

## **ediarum Setup-Anleitung**

# Inhaltsverzeichnis

<b>Kapitel 1. Basis-Setup.....</b>	<b>3</b>
Über diese Anleitung.....	3
Überblick über das Basis-Setup.....	5
Installationstipps eXist-db und Oxygen.....	5
Datenbank einrichten.....	6
ediarum.DB App installieren.....	6
Projekt in der Datenbank anlegen.....	8
ediarum-Register aktivieren.....	9
Nutzer-Account anlegen.....	10
Arbeitsumgebung in Oxygen XML Author einrichten.....	11
Oxygen mit der Datenbank verbinden.....	11
ediarum-Frameworks einrichten.....	13
Projekt im Oxygen XML Author anlegen.....	14
<b>Kapitel 2. Editionsspezifische Konfiguration.....</b>	<b>17</b>
Über Oxygen-Frameworks und ediarum-Spezifika.....	17
Ein eigenes Register einrichten.....	18

# Kapitel 1. Basis-Setup

---

Das Basis-Setup macht mit ediarum eine Arbeitsumgebung für wissenschaftliche Editionen verfügbar. Nach erfolgreichem Setup können Editor/innen in dieser visuellen, Word-ähnlichen Umgebung TEI-konforme XML-Daten mit Registern, Kommentaren u.ä. erstellen.

Ediarum ist eine Arbeitsumgebung für wissenschaftliche digitale Editionen, die auf dem DTA-Basisformat, einem Subset von TEI-XML, basieren.

Das Basis-Setup von ediarum können Sie durchführen, wenn Sie grundlegende IT-Kenntnisse in den Digital Humanities haben. Wenn das Setup einmal abgeschlossen ist, können Sie es als Add-On an die Editor/innen Ihrer Editions-Arbeitsgruppe weitergeben.

## Über diese Anleitung

---

Das Basis-Setup macht ediarum zunächst in der Standardvariante verfügbar, ohne besondere Konfigurationen für Ihr Editionsprojekt und ohne dass Sie Programmierkenntnisse benötigen.

### Über ediarum

Nutzung:

Ediarum ist eine Arbeitsumgebung für wissenschaftliche digitale Editionen. Editor\*innen können in dieser Arbeitsumgebung TEI-XML-basierte Transkriptionen von Textquellen erstellen und bearbeiten, ohne über vertiefte XML-Kenntnisse zu verfügen. Dabei können die Editor\*innen Textphänomene auszeichnen, Sacherläuterungen einfügen, Links zu Registern setzen, Registerinträge pflegen und mit Normdaten ausstatten. Das Basis-Setup stellt mit ediarum.BASE.edit ein auf historische (früh-)neuzeitliche Quellen ausgerichtetes Datenmodell zur Verfügung, das auf dem Basisformat des Deutschen Textarchivs basiert (DTA-Basisformat).

Technische Komponenten:

Auf technischer Ebene stellt ediarum folgende Komponenten zur Verfügung:

- ediarum.BASE.edit: Framework für Oxygen XML Author für Text-Transkriptionen  
# Enthält ediarum.jar, eine
- ediarum.REGISTER.edit: Framework für Oxygen XML Author für Register
- ediarum.DB: eXistdb App für die Datenverwaltung passend zu ediarum.BASE.edit und ediarum.REGISTER.edit

### Setup-Varianten

Diese Anleitung beschreibt das Basis-Setup von ediarum. Sie beschreibt nicht die editionsspezifische Konfiguration und Erweiterung, liefert aber einige Grundinformationen dazu.

Basis-Setup: Sie bringen ediarum zum Laufen, sodass die mitgelieferten Eingabefunktionalitäten für das mitgelieferte XML-Schema zur Verfügung stehen. Das Basis-Setup umfasst auch die Einrichtung eines Personenregisters und eines Ortsregisters sowie der entsprechenden Eingabefunktionalitäten.

Editionsspezifische Konfiguration und Erweiterung: Wenn ediarum läuft, können Sie der Arbeitsumgebung eigene Funktionalitäten hinzufügen, z.B. für editionsspezifische TEI-Elemente. Sie können mitgelieferte Eingabefunktionalitäten für weitere Register (z.B. Werkregister) aktivieren oder eigene Register hinzufügen.

### An wen sich diese Anleitung richtet

Das Basis-Setup von ediarum können Sie durchführen, wenn Sie grundlegende IT-Kenntnisse in den Digital Humanities haben. Wenn das Setup einmal abgeschlossen ist, können Sie es als Add-On an die Editor/innen Ihrer Editions-Arbeitsgruppe weitergeben.

Vorwissen, um das Setup durchzuführen:

- Grundverständnis von XML, XML-Schemata
- Kenntnis des Dashboards der eXist-db, d.h. der graphischen Oberfläche für die Arbeit mit Ihrer Datenbank
- Grundverständnis von github

Vorwissen, um mit ediarum editorisch zu arbeiten:

- Grundverständnis von XML und TEI
- Vertrautheit mit Oxygen XML Author

## Voraussetzungen

- Oxygen XML Author ist installiert, mindestens in der Version 20.1 (alternativ: Oxygen XML Editor).
- eXist-db ist installiert, in der Version 3.5 oder 4.4.

## Feedback

Wir sind jederzeit froh über Feedback zu dieser Anleitung an [andrea.notroff@bbaw.de](mailto:andrea.notroff@bbaw.de).

Wir planen, in naher Zukunft die XML-Dateien zu dieser Anleitung auf github zur Verfügung zu stellen, sodass Sie direkt Ergänzungen beitragen können.

# ediarum: Nutzungs- und Technologie-Beschreibung, Review  
 # ediarum: Anleitung für Editor/innen (Editionsrichtlinie, digital erweitert)  
 # DTA-Basisformat: Referenz-Dokumentation  
 # <http://exist-db.org>  
 # <https://www.oxygenxml.com>

## Was ist neu in dieser Dokument-Version?

### v1.3 vom 25.11.2020

- Korrektur in [Projekt im Oxygen XML Author anlegen \(auf Seite 14\)](#) zu zwei Editorvariablen: Am Ende des Pfades muss ein Schrägstrich stehen ("/"). Wenn diese Editorvariablen nicht korrekt eingetragen sind, können Sie zum Beispiel die Registerfunktionalitäten nicht benutzen.  
 # Für \${ediarum\_projects\_directory} zum Beispiel: /exist/rest/db/projects/  
 # Für \${ediarum\_projects\_directory\_webdav} zum Beispiel: /exist/webdav/db/projects/
- Korrektur in [Projekt in der Datenbank anlegen \(auf Seite 8\)](#) zur Umbenennung von Datenbank-Ordern für XML-Daten: Sie sollten die bei der Installation der ediarum.DB App automatisch angelegten Ordner nicht in der eXist-db umbenennen, sondern erst nachdem Sie das Basis-Setup abgeschlossen haben, per Oxygen XML Author.
- Neues Kapitel: [Installationstipps eXist-db und Oxygen \(auf Seite 5\)](#)

