

# **Helix RADE Classic 6.1.5 Release Notes**

*Copyright 1985–2009 QSA ToolWorks, LLC*



<b>1 Welcome to Helix RADE Classic 6.1.5</b>	<b>1</b>
<b>1.1 Introduction</b>	
<b>1.2 About this document</b>	
<b>1.3 About this release</b>	
1.3.1 Helix RADE Classic 6.1.5 and Helix 6 compatibility	
1.3.2 Helix Utility and Update Collection 6.0.1 are current	
<b>2 Contacting QSA ToolWorks, LLC</b>	<b>3</b>
<b>2.1 Sales and Customer Service</b>	
<b>2.2 Technical Support</b>	
2.2.1 techdb: the Helix-based source for technical support	
2.2.2 Sending files	
2.2.3 About collection repair	
<b>2.3 Bug Reports</b>	
<b>2.4 Feature Requests</b>	
<b>3 Configuring Your Computer</b>	<b>4</b>
<b>3.1 Introduction</b>	
<b>3.2 Mac OS 9.1 or higher required</b>	
3.2.1 Memory management	
<b>3.3 Installing Helix RADE</b>	
3.3.1 Enabling Helix RADE	
<b>4 Preparing to Update</b>	<b>5</b>
<b>4.1 Please read these notes carefully</b>	
4.1.1 Make a backup before you update	
4.1.2 Update Collection not required	
4.1.2.1 Update Collection	
4.1.2.2 Helix Utility	
4.1.3 Stay informed	
<b>5 What's New in Helix RADE Classic 6.1</b>	<b>6</b>
<b>5.1 Changes in Helix RADE Classic 6.1.5</b>	
5.1.1 Collection repair diagnostic info access	
<b>5.2 Changes in Helix RADE Classic 6.1.x</b>	
<b>5.2.1 Static Enter available in User Editor</b>	
5.2.1.1 OS X design impact	
<b>5.2.2 Clean Up command enhanced</b>	
5.2.2.1 Historic Perspective	
5.2.2.2 Additional Clean Up options	
5.2.2.2.1 Clean Up with the option key down	
5.2.2.2.2 Clean Up with the shift and option keys down	
<b>5.2.3 User Editor enhancements</b>	
<b>5.2.4 User Mode bypass</b>	
<b>5.2.5 Show Clipboard command key equivalent</b>	
<b>5.3 Clippings Menu</b>	
<b>5.3.1 Introduction</b>	
<b>5.3.2 Setup</b>	
<b>5.3.3 Usage</b>	
5.3.3.1 Save Selection As Clipping	
5.3.3.2 Show Clippings Folder	
5.3.3.3 Refresh Clippings Menu	
5.3.3.4 Divider	
5.3.3.5 Clippings Data	

5.3.3.6 Additional notes

5.3.3.7 See also

## **5.4 Pasting Structure: Rename Replaced Icons**

## **5.5 Design Mode enhancements**

5.5.1 Select unused icons

5.5.2 Accidental icon opening avoided

5.5.3 Prevent window opening on switch to Design Mode

5.5.4 Design Mode icon zooming performance

## **5.6 Notify On Change activation**

## **5.7 Keyword Indexing and the HKWT Resource**

5.7.1 Introduction

5.7.2 An important note on keyword index changes

5.7.3 HKWT customizing for each collection

5.7.4 Changes to the Keyword Separator Table

5.7.4.1 Missing characters added

5.7.4.2 Ligatures added

5.7.4.3 Non-breaking space added

5.7.5 Inaccurate information

5.7.5.1 Updated and corrected tables available

5.7.6 Keyword Management Utility

5.7.6.1 Direct editing

## **5.8 Keyboard shortcut modifiers supported (semi-complete feature)**

## **5.9 Navigation improvements**

# **6 Specification Changes** **13**

---

## **6.1 Compiled forms eliminated**

## **6.2 Paste Structure dialog access**

## **6.3 Default font for templates**

6.3.1 Quick entry templates

6.3.2 Quick list templates

## **6.4 Design Mode commands and command key changes**

## **6.5 User Mode commands and command key defaults**

# **7 Bug fixes** **15**

---

## **7.1 Bugs fixed in Helix RADE Classic 6.1.5**

7.1.1 7603 Error on pressing Tab in collection/relation list view (R#6734)

## **7.2 Bugs fixed in Helix RADE Classic 6.1.3**

7.2.1 Editing label rectangle containing a picture could corrupt database

7.2.2 Reports with over 32,766 records failed to display all records (BZ976)

7.2.3 Option 1 post to cold form with defaulted field corrupts index (BZ900)

7.2.4 Fixed Point posting trigger multiplied by 100 (BZ1034)

7.2.5 Eszett (ß) character (0xA7) is a word character in German

7.2.6 Keyword index progress dialog

7.2.7 Command key for Shade Defaults changed

7.2.8 Crash when pressing arrow keys

7.2.9 Registration dialog issues (BZ904)

## **7.3 Bugs fixed in Helix RADE Classic 6.1.2**

7.3.1 AutoOpen Post with omitted key field corrupts index (BZ777)

7.3.2 Crash Recovery: pictures over 64KB are lost (TS2233)

7.3.3 Quick Query references missing from Get Info

7.3.4 Keyword and Simple Validations hide true reference count

7.3.5 Potential crash on export

7.3.6 Potential write lock failure

- 7.3.7 'Mini menu' cleaned up
- 7.4 Bugs fixed in Helix RADE Classic 6.1.1**
  - 7.4.1 Paste structure with renaming on could crash
  - 7.4.2 Why? Message contains garbage text
- 7.5 Bugs fixed in Helix RADE Classic 6.1**
  - 7.5.1 Find can fail when posting changes query criteria (BZ935)
  - 7.5.2 Command rectangle duplication (TS2067)
  - 7.5.3 Structure Copy/Paste crashes (TS2068/TS2069)
  - 7.5.4 Pasting Structure fails to identify top level icons (TS2027)
  - 7.5.5 Intermittent posting failures (TS2013)
  - 7.5.6 User view permissions inadvertently restricted (TS1992)
  - 7.5.7 Crash when locked icons are selected (TS1912)
  - 7.5.8 Enter menu highlighting
  - 7.5.9 Why? message during sequence: format error
  - 7.5.10 Build number missing from splash screen

## 8 Known Problems

19

- 8.1 New in Helix RADE Classic 6.1**
  - 8.1.1 Helix Clippings menu disabled after User Mode switch
  - 8.1.2 Helix Clippings menu item names
- 8.2 Pre-existing known problems**
  - 8.2.1 Design Mode**
    - 8.2.1.1 Editing a graphics rectangle via dialog can corrupt collection (BZ1051)
    - 8.2.1.2 Deleting an icon can result in reported collection damage #1 (TS1042)
    - 8.2.1.3 Deleting an icon can result in reported collection damage #2 (BZ990)
    - 8.2.1.4 Index built on abacus can fail to invalidate
    - 8.2.1.5 Undo style change crashes
    - 8.2.1.6 Icon placement
    - 8.2.1.7 Data format dialog/dynamic popup panel
    - 8.2.1.8 Document based dynamic popups crash when edited
    - 8.2.1.9 User editor crash with large number of objects
    - 8.2.1.10 Design Mode vertical scroll bar disappears in large relations (TS1648)
    - 8.2.1.11 Cosmetic problems
      - 8.2.1.11.1 Dragging in sequence window
  - 8.2.2 User Mode**
    - 8.2.2.1 Printing Issues
      - 8.2.2.1.1 Record data reverts when printed if post on print applies to the view
      - 8.2.2.1.2 Find and Print All with more than 128 records
      - 8.2.2.1.3 Last visible item on a list view fails to print
    - 8.2.2.2 Inert field behavior
      - 8.2.2.2.1 "The Enter Key Cannot be Pressed Here"
    - 8.2.2.3 Option 4 posting in Find and Print All (BZ740)
    - 8.2.2.4 Deleted field reverts to its previous value after entry (TS958)
    - 8.2.2.5 Text to Styled Text display error
    - 8.2.2.6 Undefined indexes can compound collection damage (TSD22002)
    - 8.2.2.7 Power Query text edit field issues
    - 8.2.2.8 Dynamic popups limited to 254 items
    - 8.2.2.9 Subform or list won't open the selected entry view (TS960)
    - 8.2.2.10 Lookup tiles in dynamic popups: a caveat
    - 8.2.2.11 Page Number overflow
    - 8.2.2.12 Text Field overflow
    - 8.2.2.13 Large list record selection
    - 8.2.2.14 Document handling
      - 8.2.2.14.1 Importing external documents

- 8.2.2.14.2 External documents in text and EPS formats not displayed
- 8.2.2.15 Apple Events can be established but not fulfilled
- 8.2.2.16 Use of Helix Custom Help
- 8.2.2.17 Sleep mode crash
- 8.2.2.18 Rarities
  - 8.2.2.18.1 Potential “Save As” leads to collection damage
  - 8.2.2.18.2 Extremely long save when importing many records
- 8.2.2.19 Cosmetic problems
  - 8.2.2.19.1 Proportional scroll boxes and asynchronous views
  - 8.2.2.19.2 Color icons in user menus
  - 8.2.2.19.3 Import/Export dialog options command key
  - 8.2.2.19.4 Delete data dialog when subforms are involved
  - 8.2.2.19.5 Appearance manager compliance

## 9 Functional Clarifications

---

26

### 9.1 Design Mode

- 9.1.1 Removing icons from rectangles

### 9.2 User Mode

- 9.2.1 Two-digit date input
- 9.2.2 ‘Previous’ tile behavior when undefined fields are encountered
- 9.2.3 Index limitations
- 9.2.4 Apple Events: after a delete, subsequent counts are wrong
- 9.2.5 About the recovery file

### 9.3 Rounding in Helix

- 9.3.1 Rounding Fixed Points: Financial Rounding
- 9.3.2 Rounding Numbers: Scientific Rounding
  - 9.3.2.1 Decimal to Binary Conversion Errors



# 1 Welcome to Helix RADE Classic 6.1.5

## 1.1 Introduction

Welcome to Helix RADE Classic 6.1.5. This is the fifth maintenance release of Helix RADE Classic 6.1. This release contains feature refinements and bug fixes that have been incorporated into the code.

## 1.2 About this document

This document is a supplement to *The Helix Reference*, the complete guide to Helix 4.5.5 and earlier. It is our intention to publish an up-to-date, self-contained Helix reference manual, but that manual is not available as of this writing.

This document describes the features, enhancements, and bug fixes made to Helix RADE since the release of Helix RADE 6.0.1. This document is designed for people who are already familiar with Helix RADE 6.0.1 and want to quickly learn about the changes made since that release.

