



Project Number	IST-2006-033789
Project Title	PLANETS
Title of Deliverable	First Version of GRATE
Deliverable Number	D7
Contributing Sub-project and Work-package	PA/5
Deliverable	Internal
Dissemination Level	Public
Deliverable Nature	Report
Contractual Delivery Date	M30
Actual Delivery Date	M33
Author(s)	ALUF

About this document:

This deliverable describes the first version of GRATE (Global Remote Access To Emulation services), one of the tools for environments in PA/5, which allows access to emulation processes run remotely via a network.

This document is intended for both project participants and technical specialists. It describes the history, design, functionality, technical environment and future work of the latest release of GRATE within the Planets project. It does not contain a user manual for the tool. Installation notes can be downloaded from the appropriate project website, together with the released software.

Contributors

Person	Role	Partner	Contribution
Randolph Welte	PA/5	ALUF	Document owner and author
Dirk von Suchodoletz	PA/5	ALUF	Project group member
Klaus Rechert	IF	ALUF	Project group internal reviewer
Remco Verdegem	PA/5	NANETH	Reviewer
Jeffrey van der Hoeven	PA/5	KB-NL	Reviewer
Maurice van den Dobbelsesteen	PA/5	NANETH	Reviewer

Document Approval

Person	Role	Partner
Frank Houtman	SP lead PA	KB-NL
Remco Verdegem	WP lead PA/5	NANETH
Maurice van den Dobbelsesteen	WP lead PA/5	NANETH

Executive Summary

Within the Preservation Action sub-project, the workpackage PA/5 develops and tests both emulation tools and tools related to emulation. Three tools are under development:

1. Dioscuri (a modular hardware emulator for digital preservation)
2. UVC (Universal Virtual Computer)
3. GRATE (Global Remote Access To Emulation services)

These tools were already under development by PLANETS partners before the project started, and within the PLANETS context these tools have been developed further. Deliverable PA/5-D2 showed the first results of the evolved tools.

This deliverable, PA/5-D7, focuses on the first PLANETS release of GRATE. History, design, functionality, technical environment and future work will be described. Further user documentation can be downloaded from the project website, together with the released software.

GRATE is a demonstration of a webservice offering emulation over the Internet. It allows users to access a service that recreates obsolete target environments from their local computer. There is no need to install any emulation software, target operating systems, applications, drivers or other required software to access the target environment in question.

Table of Contents

1	<i>Introduction</i>	5
2	<i>Technical background</i>	6
2.1	GRATE Functionality	6
2.2	Options for Network Transport of Applications	6
2.2.1	Citrix Presentation Server.....	6
2.2.2	SUN Secure Global Desktop	7
2.2.3	NX NoMachine/FreeNX.....	7
2.2.4	Microsoft RDP.....	7
2.2.5	VMware Server 2 Remote Console	7
2.2.6	VNC.....	8
2.3	Object Transport from External to Target Environment	8
2.3.1	Network Transport.....	8
2.3.2	File Transfer Protocol.....	9
2.3.3	Server Message Block	9
2.3.4	Container File Inclusion	9
2.3.5	Shared Folders (VMware)	10
2.3.6	Linux Loopback Mounting	11
2.3.7	Creation of Floppy Images	11
2.3.8	ISO Images	12
2.3.9	Copy & Paste.....	12
3.	<i>GRATE (first Planets version)</i>	13
3.1	Overview	13
3.2	Description	13
3.3	Screenshots	14
3.4	History	15
3.5	Design	15
GRATE server.....	17	
GRATE client.....	18	
3.6	Functionality	18
3.7	Technical environment	18
3.8	Development notes	18
3.9	Future work	19
4.	<i>References</i>	19

1 Introduction

Within the Preservation Action sub-project the workpackage PA/5 develops and tests both emulation tools and tools related to emulation. Three tools are under development:

1. Dioscuri (a modular hardware emulator for digital preservation),
2. UVC (Universal Virtual Computer) and
3. GRATE (Global Remote Access To Emulation services).

These tools were already under development by PLANETS partners before the project started and within the Planets context these tools have been developed further. Deliverable PA/5-D2 showed the first results of the evolved tools.

The Computer Centre of the University of Freiburg (ALUF) started development of GRATE in 2003. GRATE is developed to demonstrate the accessibility of 'obsolete' digital objects - e.g. WordPerfect 5.1 documents - through emulated systems using remote technologies via networks (Internet). GRATE deploys a client-server architecture. It is a predominantly Java-based web service. The client can be executed within every Java-enabled web-browser. On the server side, GRATE enables the deployment of hardware emulators like QEMU¹ or Dioscuri² (modular emulator) and their remote GUI-usage within a Java-based web-browser.

When used specifically with QEMU's emulated operating systems, GRATE offers a web-based upload of digital objects (e.g. created with unknown word processing software). For this purpose, GRATE makes use of the DROID service (batch identification of file formats), followed by a query of the PRONOM³ database (both controlled by The National Archives of the UK). By using these results, GRATE assumes there is a suitable - and already stored - environment with pre-installed applications, like a Windows 3.11 environment with a pre-installed, out-dated WordPerfect.

GRATE also has basic functionality for controlling scripts running inside a QEMU emulated environment. These features are based on the so-called QEMU monitor, an interface for sending commands to the QEMU emulator⁴. For example, the QEMU monitor can be used to send keystrokes to the emulated operating system. Programs running inside this operating systems can receive these keystrokes and perform actions accordingly.

