Project no. IST-033576

XtreemOS

Integrated Project
BUILDING AND PROMOTING A LINUX-BASED OPERATING SYSTEM TO SUPPORT VIRTUAL ORGANIZATIONS FOR NEXT GENERATION GRIDS

D3.6.4
Requirements and specification of advanced services for mobile devices

Due date of deliverable: 31st May 2009
Actual submission date: 24th June 2009

Start date of project: June 1st 2006

Type: Deliverable
WP number: WP3.6
Task number: T3.6.4

Responsible institution: Telefónica I+D
Editor & editor's address: Santiago Prieto
Telefónica I+D, Parque Tecnológico de Boecillo
Boecillo (Valladolid)
SPAIN

Version 1.0/ Last edited by Telefónica I+D/ Date 24-09-2009

Project co-funded by the European Commission within the Sixth Framework Programme		
Dissemination Level		
PU	Public	✓
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Keyword List: design, linux, mobile device, smartphone

Revision history:

Version	Date	Authors	Institution	Sections Affected / Comments
0.1	12-Jan-09	Telefónica I+D	Telefónica I+D	First draft
0.2	08-Apr-09	Telefónica I+D	Telefónica I+D	Second draft
0.3	21-Apr-09	Telefónica I+D	Telefónica I+D	Third draft
0.4	22-Aprl-09	Toni Cortes	BSC	Added info in chapter 3
0.5	30-Apr-09	Santiago Prieto	Telefónica I+D	Added requirements and more content on scenarios
0.6	5-May-09	Toni Cortes	BSC	First version of the specification section
0.7	15-May-09	Toni Cortes	BSC	Completed the specification section
0.8	19-May-09	Santiago Prieto	Telefónica I+D	TID internal review
0.9	21-May-09	Santiago Prieto	Telefónica I+D	Project internal review
0.99	23-Jun-09	Alvaro Martínez	Telefónica I+D	General revision following reviewers' corrections and recommendations

Reviewers
Yvon Jégou (INRIA), Ian Johnson (STFC)

Tasks related to this deliverable

Task No.	Task description	Partners involved [°]
T3.6.4	Requirements and specification of advanced services for mobile devices (XtreemOS-G for MD/MP)	TID[*], BSC

° This task list may not be equivalent to the list of partners contributing as authors to the deliverable

* Task leader

Executive Summary

The XtreemOS project aims at integrating Grid functionalities into Linux and addressing the heterogeneity of current Grid computing technology, from clusters to mobile devices (MDs). The main challenge addressed in this document is to provide **advanced features** to the **Grid services** of the XtreemOS-G layer for the **mobile device** flavour, including **smartphone and PDA terminals**[1]**.** The particular characteristics of these devices and the specific mobile device use cases and scenarios will be taken into account. Addressing this heterogeneity and providing advanced services will be a great advantage for XtreemOS. This will bring Grids nearer to users and the XtreemOS' transparent design will be used to open Grid computing to the mass market, while expanding the mobile broadband marketplace.

In previous documents inside this work package, the basic version of the services for XtreemOS-MD (know as XtreemOS-MD layer G) have been defined [1] (identification of requirements and specifications), designed [2] and implemented [3] respectively. This document go a step beyond, focusing on the vision for the advanced version of the services for XtreemOS-MD, complementing the work done in D2.3.5 [4], where the requirements and specifications for layer F of XtreemOS-MD advanced version are identified..

New "advanced" use cases related to the instant messaging and job management basic scenarios already considered in [1] and [2] have been identified (like file sharing for example) and also new scenarios have been identified. More concretely, a possible "Grid player" application for video conversion and playing and also some specific scenarios for mobile phones like offline data access or agenda in the network have been described.

The specific services required for those new scenarios and use cases have been identified as improvements of the existing XtreemOS-G services like AEM or XtreemFS. New requirements for these services have been extracted such as volume sharing or on demand file uploading for XtreemFS, or active monitoring and grid resource reservation for AEM.

Finally, the specifications for XtreemOS-MD services (XtreemFS, AEM and security) have been obtained from the list of requirements. This list of specifications will be the input for the design and implementation of the advanced version of XtreemOS-MD.

[1] The concrete target mobile devices and platforms have been deeply analyzed and evaluated in deliverable D2.3.6. Just as a quick comment, the OpenMoko FreeRunner terminal, the Nokia N8x0 PDAs and terminal for Android platform are, in this order, the ones with highest priority.

Table of contents

List of figures

1 Introduction

The main goal of this deliverable is to define the requirements and specifications of the Grid services which will be offered by the advanced XtreemOS version for mobile devices. The document continues the work that was started in D3.6.1[1] for the basic version of XtreemOS-MD. This previous version has been improved and enriched with new features along with new characteristics taking into consideration the special nature of the mobile devices for which XtreemOS-MD is targeted.

Note that this deliverable is focused on the specification of the advanced Grid services (also known as XtreemOS-G layer). In order to achieve this purpose we have executed the following road map:

- First, we have identified representative use cases where Grid services of the advanced version and mobile scenarios are involved. The use case descriptions, as well as the list of relevant Grid services and their priority are collected in chapter 2.

