

Welcome to WinHelp BugHunter+ (16-bit)



WinHelp BugHunter+ allows you to quickly diagnose and fix problems that can occur when linking help systems to 16-bit Windows applications.

You'll find WinHelp BugHunter+ particularly useful when:

- You click a Help button in an application you created help for and you get the message, "Help Topic Does Not Exist" instead of your help topic.
- Your help system appears, but the wrong topic is displayed when your help system is called.
- You want to quickly determine if a particular map number is used in your help system.

See Also

[System requirements](#)

[Starting WinHelp BugHunter+](#)

[About context-sensitive help](#)

[Testing specific map IDs \(context IDs\)](#)

[Contacting eHelp Corporation](#)

System requirements

You can use WinHelp BugHunter+ (16-bit) to troubleshoot context-sensitive help for 16-bit applications running under Windows 3.1, 95, 98, NT 4.0, or 2000.



If you have a Windows NT system (4.0 or 2000), a different version of BugHunter (32-bit) is installed by default. WinHelp BugHunter+ (16-bit) will only be installed if you choose to install the tools for WinHelp 3.

See Also

[Starting WinHelp BugHunter+](#)

[About context-sensitive help](#)


Starting BugHunter+

You can start WinHelp BugHunter+ from within RoboHELP, or you can start from within RoboHELP Office as a standalone tool.

To start WinHelp BugHunter+ from RoboHELP:

- 1 From the RoboHELP Explorer Tools tab, double-click **WinHelp BugHunter 16-bit** . The WinHelp BugHunter+ window displays.

To start WinHelp BugHunter+ from RoboHELP Office:

- 1 If you are using Windows 95, 98, NT 4.0, or 2000: From the **Start** menu, select **Programs, RoboHELP Office, RoboHELP Office**. The RoboHELP Office Starter window appears.
- 2 If you are using Windows 3.1, open the **RoboHELP Office** program group.
- 3 Double-click **WinHelp BugHunter 16-bit** . The WinHelp BugHunter+ window displays.



You can choose to have BugHunter+ always displayed on your desktop while it's open. Start WinHelp BugHunter+, select the **Options** menu, and select **Always on Top** (this option is enabled when it displays a checkmark).

See Also

[About context-sensitive help](#)

About context-sensitive help

Context-sensitive help is online help that users can activate from inside a working application. The help topics provide more information about what the user sees in the application – more about a dialog, an option in a dialog, a message in the application, or an object on the screen. For example, when a user clicks on the Help button in a dialog, a help topic is displayed that is specifically associated with that dialog. The application knows which help topic to display because both the application and the help topic share the same **map ID** (also called **context IDs**).

What are map IDs?

The map string and the map number comprise the map ID. In RoboHELP, map IDs are links that contain two vital pieces of information:

- **Map string.** The map string is a "string" of characters that uniquely identify a map ID. The map string is also called the **topic ID**.
- **Map number.** The map number is a unique number (an integer) that identifies the map ID.

The map number is assigned a textual string (such as "IDH_My_First_topic"). For convenience, both the programmer and the writer use the map string instead of the map number to identify the help topic and its corresponding screen element. (The communication between the application and the help engine is still the number, however.)

Problems can arise when the link in the application source code does not match a map number in the help system. Such discrepancies result in WinHelp displaying "Help Topic Does Not Exist" messages. WinHelp BugHunter+ shows you the number that the application is calling, and thus allows you to quickly diagnose and eliminate these kinds of problems.

What are map files?

Map files are text files that contain a list of map IDs. Map files list map IDs line by line – map strings and map numbers. The link between a topic's map string and map number is through a map file. Each topic in the help file is listed in the map file and assigned a map number and map string. Sometimes the help author generates the map file (RoboHELP can generate it automatically) and provides the list of map IDs to the developer. Sometimes the developer provides the map file to the help author. The map file usually has an .HH extension.