GRATE version 0.1.1 can be downloaded from:

<http://www.planets-project.eu/private/pages/wiki/index.php/GRATE>

This deliverable, PA/5-D7, focuses on the first PLANETS release of GRATE. History, design, functionality, technical environment and future work will be described. Further user documentation can be downloaded from the project website, together with the released software.

This deliverable consists of two parts:

- Software: release 0.1.1 of GRATE
- Documentation

and is intended for both project participants and technical specialists.

¹ QEMU is an emulator written in C programming language: <http://bellard.org/qemu> or the PLANETS wiki at: http://www.planets-project.eu/private/pages/wiki/index.php/X86Emulation_tools_subpage#QEMU

² See the Dioscuri homepage at Source Forge: <http://dioscuri.sourceforge.net> or the PLANETS wiki at: http://www.planets-project.eu/private/pages/wiki/index.php/Tool_1:_Dioscuri_-_modular_emulator

³ PRONOM and DROID at the TNA: <http://www.nationalarchives.gov.uk/pronom>

⁴ The QEMU monitor can be either used via stdin of the running emulator instance or tcp-based. See <http://bellard.org/qemu/qemu-doc.html#SEC12> for further details.

2 Technical background

2.1 GRATE Functionality

Key functionality of GRATE includes:

- GRATE not only offers remote access to dedicated software (e.g. an emulator with a hosted Windows OS) via the Internet or other networks, but also allows users to upload primary digital objects.
- Then these objects can be automatically identified via services like DROID/PRONOM. GRATE is also able to display meta information about the stored secondary digital objects (like emulators, operating systems, etc.).
- Furthermore, GRATE is able to carry these objects into the emulated environments, making them available and - if necessary - copying them back to the user system.
- Finally, GRATE is able to send keystrokes to the emulated environments. This feature can especially be used for an automated control of other software running 'inside' the emulated operating system, for instance to connect to network shares, or shut down the emulated machine.

2.2 Options for Network Transport of Applications

The common commercial remote desktop applications are not suitable for long term preservation because they are only available as proprietary software with additional costs. As a result suitable interfaces are neither available nor documented, and no source code is available. Consequently, the software cannot be modified to meet our specific requirements. The GRATE system and all software components it depends on (the remote desktop functionality is completely based on TightVNC) is available as open source software.

Below follows a brief overview of selected Remote Desktop solutions:

2.2.1 Citrix Presentation Server

- Provides an extension of the Windows Terminal services based on the „Windows Server Family“. The software is more flexible than Windows Terminal server.
- Can export Applications for dedicated users. The user can use native viewers (clients) to use the GUIs of the applications
- There are Citrix Viewers for several platforms, including a platform independent Java Applet Viewer for Web browsers
- Only available for Windows Server systems
- Many features and interfaces including a resource manager or print services
- Protocol: Independent Computing Architecture (ICA) . This is similar to Microsoft's RDP (Remote Desktop Protocol)⁵ but supports a wider range of features.
- User actions like keyboard- or mouse-events or microphone recordings can be submitted to the server. USB Devices (e.g. Scanners) are supported too
- Graphic and sound output can be submitted to the clients expensive and proprietary licence model ("concurrent user" licence).
- Java client is available

⁵

Reference URL http://en.wikipedia.org/wiki/Remote_Desktop_Protocol

2.2.2 SUN Secure Global Desktop

- Originally developed by Tarantella, taken over by SUN Microsystems in 2005
- Similar to Citrix Presentation Server, but not tied to Microsoft operating systems
- Supports publishing of Windows, Unix, Linux and IBM AS/400 applications.
- Clients are available for Windows, Solaris, Mac OS X, Linux. Furthermore several thin clients (Ray, Wyse) and clients for mobile devices are available.
- User actions like keyboard/mouse events and microphone recordings can be submitted to the server. USB devices (for instance scanners) are supported too
- Graphic and sound output can be submitted to the client
- Java client is available

2.2.3 NX NoMachine/FreeNX

- Works as a kind of proxy and accelerator between two X11 partners.
- Client- and server side run a X11 server.
- Explicit enhancement of the X11 performance by a form of X11 top.
- Parts of NX - except the central client- and server application - are available under the GPL licence. FreeNX is also available under the GPL licence.
- NX session can be forwarded to VNC or RDP Server.
- Stateless connection is possible
- User actions like keyboard/mouse events and microphone recordings can be submitted to the server. USB devices (e.g. Scanners) are supported too
- Graphic and sound output can be submitted to the client
- Clients are available for Linux, Solaris, Windows, Mac OS X, Sharp Zaurus, Sony PS/2 and HP/Compaq iPAQ
- No Java client available

2.2.4 Microsoft RDP

- Originally developed by Citrix
- Based on the international T.120 protocol
- Unlike the Citrix protocol, RDP is more suitable for higher network bandwidths
- Predominantly used in Microsoft Windows environments
- Stateless protocol
- User actions like keyboard/mouse events and microphone recordings can be submitted to the server. USB devices (for instance scanners) are supported too
- Graphic and sound output can be submitted to the client
- Only the complete Desktop is exportable
- Proprietary licensing (Windows)
- Alternative RDP servers are available for Linux Systems. RDP viewers are available for Linux, Mac OS X and Amiga OS
- Java client available

2.2.5 VMware Server 2 Remote Console

- Developed as browser plug-ins by VMware as part of the VMware Server 2
- Browser plug-ins are available for Windows and Linux
- Restricted to environments executable as VMware guests
- User actions like keyboard/mouse events can be submitted to the server
- Remote USB devices (for instance scanners, ...) and remote optical drives (CD/DVD) are supported too
- Graphic output can be submitted to the client
- Sound could not be transported over the net
- Only the complete target environment is exportable
- Proprietary licensing (VMware) and protocol

