

Re: National Language – question on alphabet/sort order

Source: <http://www.tech-archive.net/Archive/VC/microsoft.public.vc.mfc/2007-05/msg01889.html>

- *From:* "Mihai N." <nmihai_year_2000@xxxxxxxxxx>
 - *Date:* Thu, 24 May 2007 22:57:10 -0700
-

On our engineering drawings there are often several sectional views and these are labelled A–A, B–B etc.
So in our code which automatically generates these sectional views there is a starting dialog which has a box for the section label ("A" in this case) and the code increments the next character until "Z" and wraps round to "A" again. (Highly unlikely that there will be more than 26 sections on one drawing but the start letter `_could_` be near the end of the alphabet and a wrap is needed)
and Yes this particular increment function assumes the English ASCII code sequence! :-(

Have you guessed my question yet?

Is there a method of incrementing and wrapping non-English alphabets?
Or : of identifying first and last characters in a set? (Sort sequence?)
European and Cyrillic alphabets might be straight forward, but obviously Asian sets are a different kettle of fish ("whole new ballgame" in modern vernacular)!

Well, there are several problems here.

The first one is that in many places the labelling rules might be different. So if A–A, B–B is the normal thing in the US, in other places it might be A1, B2, or 1–1, 2–2, etc.
This has little to do with the alphabet, but with local engineering/design rules.

So I would start by doing a bit of research to see how things are done in the potential markets that you care about.

Then, moving to languages, each might have it's own challanges. I doubt that for Japanese they use radical-stroke counts for what you need, most likely they use Gojuon (<http://en.wikipedia.org/wiki/Goj%C5%ABon>) or Iroha (<http://en.wikipedia.org/wiki/Iroha>), but they can also use numbers (http://en.wikipedia.org/wiki/Japanese_numerals), or something else.

Re: National Language – question on alphabet/sort order

You might have to use locale-specific digits.

Some conventions are also used for numbering lists
(<http://www.w3.org/TR/CSS21/generate.html#lists>)
that might also affect this.

I also doubt that accented characters are used for this even when the language uses them.

But in some languages you might skip non-accented characters that are not really used in the language (except for borrowed words).

For instance in Romanian you will probably not use Q, W, Y (but check this!)

Anyway, long story short, you need something very flexible.

An idea is this: use two different strings, both of them easy to change from UI or config file. Then iterate thru each character. I would also have a pattern that allows switching parameters (like FormatMessage)

```
CString label;  
WCHAR *str1 = ???; // load from config/ui/resources  
WCHAR *str2 = ???; // load from config/ui/resources  
WCHAR *pattern = ???; // load from config/ui/resources  
for( int i = 0; str1[i]; ++i )  
for( int j = 0; str2[j]; ++j )  
label.FormatMessage( pattern, str1[i], str2[j] );
```

Examples:

```
str1 = "ABCDEFGHJKLMNOPQRSTUVWXYZ";  
str2 = "0123456789";  
pattern = "%1!s!-%2!s!";  
will give you "A-0" "A-1" "A-2" ... "B-1" ...
```

```
str1 = "0123456789";  
str2 = "0123456789";  
pattern = "[%1!s! - %2!s!]";  
will give you "[1 - 1]" "[1 - 2]" ... "[2 - 1]"  
and so on.
```

The initial market study might show you that you need more flexibility, so you might end up with more than one character as primary element. In this case a string will not be enough, you might need an array of strings.

```
TCHAR str1[] = {  
"AA",  
"AB",  
"AB",  
...  
"ZY",  
"ZZ"  
};
```

Re: National Language – question on alphabet/sort order

```
TCHAR str2[] = {  
"00",  
"01",  
...  
"98",  
"99"  
};
```

So the labels might look like this: "AA-00" "AA-01" ... "BA-00" ... "ZZ-99"

And in fact I would also store this in resource, but in a config file.

The convention use should not be tied to the language of the UI
(a French application should not prevent from using Letter paper :-)

And you might even allow the user to change it thru the UI
(same as Word allows you to print to custom paper).

Now, this is not a 100% safe solution, but is quite flexible.

A quick market research should tell you if it is flexible enough or not.

--

Mihai Nita [Microsoft MVP, Windows – SDK]

<http://www.mihai-nita.net>

Replace _year_ with _ to get the real email

.