Magic xpa 4.6 Release Notes

OUTPERFORM THE FUTURE™

Contents

We are delighted to present Magic xpa 4.6 with new features and enhancements.

UI Enhancements

The following UI features and enhancements are added in Magic xpa 4.6.

Form Scaling

You can scale the client area of an active form of the type SDI, MDI, and child window during runtime to different sizes such as 200%, 300%, 400%, and 500% by using a combo box provided on the Status bar. You can set the flag **SpecialFormScalingByDefault** to 'Y' to set the value of **Allow Scaling** property for all the GUI Display forms in the project by default.

SpecialFormScalingByDefault

If this flag is set to 'Y', then the default value of **Allow Scaling** property will be 'True' and the form will be scalable. This means if **Allow Scaling** property is not broken, its value will be 'True'. If the property is broken, the broken value will be used.

If this flag is set to 'N' (or is nonexistent in Magic.ini), then the default value of **Allow Scaling** property will be 'False' and the form will not be scalable. This means if **Allow Scaling** property is not broken, its value will be 'False'. If the property is broken, the broken value will be used.

Valid values: Y, N

Default value: N

Note: The use of Runtime Form Designer is not supported for the forms that allow scaling.

Freezing Table Columns

Now you can set the desired number of columns in a Table control to remain frozen while scrolling the Table horizontally. The frozen columns are marked by a dark grey-colored divider by default. You can choose a color to demarcate the frozen columns from the rest. When you save a task containing tables with frozen columns and reopen it, the frozen columns that you saved last time are retained.

Note: There is no support to freeze table columns in RTL tables.

Zoom Button for Edit Control

You can add a Zoom button [...] next to the Edit control, which on clicking raises Zoom event. You can also customize the Zoom button's look and feel.

Autocomplete for Edit Control

You can autocomplete the text while typing, using a new property named **Autocomplete Mode** available for Edit control.

Flag for 3D-Sunken Radio Button Colors

In order to provide backward compatibility with UniPaaS and retain the current behavior of 3D-Sunken Radio button's background and foreground colors in xpa, a special flag named **SpecialUseColorForRadio** is added.

SpecialUseColorForRadio

If this flag is set to N, the background and foreground colors are disregarded.

Valid value: Y, N

Default value: Y

Flag for Owner-Drawn Checkbox

In order to provide backward compatibility and mark the checkboxes as owner-drawn, a new property named **Owner Drawn** and a special flag named **SpecialCheckBoxOwnerDraw** are added.

SpecialCheckBoxOwnerDraw

When this flag is set to Y, all the checkboxes with their Appearance property set to Box in all the applications will be owner drawn.

Valid values: Y, N

Default value: N

Auto Drop-down for Combo Box Control

You can make the Combo Box in focus drop-down automatically and view its items immediately using a new property **Auto Drop-down.**

Setting Color for Disabled Edit Control

You can set a color of your choice for the disabled Edit control.

Controlling Border Property of 3D-Sunken Edit Controls

While migrating your uniPaaS applications to Magic xpa, you have an option for controlling the Border property setting for 3D-Sunken Edit Controls in the Migration wizard.

Setting Table Control Title's Background Color

While migrating applications from uniPaaS to Magic xpa, you can set the Table control title's background color.

Studio Enhancements

The following enhancements related to studio are added in Magic xpa 4.6.

Copy-Paste Logic Lines across Tasks

You can copy and paste Logic, Data View, and Header lines across tasks located within the same program or outside it.

Long Names for Tasks, Programs, and Forms

You can enter Task Name, Program name, and Form name as long as 100 characters.

General Enhancements

The following general enhancements are added in Magic xpa 4.6.

Unicode Arguments for Mail Functions

Mail functions of Magic xpa support Unicode arguments.

Apache Tomcat Installation Upgraded

Apache Tomcat is upgraded from 7.0.62 to 9.0.20. The previously deployed Web Services should be copied to "**%tomcat_home%\webapps\axis2\WEB-INF\services\**".

Web Client Enhancements

The following enhancements related to Web Client are added in Magic xpa 4.6.

Display of Spinner

You can display spinner on the page(s) of your Web Client application when the server processes lengthy requests. You can also customize the spinner.

Accessing Form Controls from HTML

A new **Controls Accessor** component named **<component name>.mg.controls.g.ts** is introduced to enable you to access Form controls and their properties from the relevant HTML file. The Controls Accessor component implements easy validation for Form controls in html.

Enhanced Web Application Generation Dialog Box

You can see the list of programs you select to generate a Web application along with the folder names in which they are created.

