

Natural

Systemvariablen

Version 8.2.7

Oktober 2018

Dieses Dokument gilt für Natural ab Version 8.2.7.

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

Copyright © 1979-2018 Software AG, Darmstadt, Deutschland und/oder Software AG USA, Inc., Reston, VA, USA, und/oder ihre Tochtergesellschaften und/oder ihre Lizenzgeber.

Der Name Software AG und die Namen der Software AG Produkte sind Marken der Software AG und/oder Software AG USA Inc., einer ihrer Tochtergesellschaften oder ihrer Lizenzgeber. Namen anderer Gesellschaften oder Produkte können Marken ihrer jeweiligen Schutzrechtsinhaber sein.

Nähere Informationen zu den Patenten und Marken der Software AG und ihrer Tochtergesellschaften befinden sich unter <http://documentation.softwareag.com/legal/>.

Diese Software kann Teile von Software-Produkten Dritter enthalten. Urheberrechtshinweise, Lizenzbestimmungen sowie zusätzliche Rechte und Einschränkungen dieser Drittprodukte können dem Abschnitt "License Texts, Copyright Notices and Disclaimers of Third Party Products" entnommen werden. Diese Dokumente enthalten den von den betreffenden Lizenzgebern oder den Lizenzen wörtlich vorgegebenen Wortlaut und werden daher in der jeweiligen Ursprungssprache wiedergegeben. Für einzelne, spezifische Lizenzbeschränkungen von Drittprodukten siehe PART E der Legal Notices, abrufbar unter dem Abschnitt "License Terms and Conditions for Use of Software AG Products / Copyrights and Trademark Notices of Software AG Products". Diese Dokumente sind Teil der Produktdokumentation, die unter <http://softwareag.com/licenses> oder im Verzeichnis der lizenzierten Produkte zu finden ist.

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

Dokument-ID: NATMF-NNATVARI-827-20180523DE

Inhaltsverzeichnis

| | |
|---|-----|
| Vorwort | vii |
| 1 Anwendungsbezogene Systemvariablen | 1 |
| *APPLIC-ID | 3 |
| *APPLIC-NAME | 3 |
| *COM | 3 |
| *CONVID | 4 |
| *COUNTER (r) | 5 |
| *CPU-TIME | 5 |
| *CURRENT-UNIT | 6 |
| *DATA | 6 |
| *ERROR-LINE | 7 |
| *ERROR-NR | 7 |
| *ERROR-TA | 8 |
| *ETID | 9 |
| *ISN (r) | 9 |
| *LBOUND | 10 |
| *LENGTH (field) | 12 |
| *LEVEL | 12 |
| *LIBRARY-ID | 12 |
| *LINE | 13 |
| *LINDEX | 13 |
| *LOAD-LIBRARY-ID | 14 |
| *NUMBER (r) | 14 |
| *OCCURRENCE | 15 |
| *PAGE-EVENT | 17 |
| *PAGE-LEVEL | 18 |
| *PROGRAM | 18 |
| *REINPUT-TYPE | 18 |
| *ROWCOUNT | 19 |
| *STARTUP | 19 |
| *STEPLIB | 20 |
| *SUBROUTINE | 21 |
| *THIS-OBJECT | 21 |
| *TYPE | 21 |
| *UBOUND | 22 |
| 2 Datums- und Zeit-Systemvariablen | 25 |
| Verwendung | 26 |
| *DAT* - Datums-Systemvariablen | 26 |
| *TIM* - Zeit-Systemvariablen | 27 |
| Beispiel für Datums- und Zeit-Systemvariablen | 28 |
| 3 Eingabe/Ausgabebezogene Systemvariablen | 31 |
| *CURS-COL | 32 |
| *CURS-FIELD | 32 |

| | |
|---|----|
| *CURS-LINE | 33 |
| *CURSOR | 34 |
| *LINE-COUNT | 34 |
| *LINESIZE | 35 |
| *LOG-LS | 35 |
| *LOG-PS | 35 |
| *PAGE-NUMBER | 36 |
| *PAGESIZE | 36 |
| *PF-KEY | 36 |
| *PF-NAME | 37 |
| *WINDOW-LS | 38 |
| *WINDOW-POS | 38 |
| *WINDOW-PS | 38 |
| 4 Systemvariablen nach Natural-Umgebung | 39 |
| *BROWSER-IO | 40 |
| *DEVICE | 40 |
| *GROUP | 41 |
| *HARDCOPY | 41 |
| *INIT-USER | 42 |
| *LANGUAGE | 43 |
| *NATVERS | 47 |
| *NET-USER | 47 |
| *PARM-USER | 47 |
| *PATCH-LEVEL | 47 |
| *PID | 48 |
| *SCREEN-IO | 48 |
| *SERVER-TYPE | 49 |
| *UI | 49 |
| *USER | 50 |
| *USER-NAME | 50 |
| 5 Systemvariablen nach System-Umgebung | 51 |
| *CODEPAGE | 52 |
| *HARDWARE | 52 |
| *HOSTNAME | 52 |
| *INIT-ID | 53 |
| *INIT-PROGRAM | 53 |
| *LOCALE | 54 |
| *MACHINE-CLASS | 54 |
| *OPSYS | 54 |
| *OS | 55 |
| *OSVERS | 55 |
| *TP | 56 |
| *TPSYS | 56 |
| *TPVERS | 57 |
| *WINMGR | 57 |

| | |
|--------------------------------------|----|
| *WINMGRVERS | 57 |
| 6 XML-bezogene Systemvariablen | 59 |
| *PARSE-COL (r) | 60 |
| *PARSE-LEVEL (r) | 60 |
| *PARSE-NAMESPACE-URI (r) | 60 |
| *PARSE-ROW (r) | 61 |
| *PARSE-TYPE (r) | 61 |
| Stichwortverzeichnis | 63 |

Vorwort

Diese Dokumentation beschreibt die Natural-Systemvariablen.

Natural-Systemvariablen enthalten Informationen zur aktuellen Natural-Session, zum Beispiel die aktuelle Bibliothek (Library), die Benutzerkennung (UID) und Terminalkennung (TID), den aktuellen Status einer Schleifenverarbeitung, den aktuellen Status der Report-Verarbeitung, das aktuelle Datum und die aktuelle Uhrzeit.

Sie können die Systemvariablen an beliebiger Stelle in einem Natural-Programm referenzieren.

Einigen Systemvariablen können Sie in einem Natural-Programm auch einen anderen Wert zuweisen, d.h. ihren von Natural generierten Inhalt überschreiben. Auf diese Möglichkeit wird in den folgenden Beschreibungen durch die Bemerkung „Inhalt modifizierbar“ hingewiesen.

Die Dokumentation für die Natural-Systemvariablen ist in folgende funktionale Gruppen unterteilt:

Anwendungsbezogene Systemvariablen

Systemvariablen, die in Verbindung mit einer Natural-Anwendung von Nutzen sind: Name der Library, in der der Benutzer angemeldet ist, ID der aktuellen Library, Informationen zu Fehlersituationen, Typ bzw. Name des zurzeit ausgeführten Natural-Objekts usw.

Datums- und Zeit-Systemvariablen

Datums- und Zeitvariablen, die mit den Statements COMPUTE, DISPLAY, MOVE, PRINT, WRITE und bei logischen Bedingungen angegeben werden können.

Eingabe/Ausgabebezogene Systemvariablen

Systemvariablen, die eingabe- oder ausgaberelevante Informationen enthalten, z.B. aktuelle Cursorposition, Zeilennummer der aktuellen Zeile in der aktuellen Seite, physikalische Zeilenlänge oder Seitengröße.

Systemvariablen mit Bezug zur Natural-Umgebung

Systemvariablen, die in Bezug auf die Natural-Umgebung relevant sind: Typ/Modus des Geräts, von dem aus Natural aufgerufen wurde, Kennung des Benutzers, beim Natural Security-Logon verwendete Benutzerkennung, Sprachkennzeichen, Natural-Version usw.

Systemvariablen mit Bezug zur System-Umgebung

Systemvariablen, die in Bezug auf das Betriebssystem oder das TP-Monitor-System relevant sind: Name der Hardware-Plattform, Maschine oder Maschinenklasse, auf der Natural läuft, Name oder Versionsnummer des Betriebssystems oder des TP-Monitor-Systems, unter dem Natural läuft, Name und Version des verwendeten Window Managers usw.

XML-bezogene Systemvariablen

Systemvariablen, die in Verbindung mit dem PARSE-Statement zur Verfügung stehen.

Siehe auch die folgenden Abschnitte im Leitfaden zur Programmierung

- *Systemvariablen*
- *Beispiel für Systemvariablen und Systemfunktionen*

1 Anwendungsbezogene Systemvariablen

| | |
|--------------------------|----|
| ■ *APPLIC-ID | 3 |
| ■ *APPLIC-NAME | 3 |
| ■ *COM | 3 |
| ■ *CONVID | 4 |
| ■ *COUNTER (r) | 5 |
| ■ *CPU-TIME | 5 |
| ■ *CURRENT-UNIT | 6 |
| ■ *DATA | 6 |
| ■ *ERROR-LINE | 7 |
| ■ *ERROR-NR | 7 |
| ■ *ERROR-TA | 8 |
| ■ *ETID | 9 |
| ■ *ISN (r) | 9 |
| ■ *LBOUND | 10 |
| ■ *LENGTH (field) | 12 |
| ■ *LEVEL | 12 |
| ■ *LIBRARY-ID | 12 |
| ■ *LINE | 13 |
| ■ *LINEX | 13 |
| ■ *LOAD-LIBRARY-ID | 14 |
| ■ *NUMBER (r) | 14 |
| ■ *OCCURRENCE | 15 |
| ■ *PAGE-EVENT | 17 |
| ■ *PAGE-LEVEL | 18 |
| ■ *PROGRAM | 18 |
| ■ *REINPUT-TYPE | 18 |
| ■ *ROWCOUNT | 19 |
| ■ *STARTUP | 19 |
| ■ *STEPLIB | 20 |
| ■ *SUBROUTINE | 21 |
| ■ *THIS-OBJECT | 21 |
| ■ *TYPE | 21 |

| | |
|-----------------|----|
| ■ *UBOUND | 22 |
|-----------------|----|

*APPLIC-ID

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die ID der Library, in der der Benutzer zurzeit angemeldet ist.

*APPLIC-NAME

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Unter Natural Security

Wenn Natural Security installiert ist, enthält diese Systemvariable den Namen der Library , in der der Benutzer zurzeit angemeldet ist, oder, wenn der Benutzer über einen speziellen Link angemeldet ist, den Link-Namen. Wenn Natural Security nicht installiert ist, enthält diese Systemvariable den Namen `SYSTEM`.

Die allgemeine Option `Set *APPLIC-NAME always to library name` kann so gesetzt werden, dass `*APPLIC-NAME` immer den Namen der Library enthält, und zwar unabhängig davon, ob der Benutzer über einen speziellen Link angemeldet ist oder nicht. Siehe *Set *APPLIC-NAME always to library name* in der *Natural Security*-Dokumentation.

*COM

| | |
|----------------------|------|
| Format/Länge: | A128 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable stellt einen Kommunikationsbereich dar, der es ermöglicht, Daten von außerhalb eines Bildschirmfensters zu verarbeiten.

Wenn ein Fenster aktiv ist, können normalerweise keine Daten außerhalb des Fensters auf dem Bildschirm eingegeben werden. Wenn jedoch eine Map `*COM` als modifizierbares Feld enthält, kann ein Benutzer in dieses Feld auch dann Daten eingeben, wenn gerade ein Fenster auf dem Bildschirm aktiv ist.

Die weitere Verarbeitung kann dann vom Inhalt von `*COM` abhängig gemacht werden. Auf diese Weise können Sie Benutzeroberflächen implementieren, bei denen ein Benutzer auch dann Daten in der Kommandozeile eingeben kann, wenn ein Fenster mit eigenen Eingabefeldern aktiv ist.



Anmerkung: Obwohl `*COM` als modifizierbares Feld in einem `INPUT`-Statement verwendet werden kann, wird es nicht als Eingabefeld, sondern als Systemvariable behandelt; d.h. Eingaben, die in `*COM` gemacht werden, werden genommen, wie sie sind, ohne dass eine Eingabeverarbeitung (z.B. Umsetzung in Großbuchstaben) erfolgt. Sobald `*COM` über ein `INPUT`-Statement auf dem Bildschirm angezeigt wurde, wird mit jedem anschließenden `INPUT`- oder `REINPUT`-Statement der jeweils aktuelle Inhalt von `*COM` ausgegeben.

