Commenting

> **Copying & Pasting these conventions**
>
> Copies of these standards can be found within the Standards.fp7 file on this wiki.

Calculation commenting

- C style comments (/* */) and standard C++ comments (//) are both fine, though the former is discouraged within functions (even for multiple lines, repeat the // single-line comment).

Single comment steps within scripts

Use one or more spaces before any text is added to a script comment step. This increases the readability of the step by creating a white space margin while reviewing scripts.

`#  Use one or more spaces at the start of any script comment step to increase readability`	good
`#Not using spaces in front of comments makes them harder to read`	bad

Script commenting

Reading what a script does, what it affects, and how the script has changed is critical to understanding where it fits into the solution. The following guidelines can be used for commenting your scripts.

```
#
# ====================================================================
# Purpose:            Describe the purpose of the script [short version]
# Parameters:         one ; two ; three
#                     --------------------------
#                     $one = (enum) load, unload
#                     $two = (string) second parameter
#                     $three = (num) expected
#                     --------------------------
# Called From:        (script) any
# Author:             Matt Petrowsky
# Notes:              Additional information [long version]
# Revsion:            Last change: 10/10/10 by MP :: A single comment step with multiple lines
# ====================================================================
#
```

The **Revision** field within the comment is one single script step which contains the latest change date and a date or date/timestamp line for each change documented. Within the comment step it looks like the following.

```
 Revision:

Last change: 10/10/10 by MP
::
10/9/10 - MP - each line of history is within this one comment step
10/10/10 - MP - added the history field to the comments
```

The utility of the history field is being able to use the return key to open the comment dialog for review and close the dialog easily using the escape key.

> **Shortcuts for inserting date/time values**
>
> Using a string expansion program for inserting dates and times into your comments is very helpful. For example, typing **dts** can be expanded to the current **Date time short** equivalent. Find a list of these tools in Keyword expansion tools in Developer Tools

Custom function commenting

Custom functions should use, at minimum, the following header. Because the header of a custom function is multiple lines of text **within the calculation dialog box**, you can use spaces for indenting, instead of the proposed tabs. This makes the calculation header easier to read. **Tip:** many editors have a **co nvert spaces to tabs** and inverse functionality.

```
/**
 * =====================================================================
 * CustomFunction ( parameter1 ; parameter2 )
 *
 * PARAMETERS:
 *            @parameter1 (text) Input string
 *            @parameter2 (num) Numerical value
 * RETURNS:
 *            (bool) True or False based on proper
 *            evaluation
 * DEPENDENCIES:
 *            none
 * PURPOSE:
 *            Use this function in order to accomplish
 *            most wonderful things possible!
 * NOTES:
 *            none
 * REVISIONS:
 *            10/23/10 - Initial release
 *            10/27/10 - Modified to fix issue
 *            10/31/10 - No returning Boolean result
 * =====================================================================
 *
 */
```

Wrapping lines within the header comment

Using an external editor will make it easier to wrap individual lines to a fixed length.

Commenting conventions to borrow from

The above standards are adapted from the following sources

Wikipedia Doxygen Article
Doxygen Web Site