XtreemOS-MD: Grid Computing from Mobile Devices

Alvaro Martínez[1], Santiago Prieto[1], Noé Gallego[1], Ramon Nou[2], Jacobo Giralt[2], and Toni Cortes[2]

[1] Telefónica I+D, Spain
{amr,spm,e.xtreemos-ngm}@tid.es
[2] Barcelona Supercomputing Center, Spain
{ramon.nou,jacobo.giralt,toni.cortes}@bsc.es

Abstract. Grid and Cloud computing are well known topics, and currently one of the focus of many research and commercial projects. Nevertheless, the transparent access to Grid facilities from mobile devices (like PDAs or smartphones) is normally out of the scope of those initiatives, taking into account the intrinsic limitations of such devices: screen size and resolution, storage capacity, computational power, etc.

To provide an integrated solution for mobile access to Grid computing, aspects like Virtual Organizations (VOs) support, graphical job management, flexible authentication and access to the file system user's volume in the Grid from a mobile device should be covered. That is the purpose of XtreemOS-MD, the mobile flavour of XtreemOS - a Linux-based operating system to support VOs for Grids -, the transparent access to Grid facilities from mobile devices.

Keywords: grid, cloud computing, mobile device.

1 Introduction

The Grid [10], as a distributed computational system that allows sharing non-centralized resources to solve a single problem requiring a very significant computational power or storage capacity, is a relatively old and well known concept. Grid systems have been used specially to solve complex scientific and technical problems [17]. Middleware solutions like the Globus toolkit [9], the most extended solution, or others like Condor [26], Legion [15] or UNICORE [7], have classically been used to implement the Grid architectures. But those solutions are designed as a "sum of services" infrastructure, where tools are developed independently [12][13][14], and the lack of common programming interfaces or unified models of component interaction can impact negatively in the user-friendliness. The final consequence is that a certain level of experience and technology knowledge is required to effectively use the Grid. Precisely, XtreemOS [29] was designed to become a new Grid operating system allowing transparent and native access to the Grid environment, and then simplifying in big manner the use of Grid systems.

Y. Cai et al. (Eds.): Mobilware 2010, LNICST 48, pp. 45–58, 2010.

A more recent concept is the cloud computing, referring to a dynamically scalable style of computing, where virtualized resources are provided as a massively scalable service over the Internet [25]. Those systems can be implemented using a centralized infrastructure of servers, but also Grid platforms could be used to provide the virtualized resources forming the cloud. Commercial solutions like Amazon EC2 [1] or Google AppEngine [16] are already exploiting this idea, and other big players are currently launching their own "cloud" solutions, like Microsoft with the Windows Azure [19] platform.

But these distributed systems, Grids and Clouds, are not only useful for executing complex tasks, but also they are useful for simpler tasks, but still too complex for certain limited computational-power devices, like PDAs or smartphones. Despite the fact of their intrinsic computing limitations, the integration of an increasing number of mobile consumer devices into the Grid has been considered as potentially beneficial to the computational Grid [24]. Moreover, we believe that it opens the way for new application scenarios. For instance, a mobile device multimedia player application may use the Grid to transparently transcode any video file so that it is adapted to its media capabilities, or the Grid might be used to quickly calculate the next move of your opponent in a mobile device chess game application or, simply, the user's device can take benefit of an "unlimited" storage capability in the Grid when recording or playing multimedia content. But most importantly, all those scenarios can be implemented in a native and transparent way, so that the user does not even know that there is a Grid infrastructure behind.

1.1 State of the Art

The inclusion of mobile devices into Grid computing platforms has followed two different research lines: the first one adds mobile devices as simple Grid clients and the second tries to include them as Grid resources, extending the capabilities of the Grid thanks to the near-unlimited number of resources shared. The first approach has been followed by some Grid frameworks [20], first by just providing simple and very limited web user interfaces to mobile devices with Web proxies [8] and, one step further, by developing specific mobile device applications downloadable from Grid portals [22], either the applications being written in Java or using other development frameworks [4]. The second approach has being tackled by several initiatives, usually trying to create Grids by the simple combination of mobile devices as computing resources, essentially imitating the common Grid services and scenarios. Several architectures have been proposed, like a Proxy-Based clustered architecture [24], or more ambitious ones under the general term of "Mobile Grid Computing" [28] or even others proposing a "beyond the device as portal" model [3]. However, as far as we know, actual implementations of these approaches for mobile phones are almost inexistent. Moreover, even if this latter approach could gain acceptance with the increase of performance and storage capacity experimented in the smartphones world, issues like low battery power, lack of security and confidentiality, etc. make us think in a different kind of resource sharing, focused on network and input/output devices (GPS, camera, microphone,etc.) and data sharing, rather than computing power.

