JIO and WebViewers: interoperability for Javascript and Web Applications

Ludovic Dubost
CEO XWiki SAS

Viktor Horvath
Alixen

October 4th 2013

Agenda

- About the Speakers
- Why Interoperability
- JIO: accessing Web Applications from Javascript applications
- WebViewers: building cross-applications Javascript viewers and editors for content types
- Question and Answers

About the Speakers

Ludovic Dubost, XWiki SAS

- Creator of XWiki in 2004
- 15 years of Web Development

- French company, 30 employes
- Services and Solutions based on XWiki
- 1,6M revenue
- Independent company owned by employees
- Participating to research projects, including RESILIENCE

- LGPL Open Source Software
- Improving Information organization and sharing in companies
- Recognized project with more than 1 Million downloads
- 8 participations to the Google Summer of code
- Tens of thousand XWiki installed

Viktor Horvath, Alixen

- R&D engineer at Alixen since 2008
- Alixen: Free software development,
  maintenance,
  infrastructures,
  consulting
- Founded in 2002, merged with Systemis in 2012
- Mioga2: Collaborative Extranet
- Biblio-PAM: remote access to library resources
- Alixen participates in the Resilience project:
  J-IO, RenderJS

Resilience

- collaborative research project (FUI12)
  11 labs and small / mid-size companies
- Resilient Computing
- "decentralised cloud"
- SlapOS, J-IO, RenderJS, OfficeJS
- Members: Alcatel, Alixen, Alterway, le CEA, l'INRIA, Institut Télécom Université Paris 13, Mopho, Nexedi, Wallix et XWiki

Essonne
LE CONSEIL GÉNÉRAL

îledeFrance
Demain s'invente ici

SYSTEMATIC
PARIS REGION SYSTEMS & ICT CLUSTER

Alixen

Why Interoperability ?

Why Interoperability ?

We can make Web Applications communicate with each other

We can reduce the effort needed for Open Source projects to support more features

Alixen

JIO

http://j-io.org

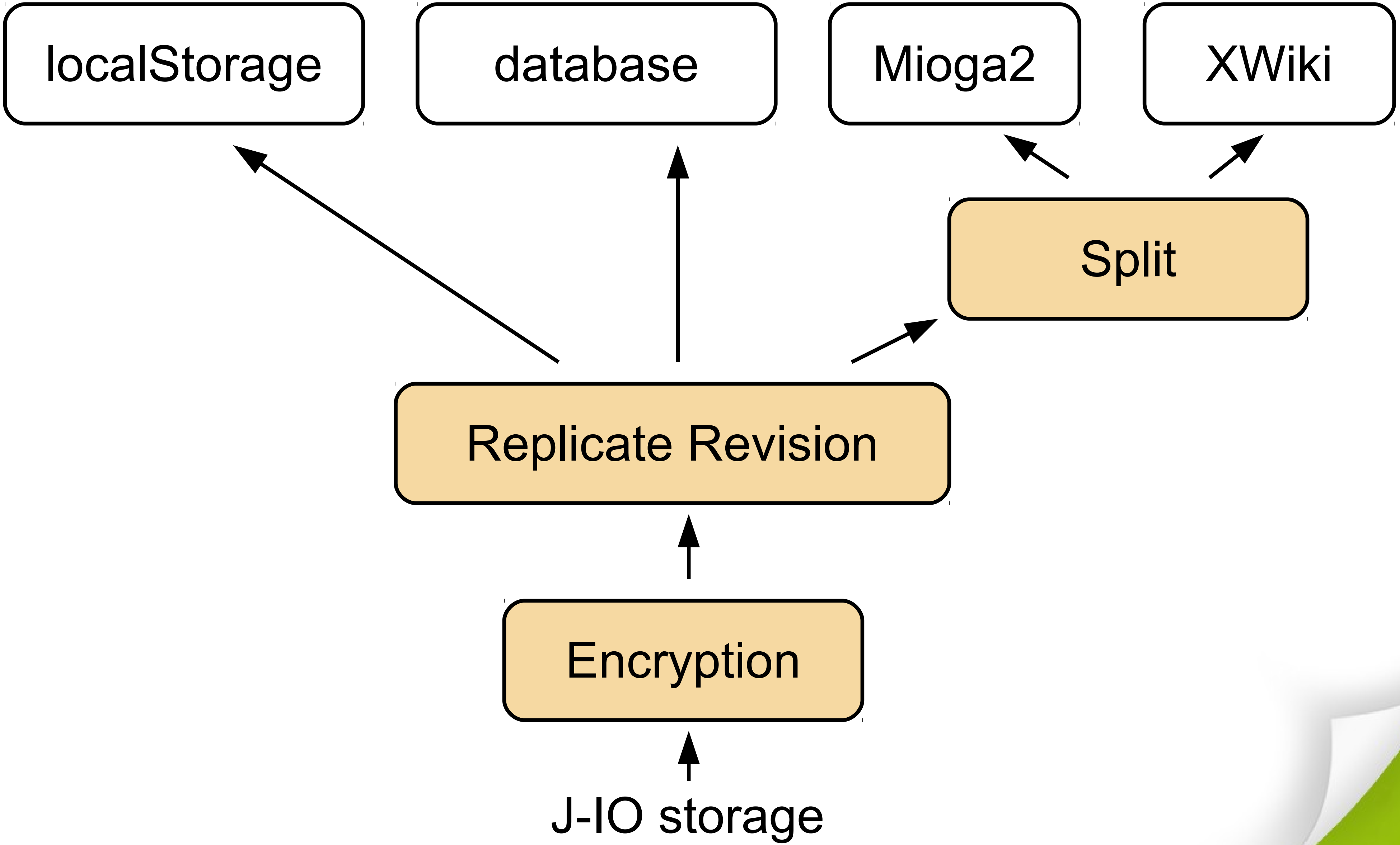
Why JIO ? - Some years ago :

