

**Create IIF media and metadata  
repositories without touching code.**

**Martin Rode**

CTO @ Programmfabrik, Berlin

# Features



- Define any data model with no coding required
- Inject data from many sources
- Upload millions of files
- Pools, Collections, Users, Groups
- **fy\_l\_r** runs on all operating systems (written in Go)
- Extensible through Plugins, including custom data types
- easydb API compatible

# Framework overview



Web	iOS / Android	fy_l_rctl
<b>API</b> <ul style="list-style-type: none"><li>• Restful JSON</li><li>• IIIF Image API 3.0</li><li>• IIIF Presentation API 3.0</li><li>• OAI/PMH</li></ul>		<b>Executor</b> <ul style="list-style-type: none"><li>• Runs any binary</li><li>• Preview Generation</li><li>• Plugin execution</li><li>• Exports</li></ul>
<b>fy_l_r</b> <ul style="list-style-type: none"><li>• Manage files &amp; metadata</li><li>• Flexible data models</li><li>• Hierarchical types support</li><li>• Users, groups, rights</li><li>• Plugin support</li><li>• Email support</li></ul>		
<b>Database</b> PostgreSQL, Sqlite	<b>S3 Storage</b> AWS / MinIO / Ceph	<b>Indexer</b> Elasticsearch

## Recipe

- Declarative .yml format
- Pre-defined for most formats
- Custom extensions possible

## Parameters

- Defined by the user in fy\_l\_r
- Many data types  
(string, list, int, choices)
- Recipes can react upon parameters

## Executes Binary

- Use well-known tools like convert, ffmpeg, dcraw
- Bring your own tools into the executor tool chain
- Manages state of file version production in **fy\_l\_r**
- Can run synchronous tasks (like IIF tile production)
- Built-in cache system

# IIIF Considerations

- Parameter & Configuration driven generation
  - Image 3.0 API based on **fy<sub>l</sub>r** base model
  - Flexible backend tooling for actual tile generation
  - Presentation 3.0 API data model & collection based (WIP)
- 
- Discussion 1: Other IIIF APIs?
  - Discussion 2: Viewer vs. App integration

Thanks, and please

[join@fy\\_l\\_r.io](mailto:join@fy_l_r.io)