for logical system files for Software AG products, see profile parameter LFILE .
Default setting	1	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See <i>SYSEXT - Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation. * Recommended.
	USR1040N *	



Anmerkungen:

1. The database type of DBID 0, which is specified in the DBMS assignments table of the Configuration Utility, and the database types of the DBID specified with the UDB parameter must be the same: ADA/ADA, ADA2/ADA2, SQL/SQL or XML/XML. As an exception, the combination ADA/ADA2 is possible. The first type is the database type of DBID 0 and the second type is the database type of the DBID specified with the UDB parameter.
2. If the DBID in the DDM used is 0, then the database type is taken from the DBMS assignments table entry DBID=0 at compilation time, whereas the database type of the DBID specified with the UDB parameter is used only at runtime.
3. If no DBID 0 is specified in the DBMS assignments table, then the default database type is set to ADA.
4. If no DBID is specified in the DDM used, the DBID specified with the UDB profile parameter determines which database is accessed. If so, the UDB profile parameter must be set to a valid DB number.

5. UDB is also used to specify the DBID for storing transaction data if the profile parameter [ETDB](#) is not specified.

197

ULANG - User Language

This Natural profile parameter specifies the language to be used for date edit masks, system messages, user messages, helptexts, help routines, and multi-lingual maps. The setting is used to set the Natural system variable *LANGUAGE.

Possible settings	1 - 60	Language code. For example, 1 is assigned to English, 2 is assigned to German, 3 is assigned to French. For a detailed list of language codes, see the table in the documentation of the *LANGUAGE variable.
Default setting	1	
Dynamic specification	yes	
Specification within session	no	
Application Programming Interface	USR1005N	See SYSEXT - <i>Natural Application Programming Interfaces</i> in the <i>Utilities</i> documentation.

Within the session, the language code can be specified using the terminal command %L=.

See also:

- *Screen Design, Skill-Sensitive User Interfaces* in the *Programming Guide*.

198

USEDIC - Common Logical Name for Dictionary Servers

This Natural profile parameter is used to specify a common logical name for dictionary servers defined with Natural RPC to enable remote dictionary access on a mainframe or UNIX host.

See also *Dictionary Server Assignments* in the *Configuration Utility* documentation.

Possible settings	any valid server name	Remote dictionary access will be possible.
Default setting	blank	Remote dictionary access will not be possible.
Dynamic specification	no	
Specification within session	no	

199

USER - User ID

This Natural profile parameter is used to enter a user ID. If the default setting is used, the login user ID from the operating system applies.

Possible settings	any valid user ID
Default setting	blank
Dynamic specification	no
Specification within session	no



Anmerkung: Under Natural Security, this profile parameter is ignored.

200 USEREP - Repository Usage

This Natural profile parameter enables you to use the repository.

Possible settings	ON	Repository usage enabled.
	OFF	Repository usage disabled.
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

201

USIZE - Size of User Buffer

This Natural profile parameter specifies the size of the user buffer in virtual memory. The user buffer contains all data dynamically allocated by Natural.

Possible settings	10 - 1024	Buffer size in MB.
	0	With USIZE=0, the memory capacity will be unrestricted.
Default setting	20	
Dynamic specification	no	
Specification within session	no	

202

WEBIO - Use Natural Web I/O Interface

This Natural profile parameter defines whether the Natural input and output (I/O) remains unchanged (i.e. terminal emulation in case of SPoD) or whether the Natural Web I/O Interface is used.

The Natural Web I/O Interface displays the input and output data in a web browser. On platforms other than Windows, it is only possible with the Natural Web I/O Interface to display and enter U format fields with characters which are not contained in the current code page.

The parameter `WEBIO` can only be used when Natural is running as a server, not in an interactive Natural session.

See also *Profile Parameters in the Unicode and Code Page Support* documentation.

Possible settings	ON	I/O via Natural Web I/O Interface.
	OFF	I/O remains unchanged (terminal emulation). If the application is running on a Natural Development Server on Windows, this setting is ignored. In this case, the Natural Web I/O Interface is always used.
Default setting	OFF	
Dynamic specification	yes	
Specification within session	no	

203

WFOPFA - Opening of Work Files

This Natural profile parameter specifies when work files are to be opened by Natural.

Possible settings	ON	A work file is opened at the time when it is first accessed by a given READ WORK FILE or WRITE WORK FILE statement. This means that only those work files which are actually accessed are opened, while the contents of unopened work files are not deleted.
	OFF	All work files referenced in a Natural object are opened when the execution of this object starts. (This may delete the content of a work file when closing if the work file was referenced by a WRITE WORK FILE statement that was never executed.)
Default setting	ON	
Dynamic specification	yes	
Specification within session	no	

WFOPFA=OFF only affects main programs; for routines, WFOPFA=ON always applies.

