AUTOMATIC DATA ENRICHMENT: MERGING METADATA FROM SEVERAL SOURCES

Frank Lützenkirchen (Duisburg-Essen University Library), Kathleen Neumann (Head Office of GBV)

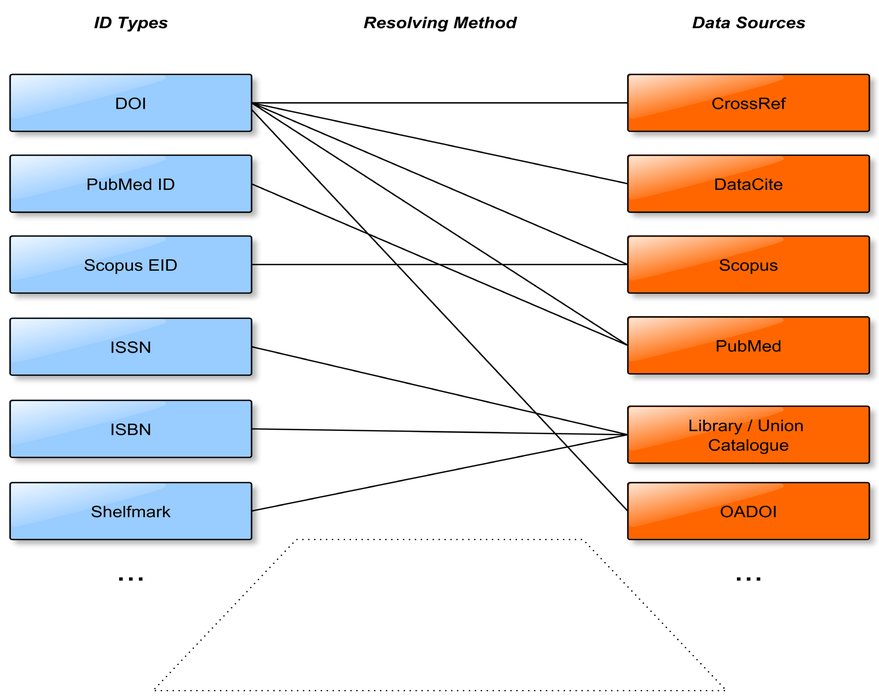
Today it is possible to uniquely identify authors and publications on common and well established ID systems such as ORCID, ISBN or DOI. Databases like CrossRef, Datacite, PubMed, IEEE or Scopus share their data using often freely accessible APIs . This opens completely new ways to automatically retrieve, merge, link, and enrich publication data.

As part of the MyCoRe Repository Framework, we implemented an improved mechanism for importing and enriching bibliographic data, the so called „Enrichment Resolver“. Incompleteness and ambiguity of publication metadata is common. Enriching data from external sources helps us to create the best possible version of every single metadata record. Looking at the author entries, it is our goal to get the most complete version of it, including person identifiers like OCRID, Scopus ID and other. Additionally, extra services like DOAJ or OADOI are used to get further information like the open access state of the publication, which also can be added to the imported metadata.

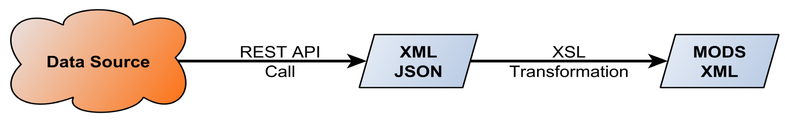
Enrich & Merge Algorithm

```
method enrich( publicationData ) {
  collectedData := {}
  publicationIDs := getIdentifiersFrom( publicationData )
  while( publicationIDs.containsNewIdentifiers() ) {
    forEach( dataSource ) {
      if(    dataSource.isNotQueriedYet()
          && dataSource.acceptsOneOf( publicationIDs ) ) {
        resolvedData := dataSource.resolvePublicationData( publicationIDs )
        if( resolvedData not null ) {
            collectedData.add( resolvedData )
            publicationIDs.addNewIdentifiersFrom( collectedData )
        }
      }
    }
  }
  publicationData.mergeWith( collectedData )
  // Recursively proceed to host, e.g. article -in- proceedings -in- series
  enrich( publicationData.getHostingPublicationOrSeries() )
}
```

