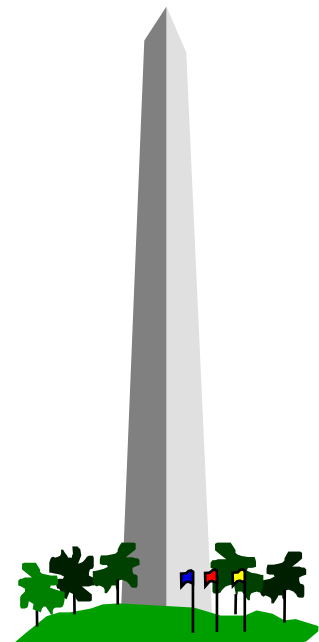




# Problem

- Many specification datasets being used by analysis applications
- Do not want engineers editing specs using FSEDIT or FSVIEW
- Changes need to be recorded for audits
- Flexible, easy to use data structure and interface needed



# Specifications Database

- Several Parameters per Spec
- Several Specs per Database
- Three Datasets per Database
  - **xxxxxREV** - Revision Control  
Vars: *spec, rev, date, time, who, specdesc, obsolete*
  - **xxxxx** - Current Specifications  
Vars: *spec, rev, \_var\_, vardesc, \_lsl\_, \_usl\_, \_target\_*
  - **xxxxxOBS** - Obsolete Specs  
Vars: *spec, rev, \_var\_, vardesc, \_lsl\_, \_usl\_, \_target\_*
- Datasets related by variables  
spec and rev

# Create Database Interface

- Only two fields need data
- Spec Dataset
  - 5 character name (xxxxx)
  - xxxxxREV and xxxxxOBS datasets created automatically
- Format for Values
  - Defaults to BEST7.
- Does not allow existing databases to be overwritten

# Specification Revision Dataset (xxxxxREV)

---

- Spec - Arbitrary name (\$8)
- Rev - Revision number
- Date - Date when revised
- Time - Time when revised
- Who - Who last revised
- Specdesc - Long description
- Obsolete - Boolean indicating if revision is obsolete or not

# Specification Parameter Dataset (xxxxx)

- Spec - Arbitrary name (\$8)
- Rev - Revision number
- Var - Variable name
- Vardesc - Parameter descript.
- \_lsl\_ - lower spec limit
- \_usl\_ - upper spec limit
- \_target\_ - spec target
  
- Dataset xxxxxOBS stores  
obsoleted specifications

# Display / Edit Interface

- Control objects trigger POPMENU selection lists
- Invalidly keyed data trigger POPMENU selection lists
- Extended table for params
- Context sensitive push buttons
- Container boxes for easy manipulation of text labels

# Differences between Display & Edit modes

- *Delete Spec* control hidden
- *Add, Move, Del* controls for parameters visible
- *New* and *Print* push buttons grayed
- *Database* and *Specification* controls grayed
- *Variable, Lower, Target* and *Upper* editable in extended table
- Context sensitive push button, *Make Changes* or *Save/Go back*

# Mode Push Button

- Make Changes  
*(switch to edit mode)*
  - Increments revision, update datetime and who
  - Edit a WORK dataset
- Save / Goback  
*(switch to display mode)*
  - Move current parameter data to obsolete dataset
  - Make edited data (WORK) the current revision
  - Update Revision Control dataset



# Using the Specs

- Data 'ready to use' with PROC CAPABILITY
  - use *where statement* to apply necessary spec from applicable database
- Range checking
  - PROC TRANSPOSE data in question
  - use PROC SQL to compare data values to spec values where data.\_NAME\_ matches spec.VAR