BASIS Offers Context Sensitive Help on the World Wide Web

By Nick Decker

With each passing year, computer application help systems evolve and improve to better assist users. Some of the more modern technological advances, such as the widespread availability of the Internet, increase access to online help systems. Other technological advances help to alleviate issues that arise due to the increasingly complex nature of computer applications. For instance, years ago it was common for help users to wade through thousands of pages of documentation compiled together in a weighty tome just to find the answer to a frequently asked question. While navigational tools, such as the table of contents and index, helped reduce the time spent finding information, plenty of room for improvement still existed.

Modern help systems now sport new features and constructs that substantially facilitate the retrieval of applicable and relevant information. For example, users can now search the same number of pages in a fraction of a second. This search returns a hyper linked list of relevant articles grouped and prioritized by subject. The links jump to pages in a local help system or to external help systems available around the world, providing the most up-to-date information possible. In addition, modern help systems can even translate these final documents into various languages on the fly!

Context Sensitive Help

Today's help systems contain exhaustive reference lists, descriptive overviews, sample training and usage guidelines, answers to frequently asked questions, and even task-oriented help. Task-oriented help exists to detail common procedures or to provide definitions and reminders for various user interface components. The help industry refers to this type of help as Context Sensitive Help (CSH) because it offers information specific to the task. CSH can be an extremely powerful function, as it bypasses all of the typical searches and instead provides a direct line into the help system resulting in an instant and topical answer.

How CSH Works

The construction of a suitable CSH-based system must take into account both the help system and the computer application. The core of a CSH system is defined by the relation and the binding of these two entities through common subject matter, or topics. Each section of the application, or perhaps even various controls in a user interface, has a corresponding section in the help system that describes its function. This relationship is explicitly defined by using shared topic IDs, which are either numeric or alphanumeric. For example, when a user wants to find the answer to a

question regarding a particular screen in an application, they can bring up context sensitive help on that exact screen. This type of help works by using a common topic ID. In other words, the application calls the help system and passes in the topic ID that uniquely identifies the particular screen. With just a click of a button, the user instantly locates the relevant topic.

Taking Advantage of BASIS CSH

Realizing the potential of a Web-based context sensitive help, BASIS recently added over 3,500 unique topic IDs to its online help system. The topics span 35 different sections of the documentation and cover a broad range of topics from Accelerators to Zooming. Topic IDs now exist for most everything in BASIS's online product documentation, including verbs, functions, error codes, and object syntax. In addition, the higher level subjects such as installation, configuration, and the BBj® Enterprise Manager now have topic IDs assigned to them. Moreover, BASIS published these topic IDs so users can now access them from any Web browser, help system, or even computer application written in several different languages.

CSH in Action

To illustrate the potential of CHS, consider the following scenario: A new employee needs access to the company's accounting database. The BBj Enterprise Manager (EM) controls this company's database, users, and permissions. The administrator needs to run the BBj EM, add the new employee to the list of users, and then set their permissions appropriately. Having guidance in the form of CSH available during this process can make the administrator's job much easier.

The traditional approach to this scenario is providing company-specific documentation on the BBj EM. This approach, albeit redundant, does not guarantee that the documentation keeps up with the rapidly evolving BBj EM. The new approach is to provide a link to the current BASIS online version of the BBj EM documentation from your own help system or the application itself.

To access the most current version the BBj EM online help, click on: **www.basis.com/onlinedocs/documentation/index.htm#<id=26**.

Clicking this link pops up a new Web browser window containing the help topic entitled "User Accounts" straight from the current BASIS online product documentation. BASIS recommends that you embed these links in platform-independent formats, such as HTML and Adobe Acrobat's Portable Document Format (.pdf).

