D4.1a - VISHNU File Management Services Package Design

	COLLABORATORS		
	TITLE : D4.1a - VISHNU File Management Services Package Design		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Benjamin Isnard, Daouda Traoré, Eugène Pamba Capo-Chichi, Kevin Coulomb, and Ibrahima Cissé	February 5, 2013	

	REVISION HISTORY		
NUMBER	DATE	DESCRIPTION	NAME
1	25/03/2011	Deliverable version	SysFera
2	05/12/2012	Rename of FMS API functions according to POSIX names	SysFera

Contents

List of Figures

Chapter 1

Document presentation

1.1 Document objectives

This document presents the detailed internal design of the File Management Services (FMS) module. The purpose of this package is to handle all aspects of file transfer and file property modification within the VISHNU system. The functional and non-functional requirements for this module are those described in the [D1.1a] and [D1.1c] document (see References). The current document is part of the design phase of the software and therefore its main goal is to define the main class used in the FMS software and their relationships.

1.2 Document structure

- Chapter 1 contains a brief overview of the document content.

- Chapter 2 contains a high-level overview of the system architecture except the deployment aspect which is well discribed in the architecture document (see [D1.1g]).

- Chapter 3 describes the internal API used for remote procedure calls through SysFera-DS.

- Chapter 4 describes the internal classes and data structures except the Vishnu core functions modelization which is included in the VISHNU Users Management Service Design document (see[D2.1a]).

1.3 References

- [D1.1a]: VISHNU General specifications

- [D1.1b]: VISHNU Spécifications techniques des besoins

- [D1.1c]: VISHNU API Detailed specifications

- [D1.1g]: VISHNU technical architecture

- [D2.1a]: VISHNU users management services package design

1.4 Acronyms

- **API**: Application programming interface

- **CLI**: Command line interface

- **DB**: DataBase

- **n/a**: Not Appliable (used for serializable capability in function descriptions)

- **SeD**: A Server Daemon is a SysFera-DS agent that provides services through the SysFera-DS API.

- **FMS**: File management services

- **WS**: Web services

1.5   Glossary

- **Components**: the software components represents a library or an executable program that provides a given interface to other components or to end-users.

- **Serialized type**: this is a class of data (C++ Class) which instances can be serialized in a XML string before being sent over an API (to or from the API). The data is deserialized on the other side of the channel in order to re-build the same instance of the class.

- **SysFera-DS**: open-source middleware developped by SysFera.

Chapter 2

System Architecture

2.1 Introduction

The purpose of this section is to precise the role of the FMS as VISHNU module. As indicated in [D1.1g], FMS is the part of VISHNU architecture which handles the file transfers between storage resources. Futhermore, FMS provides several other services to manipulate a property of a file, like changing its owner, changing its access permissions, get its complete or partial content, see the section for exhaustive list of providesd services.

2.2 Overview of the FMS software infrastructure

We present in this section a detailed description of the FMS package architecture in terms of software components. In addition we show the dependencies between components to highlight their reuse. These components follow a SOA model where each server publishes a set of services that can be used by the clients. We present the different software layers from services (provided directly to the user) to the database (used by the server). The FMS client server package has been split into seven different interrelated components. The deployment aspects have already been described in the technical architecture document (cf [D1.1g]). The diagrams shown in section 2.2 describe the relationships between these components. The definitions of the components are the following:

- **External API** contains precisely the services provided to the user as defined in the detailed specifications.

- **Internal API** publishes the services provided by the servers. The services of the external API are implemented using requests sent through the SysFera-DS middleware to this API.

- **FMS Client** contains intermediate (proxy) classes providing remote access to the business objects of **FMS Server**.

- **FMS Server** contains all classes implementing business objects by encapsulating the processing provided through the internal API.

- **Sysfera-DS Client API** is the C++ client API provided by the SysFera-DS toolbox.

- **Sysfera-DS Server API** is the C++ server API provided by the SysFera-DS toolbox.

- **Vishnu Database** has been already described in [D2.1a]. As we could see, it stores the file transfer data used by the FMS Server.

2.3 Installation prerequisites

2.3.1 Client side

The installation of FMS module on the client needs the following libraries

- VISHNU UMS libraries: the VISHNU User Management Service that allow the session creation and user authentification.

