RogueWave
SOFTWARE

Accelerating Great Code

Klocwork Insight v. 10.2.1
Quick Start Guide

RogueWave
SOFTWARE

Accelerating Great Code

≋ Quick Start Guide

If you are ready to get started with your integration, this quick start guide can get you up and running as quickly as possible. For more detailed information, you can refer to the *full installation documentation*.

Getting started with

Your installation process and product setup vary according to:

- » the programming language your team is using (C/C++, C# and Java are supported)
- » the development tools you're using

The out-of-the-box supported language/tool combinations are:

- » C/C++ with Visual Studio, Eclipse, Wind River Workbench, or QNX Momentics
- » C# with Visual Studio
- » Java with Eclipse, IntelliJ IDEA or IBM Rational Application Developer for WebSphere

If you work in another environment or build from the command-line, using your desktop tools is a matter of learning a few commands.

Getting started with the Klocwork tools

This section describes what you need to do to get started with the Klocwork tools for the first time. More in-depth details are available from our product documentation found online and from within our tools.

1. Install the Klocwork Server package that is appropriate for your platform. For more information on specific platforms, see the *online help*.
2. Set up an integration project and run an analysis. This process includes making sure Klocwork recognizes your compiler, capturing your build specification, creating an integration project and running the integration build analysis. If you'd like to test your installation using our demo project demosthenes with Visual Studio, see *Using the demosthenes demo project (Windows)* on page 2 or *Using the demosthenes demo project (Linux or Solaris)* on page 3
3. Install the desktop plug-ins that are relevant for your development team. For Visual Studio or IntelliJ IDEA, download and install the appropriate desktop plug-in. For Eclipse, install the plug-in from the Eclipse update site. The Downloads section will not appear in the portal until the Klocwork Administrator has deployed the plug-ins. For more information, see *Downloading and deploying the desktop analysis plug-ins*
4. Each developer connects to the integration project. Once connected, analysis can be run at the touch of a button while you work. If you are a Visual Studio user, Klocwork Desktop Analysis displays detected issues as developers work.

Using the demosthenes demo project (Windows)

Prerequisites:

- » The Klocwork Server package and the Klocwork Visual Studio desktop plug-in must be installed
- » The Klocwork servers must be running

To test your Klocwork installation, you can use the sample project 'demosthenes' to quickly create and analyze a project. To set up the sample project:

1. In the `<klocwork_install>\samples` directory, you can find the Demosthenes sample project. It is a C/C++ sample project that you can use with Visual Studio.
2. From the `<klocwork_install>\samples\demosthenes` folder, run `kwrun.bat`

    kwrun.bat creates a directory called demosthenes_winbuild in its current working directory. kwrun.bat creates a demosthenes project on the server and analyzes each revision of the source code.
3. In your browser, enter the following URL: http://<klocwork_server_host>:<klocwork_server_port>

4. Select and log in. A user name and password are not required if you have not set up authentication.

5. Choose demosthenes from the project list. From the issues list, you can investigate the issues and get familiar with all of the available features found in .

6. Next, open demosthenes in Visual Studio by opening the '4.sln' file found in the `<user_install>\samples\demosthenes\<visual studio version>` folder.

7. Connect the project to your Klocwork server by right-clicking the solution in the Solution Explorer and selecting Klocwork Properties. Next, select 'demosthenes' from the Klocwork project list and click OK.

8. With continuous analysis, issues are detected when you open or save a file. Make some changes to your code and find out how issues can be dealt with on the fly.

Using the demosthenes demo project (Linux or Solaris)

Prerequisites:

» The Klocwork Server package and the Klocwork Visual Studio desktop plug-in must be installed
» The Klocwork servers must be running

To test your Klocwork installation, you can use the sample project 'demosthenes' to quickly create and analyze a project. To set up the sample project:

1. In the `<klocwork_install>\samples` directory, you can find the Demosthenes sample project. It is a C/C++ sample project that you can use with Visual Studio.

