

HEG-796-22-070

Valorisation

Jan Krause-Bilvin

2023-05-31

## Mise en oeuvre

---

Contenu de cette session:

1. Valorisation et indexation
  2. Plateforme LDP: documentation de Fedora Commons
  3. Recherche : indexation vs. triple store
  4. Et pour finir ...
- 

## 1. Valorisation et indexation

---

### Rappel

Animation d'intégration

---

### Besoin de “scalability”

En français, on pourrait dire d’“évolutivité”. Il y a une dimension d’échelle.

En pratique, considérons les concepts suivants:

- Messaging
  - Conteneurs
  - Fault-tolerance
-

## Messaging

Dans les application modernes, on utilise souvent le concept de *messaging*.

Par exemple, la technologie [Apache Caml](#) dont [Apache Kafka](#).

---

## Conteneurs

Dans les application modernes, on utilise souvent le concept de *conteneurs*, une façon de déployer des applications encore plus efficiente que la virtualisation.

Par exemple, la technologie [docker](#), voir aussi cette [video](#).

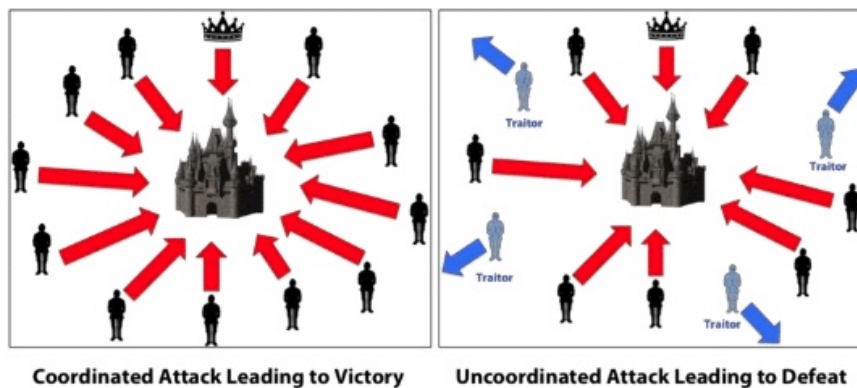
De plus, des systèmes d'orchestration permettent d'orchestrer les containers, p.ex. [kubernetes](#).

---

## Fault tolerance

Éviter la corruption: le problème des Généraux Bizantins.

### Byzantine Generals Problem (Leslie Lamport)



Il est démontré que pour tolérer  $f$  fautes,  $3f+1$  noeuds sont nécessaires. Cf. travaux de Lesslie Lamport fin des années 1970.

---

Ressources pour approfondir:

- [introduction](#)
- [introduction visuelle](#)
- [lien avec la blockchain](#)
- [proof of work](#)

- [proof of work vs. proof of stake](#)
  - [AcademicOnFlux](#)
- 

## Résumé

- Messaging : optimiser le transfert de données entre les composants du système d'archivage
- Conteneurs : optimiser la charge des dans les datacenter et la protabilité (dépendances)
- Fault-tolerance : garantir la cohérence des données (prévention piratage et intégrité des données)

Fedora Commons est construit dans cette optique.

---

## 2. Plateforme LDP: documentation de Fedora Commons

---

### [Lyrasis - Fedora Commons - Documentation](#)

- Architecture
  - [Installation](#) (one click , docker)
  - [Configuration](#)(paramètres)
    - [configuration docker](#) (users, pramètres:fcrepo.properties)
    - [liste des paramètres](#)
      - exemples: fcrepo.external.content.allowed, fcrepo.autoversioning.enabled, fcrepo.metrics.enable
- 

### [Points](#) à ne pas ratter:

- Messaging
  - External Search
  - Metrics
  - Auto-versionning
  - Namespaces
- 

### 3. Recherche : indexation vs. triple store

---

## Recherche : Indexation

Exemples de moteurs de recherche: Solr, ElasticSearch. [Introduction](#).

Des règles pour l'indexation via le messaging doivent être définies, p.ex.: rico:title  
-> index titre.

Cf. [configuration de Solr avec Fedora](#).

---

## Recherche : Triple store

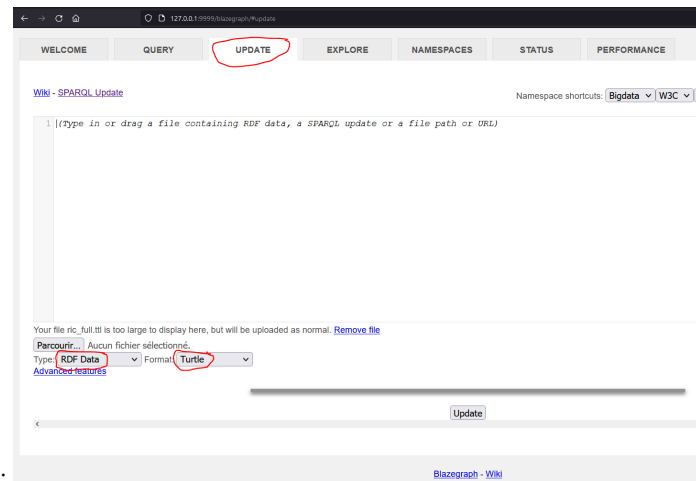
Nombreux outils disponibles, voir:

- [comparison of triplestores / structured storage](#) et
  - [large triplestores](#).
- 

## Exemple d'usage avec Blazegraph

[blazegraph](#), un triple store en java.

