



uniPaaS V1.9h

Release Notes

We are proud to introduce **uniPaaS V1.9h**. uniPaaS V1.9h is an improved and updated version of the uniPaaS V1.9 release. Read the information in this document to find out more about this latest uniPaaS version.

For more information about previous uniPaaS releases, see the [PastReleaseNotes.pdf](#) file provided with this installation.

General Information

Introducing Magic Software's uniPaaS V1.9h

We are delighted to announce the launch of a new release of Magic Software's uniPaaS SaaS-Enabled Application Platform.

uniPaaS allows organizations to quickly and cost effectively enjoy all the benefits of Rich Internet Applications (RIA) and Software-as-a-Service (SaaS) applications, whether on-premise, or on-demand.

Based upon a unique, unitary development paradigm, uniPaaS gives the power to quickly develop, enhance, and deploy business applications under multiple deployment models and at a fraction of the cost and time compared to conventional .NET or Java environments.

Further Enhancing the Platform

uniPaaS V1.9 introduces many new features and fixes. These new features enable you to accelerate application development and to produce more sophisticated and manageable business applications. This long list is added to the continuous effort of optimizing and proofing the product by addressing and fixing customer and internal reported problems.

Continued Platform Support and Compatibility

uniPaaS was tested and is confirmed to be compatible with Windows 7 32-bit and Windows Server 2008 64-bit operating systems.

Note: Due to the Windows 7 UAC, when installing and running uniPaaS on Windows 7, your user must have admin rights. You may also need to right-click on the uniPaaS Studio icon and click the **Run as administrator** option.

For more information about the various platforms on which uniPaaS V1.9h has been checked for operation by Magic Software Enterprises, refer to the **Compatibility Guide.pdf** file provided with this installation.

.NET Integration

As of uniPaaS V1.8, you can enhance your offering by easily embedding and integrating any .NET control or assembly.

You can upgrade the look and feel of your Rich Internet Application by directly placing new .NET controls into your RIA user interface. You can also enhance the functionality of your RIA by integrating any .NET assembly.

To use the .NET functionality, you need to have .NET framework V2.0 SP1 (or above) installed on your machine. You can install it by running the NetFx20SP1_x86.exe file in the Scripts\RIA folder.

For more information on how to add and manipulate .NET modules, see the sample available at: <http://kb.magicsoftware.com/article.aspx?article=1959&p=5> or the .NET Tutorial sample installed with uniPaaS.

Note that if your pre-version 1.9 programs use .NET objects, you must resave the programs that refer to .NET objects using the V1.9x service pack (use ^S). Alternatively, you can export and import the programs.

Mobile Device Support

Expand your IT offering by supporting mobile devices alongside stationary desktops. With the same high productivity of development and by keeping your own expertise in developing uniPaaS applications you can now easily produce highly interactive mobile applications and quickly expand your internal systems to serve the mobile workforce.

uniPaaS Web Page

Make sure to visit our [uniPaaS Web page](#), where you can view and download various documents.

uniPaaS Licensing

In addition to the new unitary development and deployment paradigm for RIA and SaaS, uniPaaS also supports any previous Magic Software editions and forms of development and deployment. However, in order to maintain your former development and deployment capabilities, you need to obtain new uniPaaS licenses that reflect your current eDeveloper V10 licenses.

To obtain uniPaaS licenses, please contact your local Magic Software representative.

Migrating from eDeveloper V10.1 to uniPaaS

Migrating an eDeveloper V10.1 application to uniPaaS is quite straightforward and no explicit migration procedure is required.

You can directly access and open eDeveloper V10 applications from the uniPaaS Studio and Runtime engines.

Migrating from eDeveloper V9.4 to uniPaaS

Migrating an eDeveloper V9.4 application to uniPaaS is fast and easy.

uniPaaS provides a collection of wizards to easily migrate your application, INI settings, and interface builder data.

In the Migration subfolder of the product, located in the Start menu of your desktop, you will find a shortcut to each of the available conversion wizards.

We recommend reading the Migration chapter in the *uniPaaS Help* before migrating your eDeveloper V9.4 applications.

Bundled Products

uniPaaS comes with the following bundled products:

- Limited version of DB2 Express – limited to 3 months
- Version Control Client – Jalindi Igloo
- The Magic Optimizer
- The Record Main Converter

The Magic Optimizer

Magic Software is happy to introduce Ocean Solutions's complementary utility – the Magic Optimizer. The Magic Optimizer provides extended query and analysis capabilities for any uniPaaS application.