This document covers all changes made in Helix RADE Classic 6.1; fixes and enhancements made since the initial release are noted where appropriate. For a list of changes specific to each release, visit <http://www.qsatoolworks.com/product/helix61/rade.html>

If you are not familiar with Helix RADE 6.0.1, you should also refer to the *Helix 6.0.1 Release Notes*, as that document covers changes made in earlier Helix 6 releases. You may also want to refer to the *Helix 5.3.2 Release Notes*, as that document covers changes made during the Helix 5 life-span.

QSA ToolWorks, LLC makes the information in this document available on an as is basis and is not responsible for its accuracy, use, or future compatibility with Helix products or other products such as the Macintosh® operating system.

## 1.3 About this release

Helix RADE Classic 6.1 is a feature release, primarily focused on Design Mode changes that complement the Mac OS X Helix products. There are a few new features, some specification changes, and a number of bug fixes. Helix RADE 6.1 is still a Classic application, but the new features and changes make working on collections (whether for use with the Mac OS X native Helix products or not) a more productive experience.

Helix RADE Classic 6.1.5 is a maintenance update, continuing the bug fixes and feature enhancements theme of making working on collection design more productive.

Helix RADE Classic 6.1.5 is a free update for owners of Helix RADE Classic 6.1. Helix 6.0 and earlier serial number/enablement key combinations do not work with this release. Visit our web store at <http://store.qsatoolworks.com> or contact QSA ToolWorks, LLC by email [sales@qsatoolworks.com](mailto:sales@qsatoolworks.com) or by phone 800-784-7018 for information on upgrading to Helix RADE Classic 6.1.5.

Helix RADE Classic 6.1.5 is a general release and can be used with all existing collections. We encourage all customers to update existing collections to Helix RADE Classic 6.1.5, regardless

of whether or not they are experiencing problems in the areas that are specifically addressed by this release.



*Although we are confident that Helix RADE Classic 6.1.5 is a solid release, it is still possible that problems may be found that could result in collection damage. QSA ToolWorks, LLC will do everything in its power to support users who encounter errors caused by this software, but we cannot guarantee that problems do not exist. As always, please be diligent regarding your backup and utility procedures.*

Helix RADE Classic 6.1.5 represents an interim step in our work in updating our code for Mac OS X. Helix Server, Helix Utility and Update Collection are currently Mac OS X native. Mac OS X native versions of Helix Client and Engine are available in Preview Release versions.

It is our intention to release Mac OS X native products as soon as possible, but because of the technical issues that made it impossible to reliably run Helix Server in Classic Mode in Mac OS X, we chose to focus first on the Mac OS X native Helix Server and to release it as soon as it was ready. That task was completed in December, 2005.

Shortly thereafter, Apple announced the transition from the PowerPC processor to Intel processors. That transition was completed in 2007. Unfortunately, Intel-based Macs do not support the Classic environment, and writing programs that run natively on them requires a specific programming environment (Xcode). In addition, Mac OS X 10.5 (aka: Leopard) which was released in October, 2007 does not support the Classic environment on PowerPC-based Macs. These events created a pressing need for Mac OS X native versions of all Helix products.

The technical nature of the shared components within the Helix product line make it most logical for us to deliver Mac OS X native versions of Helix Engine and Helix Client next. Helix RADE will be the last product to become Mac OS X native.

The Mac OS X Transition Journal on our web site contains the latest updates on this process.

### **1.3.1 Helix RADE Classic 6.1.5 and Helix 6 compatibility**

Helix RADE Classic 6.1.5 is “code compatible” with all Helix 6 products. Collections edited with Helix RADE Classic 6.1.5 can be used without issue with all Helix 6.0 products. You can take advantage of the new features in Helix RADE Classic 6.1.5 to create collections that can be used with your existing Helix 6.0 Client/Server and Helix Engine, as well as the Mac OS X native Helix Client and Helix Engine available in Helix 6.1. Helix RADE Classic 6.1.5 can also be used to enable new features that are available only for the Mac OS X native Helix Client.

Even so, running Helix Utility and Update Collection on your collections before using them with Helix RADE Classic 6.1.5 — and then running both utilities on a regular basis to ensure continued collection integrity — is strongly encouraged. QSA ToolWorks, LLC also sells Helix Maintenance Manager, a utility that automates the backup and utility checking process. Visit our web site and click on the Products link for more information.

### **1.3.2 Helix Utility and Update Collection 6.0.1 are current**

Because Helix RADE Classic 6.1.5 is fully code compatible with previous Helix 6 releases, the existing Helix 6.0.1 versions of Helix Utility and Update Collection should be used with Helix RADE Classic 6.1.5. There is no version 6.1 release of these utilities.

Helix Utility and Update Collection 6.0.1 are available in Mac OS X native and Classic versions. The Mac OS X versions require 10.3 or later; the Classic versions require Mac OS 9.1 or later.

The Mac OS X and Classic versions of Helix Utility and Update Collection can be used interchangeably with all Helix 6 collections. If you are using Helix RADE Classic 6.1.5 in Classic Mode in Mac OS X, you should use the Mac OS X native version of the utilities, as they provide better checking capability and run significantly faster than their Classic counterparts.

## 2 Contacting QSA ToolWorks, LLC

### 2.1 Sales and Customer Service

If you need to purchase new or upgraded Helix products, you can visit our online store at <<http://store.qsatoolworks.com>>. You can also contact our sales and customer service department by email <[sales@qsatoolworks.com](mailto:sales@qsatoolworks.com)> or by phone 800-784-7018. General questions and other administrative issues should also be directed to the customer service department. The sales office is open 9AM–5PM Eastern Time (1400–2200 GMT).

### 2.2 Technical Support

If you encounter technical problems (or have general technical questions) related to Helix, technical support is here to help. You can contact technical support by email <[support@qsatoolworks.com](mailto:support@qsatoolworks.com)> or by phone 570-662-8883. Telephone support is available 9AM–5PM Eastern Time (1400–2200 GMT). Some support services are subject to additional fees.

The Technical Support section of our web site <<http://www.qsatoolworks.com>> contains the latest information about the support services we offer.

#### 2.2.1 techdb: the Helix-based source for technical support

techdb is a Client/Server database that we use to provide technical support to our customers. In techdb you can search our knowledgebase, look up Helix error codes, submit bug reports and feature requests, chat with other Helix users, and more.

In essence, techdb is a live demo of Client/Server in action, and it provides a communal gathering spot for Helix users around the world. That alone makes it a worthwhile part of our technical support services.

Instructions on accessing techdb are found at...  
<<http://www.qsatoolworks.com/support/techdb.html>>

#### 2.2.2 Sending files

Please do not send us files via email. Because of the increasing problem of malware sent as email attachments, unsolicited attachments are automatically deleted. If you have a file you wish to send to us, please contact us first and describe the problem. We may have already isolated the source of the problem and can discuss possible remedies without needing to see your example. If we do need to see the file you have, we will send instructions so you can send it in a way that will ensure the fastest possible action on our part.

#### 2.2.3 About collection repair

Collection repair is no longer charged as a flat fee service. Repairs are now billed on an hourly basis, as per our User Support Unit (USU) policy. Many repairs can be covered with the USUs that come with each Helix upgrade, effectively costing nothing extra. We strive to turn all repairs around as quickly as possible; typical turnaround time is under 8 hours. Please see the support section of our web site for more details.

### 2.3 Bug Reports

If we are to continue to improve the product and meet the needs of our customers, we need to know when you experience problems. Bug reports can now be submitted via the interactive bug reporting section of techdb. See section 2.2.1 (*techdb: the Helix-based source for technical support*) for information on connecting to techdb.

### 2.4 Feature Requests

If you wish to submit a request for a feature you would like to see added to Helix in the future, please visit the feature request page on our website and tell us about it. This is an automated system and you will not receive a personal reply from QSA ToolWorks, LLC to items submitted there.

We hope to integrate Feature Requests into techdb soon.

# 3 Configuring Your Computer

## 3.1 Introduction

The following information is designed to help you achieve maximum performance with Helix. Other applications that you use may prefer different settings. If you discover a problem while using this version of Helix RADE, please confirm that it also occurs with these settings in effect. If you discover something that behaves differently based on your Mac OS configuration, please inform us of the details.

## 3.2 Mac OS 9.1 or higher required

Helix RADE Classic 6.1 requires Mac OS 9.1 or higher. Helix RADE Classic 6.1 also runs in Classic mode on PowerPC based Macs running Mac OS X 10.1 through 10.4.x.)

### 3.2.1 Memory management

The techniques for managing memory in Helix RADE Classic 6.1 are unchanged. Review the *Helix 5.3.2 Release Notes* or consult our web site for a discussion of Classic memory management.

## 3.3 Installing Helix RADE

Installing Helix RADE is simply a matter of inserting your Helix CD into your Mac (or downloading the software from our web site) and running the installer package. The installer asks you to choose the folder into which Helix RADE will be placed. We recommend installing into the **Applications (OS 9)** folder.

### 3.3.1 Enabling Helix RADE

After installation, you must *enable* your copy of Helix RADE Classic 6.1.5. Enabling Helix RADE requires three pieces of information:

1. Your Helix product *serial number*
2. The correct *enablement key* for the product and version you are installing
3. Your personal registration information (name, address, etc.)

Helix enablement keys for one product do not work with other Helix products and they only work with the version of Helix for which they are designed. In other words, you can not use a key for different product (e.g. Helix Server) to enable Helix RADE. Likewise you can not use a key for an older version of Helix (e.g. Helix 5.3.2) to enable Helix RADE Classic 6.1.



*When entering a key, the letter 'I' and the number '1' are interchangeable, as are the letter 'O' and the number '0'*

# 4 Preparing to Update

## 4.1 Please read these notes carefully

### 4.1.1 Make a backup before you update

Before you begin the update process, make a backup copy of your database. Refer to Chapter 18 of *The Helix Reference* for more information on maintaining your databases.

### 4.1.2 Update Collection not required

Collections that have been updated to work with prior Helix 6 products do not need to be updated before they can be used with Helix RADE Classic 6.1.5. Nonetheless, checking your collections for potential trouble should be done.

#### 4.1.2.1 Update Collection

To check a collection for structural integrity, drag it onto the **Update Collection** application icon. Two dialogs remind you to make a backup and ask you to confirm that you want to check the collection.

#### 4.1.2.2 Helix Utility

To check a collection for data integrity, drag it onto the **Helix Utility** application icon. A dialog reminds you to make a backup. After the collection is read by Helix Utility, choose the **Data Damage Repair** command found in the Tools menu. A progress dialog opens to let you know the process is working.

