

Oracle Corporation

Oracle Help for Java

File Formats for release 3.2/4.1

Revision 1.1

2/9/00

Introduction

Oracle Help for Java (OHJ) is Oracle Corporation's implementation of Sun Microsystems' JavaHelp™ standard. This document describes the OHJ file formats – formats based on the *JavaHelp™ 1.0 Specification* and implemented in the JavaHelp reference implementation. OHJ extends these file definitions to support additional features.

This document separates the help system files into three categories:

- Metadata Files - the JavaHelp HelpSet and JavaHelp Map files.
- Help Information - support files such as table-of-contents, index, etc.
- Content Files - actual help topic content files.

Each heading in this document is marked in the margin with a set of initials indicating from where the file format originated – **OHJ** for Oracle Help for Java and **JH** for JavaHelp. When both are listed together, then portions of the section were defined by Oracle Help for Java and portions were defined by JavaHelp.

Metadata Files

Two Metadata files define the Oracle Help for Java implementation of JavaHelp:

Helpset file	Identifies the map and navigational views.
Map file	Defines the association between topic ID (used by the application) and the content files (.htm/.html).

Helpset File (.hs)

JH,OHJ

The Helpset file points to a set of map files and defines an associated set of navigational views (such as the table-of-contents, index, and search). In new versions of OHJ, the HelpSet replaces the older notion of an OHJ HelpBook.

The format of the HelpSet file uses the World Wide Web Consortium Extensible Markup Language (XML) and consists of tags separating the file into sections – one for a title, one for topic ID maps, sections for navigational views, and sections for SubHelpSets.

OHJ's HelpSet file is an extended implementation of the HelpSet file described in the the *JavaHelp™ 1.0 Specification* and implemented in the JavaHelp 1.0 Reference implementation. The specific HelpSet extensions implemented by OHJ are:

- Addition of the Window Type and Link sections.
- Addition of the class attribute within the mapref tag of the Maps section, the linkref tag of the Links section, the data tag of the View section, and the subhelpset tag.

- Addition of the title tag in the View section.
- Addition of the engine attribute within the data tag of the View section.

JH

HelpSet Tag

The top-level tag in a HelpSet file is the <helpset> tag. Only one <helpset> tag is allowed in a HelpSet file.

The <helpset> tag as specified by the JavaHelp standard supports two attributes: **xml:lang** and **version**. Only **version** is supported by OHJ. The HelpSet definition outlined in this specification is version 1.1. (OHJ assumes control files are encoded in the default character set of the current default locale.)

JH

Title Section

The first section in the HelpSet – the Title section – names the HelpSet. For example:

```
<title>Developer Help</title>
```

In this example, the HelpSet is titled “Developer Help.”

When constructing Help Objects the programmer may choose to either combine HelpSets, or have each HelpSet appear in a drop-down list on the OHJ Navigator Toolbar. Users will then switch between Navigation Views for each HelpSet by picking from the drop-down list. The text that appears in the drop-down list is the contents of the <title> tag. In this example, “Developer Help” would appear in the list.

JH

Maps Section

The Maps Section is defined by the <maps> tag. This section contains information on how topic identifiers map to URLs (paths to HTML files). OHJ uses the map file specified in the JavaHelp 1.0 Specification, including the following tags:

<mapref> The location of a topic map file for this book. This tag includes a **location** attribute that specifies the URL of the map file relative to the HelpSet. Note that multiple map files can be specified within one map section. These map files are merged. Each <mapref> item is searched in order and a merged map is constructed. OHJ adds an optional **class** attribute. The location of this class is the base location for the map file. Any path information in the location attribute is relative to this base location. Note that any references to html files in this map file are relative to the map file's location.

A short sample Map section:

```
<maps>
  <mapref location="topics.xml" />
</maps>
```

OHJ

Wintype Section

Window Types are defined in the Wintype section of the HelpSet using the <wintype> tag.

