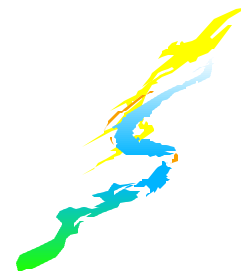
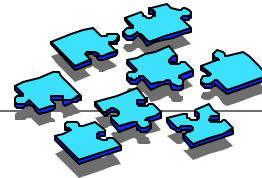




Help Development for Operations Navigator Plug-ins



Help Content

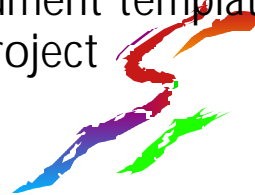


- Displays individual help topics for controls and dialogs
- Each control or dialog needs a separate html file
- Naming convention:
 - *dialog_name.html*
 - *dialog_name.control_name.html*
- Help content can link to other topics



Help Support

- Help Developers want large multi-topic files during development
- Allows a single multi-topic .html file for development -- called a HelpDocument
- Tool HDOC2HTM splits into individual help topics prior to ship
- GUI Builder generates a HelpDocument template for the controls and dialogs of a project



HelpDocument Conventions

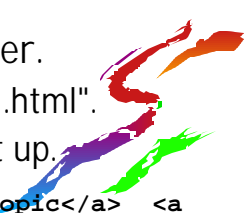
- Generated HelpDocument from GUIBuilder follows conventions
- Allows links to be active in the single file
- Allows additional topics to be created
 - Maintains active links
 - Links remain active after split-up
- Allows development using any HTML editor



HelpDocument Conventions (con't)

- Each help segment in the HelpDocument contains complete <body> content .html for a single topic file
- Upon split, a header and footer are added to complete the individual topic file
- Content for a topic is contained between tags
 - `<!-- HELPDOC:SEGMENTBEGIN --> <!-- HELPDOC:SEGMENTEND -->`
- An anchor tag immediately follows the
 - `<!-- HELPDOC:SEGMENTBEGIN -->`
 - Name of the anchor used as name of the topic file
 - Trailing ".html" required
 - Example:
 - `<!-- HELPDOC:SEGMENTBEGIN -->`
 - becomes the file: *MY_DIALOG.MY_CONTROL.html*

HelpDocument Conventions (con't)

- Each hyperlink reference in the HelpDocument uses an internal link notation to link to a topic anchor
 - Example: hyperlink to MY_DIALOG.MY_CONTROL using
 - `My info`
 - When the HelpDocument is split all href's ending with ".html" have the "#" removed
 - Convention allows links to be active in HelpDocument and separate files
 - Create true internal links in normal manner.
 - Internal links **restricted** from ending with ".html".
 - "#" not removed from internal links on split up.
 - Example: `Nice subtopic `
- 

Advanced HelpDocument Conventions

- Links to internal anchors of other topics possible

- Example:

- `Subtopic of mycontrol`

- Dialogs have only a single level name.

- Example:

- `My favorite dialog`



HelpDocument Generation

- In GUIBuilder: select Properties -> Generate Help
- Help template generated during next Save
 - For each dialog
 - For each control with a Generate Field Help set True
- Template contains
 - Header (can be overridden)
 - Segment for each control
 - Segment for each dialog
 - Footer



HelpDocument Regeneration

- If HelpDocument exists it is regenerated.
 - *myhelpdocument.html*
- Existing segments are preserved
 - Control segments copied exactly
 - Dialog segments updated with new controls
 - Controls added in list according to position
- Backup copy of original HelpDocument saved in *myhelpdocument.html.backup*



HelpDocument Regeneration


- If HelpDocument exists it is regenerated.
- Segments with PDMLSYNCH="yes" will be be synchronized with the latest level of the pdml.
 - removed if the control has been removed from PDML
 - Help writer added segments which have no corresponding control should set PDMLSYNCH="no"



Advanced HelpDocument Tags

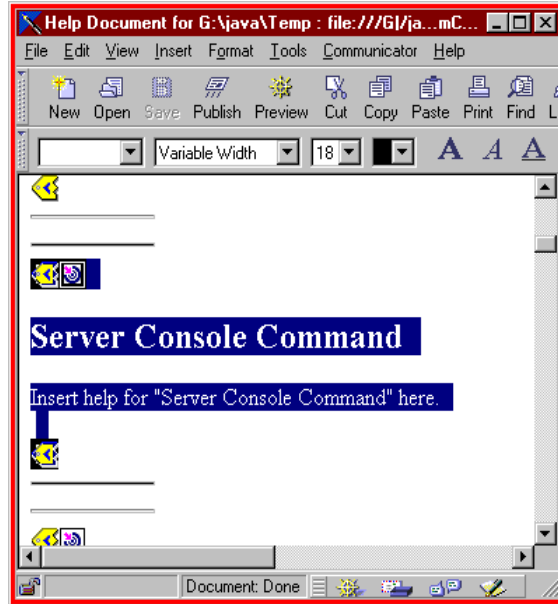
- Header of the HelpDocument is the default header used to build individual files
- Default footer is always `</BODY> </HTML>`
- Override default header or default footer on any segment
 - `header="headerfile.html"`
 - `footer="footerfile.html"`
 - Example:
 - `<!-- HELPDOC:SEGMENTBEGIN header="headerfile.html" -->`
- Override (replace) default header all default segments
 - Use `header="headerfile.html"` ON `HELPDOC:HEADEREND` tag