If no problems are found, make a backup of the updated collections and put them in a safe place. You are now ready to use Helix RADE Classic 6.1.5.

### 4.1.3 Stay informed

The transition from Mac OS 9.1 to Mac OS X is significant, and new information continues to come to light as we move forward. Be sure to check our web site for the latest information.

# 5 What's New in Helix RADE Classic 6.1

## 5.1 Changes in Helix RADE Classic 6.1.5

### 5.1.1 Collection repair diagnostic info access

Diagnostic information, useful in facilitating field repair of damaged collections, has been added to Helix RADE. Because this information is of little use outside of collection repair, instructions on accessing it is given on a need to know basis.

## 5.2 Changes in Helix RADE Classic 6.1.x

*The changes listed below were introduced in prior Helix RADE Classic 6.1.x releases.*

### 5.2.1 Static Enter available in User Editor

Of the four Enter commands (**Enter**, **Static Enter**, **Enter Override** and **Static Enter Override**) only Static Enter was excluded from the list of commands eligible for inclusion on User Mode menus. This omission has been rectified: the Static Enter command now appears in the list of available commands in the User Editor.

The Static Enter command is prohibited from adding a command key equivalent. The Option-Enter keyboard command that triggers Static Enter can not be changed.

The “Static Enter” menu does not change to “Static Replace” when a record is being modified.

#### 5.2.1.1 OS X design impact

When Static Enter is placed on a user's menu, OS X Helix applies rules to pressing the Enter key similar to the rules it applies to the Enter command. That is, if the user presses Option-Enter, either Enter or Static Enter must be found on the users menu (or the active view must have a default button) or the command is rejected. Note that the Enter command on a user menu allows both Enter and Option-Enter, but the Static Enter command on a menu only allows Option-Enter.

OS X native Helix products automatically display the Option-Enter glyphs in the menu, just as the Enter glyph is seen on Enter/Replace.

### 5.2.2 Clean Up command enhanced

#### 5.2.2.1 Historic Perspective

The specification for the Clean Up command — active only when a Design Mode window is open in ‘View by Icon’ mode — has always been that it relocates icons to the nearest point on the work area that is unoccupied by another icon. This specification is unchanged.

However, designers who work in ‘View by Name’ or ‘View by Kind’ mode, as well as those who modify their collection structure via the Clippings menu or by pasting structure, have had to deal with the fact that icons could often end up placed in distant corners of the workspace. (There is also a subtle bug in Helix that can cause icons to be placed beyond the edge of the workspace.)

#### 5.2.2.2 Additional Clean Up options

To address these issues, two new options have been added to the Clean Up command.

##### 5.2.2.2.1 Clean Up with the option key down

Holding the option key down while selecting Clean up results in all of the icons being pulled together in the currently visible area of the window.



*The width of the window controls the width of the icon row. If there are too many icons to fit the visible area, the icons continue below.*

When using this option the order of the icons may appear random. It is actually determined by the same ‘icon click order’ rule that is used when creating Quick templates and views. To arrange them in a specific order, you can click each icon in the order you want, or follow these steps:

1. Switch to 'View by Kind' mode
2. Deselect all icons by clicking in the gray area of the icon well.
3. Press and hold the 'down arrow' key. The icons are highlighted one by one from top to bottom, setting the click order by type and, within each type, in alphabetical order.
4. When the last icon has been selected, switch back to 'View by Icon' mode. Option-Clean Up arranges all of the icons alphabetically by type.

#### 5.2.2.2 Clean Up with the shift and option keys down

Holding both the shift and option keys down while selecting Clean up results in all of the icons being pulled together in the currently visible area of the window. In addition, each new icon type encountered is moved to the next line.



*Because of the 'icon click order' rule (see above) this mode can result in many short lines with one or two icons. To avoid this, use the trick outlined above to select each icon by type before cleaning them up.*

### 5.2.3 User Editor enhancements

A 'lost feature' in the User menu editor was found and revived: Helix has always supported pressing Command-Tab to create a new empty menu bar item. But Command-Tab is used to switch applications in OS 9 and later, and the command is no longer passed through to Helix. The key command has been changed to Command-Enter or Command-Return.

The 'starter slot' that is shown when creating a new menu in the User menu editor has been enlarged to make it more obvious.

### 5.2.4 User Mode bypass

When opening a collection, you can now go directly to Design Mode (bypassing User Mode) by holding down the Option key.

### 5.2.5 Show Clipboard command key equivalent

The Show Clipboard command now has a Design Mode command key equivalent: CMD-K.

## 5.3 Clippings Menu

### 5.3.1 Introduction

Built on Helix RADE's copy/paste structure feature, the Helix Clippings menu allows you to easily create structural elements for reuse in your collections. Users, sequences, relations, and the various icons within relations can be stored in Helix Clippings for application at any time.

Helix Clippings are stored in a folder on your computer in the standard Mac OS X clippings file format. They can be accessed via the Finder, and the Finder's Preview function can give you a glimpse inside the clipping. Helix Clippings made in **View by Icon** mode preview as a picture of the icons. Helix Clippings made in **View by Name** or **View by Kind** mode preview as text (the names of the icons).

Helix Clippings files can be copied like any other Macintosh file, moved from computer to computer, emailed, etc. However, they do contain a Macintosh resource fork, so care should be taken to avoid situations (FTP, FAT16 volumes, etc.) where the resource fork is stripped away.

Helix RADE Classic 6.1 includes a starter library of free Helix Clippings, and more are available on our web site. We also plan on using the Helix Clippings format to deliver more complex collection solutions, to be sold as commercial products. Both the free and commercial libraries are open for contributions from independent Helix designers. Contact us if you would like to submit Helix Clippings solutions.

### 5.3.2 Setup

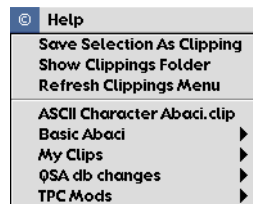
To activate the Helix Clippings menu, a folder (or alias to a folder) named **Clippings** must be found in the same folder as Helix RADE Classic 6.1. If you installed Helix RADE Classic 6.1 from our installer, a Clippings folder is created automatically and placed in the correct folder. However, we recommend that you move this folder to another location and use an alias to it, so that you do not inadvertently lose your Clippings when upgrading to a newer version of Helix RADE.

If you want to configure your Mac the way Helix RADE for Mac OS X will expect it, navigate to your user's ~/Library/Application Support folder and create a **Helix** folder in there. (If you have Helix Maintenance Manager installed, this folder already exists.) In this Helix folder, create a folder named **RADE** and put your Clippings folder in there. Then create an alias to that folder and put the alias in the folder with Helix RADE Classic 6.1. Make sure the alias is also named Clippings.

If the Clippings folder is found when a collection is opened with Helix RADE Classic 6.1, the new Clippings menu appears in the menu bar as a © symbol, immediately to the right of the Window menu. If the Clippings menu does not appear, make sure you have a collection open. If it still does not appear, quit Helix RADE, navigate to the folder where Helix RADE Classic 6.1 is stored, and make sure the folder (or alias) is named exactly. One common error is to have a trailing space after the name. It looks right, but Helix RADE does not see it.

### 5.3.3 Usage

The Clippings menu has three fixed commands, a divider, and the files found in the Clippings folder...



Helix Clippings Menu

#### 5.3.3.1 Save Selection As Clipping

**Save Selection As Clipping** creates a Helix Clipping containing the structure of the selected icons. One or more icons must be selected, or this command is disabled. When the command is chosen, a dialog box appears giving you an opportunity to name the clipping and to save it in any place you choose.



*Because this is a feature designed for Mac OS X, the Classic dialog box leaves much to be desired. The first time you use this command you must navigate to the Clippings folder, or the file will be saved in the default location as determined by your Classic OS. This is typically your **Documents** folder, so you may want to place an alias to the Clippings folder in that location to make navigation easier.*

Classic also suffers in that it does not know to hide the Mac OS X file extension. You can remove the file extension if you wish: Helix RADE understands that these are Helix Clippings either way.

**Save Selection as Clipping** uses the Classic clipboard to create Helix Clippings, so data on the clipboard will be replaced when a Clipping is created.

#### 5.3.3.2 Show Clippings Folder

**Show Clippings Folder** opens a window in the Finder and reveals the actual Clippings folder. This is useful if you want to copy your Clippings to another location, rearrange the contents of the menu, or add new items to the folder.

#### 5.3.3.3 Refresh Clippings Menu

**Refresh Clippings Menu** updates the contents of the Clippings menu by walking through the contents of the Clippings folder (as seen in the Finder). This is useful if you have added new items to the folder (or rearranged the contents) and need to force Helix to see those changes.

#### 5.3.3.4 Divider

The divider separates the fixed commands (above) from the actual Clippings data below.

#### 5.3.3.5 Clippings Data

Below the divider is the Clippings data. Anything placed in the Clippings folder is seen here. The Helix Clippings menu supports hierarchical menus up to 5 levels deep, so you can organize your Helix Clippings into logical projects.

When you select an item from the Clippings menu, that item is examined to determine if it contains Helix structural elements. If it does, the Paste Structure dialog opens allowing you to connect the structure found in the Clipping to the existing collection structure. The standard rules for pasting structural elements apply.

You can also place other files in the Helix Clippings menu. If the menu item does not appear to contain Helix structure, Helix RADE hands it off to the Finder with an Open command. Text files will open with a text editor, applications will launch, generic clippings will open as clippings, etc.



*One powerful technique is to place AppleScript (or other) applications in the Helix Clippings menu. The included **Quit, I'm Done** AppleScript demonstrates this capability by running as an applet that closes all the open windows, saves the collection, and quits Helix RADE.*

### 5.3.3.6 Additional notes

- Be sure to close all unnecessary windows before applying Helix Clippings. Helix can not update structure related to open icons. If the structure you are pasting references an icon in some other relation (e.g. via a lookup tile) and that relation window is open, the link can not be made and an incomplete icon is created.
- You can use CMD-1 (Next) to quickly cycle through the icons contained in the clipping file. For icons that are intended as updates of existing structure, choosing the icon from the 4th column establishes the link. For new icons CMD-2 triggers the New button.
- It is not necessary to explicitly create new icons. Simply clicking the OK button creates new icons for any that have not been explicitly linked to existing icons.
- The contents of Helix Clippings files are compared to any pre-selected icons on a type by type and name by name basis. If the clipping contains an icon of the same type and name as a pre-selected icon, Helix assumes you want to replace the old copy with the copy in the clipping.
- When first working with Helix Clippings, it is a good idea to open (press CMD-O) the created structure immediately after applying the clipping to make sure that all of the icons are referenced as expected.
- When pasting structure, it is possible to create structural anomalies that may be reported as damage by Update Collection or result in crashes when using Helix. One obvious example is a template used in a subform: replacing a template with a version that is larger than the original can create overlapping icons that Helix considers illegal. Should you apply a clipping that results in a crash or a broken collection, please do not throw the collection away: contact us immediately so we can help you recover from the damage without losing your work. This is almost always possible in these cases. We are also interested in seeing specific examples of structure that cause damage when applied.