Siehe auch die folgenden Abschnitte im *Leitfaden zur Programmierung*:

- *Dialog-Gestaltung*
 - *Verarbeitung von Daten außerhalb des aktiven Fensters*
 - *Positionierung des Cursors auf `*COM` – Terminalkommando `%T*`*
 - *Daten vom Bildschirm kopieren*

***CONVID**

| | |
|----------------------|----|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Konversations-ID des aktuellen konversationalen Remote Procedure Calls (RPC). Diese ID wird von einem `OPEN CONVERSATION`-Statement gesetzt.

Mit einem `OPEN CONVERSATION`-Statement kann ein Client einen Server zur alleinigen Benutzung erhalten, um eine Reihe von Services (Subprogrammen) innerhalb eines Server-Prozesses auszuführen. Diese alleinige Benutzung wird als Konversation bezeichnet. Das `OPEN CONVERSATION`-Statement dient dazu, eine Konversation zu eröffnen und die an ihr beteiligten Subprogramme anzugeben. Wenn ein `OPEN CONVERSATION`-Statement ausgeführt wird, weist es der Systemvariablen `*CONVID` eine eindeutige ID zu, die die Konversation identifiziert.

Es können mehrere Konversationen gleichzeitig offen sein. Um von einer offenen Konversation zu einer anderen zu wechseln, weisen Sie `*CONVID` die entsprechende Konversations-ID zu.

Weitere Informationen zu Natural RPC finden Sie in der *Natural RPC (Remote Procedure Call)*-Dokumentation.

*COUNTER (r)

| | |
|----------------------|-----|
| Format/Länge: | P10 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Anzahl, wie oft eine mit einem `FIND`-, `READ`-, `HISTOGRAM`- oder `PARSE`-Statement initiierte Verarbeitungsschleife durchlaufen wurde.

Durch Angabe von (r) in Klammern hinter `*COUNTER` können Sie eine bestimmte Schleife referenzieren, wobei r das Statement-Label bzw. die Sourcecode-Zeilenummer des betreffenden `FIND`-, `READ`-, `HISTOGRAM`- oder `PARSE`-Statements ist. Wenn Sie keine bestimmte Schleife referenzieren, bezieht sich `*COUNTER` auf die gerade aktive Verarbeitungsschleife.

Datensätze, die aufgrund einer `WHERE`-Klausel nicht weiterverarbeitet werden, werden im `*COUNTER` nicht mitgezählt. Datensätze, die aufgrund eines `ACCEPT/REJECT`-Statements nicht weiterverarbeitet werden, werden mitgezählt.

*CPU-TIME

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die aktuell vom Natural-Prozess benutzte CPU-Zeit in Einheiten von 10 ms.

Diese Systemvariable enthält den Wert Null für die folgenden Betriebs- oder TPMonitor-Systeme auf Großrechnern:

- *z/VSE*
- *CICS*
- *IMS TM*
- *openUTM*

*CURRENT-UNIT

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der aktuell ausgeführten Einheit. Dies ist

- im Falle des Objekttyps Function der Name der Function,
- der Name der internen Subroutine, falls eine interne Subroutine ausgeführt wird,
- der Name der externen Subroutine im Falle des Objekttyps Subroutine (siehe auch Systemvariable [*SUBROUTINE](#)),
- im Falle aller anderen Objekttypen (Programm, Subprogramm, Map, Dialog usw.) der Name des Objekts (siehe auch Systemvariable [*PROGRAM](#)).

Der Inhalt von *CURRENT-UNIT liegt immer in Großbuchstaben vor.

*DATA

| | |
|----------------------|------|
| Format/Länge: | N3 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Anzahl der im Natural-Stack gelagerten Datenelemente, die dem nächsten INPUT-Statement als Eingabedaten zur Verfügung stehen. Ist der Stack leer, enthält *DATA den Wert null (0). Ein Wert von -1 bedeutet, dass das zuoberst im Stack gelagerte Element ein Kommando oder der Name einer Natural-Transaktion ist.

Die Werte der Profil/Session-Parameter IA und ID zum Zeitpunkt der Ausführung des STACK-Statements dienen dazu, den Wert von *DATA zu bestimmen.

***ERROR-LINE**

| | |
|----------------------|------|
| Format/Länge: | N4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Sourcecode-Zeilenummer des Statements, das einen Fehler verursacht hat.

`*ERROR-LINE` wird auf Null (0) zurückgesetzt, wenn ein Level-1-Programm mit der Ausführung beginnt.

***ERROR-NR**

Als alternative Schreibweise können Sie auch `*ERROR` verwenden.

| | |
|----------------------|----|
| Format/Länge: | N7 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Fehlernummer des Fehlers, der die Verzweigung zu einer `ON ERROR`-Bedingung bewirkt hat.

Es werden nur Fehlernummern im Bereich von 0 bis 9999 unterstützt.

Normalerweise enthält `*ERROR-NR` die Natural-System-Fehlernummer, die das Eintreten der Fehlerbedingung verursacht hat; wenn allerdings ein Statement `REINPUT WITH TEXT *nnnn` ausgeführt wird, wird `*ERROR-NR` mit der betreffenden anwendungsspezifischen Fehlernummer `nnnn` gefüllt.

Sie können dieser Systemvariablen in einem Natural-Programm einen anderen Wert zuweisen, allerdings nicht innerhalb eines `ON ERROR`-Statement-Blocks.

`*ERROR-NR` wird auf Null (0) zurückgesetzt, wenn ein Level-1-Programm mit der Ausführung beginnt.

*ERROR-TA

| | |
|----------------------|----|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält den Namen des Programms, an das die Kontrolle im Falle einer Fehlerbedingung übergeben wird.

Im Falle eines Fehlers führt Natural ein `STACK TOP DATA`-Statement aus und legt folgende Informationen, welche von einer Error-Transaktion als `INPUT`-Daten verwendet werden können, oben auf dem Stack ab:

- Fehlernummer (N4 bei `SG=OFF`; N5 bei `SG=ON`)
- Zeilennummer (N4)
- Status (A1)
- Programmname (A8)
- Level (N2)

Status kann sein:

| | |
|----------|---|
| C | Kommandoverarbeitungsfehler. |
| L | Logon-Fehler. Bei Status L ist die Zeilennummer 0. |
| O | Objektzeitfehler. |
| S | Nicht korrigierbarer Syntaxfehler. |
| R | Fehler auf Remote-Server (im Zusammenhang mit Natural RPC). |

Wenn der Natural-Profilparameter `SYNERR` auf `ON` gesetzt ist und ein Syntaxfehler auftritt, ist `*LEVEL` Null, und die folgenden Informationen werden außerdem in den Stack gestellt:

- Position des Fehler verursachenden Elements in der Source-Zeile (N3).
- Länge des Fehler verursachenden Elements (N3).

*ETID

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Wert zur Identifizierung von Transaktionsdaten (End of Transaction ID) für Adabas. Dieser Wert kann folgendes sein:

- der Wert des Natural-Profilparameters `ETID`,
- die vom TP-Monitor übergebene Benutzer-ID (nur auf Großrechnern),
- der bei der Natural-Initialisierung im User Exit angegebene Wert (nur auf Großrechnern),
- die im Security-Profil des gerade aktiven Benutzers definierte ETID (falls Natural Security eingesetzt wird).

*ISN (r)

| | |
|----------------------|-----|
| Format/Länge: | P10 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Adabas-ISN des Datensatzes, der gerade in einer mit einem `FIND`- oder `READ`-Statement initiierten Verarbeitungsschleife verarbeitet wird.

Mit *r* in Klammern hinter `*ISN` können Sie eine bestimmte Schleife referenzieren, wobei *r* das Label bzw. die Sourcecode-Zeilenummer des betreffenden `FIND`- oder `READ`-Statements ist. Wenn Sie keine bestimmte Schleife referenzieren, bezieht sich `*ISN` auf die gerade aktive Verarbeitungsschleife.

Bei einer mit `HISTOGRAM` initiierten Verarbeitungsschleife enthält `*ISN` die Nummer der Ausprägung, in der der zuletzt gelesene Wert des gerade verarbeiteten Deskriptors enthalten ist. Ist der Deskriptor nicht Teil einer Periodengruppe, enthält `*ISN` den Wert null (0).

Datenbank-spezifische Anmerkungen:

| | |
|------------------------|--|
| VSAM | Bei VSAM-Datenbanken kann *ISN nur für ESDS und RRDS verwendet werden. Bei ESDS enthält *ISN die relative Byte-Adresse (RBA) und bei RRDS die relative Satznummer (RRN) des Datensatzes, der gerade in einer mit einem FIND- oder READ-Statement initiierten Verarbeitungsschleife verarbeitet wird. |
| SQL-Datenbanken | *ISN kann nicht benutzt werden. |
| DL/I | *ISN kann nicht benutzt werden. |

***LBOUND**

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

LBOUND enthält die aktuelle Untergrenze (Indexwert) eines Arrays für die angegebene(n) Dimension(en) (1, 2 oder 3) oder für alle Dimensionen (Stern-Notation ()).

Syntax:

```
*LBOUND (operand1 [,dim])
```

Operanden-Definitionstabelle:

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

operand1 ist das Array, für das die Untergrenze angegeben wird. Die Index-Notation des Arrays ist optional. Als Index-Notation ist nur die Stern-Notation (*) für den vollständigen Bereich für jede Dimension zulässig.

dim ist die Dimensionsnummer, für die die aktuelle Untergrenze zurückgegeben wird:

$$dim = \begin{Bmatrix} 1 \\ 2 \\ 3 \\ * \end{Bmatrix}$$

Wenn keine Dimension angegeben wird, wird die Untergrenze der ersten Dimension zurückgegeben.

Wenn 1, 2 oder 3 angegeben wird, wird die Untergrenze der ersten, zweiten oder dritten Dimension zurückgegeben.

Wird Stern-Notation (*) verwendet, wird die Untergrenze aller definierten Dimensionen zurückgegeben, d.h.

| | |
|---|---|
| 1 | im Falle eines eindimensionalen Arrays |
| 2 | im Falle eines zweidimensionalen Arrays |
| 3 | im Falle eines dreidimensionalen Arrays |

Wenn ein X-Array nicht zugewiesen wird und die Untergrenze der angegebenen Dimension dieses X-Arrays die variable Indexgrenze ist, d.h., wenn sie durch einen Stern (*) in der Indexdefinition dargestellt wird, dann ist die Untergrenze dieses X-Arrays nicht definiert, und der Zugriff auf *LBOUND führt zu einem Laufzeitfehler. Um den Laufzeitfehler zu vermeiden, kann die Systemvariable *OCCURRENCE benutzt werden, um auf Null-Ausprägungen abzutesten:

```
DEFINE DATA LOCAL
  1 #XA(A5/1:*)
END-DEFINE
IF *OCCURRENCE (#XA) NE 0 AND *LBOUND(#XA) > 10
  THEN ...
```

Beispiele:

```
DEFINE DATA LOCAL
  1 #I (I4)
  1 #J (I4/1:3)
  1 #XA (A5/10:*,20:*)
END-DEFINE
#I := *LBOUND(#XA) /* lower bound of 1st dimension is 10
#I := *LBOUND(#XA,1) /* lower bound of 1st dimension is 10
#I := *LBOUND(#XA,2) /* lower bound of 2nd dimension is 20
#J(1:2) := *LBOUND(#XA,*) /* lower bound of all dimensions
/* #J(1) is 10 and #J(2) is 20
END
```

Siehe auch Systemvariablen *UBOUND und *OCCURRENCE.

*LENGTH (field)

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable gibt die gerade benutzte Länge (in Bytes) eines Feldes zurück, das als dynamische Variable in Form von Code-Einheiten definiert ist. Für Format A und B ist die Länge einer Code-Einheit 1 Byte, und für Format U ist die Länge einer Code-Einheit 2 Bytes (UTF-16).

`*LENGTH(field)` gilt nur für dynamische Variablen.

Siehe auch *Aktuell für eine dynamische Variable benutzter Wertespeicher* im Leitfaden zur Programmierung.

*LEVEL

| | |
|----------------------|------|
| Format/Länge: | N2 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Stufennummer (Level Number) des Objekts (Programm, Subprogramm, externe Subroutine, Map, Help routine oder Dialog), das gerade ausgeführt wird. Stufennummer 1 bezeichnet jeweils ein Hauptprogramm. Falls zur Laufzeit höhere Stufennummern auftreten (maximal 1024), dann hat `*LEVEL` den Inhalt 99.

Interne Subroutinen werden von `*LEVEL` nicht berücksichtigt.

Siehe auch *Verwendung eines Fehlertransaktionsprogramms* im Leitfaden zur Programmierung.

