

A Note on Entering Unicode into a Form Field

AcroTeX.Net
April 15, 2007

D. P. Story

dpstory@acrotex.net
www.acrotex.net

1. The Basics

This document demonstrates how to use some of the new commands that come with AeB Pro, and some of the new keys in the eforms package that are designed to enter unicode hex encoding. In all cases, you must have the font that contains the glyph represented by the unicode hex.

You need AeB Pro with the `linktoattachments` and the `eforms` package, see preamble above.

The first step is to use the `\labelName` command to define your unicode string and associate a label with it. In the preamble, the following definition can be found.

```
\labelName{coolDPS}{Donald\u0020\u20AC.\u0020Story}
```

The string can be a combination of characters from Basic Latin or Latin-1 Supplement (latin1 option for AeB Pro required). As described in the AeB Pro manual, the string can be entered using keyboard keystrokes, or unicode notation `\uXXXX`, as above. Spaces are eaten up, so you must enter the unicode for a space `\u0020`. The command `\labelName` creates a macro that contains the pure encoding of the string. You can use `\aref` to access this encoding. The Unicode for the string `Donald\u0020\u20AC.\u0020Story` is

```
FEFF0044006F006E0061006C0064002020AC002E002000530074006F00720079
```

The hex code is computed from a table of unicodes defined in AeB Pro, with allowances for enter a “raw unicode” using a `\u` escape sequence.

Next, create an AcroForm field, a text field, in this example, and use the command `\unicodeStr` to reference the label enclosed in matching parentheses, like so

```
\textField[\Ff\FfReadOnly\textSize{10}\textFont{Arial}\W0
  \uDv{\unicodeStr(coolDPS)}\uV{\unicodeStr(coolDPS)}\BC{ }
]{dps}{1in}{11bp}
```

The result of the above code is

The technique uses some of the code brought in by the `linktoattachments` option of AeB Pro, and a new command, named `\unicodeStr`, designed to make it simple to enter unicode encoded strings into a form field.

The above field is set to read only, the one below is not:

Edit the box above, then click on the reset button. (Notice the tooltip for the reset button.)

The eforms uses an optional first argument to change the appearance of a form field, add in

action (usually JavaScript action) or to initialize the field value. In the above form fields, several unicode counterparts to the standard keys were used.

- `\uV` and `\uDv` were used instead of `\V` and `\DV` for giving the field a value and a default value, respectively. The “u” versions are used when the string is in unicode.
- Button face text for the reset button was entered with `\uCA`, `\uAC` and `\uRC` rather than `\CA`, `\AC` and `\RC`. You can mix these two types together. The normal face caption could be defined by `\uCA` and the others defined by `\AC` and `\RC`.
- The “tooltip”, as used in the reset button, normally uses the `\TU` key. When the string is meant to be unicode, use the `\uTU` key.

2. More Advanced

For combo boxes and list boxes, things are slightly more complicated. In the example below, we define a combo box. First, define the appearance values of the combo box (the string that is seen listed in the combo box).

```
\labelName{myEuro}{\u20AC\u0020(Euro)}
\labelName{myYen}{\u00A5\u0020(Yen)}
\labelName{mySheqel}{\u20AA\u0020(Sheqel)}
\labelName{myPound}{\u00A3\u0020(Pound)}
\labelName{myFranc}{\u20A3\u0020(Franc)}
```

Then we can define our combo box. According to the PDF file format, unicode strings should be enclosed in angle brackets `<XXXXXXXXXXXX>`. In initializing the combo box below, the `\unicodeStr` command is used, but this time it is enclosed in angle brackets. (That is all the “u” versions of the keys do above is to automatically insert the angle brackets for you. Here we have to do it ourselves. (I suppose one could have a helper command, but you can handle it.)

```
\comboBox[\Ff\FfEdit\uDV{1}\V{Euro}\textFont{Arial}
\BG{0.98 0.92 0.73}\BC{0 .6 0}]{myCombo}{1in}{11bp}
{[(Euro)<\unicodeStr(myEuro)>]%
  [(Yen)<\unicodeStr(myYen)>]%
  [(Sheqel)<\unicodeStr(mySheqel)>]%
  [(Pound)<\unicodeStr(myPound)>]%
  [(Franc)<\unicodeStr(myFranc)>]
}
```

The result is

Very swave!

3. More Advanced Still

You can also enter raw hex yourself (without using `\labelName`) if you have the patience to do it, and in some cases, it is the easiest.

For the purpose of the table below, the following definition was given in the preamble:

```
\newcommand{\cS}[2]{%
  \textField[
    \Ff\FfReadOnly\BC{}\textSize{10}\textFont{Arial}
    \uDv{FEFF#2}\uV{FEFF#2}
  ]{#1}{20bp}{11bp}}
```

Again the `\uDv` and `\uV` keys are used but without `\unicodeStr`. In this case, we must do it ourselves. As was mentioned earlier, the “u” versions of these keys automatically insert the angle brackets. Inside the angle brackets is the raw unicode hex code. What wasn’t mentioned before is that the first four hex digits must be FFFF to signal the beginning of unicode. Here, we insert these digits ourselves, followed, through parameter replacement, by the unicode hex digits of the characters desired. The result is the table you see below. (Again the field is readonly, the table is the one that appears in manual of usage for the `JJ_Game` class (`jj_man.pdf`)).

Description	Unicode	Symbol	Description	Unicode	Symbol
Dollar Sign	\u0024		Lira Sign	\u20A4	
Pound Sign	\u00A3		Peseta Sign	\u20A7	
Currency Sign	\u00A4		New Sheqel Sign	\u20AA	
Yen Sign	\u00A5		Dong Sign	\u20AB	
French Franc	\u20A3		Euro Sign	\u20AC	

FYI: See http://jrgraphix.net/research/unicode_blocks.php?block=0 for a display of unicode characters.