Natural for Mainframes

Natural Connection

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.

Copyright © Software AG 1979-2009. All rights reserved.

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

Table of Contents

1 Natural Connection 1
2 Main Functionality 3
    Interactive Data Transfer 4
    Integration 4
    Security 4
3 Installing Natural Connection 5
    General Information 6
    Installation Tape - z/OS 7
    Installation Tape - z/VSE 7
    Installation Tape - BS2000/OSD 7
    Installation Tape - VM/CMS 8
    Installation Procedure 9
    Installation Verification 10
4 Preparing to Use Natural Connection 11
    Displaying Work File and Printer Settings 12
    Defining Work Files and Printers 13
    Activating the PC Connection 13
    Deactivating the PC Connection 14
5 Natural Statements 15
6 Processing Work Files and Nested Loops 17
    Work File Format and Attributes 18
    Maximum File Transfer Record Length for Natural Connection 18
    Streaming 19
    Dynamic Variables in READ WORK FILE 19
    Nested READ Loops 20
    Subsequent READ Loops 21
    Buffer Allocation for Large Upload Records 22
Index 25

1 Natural Connection

Natural Connection provides an access method required for transferring data, reports, Natural objects and sources from a Natural mainframe computer to a PC.

The prerequisite for the use of Natural Connection is Entire Connection. For information on Entire Connection, refer to the relevant documentation.

This documentation is organized under the following headings:

Main Functionality	Describes the main functionality that is relevant for the mainframe.
Installing Natural Connection	Describes how to install Natural Connection on a mainframe.
Preparing to Use Natural Connection	Describes the tasks that must be accomplished to allow download and upload.
Natural Statements	Describes the Natural statements that are relevant for processing information between the mainframe and the PC by using Natural Connection.
Processing Work Files	Describes restrictions on the use of work file attributes, the support of work file formats and the impact of `READ` loops.

2 Main Functionality

- Interactive Data Transfer 4
- Integration 4
- Security 4

This chapter describes the main functionality provided by Natural Connection that is relevant for the mainframe.

For details regarding the features of the PC component, see the Entire Connection documentation.

Interactive Data Transfer

Data can be downloaded to and uploaded from the PC directly to/from an online program.

Data transfer and conversion to many formats (for example, Lotus 1-2-3, dBASE, ASCII, HTML, XML, Basic, DIF, Binary) is performed in one step. Sophisticated compression techniques are used.

Integration

Using Natural Connection, you can create integrated applications that use both mainframe and PC resources.

For example, Super Natural creates files that you can then further process on the PC. Con-nect uses Natural Connection to exchange documents with the PC. In this way, graphics, spreadsheets, or binary data can be distributed to other users.

Mainframe applications can use PC printers.

Security

A key consideration for any information network is effective security and control. Natural Security on the mainframe provides comprehensive control facilities.

Detailed security profiles are assigned on a user-by-user basis to control access to data and programs.

Other significant objects secured are the individual keywords in the Natural syntax. This can restrict some users, for example, to retrieving data from mainframe databases, while others have the full capability for retrieval and update of mainframe data.

Natural Security also protects against excessive use of system resources by individual users. This is vital when PC users may be employing powerful mainframe features for the first time.

As an additional safeguard, Natural Connection can also encrypt all data downloaded from Natural.

3 Installing Natural Connection

- General Information 6
- Installation Tape - z/OS 7
- Installation Tape - z/VSE 7
- Installation Tape - BS2000/OSD 7
- Installation Tape - VM/CMS 8
- Installation Procedure 9
- Installation Verification 10

This chapter describes how to install Natural Connection under the operating systems z/OS, z/VSE, BS2000/OSD and VM/CMS.

General Information

The following topics are covered below:

- Prerequisites
- Installation Jobs
- Using System Maintenance Aid

Prerequisites

To use Natural Connection, Entire Connection must be installed.

For further information, refer to the products and versions specified under *Natural and Other Software AG Products* and *Operating/Teleprocessing Systems Required* in the current Natural *Release Notes*.