*LIBRARY-ID

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die (mit dem LOGON-Kommando angegebene) Kennung der Bibliothek (Library-ID), in der der Benutzer gerade arbeitet.

Diese Systemvariable entspricht der Systemvariablen `*APPLIC-ID`.

*LINE

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Nummer der gerade in einem Natural-Objekt ausgeführten Zeile.

Wenn ein Natural-Programmierobjekt mit dem Natural Optimizer Compiler kompiliert wurde, reflektiert der Wert von *LINE möglicherweise nicht die korrekte Zeilennummer, wenn die Source-Zeile, in der die Systemvariable benutzt wird, Bestandteil einer Folge von Statements ist, die vom Natural Optimizer Compiler optimiert wurden.

*LINEX

| | |
|----------------------|------|
| Format/Länge: | A100 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Zeilennummer des gerade ausgeführten Statements und alle Zeilennummern der verwendeten INCLUDE-Statement-Ebenen.

Wenn diese Systemvariable im Sourcecode des Haupt-Programms (nicht innerhalb eines Copycode) verwendet wird, enthält sie nur die Nummer der Zeile, in der sie referenziert wird. In diesem Fall handelt es sich um den gleichen Wert, der von der Systemvariablen *LINE geliefert wird, lediglich das Format ist ein anderes (A100).

Wird die Systemvariable *LINEX in einer geschachtelten INCLUDE-Statement-Struktur verwendet, dann enthält dieses Feld alle Zeilennummern vom ersten bis zum letzten INCLUDE-Statement und die Zeilennummer des Statements, wobei die einzelnen Nummern durch einen Schrägstrich voneinander abgetrennt sind.

Beispiel:

```
....
3200
3210 INCLUDE COPY01
    0010 ...
    0020 ...
    ....
    0200 INCLUDE COPY02
        0010 ...
        0020 ...
```

```
....  
0050 PRINT *LINEX
```

In diesem Beispiel erzeugt das PRINT-Statement die Ausgabe 3210/0200/0050. Diese gibt den vollständigen Zeilennummernpfad an, der benötigt wird, um genau die Stelle im Programm zu lokalisieren, an der die Systemvariable *LINEX zuerst referenziert worden ist.

*LOAD-LIBRARY-ID

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die ID der Library, aus der das zurzeit ausgeführte Objekt geladen wurde.

*NUMBER (r)

| | |
|----------------------|-----|
| Format/Länge: | P10 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält eine der folgende Informationen:

- Die Anzahl der Datensätze, die mit einem FIND-Statement (unter Erfüllung der WITH-Bedingung) gelesen wurden.
- Die Anzahl der Werte, die als Ergebnis eines HISTOGRAM-Statements ausgewählt wurden.
- Den Ende-Versatz des abgerufenen Segments als Ergebnis eines READLOB-Statements.
- Den Ende-Versatz des eingefügten Segments als Ergebnis eines UPDATELOB-Statements.

Mit der Notation (r) hinter *NUMBER können Sie ein zugehöriges Statement referenzieren, wobei r das Label bzw. die Sourcecode-Zeilenummer des Statements ist. Wenn Sie r nicht angeben, bezieht sich *NUMBER standardmäßig automatisch auf die innerste aktive FIND-, HISTOGRAM- oder READLOB-Schleife. Die Notation r ist immer erforderlich, wenn sich *NUMBER auf ein UPDATELOB-Statement bezieht.



Anmerkung: Wenn *NUMBER zusammen mit einem FIND-Statement benutzt wird und die Adabas-Datei, auf die zugegriffen wird, durch die Adabas-Funktion *Security By Value* geschützt ist, enthält *NUMBER den Wert 9999999999, wenn mehr als 1 Datensatz gefunden wurde. Wenn 1 Datensatz gefunden wurde, enthält *NUMBER den Wert 1. Wenn kein Datensatz gefunden wurde, enthält *NUMBER den Wert 0.

Datenbank-spezifische Anmerkungen:

| | |
|------------------------|---|
| VSAM | Bei VSAM-Datenbanken enthält *NUMBER die Anzahl der gefundenen Datensätze nur bei einem HISTOGRAM-Statement oder bei einem FIND-Statement, in dem der Operator EQUAL TO im Suchkriterium verwendet wird. Bei jedem anderen Operator enthält *NUMBER nicht die Anzahl der gefundenen Datensätze: Werden keine Datensätze gefunden, enthält *NUMBER den Wert 0; jeder andere Wert bedeutet, dass Datensätze gefunden wurden, aber der Wert steht in keinem Zusammenhang mit der tatsächlichen Anzahl der gefundenen Datensätze. |
| SQL-Datenbanken | Bei SQL-Datenbanken enthält *NUMBER nur die Anzahl der gefundenen Zeilen, wenn *NUMBER mit einem FIND NUMBER- oder einem HISTOGRAM-Statement ohne WHERE-Klausel benutzt wird. In allen anderen Fällen enthält *NUMBER nicht die Anzahl der gefundenen Zeilen: Werden keine Zeilen gefunden, enthält *NUMBER den Wert null (0); jeder andere Wert bedeutet, dass Zeilen gefunden wurden, aber der Wert steht in keinem Zusammenhang mit der tatsächlichen Anzahl der gefundenen Zeilen. |
| DL/I | Bei DL/I-Datenbanken enthält *NUMBER nicht die Anzahl der gefundenen Segment-Ausprägungen. *NUMBER enthält 0, wenn keine Segment-Ausprägung das Suchkriterium erfüllt, und den Wert 8.388.607=X'7FFFFF', wenn mindestens eine Segment-Ausprägung das Suchkriterium erfüllt. |

***OCCURRENCE**

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable liefert die aktuelle Anzahl der Ausprägungen eines variablen Arrays. Sie kann bei allen Arten von Array-Feldern mit einer festen oder änderbaren Anzahl an Ausprägungen angewendet werden.

Dazu gehören:

- statische Arrays mit einer konstanten Anzahl an Ausprägungen

Beispiel: (1:5)

- X-Arrays mit einer änderbaren Anzahl an Ausprägungen

Beispiel: (1:*)

- Parameter-Arrays, definiert als (1:V)

Syntax:

`*OCCURRENCE (operand1 [,dim])`

Operanden-Definitionstabelle:

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

operand1 ist das Array, für das die Anzahl der Ausprägungen zurückgegeben wird. Die Index-Notation des Arrays ist optional. Falls sie angegeben wird, ist nur die Stern-Notation (*) für den vollständigen Bereich für jede Dimension zulässig, zum Beispiel `*OCC(#X(*))` oder `*OCC(#Y(*,*))`.

dim ist die Dimensions-Nummer, für die die aktuelle Anzahl der Ausprägungen zurückgegeben wird:

$$dim = \begin{Bmatrix} 1 \\ 2 \\ 3 \\ * \end{Bmatrix}$$

Erklärung:

| | |
|---|--|
| 1 | Eindimensionales Array. Dies ist die Voreinstellung, wenn <i>dim</i> nicht angegeben wird. |
| 2 | Zweidimensionales Array. |
| 3 | Dreidimensionales Array. |
| * | Alle für das entsprechende Array definierten Dimensionen sind gültig. |

In einer Parameter Data Area können Sie mit der Index-Notation `1:V` ein Array mit einer variablen Anzahl von Ausprägungen definieren (vgl. `DEFINE DATA`-Statement). Die tatsächliche Anzahl der Ausprägungen eines solchen variablen Arrays wird erst zur Laufzeit bestimmt. Mit `*OCCURRENCE` können Sie die tatsächliche Anzahl der Array-Ausprägungen ermitteln.

Beispiele:

```
DEFINE DATA
PARAMETER
  1 #PARR (I2/1:V)
LOCAL
  1 #FARR (I2/1:5)
  1 #XARR1 (I2/1:*)
  1 #XARR2 (I2/1:*,1:*)
  1 #I (I2)
  1 #J (I2)
END-DEFINE
FOR #I = 1 TO *OCC(#PARR)      /* Parameter array
```



```

WRITE 2X #I
END-FOR
FOR #I = 1 TO *OCC(#FARR)      /* Fixed array
  WRITE 4X #I
END-FOR
EXPAND ARRAY #XARR1 TO (1:4)
FOR #I = 1 TO *OCC(#XARR1)    /* X-Array
  WRITE 6X #I
END-FOR
EXPAND ARRAY #XARR2 TO (1:3,1:4)
FOR #I = 1 TO *OCC(#XARR2,1)  /* X-Array
  FOR #J = 1 TO *OCC(#XARR2,2)
    WRITE 8X #I #J
  END-FOR
END-FOR
END ↵

```

Siehe auch Beispiel-Programme OCC1P und OCC2P.

In Zusammenhang mit X-Arrays enthält *OCCURRENCE die aktuelle Anzahl der Ausprägungen:

```

DEFINE DATA LOCAL
  1 #I      (I4)
  1 #J      (I4/1:3)
  1 #XA     (A5/1:*,1:*)
END-DEFINE
EXPAND ARRAY #XA TO (1:10,1:20)
#I      := *OCC(#XA)      /* #I=10
#I      := *OCC(#XA,1)    /* #I=10
#I      := *OCC(#XA,2)    /* #I=20
#J(1:2) := *OCC(#XA,*)    /* #J(1)=10 #J(2)=20
END

```

*PAGE-EVENT

| | |
|----------------------|---------------|
| Format/Länge: | U (dynamisch) |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den von Natural for Ajax gelieferten Namen des aktuellen Ereignisses.

Diese Systemvariable wird mit dem PROCESS PAGE-Statement zur Programmierung von Rich GUI-Anwendungen verwendet.

Weitere Informationen finden Sie in der *Natural for Ajax*-Dokumentation.

*PAGE-LEVEL

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Level des aktiven `PROCESS PAGE MODAL`-Statement-Blocks.

Ist kein `PROCESS PAGE MODAL`-Statement-Block aktiv ist, dann ist der Wert von `*PAGE-LEVEL` Null (0).



Anmerkung: Ist der Wert von `*PAGE-LEVEL` größer als Null, dann ist keine Ausgabe an Report 0 über ein `INPUT`-, `PRINT`-, `WRITE`- oder `DISPLAY`-Statement möglich.

*PROGRAM

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen des Natural-Objekts, das gerade ausgeführt wird.

*REINPUT-TYPE

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable gibt an, ob sich die Anwendung in einem Status befindet, der die Ausführung eines `REINPUT`- oder eines `PROCESS PAGE UPDATE`-Statements zulässt.

Eine Anwendung kann ein `INPUT`-Statement und (falls sie mit Natural for Ajax läuft) ein `PROCESS PAGE USING`-Statement verwenden, um eine Ein-/Ausgabeverarbeitung auszuführen. Unter bestimmten Bedingungen kann eine Anwendung diese Ein-/Ausgabe-Statements mit einem `REINPUT`- oder `PROCESS PAGE UPDATE`-Statement zurückgeben oder erneut ausführen.

Der von dieser Systemvariablen zurückgegebene Wert gibt an, ob ein solches Statement zur erneuten Ausführung an dieser Stelle möglich ist oder nicht. Sie enthält einen der folgenden Werte:

| Wert | Beschreibung |
|---------------|--|
| (Leerzeichen) | Die Anwendung kann weder ein REINPUT- noch ein PROCESS PAGE UPDATE-Statement ausführen. |
| REINPUT | Die Anwendung kann ein REINPUT-Statement, aber kein PROCESS PAGE UPDATE-Statement ausführen. |
| UPDATE | Die Anwendung kann ein PROCESS PAGE UPDATE-Statement, aber kein REINPUT-Statement ausführen. |

*ROWCOUNT

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Anzahl der Zeilen, die mit einem der Natural-SQL-Statements „Searched“ DELETE, „Searched“UPDATE oder INSERT (mit *select-expression*) in einer Datenbank-tabelle gelöscht, geändert bzw. hinzugefügt wurden. *ROWCOUNT bezieht sich jeweils auf das zuletzt ausgeführte dieser Statements.

*STARTUP

| | |
|----------------------|----|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Ja |

Das Programm, dessen Name in dieser Systemvariable steht, wird immer dann ausgeführt, wenn Natural normalerweise die Kommandoeingabeaufforderung (NEXT-Prompt bzw. Direktkommando-Zeile/-Fenster) anzeigen würde.

*STARTUP enthält den Namen des Programms, das in Natural Security als Startup-Transaktion im Security-Profil der betreffenden Library eingetragen ist (außer im Batch-Betrieb; vgl. *Natural Security*-Dokumentation).