- The result of the next step is presented in chapter 3: a high level overview of the Grid services which will constitute the advanced XtreemOS-MD version. These services will be used by the applications mentioned in the use cases of chapter 2.

- The description of new features in chapter 3 and the particular needs of the evaluated use cases in chapter 2 are used to define, in chapter 4, a concrete list of requirements for each service. Chapter 4 inlcudes a subchapter for general requirements and another for concrete requirements of API's.

- Chapter 5 provides a specifications list based on the requirements of the previous chapter. This chapter describes how the services must be implemented to fulfil the identified requirements.

Finally, chapter 6 identifies the next steps to accomplish the implementation of advanced services for mobile devices. A final summary about the results of this document is included in chapter 7.

2 Advanced use cases

In order to define the requirements of XtreemOS-MD advanced services, we first analyze some interesting scenarios and reference applications that could benefit from XtreemOS-MD. We then cover the different use cases that will be used to extract the requirements.

2.1 Grid Player Scenario

The "Grid Player" is one of the possible scenarios that could benefit from XtreemOS-MD. Grid Player would be a video player including codec and resolution conversion to the optimal values for the target device. For example, a video in DIVX format [5] at 720p resolution is in general too demanding for a mobile device limited to other video codecs and lower screen resolutions.

Figure 2-1 Grid Player application running on Mobile Device

There are already some video converters in the market, but in general the mobile devices are not powerful enough to execute those applications. The idea is to run the corresponding converters in the Grid, and just send back to the mobile device the converted videos, either by streaming or simply downloading the corresponding file.

2.1.1 Conversion of "local"[2] videos

Description

A user launches the Grid Player application from his mobile device, searches for a video in the XtreemFS user's volume and selects to convert it to the optimal codec and resolution for the MD used.

The converter application will then be executed in the Grid as a new job and the resulting video will be stored inside the XtreemFS user's volume (in the folder specified by the user)

Grid services required

- Access to XtreemFS in read/write mode.
- Access to AEM services.

[2] "Local" means here (and in every use case from this scenario) "inside the user's volume", as we are considering a Grid environment. Our focus is not the conversion of videos stored inside the MD; anyway, it might be possible to store the converted video also inside the MD, in order not to need to be connected to play it.

Priority
High.

2.1.2 Conversion of "remote" videos

Description
A user launches the Grid Player application from his mobile device, searches for a video in the Internet or in the file system of a **different** user and decides to convert it to the optimal codec and resolution for the MD used.

The converter application will then be executed in the Grid as a new job and the resulting video will be stored inside the XtreemFS file system (in the folder specified by the user). In this case, either the converter application is able to work with remote files or the video is first saved (it could imply downloading it firstly) into the XtreemFS file system.

Grid services required
- Access to XtreemFS in read/write mode.
- Access to AEM services.

Priority
Medium.

2.1.3 Playing local videos (streaming)

Description
A user launches the Grid Player application from his mobile device, searches for a video in the file system and decides to play it, but optimizing the codec and resolution for the particular MD used.

The converter application will then be executed in the Grid as a new job and the content of the resulting video will be sent via streaming[6] to the mobile device in real-time.

Grid services required
- Access to XtreemFS services in read mode.
- Access to AEM services.

Priority
High.

2.1.4 Playing remote videos (streaming)

Description
A user launches the Grid Player application from his mobile device, searches for a video in the Internet or in the file system of a **different** user and decides to play it, but optimizing the codec and resolution for the particular MD used.

The converter application will then be executed in the Grid as a new job and the content of the resulting video will be sent via streaming to the mobile device in real-time.

Grid services required
- Access to AEM services.

Priority
Medium.

2.1.5 Send invitations to other users

Description
A user launches the Grid Player application from his mobile device, searches a video and sends an invitation to another user to play this video. The user may be introduced manually or selected from the contact list. The notification (invitation) may be sent via SMS or using other alternatives like instant messaging that will be analyzed.

If the video is remote, no special operation is needed except including the URI of the video in the notification. If the video is a local video stored in the user's volume, it will be needed to perform some kind of file sharing, either by transferring the file to the other user's volume, or modifying the permission associated with this file in order to let the other user access that volume.

Grid services required
- Access to XtreemFS and file/volume sharing.

Priority
Medium.

2.1.6 Preview visualization

Description
A user launches the Grid player application from his mobile device, decides to convert a video to the optimal codec and resolution for his MD and finally he also selects the option "watch preview" from the menu. The conversion process, which runs in the Grid, generates a short preview of the video with the new codec and new video resolution. This file will be sent to the user via streaming showing the conversion result in advance.

After watching the preview, the user will be able to decide whether to continue, to change some conversion parameters or to cancel the conversion.

Grid services required
- Access to AEM services.

Priority
Medium/Low.

