

## **Putting Linked Data to Work: Semantic Data "Brokering" in Practice**

**HELSINGIN YLIOPISTO  
HELSINGFORS UNIVERSITET  
UNIVERSITY OF HELSINKI**

**João da Silva  
Stefan Negru**



# OUTLINE

- ATTX Project Overview
- ATTX Semantic Broker "another" ETL tool
  - Use cases
  - Provenance tracking
- Demo
- Automation FTW ... right?
  - CI/CD Automation
  - Deployment
  - Technology stack



# KEYWORDS

- Easy Development
- Easy Deployment
- Easy Publishing and reuse of data





# SEMANTIC BROKER

## USE CASE ORIENTED

### UC1: Infrastructures and Publications

In house proof of concept use case.

### UC2: Parallel Publication Dashboard

In cooperation with the University of Jyväskylä.

### UC3: Metax

Semantic Models mapping for CSC - IT Center for Science.

### UC4: MILDRED

Developing part of the University of Helsinki Research Infrastructure.



# SEMANTIC BROKER

## JUST ANOTHER ETL TOOL?

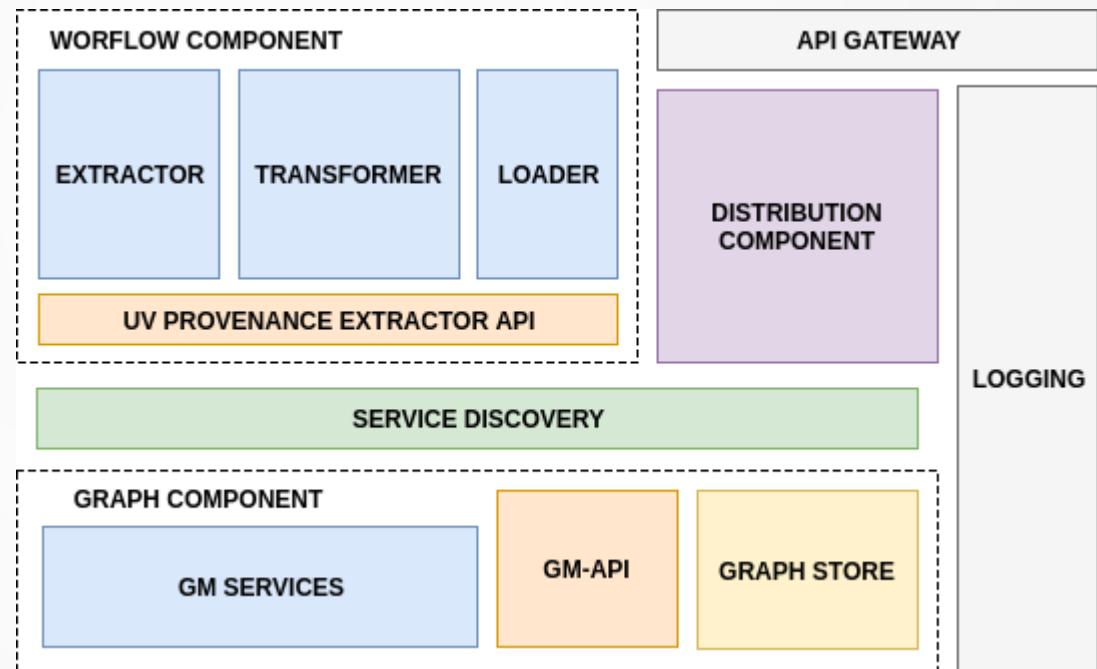
- Entity Linking (NER, reconciliation etc.)
- **Reasoning capabilities**
- **Data Validation (partly including Data Quality Standards)**
- **Statistics and Insights about the Data**
- Data Transformations
- Data Publishing (REST APIs, **HDT files etc.**)
- **Provenance Tracking**



# SEMANTIC BROKER

## ATTX SEMANTIC BROKER OVERVIEW

1. Use case oriented
2. Provenance tracking
3. Advance capabilities of enriching and generating new data
4. Flexible graph based internal data model
5. Containerised components: aiming for Micro-services Architecture





# DEMO

## DEPLOYMENT & USAGE

- Docker Swarm as platform for ATTX components: containerise everything (and run it everywhere);
- Infrastructure-as-Code approach (yml file with service stack definition);
- Automatic deployment via YML in own PC or Cloud (OpenStack)



