

Natural for OpenVMS

Release Notes for Natural Version 6.3 for OpenVMS

Version 6.3.8 for OpenVMS

February 2010

This document applies to Natural Version 6.3.8 for OpenVMS.

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

Copyright © 1984-2010 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, United States of America, and/or their licensors.

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

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

This software may include portions of third-party products. For third-party copyright notices and license terms, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". This document is part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

Table of Contents

1 Release Notes for Natural Version 6.3 for OpenVMS	1
2 Supported Operating Systems	3
3 General Information	5
License Key File Handling	6
Migrating Applications to Version 6.3	6
Natural Single Point of Development	7
Changes in the Documentation	7
Changes in Terminology	7
Unsupported Features in the Documentation	7
Examples	8
Last-Minute Changes	8
4 New Features	9
Installation	10
Unicode Support in Natural	10
Adabas 6 (Open Systems) and Adabas 8 (Mainframe) Support	10
Object Types	11
Resources	11
X-Arrays	11
Centralized DDM Storage	11
Statements	11
System Functions	12
Parameters	12
System Variables	14
System Commands	15
Constants	15
Utilities	15
Application Programming Interfaces	16
5 Changes and Enhancements	17
International Support	18
Natural System Files	18
Editors	19
Configuration Utility	19
Other Utilities	21
Parameters	23
Statements	24
System Functions	27
System Variables	28
System Commands	29
Application Programming Interfaces	30
Error Messages Corresponding to Adabas Response Codes	30
Natural Web Interface	30
XML Toolkit	35
Natural Connection	35

Miscellaneous	36
6 Known Issues	39
Batch Mode	40
Error Messages	40
Terminal Command %=	41
Usage of Database Field Short Names	41
Interrupting a Running Natural Application	41
Natural Native Interface	41
7 Natural Remote Procedure Call (RPC)	43
New Documentation Structure and Contents	44
Migration	44
New Features	45
Enhanced Features	48
Changed Features	50
SYSRPC Utility Enhancements	51
Changes to the Service Directory	53
Limitations on OpenVMS	53
8 Natural Security	55
General Information	56
Administrator Services	58
User Profiles	61
Library Profiles	61
Utility Profiles	63
RPC Server Profiles	65
Application Programming Interfaces	66
SECULD2/SECLOAD Transfer Programs	68
Logon Procedure	70
Functional Security	70
Limitations on OpenVMS	70
9 Removed Features	73
Remote Debugging	74
Versioning Software	74
Natural Web Interface	74
Utilities NATUNLD, NATLOAD and SYSTRANS	75
Profile Parameter V22COMP	75
DDM Server	75
Multi-Fetch Disabling	75
Profile Parameter DSLM - Size Limitation	75
Profile Parameter LFILMAX - Maximum Number of Dynamic Specifications of LFILE Parameter	76
Entire Screen Builder Interface	76
10 Notice of Future Changes	77
Default Setting for Adabas Database Driver	78
Handling of the TF Parameter	78
11 Release Information for Natural Version 6.3.5	79

New Features	80
Changes and Enhancements	81
Known Issues	82
Natural Remote Procedure Call (RPC)	82
Natural Security	85
Documentation	90
12 Release Information for Natural Version 6.3.6	91
Changes and Enhancements	92
Natural Remote Procedure Call	92
Natural Security	93
Notice of Future Changes	93
13 Release Information for Natural Version 6.3.7	95
New Features	96
Changes and Enhancements	96
Natural Remote Procedure Call (RPC)	98
Natural Security	99
Known Incompatibilities	102
14 Release Information for Natural Version 6.3.8	105
New Features	106
Changes and Enhancements	106
Natural Remote Procedure Call (RPC)	108
Notice of Future Changes	109









1 Release Notes for Natural Version 6.3 for OpenVMS






These Release Notes summarize the new features and enhancements that are provided with Natural Version 6.3 (as compared with the last released version which was Natural Version 5.1.1).

In addition to the new features and enhancements described in these Release Notes, this Natural version includes all error corrections, modifications and enhancements provided with previous Natural versions.

For background information, you can also refer to the Release Notes of previous Natural versions. They are available in the archive of the current Natural Documentation DVD.

This documentation is organized under the following headings:

Natural Version 6.3.4		
	Supported Operating Systems	An overview of the operating systems on which this Natural version can be installed.
	General Information	Information that applies in general to this Natural version.
	New Features	An overview of the new features in this Natural version.
	Changes and Enhancements	Information on changes in this Natural version.
	Known Issues	Additional information that you should be aware of after having installed Natural.
	Natural Remote Procedure Call (RPC)	Information on enhancements and changes to Natural Remote Procedure Call.
	Natural Security	Information on enhancements and changes to Natural Security.
	Removed Features	Information on the features that are no longer available with this Natural version.

Natural Version 6.3.4		
	Notice of Future Changes	Information on features that will change in future Natural versions.
Natural Version 6.3.5		
	Release Information for Natural Version 6.3.5	An overview of the new and changed features in Natural Version 6.3.5.
Natural Version 6.3.6		
	Release Information for Natural Version 6.3.6	An overview of the new and changed features in Natural Version 6.3.6.
Natural Version 6.3.7		
	Release Information for Natural Version 6.3.7	An overview of the new and changed features in Natural Version 6.3.7.
Natural Version 6.3.8		
	Release Information for Natural Version 6.3.8	An overview of the new and changed features in Natural Version 6.3.8.

2 Supported Operating Systems

The operating system listed below is supported with this Natural version.

- Itanium I64 with OpenVMS Version 8.3-1H1.

See also *Product Requirements* in the *Installation* documentation.

3 General Information

- License Key File Handling 6
- Migrating Applications to Version 6.3 6
- Natural Single Point of Development 7
- Changes in the Documentation 7
- Changes in Terminology 7
- Unsupported Features in the Documentation 7
- Examples 8
- Last-Minute Changes 8

License Key File Handling

A path to a valid license key file is required during the installation of Natural. The license key file is an XML file which is delivered by e-mail.

Separate license key files are required for Natural Security and Natural Development Server.

For further information, see the *Installation* documentation.

Migrating Applications to Version 6.3

Applications that were created with Natural for OpenVMS Version 3.1 and above can be executed with Version 6.3 with the following restrictions:

Natural for OpenVMS Version 6.3 supports Unicode. For this reason, Natural has internally been Unicode-enabled which means that many structures containing strings have Unicode format now. Therefore, existing data which is available in code page format will be converted internally to Unicode format in several situations. This is especially necessary when loading a Natural source into the source area and when displaying alphanumeric strings at runtime. However, a conversion from code page to Unicode and vice versa can only be performed successfully if the correct code page is used for conversion. If all objects and other alphanumeric data are coded in the system code page, no changes are necessary. For more information, see *Migrating Existing Applications* in the *Unicode and Code Page Support* documentation.

The internal Unicode structure will most probably need more memory. If you have defined a low value for the profile parameter `USIZE`, it may be necessary to increase this value.

The `CP` parameter, which was used with Natural RPC, has been renamed to `CPRPC`. A new `CP` parameter is now available which defines the default code page for Natural data. When you are working with Natural RPC and have previously used the `CP` parameter dynamically, you have to change this parameter to `CPRPC`. When you use parameter files from a previous version, you need not change anything; the Configuration Utility automatically migrates `CP` to `CPRPC`. See also [Natural Remote Procedure Call \(RPC\)](#).

The structure of the Natural library directory file `FILEDIR.SAG` has changed. The old format can still be read by Natural Version 6.3. However, it is automatically converted into the new format whenever a modify access is made. Older versions of Natural cannot read the new `FILEDIR.SAG` format. Therefore, it is no longer recommended to share the system file `FUSER` with previous Natural versions. When you have two installations of Natural (for example, version 5.1.1 (or below) and version 6.3), you have to use two different version-specific `FUSER` system files. This is because the format of the Natural library directory file `FILEDIR.SAG` has changed. `FILEDIR.SAG` is automatically converted into the new format the first time you modify an object in the `FUSER`. After `FILEDIR.SAG` has been changed to the new format, Natural Version 5.1.1, which is mentioned in

the above example, is not longer able to process the FUSER. For detailed information, see *Portable Natural System Files* and its subsection *Migrating an Old FILEDIR.SAG File* in the *Operations* documentation.

Natural Single Point of Development

Natural Single Point of Development (SPoD) connects Natural for OpenVMS Version 6.3 to

- a Natural for Windows Version 6.3 client (Natural Studio) via the Natural Development Server (NDV) Version 2.2.

This infrastructure enables the developer to create and maintain Natural applications using the Natural Development Server for OpenVMS to connect to Natural Studio under Windows.

For further information (including any limitations that may apply for your installation), see the separate Natural Single Point of Development documentation and the Natural Development Server documentation.

Changes in the Documentation

A revised and updated documentation set is available with this Natural version.

Changes in Terminology

For consistency reasons, the documentation now distinguishes between “user exits” and “application programming interfaces”. Application programming interfaces is a term used to describe SYSEXT and the USR programs. For the definitions of these terms, refer to the *Glossary*.

“Real batch mode” has been renamed to “batch mode”.

Unsupported Features in the Documentation

In some places, the documentation set for Natural for OpenVMS mentions features which are (currently) not supported with Natural for OpenVMS. Such features are:

- Natural for Ajax and the object type "adapter".
- Entire Access (connecting to an SQL database is thus not possible).
- Natural for Tamino.

Examples

The example library *SYSEXV* provides examples of the new features of Natural since version 5.1.

Last-Minute Changes

Information on last-minute changes made just before the product release can be found in the file *READ_ME_FIRST.TXT* that is provided on the Natural installation medium.

4 New Features

▪ Installation	10
▪ Unicode Support in Natural	10
▪ Adabas 6 (Open Systems) and Adabas 8 (Mainframe) Support	10
▪ Object Types	11
▪ Resources	11
▪ X-Arrays	11
▪ Centralized DDM Storage	11
▪ Statements	11
▪ System Functions	12
▪ Parameters	12
▪ System Variables	14
▪ System Commands	15
▪ Constants	15
▪ Utilities	15
▪ Application Programming Interfaces	16

Installation

When installing Natural and Natural Security, you can now choose ApplinX as an additional package to be installed. You choose this package if you plan to use ApplinX and a browser instead of a standard VT terminal emulation. Starting with this version of Natural, the Entire Screen Builder package is no longer used for ApplinX.

Unicode Support in Natural

Unicode is now supported. The new format U has been introduced and there are new/changed statements, parameters, system variables, etc. For details, see the *Unicode and Code Page Support* documentation. See also the example library SYSEXV which provides examples of the new Unicode features in this Natural version.

The new Natural Web I/O Interface is provided. In a remote development environment, it is used to display output which contains Unicode characters. When used in a production environment, the new Natural Web I/O Interface is required for Unicode programs. In the */appserver* directory of the product CD, there are implementations for both J2EE and .NET servers. See *Natural Web I/O Interface* for further information.

Adabas 6 (Open Systems) and Adabas 8 (Mainframe) Support

With Natural Version 6.3.4, a new database driver ADA2 is delivered that supports features introduced with Adabas 6 (Open Systems) and Adabas 8 (mainframe). This includes support of Adabas large objects fields, Adabas LA fields and extended Adabas buffer lengths. See also *Adabas Database Management Interfaces ADA and ADA2* in the *Programming Guide*.

By default, the “old” database driver ADA is used. This ensures that your Natural applications run as usual; there is no difference to previous Natural versions. Only when you define the database type ADA2 in the Configuration Utility, the new database driver will be used.

Object Types

Function

With this version of Natural, the object type “function” is introduced. See *User-Defined Functions* in the *Programming Guide* for further details.

Resources

As of this version, Natural supports non-Natural files (also known as resources). For more information, see *Using Non-Natural Files - Resource* in the *Programming Guide*.

X-Arrays

With this version of Natural, X-arrays (eXtensible arrays) are supported. For further information on purpose, definition and usage of X-arrays, see *X-Arrays* in the *Programming Guide*.

Centralized DDM Storage

The new system file `FDDM` is a container where all DDMs can be stored. For more information, see *System File FDDM* in the *Operations* documentation.

Statements

The following new Natural statements are provided in this version:

Statement	Description
DEFINE FUNCTION	Create new user-defined functions which may be called instead of operands in the Natural statements.
DEFINE PROTOTYPE	Specify a signature according to a certain function call.
OPTIONS	Used to specify compilation options as parameters for the current Natural programming object.
PARSE XML	Parse XML documents from a Natural program.

System Functions

As of this version, all new system functions are preceded by an asterisk (*) to avoid conflicts with, for example, user-defined variables in existing applications.

The following new Natural system functions are provided in this version:

System Function	Description
*MINVAL/*MAXVAL	Evaluate the minimum/maximum value of all given operand values.
*TRANSLATE	Translate the operand to lower/upper case characters.
*TRIM	Remove leading and/or trailing blanks from a string.

Parameters

The following new Natural profile parameters are provided in this version:

Profile Parameter	Description
CP	Defines the default code page for Natural data. Note: The existing CP parameter (used with Natural Remote Procedure Call) has been renamed to CPRPC.
CPCVERR	Specifies whether a conversion error that occurs when converting from Unicode to code page or from code page to Unicode or from one code page to another code page results in a Natural error or not.
CPOBJIN	Specifies the code page of the batch input file which is defined by the Natural profile parameter CMOBJIN.
CPPRINT	Specifies the code page of the batch output file which is defined by the Natural profile parameter CMPRINT.
CPSYNIN	Specifies the code page of the batch input file for commands which is defined by the Natural profile parameter CMSYNIN.
CPRPC	Specifies the name of the code page used by the EntireX Broker.
D0	Specifies how fields are to be interpreted for display on terminals that support bidirectional data.
EMFM	Activates/deactivates the Edit Mask Free mode at session startup.
FDDM	Defines five subparameters for the Natural system file for DDMs.
FREEGDA	Releases the GDA in Utility Activation.
KCHECK	Checks field declarations in a programming object against a set of critical Natural keywords.
MASKCME	Used to control Natural's compiler: to make the MASK option compatible to MOVE EDITED.

Profile Parameter	Description
MAXYEAR	Allows valid year values of up to 9999 instead of 2699 for date and time values.
MFSET	Specifies whether multi-fetch is used to retrieve records from Adabas databases.
NCFVERS	Enables downward compatibility with Natural Versions lower than Version 6.1 and specifies the protocol version of the Entire Connection format file (.NCF) to be used.
NENTRY	Left/right alignment of numeric field entries.
NOSSLPRX	Specifies the domain(s) to be addressed directly, that is, not via the SSL proxy.
PCHECK	Used to control Natural's compiler: to check parameters that are specified in a CALLNAT statement.
PSIGNF	Internal representation of positive sign of packed numbers.
RCFIND	Determines the action to be taken if Adabas Response Code 113 (requested ISN not found) is returned during the execution of a FIND statement processing loop.
RCGET	Determines the action to be taken if Adabas Response Code 113 (requested ISN not found) is returned during the execution of a GET statement.
RTINT	Specifies whether a running Natural application may be interrupted.
SRETAIN	Specifies that all sources have to be saved in their original encoding format.
SSLPRX	Specifies the URL of the (intranet) SSL proxy server through which all requests have to be routed.
SSLPRXPT	Specifies the port number of the SSL proxy.
SRVWAIT	Specifies the number of seconds the server is to wait for an RPC client request.
SUBCHAR	Substitution character for code page.
SUTF8	Specifies the default format to be used when Natural sources are saved.
THSEP	Dynamic thousands separator. Enables or disables the use of thousands separators at compilation time.
THSEPCH	Thousands separator. Specifies the character to be used as a thousands separator at runtime.
TQMARK	Controls the translation of a quotation mark (") within a Natural text constant.
WEBIO	Specifies whether the terminal emulation or the new Natural Web I/O Interface, which supports Unicode, is used for input and output.

The following new Natural session parameters are provided in this version:

Session Parameter	Description
DL	Specifies the display length for an alphanumeric map field of format U.
CPCVERR	Specifies whether a conversion error that occurs when converting from Unicode to code page or from code page to Unicode or from one code page to another code page results in a Natural error or not.
EMU	Edit mask in Unicode.
ICU	Insertion character in Unicode.
LCU	Leading characters in Unicode.
TCU	Trailing characters in Unicode.