204

WH - Wait for Record in Hold Status

Warten auf Datensatz im Hold

Dieser Natural Profil- und Session-Parameter gilt nur für Adabas-Datenbanken.

Er bestimmt, was geschehen soll, wenn ein angeforderter Datensatz von einem anderen Benutzer ins Hold gestellt wurde.

In einer Natural-Session kann der Profilparameter WH durch den Session-Parameter WH überschrieben werden.

Mögliche Werte	ON	Der Benutzer wird solange in den Wartestatus versetzt, bis entweder der angeforderte Datensatz wieder verfügbar ist oder eine Datenbanksystem- Zeitüberschreitung oder Überschreitung eines anderen Limits beim Versuch, den Datensatz ins Hold zu stellen, zur Ausgabe einer entsprechenden Fehlermeldung führt.	
	OFF	Eine Fehlermeldung wird ausgegeben, falls einer der gewünschten Datensätze nicht in den Hold-Status gestellt werden kann.	
Standard-Einstellung	OFF		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	



Anmerkung: Wenn ein Natural-Statement ausgeführt wird, das bewirkt, dass Adabas-Datensätze gelesen werden und eine Aktualisierungs/Lösch-Operation folgen könnte, fordert Natural Adabas an, diese Datensätze in den Hold-Status zu stellen. Weitere Informationen zur Hold-Verarbeitung siehe die Adabas *Command Reference*-Dokumentation.

Unter Natural Security: Die Einstellung dieses Parameters kann durch die Session Parameters-Option des *Library Profile* überschrieben werden.

205

WORK - Work-File Assignments

This Natural profile parameter defines the number of work files to be used during the session.

Possible settings	0 - 32	Number of work files.
Default setting	32	
Dynamic specification	no	
Specification within session	no	

WORK=*nn*

If WORK is set to *nn* (in the range 1 - 32), this setting defines the highest work file number which is available for the Natural session.

See also *Work Files* in the *Operations* documentation.

206

XADB - Database for Coordination of Transaction

Processing

This Natural profile parameter is reserved for future use.

It specifies the databases, where transaction processing is to be coordinated by a TP-monitor system. Mark all such databases in the parameter's database list box.

Possible settings	ON / OFF	Use the check boxes to specify the relevant databases.
Default setting	none	
Dynamic specification	no	
Specification within session	no	

207

XREF - Creation of XRef Data for Natural

- Extended XRef Data Generation (For Internal Use Only) 465

This Natural profile and session parameter is used to enable/disable the creation of XRef data for Natural. These are generated in two cases:

- The Natural compiler writes XRef data for Natural programs and data areas when these are cataloged (provided that the XREF parameter has been set to either ON or FORCE, see below).
- Natural Security writes XRef data for programs that are used as Startup, Restart or Error-Transaction in an application or as a special link if the XREF parameter is set to ON or FORCE in the application's Natural Security definition and a user system file is defined for the application.

This parameter controls the compilation in two aspects:

- generation of XRef data in the cases described above and
- to fulfil premise to document implementation objects. The adherence to this premise can be ensured by allowing the completion of the catalog operation only for objects that are documented in the Predict FDIC system file or in the development server file used in Natural Single Point of Development (SPoD).

This parameter also determines how XRef data are treated when Natural members are processed with the Natural utilities SYSMAIN or INPL or with the Object Handler.

Possible settings	ON	XRef data are generated in the cases described above. Documentation premise is not checked.	
	OFF	XRef data are not generated. Documentation premise is not checked.	
	FORCE	A Natural object can only be cataloged if a documentation object already exists for this implementation object. XRef data are generated in the cases described above.	
	DOC	A Natural object can only be cataloged if a documentation object already exists for this object. XRef data are not generated.	
Default setting	OFF		
Dynamic specification	yes		
Specification within session	yes	Applicable Statements:	none
		Applicable Commands:	GLOBALS, XREF

There are different ways to set the Natural XREF parameter:

- In the Natural parameter file.
- As a dynamic parameter when starting a Natural session.
- In Natural Security. If Natural Security has been used to set the XREF parameter, the XREF command may only be used to enforce this setting (by changing from ON to FORCE, from OFF to ON or FORCE).