New Concept Paper

The following concept paper is added in the help:

Using Magic Components in Web Client Applications

You can find it under the link Home > Concept Papers > Developing Web Client Applications.

Database Enhancements

The following enhancements related to databases are added in Magic xpa 4.6.

Native PostgreSQL Database Gateway

PostgreSQL database gateway is supported natively, which is compatible with PostgreSQL 11.5. You need to procure its license separately.

Support for Physical Locking in MySQL Gateway

Magic xpa now supports physical locking in MySQL gateway. The physical locking works with MySQL server version 8.0. It can be enabled using the flag **SQL_PHYSICAL_LOCKING=**Y.

Support Oracle 18 and 19

Magic xpa was tested and is compatible to work with Oracle 18 and 19.

Scrolling to the Desired Record

It is now possible to jump directly to the desired record using Table scrollbar. For more details, please refer to Magic xpa help.

WITH Clause

You can use 'WITH' clause in Direct SQL statements.

Temporary SQLite Database Files

Now SQLite database also can be used as a Temporary database in the Environment setting 'Database for Sort/Temporary' under the Preferences tab.

A database information flag named **TEMP_DB** is introduced in order to define a Temporary SQLite database. For more details, please refer to Magic xpa help.

New Keyword: 'DataTypeCompatibility'

A keyword named **DataTypeCompatibility** is introduced in DB Information, which upon setting to 'Y' in the database properties of MSSQL database makes the new MSOLEDB driver handle the data in the same way the legacy SQLOLEDB driver handles it.

GigaSpaces Enhancements

The following enhancements related to GigaSpaces are added in Magic xpa 4.6.

Determining Status Update Rate

SpecialGSWorkerStatusUpdateRate

This special flag determines the number of seconds at which the worker updates its last modification time periodically.

For example, if N = 5, then the worker will update its Last modification time every 5 seconds.

Valid values: >= 1

Default value: 10

This special keyword can assist in distinguishing between non-responsive workers and genuinely busy workers engaged in executing long requests.

Provision of Workers Dynamically

The NumberOfWorkers element in the ProjectsStartup.xml file is now enhanced. According to this enhancement, the Server will create additional Workers if the current Workers count is less than (NumberofWorkers + AdditionalOnDemand), and a request waited longer than 'LoadOnDemandTimeout' seconds before processing it, even with available licenses.

Workers Pane

It is now easy for you to distinguish non-responsive workers from the workers busy executing long duration programs while using GS middleware. A pane is provided to display the workers of the selected server.

RIA Enhancements

The following enhancement related to RIA is added to Magic xpa 4.6.

Client-side Cache Clearing

A function named **ClientClearCache()** is introduced, which on passing a logical argument 'True' clears the local cache of Rich Client tasks immediately.

Mobile Enhancements

The following enhancements related to Mobile are added in Magic xpa 4.6.

Android

64-bit Compliance

Magic xpa Android client is now 64-bit compliant.

Android and iOS

Client-side Cache Clearing

A function named **ClientClearCache()** is introduced, which on passing a logical argument 'True' clears the local cache of Rich Client tasks immediately.

Keeping the Device Awake

A property **Keep Awake** is provided for Rich Client Display forms to facilitate keeping the device awake. When the property is set to 'True' and when the current form is active, the device remains awake.

Browser Client Deprecated

As of Magic xpa 4.6 Browser Client feature is declared as deprecated, it will be removed from the Magic xpa offering in a future release.

Fixed Issues

#	Description
77124	A Window Hit event defined in a subtask with Propagate set to 'No' caused the child window to move to the background while clicking on the parent Form or the title.
149898	In a particular scenario, a RIA crashed while sorting with UserFunctionality Ctrl+T.
152600	CndRange() function was found to be executed when the 'False' value was passed as its argument.
153045	When a Model was applied to an SQLite Table name and clicked on 'Inherit All Properties', DB Column name was found to be removed.
155980	The mls_bld.exe file was not getting converted properly according to the source file.
156805	Magic xpa Documentation updated for RIA License count.
157065	If there was a space at the beginning of the task variables, 'DataViewToText() - Illegal task variables specified' message was displayed.
157296	Magic xpa Studio crashed when Default Database with XML Data Source in Database was used and clicked on Data View of a program.
157309	Two Online programs behaved differently when Cnd was set to 'No' for a batch.
157499	When a Label control was placed on the Table header, the higher Z-order was displayed in the back instead of in the front.

