software AG

Natural for Mainframes

Installation

Version 4.2.6 for Mainframes

October 2009

Natural

This document applies to Natural Version 4.2.6 for Mainframes and to all subsequent releases.

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

Table of Contents

1   Installation

This documentation describes step by step how to install Natural with Adabas or VSAM system files on IBM and compatible systems running under the operating system z/OS, z/VSE, VM/CMS or BS2000/OSD.

For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

Installation instructions for TP monitor interfaces and certain Natural subproducts are also included in this documentation.

For installation-related information on Unicode and code page support, refer to *Configuration and Administration of the Unicode/Code Page Environment* in the *Unicode and Code Page Support* document.

The Natural Installation documentation is organized under the following headings:

●	**General Installation Information**	Prerequisites, overview of the installation process, information on `INPL` format, installation options, Natural system files, using VSAM system files, using installation jobs generated by System Maintenance Aid (SMA), creating your own installation jobs.
●	**Licensing Natural**	Information on license key file, license key file installation, product license check and product license check FAQs.
●	**Installing Natural under z/OS**	How to install Natural under the operating system z/OS using Adabas system files.
●	**Installing Natural under z/VSE**	How to install Natural under the operating system z/VSE using Adabas system files.
●	**Installing Natural under VM/CMS**	How to install Natural under VM/CMS using Adabas system files.
●	**Installing Natural under BS2000/OSD**	How to install Natural under the operating system BS2000/OSD using Adabas system files.

	Installing the Natural Com-plete/SMARTS Interface	How to install the Natural Com-plete/SMARTS Interface (NCF). *
	Installing the Natural CICS Interface	How to install the Natural CICS Interface (NCI).
	Installing the Natural IMS TM Interface	How to install the Natural IMS TM Interface (NII).
	Installing the Natural TSO Interface	How to install the Natural TSO Interface (NTI).
	Installing the Natural UTM Interface	How to install the Natural UTM Interface (NUT).
	Installing the Natural TIAM Interface	How to install the Natural TIAM Interface (NRT).
	Installing the Entire System Server Interface	How to install the Entire System Server Interface which is required if the product Entire System Server or Natural ISPF is to be used.
	Installing Natural Security	How to install Natural Security (NSC) on a mainframe computer under all mainframe operating systems and TP monitors supported by Natural using an Adabas database.
	Installing the Software AG Editor	How to install the Software AG Editor which is an optional feature that represents basic functionality within Natural, exclusively used by several Natural subproducts and other Software AG products.
	Installing the Natural Net Data Interface	How to install the Natural Net Data Interface `NATNETTO` which supports the EntireX CICS 3270 Bridge and similar client/server solutions in message-oriented server environments, that is, TP monitors.
	Installation Steps for REQUEST DOCUMENT and PARSE XML	Describes the installation steps required to enable the use of the Natural statements `REQUEST DOCUMENT` and `PARSE XML`.
	Overall Installation Verification	How to verify the successful completion of the preceding installation procedures: installation verification for base Natural and installation verification for the TP-monitor interface installed.

* Corresponds to Natural Com-plete Interface.

Supplementary Documents

The Natural Installation documentation is supplemented by the following documents:

- *Natural Operations*
- *Natural TP Monitor Interfaces*
- *System Maintenance Aid* (separate documentation on the Natural Documentation CD-ROM and on ServLine24)

For details of the utility used for loading the Natural system objects, see also the `INPL` utility.

For information on the installation of the Natural add-on products, see the following documents:

- *Natural Advanced Facilities*
- *Natural Connection*
- *Natural for DB2 Database Interface*
- *Natural SQL Gateway*
- *Natural for SQL/DS*
- *Natural for DL/I Database Interface*
- *Natural for VSAM Database Interface*
- *Natural Optimizer Compiler*
- *Review Natural Monitor*
- *Entire Transaction Propagator*
- *Natural SAF Security*

For information on installing the Natural Development Server under the various operating systems, refer to the platform-specific installation instructions in the separate *Natural Development Server* documentation.

2 General Installation Information

This document contains general installation information concerning the operating systems z/OS, z/VSE, VM/CMS, BS2000/OSD, the TP monitor system interfaces and the Natural add-on units.

- Prerequisites
- Overview of the Installation Process
- INPL Format
- Installation Options
- Natural System Files
- Using VSAM System Files
- Using Installation Jobs Generated by System Maintenance Aid (SMA)
- Creating Your Own Installation Jobs

Prerequisites

Before you can install the new version of Natural, the following products must already be installed and the following requirements must be met at your site:

- A supported version of the respective operating system must be installed. For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- A supported version of Adabas must be installed. For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- As a rule of thumb, each major Software AG product requires approximately 20 MB space in the Adabas database to store the Natural objects supplied by Software AG.

To use VSAM system files, see *Using VSAM System Files*.

Overview of the Installation Process

The installation process comprises the following basic parts:

1. Creating an executable module for batch operation

2. Creating a system file containing the Natural objects (programs, maps, etc.)

   Part 1 and Part 2 are described in the sections:

   Installing Natural under z/OS

   Installing Natural under z/VSE

   Installing Natural under VM/CMS

Installing Natural under BS2000/OSD

3. Creating executable environments under different TP monitors for using Natural in online mode

   The steps required for Part 3 are described for each TP monitor in the corresponding section of this document.

   Installing the Natural/Com-plete/SMARTS Interface

   Installing the Natural CICS Interface

   Installing the Natural IMS TM Interface

   Installing the Natural TSO Interface

   Installing the Natural UTM Interface

   Installing the Natural TIAM Interface

4. Installing the optional Natural add-on products and other optional Natural features

   Natural Advanced Facilities *

   Natural Connection *

   Natural for DB2 *

   Natural for SQL/DS *

   Natural SQL Gateway *

   Natural for DL/I *

   Natural for VSAM *

   Natural Optimizer Compiler *

   Installing the Software AG Editor

   Installing the Entire Systems Server Interface

   Installing Natural Security

   Installing and Activating Natural SAF Security *

   Installing the Natural Net Data Interface

   Installation Steps for REQUEST DOCUMENT and PARSE XML

* For installation procedures not included in this document, refer to the relevant add-on product documentation.

INPL Format

The installation instructions frequently refer to the Natural utility `INPL`. `INPL` is used to load the Natural system objects (dataset `NATvrs.INPL`) into the Natural system files (see below). For details of this utility, refer to the section *INPL Utility* in the Natural *Utilities* documentation.

Installation Options

Natural offers various installation options which affect the setting up of your Natural system. For example, you can decide whether or not to use a Natural global buffer pool, a Natural shared nucleus, etc.

These options are described in the Natural *Operations* documentation or in the Natural *TP Monitor Interfaces* documentation. You are recommended to read this information before you begin with the installation of Natural.

Before you install Natural-based subproducts in a new Natural Version 4.2 environment, refer to the corresponding subproduct-specific installation description regarding `INPL` procedures.

Natural System Files

Concerning the Natural system files, the following installation-related information applies:

FNAT
   If you wish to upgrade from a previous Natural version, either a new `FNAT` system file can be used or the previous system file can be reused. For further information, see *Migration Hints* in the current Natural Release Notes for Mainframes.

FUSER
   The `FUSER` system file can be used exclusively by the new Natural version or it can be shared by different versions of Natural. See the step *Load the FUSER System File* in the appropriate operating-system-specific installation documentation.

FDIC
   The `FDIC` system file can be used exclusively by the new Natural version or it can be shared by different versions of Natural. See the step *Load the FDIC System File* in the appropriate operating-system-specific installation documentation.

FSEC

The `FSEC` system file can be used exclusively by the new Natural version or it can be shared by different versions of Natural. See the step *Load the FSEC System File* in the appropriate operating-system-specific installation documentation.

Scratch-Pad File

The Scratch-Pad File is a system file that can be used exclusively by the new Natural version or be shared by different versions of Natural. See step *Load the Scratch-Pad File* in the appropriate operating-system-specific installation documentation.

Using VSAM System Files

If you want to use the Virtual Storage Access Method VSAM for storing your system files, a supported version of Natural for VSAM is required.

For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

For further information on the Natural interface to VSAM, refer to the *Natural for VSAM* documentation.

It is possible to store the system files of different Software AG products on different data base management systems (for example, the Con-nect system file on Adabas and the Natural system file on VSAM). However, if Adabas is available, it is recommended that you use Adabas for *all* system files.

For information about which Software AG products can be used with VSAM system files, please contact Software AG support or your Software AG representative.

Using Installation Jobs Generated by System Maintenance Aid (SMA)

The installation of Software AG products is performed by installation `jobs`. These jobs are either created "manually" or generated by System Maintenance Aid (SMA) Version 2.1.2.

For each step of the installation procedures described below, the job number of a job performing the corresponding task is indicated. This job number refers to an installation job generated by SMA.

For information on the concepts and the usage of Software AG's System Maintenance Aid (SMA), refer to the *System Maintenance Aid* documentation and to SMA Readme Files.

SMA Readme Files:

For information on important parameters and options, refer to the SMA Function RM (Show Readme File) of the corresponding product.

> **Note:** System Maintenance Aid (SMA) is not available for the operating system VM/CMS.

Creating Your Own Installation Jobs

If you are not using System Maintenance Aid, you can use the example installation job of the same job number in the job library on the Natural installation tape. You must adapt this example job to your requirements.

Note that the installation job numbers on the tape are preceded by a product code (see list of job number prefixes below).

Job Number Prefixes

The installation job numbers on the tape are preceded by a product code (for example, `NATI060` or `NATI080`). The relevant product codes are mentioned at the beginning of each product-specific installation instruction. The table below contains only some examples.

Product Code	Product Name
NAF	Natural Advanced Facilities
NAT	Base Natural
NCI	Natural CICS Interface
NCF	Natural Com-plete/SMARTS Interface (corresponds to Natural Com-plete Interface)
NDB	Natural for DB2
NDL	Natural for DL/I
NDV	Natural Development Server
NII	Natural IMS TM Interface
NSB	Natural SQL Gateway
NSC	Natural Security
NSF	Natural SAF Security
NSQ	Natural for SQL/DS
NTC	Natural Connection
NTI	Natural TSO Interface
NVS	Natural for VSAM

3 Licensing Natural

License Key File

You must install a valid license key file on all mainframe platforms on which Natural is installed. Your individual product license is shipped on the installation tape. It can also be shipped by e-mail, if desired.

Once the license key file is received, you must use native FTP commands to transfer it to your mainframe host before using it during installation.

Warnings:

- **Using utilities instead of native FTP commands for the license key file transfer may corrupt the license key and thus prevent the later execution of Natural. This applies, for example, to file transfers based on 3270 terminal emulations that do not provide a true binary file transfer, but convert specific characters.**

- **The license key file has to remain in ASCII format — even on the mainframe. It must not be modified. Any modification of the license key file will invalidate the digital signature and the license check will fail. In the event of a check failure, contact your Software AG technical support representative.**

License Key File Installation

During the Natural installation process, the license key file will be converted into an object module which is linked to the Natural nucleus. The installation slightly varies depending on the operating system. See the corresponding step in the platform-specific installation procedures.

Product License Check

Each time a Natural session starts, the license information in the license object module is verified and the validity of the license key is checked. In the event of a license check failure, contact your Software AG technical support representative. For certain failures, for example, CPU ID not defined or CPU capacity exceeded, a message will be issued on the system console and the session will start without further notice.

Product License Check FAQs

1. **Why has Software AG introduced a license check for Natural on the Mainframe?**
   Software AG wants to ensure that customers run Natural only on mainframe machines for
   which they have valid and sufficient product license. This ensures that we have better control
   of the use of our software products. On other platforms (UNIX, Windows), the product license
   check has been established for many years.

2. **Which information is contained in a product license?**
   A product license is a sequential file with US-ASCII text in XML format containing the following
   items:

   - Software AG header

   - customer information (Name, ID)

   - encrypted license key

   - license expiration date (or unlimited)

   - product information (product code, version, name)

   - environment information (operating system type, CPU ID, system name, capacity)

3. **What exactly is meant by the CPU ID?**
   IBM defines the CPU ID as the "central processing complex node descriptor sequence number".
   It is the unique hexadecimal machine serial number without the machine model number.

   z/OS:

   On z/OS systems, information about the CPU can be displayed on the operator console by en-
   tering the command:

   ```
   D M=CPU
   ```

   For example, suppose the console output from this command contains these lines:

   ```
   EE174I 16.38.50 DISPLAY M 951
   PROCESSOR STATUS
   ID  CPU                   SERIAL
   00  +                      0FA10E2096
   01  +                      0FA10E2096
   CPC ND = 002096.S07.IBM.83.000000007A10E
   CPC SI = 2096.V03.IBM.83.000000000007A10E
   ....
   ```

   In this example, the CPU ID is `7A10E`, as highlighted in the example above.

z/VSE:

On z/VSE systems, information about the CPU can be displayed on the operator console by entering the operator command:

```
SIR
```

For example, suppose the console output from this command contains these lines:

```
...
AR 0015 PROCESSOR = IBM 2096-V03 83 (7A10E83)  LPAR = DAEX     No. = 0007
....
```

The CPU ID is highlighted in the example.

BS2000/OSD:

On BS2000/OSD systems, the CPU ID (8 byte hexadecimal) consists of the machine serial number, the processor ID and the machine model number. The processor ID in the second byte of the CPU ID is ignored for the license check. Information about the CPU can be displayed by entering the following BS2000/OSD command:

```
/SHOW-SYSTEM-INFORMATION INFORMATION=*CPU-ID-LIST
```

For example, suppose the output from this command contains these lines:

```
...
%CONFIGURATION                 = 7.500- S140-20A      «
%CPU-ID-LIST :  ADR   0        = 1D02301375000000 «
...
```

The most relevant information is shown in these lines. The first value identifies the machine type as `7.500- S140-20A`, the second value identifies the physical CPU ID as `1D02301375` (trailing zeros can be omitted).

z/VM

On z/VM systems, the CPU ID can be determined entering the command:

```
q cpu
```

Suppose the output from this command is:

```
CPUID = FF07A10E20968000
```

The first byte will always be `FF`, followed by the CPU-ID (highlighted in the example).

Under Natural:

The following command enables you to display machine information under all operating systems:

```
NATQVS
```

4. **Are there product licenses for Natural add-on products, e.g. for Natural for DB2?**
   No. The license key file is for the base Natural product only. At the moment there are no plans to introduce add-on product licenses.

5. **How is the product license key file installed?**
   The product license key file is supplied on the individual customer installation tape or separately via an e-mail attachment. During the installation process, it is copied to disk. It is then converted into an assembler input file which is then assembled into a linkable module. This module is linked to the Natural nucleus.

6. **How can I read the product license key file?**

   ■ The product license key file can be read on a PC by means of an XML editor (file type `.xml`) or any PC text editor.

   ■ During the conversion process, a listing of the license key file is printed by the conversion program.

   ■ When Natural is installed and runs, the license information can be displayed by using function "L" of the Natural utility `SYSTP`.

7. **When is the product license checked?**
   The product license is checked during every Natural session initialization. There are no license checks during a Natural session after session initialization.

8. **Which items of a product license are checked?**

   ■ The encrypted license key

   ■ The expiration date (if any)

   ■ The operating system (z/OS, z/VSE, z/VM or BS2000/OSD)

   ■ The product code (NAT, etc.)

   ■ The product version (either $v.r$ or $v.r.s$, for further information on product versions, see Version in the *Glossary*)

   ■ The machine CPU ID

   ■ The machine capacity (only for z/OS and z/VM)

9. **What happens if the product license is incorrect, insufficient or not installed?**
   There are two possible reactions: Either an initialization error message is issued and the Natural session continues (a), or a warning is issued on the system console (b). This behaviour may change in a future version of Natural, that is, the Natural session will terminate after such an error or warning is issued.

   a. The session starts but an error message is issued on the user terminal in one of the following cases:

      ■ the product license module is missing or incorrect, or it has been modified

      ■ incorrect operating system, product code or product version

      ■ the license expiration date has been reached

   b. The session starts but a warning message is issued on the system console in one of the following cases:

      ■ Your machine CPU ID is not defined in the license or the machine capacity is higher than the value specified in the product license.

      ■ The product license will expire within the next 30 days.

      The warning message is not repeated on the system console for every Natural session. It is issued only once per Natural buffer pool per day.

10. **How can I get a new product license key file if the delivered license key file is insufficient for my environment?**
    Contact your Software AG sales representative to get a correct product license key file according to your contract. It can be shipped by e-mail either in ASCII format or in the converted assembler input format.

11. **If I use one Natural nucleus on different machines, how can I handle the license key files?**
    It is possible to have multiple CPU IDs defined in one license key file. If all your CPU IDs are defined in your license key file, you only need one license module that you link to all your Natural nuclei. For detailed information, contact your Software AG sales representative.

12. **Is it possible to have the license module separate from the Natural nucleus?**
    Yes. The license module can be defined to be loaded dynamically during session start by means of the profile parameters RCA and RCALIAS. Specify in NATPARM or dynamically:

```
RCA=NATLIC,RCALIAS=(NATLIC,name),
```

    where *name* is the name of your separately linked license module in the Natural load library.

    Using this technique causes any statically linked license module to be replaced. This can be helpful for testing.

4 Installing Natural under z/OS

This document describes step by step how to install Natural under the operating system z/OS using Adabas system files.

- Prerequisites
- Installation Tape for Natural under z/OS
- Transferring a License Key File from PC to Host with FTP
- Installation Procedure for Natural under z/OS
- Installation Verification for Natural under z/OS

See also *z/OS Environment* in the Natural *Operations* documentation.

For installation-related information on Unicode and code page support, refer to *Configuration and Administration of the Unicode/Code Page Environment* in the *Unicode and Code Page Support* documentation.

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

- A supported version of the z/OS operating system must be installed. For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- A supported version of Adabas must be installed. For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- As a rule of thumb, each major Software AG product requires approximately 20 MB space in the Adabas database to store the Natural objects supplied by Software AG.

Installation Tape for Natural under z/OS

The installation tape contains the datasets listed in the table below. The sequence of the datasets and the number of library blocks needed are shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NAT*vrs*.LICS	Natural License Key File. For further information on license key file, license key file installation, product license check and product license check FAQs, see *Licensing Natural*. If a license key file is supplied as an e-mail attachment, see *Transferring the License Key File from PC to Host with FTP*.
NAT*vrs*.SYSF	Empty Natural system file
NAT*vrs*.ERRN	Natural error messages
NAT*vrs*.LOAD	Natural load modules
NAT*vrs*.OBJS	Natural object modules
NAT*vrs*.SRCE	Natural source modules and macros
NAT*vrs*.JOBS	Example installation jobs
NAT*vrs*.LDEL	Instructions to delete Natural system objects of Version 4.1
NAT*vrs*.INPL	Natural system objects
NAT*vrs*.EXPL	Natural example objects

Copying the Tape Contents to a z/OS Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset COPY.JOB from tape to disk.
- Modify this dataset to conform to your local naming conventions.

The JCL in this dataset is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the dataset COPY.JOB contains the JCL to unload the datasets for all delivered products from the tape to your disk.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPY.JOB from Tape to Disk
- Step 2 - Modify COPY.JOB on Your Disk

■ Step 3 - Submit COPY.JOB

Step 1 - Copy Dataset COPY.JOB from Tape to Disk

The dataset `COPY.JOB` (Label 2) contains the JCL to unload all other existing datasets from tape to disk. To unload `COPY.JOB`, use the following sample JCL:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* ------------------------------
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volume),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=volume,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

hilev is a valid high level qualifier
tape-volume is the tape volume name, for example: T12345
volume is the disk volume name

Step 2 - Modify COPY.JOB on Your Disk

Modify the `COPY.JOB` on your disk to conform to your local naming conventions and set the disk space parameters before submitting this job:

■ Set `HILEV` to a valid high level qualifier.

■ Set `LOCATION` to a storage location.

■ Set `EXPDT` to a valid expiration date.

Step 3 - Submit COPY.JOB

Submit `COPY.JOB` to unload all other datasets from the tape to your disk.

Transferring a License Key File from PC to Host with FTP

If a license key file is supplied as an e-mail attachment, you must transfer the attached license key file `nat`*vr*`.xml` from the PC to the mainframe, using native FTP commands.

⬤ **Caution:** Using utilities instead of native FTP commands for the license key file transfer may corrupt the license key and thus prevent Natural from execution later on. This applies for example to file transfer based on 3270 terminal emulations that do not provide a true binary file transfer, but convert specific characters.

To transfer a license key file from the PC to the mainframe, perform the following steps:

1. Save the product license key file e-mail attachment on your PC hard disk.

2. Open a command prompt window. In the command prompt window, change to the directory where you saved the product license key file.

3. Start an FTP session for communication with the z/OS host:

   ```
   ftp host-name
   ```

   Where *host-name* is the name of the z/OS host.

4. Enter your z/OS host login ID and password.

5. Once the FTP session has been established, specify the z/OS file system and dataset prefix for the license key file:

   ```
   ftp>cd 'hilev'
   ```

   Switch to binary data mode (the license key file must retain its ASCII format during the transfer):

   ```
   ftp>binary
   ```

6. Specify that the dataset for the license key file must be written with `RECFM=FB` and `LRECL=80`.

```
ftp>quote site RECFM=FB LRECL=80 BLKSIZE=4000
```

> **Note:** `RECFM=F` is also supported for product license key files.

7. Copy the license key file as a dataset on the z/OS system.

```
ftp>put natvr.xml NATvrs.LICS
```

This command will create a dataset called `hilev.NATvrs.LICS`.

If you requested a prefix of `NATURAL`, the dataset name would be: `NATURAL.NATvrs.LICS`.

The dataset will have `RECFM=FB`, `LRECL=80`, and the license key information stored in the dataset will be in ASCII format.

8. Stop your FTP session by entering:

```
ftp>quit
```

Installation Procedure for Natural under z/OS

▶ **Step 1: Convert and Assemble the License Key File**

Using `PROGRAM NATLICAM`, convert dataset `hilev.NATvrs.LICS` into an Assembler input file.

Assemble/link into a linkable module. This module is linked (Job I060) to the Natural nucleus.

■ (Job I007, Steps 0101, 0102 and 0104)

Job I007, Step 0101	Allocate temporary dataset for license key module (`NATLIC`)
Job I007, Step 0102	Convert license key file (`NATvrs.LICS`).
Job I007, Step 0104	Assemble and link license key module (`NATLIC`).

▶ **Step 2: Copy Natural Modules to an APF Library**

■ (Job I009, Step 1200, 1210, 1220, 1250)

Link the following Natural modules to an Authorized Program Facility (APF) library:

NATGBP42	Required if you want to install a global buffer pool.
NATASM42	Required if you want to install an Authorized Services Manager (ASM). You must use the ASM in the following cases: ■ The Natural profile parameter BPPROP is set to PLEX or GLOBAL or GPLEX (buffer pool propagation is used). ■ Natural global buffer pools are allocated in system key; see *Installing the Natural GBP Operating Program* in the *Operations* documentation. ■ Natural under CICS is used in a Parallel Sysplex (SIP function is required). ■ Natural under IMS TM is used in terminal-oriented, non-conversational mode (SIP function is required). ■ Natural under IMS TM is used, with the Accounting function writing SMF records. See also *Authorized Services Manager under z/OS* in the *Operations* documentation.
NATRSM42	Required if you want to install a Natural Roll Server. You must use the RSM in the following cases: ■ The RPC Server Front-End is used. ■ Natural under IMS TM runs in a Parallel Sysplex environment. ■ Natural under CICS runs in a Parallel Sysplex or CICSplex environment. See also *Natural Roll Server Operation* in the *Operations* documentation.
RPC Server Front-End	Required if you want to use impersonation with the Natural Remote Procedure Call (RPC). **If you use SMA:** Define the name of the RPC server front-end in the parameter NAT-RPC-FRONT. **If you do not use SMA:** The name of the RPC server front-end is freely selectable and is specified in the link-edit step of Job I009, Step 1250. It is recommended to link the RPC server front-end into an APF-authorized LINKLIST library. This eliminates the need to provide the whole STEPLIB concatenation in the execution JCL of the Natural server APF-authorized. In this case, you have to issue the MVS MODIFY command F LLA,REFRESH after the link edit.

▶ **Step 3: Create and Start the Global Buffer Pool**

■ (Job I015, Steps 0100, 0101, 0102, 0104)

Step	Creates job	Function
0100	GBNASTRT	Start Natural global buffer pool.
0101	GBNASTOP	Stop Natural global buffer pool.
0102	GBEDSTRT	Start editor global buffer pool.
0104	GBEDSTOP	Stop editor global buffer pool.

These steps are only required if you wish to use a global buffer pool.

Create and start the jobs GBNASTRT and GBEDSTRT before using Natural.

For further information on the global buffer pool, see *Natural Global Buffer Pool* in the Natural *Operations* documentation.

Installation of the Authorized Services Manager is mandatory if the z/OS parameter ALLOWUSERKEYCSA(NO) applies by default or has explicitly been specified in SYS1.PARMLIB(DIAG*xx*),

▶ **Step 4: Load the FNAT System File**

■ (Job I050, Step 0100)

If you are installing into an existing Natural 4.1 FNAT file, skip this step.

Load the empty Natural system file (dataset NAT*vrs*.SYSF) using the ADALOD utility.

This file will contain all Natural objects supplied by Software AG. Its size depends on the number of products to be installed later. As a rule of thumb, 20 MB can be assumed for each major Software AG product.

The following ADALOD parameters must not be altered:

```
ISNREUSE=YES
```

To avoid Natural errors NAT9988 and NAT7397 after reorganization of the FNAT system file using ADAULD/ADALOD, the parameter USERISN should be set to YES.

The file number *fnat* of the FNAT system file can be chosen as described under Natural profile parameter FNAT in the Natural *Parameter Reference* documentation.

▶ **Step 5: Load the FUSER System File**

■ (Job I050, Step 0101)

You have the following options:

1. You can use an existing Version 4.1 FUSER file, then you can skip this step.

2. You can use a new FUSER file for Version 4.2.

3. You can use an existing Version 4.1 FUSER file to be shared by Versions 4.1 and 4.2.

4. You can use an existing Version 4.1 FUSER file to be used by Version 4.2 only.

Use a New Version 4.2 FUSER System File - First-Time Installation

For the use of a new and empty FUSER system file for Natural Version 4.2, no additional system-file-related actions are necessary.

If you do *not* want to share the FUSER system file, proceed as follows:

Load the empty Natural user file contained in dataset NAT*vrs*.SYSF using the ADALOD utility.

In this file, all user-written Natural programs are stored.

The following ADALOD parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number *fuser* of the FUSER system file can be chosen as described under Natural profile parameter FUSER in the Natural *Parameter Reference* documentation.

Reuse an Existing Version 4.1 FUSER System File - Migration from Natural Version 4.1

If you want to use the existing Natural Version 4.1 FUSER system file and you do not want to share the FUSER system file, skip this step.

Using a Version 4.1 FUSER File to be Shared by Natural Versions 4.1 and 4.2

If you use an existing Natural Version 4.1 FUSER system file to be shared by Natural Versions 4.1 and 4.2, you must upgrade your Natural Version 4.1 installation to Version 4.1.4.

Natural Version 4.1.4 Service Pack I003 or a subsequent Service Pack is required. Service Pack I003 and all subsequent Service Packs contain all the necessary Version 4.1 based solutions for Natural Version 4.2.

User Application Programming Interfaces USR* in Library SYSEXT

The USR* programs from the delivered library SYSEXT run in a special mode. As a result, the USR* programs do not need to set further steplibs to execute related objects for processing. This reduces the impact on the Natural buffer pool search logic and improves

the performance significantly if user exits are used extensively within user written applications.

It is necessary that the user exits are cataloged with Natural Version 4.2. This implies that the user exits cannot be executed with Natural Version 4.1.

Use of USR* Programs

Usually, the access of USR* programs by an application requires that the user application programming interfaces be copied from library SYSEXT to either the application libraries on the FUSER system file or to library SYSTEM on the FUSER system file or to library SYSTEM on the FNAT system file, respectively, or any other library which is defined as steplib for the application. Library SYSEXT can also be used as steplib. Due to the fact that the delivered user application programming interfaces will always be cataloged with the latest Natural version, we recommend that the user application programming interfaces should reside on the FNAT system file. This will ensure that the right version is executed and will separate user written applications from Software AG modules.

If applications which call user application programming interfaces should run with both Natural Version 4.1 and Natural Version 4.2, it must be made sure that the user application programming interfaces delivered with the corresponding Natural version are used.

The following scenarios may be considered:

Using a Version 4.1 FUSER File for Natural Version 4.1 and 4.2

If the same FUSER system file shall be used in a Natural Version 4.1 and Version 4.2 environment in parallel the following steps are recommended:

- Remove all USR* modules you have copied from library SYSEXT into application libraries on your FUSER system file.

- In both environments, copy the used USR* modules from library SYSEXT to library SYSTEM on the corresponding FNAT system file.

- Alternatively, the USR* modules can be moved to another system library on FNAT which then must be defined as steplib, or library SYSEXT can be used as steplib for the applications. Then automatically in both environments the right versions of the user application programming interfaces are executed.

Using a Version 4.1 FUSER File for Natural Version 4.2 only

If you want to use the existing Natural Version 4.1 FUSER system file and you do not want to share the FUSER system file, then it is still possible to replace all USR* modules you have copied from library SYSEXT into application libraries with the new USR* objects from the Version 4.2 library SYSEXT.

