

Oxydition-Operations

Übersicht über das Java-Projekt für die Bereitstellung zusätzlicher Funktionalitäten in Oxygen für Oxydition

Inhalt

- [Einführung](#)
- [Projektaufbau](#)
- [Framework-Integration](#)

Einführung

Das Projekt *oxydition-operation* umfasst alle Java-Klassen für Oxydition, der Anpassung für den [Oxygen XML Editor](#) für die digitalen, historisch-kritischen Editionen der [Historischen Kommission bei der Bayerischen Akademie der Wissenschaften](#) (HiKo). Die Implementation basiert auf der Java-SDK-Schnittstelle von Oxygen. Integriert in ein Framework innerhalb von Oxygen können hiermit eine Vielzahl von Aktionen für den *Author-Modus* eingerichtet werden. Mit diesen können die Benutzer*innen die XML-Syntax in einem passenden -Dokument automatisch erstellen und bearbeiten.

Grundlage für das Projekt waren 2014 die Vorarbeiten im Projekt [ediarum](#) an der Berlin-Brandenburgischen Akademie der Wissenschaften. Die hier präsentierten Klassen sind auf die Anforderungen und Kodierungsmuster der HiKo-Projekte ausgerichtet. Um sie für ein anderes Projekt wiederzuverwenden, sind voraussichtlich weitere Anpassungen nötig.

Projektaufbau

Kernstück des Projekts ist eine Reihe von Operationsklassen, die innerhalb eines Oxygen-Frameworks konfiguriert und vom *Oxygen-Author-Modus* aus ausgerufen werden können. Alle anderen Klassen unterstützen diese Operationen. Die Klassen lassen sich dabei in folgende Kategorien einteilen, die sich auch in der Package-Struktur widerspiegeln:

- **Operationen:** Die Steuerungsklassen für die Operationsdurchführung. Alle Operationen erben von der Klasse *OxyditionOperation*. Diese implementiert wiederum das Interface *OxygenAuthorOperation*, das von der Oxygen-Java-Schnittstelle zur Verfügung gestellt wird. Abstrakte Operationsklassen sind in das Unterpackage „Abstracts“ ausgelagert.
- **Dialogs:** Dialogfenster, die von den Operationen zusammengestellt und angezeigt werden, Nutzereingaben entgegennehmen und diese an die Operationen zurückgeben.
- **Elements:** Spezialisierte Funktionselemente für Dialogfenster.
- **Beans:** Datenhaltungsobjekte, die Informationen zwischen Klassen transferieren. Am häufigsten eingesetzt, um Nutzereingaben von Dialogfenstern zu den aufrufenden Klassen zu übertragen.
- **Utilities:** Anwendungs- und Hilfsklassen, in die Funktionalitäten für bestimmte Operationen ausgelagert sind oder die häufig benötigte statische Methoden bereitstellen.
- **Data:** Datensammlungsklassen, die statische Daten enthalten und bereitstellen.
- **Adaptations:** Anpassung einzelner Klassen an die speziellen Anforderungen eines einzelnen Editionsprojekts (nur bei Registeroperationen).

Die Klassen sind außerdem gemäß des Typs der edierten Quelle, für den sie relevant sind, in Packages eingeteilt. Die obere Einteilungsebene besteht dabei aus den Packages:

- `common`: (Theoretisch) relevant für alle Quellentypen
- `brfedition`: Relevant für Briefquellen
- `protedition`: Relevant für Protokollquellen
- `register`: Operationen für die Registerdatenbank (die in allen Projekten zum Einsatz kommt)

Innerhalb dieser Packages können Klassen wiederum als *common* oder bei einem bestimmten Editionsprojekt eingeordnet sein. Die Kürzel stehen für

- `minrprot`: Die Protokolle des Bayerischen Ministerrats 1945-1962
- `weimprot`: Die Protokolle des Bayerischen Ministerrats 1919-1945
- `delbrf`: Hans Delbrück – Ausgewählte Korrespondenz

Zur Funktion der einzelnen Klassen siehe die Klassenkommentare im Source Code.

Der Aufbau des Projekts ist in einer Reihe von [Diagrammen](#) visualisiert.

Framework-Integration

Um die Klassen des Projekts vom Source Code aus in ein Framework innerhalb von Oxygen zu integrieren, müssen sie zuerst in eine `.jar`-Datei verpackt werden. Diese sollte dann in den Order des Zielframeworks im „frameworks“-Ordner der Oxygen-Installation kopiert werden. (In den Frameworks der Editionsprojekte der HiKo heißt diese Datei standartmäßig „`sdf.jar`“ und liegt im Unterordner „operations“ innerhalb des Ordners des jeweiligen Frameworks.)

Anschließend muss die `.jar`-Datei in der Framework-Konfiguration von Oxygen zum Klassenpfad der Frameworks hinzugefügt werden. Ab dann können die Klassen des Projekts bei der Konfiguration von Author-Aktionen ausgewählt werden.