A window type defines the appearance of the Help content window – size, screen placement, text color, background color, etc. Each topic in an OHJ Help system may contain a META tag specifying the window type to use to display the topic content. (See the "Content Files" section of this document for more details.) Window type is ignored if the Help system is using a web browser (such as Netscape Navigator) to display Help topics.

There may be any number of <wintype> sections; one for each window type and each with a unique name (specified by the <name> tag). The <wintype> tag carries one valid attribute – **default**. If the **default** attribute is true, the current window type is the default window type that should be used if an HTML content file does not designate a window type. If the **default** attribute is false, then this window type is not a default and will not be used except when explicitly listed in an HTML content file. When the **default** attribute is not present, its value is assumed to be false.

Valid tags in a <wintype> section are:

- <name>** The name of this window type. This name is used to associate topics with the window type.
- <height>** The height of the window. A numeric value indicates a distance in pixels. A numeric value followed by a percent sign (%) indicates a percentage of the visible screen.
- <width>** The width of the window in pixels. A numeric value indicates a distance in pixels. A numeric value followed by a percent sign (%) indicates a percentage of the visible screen.
- <x>** The horizontal position of the window. A numeric value indicates a distance in pixels. A negative value indicates distance between the right edge of the window and the right edge of the screen. A numeric value followed by a percent sign (%) indicates a percentage of the visible screen.
- <y>** The vertical position of the window in pixels. A numeric value indicates a distance in pixels. A negative value indicates distance between the bottom edge of the window and the bottom of the screen. A numeric value followed by a percent sign (%) indicates a percentage of the visible screen.
- <textfg>** The six-digit hexadecimal RGB value of the foreground color of the text in the window. A preceding '#' on the value may be present, but is ignored. For any single topic, a foreground color specified in the HTML topic file or CSS will override this value.
- <linkfg>** The six -digit hexadecimal RGB value of the foreground color of the links in the window. A preceding '#' on the value may be present but is ignored. For any single topic, a link color specified in the HTML topic file or CSS will override this value.
- <bg>** The six -digit hexadecimal RGB value of the background color of the window. A preceding '#' on the value may be present but is ignored. For any single topic, a background color specified in the HTML topic file or CSS will override this value.
- <title>** Text that appears in the title bar of the window. A <title> tag in the HTML topic file will override this text.
- <toolbar>** Defines the buttons to display in the window's toolbar. A five-digit hexadecimal value which is the sum of one or more of the following:

- 00004 Turns off the default buttons on the toolbar.
- 00040 Adds a URL address textfield – to display the current URL.
- 00400 Add the Navigator button.
- 02000 Add the Print button.
- 04000 Add the Back and Forward buttons.
- 08000 Add the Search button.
- 10000 Add the Dock/Undock buttons.

A preceding '#' on the value may be present but is ignored.

Note that attributes specified in HTML topic content files always take precedence over attributes specified in the <wintype> section of the HelpSet.

A short sample <wintype> subsection:

```
<wintype default=false>
  <name>Tutorial</name>
  <height>50%</height>
  <width>200</width>
  <x>10</x>
  <y>10</y>
  <textfg>000000</textfg>
  <linkfg>0000cc</linkfg>
  <bg>ffffff</bg>
  <title>Tutorial</title>
  <toolbar>06004</toolbar>
</wintype>
```

This example defines a window type called “Tutorial” with a default height that is 50% of the screen height and a default width of 200 pixels. This window will display 10 pixels from the top and left side of the screen. The background of the window will be white with black text and blue links. The title bar of this window will read “Tutorial.” The window’s toolbar will contain the Print, Back, and Forward button.

OHJ

Links Section

The Links section of OHJ’s HelpSet is defined by the <links> tag and contains the relative location of the link file used for Associative links. Each <linkref> item in the Link section is searched in order and a merged link database is constructed. The <linkref> item also supports an optional **class** attribute. The location of this class is the base location for the link file. Any path information in the location attribute is relative to this base location.