Editing the HelpDocument

- Can use any HTML editor
 - Tested with:
 - Netscape Composer
 - Dreamweaver
 - HR's (horizontal rules) placed before and after each segment
 - Not part of content
 - Convenience for cutting and pasting
 - Show visual beginning and end of segment
 - If you grab the HR's you grab the tags and anchors, too
- 

Adding your own segments

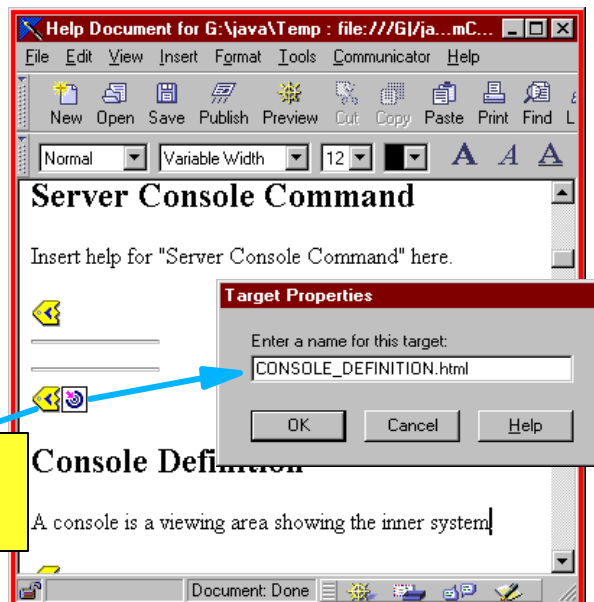
- Example: creating a general definition for "console"
- Copy an existing segment
- Get the Horizontal Rules!



Update the segment

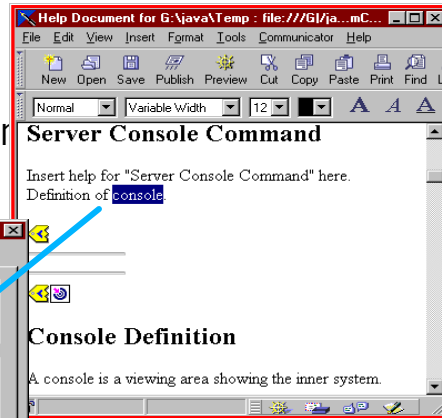
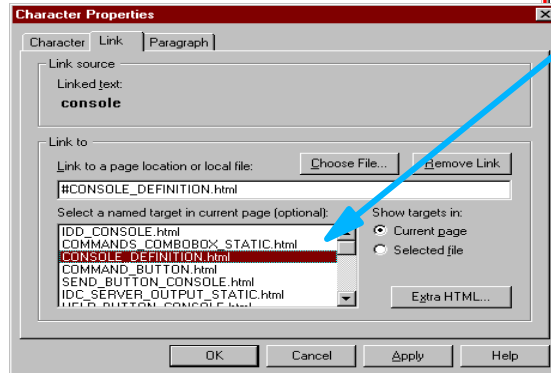
- Paste the copy (after an HR)
- Change the text
- Rename the anchor to desired .html file name (no # here!)
- Set **PDMLSYNCH="no"**

```
<--! HELPDOC:SEGMENTBEGIN
PDMLSYNCH="no" -->
```



Link to the Segment

- Create the text
- Create the link
- Select your anchor's segment - correct internal # notation



JavaHelp

- Help is displayed using the JavaHelp engine.
- JavaHelp is a new facility from Sun for providing HTML based help for Java applications and applets. JavaHelp is written entirely in Java, so consistent help is provided across all platforms.

JavaHelp components

- A HelpSet file that links all of the files together. The HelpSet file is a portal to all the information in the HelpSet.
- An XML based map file, which provides a symbolic name for the actual HTML file names and locations.
- An XML based table of contents referencing symbolic names.



JavaHelp components (cont'd)

- An XML based index referencing symbolic names.
- Search database files.

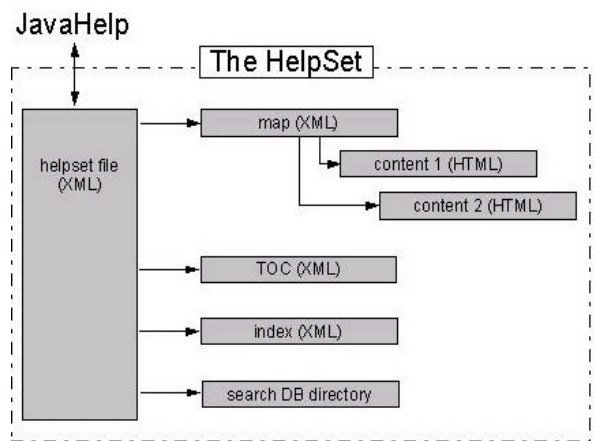
The writer creates and maintains the TOC, index, and HelpSet files.

The map file and search database are generated as part of the build.



JavaHelp components relationship

Relationship between the JavaHelp parts:



JavaHelp miscellaneous

- Interfaces to JavaHelp are provided through the HelpViewer class.
- In most cases, the programmer does not need to write the help calls - the framework does this for them.
- JavaHelp is available from Sun or the IBM Hursley Java site.

