

# **Kelp Project Documentation: Kelp 5.1, Enhydra 5.1 developer tools and IDE add-ins**

---

# Table of Contents

- 1. Introduction.....4
- 2. The Kelp Project Web Site .....5
- 3. Installing the add-in from [kelp.enhydra.org](http://kelp.enhydra.org) .....4
  - System Requirements .....4
  - Install Options .....5
  - Upgrading from Kelp 2 or Enhydra 3.x .....5
  - Configuring your IDE .....6
    - Setting up JBuilder .....6
    - Setting up NetBeans .....7
  - Defining a project class path .....7
  - ToolBox properties .....7
  - Using the installer application .....8
- 4. CVS modules .....8

---

# Chapter 1. Introduction

The Kelp add-in makes it easy to use the Enhydra XML Compiler (XMLC), the Enhydra Data Object Design Studio (DODS) and the Enhydra Application Server within an integrated development environment. Kelp 5.1 supports Borland JBuilder 4, 5, 6 and 7, NetBeans 3.3.1, 3.3.2, 3.4 and 3.4.1 and Eclipse 2.0. Refer to Using Kelp with an IDE ( [html \[using\\_kelp.html\]](#) , [pdf \[using\\_kelp.pdf\]](#) ) for information on how to use a Kelp add-in to develop web applications with XMLC.

This document supplements documentation for Kelp tools and add-ins. You'll also find an overview of the open source CVS modules at the end of this document.

You will find more details about earlier versions of Kelp at the Enhydra Kelp [<http://kelp.enhydra.org>] project web site along with FAQs (Frequently Asked Questions) [<http://kelp.enhydra.org/project/faq/>] and information on checking out the source from the CVS [<http://kelp.enhydra.org/software/cvs/>] repository.

This documentation assumes a basic understanding of a Java IDE and the servlet [<http://java.sun.com/products/servlet/>] programming model. For information on JBuilder, visit the Borland website [<http://www.borland.com/jbuilder/>].

The add-in provides wizards and tools to help you develop applications for Enhydra5.1. The add-in will add the Kelp application wizards to your IDE's File menu. You only need to use an application wizard when starting a new project. If you have an existing application that uses the Enhydra 3 make system, you can import it into an IDE project file using the Enhydra Import wizard. Once you have a project, you can use the DODS Generator, the XML Compiler and Kelp Deployer to build and deploy your application.

The DODS Generator, the XML Compiler and Kelp Deployer are added to the IDE's Tools menu. These tools give you access to Enhydra features from within your IDE. Using the JBuilder integrated build feature, you can invoke these same processes directly within the project builder without opening the wizard. Each tool also has a set of project and node property pages that you can access from your project navigation tree.

The general model for building web applications with the Kelp add-in:

- Create a new empty project from your IDE.
- Use a Kelp application wizard to generate a web application.
- Use the XML Compiler to generate Java source files for DOM classes. The DOM classes represent the dynamic pages of your application.
- Use the IDE build command to compile all Java source files.
- Deploy the application with the Kelp Deployer.
- Run or debug your application by launching Enhydra from the IDE.
- Develop your application invoking the XML Compiler whenever you add or change XMLC documents. The documents are any HTML and WML files that represent dynamic content.
- Rerun the Kelp Deployer to rebuild your web application archive (.war) to test any changes you've made to your source, template, or content files.
- When your application is complete, copy the web archive (.war) to your production server and use an administration console to add and configure the new application.

The general usage model for building Enhydra applications with the Kelp add-in:

- Create a new empty project from within your IDE.

- Use a Kelp application wizard to generate an Enhydra application.
- Develop your application invoking the XML Compiler whenever you add or change XMLC documents. The documents are any HTML and WML files that represent dynamic content.
- Use the IDE build command to compile all Java source files.
- Use the Kelp Deployer to build and auto-deploy an application archive (.jar) to the locally running instance of Enhydra.
- When your service is complete, copy the Enhydra application archive to your production server.