2.2.6 VNC

- Originally developed by Olivetti & Research Laboratory
- Basic Versions of VNC are often available for free (GPL licence), proprietary licensed products are also available
- Based on 'Thin Client' architectures
- Very popular with a wide spread user and developer community
- As a result well documented open source software
- Available for a wide range of operating systems and platforms
- VNC server and the VNC clients can be configured with a wide range of parameters
- User actions like keyboard- or mouse events can be submitted to the server
- Graphic output of one or more application-windows can be submitted to the client
- Stateless protocol.
- Sound transfer is not implemented
- Java viewer is available.

PA/5 deemed all remote desktop solutions listed above – with the possible exception of VNC – as not suitable for our purpose of remote emulation. The solutions were either not flexible enough and/or the software was only available under a proprietary license. For GRATE It was decided to develop a wrapper around the TightVNC remoting software, which is available without licensing fees (Open Source).

2.3 Object Transport from External to Target Environment

After setting up a certain view path, a digital object of interest has to be transported from a digital repository into the environment able to render that specific object. In GRATE view paths are predefined by the installed target environments.

There are several ways to transport the object to the required environment, each way requiring different effort. The options for automation of transport processes are of specific interest, as it depends on the target environments executed:

- Network transport using IP services like FTP or SMB
- Container file inclusion (preparing emulation services virtual disks with target objects of interest)
- Creation of ISO images
- Floppy images
- VMware Shared Folders
- Copy & Paste technologies (implemented for text areas in Dioscuri)

2.3.1 Network Transport

Many of the modern, more complex/advanced operating systems offer low level access to network interfaces. These interfaces came around the 1980s and were first to be found in the commercial systems. This often required the installation of additional software like hardware and protocol drivers. In the beginning there was a wider range of network protocols in use. Halfway through the 1990s the Internet Protocol (IP) became dominant the network interface and came standard on every desktop machine (either MAC, PC or Workstation Systems)

In the beginning the different protocol drivers were not part of the Operating Systems and had to be purchased separately. If planned for object exchange they should be stored in the so called software archive⁶ as well.

⁶ A software archive is needed despite the number of so called secondary digital objects which are required to open or run the primary objects of interest.

This is also possible within the DOS environments using the NCSA IP suite⁷. The main advantage of the network transport is the synchronous operation: While running the reference environment any exchange of objects is possible in both directions. The file size is only limited by the capabilities of the deployed (obsolete) operating systems.

Network transport might be needed not just for the transport of objects, but also for:

- Demonstrating old protocols, services like the webpage of a site several years ago, using the traditional applications or tools
- Computer games using networking for multi player scenarios
- Networked applications like CMS systems or networked databases like SAP systems

Protocols that have been available for a long time, like File Transfer Protocol (FTP) or Remote Procedure Call (RPC) services like Server Message Block (SMB), are good candidates to be deployed for data exchange.

2.3.2 File Transfer Protocol

FTP is one of the oldest protocols using TCP/IP. It has been around for more than 30 years and has not changed very much. In the beginning there were only simple command line tools with a small footprint. Later, modern operating systems and/or additional tools implement user friendly front ends to this protocol. A case in point are web browsers, which have handled the FTP protocol rather conveniently (at least for anonymous access) from the beginning.

The NCSA Telnet package contains an FTP client as every Windows version from Windows 98 onwards (at least a rather simple, trivial one). Telnet has been a standard for the different versions of Unix from almost the beginning.

2.3.3 Server Message Block

SMB (Server Message Block) and CIFS (Common Internet File System) are not simple file transport protocols but network file systems. They are often integrated directly into the standard file system of the running operating systems and offer more convenient transportation of files back and forth.

They were originally invented at IBM with the aim of turning DOS "Interrupt 33" (21h) local file-access into a networked file-system. Later Microsoft made considerable changes to this software (e.g. integrating the "LAN Manager" protocol 3COM was involved with). SMB has been available in Windows since the "Workstation" release of 3.11 and Windows NT (around 1992). It is implemented in the different versions of Unix and similar systems via the "Samba" software package (Open Source, often part of the OS installation). In Windows Vista Microsoft introduced a new and mostly incompatible Server Message Block 2.0.

The existence of many implementations of SMB/CIFS, especially the Open Source Samba package, should offer long-term support for that protocol well after its demise (in some unforeseeable future).

With network enabled it would be theoretically possible to allow direct network connections from out of the target environments for e.g. downloading an object directly from the user's machine. However, the security issues involved mean that internal-only networking is preferred.

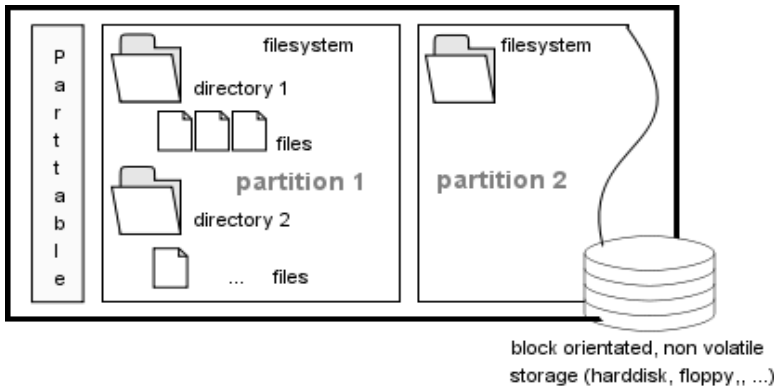
2.3.4 Container File Inclusion

It is possible to put the primary objects directly into the emulation/virtualization tool container files. This requires knowledge of the exact format of these containers and drivers for the file system used, and it is only possible before the emulator is started.