- C++ and Boost-C++ libraries

- Sysfera-DS Client libraries

- A SSH Server to perform a remote to local transfer file.

2.3.2   Server side

The installation of FMS module on the server side needs the following libraries:

- VISHNU UMS libraries: the VISHNU User Management Service that allow the session creation and user authentification

- C++ and Boost-C++ libraries

- Sysfera-DS Server libraries

- A SSH Server

- The Rsync Debian package

2.4   Architecture diagrams

2.4.1   FMS client-side components

This diagram shows the components that compose the client side of the Vishnu FMS system and their interfaces. Two services among all the services of the FMS external API (cf [D1.1c]) are shown here for example. These services are consumed by several user interfaces: command-line, web services and Python. All the interfaces of the FMS Client component are shown. We can note that the FMS External API uses the UMS Client Lib to handle user and machines authentication.
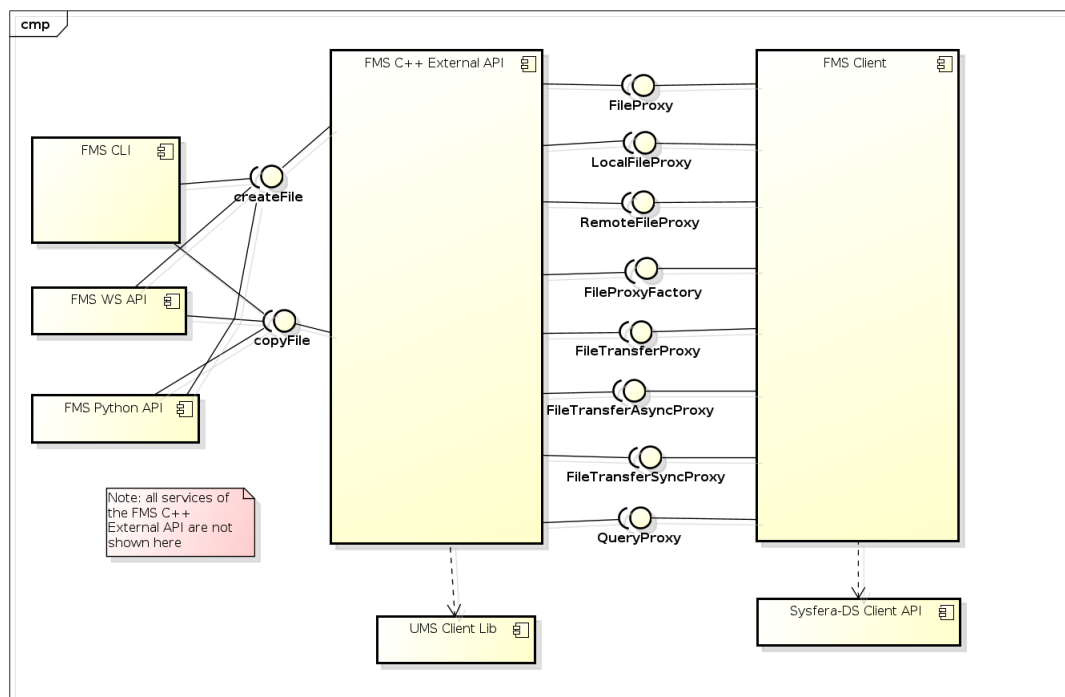
Figure 2.1: FMS client-side components