By contrast, our approach, which will be detailed below, has allowed us to implement a Grid layer providing: native access to distributed Grid file systems, seamless job execution and management in the Grid from the mobile, and a complete and secure integration with Virtual Organizations (VOs). In addition, sharing of specific resources of a mobile device is also possible. Moreover, the solution has been implemented and tested with real mobile Linux-based devices.

1.2 Transparent Mobile Grid Computing

Precisely the main focus of this paper is about the possibility of transparently and natively using a Grid infrastructure from a mobile device, to benefit from Grid facilities but hiding its complexity to the users. We believe that these goal can only be totally achieved by integrating a Grid middleware into the mobile device operating system and by avoiding portal or proxy-based only models that basically limit the potential of Grid computing for mobile devices. In addition, the sharing of mobile device resources to the Grid may be easily achieved thanks to the possibility of having native access to device resources. To this purpose, we will present XtreemOS-MD, the version for mobile devices of the XtreemOS project, a Linux-based operating system designed to work with Grids in a transparent way. XtreemOS-MD tries to cover the two research lines mentioned: the universal access to the Grid services and also the sharing of resources provided by the mobile devices, like their network capabilities, GPS and the content stored locally in the mobile device (photos, videos, etc.)

2 XtreemOS-MD General Architecture

As commented in the introduction, the approach selected by the XtreemOS project, and followed as well by its mobile version -XtreemOS-MD, is based on providing a Grid native support inside the operating system.

XtreemOS was born to provide the needed abstraction from the hardware and software and to secure the resources shared between different users, let's say what a traditional operating system provides, but focused on Grids in this case. The design of XtreemOS [6] has taken into account some particular characteristics of Grid systems and has faced challenges like scalability, transparency, interoperability, security, etc. The overall architecture of XtreemOS, from the Linux kernel to the application level, has been divided in two different levels: the Foundation layer (F-layer) providing VO support and the Grid services layer (G-layer), supported by the previous one and providing a set of services that can be used by the application layers.

The mobile device version of XtreemOS follows the same general architecture and it is adapted to the mobile device intrinsic limitations. The main targets for XtreemOS-MD have been mobile devices like PDAs and smartphones. The particularities of these mobile devices revealed the need of a specific version of the software, where aspects like mobility, limited computing and storage capabilities, and battery lifetime have been considered. The XtreemOS-MD architecture addresses those important limitations and adapts the XtreemOS general requirements and design principles to the mobile devices selected. For this reason, specific modules for providing context information and also for managing the resources shared by the mobile devices have been added to the architecture. This modules are deeply explained in section 4.

Another important aspect when thinking on mobile devices is the fact that the users are in general not computer-experts. This means that the ease of installation and use should be seriously taken into account, and the provision of graphical interfaces for installation, configuration and use of the mobile device software should be a priority, as it is for XtreemOS-MD. Other factors like the compatibility with native applications and the security model of each device have been considered when deciding the roadmap for mobile devices supported. In a first phase, XtreemOS-MD software is developed for Nokia N8x0 family of Internet tablets (based on the Maemo Linux OS [18]) and for the NeoFreeRunner smartphone (based on the SHR (Stable Hybrid Release) Linux OS from Openmoko [23]). We should highlight that XtreemOS-MD is mounted over a Linux operating system, thus potentially any mobile platforms based on Linux could support XtreemOS-MD software with minor adaptations.

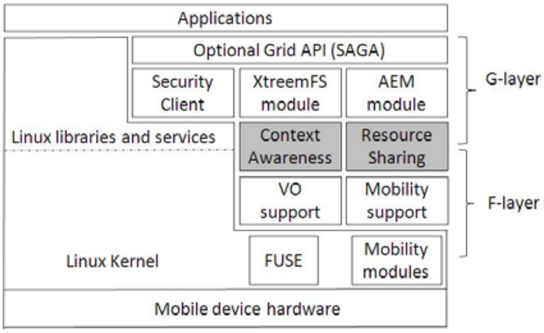

Fig. 1. XtreemOS-MD general architecture

The general architecture of XtreemOS-MD is depicted in Figure 1 where it is shown the different modules belonging to each layer that will be discussed in the following sections. The blocks in grey are precisely the most significant difference between XtreemOS and XtreemOS-MD architectures and will be further analyzed along the article.

