

jahia Day

Integrating jExperience with Machine Learning

Serge Huber, Jahia CTO & Co-founder



jahia Day

jCustomer

The Jahia
Subscription for
Apache Unomi

Get support
from the source



Subscription for Apache Unomi :

Subscription scope:

- Unlimited and prioritized bug fixes
- HotFix version (only for customers)
- Official supported docker image for cloud deployment
- Included hours of Technical Assistance with SLA at all stage of the project
- Apache Unomi Plug-in support
- (coming soon) 200+ connectors bus option
- Unlimited development environment for a given project

Additional services offering:

- Consulting: plug in development support, application development support, environment deployment and test assistance
- Training

Collaborative Development:

- New generic capabilities financed by the end client and developed by Jahia's R&D teams
- Validated and then delivered in the next version of the software published by Jahia
- Covered by the customer subscription at no additional cost

Apache Unomi

(Pronounced “You Know Me”)



2019-Top Level Project graduation

jahia Day



Apache Unomi is now a Top Level Project

Jahia proudly celebrates Apache Unomi's graduation as a Top Level Project (TLP) at the Apache Software Foundation and launches a new enterprise subscription for this powerful Open Source Customer Data Platform with privacy by design, initiated by the Jahia team in 2015.

[READ MORE](#)

What is Apache Unomi

jahia Day

- It's a **CDP** (Customer Data Platform)
- What's a CDP ?



Customer Data Platform Definition

jahia Day

A Customer Data Platform is
packaged software that creates a persistent,
unified customer database
that is accessible to other systems.