This version of the Magic Optimizer provides a fully operational documentation feature, and a limited evaluation version of the remaining features.

To obtain a fully operational license of the Magic Optimizer, please contact Ocean Solutions at info@magic-optimizer.com.

From Record Main Based to Event Driven Applications

Turn your applications into fully event driven by utilizing the new Record Main Converter utility. The Record Main Converter utility locates all of the Record Main based logic of a uniPaaS application and automatically and fully converts it to event based logic while keeping the entire functionality of the application intact.

Making your applications completely event driven enables you to utilize new features, such as Subforms and Explicit Tabbing order of controls that are not available in tasks that have the Record Main Compatible logic unit.

But most importantly, this conversion brings your application one big leap forward towards turning the application into a Rich Internet Application (RIA), allowing you to start and benefit from the great values of RIA sooner than ever before.

Note that the Record Main Converter utility might not handle specific complex logic segments. In such cases, the complex logic segment will be kept as-is to maintain the application functionality and to allow for manual intervention to complete the conversion.

Downloadable Databases

You can connect your uniPaaS application to various databases, including all of the previous eDeveloper V10 database gateways.

Below are some links to downloadable express version databases.

Make sure that the corresponding uniPaaS database gateway is included as part of the uniPaaS installation.

- [Oracle Database 10g Express Edition](#)
- [Microsoft SQL Server 2005 Express Edition](#)
- [MySQL 5.0](#)

Rich Internet Application on Apache Web Server

To run a RIA application on an Apache Web Server, the following lines should be added to the **httpd.conf** file:

```
AddType application/x-ms-application .application
AddType application/x-ms-application .manifest
AddType application/x-msdownload .dll
AddType application/octet-stream deploy
AddHandler default-handler .jpg .gif .js .txt .bat .msi
```

Web and RIA development on Windows® 7

By default, IIS 7.5 is not installed on Windows® 7.

To run Web programs or Rich Client programs using a local IIS Web server, you may need to install it manually. You can install IIS by going to **Control Panel > Programs > Advanced Options > Windows Features**.

For more information, refer to the following URLs:

<http://technet.microsoft.com/en-us/library/cc731911.aspx>

<http://technet.microsoft.com/en-us/library/cc725762.aspx>

Web and RIA Development on Windows® Vista

The uniPaaS installation cannot enable the CGI and ISAPI Extension on the IIS of Windows® Vista. If you have troubles running RIA on Windows Vista, refer to the 'Configuring IIS7 to Work with uniPaaS' technical note, which you can find in the uniPaaS Help.

RTL Applications

It is advisable to use the same const file for development and deployment; otherwise, some issues may arise, such as incorrect alignment of a Label control in RTL fields.

Sample Projects

The following samples projects were added to the product and they are now installed in the 'SampleProjects' folder:

- Rich Internet Demo – The project of the live RIA Demo that is available at <http://riademo.magicsoftware.com> and <http://riademo.magicsoftware.com/mobile>
- Online Samples – A set of Online (Client/Server) examples for many Client/Server “how to” scenarios that are described in the Mastering uniPaaS book.
- Rich Internet Samples – A set of Rich Internet examples for many Rich Client “how to” scenarios that are described in the Mastering uniPaaS book.
- Mobile Web Samples – Web templates based on HTML Merge technology for mobile devices.
- .NET Tutorial – A short tutorial with examples explaining how to use .NET in uniPaaS.

All the projects are available with their source code, so you can use them in your applications.

SQLite Gateway

The SQLite database is a single user, self-contained transactional SQL database engine that does not require a server or any installation or configuration.

The SQLite gateway is integrated in uniPaaS and available with every installation of uniPaaS.

Systinet 6.6 and Java 1.6

Since version 1.9e, uniPaaS installs Systinet 6.6 and JRE 1.6. This version of Systinet supports Java 1.6.

Java 1.4 will not be supported when working with SSJ or using the JExplore() function.

While upgrading uniPaaS from versions earlier than 1.9e and you get the "Web Application Internal Error" message, carry out the following steps:

1. Remove or rename every subfolder under the **%WASP_HOME%\work** subfolder (including the **system** subfolder and one subfolder per server package).
2. Reload Systinet – Systinet will regenerate the entire content of the **work** folder.