But the preferred way is to remove all user application programming interfaces on the
FUSER system file and copy the used user application programming interfaces from library
SYSEXT to library system of the FNAT system file or use a SYS library on FNAT as steplib.

Using a New FUSER File for Natural Version 4.2

If you want to port existing applications to a new FUSER system file, copy all application
objects but no Software AG USR* objects to the new FUSER system file. Then proceed as
described in the scenario above.

Migration

The FIND function of the Natural utility SYSMAIN can be used to search for all USR* modules
stored in a specific library on the FUSER system file or across the whole system file. In
addition, Predict cross reference data can be used to determine all referenced user applic-
ation programming interfaces.

The file number *fuser* of the FUSER system file can be chosen as described under Natural
profile parameter FUSER in the Natural *Parameter Reference* documentation.

▶ **Step 6: Load the Scratch-Pad File**

■   (Job I050, Step 0102)

The scratch-pad file (which is a Natural-internal system file) can be used exclusively by
the new Natural version or it can be shared by different versions of Natural.

If you do *not* want to use a scratch-pad file, skip this step.

If you do want to use a scratch-pad file; that is, if you want to use read-only system files
(profile parameter ROSY=ON), see also *Natural Scratch-Pad File* in the Natural *Operations*
documentation, proceed as follows:

Load the empty scratch-pad file contained in dataset NAT*vrs*.SYSF, using the ADALOD
utility as described below.

The following ADALOD parameter *must not* be altered:

```
ISNREUSE=YES
```

For the optional scratch-pad file inclusion, the following NATPARM parameters must be added or, if already present, updated with:

```
LFILE=(212,dbid,fnr)
ROSY=ON
```

If you use SMA: To be able to use a read-only system file, you have to set the parameter NAT-SCRF to Y and ROSY to ON.

▶ **Step 7: Load the FDIC System File**

■ (Job I050, Step 0103)

Skip this step:

▪ if you want to install Predict (in this case, use the corresponding installation step in the Predict *Installation* documentation), or

▪ if you want to use an existing FDIC system file (an existing FDIC system file can be shared by Natural Versions 4.1 and 4.2), or

▪ if you do not use your own FDIC system file.

Load the empty FDIC system file contained in dataset NAT*vrs*.SYSF using the ADALOD utility, as described below.

The following ADALOD parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number *fdic* of the FDIC system file can be chosen as described under Natural profile parameter FDIC in the Natural *Parameter Reference* documentation.

▶ **Step 8: Load the FSEC System File**

■ Skip this step,

▪ if you do not use Natural Security, or

▪ if you want to use an existing FSEC system file, or

▪ if you do not want to use an own FSEC system file.

If you use Natural Security: Refer to *Installing Natural Security*.

▶ **Step 9: Assemble the Natural z/OS Interface Module**

■ (Job I055, Step 0100 or 0103)

The source program `NATOS` contains a call to the macro `NTOS` which generates the Natural z/OS interface.

- Job I055, Step 0100, do the `IEBUPDATE`.
- Job I055, Step 0103, assemble and link `NATOS`.

Set the parameters in the source of the module `NATOS` to meet your requirements.

For a description of the Natural for z/OS generation parameters contained in the macro `NTOS`, refer to *NTOS Macro - Generation Parameters for Natural under z/OS* in the Natural *Operations* documentation.

Assemble and link the Natural z/OS interface module `NATOS` contained in the dataset *hilev*.NAT*vrs*.SRCE.

▶ **Step 10: Create the Natural Configuration Module**

- (Job I055, Step 0110)

  This step is only required if you need to change the delivered `NATCONFG` module, for example, to adapt the `NTDVCE` macro definition to your requirements.

  Change and assemble the source contained in the dataset *hilev*.NAT*vrs*.SRCE.

  Link the resulting Natural configuration module (`NATCONFG`) to the Natural nucleus when you link the Natural nucleus (see below).

  For more information on the configuration tables in `NATCONFG`, refer to *Natural Configuration Tables* in the Natural *Operations* documentation.

▶ **Step 11: Create the Parameter Module**

- (Job I060, Steps 0010, 0015)

  Create the batch Natural parameter module.

  The following parameters in the parameter module must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

  For *dbid*, *fnat* and *fuser*, use the values you specified when loading the system files (see **above**).

Global Natural Buffer Pool: If you wish to use a *global* Natural buffer pool, specify the parameter macro `NTBPI` with the name of the global Natural buffer pool and set the profile parameter `SUBSID` in all your Natural parameter modules.

If you use SMA: Define the name of the global Natural buffer pool in SMA parameter `NAT-GLOBAL-BP`.

For all other parameters: You can generally use the default values. Modify only the values of those parameters whose default values do not meet your requirements.

For a description of the individual parameters contained in the parameter module, refer to the *Parameter Reference* overview in the Natural *Parameter Reference* documentation.

Assemble and link the parameter module.

▶ **Step 12: Link the Natural Nucleus**

When linking the Natural nucleus, you have the following alternatives:

Link a batch front-end (Job I060, Step 0020) and link a shared nucleus (Job I060, Step 0105), or link a non-shared nucleus (Job I060, Step 0020).

■ **Using Alternative 1**

1. Link a batch front-end (Job I060, Step 0020)

   Do not link the batch front-end with the linkage editor option RENT.

   With the `INCLUDE` instruction for the parameter module, specify the name of the Natural parameter module you have created before.

   The following modules must be included (include module `NATOS` first):

Module	Function
NATOS	Batch Natural driver
NATWKFO	Work file support
PROO1B	Generated parameter module
ADAUSER	Adabas link module

   To access the shared nucleus, ensure that the `NATPARM` parameter `NUCNAME` contains the name of the module linked in Job I060, Step 0105.

2. Link a shared nucleus (Job I060, Step 0105)

   For a list of the modules included, see *Natural Shared Nucleus under z/OS and z/VSE* in the Natural *Operations* documentation.

If IBM error IEW2641S is returned when linking the shared nucleus, link the shared nucleus into a PDSE instead of a PDS.

Or:

Using Alternative 2

Link a non-shared nucleus (Job I060, Step 0020)

Do not link the non-shared nucleus with the linkage editor option RENT.

If you use SMA:
Ensure that SMA parameter `SHARED-NUC` is set to `N` (no). All the modules are then automatically linked in Step 0020.

If you do not use SMA:
Merge all `INCLUDE` statements and corresponding DD cards from Job I060, Step 0105 (shared nucleus) into Job I060, Step 0020 (front-end).
Link the executable Natural nucleus.

If IBM error IEW2641S is returned when linking the non-shared nucleus, link the non-shared nucleus into a PDSE instead of a PDS.

▶ **Step 13: Delete Natural System Objects**

■   This step is optional but recommended to avoid data inconsistencies.

If you are using a Version 4.1 Natural `FNAT` system file, you can delete obsolete Version 4.1 Natural objects.

If you are using SMA: Set the SMA parameter `NAT-FNAT41` to `Y` (yes).

If only base Natural Version 4.1 is installed

Delete the Version 4.1 Natural objects by loading the `NATvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0010)

If Natural for DB2 Version 4.1 is installed

Delete the Version 4.1 Natural for DB2 objects by loading the `NDBvrs.LDEL` data set with the `INPL` utility.

(Job I061, Step 0016)

See also *Delete Natural for DB2 Objects* in the *Natural for DB2* documentation.

If Natural Review Version 4.1 is installed

Delete the Version 4.1 Natural Review objects by loading the `RNMvrs.LDEL` data set with the `INPL` utility.

(Job I061, Step 0026)

See also *Delete Natural Review Objects* in the *Natural Review* documentation.

If Natural Security Version 4.1 is installed

Delete the Version 4.1 Natural Security objects by loading the `NSCvrs.LDEL` data set with the `INPL` utility.

(Job I061, Step 0099)

See also *Delete Natural Security Objects* in *Installing Natural Security*.

▶ **Step 14: Load the System Programs**

■ (Job I061, Step 0100)

Use the Natural `INPL` utility to load the Natural system objects (dataset `NATvrs.INPL`) into the Natural system files.

▶ **Step 15: Load the Error Messages**

■ (Job I061, Step 0102)

Load the English Natural error messages file (dataset `NATvrs.ERRN`) using the program `ERRLODUS`. In addition to the English error message short and long texts, error message short texts in German language (`ULANG=2`) are loaded. Use the program `ERRUPPER` to convert the error message texts to upper case. Both programs are described in the Natural *SYSERR Utility* documentation.

Load the Japanese Error Message Long Texts (dataset `NCJvrs.INPL`)

(Job I061, Step 8300)

This step is only required if you wish to replace the English error message long texts with their Japanese equivalents.

The Natural error message long texts in Japanese language are contained in the Natural Japanese Language Pack. This is a separate product (product code NCJ) that can be loaded optionally. If you do not load the Natural error message long texts in Japanese language, English error message long texts will appear.

Use the Natural utility `INPL` to load the Japanese error message long texts (dataset `NCJvrs.INPL`) into the Natural system files.

Load the Japanese Error Message Short Texts (dataset NCJ*vrs*.ERRN**)**

(Job I061, Step 8302)

This step is only required if you wish to load the error message short texts in Japanese language (ULANG=59).

The Natural error message short texts in Japanese language are contained in the Natural Japanese Language Pack. This is a separate product (product code NCJ) that can be loaded optionally. If you do not load the Natural error message short texts in Japanese language, English error message short texts will appear.

Use the program ERRLODUS described in the Natural *SYSERR Utility* documentation to load the error message short texts (dataset NCJ*vrs*.ERRN) into the Natural system file.

▶ **Step 16: Load the Examples**

■ (Job I061, Step 0103)

Use the Natural utility INPL to load the Natural example objects (dataset NAT*vrs*.EXPL) into the Natural system file.

▶ **Step 17: Create and Format the Roll File**

■ (Job I200, Step 0101)

This step is only required if you wish to use the Natural Roll Server.

> **Note:** If you plan to use an existing Roll File of a previous version, it is sufficient to execute the NATRSRFI RESET function. See *Natural Roll Server Operation* in the Natural *Operations* documentation.

Create and start the job FORMRF1 before using Natural.

For further information on different types of roll files, see *Natural under CICS* and *Natural under IMS TM* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 18: Create and Start the Roll Server**

■ (Job I200, Step 0102)

This step is only required if you wish to use the Natural Roll Server.

Create and start the job ROLLSV1 before using Natural.

For further information on the Natural Roll Server, see *Natural Roll Server Operation* in the Natural *Operations* documentation.

▶ **Step 19: Create and Start the Authorized Services Manager**

- (Job I200, Step 0103)

   This step is only required if you wish to use the Natural Authorized Services Manager.

   Create and start the job AUTHSV1 before using Natural.

   For further information, see *Authorized Services Manager* in the Natural *Operations* documentation.

▶ **Step 20: Create Sample JCL for Natural RPC Server**

- (Job I200, Steps 0108, 0109, 0115, 0120)

   Sample Natural RPC server.

   Step 0108, create the sample JCL to execute a Natural RPC server including the network task.

   Step 0109, create the sample JCL to execute a Natural RPC server with TCP/IP task.

   Step 0115, create the sample JCL to execute a standard Natural RPC server task without RPC server front-end.

   Step 0120, create the sample JCL to execute a Natural RPC server task by the RPC server front-end. This JCL is necessary for impersonation.

   If you want to use Natural RPC: Set the parameters in the SMA group RPC accordingly; in particular, set the parameter NAT-RPC to Y (Yes).

   For further information, see *Starting a Natural RPC Server* and *Starting a Batch Server in a Mainframe Environment* in the *Natural Remote Procedure Call (RPC)* documentation.

Installation Verification for Natural under z/OS

For base Natural, there are no specific installation verification procedures.

After the last step of the installation procedure has been successfully performed, check that the following results are available:

- Communication between Adabas and Natural is working.

- The Natural system files have been loaded.

- Batch Natural is operational.

5 Installing Natural under z/VSE

This document describes step by step how to install Natural under the operating system z/VSE using Adabas system files.

- Prerequisites
- Installation Tape for Natural under z/VSE
- Transferring a License Key File from PC to Host with FTP
- Installation Procedure for Natural under z/VSE
- Installation Verification for Natural under z/VSE

See also *z/VSE Environment* in the Natural *Operations* documentation.

For installation-related information on Unicode and code page support, refer to *Configuration and Administration of the Unicode/Code Page Environment* in the *Unicode and Code Page Support* documentation.

Notation vrs or vr: If used in the following document, the notation vrs or vr stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

- A supported version of the z/VSE operating system must be installed. For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- A supported version of Adabas must be installed. For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- As a rule of thumb, each major Software AG product requires approximately 20 MB space in the Adabas database to store the Natural objects supplied by Software AG.

Installation Tape for Natural under z/VSE

The installation tape contains the datasets listed in the table below. The sequence of the datasets and the number of library blocks needed are shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NAT*vrs*.LICS	Natural Licence Key File. For further information on license key file, license key file installation, product license check and product license check FAQs, see *Licensing Natural*. If a license key file is supplied as an e-mail attachment, see *Transferring the License Key File from PC to Host with FTP*
NAT*vrs*.SYSF	Example Natural system file
NAT*vrs*.LIBR	LIBR backup file
NAT*vrs*.LIBJ	Installation jobs
NAT*vrs*.LDEL	Instructions to delete Natural system objects of Version 4.1
NAT*vrs*.INPL	Natural system objects
NAT*vrs*.EXPL	Natural example objects
NAT*vrs*.ERRN	Natural error messages

The dataset type and the space each dataset requires on disk is shown in the *Report of Tape Creation*.

Copying the Tape Contents to a z/VSE Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

■ Copy dataset COPYTAPE.JOB from tape to disk.

■ Modify this dataset to confom with your local naming conventions.

The JCL in this member is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the member COPYTAPE.JOB contains the JCL to unload the datasets for all delivered products from the tape to your disk, except the datasets that you can directly install from tape, for example, Natural INPL objects.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk
- Step 2 - Modify COPYTAPE.JOB

■ Step 3 - Submit COPYTAPE.JOB

Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk

The dataset `COPYTAPE.JOB` contains the JCL to unload all other existing datasets from tape to disk.
To unload `COPYTAPE.JOB`, use the following sample JCL:

```
* $$ JOB JNM=LIBRCAT,CLASS=0,                                    +
* $$ DISP=D,LDEST=(*,UID),SYSID=1
* $$ LST CLASS=A,DISP=D
// JOB LIBRCAT
* ****************************************
*     CATALOG COPYTAPE.JOB TO LIBRARY
* ****************************************
// ASSGN SYS004,nnn                              <------  tape address
// MTC REW,SYS004
// MTC FSF,SYS004,4
ASSGN SYSIPT,SYS004
// TLBL IJSYSIN,'COPYTAPE.JOB'
// EXEC LIBR,PARM='MSHP; ACC S=lib.sublib'       <------- for catalog
/*
// MTC REW,SYS004

ASSGN SYSIPT,FEC
/*
/&
* $$ EOJ
```

where:

nnn is the tape address
lib.sublib is the library and sublibrary of the catalog

Step 2 - Modify COPYTAPE.JOB

Modify `COPYTAPE.JOB` to conform to your local naming conventions and set the disk space para-
meters before submitting this job.

Step 3 - Submit COPYTAPE.JOB

Submit `COPYTAPE.JOB` to unload all other datasets from the tape to your disk.

Transferring a License Key File from PC to Host with FTP

If a license key file is supplied as an e-mail attachment, you must transfer the attached license key file `natvr.xml` from the PC to the mainframe, using native FTP commands.

> ⚠ **Caution:** Using utilities instead of native FTP commands for the license key file transfer may corrupt the license key and thus prevent Natural from execution later on. This applies for example to file transfer based on 3270 terminal emulations that do not provide a true binary file transfer, but convert specific characters.

To transfer a license key file from the PC to the mainframe, perform the following steps:

1. Save the product license key file e-mail attachment on your PC hard disk.

2. Open a command prompt window. In the command prompt window, change to the directory where you saved the product license key file.

3. Start an FTP session for communication with the z/VSE host:

   ```
   ftp host-name
   ```

   Where `host-name` is the name of the z/VSE host.

4. Enter your z/VSE host login `user` and `password`.

5. Once the FTP session has been established, specify the z/VSE sublibrary for the license key file:

   ```
   ftp>cd lib.sublib
   ```

6. Switch to binary data mode (the license key file must retain its ASCII format during the transfer):

   ```
   ftp>bin
   ```

7. Copy the license key file as a member into the specified sublibrary.

```
ftp>put natvr.xml NATvrs.LICS
```

This command will create a member `NATvrs.LICS` in the specified sublibrary.

8. Stop your FTP session by entering:

```
ftp>quit
```

Installation Procedure for Natural under z/VSE

▶ **Step 1: Convert and Assemble the License Key File**

Convert member `NATvrs.LICS` in sublibrary `lib.sublib` with `PROGRAM NATLICAV` in an Assembler input file.

Assemble/link into a linkable module. This module is linked (Job I060) to the Natural nucleus.

■ (Job I006, Step 0104, and Job I007, Step 0104)

Job I006, Step 0104	Copy Natural license key into sublibrary.
Job I007, Step 0104	Convert and generate license key module (`NATLIC`).

▶ **Step 2: Load the FNAT System File**

■ (Job I050, Step 0100)

If you are installing into an existing Natural 4.1 `FNAT` file, skip this step.

Load the empty Natural system file (dataset `NATvrs.SYSF`) using the `ADALOD` utility.

This file will contain all Natural objects supplied by Software AG. Its size depends on the number of products to be installed later. As a rule of thumb, 20 MB can be assumed for each major Software AG product.

The following `ADALOD` parameters must not be altered:

```
ISNREUSE=YES
```

To avoid Natural errors NAT9988 and NAT7397 after reorganization of the `FNAT` system file using `ADAULD/ADALOD`, the parameter `USERISN` should be set to `YES`.

The file number *fnat* of the `FNAT` system file can be chosen as described under Natural profile parameter `FNAT` in the Natural *Parameter Reference* documentation.

▶ **Step 3: Load the FUSER System File**

■ (Job I050, Step 0101)

You have the following options:

1. You can use an existing Version 4.1 `FUSER` file, then you can skip this step.
2. You can use a new `FUSER` file for Version 4.2.
3. You can use an existing Version 4.1 `FUSER` file to be shared by Versions 4.1 and 4.2.
4. You can use an existing Version 4.1 `FUSER` file to be used by Version 4.2 only.

Use a New Version 4.2 FUSER System File - First-Time Installation

For the use of a new and empty `FUSER` system file for Natural Version 4.2, no additional system-file-related actions are necessary.

If you do *not* want to share the `FUSER` system file, proceed as follows:

Load the empty Natural user file contained in dataset `NATvrs.SYSF` using the `ADALOD` utility.

In this file, all user-written Natural programs are stored.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number *fuser* of the `FUSER` system file can be chosen as described under Natural profile parameter `FUSER` in the Natural *Parameter Reference* documentation.

Reuse an Existing Version 4.1 FUSER System File - Migration from Natural Version 4.1

If you want to use the existing Natural Version 4.1 `FUSER` system file and you do not want to share the `FUSER` system file, skip this step.

Using a Version 4.1 FUSER File to be Shared by Natural Versions 4.1 and 4.2

If you use an existing Natural Version 4.1 `FUSER` system file to be shared by Natural Versions 4.1 and 4.2, you must upgrade your Natural Version 4.1 installation to Version 4.1.4.

Natural Version 4.1.4 Service Pack I003 or a subsequent Service Pack is required. Service Pack I003 and all subsequent Service Packs contain all the necessary Version 4.1 based solutions for Natural Version 4.2.

User Application Programming Interfaces USR* in Library SYSEXT

The `USR*` programs from the delivered library `SYSEXT` run in a special mode. As a result, the `USR*` programs do not need to set further steplibs to execute related objects for processing. This reduces the impact on the Natural buffer pool search logic and improves the performance significantly if user exits are used extensively within user written applications.

It is necessary that the user exits are cataloged with Natural Version 4.2. This implies that the user exits cannot be executed with Natural Version 4.1.

Use of USR* Programs

Usually, the access of `USR*` programs by an application requires that the user application programming interfaces be copied from library `SYSEXT` to either the application libraries on the `FUSER` system file or to library `SYSTEM` on the `FUSER` system file or to library SYSTEM on the `FNAT` system file, respectively, or any other library which is defined as steplib for the application. Library `SYSEXT` can also be used as steplib. Due to the fact that the delivered user application programming interfaces will always be cataloged with the latest Natural version, we recommend that the user application programming interfaces should reside on the `FNAT` system file. This will ensure that the right version is executed and will separate user written applications from Software AG modules.

If applications which call user application programming interfaces should run with both Natural Version 4.1 and Natural Version 4.2, it must be made sure that the user application programming interfaces delivered with the corresponding Natural version are used.

The following scenarios may be considered:

Using a Version 4.1 FUSER File for Natural Version 4.1 and 4.2

If the same `FUSER` system file shall be used in a Natural Version 4.1 and Version 4.2 environment in parallel the following steps are recommended:

- Remove all `USR*` modules you have copied from library `SYSEXT` into application libraries on your `FUSER` system file.

- In both environments, copy the used `USR*` modules from library `SYSEXT` to library `SYSTEM` on the corresponding `FNAT` system file.

- Alternatively, the `USR*` modules can be moved to another system library on `FNAT` which then must be defined as steplib, or library `SYSEXT` can be used as steplib for the applications. Then automatically in both environments the right versions of the user application programming interfaces are executed.

Using a Version 4.1 FUSER File for Natural Version 4.2 only

If you want to use the existing Natural Version 4.1 `FUSER` system file and you do not want to share the `FUSER` system file, then it is still possible to replace all `USR*` modules you have copied from library `SYSEXT` into application libraries with the new `USR*` objects from the Version 4.2 library `SYSEXT`.

But the preferred way is to remove all user application programming interfaces on the `FUSER` system file and copy the used user application programming interfaces from library `SYSEXT` to library system of the FNAT system file or use a `SYS` library on `FNAT` as steplib.

Using a New FUSER File for Natural Version 4.2

If you want to port existing applications to a new `FUSER` system file, copy all application objects but no Software AG `USR*` objects to the new `FUSER` system file. Then proceed as described in the scenario above.

Migration

The `FIND` function of the Natural utility `SYSMAIN` can be used to search for all `USR*` modules stored in a specific library on the `FUSER` system file or across the whole system file. In addition, Predict cross reference data can be used to determine all referenced user application programming interfaces.

The file number *fuser* of the `FUSER` system file can be chosen as described under Natural profile parameter `FUSER` in the Natural *Parameter Reference* documentation.

▶ **Step 4: Load the Scratch-Pad File**

■ (Job I050, Step 0102)

The scratch-pad file (which is a Natural-internal system file) can be used exclusively by the new Natural version or it can be shared by different versions of Natural.

If you do *not* want to use a scratch-pad file, skip this step.

If you do want to use a scratch-pad file; that is, if you want to use read-only system files (profile parameter `ROSY=ON`), see also *Natural Scratch-Pad File* in the Natural *Operations* documentation, proceed as follows:

Load the empty scratch-pad file contained in dataset `NATvrs.SYSF`, using the `ADALOD` utility as described below.

The following `ADALOD` parameter *must not* be altered:

```
ISNREUSE=YES
```

For the optional scratch-pad file inclusion, the following NATPARM parameters must be added or, if already present, updated with:

```
LFILE=(212,dbid,fnr)
ROSY=ON
```

If you use SMA: To be able to use read-only system files, set the SMA parameter `NAT-SCRF=Y` and the Natural profile parameter `ROSY=ON`.

▶ **Step 5: Load the FDIC System File**

■ (Job I050, Step 0103)

Skip this step:

- if you want to install Predict (in this case, use the corresponding installation step in the Predict *Installation* documentation), or

- if you want to use an existing `FDIC` system file (an existing `FDIC` system file can be shared by Natural Versions 4.1 and 4.2), or

- if you do not use your own `FDIC` system file.

Load the empty `FDIC` system file contained in dataset `NATvrs.SYSF` using the `ADALOD` utility, as described below.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number `fdic` of the `FDIC` system file can be chosen as described under Natural profile parameter `FDIC` in the Natural *Parameter Reference* documentation.

▶ **Step 6: Load the FSEC System File**

■ Skip this step,

- if you do not use Natural Security, or

- if you want to use an existing `FSEC` system file, or

- if you do not want to use an own `FSEC` system file.

If you use Natural Security, refer to *Installing Natural Security* in the Natural *Installation* documentation.

▶ **Step 7: Assemble the Natural z/VSE Interface Module**

- (Job I055, Step 0100)

  Set the parameters in the source code of the module `NATVSE` contained in the sublibrary *lib.sublib* to fit your site requirements.

  For a description of the `NTVSE` generation parameters, refer to *NTVSE Macro - Generation Parameters for Natural under z/VSE* in the Natural *Operations* documentation.

  Assemble and link the Natural z/VSE interface module `NATVSE` contained in the sublibrary *lib.sublib*.

▶ **Step 8: Create the Natural Configuration Module**

- (Job I055, Step 0110)

  This step is only required if you need to change the delivered `NATCONFG` module, for example, to adapt the `NTDVCE` macro definition to your requirements.

  Change and assemble the source contained in the sublibrary *lib.sublib*.

  Link the resulting Natural configuration module (`NATCONFG`) to the Natural nucleus when you link the Natural nucleus (see below).

  For more information on the configuration tables in `NATCONFG`, refer to *Natural Configuration Tables* in the Natural *Operations* documentation.

▶ **Step 9: Create the Parameter Module**

- (Job I060, Step 0010)

  Create the Natural batch parameter module (Job I060, Step 0010).

The following parameters in the parameter module must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

For *dbid*, *fnat* and *fuser*, use the values you specified in the previous steps when you loaded the system files (see above).

Global Natural Buffer Pool: If you wish to use a *global* Natural buffer pool, specify the parameter macro `NTBPI` with the name of the global Natural buffer pool and set the profile parameter `SUBSID` in all your Natural parameter modules.

If you use SMA: Define the name of the global Natural buffer pool in SMA parameter `NAT-GLOBAL-BP`.

For all other parameters, you can generally use the default values.

Modify only the values of those parameters whose default values do not meet your requirements.

For a detailed description of the individual parameters contained in the parameter module, refer to the Natural *Parameter Reference* documentation.

Assemble the parameter module.

▶ **Step 10: Link the Natural Nucleus**

(Job I060, Step 0020)

The following alternatives exist:

Link a batch front-end (Job I060, Step 0020) and link a shared nucleus (Job I060, Step 0105), or link a non-shared nucleus (Job I060, Step 0020).

The Natural z/VSE batch front-end requires `RMODE(24)` due to the use of z/VSE macros which have this restriction. Therefore the use of a shared nucleus is highly recommended, as a non-shared nucleus would also require `RMODE(24)`.

■ **Using Alternative 1**

If you use SMA: Ensure that the SMA parameter `SHARED-NUC` is set to `Y` (yes).

1. Link a batch front-end (Job I060, Step 0020).

The following modules must be included (include module `NATVSE` first):

Module	Function
NATVSE	Batch Natural driver
NATVSERV	Natural z/VSE services
PRM020BA	Generated parameter module created above
ADAUSER	Adabas link module; optionally you can use `ADANAME` instead.

To access the shared nucleus, ensure that the parameter `NUCNAME` in the module `NATPARM` contains the name of the shared module linked in Job I060, Step 0105.

2. Link a shared nucleus (Job I060, Step 0105).

For a list of the modules included, see *Natural Shared Nucleus under z/OS and z/VSE* in the Natural *Operations* documentation.

Or:

Using Alternative 2

Link a non-shared nucleus (Job I060, Step 0020).

If you use SMA: Ensure that parameter `SHARED-NUC` is set to `N` (no).

If you do not use SMA: Merge all `INCLUDE` statements and corresponding DD cards from Job I060, Step 0105 (shared nucleus) into Job I060, Step 0020 (front-end).

▶ **Step 11: Link the Global Buffer Pool Modules**

■ (Job I060, Steps 0120, 0126, 0127, 0128, 0129)

```
Step 0120 LINK GLOBAL BUFFER POOL MANAGER NATGBP
     0126 EXAMPLE JOB TO START NATURAL GLOBAL BUFFER POOL
     0127 EXAMPLE JOB TO STOP NATURAL GLOBAL BUFFER POOL
     0128 EXAMPLE JOB TO START EDITOR GLOBAL BUFFER POOL
     0129 EXAMPLE JOB TO STOP EDITOR GLOBAL BUFFER POOL
```

This step is required if you want to use global buffer pools in z/VSE. See *Global Buffer Pool under z/VSE* in the Natural *Operations* documentation.

▶ **Step 12: Delete Natural System Objects**

■ This step is optional but recommended to avoid data inconsistencies.

If you are using a Version 4.1 Natural `FNAT` system file, you can delete obsolete Version 4.1 Natural objects.

If you are using SMA: Set the SMA parameter `NAT-FNAT41` to `Y` (yes).

If only base Natural Version 4.1 is installed

Delete the Version 4.1 Natural objects by loading the `NATvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0010)

If Natural Review Version 4.1 is installed

Delete the Version 4.1 Natural Review objects by loading the `RNMvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0026)

See also *Delete Natural Review Objects* in the *Natural Review* documentation.

If Natural Security Version 4.1 is installed

Delete the Version 4.1 Natural Security objects by loading the `NSCvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0099)

See also *Delete Natural Security Objects* in *Installing Natural Security*.