2.4.2 FMS server-side components

This diagram shows the components that compose the server side of the Vishnu FMS system and their interfaces. Two services among all the services of the FMS internal API are shown here for example. These services are published by the FMS Sed and are consumed by the FMS Client component through the SysFera-DS API. All the interfaces of the FMS Server component are shown.
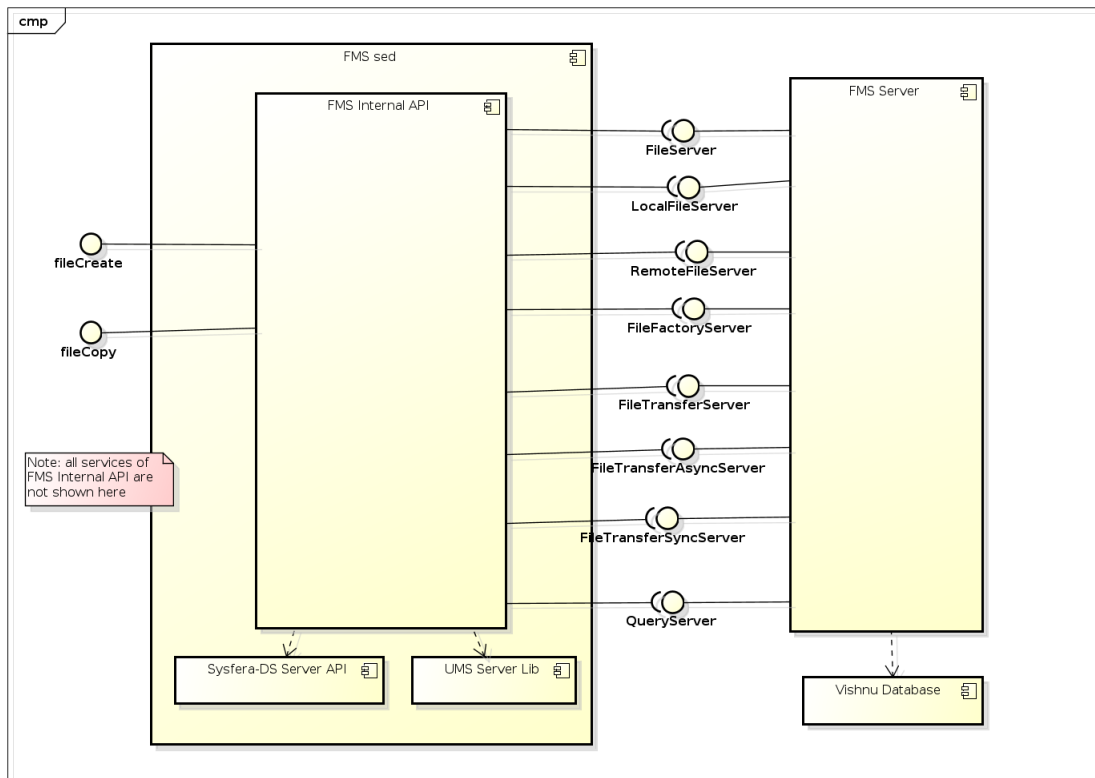
Figure 2.2: FMS server-side components

2.4.3 SysFera-DS Bus Details

This diagram shows the communication paths between the Client host and two FMS server using the SysFera-DS Bus. The SysFera-DS MasterAgent is a SysFera-DS agent that can be executed on a dedicated host or on the same host as the FMS Server. All the communications between the entities here are done using the CORBA IIOP (Internet Inter-ORB) protocol and the communications can be tunneled through SSH tunnels if necessary. The MasterAgent entity is involved in the choice of one FMS Server in the case of several available FMS servers. The choice will be transparent to the user as all FMS Servers connect to the same database. The diagram shows here all the communication paths in the case where one FMS Server (FMS Server 1) is chosen by the MasterAgent.
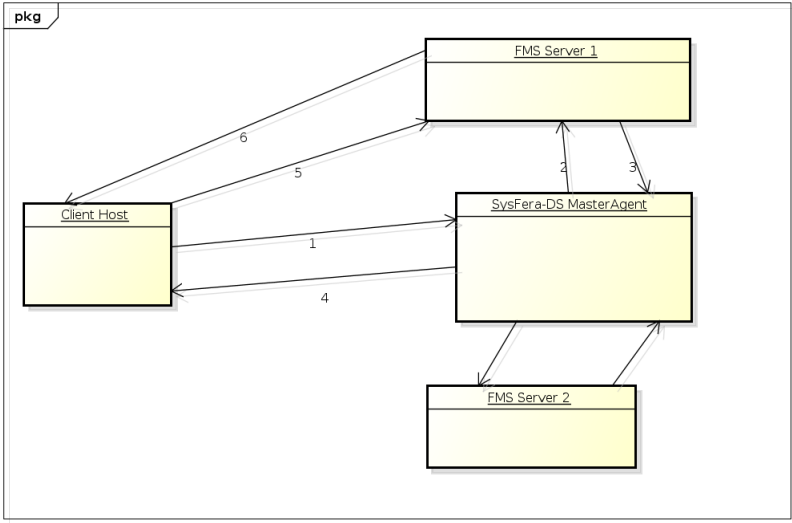
Figure 2.3: SysFera-DS Bus Details

Chapter 3

Internal API specification

3.1 Generic definition formats presentation

This section presents the formats used in this chapter to describe the services provided by the internal API.