# DEMO

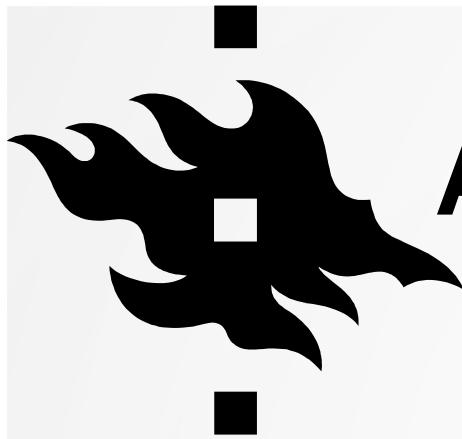
## DEPLOYMENT & USAGE

- We'll start the demo by creating a three-node Docker Swarm on our own PC (but you could do it in your own IaaS or VMs);
- Next, we'll deploy the ATTX Stack provisioned with YML;
- Then, we'll demonstrate a sample workflow for the ATTX LOD solution :-)

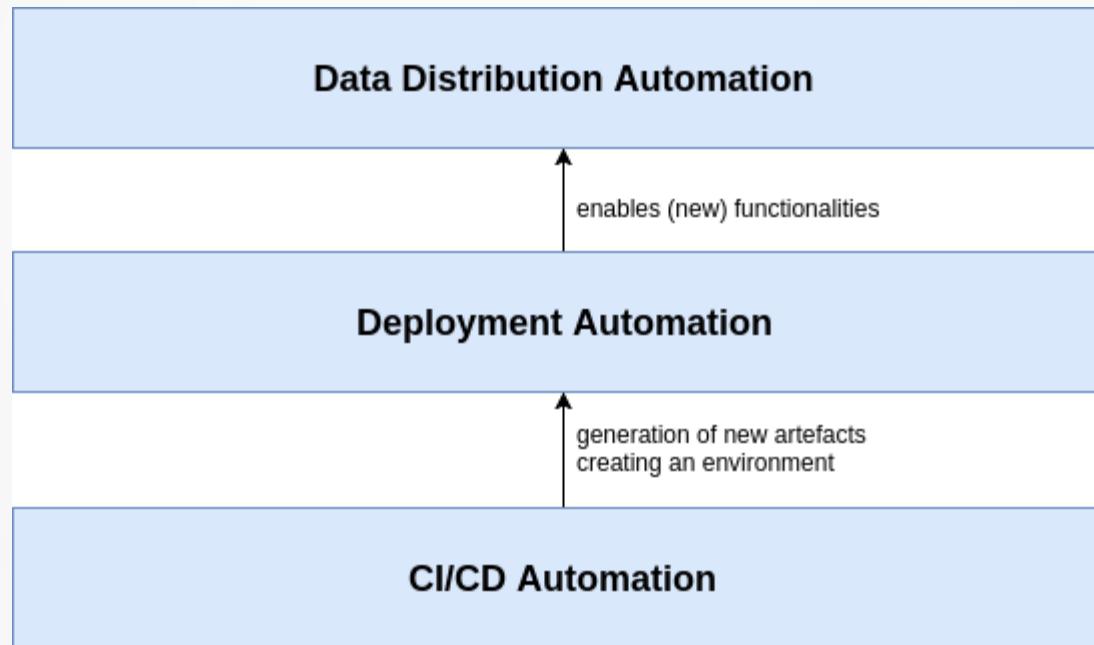


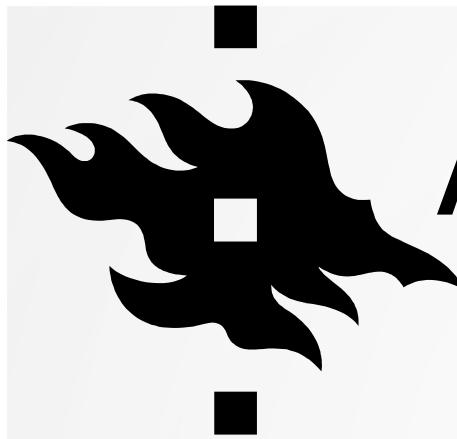
AUTOMATION FTW ...  
RIGHT?

