User Requirements Document (URD)

Computer emulator for digital preservation

Version : 1.1
Author : B. Lohman, J.R. van der Hoeven
Date : 21-02-2006
Project : Emulation project

Koninklijke Bibliotheek (National Library of the Netherlands)
Nationaal Archief of the Netherlands

I. Revision history

Revision number	Revision date	Author	Summary of changes
1.1	21-02-2006	B. Lohman	Minor changes to text; clarifications

II. Related documents

Document name	Date	Author
User Requirements Document [URD]	21-02-2006	B. Lohman
Emulation report by KB/NA [EMU]	20-06-2005	J.R. van der Hoeven

III. Table of contents

1 Introduction

1.1 Purpose of this Document

This document formally describes the user requirements for the conceptual model of a modular emulator. It is intended for review by the project manager at the Nationaal Archief of the Netherlands (NA), project member of the Koninklijke Bibliotheek, National Library of the Netherlands (KB), as well as the coordinator of the development team at Tessella Support Services plc.

1.2 Scope of this Document

The emulation report [EMU], written by the KB and NA in 2005, proposes construction of the modular emulator in three separate stages, with each stage specifying different user requirements. For efficiency reasons, a different approach has been presented by Tessella. However, the user requirements of both approaches are identical.
This document describes the requirements for the final concept of the modular emulator. Where necessary, user requirements defined in different stages will be identified as such. There are components described in the final concept that build upon results achieved in the previous stages. As the work has a research element, and project decisions will be made based on produced results during the project, it is difficult to estimate precisely how long the development of each stage will take. As a result, the implementation of these components will depend on time available.

Requirements for these modules are marked “O” (optional) and “E” (extensions). This does not imply they are not part of the scope of the project, but they will be determined at a later stage.

1.3 Definition of Terms

EMU	Emulation report by the KB and NA. See related documents
ESD	Emulator Specification Document, specifies the components of a particular emulator
KB	Koninklijke Bibliotheek, the National Library of the Netherlands
MSF	Module Specification File, specifies the hardware properties of an emulated component
NA	Nationaal Archief of the Netherlands
RWS	Reference Workstation
SRD	The Software Requirements Document, specifies the behaviour of the software system.
SMG	System Maintenance Guide, specifies how to create a development environment and create a release
URD	The User Requirements Document, catalogues the users’ requirements for the system (this document).
UVM	Universal Virtual Machine

2 General description

The goal of the project is to "…secure sustained accessibility to digital objects such as interactive multi media applications, PDF documents and database systems." These digital objects are currently accessed via a standard computer system. In this project, a model of the standard computer system, the Reference Workstation (RWS), will be used; its capabilities reflect those of the standard computer system. See Appendix B for more details.

For background on emulation the reader is referred to [EMU]. Definitions of technical terminology and jargon common to emulation and used in this document are included in the Glossary in Appendix A.

2.1 Project Context

This document lists requirements for the conceptual model of a modular emulator. Several emulators exist, most notably Bochs and QEMU, but neither implement an emulator with the same objective as desired in this project (see section 2.2). As both these systems are open source, and fall under the GNU Lesser General Public License, they can be used as a starting point for creating the emulator specified in this project.

The project consists of a number of stages, and will produce several versions during the course of development. Many of these versions should function as independent products, and will be used to assess the progress of the project and to determine the project's course.
The final product will depend on the approach used to achieve the most successful results; it is recognised that depending on the approach taken, not all user requirements may be met at the end of the project.

2.2 General Capabilities

Using the 'software-emulation-of-hardware' approach [EMU], the software will be capable of reproducing, as closely as possible, the hardware environment of the Reference Workstation. The priorities of this reproduction are, in order, execution speed, graphics, sound and network. The functional behaviour of the system of the software, compared to the RWS, will be assessed using various digital test objects.

