



Jaxy

Déployez vos Web services REST Avec 0 ligne de code

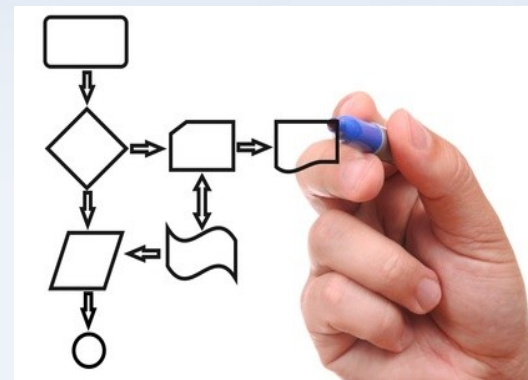


R. Yahiaoui

Agenda



Technique 



Conception 



Démo 



Code 



Petit retour d'expérience

Jaxy

- Pourquoi
- Quels usages
- Comment
 - Serveur d'App
 - War vs Jar
 - Runtime Compiler

REST

- Définition
- Architecture
- Desing API
- Critères qualité

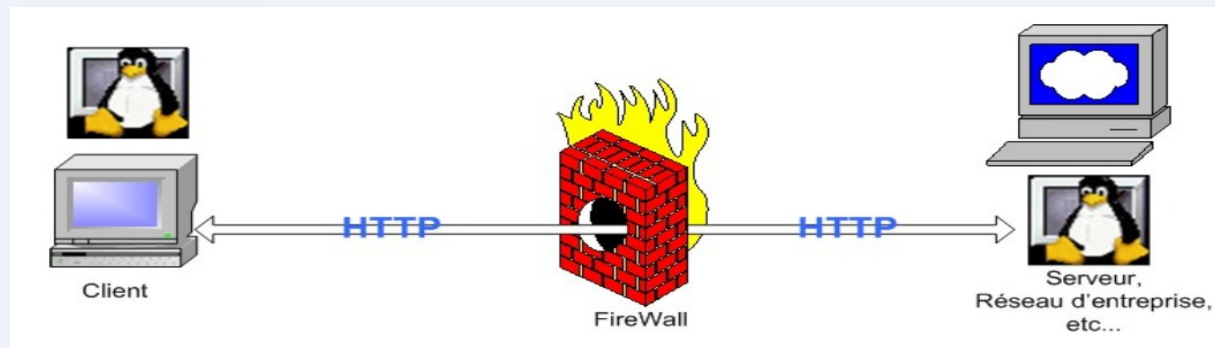
Démo



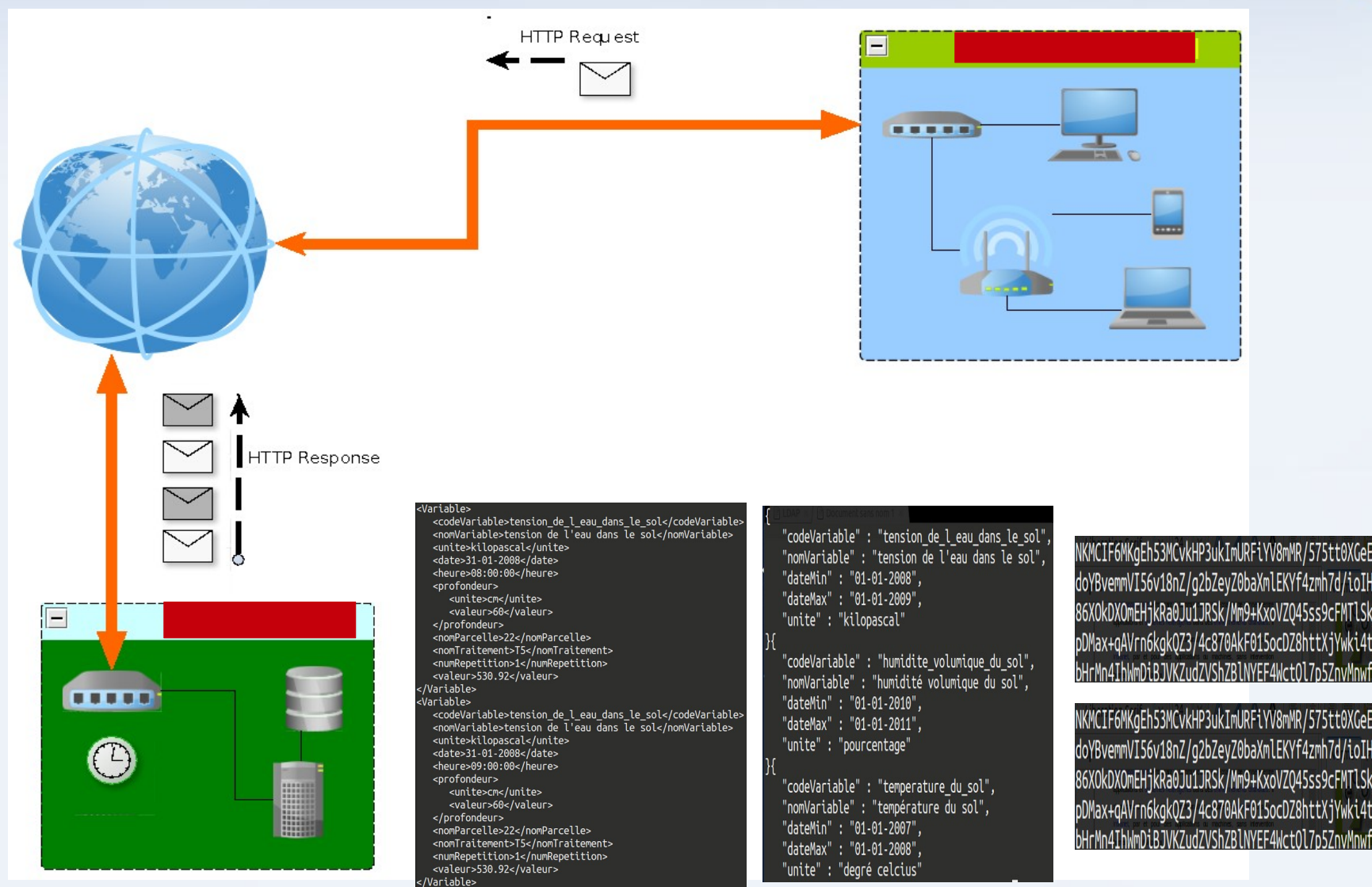


Web-Service

Un service web est un **programme informatique** de la famille des **technologies web** permettant la communication et l'échange de données entre applications et **systèmes hétérogènes** dans des **environnements distribués**. Il s'agit donc d'un ensemble de fonctionnalités exposées sur **internet** ou sur un **intranet**, par et pour des applications ou machines, sans intervention humaine... (Wikipedia)



Retour d'expérience (2014)





Ouvrir accès à la base de données ?

Oui mais ...

- SQL, peut être Complexe & Dangereux ?!
- Mise à jour des Schémas
- Forte charge sur la BDD vs caching
- Scalabilité
- Media-Type (DB- Relationnelles)
- Nouvelles architectures web
- Friendly DB access



Quelle architecture ?

SOAP (SOA)	REST (ROA)
<p>Standard (Protocole générique).</p> <p>Verbeux (XML, SEI)</p> <p>Contrats formels</p> <p>Fort Couplage Client – Serveur</p> <p>Opérations avec ou sans état (stateless)</p> <p>Couche transport générique : Http , smtp, jms...</p>	<p>Opérations sans état (stateless)</p> <p>Protocole HTTP pour le transport</p> <p>Accessible, Facile à mettre en œuvre</p> <p>Utilise les verbes standards du protocole HTTP pour faire du CRUD (GET PUT POST DELETE)</p>