2.2 Instant Messaging Scenario

This section covers use cases for the Instant Messaging scenario that will be compliant with the advanced version of XtreemOS-MD. These use cases take as scenario a basic IM application and include a short description of XtreemOS services required and their priority.

2.2.1 File Sharing

Description

Users of the same VO must be able to share files through IMA. i.e., an option like “share folder with this user” should exist that automatically shows a directory where a user could get/put files from/to other user.

Grid services required

- Access to XtreemFS

Priority:
Medium.

2.3 Job Management

This section describes the use cases related to the capacity of the XtreemOS–MD for monitoring jobs of the Grid and showing the information to the user which are going to be implemented in the advanced Xtreemos-MD version. It includes the list of Grid services required and their priority.

2.3.1 Manage Grid jobs

Description

A user from XtreemOS-MD should be able to manage Grid jobs. For this purpose, taking into account the way of working with MDs and especially with mobile phones, it will be very convenient to offer a graphical user interface (the reference application called JobMA specified in [7], is precisely designed to offer a job management GUI to MD users), allowing the following operations:

- **Launch Jobs**: the user will be able to select the desired job description file with JSDL[8] format and send it to the Grid. The job description file may be located on the XtreemFS.
- **Monitor Jobs**: the user will be able to request information ***on-demand*** about the status and details of the jobs which belong to him or to another user.
- **Active Monitor**: the user will be able to see ***in real time*** the current status and details of his own jobs. This feature allows users to be informed about a job conclusion.
- **Stop Jobs**: the user will be able to stop the execution of an active job.
- **Resume Jobs:** the user will be able to continue the execution of a stopped job.
- **Cancel Jobs:** the user will be able to cancel the execution of an active job.
- **Managing job trees:** The user will be able to manage jobs that belong to a dependency tree. The job trees may be launched, stopped, resumed and cancelled in a similar way to individual jobs.

Once the user is logged into the Grid and his XOS-Certificate is loaded onto the system, he can select a job description file with JSDL format which can be located on the local file system or on the XtreemFS.

When the user confirms the operation, the system will submit to the Grid the task described by the previous file. During the job execution, each important change of job status (i.e.: *active-finished*,) that occurs will be notified to the user by XtreemOS-MD.

Also, the user will be able to operate on the jobs: request information, cancel, resume and stop jobs previously launched to the Grid.

Grid services required

- Use of credentials: XOS-Cert obtained from Credential Distribution Agency (CDA).
- Access to XtreemFS system on the MD.
- Access to AEM system on the MD, using XATICA.

Priority
High.

2.3.2 Manage reservations

Description
An XtreemOS-MD user must be able to manage the resources reservations from a graphical user interface in the mobile devices. This feature allows users to reserve Grid resources with specific hardware characteristics for a period of time defined by a start date and finish date.

Grid services required

- Use of credentials: XOS-Cert obtained from Credential Distribution Agency (CDA).
- Access to AEM system on the MD, using XATICA.

Priority
High.

2.4 Mobile phone specific scenarios

2.4.1 Networked Agenda for my mobile phone

Description
A user of a mobile device phone could take advantage of XtreemOS by having most of her configuration data stored in the network. This feature should be particularly useful for those users willing to store her contacts list on the network. The user should be able to automatically download her contact list to her mobile phone at any time. This feature may be developed by using XtreemFS as a personal repository.

Grid services required

- Use of credentials: XOS-Cert obtained from Credential Distribution Agency (CDA).
- Access to XtreemFS system on the MD.

Priority
Medium.

2.4.2 Offline Data Access

Description
Due to the nature of connections of mobile devices it is common to lose connectivity that may cause problems in applications that are using these connections. For that reason, user might have an offline mode for data management.

In this scenario a user could set a "work offline" mode for some files and work on them without network connection. When the work is ended, user could unmark this mode and files will be sent.

Grid services required

- Access to XtreemFS services

Priority: Low

2.4.3 Interactivity with legacy SSO's

Description

A user of XtreemOS-MD may be authenticated simultaneously in their organization and in a VO of XtreemOS, only by executing one step that will be transparent to the user. This feature will be offered to the user even if the authentications mechanisms are different. The CDAProxy will be in charge of this feature.

For example: the user *A* that belongs to *Corporation-ABC* is a member of the *VO-123* (a Virtual Organization of XtreemOS) and uses Xtreemos-MD in his mobile. He will be able to authenticate his user ID in the network of *Corporation-ABC* and automatically he will be authenticated in the *VO-123* (XtreemOS VO).

Grid services required

- VO management.

3 Advanced version of XtreemOS-MD Services

Once we have identified our use cases, this chapter will provide specific services for the advanced version of XtreemOS-MD. These services will be improvements of the existing XtreemOS-G services such as AEM, XtreemFS and VO management APIs. These services come from the PC flavour, so an analysis of implications in the MD flavour is included for each service.

3.1 Application Execution Management

The AEM service for the advanced version adds the following features to the basic version:

3.1.1 Reservations

Description: applications/users may want to guarantee that the needed resources will be available when the application is to be run. For this reason, XtreemOS allows users/applications to book a set of resources according to a JSDL description and then use it to run jobs on them. It is important to notice that a reservation is not linked to a job, but can be used by several jobs. It is up to the user/application to decide which jobs use which reservations.

