Natural for z/OS

Systemvariablen

Version 9.2.5

Oktober 2026

ADABAS & NATURAL

This document applies to Natural for z/OS Version 9.2.5 and all subsequent releases.

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

Copyright © 1979-2026 Software GmbH, Darmstadt, Germany and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software GmbH product names are either trademarks or registered trademarks of Software GmbH and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software GmbH and/or its subsidiaries is located at https://softwareag.com/licenses.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software GmbH Products / Copyright and Trademark Notices of Software GmbH Products". These documents are part of the product documentation, located at https://softwareag.com/licenses and/or in the root installation directory of the licensed product(s).

Use of this software is subject to adherence to Software GmbH's licensing conditions and terms. These terms are part of the product documentation, located at https://softwareag.com/licenses and/or in the root installation directory of the licensed product(s).

Document ID: NATMF-NNATVARI-925-20261013DE

Inhaltsverzeichnis

Vorwort vii
1 Über diese Dokumentation 1
Dokumentationskonventionen 2
Online-Informationen und Support 2
Datenschutz 3
2 Über die Natural-Systemvariablen 5
3 Anwendungsbezogene Systemvariablen 7
*APPLIC-ID 9
*APPLIC-NAME 9
*COM 9
*CONVID 10
*COUNTER (r) 11
*CPU-TIME 11
*CPU-TIMEX 12
*CURRENT-UNIT 12
*DATA 13
*EDITOR 13
*ERROR-LINE 14
*ERROR-NR 14
*ERROR-TA 15
*ETID 15
*ISN (r) 15
*LBOUND 16
*LENGTH (field) 18
*LEVEL 18
*LIBRARY-ID 18
*LINE 19
*LINEX 19
*LOAD-LIBRARY-ID 20
*NUMBER (r) 20
*OCCURRENCE 21
*PAGE-EVENT 23
*PAGE-LEVEL 24
*PROGRAM 24
*REINPUT-TYPE 24
*ROWCOUNT 25
*STARTUP 25
*STEPLIB 26
*SUBROUTINE 27
*THIS-OBJECT 27
*TYPE 27
*UBOUND 28
4 Datums- und Zeit-Systemvariablen 31

Verwendung ... 32
DAT - Datums-Systemvariablen ... 32
TIM - Zeit-Systemvariablen ... 33
Beispiel für Datums- und Zeit-Systemvariablen ... 34
5 Eingabe/Ausgabebezogene Systemvariablen ... 37
*CURS-COL ... 38
*CURS-FIELD ... 38
*CURS-LINE ... 39
*CURSOR ... 40
*LINE-COUNT ... 40
*LINESIZE ... 41
*LOG-LS ... 41
*LOG-PS ... 41
*PAGE-NUMBER ... 42
*PAGESIZE ... 42
*PF-KEY ... 42
*PF-NAME ... 43
*WINDOW-LS ... 44
*WINDOW-POS ... 44
*WINDOW-PS ... 44
6 Systemvariablen mit Bezug zu JSON ... 45
*PARSE-LEVEL (r) for JSON ... 46
*PARSE-INDEX (r) for JSON ... 46
*PARSE-TYPE (r) for JSON ... 47
7 Systemvariablen mit Bezug zur Natural-Umgebung ... 49
*BROWSER-IO ... 50
*DEVICE ... 50
*GROUP ... 51
*HARDCOPY ... 51
*INIT-USER ... 52
*LANGUAGE ... 53
*NATVERS ... 56
*NET-USER ... 57
*PARM-USER ... 57
*PATCH-LEVEL ... 57
*PID ... 57
*SCREEN-IO ... 58
*SERVER-TYPE ... 58
*UI ... 59
*USER ... 59
*USER-NAME ... 59
*UUID ... 60
8 Systemvariablen mit Bezug zur System-Umgebung ... 63
*CODEPAGE ... 64
*HARDWARE ... 64

*HOSTNAME 64
*INIT-ID 65
*INIT-PROGRAM 65
*LOCALE 66
*MACHINE-CLASS 66
*OPSYS 66
*OS 67
*OSVERS 67
*TP 68
*TPSYS 68
*TPVERS 69
*WINMGR 69
*WINMGRVERS 69
9 XML-bezogene Systemvariablen 71
*PARSE-COL (r) 72
*PARSE-LEVEL (r) 72
*PARSE-NAMESPACE-URI (r) 72
*PARSE-ROW (r) 73
*PARSE-TYPE (r) 73
Stichwortverzeichnis 75

Vorwort

Diese Dokumentation beschreibt die Natural-Systemvariablen.