Installation Jobs

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

For each step of the installation procedure described below, the job number of a job performing the respective task is indicated. This job number refers to an installation job generated by SMA. If you are not using SMA, an example installation job of the same number is provided in the job library on the Natural installation tape; you must adapt this example job to your requirements. The job numbers on the tape are preceded by a product code (for example, NTCI080).

Using System Maintenance Aid

For information on using Software AG's System Maintenance Aid (SMA) for the installation process, refer to the System Maintenance Aid documentation.

Installation Tape - z/OS

The installation tape contains the dataset listed in the table below. The sequence of the dataset is shown in the Report of Tape Creation which accompanies the installation tape.

Dataset Name	Contents
NTC*nnn*.LOAD	Natural Connection load modules.

The notation *nnn* in the dataset name represents the version number of the product.

Installation Tape - z/VSE

The installation tape contains the dataset listed in the table below. The sequence of the dataset is shown in the Report of Tape Creation which accompanies the installation tape.

Dataset Name	Contents
NTC*nnn*.LIBR	LIBR backup file.

The notation *nnn* in the dataset name represents the version number of the product.

Installation Tape - BS2000/OSD

The installation tape contains the dataset listed in the table below. The sequence of the dataset is shown in the Report of Tape Creation which accompanies the installation tape.

Dataset Name	Contents
NTC*nnn*.MOD	Natural Connection module library.

The notation *nnn* in the dataset name represents the version number of the product.

Installation Tape - VM/CMS

The installation tape contains the dataset listed in the table below. The sequence of the dataset is shown in the Report of Tape Creation which accompanies the installation tape.

Dataset Name	Contents
NTC*nnn*.TAPE	Natural Connection TEXT file.

The notation *nnn* in the dataset name represents the version number of the product.

Copying the Tape Contents to Disk

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

   If the sequence number of `NTC`*nnn*`.TAPE`, as shown by the Report of Tape Creation, is *m*, you must position over 3*m* - 2 tape marks (that is, FSF 1 for the first dataset, FSF 4 for the second, etc.).
2. Access the disk that is to contain the Natural installation files as disk "A".
3. Ask the system operator to attach a tape drive to your virtual machine at the address X'181' and mount the Natural Connection installation tape.
4. When the tape has been attached, enter the CMS command:

   ```
   TAPE REW
   ```

   Position the tape by entering the CMS command:

   ```
   TAPE FSF m
   ```

   where *m* is the number of tape marks and is calculated as described above (3*m* - 2).
5. Load the Natural Connection/CMS installation material by entering the CMS command:

   ```
   TAPE LOAD * * A
   ```

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

Installation Procedure

This section describes step by step how to install Natural Connection under the operating systems z/OS, z/VSE, BS2000/OSD and VM/CMS.

Step 1: Adjust Natural Parameter Module - Job I080

Specify the keyword parameter `AM=PC` for all printer files and work files you want to use for data transfer between the host and the PC. For example:

```
NTPRINT (7),AM=PC
NTWORK  (7),AM=PC
```

Specify the parameter `PC=ON` either dynamically or in your Natural parameter module.

For information on using the parameters, refer to *Profile Parameter Usage* (in the *Operations* documentation).

For details on the parameters or modules, see the *Parameter Reference* or *Using Macros in a Natural Parameter Module* in the *Operations* documentation.

Step 2: Adapt Link Steps - Job I080

Adapt the link steps for online Natural.

- **z/OS**
  Add the following `INCLUDE` instruction and the corresponding DD-statements in the link instructions for the linkage editor:

  ```
  INCLUDE NTCLIB(NTCPCAM3)
  ```

- **z/VSE**
  Add the following `INCLUDE` instruction and the corresponding sublibrary for Natural Connection in the search chain for the linkage-editor:

  ```
  INCLUDE NTCPCAM3
  ```

- **BS2000/OSD**
  Add the following `INCLUDE` instruction to the element `LNATSHAR` in *`JOBLIB`* (see *Installing Natural under BS2000/OSD* in the *Installation* documentation for information on the naming convention for *`JOBLIB`*):

  ```
  INCLUDE NTCPCAM3,NTC<nnn>.MOD
  ```

  where *`nnn`* stands for the current version number.