System Variables

The following new Natural system variables are provided in this version:

System Variable	Description
*BROWSER-IO	Indicates that the application is running in a web browser. An application can run in a web browser via the Natural Web I/O Interface.
*CODEPAGE	Returns the IANA name of the code page currently used.
*CURRENT-UNIT	Contains the name of the currently executed unit.
LBOUND	Contains the current lower boundary (index value) of an array for the specified dimension(s) (1, 2 or 3) or for all dimensions (notation).
*LINE	Contains the number of the line currently executed in a Natural object.
*LOCALE	Contains the language and country of the current locale.
*TP	Contains the name of the TP subsystem under which Natural is running.
*TPVERS	Contains the name of the TP subsystem under which Natural is running.
*TYPE	Contains the type of the Natural object which is currently executed.
UBOUND	Contains the current upper boundary (index value) of an array for the specified dimension(s) (1, 2 or 3) or for all dimensions (notation).

The following new Natural system variables are automatically created for each PARSE statement issued.

System Variable	Description
*PARSE-COL	Specifies the column where the parser is currently working at.
*PARSE-LEVEL	Specifies the level of currently nested elements.
*PARSE-NAMESPACE-URI	Specifies the namespace URI of the current element/attribute, if the element/attributes belong to a namespace.
*PARSE-ROW	Specifies the row where the parser is currently working at.
*PARSE-TYPE	Specifies the type of the delivered data.

System Commands

The following new Natural system commands are provided in this version:

System Command	Description
LIST ERR	Displays the file information of error message(s) or error message containers of a library.
LIST RES	Displays the file information about resource object(s) of a library.
RENAME	Rename a Natural programming object. Note: This system command is not supported in a remote environment.
SYSEXV	This command invokes the SYSEXV application with examples of the new features of the current Natural versions.

Constants

The following new Natural constants are provided in this version:

Type of Constant	Description
Unicode text constant (U)	See <i>Unicode Constants</i> in the <i>Programming Guide</i> .
Unicode hexadecimal constant (UH)	See <i>Unicode Hexadecimal Constants</i> in the <i>Programming Guide</i> .

Utilities

The following new Natural utility is provided in this version:

Utility	Description
SYSAPI	Used to locate application programming interfaces (APIs) provided by Natural add-on products. A user interface is only provided via SPoD.

Application Programming Interfaces

The utility SYSEXT provides the following new application programming interfaces (APIs):

API	Description
USR1023N	Convert *TIMESTAMP and other variables containing time information into a different format.
USR2018N	Read Natural object directory.
USR2035N	Set SSL parameter string for communication via RPC.
USR2073N	Ping or terminate an RPC server.
USR2075N	Terminate EntireX Broker Service. Used with Natural Remote Procedure Call (RPC) .
USR4003N	Retrieve Natural stack information (alphanumeric).
USR4004N	Retrieve dynamic Natural profile parameters.
USR4007N	Get/set current value of profile parameter SYNERR.
USR4201N	Define a data area source.
USR4206N	List objects in a library and return directory information. Avoids combined calls of the application programming interfaces USR1055N (list objects in a library) and USR2018N (read Natural object directory) in order to retrieve directory information.
USR4208N	Read or write shared resource.
USR4209N	Return short name of subroutine.
USR4210N	Base64 conversion of alphanumeric and binary bytes.
USR6001N	Call external XSLT processor.
USR6002N	Get the current values of the internal counters MADIO, MAINPR and MAXCL.
USR6006N	Get path to system file.
USR6202N	Get and set value of an environment variable.
USR6203N	Get the path of a resource. Add and delete resources.
USR6204N	Set Natural profile parameter PROGRAM.
USR6303N	Retrieve Natural stack information (Unicode).

5 Changes and Enhancements

▪ International Support	18
▪ Natural System Files	18
▪ Editors	19
▪ Configuration Utility	19
▪ Other Utilities	21
▪ Parameters	23
▪ Statements	24
▪ System Functions	27
▪ System Variables	28
▪ System Commands	29
▪ Application Programming Interfaces	30
▪ Error Messages Corresponding to Adabas Response Codes	30
▪ Natural Web Interface	30
▪ XML Toolkit	35
▪ Natural Connection	35
▪ Miscellaneous	36

International Support

Natural for OpenVMS now includes the international support which was previously only provided with the international version of Natural 5.1.1.

Bidirectional languages and double-byte characters are now supported. For detailed information on how to activate these features, see the sections *Bidirectional Language Support* and *Double-Byte Character Support* in the *Unicode and Code Page Support* documentation.

The following has changed (as compared with the international version of Natural 5.1.1):

- In the map editor, the **Flip Map** command has been renamed to **Reverse Map**.
- The environment variable `RTL_BW_COMP` is no longer supported. Natural no longer supports logical resorting of the characters during input; the characters are always displayed in the keying sequence for left-to-right fields and in the reversed keying sequence for right-to-left-fields.
- The environment variable `NATLANG` is no longer supported. For enabling the international support, the setting of the profile parameter `CP` is now used.
- The locale files (previously installed in *etc/locale*) which were delivered with the international version of Natural 5.1.1 are no longer supported.
 - Month and day names are now retrieved from the Natural text/error message files.
 - Character property information such as “is the character upper case” is retrieved from the ICU library which is used for the Unicode support.
 - Using the AM/PM strings from the locale files is no longer supported.
 - Using the year edit masks `EC`, `EE`, `EV` and `EY` is no longer supported.
 - Characters which are entered in a right-to-left field are no longer automatically converted according to the keymap defined in the locale files. Natural now uses the characters which are delivered from the operating system, which is compatible to the Windows behavior. Changing the input language has to be achieved by executing the appropriate operating system functions.

Natural System Files

The structure of the Natural library directory file *FILEDIR.SAG* has changed. The new format of the file is now platform-independent and it can be exchanged between all Windows, UNIX and OpenVMS platforms. The old format can still be read by Natural for OpenVMS Version 6.3 and will be automatically converted into the new format whenever a modify access is made. Older versions of Natural cannot read the new *FILEDIR.SAG* format. For detailed information, see *Portable Natural System Files* in the *Operations* documentation.

Editors

Data Area Editor

2-digit level support: the level number has been increased to 99 digits.

In addition, if you use a new functionality in the data area editor (e.g. level > 9), the source will be saved in a new internal format. This means that these new types of data areas cannot be read with the previous version of Natural.

For details, see *Data Area Editor* in the *Editors* documentation.

Configuration Utility

User Interface

The user interface of the Configuration Utility (previously known as NATPARM utility) has been redesigned.

Database Management System Assignments

It is now possible to specify the database type ADA2. This database type supports alphanumeric objects (Adabas LA option) and large object database fields (LOBs), which are dynamically defined inside a view, and view sizes greater than 64KB. It can be used for Adabas as of version 6 on Open Systems and Adabas as of version 8 on mainframes. See *DBMS Assignments* in the *Configuration Utility* documentation.

HTTP Client Parameter Settings

The global configuration file now offers the possibility to specify the following parameters for HTTPS (HTTP with SSL): SSLPRX, SSLPRXPT and NOSSLPRX. See *HTTP Client Parameter Settings* in the *Configuration Utility* documentation.

System Files

The global configuration file now offers the possibility to specify alias names for the system files. This feature applies when working in a remote development environment (SPoD) with Natural for Windows. These alias names are then shown in Natural Studio's library workspace. See *System Files* in the *Configuration Utility* documentation.

Batch Mode

The new profile parameters CPOBJIN, CPSYNIN and CPPRINT can be specified. See *Batch Mode* in the *Configuration Utility* documentation.

Compiler Options

The new profile parameters KCHECK and TQMARK are now used in the Configuration Utility instead of KC and TQ.

When you export a parameter file using the Configuration Utility, `KCHECK` and `TQMARK` are now written to the resulting text file (instead of `KC` and `TQ`). When you import a parameter file which has been exported with an older version of Natural, this file is still read in correctly.

The new profile parameters `MASKCME` and `PCHECK` can be specified.

The new profile parameter `PSIGNF` can be specified.

See *Compiler Options* in the *Configuration Utility* documentation.

Error Handling

The profile parameter `NOAPPLERR` can now be specified.

The new profile parameter `RTINT` can be specified.

See *Error Handling* in the *Configuration Utility* documentation.

Device/Report Assignments

The **Close mode** option is now defined for the devices. In the previous version it was defined for the reports. See *Device Assignments* in the *Configuration Utility* documentation.

The printer device `TRANSFER` has been replaced by a new report device assignment. See *Report Assignments* in the *Configuration Utility* documentation. See also *Defining Printers* in the *Natural Connection* documentation.

Regional Settings

The new profile parameters `SUTF8`, `SRETAIN`, `CPCVERR` and `CP` can be specified.



Note: The existing `CP` parameter (used with Natural Remote Procedure Call) has been renamed to `CPRPC`. See *Remote Procedure Call* in the *Configuration Utility* documentation.

The new profile parameter `SUBCHAR` can be specified.

See *Regional Settings* in the *Configuration Utility* documentation.

Work Files

The profile parameter `TMPSORTUNIQ` can now be specified.

A new work file type, `CSV` (for comma-separated files), can now be specified.

It is now possible to open a work file in append mode and to delete a work file after closing it. Different attributes can be defined for these cases.

It is now possible to define a code page for the work file types `ASCII`, `ASCII compressed`, `Unformatted` and `CSV`.

For the work file types which write code page data (`ASCII`, `ASCII compressed`, `Unformatted` and `CSV`), it is now possible to write a byte order mark (BOM). The prerequisite is that the code page `UTF-8` is defined for the work file.

See *Work File Assignments* in the *Configuration Utility* documentation.

Remote Procedure Call

The new profile parameter `SRVWAIT` can be specified. See *RPC (Server)* in the *Configuration Utility* documentation.

Web I/O Interface

"Web I/O Interface" is a new category which is available in the parameter group "Product Configuration". The new profile parameter `WEBIO` can be specified in this category. See *Web I/O Interface* in the *Configuration Utility* documentation.

Other Utilities

Utility Activation

As of this version, Natural invokes a Natural utility without performing a logon to the corresponding utility library in the `FNAT` system file. As a result, Natural preserves the global data area (GDA) and/or application-independent variables (AIV). The current user library and the settings are maintained. To reset the GDA and/or the AIVs, see the parameter `FREEGDA`. The following utilities can now be used in this manner:

- `SYSERR`
- `SYSNCP`
- Object Handler
- `SYSRPC`
- Natural Security

The behavior described above can have an impact on batch procedures from previous Natural versions that contain logon instructions to utility libraries. For more information, see the *Tools and Utilities* documentation.

FTOUCH

The `convert` option of the utility `FTOUCH` can now be used to migrate an old `FILEDIR.SAG` file into a new portable `FILEDIR.SAG` file, as supported with Natural Version 6.3 for OpenVMS and with Natural Version 6.2.1 for UNIX and Windows.

The following new options are provided: `encoding`, `endian` and `objname`.

The new option `-ignoreext` specifies that files with unknown extensions contained in a library are ignored.

You can now define that line numbers are not written to the source code in the file system.

In the Natural main menu, a minus (-) character after the library name indicates that in this library the line numbers are suppressed in the source code.

See *FTOUCH Utility* in the *Tools and Utilities* documentation.

INPL

The INPL utility now automatically recognizes the type of the load file. It is no longer necessary to specify the correct load file type. See *INPL Utility* in the *Tools and Utilities* documentation.

Object Handler

Setting up a profile for defining individual defaults and standard procedures is now possible. You can define an individual profile for your Object Handler utility by modifying the standard Object Handler user profile and specifying general or user-defined parameter settings, such as activating Advanced-user mode by default, or specifying default settings for work files, report files and Workplans. For more information, see *Object Handler - Profile Settings* in the *Tools and Utilities* documentation.

The Transfer format can now also be used for unloading or loading sources of Natural objects in Unicode/UTF-8 (Universal Transformation Format, 8-bit form). See *Transfer Format* in the *Object Handler* section of the *Tools and Utilities* documentation.

The Object Handler has been enhanced to

- support unloading of Predict sets,
- allow the specification of a library name for the Predict XRef save set option,
- allow direct output of report data for Object Handler functions executed in batch or with Application Programming Interface without storing the data in a file.

The new command `DISPLAY STATISTICS` allows to display statistical data of the objects processed when working in command mode or when a workplan of type PROCEDURE is processed.

The Object Handler now automatically recognizes the type of the load file for the load and scan functions. It is no longer necessary to specify the correct load file type. The user interface of the Object Handler has not been changed due to compatibility reasons, however, the **Portable work file** option will not be evaluated for the load and scan functions. See the description of `WORKFILETYPE` in *Keyword Explanation of option-setting*.

The `DELETEALLOWED` option can be used in an Object Handler direct command to specify that the delete instructions from work files created with `NATUNLD` (which is no longer installed by default) are processed.

SYSRPC

See *Natural Remote Procedure Call (RPC)* later in these *Release Notes*.

Parameters

ACIVERS

The EntireX ACI Version 9 is now supported by ACIVERS. Therefore, the maximum value has been increased from 8 to 9. See also the corresponding information in the section *Natural Remote Procedure Call (RPC)* later in these *Release Notes*.

HE

The session parameter HE and the corresponding field in the map editor (both used to specify the name of a help routine to be assigned to a field) have been extended by *nX* notation. This notation allows you to omit parameters for which no values are to be passed.

LT

The range of possible values for the profile and session parameter LT has been changed from 0 - 99999999 to 0 - 2147483647. The default setting for LT is still 99999999.

LT=0 shuts off the counting mechanism for loops. This means that there is no limit to the number times that a loop is run through when LT=0.

OPRB/ETEOP

The functional scope of the profile parameter OPRB has changed. Handling of implicit END TRANSACTION calls at the end of a program are no longer handled by OPRB, but now part of the functional scope of the profile parameter ETEOP. This offers greater flexibility. The following table shows the settings necessary to achieve the same result as with the former OPRB settings in earlier Natural versions:

Former OPRB Setting	New ETEOP/OPRB Settings
OPRB=OFF (ET is issued)	OPRB=OFF and ETEOP=ON
OPRB=NOOPEN (ET is not issued)	OPRB=OFF and ETEOP=OFF
OPRB=(ACC=...) (ET is not issued)	OPRB=(ACC=...) and ETEOP=OFF

Migration takes place automatically if a parameter file of an older Natural version is accessed. No user action is necessary.

The profile parameter OPRB can now be used to specify the encoding for W fields in an Adabas user session.

STEPLIB

The profile parameter LSTEP has been renamed to STEPLIB. The information which was previously provided for the STEPLIB parameter is now available under *Additional Steplib Assignments* in the *Configuration Utility* documentation.

The default value assigned to the profile parameter STEPLIB has been changed from "STEP" to "SYSTEM". Therefore, you have to take precautions if you create new parameter files and

your applications use these newly created parameter files and depend on objects located in the library `STEP` (the former default value). The modification of the default value implies an alteration in the default library search order (`steplib`). Therefore, your applications might not run properly or might produce unexpected results.

When you are using old parameter files in which the value "STEP" is defined for the profile parameter `STEPLIB`, your existing applications which depend on objects located in library `STEP` will still run properly.

If your applications do not expect objects to be located in library `STEP` on `FUSER` or if library `STEP` on `FUSER` is empty, your applications will run properly after the default value change of `STEPLIB`. The performance is increased since the library search order decreased in size. You are encouraged to modify `STEPLIB` in your current installation to the proposed new default value in order to take advantage of the above mentioned performance increase.

USIZE

The default setting of the profile parameter `USIZE` has been changed to 20 MB (previously 10 MB).

WORK

`CLOSE MODE` is a new option of the profile parameter `WORK`. It specifies the work file / printer close mode.

YSLW

It is now possible to specify a "year fixed window" with the profile parameter `YSLW`. This enhancement is provided for compatibility with Natural for Mainframes.

Statements

CALLNAT, FETCH, PERFORM

The performance has been enhanced for the statements `CALLNAT`, `FETCH` and `PERFORM`.

DECIDE ON

The `SUBSTRING` option can now be used as *operand1* and *operand2* in the `DECIDE ON` statement. It allows you to check only a certain part of an alphanumeric, Unicode or binary field.

DEFINE DATA

The following are enhancements to `DEFINE DATA`:

- arrays of dynamic variables are now allowed;
- extensible arrays or X-arrays are now allowed;
- the `INIT` and `CONST` clauses are new;
- `DEFINE DATA` now supports up to 99 levels.