▶ **Step 13: Load the System Programs**

■ (Job I061, Step 0100)

Use the Natural `INPL` utility to load the Natural system objects (dataset `NATvrs.INPL`) into the Natural system files.

▶ **Step 14: Load the Error Messages**

■ (Job I061, Step 0102)

Load the English Natural error messages file (dataset `NATvrs.ERRN`) using the program `ERRLODUS`. In addition to the English error message short and long texts, error message short texts in German language (`ULANG=2`) are loaded. Use the program `ERRUPPER` to convert the error message texts to upper case. Both programs are described in the Natural *SYSERR Utility* documentation.

Load the Japanese Error Message Long Texts (dataset `NCJvrs.INPL`)

(Job I061, Step 8300)

This step is only required if you wish to replace the English error message long texts with their Japanese equivalents.

The Natural error message long texts in Japanese language are contained in the Natural Japanese Language Pack. This is a separate product (product code NCJ) that can be loaded optionally. If you do not load the Natural error message long texts in Japanese language, English error message long texts will appear.

Use the Natural utility `INPL` to load the Japanese error message long texts (dataset `NCJvrs.INPL`) into the Natural system files.

Load the Japanese Error Message Short Texts (dataset `NCJvrs.ERRN`)

(Job I061, Step 8302)

This step is only required if you wish to load the error message short texts in Japanese language (`ULANG=59`).

The Natural error message short texts in Japanese language are contained in the Natural Japanese Language Pack. This is a separate product (product code NCJ) that can be loaded optionally. If you do not load the Natural error message short texts in Japanese language, English error message short texts will appear.

Use the program `ERRLODUS` described in the Natural *SYSERR Utility* documentation to load the error message short texts (dataset `NCJvrs.ERRN`) into the Natural system file.

▶ **Step 15: Load the Examples**

■ (Job I061, Step 0103)

Use the `INPL` utility to load the Natural example objects (dataset `NATvrs.EXPL`) into the Natural system file.

Installation Verification for Natural under z/VSE

For base Natural, there are no specific installation verification procedures.

After the last step of the installation procedure has been successfully performed, check that the following results are available:

■ Communication between Adabas and Natural is working.

■ The Natural system files have been loaded.

■ Batch Natural is operational.

6 **Installing Natural under VM/CMS**

This document describes how to install Natural under VM/CMS.

For information on how to run Natural in a CMS environment, refer to *Natural under CMS* in the Natural *Operations* documentation.

For installation-related information on Unicode and code page support, refer to *Configuration and Administration of the Unicode/Code Page Environment* in the *Unicode and Code Page Support* documentation.

Notation vrs or vr: If used in the following document, the notation vrs or vr stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

The following products must be installed:

- A supported version of the z/VM operating system must be installed. For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- A supported version of Adabas must be installed. For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- As a rule of thumb, each major Software AG product requires approximately 20 MB space in the Adabas database to store the Natural objects supplied by Software AG.

Preparing the VM System for Natural CMS

Natural is completely reentrant under CMS. It is recommended that Natural under CMS be installed as a Discontiguous Shared Segment (DCSS) to avoid unnecessary paging activity by VM.

For better performance, Natural uses a buffer pool in which Natural object programs are stored and executed. If you execute a Natural program, it is fetched from the database and stored in the buffer pool. If you subsequently invoke the same program, it will be executed directly in the buffer pool. Thus, repeated fetching from the database and accompanying IUCV overhead is avoided.

In a shared buffer pool, one Natural under CMS user can execute Natural programs that have been placed in the buffer pool by another Natural under CMS user. A shared buffer pool is implemented by installing the buffer pool in a writeable saved segment.

You are recommended to install two DCSSs: one for the Natural nucleus and one for the buffer pool. Of course, several buffer pools at different virtual addresses can be defined and several Natural nucleus DCSSs with different functionalities, for example, different static parameter settings (`NATPARM`) or different Assembler options of the CMS driver (`NATCMS`).

The buffer pool shared segment is defined as `SN` (see the corresponding step of the installation procedure) and the first Natural session automatically initializes the buffer pool. When the last Natural user in VM exits Natural, the use count of the shared segment becomes 0, the buffer pool is purged from CP storage, and the next Natural session has to initialize the buffer pool again.

To avoid repeated initialization, the use count can be artificially held at 1 by including the following statement line in the `PROFILE EXEC` of a disconnected service machine, *segment* being the name of the Natural buffer pool shared segment:

```
SEGMENT LOAD segment
```

In this way, the Natural buffer pool is initialized only once and immediately available to a Natural session as long as the disconnected server is running.

Preparing the Adabas System for Natural

The Natural system programs are stored in the Natural system file `FNAT`. User-written Natural programs are stored in the Natural system file `FUSER`. These system files reside in Adabas files. The corresponding step of the installation procedure (see below) describes how these Adabas files can be built in an Adabas system running under CMS.

Building the Natural CMS System

Building the Natural CMS system comprises the following steps:

- building the Natural nucleus and
- loading the Natural system file.

The Natural nucleus should be built

- as a module (for testing) and
- as a saved segment (for production use).

The Natural system file is loaded directly from tape using the installation `EXEC`, `NATINPL`; see the corresponding step of the installation procedure.

The CMS driver `NATCMS` is included in the Natural module and in the saved segment. `NATCMS ASSEMBLE` consists of one macro that generates the driver.

Description of the Macro Parameters

For a detailed description of the macro parameters, refer to the corresponding step of the installation procedure. The driver `NATCMS` and the parameter module `NATPARM` are assembled during the installation process. Optionally, the modules `NATTEXT`, `NATTXT2`, `NATTXT3`, `NATCONFG` and `NATPM` can be modified and reassembled.

The list of text files to be included in the Natural module or DCSS is contained in REXX program `NAT$LOAD EXEC` (variable `LOADLIST`). To customize your Natural system, modify this `EXEC` with `XEDIT` by changing the `LOADLIST` as required.

User ID for Performing the Installation

The user ID used for performing the installation should have a minidisk of 80 cylinders on 3390-type disks or an SFS directory of 14000 blocks. CP privilege class E is needed to issue the command `CP SAVESYS`. The size of the virtual machine allowed for the user ID must be adequate to accommodate the highest address of the saved segments that you plan to generate.

Installation Tape for Natural under CMS

The installation tape was created under z/OS; it has standard z/OS labels and headers. It contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NAT*vrs*.LICS	Natural License Key File. For further information on license key file, license key file installation, product license check and product license check FAQs, see *Licensing Natural*. If a license key file is supplied as an e-mail attachment, see *Transferring the License Key File from PC to Host with FTP*
NAT*vrs*.SYSF	Empty Natural system file.
NAT*vrs*.TAPE	CMS installation material. This dataset is in TAPE DUMP format and must be loaded onto the installation minidisk.
NAT*vrs*.ERRN	Natural error messages.
NAT*vrs*.LDEL	Instructions to delete Natural system objects of Version 4.1
NAT*vrs*.INPL	Natural system objects.
NAT*vrs*.EXPL	Natural example objects.

Copying the Tape Contents to a VM/CMS Disk

1. On the Natural installation tape, a license key file is provided.

   If the sequence number of NAT*vrs*.LICS, as shown by the *Report of Tape Creation*, is *n*, you must position over 3 *n* - 2 tape marks (that is, FSF 1 for the first dataset, FSF 4 for the second, etc.).

   ```
   FILEDEF IN TAP1 (RECFM FB LRECL 80 BLKSIZE 3120
   FILEDEF OUT DISK NATLIC DATA A (RECFM F LRECL 80 BLKSIZE 80
   MOVEFILE IN OUT
   ```

2. To position the tape for the TAPE LOAD command, calculate the number of tape marks as follows:

   If the sequence number of NAT*vrs*.TAPE, as shown by the *Report of Tape Creation*, is *n*, you must position over 3 *n* - 2 tape marks (that is, FSF 1 for the first dataset, FSF 4 for the second, etc.).

3. Access the disk that is to contain the Natural installation files as disk A.

   The size of the disk must be at least 14000 4-KB blocks, for example, 80 cylinders on 3390-type disks.

4. Ask the system operator to attach a tape drive to your virtual machine at address `X'181'` and mount the Natural installation tape.

5. Position the tape by issuing the CMS command:

```
TAPE FSF fsfs
```

where `fsfs` is the number of tape marks and is calculated as described above.

6. Load the Natural under CMS installation material by issuing the CMS command:

```
TAPE LOAD * * A
```

7. Keep the tape drive attached to your virtual machine, because the tape is still needed during the installation procedure.

Transferring a License Key File from PC to Host with FTP

If a license key file is supplied as an e-mail attachment, you must transfer the attached license key file `natvr.xml` from the PC to the mainframe, using native FTP commands:

🔴 **Caution:** Using utilities instead of native FTP commands for the license key file transfer may corrupt the license key and thus prevent Natural from execution later on. This applies for example to file transfer based on 3270 terminal emulations that do not provide a true binary file transfer, but convert specific characters.

To transfer a license key file from the PC to the mainframe, perform the following steps:

1. Save the product license key file e-mail attachment on your PC hard disk.

2. Open a command prompt window. In the command prompt window, change to the directory where you saved the product license key file.

3. Start an FTP session for communication with the z/VM host:

```
ftp host-name
```

Where `host-name` is the name of the z/VM host.

4. Enter your z/VM host login ID and password; for example: `<cmsmachine.by.user>`

5. Once the FTP session has been established, specify the CMS file system (SFS) or access a minidisk for the license key file:

```
ftp>cd 'SFS'
```

6. Switch to binary data mode (the license key file must retain its ASCII format during the transfer):

```
ftp>binary
```

7. Specify that the dataset for the license key file must be written with RECFM FIX 80.

```
ftp>quote site fix 80
```

8. Write the license as a dataset on the z/VM system. For example, if the license key file name is nat*vrs*.xml, you might enter:

```
ftp>put natvr.xml
```

This command will create a dataset called NAT*vr*.LICS.

9. Rename or copy the dataset as follows:

```
NATLIC DATA A
```

The license key information stored in the dataset will be in ASCII format.

10. Stop your FTP session by entering:

```
ftp>quit
```

Installation Procedure for Natural under CMS

▶ **Step 1: Define the Natural Discontiguous Saved Segments to VM**

■ Consult your system programmer for the virtual address range where the Natural DCSS should be defined. The DCSS should not overlap with any other DCSS that you might be using during execution of a Natural session. In particular, if you plan to call 3-GL programs that use Language Environment services, or if you set NATCMS parameter LE370=YES, make sure that your Natural DCSS does not overlap with the DCSS named SCEE and SCEEX.

Execute the DEFSEG class E command for the Natural nucleus and for the Natural buffer pool.

Examples:

```
DEFSEG NATvr 3000-3FFF SR
DEFSEG NATBPvr 500-5FF SN
SAVESEG NATBPvr
```

> **Note:** Segment `NATBPvr` is saved as an empty segment. Segment `NATvr` will be saved later when the Natural nucleus has been loaded into it.

▶ **Step 2: Load the FNAT System File**

■ If you are installing into an existing Natural 4.1 `FNAT` file, skip this step.

Load the empty Natural system file (dataset `NATvrs.SYSF`) using the `ADALOD` utility.

This file will contain all Natural objects supplied by Software AG. Its size depends on the number of products to be installed later. As a rule of thumb, 20 MB can be assumed for each major Software AG product.

The following `ADALOD` parameters must not be altered:

```
ISNREUSE=YES
```

To avoid Natural errors NAT9988 and NAT7397 after reorganization of the `FNAT` system file using `ADAULD`/`ADALOD`, the parameter `USERISN` should be set to `YES`.

The file number *fnat* of the `FNAT` system file can be chosen as described under Natural profile parameter `FNAT` in the Natural *Parameter Reference* documentation.

Please, also note the following when defining a system file:

On the Natural installation tape, an input file for the `ADALOD` utility is provided.

It is easiest to perform the `ADALOD` operation from disk. You are recommended to copy the file `NATvrs.SYSF` from tape to disk and to make additional copies for loading the `FUSER`, `FDIC` and `FSEC` system files:

```
FILEDEF IN TAP1 SLn VOLIDnnnnnn (RECFM VB LRECL 9996 BLKSIZE 10000
FILEDEF OUT DISK NATvrSYS LODDATA A
MOVEFILE IN OUT
```

To make additional copies, issue the following commands, for example:

```
COPYFILE NATvrSYS LODDATA A NATvrUSE LODDATA A
COPYFILE NATvrSYS LODDATA A NATvrDIC LODDATA A
COPYFILE NATvrSYS LODDATA A NATvrSCR LODDATA A
```

Modify the member `NATvrSYS LODLIB` to meet your requirements. In particular replace the question mark (?) specified for the `FILE` parameter with the file number chosen for your `FNAT` system file.

You can now invoke the `ADALOD EXEC` for loading the Natural system file:

```
ADALOD NATvrSYS
```

▶ **Step 3: Load the FUSER System File**

■ You have the following options:

1. You can use an existing Version 4.1 `FUSER` file, then you can skip this step.

2. You can use a new `FUSER` file for Version 4.2.

3. You can use an existing Version 4.1 `FUSER` file to be shared by Versions 4.1 and 4.2.

4. You can use an existing Version 4.1 `FUSER` file to be used by Version 4.2 only.

Use a New Version 4.2 FUSER System File - First-Time Installation

For the use of a new and empty `FUSER` system file for Natural Version 4.2, no additional system-file-related actions are necessary.

If you do *not* want to share the `FUSER` system file, proceed as follows:

Load the empty Natural user file contained in dataset `NATvrs.SYSF` using the `ADALOD` utility.

In this file, all user-written Natural programs are stored.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number *fuser* of the `FUSER` system file can be chosen as described under Natural profile parameter `FUSER` in the Natural *Parameter Reference* documentation.

Reuse an Existing Version 4.1 FUSER System File - Migration from Natural Version 4.1

If you want to use the existing Natural Version 4.1 `FUSER` system file and you do not want to share the `FUSER` system file, skip this step.

Using a Version 4.1 FUSER File to be Shared by Natural Versions 4.1 and 4.2

If you use an existing Natural Version 4.1 `FUSER` system file to be shared by Natural Versions 4.1 and 4.2, you must upgrade your Natural Version 4.1 installation to Version 4.1.4.

Natural Version 4.1.4 Service Pack I003 or a subsequent Service Pack is required. Service Pack I003 and all subsequent Service Packs contain all the necessary Version 4.1 based solutions for Natural Version 4.2.

User Application Programming Interfaces USR* in Library SYSEXT

The `USR*` programs from the delivered library `SYSEXT` run in a special mode. As a result, the `USR*` programs do not need to set further steplibs to execute related objects for processing. This reduces the impact on the Natural buffer pool search logic and improves the performance significantly if user exits are used extensively within user written applications.

It is necessary that the user exits are cataloged with Natural Version 4.2. This implies that the user exits cannot be executed with Natural Version 4.1.

Use of USR* Programs

Usually, the access of `USR*` programs by an application requires that the user application programming interfaces be copied from library `SYSEXT` to either the application libraries on the `FUSER` system file or to library `SYSTEM` on the `FUSER` system file or to library SYSTEM on the `FNAT` system file, respectively, or any other library which is defined as steplib for the application. Library `SYSEXT` can also be used as steplib. Due to the fact that the delivered user application programming interfaces will always be cataloged with the latest Natural version, we recommend that the user application programming interfaces should reside on the `FNAT` system file. This will ensure that the right version is executed and will separate user written applications from Software AG modules.

If applications which call user application programming interfaces should run with both Natural Version 4.1 and Natural Version 4.2, it must be made sure that the user application programming interfaces delivered with the corresponding Natural version are used.

The following scenarios may be considered:

Using a Version 4.1 FUSER File for Natural Version 4.1 and 4.2

If the same FUSER system file shall be used in a Natural Version 4.1 and Version 4.2 environment in parallel the following steps are recommended:

■ Remove all USR* modules you have copied from library SYSEXT into application libraries on your FUSER system file.

■ In both environments, copy the used USR* modules from library SYSEXT to library SYSTEM on the corresponding FNAT system file.

■ Alternatively, the USR* modules can be moved to another system library on FNAT which then must be defined as steplib, or library SYSEXT can be used as steplib for the applications. Then automatically in both environments the right versions of the user application programming interfaces are executed.

Using a Version 4.1 FUSER File for Natural Version 4.2 only

If you want to use the existing Natural Version 4.1 FUSER system file and you do not want to share the FUSER system file, then it is still possible to replace all USR* modules you have copied from library SYSEXT into application libraries with the new USR* objects from the Version 4.2 library SYSEXT.

But the preferred way is to remove all user application programming interfaces on the FUSER system file and copy the used user application programming interfaces from library SYSEXT to library system of the FNAT system file or use a SYS library on FNAT as steplib.

Using a New FUSER File for Natural Version 4.2

If you want to port existing applications to a new FUSER system file, copy all application objects but no Software AG USR* objects to the new FUSER system file. Then proceed as described in the scenario above.

Migration

The FIND function of the Natural utility SYSMAIN can be used to search for all USR* modules stored in a specific library on the FUSER system file or across the whole system file. In addition, Predict cross reference data can be used to determine all referenced user application programming interfaces.

The file number *fuser* of the FUSER system file can be chosen as described under Natural profile parameter FUSER (in the Natural Parameter Reference documentation).

Modify the member `NAT vrUSE LODLIB` to suit your requirements. In particular replace the question mark (?) specified for the `FILE` parameter with the file number chosen for your `FUSER` system file.

You can now invoke the `ADALOD EXEC` for loading the Natural system file:

```
ADALOD NATvrUSE
```

▶ **Step 4: Load the Scratch-Pad File**

■ The scratch-pad file (which is a Natural-internal system file) can be used exclusively by the new Natural version or it can be shared by different versions of Natural.

If you do *not* want to use a scratch-pad file, skip this step.

If you do want to use a scratch-pad file; that is, if you want to use read-only system files (profile parameter `ROSY=ON`), see also *Natural Scratch-Pad File* in the Natural *Operations* documentation, proceed as follows:

Load the empty scratch-pad file contained in dataset `NATvrs.SYSF`, using the `ADALOD` utility as described below.

The following `ADALOD` parameter *must not* be altered:

```
ISNREUSE=YES
```

For the optional scratch-pad file inclusion, the following NATPARM parameters must be added or, if already present, updated with:

```
LFILE=(212,dbid,fnr)
ROSY=ON
```

Modify the member `NATvrSCR LODLIB` to suit your requirements. In particular replace the question mark (?) specified for the `FILE` parameter with the file number chosen for your scratch-pad file.

You can now invoke the `ADALOD EXEC` for loading the Natural system file:

```
ADALOD NATvrSCR
```

▶ **Step 5: Load the FDIC System File**

■ Skip this step:

- if you want to install Predict (in this case, use the corresponding installation step in the Predict *Installation* documentation), or

- if you want to use an existing `FDIC` system file (an existing `FDIC` system file can be shared by Natural Versions 4.1 and 4.2), or

- if you do not use your own `FDIC` system file.

Load the empty `FDIC` system file contained in dataset `NATvrs.SYSF` using the `ADALOD` utility, as described below.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number `fdic` of the FDIC system file can be chosen as described under Natural profile parameter `FDIC` in the Natural *Parameter Reference* documentation.

Modify the member `NATvrDIC LODLIB` to meet your requirements. In particular replace the question mark (?) specified for the `FILE` parameter with the file number chosen for your `FDIC` system file.

You can now invoke the `ADALOD EXEC` for loading the Natural system file:

```
ADALOD NATvrDIC
```

▶ **Step 6: Load the FSEC System File**

■ Skip this step,

- if you do not use Natural Security, or

- if you want to use an existing `FSEC` system file, or

- if you do not want to use an own `FSEC` system file.

If you use Natural Security, refer to *Installing Natural Security* in the Natural Installation documentation.

▶ **Step 7: Generate a TEXT file from the Software AG license key file**

■ The file `NATLIC EXEC` uses as input the `NATLIC DATA A` which you have already copied with `MOVEFILE`.

Execute `NATLIC EXEC` to generate the `TEXT` file `NATLIC TEXT A`.

This `TEXT` file must be included in the Natural nucleus, or loaded dynamically at Natural start-up by using the profile parameter `RCA`.

▶ **Step 8: Customize the Natural Parameter Module**

■ The file `NATPARM ASSEMBLE` contains sample settings of Natural profile parameters.

Use `XEDIT` to set the Natural profile parameters `FNAT`, `FUSER`, `FSEC` and `FDIC` to the correct values for your installation.

Note that these parameters are in effect for the `INPL` procedure in one of the following steps.

You must define the Natural buffer pool using the parameter macro `NTBPI`.

Example:

If you defined a buffer pool shared segment named `NATBPvr`, code:

```
NTBPI TYPE=NAT,NAME=NATBPvr
NTBPI TYPE=NAT,SIZE=1024
```

Natural will be instructed to load the shared segment `NATBPvr` and use it as its buffer pool. If the shared segment cannot be loaded, a buffer pool of 1024 KB is allocated from CMS free storage and the initialization error message NAT1074 is displayed.

You can suppress the display of initialization error messages by coding:

```
IMSG=OFF
```

You may wish to include

```
HCAM=CMS
```

to enable the *Hardcopy Function* (`%H`) as described in the Natural *Operations* documentation. See also *Hardcopy Output* in the Natural *Terminal Commands* documentation).

Set other profile parameters as required by your installation.

For a description of these parameters, refer to the Natural *Parameter Reference* documentation.

z/VM does not support CPU usage time measurement; as a result, the profile parameter `MT` reacts to elapsed time instead of CPU usage time. Therefore, set the profile parameter `MT` to `MT=0`.

▶ **Step 9: Customize the INCLUDE List for Linking of Natural**

■ The program `NAT$LOAD EXEC` is used in the following two steps to build Natural for CMS.

You may wish to use `XEDIT` to customize the contents of variable `LOADLIST`, for example, to include your own routines that you defined as `CSTATIC` in `NATPARM ASSEMBLE`.

▶ **Step 10: Linking the Adabas Interface**

■ Before linking Natural, copy the Adabas interface from the Adabas text library.

You can use the following CMS commands to perform this task:

```
FILEDEF IN DISK ADAVvrs TXTLIB fm (MEMBER ADABAS
FILEDEF OUT DISK ADABAS TEXT A
MOVEFILE IN OUT
```

where *ADAVvrs* stands for your current Adabas text library and *fm* for the filemode where you access this text library.

▶ **Step 11: Generate a Natural Module**

■ Issue the following CMS command:

```
NATBLDM
```

You are prompted for the name of the Natural module.

Enter a valid CMS file name, for example, `NATvrs`.

`NATBLDM` checks `NATCMS TEXT` and `NATPARM TEXT`.

If a `TEXT` file is not found or if it is older than the corresponding source file, `NATBLDM` asks you whether you want to assemble (reassemble) the source file. If so, you are given the opportunity to `XEDIT` the source first, in order to define installation-dependent parameters.

Note that `NATCMS TEXT` does not exist during an initial installation and that an assembly is forced.

Enter `YES` when asked whether you want to edit `NATCMS`.

The source program `NATCMS ASSEMBLE` contains a call to the macro `NTCMS` which generates the Natural VM/CMS interface.

Set the parameters that are described in `NATCMS ASSEMBLE` as required to meet your requirements.

Parameters in Macro NTCMS

The following parameters can be specified in the macro `NTCMS`:

Parameter	Description
`ATTNKEY=key`	This parameter specifies the attention key that is used to interrupt a running Natural program. When the specified key is pressed, Natural error NAT1016 will be raised. Valid keys are PA1 to PA3 and PF1 to PF24. Specify `NONE` if no attention key is to interrupt a Natural program. If the `ATTNKEY` parameter is omitted, all attention keys will cause error message NAT1016 to be issued.
`CEEXOPT=(LE run-time options)`	Specifies the LE run-time options if Natural runs LE-enabled. This parameter takes effect only with `LE370=YES`. For a description of LE run-time options, please refer to IBM documentation *Language Environment Customization*. Do not specify the `TRAP` option other than `TRAP=(ON,NOSPIE)`. The entire list of options must be enclosed in parentheses. If you plan to call subprograms that run in `AMODE24`, specify `CEEXOPT=(ALL31=OFF,STACK=(,,BELOW))`.

Parameter	Description

`LE370=YES`
 	
`LE370=NO`
	With `LE370=YES`, Natural runs as an LE program. You can call external subprograms according to the IBM calling conventions.
	
	With `LE370=NO`, you can only call main programs of the Language Environment. A new LE enclave is created and terminated for each `CALL` to a 3GL LE-enabled program.
	
	`LE370=NO` is the default.

`LOADSS=(name1,name2,...)`	This parameter specifies the names of programs that reside in saved segments and are to be loaded dynamically during execution of the Natural `CALL` statement. If the name of a program differs from the name of the saved segment in which it is contained, specify the two names in parentheses. If, for example, you specify `LOADSS=(ALPHA,(BETA,GAMMA))` and a Natural program executes `CALL 'ALPHA'`, the saved segment named `ALPHA` is loaded and control is passed to it. When a Natural program executes `CALL 'BETA'`, the saved segment named `GAMMA` is loaded and given control. Control is always passed to the start address of the saved segment. If an attempt to load a saved segment fails, normal dynamic load logic is executed, which means that a `LOAD` macro searches for `TEXT` files and for `TXTLIB` and `LOADLIB` members.
`MAXSIZE=nnn`	This parameter specifies the amount of CMS free storage (in KB) to be obtained for Natural; the default setting is `1024`. A standard way of calculating the space needed is to add all buffer sizes (`ESIZE`, etc.) explicitly defined in your `NATPARM` parameter module, plus 40 KB. If you specify `MAXSIZE=0`, each Natural buffer is obtained individually through the CMSSTOR OBTAIN service. Storage use by Natural is then only limited by the amount of free storage available in your virtual machine. See also *Using the MAXSIZE Parameter* in the Natural *Operations* documentation.
`OPID=name`	This parameter specifies the name of the CMS machine to which the specified operator message is sent after a `CALL 'CMWTO'` request has been issued from a Natural program. An asterisk (*) as value causes the message to be sent to the current CMS machine; specifying `OP` directs the message to the VM system operator. The default value is `OP`.

`NATBLDM` also generates a load map whose file name is the same as the file name of the module, with a file type of `MAP`. Keep this map, because it may be helpful in locating errors.

▶ **Step 12: Generate a Natural DCSS**

■ Issue the following CMS command:

```
NATBLDS
```

You are prompted for the name of the Natural DCSS (discontiguous shared segment) that you defined in the corresponding installation step.

Like `NATBLDM` in the corresponding step, `NATBLDS` checks `NATCMS TEXT` and `NATPARM TEXT`, and if necessary, allows you to edit and reassemble the respective source before building the saved segment.

`NATBLDS` also generates a load map whose file name is the same as the name of the DCSS, with a file type of `MAP`. Keep this map, because it may be helpful in locating errors.

Natural then offers to build the "bootstrap" for the DCSS which will load and then pass control to the saved segment. If you enter `YES`, Natural will ask you for the name you want to use to invoke Natural.

You can also build the bootstrap yourself using the following CMS command:

```
XEDIT NATBOOT ASSEMBLE
```

Set the parameter NSS to the name of the Natural DCSS and file it under the name you want to use, for example, `NATvr`.

Issue the following CMS commands:

```
GLOBAL MACLIB NATvrs DMSGPI
ASSEMBLE NATvr
LOAD NATvr (ORIGIN TRANS
GENMOD NATvr
```

▶ **Step 13: Delete Natural System Objects**

■ This step is optional but recommended to avoid data inconsistencies.

If you are using a Version 4.1 Natural `FNAT` system file, you can delete obsolete Version 4.1 Natural objects.

If only base Natural Version 4.1 is installed

Delete the Version 4.1 Natural objects by loading the `NATvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0010)

If Natural Security Version 4.1 is installed

Delete the Version 4.1 Natural Security objects by loading the `NSC`*`vrs`*`.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0099)

See also *Delete Natural Security Objects* in *Installing Natural Security*.

▶ **Step 14: Load the Natural System File and Error Messages**

- If the tape drive used when copying the contents of the installation tape to disk was detached from your virtual machine, ask the system operator to attach a tape drive to your virtual machine at address `X'181'` and mount the Natural installation tape.

  Before you can start the Natural `INPL`, the Adabas environment for your CMS machine must have been set up (as described in the Adabas documentation).

  Issue the following CMS command:

  ```
  NATINPL
  ```

  You are prompted for the name of the command to invoke Natural.

  Enter one of the following:

  - the name of the Natural module built in the corresponding step, or
  - the name of the Natural DCSS bootstrap created in the corresponding step.

  `NATINPL` then positions the tape and loads the Natural system file and error messages. In addition to the English error message short and long texts, error message short texts in German language (`ULANG=2`) are loaded.

  Use the `ERRUPPER` program to convert the error message texts to upper case. This program is provided by the Natural SYSERR utility described in the *Utilities* documentation.

  This is the most time-consuming step of the installation procedure. The elapsed time varies greatly with machine type and system load and can in some cases exceed an hour.

  The Natural system file definitions as coded in the Natural parameter module (which was created in a previous step) apply.

  Three reports are produced on your virtual printer.

  Check these reports to ensure that no errors have occurred.

Load the Japanese Error Messages

This step is only required if you wish to load the error message short texts in Japanese language (`ULANG=59`) or if you wish to replace the English error message long texts with their Japanese equivalents.