Jahia added:
and allows to protect the data privacy
of your customers



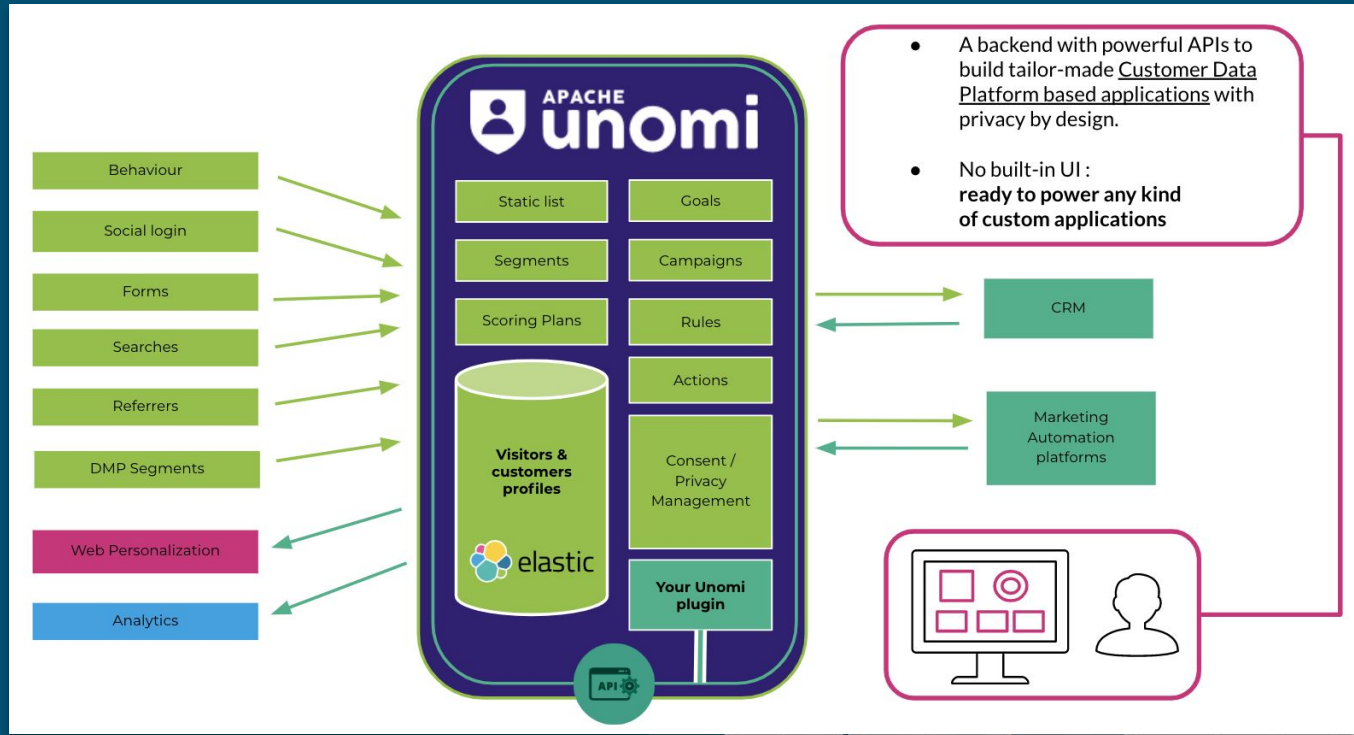
CUSTOMER
DATA PLATFORM
INSTITUTE

Why Customer Data Is Important ?

jahia Day



jCustomer Architecture



What can you do with jCustomer / Apache Unomi?

jahia Day

- Adding personalization/optimization to an existing website or PWA
- Adding personalization/optimization to a native mobile application
- Serving personalized ads
- Integrating instant messaging with personalization (Slack + CDP)

jahia Day

AI?



Artificially Intelligent

jahia Day

Phone



Artificially Intelligent

jahia Day

Watch



Artificially Intelligent

jahia Day

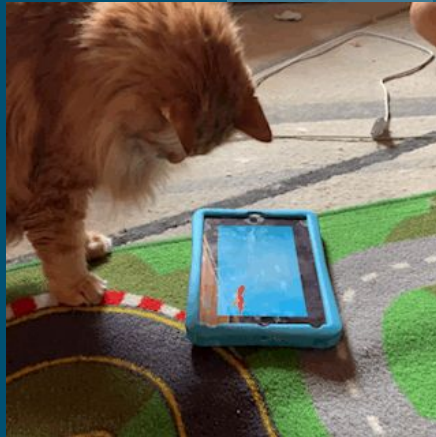
Car



Artificially Intelligent

jahia Day

Cat



Artificial Intelligence

Usually implies Machine Learning



Why AI with a CDP ?

- Help marketers target visitors
- Add some automation into personalization
- Use visitor behavior, form data, social login, CRM data to train AI
- Can leverage CDP semi-structured data to build and train new models at any time
- Allows for real-time usage of AI results

Why Open Source AI with an Open Source CDP ?

jahia Day

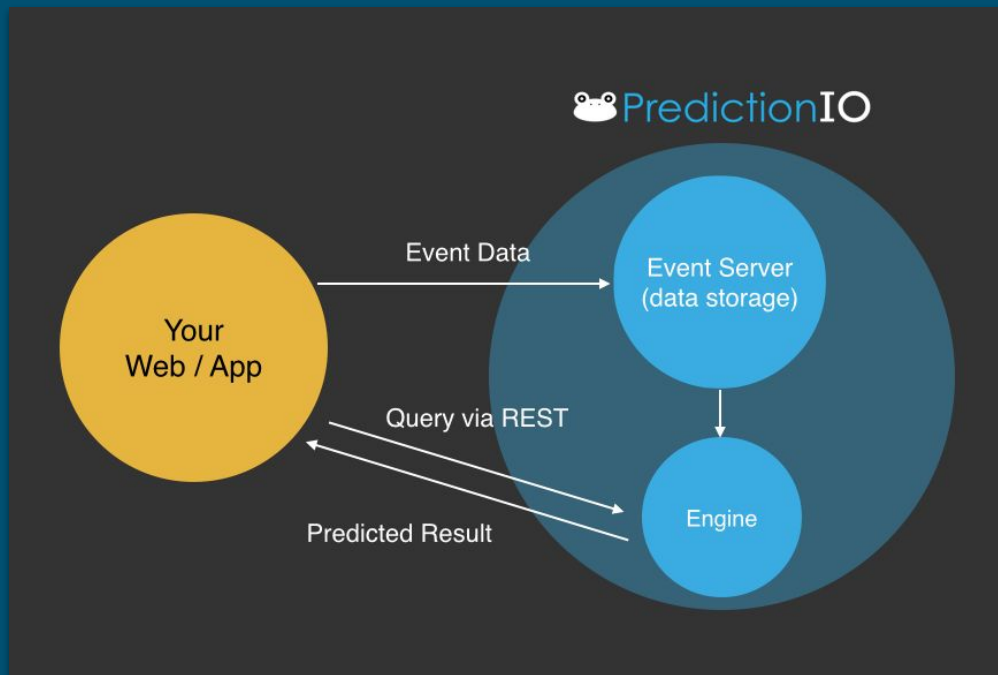
- Only way to be serious about privacy
- GDPR-compliance
- Everything can be under one roof (or one data center, or even a laptop!)
- No complicated (& costly) dependencies on third parties