Critères de Qualité



Flexibilité



Robustesse



Performance



Réutilisabilité



Sécurité

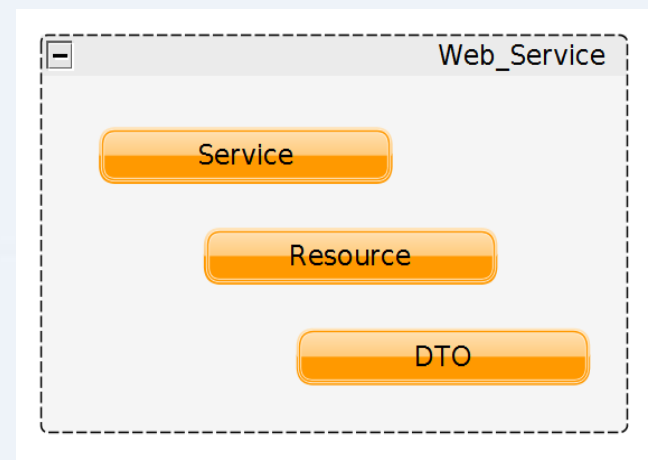
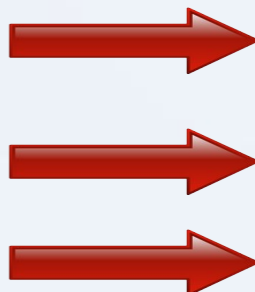


Documentation



Flexibilité

Un web service est une « *Fonction* » capable d' **intercepter** des requêtes HTTP, d'en extraire la demande, de l'**exécuter**, puis **formater** le résultat pour enfin le renvoyer au client.



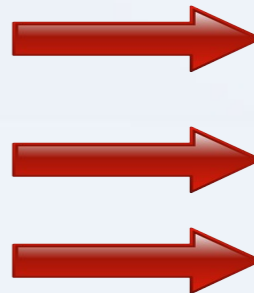
Package service web



Performances (Temps de réponse)

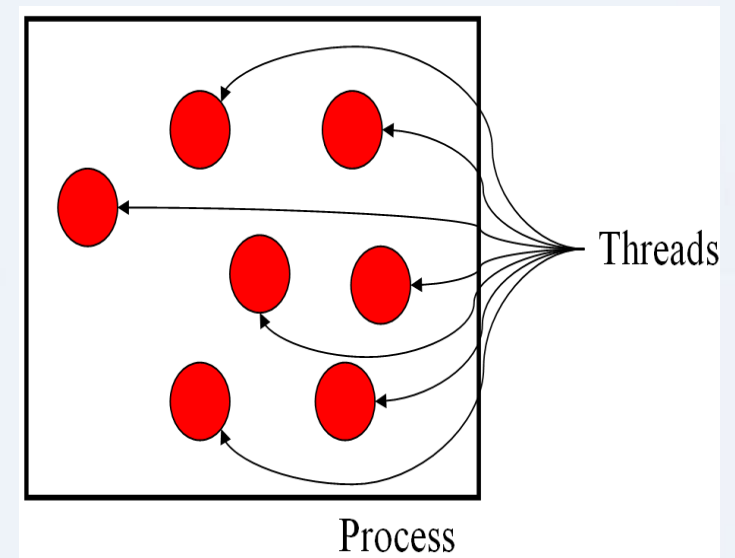
Une thread (appelée aussi processus léger) est un fil d'instructions (un chemin d'exécution) à l'intérieur d'un processus.

Contrairement aux processus, les threads d'un même processus partagent le même espace d'adressage et le même environnement..



~~Dénormalisation DB~~

VS



Débit d'extraction SANS // ~ 0,5 Mo/s

Débit d'extraction AVEC // ~ 9 Mo/s

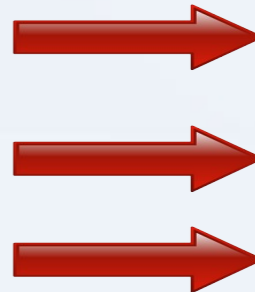


Robustesse

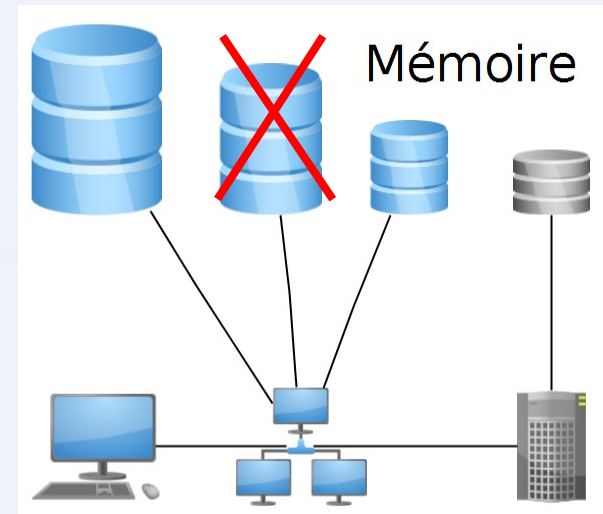
Contrôler la charge mémoire
(empreinte mémoire) afin d'éviter
toute défaillance du système.