## Änderungsverlauf

Dokument-Version	Publikations-Datum	Software-Versionen	Änderungen
v1.3	25.11.2020	Oxygen XML Author v20.1 eXist-db v3.5 oder v4.4 ediarum.BASE.edit v1.1.1 ediarum.DB v3.2.5 ediarum.REGISTER.edit v1.0.1	Korrekturen aufgrund des ediarum-Workshops 09.-11.03. 2020.
v1.2	12.02.2020	Oxygen XML Author v20.1 eXist-db v3.5 oder v4.4 ediarum.BASE.edit v1.1.1 ediarum.DB v3.2.5 ediarum.REGISTER.edit v1.0.1	Erstpublikation auf <a href="https://www.ediarum.org">https://www.ediarum.org</a>

## Überblick über das Basis-Setup

---

ediarum steht auf github zur Verfügung, in Form von Konfigurationskomponenten für den Oxygen XML Author (den sogenannten Frameworks) und für die XML-Datenbank eXist-db (den sogenannten Apps).

ediarum baut auf zwei Programmen auf:

- Oxygen XML Author: XML-Editor, mithilfe dessen die Editionstexte als XML-Dateien erstellt werden.
- eXist-db: XML-Datenbank, in der die edierten XML-Dateien gespeichert werden.

In beiden Programmen ist es möglich, die Funktionalität zu konfigurieren und zu erweitern. Desweiteren ist es möglich, diese Konfigurationen gebündelt an andere Programm-Instanzen weiterzugeben, zum Beispiel an alle Mitglieder Ihrer Arbeitsgruppe:

- Im Oxygen XML Author kann man eigene GUI-Elemente für XML-Bearbeitungsfunktionen erstellen. Der Container, in dem diese Add-On-Funktionalitäten gespeichert werden, heißt im Oxygen XML Author "Framework" oder auch "Dokumenttyp-Zuordnung". Diese Dokumentation verwendet den Term "Framework".
- In eXist-db kann man Webanwendungen mit Datenbankfunktionalitäten erstellen. Diese Webanwendungen heißen in eXist-db "Apps".

Nach erfolgreichem Basis-Setup von ediarum haben Sie mehrere solcher Oxygen-Frameworks und eXist-Apps sowie zwei "Projekte" in Ihrer lokalen Umgebung:

In Oxygen:

- Frameworks:
  - # ediarum.BASE.edit
  - # ediarum.REGISTER.edit
  - # meineEdition.edit
- Projekt: Enthält die Konfiguration einer REST-Schnittstelle mit der Datenbank.

In eXist-db:

- App: ediarum.DB
- Projekt: Enthält ediarum-spezifische Datenhaltungsstrukturen und Datenbanksettings.

Das Einrichten der Komponenten erfolgt in folgender Reihenfolge:

1. eXist-db und Oxygen XML Author installieren.
2. eXist-db: Die App ediarum.DB einrichten und ein Projekt erstellen.
3. Die WEBDAV-Verbindung zwischen eXist-db und Oxygen XML Author konfigurieren
4. Oxygen XML Author: Die Frameworks ediarum.BASE.edit und ediarum.REGISTER.edit einrichten. Ein eigenes Framework meineEdition.edit anlegen, das ediarum.BASE.edit erweitert.
5. Oxygen XML Author: Ein Projekt anlegen.

<https://github.com/ediarum>

<https://www.oxygenxml.com>

<http://exist-db.org>

## Installationstipps eXist-db und Oxygen

---

Die Installationsanleitungen für eXist-db und Oxygen finden Sie auf deren Webseiten. Die hier versammelten zusätzlichen Tipps basieren auf der Erfahrung der ediarum-Entwickler\*innen und können bei typischen Installationsproblemen hilfreich sein.

### eXist-db

- Für ediarum benötigen Sie die eXist-db-Version 3.5 oder 4.4.
- Java muss installiert und aktuell sein.
- Beachten Sie die Installationsvoraussetzungen auf der Webseite von eXist, insbesondere hinsichtlich der Java-Version.

Installation unter Windows:

- Speicher auf 1024 MB reduzieren.
- **Nicht** als "Service at Startup" installieren. "Service at Startup" kann zu Problemen beim Starten von eXist-db führen, zumindest in eXist-db Version 3.5.

Deinstallation von alten Versionen:

- Der lokale db Ordner (meist C-Laufwerk) wird bei der Deinstallation manchmal nicht automatisch gelöscht und muss deshalb manuell entfernt werden. Dazu ist nach der Deinstallation mit der uninstall.jar manchmal ein Neustart erforderlich, bevor der Ordner sich löschen lässt.

## Oxygen XML Author

- Für ediarum benötigen Sie den Oxygen XML Author, mindestens in der Version 20.1 (alternativ: Oxygen XML Editor).

# eXist-db: Installationsvoraussetzungen und -anleitungen

# eXist-db Installation auf einem Server

# eXist-db 3.5.0 - Download

# eXist-db 4.4.0 - Download

# Oxygen XML Author: Anleitungen für Installation und Benutzung

## Datenbank einrichten

---

Sie installieren sich die eXist-App ediarum.DB. Mithilfe dieser App generieren Sie in Ihrer Datenbank ein eigenes Projekt, das alle benötigten Settings für das Zusammenspiel von Datenbank und Oxygen XML Author enthält.

Diese Dokumentation bietet einen grundlegenden Einstieg, wie Sie vorgehen. Details finden Sie in den .md Dateien des ediarum.DB Repositoriums auf github, z.B. folgende Informationen:

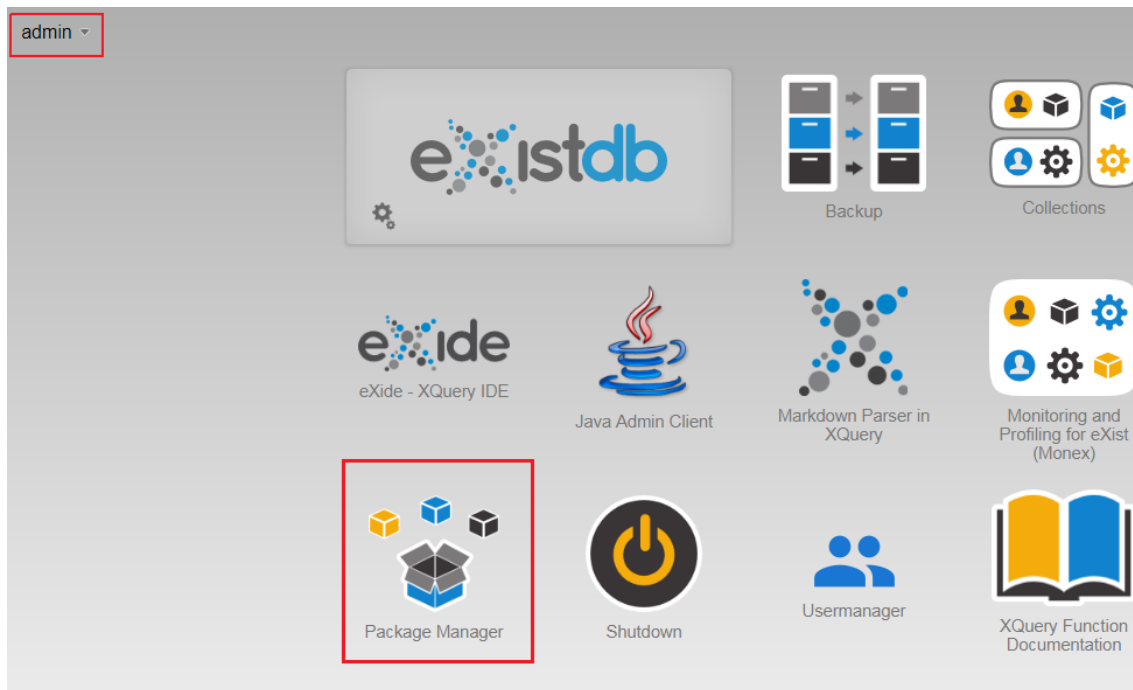
- Hintergrundinformationen und Details zu Installation und Setup
- Beschreibung der Benutzungsoberfläche der eXist-App ediarum.DB
- Konfigurationsinformationen, die Ihnen helfen, ediarum.DB besser zu verstehen und an Ihre technischen Anforderungen anzupassen