3 The Foundation Layer

In the Grid computing world, an important concept is the Virtual Organization (VO), which refers to a set of users and real organizations that collectively provide resources they want to exploit for a common goal [5]. XtreemOS and XtreemOS-MD provide a full support for VO management [21], allowing not only to join an existing VO, but even to administrate them from a graphical interface specially adapted for limited resolution screens, like the ones provided typically by mobile devices.

The F-layer comprises the components that modify the Linux OS itself, to make it aware of virtual organizations in the Grid and to provide mobility for the Grid access. The VO support module provides several VO-related facilities to Grid users like management of user's credentials, dynamic management of UID/GID, authentication and policy-based authorization and session management.

Regarding mobility, it introduces a series of difficulties for keeping sessions, loosing connectivity, etc. Mobility in XtreemOS-MD is provided by a mobility module at kernel level plus a mobility support module included in the F-layer. The solution adopted is based on an adaptation for ARM (Advanced RISC machine) architectures of the USAGI MIPv6 implementation [27], that allows users to stay connected to the Grid while they are moving, maintaining the same IP address and with only minimum delays when handing off between access networks.

The main differences that can be found in XtreemOS-MD regarding the XtreemOS architecture are the following modules:

- **Context awareness module:** That provides to higher levels information about the context of the user's terminal, including information like battery level, geographical position, current operation mode and especially the important information about network connectivity, one of the main issues when targeting mobile environments. The intrinsic mobility of mobile devices can derive in a frequent change on the connectivity conditions, moving from one network to another, changing from WiFi to 3G or even loosing temporarily the connection. XtreemOS-MD pays a special attention to this issue and thanks to this module, higher layers may define context-based rules to determine when the mobile device should be "moved" to offline mode (disconnected from the Grid). For example, a user could define a rule to move to offline mode when connected using a non-free 3G connection, or when the remaining battery falls below an specified threshold.
- **Resource sharing module:** Taking into account the limitations of mobile devices, other Grid initiatives have left them apart. Nevertheless, one of the strengths of XtreemOS is the existence of different flavours for PCs, clusters and mobile devices. It's clear that the limitations are there, and even if the computational and storage power of those devices are quickly being increased, for the moment they are not the most suitable options to run jobs or to store data. Additional problems like the lack of reliability and confidentiality appear, as the mobile devices can be manipulated and are quite accessible. Anyway, even having in mind those issues, XtreemOS-MD wants to become not only a mobile Grid client , but also a way to make available some resources of the mobile devices. Computing and storage are not the mobile devices' strengths, but we can think in the possibility of sharing resources like the camera, the GPS, or the network connectivity, and some use cases appear where those resources are needed. For example, a user could share its plane-rate 3G connection to their colleagues in a meeting, connected from their mobile devices to the local Grid via Wifi, but without a free external access to Internet.

  The resource sharing module included in the architecture is in charge of publishing in the Grid the resources offered by each mobile device and also provides the means for accessing them from a Grid client, also covering the privacy issues associated (and, for example, requesting the users authorization before giving access to their resources shared).

4 The Services Layer

Located on top of the Foundation layer, the services layer or G-layer of XtreemOS-MD is in charge of providing access to the Grid services from mobile devices. The three main services offered in the mobile device version are provided by the security client module, which enable users to authenticate against the Grid, the job management module, used for executing and managing jobs in remote machines in the Grid, and the remote file system module, which allows a transparent access to remote data.

4.1 Security Client Module

This module is implemented as a CDA (Credential Distribution Authority) client in charge of the negotiation and obtainment of user's credentials from the XtreemOS CDA server. Once the user's credentials are obtained they are passed to the VO support module in the F-layer and a new session is started in the VO. One of the issues associated with authentication in mobile devices is related to the usual lack of appropriate keyboards, and impossibility of using external authentication devices, like biometric systems (fingerprint, retina recognition, etc.). In fact, many laptops currently provide an integrated fingerprint reader, but none of the mobile devices considered including such a device.