When using the keywords `BY VALUE`, `BY VALUE RESULT` and `OPTIONAL` after a variable definition inside a `DEFINE DATA LOCAL` statement, you will receive an error. These keywords are only

allowed in the parameter data area using `DEFINE DATA PARAMETER`. Nevertheless, you may use these keywords in parameter data areas in a `USING` clause of `DEFINE DATA LOCAL`.

```
DEFINE DATA LOCAL 1 #A(A1) BY VALUE /* NAT0635 END-DEFINE
END
```

DEFINE PRINTER

When using the `OUTPUT` option `TRANSFER`, all output data written to this report number are sent to an Entire Connection terminal. In other words, output data which would normally appear on the screen are written into an NCD file.

It is now possible to specify report number 0 as a valid report number with the `DEFINE PRINTER` statement. This enhancement is provided for compatibility with Natural for Mainframes.

DEFINE WORK FILE

It is now possible to specify attribute operands with the `DEFINE WORK FILE` statement. Using the different attribute operands, it is possible to open a work file in append mode and to delete a work file after closing it.

It is now possible to define the new work file type `CSV`. See also *CSV Format* in the *Operations* documentation.

For the work file types which write code page data (`ASCII`, `ASCII compressed`, `Unformatted` and `CSV`), it is now possible to specify the attribute operand `BOM` (writes a byte order mark) or `NOBOM` (does not write a byte order mark; default) with the `DEFINE WORK FILE` statement. The prerequisite is that the code page `UTF-8` is defined for the work file.

DOWNLOAD PC FILE

For the statement `DOWNLOAD PC FILE` new options are available:

- `SYNC`

With `SYNC`, you specify that the PC returns control to Natural after executing and terminating a `COMMAND`. `SYNC` can be used, for example, to ensure that the command `SET PCFILE` has been executed before a file transfer starts.

- `ASYN`

With `ASYN`, you specify that the PC immediately returns control to Natural, regardless of whether the execution of a `COMMAND` has been terminated or not.

DISPLAY, INPUT, PRINT, REINPUT, STACK, WRITE

Dynamic variables may be used as an operand.

EXPAND, REDUCE, RESIZE

The following statements can now be used for the X-arrays:

- `EXPAND`

- `REDUCE`

- `RESIZE`

These statements are used to change the number of occurrences of an X-array.

EXAMINE, COMPRESS, SEPARATE

In addition to alphanumeric format, these statements now support binary format.

EXAMINE

The EXAMINE statement has two new clauses: DIRECTION and POSITION.

The EXAMINE statement can now be used to examine graphemes. The new clauses CHARPOSITION and CHARLENGTH are used for this purpose.

FIND

The FIND statement now allows a non-descriptor search.

FIND, HISTOGRAM and READ

Multi-fetch processing can be enabled for the statements FIND, READ and HISTOGRAM that do not involve database modification. For further details on the statements that can be used with the Multi-Fetch Clause, see *Multi-Fetch Clause* in the *Programming Guide*.

FOR

An arithmetic expression can now be used as *operand2*, *operand3* and *operand4* in the FOR statement.

MOVE

It is now possible to specify the following options with the MOVE statement:

- MOVE NORMALIZED

Converts a Unicode string into the “Unicode Normalization Form C” (NFC). The resulting Unicode string does no longer contain combining sequences for characters which are available as pre-composed characters.

- MOVE ENCODED

Translates a character string, encoded in one code page, into the equivalent character string of another code page.

READ

The READ statement has been enhanced by the new DYNAMIC clause. With DYNAMIC, you can specify the direction of loop processing dynamically.

The READ statement is enhanced by a WITH REPOSITION clause. The complete READ loop can be restarted by a ESCAPE TOP REPOSITION statement within the READ loop.

READ and HISTOGRAM

The READ and HISTOGRAM statements have been enhanced by a new TO clause. This is used for a different type of end value processing.

The READ and HISTOGRAM statements have been enhanced by new comparators for start value processing. They are:

- LT

- GT

- LE
- GE

REQUEST DOCUMENT

It is now possible to specify an encoding with the `REQUEST DOCUMENT` statement. The new `ENCODED` clause is used for this purpose.

HTTPS (HTTP with SSL) requests can now be handled by the `REQUEST DOCUMENT` statement if OpenSSL is installed.

SET GLOBALS

For the `SET GLOBALS` statement, the restriction that it can be applied in reporting mode only has been removed. This is to allow you to specify the `CPCVERR` session parameter (Code Page Conversion Error) for a Natural application that is to support Unicode also in structured mode. Also, the other session parameters available with this statement can now be set with `SET GLOBALS` in structured mode.

New Object Type Function in Statements and Expressions

Function calls can be used directly in statements or expressions, replacing operands. For further details, see *User-Defined Functions* in the *Programming Guide*.

Use of Large Constants

The limit of 253 bytes for alphanumeric and hexadecimal constants is no longer valid. The maximum size of one constant is 1073741824 bytes (1 GB).

SUBSTRING Option

The `SUBSTRING` option now supports binaries. For more information on this option, see *Relational Expression* in the *Programming Guide*.

System Functions

On OpenVMS, Windows and UNIX platforms, the behavior of the system functions has been changed in such a way so that they are compatible with the system functions on mainframe platforms. When an argument of a system function contains an array with a variable index, it is checked whether this argument is a scalar value or an array with several occurrences. A temporary variable is created as a scalar or array, according to the referenced variable in the argument which is used as the result of the whole system function.

Example:

```
DEFINE DATA LOCAL
  1 #BREAK (A1/1:11) INIT <'a','a','a','a','a','b','b','b','b','b'>
  1 #VALUE (P3/1:11) INIT <1,2,3,4,5,6,7,8,9,10>
  1 #I1      (P3)
END-DEFINE
*
F1. FOR #I1 = 1 TO 10
  PERFORM BREAK (F1.)
  AT BREAK OF #BREAK (#I1)
    /* Array with variable index which references only a scalar value!
    PRINT SUM(#VALUE (#I1)) SUM(#VALUE (#I1+1))
    /* Array with variable index which references an array with 2 occurrences!
    PRINT SUM(#VALUE (#I1:#I1+1))
    PRINT '-'(10)
  END-BREAK
END-FOR
END
```

Output:

```
15  20
15  20
-----
40  34
40  34
-----
```

System Variables

*APPLIC-NAME

If Natural Security is installed, the Natural system variable *APPLIC-NAME contains the name of the library to which the user is logged on. If the user is logged on via a special link, the system variable now contains the link name instead of the library name.

*NATVERS

The format of *NATVERS has been changed to *rr.vv.ss* (where *rr*=release, *vv*=version, *ss*=system maintenance level). It is now compatible with the format used on the mainframe.

For example, the following program will now display 06.03.04 instead of 6.3.4:

```
DISPLAY *NATVERS
END
```

*SERVER-TYPE

The new value "WEBIO" of the *SERVER-TYPE variable indicates that Natural has been started as a Natural Web I/O Interface server.

System Commands

CATALL

The system command `CATALL` can now be used in batch mode. The direct command syntax has been enhanced accordingly.

The system command `CATALL` can now also be used to check, save or stow objects. By default, the source-code lines of sources that were saved or stowed are now automatically renumbered.

When Natural Security is active, it is now checked whether the selected action (catalog, check, save or stow) is allowed under Natural Security. When it is not allowed, an occur occurs.

COMPOPT

It is now possible to specify the following parameters with the `COMPOPT` command (in addition to the parameters `DBSHORT` and `GFID`):

- KCHECK
- MASKCME
- PCHECK
- PSIGNF
- THSEP
- TQMARK

LIST DIR

When you issue the command `LIST DIR object-name`, encoding information is now shown. It shows the code page with which the source has been saved. If a code page has not been specified, this field remains empty.

STRUCT

The result of the system command `STRUCT` is now the same as on the mainframe.

TECH

The system command `TECH` has been enhanced to display the code page and locale used in the current Natural session. (The same applies when the terminal command `%<TECH` is used instead.)

Application Programming Interfaces

The following application programming interfaces (APIs) in the library SYSEXT have been enhanced:

API	Description
USR2026N	Get TECH information. This API can now read the contents of the system variables *CODEPAGE and *LOCALE.
USR2011N	Get or set a work file name. This API has the following new parameters: <ul style="list-style-type: none">■ WORK-ATTRIBUTES (A1000) Matches the ATTRIBUTES clause of the DEFINE WORK FILE statement. Currently possible values are APPEND/NOAPPEND, BOM/NOBOM and DELETE/KEEP.■ WORK-CODEPAGE (A64) Matches the work file property Code page which is allowed for the work file types ASCII, ASCII compressed, Unformatted and CSV. Both parameters are optional. They must always be specified together (i.e. you must either specify both parameters or none of them). When only one parameter is specified, it is ignored.

Error Messages Corresponding to Adabas Response Codes

The Natural error messages that correspond to Adabas response codes have been completely revised with Natural Version 6.3.

Natural displays subcodes or other information from Additions fields if they are provided by Adabas.

Natural Web Interface

The following changes/enhancements to the Natural Web Interface are provided with this version of Natural:

- [Server Extensions](#)
- [New Library SYSWEB3](#)

- [Changes to SYSWEB](#)

Server Extensions

Natural Web Interface server extensions are now available as Apache module for Apache 2.0.x or Apache 2.2.x. For further details, refer to the following sections in the *Web Technology* documentation

- *Natural Web Server Extensions - Introduction for SYSWEB3*
- *Supported HTTP Servers*
- *Natural Web Server Extensions for MOD*

Using the Web Interface on a remote machine without EntireX technology is now possible via the PAL interface as used for the SPoD server.

The server extension program initializes a communication directly to the PAL server instead of, for example, using RPC. Because all Web Interface programs are subroutines, and it is not possible to run a subroutine directly using PAL, a generic stub is necessary. This stub handles steplib setting, parameter transfer and CALLNAT of the subprogram.

For further information, see *Communication Using PAL Techniques and Functionality* in the *Web Technology* documentation.

New Library SYSWEB3

A new version of the Web Interface (SYSWEB3) is provided.

Multi-byte character sets (such as Japanese) can now be used with Natural DCOM. A prerequisite for this is that you use the new Natural library SYSWEB3. See *Natural Web Online Documentation SYSWEB3* in the *Web Technology* documentation.

For information on how to convert your old SYSWEB applications to SYSWEB3, see *Migrate Natural Web Interface SYSWEB to SYSWEB3*.

Usage of dynamic variables:

- The interface PDA W3PARAM has been modified to use (A) dynamic and (B) dynamic variables instead of (A250/1:v Array).
- A separation of transferred data is no longer necessary - this avoids errors during character conversion.
- Fewer calls are made to the EntireX Broker because of a larger communication area.
- A source change in existing programs is not necessary.
- A recatalog of existing programs with the new library SYSWEB3 is mandatory and the new Web Interface server extensions have to be used.

- The old interface and the library `SYSWEB` together with the old Web Interface server extensions are still delivered.
- The size of the incoming data (containing variables) is no longer limited to approx. 30000 bytes.
- It is no longer required to modify `W3PARM` to change the size of the incoming data or outgoing data.
- A recatalog of library `SYSWEB3` is no longer necessary to change configuration settings.
- To change the maximum size of the incoming data and the outgoing data, only the new program `W3LIMIT` needs to be changed (it may be required to increase EntireX Broker or RPC sizes).
- Incoming data no longer needs to be of mime type "application/x-www-form-urlencoded" (the check is no longer performed by the Web Interface server extension, but at `W3INIT` instead).
- The request method `PUT` is supported (the check is no longer performed by the Web Interface server extension, but at `W3INIT` instead).
- Incoming or outgoing data is no longer limited to alphanumeric data format. Binary data can be transferred e.g. via multipart-form-data upload from a browser, including pictures or delivering pictures, for example from the Natural resource directory.
- Parameter settings at the URL (`QUERY_STRING`) in combination with the `POST` method are supported.
- The number of parameters delivered to the HTTP server via the `POST` method is no longer limited to 200 parameters. The number of parameters delivered via the `POST` method now depends on the data size.
- It is now possible to set a response number other than 200 for HTTP requests from a Natural program (for example, as HTTP `PUT` response). See *Overview of Response Numbers for HTTP/HTTPs Requests* in the *Statements* documentation for a list of possible response numbers.

W3* programs for binary data and mime-types:

- There is a new `W3BINARY` call to transfer back binary data.
- `W3INFO` and PDA `W3PINFO` have been extended to be used with dynamic in/out data.
- `W3INIT` has been extended with optional parameters to allow mime types other than "application/x-www-form-urlencoded" and HTTP `PUT` requests.
- There is a new subroutine `W3READ-INPUT` to get all transferred data. For example, alpha and binary data, header and parameter (name/value pair) delivered via the `POST` method and/or `QUERY_STRING`.
- The routine `W3MIME-DATA` together with the Natural text member `MIMEDATA` controls the transport type (binary or alpha) of the return page for the given content type (`W3CONTENT-TYPE`).
- The routine `W3MIME-TYPE` together with the Natural text member `MIMETYPE` defines the file extension for given content types.

NAT* programs have been extended for the new interface:

- Program NAT-DATA displays all incoming data, also including binary data in a hexadecimal-editor like view.
- Program NAT-DIR has been extended with a version parameter to check which library (SYSWEB or SYSWEB3) has been used during compile time.
- Program NAT-DIRR has been extended to show links also for binary data (for example, *.gif) in order to display the data using program NAT-RES.
- Program NAT-RES has been extended to display binary data files (for example, *.gif).
- Program NAT-RES can be used to upload data to the Natural resource directory (this new feature has to be activated manually in the program source).

Changes to SYSWEB

The information listed below applies to the old SYSWEB library.

Internal Usage of Dynamic Variables and New Natural Features

- Internal usage of dynamic variables instead of (A1/1:v arrays).
- Existing programs need not be recataloged.
- Use new Translate commands internally.
- The predefined maximum return page is now set to 100000 (can be changed more easily).
- To change the maximum size of the return page, only the PDA W3LIMITS needs to be changed. It is no longer necessary to change w3context, too. There is no need to change the definition of the (A1/1:v) array and its redefinition) up to the maximum size that might be needed even if the majority of pages is smaller.
- Using dynamic variables decreases the data consumption of the application.
- Easier handling of environment variables greater then (A250). These variables need not be read in parts using redefined long alpha with (A1/1:v) array.
- Use of long constant values. A complete HTML page now can be defined within an init clause of the program, or an simple assign statement.

W3* Programs for Dynamic Variables and Dynamic Arrays of Dynamic Variables

- All interfaces of programs that had been defined as (A250) By Value have been changed to (A) Dynamic By Value. User programs need not be changed or recataloged to work with the changed interface.
- New W3*DYNAMIC calls have been introduced. The existing calls look for trailing blanks (because of static length strings) and remove trailing blanks. The new calls take the compile dynamic variable up to *length(), without additional checking.
- Use of constant values greater the (A253).

Use #HTTP_NEWLINE together with W3HTML

If more than one line should be produced and converted to HTML, it had been necessary to call W3HTMLLINE for every page. The changed Web Interface now can handle lines that had been compressed with ##HTTP_NEWLINE.

Old:

```
perform w3text "<pre>"
perform w3htmlline "if a > b then"
perform w3htmlline " write 'result is > then old' "
perform w3htmlline "else"
perform w3htmlline " write 'result is < then old' "
perform w3htmlline "end-if"
perform w3text "</pre>"
```

New:

```
perform w3text "<pre>"
COMPRESS "if a > b then" ##HTTP_NEWLINE
" write 'result is > then old' " ##HTTP_NEWLINE
"else" ##HTTP_NEWLINE
" write 'result is < then old' " ##HTTP_NEWLINE
"end-if" ##HTTP_NEWLINE
into W3DYN LEAVING NO
perform w3html w3dyn
perform w3text "</pre>"
```

HTML2NAT Generates Long Constant Values

To reduce calls to W3TEXT, new generated programs will use constant values longer than (A253), if possible.

Call XSLT Processor before W3END

After a complete XML document is generated, call an XSLT processor to convert your XML. See e3xslt in the library SYSWEB. Or generate a HTML template and add your converted xml to a specific place. See e3xslt3 in the library SYSWEB.