Ideally, the system will consist of distinct modules that emulate specific RWS hardware components. Once an initial version of the system is produced that fulfils the above capabilities, it will be extended with a library of emulated components, and a controller to interconnect these components. This will make the system capable of emulating multiple environments, besides that of the RWS. The hardware properties of each emulated components will be specified in a Module Specification File, while particular combinations and usage of the components that make up the environment of an emulator will be specified in an Emulator Specification Document (ESD). The ESD can be associated with stored digital objects to easily access them in the required environment. The emulated components will be re-usable and will have the possibility to be enhanced.

The ideal final software product will be platform independent, able to fulfil the above capabilities on different host hardware configurations.

2.3 User Characteristics

The target group of the software product is end-users of computer software. Their intent is to run software packages to access, read and write various digital objects, including interactive multimedia applications, PDF documents, and database systems. These objects may no longer be supported on current hardware and software systems. This raises the need to install older, unsupported operating systems in order to run the software packages necessary for accessing the digital objects.

The group can be classified as users with computer experience varying between minimal and knowledgeable.

Those with minimal computer experience will have no knowledge how to install operating systems or software packages. They will need to be provided with a complete, existing environment that is targeted towards the documents they want to access, shielding them from requiring inside knowledge of the emulator. Assuming they have developed some familiarity with the environment through continual use, simple interaction can be expected to load, read, write and save the requested objects.

More experienced users can be expected to install the operating systems required for document access, as well as installation of software packages, once provided with the correct environment. Accessing more complex objects, more involved interactions within the system can be expected, resulting in higher loads to the system.

Knowledgeable users will want to create their own environments on the fly, tailor-made to their needs. Customising various modules to create specific settings, along with custom installation of operating systems can be expected. The system will be used to the fullest, placing the highest stress on the system. Interchanging modules within one environment to test and compare settings can be expected to lead to various configurations.

2.4 System Context

Provided below is a simplified diagram, showing the information and control flows across the system boundary.

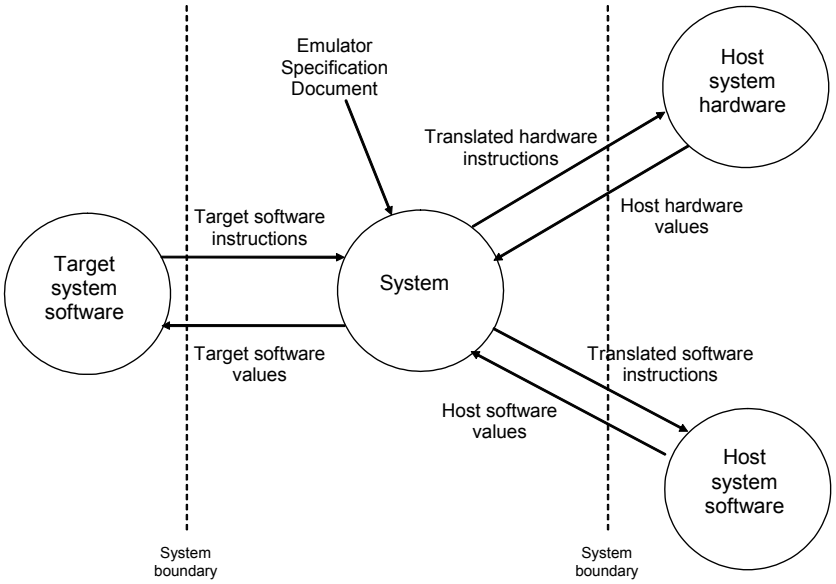

Figure 2.1: simplified diagram of system and its boundaries

2.5 Assumptions and Dependencies

There are several assumptions and dependencies which form the basis for some of the decisions following. These are the following:

2.5.1 Assumptions

- The UVM used will be the Java Virtual Machine, unless a preferable alternative is found, or initial development shows that the virtual machine approach is not feasible.
- JVM will continue to be widely used for the short to medium term and continue to be supported on a variety of platforms
- Taking in consideration the state of currently available emulators and their implementation, designing an emulator for portability using the JVM might introduce a serious overhead; this may influence the performance of the emulator in such a way that alternate strategies may need to be considered.
- All aspects of the design are assumed to be portable, that is that it can be implemented within the JVM. Initial investigations will test this assumption.