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/Ausgabezogene 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 zu JSON	Systemvariablen, die in Verbindung mit dem `PARSE JSON`-Statement zur Verfügung stehen.
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 XML`-Statement zur Verfügung stehen.

1 Über diese Dokumentation

- Dokumentationskonventionen 2
- Online-Informationen und Support 2
- Datenschutz 3

Dokumentationskonventionen

Konvention	Beschreibung
Fettschrift	>Kennzeichnet Elemente auf einem Bildschirm.
`Nichtproportionale Schrift`	Kennzeichnet Namen und Orte von Diensten im Format *`Ordner.Unterordner.Dienst`*, Programmierschnittstellen (APIs), Namen von Klassen, Methoden und Properties in Java.
`Kursivschrift`	Kennzeichnet: Variablen, für die Sie situations- oder umgebungsspezifische Werte angeben müssen. Neue Begriffe, wenn sie erstmals im Text auftreten. Verweise auf andere Dokumentationsquellen.
`Nichtproportionale Schrift`	Kennzeichnet: Text, den Sie eingeben müssen. Meldungen, die vom System angezeigt werden. Programmcode.
{ }	Zeigt eine Reihe von Auswahlmöglichkeiten an, von denen Sie eine auswählen müssen. Geben Sie nur die innerhalb der geschweiften Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole { } ein.
`|`	Trennt zwei sich gegenseitig ausschließende Auswahlmöglichkeiten in einer Syntaxzeile voneinander ab. Geben Sie eine der Auswahlmöglichkeiten ein. Geben Sie nicht das Symbol `|` ein.
[ ]	Zeigt eine oder mehrere Optionen an. Geben Sie nur die innerhalb der eckigen Klammern vorhandenen Informationen ein. Geben Sie nicht die Klammersymbole [ ] ein.
...	Zeigt an, dass Sie mehrere Auswahlmöglichkeiten desselben Typs eingeben können. Geben Sie nur die Informationen ein. Geben Sie nicht die drei Auslassungspunkte (...) ein.

Online-Informationen und Support

Produktdokumentation

Sie finden die Produktdokumentation auf unserer Dokumentationswebsite unter **https://documentation.softwareag.com**.

Zusätzlich können Sie auch über **https://www.softwareag.cloud** auf die Dokumentation für die Cloud-Produkte zugreifen. Navigieren Sie zum gewünschten Produkt und gehen Sie dann, je nach Produkt, zu „Developer Center“, „User Center“ oder „Documentation“.

Produktschulungen

Sie finden hilfreiches Produktschulungsmaterial auf unserem Lernportal unter **https://knowledge.softwareag.com**.

Tech Community

Auf der Website unserer Tech Community unter **https://techcommunity.softwareag.com** können Sie mit Experten der Software AG zusammenarbeiten. Von hier aus können Sie zum Beispiel:

- Unsere umfangreiche Wissensdatenbank durchsuchen.
- In unseren Diskussionsforen Fragen stellen und Antworten finden.
- Die neuesten Nachrichten und Ankündigungen der Software AG lesen.
- Unsere Communities erkunden.
- Unsere öffentlichen Repositories auf GitHub and Docker unter **https://github.com/softwareag** und **https://hub.docker.com/publishers/softwareag** besuchen und weitere Ressourcen der Software AG entdecken.

Produktsupport

Support für die Produkte der Software AG steht lizenzierten Kunden über unser Empower-Portal unter **https://empower.softwareag.com** zur Verfügung. Für viele Dienstleistungen auf diesem Portal benötigen Sie ein Konto. Wenn Sie noch keines haben, dann können Sie es unter **https://empower.softwareag.com/register** beantragen. Sobald Sie ein Konto haben, können Sie zum Beispiel:

- Produkte, Aktualisierungen und Programmkorrekturen herunterladen.
- Das Knowledge Center nach technischen Informationen und Tipps durchsuchen.
- Frühwarnungen und kritische Alarme abonnieren.
- Supportfälle öffnen und aktualisieren.
- Anfragen für neue Produktmerkmale einreichen.

Datenschutz

Die Produkte der Software AG stellen Funktionen zur Verarbeitung von personenbezogenen Daten gemäß der Datenschutz-Grundverordnung (DSGVO) der Europäischen Union zur Verfügung. Gegebenenfalls sind in der betreffenden Systemverwaltungsdokumentation entsprechende Schritte dokumentiert.

