

NOTE: This Technical Note has been [retired](#). Please see the [Technical Notes](#) page for current documentation.

Technical Note TE06

Changes in International Utilities and Resources

CONTENTS

[Introduction](#)

[INTL vs. itl](#)

[The International Sorting Routine](#)

[Error in APDA draft of Inside Macintosh Volume V](#)

[Bug in System 4.1 version of itl1](#)

[References](#)

[Downloadables](#)

[Mar 01 1988]

Introduction

The International Utilities package and the international resources have been changed with System file 4.1 to take advantage of the Script Manager.

[Back to top](#)

INTL vs. itl

In the past, there were two INTL resources in the System file, INTL 0 and INTL 1, which contained international formatting options. Starting with System 4.1, itl0 and itl1 replace INTL 0 and INTL 1. (INTL 0 and INTL 1 are still included with the System file so that applications that do a `GetResource` for them will still find them. Only these erroneous applications that use `GetResource` will access these resources now; the System will no longer look at them.) There can now be a set of international resources (itls) for each script that is installed. That is, where once there was one resource type (INTL) with IDs 0 and 1 there are now many different resource types (itl0, itl1, itl2, itlb, itlc, KCHR) with distinct resource IDs for the various countries. The U.S. System file uses resource ID 0 for all itl resources.

If your existing application calls `IUGetIntl` to get the appropriate international resource, you will have no problems under System 4.1. If, however, you call `GetResource` on INTL 0 or INTL 1 and then modify them, and expect that the System will use the modified resource, you will no longer work correctly. If your application has been released in the past with a modified System file to include your own INTL 0 and/or INTL 1, you may want to release your application with a modified itl0 and/or itl1.

[Back to top](#)

The International Sorting Routine

The international sorting routine is used to handle cases where letters are equal in primary ordering but different in secondary ordering (e.g., "ä" and "a"). The routine can also handle cases where one character sorts as if it were two, or two characters would sort as if they were one, or even character reversals.

Prior to System 4.1, the international sorting routine was stored starting with the `localRtn` field of INTL 1. As of

System 4.1, this routine is now stored in `itl2`. An `INTL 2` resource is also included with the international resources, just for consistency (the System never uses it). It is exactly the same as `itl2`.

Writing your own sorting routine can be dangerous if it is not done correctly. Before attempting to do this, we would highly recommend you check with Apple to determine if we are already doing one for the language of interest. We plan to do many international versions of System file 4.1, and most developers' needs will be met by these. A new policy at Apple will now make it possible to license foreign versions of the System file from Software Licensing. Contact Software Licensing at (408) 973-4667 for more information about this.

[Back to top](#)

Error in APDA draft of Inside Macintosh Volume V

There is an error in the APDA draft of the International Utilities Package chapter of *Inside Macintosh Volume V*. It says that there is a flag in the Script Manager globals that overrides the current font and always uses the `INTL 0` and `INTL 1` resources. This is not true. `INTL 0` and `1` have been replaced by `itl0` and `itl1`, and are not used except by applications that explicitly get them by doing a `GetResource`. There is a relevant Script Manager flag called `IntlForce` that is correctly but incompletely explained in the Script Manager chapter. When `IntlForce` is false, the resources used by the International Utilities are determined by the current script. The script that is in use is determined by which font is in use for the port in use. For example, if you switch to the Kyoto font, then you will be using the Japanese Script Interface System (KanjiTalk). If you do this with `IntlForce` set to false, then you will use the resources that are associated with the Japanese Script Interface System.

When `IntlForce` is true, the International Utilities always use the resources that are associated with the system script; this is usually Roman. `IntlForce` is true by default. Actually, the Script Manager initialization routine reads the `itlc` resource to determine what the default is, and Apple plans to release the various international System files with this field set to true in the `itlc` resource. Applications that want to change the value of `IntlForce` can use the Script Manager call `SetEnviron` with the `smIntlForce` verb. There is a Script Manager call, `IntlScript`, to find out the current value.

[Back to top](#)

Bug in System 4.1 version of itl1

There is a bug in the `itl1` resource in System 4.1. The `itl1` resource contains arrays of day and month names, as did the `INTL 1` resource. Their formats are:

```
days ARRAY [1..7] of STRING[15]
```

Every day and month is supposed to be coded as a Pascal string with a maximum of 15 characters, and the actual length in the first byte of the string. In System 4.0 and System 4.1, the contents of each string is always a Pascal string of length 15, with the unused characters set to nulls. In other words, the length byte is set to hex 0F. This will be fixed in a future release of the System file. If it is currently causing your application problems, you can change the lengths with `ResEdit`, or change them at run time, or use your own `itl1` resource and release your application with an installer script. The recommended approach is to wait for a fixed version of the System file from Apple.

[Back to top](#)

References

The International Utilities

The Script Manager

[Back to top](#)

Downloadables



Acrobat version of this Note (K).

[Download](#)

Technical Notes by [Date](#) | [Number](#) | [Technology](#) | [Title](#)
[Developer Documentation](#) | [Technical Q&As](#) | [Development Kits](#) | [Sample Code](#)