The Natural error message short and long texts in Japanese language are both contained in the Natural Japanese Language Pack. This is a separate product (product code NCJ) that can be loaded optionally. If you do not load the Natural error message long texts in Japanese language, English error message long texts will appear. If you do not load the Natural error message short texts in Japanese language, English error message short texts will appear.

Issue the following CMS command:

```
NCJINPL
```

You are prompted for the name of the command to invoke Natural.

Enter one of the following:

- the name of the Natural module built in the corresponding step, or
- the name of the Natural DCSS bootstrap created in the corresponding step.

You are also prompted whether you want to load error message short texts, error message long texts, or both.

`NCJINPL` positions the tape and loads the requested error message texts in Japanese language.

The Natural system file definitions as coded in the Natural parameter module (which was created in a previous step) apply.

After loading each type of error message text, a report is produced on your virtual printer.

Check these reports to ensure that no errors have occurred.

▶ **Step 15: Make Natural Modules Available to the Users**

■ Copy the Natural module and the DCSS bootstrap to a public disk or have your system programmer copy them to the CMS Y-disk.

Installation Verification for Natural under CMS

To verify your installation, perform the following steps:

1. Issue as CMS command the name of the Natural module created in a previous step.

2. Check that the following results are available:

   - Communication between Adabas and Natural is working.

   - The Natural system files have been loaded.

   - Batch Natural is operational.

7 Installing Natural under BS2000/OSD

This document describes step by step how to install Natural under the operating system BS2000/OSD using Adabas system files.

- Prerequisites
- Installation Tape for Natural under BS2000/OSD
- Transferring a License Key File from PC to Host with FTP
- Installation Procedure for Natural under BS2000/OSD
- Installation Verification for Natural under BS2000/OSD

For installation-related information on Unicode and code page support, refer to *Configuration and Administration of the Unicode/Code Page Environment* in the *Unicode and Code Page Support* documentation.

Notation vrs or vr: If used in the following document, the notation vrs or vr stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

- A supported version of the BS2000/OSD operating system must be installed. For the supported versions of the operating systems, refer to *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- A supported version of Adabas must be installed. For the supported versions, refer to *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- As a rule of thumb, each major Software AG product requires approximately 20 MB space in the Adabas database to store the Natural objects supplied by Software AG.

Installation Tape for Natural under BS2000/OSD

The installation tape contains the datasets listed in the table below. The sequence of the datasets, the dataset type, the number of library blocks needed and the space each dataset requires on disk are shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NAT*vrs*.LICS	Natural License Key File. For further information on license key file, license key file installation, product license check and product license check FAQs, see *Licensing Natural*. If a license key file is supplied as an e-mail attachment, see *Transferring the License Key File from PC to Host with FTP*.
NAT*vrs*.SYSF	Empty Natural system file
NAT*vrs*.ERRN	Natural error messages

Dataset Name	Contents
NAT*vrs*.MOD	Natural module library
NAT*vrs*.SRC	Natural source modules
NAT*vrs*.MAC	Natural macros
NAT*vrs*.JOBS	Example installation jobs
NAT*vrs*.LDEL	Instructions to delete Natural system objects of Version 4.1
NAT*vrs*.INPL	Natural system objects
NAT*vrs*.EXPL	Natural example objects

> **Note:** The dataset NAT*vrs*.JOBS will be converted automatically into an LMS-library when it is copied from tape to disk.

Copying the Tape Contents to a BS2000/OSD Disk

If you are not using System Maintenance Aid (SMA), use the procedure described below. In this procedure, the values specified below must be supplied.

To copy the datasets from tape to disk, perform the following steps:

- 1. Copy the Library SRVvrs.LIB from Tape to Disk
- 2. Copy the Procedure COPY.PROC from Tape to Disk
- 3. Copy all Product Files from Tape to Disk

1. Copy the Library SRVvrs.LIB from Tape to Disk

This step is not necessary if you have already copied the library SRV*vrs*.LIB from another Software AG installation tape. For further information, refer to the element #READ-ME in this library. The library SRV*vrs*.LIB is stored on the tape as a sequential file named SRV*vrs*.LIBS containing LMS commands. The current version *vrs* can be obtained from the *Report of Tape Creation*. To convert this sequential file into an LMS-library, execute the following commands:

```
/IMPORT-FILE  SUPPORT=*TAPE(FILE-NAME=SRVvrs.LIBS,       -
/   VOLUME=volser, DEV-TYPE=tape-device)
/ADD-FILE-LINK  LINK-NAME=EDTSAM, FILE-NAME=SRVvrs.LIBS, -
/   SUPPORT=*TAPE(FILE-SEQ=3), ACC-METH=*BY-CAT,          -
/   BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
/START-EDT
@READ  '/'
@SYSTEM 'REMOVE-FILE-LINK  EDTSAM'
@SYSTEM 'EXPORT-FILE  FILE-NAME=SRVvrs.LIBS'
@WRITE  'SRVvrs.LIBS'
@HALT
/ASS-SYSDTA  SRVvrs.LIBS
/MOD-JOB-SW  ON=1
```

```
/START-PROG   $LMS
/MOD-JOB-SW   OFF=1
/ASS-SYSDTA   *PRIMARY
```

where:

tape-device is the device-type of the tape, for example: `TAPE-C4`
volser is the VOLSER of the tape (see *Report of Tape Creation*)

2. Copy the Procedure COPY.PROC from Tape to Disk

To copy the procedure `COPY.PROC` to disk, call the procedure `P.COPYTAPE` in the library `SRVvrs.LIB`:

```
/CALL-PROCEDURE   (SRVvrs.LIB,P.COPYTAPE), -
/   (VSNT=volser, DEVT=tape-device)
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`.

3. Copy all Product Files from Tape to Disk

To copy all Software AG product files from tape to disk, enter the procedure `COPY.PROC`:

```
/ENTER-PROCEDURE   COPY.PROC, DEVT=tape-device
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`. The result of this procedure is written to the file `L.REPORT.SRV`.

Transferring a License Key File from PC to Host with FTP

If a license key file is supplied as an e-mail attachment, you must transfer the attached license key file `natvr.xml` from the PC to the mainframe, using native FTP commands.

> **Caution:** Using utilities instead of native FTP commands for the license key file transfer may corrupt the license key and thus prevent Natural from execution later on. This applies for example to file transfer based on 3270 terminal emulations that do not provide a true binary file transfer, but convert specific characters.

To transfer a license key file from the PC to the mainframe, perform the following steps:

1. Save the product license key file e-mail attachment on your PC hard disk.

2. Open a command prompt window. In the command prompt window, change to the directory where you saved the product license key file.

3. Start an FTP session for communication with the BS2000/OSD host:

```
ftp host-name
```

Where *host-name* is the name of the BS2000/OSD host.

4. Enter your BS2000/OSD host login *User* and *Password*.

5. Enter your BS2000/OSD host *Account*.

6. Specify that the file for the license key file must be written with `FCBTYPE=SAM` and `RECFORM=V`.

```
ftp>Literal file NATvrs.LICS,FCBTYPE=SAM,RECFORM=V
```

7. Switch to binary data mode (the license key file must retain its ASCII format during the transfer):

```
ftp>bin
```

8. Write the license as a file on the mainframe system.

```
ftp>put natvr.xml NATvrs.LICS
```

This command will create a file called `NATvrs.LICS`.

The file set will have `RECFORM=V`, and the license key information stored in the file set will be in ASCII format.

9. Stop your FTP session by entering:

```
ftp>quit
```

Installation Procedure for Natural under BS2000/OSD

Naming Conventions

In the following text, the library name *JOBLIB* stands for

- the example job library (`NATvrs.JOBS`) if you are *not* using SMA or
- the SMA job library (see SMA parameter `JOBLIB` in SMA Parameter Group `BASIC`) if you are using SMA.

> **Note:** Software AG uses the following naming conventions for source elements in the library *JOBLIB*:

A`<product-code><function>` = Assembler sources
L`<product-code><function>` = Instruction for `TSOSLNK`/`BINDER`

Example: `ANATFRNT`, `ANATRENT`, `ANATSTUB` or `LNATFRNT`

▶ **Step 1: Convert the License Key File**

    a   Convert Dataset `NATvrs.LICS` with `PROGRAM NATLICAB` in an Assembler input file.

        (Job I010, Step 0100)

        Convert license key file (`NATvrs.LICS`).

    b   Assemble license key module. This module is linked (Job I060) to the Natural nucleus.

        (Job I010, Step 0101)

        `ANATLIC` (Natural BS2000/OSD module `NATLIC`)

▶ **Step 2: Load the FNAT System File**

    ■   (Job I050, Step 0100)

        If you are installing into an existing Natural 4.1 `FNAT` file, skip this step.

        Load the empty Natural system file (dataset `NATvrs.SYSF`) using the `ADALOD` utility.

        This file will contain all Natural objects supplied by Software AG. Its size depends on the number of products to be installed later. As a rule of thumb, 20 MB can be assumed for each major Software AG product.

        The following `ADALOD` parameters must not be altered:

```
ISNREUSE=YES
```

        To avoid Natural errors NAT9988 and NAT7397 after reorganization of the `FNAT` system file using `ADAULD`/`ADALOD`, the parameter `USERISN` should be set to `YES`.

        The file number *fnat* of the `FNAT` system file can be chosen as described under Natural profile parameter `FNAT` in the Natural *Parameter Reference* documentation.

▶ **Step 3: Load the FUSER System File**

    ■   (Job I050, Step 0101)

        You have the following options:

        1. You can use an existing Version 4.1 `FUSER` file, then you can skip this step.

        2. You can use a new `FUSER` file for Version 4.2.

        3. You can use an existing Version 4.1 `FUSER` file to be shared by Versions 4.1 and 4.2.

4. You can use an existing Version 4.1 `FUSER` file to be used by Version 4.2 only.

Use a New Version 4.2 FUSER System File - First-Time Installation

For the use of a new and empty `FUSER` system file for Natural Version 4.2, no additional system-file-related actions are necessary.

If you do *not* want to share the `FUSER` system file, proceed as follows:

Load the empty Natural user file contained in dataset `NAT`*vrs*`.SYSF` using the `ADALOD` utility.

In this file, all user-written Natural programs are stored.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number *fuser* of the `FUSER` system file can be chosen as described under Natural profile parameter `FUSER` in the Natural *Parameter Reference* documentation.

Reuse an Existing Version 4.1 FUSER System File - Migration from Natural Version 4.1

If you want to use the existing Natural Version 4.1 `FUSER` system file and you do not want to share the `FUSER` system file, skip this step.

Using a Version 4.1 FUSER File to be Shared by Natural Versions 4.1 and 4.2

If you use an existing Natural Version 4.1 `FUSER` system file to be shared by Natural Versions 4.1 and 4.2, you must upgrade your Natural Version 4.1 installation to Version 4.1.4.

Natural Version 4.1.4 Service Pack I003 or a subsequent Service Pack is required. Service Pack I003 and all subsequent Service Packs contain all the necessary Version 4.1 based solutions for Natural Version 4.2.

User Application Programming Interfaces USR* in Library SYSEXT

The `USR*` programs from the delivered library `SYSEXT` run in a special mode. As a result, the `USR*` programs do not need to set further steplibs to execute related objects for processing. This reduces the impact on the Natural buffer pool search logic and improves the performance significantly if user exits are used extensively within user written applications.

It is necessary that the user exits are cataloged with Natural Version 4.2. This implies that the user exits cannot be executed with Natural Version 4.1.

Use of USR* Programs

Usually, the access of USR* programs by an application requires that the user application programming interfaces be copied from library SYSEXT to either the application libraries on the FUSER system file or to library SYSTEM on the FUSER system file or to library SYSTEM on the FNAT system file, respectively, or any other library which is defined as steplib for the application. Library SYSEXT can also be used as steplib. Due to the fact that the delivered user application programming interfaces will always be cataloged with the latest Natural version, we recommend that the user application programming interfaces should reside on the FNAT system file. This will ensure that the right version is executed and will separate user written applications from Software AG modules.

If applications which call user application programming interfaces should run with both Natural Version 4.1 and Natural Version 4.2, it must be made sure that the user application programming interfaces delivered with the corresponding Natural version are used.

The following scenarios may be considered:

Using a Version 4.1 FUSER File for Natural Version 4.1 and 4.2

If the same FUSER system file shall be used in a Natural Version 4.1 and Version 4.2 environment in parallel the following steps are recommended:

- Remove all USR* modules you have copied from library SYSEXT into application libraries on your FUSER system file.

- In both environments, copy the used USR* modules from library SYSEXT to library SYSTEM on the corresponding FNAT system file.

- Alternatively, the USR* modules can be moved to another system library on FNAT which then must be defined as steplib, or library SYSEXT can be used as steplib for the applications. Then automatically in both environments the right versions of the user application programming interfaces are executed.

Using a Version 4.1 FUSER File for Natural Version 4.2 only

If you want to use the existing Natural Version 4.1 FUSER system file and you do not want to share the FUSER system file, then it is still possible to replace all USR* modules you have copied from library SYSEXT into application libraries with the new USR* objects from the Version 4.2 library SYSEXT.

But the preferred way is to remove all user application programming interfaces on the FUSER system file and copy the used user application programming interfaces from library SYSEXT to library system of the FNAT system file or use a SYS library on FNAT as steplib.

Using a New FUSER File for Natural Version 4.2

If you want to port existing applications to a new FUSER system file, copy all application objects but no Software AG USR* objects to the new FUSER system file. Then proceed as described in the scenario above.

Migration

The FIND function of the Natural utility SYSMAIN can be used to search for all USR* modules stored in a specific library on the FUSER system file or across the whole system file. In addition, Predict cross reference data can be used to determine all referenced user application programming interfaces.

The file number *fuser* of the FUSER system file can be chosen as described under Natural profile parameter FNAT in the *Parameter Reference* documentation.

▶ **Step 4: Load the Scratch-Pad File**

■   This step is not supported by SMA.

The scratch-pad file (which is a Natural-internal system file) can be used exclusively by the new Natural version or it can be shared by different versions of Natural.

If you do *not* want to use a scratch-pad file, skip this step.

If you do want to use a scratch-pad file; that is, if you want to use read-only system files (profile parameter ROSY=ON), see also *Natural Scratch-Pad File* in the Natural *Operations* documentation, proceed as follows:

Load the empty scratch-pad file contained in dataset NAT*vrs*.SYSF, using the ADALOD utility as described below.

The following ADALOD parameter *must not* be altered:

```
ISNREUSE=YES
```

For the optional scratch-pad file inclusion, the following NATPARM parameters must be added or, if already present, updated with:

```
LFILE=(212,dbid,fnr)
ROSY=ON
```

▶ **Step 5: Load the FDIC System File**

■   (Job I050, Step 0103)

Skip this step:

- if you want to install Predict (in this case, use the corresponding installation step in the Predict *Installation* documentation), or

- if you want to use an existing `FDIC` system file (an existing `FDIC` system file can be shared by Natural Versions 4.1 and 4.2), or

- if you do not use your own `FDIC` system file.

Load the empty `FDIC` system file contained in dataset `NATvrs.SYSF` using the `ADALOD` utility, as described below.

The following `ADALOD` parameters *must not* be altered:

```
ISNREUSE=YES
```

The file number `fdic` of the `FDIC` system file can be chosen as described under Natural profile parameter `FDIC` in the *Parameter Reference* documentation.

▶ **Step 6: Load the FSEC System File**

- If you use Natural Security, refer to *Installing Natural Security*.

  Skip this step,

  - if you do not use Natural Security, or

  - if you want to use an existing `FSEC` system file, or

  - if you do not want to use an own `FSEC` system file.

▶ **Step 7: Assemble the Natural BS2000/OSD Stub Module and Batch Driver**

- (Job I055, Steps 0216 to 0218)

  Assemble the following source modules:

  - `ANATSTUB` (Natural BS2000/OSD stub module),

  - `ANATFRNT` (Natural BS2000/OSD front-end batch driver)

  - `ANATRENT` (Natural BS2000/OSD reentrant batch driver).

  These source modules are contained in the library `JOBLIB`.

▶ **Step 8: Create the Parameter Module**

- (Job I060, Step 0010)

Create the Natural batch parameter module.

The following parameters in the source parameter module ANATPARM, which is contained in library JOBLIB, must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

For *dbid*, *fnat* and *fuser*, use the values you specified when loading the system files (see the corresponding installation steps).

For all other parameters, you can generally use the default values.

Modify only the values of those parameters whose default values do not suit your requirements.

For the individual parameters contained in the parameter module, refer to the Natural *Parameter Reference* documentation.

Assemble the parameter module.

▶ **Step 9: Link the Natural Nucleus**

■ (Job I060, Steps 3801, 3802)

Link the source modules LNATFRNT (front-end part of batch nucleus) and LNATSHAR (shared part of batch nucleus).

These source modules are contained in the library JOBLIB .

With the INCLUDE instruction for the parameter module, specify the name of the Natural parameter module created in the corresponding step.

Using a Sort Program

If you wish to use a sort program (either Natural's internal one or an external one), include the module NAT2SORT.

It is also possible to place NAT2SORT in a load library from where it can be loaded dynamically at runtime; this requires that NAT2SORT is specified with the profile parameter RCA.

▶ **Step 10: Start All Pools**

■ (Job I061, Step 001)

Start the job E.START.ALL which is contained in the library JOBLIB.

To end all pools, start job `E.END.ALL` which is contained in the library `JOBLIB`.

▶ **Step 11: Delete Natural System Objects**

■ This step is optional but recommended to avoid data inconsistencies.

If you are using a Version 4.1 Natural `FNAT` system file, you can delete obsolete Version 4.1 Natural objects.

If you are using SMA: Set the SMA parameter `NAT-OPT-FNAT41` to `'Y'` (yes).

If only base Natural Version 4.1 is installed

Delete the Version 4.1 Natural objects by loading the `NATvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0010)

If Natural Security Version 4.1 is installed

Delete the Version 4.1 Natural Security objects by loading the `NSCvrs.LDEL` data set with the Natural `INPL` utility.

(Job I061, Step 0099)

See also *Delete Natural Security Objects* in *Installing Natural Security*.

▶ **Step 12: Load the System Programs**

■ (Job I061, Step 0100)

Use the Natural `INPL` utility to load the Natural system objects (the dataset `NATvrs.INPL`) into the Natural system files.

▶ **Step 13: Load the Error Messages**

■ (Job I061, Step 0102)

Load the English Natural error messages file (dataset `NATvrs.ERRN`) using the program `ERRLODUS`. In addition to the English error message short and long texts, error message short texts in German language (`ULANG=2`) are loaded. Use the program `ERRUPPER` to convert the error message texts to upper case. Both programs are described in the Natural *SYSERR Utility* documentation.

▶ **Step 14: Load the Examples**

■ (Job I061, Step 0103)

Use the `INPL` utility to load the Natural example objects (dataset `NAT`*vrs*`.EXPL`) into the Natural system file.

▶ **Step 15: Allocate and Format the Editor Work File**

■ (Job I081, Step 1900)

Allocate the editor work file (see also *Editor Work File* in the Natural *Operations* documentation) by issuing the command:

```
/FILE dataset-name,LINK=CMEDIT,SPACE=...
```

Then execute the formatting program (`NATEDFM2`, `NAT`*vrs*`.MOD`) by entering `FORMAT` to format and load the control record in the editor work file.

Installation Verification for Natural under BS2000/OSD

For base Natural, there are no specific installation verification procedures. After the last step of the installation procedure has been successfully performed, check that the following results are available:

■ Communication between Adabas and Natural is working.

■ The Natural system files have been loaded.

■ Batch Natural is operational.

8 Installing the Natural Com-plete/SMARTS Interface

This document describes how to install the Natural Com-plete/SMARTS Interface, which corresponds to the Natural Com-plete Interface (product code NCF).

- Structure and Functionality of the Natural Com-plete/SMARTS Interface
- Prerequisites
- Installation Tape for the Natural Com-plete/SMARTS Interface
- Installation Procedure for the Natural Com-plete/SMARTS Interface
- Installation for Recoverable Sessions in Com-plete
- Installation of the Natural Server under Com-plete
- Using the Com-plete *ULIB Function
- Installation Verification
- Customizing the Natural Com-plete Environment

> **Note:** SMARTS is an acronym for "Software AG Multi-Architecture Runtime System". It constitutes a runtime layer that allows POSIX-like applications to run on mainframe operating systems. Software AG products communicate with the operating system through the SMARTS layer.

For information on how to operate Natural in a Com-plete environment, refer to:

- *Natural under Com-plete/SMARTS* in the Natural *TP Monitor Interfaces* documentation.

- *Natural under Com-plete/SMARTS Abend Codes* in the Natural *Messages and Codes* documentation.

Notation vrs or vr: If used in the following document, the notation vrs or vr stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Structure and Functionality of the Natural Com-plete/SMARTS Interface

The Natural Com-plete/SMARTS Interface is made up by linking the following modules:

NCFNUC	TP driver interface module.
NCFPARM	Natural Com-plete parameter module.
NCFAM	Natural Com-plete print/work file access method.
NCFROLLS	Natural Com-plete interface module for Natural Roll Server.
NATPARM	Natural parameter module.
PAEAINT	Interface module for SMARTS APS C/ASM-functions. This module resides in the SMARTS APS delivery load library.
TLOPUSER	Interface module for Com-plete functions. This module resides in the Com-plete delivery load library.
ATRRCSS	Interface module for Resource Recovery Services (RRS). This module resides in the SYS1.CSSLIB library.

The resulting module has to be cataloged as `RESIDENT PAGE`. For further information, see *Com-plete System Programmer's Manual* and/or *Com-plete Utility Manual.*

In addition, it is often quite useful to have small startup programs which pass specific dynamic parameters to Natural. An example of such a startup program is created during installation.

Prerequisites

- Base Natural must be installed under **z/OS** or **z/VSE**.
  Version as specified under *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

- Com-plete must be installed.
  Version as specified under *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

Installation Tape for the Natural Com-plete/SMARTS Interface

The following topics are covered below:

- Installation Tape - z/OS Systems
- Installation Tape - z/VSE Systems

Installation Tape - z/OS Systems

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name:	Contents:
`NCFvrs.LOAD`	Com-plete-dependent load modules.
`NCFvrs.SRCE`	Com-plete-dependent source programs and macros.

Copying the Tape Contents to a z/OS Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset `COPY.JOB` from tape to disk.
- Modify this dataset to conform to your local naming conventions.

The JCL in this dataset is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the dataset `COPY.JOB` contains the JCL to unload the datasets for all delivered products from the tape to your disk.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPY.JOB from Tape to Disk
- Step 2 - Modify COPY.JOB on Your Disk
- Step 3 - Submit COPY.JOB

Step 1 - Copy Dataset COPY.JOB from Tape to Disk

The dataset `COPY.JOB` (Label 2) contains the JCL to unload all other existing datasets from tape to disk. To unload `COPY.JOB`, use the following sample JCL:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* -------------------------------
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volume),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=volume,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

hilev is a valid high level qualifier

tape-volume is the tape volume name, for example: T12345
volume is the disk volume name

Step 2 - Modify COPY.JOB on Your Disk

Modify the `COPY.JOB` on your disk to conform to your local naming conventions and set the disk space parameters before submitting this job:

- Set `HILEV` to a valid high level qualifier.
- Set `LOCATION` to a storage location.
- Set `EXPDT` to a valid expiration date.

Step 3 - Submit COPY.JOB

Submit `COPY.JOB` to unload all other datasets from the tape to your disk.

Installation Tape - z/VSE Systems

The installation tape contains the dataset listed below. The sequence of the datasets on tape is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name:	Contents:
`NCFvrs.LIBR`	Com-plete installation libraries.

The notation *vrs* in dataset names represents the version number of the product.

Copying the Tape Contents to a z/VSE Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset `COPYTAPE.JOB` from tape to disk.
- Modify this dataset to confom with your local naming conventions.

The JCL in this member is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the member `COPYTAPE.JOB` contains the JCL to unload the datasets for all delivered products from the tape to your disk, except the datasets that you can directly install from tape, for example, Natural `INPL` objects.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk
- Step 2 - Modify COPYTAPE.JOB
- Step 3 - Submit COPYTAPE.JOB

Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk

The dataset `COPYTAPE.JOB` contains the JCL to unload all other existing datasets from tape to disk. To unload `COPYTAPE.JOB`, use the following sample JCL:

```
* $$ JOB JNM=LIBRCAT,CLASS=0,                                    +
* $$ DISP=D,LDEST=(*,UID),SYSID=1
* $$ LST CLASS=A,DISP=D
// JOB LIBRCAT
* ****************************************
*     CATALOG COPYTAPE.JOB TO LIBRARY
* ****************************************
// ASSGN SYS004,nnn                           <------  tape address
// MTC REW,SYS004
// MTC FSF,SYS004,4
ASSGN SYSIPT,SYS004
// TLBL IJSYSIN,'COPYTAPE.JOB'
// EXEC LIBR,PARM='MSHP; ACC S=lib.sublib'    <------- for catalog
/*
// MTC REW,SYS004

ASSGN SYSIPT,FEC
/*
/&
* $$ EOJ
```

where:

nnn is the tape address
lib.sublib is the library and sublibrary of the catalog

Step 2 - Modify COPYTAPE.JOB

Modify `COPYTAPE.JOB` to conform to your local naming conventions and set the disk space para-meters before submitting this job.

Step 3 - Submit COPYTAPE.JOB

Submit `COPYTAPE.JOB` to unload all other datasets from the tape to your disk.

Installation Procedure for the Natural Com-plete/SMARTS Interface

Example jobs for installing Natural under Com-plete/SMARTS are contained in the job library with prefix NCO (for example, NCOI070).

▶ **Step 1: Create, Assemble and Link NCFPARM**

(Job I070, Steps 2311, 2312)

■ **Customization:** The macro `NFMPRM` contains several parameters which you can modify if their default values do not suit your requirements; these variables are described in the section *Parameters in Macro NFMPRM* in the *Natural TP Monitors* documentation.

▶ **Step 2: Create, Assemble and Link Startup Program**

(Job I070, Steps 2320, 2321)

■ This is an optional step; it should be performed based on site requirements only.

1. Create the source Natural under Com-plete/SMARTS startup program in the source library. Adapt this source to your requirements.
2. Assemble and link the startup program into your Com-plete user program library.

▶ **Step 3: Create Parameter Module**

■ (Job I080, Steps 2300, 2310)

Create the Natural parameter module for Com-plete.

The following parameters in the parameter module must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

For *dbid*, *fnat* and *fuser* use the values you specified when loading the system files; see *Installing Natural under z/OS* or *Installing Natural under z/VSE*.

Local Buffer Pools:

In contrast to previous versions, local buffer pools are allocated during the very first Natural session after Com-plete startup. If you wish to use a *local* buffer pool under Complete, review the `SIZE` subparameter of macro `NTBPI`, which determines the size of the local buffer pool, and for the Natural buffer pool review the `TXTSIZE` parameter, which determines the text segment size of the buffer pool; if necessary, change these parameters in the macro `NTBPI`.

The status of the local buffer pools can be displayed on the operator console by the following command:

```
K SERV,NCFNAT42,BPSTAT
```

Global Buffer Pool (z/OS only):

If you wish to use a Natural *global* buffer pool under Com-plete, specify the same values as in the Natural installation procedure for the profile parameter `SUBSID` in the parameter macro `NTPRM` and for the keyword subparameter `NAME` of the parameter macro `NTBPI`.

For all other parameters, you can generally use the default values. Modify only the values of those parameters whose default values do not suit your requirements.

> **Note:** For a description of the individual parameters contained in the parameter module, see the Natural *Parameter Reference* documentation.

Assemble and link the parameter module.

▶ **Step 4: Link Com-plete/Natural Nucleus - Job I080, Step 2320**

■ With the `INCLUDE` instruction for the Natural parameter module, specify the name of the Natural Com-plete parameter module created in the corresponding step.

1. Include the following modules:

   ```
   NCFNUC
   NCFROLLS
   NCFPARM
   NCFAM
   TLOPUSER
   PAEAINT
   PRM020CO
   ATRRCSS
   ```

2. Link the Com-plete/Natural nucleus to your Com-plete user program library.

3. Add the Natural Com-plete nucleus to the list of `RESIDENTPAGE` programs in your Com-plete `SYSPARM`s.

Non-shared nucleus:

If you do not wish to use a shared Natural nucleus under Com-plete, merge all `INCLUDE` statements and corresponding DD cards from Job I060, Step 0105 (shared nucleus) into Job I080, Step 2320 (front-end).

▶ **Step 5: Adapt Com-plete**

■ This step refers to

　■ **Installation of the Natural Server under Com-plete**,

　■ **Using the Com-plete** `*ULIB` **Function**.

These topics are described below under separate headings.

Installation for Recoverable Sessions in Com-plete

To benefit from Com-plete's recoverable session handling available under z/OS, you have to link the module `NCFROLLS` to the Com-plete/Natural nucleus. `NCFROLLS` serves as an interface to the Natural Roll Server, which has to be started to support recoverable sessions. Furthermore, the module `ATRRCSS` needs not to be linked to the Com-plete/Natural nucleus, because the RRS interface module is part of the Com-plete routine `TLOPUSER`.

For further information on the Natural Roll Server, refer to *Natural Roll Server Functionality* and *Natural Roll Server Operation* in the Natural *Operations* documentation.