2 Über 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.

Siehe auch die folgenden Abschnitte im Leitfaden zur Programmierung

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

3 Anwendungsbezogene Systemvariablen

- *APPLIC-ID 9
- *APPLIC-NAME 9
- *COM 9
- *CONVID 10
- *COUNTER (r) 11
- *CPU-TIME 11
- *CPU-TIMEX 12
- *CURRENT-UNIT 12
- *DATA 13
- *EDITOR 13
- *ERROR-LINE 14
- *ERROR-NR 14
- *ERROR-TA 15
- *ETID 15
- *ISN (r) 15
- *LBOUND 16
- *LENGTH (field) 18
- *LEVEL 18
- *LIBRARY-ID 18
- *LINE 19
- *LINEX 19
- *LOAD-LIBRARY-ID 20
- *NUMBER (r) 20
- *OCCURRENCE 21
- *PAGE-EVENT 23
- *PAGE-LEVEL 24
- *PROGRAM 24
- *REINPUT-TYPE 24
- *ROWCOUNT 25
- *STARTUP 25
- *STEPLIB 26
- *SUBROUTINE 27

- *THIS-OBJECT 27
- *TYPE 27
- *UBOUND 28

*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 unter *Dialog-Gestaltung* im *Leitfaden zur Programmierung*:

- *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-Zeilennummer 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

Die Systemvariable `*CPU-TIME` enthält die aktuell vom Natural-Prozess beanspruchte CPU-Zeit in Einheiten von 10 ms.

Bei folgenden TP-Monitor-Systemen auf z/OS enthält diese Systemvariable den Wert Null:

- CICS
- IMS TM

*CPU-TIMEX

Format/Länge:	P13
Inhalt modifizierbar	Nein

Die Systemvariable `*CPU-TIMEX` enthält die aktuell vom Natural-Prozess beanspruchte CPU-Zeit.

- Unter TSO und im Batch-Modus wird der Wert in Einheiten von 1 Mikrosekunde zurückgegeben.
- Unter Com-plete wird der Wert in Einheiten von 10 Millisekunden zurückgegeben.

Bei folgenden TP-Monitor-Systemen auf z/OS enthält diese Systemvariable den Wert Null:

- CICS
- IMS TM

*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.

*EDITOR

Format/Länge:	L
Inhalt änderbar	Nein

Die Systemvariable `*EDITOR` zeigt an, ob der Natural-Programm-Editor, der Datenbereich-Editor *(Data Area Editor)* und der Masken-Editor *(Map Editor)* eingeschaltet (aktiviert) sind und benutzt werden können.

Die Systemvariable `*EDITOR` kann einen der folgenden Werte enthalten:

Wert	Beschreibung
TRUE	Die Editoren sind eingeschaltet (aktiviert).
FALSE	Die Editoren sind nicht eingeschaltet (deaktiviert) und können nicht benutzt werden.

Weitere Informationen siehe *NaturalONE als Standard-Entwicklungsumgebung* in der *Editoren*-Dokumentation.

Anmerkung: Die Editoren sind auch eingeschaltet (aktiviert), wenn Natural ISPF installiert ist.

*ERROR-LINE

Format/Länge:	N4
Inhalt modifizierbar	Nein

Diese Systemvariable enthält die Sourcecode-Zeilennummer 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.

Weitere Informationen siehe *Verwendung eines Fehlertransaktionsprogramms* im *Leitfaden zur Programmierung*.

*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-Zeilennummer 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. Wird auf eine Extended ESDS VSAM-Datei zugegriffen, dann wird Format/Länge von `*ISN` zur Unterstützung der 8-Byte-RBA als P19 generiert.
SQL-Datenbanken	`*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
operand1			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:

$$dim = \left\{ \begin{matrix} 1 \\ 2 \\ 3 \\ * \end{matrix} \right\}$$

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 eindimenisionalen 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 abzuprüfen:

```
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, Helproutine 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-Zeilennummer 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.

*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
operand1			A			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:

$$dim = \left\{ \begin{matrix} 1 \\ 2 \\ 3 \\ * \end{matrix} \right\}$$

Erklärung:

1	Eindimensionales Array. Dies ist die Voreinstellung, wenn *dim* 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 Datenbanktabelle 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 Fehlerschleifensitutation 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 `SYS`*xxx* 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	Objekttyp
`PROGRAM`	Programm
`FUNCTION`	Function
`SUBPROGRAM`	Subprogramm
`SUBROUTINE`	Subroutine
`HELPROUTINE`	Helproutine
`MAP`	Map

Wert	Objekttyp
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
operand1			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:

$$dim = \left\{ \begin{array}{c} 1 \\ 2 \\ 3 \\ * \end{array} \right\}$$

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 eindimenisionalen 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 abzuprüfen:

```
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`.