3.1.1 Service definition format

Access

Here is detailed the access level of the service 'myService' (i.e. the privilege required to use it)

Parameters

The following table contains all the input and output parameters of the service, along with their type and description.

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	This is an example of a required string input parameter	IN
listOfJobs	string	ListJobs	This is an example of an object output parameter that is serialized as a string	OUT

Description

Here is detailed the purpose of the service 'myService'

Return Value

Here are detailed the different return codes provided by the service.

Name	Description
VISHNU_OK	The service has been performed successfully.
TMS_UNKNOWN_MACHINE	This is the human-readable generic message that will be available to the user of the API.

Used by this(these) API function(s):

This shows the list of functions from the external Vishnu API (see [D1_1c]) that use this service.

3.2 Definition of the services of the package

3.2.1 Service fileCreate

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file to create following the pattern [host:]file path	IN
mode	mode_t	n/a	The file access permissions in octal numeral system	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileCreate() function creates files on remote machines.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::touch

3.2.2 Service DirCreate

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The directory to create following the pattern [host:]directory path	IN
mode	mode_t	n/a	the new directories permission access in octal numeral system	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The DirCreate() function creates directories on remote machines.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::mkdir

3.2.3 Service fileRemove

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file to remove following the pattern [host:]file path	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileRemove() function removes files from remote hosts.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::rm

3.2.4 Service DirRemove

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN

Parameter	Type	Serialized type	Description	Mode
path	string	n/a	The directory to remove following the pattern [host:]directory path	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The DirRemove() function removes directories (and subdirectories) from remote machines.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::rmdir

3.2.5 Service grpChange

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file/directory following the pattern [host:]file path	IN
group	string	n/a	the new group owner of file/directory	IN
errorInfo	string	n/a	Additional information provided when an error code is returned	OUT

Description

The grpChange() function changes group owner of remote files/directories.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::chgrp

3.2.6 Service modeChange

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file/directory following the pattern [host:]file path	IN
mode	int	n/a	the access rigths of file/directory in octal system	IN
errorInfo	string	n/a	Additional information provided when an error code is returned	OUT

Description

The modeChange() function changes access rights of remote files/directories.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::chmod

3.2.7 Service fileHead

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file following the pattern [host:]file path	IN
fileContent	string	n/a	The first "nLine" lines of the file	OUT
nLine	size_t	n/a	Number of lines to display.	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileHead() function displays a few first lines of files located on remote machines.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.

Name	Description
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::head

3.2.8 Service fileTail

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file following the pattern [host:]file path	IN
fileContent	string	n/a	The last "nLine" lines of the file.	OUT
nLine	size_t	n/a	number of lines to display.	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileTail() function displays a few last lines of files located on remote machines

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::tail

3.2.9 Service fileContent

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file to display following the pattern [host:]file path	IN
fileContent	string	n/a	The content of the file.	OUT
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileContent() function displays content of files located on remote machines

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
VISHNU_OK	The service has been performed succesfully.
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::more

3.2.10   Service dirList

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The directory to list following the pattern [host:]directory path	IN
dirContent	string	DirEntryList	The content of the directory (only files names with short format, all files informations with long format).	OUT
options	string	LsDirOptions	List of options for the ls command.	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The dirList() function displays the content of a remote directory

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::ls

3.2.11   Service fileCopy

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
src	string	n/a	The source file to copy following the pattern [host:]file path	IN
dest	string	n/a	The path of the destination file.	IN
options	string	CopyFileOptions	the copy options.	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileCopy() function executes a synchronous copy of file.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_UNKNOWN_MACHINE	The machine id is unknown
ERRCODE_SESSIONKEY_NOT_FOUND	The session key is unrecognized
VISHNU_OK	The service has been performed succesfully.