In order to reduce those limitations, XtreemOS-MD offers the possibility of using a proxy for obtaining the credentials needed in the authentication process. Currently XtreemOS-MD does not use SIM card based authentication because it would limit the solution only to devices with a SIM card reader. The proxy (CDAProxy) is the module added to the XtreemOS architecture to communicate the local CDA client running on the mobile device and the remote CDA server, and which could also serve local credentials stored in the proxy. The connection between the client and proxy is secured by SSL. The main benefit is that the proxy may be executed out of the mobile device, in a more powerful user's PC for

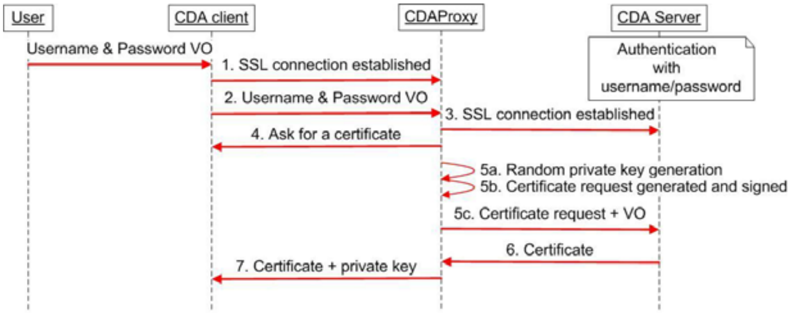
User
CDA client
CDAProxy
CDA Server
Authentication with username/password
Username & Password VO
1. SSL connection established
2. Username & Password VO
3. SSL connection established
4. Ask for a certificate
5a. Random private key generation
5b. Certificate request generated and signed
5c. Certificate request + VO
6. Certificate
7. Certificate + private key

Fig. 2. A CDAProxy can be used by XtreemOS-MD in order to reduce the time needed for certificate generation

instance. That reduces the time for generating the certificate, a heavy computational task, and offers alternative ways of authentication (biometric systems, etc). This delegated authentication process is shown in Figure 2. After the initial connections establishment between the client, the proxy and the server (steps 1 to 3), the client asks for a certificate (step 4). Then, the proxy issues the private key and also generates and signs the corresponding certificate request (steps 5a and 5b). Finally, the certificate request is sent to the CDA server (step 5c), which will answer back with the certificate (steps 6 and 7). The certificate generation is just needed from time to time, depending on the concrete policies related to certificate validity and expiration. It would be possible then to use a proxy running in a nearby machine to perform the certificate generation, which would not be anymore needed once stored the credentials in the mobile device. After credential expiration, the mobile device could connect again to any CDAProxy to renew the credentials.

Due to the fact that the proxy would be normally executed in a more powerful equipment than the mobile device, the time for generating the key would be considerably lower than the required time to generate the same key in the mobile device. Several tests have been performed comparing the average time for key generation in a Nokia N800 mobile device and a mid-range PC. Results are shown in Figure 3, where it is noticed that the average time for certificate generation is around 15 times lower (0,41 seconds against 6,1 seconds) when using the CDAProxy instead of the security client in the Nokia N800 device. Also, it's quite remarkable the high variance of the times for credential generation in the Nokia device.

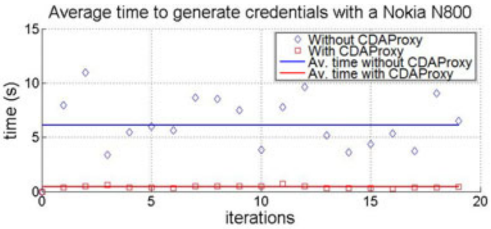

Fig. 3. Certificate generation time with and without CDAProxy

4.2 File System Service (XtreemFS Service)

Transparently integrating XtreemFS, mobile device users will access, from their usual file system clients, not only local files, but also their files stored in the Grid. Taking into account the typical storage limitations of the mobile devices, the XtreemOS-MD users will benefit from a great storage capacity increase, as they will access their remote volume in the Grid (whatever the size of this volume is) from their mobile device.

Before we present how this integration is achieved, and how mobile device limitations have been taken into account, let's first see a very brief summary of how XtreemFS works. XtreemFS is mainly composed by three components MRC (Metadata Replica Catalogue), OSDs (Object Storage Devices), and Clients. The MRC is in charge of maintaining all metadata information about file such as access rights, dates, replica locations, etc. The OSDs are the resources where data is actually stored. A file may be in a single OSD or in several if the file is stripped. OSDs also take care of coordinating stripped files and replicated files. Finally, the client is a user library built using FUSE [11], a file system in user space, which is located below the F-layer as a kernel-level module. Whenever a client wants to access a file, it firsts contacts the MRC to get the information about the OSDs storing the file and a credential to prove it has access to it. With his credential, the client can access the file in the OSDs with no further connection with the MRC (unless the credential expires, in which case a new credential will be requested for this file). The current version of XtreemOS, only allows mobile devices to act as clients (although the possibility of sharing files from the device is currently under development). Nevertheless, we had to design the file system in order to make sure that it was compliant with the mobile devices peculiarities. XtreemOS-MD offers then a transparent access to the remote user's volume based on XtreemFS.