---

## Chapter 2. The Kelp Project Web Site

The Kelp project web site, <http://kelp.enhydra.org> [<http://kelp.enhydra.org>] is dedicated to making Enhydra technologies easier to use through the use of tools such as application wizards and Java IDE add-ons.

The site serves as the repository for downloads, documentation and mail list archives. It also acts as a gateway to the CVS repository for the open source modules.

You can use the project mail list [<http://kelp.enhydra.org/project/maillingLists/>], [kelp@enhydra.org](mailto:kelp@enhydra.org) to post questions or suggestions regarding any of the Kelp modules. View the mail list archive [<http://kelp.enhydra.org/project/maillingLists/kelp/maillist.html>] to get information on recent releases and tips on how to make the most of Kelp.

You are invited to participate in the Kelp project. We welcome contributions in many forms, including documentation, patches and additional IDE implementations.

---

# Chapter 3. Installing the add-in from kelp.enhydra.org

Before installing the add-in, you should already have installed a supported Java IDE and the Enhydra Application Server.

## System Requirements

- Windows systems
  - Java IDE
    - Borland JBuilder 4.0-7.0 [<http://www.borland.com/jbuilder/>] Download Foundation
    - NetBeans 3.3.1 - 3.4.1.
    - Eclipse 2.0
  - Enhydra Application Server
    - Enhydra 5.1
  - NT 4.0, Windows 2000, Windows XP
  - Memory 256Mb
- Linux Systems
  - Java IDE
    - JBuilder 4.0-7.0 [<http://www.borland.com/jbuilder/>]
    - NetBeans 3.3.1. - 3.4.1.
  - Enhydra Application Server
    - Enhydra 5.1
- Solaris Systems
  - Java IDE
    - JBuilder 4.0-7.0 [<http://www.borland.com/jbuilder/>]

- NetBeans 3.3.1. - 3.4.1
- Enhydra Application Server
- Enhydra 5.1

Kelp 5 requires the Kelp add-in jar, the Kelp toolbox jar and Enhydra XMLC classes to be in the IDE class path. Note that Kelp 5 will not work with Enhydra 3.1.

It is recommended that you go through each step listed in the installation guide so you can run the Enhydra administration console. If you install your Java IDE prior to installing Enhydra, you can skip the step about downloading Java. You can configure the application server to work with the Java 2 files that come with the IDE.

## Install Options

Kelp add-ins are available in the following packages:

- Installer application

Installer application ask user for JDK directory, JAVA IDE directory and ENHYDRA directory and makes all necessary configuration steps. Application is based on jakarta-ant and supported by all three platforms (Windows, Linux, Solaris).

- Zip Archives

To use a zip archive, extract the contents into your <ide\_root> directory. For example, if you have JBuilder installed at: /home/toni/jbuilder5, extract the Kelp archive so you end up with:

```
/home/toni/jbuilder5/kelp5/dist  
/home/toni/jbuilder5/kelp5/doc  
/home/toni/jbuilder5/kelp5/lib  
/home/toni/jbuilder5/kelp5/samples
```

If you install from an archive, you may also need to install the toolbox.jar from the ToolBox zip. The toolbox.jar is also bundled with the Enhydra. Kelp will not run correctly if you are not running with a matching version of the toolbox.jar. If you are not sure if your Enhydra installation contains the correct toolbox.jar (Enhydra 5.1), you should install toolbox.jar from the matching archive.

If you need to install the toolbox.jar, download the zip file and extract the contents into your <Enhydra5.1\_root>/lib/build directory. If your <Enhydra5.1\_root> directory is /Enhydra5.1, extract the toolbox.jar into <Enhydra5.1\_root>/lib/build.

```
/Enhydra5.1/lib/build/toolbox.jar
```

After installing Kelp from archives, you will need to configure your IDE .