2.5.2 Dependencies

- Direct interaction with host hardware depends on the capabilities of the development language (Java).
- As suggested in section 2.5.1, developing an emulator that solely runs on the Java Virtual Machine creates a dependency on support from the manufacturer.
- Detailed knowledge of the inner workings of components is required, much of which may be considered proprietary by the manufacturer. Reverse engineering may need to be applied to gather enough information.
- Development will be incremental, so later stages depend on previous results. Given the limited timescale, prioritisation of tasks will be necessary to ensure the most important items are completed first.

2.6 Outstanding Issues

The list below indicates areas where knowledge is lacking or unclear. These areas should be resolved before the document is used as intended. The reason for listing them in this section is that they are not forgotten in the body of the document, and provide a readily available list for discussion when possible.

- Hardware abstraction layer (HAL) emulation (low level) needs to be researched as an alternative to the current JVM approach (high level).
- With the JVM as suggested approach, it needs to be seen whether this offers enough functionality and speed that is desired.
- The release license specifics need to be determined.

3 Functional Requirements - Emulator

This section contains all the users' functional requirements with regard to the emulator aspect of the system. Each requirement is prioritised as follows:

M Mandatory requirement. This feature must be built into the final system.
D Desirable requirement. This feature should be built into the final system unless its cost is too high.
O Optional requirement. This feature can be built into the final system at the Project Manager's discretion.
E Possible future enhancement. This feature is recorded here so that the idea is not lost. The decision on whether to include it in the system will depend on progress on the mandatory requirements.

3.1 Processing devices

Label	Requirement	Necessity
U3.1.1	The software should provide code that emulates a processor comparable to a socket 478 Intel Pentium 4 single core at 2.4 GHz. Issues: • What if desired speed is not reached in emulation?	M
U3.1.2	The software should provide code that emulates a motherboard comparable to a Compaq EVO D510 CMT with [type] chipset, [speed] Front Side Bus. Issues: • Design question: is this part necessary to emulate?	D

3.2 Output devices

Label	Requirement	Necessity
U3.2.1	The software should provide code that emulates a display adapter comparable to a nVidia GeForce MX 420 with 32MB memory. Issues: -	M
U3.2.2	The software should provide code that emulates sound comparable to the AC'97 specification. Issues: • This is a specification, not a hardware component – is this clear enough?	D
U3.2.3	The software should provide code that emulates a network adapter comparable to the Intel Pro/1000 T. Issues: -	O

3.3 Read / write devices

Label	Requirement	Necessity

U3.3.1	The software should provide code that emulate a hard disk, comparable to a 40GB Maxtor 6E040L0 or 40GB Western Digital WD400BB-60CJA0 Issues: -	M
U3.3.2	The software should provide code that emulates random access memory up to 1 GB. Issues: -	M
U3.3.3	The software should provide code that emulates an optical storage device, comparable to a Compaq JLMS DVD-ROM LTD-1665. Issues: -	D
U3.3.4	The software should provide code that emulates a 3.5” floppy drive, with average seek time. Issues: -	E

3.4 Input devices

Label	Requirement	Necessity
U3.4.1	The software should provide code that emulates a keyboard, comparable to IBM PC 101-key keyboard. Issues: -	M
U3.4.2	The software should provide code that emulates a mouse, supporting the PS/2 protocol. Issues: • This does not specify anything about the mouse, only the protocol	M