<https://github.com/ediarum/ediarum.DB>

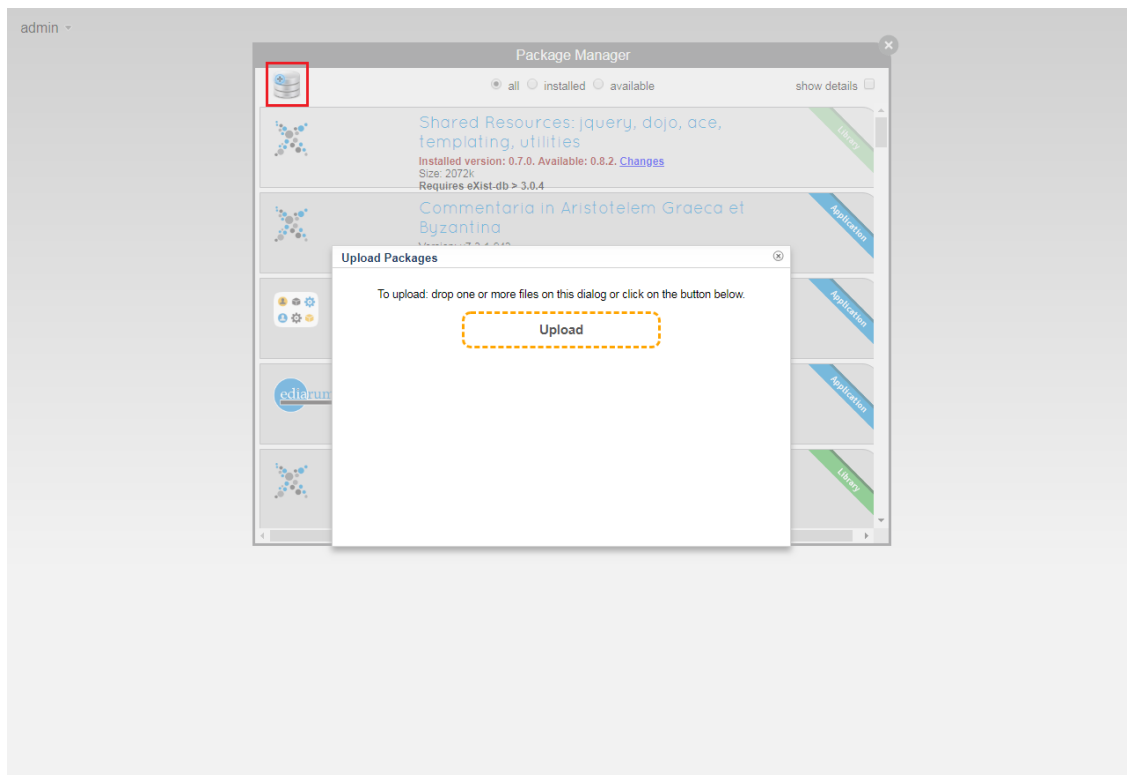
## ediarum.DB App installieren

Über das Dashboard der eXist-db installieren Sie ediarum.DB mithilfe der XAR-Datei. Danach enthält Ihre Datenbank einige grundlegende Funktionen und Zugriffs-Einstellungen und Sie können die ediarum.DB-App verwenden.

1. Klicken Sie auf der Webseite <https://github.com/ediarum/ediarum.DB/releases> auf den Button **Latest Release** und speichern Sie die Datei `ediarum.db-#version#.xar` lokal ab.
2. Melden Sie sich im Dashboard Ihrer eXist-db mit Admin-Rechten an.
3. Öffnen Sie die **Package Manager**-App.



4. Klicken Sie auf den Button **add a package** und ziehen Sie die gespeicherte Datei `ediarum.db-#version#.xar` in das Fenster **Upload Packages**. Oder: Klicken Sie auf **Upload** und wählen Sie die XAR-Datei über den Dateixplorer aus. Schließen Sie das Fenster **Package Manager**.



- Im Dashboard sehen Sie nun die ediarum.DB-App.



- In Ihrer eXist-db wurde nun Folgendes eingerichtet:
  - # Nutzer-Gruppen: website, oxygen
  - # Nutzer: exist-bot, oxygen-bot, website-user, mit entsprechend lautenden Passwörtern
  - # Zugriffsrechte zu benötigten Verzeichnissen und Prozessen

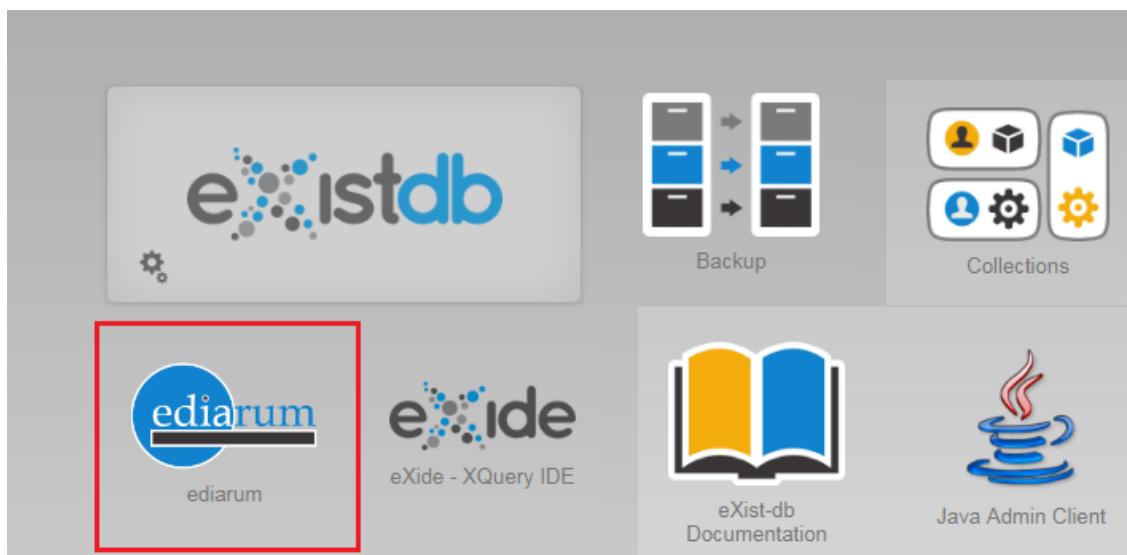
# <https://github.com/ediarum/ediarum.DB/blob/master/INSTALLATION.md>