Used by this(these) API function(s):

FMS::cp

3.2.12   Service fileAsyncCopy

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
src	string	n/a	The source file to copy following the pattern [host:]file path	IN
dest	string	n/a	The path of the destination file	IN
thrId	long	n/a	A file tranfer identifier (allowing for instance to ckeck the status of a file transfer, or to cancel it)	OUT
options	string	CopyFileOptions	the copy options	IN
errorInfo	string	n/a	Additional information provided when an error code is returned	OUT

Description

The fileAsyncCopy() function executes an asynchronous copy of file.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
ERRCODE_UNKNOWN_MACHINE	The machine id is unknown
ERRCODE_SESSIONKEY_NOT_FOUND	The session key is unrecognized
VISHNU_OK	The service has been performed succesfully.

Used by this(these) API function(s):

FMS::acp

3.2.13 Service fileMove

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
src	string	n/a	The source file to move following the pattern [host:]file path	IN
dest	string	n/a	The path of the destination file	IN
options	string	MvFileOptions	The move command options	IN
errorInfo	string	n/a	Additional information provided when an error code is returned	OUT

Description

The fileMove() function executes a synchronous move of file.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::mv

3.2.14 Service fileAsyncMove

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
src	string	n/a	The source file to move following the pattern [host:]file path	IN
dest	string	n/a	The path of the destination file.	IN
thrId	long	n/a	The file tranfer identifier (allowing for instance to ckeck the status of a file transfer, or to cancel it).	OUT
options	string	MvFileOptions	The mv command options	IN
errorInfo	string	n/a	Additional information provided when an error code is returned	OUT

Description

The fileAsyncMove() function executes an asynchronous move of file.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::amv

3.2.15   Service FileTransferCancel

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
options	string	StopTransferOptions	The stop file transfer command options	IN

Description

The FileTransferCancel() function stops an execution of a set of file transfers.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::stopFileTransfer

3.2.16 Service fileTransfersList

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
fileTransferList	string	FileTransferList	The file transfer container.	OUT
options	string	LsTransferOptions	The filter options .	IN
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The fileTransfersList() function displays the history of all file transfers submitted by User.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::listFileTransfers

3.2.17 Service FilesInfo

Access

This service can be used by any VISHNU user

Parameters

Parameter	Type	Serialized type	Description	Mode
sessionKey	string	n/a	The session key	IN
path	string	n/a	The file whose inode information will be displayed	IN
filesinfo	string	FileStat	The inode information.	OUT
errorInfo	string	n/a	Additional information provided when an error code is returned.	OUT

Description

The FilesInfo() function displays the information of files.

Return Value

An error code is returned when an error occurs during the execution of the service

Name	Description
ERRCODE_DBERR	Vishnu not available (Database error)
ERRCODE_INVALID_PARAM	an option or a parameter provided is invalid for this service
ERRCODE_INVALID_PATH	The path provided is invalid
VISHNU_OK	The service has been performed succesfully.
ERRCODE_SESSIONKEY_EXPIRED	The session key has expired. The session is closed.

Used by this(these) API function(s):

FMS::stat

Chapter 4

Internal class and data structures

4.1 Introduction

This chapter introduces the details of the implementation of the different components described in chapter 2 (Architecture). It is composed of three sections:

- **Client modelization**: describes the classes used to implement the *FMS Client* component.

- **Server modelization**: describes the classes used to implement the *FMS Server* component.

- **Data modelization**: describes the data structure used to implement the *FMS Client* component and the *FMS Server* component.

- **Vishnu core functions modelization**: describes the classes and data structures used to implement the the VISHNU cross-modules components (components that are used by FMS and other VISHNU modules).