Example:

If you open a map file in a text editor, it often contains lines that look like this:

```
#define FILENAME_My_Help_Topic_ 101
```

In this example, **FILENAME** refers to the document name. **My_Help_Topic** is the map string and **101** is the map number.



You can easily view the map string and the map number for a particular topic in RoboHELP by looking at **Topic Properties**. The map string (also called the **topic ID**) is listed on the **General** tab, and the map number is listed on the **Advanced** tab. You can also see all map IDs for a project at once. From **RoboHELP Explorer**, select the **Project** tab, open the **Project** folder, and open the **Map IDs** folder.

What causes context-sensitive help to break down?

Context-sensitive help is usually a joint effort between the help author and the programming team. Problems displaying context-sensitive help result when:

- The wrong map ID or wrong help file name is used in the map file
- The wrong map ID is assigned to the help topic
- A topic is not assigned a map ID
- A map ID is changed

See Also

[Starting WinHelp BugHunter+](#)

RoboHELP Classic online help

Determining what map ID the application is calling

If you try to access a help topic and get a message that says the topic can't be found, you can use BugHunter+ to see which map ID your application is calling. Then you can check your help topic to see if it has the same map ID.

To determine what map ID the application is calling:

- 1 [Start WinHelp BugHunter+.](#)
- 2 Start the application you need to debug.
- 3 Resize the windows on your desktop so that you can see both your application and BugHunter+.
- 4 Repeat the action that causes the error you need to fix. For example, click the Help button that is calling the wrong topic or producing an error message. Any calls that the application makes to WinHelp are captured and displayed in the BugHunter+ window.
- 5 Make a note of the help file name and map ID displayed in BugHunter+. You'll use this information to repair the broken link.

See Also

[Overview of repairing broken links](#)

[Clearing the BugHunter window](#)

[Copying BugHunter data to the clipboard](#)

[Displaying BugHunter as Always on Top](#)

[WinHelp BugHunter+ data](#)

[Viewing the map ID \(context ID\) in Hexadecimal](#)

About repairing broken links

You can approach fixing a broken link in several different ways, depending on the situation. You should discuss the problem with your application developer to determine the best solution. Here are some scenarios you may encounter:

- **One of the WinHelp calls is referencing the wrong map ID.**

In this situation, the help author can [update the map ID](#) or the developer can reprogram the WinHelp call. Discuss this with your developer to decide who will repair the problem.

- **The WinHelp call is referencing an old help file name.**

The help file name has changed. In this situation, the help author can change the name of the compiled help file (using Windows Explorer) or the developer can reprogram the WinHelp call.

- **One of the WinHelp calls is referencing a map ID that does not exist.**

This may occur if the help author has not created the topic yet. In this case, the help author can [create a topic with the specific map ID](#).


- **There are several WinHelp calls referencing several different map IDs, but you've decided you want them to call one single topic.**

In this case, the help author can [alias the map IDs](#) to a single topic.

Changing the map ID for a topic

Use this procedure when you need to choose a different map ID for a context-sensitive topic – you need to replace an "old" map ID with a "new" one. You might need to use this procedure if your programmer gives you a map ID that has changed. (Remember that "map ID" and "topic ID" mean the same thing in RoboHELP.)


To change the map ID for a topic:

- 1 Open the project in RoboHELP.
- 2 Navigate to the topic whose map ID you want to change.
- 3 From Word, right-click anywhere in the topic and select **Properties**. The **Topic Properties dialog** appears.
- 4 In the **General tab**, click  to the right of the **Topic ID**. The **Select Unused Topic ID** dialog appears. (If this button is disabled (grayed out), there are no unused or missing topic IDs.)
- 5 In **Unused Topic IDs**, select the unused topic ID you want to connect to this topic and click **OK**.
- 6 Recompile the project.
- 7 Place the HLP file in the same folder as the application and retest the link.

Creating a new topic with a specific map ID

Use this procedure when you want to assign a map ID to a topic at the same time you create the topic.