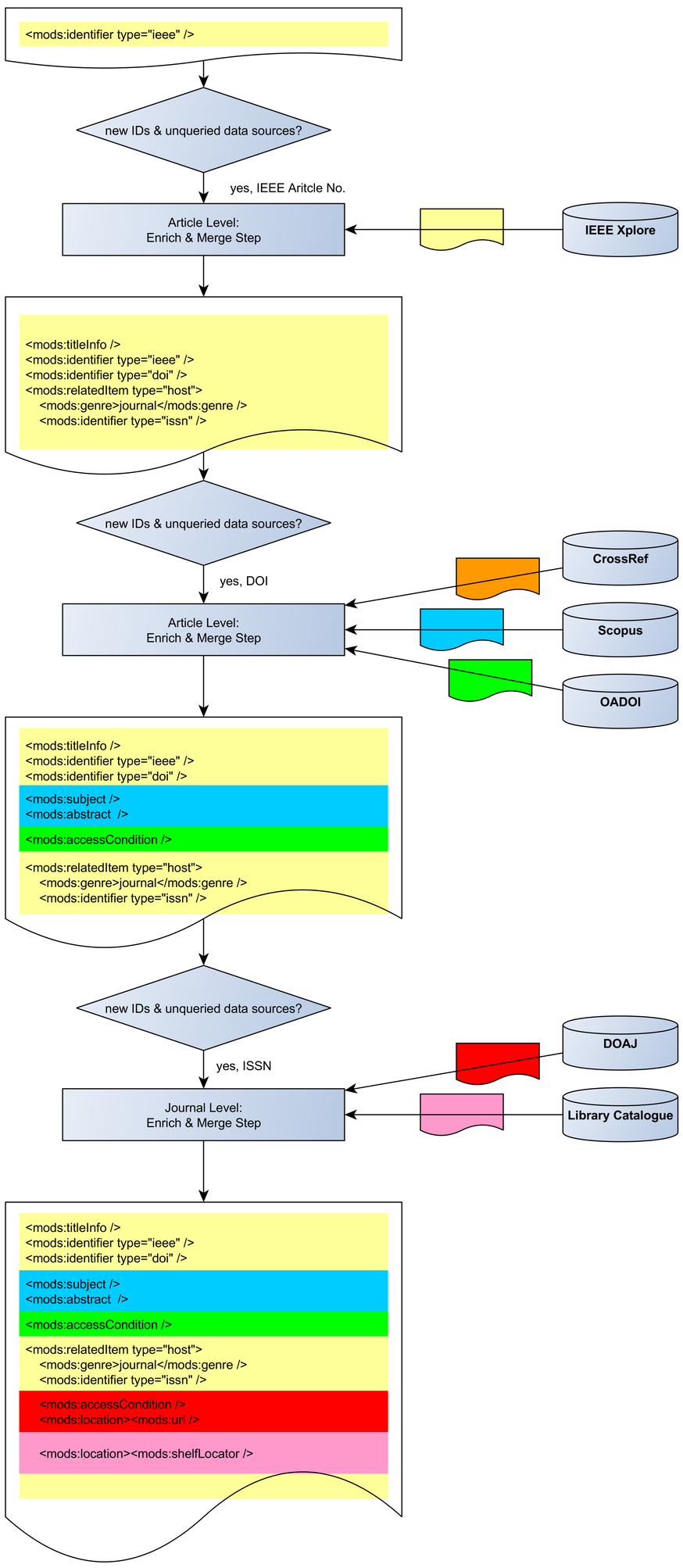
Configuration



Typical resolving method: Call data source, transform result into unified MODS XML Schema



Sample Process



Merging MODS representations

```
<mods:mods>
  <mods:identifier type="pmid">173456889</mods:identifier>
  <mods:identifier type="doi">10.12345/pub/987</mods:identifier>
  <mods:titleInfo>
    <mods:title>Merging data: heuristics, risks and limitations</mods:title>
  </mods:titleInfo>
  <mods:name type="personal">
    <mods:namePart type="family">Lützenkirchen</mods:namePart>
    <mods:namePart type="given">Frank</mods:namePart>
  </mods:name>
</mods:mods>
```