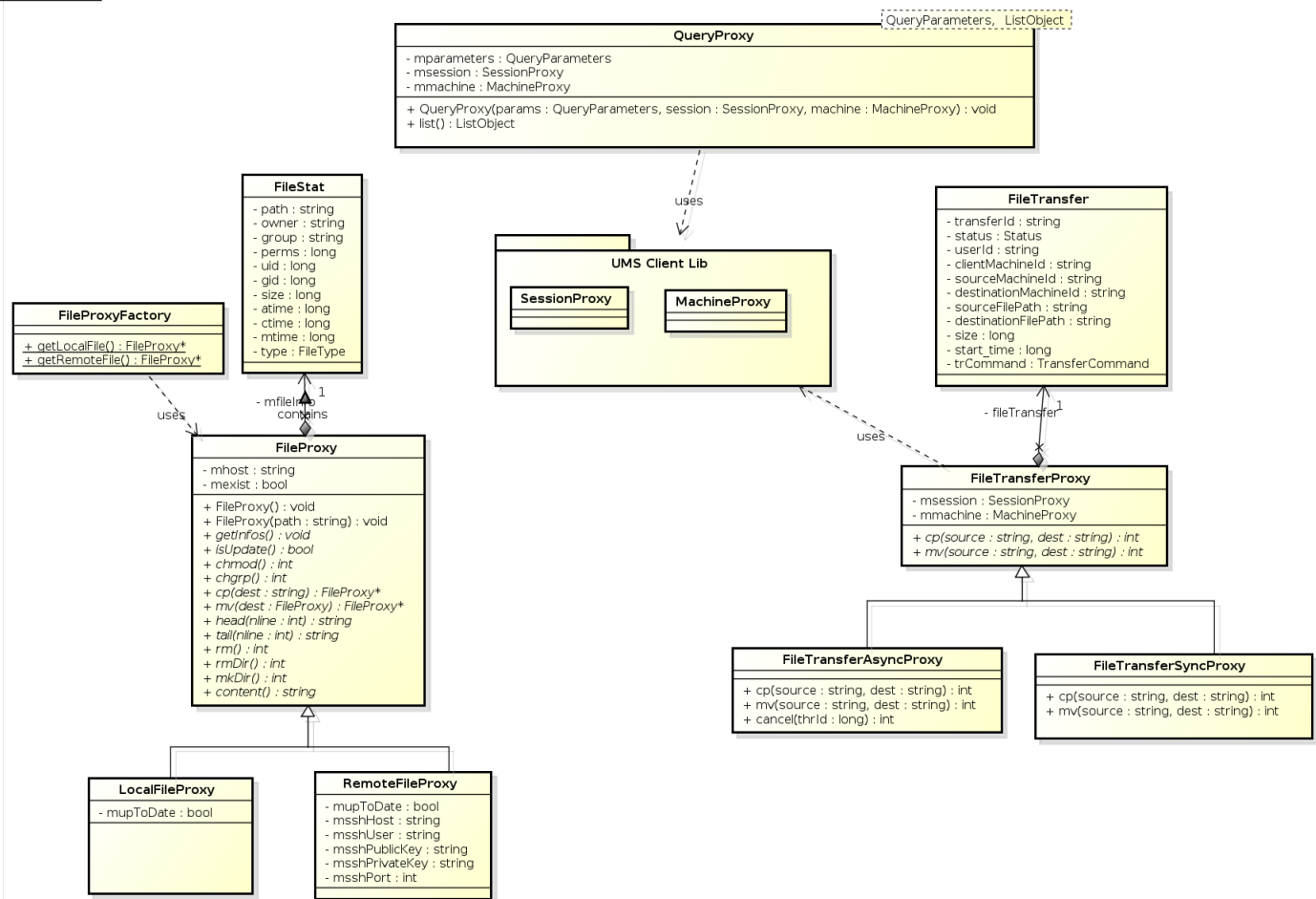## 4.2 FMS client modelization

4.2.1 Class diagrams

4.2.1.1 FMS Client Class Diagram

This diagram describes all classes used to communicate with VISHNU System. Each class proxy contains the corresponding data class illustrated on the FMS Data modelization section and the methods usable by the FMS Client component. A QueryProxy class implements a generic model to list objects of VISHNU.

pkg FMS_Proxy

QueryProxy

QueryParameters, ListObject

- mparameters : QueryParameters
- msession : SessionProxy
- mmachine : MachineProxy

+ QueryProxy(params : QueryParameters, session : SessionProxy, machine : MachineProxy) : void
+ list() : ListObject

FileStat

- path : string
- owner : string
- group : string
- perms : long
- uid : long
- gid : long
- size : long
- atime : long
- ctime : long
- mtime : long
- type : FileType

