

Reference: 2007/10/31-8.2

Outside In HTML Export Technology SDK Quick Start Guide

This document provides an overview of the Outside In HTML Export Software Developer's Kit (SDK). It includes download instructions, installation overview, architecture description and other topics that will help readers to get started working with the SDK. Pointers are given throughout to the Developer Guide and sample applications. Readers should also use the other documents available in the Outside In section of Oracle Technology Network.

Contents

Product Overview	3
Target Audience	3
Use Cases	3
Available Downloads	4
Installation	4
Directory Structure	5
Architecture	5
Integration	6
Information on common Issues	9

Product Overview

Outside In HTML Export converts the contents of more than 400 file types into HTML, rendering embedded graphics as a choice of GIF, JPEG, or PNG. Outside In HTML Export uses sophisticated template architecture, as well as a rich set of options, to allow a high degree of customization of the HTML output. When used in combination with Outside In Content Access or Outside In Search Export, SearchML, annotations, or redactions can be programmatically added to the HTML output.

Outside In HTML Export can use cascading style sheet formatting for more-complex formatting, or alternatively can be scaled down to use least-common-denominator versions of HTML for the broadest compatibility with browsers on various platforms or devices. Its ability to provide browser access to hundreds of file types without plug-ins or other proprietary applications makes it one of the most widely used SDKs of the Outside In Technology.

HTML Export is delivered as a Software Development Kit (SDK) with C based Application Programming Interfaces (APIs). It is available for a number of operating systems, listed below.

Target Audience

HTML Export is for Software Developers who wish to integrate HTML conversion into their applications.

Use Cases

HTML Export is used in applications that will benefit from HTML based content including:

- portals
- Web-based search engines
- CRM, CM, DM, KM, e-mail & unified messaging
- many other web-based applications

Available Downloads

The HTML Export SDKs for each supported platform are contained in archive files that can be downloaded from Oracle Technology Network. The link to download these files is available on the same page from where you downloaded this document, or from the link below:

http://www.oracle.com/technology/products/content-management/oit/oit_dl_otn.html

Each of the following downloads include all the files needed to evaluate/implement the HTML Export for that platform.

HP-UX (IA-64)

HP-UX 32-bit (PA-RISC)

IBM AIX 32-bit

Linux (IA-64)

Linux (IBM zSeries)

Linux (x86-32)

Linux (x86-64)

Solaris (Sun SPARC-32)

Solaris (x86-32)

Windows (IA-64)

Windows (x86-32)

Windows (x86-64)

Installation

To install the demo version of the SDK, copy the contents of the archive (available on the Web site) to a local directory of your choice.

For Windows versions, unzip the archive to the directory of your choice.

For UNIX versions of the SDK, copy the tgz file corresponding to your platform (to a local directory of your choice). Decompress the tgz file and then extract from the resulting tar file as follows:

```
gunzip tgzfile
```

```
tar xvf tarfile
```

Directory Structure

The installation directory contains the following directory structure:

<code>*</code>	Contains a working copy of the technology and compiled executables of the sample applications.
<code>\common</code>	Contains the C include files needed to build or rebuild the technology.
<code>\docs</code>	Includes HTML and PDF versions of the SDK Developer Guide.
<code>\hxfiles</code>	Contains sample files designed to exercise the technology.
<code>\lib (Windows Only)</code>	Contains the library (.lib) files for scca.dll , sccex.dll , and sccfi.dll .
<code>\resource</code>	Contains localization resource files.
<code>\samples</code>	Contains a number of subdirectories, each one holding the source code for a different sample application.
<code>\template</code>	Contains a number of sample templates designed to exercise HTML Export's template language

Architecture

The basic architecture of the HTML Export is the same across all supported platforms. The input filters form the base of the architecture. Each one reads a specific file format or set of related formats and sends the data to the normalization and caching module through a standard set of function calls. There are more than 150 of these filters that read more than 400 distinct file formats. Filters are loaded on demand by the data access module.

The normalization and caching module is responsible for caching a certain amount of data from the filter and returning this data to the export filter

The Export Filters are architecturally similar to input filters, and write out a specific format based on information coming from the normalization and caching module. The export filters generate HTML, GIF, JPEG, and PNG in this case.

The export module implements the export API and loads and runs individual export filters.

The Data Access module implements a generic API for access to files. It understands how to identify and load the correct filter for all the supported file formats. The module delivers to the developer a generic handle to the requested file, which can then be used to run HTML Export.

Integration

The best way to begin working with HTML Export is to examine the documentation and sample applications. From there you can begin to plan the integration of this technology into your own application.

