KOVAIR   LDRA

LDRA Integration Adapter/Connector Datasheet

💡 A SNEAK PEEK

Kovair LDRA Integration Adapter connects LDRA Testbed with other best-of-breed tools and Kovair ALM and provides a centralized view of the code development to build process including static code analysis report in LDRA.

⊕ BENEFITS

❖ Allows LDRA users to achieve Continuous Integration - CI from a single interface
  ❖ Enforces code quality compliance by developers during code check-in in an SCM tool
  ❖ Enables users to perform Regression testing using LDRA TBrun and view test results for Regression and Structural coverage report from the IDE itself
❖ Establishes complete traceability between code files, change sets, results, tests, requirements and defects
❖ Perform complete code coverage analysis as well as execute Build for selected code files after checking them into the SCM tool.
❖ Allows developers to automate defect tracking and removal process for each LDRA failed code
❖ Enables project managers to evaluate the ratio between lines of codes written and defects generated, thus measuring developers' performance
❖ Allows developers to view and monitor LDRA results like Quality Review Report and TBrun Regression Report from IDE.
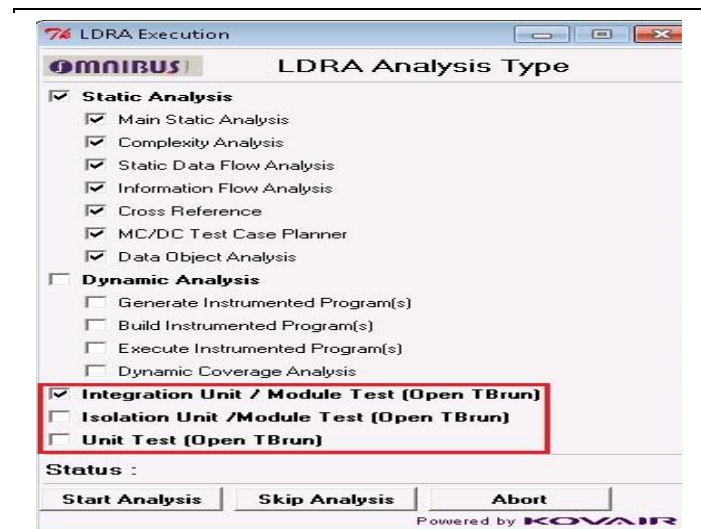
The integration of LDRA tools with Kovair Omnibus platform fuels the development of Safety Critical Software and maximizes the value of LDRA tool investment. Together with Kovair Application and Omnibus integration technology, LDRA provides a Continuous Integration (CI) platform to its users that automates and manages the build process for large-scale, system-level software projects.

The Kovair LDRA integration helps developers to perform static code analysis before code check-in using LDRA Testbed, measure code coverage, trigger Build, and execute Regression testing and view Test Reports using LDRA TBrun, right from their own IDEs.

Kovair Advantages for LDRA Testbed Integration

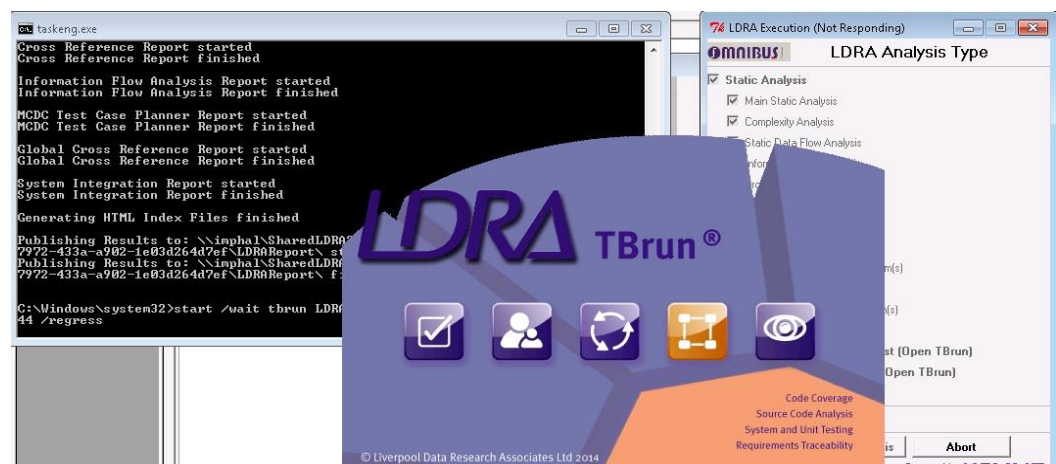**Impose code quality standards into development phase**