Pagination des requêtes SQL

LIMIT – OFFSET



Volume de données >> à la mémoire

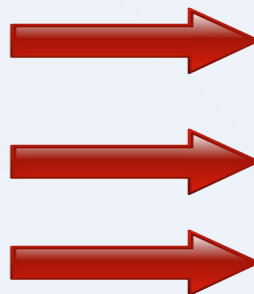




Réutilisabilité

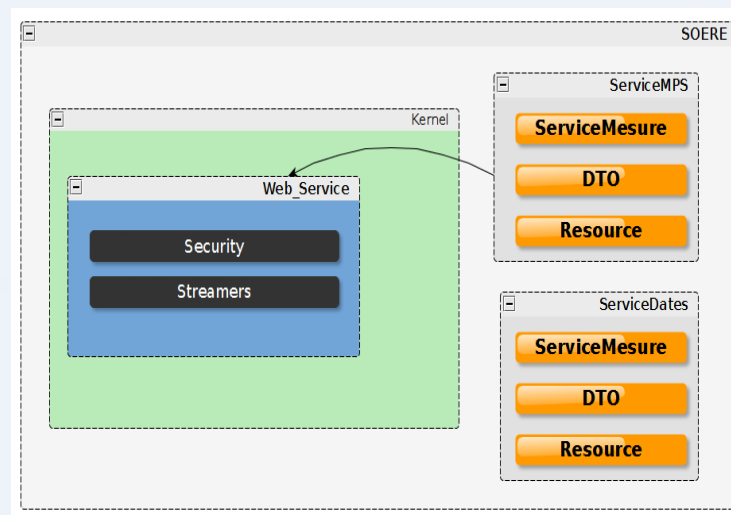
Intégration du web service (core)
dans le Kernel du S.I .

⇒ Héritage implicite des
fonctionnalités du web service.



Libre à chaque S.I d'implémenter ses
propres web services
(dans la couche plugin)

Write once, run anywhere



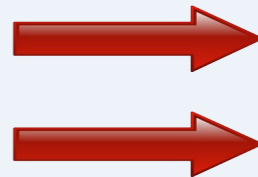


Documentation

Spécification pour documenter les API Rest
(Via annotation)



Problème : Pas de prise en charge des sub-resource locator



Propre API qui repose sur l'introspection du Code (sans annotation)

