

# How do I access jUDDI from Java

## JAXR Example

The following is an example for working with jUDDI using JAX-R

For additional samples, visit the samples section of the jUDDI code base at <http://svn.apache.org/viewvc/juddi/trunk/juddi-examples/>

```
/*
 * JAXRExamples.java
 * $Id$
 */
package com.ivan.jaxr;

import java.net.PasswordAuthentication;
import java.util.ArrayList;
import java.util.Collection;
import java.util.HashSet;
import java.util.Properties;
import java.util.Set;

import javax.xml.registry.BulkResponse;
import javax.xml.registry.BusinessLifeCycleManager;
import javax.xml.registry.BusinessQueryManager;
import javax.xml.registry.Connection;
import javax.xml.registry.ConnectionFactory;
import javax.xml.registry.FindQualifier;
import javax.xml.registry.JAXRException;
import javax.xml.registry.RegistryService;
import javax.xml.registry.infomodel.Key;
import javax.xml.registry.infomodel.Organization;

/**
 * This class contains JAXR examples doing the following:
 * - Connect to a UDDI registry.
 * - Locate organizations meeting specific criteria.
 * - Create and save a new organization.
 *
 * Configured to use a jUDDI instance running in Tomcat on
 * localhost.
 * Note that a publisher must have been inserted into the database
 * and the appropriate constant below set to reflect the publisher id.
 *
 * Requires the following libraries:
 * jaxr-api.jar, jaxr-impl.jar, jaxb-api.jar, jaxb-impl.jar,
 * jaxb-xjc.jar, jaxb-impl.jar, saaj-api.jar, saaj-impl.jar,
 * activation.jar
 *
 * @author Ivan A Krizsan
 */
public class JAXRExamples
{
    /* Constant(s): */
    private static final String PUBLICATION_MANAGER_URL =
        "http://localhost:8080/juddi/publish";
    private static final String INQUIRY_MANAGER_URL =
        "http://localhost:8080/juddi/inquiry";
    private static final String JUDDI_PUBLISHER_ID = "juddi";

    /* Instance variable(s): */
    private BusinessQueryManager mBusinessQueryMgr;
    private BusinessLifeCycleManager mBusinessLifecycleMgr;
    private Connection mConnection;

    /**
     * Prepares for access to the UDDI registry.
     */
    public void init() throws JAXRException
    {
```

```

/*
 * Retrieve a factory that creates JAXR connections.
 * A connection represents a session with a registry provider
 * and maintains state for the session.
 */
ConnectionFactory theJAXRConnectionFactory =
    ConnectionFactory.newInstance();

/*
 * Set the inquiry and publication manager URLs to be used
 * by the connection factory.
 * Also sets the authentication method, event though it is
 * not strictly necessary.
 */
Properties theUDDIConnectionProperties = new Properties();
theUDDIConnectionProperties.setProperty(
    "javax.xml.registry.queryManagerURL", INQUIRY_MANAGER_URL);
theUDDIConnectionProperties.setProperty(
    "javax.xml.registry.lifeCycleManagerURL", PUBLICATION_MANAGER_URL);
theUDDIConnectionProperties.setProperty(
    "javax.xml.registry.security.authenticationMethod",
    "UDDI_GET_AUTHTOKEN");

theJAXRConnectionFactory.setProperties(theUDDIConnectionProperties);

/* Finally we can retrieve a JAXR connection. */
mConnection = theJAXRConnectionFactory.createConnection();

/*
 * Get the RegistryService, which is used to retrieve,
 * among other things:
 * - BusinessLifeCycleManager which allows for creation and
 *   deletion of different kinds of objects, such as
 *   organizations, services etc.
 * - BusinessQueryManager which allows for searching among
 *   different kinds of objects.
 * - A CapabilityProfile object from which the capability
 *   level and JAXR version can be retrieved.
 */
RegistryService theRegistryService = mConnection.getRegistryService();

mBusinessQueryMgr = theRegistryService.getBusinessQueryManager();
mBusinessLifecycleMgr =
    theRegistryService.getBusinessLifeCycleManager();
}

/**
 * @param args Command line arguments, not used.
 */
public static void main(String[] args)
{
    try
    {
        JAXRExamples theInstance = new JAXRExamples();
        theInstance.init();

        theInstance.addOrganization("Bad Cars Inc");
    } catch (JAXRException theException)
    {
        theException.printStackTrace();
    }
}

/**
 * Queries the UDDI registry for all organizations registered.
 *
 * @throws JAXRException If error occurs querying registry.
 */
public void queryForOrganization() throws JAXRException
{
    /*