New Features, Feature Enhancements and Behavior Changes

Range and Locate on an Asterisk Value

The Range and Locate mechanism was changed to allow ranging or locating on the asterisk character.

Performing range or locate on a string that contains * will treat the asterisk as a character and not a wildcard character.

To perform the range or locate on * where * is for a wildcard character, do the range on *,

Rich Client – Post Refresh By Parent Event

A new internal event was added to execute logic every time the subform is refreshed.

This event is raised before the Record Prefix in the subform task when the subform is refreshed by the parent. The event is raised when one of the following occurs:

1. The subform is defined with Auto Refresh=Y and a variable sent to the subform as a parameter is changed.
2. The Subform Refresh event is raised.
3. The subform is executed for the first time.

The event is not raised when a 'View Refresh' is raised in the subform task.

Rich Client – Single Sign On

The Rich Client now supports Windows Integrated Authentication to proxy servers that support this authentication mode.

Sample Projects

Additional sample programs were added to the Online Samples and Rich Internet Samples projects. The new sample programs are for some of the uniPaaS functions and for connectivity to Google Calendar, Google Blogger, Facebook, LinkedIn and Twitter. Note that some of these samples are available in the Rich Internet Samples project only.

Fixed Problems

ActiveX – COM Server (QCR # 190506)

The length of data received by uniPaaS serving as a COM server was smaller than in previous versions.

Log – Daylight Savings Time (QCR # 264665)

The time was wrong in the Monitor when daylight savings time was used.

Printing – Hebrew Version (QCR # 446177)

There was a difference between uniPaaS & eDeveloper V9 in the output file when using Visual <> Logical = Yes and Flip=Yes.

Performance – Data Refresh (QCR # 508300)

The form was refreshed often and caused a performance issue compared to V9.4.

User Range – Clear Template (QCR # 513511)

The user range was not cleared after a View Refresh event even though the range was cleared using the Clear Template option (F3).

Rich Client – Multiline Edit Control (QCR # 608217)

A vertical scroll bar was displayed in a multiline Edit control defined with Vertical Scroll set to No when another multiline Edit control defined with Vertical Scroll set to Yes was placed on the form.

User Function – Null Value (QCR # 709919)

A user-defined function did not work when sending a Null value to it.

Multilingual Support – Tooltip (QCR # 711648)

The wrong translation was shown for tooltip text when the original text was longer than the translated one.

Rich Client – Subform (QCR # 713596)

The Record Suffix and Control level handlers of the subform task were wrongly executed after right clicking in the subform area.

Web Service Lite – Namespace Information (QCR # 721851)

The namespace information was not added properly to the root element.

Rich Client – Push Button Events (QCR # 722115)

The Record Suffix logic unit of a subtask was executed after clicking on a push button without exiting the record when the subform was the first control in the tab order.

Components – Context Menu (QCR # 722764)

A component program was executed when running a program from a host program's context menu.

Broker – Monitor (QCR # 725907)

The Broker Monitor displayed the wrong number of contexts after terminating a context using the RqCtxTrm() function.

Gateways – MySQL (QCR # 726510)

An error occurred when deleting a record from a MySQL table and performing view refresh in a specific scenario.

ISAM/400 – Library Name (QCR # 728620)

uniPaaS crashed after performing APG on an ISAM/400 data source if the data source name was in the format of '<library>/<file>' and the library name length was 10 chars.

Rich Client – Rich Text Edit Control (QCR # 729159)

The text decorations were removed when copying the RTF text from a Rich Text Edit control to WordPad.

Rich Client – Timeout (QCR # 729894)

An error occurred when trying to re-authenticate after a timeout had occurred.

XML Views– Decimal Numbers (QCR # 730324)

Trailing zeros were removed after inserting data into an XML data source.

Gateways – DB2/400 (QCR # 733320)

uniPaaS crashed after performing APG on a DB2/400 data source when the data source had logical names in the index fields' names.

Unicode Fields – Alternate Collating Sequence (QCR # 760508)

Alternate collating sequence did not work properly when used on Unicode fields.

ActiveX – Subforms (QCR # 768753)

The Click event on an ActiveX toolbar did not work when clicking on the ActiveX while parked on the sibling subform.

Rich Client – Hebrew Tooltip (QCR # 777767)

The tooltip alignment was set to left instead of right in the Hebrew version.

Rich Client – Subforms (QCR # 777939)

- The Empty Dataview event was raised twice after calling a program in a subform that had an empty dataview with parkable controls.
- The Empty Dataview event was not raised after calling a program in a subform that had an empty dataview with no parkable controls.