For example:

```
<links>
  <linkref location="linkfile1.xml" />
  <linkref location="linkfile2.xml" />
</links>
```

In this example, the files “linkfile1.xml” and “linkfile2.xml” contain the link items and entries used to display lists of links to related topics. See the section on the Link File for more information.

JH,OHJ

View Section

This section contains information on the navigational views defined by the HelpSet: the table-of-contents views, keyword index views, and search views.

As is defined in the JavaHelp specifications, a HelpSet may include multiple views of each type. OHJ merges all views of the same type and with the same label. That is: table-of-contents views with the same type and label are appended together; index views with the same type and label are merged in a sorted order; and search views with the same type and label are merged in no order -- order is irrelevant, since all search results are sorted by the OHJ user interface.

OHJ understands all tags of the JavaHelp Navigational Views Section as specified in the JavaHelp 1.0 specification:

<i>JH</i>	<name>	The name of the view. OHJ ignores this tag and its contents.
<i>JH,OHJ</i>	<label>	The label displayed on the navigator tab. Label is optional. If no label is provided, OHJ will use "Contents," "Index," and "Search" for the tab labels for the TOC, Index, and Search Views. Views with identical labels and identical view types are merged by OHJ. The label tag supports an optional image attribute which is set to the map file ID of the image that is to appear next to the label text in the navigator tab.
<i>OHJ</i>	<title>	A title for the view. This is displayed to the user as the top-level node in the TOC view with all top-level nodes in the TOC definition being collapsed under the title node. If a title is not provided for a TOC view, then all top-level nodes in the TOC definition will appear as top-level nodes in the TOC view. Also used for Search views as the location of any documents found in the given search index file that is displayed in the search results. If a title is not provided for a Search view, then the location in the search results is blank. Not currently used by the OHJ Index view. The title tag supports an optional image attribute which is set to the map file ID of an image. If provided for a TOC view, the image will be used as the TOC icon for the top-level node. If provided for an Index or Search view, the image is ignored.
<i>JH,OHJ</i>	<type>	The name of the Java class that will be used as the user interface for this view. The following types are provided with the default implementation: oracle.help.navigator.tocNavigator.TOCNavigator oracle.help.navigator.keywordNavigator.KeywordNavigator oracle.help.navigator.searchNavigator.SearchNavigator (If the JavaHelp types are specified, OHJ will map these correctly to OHJ types.)
<i>JH,OHJ</i>	<data>	Specifies the path to the data used by the particular type of view. This may be a filename or URL. OHJ adds an optional class attribute. The location of this class is the base location for the data file. Any path information is relative to this base location. Note that any references to html file names used in the view's data are relative to this base location.

OHJ also extends the use of the **engine** attribute to designate the other supported file formats. This attribute specifies the name of the class which will process the data file to provide the content for the designated navigator view. OHJ will use this tag to support both the JavaHelp formats and the HHC/HHK file formats supported by earlier versions of OHJ. The following table lists valid **engine** values for each View type: (default values are in bold)

Type `oracle.help.navigator.tocNavigator.TOCNavigator`:

engine	description
<code>oracle.help.engine.XMLTOCEngine</code>	OHJ XML table-of-contents (extension of JavaHelp TOC)
<code>oracle.help.engine.HHCEngine</code>	Microsoft HTMLHelp table-of-contents
<code>oracle.help.engine.TOCENGINE</code>	OHJ table-of-contents

Type `oracle.help.navigator.keywordNavigator.KeywordNavigator`:

engine	description
<code>oracle.help.engine.XMLIndexEngine</code>	OHJ XML keyword index (extension of JavaHelp keyword index)
<code>oracle.help.engine.HHKEngine</code>	Microsoft HTMLHelp keyword index
<code>oracle.help.engine.TOKENGINE</code>	OHJ keyword index

Type `oracle.help.navigator.searchNavigator.SearchNavigator`:

engine	description
<code>oracle.help.engine.SearchEngine</code>	OHJ full-text search database

JH

SubHelpSet Section

OHJ will allow for HelpSets to be included in other HelpSets. Note that views in SubHelpSets with the same type and label will be merged.