Implications in the MD-flavour: All the client interface and managing of the reservations will need to be ported to the MD-flavour.

3.1.2 Job execution in multiple resources

Description: In the basic version of XtreemOS a job could only run on a single resource (with as many processes as desired). In the advanced version, a job can specify that more than one resource is needed and specify the resource characteristics.

Implications in the MD-flavour: The entire client interface and the specification of multiple resources per job have to be prepared in order to be able to get the information from multiple resources when requesting job monitoring information.

3.1.3 Dependencies

Description: In order to allow third party implementation of a workflow manager, the advanced version of XtreemOS offers the dependency concept. The idea is that any job may add a dependency to any other job (given that it has the right security credentials). These dependencies can then be used by the jobs themselves, or by the monitoring infrastructure to take group actions on all the jobs in a dependency tree, or start jobs in a dependency tree after another job has ended (notified by SIG_CHILD signal[9]). A given job may belong to many dependency trees because each dependency tree will be tagged by the creator.

Implications in the MD-flavour: All the client interface and managing of dependencies will need to be added. In addition, monitoring infrastructure will also be affected because now the user can request monitoring information for a dependency tree.

3.1.4 Monitoring

Description: In the advanced version of XtreemOS, the monitoring will have new features such as detailed on-line information of job status, user events, buffered events (to avoid constant pulling from the user/client side) and call-backs[3].

Implications in the MD-flavour: All the client interface and managing of these new monitoring features will need to be implemented. We have to place special focus on the call-backs, because this kind of notification will mean a significant technical effort in a mobile environment.

3.1.5 Interactive jobs

Description: So far, in the basic version all jobs had to be batch jobs because there was no way for them to interact with the user. This limitation will be removed in the advanced version and the user will be able to connect/disconnect from a job using the SSH master mechanisms.

Implications in the MD-flavour: As this mechanism will be implemented using characteristics of some SSH implementations, the MD-flavour will need to find the adequate SSH for mobiles and/or fins a workaround for it.

3.1.6 Fault tolerance

Description: In the advanced version of XtreemOS, both the AEM server and the applications will be tolerant to failures.

Implications in the MD-flavour: This feature should be completely transparent to the MD-flavour because it basically affects CORE and RESOURCE nodes and services.

3.2 Data Management

The XtreemFS adds the following features to the basic version:

3.2.1 File replication using Vivaldi algorithm to locate clients

Description: In the advanced version of XtreemOS, all nodes involved in file management/usage will be placed in a 2D space (i.e. will have a pair of <X,Y> coordinates). These coordinates will be computed using the Vivaldi[10] mechanisms (slightly modified) where the distance between two points in the space reflects the latency between the two nodes. This coordinates will be used to decide what replica is the fastest one to be used by any client.

Implications in the MD-flavour: This feature has two main implications. First, the MD, as any client, will need to maintain these <X,Y> coordinates. This implies, as long as file systems are mounted, some communication (every few minutes) with one of the file servers it is currently using. This has the problem of extra communication that might be significant for a mobile device. In addition, MD are less stable as far as location is concerned, thus we will need to find solutions to this high mobility. Second, the client interface will need to be modified to include this selection of replicas based on the coordinates of the MD and the coordinates of all OSDs[11] holding a replica of the file.

---

[3] The pull and buffered pull possibilities are already supported by AEM, as indicated respectively in WP3.3 deliverables D3.3.1 and D3.3.5, while the last option (push) is a specific request (still not supported) launched from the MD side and coming from a OG Proposed Recoomendation reflected in http://www.ogf.org/documents/GFD.90.pdf

3.2.2 Caching in kernel

Description: In the basic version of XtreemOS, all caching is done at user level (i.e. in the FUSE client). Nevertheless, it's planned that in the advanced version, all this user level caching will be inserted in to the kernel page cache.

Implications in the MD-flavour: To offer this feature, the kernel used for the MD may need to be changed. Nevertheless, the real implications are still unknown because the final solution from the file system team has not been defined yet.

3.2.3 Offline mode

Description: It's planned that the advanced XtreemFS version will support offline mode, which allows the user to specify which files will be used when a client device does not have connection to the Grid nodes. The nodes will need to maintain information about offline files using some locking mechanisms. The client applications will have an interface similar to SCM[12], which allows files to be locked (after downloading them) and committing data when work is done (when uploading).

3.3 VO management and security services

3.3.1 Single Sign On support for mobile devices

Description: The XtreemOS MD must offer a service of Single Sign On[13] that is especially important in mobile device, given that the keyboards and displays of these types of devices are not appropriate to write login/password easily. So an SSO system allows users to simplify the authentication step.