Falls keine Startup-Transaktion eingetragen ist oder Natural Security nicht verwendet wird, hängt der Wert von *STARTUP davon ab, wie der Profilparameter MENU gesetzt ist:

- Wenn MENU=OFF gesetzt ist, ist *STARTUP leer.
- Wenn MENU=ON gesetzt ist, enthält *STARTUP den Namen MAINMENU, d.h. das Natural-Hauptmenü wird aufgerufen.

Über ein Natural-Programm können Sie der Systemvariablen *STARTUP einen Wert zuweisen, der dann ihren jeweils vorherigen Inhalt überschreibt.



Anmerkung: Ein im Batch-Betrieb verwendetes Startup-Programm muss ein `FETCH-` oder `STACK COMMAND`-Statement enthalten; sonst kann ein Fehler NAT9969 auftreten.

Falls Sie die Kommandoeingabeaufforderung durch Eingabe von `%%` (oder einem gleichwertigen Kommando) aufrufen — entweder in einer Nicht-Security-Umgebung oder in einer Security-Umgebung, in der der Kommando-Modus für die aktuelle Library nicht verboten ist — wird dadurch der Startup-Mechanismus deaktiviert. Um ihn wieder zu aktivieren, müssen Sie sich entweder erneut in die Library begeben oder ein Programm ausführen, das s.o. `*STARTUP` wieder einen Wert zuweist.



Wichtig: Um das Programm zu deaktivieren, das in `*STARTUP` enthalten ist, setzen Sie z.B. mittels eines Statements `RESET *STARTUP` den Wert der Systemvariable auf „leer“.

Unter Natural Security:

In einer Natural-Security-Umgebung, in der der Kommando-Modus für die aktuelle Library verboten ist, bewirkt `%%`, dass das Programm, dessen Name in `*STARTUP` steht, aufgerufen wird.

Wenn ein durch eine Startup-Transaktion (`*STARTUP`) verursachter Natural-Laufzeitfehler auftritt, kann Naturals Fehlerverarbeitung dazu führen, dass die Startup-Transaktion erneut ausgeführt wird. Dieses würde zu einer Fehlerschleifensituation führen. Um eine solche Schleife zu verhindern, steht die allgemeine Option `Logoff in error case if *STARTUP is active` zu Verfügung. Siehe *Logoff in Error Case if *STARTUP is Active* in der *Natural Security*-Dokumentation.

*STEPLIB

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der Steplib-Library, die mit der Natural-Library, in der der Benutzer gerade arbeitet, verknüpft ist.

Wenn Natural Security nicht installiert ist, enthält `*STEPLIB` den mit dem Profilparameter `STEPLIB` angegebenen Namen.

Wenn Natural Security installiert ist, kann die Steplib im Security-Profil der betreffenden Library eingetragen werden.



Anmerkung: Bei der Steplib-Library wird immer davon ausgegangen, dass sie dieselbe Datenbank-ID und Dateinummer hat wie die aktuelle Library des Benutzers. Es wird davon ausgegangen, dass außer der Library `SYSTEM` die Libraries mit dem Namen `SYSxxx` sich in der Systemdatei `FNAT` und andere Libraries sich in `FUSER` befinden.

*SUBROUTINE

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der externen Subroutine, die gerade ausgeführt wird. Der Inhalt von *SUBROUTINE steht stets in Großbuchstaben.

*THIS-OBJECT

| | |
|----------------------|------------------|
| Format/Länge: | HANDLE OF OBJECT |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält eine Handle zum gerade aktiven Objekt. Das gerade aktive Objekt benutzt *THIS-OBJECT, um entweder seine eigenen Methoden auszuführen oder eine Referenz auf sich selbst an ein anderes Objekt zu übergeben.

*THIS-OBJECT enthält nur einen tatsächlichen Wert, wenn eine Methode ausgeführt wird. Sonst enthält es NULL-HANDLE.

*TYPE

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Typ des Natural-Objekts, das gerade ausgeführt wird.

Gültige Werte für *TYPE:

| Wert | Objektyp |
|-------------|-------------|
| PROGRAM | Programm |
| FUNCTION | Function |
| SUBPROGRAM | Subprogramm |
| SUBROUTINE | Subroutine |
| HELPROUTINE | Helproutine |
| MAP | Map |

| Wert | Objektyp |
|---------|----------|
| ADAPTER | Adapter |

*UBOUND

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

***UBOUND** enthält die aktuelle Obergrenze (Indexwert) eines Arrays für die angegebene(n) Dimension(en) (1, 2 oder 3) oder für alle Dimensionen (Stern-Notation (*)).

Syntax:

```
*UBOUND (operand1 [,dim])
```

Operanden-Definitionstabelle:

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

operand1 ist das Array, für das die Obergrenze angegeben wird. Die Index-Notation des Arrays ist optional. Als Index-Notation ist nur die *-Notation für den vollständigen Bereich für jede Dimension zulässig.

dim ist die Dimensionsnummer, für die die aktuelle Obergrenze zurückgegeben wird:

$$dim = \begin{Bmatrix} 1 \\ 2 \\ 3 \\ * \end{Bmatrix}$$

Wenn keine Dimension angegeben wird, wird die Obergrenze der ersten Dimension zurückgegeben.

Wenn 1, 2 oder 3 angegeben wird, wird die Obergrenze der ersten, zweiten oder dritten Dimension zurückgegeben.

Wenn * angegeben wird, wird die Obergrenze aller definierten Dimensionen zurückgegeben, d.h.

| | |
|---|---|
| 1 | im Falle eines eindimensionalen Arrays |
| 2 | im Falle eines zweidimensionalen Arrays |
| 3 | im Falle eines dreidimensionalen Arrays |

Wenn ein X-Array nicht zugewiesen wird und die Obergrenze der angegebenen Dimension dieses X-Arrays die variable Indexgrenze ist, d.h., wenn sie durch einen Stern (*) in der Indexdefinition dargestellt wird, dann ist die Obergrenze dieses X-Arrays nicht definiert, und der Zugriff auf *UBOUND führt zu einem Laufzeitfehler. Um den Laufzeitfehler zu vermeiden, kann die Systemvariable *OCCURRENCE benutzt werden, um auf Null-Ausprägungen abzufragen:

```
DEFINE DATA LOCAL
  1 #XA(A5/1:*)
END-DEFINE
IF *OCCURRENCE (#XA) NE 0 AND *UBOUND(#XA) > 10
  THEN ...
```

Beispiele:

```
DEFINE DATA LOCAL
  1 #I  (I4)
  1 #J  (I4/1:3)
  1 #XA (A5/*:10,*:20)
END-DEFINE
#i  := *UBOUND(#XA)           /* upper bound of 1st dimension is 10
#i  := *UBOUND(#XA,1)        /* upper bound of 1st dimension is 10
#i  := *UBOUND(#XA,2)        /* upper bound of 2nd dimension is 20
#j(1:2) := *UBOUND(#XA,*)    /* upper bound of all dimensions
                                   /* (1st and 2nd)
                                   /* #J(1) is 10 and #J(2) is 20
```

Siehe auch die Systemvariablen *LBOUND und *OCCURRENCE.

2 Datums- und Zeit-Systemvariablen

| | |
|---|----|
| ■ Verwendung | 26 |
| ■ *DAT* - Datums-Systemvariablen | 26 |
| ■ *TIM* - Zeit-Systemvariablen | 27 |
| ■ Beispiel für Datums- und Zeit-Systemvariablen | 28 |

Verwendung

Die unten aufgeführten Datums- und Zeit-Systemvariablen können an den folgenden Stellen angegeben werden:

■ Statements:

- COMPUTE
- DISPLAY
- MOVE
- PRINT
- WRITE

■ Logische Bedingungen

Der von Natural generierte Inhalt der Datums- und Zeit-Systemvariablen kann *nicht* verändert werden, d.h. Sie können keiner dieser Variablen in einem Natural-Programm einen anderen Wert zuweisen.

DAT - Datums-Systemvariablen

Alle Datumsvariablen enthalten das aktuelle Datum. Das Format des Datums ist bei jeder Variablen anders, wie die folgende Tabelle zeigt.

| Datumsvariable | Format/Länge | Datumsformat |
|----------------|--------------|---------------------------------|
| *DATD | A8 | DD.MM.YY |
| *DAT4D | A10 | DD.MM.YYYY |
| *DATE | A8 | DD/MM/YY |
| *DAT4E | A10 | DD/MM/YYYY |
| *DATG | A15 | DDmonthnameYYYY (Gregorianisch) |
| *DATI | A8 | YY-MM-DD |
| *DAT4I | A10 | YYYY-MM-DD |
| *DATJ | A5 | YYJJJ (Julianisch) |
| *DAT4J | A7 | YYYYJJJ (Julianisch) |
| *DATN | N8 | YYYYMMDD |
| *DATU | A8 | MM/DD/YY |
| *DAT4U | A10 | MM/DD/YYYY |
| *DATV | A11 | DD-MON-YYYY |

| Datumsvariable | Format/Länge | Datumsformat |
|----------------|--------------|-----------------------|
| *DATVS | A9 | DDMONYYYY |
| *DATX | D | Internes Datumsformat |

*D = Day (Tag), J = Jualianischer Kalendertag, M = Monat, Y = Year (Jahr), MON = führende drei Bytes des Monatsnamens wie in *DATG.

TIM - Zeit-Systemvariablen

Zur Laufzeit wird der Inhalt einer Zeit-Systemvariablen jedes Mal von neuem ausgewertet, wenn die Variable in einem Natural-Programm referenziert wird. Das Format der Zeit ist für jede Zeit-Variable unterschiedlich (siehe unten).

| Zeitvariable | Format/Länge | Erklärung |
|--------------|--------------|--|
| *TIMD (r) | N7 | <p>Kann nur in Verbindung mit einem vorangegangenen SETTIME-Statement verwendet werden.</p> <p>Enthält die Zeit, die seit der Ausführung des SETTIME-Statements verstrichen ist.</p> <p>Format: HHIISS (*).</p> <p>(r) steht für das Statement-Label bzw. die Sourcecode-Zeilenummer des SETTIME-Statements, das als Basis für *TIMD verwendet wird.</p> <p>Wenn *TIMD benutzt wird, ohne zuvor ein SETTIME-Statement auszuführen, dann wird als Wert Null zurückgegeben.</p> |
| *TIME | A10 | Enthält die aktuelle Uhrzeit im Format HH:II:SS.T (*). |
| *TIME-OUT | N5 | <p>*TIME-OUT enthält die Anzahl der verbleibenden Sekunden, bevor die aktuelle Transaktion wegen Zeitüberschreitung abgebrochen wird (nur verfügbar unter Natural Security).</p> <p>*TIME-OUT ist 0, wenn kein datenverändernder Datenbankzugriff erfolgt.</p> <p>*TIME-OUT kommt nur zum Tragen, wenn ein FIND-, READ- oder GET-Statement einen Datensatz zum Zwecke des Änderns oder Löschens liest.</p> <p>*TIME-OUT wird auf Null zurückgesetzt, wenn ein END TRANSACTION- oder BACKOUT TRANSACTION-Statement ausgeführt wird.</p> |
| *TIMESTAMP | B8 | <p>Hardware-interner Zeitzähler (Store Clock).</p> <p>Unter BS2000 ist dieser Wert als Ortszeit oder als GMT-Zeit verfügbar. Um die Konsistenz in allen Umgebungen sicherzustellen, enthält die Systemvariable *TIMESTAMP unter BS2000 die GMT-Zeit.</p> |
| *TIMN | N7 | Enthält die aktuelle Uhrzeit im Format HHIISS (*). |

| Zeitvariable | Format/Länge | Erklärung |
|--------------|--------------|--|
| *TIMX | T | Enthält die aktuelle Uhrzeit im internen Zeitformat. |

* H = Hour (Stunde), I = Minute, S = Sekunde, T = Tenth of a second (Zehntelsekunde).