- Web servers, databases are expensive
- Server-side applications
- Apache web server

Why JIO ? - Now :

- Web servers, databases are ~~expensive~~ much cheaper
- ~~Server-side~~ applications Client-side
- Apache web server or asynchronous web and application servers
- many more users per server
- ...data storage cost remains!

Separate web application and its data



Characteristics

- document-oriented storage
- files as attachments
- similar to NoSQL / JSON storages
- enrich a basic storage:
  indexes, revisions

```
{
  "_id": "my_document",
  "title": "New Title",
  "_attachments": {
    "its_attachment": {
      "length": 3,
      "digest": "md5-e0089e[...]",
      "content_type": "text/plain"
    }
  }
}
```

Storage Connector: API methods

- post / put / putAttachment
- get / getAttachment
- remove / removeAttachment
- allDocs (complex queries)

Connection example

```
var local_jio = jIO.newJio({
    type: 'local',
    username: 'viktor',
    application: 'test' });

local_jio.put({
    _id: '123',
    title: 'My Document',
    setting: 'important_data'
  }, function(err, response) {  console.log(response); });
```

```
Object { ok=true, id="123"}
```

```
local_jio.get({ _id: '123' },
  function(err, doc) { console.log(doc.setting); })
```

```
important_data
```

Complex Queries

```
local_jio.allDocs({
    query: 'setting: "important_data"'
  }, function(err, response) { console.log(response); });
```

```
Object { total_rows=1, rows=
  [ Object { id="123", key="123", value={...}} ] }
```

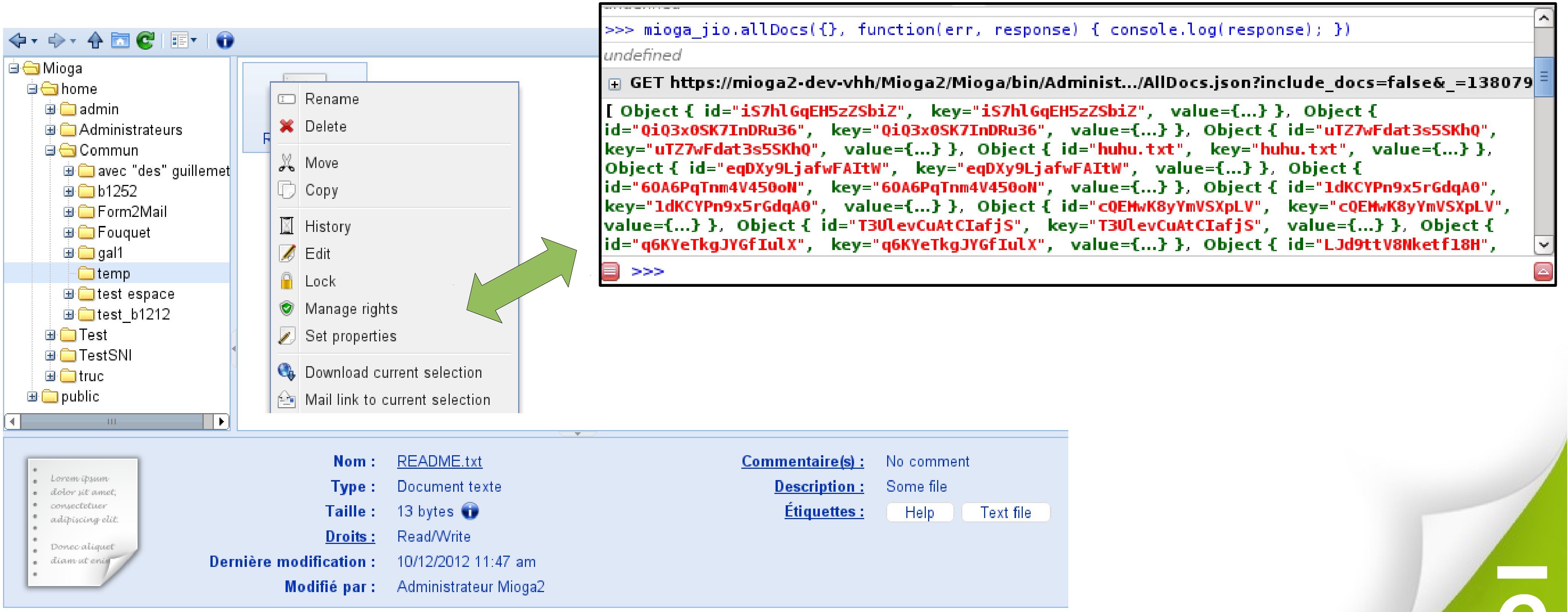
```
local_jio.allDocs({
    query: '(title:"% Document")
                 AND (setting: "important_data")',
    limit: [0, 100],
    sort_on: [['last_modified', 'descending']],
    select_list: ['title'],
    wildcard_character: '%'
  }, function(err, response) { console.log(response); });
```