Rich Client – Portuguese Language (QCR # 779678)

Some controls were not seen on an Edit control when using the ABNT2 keyboard layout on Windows 7 with the Portuguese language.

Rich Client – List Box Control (QCR # 782569)

uniPaaS crashed after pressing the Left or Right keyboard keys when a Numeric variable was attached to a List Box control.

Components – SDI (QCR # 783187)

uniPaaS crashed when two SDI programs performed Call by Name operations to two batch tasks from two different components.

Rich Client – Combo Box Control (QCR # 783602)

uniPaaS crashed when a table with a combo box field was browsed and the value in the combo box was changed.

Rich Client – Tab Control (QCR # 784275)

After clicking on the selected tab, the focus switched to another tab instead of staying on the selected one when the variable connected to the tab had an init expression, a default value or was updated previously.

Printing – PDF (QCR # 785187)

A 'Demo Version' watermark was printed in some PDFs when printing several reports simultaneously to a PDF using parallel programs,

RM Converter – Components (QCR # 786062)

Wrong control names were set when the data source was from a component. As of this version, these controls will get a generic name.

ActiveX – Direct SQL (QCR # 793879)

A memory leak occurred in some scenarios involving ActiveX controls and a Direct SQL task.

Subform – Close Event (QCR # 794247)

Close Event handlers were not executed after closing the parent window using the close button when the focus was in the subform task.

Rich Client – Subforms (QCR # 796645)

Wrong controls were displayed when several controls were placed on top of each other and the controls' visibility was set according to a value from a linked table.

Context Menu – Complex Projects (QCR # 799515)

The Context menu did not work properly in complex projects.

Rich Client – VecGet() Function (QCR # 801523)

Using the VecGet() function on a non-updated cell did not return the cell default value when VecSet() was used on the client side and VecGet() was used on the server side or on the Task Suffix.

Unicode Fields – Null Value (QCR # 802337)

Garbage characters were displayed when an expression with a Null value was attached to a Unicode field.

Image Control – Scaled to Fit (QCR # 914661)

The display of an image was distorted when the Image style was set to "Scaled to Fit".

CRR – .NET (QCR # 916024)

An error message was shown after adding a .NET DLL to the CRR when the DLL's path was a network path.

Rich Client – Link Query (QCR # 917576)

Wrong values were retrieved from a Link Query when a variable was defined in the Success Indication property in a specific scenario.

Components – Load (QCR # 919206 and 934084)

uniPaaS failed to open a project with many defined components.

Rich Client – Control Verification (QCR # 925408)

The Control Verification logic unit was not executed if Cancel was performed in a subtask.

Rich Client – Push Button Controls (QCR # 926088)

Text entered into an Edit control was displayed on a button after the cursor hovered over the button when the Button style was Hyper Text, the Transparent color was used, and the button was placed in a specific location.

AS400 – Get Definition (QCR # 928309)

uniPaaS crashed after performing a Get Definition from an AS400/ISAM collection (library) when the length of the collection name was 10 characters.

Rich Client – Direct SQL (QCR # 929156)

uniPaaS crashed when Rich Client Direct SQL tasks were closed in a different order than they were opened.

MSSQL – Application Role (QCR # 929501)

uniPaaS crashed when connecting to MSSQL using an application role,

Rich Client – Form State Persistency (QCR # 929630)

The columns in a Table control were not displayed properly after reducing the Rich Client frame size when closing and re-executing the program.

Components – Import (QCR # 930788)

uniPaaS crashed after importing a project in a specific scenario where components were used.

Rich Client – Incremental Locate (QCR # 932630)

uniPaaS crashed when performing incremental locate in a table that ran inside a frame.

Rich Client – Hebrew Version (QCR # 933000)

The H picture did not work properly when the control was running inside a subform task in the Hebrew version.

Rich Client – Incremental Locate (QCR # 933189)

uniPaaS crashed after performing an incremental locate in a specific scenario.

Gateways – Oracle (QCR # 934290)

An error message appeared when using Oracle 11gR2 in a specific scenario.

Rich Client – Checker (QCR # 935678)

The Checker hanged when performed on a Rich Client program in a specific scenario.

Rich Client – Record Suffix (QCR # 936558)

The Record Suffix of the previous record was executed when performing Cancel.