Why Apache PredictionIO ?

- Really simple yet powerful REST API
- Offers many ready-to-use templates
- Complete machine-learning stack
- Uses great open-source stacks such as Apache Spark, Spark MLlib, ElasticSearch, PostgreSQL, HBase
- Because it has real-time querying

PredictionIO architecture

jahia Day



Integration - How it works

1. Visitors interact with webpage or PWA
2. Unomi is called from Javascript Web Tracker
3. Events are processed -> rules are executed
4. Rules have actions that query the AI engine
5. Results get sent back to in-page Javascript -> personalization

All in real-time!



AI Lead scoring

- Predict probability a visitor will “convert” (buy a product, subscribe, contact)
- Based on visitor behavior on site
- Usually trained using real visitor data (for best results)
- Requires little data from the visitor (page URL, referrer URL, browser)

Already setup

- PredictionIO Event server
- Lead scoring engine
- ElasticSearch
- jCustomer / Apache Unomi
- Jahia with jExperience

jahia Day

Live demo !



Training Data

- Currently generated using Python script
- Possible to extract from Elasticsearch data collected by Apache Unomi
- Re-inject into template & train using updated data & redeploy

Integrating with other AI engines

- Build custom actions to send/retrieve AI engines output
- Setup calls to the custom actions with custom rules
- Using their own models, monitoring, training, etc...
- Nothing is specific to Apache PredictionIO in Apache Unomi
- For example (decide.ai, Amazon, Google, TensorFlow, ...)

Source code

jahia Day

<https://github.com/Jahia/unomi-predictionio-plugin/>

THANK YOU

jahia Day

SERGE HUBER

@sergehuber

shuber@jahia.com



Setting up Apache PredictionIO

- Used Docker container generation

```
git clone https://github.com/apache/predictionio.git
cd predictionio
cd docker
docker build -t predictionio/pio pio
docker-compose -f docker-compose.yml \
  -f pgsql/docker-compose.base.yml \
  -f pgsql/docker-compose.meta.yml \
  -f pgsql/docker-compose.event.yml \
  -f pgsql/docker-compose.model.yml \
  up
```

Setting up Lead scoring engine

```
git clone https://github.com/sergehuber/template-scala-parallel-leadscoreing.git
MyLeadScoring
cd MyLeadScoring
export PATH=**REPLACE_WITH_PATH_TO_PREDICTIONIO_SRC**/docker/bin/:$PATH
pio-docker build --verbose
pio-docker app list
ACCESS_KEY=cde1LgZqZxj7CI_2hDM_vy-Q3fhLDx1TQKao_UHe9DgcFLkSVm9Yfq_3ve8BTgz1
python data/import_eventserver.py --access_key $ACCESS_KEY
pio-docker train
pio-docker deploy
```

Setting up ElasticSearch

1. `curl -O https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.6.3.tar.gz`
2. `tar zxvf elasticsearch-5.6.3.tar.gz`
3. `cd elasticsearch-5.6.3`
4. change `cluster.name` to `cluster.name: contextElasticSearch` in `config/elasticsearch.yml` file
5. `bin/elasticsearch`

Settings up Apache Unomi

1. `git clone https://github.com/apache/unomi.git`
2. `cd unomi`
3. `./buildAndRunNoTests.sh`
4. `unomi:start` (in Karaf shell)