### 5.3.3.7 See also

- New Features: “Pasting Structure: Rename Replaced Icons” on page 9
- Spec Changes: “Paste Structure dialog access” on page 13
- Bug Fixes: “Structure Copy/Paste crashes (TS2068/TS2069)” on page 17
- Bug Fixes: “Pasting Structure fails to identify top level icons (TS2027)” on page 17
- Known Problems: “Helix Clippings menu disabled after User Mode switch” on page 19
- Known Problems: “Helix Clippings menu item names” on page 19

## 5.4 Pasting Structure: Rename Replaced Icons

In prior versions of Helix RADE pasting structure does not update the names of icons that are replaced with updated structure. This could cause confusion if you replaced, for example, the template *TaxForm 2007* with *TaxForm 2008*, as the contents of the template would be updated, but the name would remain *TaxForm 2007*.

Helix RADE Classic 6.1 adds a **Rename Replaced Icons** checkbox to the Paste Structure Dialog. When this box is checked (it is by default) icons whose structure is replaced also have their names updated, using the name taken from the pasted structure. Unchecking this box preserves the existing icon names, just as in prior versions of Helix.



*The **Rename Replaced Icons** checkbox's keyboard shortcut is CMD-R.*

Icon names are not updated when bypassing the Paste Structure dialog: since Helix has no way of making a match between icons with different names, icons with names that do not match existing icons are always created as new icons unless you use the Paste Structure dialog to link them to existing icons.

## 5.5 Design Mode enhancements

*Each of these Design Mode changes debuted in Helix RADE Classic 6.1.2.*

### 5.5.1 Select unused icons

When working in a collection or relation window, holding the Option key down while choosing Select All now selects only unused icons. Adding the Shift key excludes indexes from the selection. (Indexes are often 'unused' but still required for optimal performance.)

The Command key equivalents (CMD-Option-A & CMD-Option-Shift-A) are also supported.

### 5.5.2 Accidental icon opening avoided

When rapidly selecting icons in Design Mode via shift-click, Helix RADE no longer misinterprets an accidental double-click as a request to open all of the selected icons. Shift-clicking an icon now deselects it if it is already selected.

To open a group of selected icons, release the Shift key, then double click on any selected icon.

### 5.5.3 Prevent window opening on switch to Design Mode

When opening a collection in Design Mode (or when switching to Design Mode from User Mode) holding the Option key down stops Helix RADE from opening all windows except the main collection window.

### 5.5.4 Design Mode icon zooming performance

The performance of the Design Mode zoom effect when icons are rejected is improved in two ways:

1. The number of steps in the animation is reduced from 8 to 3. The animation is now much faster.
2. When multiple icons are zooming back to their original positions, the animation now stops after the third icon. By then the user has typically gotten the message and the extended animation was deemed both time-consuming and annoying.

## 5.6 Notify On Change activation

**Notify On Change** is a new feature that allows you to set an attribute for each view that makes it update the Why? message (and therefore to beep) when a change is made to the content of that view. Anytime the view is redrawn to update data, the Why? message will be updated with a message that the data on a view has changed. The name of the view is also displayed.

**Notify On Change** is activated in the same manner as other view specific attributes (such as Shade Defaults or Keep Current: with the view open, select Notify On Change from the √ menu. The command displays a √ next to it when a view has this attribute set.



***Notify On Change** does not function in Helix RADE Classic 6.1 itself. This feature is part of the Mac OS X-native Helix Engine and Helix Client. Helix RADE Classic 6.1 is used to enable this feature for views used in those products.*

## 5.7 Keyword Indexing and the HKWT Resource

*This feature first appeared in Helix RADE Classic 6.1.1 and Helix Server 6.1.*

### 5.7.1 Introduction

In testing keywords in OS X Helix, a recurring problem was identified: the HKWT resource that defines the Keyword Separator Table is stored on the Helix application, not the collection. If a collection relies on a modified Keyword Separator Table, the user must remember to update

the HKWT resource with every new version of Helix installed. Failure to maintain the HKWT resource results in inconsistent results in Keyword-based searches.

A Helix Keyword Separator Table technote containing up-to-date information is available at: <http://www.qsatoolworks.com/support/technical/knowledgebase/kbfiles/technotes/hkwt/>

## 5.7.2 An important note on keyword index changes

When the HKWT resource is changed, Keyword Indexes in all affected collections must be rebuilt. Otherwise, existing index entries remain in the index, and the index will be unreliable. Currently there is no code in Helix to detect this situation and automatically rebuild Keyword Indexes. Helix Utility includes a **Break All Indexes** command, but that also breaks regular field indexes, which are not affected by this change.



*If the Keyword Index is not rebuilt, this problem can occur: when a keyword field is modified, the words in the field are added to the Keyword Index based on the new delimiters. However, the old entries for that field, based on the old HKWT, are not removed from the Keyword Index. (The keyword reindexing code searches for words to remove based on the current HKWT, not the one in effect when the data was previously entered.*

In summary: when changing the HKWT, be sure to rebuild all existing Keyword Indexes in any affected collections.

## 5.7.3 HKWT customizing for each collection

Starting in Helix RADE Classic 6.1.1, each collection can now have its own HKWT resource, providing a customized Keyword Separator Table. Helix now checks for the HKWT resource inside the collection before loading the default HKWT from the application.

The HKWT resource is not automatically added to new collections: it must be explicitly copied into the collection when collection-level specificity is desired. (See Keyword Management Utility below.)

## 5.7.4 Changes to the Keyword Separator Table

The following information is of interest only to Helix Client users who use keyword-based searches.

### 5.7.4.1 Missing characters added

In prior versions of Helix, many 'High ASCII' characters that are used frequently in European languages were being ignored as word characters. Consequently, words containing characters such as Å and Ø were excluded from keyword searches.

### 5.7.4.2 Ligatures added

The two ligatures included in the standard Macintosh font set — fi & fl — are now included in the word character set.

### 5.7.4.3 Non-breaking space added

The non-breaking space (NBSP) character was not previously considered a word character, even though the very definition of the non-breaking space speaks for its inclusion.

Helix 6.1 treats the non-breaking space — created by typing option-space — as a word character.

## 5.7.5 Inaccurate information

It was also discovered that the previously published information, such as that found in Appendix A of *The Helix Reference* is inaccurate. For example: É is noted as being a word character, but in the actual HKWT resource, it was being treated as a separator. Many similar inconsistencies were found and corrected.

### 5.7.5.1 Updated and corrected tables available

Tables showing the correct information for the HKWT resource in prior versions of Helix (replacing Appendix A of *The Helix Reference*) — as well as a table showing the new HKWT resource and a comparison chart showing the differences — can be found in the technote referenced above.

## 5.7.6 Keyword Management Utility

A new utility is under development to help manage HKWT resources. The HKWT Resource Manager can:

- Create a map of an existing HKWT
- Copy an HKWT from one location to another
- Modify an existing HKWT

The Keyword Management Utility (with instructions) is found on the technote referenced above.

### 5.7.6.1 Direct editing

You can also edit the HKWT resource directly using a resource editor. Information on resource editors can be found at <http://www.qsatoolworks.com/faq/>.

## 5.8 Keyboard shortcut modifiers supported (semi-complete feature)

In addition to the original Command key commands, you can now add Shift-Command, Option-Command, and Shift-Option-Command modifiers to user menus.

These modified command keys are added in the user editor by holding down the modifier keys while dragging the key to the user menu, the same way standard modifier keys are assigned.

However, the Classic User Editor doesn't quite know how to support these. As soon as you drag out a standard command key the keyboard representation of that key dims, making it impossible to assign commands using that primary key to other menu items. To work around this:

1. Remove the primary command key assignment (if necessary) so that the keyboard icon for that key is re-enabled.
2. While holding down the Shift, Option, or Shift-Option modifier keys, drag the key icon to the menu item it is to be assigned to.
3. When you release the key icon on the menu item, the modifier-enhanced version of the key command is drawn.
4. When you are done assigning key commands with additional modifiers, drag the un-modified key back to the desired menu item.



*The ability to assign modified command keys is actually a Helix RADE for Mac OS X feature. We are making this feature available in its semi-complete form so you can take advantage of it when the Mac OS X native versions of Helix Engine and Helix Client are released.*

Additional Notes:

- Because the Classic User editor does not know when an extended modifier key command has been used, it is possible to assign the same extended modifier key command to multiple commands. There is no guarantee as to which command will be executed if you do this.
- Certain modified command keys do not work as expected in Classic Helix programs. (Command-Option key combinations are the most often affected.) Be sure to test shortcuts in User Mode yourself before deploying them.

## 5.9 Navigation improvements

A few small improvements were made to make navigating in Design Mode a bit more efficient:

1. When working in a collection or relation window in list mode, you can now use CMD-down arrow to open a highlighted icon.
2. When working in an abacus, you can use the left/right arrow keys to switch from the tile list to the field (left) and abacus (right) lists. (These arrow keys do not yet work to switch from the abacus or field lists to another list.)

## 6 Specification Changes

*Helix RADE Classic 6.1.5 contains no specification changes. The changes listed below were introduced in prior Helix RADE Classic 6.1.x releases.*

### 6.1 Compiled forms eliminated

Unlike Helix 6.0.1 and earlier, Helix 6.1 does not rely on compiled forms. This is a significant 'under the hood' change, but we have handled it in a way that allows you to switch back and forth seamlessly between versions. Compiled forms were eliminated because they provide no performance gain on modern (PowerPC G3 or later) hardware.



*Because Helix 6.0 products rely on compiled forms, opening a collection in Helix Server or Helix Engine requires that forms be recompiled. This will happen automatically when a collection is opened.*

The **Compile Forms** command is no longer found on the Design Mode menu.

### 6.2 Paste Structure dialog access

In Helix RADE 6.0.1 and earlier, the Paste Structure dialog was not presented unless the designer held down the Option key while pasting structure. In Helix RADE Classic 6.1 the dialog is presented by default. To bypass the Paste Structure dialog, hold down both the Shift and Option keys while pasting.