FileTransfer

- transferId : string
- status : Status
- userId : string
- clientMachineId : string
- sourceMachineId : string
- destinationMachineId : string
- sourceFilePath : string
- destinationFilePath : string
- size : long
- start_time : long
- trCommand : TransferCommand

uses

UMS Client Lib

SessionProxy

MachineProxy

FileProxyFactory

+ getLocalFile() : FileProxy*
+ getRemoteFile() : FileProxy*

uses

1
- mfileInfo
contains

FileProxy

- mhost : string
- mexist : bool

+ FileProxy() : void
+ FileProxy(path : string) : void
+ getInfos() : void
+ isUpdate() : bool
+ chmod() : int
+ chgrp() : int
+ cp(dest : string) : FileProxy*
+ mv(dest : FileProxy) : FileProxy*
+ head(nline : int) : string
+ tail(nline : int) : string
+ rm() : int
+ rmDir() : int
+ mkDir() : int
+ content() : string

uses

1
- fileTransfer

FileTransferProxy

- msession : SessionProxy
- mmachine : MachineProxy

+ cp(source : string, dest : string) : int
+ mv(source : string, dest : string) : int

LocalFileProxy

- mupToDate : bool

RemoteFileProxy

- mupToDate : bool
- msshHost : string
- msshUser : string
- msshPublicKey : string
- msshPrivateKey : string
- msshPort : int

FileTransferAsyncProxy

+ cp(source : string, dest : string) : int
+ mv(source : string, dest : string) : int
+ cancel(thrId : long) : int

FileTransferSyncProxy

+ cp(source : string, dest : string) : int
+ mv(source : string, dest : string) : int

4.3 FMS server modelization

4.3.1 Class diagrams

4.3.1.1 FMS Server Class Diagram

This diagram presents the main objects used by FMS server component to process the FMS Client component requests. The relationships between objects (simple dependancies, compositions, aggregations, and generalizations) are highlighted.The Query server performs the list operation.
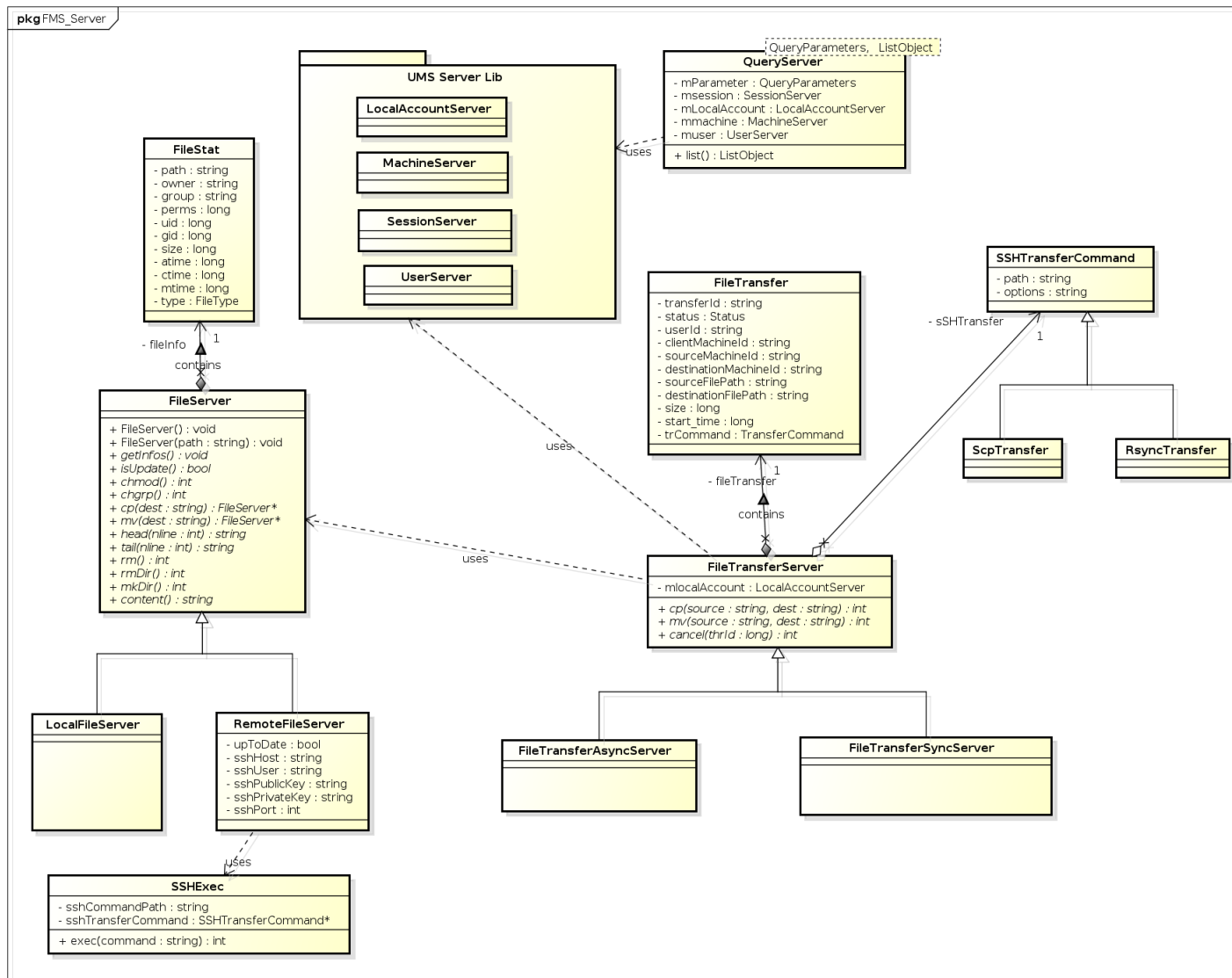

