



First Look: The Java Gateway and InterSystems IRIS

Version 2018.1
2018-06-22

First Look: The Java Gateway and InterSystems IRIS

InterSystems IRIS Data Platform Version 2018.1 2018-06-22

Copyright © 2018 InterSystems Corporation

All rights reserved.



InterSystems, InterSystems Caché, InterSystems Ensemble, InterSystems HealthShare, HealthShare, InterSystems TrakCare, TrakCare, InterSystems DeepSee, and DeepSee are registered trademarks of InterSystems Corporation.



InterSystems IRIS Data Platform, InterSystems IRIS, InterSystems iKnow, Zen, and Caché Server Pages are trademarks of InterSystems Corporation.

All other brand or product names used herein are trademarks or registered trademarks of their respective companies or organizations.

This document contains trade secret and confidential information which is the property of InterSystems Corporation, One Memorial Drive, Cambridge, MA 02142, or its affiliates, and is furnished for the sole purpose of the operation and maintenance of the products of InterSystems Corporation. No part of this publication is to be used for any other purpose, and this publication is not to be reproduced, copied, disclosed, transmitted, stored in a retrieval system or translated into any human or computer language, in any form, by any means, in whole or in part, without the express prior written consent of InterSystems Corporation.

The copying, use and disposition of this document and the software programs described herein is prohibited except to the limited extent set forth in the standard software license agreement(s) of InterSystems Corporation covering such programs and related documentation. InterSystems Corporation makes no representations and warranties concerning such software programs other than those set forth in such standard software license agreement(s). In addition, the liability of InterSystems Corporation for any losses or damages relating to or arising out of the use of such software programs is limited in the manner set forth in such standard software license agreement(s).

THE FOREGOING IS A GENERAL SUMMARY OF THE RESTRICTIONS AND LIMITATIONS IMPOSED BY INTERSYSTEMS CORPORATION ON THE USE OF, AND LIABILITY ARISING FROM, ITS COMPUTER SOFTWARE. FOR COMPLETE INFORMATION REFERENCE SHOULD BE MADE TO THE STANDARD SOFTWARE LICENSE AGREEMENT(S) OF INTERSYSTEMS CORPORATION, COPIES OF WHICH WILL BE MADE AVAILABLE UPON REQUEST.

InterSystems Corporation disclaims responsibility for errors which may appear in this document, and it reserves the right, in its sole discretion and without notice, to make substitutions and modifications in the products and practices described in this document.

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: support@InterSystems.com

Table of Contents

First Look: The Java Gateway and InterSystems IRIS	1
1 Why the Java Gateway is Important	1
2 Exploring the Java Gateway	1
2.1 Before you Begin	1
2.2 Creating a JAR File	1
2.3 Starting a Java Gateway	2
2.4 Generating Proxy Classes	2
2.5 Manipulating Java Objects in ObjectScript	3
3 Learn More About the Java Gateway	4

First Look: The Java Gateway and InterSystems IRIS

This First Look guide explains how to use the Java Gateway to interoperate with Java components in InterSystems IRIS™. In this exploration, you will first create a sample JAR file. You will then create and start a Java Gateway in the InterSystems IRIS Management Portal. You will programmatically import the sample JAR file that you created. Finally, you will create and manipulate some proxy classes and objects in ObjectScript.

To give you a taste of the Java Gateway without bogging you down in details, we've kept this exploration simple. These activities are designed to only use the default settings and features, so that you can acquaint yourself with the fundamentals of the feature without having to deal with details that are off-topic or overly complicated. When you bring the Java Gateway to your production systems, there may be things you will need to do differently. Be sure not to confuse this exploration of the Java Gateway with the real thing! The sources provided at the end of this document will give you a good idea of what is involved in using the Java Gateway in production.

1 Why the Java Gateway is Important

The Java Gateway provides an easy way for InterSystems IRIS to interoperate with Java components. After importing a JAR file using the Java Gateway, you can instantiate an external Java object and manipulate it as if it were a native object within InterSystems IRIS. You will have access to your Java classes and methods from within InterSystems IRIS and ObjectScript!

Internally, the Java Gateway generates one-to-one ObjectScript proxies for each and every Java class. In general, the best approach to using the Java Gateway is to build a small Java wrapper class that exposes just the functionality you want, and then to create a proxy for this wrapper.

2 Exploring the Java Gateway

We have developed a brief demo that shows you how to work with the Java Gateway and InterSystems IRIS.

2.1 Before you Begin

To run the demo, you'll need a single Windows 10 machine with a running, licensed instance of InterSystems IRIS and version 1.8 of the JDK.

For instructions on how to install and license a development instance of InterSystems IRIS, see [Quick Start: InterSystems IRIS Installation](#).