2. From `<klocwork_install>\samples\demosthenes`, run `[<environment_variable_settings>] kwrun.sh [<output directory>]`. If you are using an Ubuntu machine that uses the Dash shell, run: `[<environment_variable_settings>] /bin/bash ./kwrun.sh [<output directory>]`

   where `<environment_variable settings>` can include:

   » `KW_LICENSE_HOST=<host>` - specifies the License Server host (if you are not using a license file)
   » `KW_LICENSE_PORT=<port>` - specifies the License Server port (if you are not using a license file, and if the License Server is running on a non- default port. The default is 27000)
   » `KW_NO_KWINJECT=[yes|y]` - generate build specification without kwinject

   `<output_directory>` is the directory where all analysis data is stored. The default output directory is demosthenes_build, in the current working directory. `kwrun.sh` creates a demosthenes project on the server and analyzes each revision of the source code.

3. In your browser, enter the following URL: http://<klocwork_server_host>:<klocwork_server_port>

4. Select and log in. A user name and password are not required if you have not set up authentication.

5. Choose demosthenes from the project list. From the issues list, you can investigate the issues and get familiar with all of the available features found in .

6. Next, open demosthenes in Visual Studio by opening the '4.sln' file found in the `<user_install>\samples\demosthenes\<visual studio version>` folder.

7. Connect the project to your Klocwork server by right-clicking the solution in the Solution Explorer and selecting Klocwork Properties. Next, select 'demosthenes' from the Klocwork project list and click OK.

8. With continuous analysis, issues are detected when you open or save a file. Make some changes to your code and find out how issues can be dealt with on the fly.

Quick Start: Integration build

To get developers up and running fast, you need to set up a project on the Klocwork Server and connect local projects to it. Developers can then analyze code on their desktops within the context of your software system and prevent the introduction of new defects. At this point, you're also ready to assess the current health of your code with our reporting tools. Here's how to get started:

Prerequisite:

» The Klocwork Server must be installed and running

Analyze your integration build

1. Create a project. This can be done from or from the command line with kwadmin. For example:

   ```
   kwadmin --url http://server2:8080/ create-project my_project
   ```

2. Capture your build settings. In your build environment, issue your regular build command, but prefix it with the kwinject command, plus whatever kwinject options you need. For example:

   ```
   kwinject make
   ```

3. Analyze the project:

   ```
   kwbuildproject --url http://server2:8080/my_project -o my_tables kwinject.out
   ```

4. Load your results into the database:

   ```
   kwadmin --url http://server2:8080/ load my_project my_tables
   ```

View analysis results

1. Log in to through the portal. For example:

   ```
   http://server2:8080/review
   ```

2. The Projects list appears. Click **Reports** and you'll see the Summary report.
3. Click the Top 10 Issues report to see the most common types of issues in your project.
4. Click any segment in the pie chart and you'll get a list of issues.
5. Double-click any issue in the list to investigate it.

Quick Start: Desktop

If you prefer to work with an IDE or text editor that we don't support, you can use kwcheck to set up a local project and find defects using Klocwork Desktop.

> **Prerequisite:**
>
> » The Klocwork Server must be installed and running

1. Set up a local project:

   ```
   kwcheck create --url http://server2:8080/my_project
   ```

2. Capture your build settings:

   ```
   kwshell make
   ```

3. Analyze your code:

   ```
   kwcheck run
   ```

4. View detected issues:

   ```
   kwcheck list -F detailed
   ```

Essential next steps

Once you've completed your first analysis, here are the critical next steps:

» *Configuring checkers for the integration build analysis*
» *Tuning C/C++ analysis*
» *Speeding up the analysis*

Rogue Wave
SOFTWARE

Toll-free: 1.866.556.2967
Direct: 1.613.836.8899
sales@klocwork.com
support@klocwork.com

5500 Flatiron Parkway, Suite 200, Boulder, CO 80301

www.roguewave.com

www.klocwork.com

This document, as well as the software described in it, is furnished under license and may only be used or copied in accordance with the terms of such license. The information contained herein is the property of Rogue Wave Software, Inc. and is confidential between Rogue Wave Software, Inc. and the client and remains the exclusive property of Rogue Wave Software, Inc. No part of this documentation may be copied, translated, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission of Rogue Wave Software, Inc. If you find any problems in the documentation, please report them to us in writing. Rogue Wave Software, Inc. does not warrant that this document is error-free.

Klocwork is a registered trademark of Rogue Wave Software, Inc.

All other trademarks are the property of their respective owners. All help content for Klocwork's MISRA checkers is copyright by MIRA Ltd, on behalf of the MISRA Consortium.

Copyright notices for third-party software are contained in the file 3rdparty_copyright_notices.txt, located in the Klocwork installation directory.