Subtask Tree – Manipulation (QCR # 938208)

An error was shown after copying and pasting tasks in the subtasks tree and then adding a new expression to the Expression Editor.

Expressions – Stat() Function (QCR # 939278)

The wrong value was returned for an expression with the Stat() function when used in an init expression in a specific scenario.

Browser Client – Verify (QCR # 939685)

A Browser Client task ended and the message was not shown when there was a Verify Warning operation with accented characters.

RM Converter – First Control Operations (QCR # 939812)

After using the RM Converter, a wrong condition was set for operations that were written before the first control.

As of this version, the operations that were moved to the Record Prefix logic unit will be executed always and the operations that were moved to the Control Prefix logic unit of the first control will be conditioned with KbGet(1)='Next Field'EVENT so they will only be executed when tabbing to the control.

Rich Client – Public Name (QCR # 940280)

The requester crashed when there was an error in the execution of a request and the program name in the error was longer than the public name length.

Rich Client – Authenticated Proxy (QCR # 940665)

An error was returned after running a Rich Client program when an authenticated proxy server (Integrated Authentication) was set at the second position in a PAC file.

Broker – Wait Timeout (QCR # 941253)

An error occurred due to improper calculation of the wait timeout in a specific scenario.

Functions – DVal() (QCR # 967333)

The DVal() function did not work properly on a text year in the Hebrew version.

Oracle – Direct SQL (QCR # 972081)

An error occurred when trying to insert a record into a data source using an Oracle procedure with IN/OUT parameters.

Reports – Timer Events (QCR # 972364)

The engine crashed randomly after processing a large amount of data/reports on a specific customer site when Timer events were defined in a component.

Web Service Lite – Response Value (QCR # 979949)

There was a difference in the Web service response when using the Web S. Lite operation in 1.9c and in 1.9d onwards.

Rich Client – Parallel Program (QCR # 982117)

The original context was closed after reaching the timeout defined in the ContextUnloadTimeout setting when calling a parallel program.

Parallel Execution – Repeated Execution (QCR # 982907)

uniPaaS crashed randomly after repeatedly calling a parallel program from a menu.

Windows – Parent and Orphan (QCR # 987166)

uniPaaS crashed randomly when moving the focus many times between parent and orphan windows and then closing them.

Printing – Card Printer (QCR # 987907)

Printing of images on a Zebra P330i Card Printer was distorted.

A new special setting named SpecialImageResize4Printing was added to render the images differently.

Browser Client – Card Printer (QCR # 989815)

The Record Suffix was not executed in a Browser Client subform.

Rich Client – Placement (QCR # 991746)

The table columns' placement were wrongly calculated when the table was called inside a subform running in a program defined with the Window Type set to 'Fit to MDI'.

V9 Converter – Models (QCR # 993317)

The reference of a model attached to a variable was lost in some scenarios.

Web Service Lite – BLOB Variable (QCR # 993862)

uniPaaS crashed after sending a BLOB argument with a NULL value (xsi:nil="true") in an Invoke Web S, Lite operation.

PDF – Landscape (QCR # 996931)

A report set to print to PDF with Landscape orientation printed the first page correctly and all the other pages in Portrait.

Rich Client – Stat() Function (QCR # 996793)

The Stat() function returned the wrong result after reentering into a subform.

Rich Client – Checker (QCR # 997778)

The Checker crashed when performed on a large Rich Client program.

Rich Client – SDI (QCR # 997832)

uniPaaS crashed after closing the first instance and then the second instance of a Rich Client SDI (non-parallel) task that was called twice from another RC SDI non-parallel program when a Studio sort was defined in the called program.

User Function – Hebrew Version (QCR # 999775)

User function parameters defined with Hebrew characters were displayed as 'aaaa..' in the function list.

Rich Client – Empty Dataview (QCR # 999931)

The record was not updated correctly when the task was defined with 'Allow empty dataview=Yes' and the update was done by the server after a specific sequence of actions.

Magic Software Enterprises Ltd provides the information in this document as is and without any warranties, including merchantability and fitness for a particular purpose. In no event will Magic Software Enterprises Ltd be liable for any loss of profit, business, use, or data or for indirect, special, incidental or consequential damages of any kind whether based in contract, negligence, or other tort. Magic Software Enterprises Ltd may make changes to this document and the product information at any time without notice and without obligation to update the materials contained in this document.

Magic is a trademark of Magic Software Enterprises Ltd.

Copyright © Magic Software Enterprises, September, 2011