Using Kovair integration with LDRA, a developer can perform **Static and Dynamic Code analysis** from the IDE itself. Kovair enables organizations to enforce code analysis by developers during code check-in in an SCM tool like ClearCase.

Fig: Omnibus Dialog Box prompting users to select Analysis type during code check-in

Include Testing during development phase

Kovair's integration interface allows users to perform the Integration Unit, Isolation Unit and Unit Testing using TBrun from their IDEs during code check-in. The integration directly opens TBrun interface where user can create Sequence and can execute Black box / White box testing.

Fig: Triggering Test Execution using LDRA TBrun from IDE

❖ Synchronization support for all major LDRA analysis types such as Static Analysis, Instrumentation Analysis, Dynamic Analysis

❖ Invoke TBrun interface within the IDE for executing Unit test (Black /White Box mode), Integration/Module Test and Isolation Unit test.

❖ Direct access to LDRA tests from within ClearCase by Kovair ClearCase plug-in for LDRA

❖ Visibility of LDRA file, execution, execution results from 3rd party tools like DOORS

❖ Support for Omnibus dialog box where engineers can link the defects / requirements based on which they have developed and checked in the code

◇ SYSTEM REQUIREMENTS

❖ Operating Systems:
Windows Server 2008 R2

❖ Databases:
SQL Server 2008
SQL Server 2008 R2

❖ IIS: **IIS 7.0 or above**

❖ .Net: **.Net 4.5.1 or Above**

Measure Code Coverage and Execute Build
A developer can perform **Code coverage analysis** and execute Build for the selected code files directly from the IDE.

Access LDRA reports directly from the IDE
The extended integration interface of Kovair Omnibus allows user to view LDRA Testbed Quality Review report and TBrun Regression report within the IDE. Users can also view the entire Sequence from TBrun Regression report.

Set up a CI scenario
Kovair Omnibus already has connectors for more than 60 tools which includes build tools like Hudson or Jenkins. With Kovair-LDRA integration organizations can automate the entire development phase like coding, code analysis, build, unit testing, and thus set up a Continuous Integrations-CI environment.

Trace back to source requirements and defects
Developers can link LDRA compliant codes to the original requirements or defects for which they have written codes. This helps project manager understand what codes have successfully passed the LDRA test for which requirements coming from DOORS or which defects raised in JIRA, and how many codes are yet to go through the LDRA compliance test in order to complete the build process successfully.

Manage everything at one place
Developers do not need to log in to multiple tools to monitor the progress status of their development objects or LDRA results like Quality Review Report and TBrun Regression Report. Everything can be viewed from within the developers' IDEs itself. It will be which is done automatically based on a rule (File based configuration or metadata).

Integration with TBmanager
Kovair Omnibus is currently working on the next phase of integration with TBmanager to enable team-wide workflow management. The proposed integration will incorporate Dynamic Call graphs and Flow graphs as part of the reports. Implementation of Requirements based testing using the TBmanager module is also in progress.

How Kovair supports LDRA integration

Kovair facilitates this integration solution with the help of Kovair LDRA Integration Adapter/ Connector and Kovair ClearCase Plug-in for LDRA. The adapter provides bi-directional synchronization between LDRA Testbed and Kovair Omnibus, thus resulting in data flow to and fro from other development tools via Omnibus. The ClearCase Plug-in for LDRA installed locally in developer's system pulls in code files from ClearCase during code check-in and then pushes them to LDRA tool via Kovair LDRA integration adapter.

Once the LDRA tests are done, users can view the execution results from the plug-in itself. As the associated data flows to Kovair Omnibus centralized repository, it provides a vivid picture of the execution. The overall status can also be pushed to other desired tools like DOORS.