#	Description
157561	In an Online program, the message 'Valid control input range is: ' was displayed when Range was not set.
157659	When control verification had an Internal event 'Subform Refresh', the cursor was found to be parked on the second Edit control of the called program.
157969	In a specific scenario, Magic xpa Runtime engine crashes in the host application.
158486	DBErr() function returned only 255 characters when a custom error message was longer than 255 characters.
158639	Decimal Parsing snippet returned an incorrect value.
158646	On an iOS device with the language set to Japanese, Null character appeared at the end of the fields automatically.
158742	In a particular scenario, Magic xpa Studio crashed when a user-defined function from the Main program was called in a Batch program using Evaluate operation.
158798	The Push Button was found in disabled state erroneously when an event was raised using GetParam() function.
158938	When a non-interactive program was called from an interactive program, WinMinimize() function on executing second time onwards did not work.
159054	Radio Button was found to have incorrect Vertical Alignment on migrating from UniPaaS to Magic xpa.
159055	The column title was found cut after migration.
159285	On a particular version of iOS, when the Japanese language was set, the data was not extracted.
159505	When the incremental update was set to 'Y', the variable Init was recomputed at the last record.
159534	Magic xpa Studio crashed while activating F8 on an RC program if 'Else Block' was disabled in the program.

#	Description
159714	A Push Button on a phantom task having zero records was seen as disabled when its Enabled condition was set to 'True'.
159789	The Internal event 'Mark Previous Char' was found working incorrectly at the end of the first character in an Edit box.
159810	When the encrypted text in Blob was tried to save using function Blb2File(), garbage data was displayed.
159906	If Color property was set to GUI Display in Data View, and if a 3-D style was applied to a control on the form, a new color of the reference color was found to be created for it after migration.
160524	Data was not displayed on navigating among pages using Back and Forward buttons, a Route event was raised in parent program and it was invoked from one of the two sibling programs.
161179	An error "Uncaught Reference Error: global is not defined." was returned while executing Web Client programs due to the newly introduced Angular behavior.
161289	Space was getting selected in the text selection for a Virtual variable of picture size '<space>X10', when reparked on the Edit control using the Tab key.
161466	The function Rights() was found to return an incorrect value for 64-bit engine using Active Directory.
161704	On iOS device, CndRange() function was found to be executed in a particular scenario on Local DB tables although 'False' value was passed as its argument.

Past Release Notes

Magic xpa 4.5a

We are delighted to present Magic xpa 4.5a update with new features and enhancements.

Web Client New Features

The following new features are added in Magic xpa 4.5a.

Support for Calling an Overlay Window Present in a Web Module Marked as Load on Demand

Now you can call an Overlay window present inside a Web module, when the Web module marked as '**Load on Demand**' is loaded.

Note: If you have any projects generated in Magic xpa versions prior to 4.5a, you need to perform certain steps. Please refer the help topic **Home > Reference Guide > Navigation and Workspace > Repository Manipulations > Web Module Properties.**

For new projects generated in 4.5a, those steps are not required as the generation process handles them.

Web Client Enhancements

The following features are enhanced in Magic xpa 4.5a.

Support for Angular 8

Magic xpa 4.5a update supports Angular 8 for Web Client. You are required to use node.js version 10.9 or greater.

To upgrade your Magic xpa Web Client applications from Angular 7 to Angular 8, please refer the help topic **Home > Mastering Magic xpa > Web Client Application Development > How Do I Upgrade Magic xpa Web Client Applications from Angular 7 to Angular 8?.**

Enhanced Table Row Editing

To enhance the functionality of row editing in a Table, in addition to the existing Angular function **isRowInRowEditing()**, a new Angular function **mg.isCurrentInRowEditing()** is introduced. The later enables you to edit a row in a table when the controls are part of the logical record but not inside the table.

You need to use this method only while using the controls outside the Table control and if the current row is in row editing status.

Support for Customizing Error Message

Magic xpa 4.5a update enables you to write a custom error message using **mgError** tag.

Support for Web Application Generation from Inside a Program

You can generate a Web Client application while being inside a program. The program in working is automatically saved before application generation.

64-bit Runtime Features

The following features are provided in Magic xpa 4.5a Runtime.

Support of WMQSeries

WMQSeries is supported for 64-bit xpa Runtime Engine.

XXLNumericSize Special Flag Set to 'Yes'

For Magic xpa 64-bit Runtime Engine, the default value of the special flag **XXLNumericSize** is set to 'Yes' internally. If you set this flag to 'No' in Magic.ini, Magic xpa overrides your setting. For 32-bit Runtime Engine, the behavior remains unchanged.