Info : Nouvelle API Swagger : Sans annotation ni configuration particulière

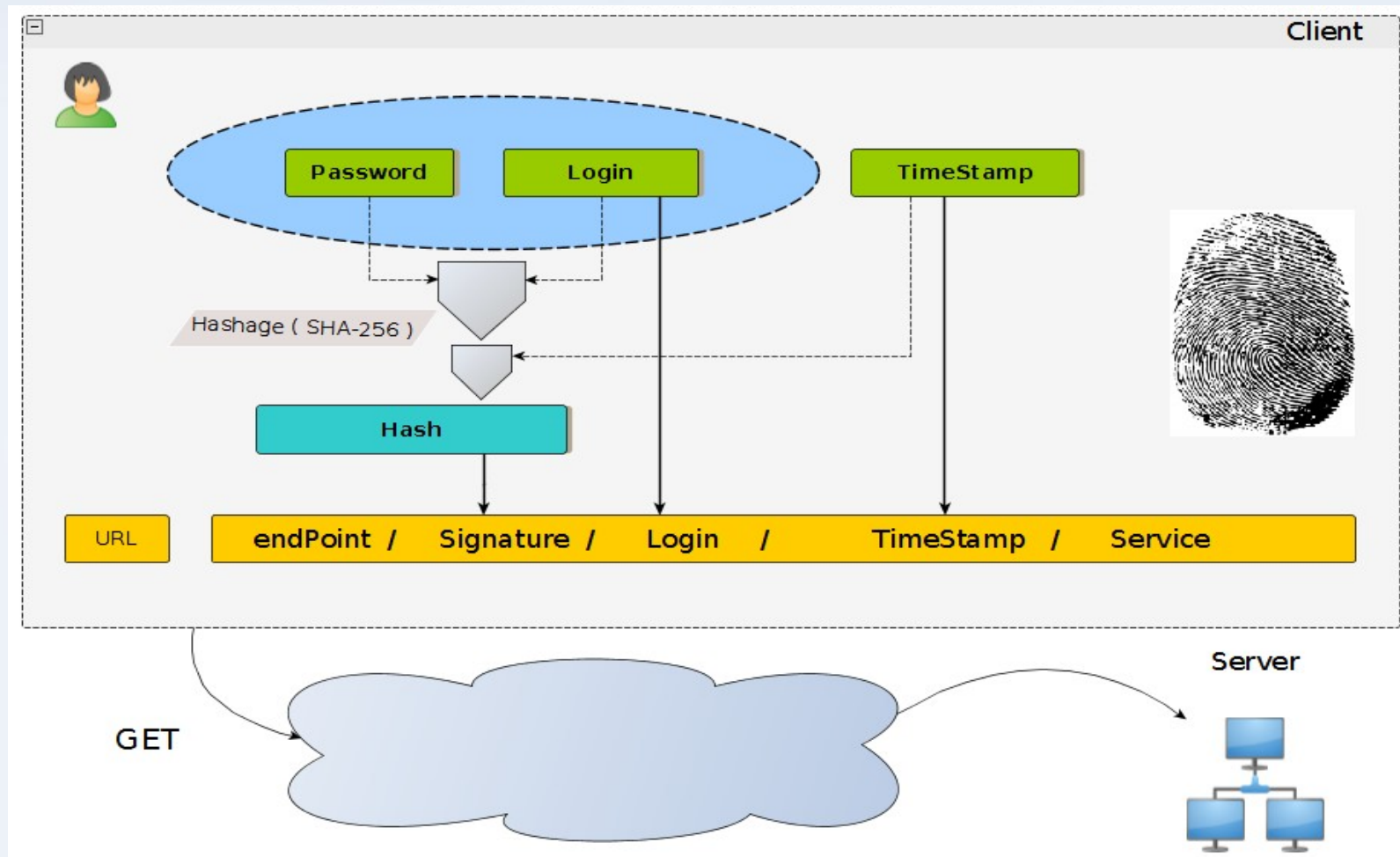
<https://github.com/sdaschner/jaxrs-analyzer>



Sécurité (1/2)