One can manage the entire development process starting from importing requirements from a requirement management tool to developing code in an IDE, analyzing coding standards in LDRA, code check-in into an SCM tool to managing builds and deployment in an integrated tool setup.
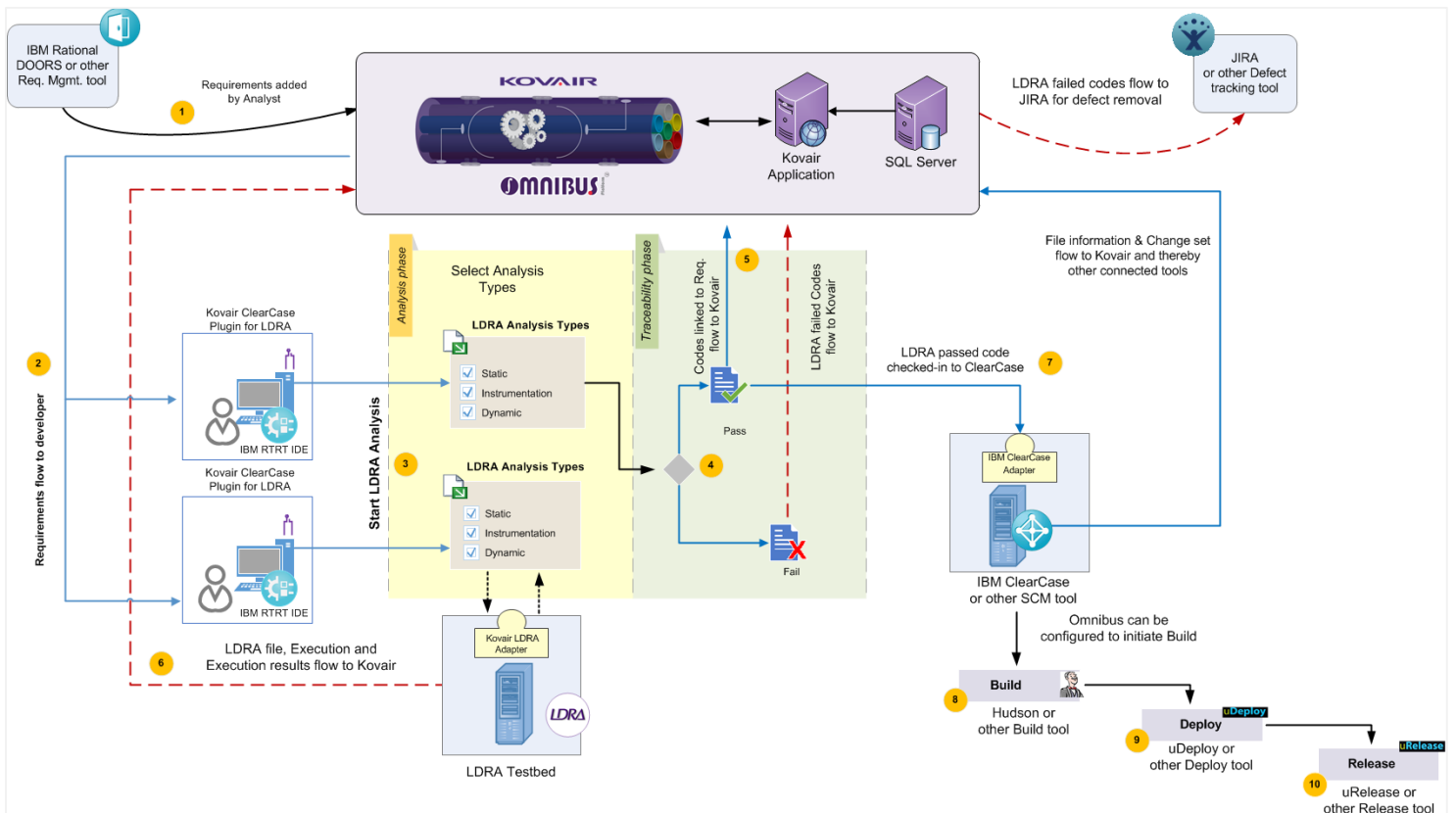
A sample integration scenario involving LDRA and other development tools

Kovair's powerful report and process engine builds complete systems with quality management and automated testing capability. By linking the LDRA tool suite to Omnibus, the platform can invoke the LDRA tool suite to perform analysis, track test and verification fulfillment, and assess overall improvements or regressions in code quality in real-time manner.

