FP7 – ICT – GA 216946

D5.5 Proof of concept demonstration definition, second version

Deliverable Type *	:	**PU**
Nature of Deliverable **	:	**R**
Version***	:	**Released**
Created	:	**30 September 2009**
Contributing Workpackages	:	**WP5**
Editor	:	**Benoit Lécroart**
Contributors/Author(s)	:	**WP5 team**
File Name	:	**[MobiThin_D5.5_POC_Demonstrator_definition_NTUK_30Sept09]**

*** Deliverable type:** PU = Public, RE = Restricted to a group of the specified Consortium, PP = Restricted to other program participants (including Commission Services), CO= Confidential, only for members of the MobiThin Consortium (including the Commission Services)
**** Nature of Deliverable**: P= Prototype, R= Report, S= Specification, T= Tool, O = Other.
*****Version**: Preliminary, Draft 1, Draft 2,…, Released

Abstract:
This document defines the demonstrations completing the second phase of the MobiThin project. Demonstrations are defined at two levels: an end-to-end system level demonstration and specific demonstrations focusing on particular protocol optimization strategies.

The end-to-end system demonstration will allow assessing the MobiThin architecture on a functional level as well as in terms of scalability, and inter-working between management level adaptation and protocol adaptation.
To allow validation of the MobiThin solutions on the actual hardware (actual thin client terminals and non-emulated network links), several additional demonstrations are described. The inter-working of thin client with SIP, as a first step towards integration into the IMS architecture, is targeted as demonstration.

For each of the six demonstrators identified, the specific objectives, the test environments and the targeted experiments are described. Care has been taken to address all major results of MobiThin in the demonstrators detailed in this document.
The realization of the different demonstrations will bring the MobiThin approach substantially closer to deployment in an operational environment.

"The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 216946"

Keyword List: Proof of Concept, Demonstration

The MOBITHIN Project Consortium groups the following Organizations:

Interdisciplinary Institute for BreedBand Technology vzw	IBBT vzw	B
Deutsche telekom Laboratories	DT Labs	G
Prologue Software	Prologue	F
Interuniversitair Micro-Electronica Centrum vzw	IMEC vzw	B
NEC Technologies (UK) Ltd	NTUK	UK
Groupe des Ecoles des Télécommunications	GET	F
JCP-Consult SAS	JCP	F

Information Society Technologies

SEVENTH FRAMEWORK PROGRAMME

Contents

1. Executive Summary ... 4
2. Introduction ... 4
3. Scope ... 5
4. Scenario ... 5
5. architecture ... 5
6. POCs demonstrations lists ... 6
7. POCs demonstrations descriptioN ... 7
7.1 Integrated Demonstration ... 7
7.1.1 Objectives ... 7
7.1.2 Description of the test environment ... 7
7.1.3 Experiments to be performed ... 8
7.2 Multimedia Thin Client based on SIP ... 11
7.2.1 Objectives ... 11
7.2.2 Description of the test environment ... 12
7.2.3 Experiments to be performed ... 14
7.3 Effects of Constraints on Image Transmission ... 14
7.3.1 Objectives ... 14
7.3.2 Description of the test environment ... 15
7.3.3 Experiments to be performed ... 16
7.4 Link Optimization NS2 Emulation ... 17
7.4.1 Objectives ... 17
7.4.2 Description of the test environment ... 17
7.4.3 Experiments to be performed ... 18
7.5 Link Optimization Hardware Emulation (XMSF) ... 18
7.5.1 Objectives ... 18
7.5.2 Description of the test environment ... 19
7.5.3 Experiments to be performed ... 20
7.6 Remote devices / Peripherals ... 21
7.6.1 Objectives ... 21
7.6.2 Description of the test environment ... 21
7.6.3 Experiments to be performed ... 22

EDITORIAL Change Management

Version	Date	Editor/Author	Comments
Preliminary	2/9/2009	Benoit Lécroart-NEC Tech (UK)	Creation of document.
Draft	18/9/2009	Benoit Lécroart-NEC Tech (UK)	First complete draft
Draft	21/9	Bert Vankeirsbilck Benoit Lécroart	Merge Wan demo Minor wording updates
Draft	25/9	IBBT, IT, Prologue, NTUK	Final updates
Release	29/9	Benoit Lécroart-NEC Tech (UK)	Final cleanup

1. EXECUTIVE SUMMARY

This document defines the demonstrations that will be performed during the second phase of the MobiThin project.
The demonstrations are extending the demonstration performed during the first phase of the project, with the aim of concretely demonstrating the benefits of MobiThin project results.
Demonstrations are defined at two levels: an end-to-end system level demonstration and specific demonstrations focusing on particular protocol optimization strategies. They will represent a major step forward toward the deployment of Thin Client services over real mobile networks

Six demonstrations are identified.

- Integrated Demonstration
  - Aims to demonstrate key features developed through the project in a single demonstration. It aims to be a first step toward the development of a future Thin Client service over wireless networks. The significant part of the Thin Client system will be demonstrated. The main feature demonstrated will be the dynamic protocol adaptation to the network impairment, the infrastructure energy adaptation to the client needs, and the overall management of the Thin Client system.

The other demonstration will focus on more specific technical research topics showing also the outcome of the project.

- Multimedia Thin Client based on SIP
  - Aims to demonstrate an implementation of a Thin Client system supporting Multimedia services over a SIP based Service Delivery Platform infrastructure. This demonstration is a first main step toward the realization of Thin Client service over IMS operated network using the full network QOS capabilities. It will be based on the IMS architecture study performed during the phase 2.