---



Permet l'upload de turtle (via interface web ou API):

---

## Fonds privés de Montreux

Le logiciel et les données sont disponibles sur cyberlearn.

En ligne, cela correspond à: [vaud.archivescommunales.ch](#) - Montreux.

---

Utilisation:

```
cd blazegraph_fedora
java.exe -server -Xmx4g -jar blazegraph.jar
```

Puis dans le navigateur: <http://127.0.0.1:9999/blazegraph/>

---

Quels prédicats RiC ont été utilisés?

---

```
PREFIX rico: <https://www.ica.org/standards/RiC/ontology#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT ?pred WHERE {
  ?sub ?pred ?obj .
  FILTER(regex(str(?pred), "RiC" ) )
}
LIMIT 100
```

---

Lister les intitulés de tous les records.

---

```
PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
  ?sub rico:title ?obj
}
LIMIT 100
```

---

Qui est le parent de *object:001228-4-PP060-A-01-04* ?

---

```
PREFIX object: <http://localhost:8080/rest/object/>
PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
  object:001228-4-PP060-A-01-04 rico:isOrWasPartOf ?obj .
}
LIMIT 100
```

---

Qui sont les parents (récursifs) de *object:001228-4-PP060-A-01-04* ?

---

```

PREFIX object: <http://localhost:8080/rest/object/>
PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
    object:001228-4-PP060-A-01-04 rico:isOrWasPartOf* ?obj .
}
LIMIT 100

```

---

Qui sont tous les enfants (récursifs) de *object:001228-4-PP060* ?

---

```

PREFIX object: <http://localhost:8080/rest/object/>
PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
    object:001228-4-PP060 ^rico:isOrWasPartOf* ?obj .
}
LIMIT 100

```

---

Quels sont les records dont les intitulés:

— contiennent “hôtel”, “hotel”, “Hôtel”, or “Hotel”

---

```

PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
    ?sub rico:title ?obj .
    FILTER(regex(str(?obj), ".*[Hh][oô]tel.*" ) )
}
LIMIT 100

```

---

Quels sont les records dont les intitulés:

— contiennent “hôtel”, “hotel”, “Hôtel”, or “Hotel”  
 — et datant des années 1930 ou 1940

---

```

PREFIX rico: <https://www.ica.org/standards/RiC/RiC-0_v0-2.html#>

```

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT DISTINCT * WHERE {
  ?record rico:title ?title .
  ?record rico:isAssociatedWithDate/rico:expressedDate ?date .
  FILTER(regex(str(?title), ".*[Hh][oô]tel.*" )) .
  FILTER(regex(str(?date) , ".*19[34]\\d*" )) .
}
LIMIT 100

```

---

4. Et pour finir ...

---

### Exemples de réalisations suisses utilisant les concepts abordés à ce cours

- [Bodmer Lab, UNIGE](#) : FedoraCommons + ElasticSearch + InterfaceAdHoc
  - [memobase.ch](#) : FedoraCommons + ElasticSearch + InterfaceAdHoc
  - [docuteam cosmos](#) : FedoraCommons + Solr + AtoM / InterfaceAdHoc.
- Exemples:
- [SIAr neuchâtelois](#)
  - [société des Arts genevoise](#)
  - [Registre Foncier valaisan](#)
  - ...
- 

### Idées de TB

De nombreuses instances AtoM existent en Suisse, notamment:

- [vaud.archivescommunales.ch](#)
- [ge.archivescommunales.ch](#)
- [caspar.cinematheque.ch](#)
- [architrave-hesge.ch](#)
- [arCHeco.info](#)
- [Registre Foncier valaisan](#)
- [f-information.org](#)
- [CIRA.ch](#)
- ...

De plus, les inventaires de Bâle-Ville, PTT et Memoriav sont déjà disponibles au format RiC.

---

Sujet possible de TP (variante à préciser/discuter):

- Intégration d'inventaires archivistiques dans le connectome
- Valorisation des archives sur Wikidata
- Diffusion d'inventaires archivistiques sur le Web sémantique
- Publication de la plateforme AtoM sur le Web sémantique
- etc.

Intéressé.e? => jan.krause@vd.ch

---

### **Évolution du cours**

=> Évaluation du cours en ligne sur le portail AGE.

---