One can also use Kovair Application as a central data repository to view all tool generated data including LDRA at one place and establish traceability relationships between any cross-tool artifacts as per the need.

Use Case Flow

1. Business analyst adds a new Requirement in IBM DOORS.

2. The Requirements flow to developer for which he or she writes codes in IBM RTRT IDE.

3. Before codes are checked-in in IBM ClearCase, developer is prompted to analyze code quality in LDRA tool.

4. If the codes pass LDRA compliance test and the results turn out as 'Pass', developer can link the tested code files to the original Requirements in DOORS or defects in JIRA so that project manager can view Requirements and Defects traceability from within DOORS itself.

5. Once linked to the Requirements or Defects, the codes get successfully checked-in in IBM ClearCase.

6. On successful check-in, one can also configure Omnibus to initiate Build process in Jenkins, followed by automated testing and Release in other tools.

7. If LDRA test turns out as 'Fail', the code check-in process automatically gets aborted and defects are raised automatically.

8. The defects flow to any other defect management tool like JIRA through Omnibus for required fixes.

9. Developer check-ins the fixed codes and then repeats the LDRA test for analysis.

In the above Use Case, Kovair Omnibus Integration Platform along with its LDRA adapter/connector facilitates LDRA users in various ways as mentioned below.

Value Propositions of Kovair-LDRA Integration Use Case

1. Manage everything at one place
Developers do not need to login to multiple tools to monitor the progress status of their development objects or LDRA results. Everything can be viewed from within the IDE itself. It will be done automatically based on a rule (File based configuration or metadata).

2. Impose code quality standards into the system
Once check-in is fired in IBM ClearCase, Kovair Omnibus bifurcates the check-in and sends the file to LDRA (in local machine) for analysis. Once the codes pass the analysis steps, they are checked in with a comment that they have passed the following phases of analysis. If the analysis fails in LDRA, the code check-in is aborted.

3. Leave no impact on product stability
Once code check-in is successful, the Jenkins's Build will be automatically fired for the project code that has been checked in to verify the build stability. This insures that the code has no impact on the overall product stability.

4. Manage Build and defects centrally and remotely
When integrated with Kovair ALM platform, off-site managers can easily track progress and quality of Build from within Kovair without logging into Jenkins or other tools. Similarly, defects raised during Build failure automatically flow to any defect management tool like JIRA connected through Omnibus.

5. Manage LDRA Results directly from the IDE
Developer seeking to check-in code in ClearCase does not necessarily need to login to LDRA tool separately to start the code analysis. Once Kovair ClearCase Plug-in for LDRA is installed on each of the Developer's system, it automatically pulls the code files from developer's machine and sends them to LDRA for code compliance test. Developer also gets a quick access to LDRA results without entering into the tool. However, having the LDRA tool installed in each developer's system is a prerequisite.

6. Take actionable decisions easily
When a developer intends to check-in code in IBM CC, Kovair ClearCase plug-in prompts him or her with a dialog box showing multiple options to select phases of analysis like Static analysis, Instrumentation analysis and Dymanic analysis.

7. Trace back to source requirements and defects
Developers can link LDRA compliant codes to the original requirements or defects for which they have written codes. This helps project manager understand what codes have successfully passed the LDRA test for which requirements coming from DOORS or which defects raised in JIRA, and how many codes are yet to go through the LDRA compliance test in order to complete the build process successfully.

8. Gain complete visibility
The management can see the entire chain of development artifacts from Requirements -> Code File -> Code analysis result (HTML) -> Defect -> Build -> Build Result.

The Kovair Omnibus Adapter for LDRA integration is already available for enterprise use. The live demonstration of the integration scenario can be scheduled in advance and run through web conferences by contacting the Kovair Omnibus Team.