Database Change Migrations with Liquibase

Leo Przybylski

Abstract

Database changes via DDL/DML can be difficult to manage. Much of the time, DDL changes will be unreversable. For example, a table rename may require a table drop and recreate. This is usually favorable because then developers can ignore the proprietary language nature of SQL across RDBMS. Liquibase provides a common language and method supporting almost every database there is. Liquibase can also provide a common methodology for applying updates from the Kuali Foundation across different software systems. This can be very useful in apply upgrades from the Kuali Foundation or managing changes for your local institution.

This session covers intuitive and simple database change management process and how to integrate it with existing data migration, change management, and foundation KC, KFS and Rice updates.

1 Objectives

- Implementors will learn best practices for structuring their projects.
- Implementors will learn how to accept database changes from the foundation and structure theirs around.
- Developers will learn how to test changes and the effects of change management within their development process.
- Implementors will learn how to rollback software versions including database changes.

- Implementors will learn how to play/fast-forward changes acrossed several revisions to update to a later version of database and source code.
- Business Intelligence Analysts will learn how to integrate database change management and ETL processes so that source code changes have minimal impact upon their conversion processes.
- Introduce a common methodology for apply upgrades across projects (KC, KFS, and Rice) using Maven and Liquibase.

2 Presenters

Leo Przybylski

3 Audience

Technical track for:

- DBAs
- developers
- DevOps

3.1 Level

Intermediate

4 Focus Area

Technical

5 Track

Kuali Financial System