2.2 Creating a JAR File

Create a Person class in Java using an IDE of your choice. Use the following sample code:

```
public class Person {  
    public int age;
```

```
public String name;

//constructor
public Person (int startAge, String Name) {

    age = startAge;
    name = Name;

}

public void setAge(int newAge) {

    age = newAge;

}

public String getName() {

    return name;

}

public int getAge() {

    return age;

}

public static void main(String []args) {

    Person myPerson = new Person (5, "Tom");
    System.out.println(myPerson.getName());
    System.out.println(myPerson.getAge());

}

}
```

Compile the Person class. Then, open a Command Prompt window and navigate to the directory containing your Person.class file. Use the following command to generate a Person.jar file:

```
jar cf Person.jar Person.class
```

Note the location of your Person.jar file.

2.3 Starting a Java Gateway

In order to start a Java Gateway, follow these steps:

1. In the InterSystems IRIS System Management Portal, navigate to the **Object Gateways** page. (**System Administration > Configuration > Connectivity > Object Gateways**)
2. Select **Create New Gateway**.
3. Select the **Java** radio button.
4. Enter JavaGatewayTest for **Gateway Name** and 55555 for **Port**.
5. Click **Save**.
6. In the row for your new Java Gateway, click **Start**.

2.4 Generating Proxy Classes

Create a new ObjectScript class in the **USER** namespace called CreateProxyClasses.cls with package name JavaGatewayTest. Paste in the following code:

```
Class JavaGatewayTest.CreateProxyClasses Extends %Persistent
{
ClassMethod run()
{
    // get a connection to the Java Gateway
    set gwyConn = ##class(%Net.Remote.Gateway).%New()
    set status = gwyConn.%Connect("127.0.0.1", 55555, "USER")
}
```

```

if $$$ISERR(status) {
    write !,"error: "._$system.OBJ.DisplayError(status)
    quit
}

// add the jar to the classpath
set classpath = ##class(%ListOfDataTypes).%New()
do classpath.Insert("YOUR FILEPATH HERE")
set status = gwyConn.%AddToCurrentClassPath(classpath)
if $$$ISERR(status) {
    write !,"error: "._$system.OBJ.DisplayError(status)
    quit
}

// create the proxy ObjectScript classes that correspond to the Java classes in the jar file
set status = gwyConn.%Import("Person",,,,1)
if $$$ISERR(status) {
    write !,"error: "._$system.OBJ.DisplayError(status)
    quit
}

// close the connection to the Java gateway
set status = gwyConn.%Disconnect()
if $$$ISERR(status) {
    write !,"error: "._$system.OBJ.DisplayError(status)
    quit
}
}
}
}

```

Replace `YOUR FILEPATH HERE` with the complete file path of your `Person.jar` file, enclosed in double quotes.

Compile and build the class.

Open the InterSystems IRIS Terminal and navigate to the **USER** namespace. Execute the **run()** method with the following command:

```
do ##class(JavaGatewayTest.CreateProxyClasses).run()
```

2.5 Manipulating Java Objects in ObjectScript

Create a new ObjectScript class in the **USER** namespace called `ManipulateObjects.cls` with package name `JavaGatewayTest`. Paste in the following code:

```

Class JavaGatewayTest.ManipulateObjects Extends %Persistent
{
ClassMethod run()
{
    // get a connection to the Java Gateway
    set gwyConn = ##class(%Net.Remote.Gateway).%New()
    set status = gwyConn.%Connect("127.0.0.1", 55555, "USER")
    if $$$ISERR(status) {
        write !,"error: "._$system.OBJ.DisplayError(status)
        quit
    }

    // manipulate some proxy objects
    set person = ##class(User.Person).%New(gwyConn,5,"Tom")
    write !,"Name: "._person.getName()
    write !,"Age: "._person.getAge()
    write !,"setting age to 100"
    do person.setAge(100)
    write !,"Age: "._person.getAge()

    // close the connection to the Java Gateway
    set status = gwyConn.%Disconnect()
    if $$$ISERR(status) {
        write !,"error:"._$system.OBJ.DisplayError(status)
        quit
    }
}
}
}

```

Compile and build the class.

Open the InterSystems IRIS Terminal and navigate to the **USER** namespace. Execute the **run()** method with the following command:

```
do ##class(JavaGatewayTest.ManipulateObjects).run()
```

You should see the following output:

```
Name: Tom  
Age: 5  
setting age to 100  
Age: 100
```

Now that you have successfully completed the exercise, stop the Java Gateway that you created. Return to the **Object Gateways** page in the Management Portal, locate your `JavaGatewayTest` gateway, and select **Stop**.

3 Learn More About the Java Gateway

To learn more about the Java Gateway and InterSystems IRIS, see:

[Using the Java Gateway in Productions](#)

[Connecting Systems in InterSystems IRIS Using Java Business Hosts](#)