- With the Natural XREF command. If Natural Security is not installed, the XREF parameter is usually set with the Natural XREF command. The Natural command XREF ? displays the current setting of the XREF parameter.

Extended XRef Data Generation (For Internal Use Only)



Wichtig: The extended XREF parameter is reserved for internal use by Natural.

The extended XREF parameter has the following syntax:

```
XREF=(normal-xref-setting,extended-xref-setting)
```


208

YSLW - Year Sliding or Fixed Window

This Natural profile parameter specifies the range of years covered by the „year sliding window“ or „year fixed window“.

The sliding-window or „year fixed window“ mechanism assumes a date with a 2-digit year to be within a „window“ of 100 years. Within these 100 years, every 2-digit year setting is uniquely related to a specific century, so that there is no confusion about which century is meant.

Possible settings	Normal Setting	0	When you set the parameter to "0", the current century is assumed. No sliding or fixed-window mechanism is used.
	Sliding Window	1 - 99	By setting the parameter to a value between "1-99", you determine when the 100-year range begins in the past. The YSLW setting is subtracted from the current year to determine the first year of the window range. Example: If the current year is 2002 and you specify YSLW=40, the sliding window will cover the years 1962 to 2061. A 2-digit year setting <i>nn</i> from "62" to "99" is then interpreted accordingly as "19 <i>nn</i> ", while a 2-digit year setting <i>nn</i> from "00" to "61" is interpreted as "20 <i>nn</i> ".
	Fixed Window	1582 - 2600	By setting the parameter to a value between "1582-2600", you determine the first year of a 100-year range. The upper boundary of the 100-year range is evaluated by adding "99" to the value specified. Example: If you specify YSLW=1985, the fixed window will cover the years 1985 to 2084. A 2-digit year setting " <i>nn</i> " from 85 to 99 is then interpreted accordingly as "19 <i>nn</i> ", while a 2-digit year setting " <i>nn</i> " from "00" to "84" is interpreted as "20 <i>nn</i> ".

Default setting	0	No sliding or fixed-window mechanism is used.
Dynamic specification	yes	
Specification within session	no	

The YSLW parameter is evaluated at runtime when an alphanumeric date setting with a 2-digit year component is moved into a date variable. This applies to data settings which are:

- used with the mathematical function VAL;
- used with the IS(D) option in a logical condition;
- read from the stack as input data;
- or entered in a map as input data.

See also the section *Processing of Date Information* in the *Programming Guide*.

209

ZD - Zero-Division Check

Teilung durch Null

Mit diesem Natural Profil- und Session-Parameter bestimmen Sie, was im Falle einer Division durch Null (0) geschehen soll.

In einer Natural-Session kann der Profilparameter ZD durch den Session-Parameter ZD überschrieben werden.

Mögliche Werte	ON	Natural gibt eine Fehlermeldung aus, falls versucht wird, eine Zahl durch Null (0) zu teilen.	
	OFF	Natural gibt bei einer Division durch Null als Ergebnis eine Null aus.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	SET GLOBALS
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Unter Natural Security: Die Einstellung dieses Parameters kann durch die *Session Parameters*-Option des Library-Profiles überschrieben werden.

210 ZP - Zero Printing

Anzeige von Nullwerten

Dieser Natural Profil- und Session-Parameter gibt an, wie ein Feld mit lauter Nullen ausgegeben werden soll, d.h. er wird benutzt, um die Anzeige eines numerischen Feldes (Format N, P, I oder F) oder eines Zeitfeldes (Format T), dessen Wert aus lauter Nullen besteht, zu unterdrücken.

In einer Natural-Session kann der Profilparameter ZP durch den Session-Parameter ZP überschrieben werden.

Mögliche Werte	ON	Bei einem Feldwert, der aus lauter Nullen besteht, wird bei numerischen Feldern eine Null rechtsbündig bzw. bei Zeitfeldern der ganze Feldwert angezeigt.	
	OFF	Ein Feldwert, der aus lauter Nullen besteht, wird nicht angezeigt.	
Standard-Einstellung	ON		
Dynamische Spezifikation	ja		
Spezifikation in Session	ja	Gültige Statements:	DISPLAY FORMAT INPUT PRINT REINPUT SET GLOBALS WRITE
		Gültiges Kommando:	GLOBALS
Programmierschnittstelle (API)	USR1005N	Siehe SYSEXT - <i>Natural Application Programming Interfaces</i> in der <i>Utilities</i> -Dokumentation.	

Siehe auch *Parameter zur Beeinflussung der Ausgabe von Feldern im Leitfaden zur Programmierung*.