Beispiel für Datums- und Zeit-Systemvariablen

```

** Example 'DATIVAR': Date and time system variables
*****
DEFINE DATA LOCAL
1 #DATE (D)
1 #TIME (T)
END-DEFINE
*
WRITE NOTITLE
'DATE IN FORMAT DD.MM.YYYY ' *DAT4D /
'DATE IN FORMAT DD/MM/YYYY ' *DAT4E /
'DATE IN FORMAT DD-MON-YYYY ' *DATV /
'DATE IN FORMAT DDMONYYYY ' *DATVS /
'DATE IN GREGORIAN FORM ' *DATG /
'DATE IN FORMAT YYYY-MM-DD ' *DAT4I /
'DATE IN FORMAT YYYYDDD ' *DAT4J /
'DATE IN FORMAT YYYYMMDD ' *DATN (AD=L) /
'DATE IN FORMAT MM/DD/YYYY ' *DAT4U /
'DATE IN INTERNAL FORMAT ' *DATX (DF=L) ///
'TIME IN FORMAT HH:II:SS.T ' *TIME /
'TIME IN FORMAT HHIISST ' *TIMN (AD=L) /
'TIME IN INTERNAL FORMAT ' *TIMX /
*
MOVE *DATX TO #DATE
ADD 14 TO #DATE
WRITE 'CURRENT DATE' *DATX (DF=L) 3X
'CURRENT DATE + 14 DAYS ' #DATE (DF=L)
*
MOVE *TIMX TO #TIME
ADD 100 TO #TIME
WRITE 'CURRENT TIME' *TIMX 5X
'CURRENT TIME + 10 SECONDS' #TIME
*
END

```

Ausgabe des Programms DATIVAR:

```
DATE IN FORMAT DD.MM.YYYY      11.01.2005
DATE IN FORMAT DD/MM/YYYY      11/01/2005
DATE IN FORMAT DD-MON-YYYY     11-Jan-2005
DATE IN FORMAT DDMONYYYY       11Jan2005
DATE IN GREGORIAN FORM         11January 2005
DATE IN FORMAT YYYY-MM-DD      2005-01-11
DATE IN FORMAT YYYYDDD         2005011
DATE IN FORMAT YYYYMMDD        20050111
DATE IN FORMAT MM/DD/YYYY      01/11/2005
DATE IN INTERNAL FORMAT        2005-01-11

TIME IN FORMAT HH:II:SS.T      14:42:05.4
TIME IN FORMAT HHIISS          1442054
TIME IN INTERNAL FORMAT        14:42:05

CURRENT DATE 2005-01-11      CURRENT DATE + 14 DAYS      2005-01-25
CURRENT TIME 14:42:05        CURRENT TIME + 10 SECONDS 14:42:15
```


3

Eingabe/Ausgabebezogene Systemvariablen

| | |
|----------------------|----|
| ■ *CURS-COL | 32 |
| ■ *CURS-FIELD | 32 |
| ■ *CURS-LINE | 33 |
| ■ *CURSOR | 34 |
| ■ *LINE-COUNT | 34 |
| ■ *LINESIZE | 35 |
| ■ *LOG-LS | 35 |
| ■ *LOG-PS | 35 |
| ■ *PAGE-NUMBER | 36 |
| ■ *PAGESIZE | 36 |
| ■ *PF-KEY | 36 |
| ■ *PF-NAME | 37 |
| ■ *WINDOW-LS | 38 |
| ■ *WINDOW-POS | 38 |
| ■ *WINDOW-PS | 38 |

*CURS-COL

| | |
|----------------------|--|
| Format/Länge: | P3 |
| Inhalt modifizierbar | Ja (aber es darf kein negativer Wert zugewiesen werden). |

Diese Systemvariable enthält die Nummer der Spalte, in der sich der Cursor zur Zeit befindet.

Die Cursor-Position bezieht sich auf das aktive Natural-Bildschirmfenster, unabhängig von seiner physischen Platzierung auf dem Bildschirm. Die Position des Cursors wird ausgehend von Zeile 1 / Spalte 1 der logischen Seite bestimmt.

Wenn *CURS-COL einen negativen Wert enthält, bedeutet dies, dass der Cursor außerhalb des aktiven Fensters steht. Wenn *CURS-COL negativ ist, enthält auch *CURS-LINE einen negativen Wert. In diesem Fall bestimmen die absoluten Werte beider Systemvariablen die Position des Cursors auf dem physischen Bildschirm.



Anmerkung: Meldungszeile, Funktionstastenleiste und Statistikzeile/Infoline werden nicht als Zeilen mitgezählt.

Siehe auch *Dialog-Gestaltung, Spalten-sensitive Verarbeitung im Leitfaden zur Programmierung*.

*CURS-FIELD

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable gibt eine Identifikation des Feldes zurück, in dem sich der Cursor zur Zeit befindet. Der zurückgegebene Wert ist die interne Darstellung der Feldadresse.

*CURS-FIELD kann nicht alleine, sondern nur in Verbindung mit der Systemfunktion POS verwendet werden. Sie können beide zusammen benutzen, um zu prüfen, ob sich der Cursor gerade in einem bestimmten Feld befindet, und die weitere Verarbeitung von dieser Bedingung abhängig machen. Nähere Informationen siehe Systemfunktion POS.

Wenn der Cursor nicht in einem Feld steht oder wenn kein REINPUT möglich ist, enthält *CURS-FIELD den Wert 0.

In Natural for Ajax-Anwendungen dient *CURS-FIELD zur Identifikation des Operanden, der den Wert des Control darstellt, welches den Eingabefokus hat. Sie können *CURS-FIELD in Verbindung mit der POS-Funktion benutzen, um eine Prüfung auf das Control, das den Eingabefokus hat, zu veranlassen und die Verarbeitung in Abhängigkeit von diesem Zustand durchzuführen.



Anmerkung: Die Systemvariable `*CURS-FIELD` kann nicht zwischen zwei verschiedenen Variablen unterscheiden, die am selben Speicherplatz beginnen (`REDEFINE`-Variablen), weil die in `*CURS-FIELD` zurückgegebene Feldadresse für beide Variablen dieselbe ist.

Der Wert von `*CURS-FIELD` dient nur zur internen Identifikation des Feldes und kann nicht für arithmetische Operationen verwendet werden. Wenn `*CURS-FIELD` eine Ausprägung eines X-Arrays (ein Array, für das wenigstens eine Dimension als erweiterbar spezifiziert ist) identifiziert, kann sich der Wert von `*CURS-FIELD` ändern, nachdem die Anzahl der Ausprägungen für eine Dimension des Arrays mittels eines der Statements `EXPAND`, `RESIZE` oder `REDUCE` geändert wurde.

Siehe auch *Dialog-Gestaltung, Feld-sensitive Verarbeitung im Leitfaden zur Programmierung*.

*CURS-LINE

| | |
|----------------------|---|
| Format/Länge: | P3 |
| Inhalt modifizierbar | Ja (aber es darf kein negativer Wert oder 0 zugewiesen werden). |

Diese Systemvariable enthält die Nummer der Zeile, in der sich der Cursor zur Zeit befindet.

Die Cursor-Position bezieht sich auf das aktive Natural-Bildschirmfenster, unabhängig von seiner physischen Platzierung auf dem Bildschirm. Die Position des Cursors wird ausgehend von Zeile 1 / Spalte 1 der *logischen* Seite bestimmt.



Anmerkung: Meldungszeile, Funktionstastenleiste und Statistikzeile/Infoline werden nicht als Zeilen mitgezählt.

`*CURS-LINE` kann auch einen der folgenden Werte enthalten:

| Wert | Cursor-Position |
|------|--|
| 0 | Auf der oberen oder unteren horizontalen Rahmenzeile eines Bildschirmfensters. |
| -1 | Auf der Natural-Meldungszeile. |
| -2 | Auf der Natural-Statistikzeile/Infoline. |
| -3 | Auf der oberen (Tastennummern)-Funktionstastenzeile. |
| -4 | Auf der unteren (Tastennamen)-Funktionstastenzeile. |

Wenn `*CURS-COL` einen negativen Wert enthält — was bedeutet, dass der Cursor außerhalb des aktiven Fensters steht —, dann enthält auch `*CURS-LINE` einen negativen Wert. In diesem Fall bestimmen die *absoluten* Werte beider Systemvariablen die Position des Cursors auf dem *physischen* Bildschirm.

Siehe auch *Dialog-Gestaltung, Zeilen-sensitive Verarbeitung im Leitfaden zur Programmierung*.

*CURSOR

| | |
|----------------------|------|
| Format/Länge: | N6 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Position des Cursors auf dem Eingabebildschirm, wenn die EINGABE-Taste oder eine Funktionstaste gedrückt wird.



Anmerkung: Es empfiehlt sich, anstelle von *CURSOR die Systemvariablen *CURS-LINE und *CURS-COL zu verwenden. *CURSOR ist nur noch aus Gründen der Kompatibilität zu früheren Natural-Versionen verfügbar.

*LINE-COUNT

| | |
|----------------------|------|
| Format/Länge: | P5 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Zeilennummer der aktuellen, d.h. zuletzt ausgegebenen Zeile.

Natural benutzt diese Variable, um die Zeilennummer für die nächste Zeile eines Reports zu bestimmen.

Der Wert von *LINE-COUNT erhöht sich jedesmal, wenn ein WRITE-, SKIP-, DISPLAY-, PRINT- oder INPUT-Statement ausgeführt wird. Der Wert erhöht sich mit jeder ausgegebenen Zeile um 1; *LINE-COUNT enthält also die Nummer der zuletzt ausgegebenen Zeile auf der aktuellen Seite.

Mit jedem EJECT- und NEWPAGE-Statement wird *LINE-COUNT wieder auf 1 gesetzt (außer bei NEWPAGE WITH TITLE, wo sich der Wert von *LINE-COUNT aus der Anzahl der Zeilen der mit WITH TITLE ausgegebenen Seitenüberschrift ergibt).

Die höchstmögliche Zeilennummer ist 250.

Erzeugt ein Programm mehrere Reports, können Sie mit der Notation (*rep*) hinter *LINE-COUNT den Report bestimmen, dessen aktuelle Zeilennummer Sie wünschen.

***LINESIZE**

| | |
|----------------------|------|
| Format/Länge: | N7 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die physische Zeilenlänge des Eingabe/Ausgabe-Geräts, von dem aus Natural aufgerufen wurde (falls das eingesetzte TP-System diese Informationen liefern kann).

***LOG-LS**

| | |
|----------------------|------|
| Format/Länge: | N3 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Zeilenlänge der logischen Seite, die mit dem Primär-Report ausgegeben wird.

*LOG-LS gilt nur für den Primär-Report, nicht für etwaige weitere Reports.

***LOG-PS**

| | |
|----------------------|------|
| Format/Länge: | N3 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Seitenlänge der logischen Seite, die mit dem Primär-Report ausgegeben wird.

*LOG-PS gilt nur für den Primär-Report, nicht für etwaige weitere Reports.

*PAGE-NUMBER

| | |
|----------------------|----|
| Format/Länge: | P5 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Nummer der gerade ausgegebenen Seite eines Reports.

Erzeugt ein Programm mehrere Reports, können Sie mit der Notation (*rep*) hinter *PAGE-NUMBER den Report bestimmen, auf den sich *PAGE-NUMBER beziehen soll.

Natural initialisiert diese Variable erst, wenn die Formatierung des Reports erfolgt, d.h. sie wird erst relevant, wenn das erste FORMAT-, WRITE- oder DISPLAY-Statement ausgeführt wird. Diese Variable kann von einem Natural-Programm modifiziert werden.

Natural benutzt diese Variable, um die Seitennummer für die nächste Seite eines Reports zu bestimmen. Der Wert von *PAGE-NUMBER erhöht sich jedesmal um 1, wenn mit einem DISPLAY-, WRITE-, SKIP- oder NEWPAGE-Statement eine neue Seite erzeugt wird. Ein EJECT-Statement hat keinen Einfluss auf *PAGE-NUMBER.

*PAGESIZE

| | |
|----------------------|------|
| Format/Länge: | N7 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die physische Seitenlänge des Eingabe/Ausgabe-Geräts, von dem aus Natural aufgerufen wurde (falls das eingesetzte TP-System diese Informationen liefern kann).

*PF-KEY

| | |
|----------------------|------|
| Format/Länge: | A4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Identifikation der Funktionstaste, die zuletzt gedrückt wurde.

*PF-KEY kann einen der folgenden Werte enthalten:

| Wert | Beschreibung |
|--------------|---------------------------|
| PA1 bis PA3 | PA-Tasten 1 bis 3 |
| PF1 bis PF48 | PF-Tasten 1 bis 48 |
| ENTR | ENTER- bzw. EINGABE-Taste |
| CLR | CLEAR- bzw. LÖSCH-Taste |
| PEN | Lichtstift |
| PGDN | PAGE DOWN-Taste |
| PGUP | PAGE UP-Taste |

*PF-KEY enthält die Identifikation der Taste nur, falls die Taste auf der jeweiligen Stufe aktiviert ist; falls nicht, enthält *PF-KEY den Wert ENTR.