Figure 4.2: FMS Server Class Diagram

4.4 FMS data modelization

4.4.1 Class diagrams

4.4.1.1 FMS Data Class Diagram

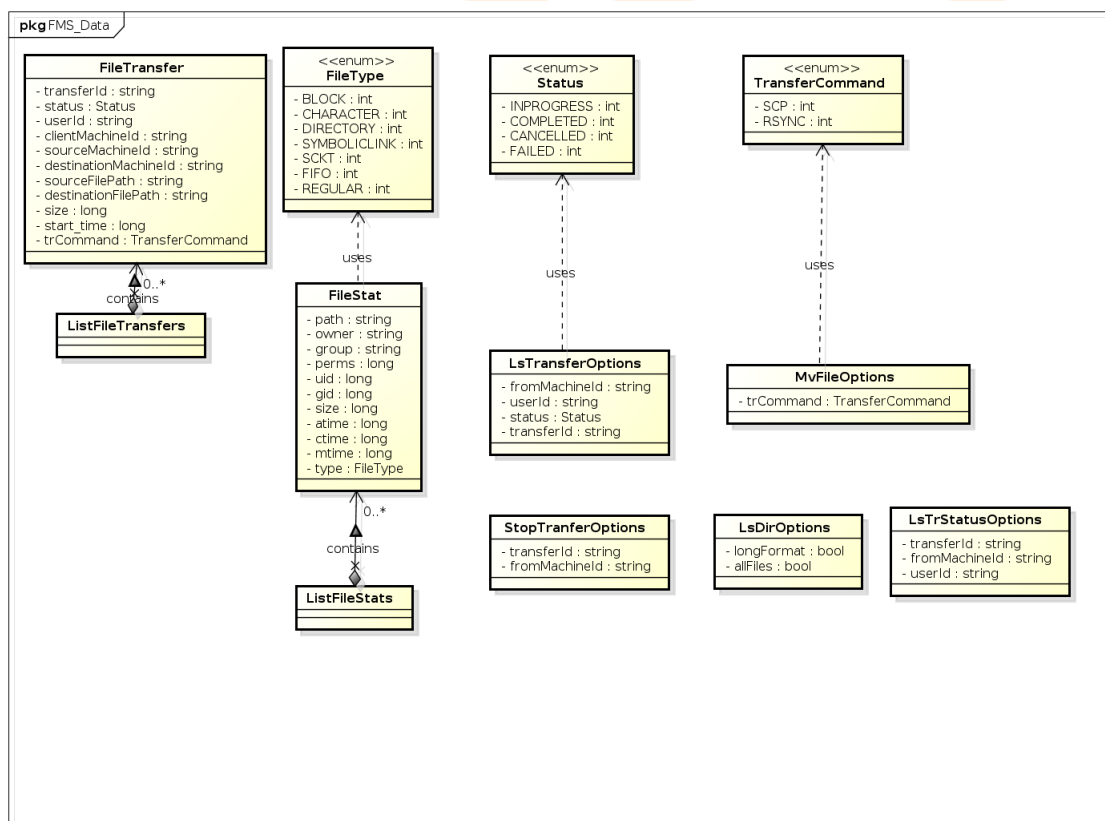This diagram illustrates the structure and the relationship between data manipulated by the components Client and Server.

Figure 4.3: FMS Data Class Diagram