Using SysFunc with IFC

Replacement for Macro %IF

Ronald J. Fehd
Centers for Disease Control

SCSUG 2009

Outline

Sound Byte

Any sufficiently advanced technology
is indistinguisible
from magic

Arthur C. Clarke

Sysfunc and ifc replace macro %if

Sound Byte

Any sufficiently advanced technology
is indistinguisible
from magic

Arthur C. Clarke

Sysfunc and ifc replace macro %if

Macro %if

```
%macro TestThis
      (Condition = 0);
%if   &Condition          %then %do;
      * true;
      %end;
%else %if not &Condition %then %do;
      * false;
      %end;
%mend;
%TestThis
```

Data step function ifc

```
Do N = ., 0, 1, -1;
   Text = ifc( N
             , 'true'
             , 'false'
             , 'missing'
             ) ;
```

Testing IFC listing

N	Text
.	missing
0	false
1	true
-1	true

%Sysfunc

- Is.a Macro function
- Access data step functions
- Usage: %put %sysfunc(max,3,2);

Open Code

```
%sysfunc(ifc(&Condition.
             ,* true;
             ,* false;
         )   )
```

Assert: exists Data

```
%Let Data = sashelp.class;

%sysfunc(ifc(%sysfunc(exist(&Data.))
     ,%nrstr(%Put Note: exists &Data.;)
     ,%nrstr(%Put Note: not exist &Data.;
              endSAS;) ))
```

Usage Examples

- http://www.sascommunity.org/wiki/
- Conditionally Executing Global Statements
- Assertions

Summary
Magic

- Sysfunc macro function
- IFC data step function
- NrStr macro function delays resolution

Conclusion

- Sysfunc with IFC can replace small macros
- Yields easily-readable code

Author Information

Ronald J. Fehd **RJF2@cdc.gov**
Stat Software HelpDesk
SAS Site Rep
Centers for Disease Control
Atlanta, GA, USA

Presentation pdf: LaTeX Beamer class