

# Magic xpi 4.0 Release Notes



OUTPERFORM THE FUTURE™

## Enhanced Architecture

Magic xpi 4.0 provides an enhanced and robust architecture. The messaging layer has been replaced with the award winning XAP middleware from GigaSpaces, which uses a distributed, self healing and scalable architecture. These benefits directly enhance Magic xpi 4.0 projects. Existing projects upgraded to this version can be easily scaled to multiple processes and even to multiple machines (scale out) just by changing external configurations.

For more information about Magic xpi 4.0's architecture, please have a look at the [Magic xpi 4.0 Architecture](#) concept paper and the **Quick Start Guide**, which you can find in the **help** folder.

Many of the changes to this version were made to accommodate the new architecture.

## Highlights

- **Self Healing** – Health monitoring of all Magic xpi entities, such as servers, workers and triggers, is now handled in the Space. These entities are automatically reconvered if they fail to report their status.

The Recovery mechanism of flows was enhanced as well to give a robust solution for data recovery.

- **Scalability** – In Magic xpi 4.0's new architecture, the project entity is no longer bound to a single process. The project is now an entity in the Space, which means that many Magic xpi processes can serve the same project.

All operations and data that forced the Project = Server equation are now handled in the Space and, as such, are shared by all Magic xpi servers that run the project. A few examples are Locking, Global, and Business Process (BP) variables, and so on.

- **Management** – Magic xpi 4.0 introduces an **Advanced Monitoring Console**, which displays the project's real time data. The Monitor displays information about the project's servers, the server's workers, the status of each worker, the **Max instance** values of each flow, the license consumption, the trigger's state, and much more.

In addition, it is possible to add more servers to a project and shut down individual servers.

## Feature Enhancements and Behavior Changes

### Monitor Changes

In addition to the new **Advanced Monitoring Console**, some of the columns and parameters in the Monitor have been changed and others have been removed. For additional information, please see the [What's Different in Magic xpi 4.0](#) topic in the *Magic xpi Help*.

## Debugger Changes

The **Attach to Server** dialog box has been renamed to **Attach to Project**.

The **Space Name** and **Lookup Group** parameters have been added to this dialog box, and the **Server Address** and **Server Password** parameters have been removed.

## Licensing

The license mechanism now supports a combination of floating and fixed license thread allocation.

A new property called **ReservedLicenseThreads** was introduced to allow the user to reserve a fixed amount of threads to a project.

## Calling a Flow Using the uniPaaS Call Remote Operation

To call a flow using the uniPaaS Call Remote operation, follow these steps:

1. In Magic xpi 4.0, the following jar files need to be made available in the classpath (either in the machine's environment variable or in the CLASSPATH entry in the **Magic.ini** file):
  - From **<magic xpi>\java\lib**:
    - mgxpi-commons.jar
    - log-commons-logging.jar
    - uniRequester.jar
    - mgxpi-gs.jar
    - log4j-1.2.16.jar
  - From **<magic xpi>GigaSpaces\lib\required**:
    - All the jar files

**Note:** If you are using the Magic xpi 4.0 installation for your uniPaaS development, the jars required for GigaSpaces are already defined in the **Magic.ini** file's CLASSPATH entry. You should then add the jars that are located under the **<magic xpi>\java\lib** defined above.

2. From uniPaaS, create a new Magic xpi type server. In the **Server Address** field, define the following value:

```
SpaceName=MAGIC_SPACE&LookupGroupName=<name>&LookupLocators=<host:[port]>
```

**Note:** The **LookupGroupName** and/or the **LookupLocators** values should be taken from the Magic xpi **Magic.ini** file. If you do not have a value for the **LookupLocators** property, the entry should still exist with the equal sign (=) at the end.

3. Define a new service, and choose the server that you created in the previous step.
4. Set the Magic xpi project name in the **Endpoint** field.
5. You can then use the Call Remote operation with the service that you defined.

## Configuration File Changes

The **Mgrqispi.dll** Web requester has been replaced with a new Web requester called **MgWebRequester.dll**.

## Customizing the Magic xpi Studio

The **Customize** dialog box's **General** tab now contains read-only **Space Name** and **Lookup Group** fields.

