



# **Cascading Style Sheets CodeCount™**

## **Counting Standard**

*University of Southern California*

**Center for Systems and Software Engineering**

October , 2008

**Revision Sheet**

Date	Version	Revision Description	Author
10/2/2008	1.0	Original Release	CSSE
1/2/2013	1.1	Updated document template	CSSE

# Table of Contents

No.	Contents	Page No.
1.0	<a href="#">Definitions</a>	4
1.1	<a href="#">SLOC</a>	4
1.2	<a href="#">Physical SLOC</a>	4
1.3	<a href="#">Logical SLOC</a>	4
1.4	<a href="#">Data declaration line</a>	4
1.5	<a href="#">Blank line</a>	5
1.6	<a href="#">Comment line</a>	5
1.7	<a href="#">Executable line of code</a>	5
2.0	<a href="#">Checklist for source statement counts</a>	6

# 1. Definitions

- 1.1. **SLOC** – Source Lines of Code is a unit used to measure the size of software program. SLOC counts the program source code based on a certain set of rules. SLOC is a key input for estimating project effort and is also used to calculate productivity and other measurements.
- 1.2. **Physical SLOC** – One physical SLOC is corresponding to one line starting with the first character and ending by a carriage return or an end-of-file marker of the same line, and which excludes the blank and comment line.
- 1.3. **Logical SLOC** – Lines of code intended to measure “statements”, which normally terminate by a semicolon (C/C++, Java, C#) or a carriage return (VB, Assembly), etc. Logical SLOC are not sensitive to format and style conventions, but they are language-dependent.
- 1.4. **Data declaration line or data line** – All executable lines are contained within a class. Each class begins and ends with a curly brace. The following example shows how a class can be defined.

```
<tag name> {
    .....
}
```

Irrespective of how many executable lines are contained within the curly braces, the number of data declaration lines for the above example is 1. Therefore a data declaration line for CSS corresponds to those lines that define a class.

- 1.5. **Blank Line** – A physical line of code, which contains any number of white space characters (spaces, tabs, form feed, carriage return, line feed, or their derivatives).
- 1.6. **Comment Line** – A comment is defined as a string of zero or more characters that follow language-specific comment delimiter.  
CSS comment delimiters are “/\*” and end with “/\*”. A whole comment line may span one line and does not contain any compliable source code. An embedded comment can co-exist with compliable source code on the same physical line. Banners and empty comments are treated as types of comments.
- 1.7. **Executable Line of code** – These include the lines contained within the definition of a CSS class such as p, td etc or even user defined classes. Every executable line needs to end with a semicolon barring the last line before the container class is closed (using a curly brace).

## 2. Checklist for source statement counts

### PHYSICAL SLOC COUNTING RULES

MEASUREMENT UNIT	ORDER OF PRECEDENCE	PHYSICAL SLOC	COMMENTS
<b>Executable lines</b>	1	One per line	Defined in 2.4
<b>Non-executable lines</b>			
Declaration (Data) lines	2	One per line	Defined in 2.5
Comments			Defined in 2.6
On their own lines	3	Not included (NI)	
Embedded	4	NI	
Empty comments	5	NI	
Blank lines	6	NI	Defined in 2.7

### LOGICAL SLOC COUNTING RULES

NO.	STRUCTURE	ORDER OF PRECEDENCE	LOGICAL SLOC RULES	COMMENTS
R01	<tagname> { ... }	1	Count once.	Each tag is counted once. It can be an HTML tag or a user defined class name.
R02	.. {<keyword> : <value>; ... }	2	Count number of semicolon separated statements.	The last statement need not end with a semicolon.