The subhelpset section supports the **location** attribute which specifies the URL of the HelpSet to be merged. OHJ adds an optional **class** attribute. The location of this class is the base location for the subhelpset file. Any path information is relative to this base location.

Sample HelpSet

```
<?xml version='1.0'?>
<helpset version="1.1">

<maps>
  <mapref location="topics.xml" />
</maps>

<wintype>
  <name>Tutorial</name>
  <height>50%</height>
  <width>200</width>
  <x>10</x>
```

```

        <y>10</y>
        <title>Tutorial</title>
        <toolbar>06004</toolbar>
    </wintype>

    <links>
        <linkref location="linkfile1.xml" />
        <linkref location="linkfile2.xml" />
    </links>

    <view>
        <label image="tocgif">Table of Contents</label>
        <title image="uggif">Forms User's Guide</title>
        <type>oracle.help.navigator.tocNavigator.TOCNavigator</type>
        <data engine="oracle.help.engine.XMLTOCEngine">ugcontents.xml</data>
    </view>

    <view>
        <label>Keyword Index</label>
        <type>oracle.help.navigator.keywordNavigator.KeywordNavigator</type>
        <data engine="oracle.help.engine.XMLIndexEngine">ugindex.xml</data>
    </view>

    <view>
        <label image="tocgif">Table of Contents</label>
        <title image="dggif">Forms Developer's Guide</title>
        <type>oracle.help.navigator.tocNavigator.TOCNavigator</type>
        <data engine="oracle.help.engine.HHCEngine"
            class="oracle.forms.documentation.dev">
            dgcontents.hhc
        </data>
    </view>

    <view>
        <label>Keyword Index</label>
        <type>oracle.help.navigator.keywordNavigator.KeywordNavigator</type>
        <data engine="oracle.help.engine.HHKEngine"
            class="oracle.forms.documentation.dev">
            dgindex.hhk</data>
    </view>

    <view>
        <label>Search</label>
        <title>Forms Documentation</title>
        <type>oracle.help.navigator.searchNavigator.SearchNavigator</type>
        <data engine="oracle.help.engine.SearchEngine"
            class="oracle.forms.documentation.dev">
            search.idx</data>
    </view>

    <subhelpset location="prodsup.hs" />
    <subhelpset location="advanced.hs" />

</helpset>

```

Map File (.xml)

JH

A map file is used to associate Help topic IDs with the HTML files containing the Help content and with Window Types defined in the HelpSet. These IDs and the map file are used in the API for context-sensitive calls, in the table-of-contents files, and in index files. The format of the map file for OHJ is similar to the file format outlined in the JavaHelp 1.0 Specification. OHJ has added the **wintype** attribute to the <mapID> tag.

The map file provides a level of indirection that allows developers to program topic IDs that can be associated with actual content files later in the development process. The map file

also makes upgrading a Help system less problematic. (Most authoring tools today generate the content HTML files from source files by automatically creating and assigning file names. When an author updates a Help system, perhaps adding or subtracting a topic, these file names may become out-of-sync. The map file provides a way for Help authors to match old topic IDs with new HTML content files.)

Valid characters for topic IDs or file names are any alphanumeric character or punctuation symbol except the equals sign. Topic IDs should be unique within the HelpSet.

Relative URLs are resolved against the absolute URL for the map file.

An example file follows:

```
<?xml version='1.0' ?>
<map version="1.0">

  <mapID target="topic1" url="file1.html" />
  <mapID target="topic2" url="file2.html#a1" />
  <mapID target="t3" url="file3.html" wintype="intro" />
  <mapID target="t4" url="file4.html#a2" wintype="intro" />
  <mapID target="t5.tsk" url="file5.html" wintype="task" />
  <mapID target="t5.cncpt" url="file5.html" wintype="concept" />
</map>
```

The map file also allows authors to assign Window Types to topic IDs and URLs. In the example above, mapID “topic1” and “topic2” do not have an associated window type and therefore use the default window type defined in the HelpSet. MapIDs “t3” and “t4” map to HTML content files that will be displayed in the “intro” window type. MapID “t5.tsk” will display “file5.html” in the window defined by the “task” window type. MapID “t5.cncpt” will display the same content file (“file5.html”) in a different window type (“concept”).

