

# Metadatenschema für die Langzeitarchivierung von Publikationsprozessen am Beispiel OJS

Karl-Arnold Bodarwé, UB Regensburg



# Ausgangslage

- Wissenschaftliche Zeitschriftenartikel werden üblicherweise nur mit der fertigen Druckfahne und dazugehörigen Metadaten archiviert (PDF + DC Terms).
- Der Entstehungsprozess des Artikels wird üblicherweise nicht mit veröffentlicht und damit auch nicht mit archiviert.
- Informationen über die Entstehung des Artikels, Kontext, Reviews, Kommunikation, gehen damit verloren.



# Datenstruktur

## Kommunikation

Mails

Notizen

Kommentare

## Peer Reviews

Eingereichte Dokumente

Revisionen

Entscheidungen

Daten der Beteiligten

## Preprints

Manuskripte

Vorabversionen

Layout/Copyediting



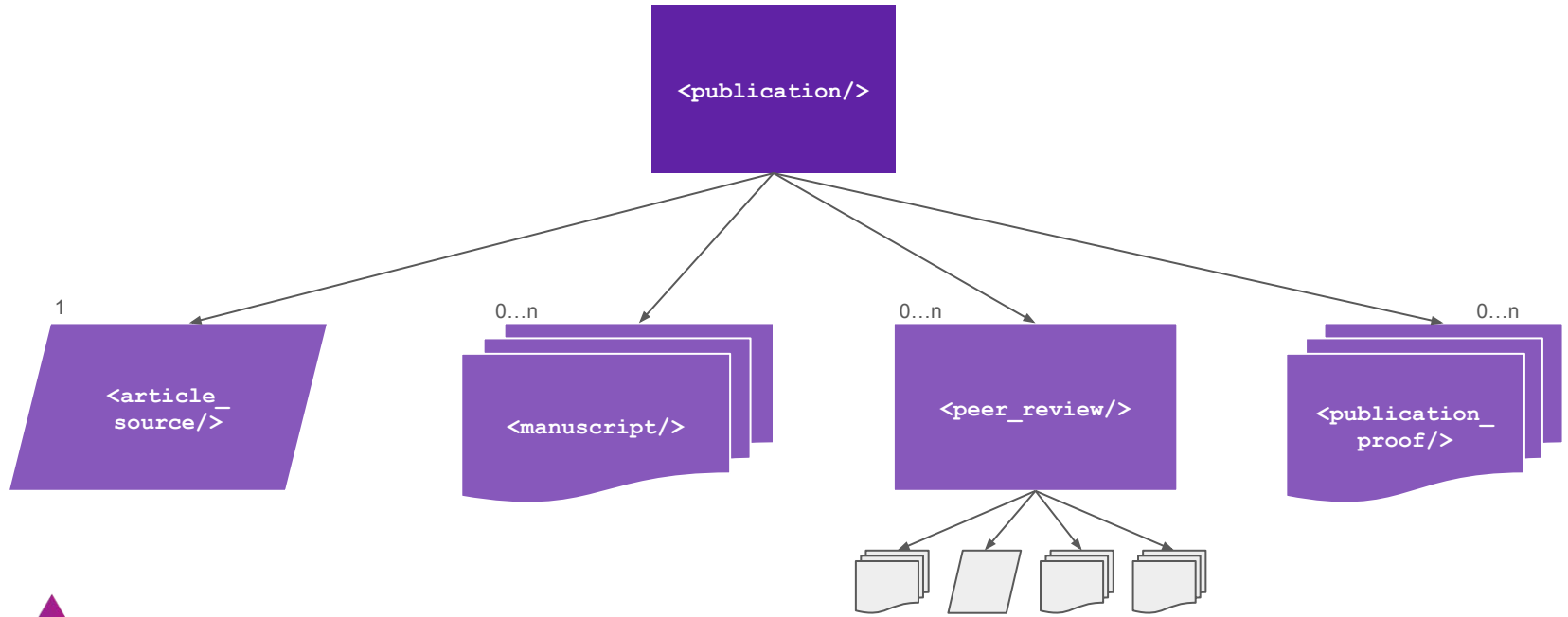
# Eigenes Metadatenschema

Ziele:

- Wiederverwendbarkeit für verschiedene Anwendungen
- Flexibel nutzbar für verschiedene Fälle
- Möglichst wenige proprietäre Lösungen