For those who use the keyboard shortcuts: both Command-V and Command-Option-V now bring up the Paste Structure dialog. Command-Shift-Option-V bypasses it.

### 6.3 Default font for templates

The default font for label and data rectangles has changed from Geneva 12 point to Geneva 9 point. This was done to make 'default' code look better in collections used with Mac OS X native Helix products.

Helix font defaults can be changed by editing the HFAC resources. If you are not familiar with resource editing, contact technical support for assistance.

This change has no effect on existing structure. Only new structure is affected.

#### 6.3.1 Quick entry templates

A few changes were made to the default entry (aka: Quick Entry) templates:

1. The OS X style of indicating focus creates visual anomalies when rectangles are butted up against, or overlap, one another. Helix RADE has been modified to put a 4-pixel vertical gap between each field when creating a Quick Entry template. A 4-pixel horizontal gap is also created between the label and data rectangles.
2. The OS X style for data entry rectangles has a subtle 3D effect that takes some of the available space away from the text within. To compensate for this, the default font size for Quick Entry templates is now 9 point.
3. The width of label rectangles has been reduced slightly, and the width of data rectangles increased.

Quick Templates are created:

- When a designer fills a new template via the Quick Entry command
- When a designer creates a new entry view with no template specified and chooses Clear Form, Find First or Find Last.

This change has no effect on existing structure. Only new templates are affected.

#### 6.3.2 Quick list templates

Two changes were made to the default list (aka: Quick List) templates:

1. The width of rectangles created in default list (aka: Quick List) templates is slightly smaller.

2. The default font size for Quick Entry templates is now 9 point.

This change has no effect on existing structure. Only new templates are affected.

## 6.4 Design Mode commands and command key changes

The following Design Mode command keys have changed to avoid conflicts or to be more consistent with Mac OS X:

- **Quick Entry:** Shift-Command-E
- **Quick List:** Shift-Command-L
- **Quick Query:** Shift-Command-G
- **Open Query:** Shift-Command-H
- **Show Setup:** Shift-Command-I
- **Show Posting:** Shift-Command-J
- **Close Collection:** Shift-Command-Q (Also moved to just above the Quit command.)
- **Preferences:** Command-,

The following commands are now available in Design Mode:

- **Sort Order...**
- **Refresh**

The following commands have been removed from Design Mode:

- **Compile Forms**

## 6.5 User Mode commands and command key defaults

Commands and command keys can be added, removed, rearranged, etc. in the User Editor. The following changes have been made to the default set that is created when you create a new user.

When creating a new user, the following default command keys have changed to avoid conflicts or to be more consistent with Mac OS X:

- **Quick Query:** Shift-Command-G
- **Open Query:** Shift-Command-H
- **Preferences:** Command-,

When creating a new user, the following default command keys have been added:

- **Page Setup:** Shift-Command-P
- **Close Collection/Leave:** Shift-Command-Q
- **Close All:** Command-Option-W

When creating a new user, the default position for the following command has moved:

- **Autosave:** from the Set menu to the File menu, with the other Save related commands.

When creating a new user, the following commands are no longer part of the default set:

- **Custom Paper...**
- **Export Ordered...**

When creating a new user, the following commands are now part of the default set:

- **Sort Order...**
- **Refresh**

## 7 Bug fixes

### 7.1 Bugs fixed in Helix RADE Classic 6.1.5

#### 7.1.1 7603 Error on pressing Tab in collection/relation list view (R#6734)

If a collection or relation window is in list mode (**View by Kind** or **View by Name**) with a single icon selected, Helix RADE would crash if the window was toggled from **Edit** to **Fast Find** mode and then the Tab key was pressed. This has been fixed.

### 7.2 Bugs fixed in Helix RADE Classic 6.1.3

#### 7.2.1 Editing label rectangle containing a picture could corrupt database

A longstanding bug was discovered in Helix that could result in database corruption when editing a label rectangle containing a picture over 32K by double clicking the rectangle to access the settings dialog.

Fixing the actual bug would take more time than we want to devote to Classic Helix now, so to avoid the bug, Helix now displays a dialog instructing you to use menu items to make the desired attribute changes. (You can also cut the picture from the rectangle, open the dialog, make the desired changes, close the dialog, and repaste the picture, if preferred.)

#### 7.2.2 Reports with over 32,766 records failed to display all records (BZ976)

A bug that caused Helix Client Classic (and RADE) to fail to show all of the records in a report (list) when there were more than 32,766 records in the report is fixed.

#### 7.2.3 Option 1 post to cold form with defaulted field corrupts index (BZ900)

A rare combination of circumstances, an indexed field with a default value on a cold form using Option 1 posting to change the field value from undefined to the default value, resulted in corruption to the index built on that field. This bug is fixed.

#### 7.2.4 Fixed Point posting trigger multiplied by 100 (BZ1034)

A bug that caused conditional sequences to run 100 times the expected value when a fixed point value is used as the trigger *and* that same value is posted to another relation *and* that same value is displayed on the view has been fixed.

#### 7.2.5 Eszett (ß) character (0xA7) is a word character in German

When the keyword separator table was updated, the German eszett character was inadvertently left out of the word character list.



*Helix customers who use the eszett character in text stored in keyword fields need to rebuild all keyword indexes or keyword search results will be inaccurate.*

See “Keyword Indexing and the HKWT Resource” on page 10 for more information about keyword indexes.

#### 7.2.6 Keyword index progress dialog

The Keyword Index progress dialog would sometimes fail to show the name of the keyword field for which the index was being built. This was fixed in Helix Server OS X 6.0, but the fix had not been applied to Helix RADE until now.

#### 7.2.7 Command key for Shade Defaults changed

The Design Mode command key equivalent for Shade Defaults is changed to CMD-D to avoid conflict with OS X's use of CMD-OPT-D for Show/Hide Dock.

#### 7.2.8 Crash when pressing arrow keys

Opening a collection or relation window in **View by Icon** mode, clicking in the gray area of the icon well, and pressing an arrow key would cause a crash. This has been fixed.

Likewise, pressing an arrow key in a window in **View by Name** or **View by Kind** mode in a window had no icons would cause a crash. This has also been fixed.

### 7.2.9 Registration dialog issues (BZ904)

The registration dialog would not allow the user to switch to another program, copy data, switch back and paste that new data into a field. This is fixed.

## 7.3 Bugs fixed in Helix RADE Classic 6.1.2

### 7.3.1 AutoOpen Post with omitted key field corrupts index (BZ777)

A rare combination of factors would result in the corruption of an index, resulting in records appearing twice in lists, or deleted records appearing as a blank line in lists. This has been fixed.

Helix Utility has always detected this problem and scheduled the index for rebuilding, but lists could display inaccurate data between Helix Utility runs.

### 7.3.2 Crash Recovery: pictures over 64KB are lost (TS2233)

Pictures over 64KB, pasted into Picture fields, were not being recovered when the log file was applied after a crash. Upon recovery, the records containing the pictures were recovered, but the picture field was left empty.

This is now fixed. All data is now properly recovered from the log file after a crash.

### 7.3.3 Quick Query references missing from Get Info

If a field icon is currently used in a quick query, Get Info on that field would not report that it was used by the view. The field could not be thrown away (because it was in use) but Get Info would not reveal where it was used. This has been fixed. Get Info now shows the view in the list of references.

In addition, when such a view is opened by double clicking in the Get Info reference list, a Why? message is set informing that the field is used in the Quick Query.

### 7.3.4 Keyword and Simple Validations hide true reference count

Icons that are not referenced by other icons (and may therefore be thrown away) are indicated by text marker [-] at the beginning of the Contents column (in View by Name and View by Kind modes). Fields that are designated as Keyword fields or had simple validations attached were not showing the [-] marker even though they could be discarded. Now they do.

### 7.3.5 Potential crash on export

During export operations (both regular and Apple Event) an internal counter was set too high. This would occasionally cause a crash when Helix was running natively in Mac OS X. This bug has existed since at least Helix Express 4.0 (and probably since Helix 1.0) and although we never had a crash report in Classic Helix directly attributable to this bug, it was a theoretical possibility. Therefore the fix was also applied to Classic Helix.

### 7.3.6 Potential write lock failure

It was discovered that every four billionth attempt to write lock a record would fail, and the record could potentially be edited by two users simultaneously. A user would have to write lock a record once every second of every day for 136 years to reach this number, so we are confident this bug has never affected any Helix user to date. Nonetheless, it is now fixed and will not affect users no matter how many centuries their databases are kept in use.

### 7.3.7 'Mini menu' cleaned up

The items available in the 'mini menu' — the one that appears when you launch Helix RADE but have not yet opened a collection — contained a number of pointless commands that were always disabled. Those have been removed.

## 7.4 Bugs fixed in Helix RADE Classic 6.1.1

### 7.4.1 Paste structure with renaming on could crash

When pasting structure in View by Kind or View by Name mode, a 0305 error (and subsequent crash) could occur if Rename Replaced Icons was checked and the resultant renaming caused icons to move to a new location in the list. This bug is fixed.

### 7.4.2 Why? Message contains garbage text

In Helix 6.1 only: If a sequence encountered an error that prevented record entry, the Why? Message could contain garbage characters where the actual problem should have been reported. The same Enter command outside a sequence would properly report the specific error. From within a sequence only the generic 'The record could not be entered.' error was shown.

This bug is fixed: Helix now properly reports both the generic (sequence level) error and the specific (view level) error.

## 7.5 Bugs fixed in Helix RADE Classic 6.1

### 7.5.1 Find can fail when posting changes query criteria (BZ935)

A query restricted view could fail to find the correct records if the view contains a post that modifies a value that is used to calculate the query. This has been fixed.

### 7.5.2 Command rectangle duplication (TS2067)

When duplicating a command rectangle with the Button Name option set to Use Abacus Output, the referenced abacus is not copied to the new rectangle, but the Use Abacus Output flag is not reset to use the typed-in name. The result in older versions of Helix is a button with no label, unless you had also supplied a static label, in which case the static label is used.

Helix RADE Classic 6.1 and later properly reset the Button Name option and avoid the confusion. The duplicated rectangle is labeled using the name of the sequence until another name is designated by the collection designer.

### 7.5.3 Structure Copy/Paste crashes (TS2068/TS2069)

Crashes when copying and pasting structure have been fixed:

- Helix RADE may crash when attempting to paste a view if the view contains both posting and subforms.
- Helix crashes when attempting to copy/paste large amounts of structure.