Before modifying the primary objects in the emulation/virtualization tool container file the emulator is stopped. Otherwise any changes to the block layer would corrupt the virtual disks with unforeseeable consequences.

⁷ http://www.planets-project.eu/private/pages/wiki/index.php/EmulationSoftwareArchive#Additional_Software__28drivers.2C.....29

Thus no "online" access to objects within the emulation/virtualization tool container file via network connections is required. This has the advantage that the hardware platform and the software running on it (e.g. for home computers) do not to support networking.



The container file option requires several special workflows:

- Creating the container in the required format
- Formatting it with a file system known/supported by the target environment
- Adding it as a secondary disk to the starting emulation/virtualization environment.
- If needed – Embedding an "auto start" kind of file (depending on the target environment)

2.3.5 Shared Folders (VMware)

Shared folders are a feature of virtualization tools like [VMware \(Version 5.0+\)](#) or [Virtual Box](#).

They depend on some newer guest operating systems like Windows XP. With shared folders it is possible to use a virtual network environment for direct exchange of data of some specified folders within the host system. To make this work tools have to be installed within the guest environments.

VMware has provided the tool "vmware-mount.pl" for quite a while, but does not advertise it. This tool is able to mount single partitions directly from "complex" containers (virtual hard disks with partition table and several partitions):

```
x60s:/planets # vmware-mount.pl/tmp/win98.vmdk 1/mnt
Subroutine shell_string redefined at/usr/bin/vmware-mount.pl line 293.

-----
VMware for Linux - Virtual Hard Disk Mounter
Version: 1.0 build-45731
Copyright 1998 VMware, Inc. All rights reserved. -- VMware
Confidential
-----

It has been reported that this program does not work correctly with
2.4+ Linux
kernels in some cases, and you are currently running such a kernel. Do
you
really want to continue? [N] y

No Network Block Device detected.
```

```

There is no Network Block Device defined on this machine. This script
is about
to create the/dev/nb0 Network Block Device. Continue? [Y]

Creating the/dev/nb0 Network Block Device

No Network Block Device driver detected.

Trying to load the Network Block Device driver kernel module...
Success.

Client: The partition is now mapped on the/dev/nb0 Network Block
Device.

Using another terminal, you can now browse your Virtual Hard Disk
partition in
/mnt. Hit Control-C in this terminal when done.
Warning: The NBD_DO_IT ioctl returned an error: Broken pipe
Warning: FILE: FileIO_Unlock Unlock of file '/tmp/win98.vmdk' failed:
Numerical argument out of domain.

```

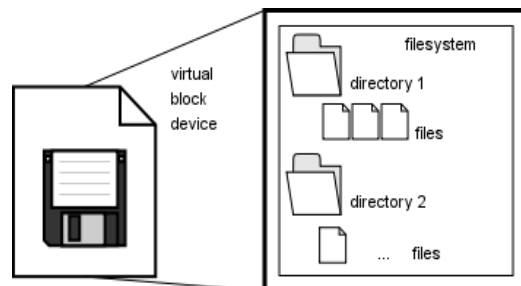
The warnings regarding the tool usage seem to refer to the error message during the unmounting process. Nevertheless, it was possible to access the target file system (Windows 98 FAT formatted) and add, edit, delete files within the container. The container was not harmed in any way as the virtual machine could be started properly with it, and the virtual machine found the altered files. However, the procedure is a little bit complex and should be re-hacked to use in unattended environments e.g. for automated use⁸ in some frameworks.

2.3.6 Linux Loopback Mounting

Simple container formats - like the "dd" images created for use with QEMU - allow for "loopback mounting", where files behave like disk partitions in the host operating system. The same applies for ISO and Floppy images.

2.3.7 Creation of Floppy Images

The floppy image option was the simplest one. The 1.44 MByte size is supported by most emulation/virtualization tools, the larger standard of 2.88 MByte usually not.



The standard 3.5" floppy drive is largely deprecated nowadays, and most modern computer do not come equipped with such drives anymore. The creation of empty unformatted floppies is straightforward (e.g. under Linux, writing 512 Byte blocks):

⁸ http://www.planets-project.eu/private/pages/wiki/index.php/Automation_Options_for_Emulation

```
dd if=/dev/zero of=floppy-1440.img count=2880
```

The same command could be used to copy an entire disk to a file:

```
dd if=/dev/fd0 of=any-floppy-image.img
```

Any floppy could be transferred this way - the underlying operating system does not need to know about the floppy's file system, even Mac, Solaris and other operating system floppies can be copied to image files. Floppy disk drives of older machinery like home computers are not as standardized as the more recent ones and (compatibility) problems might be expected there.

For Windows the Rawrite tool can be used to copy floppy disk to images - this imaging tool can be found in the Software Archive⁹. The images made by the different tools and understood by the various emulation/virtualization tools are interchangeable/compatible.

2.3.8 ISO Images

The direction of data exchange is limited from the outside into the rendering environment. The maximum size of an ISO image is 4 GByte of data which is ample and should serve most purposes. The ISO standard was extended over the time to overcome the old DOS file naming limitation. CDs have been around for a while and with successors like DVD and Blu-ray Disk (BD) readers are backwards compatible.