Anmerkung:



Anmerkungen:

1. Wenn ein Seitenumbruch auftritt, ändert sich der Wert von *PF-KEY nach ENTR. Dies gilt für alle Umgebungen (Terminal oder Nicht-Terminal).
2. Wenn Sie abfragen, ob der Wert von *PF-KEY innerhalb eines bestimmten Wertebereichs liegt, berücksichtigen Sie bitte, dass der Wert von *PF-KEY alphanumerisch ist.

Siehe auch

- SET KEY-Statement (in Bezug auf Auswirkungen auf den Inhalt von *PF-KEY).
- *Verarbeitung aufgrund von Funktionstasten – Systemvariable *PF-KEY im Leitfaden zur Programmierung.*

*PF-NAME

| | |
|----------------------|------|
| Format/Länge: | A10 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der Funktionstaste, die zuletzt gedrückt wurde. Dies ist der Name, der der Taste mit der NAMED-Klausel eines SET KEY-Statements zugewiesen wurde.

Damit haben Sie die Möglichkeit, die Verarbeitung von einem bestimmten Funktionsnamen abhängig zu machen, anstatt von einer bestimmten Taste. Wollen Sie beispielsweise erreichen, dass Benutzer Hilfe durch Drücken von wahlweise PF1 oder PF13 anfordern können, weisen Sie beiden Tasten den Namen HILFE zu und knüpfen das Aufrufen der Hilfe an die Bedingung *PF-NAME= 'HILFE': die Hilfe wird dann aufgerufen, ganz gleich ob der Benutzer sie mit PF1 oder PF13 angefordert hat.

Siehe auch *Dialog-Gestaltung, Verarbeitung aufgrund der Namen von Funktionstasten im Leitfaden zur Programmierung*.

***WINDOW-LS**

| | |
|----------------------|------|
| Format/Länge: | N3 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die logische Zeilenlänge (ohne Rahmen) des aktiven Natural-Windows. Siehe `DEFINE WINDOW-Statement`.

***WINDOW-POS**

| | |
|----------------------|------|
| Format/Länge: | N6 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die logische Position der oberen linken Ecke des aktiven Natural-Windows. Siehe `DEFINE WINDOW-Statement`.

Die logische Position wird über mehrere Zeilen hinweg ab der Position 0 (obere linke Ecke) in Zeichen gezählt.

***WINDOW-PS**

| | |
|----------------------|------|
| Format/Länge: | N3 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die logische Seitenlänge (ohne Rahmen) des aktiven Natural-Windows. Siehe `DEFINE WINDOW-Statement`.

4 Systemvariablen nach Natural-Umgebung

| | |
|----------------------|----|
| ■ *BROWSER-IO | 40 |
| ■ *DEVICE | 40 |
| ■ *GROUP | 41 |
| ■ *HARDCOPY | 41 |
| ■ *INIT-USER | 42 |
| ■ *LANGUAGE | 43 |
| ■ *NATVERS | 47 |
| ■ *NET-USER | 47 |
| ■ *PARM-USER | 47 |
| ■ *PATCH-LEVEL | 47 |
| ■ *PID | 48 |
| ■ *SCREEN-IO | 48 |
| ■ *SERVER-TYPE | 49 |
| ■ *UI | 49 |
| ■ *USER | 50 |
| ■ *USER-NAME | 50 |

*BROWSER-IO

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable gibt an, ob die Anwendung in einem Web Browser läuft.

Damit eine Anwendung in einem Web Browser laufen kann, müssen Sie entweder das Natural Web I/O Interface oder Natural for Ajax benutzen.

In einer Anwendung, die über das Natural Web I/O Interface läuft, können Maps verwendet werden. In einer Anwendung, die über Natural for Ajax läuft, können (mittels `PROCESS PAGE-Statement`) sowohl Maps als auch Rich UI-Pages verwendet werden.

Diese Systemvariable kann einen der folgenden Werte enthalten:

| Wert | Beschreibung |
|----------------|--|
| <i>(empty)</i> | Die Anwendung läuft nicht in einem Web Browser. |
| WEB | Die Anwendung läuft über das Natural Web I/O Interface. Die Verwendung des <code>PROCESS PAGE-Statements</code> ist nicht möglich. |
| RICHGUI | Die Anwendung läuft über Natural for Ajax. Die Verwendung des <code>PROCESS PAGE-Statements</code> ist möglich. |

*DEVICE

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Typ des Geräts, von dem Natural aufgerufen wurde. Sie kann einen der folgenden Werte enthalten:

| Wert | Beschreibung |
|-------|---|
| BATCH | Batch-Betrieb. |
| COLOR | 3279-Kompatibilität. 3278-Bildschirm (Gerät mit erweiterter Attribut-Unterstützung). |
| VIDEO | 3270-Bildschirm, PC-Bildschirm, VT- oder X-Terminal oder ein beliebiger UNIX-Terminaltyp. |
| TTY | Teletype- oder anderer Start/Stop-Typ. |

| Wert | Beschreibung |
|---------|--|
| PC | Benutzung von Natural Connection ist aktiviert (durch Profilparameter PC=ON bzw. Terminalkommando %+). |
| BTX | BTX-Gerät. |
| SPOOL | 3270-Drucker. |
| ASYNCH | Asynchrone Session. |
| BROWSER | Web I/O-Bildschirmgerät (Server-Umgebung). |

*GROUP

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable ist nur unter Natural Security relevant. Sie enthält die ID, über die der Benutzer in einer geschützte Library angemeldet angeloggt ist, also die ID, über die er an die Library gelinkt ist. Dies ist entweder die ID der Gruppe, über die der Benutzer gelinkt ist, oder die Benutzerkennung (User-ID) des Benutzers selbst (falls er direkt an die Library gelinkt ist).

*GROUP enthält keinen Wert

- bei einem Logon in eine nicht geschützte Library (wo kein Link verwendet wird),
- wenn Natural Security nicht aktiv ist.

*HARDCOPY

| | |
|----------------------|----|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält die Kennung des Druckers, der verwendet wird, wenn das Terminalkommando %H eingegeben wird.

*INIT-USER

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Benutzerkennung (User-ID) des Benutzers.

Im Online-Modus:

| | |
|------------------|---|
| Com-plete | Der Wert von *INIT-USER ist die Com-plete-Logon-Benutzerkennung. |
| CICS | <p>Der Wert von *INIT-USER wird durch den Parameter USERID des Macros NCIPARM oder NTCICSP (abhängig von der Version des installierten Natural CICS Interface) bestimmt, wobei der erste Subparameter für CICS-Terminal-Sessions, der zweite für Nicht-Terminal-Sessions, d.h. für asynchrone, DPLeD- usw. Sessions und der dritte für DTP-Sessions gilt.</p> <p>Bei Angabe von ANY für einen USERID-Subparameter wird jedes durch EXEC CICS ASSIGN USERID (. .) zurückgegebene Zeichen, außer Leerzeichen, für den relevanten Session-Typ (Terminal- oder Nicht-Terminal oder DTP) als gültig angesehen.</p> <p>Bei Angabe von YES (NCIPARM) oder ON (NTCICSP) für einen USERID-Subparameter wird jedes durch EXEC CICS ASSIGN USERID (. .) zurückgegebene Zeichen, außer Leerzeichen, als gültig angesehen, wenn es nicht mit der CICS-Standard-Benutzerkennung übereinstimmt und (nur bei Terminal-Session) wenn der Benutzer sich in CICS angemeldet hat.</p> <p>Die Angabe von NO (NCIPARM) oder NO (NTCICSP) für einen USERID-Subparameter bedeutet, dass der CICS-Benutzerkennungswert ignoriert wird.</p> <p>Wenn eine CICS-Benutzerkennung ungültig ist oder ignoriert wird, wird die editierte (nicht gepackte) CICS-Task-Nummer bei Nicht-Terminal-CICS-Sessions genommen, d.h. bei asynchronen oder DPLeD usw.; bei Terminal-Sessions wird die 3-Byte-Operator ID genommen, wenn sie nicht leer ist, andernfalls wird die CICS-Terminalkennung genommen; bei DTP-Sessions wird die Pseudo-Terminalkennung genommen.</p> |
| IMS TM | Der Wert von *INIT-USER ist die IMS TM Signon-Benutzerkennung. |
| TSO | Der Wert von *INIT-USER ist die TSO Logon-Benutzerkennung. |
| TIAM | <p>Der Wert von *INIT-USER wird durch den Parameter USERID des Macros NAMTIAM bestimmt:</p> <p>Bei USERID=USER oder NO (Standardeinstellung) enthält die Variable den mit dem LOGON-Kommando angegebenen BS2000-Jobnamen.</p> <p>Wurde kein BS2000-Jobname angegeben, enthält die Variable das gleiche wie bei USERID=SYSTEM (oder YES), nämlich die BS2000-Benutzerkennung.</p> |
| openUTM | <p>*INIT-USER enthält die für die openUTM-Anwendung definierte Benutzerkennung.</p> <p>Sind für die openUTM-Anwendung keine Benutzerkennungen definiert, entspricht der Wert von *INIT-USER dem von *INIT-ID.</p> |

Im Batch-Modus:

*INIT-USER enthält den Namen des Jobs oder die Benutzerkennung, unter dem oder der die Natural-Session läuft.

| | |
|-------------------------|---|
| z/OS | Der Wert von *INIT-USER wird durch den Subparameter USERID des Profilparameters OSP bzw. des Parameter-Makros NTOSP bestimmt: |
| Natural for IMS: | Der Wert von *INIT-USER wird durch den Subparameter USERID des Makros NTIMSPE bestimmt. |
| BMP DLIBATCH | Bei USERID=ON wird der Wert vom Security-Zugriffskontrollblock (ACEE) des verwendeten Security-Pakets (z.B. RACF oder ACF2) genommen. Bei USERID=OFF wird der Wert von *INIT-USER durch den Namen des Jobs bestimmt, unter dem die Natural-Session läuft. Wenn kein Security-Paket verwendet wird, wird der Wert von *INIT-USER vom USER-Parameter der Job-Karte genommen. Wurde dort kein USER-Parameter angegeben, ist der Wert der gleiche wie bei der Einstellung USERID=OFF. |
| z/VSE | Der Wert von *INIT-USER wird durch den Subparameter USERID des Profilparameters VSEP bzw. des Parameter-Makros NTVSEP bestimmt. |

***LANGUAGE**

| | |
|----------------------|----|
| Format/Länge: | I1 |
| Inhalt modifizierbar | Ja |

Diese Systemvariable enthält den Sprachindikator (Sprachcode). Dieser Sprachindikator wird bei Editiermasken von Datumsfeldern sowie bei Natural- oder benutzergeschriebenen Fehlermeldungen, die über INPUT- oder REINPUT-Statements ausgegeben werden, verwendet.

Jedem Sprachcode ist ein einbuchstabiger Code zugewiesen; in sprachabhängigen Anwendungen werden alle Und-Zeichen (&) in Namen von Objekten (z.B. Maps, Dialoge, Helprouninen, Subprogramme) durch diesen einbuchstabigen Code ersetzt. Einzelheiten zum Gebrauch des Und-Zeichens in Objektnamen finden Sie in den Beschreibungen der Statements CALLNAT (*operand1*), FETCH (*operand1*), INCLUDE (*copycode-name*) und INPUT (USING MAP) und beim Session-Parameter HE (*operand1*).

Es stehen Ihnen 60 verschiedene Sprachcodes zur Verfügung. Die Codes sind unten aufgeführt.

Die Systemvariable *LANGUAGE wird vom Natural-Profilparameter ULANG gesetzt, der die für Datums-Editiermasken, Systemmeldungen, Benutzermeldungen, Hilfetexte, Helprouninen und mehrsprachige Masken zu benutzende Sprache festlegt.

Auf Großrechnern benutzt der Compiler zur Bestimmung des Map-Namens immer den aktuellen Wert von `*LANGUAGE`. Zur Laufzeit versucht Natural für Großrechner zuerst die Map mit der aktuellen `*LANGUAGE`-Einstellung zu lesen. Wenn sie nicht gefunden wird, versucht Natural die Map mit der Standard-Sprache zu finden.

Einzelheiten zum Einsatz von Sprachcodes entnehmen Sie dem Abschnitt *Mehrsprachige Benutzeroberflächen im Leitfaden zur Programmierung*.