## Upgrading from Kelp 2 or Enhydra 3.x

If you have configured your IDE for use with earlier releases of Kelp or the Enhydra application server, you should uninstall and remove all references to those releases. This is particularly important if you plan to use a configure script or the Windows installer to configure your IDE. The following items provide steps that should be taken prior

to installing Kelp 5.1

- **JBuilder**

Before running the configure script, remove addpath statements that reference Kelp or Enhydra application server jar files from the

```
<ide_root>/bin/jbuilder.config.
```

The configure script for JBuilder will not remove addpath statements that reference earlier releases of the application server or Kelp. If classes are referenced by more than one addpath statement, the classes in the first addpath statement will take precedence. Since the script inserts addpath statements at the end of the jbuilder.config file, these addpath statements will have the least precedence.

## Configuring your IDE

You will need to modify some IDE configuration files to enable the Kelp add-in. The configuration process depends on what IDE you are running. Refer to the steps for your IDE.

After configuring your IDE, you can create an Enhydra library to use with your Enhydra projects. The library should include the Enhydra classes required to compile the source files generated by XMLC.

## Setting up JBuilder

For usage information run the configure script without any parameters from the kelp5 directory.

Follow the steps below when you manually install Kelp 5.1 or if the add-in does not appear to be functioning properly after using the Windows installer or the configure script.

- **Setup the IDE class path.**

Add toolbox.jar and KelpAddinJBuilder.jar classes to your IDE class path. JBuilder uses a JBuilder.config file for setting the IDE class path. You can use the addpath setting to add jars to the IDE class path.

With JBuilder closed, edit the JBuilder.config file to add an addpath statement for each of the jar files. Also be sure to remove any addpath statements that refer to older versions of Enhydra, ToolBox or Kelp.

If you have JBuilder installed to /home/toni/jbuilder6 and Enhydra 5.1 installed in an <Enhydra5.1\_root> of /Enhydra5.1, your addpath statements would be:

```
addpath /Enhydra5.1/lib/build/toolbox.jar
addpath /home/toni/jbuilder6/kelp5/lib/KelpAddinJBuilder6.jar
```

- **Create an Enhydra library for defining project class paths**

JBuilder uses libraries to setup the class paths used to build and run your projects. You can create and edit library definitions by selecting Tools | Configure Libraries. Refer to the Defining a project class path for the list of jars required in your Enhydra library.

Note that if the IDE class path and library entries reference different releases of Enhydra, the XML Compiler may generate java source files that fail to compile.

If you want to use the XML Compiler with non-HTML document types on Linux, you will also need to comment out the vmparam -Xverify:none option in the jdk.config file. Note that commenting out this vmparam setting may prevent access to the project Run properties tab.

You can check to see if you have installed the tools correctly by starting JBuilder and looking for the XML Compiler in the Tools menu. If the menu does not appear, check the IDE class path setting from the Info tab of the JBuilder about box.

## Setting up NetBeans

The add-in includes an installer that automates the configuration process when installing into NetBeans 3.3.1 or 3.3.2. It will setup Kelp add-in and add libraries to IDE classpath.

Follow the steps below if the add-in does not appear to be functioning properly after using the installer, or if you want to manually install add-in. Note that when manually configuring NetBeans, you will be modifying your existing configuration files instead of using the Kelp provided installer.

## Configuring your IDE

Add the toolbox.jar, enhydra.jar, xmlc.jar, xerces.jar, tomcat.jar, core.jar and gnu-regexp.jar to your IDE class path. NetBeans include a configuration file that you can use to add jars to the IDE class path.

## Defining a project class path

In JBuilder, the project class path is independent from the IDE class path. Both IDEs also allow you to define sets of jars as libraries. Libraries may be shared between projects.

The following jars are required to successfully compile and run projects that use the servlet, Enhydra application or Enhydra XMLC.