## Projekt in der Datenbank anlegen

In der Datenbank muss für Ihre Edition ein Projekt mit einer standardisierten Ablagestruktur und einigen standardisierten Settings vorhanden sein. Um die Standards automatisiert korrekt anzulegen, benutzen Sie die ediarum.DB-App.

Sie sind im Dashboard Ihrer eXist-db mit Admin-Rechten angemeldet.

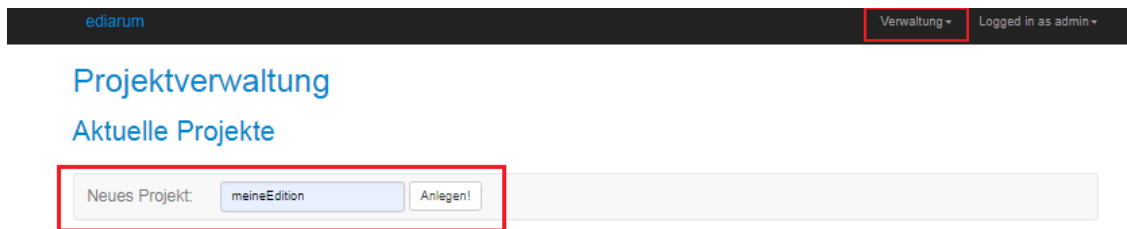
1. Öffnen Sie die ediarum.DB-App.



2. Wählen Sie in der App das Menü **Verwaltung** → **Projekte** aus.
3. Schreiben Sie in das Feld **Neues Projekt** einen Projektnamen für Ihre Edition und klicken Sie auf **Anlegen!**.



Den Projektnamen müssen Sie später für die Datenbankanbindung angeben. **Verwenden Sie keine Umlaute, Leer- oder Sonderzeichen!**



In der Datenbank wird ein neuer Ordner mit dem gewählten Namen angelegt.

Die angezeigte Projektliste enthält nun Ihr Projekt. Wenn Sie Ihr Projekt anklicken, sehen Sie die Ordnerstruktur, in der die XML-Dateien Ihrer Edition gespeichert werden. Später, nachdem Sie das Basis-Setup abgeschlossen haben, können Sie die Ordner im Oxygen XML Author umbenennen.

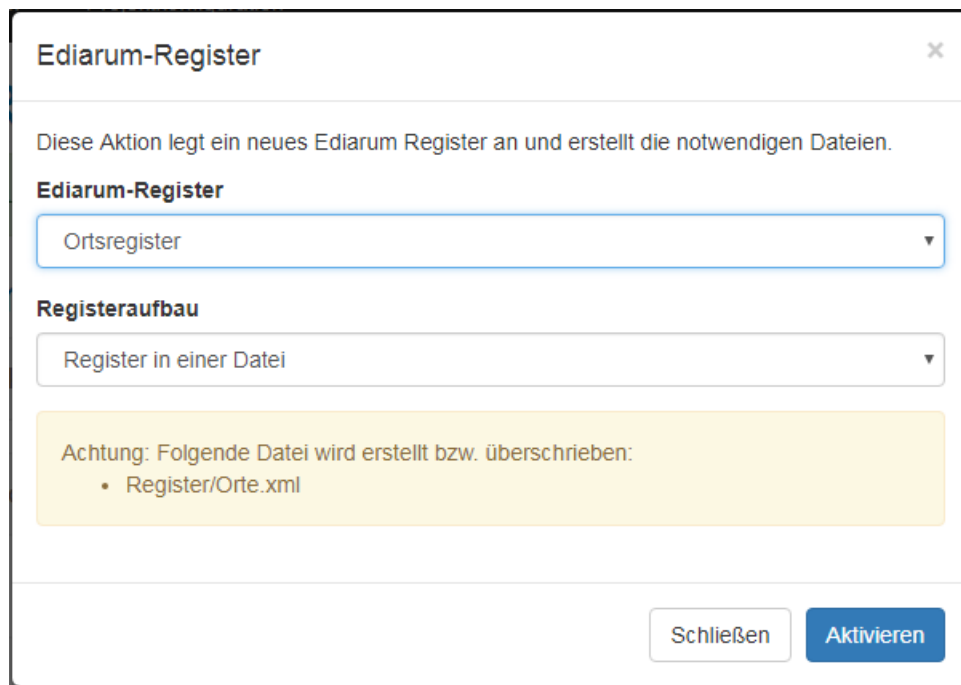


## ediarum-Register aktivieren

Sie aktivieren die mitgelieferten Personen- und Ortsregister, die typischerweise in einer Edition benötigt werden. Sie haben die Möglichkeit, zusätzlich weitere ediarum-Standardregister zu aktivieren: Sachbegriffe, Körperschaften, Werke, Briefe, Anmerkungen.

Sie haben in Ihrer eXist-DB die ediarum.DB-App mit Admin-Rechten geöffnet.

1. Wählen Sie das Menü **Verwaltung** → **Projekte** aus.
2. Klicken Sie Ihr Projekt an, um die Übersichtsseite des Projekts zu öffnen.
3. Wählen Sie das Menü **Projektkonfiguration** → **Register** aus.
4. Führen Sie die folgenden Schritte für jedes Register aus, das Sie in Ihrer Editionsarbeit benutzen möchten, mindestens für das Personen- und das Ortsregister.
  - a. Klicken Sie in der Liste **ediarum-Register** im Listeneintrag **Neues Register** auf **Aktivieren**.
  - b. Wählen Sie den Register-Typ (im Feld **Ediarum-Register**) und den Register-Aufbau und klicken Sie auf **Aktivieren**.



Die Standard-Ordnerstruktur auf der Übersichtsseite Ihres Projekts ist um die aktivierten Register ergänzt.



**Ein eigenes Register einrichten** [Wenn Sie ein Register benötigen, das nicht bereits als Standardregister in ediarum enthalten ist, legen Sie dieses in eXist-db in der ediarum.DB-App an.] (auf Seite 18)

## Nutzer-Account anlegen

Die Nutzer-Accounts in der ediarum.DB-App sind Voraussetzung, damit die Editor/innen ihre Oxygen-Instanzen mit der eXist-db verbinden können.

