

Migrating Existing Applications

This chapter covers the following topics:

- Impact of Unicode on Existing Applications
 - Migrating Existing Objects on Windows, UNIX and OpenVMS Platforms
 - Migrating Existing Objects on Mainframe Platforms
 - Adding Unicode Support to Existing Applications
 - Migrating Natural Remote Procedure Calls (RPC)
-

Impact of Unicode on Existing Applications

Windows, UNIX and OpenVMS Platforms

On Windows, UNIX and OpenVMS platforms, Natural has internally been Unicode-enabled which means that many structures containing strings have Unicode format now. For example, the Natural source area has now Unicode format. For this reason, data which is only available in code page format is internally converted to Unicode format. This applies, for example, to the Natural sources and to the Natural library names and object names. However, a conversion from code page to Unicode and vice versa can only be performed successfully if the correct code page is used for conversion. Even if an application is not changed but only re-cataloged, the code page information is important because for cataloging an object is loaded into the Natural source area. If all objects are coded in the system code page, no changes are necessary. If the objects are not coded in the system code page, see *Migrating Existing Objects on Windows, UNIX and OpenVMS Platforms* for further information.

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

Mainframe Platforms

There is no impact of Unicode on existing applications. The internal structures have not been changed and no conversion of A format fields is enforced. This means that existing Natural applications should execute without any effort. Make sure that the parameters `CFICU` and `CP` have to be set to `OFF`. Even the need to link one of the ICU load modules (`NATICU`, `NATICUCV` or `NATICUXL`) becomes redundant in this case. Only the I/O buffers have been noticeably increased since the attributes have been enhanced to support potential Unicode fields. If `CP` is set to `OFF`, the system variable `*CODEPAGE` is cleared and the well-known translation tables (such as standard table or alternative table) are continued to be used for I/O translations.

Migrating Existing Objects on Windows, UNIX and OpenVMS Platforms

Natural has been extended so that the code page information can be defined on several levels:

- The Natural profile parameter CP defines the default Natural code page.
- For several objects (Natural sources, Natural batch input/output files, work files of type ASCII, ASCII compressed, Unformatted and CSV) an object-specific code page can be defined.

If neither an object-specific code page nor a default code page is defined, Natural will use the operating system's code page.

Since it is not possible to identify the correct code page automatically, it is important that you define the required code page information yourself. The following scenarios are possible:

Status	Effort	Action
All data is available in the operating system's code page.	No effort	No action.
All data is stored with one code page, but this code page differs from the operating system's code page.	Easy	The Natural profile parameter CP has to be set to the correct code page.
The data is available in different code pages.	Depends on the number of sources and code pages	<p>The correct code page has to be defined for every Natural object:</p> <ul style="list-style-type: none"> • Sources If only few objects are affected, change the code page via the Properties dialog box. If several objects (for example, an entire library) are affected, use the FTOUCH utility for changing the code page. • Batch Files Set the Natural profile parameters CPOBJIN, CPSYNIN and CPPRINT to the correct code page. • Work Files Set correct code page for the work files in the Configuration Utility.
Different code pages are mixed in one object (for example, in a source)	High	The object has to be rewritten in UTF-8 format.

Migrating Existing Objects on Mainframe Platforms

Natural has been extended so that the code page information can be defined on several levels:

- The Natural profile parameter CP defines the default Natural code page.
- For several objects (Natural sources, Natural batch input/output files, print reports, Adabas files) an object-specific code page can be defined.

If neither an object-specific code page nor a default code page is defined (that is, CP=OFF applies), Natural does not convert any data.