Template Processing of Documents Saved at Resource Directory

It is possible to load templates defined with an HTML editor from the Natural resource directory. Your specific character strings like \$NWWW\$ can be searched and replaced with the actual value. See e3templ in the library SYSWEB.

XML Toolkit

All XML Toolkit-relevant programs, dialogs etc. are now delivered in the library SYSXTK (no longer in the library SYSEXXT).

A **File Selection** dialog has been implemented. File names, whenever possible, can be selected from a selection list.

XML schemas containing `<import>` or `<include>` elements can now be used for generation. For the schema "Location", only local saved files can be used.

`<any>` and `<anyAttribute>` elements can now be specified. The content of an `<any>` or `<anyAttribute>` element is saved unparsed as a dynamic variable.

Conditional sections inside a DTD are now supported.

The element type ANY is now supported (equal to the `<any>` XML schema construct). The content of an ANY element is saved unparsed as a dynamic variable.

For information on the current limitations, see *Considerations and Limitations* in the *XML Toolkit* section of the *Web Technology* documentation.

Natural Connection

Support of Long and Dynamic Variables

The new feature now enables you to upload or download long and dynamic variables in Natural Connection work files.

Work File Format and Attributes

Below are the restrictions that apply to the use of work file attributes:

Accessing PC work files is restricted to a fixed record length of 1073741823 bytes or 32767 bytes when using the statement `WRITE WORK FILE VARIABLE`. Depending on the Entire Connection version installed on the PC, additional restrictions may apply as described below.

Maximum File Transfer Record Length for Natural Connection

The maximum record length supported for file transfer is 1 GB - 1 byte = 1073741823 bytes. For more information, see *Maximum File Transfer Record Length for Natural Connection* in the *Natural Connection* documentation.

Miscellaneous

Single-Fetch as Default Fetch Mode for Adabas Databases

By default, Natural uses single-fetch to retrieve data from Adabas databases. This default processing mode can be set to single-fetch or multi-fetch using the profile parameter `MFSET`. The setting can be changed on statement level if required.

For further information, see *Accessing Data in an Adabas Database* in the *Programming Guide*. You can also consult the statements `READ`, `FIND` and `HISTOGRAM` for more information.

Adabas Access

Multiple-value fields and periodic groups with more than 191 occurrences are now supported as of Adabas Version 5. The size of one Adabas record is still limited to 64 KB.

Characters in a Mask

The JJJ character (Julian date) may now be used within a mask definition. This enhancement is provided for compatibility with Natural for Mainframes. See *Characters in a Mask* in the *Programming Guide*.

Programs in Sample Libraries SYSEXPB and SYSEXSIN

The library `SYSEXRM` has been renamed to `SYSEXSIN`.

The sample programs included in the Natural documentation that are located in the libraries `SYSEXPB` and `SYSEXSIN` have been completely revised, enhanced and renamed.

Application Programming Interfaces USR* in Library SYSEXT

The `USR*` programs delivered in the library `SYSEXT` now run in a special mode. As a result, the `USR*` programs need no longer set further steplib to execute related Natural objects for processing. This reduces the impact on the Natural buffer pool search logic and improves performance significantly if application programming interfaces are used extensively within user-written applications.

Buffer Pool

The SHUTDOWN command of the NATBPMON utility has been enhanced to quickly shut down a buffer pool. It is now possible to specify the FORCE option with an optional parameter. If NATBPMON SHUTDOWN FORCE *grace-period* is executed with super user rights, the terminate signal SIGTERM is sent to all active Natural sessions. After the expiry of the seconds defined with the *grace-period* parameter, the IPC resources held by the buffer pool are removed from the system. This command only has effect if executed with super user rights; without these rights, NATBPMON will reject its execution. See *NATBPMON Commands* in the *Operations* documentation.

Printing

When you print an object, the database ID and file number are now included on the printout.

NCF Format Files

The content of the NCF format files which are created for the work file type Entire Connection has changed. Now the information fields that were empty before are also filled with the work file processing information. The content of the information field **File name** has changed; the file name and path are now included in the information field. Since Unicode is now supported for NCD files, a new information field is available in the NCF file, containing code page information.

6 Known Issues

- Batch Mode 40
- Error Messages 40
- Terminal Command %= 41
- Usage of Database Field Short Names 41
- Interrupting a Running Natural Application 41
- Natural Native Interface 41

This chapter provides additional information that you should be aware of after having installed Natural.

Batch Mode

Natural system commands which require a screen-based user interface (for example, `CATALL`) are not recognized in batch mode (formerly called “real batch mode”).

Error Messages

Some error messages have been changed for this version of Natural to improve compatibility between Natural for Windows, Natural for UNIX, Natural for Mainframes and Natural for OpenVMS.

The following error messages have been changed:

- NAT6254 has been replaced by NAT1254;
- NAT0968 has been replaced by NAT0935;
- NAT6010 has been replaced by NAT1222 for X-arrays and large and dynamic variables.

The following error messages are new for this version of Natural to improve compatibility between Natural for Windows, Natural for UNIX, Natural for Mainframes and Natural for OpenVMS.

Error Message	Description
NAT1145	Input does not match edit mask :1:. NAT1145 replaces NAT1143 in case of interactive input/output. This new error advises the user of the expected input format for the current field.
NAT1257	Access to a non-materialized X-array field not permitted. NAT1257 replaces NAT1316 in case of references to non-materialized X-arrays.

Terminal Command %=

If the background color of a window is changed with the terminal command `%=`, this will now also change the background color of the text constants. In earlier versions, the background color of text constants was not changed which caused an uncommon color layout.

Usage of Database Field Short Names

In previous versions, a database field short name was not rejected during compilation if the parameter `DBSHORT=ON` and the `DEFINE DATA LOCAL` statement were specified. This problem was corrected. The syntax error NAT0981 is now returned. For further information, see the enhanced description of the `DBSHORT` parameter.

Interrupting a Running Natural Application

The profile parameter `RTINT` determines whether it is allowed to interrupt a running Natural application that does not respond anymore. This mechanism may cause problems afterwards, when continuing or terminating the application processing. Therefore, it is not recommended to set this parameter to `ON`. By default, interrupts are not allowed.

Natural Native Interface

The Natural Native Interface, which enables an application to execute Natural code in its own process context through function calls according to the C calling convention, is not yet supported.

7

Natural Remote Procedure Call (RPC)

▪ New Documentation Structure and Contents	44
▪ Migration	44
▪ New Features	45
▪ Enhanced Features	48
▪ Changed Features	50
▪ SYSRPC Utility Enhancements	51
▪ Changes to the Service Directory	53
▪ Limitations on OpenVMS	53

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the enhancements listed in this chapter are available.

New Documentation Structure and Contents

The Natural RPC documentation has been restructured and revised completely, also taking into account the new features that are now available with EntireX Communicator (EntireX Broker).

Important:

Information on prerequisites and restrictions that you should observe before you start to implement or modify a Natural RPC environment has been summarized in the following sections of the *Natural Remote Procedure Call (RPC)* documentation:

- *Prerequisites and Preparatory Information*
- *Restrictions and Limitations*

Migration

EBCDIC/ASCII Conversion Removed

With Natural RPC Version 5.1.1 (and below), an RPC built-in conversion from EBCDIC to ASCII and vice versa took place if the EntireX attribute `TRANSLATION` or `CONVERSION` was missing in the Broker attribute file and the encoding of the client data had to be translated to the required encoding of the server platform.

The RPC built-in conversion from EBCDIC to ASCII and vice versa has been removed. The conversion of client data from EBCDIC to ASCII and vice versa is already part of the EntireX functionality. In addition, the EntireX conversion is more versatile than the RPC built-in conversion. Specify the EntireX attributes `TRANSLATION` or `CONVERSION` in the Broker attribute file to ensure an appropriate conversion.

Setting of Steplib SYSRPC Omitted on Client Side

With Natural RPC Version 5.1.1 (and below), the library `SYSRPC` is implicitly set as a steplib when the service is accessed on the client side.

The library `SYSRPC` is no longer set as a steplib during execution of a remote `CALLNAT` request. This is to improve the performance and to avoid that an old service directory in the library `SYSRPC` is accidentally accessed. The `SYSRPC` utility places the service directory in the current user library instead of library `SYSRPC` in the `FNAT` system file. Therefore, the library `SYSRPC` should not contain any user-specific information, including the service directory.

If you want to continue to use a service directory that has been placed in library `SYSRPC` before, you have the following choices:

- Move the service directory from the library `SYSRPC` to the client library.
- Move the service directory from the library `SYSRPC` to the library `SYSTEM` in the `FUSER` system file.
- Define the library `SYSRPC` as an additional steplib using Natural Security.
- Use the application programming interface `USR3025N` in the library `SYSEXT` to set the library `SYSRPC` as an additional steplib in an environment without Natural Security.

New Features

Support of the New EntireX RPC Protocol RPC2000

The new EntireX RPC protocol `RPC2000` introduces the following new features that are supported by Natural RPC Version 6.3:

- The new header layout of the `RPC2000` protocol has removed all length restrictions for names. However, Natural-specific names such as library names, program names and user IDs remain limited to 8 characters (this feature requires that `RPC2000` is enabled).
- Base 64 encoding is used for marshalling of binary data. This saves up to 30 percent space in the send/receive buffer (this feature requires that `RPC2000` is enabled).
- If the parameter data area (PDA) contains arrays of groups, the physical field sequence of a PDA may be different on the client and server sides if the PDA contains arrays of groups. With Natural RPC Version 6.3, the group structure of a PDA may be kept when an RPC server is called by a Natural client. This is to ensure the same physical field sequence in the PDA for both the client and the server.

X-Arrays

X-arrays are supported in the parameter list of a remote `CALLNAT` execution. The server may increase or decrease the number of occurrences.

Restrictions:

- In case of a multidimensional array, all dimensions of the array must be extensible.
- The lower bound must not be extensible, that is only extensible upper bounds are allowed.

Format U Data

Variables of format U (Unicode data in UTF-16) are supported in the parameter list of a remote `CALLNAT` execution. The variables may either be fixed length variables or dynamic variables.

If U data is sent from a platform with big endian encoding to a platform with little endian encoding or vice versa, the encoding is adapted so that it conforms with the encoding on the receiving platform. For example, when U data in little endian encoding arrives on a big endian platform, this data is converted to big endian encoding before it is handed over to the program. When this data is sent back, it is converted back to little endian encoding.

Optional Parameters

Optional parameters (*nX* notation in the `CALLNAT` statement) are supported in the parameter list of a remote `CALLNAT` execution.

Support of EntireX ACI Version 6 and ACI Version 7 Features

With Natural RPC Version 6.3, the following features available with ACI Version 6 and ACI version 7 are supported:

- Data compression to compress the send/receive buffer on client and server side. This feature reduces the amount of data that is to be transferred from client to server and vice versa. To use this feature, ACI Version 7 is required.
- Data that is exchanged between client and server may be encrypted for reasons of data security. To use this feature, ACI Version 6 is required.

Application Programming Interface to Set the Target Library for CALLNAT Execution on the Server Side

To unbundle the client and the server environment, an application programming interface (API) is provided. This API (USR4008) enables the client to set the library in which the CALLNAT request is to be executed on the server side. The LOGON option must be set to ON to use this feature.

Application Programming Interface to Set EntireX Specific Parameters

A new application programming interface (USR4009) is provided to enable the client to set EntireX-specific parameters for RPC request execution.

Application Programming Interface to Terminate the EntireX Broker Service

The new application programming interface USR2075N enables you to terminate the EntireX Broker Service from within your application. See *Terminating an EntireX Broker Service* in the *Natural Remote Procedure Call (RPC)* documentation.

New RPC User Exits Called by Natural RPC Server at Certain Events

With Natural RPC Version 6.3, two optional user exits are introduced. These exits are called by the Natural RPC server at certain events and must be implemented by the user as a subprogram. The following two user exits have been introduced:

- A before-service-execution user exit that is called after the data is unmarshalled and before the service (the remote CALLNAT) is executed.
- An after-service-execution user exit that is called on return from the service (the remote CALLNAT) and before the data is marshalled.

New Parameter CPRPC

The new parameter CPRPC is used to specify the name of the code page used by the EntireX Broker. CPRPC is applied to the Natural RPC facility when the transport protocol ACI (that is EntireX Broker) is used. See also [Migrating Applications to Version 6.3](#).

Wait Time of RPC Server

The new parameter `SRVWAIT` is used to specify the number of seconds the server is to wait for an RPC client request.

Enhanced Features

Server Trace Enhanced to Aid in Error Diagnostics

With Natural RPC Version 6.3, additional information is logged in the Natural RPC server trace file to aid in error diagnostics.

Information about Server Status Provided to Aid in Error Diagnostics

For better error diagnostics, the server will provide additional status information. An application programming interface (API) is provided to return this information to the client.

Status information includes:

- the `STEPLIB` chain,
- the Natural buffer pool that is used,
- important Natural profile parameter settings (`FNAT`, `FUSER`, RPC parameters, ...).

Performance of Request Processing Improved

Several administrative database calls, that have been issued after processing of the client RPC request has been completed, have been eliminated with Natural RPC Version 6.3. This improves the performance of the RPC server especially for those requests which generate only a small workload.

Natural Security Password Change via RPC

The Natural RPC server has been enhanced by the possibility to change the Natural Security (NSC) password on the `FSEC` by a remote procedure call.

The new application programming interface `USR2074N` allows a Natural RPC client which is running in a non-NSC environment to change the NSC password on the `FSEC` via a remote procedure call. It is no longer required for the client to use a terminal emulation to change the password on the Natural logon screen.

Similar interfaces are provided by EntireX to enable an EntireX RPC client to change his NSC password.

Case-Sensitive User ID and Password

The sample callers USR1071P, USR2071P and USR2072P of the application programming interfaces USR1071N, USR2071N and USR2072N allow the use of case-sensitive user IDs and passwords.

By default, user IDs and passwords are translated into upper case for compatibility reasons.

Prevent Password Attacks on Server

To prevent password attacks if a Natural RPC server is running under Natural Security, each invalid logon attempt is now recorded on the FSEC. If the limit for logon attempts is reached for a user, this user is locked and any further logon attempt (valid or invalid) is rejected. The counter for the logon attempts is reset for a specific user after a successful logon of this user.

Please note that this feature leads to the intended incompatibility that a user must be unlocked after the limit of logon attempts has been reached. In previous versions, the number of attempts was unlimited.

Refer to *RPC Server Session Options in Library and User Preset Values* in the *Natural Security* documentation for more details.

Logon Forced after Timeout

If your application has once issued an EntireX Broker logon using the application programming interface USR2071N, a new logon is enforced after a timeout. In previous versions, the behavior after a timeout depends on the AUTOLOGON setting in the EntireX Broker attribute file.

- With AUTOLOGON=YES an implicit logon is issued.
- With AUTOLOGON=NO a new logon is enforced.

This feature avoids problems with external security systems such as RACF if AUTOLOGON=YES is specified but an explicit logon is required.

Support of Simplified EntireX Security Installation Provided with ACI Version 8

If the profile parameter ACIVERS is set to 8, the Natural RPC issues a KERNELVERS call to get the correct value for the ACI field KERNELSECURITY. In this case, it is no longer required to perform the special installation steps to install the EntireX Broker stubs for EntireX Security (install the secuexit module).

This enhancement applies to the Natural RPC server as well as to the Natural RPC client. In case of a Natural RPC client, the KERNELVERS call is done as part of the EntireX logon using the application programming interface USR2071N.

DFS Profile Parameter Enhanced

The profile parameter `DFS` has been enhanced to prevent the Natural RPC from accessing and using a service directory. This avoids that a service directory has to be created before Natural RPC can be used. Using this feature will result in a reduction of administrative overhead and a corresponding performance enhancement.

TRACE Profile Parameter Enhanced

The `TRACE` profile parameter has been enhanced by an option to write a trace entry only if an error has occurred. This enables you to analyze errors on the Natural RPC server side without the need to have the Natural RPC trace permanently turned on.

ACIVERS Profile Parameter Enhanced

The EntireX ACI Version 9 is now supported by `ACIVERS`. This allows you to enable the EntireX Broker stub to send additional environmental information about client and server to the EntireX Broker.

Changed Features

Reset Natural Limits on Server

The following Natural limits apply to the Natural RPC server:

- `LT` - Limit for Processing Loops,
- `MAXCL` - Maximum Number of Program Calls,
- `MADIO` - Maximum DBMS Calls between Screen I/O Operations.

