



## uniPaaS V2.0a

### Release Notes

**W**e are proud to introduce **uniPaaS V2.0a**. uniPaaS V2.0a is an improved and updated version of the uniPaaS V2.0 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 V2.0a*

We are delighted to provide you with the new major release of Magic Software's uniPaaS application platform.

uniPaaS enables software vendors and enterprises to quickly and cost effectively develop and deploy complex business applications and enjoy all the benefits of .NET applications, Rich Internet Applications (RIA) and Mobile applications whether on-premise, or on-demand.

### *Genuine .NET Runtime Engine*

Since uniPaaS V2.0, the Runtime engine is now a genuine .NET framework based engine (using the Windows Forms .NET library as the GUI front-end), making any application that is run using this engine a genuine .NET application.

In addition to your application being a true .NET application, you can further harness the .NET framework potential by utilizing uniPaaS's seamless .NET integration capabilities.

### *.NET Integration*

You can enhance your application offering by easily embedding and integrating any .NET control or assembly.

You can upgrade the look and feel of your application by directly placing new .NET controls into your application's user interface. You can also enhance the functionality of your application by integrating any .NET assembly.

For more information on how to add and manipulate .NET modules, see the **.NET Tutorial** sample installed with uniPaaS.

### *Mobile Device Support*

Expand your IT offering by supporting mobile devices alongside stationary desktops. With the same high productivity of development and by harnessing 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 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 licenses.

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

### ***Migrating from eDeveloper V10.1 or uniPaaS V1.x to uniPaaS V2.x***

Migrating an eDeveloper V10.1 or a uniPaaS V1.x application to uniPaaS V2.x requires a migration procedure.

The migration process automatically starts when you open (or import) your project. At the end of the process, you will see a log specifying the changes made to your project.

Before migrating a project, you need to migrate its components. This is important since the project may use models from a component and they must be migrated first.

Note that after the migration to V2.x, you should not open your project in an earlier version.

For more information on the migration process and the major changes between the versions, refer to the [Migration Guide](#) available on Magic's DevNet.

For a full list of changes, refer to the *What's Different in uniPaaS V2.0* help topic.

### ***Migrating from eDeveloper V9.4 to uniPaaS V2.x***

Migrating an eDeveloper V9.4 application to uniPaaS V2.x is a two-phase migration process.

First, you need to migrate your eDeveloper V9.4 application to a uniPaaS V1.0 application. uniPaaS provides a collection of wizards to easily migrate your 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. Note that these tools are also provided with uniPaaS V2.x, so you do not need to install uniPaaS V1.x for the migration.

Then, migrate the resulting uniPaaS V1.0 application to uniPaaS V2.0 as described in the section above.

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

### ***From Record Main Based to Event Driven Applications***

Turn your applications into fully event driven applications by utilizing the Record Main Converter utility. The 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.

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.

### *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](mailto:info@magic-optimizer.com).

### *Prerequisites*

uniPaaS is a native .NET application, so in order to run uniPaaS on your machine, you must have .NET framework installed on your machine with one of the following configurations:

- Both .NET Framework V2.0 SP1 and Microsoft Visual C++ 2008 Redistributable. You can install them by running the **NetFx20SP1\_x86.exe** and **VCRedist\_x86.exe** files from the **Scripts\RIA** folder.
- .NET Framework V3.5

### *Network Installation*

The .NET runtime security policy (by default) disables code from running if it exists on a network drive. To run uniPaaS from a network drive you need to adjust your security policy.

This can be done via the Microsoft .NET Configuration tool or by running the following command:

```
c:\Windows\Microsoft.NET\Framework\v2.0.50727\caspol.exe -machine -addgroup All_Code -strong -file <path>uniRTE.exe -noname -noversion FullTrust -name uniPaaS_Assemblies_Access -description "Code group granting trust to uniPaaS assemblies"
```

This command needs to be executed only once in each of the computers.

You can also add a **-silent** key for a silent installation.

### *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>

### ***Online Changes***

In uniPaaS V2.0, the Online GUI is based on the Windows Forms .NET library in the .NET framework instead of Win32.

The .NET framework does not support all the appearances and control behaviors that were supported in previous versions, so there is a change in the uniPaaS Runtime appearance as well.

For a full list of changes, refer to the *What's Different in uniPaaS V2.0* help topic.

Some of the major enhancements are described below.

### ***Sample Projects***

The following sample projects are 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 and some additional samples of uniPaaS functionality.
- Rich Internet Samples – A set of Rich Internet examples for many Rich Client “how to” scenarios that are described in the Mastering uniPaaS book and some additional samples of uniPaaS functionality.
- 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 of the projects are available with their source code, so you can use them in your applications.

### ***uniPaaS Web Page***

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

## New Features, Feature Enhancements and Behavior Changes

### ***MDI Frame Color***

A new color was added to the default color file with the value corresponding to an MDI Frame background.

This color is used as the MDI Form color in newly created projects (and can be changed in the properties of the Main Program form).

### ***End-User Functionality Component***

The End-User Functionality component is now added by default to newly created projects.

A new option was added in the migration wizard, which lets you select whether to also add this component to the migrated project.

Note that if you want to use both this component and the Report Generator, then the Report Generator component should be defined above the End-User Functionality component.

### ***Subform Behavior***

A new property named Refresh When Hidden was added to the Subform control in Online tasks.

This property provides the ability to load and refresh the task running in the Subform control the same way it was done in uniPaaS V1.9.

A value of Yes means that the subform tasks will be called for the first time after the Record Prefix of the host and again after each refresh of the subform. (This is similar to the behavior in uniPaaS V1.9 when the subform was attached to a container control and the container control was hidden.)

A value of No means that the subform tasks will be called only if the subform is visible. (This is similar to the behavior in uniPaaS V1.9 when the subform had a Visible expression.)

For nested subforms defined with a value of No, if the parent subform has a value of Yes at runtime, then the nested subform will also be executed with a value of Yes.

The migration from uniPaaS V1.x will set the value of this property as follows:

- No – When the subform is attached to a container control.
- Yes – When the subform has a Visible expression.
- Exp – When the subform is attached to a container control and also has a Visible expression. The expression is the expression of the Visible property.

In addition, up until this version, when clicking on a Subform control, the Control Prefix and Control Suffix of the first control were always executed. This is now fixed, so only the Control Prefix of the clicked control will be executed.

### ***EDP File Structure***

The EDP file was changed a bit to prevent opening a uniPaaS V2.0 project in previous versions, in order to maintain the integrity of the sources.

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