The first important issue in a mobile device is that a stable connection can't be assumed. In XtreemFS, OSDs, do not keep state about the clients (the little state that is actually stored is just for some performance optimizations). A client just needs the credential to be able to access a file from the OSDs, and there is no need for the OSD to recognize the client as a previous one. In the same line, avoiding any kind of persistent link between the file system and the client reduces the battery wasted as only the needed data are sent through the network device. In addition, this mechanism allows the mobile device to go to standby (or even disconnect) between two accesses (i.e. reads) to the same file. Finally, manually mounting the volume is not even needed, as XtreemOS-MD provides an auto-mounting mechanism, which will be also in charge of requesting the user's credential when the volume is going to be used and is still not mounted. Once authenticated the first time, the credentials are stored and the user's password will not be requested again during the current session.

4.3 Execution of Jobs in the Grid Service

The job management service delivers to the user the control to launch, pause, resume, stop or check if a job is running on the Grid. This service is covered by the AEM (Application Execution Management) module which not only manages jobs but also acts as a gateway for the user to the resources associated to them. This component was designed with mobile devices in mind, so main issues were resolved at design level while only minor ones were exposed later when the actual implementation on device platforms took place. AEM is composed of several services running on an XOSD (XtreemOS Daemon) providing a single access point to the system. AEM architecture is distributed, in every node of XtreemOS we will find an XOSD running with several different services depending on the nature of the node. Mainly there are core and resource node types; in a core node we have Job, Execution and Resource management services. On the other hand, a resource node should have Execution and Resource management. However, other services can be run in order to distribute functionality over the Grid. Mobile devices benefit from the non-persistent connection between calls, so the client can change its IP address and continue interacting with the system using his credentials. Client identification and authentication is only based on the certificate presented on each request. Also AEM supports that the clients can change their entry point to the whole Grid, represented by an XOSD. This feature benefits mobile devices in the way that they can simply access the closest node removing the service discovery overhead by leaving it to the grid. Any XOSD is an entry point to the whole AEM. Requests are internally redirected to the correct XOSD and service which is located using the distributed overlay. For mobile devices usage, AEM has a C interface (C-XATI), which is generated by DIXI as well, in addition to the Java one. With this C interface there is no difference between the accessing flavours of AEM. A client in an XtreemOS node can submit a Job, and see its status in the mobile device, set a user monitoring event in the job and receive the feedback in the mobile. C-XATI takes care of some issues affecting mobile devices and other limited devices; external libraries not available in the mobile device and connections. AEM also works as a proxy to register a mobile component (GPS, camera) with the resource discovery service to share it with the Grid or Cloud. The whole monitoring infrastructure of XtreemOS is exposed to the mobile devices through XATI (C-XATI). Every functionality present in the current version of the software has been directly ported to the mobile devices flavour of XtreemOS. Nevertheless, we are currently exploring some specific adaptations, for the case of connection loss from the client side. Particularly, a user might request a call-back on a monitoring event, like a job changing its state, but its connection may be closed before the call-back is sent. In that scenario, AEM would offer the client a way to request pending call-backs whenever it detects the reconnection. This functionality is not only useful for mobile devices, but also for laptops or even desktop clients, although is more of a requisite on them because of the frequency this connection losses could happen.

5 Performance of XtreemOS-MD Services

One remarkable aspect of XtreemOS-MD services is their performance. Regarding the file system and job execution services, we have designed some scenarios to compare the performance of those services when using a mobile device versus the performance of an XtreemOS client running in a regular PC. We have repeated this tests directly on a mobile device, a Nokia N800, to show the benefits of performing those operations in the Grid instead of doing them directly on the device itself, which is less powerful in terms of computation. The results are reflected in Figure 4, where it can be appreciated than the performance of XtreemOS-MD running in the Nokia N800 is similar to the one achieved with the XtreemOS client running in a PC (a Pentium 4). But also it's remarkable the performance improvement obtained when using XtreemOS-MD while compared to the direct execution of the operations in the mobile device. The four different tests reflected in Figure 4, where the values shown in axis Y are the average from a series of different realizations, correspond to:

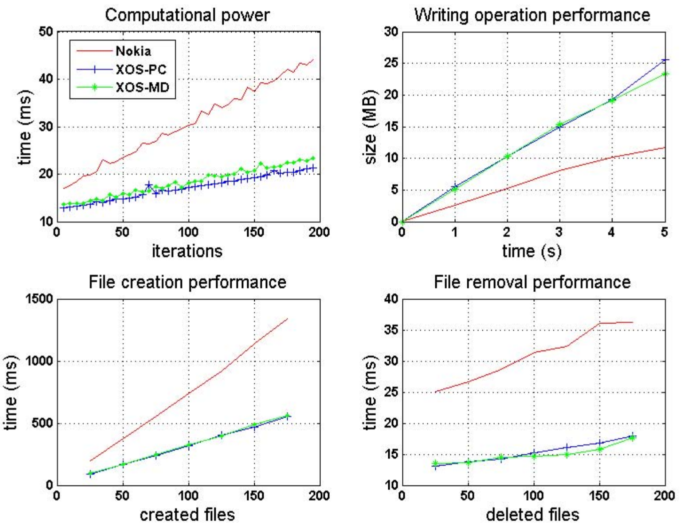

Fig. 4. Comparison of computational power (job execution) and performance of some file operations when using directly the mobile device (Nokia N800), XtreemOS client running in a PC and XtreemOS-MD running in the Nokia N800

- Computational power: a simple script including a loop incrementing a variable that is passed a number of iterations.
- File creation performance: a script that creates a number of empty files in the file system (XtreemFS or local file system in the Nokia realization).
- File removal performance: a script that removes a number of empty files in the file system (XtreemFS or local file system in the Nokia realization).
- Writing operation performance: a script that writes some data in a file of the file system (local in the Nokia or in the XtreemFS when using XtreemOS and XtreemOS-MD) for a defined time. The axis Y represents the average size of the file generated during the time represented in axis X.

These results show that XtreemOS-MD services have a similar performance than XtreemOS ones. That means that the time for completing any operation on the Grid does not depend on the client used (PC with XtreemOS or mobile device with XtreemOS-MD). The reason behind is that the communications and protocols designed to interact with the Grid are light enough to be implemented even in limited mobile devices. The use of native code in the mobile devices against the Java code used by the implementation of the services for the XtreemOS PC version is one of the reasons permitting a quite similar performance of XtreemOS-MD services running on a mobile device and XtreemOS services running on a more powerful PC.

Those tests are simple tests for performance evaluation, but thinking on a real use case, we have designed a different scenario, related with video conversion in the Grid. To test this use case we have launched several jobs of video conversion in the Grid, using the job execution service for launching the jobs, and also directly in a Nokia N800 mobile device. For the Grid-aware tests, the input video file is read from the user's grid volume, then it is converted to a different video quality and finally the output video file is stored in the same user's volume in the grid. In both cases, mobile and PC, the Grid testbed used was the same. The result, obviously dependent of the file size, has shown to be completely independent of the client used, be it either a PC or mobile device, as stated in Figure 5 where the red line shows the average time for the video conversion job when launching the job from a PC running XtreemOS and the green line corresponds to the average time when launching the job from a mobile device running XtreemOS-MD.

But the most remarkable result is the comparison with the time spent by the Nokia N800 in the video conversion process, especially with the increase of the file size. For a video file of just around 30 MB, the time spent by the Nokia is around 20 minutes, 10 times more than the Grid. And it's not just a matter of time, but also a problem of battery consumption. Some measurements relative to the battery consumed by the Nokia N800 during the conversion process have been taken and the consumption for a file of 30 MB is around 4% of the total capacity (around 10 times higher than the battery power that would have been consumed during a similar period of inactivity).

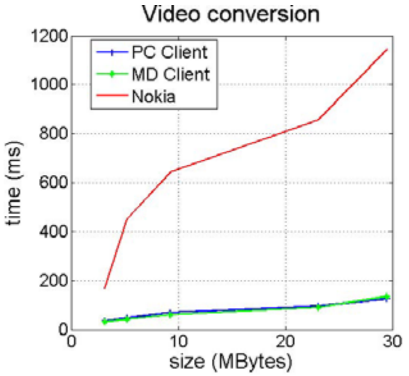

Fig. 5. Video conversion performance in the Grid (PC and mobile device client) and in a Nokia N800

6 XtreemOS-MD Applications

One of the strenghts of XtreemOS-MD is the possibility of using all the existing classical non-Grid applications, and even modified them slightly to add Grid features to them. For example, it's much more simple to extend an existing instant messaging client (like Pidgin for example) to incorporate the support for writing logs, etc. in the XtreemFS or to allow chatting with anyone connected to your same VO than writing from scratch a new instant messaging Grid-aware application as would be necessary if the Grid client would be based in a simple web portal solution. Apart from those existing or modified applications, some additional applications have been developed to be released together with the XtreemOS releases, offering some interesting specific functionalities.