The advanced XtreemOS version provides a SSO service based on User Session Service (USS) which multiplexes connections inside just one persistent and authenticated connection with a remote server, the *Session Manager Daemon* (SMD). New connections are multiplexed over the existing connection, not needing a new authentication. Applications connect with USS, which either uses SMD to connect with destination services or uses a modified *openSSH* tool.

Implications in the MD-flavour: In mobile devices, usually *dropbear* is included instead of *openssh*, because is considerable lighter. But "control master" is an *openssh* specific feature, not implemented in *dropbear*[14]. According to the idea of using a unique solution for XtreemOS and XtreemOS-MD, the SSO mechanisms included in the advanced XtreemOS version will be migrated to the mobile device version. Some modifications will be done to fit the SSO service in accordance with mobile devices characteristics.

At time of writing this document, the final SSO design has not been completed, so the MD version of SSO has not been closed yet.

4 Requirements

In the previous chapters, we have studied the uses cases which have been defined in order to identify the required functionalities and needed services for the advanced version of XtreemOS-MD. Now a provision and description of the concrete requirements the services must carry out is included in this section.

To define the requirements, first of all, each requirement has been identified with a number *R3.6.x* that continues the established numeration for requirements of D3.6.1 [1]. Next a description of the requirements is included. Finally we have indicated the origin of each requirement (focusing on advanced use cases) and the requirement importance to define a specific order of priority for the implementation.

4.1 General requirements for advanced services

R3.6.28: Interactive Applications

XtreemOS-MD must provide mechanism to support the interactivity *in real-time* between the users and the jobs running in the Grid. The user will be able to connect with jobs using the standard input/output, X-windows, mouse events, etc... The feature implementation will be based on SSH master mechanisms. The used mechanism will be able to operate properly with data flows for real-time applications (i.e.: streaming)

Comes from: Use Cases, D 3.6.4 (section 2.1)
Importance: Medium

4.2 Requirements for advanced application execution management (AEM) services

R3.6.29: Active monitoring

XtreemOS-MD must offer a monitoring service that allows the users to see the current details of their jobs *in real-time.* The requested information for jobs [15] should include: job id, job status, owner, submit time, submit node, reservation id and also exit status when applicable. Besides, for every process in the job it should provide its status and user/system time consumed. This service must support several clients requesting job information at the same time.

Comes from: Use Cases, D 3.6.4 (section 2.3)
Importance: Medium

R3.6.30: Buffering events

The service must offer the possibility of creating buffers in the AEM server in order to store users and/or jobs events. The buffer size must be defined by the users (3rd party applications) as well as the events type that will be stored in it. The buffer implementation will be focused on keeping the most recent events.

Comes from: Use Cases, D 3.6.4 (section 2.3)

Importance: Medium

R3.6.31: Asynchronous event notifications

XtreemOS-MD must be able to notify jobs events to 3rd party applications in asynchronous mode; these applications act as events listeners to jobs finalization, jobs errors, or any other special notification. For example an application could subscribe to events of the form "metric A changed its value to X", such as "job changed its status to Running" and thus be notified whenever the job is executed. Another use case would be the notification of buffer almost full when metric buffering is active, so that the user can retrieve every value, not losing data nor having to poll the system periodically.

Comes from: Use Cases, D 3.6.4 (section 2.1)
Importance: Low

R3.6.32: Format of job information

XtreemOS-MD must return monitoring information of the user's jobs. This information will be formatted in XML files and in data objects (i.e. DataBeans of JAVA, structs of C, etc.). The details of both methods (files and objects) must be well known and documented for applications developers.

Comes from: Use Cases, D 3.6.4 (section 2.3)
Importance: Medium

R3.6.33: Grid Resources Reservation

XtreemOS-MD must be able to offer a reservation mechanism for Grid resources using JSDL files[4]. The reservations must contain as minimum start date, finish date, number and type of nodes. To use the reservation service, it must be possible from XATICA using calls to specific methods of the API.

Comes from: Use Cases, D 3.6.4 (section 2.3.2)
Importance: Medium

R3.6.34: Dependencies among jobs in different resources

XtreemOS-MD must provide a job execution control system that allows users to create dependencies among jobs executed in different resources[5]. So a complex job could be divided into some simple jobs that will run in different resources without synchronization problems. Hence, the XATICA API must offer the necessary methods to create, delete, remove and consult the dependencies among the jobs that belong to a specific dependency tree.

Comes from: Use Cases, D 3.6.4 (section 2.1.6)
Importance: Medium

[4] Already supported by AEM, as specified in WP3.3 deliverables (concretely in D3.3.3 and D3.3.4)
[5] Already supported by AEM, as specified in WP3.3 deliverables (concretely in D3.3.1, D3.3.3 and D3.3.4)

4.3 Requirements for advanced data management services

R3.6.35: Volume sharing

XtreemFS must allow two users to mount the same volume and ensure that information stored in this volume remains synchronized between both users.

Comes from: Use Cases, D 3.6.4 (section 2.2)
Importance: Medium

R3.6.36: File replication using Vivaldi algorithm

When XtreemFS client requests the data from a node it must be from the nearest node. The distance from a node must be calculated using Vivaldi algorithm when this information is available. If not, the node will be selected using algorithm from basic version of XtreemFS.