Note also that the association between URL and wintype will be used when linking from topic to topic using URLs instead of topic IDs. For example, if a topic had a hard-coded target to “file5.html”, clicking the link would display the HTML content in a “task” window type.

This scheme allows authors to assign window types to HTML files, and also override those associations by linking to an alternate topic ID. For example, for help system topic-to-topic links, TOC links, Index links, and hard-coded links to “file5.html” the author might use t5.tsk, but for links from an Introductory Tutorial, the author might use t5.cncpt.

By keeping this information in the map file, the author has one central repository for managing these assignments.

Help Information

Help information consists of:

Table of contents file	Defines the table-of-contents entries and their associated files.
Index file	Defines the index entries and their associated files.
Search file	Contains the database used for full-text search.
Link file	Contains the link entries for associative links.

Table-of-Contents File (.xml)

JH, OHJ

The table-of-contents (TOC) file describes to the TOC View the content and layout. This file uses XML and follows the TOC file format described in the JavaHelp 1.0 Specification. Note that the OHJ TOC file does not support the **xml:lang** attributes.

The following is a very small example of a TOC file:

```
<?xml version='1.0'?>
<toc version="1.0">
<tocitem text="Introduction to My Product">
  <tocitem target="aboutmyp" text="About My Product" />
  <tocitem target="architec" text="My Product Architecture" />
  <tocitem target="feature1" text="About Feature One" />
  <tocitem target="feature2" text="About Feature Two" />
  <tocitem target="feature3" text="About Feature Three" />
  <tocitem text="Learning My product">
    <tocitem target="qtour" text="Quick Tour" />
    <tocitem target="docguide" text="Guide to Documentation" />
  </tocitem>
</tocitem>
<tocitem text="Step-by-step Procedures">
  <tocitem text="Basic Tasks">
    <tocitem target="task1" text="To ..." />
    <tocitem target="task2" text="To ..." />
    <tocitem target="task3" text="To ..." />
  </tocitem>
  <tocitem text="Working with ...">
    <tocitem target="task4" text="To ..." />
    <tocitem target="task5" text="To ..." />
  </tocitem>
</tocitem>
</toc>
```

This definition produces the following table-of-contents tree:

```
+ Introduction to My Product
  o About My Product
  o My Product Architecture
  o About Feature One
  o About Feature Two
  o About Feature Three
+ Learning My Product
  o Quick Tour
  o Guide to Documentation
+ Step-by-step Procedures
  + Basic Tasks
    o To ...
    o To ...
    o To ...
  + Working with ...
    o To ...
    o To ...
```

Table-of-Contents Tags

The following are descriptions of the TOC tags supported by OHJ.

- <toc>** Defines the TOC -- contains <tocitem> tags.
- <tocitem>** Defines a TOC entry. Nesting *entry1* within *entry2* defines *entry1* to be hierarchically contained within *entry2*. The **target** attribute specifies the topic ID of the content file to display when the entry is chosen by the user.

IDs are defined (associated with a URL) in the map file. The **image** attribute is supported. Note that the **image** attribute specifies an id that the map file resolves to an image file. The **xml:lang** attributes specified in the JavaHelp 1.0 Specification are not supported by OHJ and will be ignored.

Index File (.xml)

JH, OHJ

The index file describes to the index navigator the content and layout of the index. This file uses XML and follows the index file format described in the JavaHelp 1.0 specification. OHJ adds the <indexentry> tag.