6.1 The JobMA Application

The classical Linux command-line interface is not the most appropriate one when targeting mobile phone users. In order to simplify the process for managing jobs in the Grid, XtreemOS-MD provides a graphical application called JobMA (Job Management Application). This application provides an intuitive access to XtreemOS job management facilities like create and launch jobs, stop, resume, or cancel running jobs, and also job monitoring. Advanced jobs, specified by JSDL (Job Submission Description Language) file [2], can be loaded and executed, but also basic jobs can be defined with a specific GUI form.

6.2 The Grid-Player Application

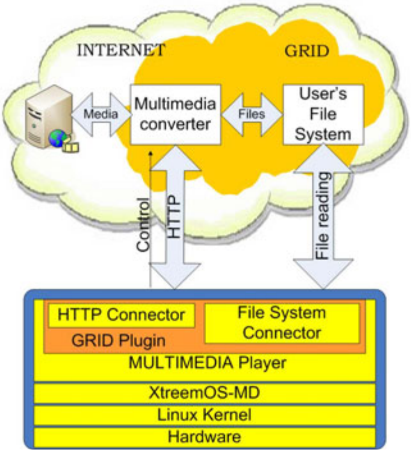

Fig. 6. Grid-Player application architecture

Nowadays, a video player is one of the features with more demand in a mobile device. Nevertheless, the video capabilities and the low screen resolutions of those devices highly limit their possibilities. This opens the door to a new multimedia application, that we called the Grid-Player, consisting on a video player capable of translating the original format and resolution of a video when not supported in the mobile device (for example, when codec is not available, etc.). The process for video transcoding is really heavy from a computational point of view, so that the remote processing in the Grid instead of directly on the local device is an important strength of the application; and this is not just a question of saving time, but also saving battery power, one of the main limitations of current mobile devices. The architecture of this Grid-Player application is shown in Figure 6, where it's reflected that the Multimedia Player application running in the mobile device will connect to the Grid in order to transcode one video stored in the user's file system (or accessible via HTTP), storing again in the user's volume the resulting video, optimized for the mobile device that requested the conversion (codec, resolution, etc.)

A Grid-Player first prototype, based on the JobMA application and making use of open source video-conversion software (*ffmpeg* library) running on the Grid, has already been demonstrated with a very good acceptance. In fact, this prototype has been used to execute the tests which results are shown in Figure 5. This kind of application could serve as a motivation to use the Grid (and concretely XtreemOS-MD), even not having the users a real knowledge on the infrastructure behind. For them, it will be just an application for playing local or remote videos stored in whatever format, resolution or video codec used.

7 Conclusions

Along this article we have presented XtreemOS-MD as a new way to enter in the Mobile Grid Computing arena. The transparent Grid access from mobile devices will offer the possibility of using a wide range of applications from a quite limited device like a PDA or a smartphone. Thinking on the increasing importance of mobility, XtreemOS-MD could be very interesting as a new way to use remote desktop applications, like typical office packages for instance, or to improve mobile device applications, including multimedia or games, by using Grid capabilities directly from the mobile device. Finally, taking into account the number of mobile devices around the world, and the XtreemOS-MD ease of use, it could serve as a catalyst for popularizing the Grid, extending their use to non computer savvy users. It should be noticed that XtreemOS-MD first prototype has been released before summer 2009. This 1.0 release includes the access to the main XtreemOS services from mobile devices like the Nokia N8x0 PDA and the NeoFreeRunner smartphone.

Acknowlegments

This work was partially supported by the EU IST program as part of the XtreemOS project (contract FP6-033576), by the Spanish Ministry of Science and Technology under the TIN2007-60625 grant, and by the Catalan Government under the 2009-SGR-980 grant.