A mobile device may be moving between different networks (Wifi, 3G, different cells, roaming) and the "Vivaldi position" should be recalculated when those movements cause a network change. The Vivaldi algorithm willl then be applied and the result will be an <X, Y> position for any mobile device location and the nearest node (in Vivaldi distance terms) at any time.

Comes from: Use Cases, D 3.6.4 (section 3.2.1)
Importance: Medium

R3.6.37: Caching in kernel

Many Linux functionalities assume that data is kept in the kernel cache and thus a *mmap* can be done. In the current version of XtreemFS, the cache is kept in the FUSE-user space and thus *mmaps* cannot be implemented. In the advanced version for XtreemFS this problem will be solved by using the kernel page-cache, and thus this will need to be implemented in the MD version too.

Comes from: Use Cases, D 3.6.4 (section 3.2.3)
Importance: Low

R3.6.38: Offline mode

XtreemFS should provide an offline work mechanism. Files should be locked and unlocked from client applications in order to avoid other client applications modifying those files meanwhile they remain locked. Client applications could work on locked files safely without need of a permanent connection to a Grid node. Once that user finishes working with these files, the client application should allow uploading and unlocking those files easily.

Comes from: Use Cases, D 3.6.4 (section 2.4)
Importance: Low

R3.6.39: On demand file uploading

XtreemFS should provide a lazy upload mechanism which allows uploading files from the MD only when other nodes request the file contents. This affects only MDs, so it will be implemented in the XtreemOS-MD version. In other words, XtreemFS should implement a system similar to Samba/NFS, where a user marks files to share without the need of uploading them to a server. This is especially important in the mobile implementation due to bandwidth constraints and 3G traffic cost.

Comes from: Use Cases, D 3.6.4 (section 2.4)
Importance: Low

R3.6.40: Transparent file sharing

XtreemFS should provide a mechanism to implement the file sharing described in R3.6.39: "On demand file uploading" in a transparent way from the users perspective. This requirement will improve the mobile device user experience taking into account the interface limitations inherent in MDs.

Comes from: R3.6.39: **'**On demand file uploading'.
Importance: Low

R3.6.41: Auto-mounting

XtreemFS should provide auto-mounting mechanisms: when a user access a local folder linked to XtreemFS volume, this volume should be mounted automatically, without user action. I.e an application saves some configuration files inside a folder and the user wants that these files are stored in an XtreemFS volume to use this configuration in other terminals. The user may specify in the XtreemFS configuration that a local folder is linked with a xtreemFS volume and when the application access to the file, xtreemfs automatically mounts the volume providing files to the application in a transparent way.

Comes from: Use Cases, D 3.6.4 (section 2.4)
Importance: Low

R3.6.42: Manage IP address changes

Mobile devices are exposed to connectivity losses, IP address changes and other network errors not directly managed by the OS. XtreemFS should manage those network problems, especially IP changes, in a transparent way from the users perspective, at least whenever is possible to automatically recover from those situations.

Comes from: Use Cases, D 3.6.4 (section 2.4)
Importance: Low

4.4 Requirements for VO management and advanced security services

R3.6.43: Interaction with other SSO's

XtreemOS should provide a SSO system that facilitates the interaction with other SSO's in order to handle the VO authentication from the XtreemOS-SSO maintaining the user authentication with the legacy SSO. For example, to guarantee a real integration with legacy SSO's, XtreemOS-MD must offer authentication against LDAP's.

Comes from: Use Cases, D 3.6.4 (section 2.4.3)
Importance: Low

5 Specifications

The current section contains the detailed descriptions of the necessary operations that the Grid services will be able to do in order to fulfill the requirements mentioned in the previous chapter.

To define the specifications, first of all, each specification has been identified with a number *S3.6.x* that continues the established numeration for specifications of D3.6.1 [1]. Next a description of the operations is included. Finally we have indicated the origin of each specification focusing on requirements.

The specifications have been divided into categories following the same structure as the requirements chapter.

5.1 Specifications for advanced application execution management (AEM) services

S3.6.16 Creation of reservations

Description: When MD user wants to run jobs using multiple resources or at a given time, it needs to create a reservation using a JSDL to describe the characteristics of the resources needed (number and type of machines, memory, input and output files, etc). Once the reservation is created, a reservation ID will be available to be used when creating jobs on top of this reservation.

Comes from: R3.6.33

S3.6.17 Job submission using an already created reservation

Description: Once a reservation has been created, MD users can use them to run jobs on top of these reservations. This is the main mechanism to run jobs using more than a single resource. They will need a JSDL describing the job and the reservation ID.

Comes from: R3.6.33

S3.6.18 Create/remove dependencies tree among jobs

Description: MD users have to be able to dynamically add dependencies among the jobs. To create a dependency, the user will need to specify what job depends on what other job plus a tag to identify the dependency tree.