* Note that the links used in this article invoke a new instance of your Web browser that displays the appropriate section of the BASIS online documentation. In order for CSH to work correctly, you must have an Internet connection and your browser configured to not block popup windows.
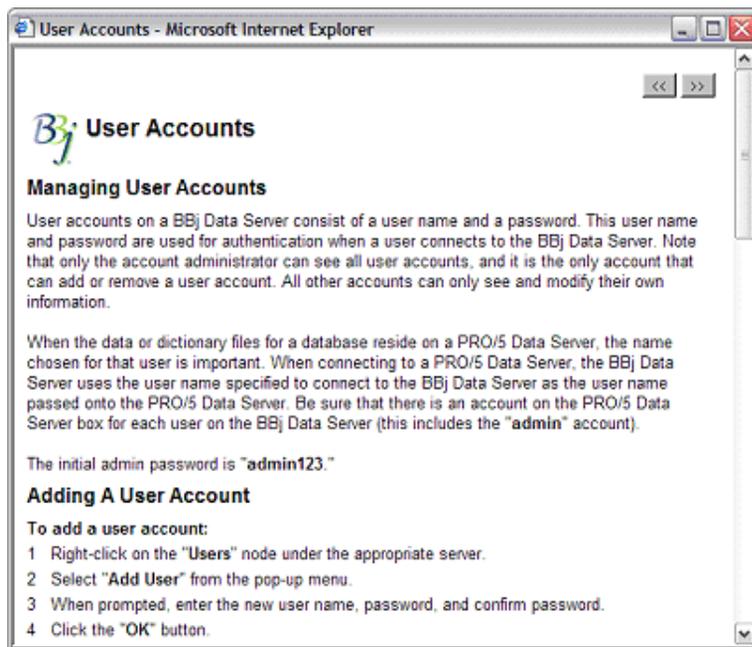

Figure 1. Resultant CSH Popup.

Additional Uses for CSH

There are many potential use cases for integrating BASIS's help into your application or help system. BASIS's online help system covers various subjects and concepts that directly relate to the configuration and performance of your application. All you need to do is add a link or a couple of lines of code and you can instantly have access to documentation detailing everything from adding and configuring printers to memory management and application optimization.

For example, you could take advantage the BASIS SETOPTS documentation by providing links into the SETOPTS verb or Configurator online help topics:

Configurator SETOPTS:
www.basis.com/onlinedocs/documentation/index.htm#<id=3047

SETOPTS Verb:
www.basis.com/onlinedocs/documentation/index.htm#<id=2094

Creating a CSH Link

As a working example, this article describes the process of creating a CSH link from beginning to end using the new BBjInputD control.

The first step is to locate the topic in the documentation that describes the creation of a BBjInputD control by using the BASIS Web site's search feature. Just point your Web browser to **www.basis.com/**, select "**Documentation**" from the Search drop down list, type **InputD**, then press **Go**. Click on the first item returned in the list, the **INPUTD Mnemonic - Create Date Edit (INPUTD) Control**.

The second step is to create an embedded version of this link. In order to do this, you need the topic ID number. To get the topic ID, simply search for the topic name in the `MapReport.txt` file available from **www.basis.com/devtools/ documentation/index.html**. A quick search of the topic name, "INPUTD Mnemonic - Create Date Edit (INPUTD) Control" brings you to the following section in the `MapReport.txt` file:

```
INPUTD Mnemonic - Create Date Edit (INPUTD) Control -
inputd_mnemonic_create_date_edit_(inputd)_control.htm
Map ID references:
INPUTD_Mnemonic_Create_Date_Edit_(INPUTD)_Control #14172(mnemonic.hh)
```

