To: Distribution
From: J. Falksen
Date: 10/19/78
Subject: Version 2 exec_com if-then-else-fi

I like the idea of the version 2 exec_com as specified in MTB-392. It looks like a good way to make an incompatible syntax change. I feel, however, that while a change is being made, the spectre of flow-of-control should be attacked. If not, then it seems as though a version 3 would become necessary.

First of all, I think that the if construct should look like this:

```
&if [...] &if [...] 
&then {line} &then {line} 
{line} ... {line} ...
&else {line} &fi 
{line} ... &fi 
```

It is very easy for convert_ec to change the existing &if's into this form. This form requires no &do-&end for its functioning. (I think PL/I made a mistake in its if construct.)

The issue of branching into an &if is easily explained. After having branched somewhere, if an &fi is encountered then there is no action to take. If an &else is encountered, then everything is skipped until the matching &fi is reached.

I have a macro processor which uses this type of conditional testing. I and others who have used it have found this form to be quite nice. Nesting is no problem.

Second, the need for a &do-&end is of use only if it is an iteration construct. I think this would probably be nice, but I am not attacking that problem since it represents no incompatible changes.

Multics Project internal working documentation. Not to be reproduced or distributed outside the Multics Project.