Relink your Natural shared nucleus as described in *Installing Natural under BS2000/OSD* in the *Installation* documentation.

- **VM/CMS**
  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.

  Add the following statement to the program `NAT$LOAD EXEC` or `NATBLDS` respectively.

```
LOADLIST = LOADLIST 'NTCPCAM3'
```

  Relink your Natural nucleus with the procedure `NATBLDM`.

Installation Verification

To verify the successful installation of Natural Connection

1. Invoke Entire Connection on the PC and start the terminal emulation.
2. Use the terminal emulation to invoke Natural on the mainframe.
3. Enter the terminal command `%+` to activate the PC connection.
4. Use the Natural Object Handler (see the relevant documentation) to download Natural source code to the PC. For further information on transferring Natural objects, see *Data Transfer* which is part of the *Terminal Emulation* section of the Entire Connection documentation.
5. Verify that the downloaded program is now on your PC.

4 Preparing to Use Natural Connection

- Displaying Work File and Printer Settings 12
- Defining Work Files and Printers 13
- Activating the PC Connection 13
- Deactivating the PC Connection 14

This chapter describes the tasks that must be accomplished to allow download and upload.

In order to download and upload data, a work file must be designated as a PC file.

To download reports, a printer must be designated as a PC printer.

Displaying Work File and Printer Settings

To display your current work file and printer settings, enter the Natural system command `SYSFILE` at the `NEXT` prompt. This invokes the `SYSTP` utility and a screen similar to the following appears:

```
11:07:18              ***** NATURAL SYSTP UTILITY *****             2005-06-16
 User SAG                   - Work File Information -             TID DAEFTCI7

 M No.   Type      Name   Recfm  Lrecl  Blksz  Status
 ------------------------------------------------------------------------------
    1  COMPLETE  CMWKF01   VB            4628  Available for Input/Output
    2  COMPLETE  CMWKF02   VB            4628  Available for Input/Output
    3  COMPLETE  CMWKF03   VB            4628  Available for Input/Output
    4  COMPLETE  CMWKF04   VB            4628  Available for Input/Output
    5  PC                  VB            4628  Available for Input/Output
    6  PC                  VB            4628  Available for Input/Output
    7  PC                  VB            4628  Available for Input/Output









   Top of List
Command ===>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
Cont  Help  Menu  Exit  Sel   Pos   --    -     +     ++    Print Work  Canc
```

The standard PC files are work files 5, 6 and 7 as shown in the example above. For more information, see *SYSTP Utility* in the *Utilities* documentation.

Defining Work Files and Printers

You can define work files and printers statically in your Natural parameter module, or dynamically when invoking Natural:

- **Work Files**
  Use the `NTWORK` macro or `WORK` parameter for work file definitions.
- **Printers**
  Use the `NTPRINT` macro or `PRINT` parameter for printer definitions.

For detailed information on these macros and parameters, see the *Parameter Reference*.

Remember, however, that the download and upload modules provided by Natural Connection use work file 7. For further information on transferring Natural objects, see *Data Transfer* which is part of the *Terminal Emulation* section of the Entire Connection documentation.

Activating the PC Connection

To upload and download data to/from a PC, the PC connection must be active. You activate the PC connection from the mainframe.

To activate the PC connection

- Use the Natural terminal command `%+`.

  Or:

  Invoke Natural with the dynamic parameter `PC=ON`.

  Or:

  Use the `SET CONTROL '+'` statement in a Natural program.

With the terminal command %+, you can also set an additional option:

Option	Description
%+N	The PC connection is activated. In addition, no field names are sent when downloading or uploading data.

If you attempt to upload or download data without the PC connection being activated, a message appears indicating that the PC connection is not active.

If you enter %+ and the PC connection is already active, a message appears indicating that the PC connection is already active.

Deactivating the PC Connection

To deactivate the PC connection

- Use the Natural terminal command %-.