The last line shows that the corresponding topic ID number for the BBjInputD control is "14172." Use this number at the end of the embedded URL to popup the appropriate topic in the documentation.
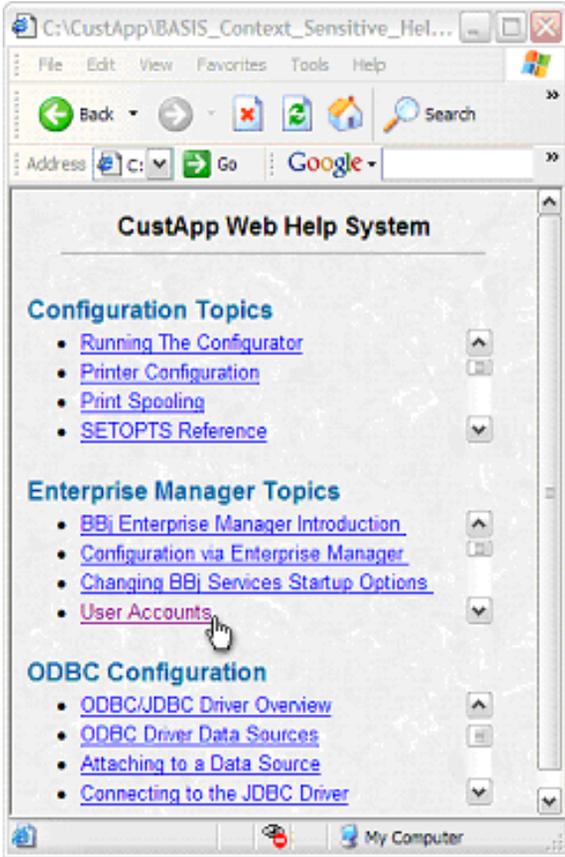
The third step is to construct the final URL. Using the previous examples, modify the existing topic ID at the end of the URL to reference the one corresponding to the BBjInputD control. The new URL is **www.basis.com/ onlinedocs/documentation/index.htm#<id=14172**.

This URL is completely functional, but BASIS recommends that you add one more parameter to improve usability. Adding a `target=` parameter in the HTML anchor tag defines a customizable window used for the popup. This offers the following benefits. First, assuming that the user is viewing the resultant link in a Web browser, the addition of the target parameter leaves the original browser session intact and performs the redirection with a new instance. Second, by providing a common target parameter, the same popup is used for each link, which prevents a new browser instance each time a user clicks on a CSH link. Listed below is the final HTML code utilizing the link just created:

```
<A HREF="http://www.basis.com/onlinedocs/
documentation/index.htm#<id=14172"
TARGET=CSH>Creating an InputD Control</a>
```

You can use this link in any HTML page or many other types of help systems that support URL links.

More Than Just Links – A Common API

The links provided in this article illustrate how to use the remotely accessible CSH. Not only do the practical applications of CSH go far beyond this example, the access methods do as well. BASIS uses RoboHelp®, the industry standard for help authoring to create our online help system. RoboHelp provides the libraries that implement a common ApplicationProgramming Interface (API) for CSH. These libraries exist for Web pages (HTML/JavaScript), Java applications, C/ C++, and Visual Basic. The libraries allow you to access CSH with just a few lines of code, regardless of which language is used.

Recap

Help systems are a vital part of any application. Today's technologies open the doors for robust help systems that offer live Web links, up-to-date content, content sensitive popups, and multi-media. BASIS's foray into in the world of remotely accessible context sensitive help explores the many possibilities of CSH, making the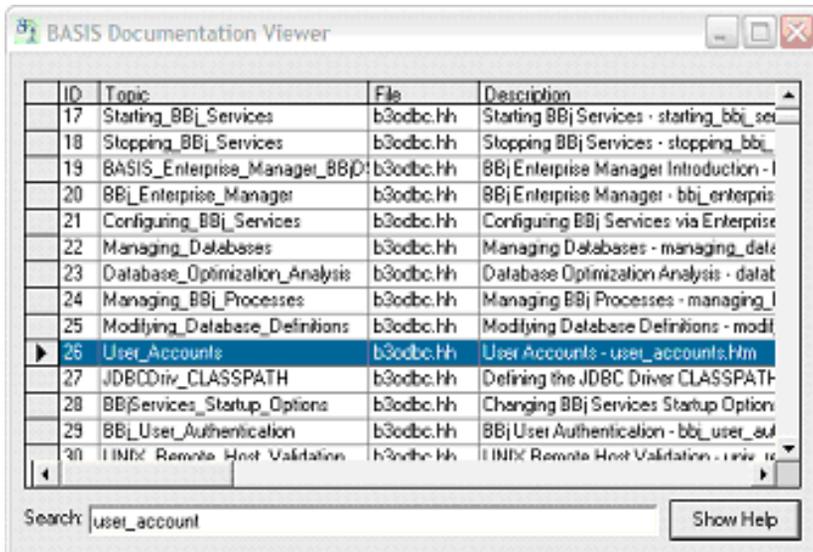 benefits available to our customers.