Sprachcode-Zuordnungen

Den einzelnen Sprachcodes sind die folgenden Sprachen zugeordnet (die rechte Spalte zeigt die jeweiligen einbuchstabigen Codes für sprachabhängige Objektnamen):

- Einbyte-Sprachen (Schreibrichtung von links nach rechts) mit lateinischen Kleinbuchstaben
- Einbyte-Sprachen (Schreibrichtung von links nach rechts) ohne lateinische Kleinbuchstaben
- Einbyte-Sprachen (beide Schreibrichtungen) ohne lateinische Kleinbuchstaben
- Vom Benutzer zugewiesene Sprachen
- Multibyte-Sprachen
- Doppelbyte-Sprachen

Einbyte-Sprachen (Schreibrichtung von links nach rechts) mit lateinischen Kleinbuchstaben

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|-----------------------------------|---|
| 1 | Englisch | 1 |
| 2 | Deutsch | 2 |
| 3 | Französisch | 3 |
| 4 | Spanisch | 4 |
| 5 | Italienisch | 5 |
| 6 | Niederländisch | 6 |
| 7 | Türkisch | 7 |
| 8 | Dänisch | 8 |
| 9 | Norwegisch | 9 |
| 10 | Albanisch | A |
| 11 | Portugiesisch | B |
| 12 | Chinesisch Lateinschrift (Taiwan) | C |
| 13 | Tschechisch | D |
| 14 | Slowakisch | E |
| 15 | Finnisch | F |
| 16 | Ungarisch | G |
| 17 | Isländisch | H |
| 18 | Koreanisch | I |

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|-------------|---|
| 19 | Polnisch | J |
| 20 | Rumänisch | K |
| 21 | Schwedisch | L |
| 22 | Kroatisch | M |
| 23 | Katalanisch | N |
| 24 | Baskisch | O |
| 25 | Afrikaans | P |

Einbyte-Sprachen (Schreibrichtung von links nach rechts) ohne lateinische Kleinbuchstaben

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|----------------------|---|
| 26 | Bulgarisch | Q |
| 27 | Griechisch | R |
| 28 | Japanisch (Katakana) | S |
| 29 | Russisch | T |
| 30 | Serbisch | U |

Einbyte-Sprachen (beide Schreibrichtungen) ohne lateinische Kleinbuchstaben

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|--|---|
| 31 | Arabisch | V |
| 32 | Farsi (Iran) | W |
| 33 | Hebräisch | X |
| 34 | Urdu (Pakistan) | Y |
| 35 | (für zukünftige Verwendung reserviert) | Z |
| 36 | (für zukünftige Verwendung reserviert) | a |
| 37 | (für zukünftige Verwendung reserviert) | b |
| 38 | (für zukünftige Verwendung reserviert) | c |
| 39 | (für zukünftige Verwendung reserviert) | d |
| 40 | (für zukünftige Verwendung reserviert) | e |

Vom Benutzer zugewiesene Sprachen

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|--|---|
| 41 | (Sie können diesem Code eine Sprache zuweisen) | f |
| 42 | (Sie können diesem Code eine Sprache zuweisen) | g |
| 43 | (Sie können diesem Code eine Sprache zuweisen) | h |
| 44 | (Sie können diesem Code eine Sprache zuweisen) | i |
| 45 | (Sie können diesem Code eine Sprache zuweisen) | j |
| 46 | (Sie können diesem Code eine Sprache zuweisen) | k |
| 47 | (Sie können diesem Code eine Sprache zuweisen) | l |
| 48 | (Sie können diesem Code eine Sprache zuweisen) | m |
| 49 | (Sie können diesem Code eine Sprache zuweisen) | n |
| 50 | (Sie können diesem Code eine Sprache zuweisen) | o |

Multibyte-Sprachen

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|--|---|
| 51 | Hindi | p |
| 52 | Malaiisch | q |
| 53 | Thai | r |
| 54 | (für zukünftige Verwendung reserviert) | s |
| 55 | (für zukünftige Verwendung reserviert) | t |
| 56 | (für zukünftige Verwendung reserviert) | u |

Doppelbyte-Sprachen

| Code | Sprache | Zeichencode in sprachspezifischen Objektnamen |
|------|----------------------------------|---|
| 57 | Chinesisch (Volksrepublik China) | v |
| 58 | Chinesisch (Taiwan) | w |
| 59 | Japanisch (Kanji) | x |
| 60 | Koreanisch | y |

*NATVERS

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Natural-Version (außer den Cumulative Fix-Informationen).

Informationen zum Cumulative Fix sind in der Systemvariable [*PATCH-LEVEL](#) enthalten.

Weitere Informationen siehe *Version* im *Glossary*.

*NET-USER

| | |
|----------------------|------|
| Format/Länge: | A253 |
| Inhalt modifizierbar | Nein |

Standardmäßig ist der Wert von *NET-USER identisch mit dem von [*USER](#).

*PARM-USER

| | |
|----------------------|------|
| Format/Länge: | A253 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen des aktuell benutzten Parametermoduls (wenn `PARM=name` nicht als ein dynamischer Parameter angegeben wurde, ist *PARM-USER leer).

*PATCH-LEVEL

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die aktuelle Cumulative Fix-Nummer.

Informationen zur Versionsnummer, siehe [*NATVERS](#) und *Version* im *Glossary*.

*PID

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält eine eindeutige Session-ID.

*SCREEN-IO

| | |
|----------------------|------|
| Format/Länge: | L |
| Inhalt modifizierbar | Nein |

Diese Systemvariable zeigt an, ob eine Bildschirm-Eingabe/Ausgabe möglich ist oder nicht.

Sie kann einen der folgenden Werte enthalten:

| | |
|--------------|---|
| TRUE | Bildschirm-Eingabe/Ausgabe ist möglich. |
| FALSE | Bildschirm-Eingabe/Ausgabe ist nicht möglich. |

Im Falle einer dialog-orientierten Anwendung wird *SCREEN-IO mit TRUE initialisiert.

In einer Natural-Batch-Session wird *SCREEN-IO mit FALSE initialisiert (außer bei einem Remote Development Server).

Wenn Natural als DB2 Stored Procedures Server (*SERVER-TYPE = DB2-SP) oder als RPC Server (*SERVER-TYPE = RPC) *SCREEN-IO gestartet wird, wird *SCREEN-IO auf FALSE gesetzt.

Wenn *SCREEN-IO auf FALSE gesetzt ist und ein Statement ausgeführt wird, das einen Dialog mit dem Benutzer bedingt, gibt Natural den Fehler NAT0723 aus.

*SERVER-TYPE

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable zeigt an, als welcher Server-Typ Natural gestartet wurde.

Sie kann einen der folgenden Werte enthalten:

| | |
|----------------|--------------------------------------|
| DB2-SP | Natural DB2 Stored Procedures Server |
| DEVELOP | Natural Development Server |
| RPC | Natural RPC Server |
| WEBIO | Natural Web I/O Interface Server |

Wenn Natural nicht als Server gestartet wird, wird *SERVER-TYPE auf Leerzeichen gesetzt.



Anmerkung: *SERVER-TYPE bezieht sich auf Natural als Ganzes, nicht auf das gerade ausgeführte Natural-Programm (das innerhalb eines Server-Naturals als Client- oder Server-Programm ausgeführt werden kann).

*UI

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Typ der verwendeten Benutzeroberfläche:

| | |
|------------------|---|
| CHARACTER | Zeichen-orientierte Benutzeroberfläche. |
| GUI | Graphische Benutzeroberfläche. |

***USER**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Benutzerkennung (User-ID), mit der der Benutzer Natural über die Natural-Security-Logon-Prozedur aufgerufen hat.

Wenn der Profilparameter `AUTO=ON` (Automatic Logon) gesetzt ist oder wenn Natural Security nicht aktiv ist, entspricht der Wert von `*USER` dem von `*INIT-USER`.

***USER-NAME**

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Wenn Natural Security installiert ist, enthält diese Systemvariable den Namen des gerade aktiven Natural-Benutzers.

Wenn Natural Security nicht installiert ist, enthält diese Systemvariable den Wert `SYSTEM`.

5

Systemvariablen nach System-Umgebung

| | |
|------------------------|----|
| ■ *CODEPAGE | 52 |
| ■ *HARDWARE | 52 |
| ■ *HOSTNAME | 52 |
| ■ *INIT-ID | 53 |
| ■ *INIT-PROGRAM | 53 |
| ■ *LOCALE | 54 |
| ■ *MACHINE-CLASS | 54 |
| ■ *OPSYS | 54 |
| ■ *OS | 55 |
| ■ *OSVERS | 55 |
| ■ *TP | 56 |
| ■ *TPSYS | 56 |
| ■ *TPVERS | 57 |
| ■ *WINMGR | 57 |
| ■ *WINMGRVERS | 57 |

*CODEPAGE

| | |
|----------------------|------|
| Format/Länge: | A64 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable gibt den IANA-Namen der vorgegebenen (Default-)Codepage zurück, der intern von Natural für Konvertierungen in und von Unicode benutzt wird und der mit dem Natural-Profilparameter `CP` gesetzt wird.



Anmerkung: *CODEPAGE liefert auch die Voreinstellung, wenn bei einem `MOVE ENCODED`-Statement keine Codepage angegeben wird.

Example:

```
IBM01141
```

Um den Inhalt der Systemvariablen *CODEPAGE anzuzeigen, können Sie das Systemkommando `CPINFO` oder die Utility `SYSCP` benutzen.

*HARDWARE

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der Hardware-Plattform, auf der Natural läuft. Dieser Wert wird vom Betriebssystem geliefert.

*HOSTNAME

| | |
|----------------------|------|
| Format/Länge: | A64 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der Maschine, auf der Natural läuft.

***INIT-ID**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

*INIT-ID enthält die Kennung (Terminal-ID) des Terminals, von dem aus Natural aufgerufen wurde (gemäß den Konventionen des eingesetzten TP-Systems).

Im Batch-Betrieb enthält *INIT-ID den Step-Namen des Natural-Jobs.

In einer asynchronen Natural-Session unter Com-plete oder UTM enthält *INIT-ID die Terminalkennung der Task, von der die asynchrone Session gestartet wurde.

In einer asynchronen Session unter CICS enthält *INIT-ID die CICS-Task-Nummer der asynchronen Task.

In einer Server-Umgebung, zum Beispiel Natural Development Server oder Web I/O Interface Server, enthält *INIT-ID die laufende Session-Nummer, die für jede neue Session innerhalb des Servers um ein erhöht wird.

***INIT-PROGRAM**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen des Programms (bzw. der Transaktion), das gerade als Natural ausgeführt wird.

Im Batch-Betrieb unter z/OS enthält *INIT-PROGRAM den Namen des Jobs, unter dem die Natural-Session läuft.

Unter TSO enthält *INIT-PROGRAM den Namen des TSO-Kommandos, unter dem Natural läuft. Falls kein Kommando abgesetzt wurde, zum Beispiel, weil der Task über CALL gestartet wurde, wird stattdessen der Modulname des Natural-Front-Ends verwendet.

In einer Natural Development Server-Umgebung, die Com-plete oder Com-plete/SMARTS verwendet, wird *INIT-PROGRAM entsprechend der Com-plete/SMARTS-Inbetriebsname-Option INSTALLATION gesetzt. Der Standardinhalt ist *****.

***LOCALE**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Sprache und das Land der aktuellen Region, die die Unicode Collation Sequence angibt.

Beispiel:

```
en_US
```

Um den Inhalt der Systemvariablen `*LOCALE` anzuzeigen, können Sie das Systemkommando `CPINFO` oder die Utility `SYSCP` benutzen.

***MACHINE-CLASS**

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen der Maschinenklasse, auf der Natural läuft.

Sie kann einen der folgenden Werte enthalten:

MAINFRAME
PC
UNIX
VMS

***OPSYS**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Natural-Namen des eingesetzten Betriebssystems.

Sie kann einen der folgenden Werte enthalten:

| | | |
|----------|----------|----------|
| ATT_OSX | MSDOS | SUN_SOLA |
| AVIION | MS_OS/2 | SUN_SUNO |
| BS2000 | MVS/ESA | UNISYS 5 |
| BS2/XS | MVS/XA | UNISYS 6 |
| BULL/BOS | NCR 3000 | VSE/ESA |
| DEC-OSF/ | OS | WNT-X86 |
| DOS/VS | OS/400 | |
| DPS300 | RS_6000 | |
| DRS 6000 | SCO | |
| FUJI M73 | SINIX_52 | |
| HP_HPUX | SINIX_54 | |



Anmerkung: Es empfiehlt sich, statt *OPSYS die Systemvariablen *MACHINE-CLASS, *HARDWARE und *OS zu verwenden, da diese eine genauere Unterscheidung der Umgebung, in der Natural läuft, ermöglichen.

*OS

| | |
|----------------------|------|
| Format/Länge: | A32 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen des Betriebssystems, unter dem Natural läuft. Dieser Wert wird vom Betriebssystem geliefert und kann sich ändern.

*OSVERS

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Versionsnummer des Betriebssystems, unter dem Natural läuft. Dieser Wert wird vom Betriebssystem geliefert und kann sich ändern.