Installation of the Natural Server under Com-plete

The Natural server is used to maintain common storage and tables across Natural sessions, e.g. local buffer pools.

To install the Natural server under Com-plete, link module `NCFNAT42` with module `TLOPUSER` from the APS load library, using the following linkage editor commands (see Job I080, Step 2350):

Platform:	Requirement:
| z/OS | ```
MODE RMODE(ANY)
INCLUDE natlib(NCFNAT42)
INCLUDE comlib(TLOPUSER)
ENTRY NCFNAT42
NAME NCFNAT42(R)
```<br><br>where *natlib* is the Natural load library and *comlib* is the Com-plete load library. |
| z/VSE | ```
MODE RMODE(ANY)
PHASE NCFNAT42,*
INCLUDE NCFNAT42
INCLUDE TLOPUSER
ENTRY NCFNAT42
``` |

Define the Natural server for Natural Version 4.2 in the Com-plete startup:

```
SERVER=(NCFNAT42,NCFNAT42)
```

The first parameter defines the name of the server as defined by the parameter `SERVER` of macro `NFMPRM`. The second parameter defines the module name.

The Natural server module is loaded dynamically during Com-plete initialization. The linked module must therefore be placed in a load library contained in the `COMPINIT` concatenation (see also the Com-plete *Installation* documentation).

Using the Com-plete *ULIB Function

If you are running Natural under Com-plete/SMARTS in threads "below" (`NCFPARM THABOVE=NO`), you must catalog the Natural Com-plete/SMARTS Interface using `*ULIB`.

The Natural Com-plete/SMARTS Interface must also be cataloged if Natural work pools below the 16 MB line are desired. The ULIB region size will then depend on the value that was chosen for `WPSIZE` in the Natural *Parameter Reference* documentation.

See also Storage Usage, section *Natural under Com-plete/SMARTS*, in the Natural *TP Monitor Interfaces* documentation.

The region size actually required depends on the buffer sizes specified in the Natural parameter module. To determine the region size actually used, you can use the Natural utility `SYSTP` as described in the Natural *Utilities* documentation.

Installation Verification

Perform the following steps to verify the successful installation of the Natural/Com-plete interface:

1. Stop and restart Com-plete.

2. Enter the Com-plete user menu, type in the name of the Natural Com-plete driver. The Natural initial screen should appear.

3. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

Customizing the Natural Com-plete Environment

To customize your Natural Com-plete environment, you can modify the parameters in the macro `NFMPRM`.

Refer to *Parameters in Macro NFMPRM* in the Natural *TP Monitor Interface* documentation.

9 Installing the Natural CICS Interface

This document describes how to install the Natural CICS Interface (product code NCI).

- Prerequisites
- Installation Tape for the Natural CICS Interface
- Naming Conventions for the Natural CICS Interface
- Installation Procedure for the Natural CICS Interface
- Installation Verification

⊘ **Caution:** Before starting the installation procedure of Natural under CICS, you should have read the section concerning the system control mechanism, see *System Control under CICS* in the Natural *TP Monitor Interfaces* documentation.

See also *Natural under CICS* in the Natural TP Monitor Interfaces documentation for information on the following topics:

- *NCISCPCB Generation Parameters*

- *NCMPRM Macro Parameters*

- *Customization of VSAM RRDS Roll Files*

- *NCISCPRI Warnings and Error Messages*

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

The following software must be installed and running before you install the Natural CICS Interface:

- Base Natural under **z/OS** or **z/VSE**.

- Adabas CICS Interface (product code ACI), version as specified under *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

- CICS/TS or CICS/VSE, version as specified under *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes* for Mainframes.

Installation Tape for the Natural CICS Interface

Installation Tape - z/OS Systems

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
`NCIvrs.LOAD`	CICS-dependent load modules
`NCIvrs.OBJS`	CICS-dependent object modules.
`NCIvrs.SRCE`	CICS-dependent source programs and macros

Copying the Tape Contents to a z/OS Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset `COPY.JOB` from tape to disk.
- Modify this dataset to conform to your local naming conventions.

The JCL in this dataset is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the dataset `COPY.JOB` contains the JCL to unload the datasets for all delivered products from the tape to your disk.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPY.JOB from Tape to Disk
- Step 2 - Modify COPY.JOB on Your Disk

■ Step 3 - Submit COPY.JOB

Step 1 - Copy Dataset COPY.JOB from Tape to Disk

The dataset COPY.JOB (Label 2) contains the JCL to unload all other existing datasets from tape to disk. To unload COPY.JOB, use the following sample JCL:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* -------------------------------
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volume),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=volume,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

hilev is a valid high level qualifier
tape-volume is the tape volume name, for example: T12345
volume is the disk volume name

Step 2 - Modify COPY.JOB on Your Disk

Modify the COPY.JOB on your disk to conform to your local naming conventions and set the disk space parameters before submitting this job:

■ Set HILEV to a valid high level qualifier.

■ Set LOCATION to a storage location.

■ Set EXPDT to a valid expiration date.

Step 3 - Submit COPY.JOB

Submit `COPY.JOB` to unload all other datasets from the tape to your disk.

Installation Tape - z/VSE Systems

The installation tape contains the dataset listed below. The sequence of the datasets on tape is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
`NCIvrs.LIBR`	Natural CICS interface installation libraries

Copying the Tape Contents to a z/VSE Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset `COPYTAPE.JOB` from tape to disk.
- Modify this dataset to confom with your local naming conventions.

The JCL in this member is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the member `COPYTAPE.JOB` contains the JCL to unload the datasets for all delivered products from the tape to your disk, except the datasets that you can directly install from tape, for example, Natural `INPL` objects.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk
- Step 2 - Modify COPYTAPE.JOB

Step 1 - Copy Dataset COPYTAPE.JOB from Tape to Disk

The dataset COPYTAPE.JOB contains the JCL to unload all other existing datasets from tape to disk. To unload COPYTAPE.JOB, use the following sample JCL:

```
* $$ JOB JNM=LIBRCAT,CLASS=0,                                    +
* $$ DISP=D,LDEST=(*,UID),SYSID=1
* $$ LST CLASS=A,DISP=D
// JOB LIBRCAT
* ****************************************
*      CATALOG COPYTAPE.JOB TO LIBRARY
* ****************************************
// ASSGN SYS004,nnn                                 <------  tape address
// MTC REW,SYS004
// MTC FSF,SYS004,4
ASSGN SYSIPT,SYS004
// TLBL IJSYSIN,'COPYTAPE.JOB'
// EXEC LIBR,PARM='MSHP; ACC S=lib.sublib'          <------- for catalog
/*
// MTC REW,SYS004

ASSGN SYSIPT,FEC
/*
/&
* $$ EOJ
```

where:

nnn is the tape address
lib.sublib is the library and sublibrary of the catalog

Step 2 - Modify COPYTAPE.JOB

Modify COPYTAPE.JOB to conform to your local naming conventions and set the disk space parameters before submitting this job.

Step 3 - Submit COPYTAPE.JOB

Submit `COPYTAPE.JOB` to unload all other datasets from the tape to your disk.

Naming Conventions for the Natural CICS Interface

The following naming conventions apply for Natural under CICS:

`ncip`	A common system prefix of 1 to 5 characters; for example, `NCI42`. This prefix is determined by the value of the parameter `PREFIX` in the Natural CICS interface parameter module; see also the corresponding step of the Installation Procedure described below. It is followed by specific characters to make up the names of the following objects:
`ncipCB`	Natural CICS interface system directory; for example, `NCI42CB`.
`ncipR1` to `ncipR9`	Natural CICS interface VSAM RRDS roll files (optional).
`ncipXFA`	Natural CICS interface 3270 Bridge `XFAINTU` exit.

Installation Procedure for the Natural CICS Interface

This section describes the actual installation steps.

When link-editing the Natural nucleus or its subcomponents, you may receive `IEW2646I` or `IEW2660W` messages, which can be ignored.

▶ **Step 1: Allocate the VSAM RRDS Roll Files for Natural (optional)**

■ (Job I008, Step 2200)

This step must only be performed if VSAM roll files are used as CICS roll facility.

For optimum performance, which means without CI/CA splits, the Natural CICS Interface uses VSAM RRDS files for roll files.

▶ **Step 2: Assemble the Roll-File Initialization Module (optional)**

■ (Job I070, Step 2205)

This step must only be performed if VSAM roll files are used as roll facility.

This step creates an executable batch module which is used in the step *Link the VSAM Roll-File Initialization Module* (optional).

Assemble and link/catalog the module `NCISCPRI` for initializing a roll-file.

> **Step 3: Create the Natural CICS Interface Parameter Module**

- (Job I070, Steps 2220, 2225)

  The Natural CICS interface parameter module `NCIPARM` contains a macro named `NCMPRM`, which contains parameters specific to the Natural CICS Interface.

  You can generally use the default values for all parameters. Modify only the values of those parameters whose default values do not suit your requirements. The only mandatory parameter without a default value is the common Natural CICS interface prefix.

  To simplify the Natural CICS interface parameter module installation process, the source module `NCIPARM` contains the `NCMPRM` macro request with parameter `PREFIX=&SYSPARM`. Thus, when generating a Natural CICS interface parameter module, assemble the `NCIPARM` source module with assembler option `SYSPARM=`*prefix* rather than editing the source module.

  The individual parameters are described in the section *NCMPRM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

  Edit, assemble and link/catalog the Natural CICS interface parameter module `NCIPARM`.

> **Step 4: Assemble the Natural CICS Interface Starter Module NCISTART**

- (Job I070, Step 2230)

  Translate, assemble and link/catalog the Natural CICS interface module `NCISTART`.

  Keep in mind that with each installation of a new CICS release, the `NCISTART` module must be reassembled and linked.

  ⚠️ **Important:** If you are not using the most recent CICS version, the translate step may result in a non-zero return code (4 - 16, depending on your CICS version) because of CICS commands being used that are unknown to your CICS translator. This return code can be ignored as long as the subsequent assembly step will end with a return code of 0 by using conditional assemblies to bypass potential CICS translator messages.

  📄 **Notes:**

  1. Under CICS/TS 3.1 and later, the CICS translator option `LEASM` has to be set for LE enablement, see *Natural CICS Interface and IBM Language Environment (LE)*.
  2. The IBM Language Environment macro library must be available in the `SYSLIB` chain in the assembler step.

3. When linking `NCISTART`, the following modules receive an `IEW0461/IEW2454W` error message: `NCIROOT, DFHEAIO, DFHEI1 and some IBM Language Environment modules (CEE...)`. This is normal and is resolved in the final link-edit.

▶ **Step 5: Assemble the Natural CICS Interface Root Module NCIROOT**

■ (Job I070, Step 2235)

Translate, assemble and link/catalog the Natural CICS interface module `NCIROOT`.

Keep in mind that with each installation of a new CICS release, the `NCIROOT` module must be reassembled and linked.

> **Notes:**

1. If you are not using the most recent CICS version, the translate step may result in a non-zero return code because of CICS commands being used that are unknown to your CICS translator. The subsequent assembly step will end with a return code of `0` by using conditional assemblies to bypass potential CICS translator messages.

2. When linking `NCIROOT`, the following modules receive an `IEW0461/IEW2454W` error message: `NCIPARM, NCISERV, DFHEAIO and DFHEI1`. This is normal and is resolved in the final link-edit.

▶ **Step 6: Create the System Directory**

■ (Job I070, Steps 2245, 2250)

The Natural CICS Interface system directory is generated by assembling and linking the source module `NCISCPCB`.

For z/OS, you find a basic example source in dataset `NATvrs.JOBS` and a comprehensive example source in dataset `NCIvrs.SRCE`.

For z/VSE, you find a basic example source in dataset `NATvrs.LIBJ` and a comprehensive example source in dataset `NCIvrs.LIBR`.

For a description of the individual macros and parameters contained in `NCISCPCB`, refer to *NCISCPCB Generation Parameters* in the Natural *TP Monitor Interfaces* documentation.

Edit, assemble and link/catalog module `NCISCPCB`.

▶ **Step 7: Assemble the Natural CICS Interface External CALLNAT Interface Module (optional)**

■ (Job I070, Step 2270)

This step must only be performed if you want to use the Natural CICS interface external `CALLNAT` interface.

Translate, assemble and link/catalog the Natural CICS interface external `CALLNAT` interface module `NCIXCALL`.

If you have an `NCIXCALL` module from a previous Natural version, enter a new name for this old `NCIXCALL` in the `SYSPARM` parameter; for example, `NCIXCIOV`. In Job I080, Step 2270, the old `NCIXCALL` must be relinked with the name you have specified here.

Keep in mind that with each installation of a new CICS release, the `NCIXCALL` module must be reassembled and linked.

> **Notes:**

1. The translate step may result in a non-zero return code because of CICS commands being used that are unknown or no longer applicable to your CICS translator. The subsequent assembly step will end with a return code of 0 by using conditional assemblies to bypass potential CICS translator messages.

2. When linking `NCIXCALL`, the following modules receive an `IEW0461/IEW2454W` error message: `DFHEAIO and DFHEI1`. This is normal and is resolved in the final link-edit.

▶ **Step 8: Assemble the Natural CICS Interface NEP Interface Module (optional)**

■ (Job I070, Step 2275)

This step must only be performed if you want to use the Natural CICS interface node error program (NEP).

Translate, assemble and link/catalog the Natural CICS Interface module `NCIZNEP`.

Keep in mind that with each installation of a new CICS release, the `NCIZNEP` module must be reassembled and linked.

> **Notes:**

1. The translate step may result in a non-zero return code because of CICS commands being used that are unknown or no longer applicable to your CICS translator. The subsequent assembly step will end with a return code of 0 by using conditional assemblies to bypass potential CICS translator messages.

2. When linking `NCIZNEP`, the following modules receive an `IEW0461/IEW2454W` error message: `DFHEAIO and DFHEI1`. This is normal and is resolved in the final link-edit.

▷ **Step 9: Assemble the Natural CICS Interface XFAINTU Exit (optional)**

■ (Job I070, 2280)

This step must be performed only if you want to use Natural via the CICS 3270 Bridge.

Translate, assemble and link/catalog the Natural CICS interface module `NCIXFATU`.

Keep in mind that with each installation of a new CICS release, the `NCIXFATU` module must be reassembled and linked.

📄 **Notes:**

1. The translate step may result in a non-zero return code because of CICS commands being used that are unknown or no longer applicable to your CICS translator. The subsequent assembly step will end with a return code of 0 by using conditional assemblies to bypass potential CICS translator messages.

2. When linking `NCIXFATU`, the following modules receive an `IEW0461/IEW2454W` error message: `DFHEAIO and DFHEI1`. This is normal and is resolved in the final link-edit.

▷ **Step 10: Assemble the Natural CICS Front-End Driver NCISFED (optional)**

■ (Job I070, 2290)

This step must be performed only if you want to use the Natural RPC server front-end under CICS. If you use SMA, this step is only generated if the SMA parameter `NAT-RPC` is set to `Y` (Yes). `NAT-RPC` is used in connection with the SMA parameter `NCI-RPC-FRONT`.

Translate, assemble and link/catalog the Natural CICS front-end driver `NCISFED`.

Keep in mind that, with each installation of a new CICS release, the `NCISFED` module must be reassembled and linked.

📄 **Notes:**

1. The translate step may result in a non-zero return code because of CICS commands being used that are unknown or no longer applicable to your CICS translator. The subsequent assembly step will end with a return code of `0` by using conditional assemblies to bypass potential CICS translator messages.

2. Under CICS/TS 3.1 and later, the CICS translator option `LEASM` has to be set for LE enablement, see *Natural CICS Interface and IBM Language Environment (LE)* in the *TP Monitor Interfaces* documentation.

3. The IBM Language Environment macro library must be available in the `SYSLIB` chain in the assembler step.

4. When linking `NCISFED`, the following modules receive an `IEW0461/IEW2454W` error message: `DFHEAIO and DFHEI1 and some IBM Language Environment modules (CEE...)`. This is normal and is resolved in the final link-edit.

▶ **Step 11: Create the Natural Parameter Module**

■ (Job I080, Steps 2210, 2220)

Create the Natural parameter module for CICS.

The following parameters in the parameter module must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

For *dbid*, *fnat* and *fuser* use the values you specified when loading the system files; see *Installation Procedure for Natural under z/OS* or *Installation Procedure for Natural under z/VSE*.

For all other parameters, you can generally use the default values. Modify only the values of those parameters whose default values do not suit your requirements.

For a description of the individual parameters contained in the parameter module, refer to the Natural *Parameter Reference* documentation.

Edit, assemble and link/catalog the parameter module.

▶ **Step 12: Link the Natural CICS Interface Nucleus**

■ (Job I080, Step 2230)

Link the executable Natural CICS interface nucleus *ncistart* into your CICS user library.

Adapt the `INCLUDE` instruction for the parameter module to the name of the parameter module created in the previous step.

Non-Shared Nucleus:

The following paragraph only applies if you are not using System Maintenance Aid.

If you do not wish to use a shared Natural nucleus under CICS:

Merge all `INCLUDE` instructions and corresponding DD cards from Job I060, Step 0105 (shared nucleus) into Job I080, Step 2230 (front-end).

> **Step 13: Link the Natural System Directory**

- (Job I080, Step 2250)

  Link the Natural system directory into your CICS user library with module name *ncip*CB (see *Naming Conventions for the Natural CICS Interface*).

  The Natural system directory must be linked with option `NORENT` in z/OS.

> **Step 14: Link the VSAM Roll-File Initialization Module (optional)**

- (Job I080, Step 2265)

  This step must only be performed if VSAM roll files are used as CICS roll facility.

  Link the VSAM roll file initialization module into your CICS user library.

> **Step 15: Link the Natural CICS Interface External CALLNAT Interface Module (optional)**

- (Job I080, Steps 2270, 2271)

  This step must only be performed if you want to use the Natural CICS external `CALLNAT` interface.

  Link the Natural CICS external `CALLNAT` interface module `NCIXCALL`.

  Step 2270 is needed if `NCIXCALL` has been installed in the same CICS with a previous Natural version. The previous Natural version `NCIXCALL` must then be relinked with a new name, for example `NCIXCIOV` (see the corresponding step). This way, you can preserve the old version's name `NCIXCALL` for the current version, which avoids that you have to relink all your programs using `NCIXCALL`.

  Step 2271 links the module `NCIXCALL` for the previous Natural version.

> **Step 16: Link the Natural CICS Interface NEP Interface Module (optional)**

- (Job I080, Step 2275)

  This step must only be performed if you want to use the Natural CICS interface node error program. See also *CICS Node Error Program Considerations for Natural* in the *Natural TP Monitor Interface* documentation.

  Link the Natural CICS Interface module `NCIZNEP`.

> **Step 17: Link the Natural CICS Interface XFAINTU Exit (optional)**

- (Job I080, Step 2280)

This step must be performed only if you want to use Natural via the CICS 3270 Bridge. See also *CICS 3270 Bridge Support* in the *Natural TP Monitor Interface* documentation.

Link the CICS Interface module `NCIXFATU` under the name *`ncipXFA`*; see *Naming Conventions for the Natural CICS Interface* in the *Natural TP Monitor Interface* documentation.

▶ **Step 18: Link the Natural CICS RPC Server Front-End (optional)**

■ (Job I080, Step 2290)

This step must be performed only if you want to use the Natural RPC server front-end under CICS. If you use SMA, this step is only generated if the SMA parameter `NAT-RPC` is set to `Y` (Yes). `NAT-RPC` is used in connection with the SMA parameter `NCI-RPC-FRONT`.

If you use SMA: Define the name of the RPC server front-end in the parameter `NCI-RPC-FRONT`.

If you do not use SMA: The name of the RPC server front-end is freely selectable and is specified in the link-edit step.

▶ **Step 19: Initialize the VSAM Roll Files (optional)**

■ (Job I081, Step 2200)

This step must only be performed if VSAM roll files are used as CICS roll facility.

Initialize the VSAM roll files.

This step must be repeated for all roll files used if roll-files are the primary roll-facility.

As a direct (random) access type file, a VSAM RRDS file has to be formatted.

For the Natural CICS interface VSAM roll files, formatting is done by the `NCISCPRI` program. To execute the `NCISCPRI` program, the Natural roll file to be initialized has to be assigned with file name `ROLL` in the JCL `DD` (z/OS) or `DLBL` (z/VSE) statement, respectively. No other parameter input is required for `NCISCPRI`; all data required for file initialization is obtained by `SHOWCB` VSAM macro calls.

> **Note:** When running the Natural CICS interface under z/VSE-type operating systems, an end-of-data (/*) statement must be supplied in the JCL stream for compatibility reasons, although no parameter input is required by the `NCISCPRI` utility.

For a description of the messages that may be output during this step, refer to *NCISCPRI Warnings and Error Messages* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 20: Customize CICS**

■ (Job I005)

Create CICS RDO entries for CICS.

> **Note:** It is generally recommended to put all Natural version-dependent compon-
> ents such as programs, transactions and files in a separate resource group, in the
> following denoted as *natgroup*.

Add the following definitions to your CICS system:

a) Program Definitions

■ A program definition for the executable Natural module, where *ncistart* is the name
of the Natural CICS interface nucleus as specified in the corresponding step:

```
DEFINE PROGRAM (ncistart) GROUP (natgroup) LANGUAGE(ASSEMBLER)
        *
  DESCRIPTION (NATURAL CICS INTERFACE V42 DRIVER COMPONENT)
```

Platform:	Requirement:
All CICS/TS versions	The following parameter value can and should be set in the CICS program definition: DATALOCATION(ANY)
CICS Version 5.3 or above	The following parameter value can and should be set in the CICS program definition: CONCURRENCY(THREADSAFE)

■ A program definition for the Natural CICS interface system directory (mandatory):

```
DEFINE PROGRAM (ncipCB) GROUP (natgroup) LANGUAGE(ASSEMBLER)
        *
  DESCRIPTION (NATURAL CICS INTERFACE V42 SYSTEM DIRECTORY)
```

■ A program definition for the shared Natural nucleus (optional):

```
DEFINE PROGRAM (natshr42) GROUP (natgroup) LANGUAGE(ASSEMBLER)
        *
  DESCRIPTION (NATURAL V42 SHARED NUCLEUS)
```

where `natshr42` is the name of the environment-independent nucleus part - see also *Natural Shared Nucleus* in the Natural *Operations* documentation - as specified with the session parameter NUCNAME. The default name is NATSHR42. Note that specifications of EXECKEY or DATALOCATION for the shared Natural nucleus are irrelevant, as the shared Natural nucleus "inherits" all attributes from the executable Natural module (see NCISTART above) because it is involved by BALR and not by CICS means.

■ To access a shared Natural nucleus in the LPA/ELPA under z/OS, specify USELPACOPY(YES) for this program definition and LPA=YES in the CICS start-up parameters.

■ To access a shared Natural nucleus in the SVA under z/VSE with CICS Version 4.1, specify USESVACOPY(YES) for this program definition and SVA=YES in the CICS start-up parameters.

■ A program definition for the 3GL CALLNAT interface (optional):

```
DEFINE PROGRAM (ncixcall) GROUP (natgroup) LANGUAGE(ASSEMBLER)
      *
  DESCRIPTION (NATURAL CICS INTERFACE V42 EXTERNAL CALLNAT INTERFACE)
```

Platform:	Requirement:
All CICS/TS versions	The following parameter value can and should be set in the CICS program definition: DATALOCATION(ANY)

■ A program definition for the node error program (optional):

```
DEFINE PROGRAM (nciznep) GROUP(natgroup) LANGUAGE(ASSEMBLER)
       *
  EXECKEY(CICS) DESCRIPTION (NATURAL CICS INTERFACE V42 NODE ERROR PROGRAM)
```

■ A program definition for the `XFAINTU` global user exit (optional):

```
DEFINE PROGRAM (ncipXFA) GROUP (natgroup) LANGUAGE(ASSEMBLER)
       *
  EXECKEY(CICS) DESCRIPTION (NATURAL/CICS XFAINTU GLUE)
```

■ A program definition for the Natural CICS RPC server front-end (optional):

```
DEFINE PROGRAM (ncisfe) GROUP (natgroup) LANGUAGE(ASSEMBLER)
  DESCRIPTION (NATURAL CICS V42 RPC SERVER FRONT-END)
```

Platform:	Requirement:
All CICS/TS versions	The following parameter value can and should be set in the CICS program definition: `DATALOCATION(ANY)`

This definition is only required if you want to use the Natural RPC server front-end under CICS. If you use SMA, this definition is only generated if the SMA parameter `NCI-RPC-FRONT` is set.

If you use SMA: The name of the RPC server front-end `ncisfe` is set to the parameter `NCI-RPC-FRONT`.

If you do not use SMA: The name of the RPC server front-end `ncisfe` must be identical to the name specified in Job I080 Step 2290, see the corresponding step.

Important Note for CICS/TS Versions:
Programs being involved by BALR, for example, shared Natural nucleus, Adabas linkage interface or external programs called by Natural using standard linkage conventions (`SET CONTROL 'P=S'`), "inherit" the program attributes (for example, `EXECKEY` or `DATALOCATION`) from the Natural module available (see `NCISTART` above).

b) Transaction Definitions

It is recommended to define or choose a CICS profile for the Natural transactions with the following:

```
DEFINE PROFILE (natprof) GROUP (natgroup) SCRNSIZE (ALTERNATE) INBFMH (ALL)
```

- A transaction definition for the Natural transaction:

```
DEFINE TRANSACTION (ncitransact) GROUP (natgroup)
        *
   PROGRAM (ncistart) TWASIZE(128) DUMP(NO)
        *
   PROFILE (natprof)
```

where:
ncitransact is the name of the Natural CICS interface user transaction code and *ncistart* is the name of the Natural under CICS nucleus as specified in the corresponding step.

- A transaction definition for the Natural internal message switching transaction:

```
DEFINE TRANSACTION (nmsg) GROUP (natgroup)
        *
   PROGRAM (ncistart) TWASIZE(128) DUMP(NO)
        *
   PROFILE (natprof)
```

ncistart being the name of the Natural CICS nucleus as specified in the corresponding step and *nmsg* being the name of the Natural CICS interface message switching transaction code as defined in the MSGTRAN parameter, in the section *Parameters in Macro NCMPRM* in the Natural *TP Monitor Interfaces* documentation. The default name is NMSG.

- The Natural message switching facility requires this transaction to be defined in CICS.

Platform:	Requirement:
All CICS/TS versions	The following parameter values can and should be set in the CICS transaction definitions for Natural: TASKDATALOC(ANY) Setting TASKDATALOC(ANY) may have an impact on non-Natural programs called by Natural, see the *CICS Manual* for details.
All CICS/TS versions (z/OS only)	The following parameter value can be set in the CICS transaction definition: ISOLATE(YES)

For the impact of transaction isolation, see also *THRDSZE - Thread Size* in the *TP Monitor Interfaces* documentation.

■ A transaction definition for the Natural CICS RPC server front-end transaction:

```
DEFINE TRANSACTION (ncisfetransact) GROUP (natgroup)
       *
  PROGRAM (ncisfe) TWASIZE(128) DUMP(NO)
       *
  PROFILE (natprof)
```

where:
ncisfetransact is the name of the Natural CICS RPC server front-end transaction code and
ncisfe is the name of the Natural CICS RPC server front-end as specified in the corresponding step.

c) File Definitions

■ One entry in the FCT for each Natural CICS Interface VSAM roll file (only if VSAM roll files are to be used):

```
 DFHFCT TYPE=FILE,                         *
        FILE=ncipR1,                       *
        ACCMETH=VSAM,                      *
        RECFORM=(FIXED,BLOCKED),           *
        SERVREQ=(UPDATE,DELETE,ADD),       *
        FILSTAT=(ENABLED,OPENED),          *
        BUFND=5,STRNO=3
```

and correspondingly for all other roll files.

Local shared resources (LSR) should be used whenever possible. If multiple LSR pools are supported, one pool should be dedicated exclusively to Natural under CICS roll files.

d) Transient Data Destinations

■ Two or (alternatively) three entries in the DCT for the Natural remote job entry facility `NATRJE` (z/OS only): When submitting a job to JES with the following *two* entries, the internal reader is started on `CLOSE` of the destination:

```
DFHDCT TYPE=SDSCI,DSCNAME=NATRJE,TYPEFLE=OUTPUT
DFHDCT TYPE=EXTRA,DSCNAME=NATRJE,DESTID=nrje,OPEN=DEFERRED
```

where *nrje* is the name of the Natural CICS submit destination as defined in the `RJEDEST` parameter, see the section *NCMPRM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation. The default name is `NRJE`.

■ Additionally, the following DD statement is required in the CICS startup JCL:

```
//NATRJE DD SYSOUT=(*,INTRDR),DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
```

When submitting a job to JES with the following *three* entries, the Natural CICS Interface deals with an indirect destination that will not be closed:

```
DFHDCT TYPE=SDSCI,DSCNAME=NATRJE,TYPEFLE=OUTPUT
DFHDCT TYPE=EXTRA,DSCNAME=NATRJE,DESTID=name,OPEN=INITIAL
DFHDCT TYPE=INDIRECT,DESTID=nrje,INDDEST=name
```

where:

■ *nrje* is the name of the Natural CICS submit destination as defined in the `RJEDEST` parameter, see the section *NCMPRM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation. The default name is `NRJE`.

■ *name* is the name of the corresponding indirect destination.