3.5 Port devices

Label	Requirement	Necessity
U3.5.1	The software should provide code that emulates a communications (COM) port. Issues: • Hardware specifications?	M
U3.5.2	The software should provide code that emulates a parallel communications (LPT) port. Issues: • Hardware specifications?	D
U3.5.3	The software should provide code that emulates a Universal Serial Bus (USB) port. Issues: • Hardware specifications?	E

3.6 External files

Label	Requirement	Necessity
U3.6.1	The software should be able to read and write Emulator	M

	Specification Documents (ESDs), which contain a list of components that make up a specific emulator. Issues: -	
U3.6.2	The Emulator Specification Document should contain properties for the emulator that are user-configurable Issues: -	D
U3.6.3	Each emulated component should have an associated Module Specification File (MSF) or metadata that specifies the hardware properties of that component. Issues: -	M

3.7 User interface characteristics

Label	Requirement	Necessity
U3.7.1	An Emulator Specification Document (ESD) can be created via a user interface. Issues: -	D
U3.7.2	A specific emulator should be started when an Emulator Specification Document (ESD) is loaded in the user interface. Issues: -	D
U3.7.3	The user interface contains logic to determine compatibility between components when loading / creating an Emulator Specification Document (ESD). Issues: -	E

3.8 Target software

Label	Requirement	Necessity
U3.8.1	The software should be able to host operating systems comparable to Microsoft Windows 2000 Professional. Issues: -	M
U3.8.2	The software should be able to host applications software within the operating system capable of running the test objects. Issues: • List of applications necessary?	M

3.9 Components

Label	Requirement	Necessity
U3.9.1	There must be at least one emulated component (and associated Module Specification File) for every type of hardware component listed in section **Fout! Verwijzingsbron niet gevonden.**. Issues: -	M
U3.9.2	It should be possible to refine existing emulated components for	D

	highest compatibility, while keeping the previous versions. Issues: -	

4 Functional Requirements – Library

This section contains all the users' functional requirements with regard to the library aspect of the system. Each requirement is prioritised as follows:

M Mandatory requirement. This feature must be built into the final system.
D Desirable requirement. This feature should be built into the final system unless its cost is too high.
O Optional requirement. This feature can be built into the final system at the Project Manager's discretion.
E Possible future enhancement. This feature is recorded here so that the idea is not lost. The decision on whether to include it in the system will depend on progress on the mandatory requirements.

4.1 Component management

Label	Requirement	Necessity
U4.1.1	Emulated components should be organised in a library, associating emulated component code with Module Specification Files. Issues: -	M
U4.1.2	The library should maintain a component list consisting of Module Specification Files (MSF) or metadata that can be used in an emulator. Issues: -	M
U4.1.3	The component list should be updateable with newer components, while maintaining a versioning system of modified emulated components. Issues: -	M
U4.1.4	The library should be able to provide the emulator with the emulated component code based on information from the Emulation Specification Document (ESD). Issues: -	M

4.2 Component use

Label	Requirement	Necessity
U4.2.1	It should be possible to re-use emulated components for multiple emulator configurations. Issues: -	M

4.3 Interface

Label	Requirement	Necessity
U4.3.1	There should exist a mechanism ("controller") capable of interconnecting modules that results in a working emulator.	M

	Issues: -	
U4.3.2	The process of interconnecting modules should be automated. Issues: -	D

5 Non-functional Requirements

5.1 Speed and Time Requirements

Label	Requirement	Necessity
U5.1.1	The software should be able to emulate the RWS at a reasonable speed. However, this is not a mandatory requirement because it is assumed that computers will continue to become faster (based on Moore's law). As this emulator will be focused on future use, speed is currently not an issue. Issues: How fast is Java?	D

5.2 Capacity Requirements

Label	Requirement	Necessity
U5.2.1	The software must be able to handle at least 40 GB target disk images. Issues: -	M

5.3 Reliability Requirements

Label	Requirement	Necessity
U5.3.1	The software should be able to give a correct representation of the test objects. Issues: this requires comparison tests between RWS and emulated environment.	M

6 Design and Implementation Constraints and Standards

6.1 Safety

There are no relevant requirements in this section.