  Or:

  Use the SET CONTROL '-' statement in a Natural program.

5 Natural Statements

This chapter lists the Natural statements required to process information between the mainframe and the PC by using Natural Connection.

For detailed information on these statements, refer to the relevant sections in the *Statements* documentation.

The Natural statements that apply to Natural Connection can be divided into the following groups:

Transfer Data

- `DOWNLOAD PC FILE` (synonym for `WRITE WORK FILE`)
- `UPLOAD PC FILE` (synonym for `READ WORK FILE`)

Download Reports

- `DISPLAY`
- `PRINT`
- `WRITE`

Close a PC File

- `CLOSE PC FILE` (synonym for `CLOSE WORK FILE`)

6 Processing Work Files and Nested Loops

- Work File Format and Attributes 18
- Maximum File Transfer Record Length for Natural Connection 18
- Streaming 19
- Dynamic Variables in READ WORK FILE 19
- Nested READ Loops 20
- Subsequent READ Loops 21
- Buffer Allocation for Large Upload Records 22

This chapter describes restrictions on the use of work file attributes, the support of work file formats and the impact of READ loops.

Work File Format and Attributes

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

- Accessing PC work files is restricted to a fixed record length of 1073741823 bytes or 32767 bytes when using the statement WRITE WORK FILE VARIABLE. Depending on the Entire Connection version installed on the PC, additional restrictions may apply as described below.
- Natural Connection does not support work files of the type UNFORMATTED. A work file is always transferred in formatted mode and contains record-oriented data only. When a work file of the type UNFORMATTED is opened, Natural Connection switches to the type FORMATTED and executes any WRITE WORK FILE statement with the option VARIABLE. To transfer byte-streamed data, see *Streaming* below.

Maximum File Transfer Record Length for Natural Connection

As of Natural Connection Version 4.1, the maximum record length supported for file transfer depends on the version of Entire Connection installed on the PC.

For Entire Connection up to Version 4.2, the maximum record length is limited by the number of bytes that can be displayed on the appropriate 3270 model. For example, for a 3270 Model 2 device the record length is 24*80 = 1920 bytes. Since all data buffers are enclosed by a header and trailer, the resulting net record length is 1887 bytes.

For Entire Connection Version 4.3.1, the maximum record length is limited to 32 KB - 1 byte = 32767 bytes.

As of Entire Connection Version 4.3.2 Patch Level 1 and Entire Screen Builder Version 5.2.1, the maximum record length is increased to 1 GB - 1 byte = 1073741823 bytes. But writing work files in variable format (WRITE WORK VARIABLE) is still restricted to a maximum record length of 32 KB - 1 byte.

Streaming

Entire Connection provides the option to transfer byte-streamed data that are non-record-oriented. A byte-streamed data transfer is activated when a `READ WORK FILE` or `WRITE WORK FILE` statement is coded with only one single operand of binary format.

Downloading and Uploading Binary Data

Binary data is usually object code or executable code that does not contain displayable or printable characters. To prevent standard character translations being performed during data transfer, Natural and Entire Connection use special methods for transferring binary data.

To download binary data

1 Define a binary variable.

2 If the last block of downloaded data contains less data than the block size chosen, insert `X'FF'` at the position that marks the end of the binary data. (If you omit `X'FF'`, the rest of the last block will be filled with `X00`.)

To upload binary data

1 Define a binary variable.

2 Remove `X'FF` from the last block. `X'FF` marks the end of the binary data.

Dynamic Variables in READ WORK FILE

If you define a dynamic variable of the format binary or alphanumeric as operand of a `READ WORK FILE` statement, when processing the corresponding `READ` loop, any resize operation on this variable will only be valid until the next `READ` is performed. While processing the `READ`, Natural resizes all dynamic variables to the size they had when the work file was opened. This is required in the open process which determines the record layout. The record layout is mandatory for processing the corresponding work file. The record layout is valid until the next close of the work file occurs.