References

1. Amazon Web Services LLC. Amazon Elastic Compute Cloud (Amazon EC2), http://aws.amazon.com/ec2
2. Anjomshoaa, A., Drescher, M., Fellows, D., Ly, A., McGough, S., Pulsipher, D., Savva, A.: Job Submission Description Language (JSDL) Specification (November 2005), http://www.gridforum.org/documents/GFD.56.pdf
3. Clarke, B., Humphrey, M.: Beyond the "Device as Portal": Meeting the Requirements of Wireless and Mobile Devices in the Legion Grid Computing System. In: Proceedings of the 16th International Parallel and Distributed Processing Symposium table of contents. IEEE Computer Society, Washington (2000)
4. Chu, D.C., Humphrey, M.: Mobile OGSI.NET: Grid Computing on Mobile Devices. In: Proceedings of the Fifth IEEE/ACM International Workshop on Grid Computing (GRID 2004), November 08-08, pp. 182–191 (2004)
5. Coppola, M., Jégou, Y., Matthews, B., Morin, C., Prieto, L.P., Sanchez, O.D., Yang, E.Y., Yu, H.: Virtual organization support within a grid-wide operating system. IEEE Internet Computing 12(2) (2008)
6. Cortes, T., et al.: XtreemOS: a Vision for a Grid Operating System, White paper (May 2008)
7. Erwin, D.W., Snelling, D.F.: UNICORE: A Grid computing environment. In: Sakellariou, R., Keane, J.A., Gurd, J.R., Freeman, L. (eds.) Euro-Par 2001. LNCS, vol. 2150, p. 825. Springer, Heidelberg (2001)
8. González-Castaño, F.J., Vales-Alonso, J., Livny, M., Costa-Montenegro, E., Anido-Rifón, L.: Condor grid computing from mobile handheld devices. ACM SIGMOBILE Mobile Computing and Communications Review 6(2), 18–27 (2002)
9. Foster, I.: Globus Toolkit Version 4: Software for Service-Oriented Systems. In: Jin, H., Reed, D., Jiang, W. (eds.) NPC 2005. LNCS, vol. 3779, pp. 2–13. Springer, Heidelberg (2005)

10. Foster, I., Kesselman, C.: The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
11. FUSE: Filesystem in Userspace, `http://fuse.sourceforge.net/`
12. Globus Alliance. The Globus Toolkit 4.1.3 Developer's Guide, `http://www.globus.org/toolkit/docs/development/4.1.3/developer/index.html`
13. Condor Team. Condor Manual. University of Wisconsin-Madison, `http://www.cs.wisc.edu/condor/manual/v7.5/`
14. Grimshaw, A.S., Humphrey, M.A., Natrajan, A.: A philosophical and technical comparison of Legion and Globus. IBM Journal of Research and Development 48(2), 233–254 (2004)
15. Grimshaw, A.S., Wulf, W.A., the Legion Team: The Legion vision of a worldwide virtual computer. Communications of the ACM 40(1), 39–45 (1997)
16. Google Inc.: Google Application Engine, `http://code.google.com/intl/appengine/appengine/`
17. Gridipedia web site. The history of Grid, `http://www.gridipedia.eu/historyofgrid.html`
18. Nokia. Maemo operating system, `http://nokia.maemo.com`
19. Microsoft Corporation. Windows Azure Platform, `http://www.microsoft.com/azure/default.mspx`
20. Millard, D., Woukeu, A., Tao, F.B., Davis, H.: Experiences with Writing Grid Clients for Mobile devices (2005)
21. Morin, C.: XtreemOS: A grid operating system making your computer ready for participating in virtual organizations. In: Proceedings of the 10th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC 2007), pp. 393–402 (May 2007)
22. Novotny, J., Russell, M., Wehrens, O.: GridSphere: a portal framework for building collaborations. Journal of Concurrency and Computation: Practice and Experience 16(5), 503–513 (2004)
23. OpenMoko. SHR (Stable Hybrid Release) distribution, `http://wiki.openmoko.org/wiki/SHR`
24. Phan, T., Huang, L., Dulan, C.: Challenge: integrating mobile wireless devices into the computational grid. In: Proceedings of the 8th Annual International Conference on Mobile Computing and Networking, Atlanta, Georgia, USA, September 23-28 (2002)
25. SYS-CON Media Inc. Twenty-One Experts Define Cloud Computing (2008), `http://cloudcomputing.sys-con.com/node/612375/print`
26. Thain, D., Tannenbaum, T., Livny, M.: Condor and the Grid in Grid Computing: Making the Global Infrastructure a Reality. In: Berman, F., Fox, G., Hey, T. (eds.), Wiley, Chichester (2002)
27. USAGI Project. Linux IPv6 Development Project, `http://www.linux-ipv6.org/`
28. Wesner, S., Jähnert, J.M., Toro, M.A.: Mobile Collaborative Business Grids - A short overview of the Akogrimo Project
29. XtreemOS project. Building and Promoting a Linux-based Operating System to Support Virtual Organizations for Next Generation Grids, `http://www.xtreemos.org/`