A major problem is the need for specific drivers in ancient operating systems. Although included in BIOS for booting, a separate driver needs to be installed for all DOS versions and for Windows 3.X. The drivers are available in the various Unix flavours and from Windows 95 onwards.

ISO images are of specific interest as installation media for intermediate software such as operating systems and applications. They can be produced from the original medium using tools like dd (e.g. the device sr0 is first optical drive in some Linux machine):

```
dd if=/dev/sr0 of=Windows98-install.iso
```

Digital rights management (DRM) – when present - could possibly prevent copying of the content from the medium handicapping proper archiving of that specific object. ISO image creation can be done with every CD/DVD burning tool suite too.

It is even possible to create multi-boot ISO images: It would be possible to put more than one installation medium to a virtual CD allowing the choice between several DOS versions and vendors to install. This would ease the handling of installation processes. Especially if floppy disk sets are combined with a CD there is no need for changing of media during installation. There is an example of such a multi boot disk available in the Software Archive.

Another interesting option is to pack several CD-based software packages (like games, encyclopaedias) into a single image via virtual CD drivers within target environment. There are several virtual CD drivers available for Windows OS or the loopback mount option within Linux

2.3.9 Copy & Paste

Copy and paste is possible of text regions in Dioscuri.

⁹ PLANETS sample software archive run by ALUF: <ftp://archive.ruf.uni-freiburg.de>, use "planets" and for password "pp2006-10"

3. GRATE (first Planets version)

3.1 Overview

Name: GRATE – Global Remote Access to Emulation-Services

Current version: 0.1.1

Release date: 12 August 2008

Availability: public, Open Source (GPL v2)

License: Some necessary components: GNU GPL version 2, ASL 2.0, Apache License v2.0, PHP Licence, Artistic License, MIT, Mozilla Public License 1.1

Programming language: Java (>= version 1.4.2), PHP v4 or higher, Perl, Java Script/Ajax Technologies

Organisation: ALUF

Reference: <http://www.planets-project.eu/private/pages/wiki/index.php/GRATE>

User manual: not fully available yet. See project site for further information.

3.2 Description

GRATE is a webservice written in Java/PHP/Perl and JavaScript (Ajax) and allows for location-independent remote access to designated emulation-services over the Internet.

The main goal is to provide a user-friendly service, which can be used to view and edit out-dated local existing digital objects (with an upload functionality). Additionally, modified objects can be saved back to the user's local system.

The GRATE client consists of two components: The GRATE Java applet and a Java Tight VNC client applet, embedded in PHP/JavaScript/Ajax code. Since Java applets are platform independent, every Java-enabled web-browser is suitable for executing GRATE.

The client communicates with the GRATE server component, which is responsible for session management (establishing and terminating VNC sessions, executing emulators, delivering meta-information, etc.) as well as transporting uploaded digital objects into the emulated environments.

As the GRATE is Java-based, it is possible - in principle - to port the server to various different operating systems. Currently only a Linux-based version is available.