The HTML Export Developer Guide

The Developer Guide for HTML Export provides more detailed information about getting started, including

- Implementation on Windows and Unix
- Data Access and Export Functions
- Messages
- Using Redirected I/O
- Options
- Sample Applications

The Developer Guide also contains a list of the filter libraries and the supported formats and platforms for the specific Outside In SDK.

HTML Export Sample applications

HTML Export includes the executables and source code for a number of sample applications. Use the sample application executables to see examples of some of the features of the SDK. The source code for each sample application will illustrate how that functionality is implemented.

HTML Export includes the following sample applications.

export (Windows Only)

This sample application allows the user to run a single source file through HTML Export. The user can choose the source file, an output file and set the various HTML Export options.

exsimple

This simple command line driven program allows the user to run a single source file through HTML Export. The user can choose the source file, an output file and set the various HTML Export options.

exredir

This sample application is based on the **EXSIMPLE** sample application. It is designed to demonstrate how to use redirected IO and callbacks in HTML Export.

hxanno

This sample application provides an example of how to integrate Content Access with HTML Export to perform search hit highlighting. It is provided more for the instructional value its sample code offers than for the functionality it provides when executed. The general principles of how to get ACC text positions from Content Access should be evident from perusing the source code.

hxsample

This sample application is an extremely basic implementation of HTML Export that uses all of the default settings for every option unless the option *template* parameter is used and a template is specified that includes overrides of default option settings.

ExJava / exporter

The **ExJava** Java wrapper, working in tandem with the **exporter** sample application, provides a working example of one method of interfacing with Oracle's C-based SDK products from a Java application. **Export.jar** is a Java API wrapper used by a Java application to control the **exporter** executable and set conversion options. **exporter** is a C-based executable which performs conversions using the modules in the Outside In Export SDK.

HTML Export Sample Templates

Much of the power, flexibility and complexity of HTML Export is realized through its use of templates to drive the export process. Templates give the developer (or the developer's customer) flexibility in the visual and navigational properties of the resulting output. Templates also isolate the HTML Export code from the ever-changing face of HTML and its associated plug-ins, components and scripting languages.

When you install HTML Export, a **template** directory is created which contains the following sample templates.

\template\standard

The standard template features convenient navigation elements, including a table of contents and a preview window, to help users quickly access a document's information.

\template\standard_mhtml

The **standard_mhtml** template is a version of the standard template that creates output to a single MHTML file.

\template\navigation

The navigation template has many of the same features as the standard template, such as convenient navigation elements, and adds drop-down table of contents.

\template\navigation_mhtml

The navigation_mhtml template is a version of the navigation template that creates output to a single MHTML file.

\template\newsletter

The newsletter template supports all document types except archives. It displays the content in a style similar to a news Web site. The table of contents contains each top level heading (the "Heading 1" style). When a user clicks these hyperlinks, the corresponding section's content fills the main window.

\template\newsletter_mhtml

The newsletter_mhtml template is a version of the newsletter template that creates output to a single MHTML file.

\template\noframes

The noframes template displays an entire document in a single frame, with table-of-contents style navigation. It is ideal for use in the most straightforward publishing applications.

\template\tableofcontents

The tableofcontents template is simpler than the standard or the navigation templates, and contains fewer navigation elements. It shows a table of contents on the left side of the screen, and the selected document content on the right.

\template\tableofcontents_mhtml

The tableofcontents_mhtml template is a version of the tableofcontents template that creates output to a single MHTML file.

\template\textonly

The textonly template is designed for use by developers wishing to convert documents for inclusion in an index for a search engine. It should not be used in publishing applications. All of the document's elements, including properties, headers and footers, are converted.

\template\tutorial

This is a directory of templates containing comment text intended to help users interested in more thoroughly understanding the HTML Export template language. These templates are described in detail in the Template Tutorials section of the Developer Guide.

Information on common Issues

The following sections of the Developer Guides for each SDK address issues that new users of the Outside In Technology often ask. You may want to read these sections in particular during your evaluation process.

The Basics

These sections of the Windows / UNIX Implementation Details chapters describe how to start calling the API, sending and receiving messages, and other functions specific to each SDK. Most of the topics covered in this section are illustrated by the sample application code.

Linux Compiling and Linking

This section of Unix Implementation Details chapter contains information about library compatibility concerns for your specific flavor of Linux

Runtime Considerations (UNIX)

This section of the Unix Implementation Details chapter contains information about running in a variety of UNIX environments. See especially the *X Server Requirement* and *System Fonts* sub-sections.

Running in a 24x7 environment

This section of the Implementation Issues chapter discusses process isolation when running Outside In Technology in 24 x 7 environments.