Sie haben in Ihrer eXist-db die ediarum.DB-App mit Admin-Rechten geöffnet.

Sie benötigen mindestens einen Nutzer-Account, um im späteren Setup-Verlauf die Datenbankanbindung konfigurieren zu können. Wenn Sie möchten, können Sie gleich weitere Nutzer-Accounts anlegen - pro Editor\*in einen Account.

1. Wählen Sie das Menü **Verwaltung** → **Projekte** aus.
2. Klicken Sie Ihr Projekt an, um die Übersichtsseite des Projekts zu öffnen.
3. Wählen Sie das Menü **Projektkonfiguration** → **Benutzer** aus.
4. Geben Sie im Feld **Neues Projektmitglied** einen Benutzernamen und ein Passwort ein.

Der Benutzername ist der Name, den Sie später für die Datenbankanbindung angeben müssen. Verwenden Sie keine Umlaute, Leer- oder Sonderzeichen!

5. Klicken Sie auf **Anlegen**.

## Arbeitsumgebung in Oxygen XML Author einrichten

Sie verbinden Oxygen XML Author mit eXist-db (Datenbank) und machen alle in ediarum enthaltenen Funktionalitäten verfügbar.

Verwenden Sie Oxygen XML Author, mindestens in der Version 20.1. Für die gesamte Einrichtung und Benutzung von ediarum ist es auch möglich den Oxygen XML Editor zu nutzen.

### Oxygen mit der Datenbank verbinden

Sie konfigurieren die Verbindung für den Lese- und Schreibzugriff von der Eingabeoberfläche auf die XML-Dateien. Die WebDAV-Verbindung verhindert Schreibkonflikte zwischen verschiedenen Benutzer/innen.

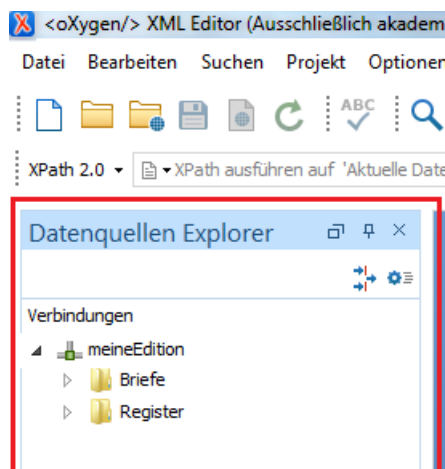
1. Öffnen Sie den Oxygen XML Author.
2. Wählen Sie das Menü **Optionen** → **Einstellungen** aus.
3. Wählen Sie in der linken Spalte des nun geöffneten **Einstellungen**-Fensters den Eintrag **Datenquellen** aus.
4. Klicken Sie in der unteren Hälfte (**Verbindungen**) auf den + Button, um eine neue Verbindung zu erstellen.
5. Füllen Sie die Angaben im nun geöffneten Fenster **Verbindung** aus.

- **Name:** Name dieser Verbindung, z.B. meineEdition. Dient Ihnen zur Unterscheidung dieser Verbindung von anderen Datenbankverbindungen, die Sie in Oxygen definiert haben. Muss nicht identisch mit dem Projektnamen in der eXist-db sein.
- **Datenquelle:** WebDAV (S)FTP
- **WebDAV:** Pfad zu Ihren Projektdaten in Ihrer eXist-Instanz, nach dem Muster: `http://HOST:PORT/webdav/db/projects/PROJEKT/data`. Zum Beispiel:  
 # `https://meinInstitut.de/exist/webdav/db/projects/meineEdition/data`  
 # `http://localhost:8080/exist/webdav/db/projects/meineEdition/data`
- **Benutzer, Passwort:** Angaben des Nutzeraccounts, den Sie in der ediarum.DB-App in eXist-db angelegt haben (**Verwaltung**→**Projekte**→**Projektkonfiguration**→**Benutzer**).

6. Klicken Sie auf **OK**.

7. Kontrollieren Sie die Datenbankverbindung:

- Wählen Sie das Menü **Fenster**→**Ansicht zeigen**→**Datenquellen Explorer** aus.  
Die hergestellte Verbindung wird im Explorer angezeigt (hier: meineEdition).
- Klicken Sie den Pfeil neben der Verbindung an, um die Ordnerstruktur zu öffnen.



Die Struktur ist identisch mit der Ordnerstruktur, die in eXist-db auf der Übersichtsseite Ihres Projekts angezeigt wird (ediarum.DB-App: **Verwaltung**→**Projekte**). Sie haben Zugriff auf die Ordner **Briefe** und **Register**.

Wenn sich die Ordner nicht aufklappen lassen und eine Fehlermeldung erscheint:

- Kontrollieren Sie die Richtigkeit Ihrer Angaben im **Verbindung**-Fenster, insbesondere:
  - **WebDAV:** Ist der Pfad bis zu der Ebene `.../data` eingetragen?
  - **Benutzer, Passwort:** Sind die Angaben identisch mit den Angaben in der **ediarum** App in eXist-db?
  - **Verfügbarkeit:** Sie können die Korrektheit und Verfügbarkeit auch dadurch prüfen, dass Sie den entsprechenden Pfad in Ihrem Browser öffnen (unter Eingabe von Benutzer und Passwort). Ihr Browser sollte Ihnen dann einen XML-Output mit den Ordnerangaben anzeigen.
- Machen Sie im **Datenquellen Explorer** einen Rechtsklick auf Ihre Edition und klicken Sie im Kontextmenü **Erneut laden** an.

## ediarum-Frameworks einrichten

Sie machen die ediarum-Funktionalitäten in Ihrer Oxygen-Editiungsumgebung verfügbar, indem Sie zwei ediarum-spezifische Oxygen-"Frameworks" einbinden.

Überblick über den Ablauf:

- Sie laden sich die beiden ediarum-Frameworks auf Ihren lokalen Rechner.
- Sie ändern den Standardordner, in dem Oxygen XML Author die Frameworks verwaltet. Dadurch vermeiden Sie Probleme mit Zugriffsrechten seitens des Betriebssystems (insbesondere Windows).
- Sie legen ein eigenes Framework für Ihre Edition an. Sie legen dieses Framework als eine Erweiterung des ediarum.BASE.edit-Frameworks an, damit Sie alle Funktionalitäten von ediarum zur Verfügung haben. Nach Abschluss des Basis-Setups können Sie in diesem Framework Ihre eigenen Konfigurationen der Editiungsumgebung speichern.

Benötigtes Vorwissen:

- Grundverständnis von github
1. Legen Sie in Ihrem lokalen Dateisystem einen Ordner für alle Frameworks an.
  2. Laden Sie sich die folgenden github-Repositorien in den Ordner, den Sie gerade angelegt haben (in github: Button **Latest release**, dann gezippte Datei lokal speichern und entpacken):

- a. <https://github.com/ediarum/ediarum.BASE.edit/releases>
- b. <https://github.com/ediarum/ediarum.REGISTER.edit/releases>

3. Legen Sie in dem Ordner zusätzlich gleich ein Verzeichnis an, in das Sie später Ihr editionsspezifisches Framework legen können.

