BioHackathon 2018 - Paris
http://bh2018paris.info/

# Building a semantic search engine for biology publications using event stream processing

Proponent: Mustafa Anil Tuncel, Kim Phillip Jablonski, Ivan Topolsky
- ETH Zürich, Department of Biosystems Science and Engineering

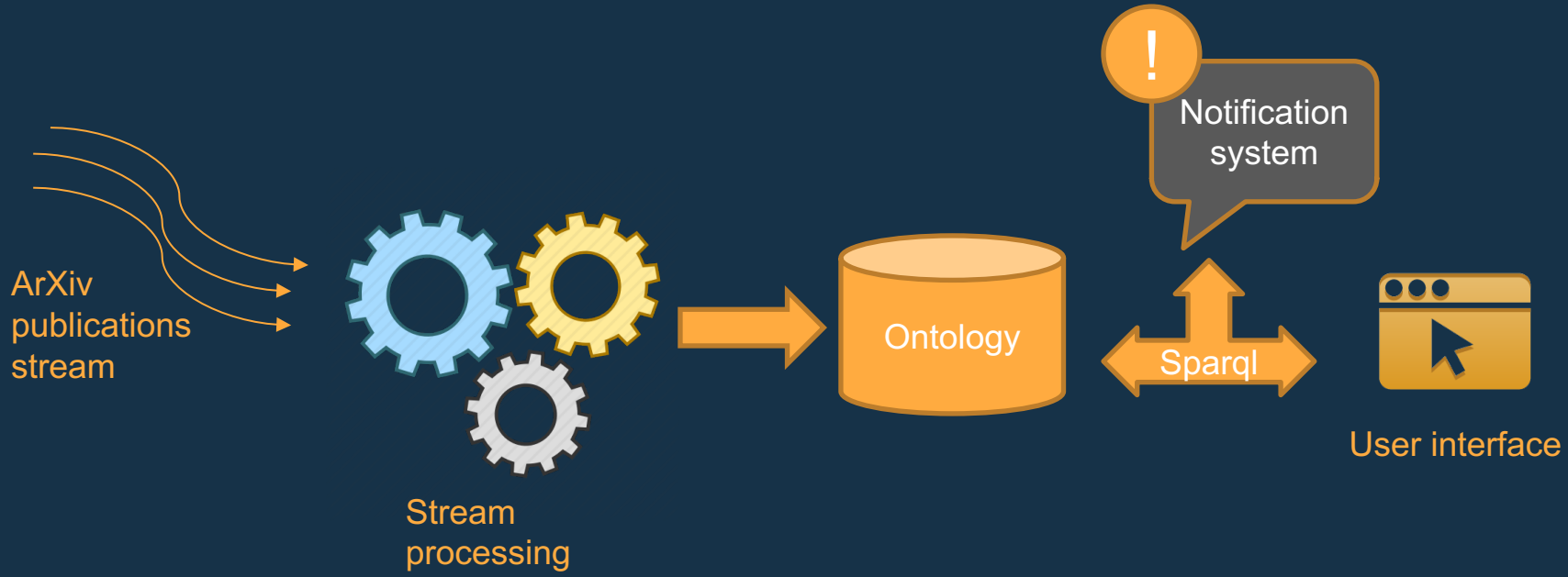
Project link:
- https://github.com/elixir-europe/BioHackathon

# Background information

# Background

- Keeping up with the constant flow of new articles being published in various journals is a challenge
- Using event stream processing, we aim at updating the biomedical publications ontology in real time

# System overview



# Goals of the hacking project

# Goal and expected outcome

- **General goal of the hacking project**
  - The goal is to create an ontology for biomedical publications and to update it in real-time using event stream processes
- **Expected results** at the end of the hackathon
  - A service that monitors the ArXiv/BioRxiv twitter feeds and continuously parses relevant metainformation into easily machine-readable BioSchemas.
  - An interface to allow users to perform SPARQL queries on the continuously updated publications ontology
  - [Optional] A notification system that informs the user on the most relevant subset of topics within the stream of publications

# Post-biohackathon perspectives

- Expanding the knowledge base
- Integration with other platforms such as PubMED
- Integrating our ontology with outcome of the BioTea to BioSchemas project

# Hack organisation

# Organisation of the hacking project

- Duration: 5 hacking days

- **Call for additional expertise from biohackathon attendees**
  - Experience in web-technologies such as nodejs/react/vue/css/..

# Steps and tasks

- Retrieving stream data from ArXiv/BioarXiv feeds using twitter stream api
- Retrieving the pdf/latex of the publication from ArXiv/BioarXiv/PubMed
- Extracting information from the latex/pdf files
- Creating the ontology
- Updating the ontology whenever a new paper is published
- Starting the sparql server (jena fuseki, python flask/rdflib, etc.)
- User interface
  - Querying page UI
  - Results in both text and graph

# Contact and links

- Contact (s)
  - Mustafa Anil Tuncel (mtuncel@ethz.ch)
  - Kim Philipp Jablonski (kim.jablonski@bsse.ethz.ch)
  - Ivan Topolsky (ivan.topolsky@bsse.ethz.ch)

- Links related to the project
  - Event stream processing: https://en.wikipedia.org/wiki/Event_stream_processing
  - RDFLib: https://github.com/RDFLib/rdflib
  - Twitter stream api: https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data.html
  - Apache Jena Fuseki Sparql server: https://jena.apache.org/documentation/fuseki2/