+

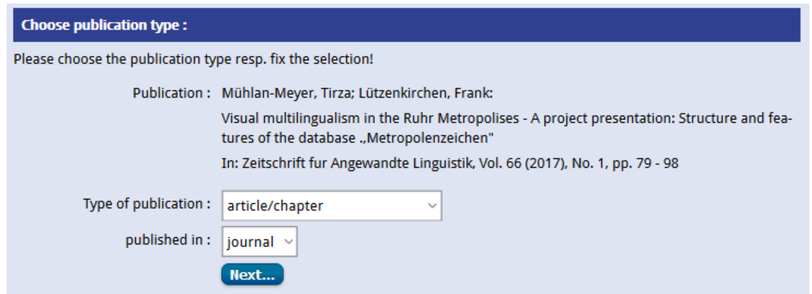
```
<mods:mods>
  <mods:identifier type="doi">10.12345/pub/987</mods:identifier>
  <mods:titleInfo xml:lang="en">
    <mods:title>Merging data</mods:title>
    <mods:subTitle>heuristics, risks and limitations</mods:subTitle>
  </mods:titleInfo>
  <mods:name type="personal">
    <mods:displayForm>Luetzenkirchen, F.</mods:displayForm>
    <mods:nameIdentifier type=”orcid”>0000-0001-5065-6970</mods:nameIdentifier>
  </mods:name>
</mods:mods>
```

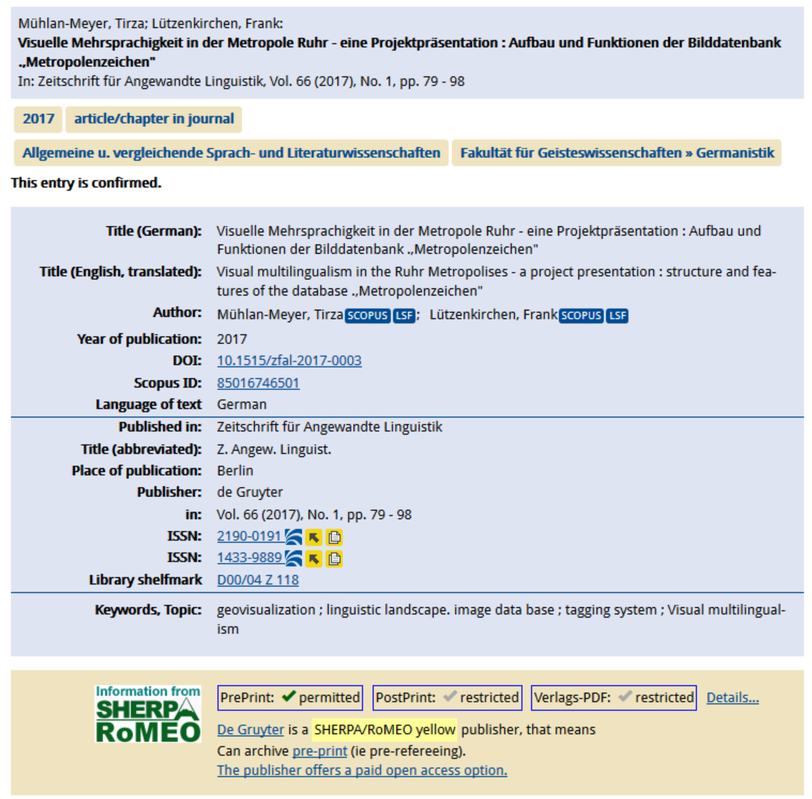
=

```
<mods:mods>
  <mods:identifier type="pmid">173456889</mods:identifier>
  <mods:identifier type="doi">10.12345/pub/987</mods:identifier>
  <mods:titleInfo xml:lang="en">
    <mods:title>Merging data</mods:title>
    <mods:subTitle>heuristics, risks and limitations</mods:subTitle>
  </mods:titleInfo>
  <mods:name type="personal">
    <mods:namePart type="family">Lützenkirchen</mods:namePart>
    <mods:namePart type="given">Frank</mods:namePart>
    <mods:nameIdentifier type=”orcid”>0000-0001-5065-6970</mods:nameIdentifier>
  </mods:name>
</mods:mods>
```

Application & Use Cases

- Use in repository, CRIS, bibliography
- Import publication metadata via DOI
- Improve publication lists, e.g. from ORCID
- Enrich existing publication data





Lessons learned

- Ease of import encourages deposit of publications
- Merging uses heuristics and is not perfect
- but it highly reduces manual editing effort
- Framework simplifies adding more data sources
- Stable, (freely) available APIs required
- Providers should use more standard metadata schemes