4 Datums- und Zeit-Systemvariablen

- Verwendung 32
- *DAT* - Datums-Systemvariablen 32
- *TIM* - Zeit-Systemvariablen 33
- Beispiel für Datums- und Zeit-Systemvariablen 34

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	DD*monthname*YYYY (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).

Zeit-Variable	Format/Länge	Erklärung
*TIMD (*r*)	N7	Kann nur in Verbindung mit einem vorangegangenen SETTIME-Statement verwendet werden. Enthält die Zeit, die seit der Ausführung des SETTIME-Statements verstrichen ist. Format: HHIISST (*). (*r*) steht für das Statement-Label bzw. die Sourcecode-Zeilennummer des SETTIME-Statement, das als Basis für *TIMD verwendet wird. Wenn *TIMD benutzt wird, ohne zuvor ein SETTIME-Statement auszuführen, dann wird als Wert Null zurückgegeben.
*TIME	A10	Enthält die aktuelle Uhrzeit im Format HH:II:SS.T (*).
*TIME-OUT	N5	*TIME-OUT enthält die Anzahl der verbleibenden Sekunden, bevor die aktuelle Transaktion wegen Zeitüberschreitung abgebrochen wird (nur verfügbar unter Natural Security). *TIME-OUT ist 0, wenn kein datenverändernder Datenbankzugriff erfolgt. *TIME-OUT kommt nur zum Tragen, wenn ein FIND-, READ- oder GET-Statement einen Datensatz zum Zwecke des Änderns oder Löschens liest. *TIME-OUT wird auf 0 zurückgesetzt, wenn ein END TRANSACTION- oder BACKOUT TRANSACTION-Statement ausgeführt wird.
*TIMESTMP	B8	Hardware-interner Zeitzähler (Store Clock), dessen Wert von der Großrechner-Maschineninstruktion STCK (im Store Clock-Format) zur Verfügung gestellt wird. Weitere Informationen zur Bedeutung siehe IBM-Dokumentation.

Zeit-Variable	Format/Länge	Erklärung
		Die Verarbeitung von Store Clock-Werten wird im *Leitfaden zur Programmierung > Weitere Programmieraspekte > Verarbeitung von Store Clock-Werten* beschrieben.
`*TIMESTMPX`	B16	Hardware-interner Zeitzähler (Store Clock), dessen Wert von der Großrechner-Maschineninstruktion `STCK` (im Store Clock Extended-Format) zur Verfügung gestellt wird. Weitere Informationen zur Bedeutung siehe IBM-Dokumentation. Die Verarbeitung von erweiterten Store Clock-Werten wird im *Leitfaden zur Programmierung > Weitere Programmieraspekte > Verarbeitung von Store Clock-Werten* beschrieben.
`*TIMN`	N7	Enthält die aktuelle Uhrzeit im Format `HHIISST` (*).
`*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 /
  'TIME IN STORE CLOCK FORMAT '   *TIMESTMP /
  'TIME IN EXT. STCK FORMAT   '   *TIMESTMPX /
*
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   02.12.2021
DATE IN FORMAT DD/MM/YYYY   02/12/2021
DATE IN FORMAT DD-MON-YYYY  02-Dec-2021
DATE IN FORMAT DDMONYYYY    02Dec2021
DATE IN GREGORIAN FORM      02December  2021
DATE IN FORMAT YYYY-MM-DD   2021-12-02
DATE IN FORMAT YYYYDDD      2021336
DATE IN FORMAT YYYYMMDD     20211202
DATE IN FORMAT MM/DD/YYYY   12/02/2021
DATE IN INTERNAL FORMAT     2021-12-02


TIME IN FORMAT HH:II:SS.T   14:12:43.2
TIME IN FORMAT HHIISST      1412432
TIME IN INTERNAL FORMAT     14:12:43
TIME IN STORE CLOCK FORMAT  DAB39D476A7E3405
TIME IN EXT. STCK FORMAT    00DAB39D476A7E340500000008150001                    ↩