~~SSL~~

Client

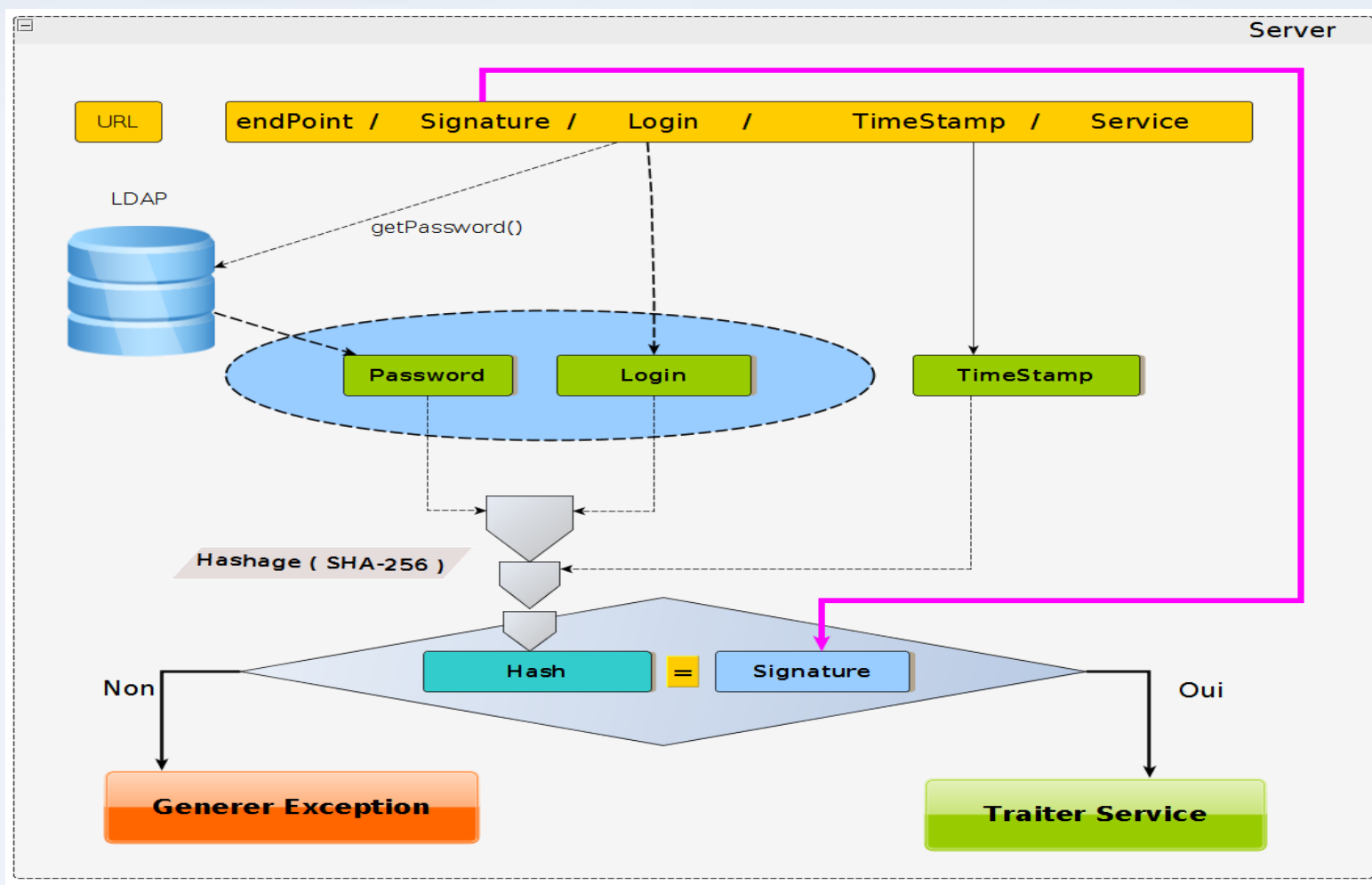




Sécurité (2/2)

~~SSL~~

Serveur





Sécurité (Alternative)

Keycloak

JWT

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJhdXRoLWJhY2t1bmQiLCJzdiI0Ijoc2FibG9ubmllcmUiLCJhdWQiOiJ3ZWItZnJvbnRlbmQiLCJleHAiOjE0NTg2Mzg2MTgsIm5iZiI6bnVsbCwiaWF0IjoxNDU4NjM1MDE4LCJqdGkiOiI4NWEzY2I0Yy01NzIwLTRkYmEtYWU5NC05MzFkNjA5MzdjNDciLCJ1YW1lIjoiaSHViZXJ0IFNBQkxPTk5Jw4hSRSI0InBlcm1pc3Npb25zIjpbIkVESVRPUiIsIlJFVklFV0VSIl19.fwxQ6GrPRyEi8wKh1BpYrWPuiF0pPswSahgM1WFWIxo
```




Sécurité (Alternative)

JWT

```
> base64URLdecode(encodedHeader);  
'{"alg":"HS256","typ":"JWT"}'
```

```
> base64URLdecode(encodedPayload);  
'{"iss":"auth-backend","sub":"hsablonniere","aud":"web-frontend",  
"exp":1458638618,"nbf":null,"iat":1458635018,"jti":  
"85a3cb4c-5720-4dba-ae94-931d60937c47","name":"Hubert  
SABLONNIÈRE","permissions":["EDITOR","REVIEWER"]}'
```