Since it is not possible to identify the correct code page automatically, it is important that you define the required code page information yourself. The following scenarios are possible:

Status	Effort	Action
All data is available in the operating system's code page.	No effort	No action.
All data is stored with one code page, but this code page differs from the operating system's code page.	Easy	The Natural profile parameter CP has to be set to the correct code page. Make sure that the I/O device supports this code page. CP=AUTO forces Natural to run with the code page of the I/O device.
The data is available in different code pages.	Depends on the number of sources and code pages	The correct code page has to be defined for every Natural object: <ul style="list-style-type: none"> ● Sources Save each object in the session with the correct code page. ● Batch Files Set the Natural profile parameters CPOBJIN, CPSYNIN and CPPRINT to the correct code page. ● Adabas Files (ECS enabled) Set the Natural profile parameter OPRB with the ACODE option.
Different code pages are mixed in one object (for example, in a source)	High	The object has to be rewritten in the appropriate code page format.

Sources which have been saved or stowed with previous Natural versions do not have code page information. The code page field of the directory is empty.

Since Natural sources are not saved in Unicode format, the source has to be converted into the default code page (value of the system variable *CODEPAGE) that applies to the session. If code page support is switched off (CP=OFF), the code page information of the source is ignored and no conversion is performed. Alphanumeric constants have to be adjusted to the default code page when they are loaded into the source area.

Since Natural sources are not saved in Unicode format, alphanumeric constants have to be adjusted to the default code page during start of the object. This can be achieved with the CPAGE compiler option. If CPAGE is set to ON, an additional table is generated into the object. The Natural loader uses this table to

convert every alphanumeric constant to the default code page (value of the system variable *CODEPAGE). Depending on the amount of alphanumeric constants, the additional table increases the size of the resulting object and the conversion consumes additional CPU time.

It is important that dependent objects (for example, a program and a local data area used by the program) use the same code page. If dependent objects use different code pages, it should be ascertained that the used characters (for example, "#") are mapped to the same code points in the used code pages. The following objects and data do not have an associated object-specific or data-specific code page:

- Data definition modules (DDMs),
- Predict rules,
- Predict XRef data.

Care should be taken if such data is used in or produced by objects for which an object-specific code page has been defined. If the application itself does not necessarily have to be code page enabled and you want the application to be code page sensitive with respect to the data that is being processed, you should consider to use the profile parameter SRETAIN with the value (ON, EXCEPTNEW).

Adding Unicode Support to Existing Applications

It is easy to extend existing applications with new source code based on the U format. The following rules have to be regarded for the U format (as compared with the A format):

- A REDEFINE of U to a format other than U should be avoided because this may result in split characters.
- U format is endian-dependant. This has to be considered when moving between the formats B and U.
- Align U in DEFINE DATA for performance reasons (better performance on UNIX and OpenVMS).
- Keep in mind that characters may be lost when moving U to A.

If you want to change existing fields from A format to U format, the following rules have to be regarded:

- Code which assumes a specific encoding of strings has to be changed (for example, comparison with a B field).
- All REDEFINE statements of the field have to be checked for their validity.
- A REDEFINE to N is not possible (that is: you will not get the expected result).
- The database field has to be migrated to Unicode (provided that this is supported by your database).
- You may have to change the length of the field: if the A field contains DBCS characters, half the length is required for the U field.

Migrating Natural Remote Procedure Calls (RPC)

The profile parameter `CP` has been renamed to `CPRPC`. In earlier Natural versions, `CP` was used to specify the name of the code page used by the Entire Conversion Service (ECS) and applied only to the Natural Remote Procedure Call when the transport protocol ACI (that is EntireX Broker) was used.

As of version 6.2 (Windows and UNIX), version 6.3 (OpenVMS) and version 4.2 (mainframe), a new `CP` parameter is available which defines the default code page for Natural data. When you are working with Natural RPC and have previously used the `CP` parameter dynamically, you have to change this parameter to `CPRPC`.

Windows, UNIX and OpenVMS Platforms

When you use parameter files from a previous version, you need not change anything; the Configuration Utility automatically migrates `CP` to `CPRPC`.

Mainframe Platforms

The parameter `CP` is used in conjunction with the parameter macro `NTCPAGE` (in the source module `NATCONFIG`) to specify the name of the default code page for Natural data or to automatically take the code page name from the user terminal.

The parameter `CPRPC` is used with the profile parameter `RPC` and the corresponding macro `NTRPC`.