Comes from: R3.6.34

S3.6.19 Connect/disconnect to/from a running interactive application using the SSH mechanism

Description: The way interactive applications are run in XtreemOS is that when the user wants to interact with the application, it connects to it though SSH, then uses the SSH tunneling to interact with the application. This option should also be a possibility from MD.

Comes from: R3.6.28

S3.6.20 Retrieve monitoring information for a job

Description: The monitoring information has increased since the basic release and MD applications should be able to retrieve all the available monitoring information (in XML format). This information can be requested polling the job manager every time the user wants an update.

Comes from: R3.6.29, R3.6.32

S3.6.21 Retrieve monitoring information that has been buffered (several events of the same type packed in a single request)

Description: In order to reduce the number of times MD connects to the job manger to get monitoring information, the job manager maintains a set of buffers with the last events of each type (as long as it has been specified to be buffered by the application). MD devices should be ready to handle monitoring information where more than one event of a type is returned by the job manager.

Comes from: R3.6.29, R3.6.30

S3.6.22 Receive event notification when some specific event occur

Description: Again, to reduce the number of time MD connects to the job manger for monitoring information, the job manager may send an event when a given monitoring event occurs. MD should be ready to receive these monitoring events and pass them onto the application "subscribed" to them.

Comes from: R3.6.29, R3.6.31

S3.6.23 Retrieve dependency tree among jobs

Description: It might be important when displaying the information of a job and its related ones to know the dependency tree (for a given tag). This tree should be obtained by the MD by contacting the job manager of the root job.

Comes from: R3.6.34

S3.6.24 Retrieve monitoring information for a group of jobs following the dependency chains

Description: Many applications are made by several jobs, thus it might be interesting for monitoring of all these jobs at a time. This can be done using dependencies and requesting the monitoring information to the jobs in the dependency tree.

Comes from: R3.6.29, R3.6.34

5.2 Specifications for advanced data management services

S3.6.25 Place MD clients in the Vivaldi 2D space

Description: In order to be able to use file replicas efficiently, we need to place the MD in a 2D space to be able to decide the closest replica to our MD. Computing this coordinated implies a periodical communication between the client and some OSDs. In addition, it needs to be able to react to frequent

and large movements because MD will tend to be much less stable (in a single location) than the rest of resources in XtreemOS.

Comes from: R3.6.36

S3.6.26 Use the Vivaldi coordinates as a parameter to choose the closest replica in the system

Description: Once the MD has its own coordinates, it needs to be able to use them to decide the best replica according to its current location.

Comes from: R3.6.36

S3.6.27 Allow file system caching at kernel level

Description: In the basic version of XtreemFS, all caching is done at user level in the FUSE demon. This is not the desirable solution because MMAPS cannot be implemented using XtreemFS files. For this reason we need to be able to cache objects in the kernel cache and be able to manage them from the FUSE client.

Comes from: section 3.2.2

S3.6.28 Allow access to files while offline

Description: MD will be offline much more frequently than other resources in XtreemOS, for this reason it is desirable to allow the user to specify which files it wants to access offline. These files should cached in the MD and used while offline. Any modification will be updated once the device in on-line again.

Comes from: R3.6.38

S3.6.29 On demand file uploading

Description: MD should be able to share files with the rest of the system without having to download unless the file is actually viewed to reduce the cost of downloading files that are never seen. This is like converting the MD is a **light-OSD** (with very limited functionality)

Comes from: R3.6.39, R3.6.40

S3.6.30: Changing IP addresses

Description: MDs are constantly moving, thus the IP address of a device may change over time (even within a give mount session). This means that XtreemFS will need to be able to recognize the client regardless of the IP address that stops being an ID.

Comes from: R.3.6.42

S3.6.31: Auto-mounting

Description: XtreemOS-MD might implement an auto-mounting mechanism, similar to the system provided by *autofs*: XtreemFS volume will be mounted automatically on-demand, when an application tries to access a file under the mount point directory.

The configuration file would include an option "on-demand mounting" to specify if the volume will be either mounted on-demand or when the user starts a XtreemOS session.

Comes from: R.3.6.41

5.3 Specifications for VO management and advanced security services

S3.6.32: SSO interactivity

Description: *CDAProxy* (explaind in [3] and detailed in [16]) works as an interface between the user and the *CDAServer* executed by the VO. *CDAProxy* may be executed to authenticate users in the *real organization* using its own methods and at the same time authenticating users in the v*irtual organization* following the VO methods. In short the CDAProxy must offer both authentications: in the real and the virtual organization.

Comes from: R3.6.43

S3.6.33: Supporting Authentication methods

Description: PAM modules[17] will be added to the CDAProxy to extend the list of authentication methods supported.

Comes from: R3.6.44

6 Future work

Once the concrete requirements and specifications have been defined for the Grid services of the advanced XtreemOS-MD version, the following step is to design the aforementioned services. This process will be focused on:

- The Grid services components will be described and detailed as well as a description of interactions among them and other XtreemOS components.

- The design will take into account that the services implemented for the advanced XtreemOS-MD version will be used by applications such as the IMA, the JobMA and the Grid Player.