Empfehlung: Behalten Sie im Namen des Ordners `ediarum` als ersten Teil bei, um kenntlich zu machen, dass Ihr editionsspezifisches Framework eine Erweiterung von ediarum ist. Zum Beispiel: `ediarum.meineEdition.edit`.

4. Stellen Sie sicher, dass der Framework-Ordner folgende Struktur hat:

```

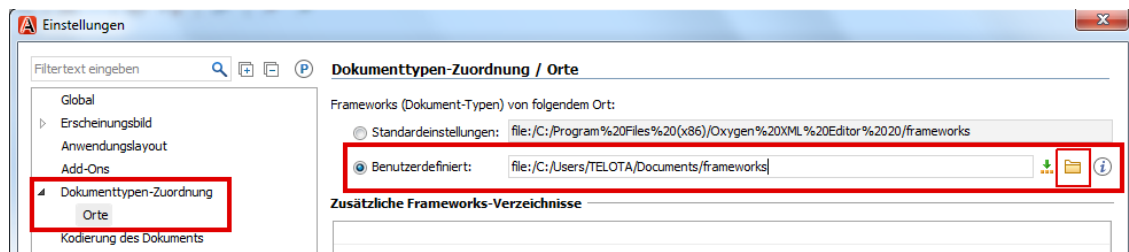
• frameworks
  # ediarum.BASE.edit
    # css
    # fonts
    # icons
    # resources
    # schemata
    # templates

  # ediarum.REGISTER.edit
    # css
    # resources
    # schema

  # ediarum.meineEdition.edit
```

5. Ändern Sie in Oxygen den Speicherort für Frameworks, sodass der Speicherort auf den Ordner verweist, der alle Framework-Ordner enthält:

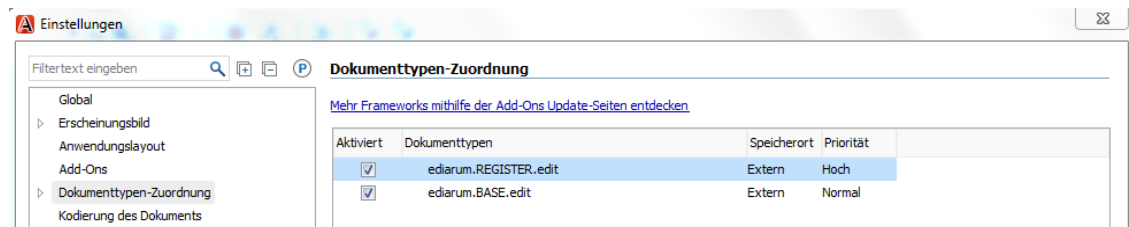
- a. Wählen Sie in Oxygen das Menü **Optionen** → **Einstellungen** aus.
- b. Wählen Sie im nun geöffneten **Einstellungen**-Fenster den Eintrag **Dokumenttypen-Zuordnung** → **Orte** aus.
- c. Aktivieren Sie die Option **Benutzerdefiniert**.
- d. Benutzen Sie das Ordner-Icon, um den Pfad auszuwählen.



6. Klicken Sie auf **OK**, um das **Einstellungen**-Fenster zu schließen.

7. Starten Sie Oxygen XML Author neu.

Unter **Optionen** → **Einstellungen** → **Dokumenttypen-Zuordnung** werden die zwei ediarum-Frameworks angezeigt.



8. Legen Sie das Erweiterungsframework für Ihre Edition an.

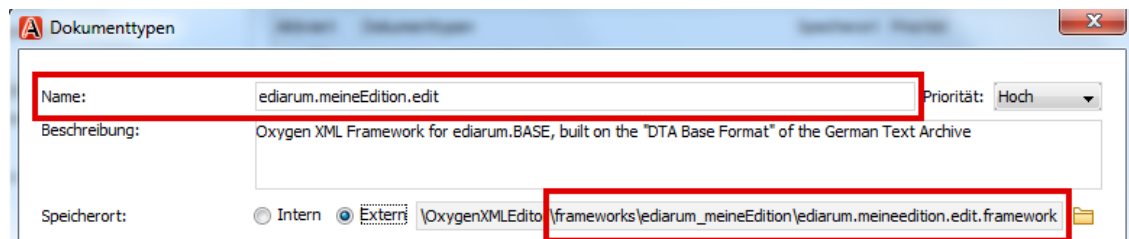
a. Wählen Sie im Fenster **Dokumenttypen-Zuordnung** das **ediarum.BASE.edit** Framework aus (nicht doppelklicken!) und klicken Sie auf **Erweitern**.

b. Geben Sie im Feld **Name**: den Namen für Ihr editionsspezifisches Framework ein.

Empfehlung: Verwenden Sie denselben Namen, den Sie dem editionsspezifischen Ordner in Ihrem Dateisystem gegeben haben. Zum Beispiel: `ediarum.meineEdition.edit`

c. Aktivieren Sie für **Speicherort**: die Option **Extern**.

d. Geben Sie den Pfad bis zu der .framework-Datei an, die innerhalb Ihres Editionsordners liegt. Zum Beispiel: `ediarum.meineEdition.edit/ediarum.meineEdition.edit.framework`. Benutzen Sie das Ordner-Icon, um die Datei auszuwählen.



9. Klicken Sie auf **OK**, um das **Einstellungen** Fenster zu schließen.

Über **Oxygen-Frameworks** und **ediarum-Spezifika** [Das grundsätzliche Vorgehen für die Konfiguration via sogenannter "Frameworks" finden Sie in der Oxygen-Dokumentation. Ediarum stellt für diese Konfiguration einige Standardbausteine bereit.] (auf Seite 17)