To create a new topic with a specific map ID:

- 1 Open the project in RoboHELP.
- 2 Click the **New Topic** button. The **New Topic dialog** appears.
- 3 Click the **General tab**, then click  to the right of **Topic ID**. The **Select Unused Topic ID** dialog appears. (If this button is disabled (grayed out), there are no unused or missing topic IDs.) Complete the remaining information for your new topic.
- 4 In **Unused Topic IDs**, select the unused topic ID you want to connect to this topic and click **OK**.
- 5 Recompile the project.
- 6 Place the HLP file in the same folder as the application and retest the link.

About aliases

Aliases are used for two main reasons: to fix broken links caused by changing a map ID (topic ID) and to connect topics to map IDs for context-sensitive help.

- **Fixing broken links.** Whenever a topic is referenced in a help system, the reference uses the topic ID as the unique identifier to locate the destination topic. If you change the topic ID, you "break" the links referencing the topic. Ways to reference a topic include hotspots (jumps, popups, macros) and pages in the table of contents. You can repair a broken link by aliasing the old topic ID to the new topic ID.
- **Connecting map IDs to context-sensitive help topics.** There are two reasons you might use an alias to connect map IDs and topics:
 - You need to connect more than one map IDs to the same topic. For example, your programming team provides you with seven map IDs for a dialog. But you've planned one topic to cover the entire dialog, so you need to connect all seven map IDs to the same topic.
 - You need to connect an existing topic to a map ID. For example, you and your programming team are developing simultaneously. When you receive the map files from your programmer, you have already created the topics – each with its own topic ID. Now you need to connect your existing topic to the map ID provided by the programmer.

See Also

[Determining if a map ID is in your project](#)

[Creating a new topic with a specific map ID](#)

[Changing the map ID for a topic](#)

[Creating an alias](#)

Determining if a map ID is in your project

You've ran BugHunter+ and found the map ID the application is calling. Now you can find out if that map ID is in your project so that you can:

- [create an alias](#) to the map ID if it is already in your project
- [import the map file into your project](#) if it hasn't been imported yet
- [change the map ID](#) for the topic if the map ID has changed

The following procedure shows you how to find a map ID using its map number.

To determine if a map ID is in your project:


- 1 Open your project in RoboHELP.
- 2 From **RoboHELP Explorer**, select the **Project** tab.
- 3 Open the **Project** folder.
- 4 Open the **All Map IDs** folder.
- 5 From **RoboHELP Explorer**, select the **Edit** menu and select **Find Map ID**.
- 6 Type the map number.
- 7 Click **OK**.

If the map ID is in your project, it will be selected in Project view and you can [create an alias](#) to it. If the map ID is not in the project or is not being used, you can [add it to your project](#).

Creating an alias

If you run BugHunter+ and determine that the map ID exists in your project but it's not connected to the right topic, you can create an alias to it.

To create an alias:

- 1 From **RoboHELP Explorer**, open the **Project** folder.
- 2 Right-click the **Aliases** folder and select **New Alias**. The **New Alias** dialog displays.
- 3 Click the  icon select an unused topic ID.
- 4 Select the topic title or topic ID you want this alias to be connected to.
- 5 Save and compile the help project then re-test the links.

See Also

[About aliases](#)

RoboHELP Classic online help

Importing a map file into your project

Use this procedure to add an unused map ID to your project. This procedure assumes you have already created the topic.

To add a map ID to your project:

- 1 From the **File** menu, select **Import**, then select **Map File**. The **Import Map File** dialog displays.
- 2 Navigate to the folder that contains the map file you want to include.
- 3 Select the map file and click **Open**.
- 4 If you want to copy the map file to the help project directory, select **Copy to project directory**.
- 5 Click **OK**.
- 6 Now create an alias from the map ID to the topic.