- Enhydra 5.1

```
<eas_root>/lib/enhydra.jar  
<eas_root>/lib/xerces.jar  
<eas_root>/lib/xmlc.jar  
<eas_root>/lib/tomcat.jar  
<eas_root>/lib/core.jar  
<eas_root>/lib/xhtml.jar  
<eas_root>/lib/wireless.jar  
<eas_root>/lib/build/toolbox.jar  
<eas_root>/lib/build/dods.jar  
<eas_root>/lib/build/ejen.jar  
<eas_root>/build/log4j.jar  
<eas_root>/lib/tools/Xalan/xalan.jar
```

## ToolBox properties

The IDE add-in uses wizards within the ToolBox library. The ToolBox persists user settings in a toolbox.properties file. This file is created in an .enhydra directory under your home directory:

```
/home/toni/.enhydra/toolbox.properties
```

if you experience any problems using the application wizards, verify that you have rights to create files and directories within your home directory.

## Using the installer application

You can use the open source installer to install Kelp add-in files and configure your IDE.

The setup program will look up your Java IDE home directory in the Windows registry. If you have installed more than one supported IDE, you may need to click on Browse in the Choose IDE directory step to select the location of the IDE you want to use with Enhydra. The sample projects, documentation and libraries will be installed in a kelp5 directory within the Java IDE's home directory.

After selecting the Java IDE root directory to use for installing, you need to tell the setup program where your Enhydra root directory is located. The setup program will look for a lib directory containing one or more Enhydra jar files within your Enhydra root directory. The setup program will check the Windows registry and common locations to determine a default such as:

```
c:\Enhydra5.1
```

Enter the correct location or use Browse to navigate to your Enhydra root directory. The Enhydra jar files will be added to the IDE class path. The setup program also creates an Enhydra library to use with your projects. The toolbox.jar will also be installed.

After selecting the type of installation you want, setup prompts you for the program group and lists all your selections for confirmation. When you confirm the settings, the installer will copy files into a kelp5 directory within the IDE home directory. The installer also copies the toolbox.jar to <Enhydra5.1\_root>/lib/build.

The installer modifies IDE configuration files. Some of the modifications will not be reversed if you uninstall Kelp. To allow you to restore your original configuration, the setup program creates a backup of each of the modified files and copies the backups to the kelp5 directory.

JBuilder automatically detects and loads add-ins from the jar files that are listed in the IDE class path. It checks the manifest of each jar in the IDE class path to see if it contains a loadable add-in. If you want to know more about what the installer does to update your IDE configuration, refer to the section configuring your IDE .

---

# Chapter 4. CVS modules

The source files for the Kelp ToolBox and add-in modules are available from the CVS page [<http://kelp.enhydra.org/software/cvs/>] of the Kelp project web site. Each module contains an Ant build.xml file. Some of the modules also contain JBuilder projects that you can use to view, debug and build the source.

Kelp includes the following modules.

- AddinCore contains classes shared by IDE implementation modules. The core classes are used to create property pages, wizards and build tasks for working with XMLC and the Enhydra application server from a Java IDE. It also contains two sample projects.
- AddinJBuilder6 contains Borland JBuilder 4-6 implementations of AddinCore interfaces.
- AddinJBuilder7 contains Borland JBuilder 7 implementations of AddinCore interfaces.
- Eclipse plug-in org.enhydra.eclipse\_1.0.0 contains all necessary jars for creating enhydra applications in Eclipse.
- AddinForte contains Sun Forte implementations of AddinCore interfaces for NetBeans 3.3\*.
- AddinForte34 contains Sun Forte implementations of AddinCore interfaces for NetBeans 3.4\*.
- Launcher contains a partial implementation of a GUI launcher for EAS. It is not being actively worked on this subject at this time.
- Toolbox contains wizards, templates and command line tools that can be run without an IDE. These tools are built into the toolbox.jar and related scripts that are included with Enhydra 5.1.
- Winstall contains an InstallShield project for configuring an IDE for use with the Kelp tool set. It currently supports JBuilder, NetBeans and Eclipse add-ins.

In Kelp 5.1 only toolbox, AddinCore, AddinJBuilder6, AddinJBuilder7, AddinForte and AddinForte34 modules can be built with Jakarta Ant 1.5 provided that you have setup the class paths in each modules lib.properties file.