CURRENT DATE 2021-12-02   CURRENT DATE + 14 DAYS    2021-12-16
CURRENT TIME 14:12:43     CURRENT TIME + 10 SECONDS 14:12:53
```

5 Eingabe/Ausgabebezogene Systemvariablen

- *CURS-COL 38
- *CURS-FIELD 38
- *CURS-LINE 39
- *CURSOR 40
- *LINE-COUNT 40
- *LINESIZE 41
- *LOG-LS 41
- *LOG-PS 41
- *PAGE-NUMBER 42
- *PAGESIZE 42
- *PF-KEY 42
- *PF-NAME 43
- *WINDOW-LS 44
- *WINDOW-POS 44
- *WINDOW-PS 44

*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`.

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.

6 Systemvariablen mit Bezug zu JSON

- *PARSE-LEVEL (r) for JSON 46
- *PARSE-INDEX (r) for JSON 46
- *PARSE-TYPE (r) for JSON 47

Dieses Dokument behandelt folgende Systemvariablen:

Die Systemvariablen, die bei Verwendung des Statements `PARSE JSON` zur Verfügung stehen, gelten nur im Kontext der aktuellen Verarbeitungsschleife.

*PARSE-LEVEL (r) for JSON

Format/Länge:	I4
Inhalt modifizierbar	Nein

Diese Systemvariable enthält die Ebene der zurzeit verschachtelten Elemente.

Die Notation `(r)` nach `*PARSE-LEVEL` wird verwendet, um das Statement-Label oder die Quellcode-Zeilennummer des `PARSE JSON`-Statements anzugeben. Wenn `(r)` nicht angegeben wird, bezeichnet `*PARSE-LEVEL` die Ebene, auf der der Parser in der gerade aktiven Verarbeitungsschleife arbeitet.

*PARSE-INDEX (r) for JSON

Format/Länge:	I4
Inhalt modifizierbar	Nein

Diese Systemvariable enthält den Index des zurzeit bearbeiteten Array-Elements.

Die Notation `(r)` nach `*PARSE-INDEX` wird verwendet, um das Statement-Label oder die Quellcode-Zeilennummer des `PARSE JSON`-Statements anzugeben. Wenn `(r)` nicht angegeben ist, enthält `*PARSE-INDEX` den Index des Array-Elements, an dem der Parser in der gerade aktiven Verarbeitungsschleife arbeitet.

Hinweis: Die Systemvariable `*PARSE-INDEX` ist nur auf Array-Elemente anwendbar und ist spezifisch für das `PARSE JSON`-Statement.

*PARSE-TYPE (r) for JSON

Format/Länge:	A1
Inhalt modifizierbar	Nein

Diese Natural-Systemvariable wird automatisch für jedes ausgegebene `PARSE JSON`-Statement erstellt.

Diese Systemvariable enthält den Typ der gelieferten Daten.

Die Notation `(r)` nach `*PARSE-TYPE` wird verwendet, um das Statement-Label oder die Quellcode-Zeilennummer des `PARSE JSON`-Statements anzugeben. Wenn `(r)` nicht angegeben ist, steht `*PARSE-TYPE` für den Typ der gelieferten Daten in der gerade aktiven Verarbeitungsschleife.

Mögliche Werte für ASCII-basierte Systeme:

<	Anfang einer Objekt-Daten-Struktur
>	Ende einer Objekt-Daten-Struktur
(	Anfang einer Array-Datenstruktur
)	Ende einer Array-Datenstruktur
A	Daten-Element vom Typ String
I	Daten-Element vom Typ Numerisch
N	Daten-Element vom Typ Null
L	Daten-Element vom Typ Boolesch
K	Schlüssel (oder) Daten-Name

7 Systemvariablen mit Bezug zur Natural-Umgebung

- *BROWSER-IO 50
- *DEVICE 50
- *GROUP 51
- *HARDCOPY 51
- *INIT-USER 52
- *LANGUAGE 53
- *NATVERS 56
- *NET-USER 57
- *PARM-USER 57
- *PATCH-LEVEL 57
- *PID 57
- *SCREEN-IO 58
- *SERVER-TYPE 58
- *UI 59
- *USER 59
- *USER-NAME 59
- *UUID 60

*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
(empty)	Die Anwendung läuft nicht in einem Web Browser.
WEB	Die Anwendung läuft über das Natural Web I/O Interface. Die Verwendung des `PROCESS PAGE`-Statements ist nicht möglich.
RICHGUI	Die Anwendung läuft über Natural for Ajax. Die Verwendung des `PROCESS PAGE`-Statements ist möglich.

*DEVICE

Format/Länge:	A8
Inhalt änderbar	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).