# Aufbau Metadatenschema

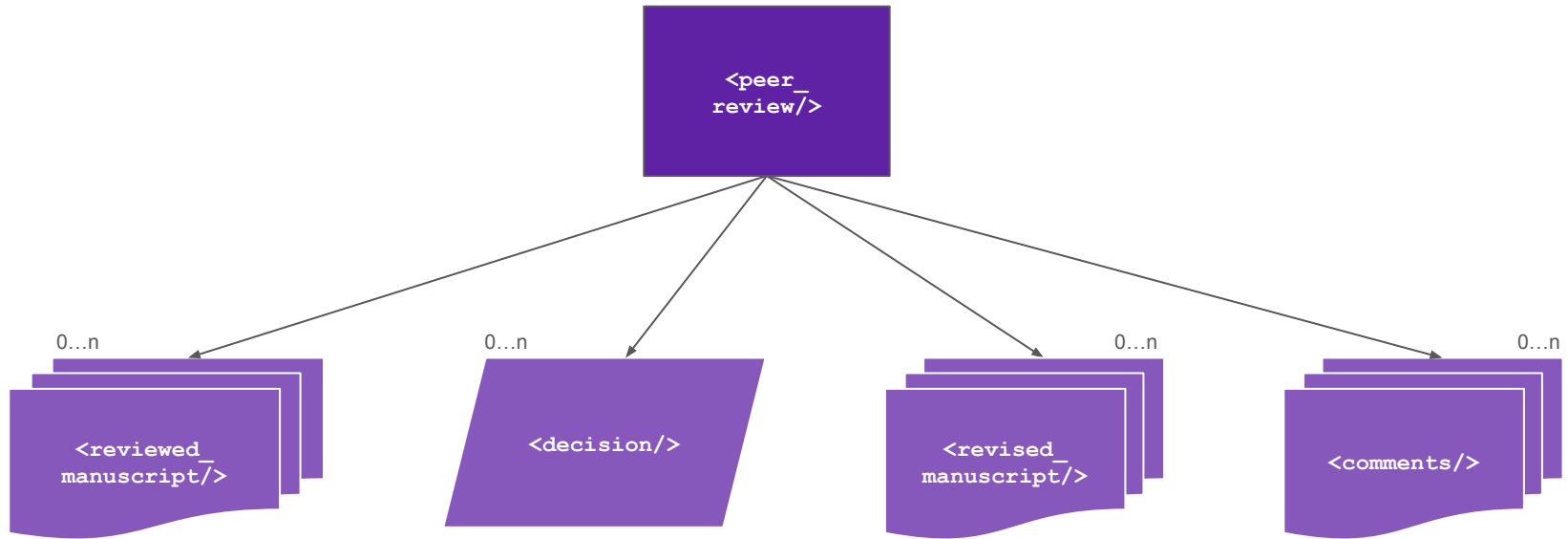


# Dokumenten Nodes

- Enthalten Metadaten über angehängte Dokumente
- alle DC Terms 1.1 stehen zur Verfügung
- Sinnvoll mindestens:
  - creator
  - date
  - format
  - type
  - relation (zur Verlinkung des Dokuments)



# Node peer\_review



# Verwendung im LZV-Bayern Projekt

- Archivierung von Open Access Zeitschriften
- Publikationsprozesse aus der Software OJS werden exportiert



OJS  
Open  
Journal  
Systems



# OJS - Open Journal Systems

- Software zum Verwalten und Publizieren von wissenschaftlichen Zeitschriften
- Entwickelt von PKP Public Knowledge Project
- Besteht seit 2012, wird stetig aktuell gehalten
- Über 30.000 aktive Journals weltweit (2021), Tendenz steigend



# OJS - Vorteile

- Open Source und Kostenfrei
- Kompletter Publikationsprozess von Einreichung bis Veröffentlichung
- Vielzahl an vorinstallierten Plugins für Metadaten, Anbindung an Crossref, DOAJ etc.
- Aktive Community, regelmäßige Updates



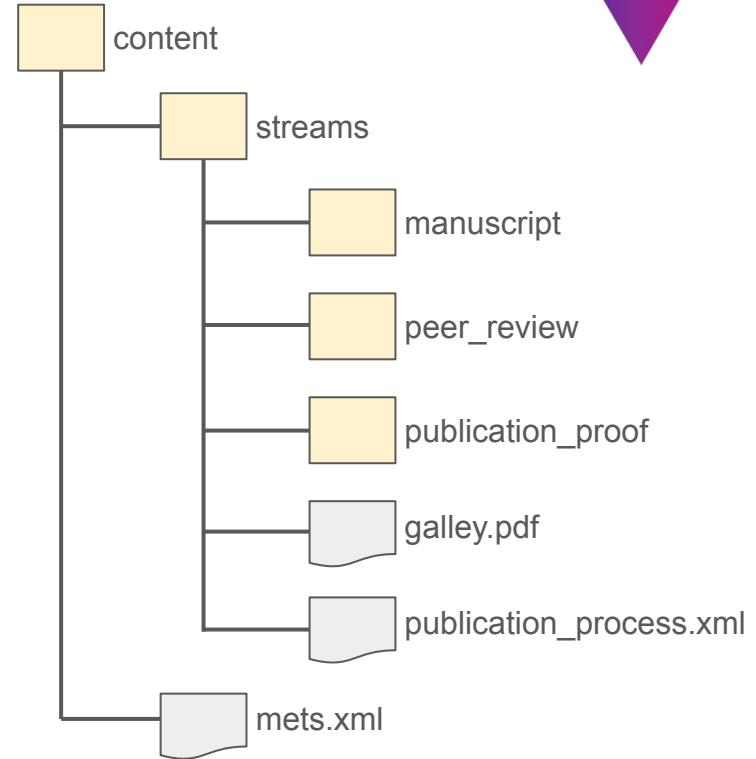
# Funktion OJS-Plugin

- Ansteuerung über Kommandozeile (Cronjob mit Shellsript)
- Alle zusätzlichen Daten des Entstehungsprozesses werden mit exportiert:
  - Submission Files
  - Entscheidungen
  - Diskussionen / Reviews
  - Daten der Beteiligten: Name, Affiliation, ORCID, soweit in OJS angegeben.
- Fertiges SIP für Rosetta wird ausgegeben und an einem Zielort abgelegt.
- Moderater Installations- und Konfigurationsaufwand
- Erhöhter rechtlicher Aufwand auf Datengeberseite



# Archivierung in Rosetta

- publication\_process.xml ist Teil des exportierten SIP



# Ausblick

- Feedback ausdrücklich erwünscht.
- Schema herunterladbar unter [https://github.com/ub-regensburg/publication\\_process](https://github.com/ub-regensburg/publication_process)
- Integration in Rosetta angestrebt.



# Vielen Dank

Bei Fragen zu oder Interesse an dem Thema freue ich mich über eine email:

[karl-arnold.bodarwe@ur.de](mailto:karl-arnold.bodarwe@ur.de)