See Also

[Creating an alias](#)

RoboHELP Classic online help

Copying BugHunter+ data to the Clipboard

The information that appears in the BugHunter+ window is text that you can copy to the Windows Clipboard and paste into other files. This can be useful if you want to send the contents of the BugHunter+ window to a developer in an e-mail message.

To copy BugHunter text to the clipboard:

- 1 In the BugHunter window, highlight the text to be copied.
- 2 From the **Edit** menu, select **Copy**. The information is copied to the Windows Clipboard so you can paste it into an e-mail message or other files.

Clearing the BugHunter+ window

The WinHelp BugHunter+ window clears when you close it so that each time BugHunter is started, an empty window displays. As you are working with BugHunter, you may want to clear the window without restarting the program.

To clear the BugHunter window:

- From the BugHunter+ **Edit** menu, select **Clear**.

Displaying BugHunter+ as "Always on Top"

When you click on another application while BugHunter+ is open, the BugHunter+ window may be hidden. If you want to be able to see the BugHunter+ window at all times, you can use the "always on top" option.

To set BugHunter+ to "Always on Top":

- From the BugHunter+ **Options** menu, select **Always on Top**. A check mark indicates this option is active. To disable, select Always on Top again.

Testing map numbers (context IDs)

BugHunter+ records all WinHelp calls, but it also allows you to test specific calls. You can enter a map number (formerly called context ID or context number), and then use BugHunter to display the topic that corresponds to that map number.

To test a map number (context ID):

- 1 From the BugHunter+ **Test** menu, select **Context ID**.
- 2 Click **Browse** and select the help file (HLP) you want to use. The location of the help file is shown in **Windows Help File**.
- 3 Enter the map number (context ID).
- 4 To see the topic in the BugHunter+ window, click **Test**.
- 5 To view the topic in a popup window, click **View**.
- 6 When you finish testing map numbers, click **Close**.

Note: To navigate in the BugHunter+ window again, you must first close the **Test Context ID** dialog.

Viewing map IDs in hexadecimal

By default, the map ID (context ID) is shown as decimal numbers. You can view the map ID in hexadecimal numbers if you want.

To view the map ID in hexadecimal:

- From the **Options** menu, select **View as Hex**.

Interpreting BugHunter+ (16-bit) data

WinHelp BugHunter+ records an "event" each time a call is made to WinHelp – that is, any time Windows help is called while BugHunter is active. The event log specifies the type of help call and any parameters that were passed with the call, enabling you to track the information that the application is passing to WinHelp.

BugHunter reports the following:

Event Time, Function Call, Help System Name, Help Command, Map Number.

- If Windows 95 commands are called from your application, the following additional data may be displayed:
Event Time, Function Call, Window Handle, Help System Name, Help Command, Pointer or Help Topic ID, Control Type, Control Text, Control ID, Map Number.

Use the following list of help commands to interpret the BugHunter+ window.

Help Commands:

HELP_COMMAND,<macro name>(<parameters>)

A call was made to a WinHelp macro. The macro name and any applicable parameters are also listed.

HELP_CONTENTS

A call was made to the Contents topic of the help system.

HELP_CONTEXT,<context ID>

A call was made to the context sensitive help topic indicated by the context ID.

HELP_CONTEXTMENU

A Windows 95 call was made from a right mouse button *What's This?* click.

HELP_CONTEXTPOPUP,<context ID>

A call was made to the context sensitive help topic indicated by the context ID. The call specified that the topic was to be displayed in a popup window.

HELP_FORCEFILE,<file name>

A call was made to WinHelp, specifying the help file to be displayed. The name of the help file is passed as a parameter.

HELP_HELPONHELP

A call was made to the standard help system "Help on Help." This is sometimes labeled "How to Use Help" in application's help menus.

HELP_KEY,<keyword>

A call was made to WinHelp to display the topic containing the specified keyword. The keyword is passed as a parameter.