Anmerkung: In einer Server-Umgebung enthält *DEVICE den Wert BROWSER. Das gilt selbst dann, wenn der Server in einer Batch-Umgebung läuft. Der Grund dafür ist, dass das Gerät, das die Ausgabe macht, der Browser ist.

*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	Der Wert von `*INIT-USER` wird durch den Parameter `USERID` des Macros `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. 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. 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. Die Angabe von `NO` (`NCIPARM`) oder `NO` (`NTCICSP`) für einen `USERID`-Subparameter bedeutet, dass der CICS-Benutzerkennungswert ignoriert wird. 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.
IMS TM	Der Wert von `*INIT-USER` ist die IMS TM Signon-Benutzerkennung.
TSO	Der Wert von `*INIT-USER` ist die TSO Logon-Benutzerkennung.

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: **BMP** **DLIBATCH**	Der Wert von `*INIT-USER` wird durch den Subparameter `USERID` des Makros `NTIMSPE` bestimmt. Der Wert von `*INIT-USER` wird durch den Subparameter `USERID` des Makros `NTIMSPE` bestimmt. 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`.

*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 aus einem Zeichen bestehender Code zugewiesen; in sprachabhängigen Anwendungen werden alle Und-Zeichen (&) in Namen von Objekten (z.B. Maps, Dialoge, Helproutinen, Subprogramme) durch diesen Ein-Zeichen-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, Helproutinen 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 for z/OS 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
19	Polnisch	J
20	Rumänisch	K
21	Schwedisch	L
22	Kroatisch	M

Code	Sprache	Zeichencode in sprachspezifischen Objektnamen
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

Code	Sprache	Zeichencode in sprachspezifischen Objektnamen
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`.

*UUID

Format/Länge:	B16
Inhalt modifizierbar	Nein

Die Systemvariable `*UUID` enthält eine binäre (16-Byte-)Zeichenkette, die einen Universally Unique Identifier (UUID) darstellt.

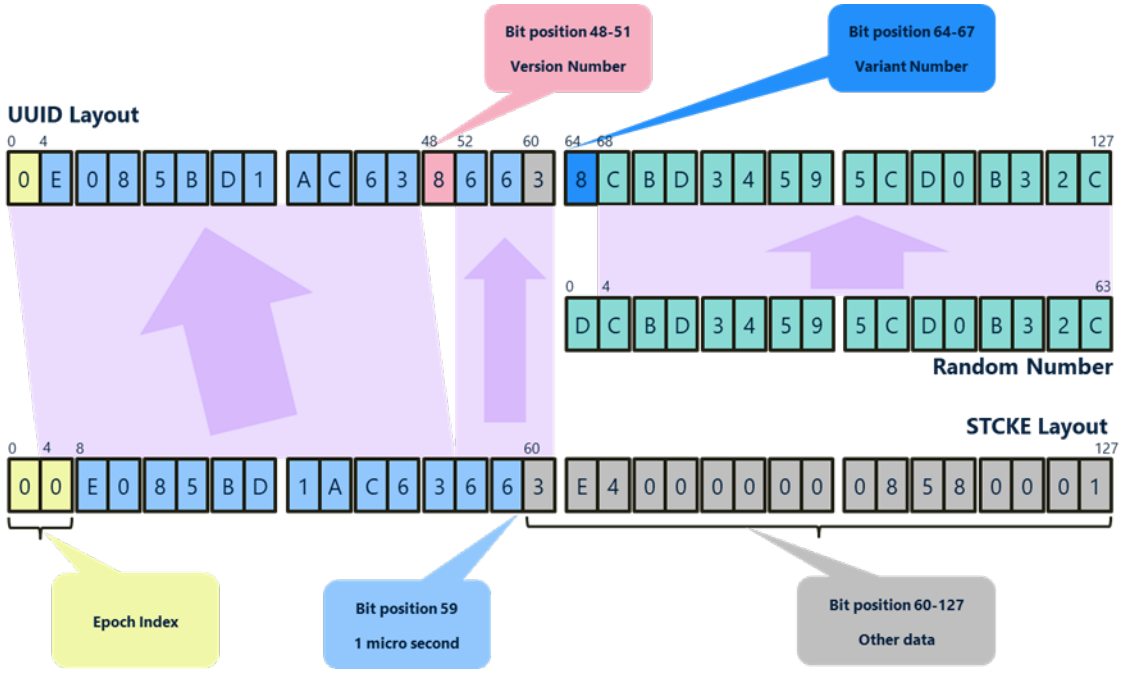
Natural generiert `*UUID` als UUID der Version 8, entsprechend den Spezifikationen von RFC 9562.

`*UUID`-Struktur:

Die UUID wird anhand der folgenden Informationen erzeugt.

- **Erweiterter TOD-Clock-Wert**
  Dieser Wert wird mit der Anweisung STCKE (Store Clock Extended) ermittelt und stellt einen Zeitstempel dar, der auf einer Zeitperiode ab dem 1. Januar 1900 basiert.
- **Zufallswert**
  Die Zufallszahl wird aus dem STCKE-Wert, der CPU-Seriennummer und dem LPAR-Namen abgeleitet, zusammen mit zufälligen XORs und Primzahl-basierten Multiplikationen, um die Entropie zu erhöhen.
- 4 Bits, die die Versionsnummer darstellen
- 4 Bits, die die Variantennummer darstellen

Beispiel:


Bit position 48-51
Version Number
Bit position 64-67
Variant Number
UUID Layout
0 4 48 52 60 64 68 127
0 E 0 8 5 B D 1 A C 6 3 8 6 6 3 8 C B D 3 4 5 9 5 C D 0 B 3 2 C
0 4 63
D C B D 3 4 5 9 5 C D 0 B 3 2 C
Random Number
STCKE Layout
0 4 8 60 127
0 0 E 0 8 5 B D 1 A C 6 3 6 6 3 E 4 0 0 0 0 0 0 0 8 5 8 0 0 0 1
Epoch Index
Bit position 59
1 micro second
Bit position 60-127
Other data

8 Systemvariablen mit Bezug zur System-Umgebung

- *CODEPAGE 64
- *HARDWARE 64
- *HOSTNAME 64
- *INIT-ID 65
- *INIT-PROGRAM 65
- *LOCALE 66
- *MACHINE-CLASS 66
- *OPSYS 66
- *OS 67
- *OSVERS 67
- *TP 68
- *TPSYS 68
- *TPVERS 69
- *WINMGR 69
- *WINMGRVERS 69

*CODEPAGE

Format/Länge:	A64
Inhalt modifizierbar	Nein

Diese Systemvariable gibt den IANA-Namen der Standard-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.

Beispiel:

```
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 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-Inbetriebname-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

*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	SINIX_54
AVIION	MS_OS/2	SUN_SOLA
BULL/BOS	MVS/ESA	SUN_SUNO
DEC-OSF/	NCR 3000	UNISYS 6
DPS300	OS	WNT-X86
DRS 6000	RS_6000	
FUJI M73	SCO	
HP_HPUX	SINIX_52	

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)
`TSO`
`TSS`

Auf Großrechnern ist `*TPSYS` im Batch-Betrieb leer.

Auf Windows- oder UNIX-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`.

9 XML-bezogene Systemvariablen

- *PARSE-COL (r) 72
- *PARSE-LEVEL (r) 72
- *PARSE-NAMESPACE-URI (r) 72
- *PARSE-ROW (r) 73
- *PARSE-TYPE (r) 73

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-Zeilennummer 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-Zeilennummer 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-Zeilennummer 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-Zeilennummer 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-Zeilennummer 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, 9
*APPLIC-NAME
system variable
application related, 9
*CODEPAGE
system variable
system environment related, 64
*COM
system variable
application related, 9
*CONVID
system variable
application related, 10
*COUNTER (r)
system variable
application related, 11
*CPU-TIME
system variable
application related, 11
*CPU-TIMEX
system variable
application related, 12
*CURRENT-UNIT
system variable
application related, 12
*CURS-COL
system variable
input/output related, 38
*CURS-FIELD
system variable
input/output related, 38
*CURS-LINE
system variable
input/output related, 39
*CURSOR
system variable
input/output related, 40
*DAT4D
system variable
date, 32
*DAT4E
system variable
date, 32
*DAT4I
system variable
date, 32
*DAT4J
system variable
date, 32
*DAT4U
system variable
date, 32
*DATA
system variable
application related, 13
*DATD
system variable
date, 32
*DATE
system variable
date, 32
*DATG
system variable
date, 32
*DATI
system variable
date, 32
*DATJ
system variable
date, 32
*DATN
system variable
date, 32
*DATU
system variable
date, 32
*DATV
system variable
date, 32
*DATVS
system variable
date, 33
*DATX
system variable
date, 33
*DEVICE
system variable
Natural environment related, 50
*EDITOR
system variable
application related, 13
*ERROR-LINE
system variable
application related, 14
*ERROR-NR