6.2 Security

There are no relevant requirements in this section.

6.3 Target Platforms

Label	Requirement	Necessity
U6.3.1	The software should be able to run on the x86 platform. Issues: -	M
U6.3.2	The software should be able to run on the PowerPC platform. Issues: -	E
U6.3.3	The software should be able to run on any platform that the emulation approach supports (i.e. if the Java Virtual Machine is chosen, any platform that supports JVM). Issues: -	E

6.4 Development Tools

Label	Requirement	Necessity
U6.4.1	The software design and development will be based existing emulators such as Bochs and QEMU, or any virtualisation technology. Issues: -	D
U6.4.2	Java should be used as a virtual layer. Issues: -	D

6.5 Project Requirements

Label	Requirement	Necessity
U6.5.1	Emulation components should be defined as distinct modules – “a finite set of programming code that, when executed, emulates a specific part of a computer system”. Issues: • It is recognized that this approach makes a trade-off for speed in favour of modularity	M
U6.5.2	The software should be constructed in a modular way. Issues: -	M

U6.5.3	Object-oriented methodology should be used in designing the software. Issues: -	M
U6.5.4	Knowledge, expertise and code should be shared with the open source community. Issues: -	M
U6.5.5	All emulator information should be documented, including device specifications, protocol descriptions and standards. Issues: -	M
U6.5.6	All modules should be documented. Issues: -	M
U6.5.7	The software should be able to read and write the format of disk images that are used as test objects. This format is equivalent to the disk images used in Bochs and QEMU. Issues: • Is a description of this format available for design purposes?	M
U6.5.8	Differences in performance between the RWS and the modular emulator should be made available. Issues: -	D

7 Delivery Requirements

7.1 Delivery

Describe any constraints or requirements the client has specified relating to how the system should be delivered to them.

Label	Requirement	Necessity
U7.1.1	The emulator should be released under the GNU (Lesser) General Public License or compatible license. Issues: • To be decided officially when publicly releasing first set of code.	M
U7.1.2	The project should be delivered on the next working day after July 1, 2007 Issues: -	M

7.2 Post-Delivery

Describe any constraints or requirements the client has specified relating to activities which are to take place following successful delivery of the system

Label	Requirement	Necessity
U7.2.1	Other interested parties should be able to freely use, modify and distribute the developed emulator in accordance to the license it is released under. Issues: -	M
U7.2.2	Future developers should be able to recreate the emulator, using the System Maintenance Guide (SMG) document along with the available source code. Issues: -	M

Appendix A Glossary

This glossary contains definitions of terms used within the User Requirement Specification document. It defines terms that are used within the emulation context, but because of their common use or multiple meanings, require clarification.
[EMU] and Wikipedia have been used as a basis for some of the definitions.

Environment	The collection of logical and physical resources used to support a platform
Host	The platform providing the emulator access to its services and hardware for it to run
Logical resources	The manifestation of physical devices in software, including operating systems.
Platform	The framework, either in hardware or software, which allows software to run. This typically includes architecture, operating system, and programming languages and their runtime libraries.
Target	The platform that is to be emulated, using a host for its resources