<https://www.flickr.com/photos/spacex/27294261525/>



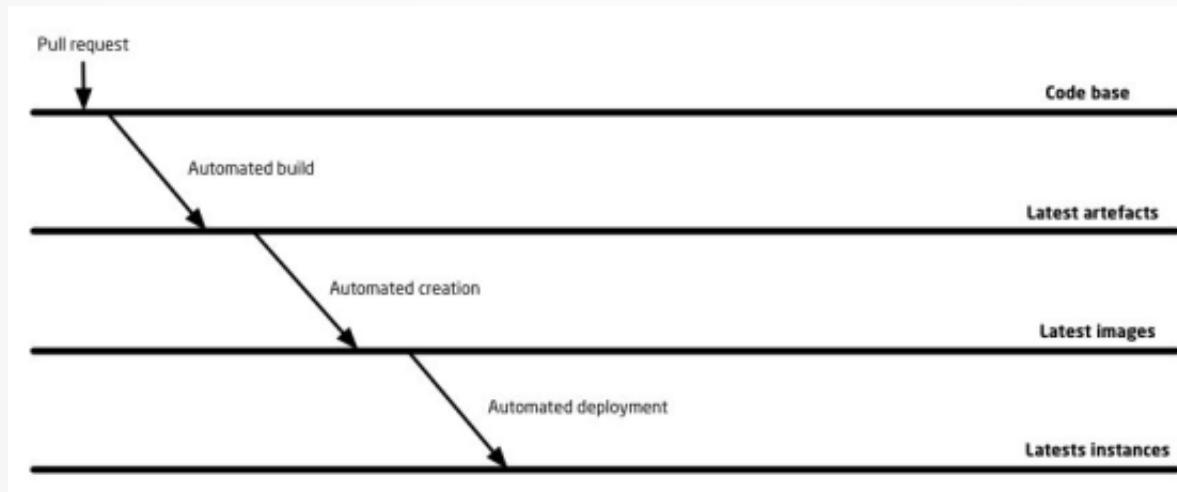
# AUTOMATION LAYERS

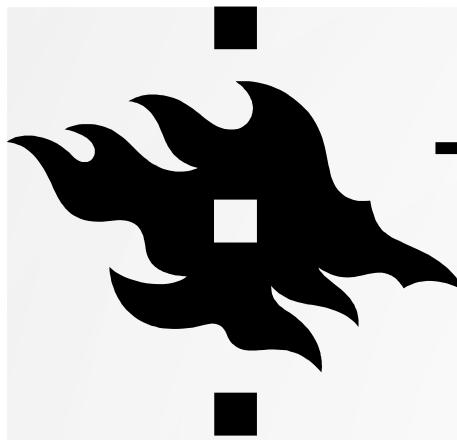




# AUTOMATION LAYERS

## CI/CD WORKFLOW

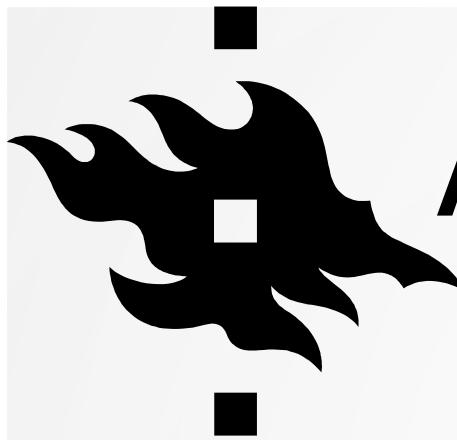




# TECHNOLOGY STACK

## CI/CD ENVIRONMENT

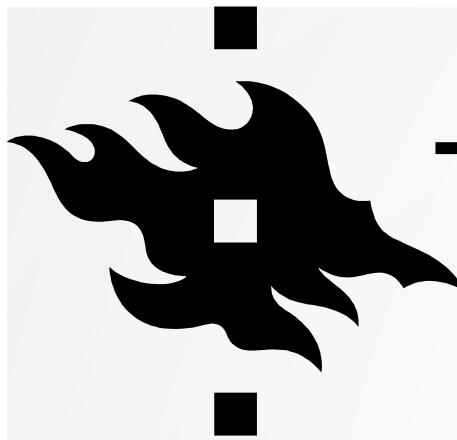
- Gradle build tool - <https://gradle.org/> with {py}Gradle
- Github - <https://github.com/>
- Jenkins - <https://jenkins.io/>
- Docker - <https://www.docker.com/>
- Archiva - <https://archiva.apache.org/>



# AUTOMATION LAYERS

## DEPLOYMENT

- Deploying updated components/services to each instance of the Semantic Broker
- Provisioning and deploying ATTX stack on own PC or Cloud (OpenStack)
- Rolling Updates with Docker Swarm



# TECHNOLOGY STACK

## ATTX SEMANTIC BROKER

- ElasticSearch - <https://www.elastic.co/>
- UnifiedViews - <https://unifiedviews.eu/>
- Jena + Fuseki - <https://jena.apache.org>

Next version:

- Consul - <https://www.consul.io/> or  
Kontena (<https://www.kontena.io/>) 
- RML - <http://rml.io/>
- SHACL - <https://www.w3.org/TR/shacl/> & ShEx - <http://shex.io/>



# THANK YOU

"They say it could not be done,  
but we did it anyway, and it (kinda) worked"

*Joonas Kesäniemi - @lunch 2017*

- <https://www.helsinki.fi/en/projects/attx-2016>
- <https://attx-project.github.io/>
- <https://github.com/ATTX-project/elag2017-demo>