Exception: An internal resize cannot be performed for inner loops if nested `READ` loops are processed on the same work file. See also the programming recommendations about nested loops below. If a dynamic variable of size 0 is used as the only operand of a `READ WORK FILE` statement, Natural issues the error NAT1500.

Nested READ Loops

Do not specify nested READ loops on one work file. The result of the inner loop(s) can be unpredictable if the operands of the inner loop do not correspond to the operands of the outer loop. The reason is that all records uploaded from the PC are processed in the format that was determined when the work file was opened in the outermost loop.

Below are example programs that demonstrate the unpredictable results the inner loop(s) of nested READ loops can have:

- Example of Inner READ Loop
- Example of READ Loop and CALLNAT

Example of Inner READ Loop

In the example program PCNESTED, during READ processing, another READ is performed:

```
/* PCNESTED
/*
DEFINE DATA LOCAL
  1 #REC1   (A) DYNAMIC
  1 #NUMBER (N10)
END-DEFINE
*
MOVE ALL 'TEST RECORD 1' TO #REC1 UNTIL 100
READ WORK FILE 1 #REC1
   READ WORK FILE 1 #NUMBER
     DISPLAY #NUMBER
   END-WORK
END-WORK
END
```

Example of READ Loop and CALLNAT

In the example program PCMAIN and subprogram PCRSUB01, during READ loop processing, an external object is called:

```
/* PCMAIN
/*
DEFINE DATA
LOCAL
  1 RECL (A2000)
  1 REDEFINE RECL
    2 RECNR (N4)
  1 CO   (N4
END-DEFINE
```

```
*
WRITE WORK 1 COMMAND
 'SET PCFILE 2 UP DATA C:/TSTPCAM/PCMAIN.TXT'
READ WORK 2 RECL
  DISPLAY RECL (AL=72)
  CALLNAT 'PCRSUB01' RECL
END-WORK
END
```

Subprogram `PCRSUB01`:

```
/*Subprogram PCRSUB01
/*
DEFINE DATA
PARAMETER
  1 RECL (A2000)
LOCAL
  1 #CC1 (A20)
  1 #CC2 (N4)
*
END-DEFINE
READ WORK 2 RECL
  #CC1 #CC2
  DISPLAY #CC1 #CC2
END-WORK
END
```

Subsequent READ Loops

If a `READ` loop is terminated by a conditional `ESCAPE`, close the work file explicitly with the `CLOSE WORK FILE` statement so that the same work file can be processed in a subsequent `READ` in the same object.

Exception: You can omit the `CLOSE WORK FILE` if you need not read the file again from the beginning, and if the subsequent `READ` uses the same record layout as the preceding one.

Below is an example that demonstrates how to correctly code a program with two `READ` loops on one work file.

Example of Loop with ESCAPE and CLOSE

In the example program PCESCAPE, the work file is explicitly closed after the first READ loop has been terminated by ESCAPE BOTTOM so that the second READ loop must reopen the work file:

```
/*PCESCAPE
/*
DEFINE DATA
LOCAL
  1 #CC1        (A20)
  1 #CC2        (A40)
  1 #COUNTER    (I2)
*
END-DEFINE
READ WORK 2 #CC1
  DISPLAY #CC2
  ADD 1 TO #COUNTER
  IF #COUNTER GE 3
    ESCAPE BOTTOM
  END-IF
END-WORK
CLOSE WORK FILE 2
*
READ WORK 2 #CC2
  DISPLAY #CC2
END-WORK
END
```

Buffer Allocation for Large Upload Records

If Natural Connection uploads a record that is larger than one physical block, Natural Connection collects all blocks that belong to the record in the appropriate work file area. The record will then be decompressed and passed to the Natural data area.

The total space allocated by all temporary buffers is up to 3 times the size of the record to be uploaded.

Example Statement

```
READ WORK FILE 1 #var
```

where 1 is the number of the work file and `#var` a variable of the format B 10000. In this case, the temporary Natural work area requires approximately 30000 bytes.

Index

D

data transfer
    to PC, 1

N

Natural Connection, 1

P

PC
    transfer data with Natural Connection, 1

U

use
    Natural Connection, 1