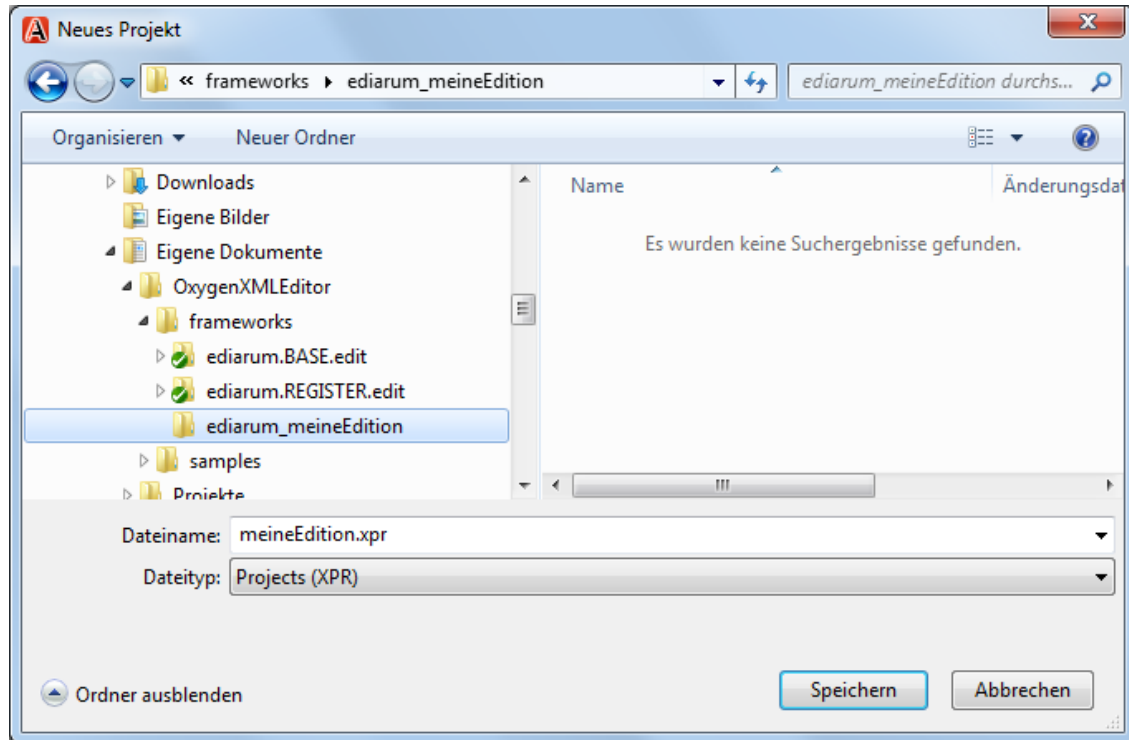
## Projekt im Oxygen XML Author anlegen

Für das Register-Lookup ist eine eigene Datenbankverbindung nötig, die mithilfe von sogenannten "Editorvariablen" in einem Oxygen-Projekt hinterlegt wird.

Für das Basis-Setup von ediarum müssen mindestens die für das Register-Lookup benötigten Editorvariablen im Projekt gespeichert werden. Eine "Editor"-Variable ist eine generische Angabe in dem Tool Oxygen XML "Editor" (oder Oxygen

XML Author). Oxygen XML Author speichert diese Einstellungen in einer Datei mit der Endung XPR. Im Anschluss an das Basis-Setup dient das "Projekt" später als Container für weitere editionsspezifische Konfigurationen.

1. Wählen Sie **Projekt** → **Neues Projekt** aus.
2. Wählen Sie im nun geöffneten **Neues Projekt**-Fenster den Ordner Ihres editionsspezifischen Frameworks aus.
3. Geben Sie einen Namen für die Projektdatei ein (zum Beispiel: `meineEdition.xpr`) und klicken Sie auf **Speichern**.



4. Legen Sie fest, dass die Editorvariablen im Projekt gespeichert werden:
  - a. Wählen Sie das Menü **Optionen** → **Einstellungen**.
  - b. Wählen Sie im nun geöffneten **Einstellungen**-Fenster den Eintrag **Benutzerspezifische Editorvariablen** aus.
  - c. Aktivieren Sie die Option **Projekt-Optionen** und klicken Sie auf **Anwenden**.

Der Eintrag **Benutzerspezifische Editorvariablen** ist jetzt mit einem **[P]** gekennzeichnet und fett dargestellt.

5. Klicken Sie auf **Neu** und tragen Sie im nun geöffneten **Editorvariablen**-Fenster die Editorvariablen `${ediarum_project_name}` und `${ediarum_project_domain}` samt Werten ein.

<b>Name:</b> <code>\${ediarum_project_name}</code>	Der Wert muss mit dem in eXist-db festgelegten Projektnamen übereinstimmen.	<b>Wert:</b> (Beispiel) meineEdition
<b>Name:</b> <code>\${ediarum_project_domain}</code>	Basis-URL zur Datenbank. Der Wert muss Domain, ggf. Server-Port und Server-Anmeldedaten enthalten.	<b>Wert:</b> (Beispiel) <code>http://oxygen-bot:*****@example.de:8028</code>

6. Klicken Sie auf **Anwenden** und **OK**.
7. Prüfen Sie, ob folgende Editorvariablen für die Datenbankanbindung passend zu Ihrer Infrastruktur gesetzt sind.

<b>Name:</b> <code>\${ediarum_projects_directory}</code>	Pfad zu den Projektordnern in der eXist-DB via REST-Verbindung.	<b>Wert:</b> (Standard) <code>/exist/rest/db/projects/</code> Anstelle von <code>exist</code> ggf. eine andere Angabe, z.B. <code>ediarum</code>
<b>Name:</b> <code>\${ediarum_projects_directory_webdav}</code>	Pfad zu den Projekt-Ordern in der eXist-DB via WebDAV-Verbindung.	<b>Wert:</b> (Standard) <code>/exist/webdav/db/projects/</code> Anstelle von <code>exist</code> ggf. eine andere Angabe, z.B. <code>ediarum</code>

8. Falls Sie Änderungen vorgenommen haben, klicken Sie auf **Anwenden** und **OK**.

Wenn Sie das Dokument-Template aus dem ediarum.BASE.edit-Framework öffnen (**Datei** → **Öffnen...** → **.../frameworks/ediarum.BASE.edit/templates**), werden im Autor-Modus die ediarum-spezifischen Schaltflächen angezeigt. Sie können nun Dokumente und Briefe edieren und mit den aktivierten Registern arbeiten.

[Erklärung der ediarum-Schaltflächen](#)

[Was ist eine Editorvariable?](#)



## Kapitel 2. Editionsspezifische Konfiguration

Wenn für Ihre Edition ein umfassenderes XML-Schema benötigt wird als das mit ediarum abgedeckte, können Sie das Schema und die Arbeitsumgebung erweitern. Sie können außerdem zusätzliche Register erstellen.

Ausgehend vom ediarum Basis-Setup haben Sie die Möglichkeit, für Ihr Editionsprojekt folgende Einstellungen vorzunehmen:

- Buttons und Funktionen hinzufügen oder ändern, indem Sie das für Ihre Edition angelegte Framework konfigurieren.
- Eigene Register einrichten.
- Literaturbestände, die Sie im Programm Zotero verwalten, als Register einbinden.

## Über Oxygen-Frameworks und ediarum-Spezifika

Das grundsätzliche Vorgehen für die Konfiguration via sogenannter "Frameworks" finden Sie in der Oxygen-Dokumentation. Ediarum stellt für diese Konfiguration einige Standardbausteine bereit.

