**REVIEW OF MULTIPLE AMPERSANDS &&&**

Multiple ampersands can be used to allow the value of a macro variable to become another macro variable reference. The macro variable reference will be rescanned until the macro variable is resolved.

The following demonstrates how macro variables with multiple ampersands are resolved.

Symbol Table

|  |  |
| --- | --- |
| Macro Variable Name | Macro Variable Value |
| A | FREIGHT |
| B | PASSENGER |
| C | SPECIAL |
| CODE | A |
|  | |

We can use any number of ampersands (&) in an indirect macro variable reference.

It resolves the entire reference from left to right. if a pair of ampersands (&&) is encountered, the pair is resolved to single ampersand, then the next part of the reference is processed.

%let a = usa;

%let usa = america;

%let America = states;

%put &a &&a &&&a &&&&a &&&&&a &&&&&&a;

%put &a;   \* usa \*;

%put &&a;  \* usa \*;

%put &&&a; \* America \* ;

explaination : && resolves to single & (put in reserve)  resolve &a to usa, now with reserve & sas reads &usa , result is america.

%put &&&&a; \* usa . pairing && and && comes to && resolves to & \*;

explaination : && resolves to single & (put in reserve) again && resolve to single & (put in reserve) , now we have only a, no execution. in reserve we have && which is &a. again result is usa.

%put &&&&&a;

explaination : && resolves to single & (put in reserve) again && resolve to single & (put in reserve), now we have &a , resolve to usa. In reserve we have two ampersand, which is one (&) and now it is &usa, resolve to America.

%put &&&&&&a;

explaination : && resloves to single & (put in reserve) again && resolve to single & (put in reserve) , && resolve to single & (put in reserve), now we have only a , no execution. In reserve we have &&& so again result is America.

%put &&&&&&&a .... would resolve to state, because we would have three ampersand (in reserve) and &a resolve to usa. &&&usa resolve to &america and it would resolve to States.