This makes it difficult to enforce certain limits on every `CALLNAT` request.

With Natural RPC Version 6.3, these limits are reset after each remote `CALLNAT` execution to avoid unpredictable error situations when the execution of a client request reaches one of the limits due to the processing of a previous client.

Long Names for Node and Server

The maximum length for node and server names in the sample callers `USR2007P` and `USR2071P` of the application programming interfaces `USR2007N` and `USR2071N` has been increased from 8 to 32.

SYSRPC Utility Enhancements

The following changes/enhancements have been provided in the `SYSRPC` utility:

- Calculate Space Requirements without Stub Generation
- Generate Stubs in Command Mode
- Comment Added to Parameters during Stub Generation
- Maximum Length of Format Buffer Increased
- Stub Generation Supports Dynamic Variables
- Stub Generation Supports Group Structure
- Stub Generation Supports X-Arrays
- Number of Parameters for Stub Generation Increased
- Stub Generation Supports Format U Data
- Allow Wildcard Notation for Subprogram Names
- Search Function Provided
- Terminate Server Tasks

Calculate Space Requirements without Stub Generation

For stubless RPC calls, the required space for the send/receive buffers can be calculated without generating a stub to allow the user to determine the space requirements.

Generate Stubs in Command Mode

For stubs with the default field attribute `M` (inout), the stub generation can now also be executed in command mode with wildcard notation for a set of stubs. A stub can be generated in batch mode on the basis of an existing subprogram. In this case, all field attributes will be `M`.

Comment Added to Parameters during Stub Generation

When a stub is generated, a comment is added to the parameters to indicate whether a parameter is an input or an output parameter.

Maximum Length of Format Buffer Increased

The format buffer length limit is dynamically calculated depending on the buffer space currently used for data.

Stub Generation Supports Dynamic Variables

It is now possible to generate stubs that contain dynamic variables in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Stub Generation Supports Group Structure

It is now possible to enter field levels and groups on the **Stub Generation** screen. The definition of groups is required to access certain EntireX RPC servers.

Only contiguous groups are supported.

Stub Generation Supports X-Arrays

It is now possible to generate stubs that contain X-arrays in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Number of Parameters for Stub Generation Increased

In the **Stub Generation** screen (which is used to enter the parameter definitions that are used in the stub subprogram), you can now specify a maximum of 5000 parameters. See *Specifying Parameters* in the SYSRPC section of the *Tools and Utilities* documentation.

Stub Generation Supports Format U Data

It is now possible to generate stubs that contain format U variables in their parameter data area (PDA). The **Stub Generation** screen has been adapted accordingly.

Allow Wildcard Notation for Subprogram Names

To reduce the size of the service directory and to enhance its readability, it will be possible to use wildcard notation to specify subprogram names.

Search Function Provided

To ease the inspection of a large service directory, a search function is provided for subprogram names.

Terminate Server Tasks

It is now possible to terminate all server tasks associated with a given EntireX Broker service. See *Terminating a Server* in the *SYSRPC* part of the *Utility* documentation. In previous versions, it was only possible to terminate a single server.

Changes to the Service Directory

Support Local Directory Layout by Remote Directory Server

With Natural RPC Version 6.3, an application programming interface (API) is provided to enable access to information stored in the local directory located on the remote server.

Version Number and Copyright Notice Added to Service Directory

A comment is added to the service directory containing the Natural version that was used to generate the service directory. In addition, a copyright notice of Software AG will be contained in the service directory.

Limitations on OpenVMS

In some places, the documentation set for Natural for OpenVMS mentions features which are (currently) not supported with Natural for OpenVMS. Such features are:

Integrated Authentication Framework (IAF)

The Integrated Authentication Framework (IAF), which is an optional feature that can be used on the Natural RPC server side and which is controlled by Natural Security, is not supported on OpenVMS platforms. Therefore, the following section in the *Natural Remote Procedure Call (RPC)* documentation does not apply on OpenVMS: *Using the Integrated Authentication Framework*.

8 Natural Security

▪ General Information	56
▪ Administrator Services	58
▪ User Profiles	61
▪ Library Profiles	61
▪ Utility Profiles	63
▪ RPC Server Profiles	65
▪ Application Programming Interfaces	66
▪ SECULD2/SECLOAD Transfer Programs	68
▪ Logon Procedure	70
▪ Functional Security	70
▪ Limitations on OpenVMS	70

The enhancements listed in this chapter are available with Natural Security Version 6.3.

General Information

The following topics are covered below:

- FSEC System File
- Shared FSEC System File
- Concurrent Modifications of a Security Profile
- WYSIWYG
- Direct Commands
- Link Functions

FSEC System File

With version 6.3, you can continue to use your existing FSEC system file. No migration of Natural Security data from the previous version to the current version is necessary.

Shared FSEC System File

If you use an FSEC system file shared by multiple versions of Natural Security, be aware that the new features described below are not available with versions prior to 6.3 for Windows and UNIX and 4.2.4 for mainframe platforms.

Concurrent Modifications of a Security Profile

A new general option **Concurrent Modifications Without Notification** allows you to determine how Natural Security is to react in a situation in which two administrators simultaneously modify the same security profile. Such a situation would occur as follows:

1. Administrator 1 invokes a security profile for modification.
2. Administrator 2 invokes the same security profile for modification.
3. Administrator 1 leaves the function after having made his/her modifications - the modifications are applied to the security profile. This means that, at this point, Administrator 2 is working on data which are “out of date”, but is not aware of this fact.
4. Administrator 2 leaves the function after having made his/her modifications. Depending on the setting of the new general option **Concurrent Modifications Without Notification**, there are two possible reactions by Natural Security:
 - The modifications made by Administrator 2 are applied - unknowingly overwriting the modifications made by Administrator 1.

- Administrator 2 receives a window, informing him/her that the security profile in question was in the meantime modified by another administrator. He/she can then contact the other administrator to discuss the changes made, and can then decide to either cancel his/her own modifications or apply them, thus overwriting the modifications made by Administrator 1.

WYSIWYG

If you use functional security within Natural Security, that is, if you use the command processor NSCCMD01 to disallow functions within the library SYSSEC, this now has a “WYSIWYG” effect: If functions are disallowed in NSCCMD01, the corresponding menu items will not be visible on the Natural Security menus. This means that within SYSSEC, you will only see the functions you are allowed to use.

Direct Commands

As of this version, you can also issue a Natural Security direct command from outside of the Natural Security library SYSSEC. This allows you to perform a Natural Security function from anywhere in your Natural session without having to log on to the library SYSSEC. To do so, you enter the direct command - prefixed by "SYSSEC" - in the Natural command line. After the function invoked by the direct command has been performed, you will be returned to the Natural screen from which you have issued the command.

The following new direct commands - and keywords in NSCCMD01 - are available to invoke Administrator Services functions directly:

New Direct Command	Function Invoked
ADMIN_D	Library And User Preset Values
ADMIN_I	Application Programming Interfaces
ADMIN_N	Maintenance Log Records
ADMIN_S	System-Library Definitions
ADMIN_U	User Default Profiles
ADMIN_X	Utility Defaults/Templates
ADMIN_Y	Library Default Profiles
ADMIN_1	Environment Profiles

Link Functions

With earlier versions, when you invoke a function for the maintenance of links, you would get a list of all objects to which the selected object can be linked, that is, those for which links already exist and those for which not.

As of this version, a new option **Select only defined links** is provided; it allows you to display either a list of all linkable objects or a list of only those objects which are already linked. This new option is available for all link maintenance functions; it appears in the window displayed when you invoke a link function.

Administrator Services

The following enhancements are provided:

- [New General Options](#)
- [Logon Records](#)
- [Logon/Countersign Errors](#)
- [System-Library Definitions](#)
- [Environment Protection](#)

New General Options

Record Each User's Initial Logon Daily

The new general option **Record Each User's Initial Logon Daily** may be used to detect unused user IDs, that is, user security profiles which have not been used for a long time. This may be helpful when you decide to delete user security profiles which are no longer used.

When this option is active, the new application programming interface NSCXRUSE (see [below](#)) can be used to obtain a list of users who have not logged on since a specified date.

Allow Deletion of Users Who Are Owners/DDM Modifiers

With previous versions, it was not possible to delete a user profile as long as the user was specified as owner or DDM modifier in any security profile.

As of this version, you can use the new general option **Allow Deletion of Users Who Are Owners/DDM Modifiers** to determine whether or not the security profile of a user who is owner or DDM modifier can be deleted.

Generation of ETIDs

The ETID option in the **Library and User Preset Values** has been enhanced: The new setting "S" causes an ETID to be generated by Natural Security for every user at the start of his/her Natural session. The ETID contains a time-stamp component, which makes it unique, and it will remain in effect for the entire session. See *Library and User Preset Values* in the *Natural Security* documentation for details.

A similar option **Time-Stamp-Related ETID** is available for Natural RPC servers (see [below](#)).

Profile Parameter Override of Library-Profile Settings

For the following **Library and User Preset Values**, an additional option is available: **Active cross-reference for Predict and Natural programming mode**. You can specify an asterisk (*) instead of their other possible values. This will then apply to all libraries as follows:

- The generation of active cross-references will be determined by the value of the Natural profile parameter `XREF`, regardless of the “cross-reference” setting in individual library profiles.
- The programming mode will be determined by the value of the Natural profile parameter `SM`, regardless of the “programming mode” setting in individual library profiles.

Profile Parameters for Undefined Libraries

A new **Library and User Preset Values** option, **Profile parameters for undefined libraries**, is available. Provided that the general option **Transition Period Logon** is set to "Y", this new option applies to all libraries for which no security profiles have been defined yet. If you set it to "Y", the permission to use Natural system commands (which for defined libraries is determined by the security option **Allow system commands** in the library profiles) will be determined by the Natural profile parameter `NC`.

Set *APPLIC-NAME Always to Library Name

With previous versions, the Natural system variable `*APPLIC-NAME` either contained the name of the library to which the user was logged on, or, if the user was logged on via a special link, the special-link name.

With this version, a new general option **Set *APPLIC-NAME always to library name** is available. It can be set so that `*APPLIC-NAME` always contains the library name, regardless of whether the user is logged on via a special link or not.

Mailboxes in Batch Mode

With previous versions, mailboxes are not output in batch mode.

With this version, a new general option **Suppress mailboxes in batch mode** is available. It determines whether mailboxes are output in batch mode or not.

Logon Records

Similar to logon records for users and libraries, it is now possible to write records when users invoke a utility. The writing of such utility access records is activated by the new session option **Access Recorded** in the utility's default profile (see [below](#)). The records can be reviewed with the **Logon Records** function.

Logon/Countersign Errors

The **Logon/Countersign Errors Menu** provides the new selection options **Date from/to** and **Time from/to**: They allow you to restrict the range of error records to which a function is applied to logon/countersign errors which occurred in a specific period of time.

Moreover, the selection option **Order of Records** has been enhanced: In addition to logon and countersign errors, you can select utility access errors.

The functions for the handling of logon/countersign error records have been enhanced. They allow you to selectively handle logon errors which occurred in conjunction with Natural RPC service requests and Natural Web I/O service requests. To do so, you specify the following in the **Start Value** field on the **Logon/Countersign Errors Menu**:

- RPCSRVRQ - for logon errors in conjunction with Natural RPC service requests.
- NWOSRVRQ - for logon errors in conjunction with Natural Web I/O service requests.

System-Library Definitions

Several new system libraries (that is, libraries whose names begin with "SYS") are provided with this Natural version. They are included in the list of libraries provided by the Administrator Services function **System-Library Definitions**, which you can use to automatically create security profiles for them.

Environment Protection

With this version, Natural Security allows you to make users' access to a library environment-specific. A Natural environment is determined by the combination of the system files FNAT, FUSER, FSEC and FDIC. You define a security profile for each environment (that is, for each system-file combination) you wish to protect, and control users' access to it. You can also make a library accessible in some environments, but not in others.

Whenever a user logs on to a library in another environment, Natural Security will check whether:

- access to the library is allowed in that environment, and
- the user is authorized to access that environment.

Such a check is performed not only when a user explicitly logs on to a library, but also when the user invokes a function which implicitly accesses another library or processes the contents of another library.

Environment protection is activated with a new general option **Environment Protection**. When environment protection is active, the following applies:

- Access to undefined environments is not possible.

- For every environment to be accessed, an environment security profile has to be defined.
- By default, access to a library is allowed in any defined environment, and access to a defined environment is allowed for all users.
- For individual libraries and users, you can disallow access to a defined environment.

See the section *Protecting Environments* in the *Natural Security* documentation for details on environment protection.



Note: The above-mentioned environment-protection functionality replaces the Administrator Services function **Definition of System-File Access** which was available with earlier versions of Natural Security.

User Profiles

Copying User's Links

The new user maintenance function **Copy User's Links** allows you to copy links which are defined for one user profile to another existing user profile. The function allows you to individually select the links you wish to copy.

Library Profiles

The following enhancements are provided:

- Private Libraries
- Linking Administrators to an Unprotected Library
- Copying Libraries
- Copying, Renaming and Deleting Libraries
- Default Profiles
- Statement and Command Restrictions
- Library SYSEXRM

- [Copy Library Profile](#)

Private Libraries

With earlier versions, access to a private library is restricted to the user for whom the private library is defined.

As of this version, it is possible to remove this restriction and control access to private libraries in the same way as access to other “normal” libraries.

With the new general option **Private Libraries in Public Mode**, you can choose “public mode”. In this mode, private libraries are handled like any other libraries:

- You can choose not to protect a private library, which means that it can be accessed by any user.
- You can make a private library protected, which means that it can only be accessed by the user whose ID is the same as the library ID, and by users who are linked to it.

For details, see the description of the additional option **Private Library** in *Components of a User Profile*, which can be found under *User Maintenance* in the *Natural Security* documentation.

In private mode, private libraries continue to be maintained in the user maintenance section of Natural Security; in public mode, private libraries are maintained in the library maintenance section.

Linking Administrators to an Unprotected Library

With earlier versions, it is not possible to establish a link between a user and an unprotected library. The conditions of use of the library are determined by the library profile.

As of this version, it is possible to establish a special link between a group and an unprotected library. This special link applies only to administrators contained in that group. Thus it is possible to define special conditions of use for administrators if this should be required for administration or maintenance tasks. For this purpose, the protection combination in the library profile has to be set to “People-protected=L, Terminal-protected=N”.

Copying Libraries

The Copy Library provides a new option **with links**. It allows you to copy not only the library profile, but also existing links associated with that library profile.

Copying, Renaming and Deleting Libraries

With earlier versions, when you copy, rename or delete a library security profile, this has no effect on the library itself and its contents stored on the FUSER system file.

As of this version, the functions **Copy Library**, **Rename Library** and **Delete Library** provide a new option **with Natural objects** which allows you to also adjust the FUSER system file accordingly when a library profile is copied, renamed or deleted, which means that the contents of the library on the FUSER file are also copied to another library, moved to another library, or deleted.

Default Profiles

With earlier versions, default security profiles can only be defined for users. As of this version, you can also define default profiles for libraries.

Statement and Command Restrictions

The use of the new statements and system commands provided with the current version of Natural can also be controlled with Natural Security.

The following commands have been removed: ADHOC, CREATE, GRAPHICS and IDL.

Library SYSEXRM

The Natural example library SYSEXRM is now called SYSEXSYN. If you have a library profile defined for SYSEXRM, you should therefore rename it to SYSEXSYN.

Copy Library Profile

When you use the **Copy Library** function to copy a library profile, the **Free List of Modules** (which is obtained by pressing PF9 on the **Disallow/Allow Modules** screen of a library profile) is copied as well. With previous versions, the list was not copied.