What is Mioga2 ?

- Collaborative Extranet for 14 years
- Project and mission management
- within and across organisational borders
- workgroups, document sharing, applications
- GPLv2 download: http://www.alixen.org

Mioga2 as a JIO Storage Connector

- Workgroups have WebDAV spaces
- WebDAV - JIO / filesystem - documents



```
>>> mioga_jio.allDocs({}, function(err, response) { console.log(response); })
undefined
GET https://mioga2-dev-vhh/Mioga2/Mioga/bin/Administ.../AllDocs.json?include_docs=false&_=138079
[ Object { id="iS7hlGqEH5zZSbiZ",  key="iS7hlGqEH5zZSbiZ",  value={...} }, Object {
id="QiQ3x0SK7InDRu36",  key="QiQ3x0SK7InDRu36",  value={...} }, Object { id="uTZ7wFdat3s5SKhQ",
key="uTZ7wFdat3s5SKhQ",  value={...} }, Object { id="huhu.txt",  key="huhu.txt",  value={...} },
Object { id="eqDXy9LjafwFAItW",  key="eqDXy9LjafwFAItW",  value={...} }, Object {
id="60A6PqTnm4V450oN",  key="60A6PqTnm4V450oN",  value={...} }, Object { id="1dKCYPn9x5rGdqA0",
key="1dKCYPn9x5rGdqA0",  value={...} }, Object { id="cQEHwK8yYmVSXpLV",  key="cQEHwK8yYmVSXpLV",
value={...} }, Object { id="T3UlevCuAtCIafjS",  key="T3UlevCuAtCIafjS",  value={...} }, Object {
id="q6KYeTkgJYGfIulX",  key="q6KYeTkgJYGfIulX",  value={...} }, Object { id="LJd9ttV8Nketf18H",
>>>
```

Nom :	README.txt	Commentaire(s) :	No comment
Type :	Document texte	Description :	Some file
Taille :	13 bytes	Étiquettes :	Help · Text file
Droits :	Read/Write		
Dernière modification :	10/12/2012 11:47 am		
Modifié par :	Administrateur Mioga2		

When should you use JIO ?

- enable the user to control his data
- ease your burden
- unify your data access
- custom user interface

- ...towards democratic, truly Free web apps

Web Viewers

WebViewers: the Problem

- Many web applications store files (pdf, office, graphs, etc..)
- We need ways to view/edit these files in your browser
- JavaScript viewers and editors exist but they are complex to integrate

WebViewers: Objectives

- One viewer/editor for all Web Applications
- One viewer/editor macro for all viewers
- Automatic selection of viewer based on content-type
- JavaScript only viewers (backend technology agnostic)

WebViewers: How does it work in a Web Application

- Viewers are distributed as zip files
- A module to install viewers
- A way to activate the viewer
  - Macro to use in content
  - View/Edit button next to a file
- A way to read/write the file content in JavaScript

WebViewers: technologies used

- RenderJs: gadget technology
- JIO: api to read/write files (optional)
- Require.js

WebViewers: demo on XWiki, Wordpress

- Install a viewer in XWiki
- Attach a file in XWiki
- Use a macro to view/edit the file in XWiki
- Show viewers in Wordpress

WebViewers: support level

- Web Applications
  - Available: XWiki view/edit, Wordpress view
  - Coming: ERP5, Mioga2, Drupal, MediaWiki, ?
- Web Viewers
  - Available: text, pdf, jquery sheet
  - Coming: SVG Edit, Realtime editor, ?

WebViewers: future

- Reaching out to Web Applications developers for native support
- Reaching out to Viewer/Editor developers for native support

http://www.webviewers.org

Alixen

Questions ?

Ludovic Dubost:
ludovic@xwiki.com, @ldubost

Viktor Horvath
info@alixen.fr