***TP**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namen des verwendeten TP-Monitors, unter dem Natural läuft. Dieser Wert wird vom Betriebssystem geliefert und kann sich ändern.

***TPSYS**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Natural-Namen des verwendeten TP-Monitors oder der verwendeten Umgebung.

Sie kann einen der folgenden Werte enthalten:

AIM/DC
CICS
COMPLETE
IMS/DC
OS/400
SERVSTUB (Natural Development Server)
TIAM
TSO
TSS
UTM

Auf Großrechnern ist *TPSYS im Batch-Betrieb leer.

Auf Windows-, UNIX- oder OpenVMS-Plattformen enthält *TPSYS den Wert NONE.

***TPVERS**

| | |
|----------------------|------|
| Format/Länge: | A8 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Version des verwendeten TP-Monitors, unter dem Natural läuft. Dieser Wert wird vom Betriebssystem geliefert und kann sich ändern.

Wenn kein TP-Monitor verwendet wird, ist *TPVERS leer.

***WINMGR**

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Bei Verwendung einer graphischen Benutzeroberfläche enthält diese Systemvariable den Namen des verwendeten Window-Managers (z.B.: MOTIF oder PM).

Bei Verwendung einer zeichenorientierten Benutzeroberfläche ist *WINMGR leer.

Die Art der Benutzeroberfläche ergibt sich aus dem Wert der Systemvariablen [*UI](#).

***WINMGRVERS**

| | |
|----------------------|------|
| Format/Länge: | A16 |
| Inhalt modifizierbar | Nein |

Bei Verwendung einer graphischen Benutzeroberfläche enthält diese Systemvariable die Versionsnummer des verwendeten Window-Managers.

Bei Verwendung einer zeichen-orientierten Benutzeroberfläche ist *WINMGRVERS leer.

Die Art der Benutzeroberfläche ergibt sich aus dem Wert der Systemvariablen [*UI](#).

6 XML-bezogene Systemvariablen

| | |
|----------------------------------|----|
| ■ *PARSE-COL (r) | 60 |
| ■ *PARSE-LEVEL (r) | 60 |
| ■ *PARSE-NAMESPACE-URI (r) | 60 |
| ■ *PARSE-ROW (r) | 61 |
| ■ *PARSE-TYPE (r) | 61 |

Diese Systemvariablen, die für die Benutzung des Statements `PARSE XML` zur Verfügung stehen, gelten nur in der jeweils aktuellen Schleife.

***PARSE-COL (r)**

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Spalte, die der Parser gerade abarbeitet.

In Großrechnerumgebungen weist die Variable immer auf den Anfang des aktuellen Markup-Elements.

Die Notation `(r)` wird nach `*PARSE-COL` benutzt, um das Statement-Label oder die Sourcecode-Zeilenummer des Statements `PARSE XML` zu spezifizieren. Wenn `(r)` nicht angegeben wird, stellt `*PARSE-COL` die Spalte dar, die der Parser in der gerade aktiven Verarbeitungsschleife abarbeitet.

***PARSE-LEVEL (r)**

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Stufe der gerade abgearbeiteten verschachtelten Schleifen.

Die Notation `(r)` wird nach `*PARSE-LEVEL` benutzt, um das Statement-Label oder die Sourcecode-Zeilenummer des Statements `PARSE XML` zu spezifizieren. Wenn `(r)` nicht angegeben wird, stellt `*PARSE-LEVEL` die Stufe dar, die der Parser in der gerade aktiven Verarbeitungsschleife abarbeitet.

***PARSE-NAMESPACE-URI (r)**

| | |
|----------------------|-------------|
| Format/Länge: | A (dynamic) |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält den Namespace-URI des aktuellen Elements oder Attributs, wenn das Element oder die Attribute zu einem Namespace gehören. Wenn der Wert `NAME (operand3)` des Statements `PARSE XML` leer ist, dann gibt es auch keinen Namespace, und `*LENGTH(*PARSE-NAMESPACE-URI)` wird auf 0 gesetzt.

Die Notation (*r*) nach *PARSE-NAMESPACE-URI wird benutzt, um das Statement-Label oder die Sourcecode-Zeilenummer des Statements PARSE XML zu spezifizieren. Wenn (*r*) nicht angegeben wird, stellt *PARSE-NAMESPACE-URI den Namespace-URI des aktuellen Elements oder Attributs in der gerade aktiven Verarbeitungsschleife dar.

*PARSE-ROW (r)

| | |
|----------------------|------|
| Format/Länge: | I4 |
| Inhalt modifizierbar | Nein |

Diese Systemvariable enthält die Zeile, die der Parser gerade abarbeitet.

Die Notation (*r*) nach *PARSE-ROW wird benutzt, um das Statement-Label oder die Sourcecode-Zeilenummer des Statements PARSE XML zu spezifizieren. Wenn (*r*) nicht angegeben wird, stellt *PARSE-ROW die Zeile dar, die der Parser in der gerade aktiven Verarbeitungsschleife abarbeitet.

*PARSE-TYPE (r)

| | |
|----------------------|------|
| Format/Länge: | A1 |
| Inhalt modifizierbar | Nein |

Diese Natural-Systemvariable wird automatisch für jedes abgesetzte PARSE XML-Statement generiert.

Diese Systemvariable enthält den Typ der übergebenen Daten.

Die Notation (*r*) nach *PARSE-TYPE wird benutzt, um das Statement-Label oder die Sourcecode-Zeilenummer des PARSE XML-Statements zu spezifizieren. Wenn (*r*) nicht angegeben wird, stellt *PARSE-TYPE den Typ der übergebenen Daten in der gerade aktiven Verarbeitungsschleife dar.

Mögliche Werte für ASCII-basierte Systeme sind:

| | |
|----|--|
| ? | Verarbeitungsanweisung (aber nicht erstes <?XML ... ?>). |
| ! | Kommentar. |
| C | CDATA-Abschnitt. |
| T | Start-Tag. |
| @ | Attribut (auf Großrechnern: § oder @, je nach Code Page und Terminal Emulation). |
| / | Abschluss-Tag. |
| \$ | Geparste Daten. |

Stichwortverzeichnis

Symbole

- *APPLIC-ID
 - system variable
 - application related, 3
- *APPLIC-NAME
 - system variable
 - application related, 3
- *CODEPAGE
 - system variable
 - system environment related, 52
- *COM
 - system variable
 - application related, 3
- *CONVID
 - system variable
 - application related, 4
- *COUNTER (r)
 - system variable
 - application related, 5
- *CPU-TIME
 - system variable
 - application related, 5
- *CURRENT-UNIT
 - system variable
 - application related, 6
- *CURS-COL
 - system variable
 - input/output related, 32
- *CURS-FIELD
 - system variable
 - input/output related, 32
- *CURS-LINE
 - system variable
 - input/output related, 33
- *CURSOR
 - system variable
 - input/output related, 34
- *DAT4D
 - system variable
 - date, 26
- *DAT4E
 - system variable
 - date, 26
- *DAT4I
 - system variable
 - date, 26
- *DAT4J
 - system variable
 - date, 26
- *DAT4U
 - system variable
 - date, 26
- *DATA
 - system variable
 - application related, 6
- *DATD
 - system variable
 - date, 26
- *DATE
 - system variable
 - date, 26
- *DATG
 - system variable
 - date, 26
- *DATI
 - system variable
 - date, 26
- *DATJ
 - system variable
 - date, 26
- *DATN
 - system variable
 - date, 26
- *DATU
 - system variable
 - date, 26
- *DATV
 - system variable
 - date, 26
- *DATVS
 - system variable
 - date, 27
- *DATX
 - system variable
 - date, 27
- *DEVICE
 - system variable
 - Natural environment related, 40
- *ERROR-LINE
 - system variable
 - application related, 7
- *ERROR-NR
 - system variable
 - application related, 7
- *ERROR-TA
 - system variable
 - application related, 8
- *ETID

- system variable
 - application related, 9
- *GROUP
 - system variable
 - Natural environment related, 41
- *HARDCOPY
 - system variable
 - Natural environment related, 41
- *HARDWARE
 - system variable
 - system environment related, 52
- *HOSTNAME
 - system variable
 - system environment related, 52
- *INIT-ID
 - system variable
 - system environment related, 53
- *INIT-PROGRAM
 - system variable
 - system environment related, 53
- *INIT-USER
 - system variable
 - Natural environment related, 42
- *ISN (r)
 - system variable
 - application related, 9
- *LANGUAGE
 - system variable
 - Natural environment related, 43
- *LBOUND
 - system variable
 - application related, 10
- *LENGTH (field)
 - system variable
 - application related, 12
- *LEVEL
 - system variable
 - application related, 12
- *LIBRARY-ID
 - system variable
 - application related, 12
- *LINE
 - system variable
 - application related, 13
- *LINE-COUNT
 - system variable
 - input/output related, 34
- *LINESIZE
 - system variable
 - input/output related, 35
- *LINEX
 - system variable
 - application related, 13
- *LOAD-LIBRARY-ID
 - system variable
 - application related, 14
- *LOCALE
 - system variable
 - system environment related, 54
- *LOG-LS
 - system variable
 - input/output related, 35
- *LOG-PS
 - system variable
 - input/output related, 35
- *MACHINE-CLASS
 - system variable
 - system environment related, 54
- *NATVERS
 - system variable
 - Natural environment related, 47
- *NET-USER
 - system variable
 - Natural environment related, 47
- *NUMBER (r)
 - system variable
 - application related, 14
- *OCCURRENCE
 - system variable
 - application related, 15
- *OPSYS
 - system variable
 - system environment related, 54
- *OS
 - system variable
 - system environment related, 55
- *OSVERS
 - system variable
 - system environment related, 55
- *PAGE-EVENT
 - system variable
 - application related, 17
- *PAGE-LEVEL
 - system variable
 - input/output related, 18
- *PAGE-NUMBER
 - system variable
 - input/output related, 36
- *PAGESIZE
 - system variable
 - input/output related, 36
- *PARAM-USER
 - system variable
 - Natural environment related, 47
- *PARSE-COL (r)
 - system variable
 - XML related, 60
- *PARSE-LEVEL (r)
 - system variable
 - XML related, 60
- *PARSE-NAMESPACE-URI (r)
 - system variable
 - XML related, 60
- *PARSE-ROW (r)
 - system variable
 - XML related, 61
- *PARSE-TYPE (r)
 - system variable
 - XML related, 61
- *PATCH-LEVEL
 - system variable
 - Natural environment related, 47
- *PF-KEY
 - system variable
 - input/output related, 36
- *PF-NAME
 - system variable
 - input/output related, 37

- *PID
 - system variable
 - Natural environment related, 48
- *PROGRAM
 - system variable
 - application related, 18
- *REINPUT-TYP
 - system variable
 - application related, 18
- *ROWCOUNT
 - system variable
 - application related, 19
- *SCREEN-IO
 - system variable
 - Natural environment related, 48
- *SERVER-TYPE
 - system variable
 - Natural environment related, 49
- *STARTUP
 - system variable
 - application related, 19
- *STEPLIB
 - system variable
 - application related, 20
- *SUBROUTINE
 - system variable
 - application related, 21
- *THIS-OBJECT
 - system variable
 - application related, 21
- *TIMD
 - system variable
 - time, 27
- *TIME
 - system variable
 - time, 27
- *TIME-OUT
 - system variable
 - time, 27
- *TIMESTAMP
 - system variable
 - time, 27
- *TIMN
 - system variable
 - time, 27
- *TIMX
 - system variable
 - time, 28
- *TP
 - system variable
 - system environment related, 56
- *TPSYS
 - system variable
 - system environment related, 56
- *TPVERS
 - system variable
 - system environment related, 57
- *TYPE
 - system variable
 - application related, 21
- *UBOUND
 - system variable
 - application related, 22
- *UI

- system variable
 - Natural environment related, 49
- *USER
 - system variable
 - Natural environment related, 50
- *USER-NAME
 - system variable
 - Natural environment related, 50
- *WINDOW-LS
 - system variable
 - input/output related, 38
- *WINDOW-POS
 - system variable
 - input/output related, 38
- *WINDOW-PS
 - system variable
 - input/output related, 38
- *WINMGR
 - system variable
 - system environment related, 57
- *WINMGRVERS
 - system variable
 - system environment related, 57

B

- *BROWSER-IO
 - system variable
 - Natural environment related, 40

D

- date system variables, 25

S

- system variables, vii
 - application related, 1
 - input/output related, 31
 - Natural environment related, 39
 - system environment related, 51
 - XML related, 59

T

- time system variables, 25