- In parallel with this work, first tests with the components in a mobile device will be done, to detect and prevent possible issues that could appear in the implementation of the services or applications.

The design tasks will be reflected and described in the deliverable: D3.6.5: Design of advanced services for mobile devices.

7 Conclusions

The aim of this document is to be the first step for the implementation of Grid services in the advanced XtreemOS-MD version. Part of this implementation will be based on the XtreemOS general version 2.0 (PC flavour).

The use cases, which have been chosen as representative examples, are focused mainly on AEM XtreemFS and the VO management services. These enhanced services provide a set of new features that improve the basic version and allow developers to design more advanced Grid applications such as JobMA, IMA and the Grid Player.

The main features of this advanced version can be summarized in the following list:

- **AEM** will be able to reserve Grid resources, to create dependencies among jobs and to provide fault tolerance mechanisms, thus allowing to run interactive jobs, active monitoring of jobs and job execution in multiple resources. The respective specifications define *how* the AEM must be implemented to offer those features.

- **XtreemFS** will offer an offline mode and a caching in kernel features. In addition, the service will be improved by adding the Vivaldi algorithm that allows to decide which file replica of XtreemFS is the best to be used by the client in terms of the Vivaldi metrics.

- **VO management** will able to guarantee the interoperability among the XtreemOS-SSO and legacy SSO's of 3rd party organizations.

These new features will be added to the existing basic version of AEM, XtreemFS and VO.

Given that they will be used by the advanced applications for XtreemOS-MD, the Grid advanced services have been taken into account in this deliverable to define the requirements and specifications for XtreemOS-MD.

The implementation of advanced applications will be done in the task: *T3.6.6. Implementation and optimization of advanced services in mobile devices* but the first approach and the high level description have been included in this document.

8 Bibliography

[1] XtreemOS Consortium. *Requirements and Specification of Basic Services for Mobile Devices* D3.6.1. Integrated Project, November 2007.

[2] XtreemOS Consortium. *Design of Basic Services for Mobile Devices* D3.6.2. Integrated Project, May 2008.

[3] XtreemOS Consortium. *XtreemOS-G for MDs/PDA* D3.6.3. Integrated Project, December 2008.

[4] XtreemOS Consortium. *Requirements and Specifications for Advanced VO Support in Mobile Devices* D2.3.5. Integrated Project, November 2008

[5] http://en.wikipedia.org/wiki/DivX

[6] http://service.realnetworks.com/help/player/plus_manual.8/htmfiles/whatisrpp.htm

[7] XtreemOS Consortium. *Early Experiments and Evaluation* D4.2.2, November 2006.

[8] GFD-R.056: *Job Submission Description Language (JSDL) Specification, Version 1.0* http://www.gridforum.org/documents/GFD.56.pdf

[9] http://linux.about.com/od/commands/l/blcmdl7_signal.htm

[10] Frank Dabek, Russ Cox, Frans Kaashoek, Robert Morris. *Vivaldi: A Decentralized Network Coordinate System*.In the Proceedings of SIGCOMM 2004, Portland, OR, Aug, 2004.

[11] Felix Hupfeld1, Toni Cortes, Bj¨orn Kolbeck1, Jan Stender, Erich Focht, Matthias Hess, Jesus Malo, Jonathan Marti, Eugenio Cesario. *The XtreemFS architecture a case for object-based file systems in Grids*. In CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE Concurrency Computat.: Pract. Exper. 2008; 8:1–12.: http://www.storagebox.org/private/felix/hupfeld_casefs_journal_online.pdf

[12] Wheeler, David. *Comments on Open Source Software / Free Software (OSS/FS) Software Configuration Management (SCM) Systems*. http://www.dwheeler.com/essays/scm.html Retrieved on May 8, 2007.

[13] Yvon Jégou. Credential Management and Security Architecture in XtreemOS: Proposition for Single-Sign-On and Delegation. XtreemOS Technical Report.

[14] Dropbear: http://matt.ucc.asn.au/dropbear/dropbear.html

[15] XtreemOS Consortium. Job Monitoring Description D3.3.5. Integrated Project, November 2008.

[16] CDAProxy Manual in XtreemOS Consortium repository,

https://gforge.inria.fr/plugins/scmsvn/viewcvs.php/distribution-mobile/doc/CDAProxy.pdf?root=xtreemos

[17] http://www.kernel.org/pub/linux/libs/pam/

A. Appendix: Acronyms and abbreviations

AEM	Application Execution Management
FUSE	Filesystem in Userspace
IMA	Instant Messaging Application
JSDL	Job Submission Description Language
LDAP	Lightweight Directory Access Protocol
MD	Mobile device
MMAP	Memory Mapped
OSD	Object Storage Devices
SCM	Source Code Management
SMS	Short Message Service
SSH	Secure SHell
SSO	Single Sign-on
VO	Virtual Organization
XATICA	XOS Application Toolkit Interface C Adaptation
XtreemFS	XtreemOS File System