3.3 Screenshots

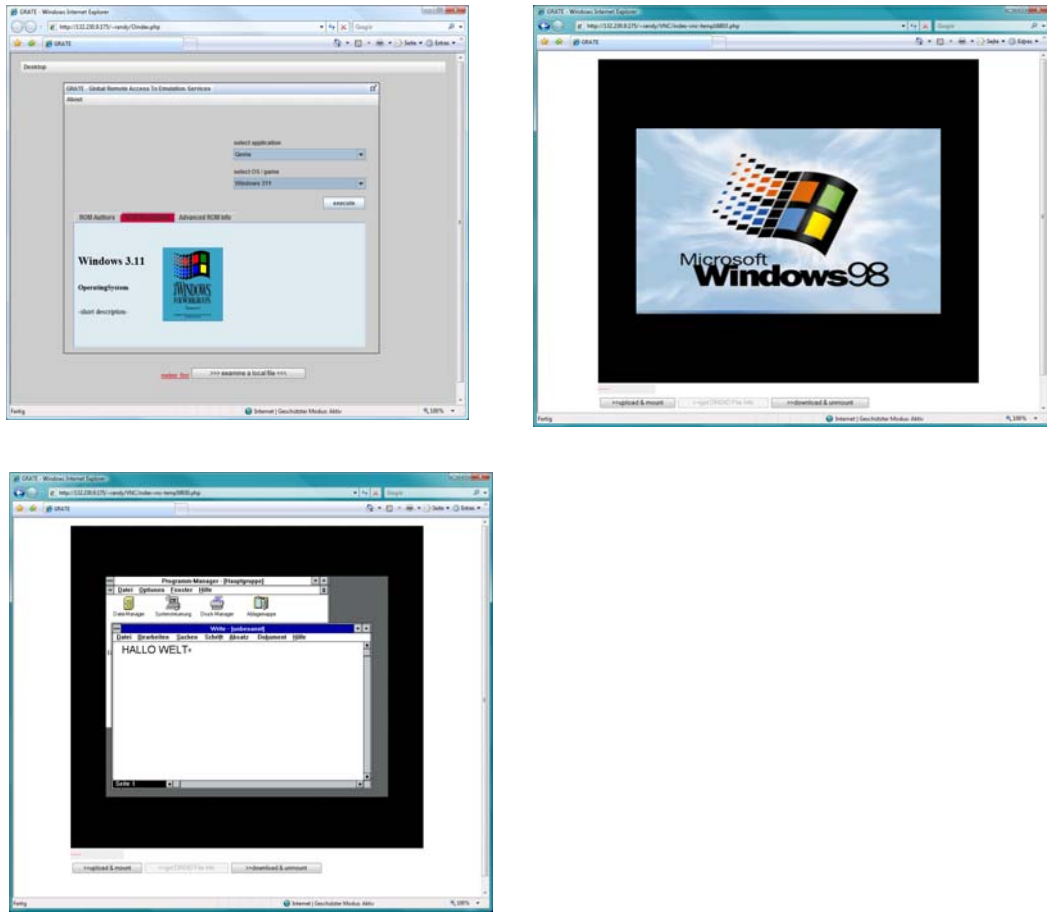


Figure 1: Screenshots of the GRATE client component

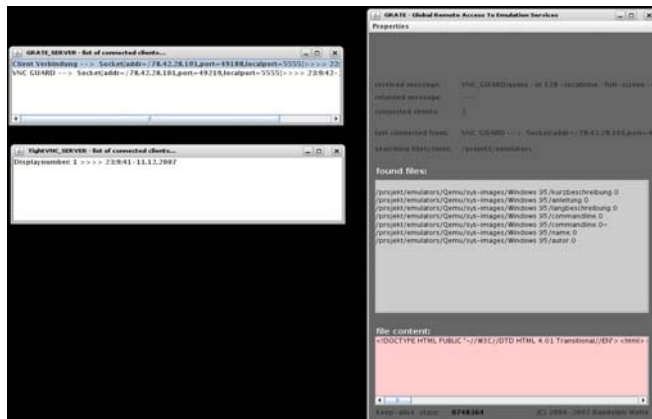


Figure 2: Screenshot of the GRATE server component

3.4 History

A predecessor and design study of GRATE was implemented in 2003 as a demonstration system for emulation in long-term preservation. It was presented to a wider audience at a University public event called “scientific market”. At that time, we wanted to demonstrate an automated execution of VMware environments via a web-browser hosted ActiveX control.

The main goal was to build a web-based demonstration system to use obsolete digital objects via “remote emulation”. The system and the “remoting part” should be based on open-source components like TightVNC.

3.5 Design

To develop a demonstration system, several “remote transport” solutions like Citrix or SUN Global Secure Desktop were evaluated (see the chapter on Technical Background)

Figure 3 shows the deployment diagram of GRATE:

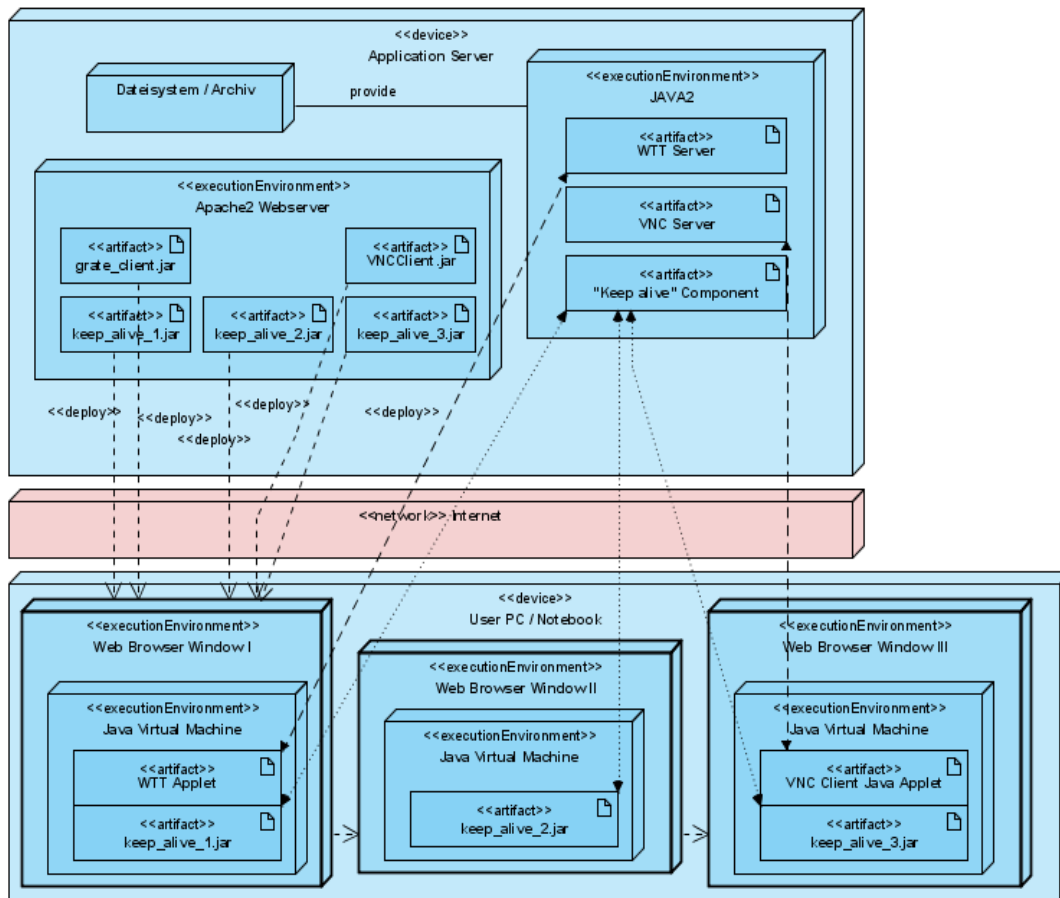


Figure 3: Deployment diagram of the GRATE demonstrator.