```
> signature  
fwxQ6GrPRyEi8wKh1BpYrWPuiF0pPswSahgM1WFWIxo
```




Code.. from scratch !

Happy Dev - Happy users





Jaxy, Use Cases :

- Approche « **Database** as 'secured' web service »
- Déployer un web service avec « **zéro** » compétence informatique (ou presque)
- Faire **interagir** 2 (ou plusieurs) S.I
- Peut être utilisé comme brique (applicative) dans une **architecture micro services**



Jaxy (v_1.0 : 2017 / v_2.2 : 2020)

→ Open source Project



**ZÉRO LIGNE DE
CODE !**

→ Liens :

<https://github.com/rac021/jaxy>

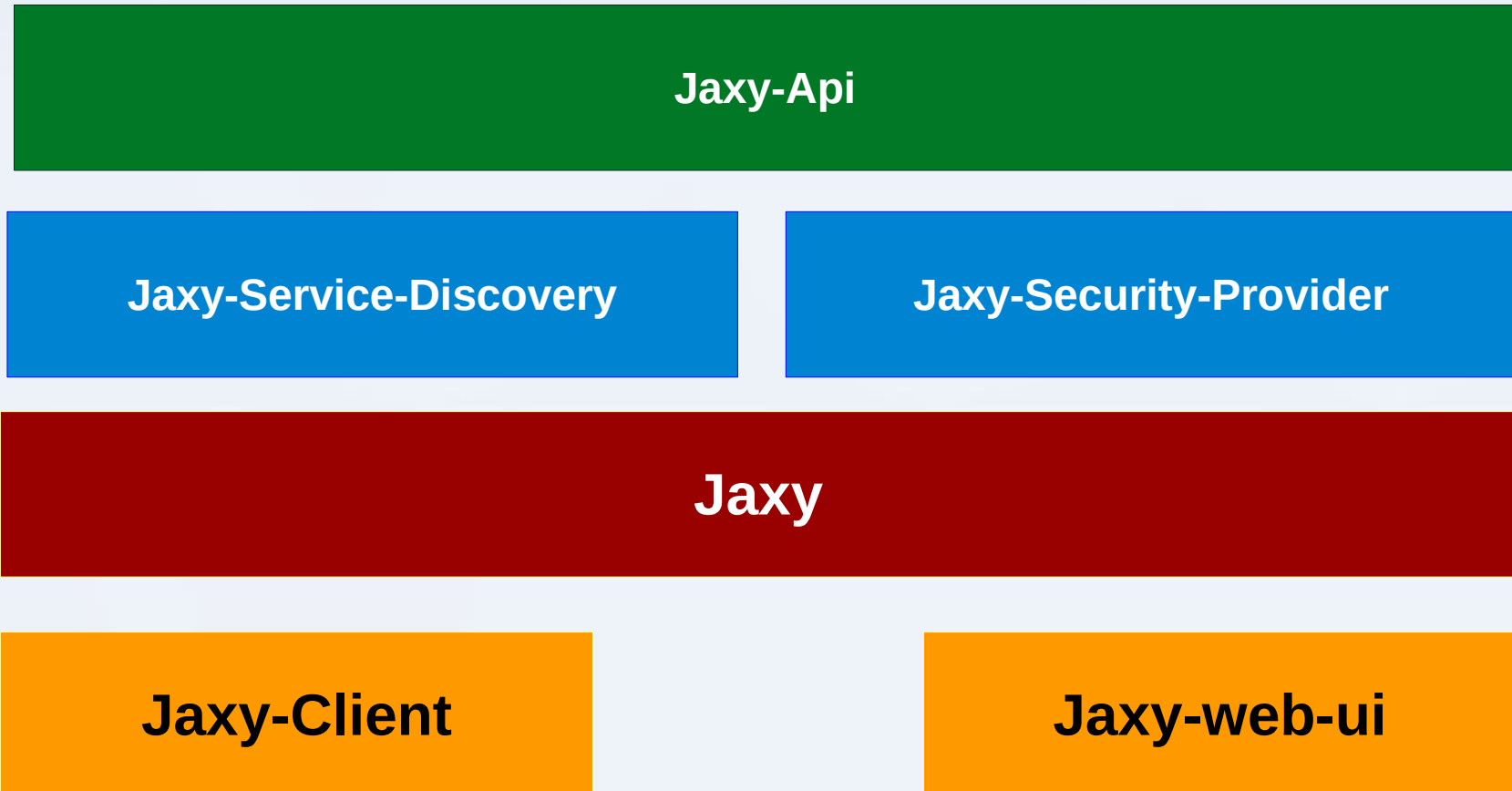
https://github.com/rac021/Jax-Y/tree/master/demo_sourceForge

→ SourceForge (Jaxy + Client-UI)

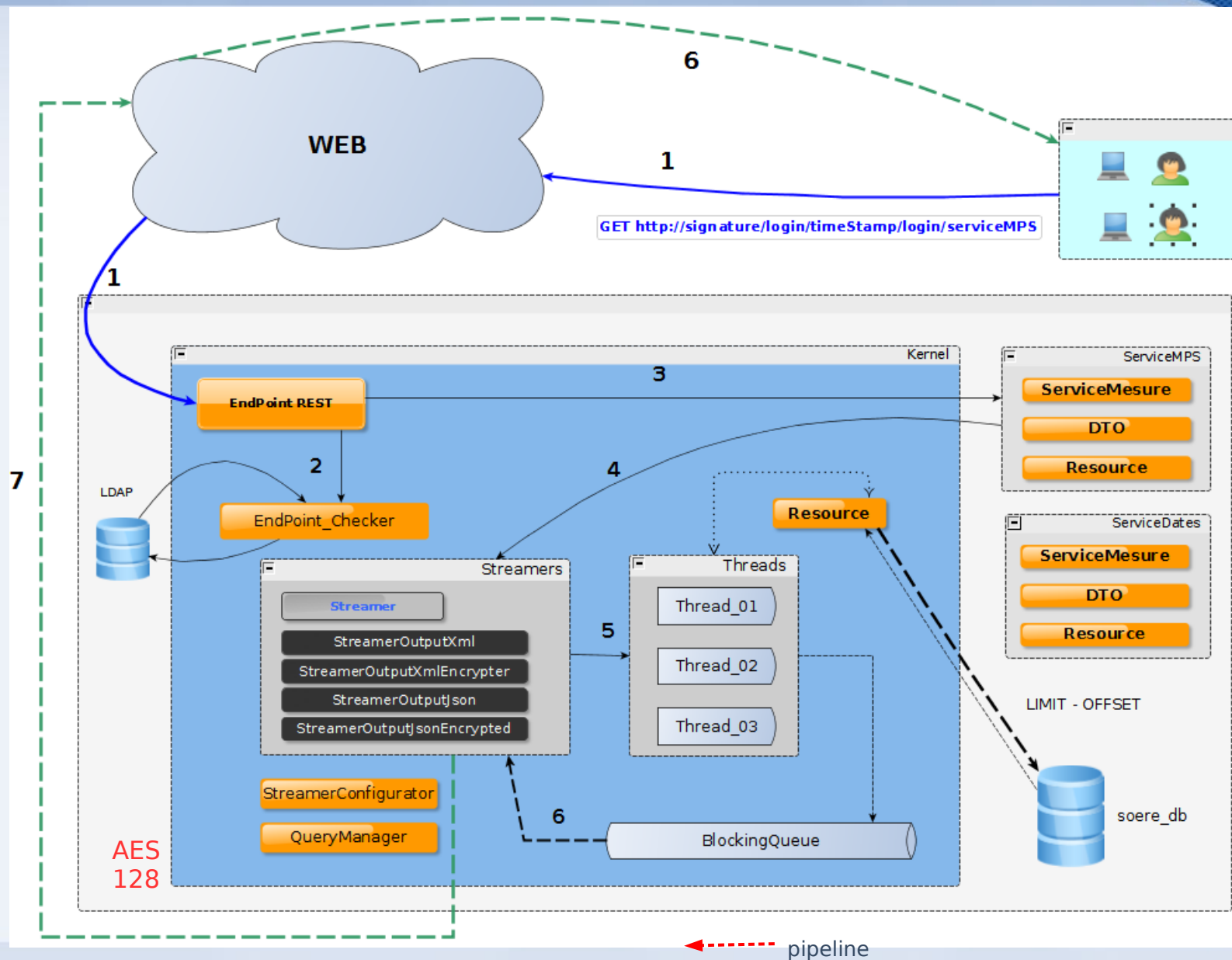
<https://sourceforge.net/projects/jaxy>



Architecture



Fonctionnement du système





Démo



MERCI DE VOTRE
ATTENTION

Question ?