Utility Profiles

The following enhancements are provided:

- [Session Options](#)
- [SYSDDM](#)
- [SYSOBJH - Object Handler](#)

- NATLOAD, NATUNLD, SYSTRANS

Session Options

The **Additional Options** section of the default utility profiles has been enhanced to provide the following session options:

Option	Explanation
Access Recorded	This option allows you to record access to a utility. It corresponds to the option Logon Recorded in user and library profiles. When a user invokes a utility, a record will be written. The Logon Records function of Administrator Services can be used to view these records.
Privileged Groups	This option may be used to influence the order in which Natural Security searches for the appropriate utility profile to apply. It determines whether or not utility profiles defined for the "privileged groups" (which are specified in a user security profile) will become part of the search order.
*GROUP Only	When a user invokes a utility function and Natural Security searches for appropriate utility profile to be applied, the search sequence, by default, includes user-library-specific and user-specific utility profiles of all groups in which the user is contained. With this option, you can restrict the search to utility profiles of the current group (as determined by the current value of the Natural system variable *GROUP) and exclude the utility profiles of other groups from the search sequence. See the section <i>Which Utility Profile Applies?</i> in the <i>Natural Security</i> documentation for details.
MAINUSER API	This option (which is only available for the SYSMAIN utility) controls the use of SYSMAIN functions invoked via the application programming interface MAINUSER. It allows you to create a separate set of utility profiles to allow/disallow the use of SYSMAIN functions when invoked via the MAINUSER API. These profiles are independent of the "normal" SYSMAIN utility profiles which control the use SYSMAIN functions when invoked via the SYSMAIN command.
Utilities option	This this option is only available for the Object Handler (SYSOBJH utility). It can be used to make the Utilities option in library profiles apply to SYSOBJH.

SYSDDM

As of this version, the use of the SYSDDM utility can be controlled by Natural Security utility profiles not only for Natural on mainframe computers, but also for Natural on UNIX and OpenVMS.



Note: With one of the next releases, SYSDDM utility profiles will also be applicable to Natural on Windows.

SYSOBJH - Object Handler

The utility profiles for SYSOBJH (Object Handler) provide the following enhancements:

- The Unload, Load and Delete functions can be allowed/disallowed selectively for mainframe DDMs, mainframe-related objects, and applications.
- The REPLACE parameter can be allowed/disallowed selectively for the Load function in user-library-specific profiles.

The initialization of the Natural utility SYSOBJH (Natural Object Handler) under Natural Security has been improved to make the use of the utility more user-friendly: With previous versions, users were not notified that they were not allowed to use a selected function/option until they actually attempted to execute it. Now a disallowed function/option is intercepted at the earliest possible stage in the selection process.

NATLOAD, NATUNLD, SYSTRANS

The Natural utilities NATLOAD, NATUNLD and SYSTRANS are no longer available (see also [Removed Features](#)). For compatibility reasons, however, utility protection for them is still possible: If utility profiles for these utilities already exist, you can continue to define and maintain them. If none exist, utility profile maintenance for these utilities will be disabled.



Note: NATLOAD and NATUNLD were only available on mainframe, UNIX and OpenVMS platforms.

RPC Server Profiles

The following enhancements are provided:

- [Time-Stamp-Related ETID](#)
- [Single-Library RPC Servers](#)

Time-Stamp-Related ETID

The new option **Time-Stamp-Related ETID** allows the generation of time-stamp-related ETIDs for RPC server sessions, in the same way as described above for user session (see [Generation of ETIDs](#) above).

Single-Library RPC Servers

For Natural RPC servers which provide services performed by subprograms contained in a single library, a new option **Logon Mode** is available. It can be specified in the security profiles of Natural RPC servers to improve performance.

Setting the option to "S" (Static Mode) has the following effects:

- The library on the server is set at the start of the server session, and will remain unchanged until the end of the server session.
- The server will process only service requests for this library. Service requests for any other library will be rejected.
- If the library is unprotected (People-protected = N), the user's authorization to access the library is not checked. If the library is protected (People-protected=Y), the user's authorization to access the library is checked.
- After a successful check, the user's conditions of use of the library are determined by the library profile. Even if a special link exists between the user and the library, any settings in the special link profile will be ignored.

See the section *Validation of an RPC Service Request* in the *Natural Security* documentation for details.

Application Programming Interfaces

The following enhancements are provided:

- [Enhancements to All APIs - Error Texts](#)
- [New APIs](#)
- [Enhanced APIs](#)

Enhancements to All APIs - Error Texts

All Natural Security application programming interfaces (APIs) have been enhanced: In the case of an error, they now provide the option to return not only the error message number, but also the associated error text. For details, see the parameter descriptions in the source codes of the API examples provided in the library `SYSSEC`.

New APIs

As of this version, “interface subprograms” are called “application programming interfaces”.

The following new application programming interfaces (APIs) are provided:

API	Function
NSCADM	This application programming interface is used to: <ul style="list-style-type: none"> ■ Display the settings of General Options in Natural Security's Administrator Services. ■ List and delete logon records. See the example program PGMADM02 in the library SYSSEC for details. ■ Remove/re-establish Natural Security's security profile maintenance/retrieval sections for base/compound applications and RPC servers. See the example program PGMADM03 in the library SYSSEC for details.
NSCXRUSE	This application programming interface is used to: <ul style="list-style-type: none"> ■ Display users whose number of unsuccessful logon attempts is greater than "0". ■ Display users who have not logged onto Natural since a specified date. See also Record Each User's Initial Logon Daily above.
NSCXLO	This application programming interface is used to read the maintenance log records which are created by Natural Security if the general option Logging of Maintenance Functions has been activated.
NSC---P	This application programming interface is used for password verification.
NSC----P	This application programming interface is used for password verification and password change.
NSCXRIER	This application programming interface is used to display individual logon error records.

Enhanced APIs

The following application programming interfaces (APIs) have been enhanced:

API	Function
NSC---L	This application programming interface now allows you to ascertain which modules in a library are available to a user. For details, see the example program PGM---LM in the library SYSSEC.
NSCUS	With previous versions, when you used the application programming interface NSCUS to modify the owner specifications in a user profile, any existing owner specifications in the profile were overwritten by the newly specified ones. As of this version, NSCUS allows you to choose between either overwriting the existing owners entries or appending the new owner entries to the existing ones without overwriting them.

API	Function
NSCXR	<p>This application programming interface now allows you to do the following:</p> <ul style="list-style-type: none"> ■ Retrieve the user ID belonging to a specified user name. For details, see the example program PGMXR014 in the library SYSSEC. ■ List all libraries of the current FNAT or FUSER system file which are not defined in Natural Security. For details, see the example program PGMXR015 in the library SYSSEC. ■ Translate the 2-character object-type code into the corresponding object type. For details, see the example program PGMXR016 in the library SYSSEC. ■ Display a list of users who are linked to an external object. For details, see the example program PGMXR017 in the library SYSSEC.

SECULD2/SECLOAD Transfer Programs

The following enhancements are provided:

- [SECULD/SECULD2](#)
- [New Features for SECULD2](#)
- [Conversion Table and EBCDIC-ASCII Conversion](#)
- [Batch Processes](#)

SECULD/SECULD2

SECULD2, which is a copy of SECULD with additional functionality, provides enhanced selection criteria to determine the range of objects to be unloaded.

It is recommended that you use SECULD2 instead of SECULD. SECULD will be removed in a future version.

New Features for SECULD2

New Option

SECLOAD provides a new option **Simulate Loading** which allows you to ascertain whether all data from the work file can be loaded to the system file, before you actually load them.

Enhanced Selection Criteria

With some SECULD functions, asterisk notation cannot be used in the **Start Value** field, for example, functions related to external objects. The reason is that asterisk notation would not work with external objects, where an asterisk (*) can be a valid character as part of the object ID itself.

For the sake of consistency, asterisk notation for any function or any object type is no longer possible in the **Start Value** field. Instead, an additional **Range** field is provided, which allows

you to determine how the value you specify in the **Start Value** field is to be treated. Three options are possible:

- The range of objects begins with the one whose object ID begins with the Start Value (“genuine” start value).
- The range of objects comprises only those whose IDs begin with the Start Value (corresponds to asterisk notation).
- The range of objects comprises only the one object whose ID is specified as Start Value.

For the unloading of links between users and objects (function code “L”), an additional selection criterion can be specified to determine the range of links to be unloaded: A new field **Link ID** is provided, where you can specify a user ID to unload only links of a certain user or range of users.

A second **Range** field is provided, which allows you to determine how the value you specify in the **Link ID** field is to be treated. Its options are in analogy to those of the Start Value’s **Range** field.

For function code “L”, the **Start Value** and **Link ID** fields, and their corresponding **Range** fields, can be used in combination with one another.

Conversion Table and EBCDIC-ASCII Conversion

With previous versions, SECULD/SECLOAD used the corresponding unload/options of the SYSTRANS utility for character conversion. With the new version of SECULD2/SECLOAD, an API subprogram NSCCONV in the library SYSSEC is provided for this purpose. You can adjust the source of this subprogram to suit your requirements.

Batch Processes

The new unloading program with the enhanced functionality as described above is called SECULD2. This means that you have to adjust your batch processes to invoke SECULD2 instead of SECULD.

To ease the transition, the old SECULD program is - for the time being - still available in the library SYSSEC. It will be removed in a future version of Natural Security.

Logon Procedure

The following enhancements are provided:

- [Unsuccessful Logon Attempts](#)
- [Password Expiration Message](#)

Unsuccessful Logon Attempts

As of this version, the number of unsuccessful logon attempts is passed as a parameter to the logon-related user exit LOGONEX1. Thus, it is possible, for example, to display corresponding information to the user before the maximum number of logon attempts is reached. For details, see the source of LOGONEX1.

Password Expiration Message

As of this version, you can issue a warning message “Your password will expire on date” (NAT1691) to users at the initial logon. The output of this message is activated with the new option **Message Before Password Expiration** in Administrator Services.

Functional Security

Status of Command Processor

If the status of a command processor is “modified” (that is, modified with `SYSNCP`), you have to update the functional security defined in Natural Security for the command processor. With earlier versions, you have to make this adjustment for each command processor individually. As of this version, the new function UC of application-interface subprogram NSCLI allows you to simultaneously update the functional security of all “modified” command processors in a library.

Limitations on OpenVMS

In some places, the documentation set for Natural for OpenVMS mentions features which are (currently) not supported with Natural for OpenVMS. Such features are:

- [Enable Error Transaction Before NAT1700/1701 Logoff](#)
- [Suspend Line Protection](#)
- [Time Differential and Time Zone](#)
- [Adapters](#)

- [Support of Integrated Authentication Framework](#)

Enable Error Transaction Before NAT1700/1701 Logoff

Restricted to mainframes.

The new general option **Enable Error Transaction Before NAT1700/1701 Logoff** allows you to have the current application's relevant `ON ERROR` statement and/or error transaction (`*ERROR-TA`) processed in the case of Natural errors NAT1700 (time window exceeded) and NAT1701 (non-activity time limit exceeded). This option only applies on mainframe computers; on non-mainframe platforms, its setting has no effect.

Suspend Line Protection

Restricted to Windows.

The Natural Studio program editor function **Suspend Line Protection** can be allowed/disallowed in the Session Options section of user security profiles.

Time Differential and Time Zone

Restricted to mainframes.

With earlier versions, the **Time Differential** option in user security profiles is only available for users of type `TERMINAL`.

As of this version, it is also available for `GROUPs`. In addition, a new option **Time Zone** is provided, which can be used as an alternative to **Time Differential**.

Adapters

Restricted to Windows.

The use of the new Natural object type “adapter” can also be controlled by Natural Security: This is done by allowing/disallowing the editing of the object type “adapter” in the **Editing Restrictions** section of library profiles.

Support of Integrated Authentication Framework

Restricted to Non-OpenVMS environments.

As of this version, Natural Security supports Natural RPC servers which use an Integrated Authentication Framework (IAF) server for token validation. See also [Limitations on OpenVMS](#) in the RPC section of these *Release Notes*.

See the section *IAF Support* in the *Natural Security* documentation for details.

9 Removed Features

▪ Remote Debugging	74
▪ Versioning Software	74
▪ Natural Web Interface	74
▪ Utilities NATUNLD, NATLOAD and SYSTRANS	75
▪ Profile Parameter V22COMP	75
▪ DDM Server	75
▪ Multi-Fetch Disabling	75
▪ Profile Parameter DSLM - Size Limitation	75
▪ Profile Parameter LFILMAX - Maximum Number of Dynamic Specifications of LFILE Parameter	76
▪ Entire Screen Builder Interface	76

This chapter provides an overview of functionality no longer supported with this Natural version.

Remote Debugging

Remote debugging (that is debugging a native Natural for OpenVMS application from a Windows computer) is no longer supported. The Windows client is no longer delivered.

Debugging an OpenVMS application is only possible with Natural Studio when you are working in a remote OpenVMS environment using SPoD.

Versioning Software

If you want to use third-party versioning software together with Natural, it is no longer required that you write your own interface. Therefore, NATNCVC is no longer supported.

It is recommended that you use Local Versioning which is available with Natural Studio when you are working in a remote OpenVMS environment using SPoD. Using Local Versioning, you can use the source control system Concurrent Versions System (CVS) or Subversion (SVN) from within your Natural environment. To use this functionality, you must have the Subversion or CVS client part (*svn.exe* or *cvs.exe*) installed on your PC and you must have access to a Subversion or CVS repository server which has been installed locally or on a server. When you search the internet, you can find these Subversion or CVS components on several download sites. For detailed information on Local Versioning, see *Natural Studio Extensions* which is part of the Natural for Windows documentation.

Natural Web Interface

Natural 6.3 is the last version to deliver SYSWEB.

The interface SYSWEB3 is now to be used instead of SYSWEB. SYSWEB is only delivered on the product CD and can only be installed manually. It is therefore recommended that you update your programs written for SYSWEB to run with the new interface SYSWEB3. See *Migrate Natural Web Interface SYSWEB to SYSWEB3* in the *Web Technology* documentation for further information.

Utilities NATUNLD, NATLOAD and SYSTRANS

Natural 6.3 is the last version to deliver the utilities NATUNLD, NATLOAD and SYSTRANS.

The utilities NATUNLD, NATLOAD and SYSTRANS are no longer installed by default. The Natural Object Handler is to be used instead.

However, if you want to continue using these utilities, you can still issue the system commands which invoke these utilities (SYSUNLD and SYSTRANS). When you issue such a system command for the first time, you have the possibility to install the corresponding utility. Follow the installation instructions displayed on the screen. When the installation for a utility has been completed, you can invoke it using the corresponding system command.

The documentation for these utilities can be found as a PDF book in the archive of the Natural Documentation CD.

Profile Parameter V22COMP

V22COMP is no longer supported.

DDM Server

The dictionary server DDM Server is no longer supported.

Multi-Fetch Disabling

The global configuration setting Multi-Fetch Disabling is no longer supported.

Profile Parameter DSLM - Size Limitation

The parameter DSLM which was used for compatibility with Natural Version 3.1 for Mainframes is no longer necessary and has therefore been removed.

Profile Parameter LFILMAX - Maximum Number of Dynamic Specifications of LFILE Parameter

The parameter `LFILMAX`, which was used to specify the maximum number of dynamic specifications of the profile parameter `LFILE`, is no longer necessary and has therefore been removed. On the mainframe, this parameter has already been removed in a previous version.

Entire Screen Builder Interface

The Entire Screen Builder interface is not part of the installation kit.

10 Notice of Future Changes

- Default Setting for Adabas Database Driver 78
- Handling of the TF Parameter 78

The following changes are planned for future versions of Natural.

Default Setting for Adabas Database Driver

With Natural Version 6.3.4, the database driver ADA is still used by default.

With a future version of Natural, the default setting for the database driver will change to ADA2.

Handling of the TF Parameter

With a future version of Natural, the handling of the TF parameter will be changed.

Currently, it is only possible to use the asterisk (*) notation for the database ID (DBID) and file number (FNR) of the production database. In a future version, it will also be possible to use the asterisk notation for the database ID and file number of the test database. It may then be possible that your applications will behave in a different way.

11 Release Information for Natural Version 6.3.5

▪ New Features	80
▪ Changes and Enhancements	81
▪ Known Issues	82
▪ Natural Remote Procedure Call (RPC)	82
▪ Natural Security	85
▪ Documentation	90

New Features

Parameters

The following new Natural parameters are provided in this version:

Parameter	Description
RPCSDIR	Specifies the name of the Natural library in which the service directory is located.
SRVCMIT	Specifies the time at which a Natural RPC server automatically commits an RPC conversation or a non-conversational RPC request.
SRVTERM	Specifies the event at which a Natural RPC server is automatically terminated.