The following is a small example of an index file:

```
<?xml version='1.0' ?>
<index version="1.0">

  <indexitem target="Add_Icon"
    text="Add Icon">
    <indexitem target="addtosheet"
      text="adding an icon to a sheet" />
    <indexitem target="addtobook"
      text="adding an icon to a workbook" />
  </indexitem>

  <indexitem target="Sheet_Background"
    text="adding a background to a sheet" />

  <indexitem text="adding a new sheet to a workbook">
    <indexentry target="New_Sheet_command"
      text="New Sheet command" />
    <indexentry target="Add_new_sheet"
      text="To add a new sheet to a workbook" />
  </indexitem>

  <indexitem target="Add_item_to_sheet"
    text="adding an item to a sheet" />
</index>
```

This produces the following keyword list:

```
Add Icon
  adding an icon to a sheet
  adding an icon to a workbook
adding a background to a sheet
adding a new sheet to a workbook
adding an item to a sheet
```

The 'adding a new sheet to a workbook' item, if selected, will return a topic list of:

```
New Sheet command
To add a new sheet to a workbook
```

This topic list appears at the bottom of the index pane, as opposed to "indented" topics ("adding an icon to a sheet" and "adding an icon to a workbook") which show up in the keyword list at the top of the index pane.

Note that selecting a keyword that does not have index entries but instead has a directly associated target (for example, "adding an item to a sheet") will display the same text in the topics list as it does the keyword list. Given the UI of OHJ's Keyword View, it is better practice to never use targets in <indexitem> tags and instead always use <indexentry>

tags to specify topics associated with an <indexitem> -- even when there's only one target for a keyword.

In other words, this:

```
<indexitem text="sheet backgrounds">
  <indexentry target="Sheet_Background"
    text="adding a background to a sheet" />
</indexitem>
```

Is better than this:

```
<indexitem target="Sheet_Background"
  text="adding a background to a sheet" />
```

Index Tags

The following are descriptions of the index tags:

<i>JH</i>	<index>	Defines the index. It can contain <indexitem> and <indexentry> tags.
<i>JH</i>	<indexitem>	Defines an index item that appears in the keyword list. Nesting <i>entry1</i> within <i>entry2</i> defines <i>entry1</i> to be hierarchically contained within <i>entry2</i> , listed and indented below <i>entry2</i> in the index. If a particular index item has more than one topic associated with it, then these topics should be listed as index entries contained by the <indexitem> tag. This tag uses the target attribute which specifies the ID of the content to display when the entry is chosen by the user. IDs are defined (associated with a URL) in the map file. OHJ currently supports only two levels of keywords. The OHJ index view will collapse any nesting beyond two levels.
<i>OHJ</i>	<indexentry>	Defines an index entry displayed in the topics list when the parent index item is selected in the index list. This tag uses the target attribute which specifies the ID of the topic to display when the entry is chosen by the user. IDs are defined (associated with a URL) in the map file. This tag is not supported by JavaHelp.

Link File (.xml)

OHJ

The link file is used by OHJ to generate the list of topics associated with a particular Associative Link keyword. This file uses XML and is unique to OHJ.

The following is a small example of a link file:

```
<?xml version='1.0' ?>
<link version="1.0">
  <linkitem topic="link1">
    <linkentry target="file1" text="Item in the first link" />
    <linkentry target="file2" text="Another item in the first link" />
    <linkentry target="file3" text="Last item in the first link" />
  </linkitem>
  <linkitem topic="link2">
    <linkentry target="topic1" text="Item in another link" />
    <linkentry target="topic2" text="A second item in the second link" />
  </linkitem>
```

</link>

With this example, if an HTML link is clicked that specifies "alink:link2" as the target, then a window will pop up containing the following links:

```
Item in another link
A second item in the second link
```

Clicking "Item in another link" will display the topic mapped to the "topic1" ID. (OHJ uses the HelpSet's map file to associate this ID with an HTML content file.)