It is the user's responsibility to either add an `/*EOF` card as the very last card in the job stream or use the corresponding `NATRJE` exit. When detecting the `/*EOF` card, JES submits the previous job stream.

■ Optionally, you can add entries in the DCT for the Natural CICS error message logging facility. For Natural error messages, you can use:

■ A destination that is already defined in CICS (for example, CSSL); then no extra DCT entry is required.

- An extra partition destination as a synonym for an existing CICS message destination:

```
DFHDCT TYPE=INDIRECT,DESTID=nerr,INDDEST=name
```

- An extra file:

```
DFHDCT TYPE=SDSCI,              *
       DSCNAME=NATMSG,          *
       RECFORM=VARUNB,          *
       RECSIZE=nnn,             *
       TYPEFLE=OUTPUT

DFHDCT TYPE=EXTRA,              *
       DSCNAME=NATMSG,          *
       DESTID=nerr,             *
       OPEN=INITIAL
```

where *nerr* is the name of the Natural CICS error message destination as defined in the MSGDEST parameter described in the section *NCMPRM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation. The default name is NERR.

You can change the RECFORM format from VARUNB (variable unblocked) to VARBLK (variable blocked), for example. Natural and the Natural CICS interface messages have a length of up to 120 bytes. Therefore, the record size (RECSIZE= *nnn*) should be at least 124 bytes for variable record format or 120 bytes for fixed record format.

When using a disk file:
Sufficient disk space must be reserved for this dataset; under z/OS, a DD statement, or, under z/VSE, a DLBL statement must be added to the CICS startup JCL.

- Optionally, you can add two entries in the DCT for the Natural CICS interface session statistics:

```
DFHDCT TYPE=SDSCI,              *
       DSCNAME=NATLOG,          *
       RECFORM=VARBLK,          *
       BLKSIZE=4628,            *
       RECSIZE=4624,            *
       DEVICE=DISK (parameter valid for z/VSE only)

DFHDCT TYPE=EXTRA,              *
       TYPEFLE=OUTPUT,          *
       DSCNAME=NATLOG,          *
       DESTID=nlog,             *
       OPEN=INITIAL
```

where *nlog* is the name of the Natural CICS logging destination as defined in the parameter `LOGDEST`, which is described in the Natural *TP Monitor Interfaces* documentation. The default name is `NLOG`.

Sufficient disk space must be reserved for this dataset; under z/OS, a DD statement, or, under z/VSE, a `DLBL` statement must be added to the CICS startup JCL.

▪ Optionally, you can add two entries in the DCT for the Natural CICS profile parameters file:

```
DFHDCT TYPE=SDSCI,                          *
       RECSIZE=80,            card image    *
       BLKSIZE=nnnn,                        *
       BUFNO=1,                             *
       DSCNAME=cmprmin,                     *
       RECFORM=FIXBLK,                      *
       TYPEFLE=INPUT
```

```
DFHDCT TYPE=EXTRA,                          *
       DESTID=nprm,                         *
       DSCNAME=cmprmin,                     *
       OPEN=DEFERRED
```

where *nprm* is the name of the Natural CICS profile parameter input destination as defined in the `PRMDEST` parameter; see *NCMPRM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation. The default name is `NPRM`.

A `DD` statement, or, under z/VSE, a `DLBL` statement must be added to the CICS startup JCL.

e) Other Definitions

z/OS only:

Add the following system abend codes to a CICS System Recovery Table (SRT):

`0D6` To protect CICS against failing Roll Server and/or Authorized Services Manager requests (via PC instructions) by Natural.

`01D` To protect CICS against failing data space cache requests by Natural.

`DC2` To protect CICS against failing memory object cache requests by Natural.

Installation Verification

1. Enter CICS from a user's terminal and type in the Natural transaction code.

2. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

10 Installing the Natural IMS TM Interface

This document describes how to install the Natural IMS TM Interface (product code NII).

- Prerequisites
- Installation Tape for the Natural IMS TM Interface
- Installation Procedure for the Natural IMS TM Interface
- Common Installation Steps
- Installing the Batch Message Processing BMP Environment
- Installing the Message-Oriented NTRD Environment
- Installing the Dialog-Oriented MPP Environment
- Installing the Server Environment
- Customizing the IMS TM Environment
- Installing the Optional Multi-Session Feature
- Installation Verification

For detailed information on using the Natural IMS TM Interface, refer to *Natural under IMS TM* in the Natural *TP Monitor Interfaces* documentation.

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

The following software must be installed and running before you install the Natural IMS TM Interface:

- Base Natural under **z/OS**.

- Natural global buffer pool if you are using the MPP environment (strongly recommended).

- Natural roll server if the ROLLSRV parameter of the Natural IMS TM Interface is set to YES.

- Natural Authorized Services Manager with the SIP Server function, if the Non-Conversational MPP Interface, the monitoring or broadcasting function of the Natural IMS TM Interface is used.

- Authorized Services Manager, if the Accounting to SMF function of the Natural IMS TM Interface is used.

- Adabas IMS TM Interface (product code AII), version as specified under *Natural and Other Software AG Products* in the current Natural *Release Notes for Mainframes*.

For details, see the relevant sections in the *Natural under IMS TM* or Natural *Operations* documentation.

Installation Tape for the Natural IMS TM Interface

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NII*vrs*.LOAD	Load modules of the Natural IMS TM Interface
NII*vrs*.SRCE	Source programs and macros of the Natural IMS TM Interface

Copying the Tape Contents to a z/OS Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset COPY.JOB from tape to disk.
- Modify this dataset to conform to your local naming conventions.

The JCL in this dataset is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the dataset COPY.JOB contains the JCL to unload the datasets for all delivered products from the tape to your disk.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPY.JOB from Tape to Disk
- Step 2 - Modify COPY.JOB on Your Disk
- Step 3 - Submit COPY.JOB

Step 1 - Copy Dataset COPY.JOB from Tape to Disk

The dataset COPY.JOB (Label 2) contains the JCL to unload all other existing datasets from tape to disk. To unload COPY.JOB, use the following sample JCL:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* ------------------------------
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volume),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=volume,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

hilev is a valid high level qualifier
tape-volume is the tape volume name, for example: T12345
volume is the disk volume name

Step 2 - Modify COPY.JOB on Your Disk

Modify the COPY.JOB on your disk to conform to your local naming conventions and set the disk space parameters before submitting this job:

- Set HILEV to a valid high level qualifier.

- Set LOCATION to a storage location.

- Set EXPDT to a valid expiration date.

Step 3 - Submit COPY.JOB

Submit `COPY.JOB` to unload all other datasets from the tape to your disk.

Sample Jobs

The sample jobs are contained in the dataset `NATvrs.JOBS` and are prefixed with `NII`.

Installation Procedure for the Natural IMS TM Interface

The installation procedure comprises the following:

1. **Common Installation Steps**
2. **Installing the Batch Message Processing BMP Environment**
3. **Installing the Message-Oriented NTRD Environment**
4. **Installing the Dialog-Oriented MPP Environment**
5. **Installing the Server Environment**
6. **Customizing the IMS TM Environment**

Perform the steps in the sequence indicated above.

Common Installation Steps

The following steps are required to install the Natural IMS TM Interface in a BMP, an NTRD an MPP and/or a server environment:

▶ **Step 1: Create and Assemble the Transaction Code Table**

■ (Job I055, Steps 2555 and 2556)

1. Create the transaction code table by including a `NIMTRNTG` macro for each transaction code used for Natural transactions.

   For further information on the parameters in the `NIMTRNTG` macro, refer to *NIMTRNTG Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

   If you want to use Natural in non-message-driven BMP or a batch environment, add a `NIMTRNTG` macro for the PSB used with an arbitrary transaction code.

2. Assemble and link the transaction code table.

▶ **Step 2: Create and Assemble the Parameter Module**

■ (Job I055, Steps 2580 and 2581)

1. Create the parameter module by including a `NIMPARM` macro for each environment needed.

   For information on the parameters for the `NIMPARM` macro, refer to *NIMPARM Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

2. Assemble and link the parameter module.

▶ **Step 3: Link the Interface Module**

■ (Job I055, Step 2582)

Link the interface module.

The same interface module can be used in a BMP, an NTRD, and MPP and/or a server environment.

The name of the interface module must be specified with the parameter `NIINAME` in the call to macro `NIMDRIV` for all environment interfaces described in the following sections.

Installing the Batch Message Processing BMP Environment

The following steps are required to install the Natural IMS TM Interface for the BMP environment:

▶ **Step 1: Create and Assemble the BMP Interface**

■ (Job I055, Steps 2583 and 2584)

1. Create the source `NIIBMP` which contains a call to macro `NIMDRIV` with the parameter `TYPE` set to `BMP`.

   For further information on the macro `NIMDRIV`, refer to the *NIMDRIV Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

2. Assemble and link the BMP interface.

   For `CMPRMTB` you receive the warning `IEW0461`. You can ignore this.

If the `NIMDRIV` parameter `LE370` is not set to `NO`, you receive the warning `IEW0461` for modules starting with `CEE`. You can ignore this.

▶ **Step 2: Link the BMP Front-End**

■   (Job I070, Step 2510)

The front-end consists of the BMP interface created in **Step 1** and your batch Natural parameter module `NATPARM`.

1. Specify the name of the Natural batch parameter module with the `INCLUDE` instruction in the parameter module (Job I060, Steps 0010, 0015).

2. Specify the name of the front-end module used for this link.

   🛑 **Caution:** This name must also be specified in your BMP region job as application program name (parameter `MBR` of the `IMSBATCH` procedure invocation).

3. Link the front-end for the BMP environment.

Installing the Message-Oriented NTRD Environment

The following steps are required to install the Natural IMS TM Interface for the NTRD environment:

> **Step 1: Create and Assemble the NTRD Interface**

■ (Job I056, Steps 2500, 2501)

1. Create the source `NIINTRD` which contains a call to macro `NIMDRIV` with the parameter `TYPE` set to `NTRD`.

    For further information on the macro `NIMDRIV`, refer to the *NIMDRIV Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

2. Assemble and link the NTRD interface.

    For `CMPRMTB` you receive the warning `IEW0461`. You can ignore this.
    If the `NIMDRIV` parameter `LE370` is not set to `NO`, you receive the warning `IEW0461` for modules starting with `CEE`. You can ignore this.
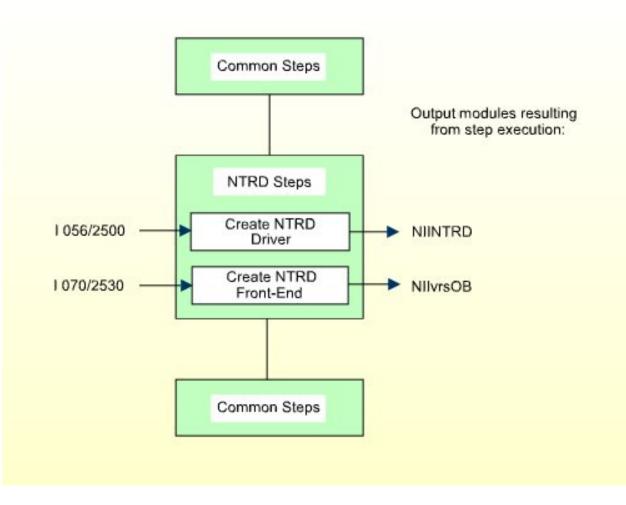
▶ **Step 2: Link the NTRD Front-End**

■ (Job I070, Step 2530)

The front-end consists of the NTRD interface created in **Step 1** and of your batch Natural parameter module `NATPARM`.

1. Specify the name of the Natural batch parameter module with the `INCLUDE` instruction in the parameter module (Job I060, Steps 0010, 0015).

2. Specify the name of the front-end module used for this link.

   Caution: This name must also be specified in the `APPLCNT` macro as application program name (parameter `PSB`), if the NTRD front-end is invoked directly by transaction code and not by bootstrap module.

3. Link the front-end for the NTRD environment.

Installing the Dialog-Oriented MPP Environment

The following steps are required to install the Natural IMS TM Interface for the MPP environment:

▶ **Step 1: Create and Assemble the Online Natural Parameter Module**

■ (Job I080, Steps 2500, 2510)

1. Set the values of the following parameters in the parameter module:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

where *dbid*, *fnat*, and *fuser* are the values you specified when loading the system files in your base Natural installation. For further information, see *Installing Natural Under z/OS*.

2. To use a *global* Natural buffer pool, specify the macro `NTBPI` in the parameter module with the name of the global Natural buffer pool and set the profile parameter `SUBSID` in the profile parameter module.

   If you want to use any other buffer pool, specify the macro `NTBPI` in the parameter module for each required buffer pool type.

   It is strongly recommended that you use a global buffer pool for each buffer pool type.

   If an Editor buffer pool is required, you must use a global Editor buffer pool.

3. Modify any other parameters in the parameter module whose default values do not meet your requirements. For further information on the parameters contained in the parameter module, refer to *Using a Natural Parameter Module* in the Natural *Operations* documentation.

4. Assemble and link the Natural parameter module for the dialog-oriented environments.

▶ **Step 2: Create and Assemble the Conversational MPP Interface**

■ (Job I070, Steps 2565,2566)

1. Create the source `NIICONV` which contains a call to macro `NIMDRIV` with the parameter `TYPE` set to `CONV`.

   For further information on the macro `NIMDRIV`, refer to the *NIMDRIV Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

2. Assemble and link the Conversational MPP interface.

   For `CMPRMTB` you receive the warning `IEW0461`. You can ignore this.
   If the `NIMDRIV` parameter `LE370` is not set to `NO`, you receive the warning `IEW0461` for modules starting with `CEE`. You can ignore this.
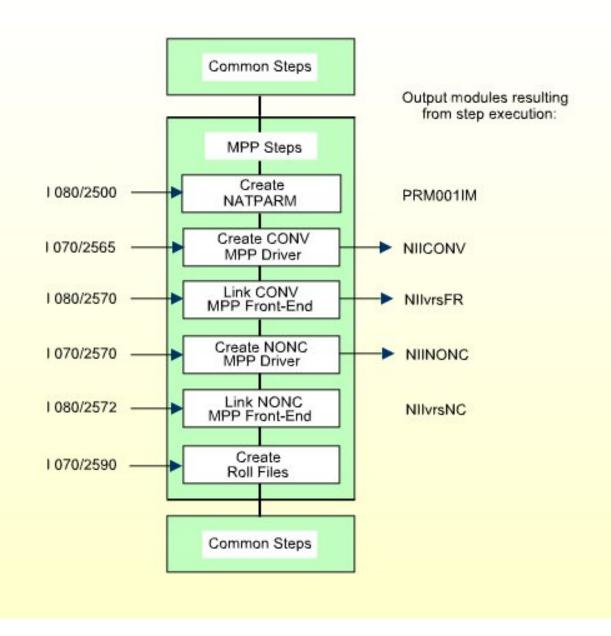
▶ **Step 3: Link the Conversational MPP Front-End**

■ (Job I080, Step 2570)

The front-end consists of the Conversational MPP interface created in **Step 2** and the online Natural parameter module `NATPARM` created in **Step 1**.

1. Specify the name of the online Natural parameter module in the `INCLUDE` instruction for the parameter module.

2. Specify the name of the front-end module used for this link.

   ⛔ **Caution:** This name must also be specified in the `APPLCNT` macro as application program name (parameter `PSB`).

3. Link the front-end for the Conversational MPP environment.

▶ **Step 4: Create and Assemble the Non-Conversational MPP Interface**

■ (Job I070, Steps 2570, 2571)

   1. Create the source `NIINONC` which contains a call to macro `NIMDRIV` with the parameter `TYPE` set to `NONC`.

      For further information on the macro `NIMDRIV`, refer to *NIMDRIV Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

   2. Assemble and link the Non-Conversational MPP interface.

      For `CMPRMTB` you receive the warning `IEW0461`. You can ignore this.
      If the `NIMDRIV` parameter `LE370` is not set to `NO`, you receive the warning `IEW0461` for modules starting with `CEE`. You can ignore this.

▶ **Step 5: Link the Non-Conversational MPP Front-End**

■ (Job I080, Step 2572)

   The front-end consists of the Non-Conversational MPP interface created in **Step 4** and the online Natural parameter module `NATPARM` created in **Step 1**.

   1. Specify the name of the online Natural parameter module in the `INCLUDE` instruction for the parameter module.

   2. Specify the name of the front-end module used for this link.

      ⛔ **Caution:** This name must also be specified in the `APPLCNT` macro as application program name (parameter `PSB`).

   3. Link the front-end for the Non-Conversational MPP environment.

▶ **Step 6: Allocate and Format the Roll Files**

- (Job I070, Step 2590)

  This step is only required if you do not use the Natural roll server.

  If you do not want to use the Natural roll server, you have to allocate and format the roll files to be used by the Natural IMS TM Interface.

  You can allocate up to 5 sequential datasets with a fixed-record format for use as roll files.

  1. Allocate the roll files.

  2. Format the roll files using the module `NATRSRFI`.

  The roll file initialization program produces a WTO message indicating the number of concurrent users which can be serviced by the roll file. For information on the roll file facility, refer to *Natural Roll Server Functionality* in the Natural *Operations* documentation.

Installing the Server Environment

The following steps are required to install the Natural IMS TM Interface for the sever environment:

▶ **Step 1: Create and Assemblethe Server Interface**

■ (Job I056, Steps 2530, 2531)

1. Create the source `NIISRVD` which contains a call to macro `NIMDRIV` with the parameter `TYPE` set to `SRVD`.

   For further information on the macro `NIMDRIV`, refer to *NIMDRIV Macro Parameters* in the Natural *TP Monitor Interfaces* documentation.

2. Assemble and link the server interface.

For `CMPRMTB` you receive the warning `IEW0461`. You can ignore this.
If the `NIMDRIV` parameter `LE370` is not set to `NO`, you receive the warning `IEW0461` for modules starting with `CEE`. You can ignore this.

▶ **Step 2: Link the Server Front-End**

■ (Job I070, Step 2550)

The front-end consists of the server interface created in **Step 1** and of your batch Natural parameter module `NATPARM`.

1. Specify the name of the Natural batch parameter module with the `INCLUDE` instruction in the parameter module (Job I060, Steps 0010, 0015).

2. Specify the name of the front-end module used for this link.

⚠ **Caution:** This name must also be specified in the `NIMBOOT` macro as driver name (parameter `DRIVERN`).

3. Link the front-end for the server environment.

▶ **Step 3: Create and Assemble the Bootstrap Module NIIBOOTS**

■ (Job I056, Steps 2540, 2541)

1. Create the source `NIIBOOTS` which contains a call to macro `NIMBOOT` with the parameter `SERVER` set to `YES`. For the `DRIVERN` parameter, specify the name of the front-end module created in **Step 2**.

2. Assemble and link the bootstrap module.

Customizing the IMS TM Environment

The following steps require system modifications to your IMS TM environment.

▶ **Step 1: Create the APPLCTN Table Definitions for MPP, BMP and NTRD**

■ **MPP Define Sample:**

```
APPLCTN PSB=NIIvrsFR,PGMTYPE=TP
    TRANSACT CODE=NATvrs,MODE=SNGL,SPA=512,
        MSGTYPE=(SNLGSEG,RESPONSE,9)
```

🛑 **Caution:** The size of the SPA must be set to at least 157 bytes plus the `NRASTART` value.

BMP Define Sample (Message-Driven or NAF-specific BMP):

```
APPLCTN PSB=NIIvrsBM,PGMTYPE=BATCH
    TRANSACT CODE=NATvrsBM,MODE=SNGL,
        MSGTYPE=(SNLGSEG,RESPONSE,9)
```

This `APPLCTN` definition is required if you use the `CMGETMSG` feature.

BMP Define Sample (without Message Queue Processing):

```
APPLCTN PSB=NIIvrsBM,PGMTYPE=BATCH
```

NTRD Define Sample:

```
APPLCTN PSP=NIIvrsOB,PGMTYPE=TP
    TRANSACT CODE=NATvrsOB,MODE=SNGL,
        MSGTYPE=(MULTSEG,NONRESPONSE,10)
```

▶ **Step 2: Create the PSB/ACB for both the MPP and BMP**

■ **Example for MPP:**

PSB for conversational Natural:

```
PCB TYPE=TP,MODIFY=YES
PCB TYPE=TP,MODIFY=YES
PCB TYPE=TP,MODIFY=YES
PCB TYPE=DB,DBNAME=dliddm,PROCOPT=A,KEYLEN=16 sample for NDL
SENSEG NAME=EMPLOY,PROCOPT=A sample for NDL
SENSEG NAME=VEHICL,PROCOPT=A,PARENT=EMPLOY sample for NDL

PSBGEN PSBNAME=NIIvrsFR,LANG=ASSEM,MAXQ=3,IOASIZE=132
```

At least one modifiable TP-PCB must be defined for default use of hardcopy, sending messages and transaction switching. To avoid a Natural initialization error, the value of the WRKPCBS parameter in the current environment table must be less than or equal to the number of PCBs minus 1.

Example for BMP:

```
PCB TYPE=TP,MODIFY=YES
PCB TYPE=TP,MODIFY=YES
PCB TYPE=DB,DBNAME=dliddm,PROCOPT=A,KEYLEN=16 sample for NDL
SENSEG NAME=EMPLOY,PROCOPT=A sample for NDL
SENSEG NAME=VEHICL,PROCOPT=A,PARENT=EMPLOY sample for NDL

PSBGEN PSBNAME=NIIvrsBM,LANG=ASSEM,MAXQ=3,IOASIZE=132
```

At least one modifiable TP-PCB must be defined for default use of hardcopy and sending messages. To avoid a Natural initialization error, the value of the WRKPCBS parameter in the current environment table must be less than or equal to the number of PCBs minus 1.

After creating the required APPLCTNs for the BMP and MPP environments, you must generate the PSB, DBD and ACB.

After the ACB generation, the following commands activate the new definitions:

```
/MODIFY PREP ACBLIB
/MODIFY COMMIT
```

▶ **Step 3: Create the BMP and MPP Regions**

■ Use the sample members as guidelines when creating the specific regions for your install-
  ation.

```
BMPJOB
MPPJOB
```

▶ **Step 4: Create the PRELOAD List**

■ Update the PRELOAD list using a PRELOAD member DFSMPL *xx* with the following module
  names:

  ▪ the Natural nucleus name,

  ▪ the interface module name,

  ▪ the front-end name,

  ▪ the Adabas link module name

  Example for MPP:

```
NATvrsSH,NIIvrsIF,NIIvrsFR,ADALNI
```

  Example for BMP:

```
NATvrsSH,NIIvrsIF,NIIvrsBM,ADALNK
```

  If alias names are used for any members in the PRELOAD list, these names should be added
  to the PRELOAD list as well. Failure to do so leads to performance degradation.

  Special Considerations

  ▪ Set the REGION parameter to at least 2MB.

  ▪ Include the load libraries used by the Natural IMS TM Interface.

- Include the `DD` statement for the roll file created in Job I070, Step 2590:

```
//ROLLFn    DD  DSN=....DISP=SHR
```

where *n* is a value from 1 - 5.

- Include the `DD` statement for `NATRJE`:

```
//NIIRJEDD   DD  SYSOUT=(X,INTRDR)
```

Installing the Optional Multi-Session Feature

The multi-session feature is an optional feature of the Natural IMS TM Interface. For details, see the relevant section in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 1: Create the Multi-Session Database**

■ The following steps have no corresponding example jobs in `NATvrs.JOBS`.

1. Create the DBD using the member `NIIMSDBD` in `NIIvrs.SRCE`.

2. Create the PSB for the initial load.

3. Add the DBD to all PSBs intended for use with the multi-session feature.

4. Define and load the database using the JCL `INITDM` in `NATvrs.JOBS`.

▶ **Step 2: Adapt the Parameter Module of the Natural IMS TM Interface**

■

Parameter	Description
`MSACTV=YES`	Activates the session manager
`MSDBD=dbdname`	Where *dbdname* is the name used in `MSDBD`.
`MSCRKEY`	The key to create a new session
`MSRSKEY`	The key to switch to a resumed session
`MSMAX=nn`	Where *nn* is the number of active sessions (max. 8)

For further information, see the sections *Installing the Batch Message Processing BMP Environment* and *Installing the Dialog-Oriented MPP Environment*.

▶ **Step 3: Adapt the Transaction Code Table**

■ Set the following parameter:

Parameter	Description
MSPCB	Number of the multi-session PCB

Installation Verification

1. From an IMS TM session, start a BMP with the following IMS TM command:

```
/STA REG BMPJOB
```

2. Check the output. The output results from the Natural system command TECH. Verify the output with your environment.

3. From an IMS TM session, issue the following IMS TM commands:

```
/STA REG MPPJOB
/STA TRAN NATvrs
/STA PROG NIIvrsFR
```

4. Type in transaction name NATvrs.

5. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

11 Installing the Natural TSO Interface

This document describes how to install the Natural TSO Interface (product code NTI).

The Natural TSO Interface (`NATTSO`) consists of a number of service routines interfacing with the z/OS operating system. `NATTSO` is supplied as a source module and can be customized to meet your requirements.

You have two options:

■ assemble and link it to the Natural nucleus, or

■ run it separately, connecting with a shared nucleus.

`NATTSO` is fully reentrant and can run above the 16 MB line.

This document describes step by step how to install Natural in a TSO environment.

- ■ Prerequisites
- ■ Installation Tape for the Natural TSO Interface
- ■ Installation Procedure for the Natural TSO Interface
- ■ Installation Verification

For information on the following topics, refer to *Natural under TSO* in the Natural *TP Monitor Interfaces* documentation:

■ General Information about the Natural TSO Interface

■ Natural TSO Datasets

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

Base Natural must be installed under z/OS.

See **Installing Natural under z/OS**.

Installation Tape for the Natural TSO Interface

The installation tape contains the dataset listed in the table below. The position of the dataset is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
`NTIvrs.SRCE`	TSO-dependent source programs

Copying the Tape Contents to a z/OS Disk

If you are using SMA, refer to the *System Maintenance Aid* documentation (included in the current edition of the Natural documentation CD).

If you are *not* using SMA, follow the instructions below.

This section explains how to:

- Copy dataset `COPY.JOB` from tape to disk.
- Modify this dataset to conform to your local naming conventions.

The JCL in this dataset is then used to copy all datasets from tape to disk.

If the datasets for more than one product are delivered on the tape, the dataset `COPY.JOB` contains the JCL to unload the datasets for all delivered products from the tape to your disk.

After that, you will have to perform the individual install procedure for each component.

- Step 1 - Copy Dataset COPY.JOB from Tape to Disk
- Step 2 - Modify COPY.JOB on Your Disk
- Step 3 - Submit COPY.JOB

Step 1 - Copy Dataset COPY.JOB from Tape to Disk

The dataset `COPY.JOB` (Label 2) contains the JCL to unload all other existing datasets from tape to disk. To unload `COPY.JOB`, use the following sample JCL:

```
//SAGTAPE JOB SAG,CLASS=1,MSGCLASS=X
//* -------------------------------
//COPY EXEC PGM=IEBGENER
//SYSUT1 DD DSN=COPY.JOB,
// DISP=(OLD,PASS),
// UNIT=(CASS,,DEFER),
// VOL=(,RETAIN,SER=tape-volume),
// LABEL=(2,SL)
//SYSUT2 DD DSN=hilev.COPY.JOB,
// DISP=(NEW,CATLG,DELETE),
// UNIT=3390,VOL=SER=volume,
// SPACE=(TRK,(1,1),RLSE),
// DCB=*.SYSUT1
//SYSPRINT DD SYSOUT=*
//SYSIN DD DUMMY
//
```

where:

hilev is a valid high level qualifier
tape-volume is the tape volume name, for example: T12345
volume is the disk volume name

Step 2 - Modify COPY.JOB on Your Disk

Modify the COPY.JOB on your disk to conform to your local naming conventions and set the disk space parameters before submitting this job:

- Set HILEV to a valid high level qualifier.

- Set LOCATION to a storage location.

- Set EXPDT to a valid expiration date.

Step 3 - Submit COPY.JOB

Submit COPY.JOB to unload all other datasets from the tape to your disk.

Installation Procedure for the Natural TSO Interface

▶ **Step 1: Create CLIST**

■ (Job I070, Step 2400)

Create a source Natural TSO CLIST.

This CLIST is used later to invoke Natural under TSO. Please note that this is only a basic example which you can adapt to your requirements and to your TSO environment.

▶ **Step 2: Create ADARUN Cards**

■ (Job I070, Step 2410)

Create source `ADARUN` cards.

This source member is required by the CLIST created in Step 1.

▶ **Step 3: Assemble the Natural TSO Interface Module**

■ (Job I070, Step 2415,Step 2420)

The source program `NATTSO` contains a call to the macro `NTTSO` which generates the Natural TSO interface.

- Job I070, Step 2415 does the `IEBUPDTE`.
- Job I070, Step 2420 assembles and links `NATTSO`.

Set the parameters in the source of the module `NATTSO` to fit your requirements.

For a description of the Natural TSO generation parameters contained in the macro `NTTSO`, refer to *Natural TSO Interface Generation Parameters* in the Natural *TP Monitor Interfaces* documentation.

Assemble and link the Natural TSO interface module `NATTSO` contained in the dataset `NTIvrs.SRCE`.

▶ **Step 4: Create and Assemble Parameter Module**

■ (Job I080, Step 0010, Step 0015)

Create the Natural parameter module for TSO.