Application Programming Interfaces

The utility SYSEXT provides the following new application programming interfaces (APIs):

API	Description
USR4005N	Read current PF-key settings.
USR4212N	Read a data area from the system file/file system and return the single entries in a table. This allows you to analyze a data area independent of the way the definitions were done in the data area editor. It also allows you to process data areas in DEFINE DATA format.
USR6304N	Set/get the reliable state for the reliable Natural RPC.
USR6305N	Commit/rollback reliable RPC message(s). This API is required if the reliable RPC state has been set to "client commit".
USR6306N	Retrieve the status of all reliable RPC messages of the user who is currently logged on to the EntireX Broker.

Configuration Tool for the Natural Web I/O Interface Client

A configuration tool is now available on J2EE servers. It is used to manage the contents of the configuration files for the session (*sessions.xml*) and for logging (*natlogger.xml*). See *Using the Configuration Tool* in the *Natural Web I/O Interface* documentation.

Changes and Enhancements

Configuration Utility

Remote Procedure Call

The new profile parameters `RPCSDIR`, `SRVCMIT` and `SRVTERM` can be specified. See *Remote Procedure Call* in the *Configuration Utility* documentation.

Session Configuration for the Natural Web I/O Interface Client

A number of features that was previously controlled by an XSLT file (colors, fonts, PF key buttons) is now controlled by a style sheet (CSS). As of this version, the XSLT file only provides restricted possibilities (see *Modifying the Field Attributes* in the *Natural Web I/O Interface* documentation). The XSLT files of the previous versions are no longer supported.



Note: As of version 6.3.7, the above chapter is no longer available.

The new URL parameter `natparamext` extends an existing Natural parameter definition in the configuration file. The extension works in the following way: the Natural parameters defined in the configuration file come first. Then, the Natural parameters defined in the URL parameter `natparamext` are added, separated by a space character. If you want to overrule the definition in the configuration file, use the URL parameter `natparam` instead. See *Starting a Natural Application with a URL* in the *Natural Web I/O Interface* documentation.

It is now possible to define the following new settings in the configuration file for the sessions. The headings below correspond to the options that are used in the new configuration tool (see *Overview of Session Options* in the *Natural Web I/O Interface* documentation). These new options are only available for J2EE servers, not for IIS.

Use SSL

You can enable SSL. A secure connection is then established between the Natural Web I/O Interface client on the application server and the Natural Web I/O Interface server.

Corresponds to the `ssl` attribute of the `session` element in *sessions.xml*.

Show function key numbers

You can determine whether the PF key numbers are shown next to the PF keys.

Corresponds to the `showfkeynumbers` attribute of the `screen` element in *sessions.xml*.

It is now possible to specify a trust file in the configuration file for the sessions. Trust files are used for a secure connection between the Natural Web I/O Interface server and the Natural Web I/O Interface client. The headings below correspond to the options that are used in the new configuration tool (see *Global Settings* in the *Natural Web I/O Interface* documentation). These new options are only available for J2EE servers, not for IIS.

SSL trust file path

The path to your trust file. For further information, see *Trust Files (J2EE only)* in the *Natural Web I/O Interface* documentation.

Corresponds to the `trustfile_name` element of the `global` section in *sessions.xml*.

SSL trust file password

If your trust file is password-protected, the appropriate password is required.

Corresponds to the `trustfile_password` element of the `global` section in *sessions.xml*.

Known Issues

The information provided for Version 6.3.4 in the section *Known Issues* still applies for Version 6.3.5. In addition, the following applies:

EntireX Broker Stub for Natural Remote Procedure Call (RPC)

Natural Remote Procedure Call (RPC) cannot be used until the EntireX Broker Stub is available.

Support of Integrated Authentication Framework (IAF) on Server Side

The Integrated Authentication Framework cannot be used by a Natural RPC server until the EntireX Broker Stub is available.

Natural Remote Procedure Call (RPC)

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

With Natural Version 6.3.5, an enhanced Natural Remote Procedure Call Version 6.3.2 is delivered that replaces the existing Natural RPC Version 6.3.1.

See also *Known Issues* above.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the following changes, enhancements and new features are provided:

- Profile Parameters
- Reliable RPC
- Support Logging and Accounting of RPC Program and RPC Library within the EntireX Broker

- Availability of *SERVER-TYPE=RPC Enhanced
- Dynamic Resize of Buffer (MAXBUFF)
- New RPC-Specific Application Programming Interfaces
- Support of Integrated Authentication Framework (IAF) on Server Side

Profile Parameters

The following new profile parameters are available:

- SRVCMIT - Server Commit Time
- SRVTERM - Server Termination Event
- RPCSDIR - Library for Service Directory

The following profile parameter has been changed:

- MAXBUFF - Maximum Buffer Size

For further information, see *Dynamic Resize of Buffer* and the corresponding parameter description in the *Parameter Reference*.

Reliable RPC

Reliable RPC is the Natural RPC implementation of a reliable messaging system. It combines the Natural RPC technology and persistence, which is implemented by means of units of work that are offered by the EntireX Broker. Reliable RPC is characterized by following features:

- The Natural RPC client executes a `CALLNAT` statement without waiting for a reply from the server (the RPC message is sent in asynchronous mode).
- An RPC server needs not be active at the time the `CALLNAT` is executed.
- The reliable RPC message is stored in the Broker's persistent store until an RPC server is available.
- The Natural RPC server executes the reliable RPC by calling the requested subprogram but does not send a reply to the RPC client.
- A Natural RPC client may ask the status of the sent reliable RPC messages.
- A Natural RPC client may send a reliable RPC message to an EntireX RPC server.
- A Natural RPC server may receive a reliable RPC message from an EntireX RPC client.

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

Support Logging and Accounting of RPC Program and RPC Library within the EntireX Broker

The Natural RPC client provides the name of the subprogram that is to be executed and the name of the library from which the subprogram is to be executed to the EntireX Broker.

The Natural RPC server returns the name of the subprogram that has been executed and the name of the library from which the subprogram has actually been executed.

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

Availability of *SERVER-TYPE=RPC Enhanced

The Natural RPC server shows the system variable content *SERVER-TYPE=RPC already during the processing of the commands that have been placed on the Natural stack with the Natural profile parameter STACK. With this enhancement, all Natural objects that are executed by the Natural RPC server can check the system variable *SERVER-TYPE for RPC.

In previous versions, *SERVER-TYPE=RPC was only available during the execution of an RPC request.

Dynamic Resize of Buffer (MAXBUFF)

The size of the buffer which is used to exchange data between client and server will be dynamically increased on demand. The size specified with the profile parameter MAXBUFF is used as default value.

This measure will avoid most Natural errors that are reported with Natural error message NAT6964 and reason codes 4, 5 and 7.

New RPC-Specific Application Programming Interfaces

The following RPC-specific application programming interfaces have been added:

- USR6304N - Set/get the reliable state for the reliable Natural RPC
- USR6305N - Commit/rollback reliable RPC message(s). This API is required if the reliable RPC state has been set to "client commit"
- USR6306N - Retrieve the status of all reliable RPC messages of the user who is currently logged on to the EntireX Broker.

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

Support of Integrated Authentication Framework (IAF) on Server Side

If Natural Security is installed on the Natural RPC server side and if the EntireX Broker uses IAF for authentication, the Natural RPC server can optionally be configured to use an IAF token for client authentication instead of the Natural Security logon data. The IAF token is provided by the EntireX Broker and contains the user ID that the client has used to log on to the EntireX Broker. As a consequence, after a successful authentication the Natural user ID *USER is always identical to the client user ID used by the EntireX Broker. It is no longer possible to use a user ID within Natural that is different from the client user ID used by the Entirex Broker.

To use this feature, the Natural RPC server and IAF must be configured in Natural Security. See the section *Protecting Natural RPC Servers and Services* in the *Natural Security* documentation for details.

No changes are required on the client side.



Note: The **limitations** for the Integrated Authentication Framework (IAF) as described for Version 6.3.4 no longer apply.

Natural Security

The following enhancements are provided with Natural Security Version 6.3.5:

- Administrator Services
- Libraries
- DDMs
- Utilities
- Natural RPC Server Profiles
- Other Enhancements

Administrator Services

The following enhancements are provided in Administrator Services:

- Logging of Maintenance Functions
- Maintenance Log Records
- Module Protection Mode
- Definition of Undefined Libraries

- [Disable Rename and Delete of Library Node](#)

Logging of Maintenance Functions

The general option **Logging of Maintenance Functions** has been enhanced: When you activate the logging for user and library security profiles, you have the option to log the following additional data (extended logging):

- When the functions Copy User and Copy Library are used with the **with links** option, any relationship which the copying has established between the user/library and other objects is logged.
- When the Delete User function is used, any relationship which existed between the user and other objects and which was removed by the deletion is logged.
- When a link between a group and a library is maintained, a list of the group's members is logged.
- When the Disallow/Allow Modules section of a library (or special link) profile is maintained, information on the changed status of any module is logged.

A new **Log File Maintenance** function, **List Log Records Extended**, is available to view the additional data.

The modifications which are logged for a type of security profiles now also include the transfer of security profiles of this type via the Natural Security data transfer programs SECULD2 and SECLoad.

Maintenance Log Records

With the **Maintenance Log Records** functions you can display for each log record the screen which was modified. These screens now also show the Natural Security version and the FSEC system file with/on which the modification was performed.



Note: This information is not shown for log records which were written with Natural Security versions prior to 4.2.5 on mainframes and 6.3.5 on non-mainframes.

Module Protection Mode

The new option **Module Protection Mode** affects how the **Disallow/Allow Modules** settings in library security profiles are evaluated: You can set it so that they are evaluated in the same way on mainframes and non-mainframe platforms. This may be useful if you transfer a Natural application from a mainframe to a non-mainframe platform and wish to keep your module protection unchanged.

For details, see *Module Protection Mode* under *Library and User Preset Values* in the *Administrator Services* section of the *Natural Security* documentation.

Definition of Undefined Libraries

The new Administrator Services function **Definition of Undefined Libraries** serves the same purpose as the library maintenance enhancement described under [Undefined Libraries](#) below.

Disable Rename and Delete of Library Node

The new option **Disable Rename and Delete of Library Node** allows you to prevent the inadvertent deletion/rename of a library in the mapped environment of the Natural Development Server. If it is set, the actions Rename and Delete cannot be selected from the context menu of the library node.

For details, see *Disable Rename and Delete of Library Node* under *Library and User Preset Values* in the *Administrator Services* section of the *Natural Security* documentation.

Libraries

The following enhancements are provided for libraries:

- [Source Locking](#)
- [Undefined Libraries](#)

Source Locking

Source locking in the case of concurrent updates of Natural source members - as controlled for the Natural session by the Natural profile parameter `SLOCK` - can now also be controlled for individual libraries by a corresponding setting in the *Session Parameters* section of library profiles.

Undefined Libraries

Library maintenance has been enhanced allowing you to search for undefined libraries - that is, libraries which exist on the system file, but for which no security profiles have been created in Natural Security: You can expand the Library Maintenance selection list to list either all (defined and undefined) libraries or only the undefined ones. You can apply the search for undefined libraries to the current `FUSER` system file or to another system file of your choice.

For details, see *Listing Undefined Libraries* in the *Library Maintenance* section of the *Natural Security* documentation.

DDMs

The following enhancement is provided for DDMs:

- [Support of FDDM Profile Parameter on Non-Mainframes](#)

Support of FDDM Profile Parameter on Non-Mainframes

If a central system file for non-mainframe DDM storage (outside of libraries) is specified with the Natural profile parameter `FDDM`, the protection of non-mainframe DDMs and the maintenance of their security profiles is performed in the same way as with mainframe DDMs (as described in the section *Protecting DDMs On Mainframes* of the *Natural Security* documentation).

Utilities

The following enhancements are provided for utilities:

- [All Utility Profiles](#)
- [SYSDDM](#)
- [SYSMAIN](#)
- [SYSOBJH - Object Handler](#)

All Utility Profiles

In a utility profile, you allow or disallow each option by marking it with "A" or "D" respectively. For ease of maintenance, you can now set all options in a utility profile simultaneously to "A" or "D" by pressing PF16 or PF17 respectively.

SYSDDM

A new `SYSDDM` function **SQL Services (NSB)** for Natural SQL Gateway support is provided with this Natural version. Its use can also be controlled in `SYSDDM` utility profiles.

SYSMAIN

In the default utility profile of the `SYSMAIN` utility, a new **Additional Option** named **Utilities Option** is available. With it, you can make the **Utilities** option in library profiles apply to `SYSMAIN`.

SYSOBJH - Object Handler

Several new Object Handler functions are provided with this Natural version. Their use can also be controlled in SYSOBJH utility profiles.

Natural RPC Server Profiles

Support of Integrated Authentication Framework

As of this version, Natural Security supports Natural RPC servers which use an Integrated Authentication Framework (IAF) server for token validation. See also [Support of Integrated Authentication Framework \(IAF\) on Server Side](#) in the RPC section of these Release Notes.

See also [Known Issues](#) above.

See the section *IAF Support* in the Natural Security documentation for details.



Note: The **limitations** for the Integrated Authentication Framework (IAF) as described for Version 6.3.4 no longer apply.

Other Enhancements

The following other enhancements are provided:

- [Selection Criterion for Link Functions](#)
- [Control Use of TEST Command in Environments](#)
- [Application Programming Interface NSCLI Enhanced](#)
- [New User Exit for LOGOFF](#)

Selection Criterion for Link Functions

Whenever you invoke a link function, a window appears before the list of objects to be linked is displayed. This window, which also allows you to specify a start value for the list to be displayed, used to provide an option **Select only defined ones**. This option has been enhanced and is now called **Selection Criterion**: It allows you to select whether the list is to contain:

- all objects (linked and not linked),
- only objects which are already linked,
- only objects which are not yet linked.

This enhancement is provided for all Natural Security maintenance functions available to link users to libraries, users to applications, users to external objects, and libraries to files.

Control Use of TEST Command in Environments

A new option in the security profiles of environments allows you to control the use of the Natural system command `TEST` in an environment. You can allow or disallow it altogether, or restrict the use of the debugger. For details, see *Components of an Environment Profile* in the section *Protecting Environments* of the *Natural Security* documentation.

This option only applies to environments on mainframe computers.

Application Programming Interface NSCLI Enhanced

The application programming interface (API) `NSCLI` has been enhanced: It allows you to list library profiles (and special link profiles) which contain a specific Adabas password, and you can then change the password. For details, see example program `PGMLI006` and text member `TXTLI006` in the library `SYSSEC`.

New User Exit for LOGOFF

A new user exit, `LOGONEX5`, is available; it is invoked by the Natural Security logon program whenever the `LOGOFF` system command is executed.

Documentation

Editors

The documentation for the program editor and for the data area editor has been revised. See *Program Editor* and *Data Area Editor* in the *Editors* documentation.

Natural Web I/O Interface

The documentation for the Natural Web I/O Interface has been revised.

An introduction to the Natural Web I/O Interface is now available.

In addition to the information about the new configuration tool, the installation and configuration information for the Natural Web I/O Interface daemon has been moved to the *Natural Web I/O Interface* documentation. This information was previously available in the *Installation* documentation.

The general term “Natural Web I/O Interface server” is now used instead of “Natural Web I/O Interface daemon”.

12

Release Information for Natural Version 6.3.6

▪ Changes and Enhancements	92
▪ Natural Remote Procedure Call	92
▪ Natural Security	93
▪ Notice of Future Changes	93

Changes and Enhancements

Utilities

Object Handler

The Object Handler now provides the commands `NATUNLD`, `NATLOAD` and `SYSTRANS`, which are used to migrate the old utilities `NATUNLD`, `NATLOAD` and `SYSTRANS` to the Object Handler. For details, see *Migration from NATUNLD/NATLOAD and SYSTRANS to the Object Handler* in the *Object Handler* section of the *Utilities* documentation.

The Object Handler now provides the `UNDELI` command to process delete instructions with the unload function. For details, see *Basic Command Syntax* in the *Object Handler* section of the *Utilities* documentation.

Natural Web I/O Interface Client

Support is now provided for automatic style sheet selection for different screen models. See *Modifying the Font Size* in the *Natural Web I/O Interface* documentation.

On IIS, you can now determine whether the PF key numbers are shown next to the PF keys. This is done with the `showkeynumbers` attribute of the `screen` element in `sessions.xml`. See *Overview of Configuration File Elements* in the *Natural Web I/O Interface* documentation. (This feature is already available for J2EE.)