Link Tags

The following are descriptions of the link tags:

- <link>** Defines the link file. Contains only <linkitem> tags.
- <linkitem>** The required **topic** attribute of the <linkitem> tag defines the ID that should be used to define the HTML link. (See "Associative Links" in the "Content Files" section for more information on the Topic popup window.) The <linkitem> tag can only include <linkentry> tags.
- <linkentry>** Defines an entry displayed in the Topic popup window when an A-link is clicked. This tag requires the **target** attribute which specifies the ID of the topic to display when the entry is chosen by the user. IDs are defined (associated with a URL) in the map file. The text contained in a <linkentry> tag is displayed to the user in the Topic popup window.

Search Index File (.idx)

OHJ

The Search Index file uses a proprietary binary format. Applications to create search index files are provided as Java executables.

Content Files

At this time, the Help display component that ships with Oracle Help for Java is ICESoft AS's ICE Browser. Version 5.01 of the ICE Browser is compliant with the HTML 4.0 standard and can display tables and frames as well as run Java Applets.

To support Topic Id Links, Associative Links, Popups, and Window Types, several OHJ specific HTML conventions are required.

Topic Id Links

OHJ

The ICE Browser Edition of OHJ supports links specified using either hard-coded html file names or topic IDs. For example:

```
<a href="topicid:topic1">Link to Topic 1</a>
```

When this link is clicked, OHJ will reference the HelpSet map file information and jump to the html file associated with the link's topic ID.

Associative Links

OHJ

Most HTML help authoring tools generate applet tags to support Associative or Related-Topics Links. For release 3.1 of OHJ, Associative Links will be supported through a link protocol. For example, an Associative Link that would display all the topics associated with the "worksheet" keyword would be specified with:

```
<a href="alink:worksheet">Related Topics</a>
```

OHJ will use the Associative Link keyword (in this case "worksheet") to search the merged link files and display a popup window with a list of related topics.

Note that this feature only works with the ICE Browser Edition of OHJ -- systems that use other HTML display engines should probably use applets to display related topics.

Popups

OHJ

The ICE Browser Edition of OHJ supports popups – useful for tips and glossary definitions. For example:

```
<a href="popup:sheetdefinition">Sheet Definition</a>
```

The keyword that follows the popup protocol specifies a topic ID. When the popup link is clicked, the contents of the file associated with the topic ID in the HelpSet's map file is displayed in a lightweight popup window.

Topic IDs

OHJ

To support topic ids and the generation of the map file, an author may use a META tag placed in each HTML content file. A **topic-id** META tag identifies the topic id for this topic. For example:

```
<META name="topic-id" content="topic1">
```

If the authoring wizard supplied with OHJ is used to process this file, the topic id will be read and used to construct the map file – associating this topic id with the actual topic file name. See the **Map File** section for more information.

Window Types

OHJ

OHJ also uses the map file to associate topics with window types. To support the authoring of window types, an author may use a META tag placed in each HTML content file. A **window-type** META tag marks the window type for this topic. For example:

```
<META name="window-type" content="tutorial">
```

When the Authoring Wizard is run, the window type will be read and used along with the topic id META tag to construct the map file. See the **Wintype, Topic IDs, and Map File** sections of the HelpSet file for more information.

Note that in previous version of OHJ, the OHJ display engine actually read and used this META tag. This is no longer the case – the map file is now the central repository for this information.

*Oracle Help for Java
File Formats for release 3.2/4.1
Revision 1.1, 2/9/00*

Please send comments to ohjinfo%oceania@us.oracle.com.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

All statements regarding future intent are subject to change or withdrawal without notice and represent goals and objectives only. Information in this document regarding non-Oracle products was obtained from the suppliers of those products, published announcement material, or other publicly available sources. The information contained in this document represents the current views of Oracle on the issues discussed as of the date of publication. Oracle cannot guarantee the accuracy of any information after the date of publication.

Oracle is a registered trademark, and other Oracle product names, service names, slogans or logos referenced in this paper are trademarks or registered trademarks of Oracle Corporation. All other company, product or service names referenced in this paper are used for identification purposes only and may be trademarks of their respective owners.

Copyright © 1999,2000 Oracle Corporation, All Rights Reserved.