```

```

    * Create list of query qualifiers. In this case:
    * - Case sensitive matching of organization names.
    * - Sort found organizations by name in descending order.
    */
ArrayList<String> theQueryQualifiers = new ArrayList<String>();
theQueryQualifiers.add(FindQualifier.CASE_SENSITIVE_MATCH);
theQueryQualifiers.add(FindQualifier.SORT_BY_NAME_DESC);

/*
 * Create list of organization names to search for.
 * The '%%' means that any name will match.
 */
ArrayList<String> theOrganizationNames = new ArrayList<String>();
theOrganizationNames.add("%%");

/* Use the business query manager to query for the organization(s). */
BulkResponse theQueryResponse =
    mBusinessQueryMgr.findOrganizations(theQueryQualifiers,
        theOrganizationNames, null, null, null, null);

Collection theQueryExceptions = theQueryResponse.getExceptions();
Collection<Organization> theOrganizations = theQueryResponse.getCollection();

if (theQueryExceptions == null)
{
    /* Output the result to the console. */
    System.out.println("Listing organizations:");
    for (Organization theOrganization : theOrganizations)
    {
        System.out.println("An organization: "
            + theOrganization.getName().getValue());
    }
} else
{
    System.out.println("An error occurred querying for organizations.");
}
}

/**
 * Adds an organization with the supplied name to the registry.
 *
 * @param inOrgName Name of organization to add.
 * @throws JAXRException If error occurred inserting organization
 * into registry.
 */
public void addOrganization(final String inOrgName) throws JAXRException
{
    /*
     * Set the credentials for the connection.
     * JAXR will help us retrieve an UDDI authorization token and
     * include it in subsequent requests.
     * JUDDI will, in its default configuration, authenticate users
     * which are present in the PUBLISHER table, PUBLISHER_ID column
     * regardless of the password supplied.
     */
    PasswordAuthentication thePswAuthentication =
        new PasswordAuthentication(JUDDI_PUBLISHER_ID, "".toCharArray());
    Set<PasswordAuthentication> theConnectionCredentials =
        new HashSet<PasswordAuthentication>();
    theConnectionCredentials.add(thePswAuthentication);
    mConnection.setCredentials(theConnectionCredentials);

    /* Create an organization object and give it a name. */
    Organization theOrganization =
        mBusinessLifecycleMgr.createOrganization(inOrgName);

    /* Place all organizations to be added in a collection. */
    Collection<Organization> theOrgCollection =
        new ArrayList<Organization>();
    theOrgCollection.add(theOrganization);

```

```

/* Save the organization(s) in the registry. */
BulkResponse theQueryResponse =
    mBusinessLifecycleMgr.saveOrganizations(theOrgCollection);

/*
 * If no exceptions in the response, then the operation
 * succeeded and there will be a key for the saved organization.
 */
if (theQueryResponse.getExceptions() == null)
{
    Collection<Key> theResponseCollection =
        theQueryResponse.getCollection();
    Key theOrgKey = theResponseCollection.iterator().next();
    System.out
        .println("Successfully created and saved an organization.");
    System.out.println("Organization Key = " + theOrgKey.getId());
} else
{
    System.out.println("An error occurred adding the organization: " +
        inOrgName);
}
}
}

```