General

The following concept paper is added in the help and in the documentation:

Connecting Magic xpa with Git Repository.pdf.

Fixed Issues

Defect	Description
157537	In a Web Client task, 'Index out of bounds' error occured when the task had a Link Query, the number of records were less than a page, and the Create Line event was raised.
157839	While invoking an Overlay window, when the second record was attempted to create in a table, 'Duplicate index' error occurred.
158366	In SFDC project FactoryMaster(Access), the server requests in a Subtask triggered the server requests for parent task when the parent task had main source with Link Query/Locate and an Expression in Init of a Variable causing server-side operation.
158732	Studio crashed when 'vcgantt', a third-party control was dropped on a Form.

Magic xpa 4.5

We are delighted to present Magic xpa 4.5 with the addition of some interesting features and enhancements, as well as removal of some features.

Support for 64-bit Server Engine

Magic xpa version 4.5 comes with 64-bit server engine supported in background mode in order to take utmost benefit of a 64-bit operating system. The following points are concerned with 64-bit server engine support:

- In case of Studio product installation, the default selection for 64-bit engine support is according to the Operating System (OS). For 64-bit OS, both 32-bit and 64-bit xpa engines are installed. For 32-bit OS, only 32-bit xpa engine is installed.
- In case of Server product installation you can select either from 32-bit and 64-bit Server product installation.
- While adding 64-bit pointers to buffer you need to take care of adjusting positions of the next members in the buffer so that the application can be used with 32-bit and 64-bit xpa engines.

Unsupported Gateways

All gateways except the below ones are supported:

- AS400/Db400 gateway
- Gigaspaces gateway
- Web Service gateway

Unsupported Runtime Features

- MgComCreator
- DDE
- WMQ

Discontinued Provision of Other Runtime Features

- Java installation that earlier came bundled with Magic products installation is now discontinued. You need to install Java separately.
    - **For Studio product installation:**
      - Always set 32-bit Java in all configuration files (even Web Requester of GS is 32-bit).
      - In Magic.ini, set both JAVA_HOME_32 and JAVA_HOME_64 (if installed).

- **For Server product installation:**
  - 64-bit: Always set 64-bit Java in all configuration files.
  - 32-bit: Always set 32-bit Java in all configuration files.
- **For Desktop Client product installation:**
  - Always set 32-bit Java in all relevant configuration files.
  - In Magic.ini, set only JAVA_HOME_32 (if installed).

Once you install Java, you can use Magic's utility **MagicJavaConfig** to set a new/update the existing JAVA_HOME path in various configuration files of xpa.

- Hardware Based Licensing (HASP) will be supported only in 32-bit products and components installation.

Web Client New Features

The following new features are added in Magic xpa 4.5.

New Events

The following new internal events are introduced:

Subform Close

The **Subform Close** event explicitly closes the running task in the specified Subform control and returns the Subform to 'None' state.

No Program Execution Right

The **No Program Execution Right** event is raised when the user trying to execute the program did not have the program execution right.

New Function mg.SimulateClick()

A new function **mg.SimulateClick()** is introduced for enabling you to raise Magic events from Angular.

New Authentication Functions

Magic xpa 4.5 provides the following new functions to enable you to work in anonymous and authenticated sessions.

1. **IsLoggedIn()**: Checks if a user is logged in or not.
2. **Logout()**: Logs-out a user from the on-going session.

Sophisticated Web Application Generation

Magic xpa update brings you a sophisticated Web generation output dialog with the provision to display the generation progress. The dialog shows the progress of generation in the step-by-step manner in the order of the tasks completion. The generation output dialog box stays on the screen until you close it. You can press the image buttons next to each step of generation to view the filtered log for the desired task.

Support for Sophisticated Functions under Web Menu

Integration with Microsoft® Visual Code

You can now open the generated project in Visual Studio Code by selecting the option **Open in Visual Studio Code** under Web menu.

Support to View Recent Generation Log

You can view the log of Web application generation recorded in the file nodejs.log file by selecting the option **Show Recent Log** under Web menu.

One-click Minimization of Modals

Now you can minimize and restore multiple modal windows in a single click. Magic xpa update introduces a new flag named **SpecialAllowMinimizeAllModals** to provide minimization of modal windows.

SpecialAllowMinimizeAllModals

When you set this flag to 'Y' and when you have multiple modal windows open, then clicking on the minimize button of the modal window in focus minimizes all the windows in the runtime tree.