*Both of these bugs are addressed in Helix RADE Classic 6.1. However, it is still possible to crash Helix by copying or pasting very large structures.*

### 7.5.4 Pasting Structure fails to identify top level icons (TS2027)

A longstanding bug in the Paste Structure dialog has been fixed. In previous versions of Helix RADE, clicking the Next button bypasses sequence and user icons. In addition, if a sequence or user icon is part of the structure to be pasted, the Next button eventually selects 'nothing' (even though the status message indicates that there are still unlinked icons) and loses the ability to select other icons.

### 7.5.5 Intermittent posting failures (TS2013)

In rare situations, posting mysteriously stops working after a view is in use for a while. When this bug strikes, the post stops working and does not work again until remedial action is taken.

This bug is extremely rare and only seems to affect relations where there are a large number of post icons. The bug actually affects the internal posting package that is created when you select various post icons in the view's four posting action columns.

This bug has been fixed by doing away with these packages altogether in Helix 6.1. The performance gain on modern hardware was negligible, and the added complexity was allowing subtle bugs like this to occur.

### 7.5.6 User view permissions inadvertently restricted (TS1992)

An obscure bug in the Classic Helix User Editor has been fixed. Helix keeps an internal table of user permissions for each view where a permission has been turned off. In a very specific situation, editing the menu of a user who did not have a particular view restricted in any way would result in that user inheriting restrictions for that view from the user doing the editing. This has been fixed.

### 7.5.7 Crash when locked icons are selected (TS1912)

Earlier versions of Helix RADE crash when trying to open a collection or relation window if *all* of the following conditions are true:

1. One and only one icon is selected
2. That icon is locked
3. The collection (or relation) window is in List Mode (**View by Name** or **View by Kind**)
4. The window was last used in Edit Mode (The toggle button at the top of the list has an A in it.)

Earlier versions of Helix RADE also crash if you select a locked icon and then click on the Edit/Fast Find button to toggle modes.

Both of these bugs are fixed in Helix RADE Classic 6.1.

### 7.5.8 Enter menu highlighting

Pressing a command key to execute a menu item should always highlight that menu briefly to provide feedback to the user. In Helix 4.5.3 this function was disabled in the case of the Enter command. The rationale was that when Enter is called repeatedly in a sequence, highlighting the menu is a distraction and reduces performance.

In Helix RADE Classic 6.1, pressing the Enter key properly highlights the menu when an Enter command is executed. When the Enter command is executed as part of a sequence, the menu is not highlighted.

### 7.5.9 Why? message during sequence: format error

When an error occurs during the running of a sequence, there can sometimes be two error messages generated. For example, if a sequence attempts to enter a record that can not be entered because a validation check fails, there are two errors involved: the error because of the validation failure and the error because the record can not be entered.

Prior versions of Helix concatenated these two errors into a single entity, sometimes producing messages that did not make sense as presented. Helix 6.1 splits these messages onto separate lines, making it more apparent that two errors are involved.

### 7.5.10 Build number missing from splash screen

Prior versions of Helix RADE were not displaying the build number in the splash screen, making it difficult to determine precisely which version of Helix was in use. The build number is now shown when Helix RADE launches and when selecting About Helix RADE from the Apple menu.

## 8 Known Problems

This document contains the complete list of known problems, taken from previous release notes as well as recent additions made to our web site Knowledgebase. Check the Knowledgebase at [www.qsatoolworks.com](http://www.qsatoolworks.com) for the latest updates to the known problem list.

### 8.1 New in Helix RADE Classic 6.1

#### 8.1.1 Helix Clippings menu disabled after User Mode switch

The Helix Clippings menu should always be enabled in Design Mode. However, if an active collection or relation window has icons selected at the time Helix RADE is switched to User Mode, when switching back to Design Mode the Helix Clippings menu becomes disabled from that point on unless at least one icon is selected in the active collection/relation window.

When this bug strikes, you can still access the Helix Clippings menu by simply selecting an icon in the active collection/relation window. You can also restore the correct behavior by deselecting all icons, switching to User Mode, and back to Design Mode.

To avoid this bug, be sure to deselect all icons before switching to User Mode.

#### 8.1.2 Helix Clippings menu item names

Because Classic file names are limited to 31 characters and because the Mac OS X extension for a clipping is `.pictClipping` or `.textClipping`, files that are given long names in the Finder appear in a shortened form with what appears to be random characters at the end. This is normal and should cause no concern. The clippings will function as intended. This problem will disappear when Helix RADE for Mac OS X is released.

Another problem with menu item names is that occasionally an item name in the Helix Clippings menu is replaced by the word *Bumper*. This is a harmless bug that can usually be resolved by selecting the **Refresh Clippings Menu** command and then waiting 5–10 seconds before doing anything else.

### 8.2 Pre-existing known problems

#### 8.2.1 Design Mode

##### 8.2.1.1 Editing a graphics rectangle via dialog can corrupt collection (BZ1051)

Changing the attributes of a label or command rectangle containing a graphic can result in collection damage in the following situation:

1. The label or command rectangle contains a graphic larger than 64KB
2. The attributes of the rectangle are modified by opening the rectangle attributes dialog

In this situation, the graphic is corrupted when the **OK** button is clicked (accepting the changes made in the dialog).

Once this corruption has occurred, the collection will fail to pass the Helix Utility **Data Damage Check**. Attempts to delete the rectangle may succeed, but it is more likely that Helix RADE will crash with a 5xxx error.

You can work around this bug in two ways:

1. Do not open the attributes dialog: use the menu commands instead.
2. Cut the graphic out of the rectangle, then open the dialog, make the desired attribute changes, and click **OK**. You can now safely paste the graphic back into the rectangle.

Helix RADE Classic 6.1.3 and later do not open the dialog when this bug is possible. Instead a Why? message informs the user:

This dialog can not be opened because the rectangle contains an image larger than 64KB. Use menu commands to change the attributes instead.

### 8.2.1.2 Deleting an icon can result in reported collection damage #1 (TS1042)

Throwing away a field or abacus that is no longer used can result in Update Collection reporting that the collection is damaged. (Diagnostic Code: 30/xxxx/xxxxxxxx)

The problem is that some other icon that formerly referred to the deleted icon has a corrupted 'backlink table' that causes it to believe it still needs the deleted icon. This is a known bug from older versions of Helix. Fortunately, the damage is benign and can be ignored until an opportunity to repair the collection is presented.

There are a number of options for dealing with this situation:

1. Reuse the icon. There is no harm in this at all as the affected backlink table will continue believing that the icon is needed until the object that uses the backlink table is modified in a way that causes the backlink table to be recreated, removing the incorrect reference.
2. If you can recall which icons may have referred to this icon in the past, recreate those icons and throw the originals away. This will also discard the associated backlink tables and allow you to delete the unused icon.
3. If you can recall which icons may have referred to this icon in the past, copy those icons to the clipboard and paste them back into the collection, replacing the originals with exact duplicates. This will force the backlink tables to be reconstructed. If you can not remember which icons referred to the deleted icon, you can also try copy/paste of entire relations.
4. Go to a recent backup, copy the icons you deleted, and paste them into the damaged collection. The backlink tables should now find the needed icons.
5. Throw away the icons and arrange for a collection repair. See "About collection repair" on page 3 for information on collection repair procedures and the charges involved.

### 8.2.1.3 Deleting an icon can result in reported collection damage #2 (BZ990)

Throwing away a field that was previously used in an abacus that is currently used by a query can result in Update Collection reporting that the collection is damaged. (Diagnostic Code: 36/xxxx/xxxxxxxx)

The damage caused by this issue is trivial and can be fixed simply by opening the query that contains the reference and making a change to the query specification. (In the case of a Permanent Power Query item, you have to go into the Permanent tab and deselect/reselect the items.)

The ultimate fix for this bug is quite involved and has been deferred until work on Helix RADE for OS X begins.

### 8.2.1.4 Index built on abacus can fail to invalidate

When an index is built on an abacus, and the contents of the abacus change, the index is not always invalidated.

This bug seems to be triggered by the fact that the abacus is used as the target in a lookup. However, it does not always occur.

### 8.2.1.5 Undo style change crashes

Select a rectangle on a template, make a style change (Bold, italic, etc.), and choose Undo. Helix RADE crashes. This bug has existed for years.

### 8.2.1.6 Icon placement

New icon placement when working in **View by Name** or **View by Kind** mode, or when pasting structure into a collection/relation window sometimes results in the icon being placed at coordinates that are far away from the other icons when **View by Icon** mode is used.

Beside the annoyance factor, this can also cause odd behavior, such as icons that cannot be deselected, icons that cannot be opened, icons that open even if they are not selected, icons that can not be dragged, icons that appear to be in the proper workspace but can not be manipulated, etc. When in **View by Icon** mode, these icons that appear partially hidden in a gray area at the edge of the window's work space. Dragging those icons back into the work space corrects the odd behavior.

### 8.2.1.7 Data format dialog/dynamic popup panel

If you open the field or data rectangle format dialog, switch to the dynamic popup panel, and drag the dialog to another location, the screen does not redraw properly.

**Workaround:** Toggle to another style, then back to dynamic popup.

### 8.2.1.8 Document based dynamic popups crash when edited

In versions of Helix we tested (as far back as 4.5.1) it has been possible to create a dynamic popup with a document field by following a specific order of steps. This functionality was never intended (in fact there is no code in place to handle such a case) and the fact that it works in some versions of Helix and not in others is a quirk of the compiler. Attempting to create such a popup in Helix RADE Classic 6.1 results in a crash. We will investigate the possibility of officially adding this capability in a future revision.

### 8.2.1.9 User editor crash with large number of objects

Attempting to open a user icon in a collection with a very large total number of relations, users, and sequences can result in a crash, reported by Helix as an “object has become too long” error.

### 8.2.1.10 Design Mode vertical scroll bar disappears in large relations (TS1648)

When working in Design Mode in **View by Kind** or **View by Name** mode, the vertical scroll bar becomes flaky or disappears altogether when there is large number of icons in the relation.

This is a known bug from older versions of Helix. No problems are seen with 2720 icons or less, but at exactly 2721 icons, the list begins to act flaky, and scrolling at the bottom of the list becomes erratic. At 2731 icons the vertical scroll bar disappears altogether.



*2700+ icons in a relation is extremely unusual. If you are affected by this bug, you may want to reevaluate your database design. It may be that you are trying to combine the function of multiple relations into one. Look for multiple fields holding the same data class (e.g. “item#1”, “item#2”, “item#3”, “item#4”, etc....). If you have this, you should move the “item” field to another relation and take advantage of the relational capabilities of Helix that show its true power.*

There are two workarounds:

1. Switch to **View by Icon** mode. The problem does not manifest in this mode.
2. Reduce the number of icons to 2720 or less.