Appendix B RWS characteristics

Software		
Application	*Version*	*Files*
MS Windows 2000 Professional	Service Pack 3	w2ksp3.exe
MS Direct X for NT	8	DX80NTeng.exe
Display Drivers (Compaq nVidia LP AGP GeForce2MX-400)	6.13.10.4103	41.03_win2kxp.exe
Intel Chipset		SP22545.exe
Intel Ide		SP21356.exe
Intel pro 1000T		pro2kxpm.exe
Keyboard		SP21795.exe (not used because it only enables the function keys on the top of keyboard)
Mouse		SP21424.exe
Sound		SP22286.exe
Additional Software		
Application	*Version*	*Files*
MS Internet Explorer	5.50.4134.0600	ie552000 directory, setup is started via ie5setup.exe
MS Windows Media Player	7.1	mp71.exe
MS SysPrep (in deploy.cab)	1.1	W2KCD:\support\tools\deploy.cab and Windows 2000 System Preparation Tool (version 1.1) from Microsoft (sysprep_update)
Power Quest Drive Image Pro	4.0	Drive Image Pro 4.0 directory, subdirectory dp40en, subdir setup, contains setup.exe
Adobe Acrobat Reader	5.01	Acrobat Reader 5.01 directory, rp501enu.exe (only used on development to read the documentation)
Java Plug-in for the browser [=IE]	1.3.1_02 Standard Edition International	j2re-1_3_1_02-win-i.exe
InterVideo WinDVD	2000	Install CD delivered with DVD-ROM Drive (COMPAQ JLMS DVD-ROM LTD-1665)
Paragon CD-ROM emulator	2.5 Personal Edition	Install cd : iso ‘paragon cd.iso’
Hardware		
Component	*Specification*	
Main board	Compaq EVO D510 CMT	
Processor	Intel Pentium 4 system running at 2,4GHz	
Internal memory	1 GB RAM	
Hard disk devices	2 internal disks: 40 GB Maxtor 6E040L0 40 GB WDC WD400BB-60CJA0	
Display adapter	nVidia GeForce4 MX 420 display adapter Fill Rate : 1 Billion Texels/Sec.	

	Triangles per Second : 31 Million Memory Bandwidth : 2.7GB/Sec. Maximum Memory : 32MB
Sound device	SoundMAX AC97 Integrated Digital Audio
Optical storage devices	DVD-ROM Drive (COMPAQ JLMS DVD-ROM LTD-1665), Region 2
Floppy disk drives	3.5”floppy drive
Ports (external)	4 Universal Serial Bus ports (USB) 2 serial communications ports (COM) 1 parallel communications port (LPT)
Network adapters	100 Megabit Ethernet network adapter 1 Gigabit Ethernet network adapter
Monitor	CTX S500B *Viewing* Display Technology : Active Matrix TFT Panel Display Panel : 15" (Diagonal) Resolutions : 640 x 350 @ 70Hz 720 x 400 @ 70Hz 640 x 480 @ 60Hz, 72Hz, 75Hz 800 x 600 @ 60Hz, 72Hz, 75Hz 1024 x 768 @ 60Hz, 70Hz, 75Hz 1024 x 768 (Max) Contrast Ratio : 400:1 Brightness (Typical) : 250 cd/m2 Viewing Angle (H/V) : 120/100 (degrees) Pixel pitch : 0.297 mm Scanning characteristics : 30kHz - 60kHz (Hor. Freq.), 58Hz ~ 75Hz (Vert. Freq.) Response time : tr:13ms / tf:27ms Maximum color support : 16.7 million *Bodywork* Cabinet Color(s) : Black VESA compliant : yes Plug & Play : yes Power usage : 35W maximum Dimensions : 14.65" W x 14.02" Hx 6.77" D Weigth : Net: 8.16 lbs. Gross: 14.1 lbs Inputs signals : Video= RGB Analog (0.7Vp-p) Sync= H/V Separated (TTL) User controls : Front Panel Controls: Power Switch, LCD Indicator, ESC, Up, Down, Enter On-screen menu controls : Contrast, Brightness, Auto Tune, Color Temp, Size, Phase, Focus, Dithering, Text/Gfx, Position, Languages, Recall Environments : PC and Mac *miscellaneous* Safety regulations : FCC Class B, UL, cUL, CE, TCO'99
Keyboard	Compaq US keyboard for Microsoft Windows
Mouse	Compaq PS/2 (Logitech) mouse, wheel, wired, mouse ball