### Framework-Komponenten

Im Framework ist die Oxygen-Benutzungsoberfläche für Ihr Oxygen-Projekt mit Ihrem Datenmodell (XML-Struktur) definiert, ggf. für verschieden strukturierte Dokumenttypen (z.B. Brief, Manuskript, Notizbuch). Diese Benutzungsoberfläche umfasst folgende Elemente:

- Vorlagen für neue Dokumente, sodass die Editor/innen die XML-Struktur nicht "von null" alleine aufbauen müssen (XML-Template)
- Lesefreundliches Layout der XML-Inhalte im "Autor"-Tab, d.h. eine "What-you-see-is-what-you-get"-Schreiboberfläche, sodass die Editor/innen nicht direkt in XML schreiben müssen und ihre Textarbeiten übersichtlich und publikationsnah überprüfen können.
- Buttons und Aktionen in der Werkzeugleiste, sodass die Editor/innen XML-Elemente und Attributwerte per Klick und in Eingabefeldern einfügen können, ohne die benötigten XML-Elemente kennen zu müssen
- ggf. Transformationsszenarien, die verschiedene Publikations-Layouts zur Verfügung stellen, z.B. für Korrekturdurchläufe

In Ihrem eigenen Erweiterungs-Framework können Sie eigene Elemente für die Oxygen-Benutzungsoberfläche konfigurieren.

### Konfigurationskomponenten

Im Hintergrund besteht das Framework aus einer Sammlung von Dateien. Einige Aspekte der Konfiguration nehmen Sie direkt in den Dateien vor, für andere benutzen Sie das **Dokumenttypen**-Fenster.

**Tabelle 1: Konfigurationskomponenten**

Komponente		Speicherung	Änderung
Was können Sie konfigurieren?		Wo speichert Oxygen XML Author Ihre Konfigurationen?	Auf welchem Weg können Sie die Konfigurationen ändern?
Schema	XML-Schema (Strukturvorgaben)	RNG-Datei	Die RNG-Datei bearbeiten
Template	Vorlage für neue XML-Dokumente	XML-Datei	Die XML-Datei bearbeiten
GUI-Funktionalitäten	GUI-Elemente und -Funktionalitäten, die Oxygen XML Author dann im Autor-Modus bereitstellt	FRAMEWORK-Datei	Im <b>Autor</b> Tab des <b>Dokumenttypen</b> Fensters: <b>Optionen#Einstellungen#Dokumenttypen-Zuordnung</b> , Framework auswählen

### ediarum-Spezifika für die Konfiguration per Framework

Ableitungsstruktur: ediarum.BASE.edit arbeitet mit dem Datenmodell des DTA-Basisformats (Subset von TEI-XML) und erweitert dieses an einigen Punkten. Für Ihr Editionsprojekt erweitern Sie wiederum das Datenmodell von ediarum.BASE.edit. Das Datenmodell für Ihr Editionsprojekt enthält also am Ende unterschiedlich generische Anteile, die voneinander abgeleitet sind: DTA-Basisformat > ediarum.BASE.edit > projektspezifische Schema-Anteile. Entsprechend dieser Ableitungsstruktur beim Datenmodell sind auch die Frameworks voneinander abgeleitet.

Dateinamen und Ordnerstrukturen: Für die Benennung und den Ablageort der Konfigurationsdateien folgt ediarum eigenen Konventionen, unter anderem, um eine einfache gebündelte Auslieferung eines Frameworks zu ermöglichen. Orientieren Sie sich an Struktur und Inhalten des ediarum.BASE.edit Ordners.

Oxygen-"Operationen": Die GUI-Funktionalitäten bestehen im Kern aus sogenannten "Operationen", d.h. Befehls-Bausteinen, aus denen Sie eine eigene Funktionalität zusammensetzen und mithilfe von Parametern auf Ihre Edition anpassen können. Dabei gibt es Operationen, die der Oxygen XML Author mitliefert, und solche, die eigens für ediarum programmiert und bereitgestellt wurden. Die Datei ediarum.jar enthält die ediarum-spezifischen Operationen, die Sie benutzen können.

Oxygen-"Editorvariablen": Die von Oxygen XML Author und von ediarum bereitgestellten Operationen und einige andere Konfigurationskomponenten sollen einfach und vielfältig wiederverwendbar sein. Deshalb sind, wo immer kontextspezifische Angaben nötig sind, anstatt konkreter Werte Platzhalter (Variablen) eingesetzt, z.B. für Projektname, Datei-, Ordner- und URL-Pfade. Die Werte dieser sogenannten Editor-Variablen sind einmal an zentraler Stelle definiert. Für Ihre eigene editionsspezifische Konfiguration müssen Sie die Werte für projektspezifische Editor-Variablen eintragen und gegebenenfalls zusätzliche, eigene Editor-Variablen erstellen, z.B. für zusätzliche Registerarten. Es empfiehlt sich, dabei die Namenskonventionen der bestehenden Editorvariablen konsistent weiter zu führen.

[Oxygen-Anleitung zur Anpassung eines Frameworks](#)

[Tutorial: Wie baue ich ein eigenes Framework für Oxygen XML?](#)

## Ein eigenes Register einrichten

---

Wenn Sie ein Register benötigen, das nicht bereits als Standardregister in ediarum enthalten ist, legen Sie dieses in eXist-db in der ediarum.DB-App an.

1. Wählen Sie das Menü **Verwaltung** → **Projekte** aus.
2. Klicken Sie das Feld Ihres eigenen Projekts an.
3. Wählen Sie das Menü **Projektkonfiguration** → **Register** aus.
4. Stellen Sie sicher, dass das gewünschte Register nicht bereits in der angezeigten Registerliste enthalten ist.
5. Klicken Sie im Feld **Neue Registeranbindung (Projekt)** auf **Hinzufügen**.
6. Füllen Sie in dem geöffneten Fenster alle Parameter aus.

Abbildung 10: Fenster für den Eintrag von Register-Parametern

Register (Projekt) ✕

API-ID

ID

Bezeichnung

Name

Registerordner / -datei

Register/Personen

Namespace

tei:http://www.tei-c.org/ns/1.0

Node

tei:TEI

XML-ID

@xml:id

Span

//tei:title

Schließen

Speichern

Tabelle 2: Beispielhafte Werte, anhand eines Registers für Maßangaben		
	Beispielwert	
Struktur eines Registereintrags	<pre>&lt;item xml:id="mark_koelnisch"&gt;   &lt;label&gt;Kölnische Mark&lt;/label&gt; &lt;/item&gt;</pre>	
API-ID	units	frei wählbar, wird bei den Editorvariablen angegeben (sollte nicht leer sein)
Bezeichnung	Maße	Name des Registers (frei wählbar)
Registerordner/ -datei	Register/Maße.xml	Register/NAME.xml
Namespace	tei:http://www.tei-c.org/ns/1.0	tei:http://www.tei-c.org/ns/1.0
Node	tei:item	orientiert sich am Aufbau des Registers
XML-ID	@xml:id	@xml:id
Span	//tei:label	orientiert sich am Aufbau des Registers

7. Klicken Sie auf **Speichern**.

In der Registerliste sehen Sie nun das neue Register. Über den Button **API-Links** sind die URLs für den Aufruf des Registers zugänglich.

8. Auf Basis der API-Links müssen Sie nun in Ihrem Oxygen-Projekt Editorvariablen definieren, sodass Sie registerspezifische Buttons definieren können, z.B. für Einfügeaktionen. Orientieren Sie sich dafür an den Editorvariablen und Aktionen, die für die Standardregister bereits in ediarum enthalten sind.