The following parameters in the parameter module must be modified for the installation:

```
FNAT=(dbid,fnat)
FUSER=(dbid,fuser)
```

For *dbid*, *fnat* and *fuser* use the values you specified when loading the system files; see *Installation Procedure for Natural under z/OS*

Global Buffer Pool: If you wish to use a *global* buffer pool, specify the macro `NTBPI` in all your Natural parameter modules.

For all other parameters: You can generally use the default values.

Modify only the values of those parameters whose default values do not suit your requirements.

For a description of the individual parameters contained in the parameter module, refer to the Natural Parameter Reference documentation.

For dynamic assignment of profile parameters, see also the `CMPRMIN` dataset described in the section *Natural in Batch Mode* in the Natural *Operations* documentation.

Assemble and link the parameter module.

▶ **Step 5: Link Natural TSO Nucleus**

■ (Job I080, Step 0020)

With the `INCLUDE` instruction for the parameter module, specify the name of the Natural TSO parameter module you created before.

> **Non-Shared Nucleus:** If you do not wish to use a shared Natural nucleus under TSO, merge all `INCLUDE` statements and corresponding DD cards from Job I060, Step 0105 (shared nucleus) into Job I080, Step 0020 (front-end).

Link the executable Natural TSO nucleus.

▶ **Step 6: Make Adabas Interface Available**

■ You have two options:

■ Include the Adabas load library in the steplib of your TSO user procedures.

■ Or copy the modules listed in the section referring to installation with TSO of the appropriate Adabas installation manual to a library of your TSO user steplib.

Installation Verification

Perform the following steps to verify the successful installation of the Natural TSO Interface:

1. Enter TSO, `READY` mode.
2. Invoke the CLIST you created at the beginning of this Installation Procedure.

Example:

```
EX 'SAGLIB.SMA.SRCE(NATvr)'
```

3. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

12 Installing the Natural UTM Interface

This document describes how to install the Natural UTM Interface (product code NUT).

- Prerequisites
- Installation Tape for the Natural UTM Interface
- Installation Procedure for the Natural UTM Interface
- Installation Verification

For information on how to operate the Natural UTM Interface, refer to *Natural under UTM* in the Natural *TP Monitor Interfaces* documentation.

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

The following software must be installed and running before you install the Natural UTM Interface:

- Base Natural under BS2000/OSD. See *Installing Natural Under BS2000/OSD*.

- Adabas UTM Interface (product code AUT), version as specified under *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

Installation Tape for the Natural UTM Interface

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NUT*vrs*.MAC	Macros necessary for Natural/UTM
NUT*vrs*.SRC	Sources necessary for Natural/UTM
NUT*vrs*.JOBS	Example job library for Natural/UTM

Copying the Tape Contents to a BS2000/OSD Disk

If you are not using System Maintenance Aid (SMA), use the procedure described below. In this procedure, the values specified below must be supplied.

To copy the datasets from tape to disk, perform the following steps:

- 1. Copy the Library SRVvrs.LIB from Tape to Disk
- 2. Copy the Procedure COPY.PROC from Tape to Disk
- 3. Copy all Product Files from Tape to Disk

1. Copy the Library SRVvrs.LIB from Tape to Disk

This step is not necessary if you have already copied the library SRV*vrs*.LIB from another Software AG installation tape. For further information, refer to the element #READ-ME in this library. The library SRV*vrs*.LIB is stored on the tape as a sequential file named SRV*vrs*.LIBS containing LMS commands. The current version *vrs* can be obtained from the *Report of Tape Creation*. To convert this sequential file into an LMS-library, execute the following commands:

```
/IMPORT-FILE  SUPPORT=*TAPE(FILE-NAME=SRVvrs.LIBS,      -
/   VOLUME=volser, DEV-TYPE=tape-device)
/ADD-FILE-LINK  LINK-NAME=EDTSAM, FILE-NAME=SRVvrs.LIBS, -
/   SUPPORT=*TAPE(FILE-SEQ=3), ACC-METH=*BY-CAT,              -
/   BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
/START-EDT
@READ  '/'
@SYSTEM 'REMOVE-FILE-LINK  EDTSAM'
@SYSTEM 'EXPORT-FILE  FILE-NAME=SRVvrs.LIBS'
@WRITE  'SRVvrs.LIBS'
@HALT
/ASS-SYSDTA  SRVvrs.LIBS
/MOD-JOB-SW  ON=1
/START-PROG  $LMS
/MOD-JOB-SW  OFF=1
/ASS-SYSDTA  *PRIMARY
```

where:

tape-device is the device-type of the tape, for example: TAPE-C4
volser is the VOLSER of the tape (see *Report of Tape Creation*)

2. Copy the Procedure COPY.PROC from Tape to Disk

To copy the procedure `COPY.PROC` to disk, call the procedure `P.COPYTAPE` in the library `SRVvrs.LIB`:

```
/CALL-PROCEDURE  (SRVvrs.LIB,P.COPYTAPE), -
/   (VSNT=volser, DEVT=tape-device)
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`.

3. Copy all Product Files from Tape to Disk

To copy all Software AG product files from tape to disk, enter the procedure `COPY.PROC`:

```
/ENTER-PROCEDURE  COPY.PROC, DEVT=tape-device
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`. The result of this procedure is written to the file `L.REPORT.SRV`.

Installation Procedure for the Natural UTM Interface

Naming Conventions

In the following text, the library name *JOBLIB* stands for

- the example job library (`NUTvrs.JOBS`) if you are *not* using SMA or

- the SMA job library (see SMA parameter `JOBLIB` in SMA Parameter Group `BASIC`) if you are using SMA.

> **Note:** Software AG uses the following naming conventions for source elements in the library *JOBLIB*:

A`<product-code><function>` = Assembler sources
L`<product-code><function>` = Instruction for `TSOSLNK/BINDER`

Example: `ANATFRNT, ANATRENT, ANATSTUB` or `LNATFRNT`

▶ **Step 1: Generate KDCROOT Source**

■   (Job I070, Step 0110)

Prepare the sample `KDCDEF` instruction source `NUT.KDCDEF` (LMS element type = D) in the library *JOBLIB* so that it suits your environment.

Therefore, refer to *Defining the UTM Resources (KDCDEF)* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 2: Assemble the Natural UTM BS2STUB Module**

■ (Job I070, Step 0111)

Assemble source module `ANUTSTUB` in the library `JOBLIB`.

See also *Local Common Memory Pools* in the Natural *Operations* documentation.

▶ **Step 3: Assemble the Natural UTM Front-End Part**

■ (Job I070, Step 0112)

Assemble source module `ANUTFRNT` in the library `JOBLIB`.

See also *Keyword Parameters of Macro NATUTM* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 4: Assemble the Natural UTM Reentrant Part**

■ (Job I070, Step 0113)

Assemble source module `ANUTRENT` in the library `JOBLIB`.

See also *Keyword Parameters of Macro NURENT* in the Natural *TP Monitor Interfaces* documentation).

▶ **Step 5: Generate the Swap Pool Parameter Module NTSWPRM**

■ (Job I070, Step 0114)

Assemble and link module `ANUTSWP` in the library `JOBLIB`.

See also *Keyword Parameters of Macro NTSWPRM* in the section *Defining the Swap Pool* in the Natural *Operations* documentation.

▶ **Step 6: Assemble KDCROOT**

■ (Job I070, Step 0120)

Assemble the sample source `ANUTKDCR` (generated under Step 1) in the library `JOBLIB` for the `KDCROOT` module.

See also *Generating KDCROOT* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 7: Generate KDCFILE**

■ (Job I075, Step 0100)

Prepare the sample `KDCDEF` instruction source `NUT.KDCDEF` (LMS element type = D) in the library `JOBLIB` so that it suits your environment.

Therefore, refer to *Defining the UTM Resources (KDCDEF)* in the Natural *TP Monitor Interfaces* documentation.

▶ **Step 8: Assemble the Natural UTM Parameter Module**

■ (Job I080, Step 0200)

1. In source module `ANUTPARM`, modify the Natural profile parameters so that they suit your environment, but at least have a look at the following parameters:

   ```
   FNAT=(dbid,fnat)
   FUSER=(dbid,fuser)
   ```

2. Assemble source module `ANUTPARM` in the library *JOBLIB*.

▶ **Step 9: Link the Natural UTM Front-End Part**

■ (Job I080, Step 0210)

Use the `INCLUDE` statements for `TSOSLNK` contained in `LNUTFRNT` in the library *JOBLIB*.

Installation Verification

1. Submit job `E.START.SWP` in the library *JOBLIB* to start the Natural Swap Pool.

2. Submit job `E.START.NUT` in the library *JOBLIB*, which is a sample job for starting Natural under UTM.

3. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

13 Installing the Natural TIAM Interface

This document describes how to install the Natural TIAM Interface (product code NRT).

- Prerequisites
- Installation Tape for the Natural TIAM Interface
- Installation Procedure for the Natural TIAM Interface
- Installation Verification

For detailed information on the following topics, refer to *Natural under TIAM* in the Natural *TP Monitor Interfaces* documentation.

- *Structure of the Natural TIAM Interface*

- *Parameters in Macro NAMTIAM*

- *Common Memory Pools under TIAM*

- *Natural Shared Nucleus*

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

Base Natural must be installed under BS2000/OSD.

See **Installing Natural under BS2000/OSD**.

Installation Tape for the Natural TIAM Interface

The installation tape contains the dataset listed below. The sequence of the datasets on tape is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NRT*vrs*.MAC	Macros necessary for Natural/TIAM
NRT*vrs*.JOBS	Example job library for Natural/TIAM

Copying the Tape Contents to a BS2000/OSD Disk

If you are not using System Maintenance Aid (SMA), use the procedure described below. In this procedure, the values specified below must be supplied.

To copy the datasets from tape to disk, perform the following steps:

- 1. Copy the Library SRVvrs.LIB from Tape to Disk
- 2. Copy the Procedure COPY.PROC from Tape to Disk
- 3. Copy all Product Files from Tape to Disk

1. Copy the Library SRVvrs.LIB from Tape to Disk

This step is not necessary if you have already copied the library `SRVvrs.LIB` from another Software AG installation tape. For further information, refer to the element `#READ-ME` in this library. The library `SRVvrs.LIB` is stored on the tape as a sequential file named `SRVvrs.LIBS` containing LMS commands. The current version *vrs* can be obtained from the *Report of Tape Creation*. To convert this sequential file into an LMS-library, execute the following commands:

```
/IMPORT-FILE  SUPPORT=*TAPE(FILE-NAME=SRVvrs.LIBS,      -
/   VOLUME=volser, DEV-TYPE=tape-device)
/ADD-FILE-LINK  LINK-NAME=EDTSAM, FILE-NAME=SRVvrs.LIBS, -
/   SUPPORT=*TAPE(FILE-SEQ=3), ACC-METH=*BY-CAT,              -
/   BUF-LEN=*BY-CAT, REC-FORM=*BY-CAT, REC-SIZE=*BY-CAT
/START-EDT
@READ  '/'
@SYSTEM 'REMOVE-FILE-LINK  EDTSAM'
@SYSTEM 'EXPORT-FILE  FILE-NAME=SRVvrs.LIBS'
@WRITE  'SRVvrs.LIBS'
@HALT
/ASS-SYSDTA  SRVvrs.LIBS
/MOD-JOB-SW  ON=1
/START-PROG  $LMS
/MOD-JOB-SW  OFF=1
/ASS-SYSDTA  *PRIMARY
```

where:

tape-device is the device-type of the tape, for example: `TAPE-C4`
volser is the VOLSER of the tape (see *Report of Tape Creation*)

2. Copy the Procedure COPY.PROC from Tape to Disk

To copy the procedure `COPY.PROC` to disk, call the procedure `P.COPYTAPE` in the library `SRVvrs.LIB`:

```
/CALL-PROCEDURE  (SRVvrs.LIB,P.COPYTAPE), -
/   (VSNT=volser, DEVT=tape-device)
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`.

3. Copy all Product Files from Tape to Disk

To copy all Software AG product files from tape to disk, enter the procedure `COPY.PROC`:

```
/ENTER-PROCEDURE  COPY.PROC, DEVT=tape-device
```

If you use a TAPE-C4 device, you may omit the parameter `DEVT`. The result of this procedure is written to the file `L.REPORT.SRV`.

Installation Procedure for the Natural TIAM Interface

Naming Conventions

In the following text, the library name *JOBLIB* stands for

- the example job library (`NRTvrs.JOBS`) if you are *not* using SMA or
- the SMA job library (see SMA parameter `JOBLIB` in SMA Parameter Group `BASIC`) if you are using SMA.

> **Note:** Software AG uses the following naming conventions for source elements in the library *JOBLIB*:

A`<product-code><function>` = Assembler sources
L`<product-code><function>` = Instruction for `TSOSLNK/BINDER`

Example: `ANATFRNT, ANATRENT, ANATSTUB` or `LNATFRNT`

▶ **Step 1: Assemble the Natural/TIAM Non-Reentrant Front-End Part**

■ (Job I070, Step 0102)

The front-end part of Natural/TIAM is assembled by generating the macro `NAMTIAM`. You can generally use the default values of the parameters in macro `NAMTIAM`.

Modify only the values of those parameters whose default values do not suit your requirements.

For a description of the individual parameters, see *Parameters in Macro NAMTIAM* in the Natural *TP Monitor Interfaces* documentation.

Assemble source module `ANRTFRNT` in the library *JOBLIB* to generate macro `NAMTIAM` for the front-end part.

▶ **Step 2: Assemble the Natural/TIAM Reentrant Part**

■ (Job I070, Step 0103)

The reentrant part of Natural/TIAM is assembled by generating the macro `NAMTIAM`. You can generally use the default values of the parameters in macro `NAMTIAM`.

Modify only the values of those parameters whose default values do not suit your requirements.

For a description of the individual parameters, see *Parameters in Macro NAMTIAM* in the Natural *TP Monitor Interfaces* documentation.

Assemble source module `ANRTRENT` in the library *JOBLIB* to generate macro `NAMTIAM`.

▶ **Step 3: Assemble the Natural/TIAM Parameter Module**

■ (Job I080, Step 0109)

Assemble source module `ANRTPARM` in the library *JOBLIB*.

▶ **Step 4: Link the Natural/TIAM Front-End Part**

■ (Job I080, Step 0110)

Use the `INCLUDE` statements for `TSOSLNK` contained in `LNRTFRNT` in the library *JOBLIB*.

> **Note:** The source module `ANATSTUB` has been assembled during the installation of Natural for BS2000/OSD.

Installation Verification

1. Call procedure `P.STARTNRT` in library *JOBLIB* to start Natural under TIAM.

2. Proceed with the steps described in the section *Installation Verification for the TP Monitor Interface*.

14 Installing the Entire System Server Interface

Installing the Entire System Server Interface is required if the product Entire System Server or Natural ISPF is to be used.

You may choose between standard installation (recommended) and customized installation.

- Prerequisites
- Standard Installation
- Customized Installation
- Assembling the Parameter Modules for the Entire System Server Interface Component
- Linking the ESXNUC Module
- Running Entire System Server in Single-User Mode under z/OS

Notation *vrs* or *vr*:

If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

A supported version of either of the following products must be installed before you can install the Entire System Server Interface:

- Entire System Server or Natural ISPF, version as specified under *Natural and Other Software AG Products* in the current *Natural Release Notes for Mainframes*.

Standard Installation

If you want to use the recommend standard value settings in the modules `ESYNODTB` and `NATPNIP` (used by the Entire System Server and Natural ISPF), proceed with *Linking the ESXNUC Module*.

Customized Installation

If you do *not* want to use the recommend standard value settings, first edit the module `NATPNIP` in the Natural source library and then relink the `ESXNUC` module.

The content of module `NATPNIP` is:

```
NAMVIEWP BUFLEN=8192,NUMREQ=5,MAXCBL=3000,MAXEDL=3000,EXTUSER=INIT-USER
```

The default values are included. The `NATPNIP` parameters are explained below:

`BUFLEN`	Length of all Adabas buffers in bytes.
`NUMREQ`	Number of possible nested `FIND` loops in Natural calling Entire System Server.
`MAXCBL`	Complex `FIND` buffer length.
`MAXEDL`	Editor session buffer length.
`EXTUSER`	External user ID which will be passed to Entire System Server for security checks.

Possible values for `EXTUSER` are:

`EXTUSER=INIT-USER`	Recommended for z/OS platforms. The value of the Natural variable `*INIT-USER` is transferred via Entire System Server to an external security system, for example, RACF, ACF2, TOP-SECRET, and all calls to security restricted resources are handled under this user ID.
`EXTUSER=USER`	Recommended for UTM on BS2000/OSD. Similar processing using Natural variable `*USER`.
`EXTUSER=ADDRESS-SPACE`	Recommended for TIAM and batch mode on BS2000/OSD. The security description of this address space is used for security evaluation.

You may also edit the module `ESYNODTB` in the Natural source library which contains the following default values:

```
NAMXNOD ID=148,NAME=PRODUCTION-1
NAMXNOD ID=149,NAME=PRODUCTION-2,LAST=Y
END
```

When do I need to change these values?

Normally, calls to Entire System Server from Natural are handled via a `NODE` parameter, for example:

```
FIND ACTIVE-JOBS WITH JOB-NAME = 'ADA*' AND NODE = 148
```

where `NODE` contains a numeric value, but no character-oriented values.

Programmers need to know a numeric value pointing to a real Entire System Server. Using the value `NODE-NAME`, they may code:

```
FIND ACTIVE-JOBS WITH JOB-NAME = 'ADA*' AND NODE-NAME = 'DAEF'
```

where `DAEF` is the logical name of the production Entire System Server.

The table `ESYNODTB` defines these mappings and contains the following parameters:

`ID`	Entire System Server node number (also known as DBID).
`NAME`	Entire System Server node name.
`LAST`	Indicator for last entry in table.

Assembling the Parameter Modules for the Entire System Server Interface Component

With Natural ISPF

This section applies if Natural ISPF is used as INCORE database.

The jobs mentioned below can be found in the data set `ISPvrs.JOBS`.

Under z/OS and z/VSE

Link the parameter module `NATPNIP`, using Job I055, Step 1106. In this case, the module `ESYNODTB` is not required.

Under BS2000/OSD

Link the parameter module `NATPNIU` (for UTM) or `NATPNIT` (for TIAM/batch mode), using Job I055, Steps 1106 and 1108. In this case, the module `ESYNODTB` is not required.

With Entire System Server

This section applies if the Entire System Server is used.

The jobs mentioned below can be found in the data set `NPRvrs.JOBS`.

Under z/OS and z/VSE

Assemble and link the modules using Job I055, Step 1106 (`NATPNIP`) and, optionally, 1107 (`ESYNODTB`).

Under BS2000/OSD - UTM Environments

Assemble and link the modules using Job I055, Step 1108 (`ANATPNIU`) and, optionally, 1107 (`AESYNDTB`).

`ANATPNIU` must be assembled with `NAMVIEWP EXTUSER=USER` in order to use the Natural Security user ID as user ID for Entire System Server calls (if no `NATPROC-LOGON` is issued).

Under BS2000/OSD - TIAM and Batch Environments

Assemble and link the modules using Job I055, Step 1106 (`ANATPNIT`) and, optionally, 1107 (`AESYNDTB`).

`NATPNIT` must be assembled with `NAMVIEWP EXTUSER=ADDRESS-SPACE`. The `LOGON` user ID will be used as Entire System Server user ID if no `NATPROC-LOGON` has been issued.

Linking the ESXNUC Module

Under z/OS

You can either link all modules to the shared nucleus or link them all to the front-end.

Include the following instructions in your link job:

`INCLUDE NATLIB(NATPNIP)`	Entire System Server Interface parameters
`INCLUDE NATLIB(ESXNUC)`	Entire System Server Interface nucleus
`INCLUDE NATLIB(ESYNODTB)`	Optional, node table

Optionally:

If you want to run Entire System Server in single user mode, proceed as described in the section *Running Entire System Server in Single-User Mode* below.

Note that you may do so only under TSO or in batch mode.

Under z/VSE

You can either link all modules to the shared nucleus or link them all to the front-end.

Include the following instructions in your link job:

`INCLUDE NATPNIP`	Entire System Server Interface parameters.
`INCLUDE ESXNUC`	Entire System Server Interface nucleus.
`INCLUDE ESYNODTB`	Optional, node table.

Optionally:

If you want to run Entire System Server in single user mode, you may do so only in batch mode.

`INCLUDE NATLIB(NATPSNGL)` to the front-end and link it now as `NON-REENTRANT`.

This allows you to use Entire System Server in single-user mode. This module *must not* be linked to the shared nucleus.

Under BS2000/OSD

All modules have to be linked to the front-end part of Natural to handle different configuration settings.

Include the following instructions in your link job:

`INCLUDE NATPNIP, NATvrs.MOD`	Entire System Server Interface parameters.
`INCLUDE ESXNUC, NATvrs.MOD`	Entire System Server Interface nucleus.
`INCLUDE NATPRBSU, NATvrs.MOD`	BS2000/OSD service module.
`INCLUDE NATPSNGL, NATvrs.MOD`	Optional, allow use of Entire System Server in single-user mode (only possible under TIAM or in batch mode).
`INCLUDE ESYNODTB, NATvrs.MOD`	Optional, node table.

Under VM/CMS

Modify the variable `LOADLIST` in `NAT$LOAD EXEC` to include the following modules:

Module	Explanation
`NATPNIP`	Entire System Server Interface parameters.
`ESXNUC`	Entire System Server Interface nucleus.
`ESYNODTB`	Optional, node table.

Proceed as described in the VM/CMS installation procedure under the heading *Generate a Natural Module*.

Running Entire System Server in Single-User Mode under z/OS

- General
- Installing ESY for Single-User Mode

General

Single-user ESY under z/OS is only available in batch mode and under TSO.

Running the Entire System Server (ESY) in single-user mode is advantageous, for example, in the following cases:

- Executing long running batch jobs comprising a large number of call to the Entire System Server.

- Performing test scenarios using a Natural session under TSO performing many calls to the Entire System Server, without disturbing the production environment.

- Exploring new Entire System Server functionality or versions.

From a Natural point of view, the ESY single-user mode is accessible as ESY node 148, irrespective of whether such a node does already exist on your machine or network.

The following is an example of a Natural program that is running in a single-user environment:

```
FIND ACTIVE-JOBS WITH NODE = 148 AND JOB-NAME = 'XCOM*'
```

This statement calls a single-user Entire System Server that runs within the same address-space. Calling a different ESY node that runs elsewhere in the network is possible by using a different node number, as shown in the following example:

```
FIND ACTIVE-JOBS WITH NODE = 53 AND JOB-NAME = 'NUC*'
```

This statement calls a multi-user ESY with node number 53 out of the same Natural program which called the single-user ESY session.

Installing ESY for Single-User Mode

Single-user ESY is only available in batch mode and in a TSO environment.

The following topics are covered below:

- Installing for Batch Mode
- Installing for TSO

Installing for Batch Mode

Perform the following steps:

1. Create a new PDS load library, which must be APF authorized.

2. Copy all members of the ESY load library into the new load library.

3. Link-edit the Module `NATPSNGL`, which handles ESY single user features, with the standard batch Natural nucleus front-end. Note that this module *must not* be linked with a shared Natural nucleus.

   Alternatively, you may also use the `RCA` parameter to load this module by setting `RCA=NATPSNGL`.

   In both cases, the Natural front-end must be link-edited, authorized by parameter `AC=1`, into the new load library.

4. To handle the ESY-relevant steps, add the mandatory cards `PARMS` and `SYSPRINT` to your JCL. In addition, you may add the optional cards `ESYTRACE` and `CLOG`.

```
//ESYTRACE DD  SYSOUT=*       <== internal trace
//SYSPRINT DD  SYSOUT=*       <== modules/zap directory of NPR

//CLOG     DD  DISP=SHR,DSN=xxx   <== Command log dataset
//PARMS    DD  DISP=SHR,DSN=xxx   <== NPR Parameter
```

5. Add an Adabas load library to your JCL, which also has to be APF authorized.

6. Edit your ESY parameter member. Here, the parameter `NODE` will be ignored, since Natural routes any calls to node number 148 through to the single-user ESY node.

Installing for TSO

Perform the following steps:

1. Ask your system programmer to add the name of the Natural nucleus as `AUTHPGM` in the TSO definition member named `IKJTSO00`. Usually, this member is in dataset `SYS1.PARMLIB`.

2. Proceed analogously as described in the batch mode installation above.

15 Installing Natural Security

This document describes how to install Natural Security (product code NSC) on a mainframe computer under all mainframe operating systems and TP monitors supported by Natural using an Adabas database.

- Prerequisites
- Installation Tape - z/OS and z/VSE
- Installation Tape - BS2000/OSD and VM/CMS
- Installation Procedure
- Installation Verification

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

The following software must be installed and running before you can install Natural Security:

- Natural for batch mode
- TP monitor(s)

You are recommended to install Natural Security *after* all other subproducts of Natural, as this makes defining the subproducts' system libraries to Natural Security easier.

🛑 **Caution:** If you are using parallel versions of Natural Security it is strongly recommended that you always use the highest version to perform data maintenance.

For further information, see *Application Programming Interfaces* in the *Natural Security* documentation.

Additional Prerequisites for Natural Security in a Heterogeneous Environment

In addition to the prerequisites described above, the following software must be installed and running in order to use Natural Security in a heterogeneous environment:

- Entire Net-Work *
- Natural Security for Mainframes (z/OS, z/VSE, BS2000/OSD and VM/CMS) *

* Version as specified under *Natural and Other Software AG Products* in the current Natural *Release Notes* for Mainframes.

The following software must be installed as required:

- Natural Security for UNIX
- Natural Security for Windows

- Natural Security for OpenVMS

For further information, see *Using Natural Security on Multiple Platforms* in the *Natural Security* documentation.

Prerequisite for the Natural Development Server Part

The following is required:

- INPL of the current Natural Security version.
- Enable SYSSEC to maintain Natural Development Server specific profiles.
- Add basic application profiles to the FSEC system file which are prerequisites for maintaining Natural Development Server specific profiles.

See also Application Protection, Prerequisites in the Natural Security documentation.

Installation Tape - z/OS and z/VSE

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NSC*vrs*.SYSL	Natural Security log file
NSC*vrs*.INPL	Natural Security system objects
NSC*vrs*.LDEL	Instructions to delete Natural Security system objects of Version 4.1
NSC*vrs*.VINI	Natural Security FDIC initialization file for VSAM system files

For instructions on copying the datasets from the installation tape to a disk, see the following sections: *Copying the Tape Contents to a z/OS Disk* and *Copying the Tape Contents to a z/VSE Disk*.

Installation Tape - BS2000/OSD and VM/CMS

The installation tape contains the datasets listed in the table below. The sequence of the datasets is shown in the *Report of Tape Creation* which accompanies the installation tape.

Dataset Name	Contents
NSC*vrs*.SYSL	Natural Security log file
NSC*vrs*.INPL	Natural Security system objects
NSC*vrs*.LDEL	Instructions to delete Natural Security system objects of Version 4.1

For instructions on copying the datasets from the installation tape to a disk, see the following sections: *Copying the Tape Contents to a BS2000/OSD Disk* and *Copying the Tape Contents to a VM/CMS Disk*.

Installation Procedure

It is recommended that you install Natural Security *after* all other subproducts of Natural, as this makes defining the subproducts' system libraries to Natural Security easier.
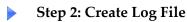
► **Step 1: Create System File**

- ■ (Job I050, Step 9900)

    Only perform this step if you wish to use a new FSEC system file for Natural Security Version 4.2. If you wish to use an existing Version 4.1 FSEC system file, omit this step.

    If you wish to have the FSEC as VSAM system file, please refer to the *Natural for VSAM* documentation for information on how to install the FSEC system file and details concerning restrictions on the use of the FSEC VSAM system file.

    Load the Natural Security system file (FSEC) using the Adabas utility ADALOD (ADALOD is described in the *Adabas Utilities* documentation). Input for ADALOD is the dataset NAT*vrs*.SYSF. The file number you specify for the FSEC system file must be as yet unused.

    This step creates an empty system file for Natural Security.

► **Step 2: Create Log File**

- ■ (Job I050, Step 9901)

This step must only be performed if the Natural Security function "Logging of Maintenance Functions" is to be used. Otherwise, omit this step.

Load the log file using the Adabas utility ADALOD (ADALOD is described in the *Adabas Utilities Manual*). Input for ADALOD is the dataset NSC*vrs*.SYSL.

This step creates a log file to be used by the above-mentioned function.

▶ **Step 3: Adjust Natural Parameter Modules**

■ (Jobs I060, I080)

Add the following profile parameter to all your Natural parameter modules:

```
FSEC=(,file-number)
```

where `file-number` represents the number of the Natural Security system file - either the new one loaded in Step 1 or the existing Version 4.1 one. (If required, you can also specify a database ID, password and cipher code with the FSEC profile parameter; refer to *FSEC - Natural Security System File* in the Natural *Parameter Reference* documentation.)

Repeat jobs I060 and I080 for all your TP monitors.

▶ **Step 4: Delete Natural Security Objects**

■ (Job I061, Step 0099)

This step is optional but recommended to avoid data inconsistencies.

If you are using a Version 4.1 Natural FNAT system file, delete obsolete Version 4.1 Natural Security objects by loading the NSC*vrs*.LDEL data set with the Natural INPL utility.

See also the corresponding step *Delete Natural System Objects* in the following sections:

Installation Procedure for Natural under z/OS
Installation Procedure for Natural under z/VSE
Installation Procedure for Natural under BS2000/OSD
Installation Procedure for Natural under CMS

▶ **Step 5: Load Natural Security Modules**

■ (Job I061, Step 9900)

Once this step has been performed, it is not possible to remove Natural Security from the Natural system file; to remove Natural Security from the system file again, you would have to delete the entire contents of the system file and re-install all Natural components again.

Load the Natural Security modules using the Natural system command INPL described in the in the Natural *System Commands* documentation.

This step loads the Natural Security modules into your Natural system file (FNAT) under the library ID "SYSSEC" and the logon-processing programs under the library ID "SYSLIB".

This step also results in the creation of the following security profiles and relationships:

- A library security profile with library ID "SYSSEC". The library is people-protected ("People-protected" set to "Y" and "Terminal-protected" set to "N").

- A user security profile with user ID "DBA", user type ADMINISTRATOR, and password set to "DBA".

- User "DBA" is linked to library "SYSSEC" (ordinary link, no special link).

If INPL is executed again for Natural Security, the Natural Security modules will be newly loaded without affecting any objects and relationships already defined after the initial execution of INPL.

If you execute INPL with the option "RECOVER", the user "DBA", the library "SYSSEC", and the link between the two will be redefined as after the initial installation, while all other links to "SYSSEC" will be cancelled.

▶ **Step 6: Change Password of User "DBA"**

■ Invoke Natural.

On the Natural Security logon screen, type in library ID "SYSSEC", user ID "DBA", password "DBA", and a new password, and press ENTER.

Type in the new password again and press ENTER to confirm the password change.

▶ **Step 7: Define Administrators**

■ This step must only be performed if Version 4.2 is your first version of Natural Security; that is, if you have not used any previous version of Natural Security. Otherwise, omit this step.

Create a user security profile for each person who is to be a Natural Security administrator; and then link each Natural Security administrator to the library SYSSEC. The following is an *example* of how to do this.

1. On the logon screen, type in library ID "SYSSEC", user ID "DBA" and the password as established in Step 5.

2. The Natural Security Main Menu will be displayed. On this, enter code "M".

3. A window will be displayed. In this window, mark object type "User" with a character or with the cursor.

4. The User Maintenance selection list will be displayed. In the command line of the User Maintenance selection list, enter the command "ADD".

5. A window will be displayed. Choose a user ID for your Natural Security administrator (for example, if the administrator's name were Arthur Dent, you may choose "AD" as his user ID; the following steps will take this as an example). In the window, type in user ID "AD" and user type "A".

6. The Add User screen will be displayed. Enter the user name "Arthur Dent" and set Private Library to "N" (and press ENTER).

7. Press PF3. User Arthur Dent is now defined to Natural Security under the user ID "AD".

8. The User Maintenance selection list will be displayed again. In the "Co" column of the selection list, mark user "AD" with function code "LL".

9. A window will be displayed. In the window, enter library ID "SYSSEC".

10. The Link User To Libraries selection list will be displayed. In the "Co" column of the selection list, mark library "SYSSEC" with function code "LK". User Arthur Dent is now linked to library SYSSEC.

11. In the command line, enter the direct command "LOGOFF". The Natural Security logon screen will be displayed.

Now you can log on to SYSSEC with user ID "AD" and password "AD". When you log on with the new user ID for the first time, you must change the password (by typing in a new password in addition to the user ID and password).

Once you have successfully defined administrators, it may be advisable to delete user DBA to make sure that the user ID "DBA" cannot be used by unauthorized users to gain access to SYSSEC.

To delete user DBA:

■ Log on to SYSSEC with user ID "AD".

■ Go to the User Maintenance selection list as described above.

■ On the list, mark user "DBA" with function code "DE". A window will be displayed, in which you enter and confirm user ID "DBA".

■ The user DBA is now deleted.

▶ **Step 8: Define System Libraries**

■ Only perform this step if Version 4.2 is your first version of Natural Security; that is, if you have not used any previous version of Natural Security. Otherwise, omit this step.

Create security profiles for all system libraries of Natural and Natural subproducts installed at your site; see also *Adding a New Library* in the *Natural Security* documentation. Refer to the installation instructions for other Software AG products for the corresponding security definitions to be performed.

To automatically create security profiles for system libraries (that is, all libraries whose IDs begin with "SYS"), proceed as follows:

- Log on to library "SYSSEC".
- On the Natural Security Main Menu, select "Administrator Services".
- On the Administrator Services Menu, select the function "Definition of System Libraries".

When you invoke the function, a list of the system libraries of Natural and all Natural subproducts installed at your site will be displayed. For each system library, a library-specific security profile is provided in which all the necessary components are already defined appropriately.

On the list, you can either mark with "AD" individual libraries to which you wish their pre-defined profiles to be applied one by one, or you can choose to have the pre-defined profiles applied to all product system libraries simultaneously by marking the corresponding product with "AD".

Note: This step should not be performed for SYS libraries containing Natural utilities, as it is recommended that these utilities be protected as described in the section *Protecting Utilities* in the *Natural Security* documentation.

If you use the function "System Libraries Definition" in an initial installation, you have to set the Natural profile parameter MADIO to a value of at least "2000".

Installation Verification

After Step 5 of the installation procedure has been completed successfully, Natural Security is operational.

Repeat the *Installation Verification* procedure for all Natural subproducts installed at your site.

16 Installing the Software AG Editor

The Software AG Editor is an optional feature that represents basic functionality within Natural, which is exclusively used by several Natural subproducts and other Software AG products.

- Prerequisites
- Support of a z/OS Parallel Sysplex Environment
- Migrating from Previous Versions
- Installation Procedure under z/OS and z/VSE
- Installation Procedure under BS2000/OSD
- Installation Procedure under CMS
- Installation Verification for the Software AG Editor

For details on the following topics, see also the documentation listed below:

- *Operating the Software AG Editor* in the Natural *Operations* documentation

- *SYSEDT Utility - Editor Buffer Pool Administration* in the Natural *Utilities* documentation

- Software AG Editor Functionality and Usage, refer to *Software AG Editor* in the Natural *Editors* documentation.

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Prerequisites

Installation Modules

As the Software AG Editor is an integral part of Natural, it can only be installed together with the current Natural version; that is, the version it is supplied with. The installation modules are provided as part of the Natural installation datasets.

Software AG Editor as an Alternative to the Natural Program Editor

If you want to use the Software AG Editor as an alternative to the Natural program editor, you still need to have Natural ISPF installed and to set your editor profile appropriately. For detailed information, see *Editor Profile* in the Natural *Editors* documentation.

Using SMA

If you install the Software AG Editor by using SMA, you must set a flag to `Y` (yes). This flag is called:

Platform:	Requirement:
z/OS or z/VSE	`SAG-EDITOR`
BS2000/OSD	The installation of the Software AG editor takes place automatically when installing Natural for BS2000/OSD.

Support of a z/OS Parallel Sysplex Environment

To support a z/OS Parallel Sysplex environment, that is, to be able to switch the z/OS host during a Natural session, the Software AG Editor must run without a buffer pool.

For this purpose, the profile parameter `EDPSIZE` (described in the Natural *Parameter Reference* documentation) is supplied where you can specify the size of an auxiliary editor buffer pool. All editor data are kept in the user storage thread. The total editor work space per user is limited by the `EDPSIZE` parameter. No editor work file is required. The `RECOVER` mechanism of the Software AG Editor is not supported.

Migrating from Previous Versions

Since the `RECOVER` mechanism from Software AG Editor is upward compatible, you can continue using a buffer pool work file from previous versions. It is possible to share buffer pools and their work files between Versions 4.1 and 4.2.

If you want to use **Version 4.1 and Version 4.2** editor work files concurrently in the same region, you have to use different logical dataset names. To define the dataset names, use the parameter `DDNAME` in the Natural parameter module.

Installation Procedure under z/OS and z/VSE

The installation procedure for the integrated Software AG Editor under z/OS and z/VSE consists of the following steps:

▶ **Step 1: Adapt the Editor Buffer Pool Parameter Macro NTEDBP**

■ The editor buffer pool parameters contained in the Natural parameter module (`NATPARM`) are stored in the editor buffer pool work-file control record during formatting (see **below**). `NATPARM` can be linked to the `EDITOR` work-file batch formatting utility.

⚠ **Important:** Before adapting the editor parameter module, first refer to the editor buffer pool parameter `EDBP` and its corresponding macro `NTEDBP` (described in the Natural *Parameter Reference* documentation).

Assemble the Natural parameter module and link it together with the Editor Work File Batch Format utility (`NATEDFM`) contained in the Natural load library.

For first-time installations:
You can leave the defaults. In this case, it is not necessary to assemble and link the Natural parameter module.

▶ **Step 2: Allocate the Editor Work File**

■ (Job I008, Steps 1900, 1901, 1903, 1905)

Skip this step for Com-plete.

A VSAM RRDS is used as the editor work file. To best exploit the VSAM dataset space, the record length should be defined 8 bytes less than the control interval length; see also *Editor Work File* in the Natural *Operations* documentation.

Depending on your TP-monitor environment, run Steps 1900, 1901, 1903 and 1905 of Job I008 to allocate the dataset:

```
Step 1900 ALLOCATE EDITOR WORKFILE BATCH
     1901 ALLOCATE EDITOR WORKFILE CICS
     1903 ALLOCATE EDITOR WORKFILE TSO
     1905 ALLOCATE EDITOR WORKFILE GLOBAL
```

▶ **Step 3: Format the Editor Work File**

■ (Job I081, Steps 1900, 1901, 1903, 1905)

Skip this step for Com-plete.

Use the Editor Work File Batch Format program (`NATEDFM`) to format and load the control record in the editor work file.

Depending on your TP-monitor environment, Steps 1900, 1901, 1903 and 1905 of Job I081 to format the dataset:

```
Step 1900 FORMAT EDITOR WORK FILE BATCH
     1901 FORMAT EDITOR WORK FILE CICS
     1903 FORMAT EDITOR WORK FILE TSO
     1905 FORMAT EDITOR WORK FILE GLOBAL
```

Platform:	Requirement:
z/VSE	To be able to use the batch format utility, the phase `NATEDFMV` must be created by linking the following modules:

`NATEDFMV NATPARM` (optional)

Link the modules (Job I060, Step 1900).

You may receive error message `IEC070I 203-204`, which can be ignored.

▶ **Step 4: Modify the Start-Up JCL and Subsystem Definitions**

■ You can change the DD name in the editor buffer pool's parameter macro. If you use another DD name than `CMEDIT`, you must specify it in your JCL.

Platform:	Requirement:
z/OS and TSO	Add a DD card for the editor work file:

```
//CMEDIT DD DSN=dataset-name,DISP=SHR
```

If you have specified the correct dataset name in the editor buffer pool's parameter macro in `NATPARM`, this step can be omitted, as Natural will allocate the file dynamically.

z/VSE	Add a DLBL card for the editor work file:

```
// DLBL CMEDIT,'dataset-name',,VSAM,CAT=catalog-name
```

In z/VSE batch mode, however, instead of using an actual VSAM file, a dummy editor work file can be defined by using JCL or standard labels, see *Example for z/VSE Batch Mode* below. Such a label definition enables you to use the editor and the editor buffer pool, but any write access to the editor work file will lead to an error, and recovering will not be possible.

Example for z/VSE Batch Mode:

```
// DLBL CMEDIT,'SAG.EDITOR.WORK.FILE',,VSAM,CAT=catalog-name
// EXTENT SYSnnn */nnnbeing any valid SYS number
// ASSGN SYSnnn,IGN
```

Platform:	Requirement:
Com-plete	The Software AG Editor uses an SD file as its work file under Com-plete, therefore no separate VSAM file is required. Therefore, a global buffer pool cannot be shared between Com-plete and other environments.
	The name of the SD file is indicated in the subparameter `DDNAME` of the `NTEDBP` macro. The `DSNAME` subparameter has therefore no significance.
	The number of work file records is set with the subparameter `RECNUM` and the work file record length is set with the subparameter `LRECL` of the `NTEDBP` macro.
	If you want to use a local editor buffer pool, add the definitions of the editor buffer pool to the `SERVER` parameter of your startup parameters as indicated in the section *Installation of the Natural Server under Com-plete*.
CICS	Add an entry in the CICS File Control Table (Job I005)

▶ **Step 5: Adapt all Natural Parameter Modules**

■ (Job I080)

Add the following parameter to your parameter module to specify the size of the editor area:

```
SSIZE=nn
```

where *nn* must be set to at least 54.

The following buffer-pool type-dependent steps must be made:

```
NTBPI TYPE=EDIT,SIZE=size
```

The region size must be defined large enough to be able to allocate the buffer pool.

Using a Global Buffer Pool

If you want to use a global buffer pool, you must specify the `NTBPI` macro (see *Assembling a Natural Parameter Module* in the Natural *Operations* documentation) in a different way so that you are able to specify the global editor buffer pool ID.

Specify the `NTBPI` macro as follows:

```
NTBPI TYPE=EDIT,NAME=name
```

Reassemble and link the parameter module.

If you want to use a global buffer pool, the Natural subsystem interface must be installed. For more information, see *Natural Global Buffer Pool* in the Natural *Operations* documentation.

▶ **Step 6: Define a Global Editor Buffer Pool**

■ Optionally, the editor buffer pool can be defined as a global buffer pool.

Similarly to the global Natural buffer pool, the global editor buffer pool can be shared by several regions. It is defined and started with the same procedure as a global Natural buffer pool; see *Natural Global Buffer Pool* in the Natural *Operations* documentation, only that parameter `TYPE=EDIT` has to be added. However, it must be kept separate from the global Natural buffer pool.

Make sure that all users of a global editor buffer pool have access to the same shared editor work file.

This will be checked by the editor buffer pool manager and can lead to abends if the current work file is different from the work file used during initialization.

⚠ **Important:** Under IMS TM, the use of a global buffer pool is mandatory.

▶ **Step 7: Relink Natural with the Software AG Editor**

■ (Job I080)

Natural must be relinked with the editor module `NATEDT`.

Assuming your Natural load library is referenced by `NATLIB`, add the following line to your linkage-editor input:

```
INCLUDE NATLIB(NATEDT)
```

The editor module can be linked to a shared nucleus or to an environment-dependent nucleus. You can link it to an alternate parameter module, or as a separate module named `EDITOR` if you are running with profile parameter `RCA` described in the Natural *Parameter Reference* documentation.

In this case, the editor module `NATEDT` must be linked as load module `EDITOR` and `RCA=EDITOR`.

Installation Procedure under BS2000/OSD

The installation procedure for the integrated Software AG Editor under BS2000/OSD consists of the following steps:

▶ **Step 1: Adapt the Natural Parameter Macro NTEDBP**

■ The editor buffer pool parameters contained in Natural parameter module (`NATPARM`) are stored in the editor buffer pool work-file control record during formatting (see below). `NATPARM` can be linked to the `EDITOR` work-file batch formatting utility.

🛑 **Caution:** Before adapting the editor parameter macro, first read the documentation of the editor buffer pool parameter `EDBP` and its corresponding macro `NTEDBP` in the Natural *Parameter Reference* documentation.

Assemble and link the Natural parameter module.

For first-time installations:
You can leave the defaults. In this case, it is not necessary to assemble and link the Natural parameter module.

▶ **Step 2: Adapt all Natural Parameter Modules**

■ (Jobs I080 and I060)

Add the following parameter to your parameter module to specify the size of the editor area:

```
SSIZE=nn
```

where *nn* must be set to at least 54.

Add the following macro to your parameter module to specify the size of the editor buffer pool:

```
NTBPI TYPE=EDIT
```

The region size must be defined large enough to be able to allocate the buffer pool. For a detailed description of the `NTBPI` macro, refer to *Assembling a Natural Parameter Module* in the Natural *Operations* documentation.

Reassemble and link the parameter module.

▶ **Step 3: Allocate and Format the Editor Work File E.FRM.EDITWORK**

■ (Job I081, Step 1900)

Allocate the editor work file (see also *Editor Work File* in the Natural *Operations* documentation) by issuing the command:

```
/FILE dataset-name,LINK=CMEDIT,SPACE=...
```

Then execute the formatting program (`NATEDFM2`, `NATvrs.MOD`) by entering `FORMAT` to format and load the control record in the editor work file.

▶ **Step 4: Modify the Start-Up Procedures for your Natural Nuclei**

■ For TIAM and/or UTM:

Add a `FILE` command for the editor work file:

```
/FILE dataset-name,LINK=CMEDIT
```

▶ **Step 5: Start the Global Editor Buffer Pool**

■ (Job I061, Step 0001)

Once defined, a global editor buffer pool must be started.

To start the global editor buffer pool, execute the module `CMPSTART`.

▶ **Step 6: Relink Natural with the Software AG Editor**

■ (Job I080)

Natural must be relinked with the editor module `NATEDT` and the work file access module `NATEDWA2` (see the source member `LNATSHAR` in your job library).

Assuming your Natural load library is referenced by `NATLIB`, add the following lines to the linkage-editor input:

```
INCLUDE NATLIB(NATEDT)
INCLUDE NATLIB(NATEDWA2)
```

Installation Procedure under CMS

The installation procedure for the integrated Software AG Editor under CMS consists of the following steps:

▶ **Step 1: Adapt the Natural Parameter Module**

■ The editor buffer pool parameters contained in the Natural parameter module (`NATPARM`) are stored in the editor buffer pool work file control record during the first initialization. Read about the editor buffer pool parameter macro `NTEDBP` described in the Natural *Parameter Reference* documentation.

For first-time installations:
You can leave the defaults for macro `NTEDBP`.

Add the following parameter to your `NATPARM` module to specify the size of the editor area:

```
SSIZE=nn
```

where *nn* must be set to at least 54.

Add the following macro to your `NATPARM` module to specify the size of the local editor buffer pool:

```
NTBPI TYPE=EDIT,SIZE=size
```

For a detailed description of the `NTBPI` macro, refer to *Assembling a Natural Parameter Module* in the Natural *Operations* documentation.

Reassemble and link the `NATPARM` module.

> **Caution:** The CMS machine must be defined large enough to allocate the editor buffer pool.

▶ **Step 2: Relink Natural with the Software AG Editor**

■ Natural must be relinked with the editor module `NATEDT`.

Run `NATBLDM EXEC` to rebuild Natural for CMS.

Check for the following line in `NAT$LOAD EXEC` and make sure that the value 50 is set:

```
SET LDRTBLS 50
```

Installation Verification for the Software AG Editor

Once you have installed the editor, you can verify the installation.

Invoke Natural and enter `SYSEDT` on the command line.

The Natural utility `SYSEDT` displays all buffer pool parameters and usage statistics; see *SYSEDT Utility - Editor Buffer Pool Administration* in the Natural *Utilities* documentation.

To be able to test the full operation of the Software AG Editor, a subproduct (for example, Natural ISPF) must be installed which uses its functionality.

17 Installing the Natural Net Data Interface

The Natural Net Data Interface `NATNETT0` supports the EntireX CICS 3270 Bridge and similar client/server solutions in message-oriented server environments, that is, TP monitors.

Notation vrs or vr: If used in the following document, the notation *vrs* or *vr* stands for the relevant **v**ersion, **r**elease, **s**ystem maintenance level numbers. For further information on product versions, see Version in the *Glossary*.

Device Configuration in NATCONFG

Logical net data devices are configured with the flag byte `IONET`, all other flag and value settings must be made as in the sample definition given below. The module entry is `VCNETT0`, additional device entries must be specified with `WXTRN=OFF`. The settings of `FLAG1`, `FLAG2` and `RTAL` must be according to the example below.

If the delimited mode is set, the delimiter character which separates the fields in the value buffer can be set via cmbel.

IONET Settings

IONET	DS	XL1	NETDATA CONTROL FLAG
NECUFNR	EQU	X'01' 1	CURSOR POSITION = FIELDNR
NEMSG	EQU	X'02' .... ..1.	SEND MESSAGE LINE (if not set, message line will be skipped)
NEABO	EQU	X'04' .... .1..	ATTRIBUTE BUFFER OPTION
NEFBO	EQU	X'08' .... 1...	FORMAT BUFFER OPTION

NEFLG	EQU	X'10' ...1	FIELD LENGTH OPTION
NEDLM	EQU	X'20' ..1.	DATA DELIMITED OPTION
NEFIX	EQU	X'40' .1..	FIXED FORMAT OPTION
NEFBOPTE	EQU	X'80' 1...	EXTENDED FORMAT BUFF. OPT.

NATCONFG already has a device entry for NATNETTO.

The TTYPE is NETF. The protocol options are set as follows:

- Value buffer structure is fixed (without delimitation between the fields).
- Format buffer extended format buffer and attribute buffer options are set.
- The cursor position is in field number notation.

Message line and PF-key line are suppressed

Example:

```
NTDVCE TYP=NETF,NAME=NETTF,ENTRY=VCNETTO,MSG=BOT,              HS06-
       FLAG1=CMNIXD,FLAG2=CMTNOPT,RTAL=255,                    HS07-
       FLAGS=(IONET,-,CO,IONET,+,NEFIX+NEFBO+NEABO+NECUFNR+NEFB-
       OPTE,WINDTITI,+,PFKNDISP)                              HS06
```

Installation Procedure under z/OS

▶ **To install the Natural Net Data Interface under z/OS, adapt the following link step for Natural.**

■ Add the following `INCLUDE` instruction and the corresponding DD-statements in the link instructions for the linkage editor:

```
INCLUDE NATLIB(NATNETTO)
```

Installation Procedure under z/VSE

▶ **To install the Natural Net Data Interface under z/VSE, adapt the following link step for Natural.**

■ Add the following `INCLUDE` instruction in the search chain for the linkage-editor:

```
INCLUDE NATNETTO
```

Installation Procedure under BS2000/OSD

▶ **To install the Natural Net Data Interface under BS2000/OSD, adapt the following link steps for Natural.**

1 Add the following `INCLUDE` instruction to the element `LNATSHAR` in `LIB.NAT`*vrs*:

```
INCLUDE NATNETTO,NATvrs.MOD
```

2 Relink your Natural shared nucleus with procedure `P.LINKMOD` in `LIB.NAT`*vrs*

18 Installation Steps for REQUEST DOCUMENT and PARSE XML

This document describes the installation steps required to enable the use of the Natural statements `REQUEST DOCUMENT` and `PARSE XML`.

- Platforms Supported
- Prerequisites
- Building the Natural Nucleus
- Building the Natural Front-End
- Installation Verification

See also

- *Statements for Internet and XML Access* in the *Programming Guide*

- `REQUEST DOCUMENT` in the *Statements* documentation

- `PARSE XML` in the *Statements* documentation

Platforms Supported

For a list of mainframe computer platforms on which the Natural statements `REQUEST DOCUMENT` and `PARSE XML` are supported, see the following sections in *Statements for Internet and XML Access* in the Natural *Programming Guide*.

- *Platform Support for REQUEST DOCUMENT*

- *Platform Support for PARSE XML*

Prerequisites

See *General Prerequisites*, *Installation Prerequisites* in the section *Statements for Internet and XML Access* in the Natural *Programming Guide*

Building the Natural Nucleus

The runtime for the statements `REQUEST DOCUMENT` and `PARSE XML` is provided in the module `NATXML`.

To enable the runtime, the module `NATXML` has to be linked to the Natural shared nucleus.

For further information, refer to the appropriate installation step of your environment.

Building the Natural Front-End

Preliminary Remarks

Concerning all batch and TP-system drivers besides Com-plete on z/OS, z/VSE and VM/CMS

- The Natural batch and TP monitor system drivers have to be installed LE enabled, that is, the parameter `LE370` must be set to `YES`. Refer to the corresponding part of the Natural documentation for your specific environment.

Concerning BS2000/OSD

- The keyword parameter `ILCS` of macro `NAMTIAM` must be set to the value `CRTE` in the assembly step of the reentrant part of the TIAM driver.

- The keyword parameter `ILCS` of macro `NAMBS2` must be set to the value `CRTE` in the assembly step of the reentrant parts of the batch driver.

- The keyword parameter `ILCS` of macro `NURENT` must be set to the value `CRTE` in the assembly step of the reentrant part of the UTM driver.

Linking the Natural Front-End Modules

Under z/OS:

Batch and TSO	Add `INCLUDE`s for the modules `NAT2LE` and `NAT2TCP` to the link step of the Natural front-end module.
	Resolve unresolved external references from the LE library (usually `hlq.SCEELKED`), that is, add this library to the `SYSLIB` definition of your link job and do *not* specify the `NCAL` parameter for the link.
CICS	Add an `INCLUDE` for the CICS socket module `EZACIC17` contained in the CICS socket library (usually `hlq.SEZARNT1`, `hlq.SEZATCP` or `hlq.SEZACMTX`).
	Resolve unresolved external references from the CICS socket library and the current LE library (usually `hlq.SCEELKED`), that is, add these libraries to the `SYSLIB` definition of your link job and do *not* specify the `NCAL` parameter for the link.
	Configure the CICS TCP/IP environment as described in the IP CICS Socket Guide by IBM.
Com-plete	Link the module `NAT2LE` to the Natural Com-plete front-end module.
	Copy the module `NCFTCP42` to the Com-plete load library.

	The TCPIP CDI driver and the `POSIX SERVER` statement in the Com-plete parameter module `SYSPARM` are required.
IMS TM	Add `INCLUDE`s for the modules `NAT2LE` and `NAT2TCP` to the link step of all NII front-end modules. Resolve unresolved external references from the LE library (usually `hlq.SCEELKED`), that is, add this library to the `SYSLIB` definition of your link job and do not specify the `NCAL` parameter for the link.

Under z/VSE:

Batch and CICS	Add `INCLUDE`s for the modules `NAT2LE` and `NAT2TCP` to the link steps of the batch front-end module and the NCI front-end module. Resolve unresolved external references from the LE library (usually `PRD2.SCEEBASE`), that is, add this library to the `OBJ-SEARCH` definition of your link job and do *not* specify the `NOAUTO` parameter in the `ACTION` statement for the link.
Com-plete/NDV-SMARTS	Link the module `NAT2LE` to the Natural Com-plete front-end module. Copy the phase `NCFTCP42` to the Com-plete/SMARTS load-library. The TCPIP CDI driver and the `POSIX SERVER` statement in the Com-plete parameter module `SYSPARM` are required.

Under VM/CMS:

VM/CMS	Xedit `NAT$LOAD EXEC` and append `NAT2LE` and `NAT2TCP` to the variable `loadlist`. For enabling the access to TCP/IP, refer to *Using TCP/IP Communication* in the *Operations* documentation.

Under BS2000/OSD:

Batch, TIAM and *open***UTM**	Add `INCLUDE`s for the module `NAT2LE` and the LLM `NAT2TCP` to the front-end link steps for batch mode and TIAM. Add an `INCLUDE` for the CRTE initialization routine `ITOSL#`, if `ITOSL#` is not already included into the front-end module. Add a `RESOLVE-BY-AUTOLINK LIB=($TSOS.SYSLNK.CRTE)` statement to the link step for the Natural front-end module. **Note:** Since `NAT2TCP` is delivered in LLM format, the front-end has to be linked with the binder!
NDV	The `REQUEST DOCUMENT` statement is not currently supported by the Natural Development Server for BS2000/OSD.

Installation Verification

Activate the statements in the runtime environment; see *Activation/Deactivation* in *General Prerequisites* in the section *Statements for Internet and XML Access* in the Natural *Programming Guide*.

Try the sample programs contained in library `SYSEXV`.

For information on profile settings to enable the support of the `REQUEST DOCUMENT` and/or `PARSE XML` statement, see the following documents:

- *Profile Settings* in *General Prerequisites* in the section *Statements for Internet and XML Access* in the Natural *Programming Guide*
- Profile parameter `XML` in the *Parameter Reference* documentation

19 **Overall Installation Verification**

This document describes how to verify the successful completion of the preceding installation procedures.

- Installation Verification for Base Natural
- Installation Verification for the TP-Monitor Interface

Installation Verification for Base Natural

For base Natural, there are no specific installation verification procedures. After the last step of the installation procedure has been successfully performed, check that the following results are available:

- Communication between Adabas and Natural is working.

- The Natural system files have been loaded.

- Batch Natural is operational.

Installation Verification for the TP-Monitor Interface

Once you have installed a Natural/TP-monitor interface (as outlined at the beginning of this documentation in the section *Overview of the Installation Process*), you should verify the successful installation of the TP-monitor interface.

Before you start the verification procedure, ensure that you have already created the EMPLOYEES sample file; if not, refer to the appropriate Adabas documentation for information on how to create it.

▶ **To verify the successful installation of the TP-monitor interface, proceed as follows:**

1   Invoke the Natural utility SYSDDM by entering the following command in the Natural command line:

```
SYSDDM
```

On the SYSDDM menu, enter Code "E" (Edit DDM) and DDM Name "EMPLOYEES". Press PF3 to quit.

Enter Code "C" (Catalog DDM), the file number and the database ID of your EMPLOYEES demo file and Replace "Y".

After these functions have been performed, exit from SYSDDM.

2   Log on to library SYSEXSYN:

```
LOGON SYSEXSYN
```

EDIT, CHECK, STOW, RUN, EXECUTE the example programs FNDSOR and REAEX1S in this library.

3   Invoke SYSEXV:

```
SYSEXV
```

Try the programs in this library; they demonstrate the new features provided by the current Natural version.

Index

E

Entire System Server Interface Installation, 166