## Data Mapper – 32K Limitation Removed

The 32K row limitation was removed from the Data Mapper when creating a flat file.

## Office 365 Support – Dynamics CRM and SharePoint

Office 365 is now supported in the Dynamics CRM and SharePoint connectors.

Similar to the Dynamics CRM resource, the SharePoint resource now has a **Deployment Type** parameter for selecting **On Premise** or **On Demand**.

The SharePoint resource now enables selecting a site from the new Sites list.

## Salesforce Connector Enhancements

In Bulk operations, the field picture and type can now be changed in the Objects fields screen.

## GigaSpaces-Related Memory Settings

### Memory Allocation Errors

From Magic xpi 4.0, any data from triggers and any data sent to parallel or stand alone branches goes through the Magic Space. This means that the default memory allocation settings may not be enough if a very large amount of content is sent, or if external triggers such as HTTP are being used.

The memory allocation settings for the various GigaSpaces entities can be found in the following location: **<magic xpi>\GigaSpaces\bin\magicxpi-gs-agent.bat**.

The GigaSpaces container (GSC) that holds the data in the Space is currently configured to 256MB. If you encounter any memory allocation errors, you should consider allocating at least 512MB to the GSC by changing the following line:

```
set GSC_JAVA_OPTIONS=-Xmx256m -  
Dcom.magicsoftware.ibolt.home="%MAGIC_XPI_HOME%"
```

To:

```
set GSC_JAVA_OPTIONS=-Xmx512m -  
Dcom.magicsoftware.ibolt.home="%MAGIC_XPI_HOME%"
```



## Development Machine Settings

Currently, the default memory and clustering settings are configured for a production machine. In a development machine, the settings can be changed to decrease the application's footprint.

The recommended settings for the `<magic xpi>\GigaSpaces\bin\magicxpi-gs-agent.bat` file are:

### Memory

```
set GSA_JAVA_OPTIONS=-Xmx64m
set GSC_JAVA_OPTIONS=-Xmx256m -
Dcom.magicsoftware.ibolt.home="%MAGIC_XPI_HOME%"
set GSM_JAVA_OPTIONS=-Xmx64m
set LUS_JAVA_OPTIONS=-Xmx64m
```

In the line starting with `call gs-agent.bat`, set `gsa.gsc` to `1` instead of `2`.

For example:

```
call gs-agent.bat gsa.gsc 1 gsa.global.gsm 1 gsa.global.lus 1 gsa.mgdeploy 1
```

### Clustering

Clustering is defined in the `<magic xpi>\config\magicxpi_sla.xml` file.

By default, the Space is defined with two partitions and a backup for each partition. For a development machine, a single partition should be enough and the backup is therefore not required.

Use the following as the content of the `magicxpi_sla.xml` file:

```
<beans xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:os-
sla="http://www.openspaces.org/schema/sla"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
http://www.openspaces.org/schema/sla
http://www.openspaces.org/schema/8.0/sla/openspaces-sla.xsd">
    <os-sla:sla cluster-schema="partitioned-sync2backup" number-of-
instances="1" number-of-backups="0" max-instances-per-vm="1">
        </os-sla:sla>
</beans>
```

## Non-Supported Features

The following features are no longer supported in Magic xpi:

- Two phase processing
- Routing

The following features are currently not supported in Magic xpi 4.0:

- The WS bridge
- Non-persistent locking
- The Web Monitor
- The SharedValGet and SharedValSet functions. For additional information, please see the [Migration from V3.x to V4](#) topic in the *Magic xpi Help*.
- The COM component
- The EJB component
- The Domino component as a trigger
- The W4 connector
- Outputting a form to a requester I/O device using the uniPaaS component

In this version:

- The Post Event service and the Wait for Event service cannot be used between multiple servers.
- The Refresh Conversions service only refreshes the conversion table for the server that it runs on.
- IIS6 requires manual setup.
- Only the Magic xpi Suite is available during product installation. You cannot install the Monitor or the Server separately.

## Magic xpi 3.x Release Notes

For information about Magic xpi 3.x releases, see the **PastReleaseNotes.pdf** file included in the Magic xpi 4.0 installation.

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, 2013