Valid values: Y, N

Default value: N

Support for Ability to Modify the Code during Runtime

Now you can make changes to the Web Client project files while the runtime engine is executing in the background. All you need to do is set the value of the option '**Run project as**' to 'Web Client' and set the '**Edit Project mode**' on.

Support for Ability to Change the Program during Runtime

Now you can select another startup program and execute it without the need to stop and start the runtime engine while one program is already executing.

Web Modules for Better Folder Management

Instead of the **ngModule** that includes all the Magic components, you can now create independent Web Modules per folder for your Web Client applications, which can be generated and loaded on demand.

Utility to Check Availability of Required npm and node.js Packages

Magic xpa now provides a way to verify the Web environment on your computer. It determines if the **npm** and **node.js** Angular packages are installed and if they match the required versions.

Provision of New Property: ConfirmUpdateForWebClient

In order to override the default value of Confirm Update for Web Client tasks, a new property named **ConfirmUpdateForWebClient** is introduced. It is added to Magic.ini to control the default value for Confirm Update on Web Client tasks.

Provision of New Directives

mgFormat

Magic xpa 4.5 provides you a free hand to be able to implement masking of mobile number formatting, currency, etc., your way. Magic provides you a directive named **mgFormat** that replaces the Pattern attribute and the two earlier directives **AlphaMagicDirective** and **BooleanMagicDirective**.

NonMagicControl

You can assign a new directive **NonMagicControl** to a focusable non-Magic html element to intimate the Magic engine about a new html element in focus. If the previous control-in-focus was a Magic control, then this directive updates the variable attached to that Magic control.

Support for Customization of Alert and Confirmation Messages

Magic xpa 4.5 allows you to use your own component to customize the alert and confirmation messages to the desired ones in your Web application in place of the default JavaScript alert and confirmation messages. All you need to do is to employ the service **ConfirmationComponentsMagicProvider** to customize the UI messages.

Web Client Enhancements

The following enhancements are done in Magic xpa 4.5.

Support of Web Client for Single User Edition

The Single User Edition supports Web Client tasks. You can use the standard studio during development as in case of the other tasks. You need to set **LicenseName=MGWEB** during deployment in order to load the correct default license.

Enhanced Table Display

The tables with empty records will not show default values in columns now when 'Allow Empty DataView' property is set to 'Yes' in Web Client tasks.

Enhanced HTML Generation

With Magic xpa update, you can generate HTML code for the selected individual control, for the selected control with its child controls, and for the entire Web Client form.

You can also avail the generated HTML code for future use. You can press the Copy button and the generated HTML code is copied to Windows clipboard without having you to mark the code before copying.

No Generation or Overriding of CSS Files per Component

Magic xpa does not generate separate CSS files for each component. If you already have a project in Magic xpa 4.1/4.0 and you upgrade to Magic xpa 4.5, you need to update the file **Magic-style.css** with the following styles:

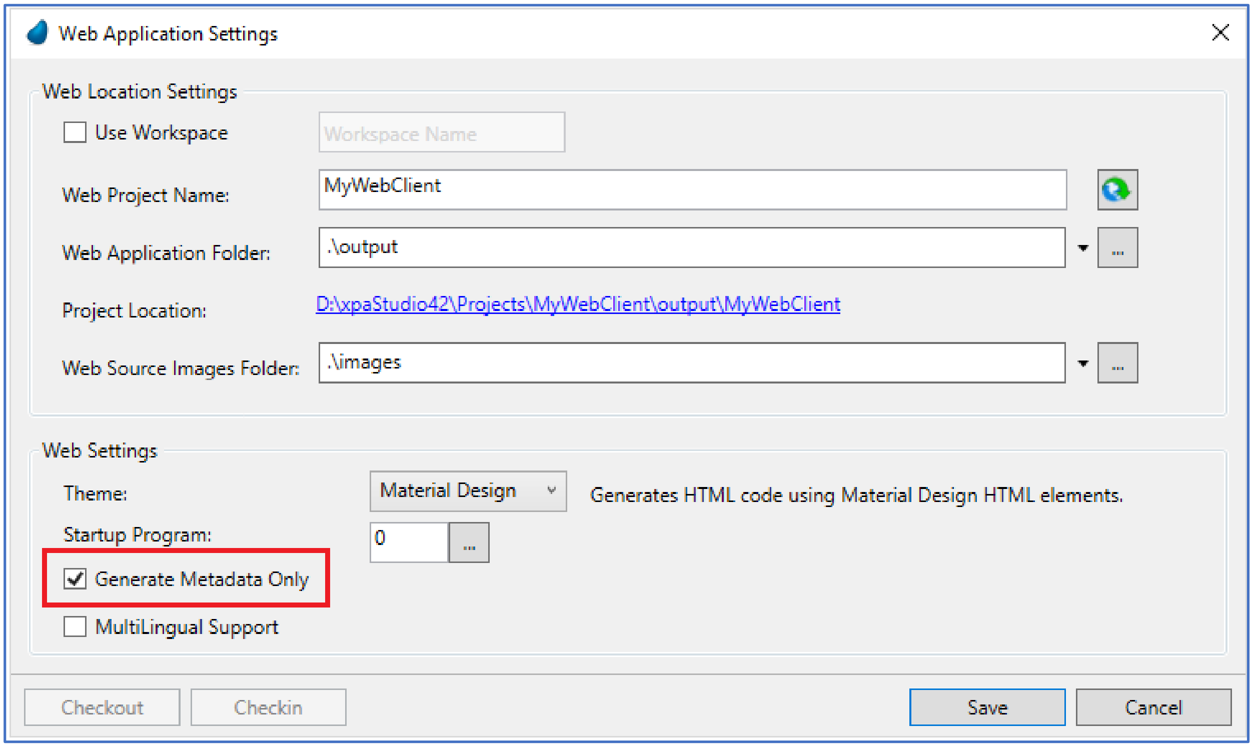
```
.button_image_background_size{
  background-size: 100% 100%;
}
.lable_overflow{
  overflow: hidden;
}
.table_scrollBar{
  height: 600px;
  overflow-y: scroll;
}
.container_border{
  border: lightskyblue solid 1px;
}
.selected {
  background: #F5F5F5;
}
.table_row:hover {
  background-color: #EEEEEE;
}
```