system variable
application related, 14
*ERROR-TA
system variable
application related, 15
*ETID
system variable
application related, 15
*GROUP
system variable
Natural environment related, 51
*HARDCOPY
system variable
Natural environment related, 51
*HARDWARE
system variable
system environment related, 64
*HOSTNAME
system variable
system environment related, 64
*INIT-ID
system variable
system environment related, 65
*INIT-PROGRAM
system variable
system environment related, 65
*INIT-USER
system variable
Natural environment related, 52
*ISN (r)
system variable
application related, 15
*LANGUAGE
system variable
Natural environment related, 53
*LBOUND
system variable
application related, 16
*LENGTH (field)
system variable
application related, 18
*LEVEL
system variable
application related, 18
*LIBRARY-ID
system variable
application related, 18
*LINE
system variable
application related, 19
*LINE-COUNT
system variable
input/output related, 40
*LINESIZE
system variable
input/output related, 41
*LINEX
system variable
application related, 19
*LOAD-LIBRARY-ID
system variable
application related, 20
*LOCALE
system variable
system environment related, 66
*LOG-LS
system variable
input/output related, 41
*LOG-PS
system variable
input/output related, 41
*MACHINE-CLASS
system variable
system environment related, 66
*NATVERS
system variable
Natural environment related, 56
*NET-USER
system variable
Natural environment related, 57
*NUMBER (r)
system variable
application related, 20
*OCCURRENCE
system variable
application related, 21
*OPSYS
system variable
system environment related, 66
*OS
system variable
system environment related, 67
*OSVERS
system variable
system environment related, 67
*PAGE-EVENT
system variable
application related, 23
*PAGE-LEVEL
system variable
input/output related, 24
*PAGE-NUMBER
system variable
input/output related, 42
*PAGESIZE
system variable
input/output related, 42
*PARM-USER
system variable
Natural environment related, 57
*PARSE-COL (r)
system variable
XML related, 72
*PARSE-INDEX (r)
system variable
JSON related, 46
*PARSE-LEVEL (r)
system variable
JSON related, 46
XML related, 72
*PARSE-NAMESPACE-URI (r)
system variable
XML related, 72
*PARSE-ROW (r)
system variable
XML related, 73
*PARSE-TYPE (r)
system variable

JSON related, 47
XML related, 73
*PATCH-LEVEL
system variable
Natural environment related, 57
*PF-KEY
system variable
input/output related, 42
*PF-NAME
system variable
input/output related, 43
*PID
system variable
Natural environment related, 57
*PROGRAM
system variable
application related, 24
*REINPUT-TYP
system variable
application related, 24
*ROWCOUNT
system variable
application related, 25
*SCREEN-IO
system variable
Natural environment related, 58
*SERVER-TYPE
system variable
Natural environment related, 58
*STARTUP
system variable
application related, 25
*STEPLIB
system variable
application related, 26
*SUBROUTINE
system variable
application related, 27
*THIS-OBJECT
system variable
application related, 27
*TIMD
system variable
time, 33
*TIME
system variable
time, 33
*TIME-OUT
system variable
time, 33
*TIMESTMP
system variable
time, 33
*TIMN
system variable
time, 34
*TIMX
system variable
time, 34
*TP
system variable
system environment related, 68
*TPSYS
system variable
system environment related, 68
*TPVERS
system variable
system environment related, 69
*TYPE
system variable
application related, 27
*UBOUND
system variable
application related, 28
*UI
system variable
Natural environment related, 59
*USER
system variable
Natural environment related, 59
*USER-NAME
system variable
Natural environment related, 59
*UUID
system variable
application related, 60
*WINDOW-LS
system variable
input/output related, 44
*WINDOW-POS
system variable
input/output related, 44
*WINDOW-PS
system variable
input/output related, 44
*WINMGR
system variable
system environment related, 69
*WINMGRVERS
system variable
system environment related, 69

B

*BROWSER-IO
system variable
Natural environment related, 50

D

date system variables, 31

S

system variables, vii
application related, 7
input/output related, 37
JSON related, 45
Natural environment related, 49
system environment related, 63
XML related, 71

T

time system variables, 31