HELP_MULTIKEY,<table footnote>,<keyword>

A call was made to WinHelp to display the topic containing the specified keyword. This type of call is used when the keyword is in an alternate keyword table. The table footnote character and the keyword are passed as parameters.

HELP_PARTIALKEY,<keyword>

A call was made to WinHelp to display the topic containing the specified keyword. The keyword is passed as a parameter.

HELP_QUIT

A call was made to WinHelp informing it that help is no longer needed.

HELP_SETCONTENTS,<topic name>

A call was made to the SetContents macro. This macro defines the context string of the topic that is the "Contents" topic of the help system. The context string of the topic is included as a parameter.

HELP_SETWINPOS

A call was made to the SetWinPos macro. This macro specifies the size and position of a primary or secondary help window.

HELP_UNKNOWN

A non-standard call was made to WinHelp.

HELP_WM_HELP

A Windows 95 call was made from either the F1 key or the ? in the title bar of a window.

eHelp Corporation headquarters

eHelp Corporation is the worldwide leader in Help authoring solutions. Software developers, Help authors, technical writers, and documentation experts worldwide turn to eHelp Corporation for state-of-the-art Help authoring and Internet/intranet solutions.

Corporate Headquarters

Address: eHelp Corporation
7777 Fay Avenue, Suite 201
La Jolla, CA 92037
U.S.A.

Telephone: 1-858-551-2485

Fax: 1-858-551-2486

World Wide Web: www.blue-sky.com

Office Hours: Monday to Friday
6:00 A.M. to 5:00 P.M. Pacific Time

[eHelp Corporation Sales Department](#)

[eHelp Corporation Technical Support Department](#)

Sales

eHelp Corporation's professional sales staff is ready to answer your sales questions Monday to Friday from 6:00 A.M. to 5:00 P.M. Pacific Time.

Contact your sales representative to hear about the latest products, the Subscription Plan, training, upgrade options and prices, and more. If you have a technical question or problem, please contact [Technical Support](#).

Contacting your eHelp Corporation Sales Representative

Sales Phone: 1-800-677-4946

International (including Canada): 1-858-459-6365

Sales Fax: 1-858-459-6366

World Wide Web: www.blue-sky.com

Sales Email: sales@blue-sky.com

When sending email, please include your company name, city, and state to ensure your request is handled promptly.

Technical Support

At eHelp Corporation, we work hard to provide you with intuitive technical products. Additionally, we try to provide superior online and print documentation to enable you to work independently. If you have a technical question that you can't answer with the provided tools, please contact our Technical Support department. All of our Technical Support representatives are eager to answer your questions and help you create the best Help systems possible.

Note: You must be a registered user to access eHelp Corporation support services. eHelp Corporation support services are subject to the prices, terms, and conditions in place at the time the service is used.

eHelp Corporation's Technical Support services includes email, telephone, and Web-based support on our corporate Web site (www.blue-sky.com).

Help Us Help You

When contacting Technical Support via phone or email, please provide the following information for the fastest possible service:

- Product serial number (registration number)
- Your name, company name, and phone number
- Exact product name and version number
- Type of operating system (e.g., Windows 95)
- Version of Microsoft Word (if using RoboHELP for WinHelp)
- Browser and version (if creating HTML-based Help)
- Complete description of the issue, including steps to reproduce it
- Exact wording of any messages displayed when you encountered the problem
- Steps taken to resolve the problem
- All previous email threads with eHelp Corporation about the issue, if any

For current hours of operation and details about all support offerings, please:

- Visit our Web site (www.blue-sky.com)
- Email us at **supportnews@blue-sky.com** to receive an automated reply listing all of your support options.
- Call us at 858-551-2194.

Premium Support

Premium Support entitles plan members to priority Technical Support access through a dedicated telephone line and a dedicated email address. Plan members receive direct access to eHelp Corporation's seasoned Senior Technical Support Engineers.

Premium Support plans are available for current and previous versions of eHelp Corporation products.

{ewl RoboEx32.dll, WinHelp2000, }

