

VOSpace 2.0 implementation status at CDS

André Schaaff
CDS

Vincent Meslard (now in an another lab)

Interop meeting, Nara, 7-11 December 2010

VOSpace effort

- VOSpace 1.15 & iRODS implementation
 - no maintenance and no support
- Whole effort on the VOSpace 2.0 version
 - adaptable to several data storage systems
 - tested and documented to be distributed

VOSpace 2.0 implementation

- Developed in Java and based on the Restlet framework
- Flexible backend based on the use of a GenericBackend interface which has to be implemented for each kind of backend
 - Available backends
 - LocalFileSystem
 - iRODS
- Name for the project : XVOSI (version 0.9, 1.0 when Rec.)
- eXtensible VOSpace Implementation

Existing backends

- Requirement : XVOSI installed
- Choose a backend : LocalFileSystem
 - Set the path
 - to the data (e.g. /home/myvospace)
 - to the metadata (e.g. /home/myvospace.meta)
 - Set the port
 - Set or update the service metadata

New backend

- Creation of a new backend : myOwnBackend
- write MyOwnBackend.java
 - createNode, moveNode, deleteNode (must be tested before the production phase !), ... must be defined here
 - you can use the 2 existings backends as models
- change the backend in VOspace

MyOwnBackend.java

```
package cds.vospace2.backend;

import ... // all the needed classes

public class MyOwnBackend implements GenericBackend {

    // method from GenericBackend must be implemented
    void storeNode(VOUnstructuredDataNode node) throws Exception
    { ... }
    void storeNode(VOStructuredDataNode node) throws Exception
    { ... }
    void storeNode(VOContainerNode node) throws Exception
    { ... }
    void storeNode(VOLinkNode node) throws Exception { ... }
    void storeNode(VONode node) throws Exception { ... }

    void copyNode(VOUnstructuredDataNode origin,
VOUnstructuredDataNode destination) throws Exception { ... }
    ...

    void deleteNode(VOUnstructuredDataNode node) throws
Exception { ... }
    ...
}
```

Set the backend

```
change the backend when VOspace is started :VOspace.setBackend
( new MyOwnBackend() );
```

VOSpace implementation (2)

- Provide other backends ?
- Not easy to write a «generic» backend for a database, it depends on the description of the DB. Maybe an example for a specific DB.

Security

- TLS with password herited from our 1.15 implementation
- We have tested SecurityFacade

StructuredDataNode case

- We have made tests with VOTable files and SAVOT
 - parsing of the files to SAVOT internal model, serialization/unserialization, SavotWriter to restore the VOTable file
 - in the case of StructuredDataNode you are not sure to retrieve exactly the same file (metadata and data ok, but presentation can be changed, comments (<!-- ... -->) can be removed)
 - the user of the VOspace must be aware of that fact !!!

Compliance of a VOSSpace

- A VOSSpace implementation should be compliant with the compliance Matrix
- Writing of a partial compliant backend should perhaps be possible ?
- ...

before XVOSI release

- Current version uses an adhoc implementation of UWS which will be replaced by the new CDS UWS library before the first release
- Take into account the transfer change

XVOSI release

- Sources (GPL3) available when VOSpace 2.0 will be a Recommendation
- Documentation about the developement of other backends

Additional tools

- XVO SI admin for
 - the changing of the metadata of the service
 - the checking of the coherence between data and metadata of the VOSpace
- XVO SI client to
 - access the service
 - launch unit tests

Other work for next versions

- Optimization of the storage
 - data compression
- Study about the use in workflows
- Update the VOSpace Explorer developed for the 1.15 version ?
- Following the Victoria presentation about clouds, perhaps some real tests

Which use at CDS ?

- We use iRODS
 - without VOSpace for internal needs
 - no time overhead due to an additional layer
- VOSpace 2.0 implementation used to make tests with our tools and services and to make them VOSpace 2.0 compatible

Conclusion

- Near the first release
- Period of deep tests in progress
 - focus of the integrity of the service (loose of files, etc.)