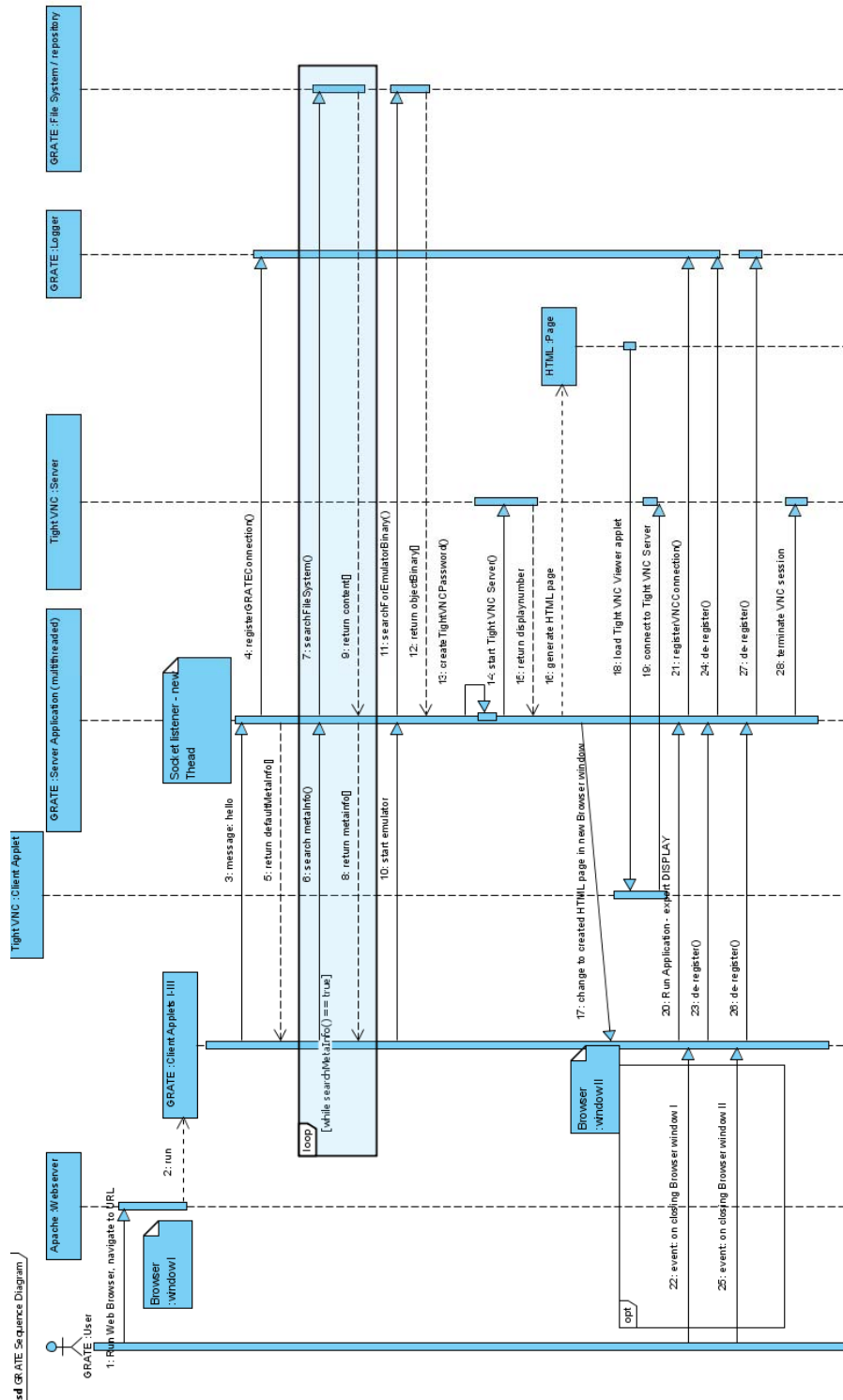


Figure 4: Sequence diagram of the GRATE demonstrator.

GRATE server

The GRATE server is the core component of the system. It handles the following tasks:

- Providing a “read only” GUI (graphical user interface) for various information (number of connected clients, content of the last file system archive search etc.) - fig. 2
- Complete session management (sessions to the GRATE client, VNC sessions)
- TCP based communication with the GRATE client
- Searching and querying the local archive/file system for emulator ROMs, meta information, etc. (simple and rudimentary folder-based “archive”)
- Execution and termination of emulators/virtualisers
- Upload and transport of the digital objects “inside” the running environments
- Script executing inside executed environments.
- DROID/PRONOM queries
- Assigning suitable environments for the obtained information.