Natural Remote Procedure Call

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

With Natural Version 6.3.6, an enhanced Natural Remote Procedure Call Version 6.3.3 is delivered that replaces the existing Natural RPC Version 6.3.2.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the following changes, enhancements and new features are provided.

Trace Enhancement

In case of a conversion error, which is reported with Natural error number NAT6974 and reason codes 2 and 3, the position of the erroneous data in the buffer is indicated. This will help you to identify which application data is causing the conversion error.

For further information, see *Using the Server Trace Facility* in the *Natural Remote Procedure Call (RPC)* documentation.

Natural Security

The following enhancements are provided with Natural Security Version 6.3.6.

Logon Records Enhanced

The information stored in the logon records has been expanded: now it also includes the user type and system-file information (FUSER, FNAT, FDIC). With this Natural Security version, this information can only be retrieved via the application programming interface NSCXR; see the example program PGMXR006 in the library SYSSEC.

Notice of Future Changes

The following changes are planned for future versions of Natural.

Change in Return Code Handling

With a future version of Natural, the return code handling of batch Natural processing will be changed:

- A `TERMINATE nnn` statement will cause `nnn` to be always set as the return code of a Natural run.
- The return code set by a `TERMINATE` statement will override the return code 61 which is set due to a runtime error. Currently, return code 61 overrides the return code set by a `TERMINATE` statement.
- Error message NAT9987 will appear in the file identified by Natural parameter `CMPRINT` if a Natural run was stopped by a `TERMINATE` statement. Currently, the message NAT9978 is displayed if either a runtime error occurred or the session was terminated by executing a `TERMINATE` statement.

Review your batch procedures and check whether there are dependencies related to the above mentioned changes.

Natural Web I/O Interface Client on IIS

With a future version of Natural, it will no longer be possible to use the Natural Web I/O Interface client with Microsoft Internet Information Services (IIS). IIS will no longer be supported.

13 Release Information for Natural Version 6.3.7

▪ New Features	96
▪ Changes and Enhancements	96
▪ Natural Remote Procedure Call (RPC)	98
▪ Natural Security	99
▪ Known Incompatibilities	102

New Features

Application Programming Interfaces

The utility SYSEXT provides the following new application programming interfaces (APIs):

API	Description
USR4215N	Return a list of resources (short object and long resource name) of a Natural library according to the forwarded range parameters.

Natural Web I/O Interface

The Natural Web I/O Interface is now delivered in the Version 1.3.5.

Support is provided for the following:

- It is now possible to define different styles for output fields which are based on variables and output fields which are based on literals (J2EE only). See *Defining Different Styles for Output Fields in Configuring the Client* which is part of the *Natural Web I/O Interface* documentation.

Changes and Enhancements

PROCESS PAGE Statement Enhanced

A field in a PROCESS PAGE statement can now be associated with a control variable. When the value returned for that field is not equal to the value sent, the field is regarded as modified and the control variable is set to modified.

Compiler Option PCHECK Enhanced

The compiler option PCHECK has been enhanced for use in conjunction with objects of type adapter. With PCHECK set to ON, the PROCESS PAGE USING statement now checks the parameter number, format, length and array index boundaries not only at execution time, but already at compile time.

In addition, the parameter check at compile time has been expanded to other invoking statements, such as PERFORM (external subroutine), INPUT USING MAP, and to the calling of help routines.

Result Format and Length of an Exponentiation with an Exponent of Format Numeric (N) or Packed (P)

The precision of the result field in an exponentiation with an exponent of format numeric (N) or packed (P) has changed in Natural Version 6.3.7 in order to achieve a result which is more precise.

In case of an exponentiation where the exponent is a numeric or packed operand with one or more digits after the decimal point (for example, N1.1 or P1.1), a result field of format float (F8) instead of format packed (P) is now generated.

The change is effective after a RUN, CATALOG or STOW in Natural Version 6.3.7. An EXECUTE of programs which are compiled under a previous version (Natural Version 6.3.6 or below) has no effect.

Example:

```
DEFINE DATA LOCAL
1 #F8 (F8)
END-DEFINE
COMPUTE ROUNDED #F8 = 1 + 5 ** 0.5
/* The result field of the exponentiation 5 ** 0.5 is now F8 instead of P15.0
PRINT #F8
/* After CATALOG, the result in Natural Version 6.3.7 is +3.236067977499789E+00
/* Result in Natural Version 6.3.6 is +3.000000000000000E+00
END
```

The documentation has been adapted accordingly. See *Performance Considerations for Mixed Format Expressions* in the *Programming Guide*.

New DDM for EMPLOYEES File

With Natural Version 6.3.7, the new DDM EMPLOYEES-V2009 is delivered. This DDM is used by the new sample programs supplied with the SYSEXV utility. This DDM makes use of new Adabas data types introduced with Adabas Version 6. The respective EMPLOYEES data requires Adabas Version 6.1.6 or higher.

Enhanced Application Programming Interfaces

The following application programming interfaces (API) in the library SYSEXT have been enhanced:

USR0330N

The complete timestamp value for Natural objects is now returned.

USR0360N, USR0421N **and** USR1020N

A new optional parameter named LAYOUT-CHECK is available. The layout definition is checked when a value other than blank is specified for this parameter. Error messages that do not fit into the ranges are not processed.

Natural Remote Procedure Call (RPC)

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

With Natural Version 6.3.7, an enhanced Natural Remote Procedure Call Version 6.3.3.1 is delivered that replaces the existing Natural RPC Version 6.3.3.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the following changes and enhancements are available:

- [System Command RPCERR Enhanced](#)
- [RPCINFO/RPCINFOL Enhanced](#)
- [SYSRPC Stub Generation](#)
- [Documentation](#)

System Command RPCERR Enhanced

The system command `RPCERR` has been enhanced to show the full error message text returned by an EntireX RPC server.

RPCINFO/RPCINFOL Enhanced

The following enhancements have been applied to the user application programming interface `RPCINFO` and the parameter data area `RPCINFOL`:

- `RPCINFO` has been enhanced to retrieve the error message text returned by an EntireX RPC server.
- The fields `S-NODE`, `S-NAME`, `C-NODE`, `C-NAME` have been enlarged from A8 to A32 to be more compliant with EntireX.
- `RPCINFO` is loaded to the library `SYSTEM` on system file `FNAT` and is immediately available to all user applications.

All enhancements are compatible with existing applications.



Notes:

1. If you recatalog an existing application that references `RPCINFOL`, you may be concerned if the enlarged fields are displayed. Either adapt your output layout accordingly or add the session parameter `AL=8` where appropriate.
2. You are strongly recommended to remove any copies of `RPCINFO` from the `FUSER` system file.

SYSRPC Stub Generation

The **Stub Generation** function of the SYSRPC utility has been enhanced by the direct command `COMPAT IDL` in order to optionally generate stub subprograms that are more compliant with EntireX RPC servers and the reliable RPC. If `COMPAT IDL` is specified and an EntireX RPC server is called by a Natural client, the parameter definitions on the **Stub Generation** screen must correspond to the IDL definition for the EntireX RPC server.

With the next version of Natural, `COMPAT IDL` will be the default. For compatibility reasons, an option will be provided to generate stub subprograms as in previous Natural versions.

Documentation

Please note that the term “stub subprogram”, which in earlier versions of EntireX was also used to refer to application-dependent, Workbench-generated pieces of code for issuing and receiving remote procedure calls, will no longer be used. In the EntireX documentation, these objects are now referred to as “interface objects”. In the Natural Remote Procedure Call documentation, however, the term “stub subprogram” is still prevailing. It will be replaced in one of the next versions of Natural RPC.

Natural Security

The following enhancements are provided with Natural Security Version 6.3.7:

- [Users](#)
- [Libraries](#)
- [Utilities](#)
- [Application Programming Interfaces](#)
- [Other Enhancements](#)

Users

The following enhancements are provided for users:

- [Copying a User](#)

- [New User Exit NSCUSEX2](#)

Copying a User

The function **Copy User** has been enhanced: You can enter the name of the new user not only in the new profile but already in the window which is displayed when you invoke the **Copy** function. This name is then written into the new profile (where you can still change it, of course).

New User Exit NSCUSEX2

A new user exit, NSCUSEX2, is provided in the library SYSSEC. It allows you to review the changes you have made to a list of group members. It is invoked when you use the user-maintenance function **Edit Group Members** and CATALOG the changes to the group. It displays a list of the group's members, indicating which members have been added to the group and which have been removed from it.

Libraries

The following enhancements are provided for libraries:

- [Copying a Library](#)
- [Use of UNIX Shell Commands](#)

Copying a Library

The function **Copy Library** has been enhanced: You can enter the name of the new library not only in the new profile but already in the window which is displayed when you invoke the **Copy** function. This name is then written into the new profile (where you can still change it, of course).

Use of UNIX Shell Commands

UNIX shell commands can be executed from within a Natural program by invoking the Natural user exit SHCMD via the statement CALL SHCMD. With Natural Security you can now control for each library the execution of UNIX shell commands from within Natural programs by allowing/disallowing the execution of the CALL SHCMD statement in the **Command Restrictions** section of library profiles.

Utilities

The following enhancements are provided for utilities:

- [SYSERR - Direct Commands](#)
- [SYSMAIN and SYSOBJH - Exclude Library Profiles With Co-Owners](#)

SYSERR - Direct Commands

In the security profiles for the `SYSERR` utility, you can now control the use of `SYSERR` direct commands. You can allow or disallow each command for all users, or allow it for Natural Security administrators only.

SYSMAIN and SYSOBJH - Exclude Library Profiles With Co-Owners

The session option **Utilities option** in the default profiles of utilities `SYSMAIN` and `SYSOBJH` has been enhanced: In addition to the existing values `Y` and `N`, you can specify the new value `0`: This is identical to `Y`, but has the following additional effect:

If you use `SYSMAIN` or `SYSOBJH` to process the contents of a library in whose library profile the general option **Utilities** is set to `0` (= processing permitted for owners), and you as an owner require a countersignature, the countersignature prompt will be suppressed; instead, the library will be excluded from `SYSMAIN/SYSOBJH` processing.

This enhancement is only relevant online, because in batch mode countersignatures cannot be processed anyhow.

Application Programming Interfaces

The following application programming interfaces (APIs) have been enhanced:

- [NSCADM](#)
- [NSCXR](#)

NSCADM

The API `NSCADM` has been enhanced. It allows you to compare a preset value (as set in the **Library and User Preset Values**) with the the corresponding actual value in user profiles, and provides a list of all user profiles in which the value differs from the preset value. For details, see example program `PGMADM04` and text member `TXTADM04` in the library `SYSSEC`.

NSCXR

The API `NSCXR` has been enhanced. It allows you to retrieve information about the access rights to an individual module in a library, for all users who have access to the library. For details, see example program `PGMXR018` and text member `TXTXR018` in the library `SYSSEC`.

Other Enhancements

The following other enhancements are provided:

- [Suppress Startup Transaction in NDV Mapped Environment](#)
- [Copying a Mailbox](#)

Suppress Startup Transaction in NDV Mapped Environment

If a startup transaction is specified in a Natural Security library profile, it will be executed after a successful logon to the library. In a Natural Development Server environment, however, when a library is selected from the tree view in the mapped environment, the execution of a startup transaction may not be desired. A new Administrator Services option **NDV Startup Inactive**, which can be set in the **Library And User Preset Values**, allows you to suppress the execution of startup transactions in a mapped environment.

Copying a Mailbox

The function **Copy Mailbox** has been enhanced: You can enter the name of the new mailbox not only in the new profile but already in the window which is displayed when you invoke the **Copy** function. This name is then written into the new profile (where you can still change it, of course).

Known Incompatibilities

This section provides additional information that you should be aware of after having installed Natural.

Canceling File Transfer with Entire Connection

The following information applies when using Entire Connection Version 4.5.2 or above.

A different error message is now issued when the user cancels a file transfer that has been started with the Natural statement `DOWNLOAD PC FILE` or `UPLOAD PC FILE`. As of Natural Version 6.3.7, the error message `NAT1016` is issued.

Up to Natural Version 6.3.6, one of the following error messages was issued: `NAT1500`, `NAT6220`, `NAT6221`.

This change may affect existing Natural applications which handle the error situation inside an ON ERROR statement.

14 Release Information for Natural Version 6.3.8

- New Features 106
- Changes and Enhancements 106
- Natural Remote Procedure Call (RPC) 108
- Notice of Future Changes 109

New Features

Application Programming Interfaces

The utility SYSEXT provides the following new application programming interfaces (APIs):

API	Description
USR4217N	Send data to the webMethods Optimize for Infrastructure Data Collector.
USR4218N	Provide information on the context in which the Natural Development Server runs, for example, the version of the client.

Natural Functions for URL and Base64 Encoding

Natural now provides functions for encoding or decoding URL addresses, and for converting binary data into printable, network-compatible data (or vice versa) using Base64 encoding. For further information, see *Natural Functions* in the *System Functions* documentation.

Natural Web I/O Interface Client

The Natural Web I/O Interface is now delivered in the Version 1.3.6.

Support is provided for the following:

- It is now possible to define underlined, blinking, italic and bold text using style sheets. XSLT files are no longer used for this purpose. See *Using Style Sheets (J2EE only)* in *Configuring the Client* which is part of the *Natural Web I/O Interface* documentation.

Changes and Enhancements

Assigning a Floating Point Operand (Format F) to a Numeric or Packed Operand (Format N or P)

When a floating point operand (format F) is assigned to a numeric or packed operand (format N or P), the result as compared to Natural Version 6.3.7 may be different. As of Natural Version 6.3.8, the result of the COMPUTE statement will not be rounded any more if the ROUNDED option is not used. This applies regardless of whether the operand has been explicitly defined by the user or implicitly created by Natural to hold intermediate results (see example 2 below). This change has been made in the runtime and is therefore effective when executing programs with Natural Version 6.3.8. It is not necessary to recatalog the programs.

Example 1:

```

DEFINE DATA
  LOCAL
    1 #SRC (F8)
    1 #TAR (N2.6)
END-DEFINE
#SRC := 14.9999999
COMPUTE #TAR := #SRC
PRINT #TAR
COMPUTE ROUNDED #TAR := #SRC
PRINT #TAR
END

```

Results in Natural Version 6.3.7:

```

15.000000
15.000000

```

Results in Natural Version 6.3.8:

```

14.999999
15.000000

```

Example 2:

```

DEFINE DATA
  LOCAL
    1 #SRC (N11.7)
    1 #TAR (N2.7)
END-DEFINE
#SRC := 224.9999999
COMPUTE #TAR = SQRT(#SRC) /* INTERMEDIATE F8 FIELD
PRINT #TAR
COMPUTE ROUNDED #TAR = SQRT(#SRC)
PRINT #TAR
END

```

Results in Natural Version 6.3.7:

```
15.0000000  
15.0000000
```

Results in Natural Version 6.3.8:

```
14.9999999  
15.0000000
```

Natural Remote Procedure Call (RPC)

Natural Remote Procedure Call (RPC) is available as a separate subcomponent of Natural. It has its own version number. This measure takes into account that Natural RPC is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

With Natural Version 6.3.8, an enhanced Natural Remote Procedure Call Version 6.3.3.3 is delivered that replaces the existing Natural RPC Versions 6.3.3.1 and 6.3.3.2.

As of Version 6.3 of Natural Remote Procedure Call (RPC), the following changes and enhancements are available.

Stub Subprograms from IDL File

If you have generated a stub subprogram from an IDL file using the **Generate Natural from Software AG IDL** command of NaturalONE or EntireX Workbench, program and library name of the IDL file are incorporated into the generated stub subprogram and passed to the RPC server. This means especially that the name of the program that is executed on the RPC server may be different to the name of the stub subprogram.

Note for Natural RPC servers:

If the Natural client calls a Natural RPC server, the library name of the IDL file is only evaluated by the Natural RPC server if the Natural client has set the **Logon** option.

Notice of Future Changes

The following changes are planned for future versions of Natural.

Adabas Records no Longer in Hold Status when Database Update is not Permitted

With a future version of Natural, Adabas records will no longer be put in hold status if the Natural profile parameter `DBUPD` is set to `OFF` (that is, when a database update is not permitted). When set to `OFF`, the Adabas calls `L1`, `L2`, `L3`, `S1`, `S2`, `S3` will be generated instead of `L4`, `L5`, `L6`, `S4`, `S5`, `S6`.