Now onwards, when a program or a form will be generated, the css file will not be created inside the component. The common css file will be referred.

Support to Generate Metadata

Support to Maintain Identical Metadata between Magic and Angular Projects

The Magic xpa update now copies the generated metadata files from the Magic project's **\input** folder to the Angular folder **\magic-metadata**.



Web Application Settings

Web Location Settings

☐ Use Workspace — Workspace Name

Web Project Name: MyWebClient

Web Application Folder: .\output

Project Location: D:\xpaStudio42\Projects\MyWebClient\output\MyWebClient

Web Source Images Folder: .\images

Web Settings

Theme: Material Design — Generates HTML code using Material Design HTML elements.

Startup Program: 0

☑ Generate Metadata Only

☐ MultiLingual Support

Thus, you can maintain the same versions of the metadata files between xpa and Angular projects.

Support to Maintain the Most Recent Metadata

When you generate the same Web application second time onward, Magic xpa 4.5 clears the previously created **\input** folder.

The Overlay Windows

An overlay window in Magic xpa 4.5 behaves identical to a Modal window. Means, all operations after the Call operation are executed only after the Called window is closed.

You can now control the properties of overlay windows from Angular/HTML/CSS. For more information, please see the concept paper Developing Web Client Applications.

Support for Same Workspace for Multiple Projects

Magic xpa update now provides you the ability to have multiple applications or sites under one Angular (CLI) workspace. Additionally, you can create shared set of components, directives, and services for your applications within the same Angular workspace.

New Concept Papers

The following concept papers are added in the help and in the **\Support** folder:

- Developing Web Client Applications
  - Row Editing in Web Client
  - Routing in Web Client
  - Customizing Overlay Windows
  - Navigating Browser History
  - Customizing Alert and Confirmation Messages
  - Concepts of Login and Logout
  - Understanding Web Modules
- Deploying Web Client Applications

Removal of Mouse Events for Web Client

The support for two events ‘Mouse Over’ and ‘Mouse Out’ is removed for Web Client tasks.

Non-Web Client New Features

The following non-web client features are added in 4.5.

Provision to Disconnect from SQLite Database

You can disable all the existing database connections to the SQLite database once the special flag **SpecialPostRequestDBDisconnect** is set to Y.

SpecialPostRequestDBDisconnect

The runtime engine disconnects all the existing connections to the connected SQLite databases when all of the following conditions are true:

- This flag is set to Y.
- The runtime engine was executed via the xpa studio.
- The runtime engine is activated as an enterprise server (connected to a middleware).
- The runtime engine finished executing the request (the engine has just moved from available status) and is now returning control to the studio. This status can be seen when the engine is connected to a middleware.
- The connection is to a SQLite database.

Valid values: Y, N

Default value: N

Non-Web Client Enhancements

The following non-web client features are enhanced in 4.5.