### 8.2.1.11 Cosmetic problems

#### 8.2.1.11.1 Dragging in sequence window

Dragging a step up and down in a sequence list leaves some harmless visual artifacts.

## 8.2.2 User Mode

*User Mode problems also apply to the Classic version of Helix Client.*

### 8.2.2.1 Printing Issues

#### 8.2.2.1.1 Record data reverts when printed if post on print applies to the view

If an existing record is displayed and a text field is modified, the record can be printed with the altered (but not entered) text. Subsequently closing the record appropriately asks the user whether they want to save the changes or not.

If a post set to trigger “on print” is attached to the view, the record is printed with the altered text, but the record is immediately reverted to its original state after printing, and the user is not given an opportunity to save the changes.

To work around this problem, use a sequence to print, starting with a Static Enter command to prevent the record from being reverted after printing.

#### 8.2.2.1.2 Find and Print All with more than 128 records

Using **Find and Print All** to print more than 128 records with ‘n-up’ layout only prints the first 128 records in n-up form. Subsequent pages print with the 1-up setting. For example, printing 150 records with 4-up layout results in the first 128 records printing (at the properly reduced size) on 32 pages, followed by records 129–150 printing one per page.

To work around this problem, select your desired n-up setting in the print dialog, and then click the “Save Settings” button. This will tell the LaserWriter driver to use that setting for the entire print job. Remember to reset the print dialog to 1-up layout when you want to print records at full-size.

#### 8.2.2.1.3 Last visible item on a list view fails to print

The last item on a list view can fail to print even though it displays properly on screen. The problem is that the last time the “Page Setup” command was used on the view, a different printer (or different printer page size) was selected. When page sizes are changed, Apple’s printer drivers display a dialog that remind the user to “be sure to choose Page Setup in all open applications” and avoid printing issues. Because of Helix’s ability to remember page settings for playback in sequences, it always retains the previous Page Setup information. During printing, Helix first calculates how many records fit on each page. Unfortunately it uses the stored Page Setup information to do this calculation. If the printable area becomes slightly larger, as is common with newer printers and their smaller page margins, Helix’s old calculations are incorrect, and one or more records is trimmed off.

To solve the problem, choose Page Setup and click “OK” — all displayed records will now print.

#### 8.2.2.2 Inert field behavior

In certain instances (e.g. posting) the value contained in an inert field is only recognized after tabbing out of it. Even if the value showing in the data rectangle appears to be changed, processes depending on the changed value do not occur unless the inert field is tabbed through.

##### 8.2.2.2.1 “The Enter Key Cannot be Pressed Here”

If the first data rectangle on a view contains an inert field, Helix may refuse attempts to enter a record with a Why? message indicating that “The enter key cannot be pressed here.” There are a number of known workarounds, among them...

- When Helix refuses to enter the record, open the Why? message and dismiss it. The record can now be entered.
- When Helix refuses to enter the record, close the view. Helix will display the standard “Enter, Discard, or Cancel?” dialog. Clicking the Enter button stores the record. Note that this only works if the User Mode user has add permission for this view.
- Rearrange the view so the first field is not inert, or convert the field into a stored field.

#### 8.2.2.3 Option 4 posting in Find and Print All (BZ740)

A bug has been discovered that occurs during record printing if and only if you have all of the following conditions at the same time:

- The view is an entry view
- You use Find and Print All to print the records
- There are multiple records to be printed
- The view includes a subform
- At least one of the host view records has enough subform data to require more than a single page
- The view is set to Post on Print
- The post is an Option 4 post designed to delete the host record

In this situation, the records print properly until the first multi-paged record is encountered. After that record is printed, subsequent host records are deleted before they are printed, and their pages are printed with no data.

This bug can be avoided in one of two ways:

1. Make the host form into a list and use Print Form instead of Find and Print All to print.
2. Create a recursive sequence with Find First and Print Form commands and use that to print.

This bug has existed in Helix for a very long time, but it was not discovered until the Helix 5.2 test cycle. If you have never encountered it before, you do not need to worry that your collection will change behavior due to a Helix version upgrade.

#### 8.2.2.4 Deleted field reverts to its previous value after entry (TS958)

A field with a default abacus will not store a default value if...

1. The record was previously entered with a value in this field, and
2. You simply delete the stored value and tab out, letting the default 'drop in' to the field.

The conditions under which this bug strikes have been pinned down and it only strikes if *all* of the following are true...

1. The view has an option 0 post attached and
2. The view's Keep Current attribute is off.

This bug has existed since the release of Helix Express 3.0. To avoid this problem, turn the Keep Current attribute off for views that include both defaulted values and an option 0 post.

#### 8.2.2.5 Text to Styled Text display error

Changing the data type of a field from Text to Styled Text correctly changes the type of any abaci that the field is used in, but strange characters may appear on a view displaying such an abacus. If this occurs, open the template that the view uses and drag the abacus to the data rectangle in which it already resides. This will clear up the problem.

#### 8.2.2.6 Undefined indexes can compound collection damage (TSD22002)

Damage sustained by a collection is exacerbated if...

1. One or more indexes are created, but left undefined, or
2. Icons are removed from an existing index leaving an undefined index, but the index is not deleted.

When you are working in Design Mode, do not leave indexes undefined. Either fill them with at least one icon or delete them.

#### 8.2.2.7 Power Query text edit field issues

The text edit field in a power query window does not properly respond to mouse actions such as clicking or dragging.

**Workaround:** Arrow key editing functions do work as expected.

#### 8.2.2.8 Dynamic popups limited to 254 items

Dynamic popup menus are limited to 254 items. Popup lists with more than 254 items will display the non-functional text "More..." as the 255th item. This item does nothing. If the relation from which the dynamic data is drawn contains more than 254 items, you must use a query to limit the popups content.

#### 8.2.2.9 Subform or list won't open the selected entry view (TS960)

A list view with an attached entry view does not respond when attempting to open a record. Examining the repeat rectangle's dialog on the view's template shows that the selection is still active, but when you open the view and try to double click a record, one of the following is observed:

- As a subform: the cursor does not change to the hand symbol and double clicking does not open the entry view.
- As a list: the cursor changes to the hand symbol and the record highlights, but double clicking does not open the entry view.

The root issue is that the attached entry view has been put into show posting or show setup mode. The bug is that Helix RADE does nothing to indicate why the entry view is not responding to the attempts to open a record. The list view that references the entry view *should* beep, and the Why? message should inform you that the entry view is not in Show Form mode.

This bug is fixed in the upcoming Mac OS X native Helix products.

#### 8.2.2.10 Lookup tiles in dynamic popups: a caveat

In Helix RADE 6.0, a limitation in dynamic popup construction was removed. Earlier versions of Helix exclude abaci that include lookup tiles from the slot machine selection lists. We have since learned why this restriction existed.

Dynamic popup menus are truly dynamic: if you have a view open with a dynamic popup and data in the popup's source relation changes, your popup menu will be updated to reflect that change. However, in the case of a lookup value, the abacus containing the lookup tile is in one relation and the data being looked up is in another. A change in the "indirect" relation (the one

the lookup retrieves data from) is not reported to the dynamic popup, so no update occurs, and the data seen in the popup becomes inaccurate. In effect, a dynamic popup menu that relies on data values derived from an abacus that contains a lookup always has the characteristic of a cold form; i.e: Keep Current is deselected.

#### 8.2.2.11 Page Number overflow

When the “P# is Page Number” feature is used on a view that has more than 65,535 pages in it, or on a list that has more than 65,535 records in it, the counter will overflow and begin counting up from 0 again on the 65,536th element.

#### 8.2.2.12 Text Field overflow

Attempting to type more than 32,767 characters into a field will result in a Why? message stating the field cannot contain more than 32,767 characters. Other methods of adding characters to a field (e.g: importing, posting, Apple Events) can bypass this check, resulting in a crash.

#### 8.2.2.13 Large list record selection

Although Helix can view and scroll through lists containing millions of records without difficulty, you may still have problems viewing and scrolling through larger lists. However, Helix cannot properly select more than 16,382 records in a list.

Using Shift-click or Command-click to select more than 16,382 records results in a dialog box showing the total record count you originally selected, but noting that “The list has been truncated to 16,382 records.” The actual number of records selected is indeed 16,382, beginning with the first record you selected. Continued attempts to select more records via Shift-click or Command-click results in the selected record count being reported as 16,382, and only the original 16,382 records are actually selected. A subsequent Copy Selected Records copies only the highlighted records to the clipboard.

Choosing Select All Records in a list with more than 16,382 records results in a dialog box stating the total (correct) record count in the list but noting that only 16,382 records have been selected. In fact, *all records are selected* even though only the first 16,382 records are highlighted. A subsequent Copy Selected Records, Paste Selected Records, or Delete Selected Records acts on all of the records in the list, not just the highlighted records.

#### 8.2.2.14 Document handling

##### 8.2.2.14.1 Importing external documents

When importing into Helix, an external document path will result in an error if the file that the path resolves to can not be found. Because of the variety of ways that Helix is used to import and export document references, this problem requires further study before a solution can be proposed.

##### 8.2.2.14.2 External documents in text and EPS formats not displayed

The text and EPS document display code was originally custom written for use with Helix’s internal documents. Neither of these document types are supported by the QuickTime conversion engine, so they are not viewable when stored externally. To display the contents of text and EPS documents, the documents must be stored internally.

#### 8.2.2.15 Apple Events can be established but not fulfilled

Apple Events that are initiated while a collection is being opened, or when an expired copy of Helix is displaying the expiration dialog, can not be fulfilled and are left open indefinitely.

#### 8.2.2.16 Use of Helix Custom Help

Helix versions 4.5.3 and beyond do not officially support the old Helix Help or Helix Custom Help files. You can use old files if you insert the registered trademark symbol (®) after the word Helix (e.g. Helix® Custom Help).

#### 8.2.2.17 Sleep mode crash

Helix does not get along well with Sleep Mode. (Monitor sleep is not a problem.) Collections opened with Helix RADE should be closed if there is a chance that you will be away long enough to have your system enter the sleep mode.

Users who run in OS X Classic Mode should set their Classic Preferences Panel (Advanced Tab) to never put Classic to sleep or be conscientious about quitting Helix when not in use.

## 8.2.2.18 Rarities

### 8.2.2.18.1 Potential “Save As” leads to collection damage

It has been reported that in some instances, collections created via “Save As” are saved in a damaged state. We have never been able to reproduce this, but the seriousness of the issue leads us to warn against the potential data loss.