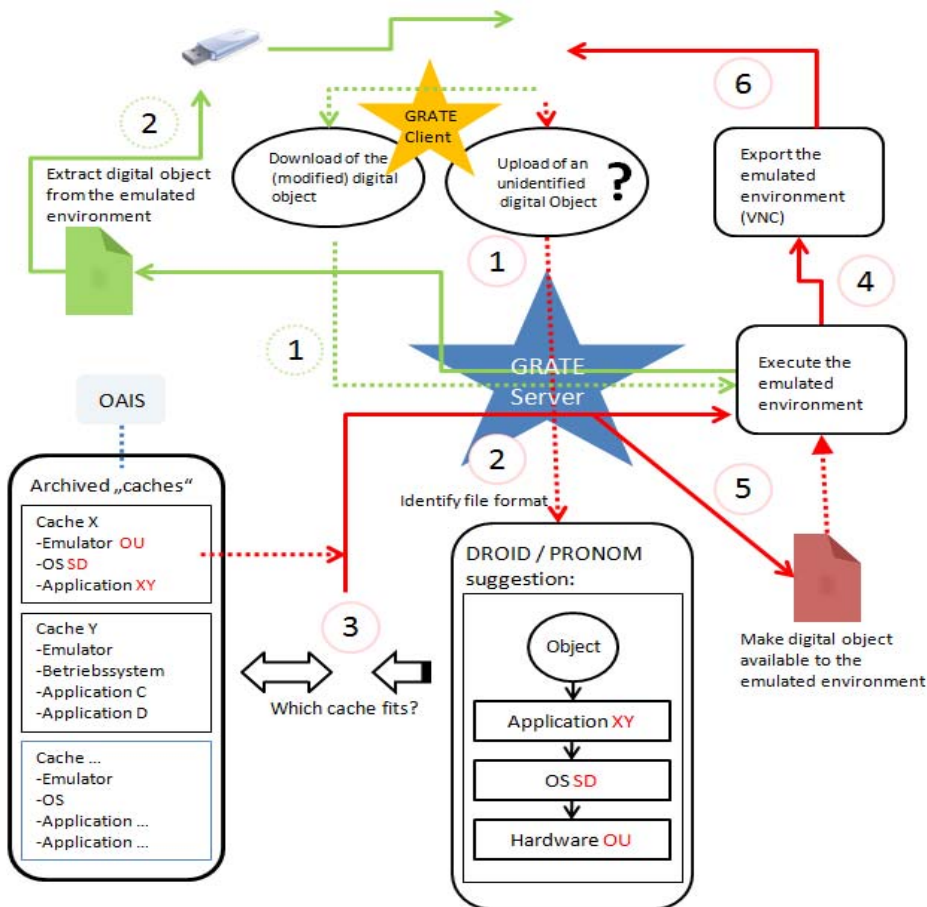


Figure 5: Model of the GRATE server.

GRATE client

The GRATE client consists of several Java applets (see fig. 3). The main applet provides a GUI (see fig. 1) with an overview of all installed emulators/environments. For each of them, meta information can be displayed in an HTML viewer. There is also the possibility to upload a digital object and let the system do something like an automated operation. The systems will try to identify the object (see 3.5.1) and will automatically execute a suitable environment. To do this, the applet forces the browser to load another page where a VNC applet is hosted. This applet connects to an already-running VNC server (see fig. 4).

3.6 Functionality

Version 0.1.1 of GRATE offers the following functionality:

Server-side:

- Embedding of several emulators/other software
- Controlling and logging sessions

Client-side:

- Displaying meta information for all (server-side) archived objects (HTML-viewer)
- Remote usage of emulator GUIs (or other GUI based software) via TightVNC inside a "normal" web browser. For the user there are no further installations necessary.
- Upload of local digital objects and their usage inside emulated environments (only implemented for QEMU yet)
- Download of (modified) digital objects back to the local file system of the user
- Auto-detection of these objects via DROID/PRONOM services.
- Auto-execution of suitable environments for the detected digital objects
- Execution of scripts/software running inside emulated environments (via sending of Keystrokes)

3.7 Technical environment

At present, the GRATE server only runs on Linux based systems. Since it is coded in Java/PHP/Perl it is possible to port it to other operating systems like Windows XP or Vista. The GRATE client runs on almost every Java-enabled Web-browser (Java >= 1.5.2). For other needed components see "used technologies":

<http://www.planets-project.eu/private/pages/wiki/index.php/GRATE>

3.8 Development notes

Java has been chosen as the main programming language because of its platform-independent features. For the client-side we have chosen the Java applets technology, since in most cases no additional software installations are necessary. The service can be used location independent with every web-browser. Other programming languages like PHP and Perl were used for deploying the Java applets or for the upload of files. JavaScript/Ajax technology was used for this purpose as well. All of the used programming languages are available without licensing fees as open-source software.

3.9 Future work

Within Planets, the core functionality of GRATE will not be extended. However, integration will take place with the PLANETS Interoperability Framework (IF) for which the GRATE application programming interface will be aligned with the PLANETS Service Oriented Architecture. This will help bring about close integration with the PLANETS components Testbed and PLATO.

Future development on GRATE's functionality can be expected within upcoming projects, such as KEEP (Keeping Emulation Environments Portable). Source code will continue to be made available which allows any interested developer to continue extending GRATE's capabilities.

4. References

- GRATE website and documentation: <http://www.planets-project.eu/private/pages/wiki/index.php/GRATE>
- [Longterm Preservation of Digital Objects](#) (presentation at the emulation expert meeting October 2006 in The Hague)
- [Emulation und ihre Randbedingungen als Langzeitarchivierungsstrategie dynamischer digitaler Objekte](#) (presentation of emulation and GRATE at the Workshop of the Nestor Media Working Group at the "Kunstgewerbemuseum Berlin" in March 2008.)
- ["Emulation: From Digital Artefact to Remotely Rendered Environments"](#) (Presentation at iPRES2008)
- ["Bridging the Past to the Future without Altering the Object"](#) (Presentation at [3rd Annual WePreserve Conference 2008](#) Nice)
- ["Bridging the Past to the Future without Altering the Object"](#) (Presentation and Demonstration at the PLANETS All Staff Meeting November 2008 in Marbella/Spain)