Improve Git support

To overcome the limitations of PushOK Git and to enable you to work with Local and remote Git repositories, xpa integrated with TortoiseGit and work with repositories directly from the xpa studio interface.

With TortoiseGit, xpa makes it possible for you to:

- Add project files to the repository, commit your changes, pull the changes from and push them to repositories, compare, and merge for individual project repository items; without leaving the studio.
- Cherry-pick instead of merging to have exact control over which commits are ported back to the master branch.

Session Hijacking Prevention for Web Client

The flag **SpecialRIAGenerateAndValidateSessionID** is renamed as **SpecialClientGenerateAndValidateSessionID** in Magic xpa 4.5. The latter is now effective for Web Client as well. In the previous versions (starting from Magic xpa 3.2d up until 3.3c), the flag retains its old name: **SpecialRIAGenerateAndValidateSessionID**. The flag **SpecialClientGenerateAndValidateSessionID** is not supported for Broker.

Android: Migration to Android Jetpacks

Magic app now uses AndroidX for management of Android Support Library using Jetpacks.

iOS: Support of iOS 10

Magic xpa Rich Client Applications now support minimum iOS 10.

Non-Web Client Removed Features

The following non-web client features is removed from Magic xpa 4.5.

Support for Windows 10 Mobile

Support for Windows 10 Mobile is discontinued in Magic xpa 4.5.

Support for FlexLM Licensing

Support for FlexLM License Server is discontinued from the installation of Magic 4.5 for all product types.

Magic xpa 4.1

New Features and Enhancements

We are delighted to present the Magic xpa 4.1 update with new features and enhancements.

New Features

The following new features are added in Magic xpa 4.1.

Magic xpa Needs Angular 7

Magic xpa 4.1 update now works with Angular 7.

Support for Row Editing in Table Control

Magic xpa now supports editing a row in Table control. The following new features are added to be able to achieve the row editing functionality.

- A new internal event named 'Enter Row Editing'
- A new Magic function named 'IsRowEditing()'

Support for Updating Magic Data View from Outside

Magic xpa now adds the ability to update the variables defined in Data View from Angular code. Consider it to implement uploading files to the server.

New Function DataViewToJSON()

Magic xpa introduces a new function DataViewToJSON() to return the data from program Data View to JSON format to be able to read it from Angular code.

Support for Date Field Using Form Control

Magic xpa now supports editing Date field using Input Type = Date. This is applicable for Native theme. The Date value accessor is added to support editing Date field, which is inspired by angular-date-value-accessor: https://www.npmjs.com/package/angular-date-value-accessor.

Enhancements

The following enhancements are added in Magic xpa 4.1.

Easy Row Selection

You can select a row by clicking anywhere on the desired row. If you click on the disabled controls placed on the desired row, the control does not select the desired row.

Provision for Better Clarity in Console Messages

In order to bring more clarity with the source of errors in console, Magic xpa now marks the errors to indicate that they are distinctively from Magic.

Record Flush Event - Control Re-fetching from DB

An optional parameter named 'Refetch Record' is now added to the Internal Event 'Record Flush' to control whether the event will re-fetch the record from the database.

Fixed Issue

Defect	Description
154736	In a particular case, not all records are displayed while calling a program using Route.

The Magic Web Application Framework

We are delighted to introduce our most awaited offering of Magic xpa, the Magic Web Application Framework, to provide rapid business application development with modern UI development capabilities using Angular, HTML5, and CSS.

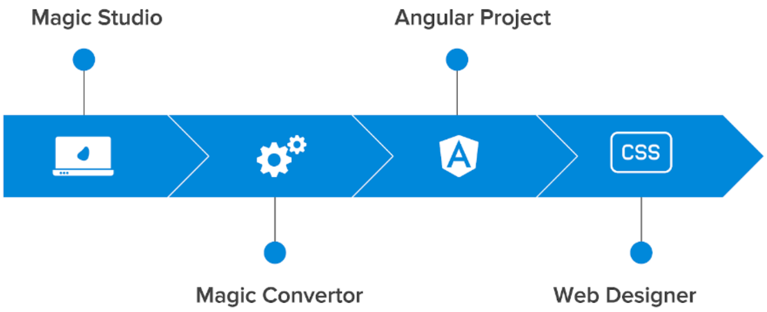
Apps Powered by Angular

You can now easily create modern business apps powered by Angular to provide a rich user experience and meet the increasingly complex enterprise business expectations for digital transformation.