**Workaround:** Do not use Save As; Close the collection and use the Finder to duplicate it.

### 8.2.2.18.2 Extremely long save when importing many records

Many Helix users periodically experience long (15 minutes or more) save durations. The problem has to do with the fact that a page of the DAT (Data Allocation Table) has filled up. The DAT is used to track gaps in the collection file as they are reused by new data. When there are many gaps, the DAT table can become very large, resulting in very long save times. There are two workarounds. The preferred option is to compress the collection, which removes the gaps and prevents the long save times. The second option is to edit the HDTP resource, which controls the operation of the DAT table. Open the HDTP resource with a resource editor and change the first three occurrences of 16384 in that resource to 32768. Remember that this workaround actually just postpones the inevitable. Sooner or later the problem surfaces with large imports because all this change does is increase the size of a DAT page. It should postpone the occurrence of the long save that happens when a DAT page fills, and if a save is induced at some point before the DAT page fills no long save will occur.

It is also believed that turning off the HDTM resource can affect save times. If you are experiencing long saves on a regular basis, please contact technical support for details on testing this theory.

## 8.2.2.19 Cosmetic problems

### 8.2.2.19.1 Proportional scroll boxes and asynchronous views

Because Helix draws views asynchronously (that is, it draws the view first, then dispatches a separate routine to draw the data while it carries on with other tasks) it is impossible to properly set the vertical proportional scroll box of a view when it is opened. After the view has filled in the data, the scroll box may or may not recalculate and display with correct proportionality.

**Workaround:** Deactivating and reactivating the view, or simply resizing it slightly after the data has filled in enables Helix to properly calculate the proportional scroll box.

### 8.2.2.19.2 Color icons in user menus

All of the color swatches in the color menus of newly created collections include a 1 pixel black border to make it easier to distinguish light shades against the grayscale dialog and menu background. However collections created in Helix 5.0.2 or earlier do not show this border because the cicon resources already exist in the collection and writing code to alter them is impractical at this time.

**Workaround:** If you want the borders to appear in your collections, you have two options...

1. Open the collection with a resource editor and delete the cicon resource. The next time you open the collection, the cicon resource will be rebuilt by Helix with the 1 pixel black border around the swatches. Important: if you have defined any custom colors for your collection, you will have to redefine them.
2. Open the collection with a resource editor, open the cicon resource, and manually draw a black border around each color swatch that appears in your color menu.

### 8.2.2.19.3 Import/Export dialog options command key

In import and export dialogs, CMD-O is the keyboard equivalent for the “Options” button. If you select a folder in the directory list, the “Import” or “Export” button changes to read “Open” as it should. Pressing CMD-O while the button reads “Open” flashes both the Options and the Open buttons, even though only the Options command is executed. This is a cosmetic problem only.

### 8.2.2.19.4 Delete data dialog when subforms are involved

The delete data confirmation dialog that appears when a form with subforms is used does not center over the parent window as it should.

### 8.2.2.19.5 Appearance manager compliance

The “Standard File” dialogs (Open collection, Get Document, etc.) can’t be made appearance compliant. They will be replaced with Navigation Services style dialogs in the Mac OS X release.

# 9 Functional Clarifications

## 9.1 Design Mode

### 9.1.1 Removing icons from rectangles

In early versions of Helix you could remove a field or abacus from a rectangle by double clicking on the rectangle, scrolling down to the end of the icon list, and clicking in the thin white space below the last entry in the list. This is no longer possible.

To remove an icon from a rectangle, open the rectangle, click anywhere in the icon's list, and drag the pointer out of the list box to the top, bottom, or right. When the pointer leaves the list area, the item will be deselected.

## 9.2 User Mode

### 9.2.1 Two-digit date input

Helix versions 4.5.1 and earlier always interpret a two digit year input as being in the 1900s. Helix 4.5.3 and later use an HY2K resource to determine the beginning of a 100 year range which is used to interpret a user's two digit year input. The default value for this resource is 1920, which sets the interpreted range to 1920–2019. Users should be informed that if they enter a year as a two digit number, Helix will interpret the date according to these rules.

### 9.2.2 'Previous' tile behavior when undefined fields are encountered

If a 'Previous' tile (Previous, Form Previous, and Previous For) encounters a record whose target field is undefined, it continues to search back (even through multiple records) until it finds a defined value to return. Two previous tiles on the same view, displaying the 'previous' value of two different fields, will actually display data from *two different records* if one of the target fields is undefined in the actual record that precedes the displayed record.



*This behavior has been consistent since at least Double Helix 3.5r13 (and probably earlier).*

If you need to know whether a previous record's field is defined or not, use the previous tile to return a value known to always be defined and unique (e.g.: record number) then use a Lookup tile to look up the target field. Alternatively you can use an 'Undefined Becomes' tile to make sure the value is always defined and reference that abacus in the Previous tile.

### 9.2.3 Index limitations

Indexes are limited in the length of the significant data. Each additional level of sorting reduces the total length of significant data. The exact length of the significant data is dependent on the types of data in the index. You cannot rely on precise sorting if the total length of the data for all fields and abaci in the index exceeds 200 characters. Each additional sorting level further reduces the length of significant data by approximately 4 characters.

### 9.2.4 Apple Events: after a delete, subsequent counts are wrong

After deleting a record (selector 150) another count (selector 120) results in the original record count being returned again.

When a view is opened via an Apple Event, it is always treated as though the Keep Current option is turned off. Consequently the count is not revised when records are deleted or added. You must close and reopen a connection to the view to retrieve the updated record data.

### 9.2.5 About the recovery file

Helix's crash recovery file (i.e. the suitcases icon) normally resides in an invisible Temporary Items folder, which is hidden in the trash. When you restart your computer after a crash, you will notice that your trash can appears full. Do not empty the trash or touch anything in the Temporary Items folder. Relaunch your collection and it will retrieve what it needs from this folder. Then save the collection and quit Helix. The computer will return to the Finder, and you may now safely empty the trash.

Advanced users can control the destination of the recovery by setting the HTMP resource in the Helix application and the HRFL resource in the Helix collection. The default value of the HTMP is 0000, and the default value of the HRFL is blank, which puts the recovery file in the Temporary Items folder of the local volume with the most free space. Changing the HTMP resource to 0001 puts the recovery file at the root level of the volume with the most free space. Changing the HRFL resource allows you to specify the volume the file is stored on. Contact technical support if you need assistance modifying these resources.

## 9.3 Rounding in Helix

Rounding is generally a pretty straightforward topic: when you round, round to the nearest number for the level of precision specified. Confusion is introduced when the value to be rounded is exactly halfway between the two possible values. This section is designed to explain and document how Helix rounds those values.



*For the sake of illustration, we will deal with rounding to the nearest integer, typically done with Helix's "Round" tile. Keep in mind that these same rules apply to the "Round to Nearest" tile, which allows you to specify the rounding precision.*

Rounding always seeks the nearest value to round to, but when the original value is exactly halfway between the target values, Helix employs one of two distinct sets of rounding rules. The rounding rules used depend on the data type being rounded.

The rules below only apply to values that fall exactly halfway between the two possible values. All other values round to the nearest number.

### 9.3.1 Rounding Fixed Points: Financial Rounding

Fixed Point data (technically, *integers*) were introduced much later in Helix's history and have always used the financial rounding method. Financial rounding seeks to make rounding more predictable, so it rounds *away from zero*, thereby guaranteeing consistent rounding.



*Financial rounding is also known as "symmetric arithmetic rounding," "banker's rounding," and a few others names. There are also variations on the basic rules for financial rounding — always rounding toward zero is one common variation — so be aware that not everybody has the same rules in mind when speaking about financial rounding.*

To see financial rounding in another setting, open Apple's ScriptEditor and run this AppleScript (make sure the event log is open so you can see the results).

```
repeat with i from -10 to 10
  get round (i + 0.5) rounding as taught in school
end repeat
```

In AppleScript, as taught in school uses financial rounding, and you will get the exact answers that Helix produces when rounding data in fixed point format.

### 9.3.2 Rounding Numbers: Scientific Rounding

Number data (technically, *floating point numbers*) were introduced in the very first version of Helix and have always used the scientific rounding method. Scientific rounding seeks to minimize cumulative errors, so it rounds to the nearest *even* number, thereby reducing the possibility that the result will be skewed.

To see scientific rounding in another setting, open Apple's ScriptEditor and run this simple AppleScript (make sure the event log is open so you can see the results).

```
repeat with i from -10 to 10
  get round (i + 0.5) rounding to nearest
end repeat
```

In AppleScript, to nearest uses scientific rounding, giving you the exact answers Helix produces when rounding data in number format.

#### 9.3.2.1 Decimal to Binary Conversion Errors

Certain numbers can be seen to violate the specification for scientific rounding. This problem is introduced because computers typically convert decimal (as in base 10) numbers into their

binary (base 2) equivalents before doing mathematical operations. In decimal math, we have many fractional values (e.g.  $1/3$ ) that we understand to be 'infinitely repeating decimal numbers.' Attempting to divide 100 identical items evenly between three people is impossible. Doing mathematical operations along this line introduces rounding errors that we naturally understand and compensate for. A human being can look at  $((100/3)*3)$  and understand that the answer is 100, but if you work it out, the answer comes out as 99.999... and we simply 'round it off' to 100. But it is important to keep in mind that 100 is an approximation, the 'real' answer is 99.999...

When examining binary numbers, you find that a whole different series of fractional numbers turn out to be infinitely repeating. For example  $1/10$  is an infinitely repeating binary number.

Now consider how this applies to rounding. Given the number 0.235 and being asked to round to the nearest 0.01, you would apply the financial rounding rules and arrive at the (correct) answer of 0.24. However, Helix rounds this to 0.23.

Why? The answer is binary conversion error.  $235/1000$  is, when expressed as a binary number, an infinitely repeating number. Converting  $235/1000$  to binary and then back to decimal will yield (approximately) 0.23499999... Because *this* value is not exactly halfway between the two numbers we are potentially rounding to, it is rounded to the nearest value (down, in this case) and the result appears incorrect.

Why does Helix make this mistake? The math routines that Helix uses are part of the Macintosh CPU's ROM: they are part of the common package that most programs use. Open your Classic Calculator DA and type  $1 \div .9 = .1 =$  and you'll see that the answer is  $2.032879E-20$ , *not* 0. Remember that  $1/10$  (0.1 in decimal notation) can not be accurately represented in binary: it is an infinitely repeating value in binary. The value has to be approximated, and the minute error is seen when math operations are performed.

If you do not need more than two decimal places of precision, you can avoid these rounding errors by converting the number into a Fixed Point data type before performing math operations, converting the result back to a number data type if necessary.