You can now take your Magic xpa business logic for the back and the front end, and easily generate Angular code from it. The new Angular-based application can then work seamlessly with any Web presentation framework. The new Web application framework is beneficial in the following ways:

- You are free to tap into the world of Angular.
- Your apps are client-free since the deployed application is pure JS, HTML5, and CSS3.
- You can outsource the screens design.

The Web Client application generation can be viewed in the following illustration:



3. In the **Magic Studio,** you define your business logic.
4. In the Magic studio, with just a click of a button, you generate your Angular code using the **Magic Converter**, which is customizable by advanced users.
5. In **Angular,** your code is now ready-to-use, but you can also add additional Angular code that is pure client-side logic using some Magic tags.
6. Outside the Magic, you or a designer can update your screens using **HTML and CSS**.

In this way, you can develop highly-responsive Web applications in no time. You can use the structured, low-code Magic xpa development platform for creating the business logic together with an open platform for the presenting your screens including the UI and the UX.

Why Do You Need Magic's Web Application Framework?

Here is why you need Magic's Web Application:

Magic and Angular Fit Well Together

Applications created with Magic's Web application framework will be pure Angular applications, meaning they will be deployed and managed exactly like any Angular application.

State-of-the-art Technology

Magic's Web application framework uses state-of-the-art client-side technologies including JavaScript, Angular, CSS3, HTML5, and TypeScript. You can create full-fledged business heavy-duty, transaction intensive web-based applications including data entry, enabling you to integrate your applications with Web Technology.

Magic xpa's Client-side Architecture

All of the UI-related work, including the UI logic, is done by the browser (the client), which enables you to create fast and dynamic Web applications. In this architecture, the server only sends raw data to the client and the client generates the screen.

Manageable Large Scale Angular Applications

Over the years, the number of lines of code needed to develop functionality-intensive Web applications has been growing at a very rapid pace. As applications are evolving, the market keeps coming up with better ways to handle the ever-increasing lines of code.

Advanced Customization of Your UIs

The beauty of the decoupling of the UI and logic, which Magic's Web application framework makes possible, is the flexibility to split the effort and time invested, between the business logic and the UI/UX.

Less Code; More Versatility

The Magic engine's robust business logic paradigm can save a lot of code in Angular. It helps to set-up most of the behind-the-scenes logic for presenting the UI dynamically.

Magic offers a wide range of control and form properties, which are very simple to use and can save a lot of lines of coding in Angular. For example, validation, recompute, conditions, visible, enabled, tooltips, and more.

Expressions on any of the properties are very easy to use in Magic, which otherwise would take dozens of lines of codes to develop in Angular, to achieve dynamic behavior in apps.

The appearance properties, such as Font, were removed and the basic appearance is done in HTML/CSS/Angular. The remaining properties are for either binding or behavior and these are set in Magic with no need to code these functionalities in Angular. So, by adding a tag to Angular for one of these binding or behavior functionalities, you are basically linking Magic to Angular.

Fully Compliant with Single-Page Applications (SPAs)

Magic's Web application framework is fully compliant with single-page applications (SPA), which provide a smoother and richer user experience.

Cloud Enabler: The Sky's the Limit

Since the Web client is Web-enabled, running in the browser, the Web client is cloud enabled. This means that your apps can be deployed on any cloud.

Cross Browser Support and Portability

Your Web applications will be platform-agnostic. They will run on any browser and on any device with zero footprint. You do not need to install anything.

Ease of Maintenance in the Future

You can change your application logic from Magic and it will be reflected automatically in your Angular-based apps. Thus, there are less efforts to make changes in your application in future.

Full Git Support

Magic xpa is equipped to empower team development with its full Git support (including branching and merging).

About Magic Software Enterprises

Magic Software Enterprises (NASDAQ: MGIC) empowers customers and partners around the globe with smarter technology that provides a multi-channel user experience of enterprise logic and data.

We draw on 30 years of experience, millions of installations worldwide, and strategic alliances with global IT leaders, including IBM, Microsoft, Oracle, Salesforce.com, and SAP, to enable our customers to seamlessly adopt new technologies and maximize business opportunities.

For more information, visit www.magicsoftware.com.

Magic is a registered trademark of Magic Software Enterprises Ltd. All other product and company names mentioned herein are for identification purposes only and are the property of, and might be trademarks of, their respective owners.

Magic Software Enterprises has made every effort to ensure that the information contained in this document is accurate; however, there are no representations or warranties regarding this information, including warranties of merchantability or fitness for a particular purpose. Magic Software Enterprises assumes no responsibility for errors or omissions that may occur in this document. The information in this document is subject to change without prior notice and does not represent a commitment by Magic Software Enterprises or its representatives.