- Effects of Constraints on Image Transmission
  - Aims to demonstrate the benefits of Mobithin Thin Client image transmission. A comparison of the state of the art image compression protocols (BiFS, LASer) used in a thin client environment will be performed with the legacy thin protocol VNC. The demonstration will attach to measure the benefits of each solution in term of bandwidth, resilience to the wireless-due packet losses, and CPU client consumption. The demonstration will be performed both in labs controlled network environment, and also on a real operated UMTS network.

- Link Optimization NS2 Emulation
  - Aims to perform measurement of the wireless link optimization developed during the project. The test environment uses a simulated SDR (Software Define Radio) platform based on NS2 simulator. This is a demonstration with enhanced features of the phase 1 Wireless medium optimization demonstration working with “live” traffic from upper layers.

- Link Optimization Hardware Emulation (XMSF)
  - Aims to improve the accuracy of the precedent demonstration, by using a Cross Layer Simulation framework containing more precise models of the physical components. All those demonstration will allow measuring the energy and consumption gain of the wireless components due to the project developed optimizations, without the flexibility of the NS2 simulator.

- Remote devices / Peripherals
  - Aims to demonstrate the capacity to remotely access and use devices connected to terminals, in a Thin Client server environment. The demonstration will first show the capability of accessing data from a remote a peripheral connected to a thin client. It will then show a demonstration of the use by a thin client, of a remote peripheral connected to an other thin client

2. INTRODUCTION

The objective of the WP5 of the MobiThin project is to design, construct and evaluate various demonstrations, in order to validate the MobiThin concepts and implementation, as well as to feed project dissemination actions. This validation will be done by both simulation and emulation, as well as through well-selected lab trials.

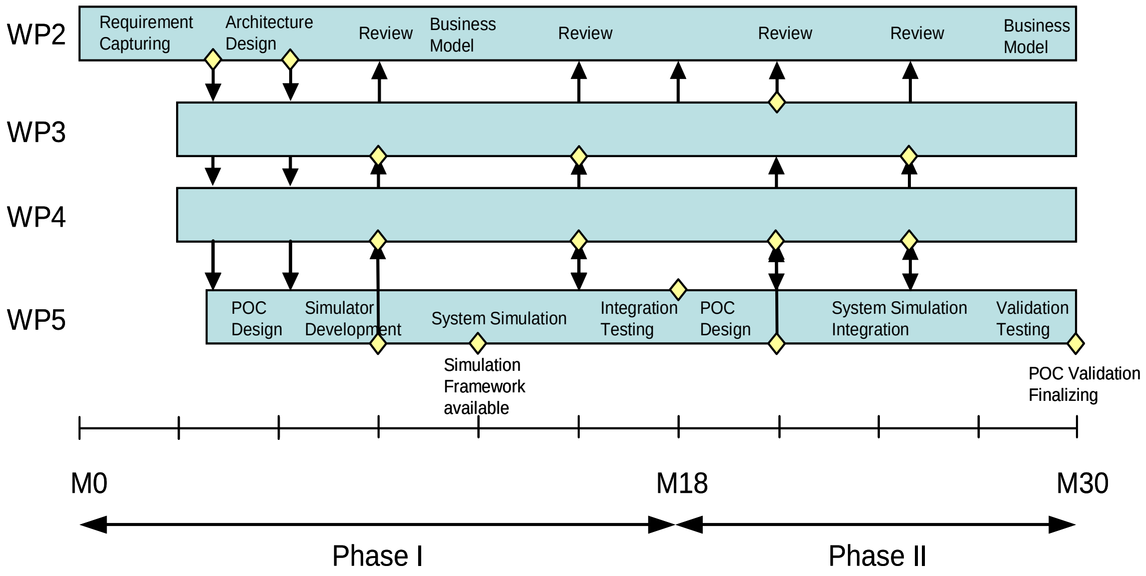


Figure 1: WP5 timing and relation with other work packages

During the phase 1 of MobiThin, WP5 played an important role of integrating all the components developed in WP3 and WP4, and by performing their validation based on the requirements and architecture identified in WP2. The results obtained in WP5 have been used for disseminating the project results through the WP6.

During the phase 2 of Mobithin, the demonstration performed will extend the demonstration performed in phase 1 to identify the progress of Mobithin project toward the development of Mobile Thin Client services.

3. SCOPE

This document identifies the demonstrations aiming to validate the components developed in work package 3 and 4 against the project objectives. The document identifies the demonstrations that will be performed during the last phase of the project that will complete at M30.

The main partners involved in the development of components in WP3 and WP4 have identified different demonstrations for this purpose.

For each demonstration, the following information is given:

- The objectives of the demonstration, showing also the links with the MobiThin Scenarios identified in D.2. The demonstrations are defined based on the interim phase 1 project results such as the D2.4 system review document, and the D5.3/D5/4 results achieved.
- A description of the test environment, with a logical view of the component and interfaces concerned, based on the system architecture defined in D2.2
- A physical view of the test environment
- The experiments that will be performed with the expected results

4. SCENARIO

The following scenarios have been identified in the requirements documents D2.1:

- Scenario 1: Bringing the hospital to the patient’s home
- Scenario 2: Getting in touch with Paris
- Scenario 3: Keep scoring at school
- Scenario 4: Mobile thin office

The demonstrations identified in this document, aim to validate that the MobiThin system developed in WP3 and WP4 will be capable of realizing significant parts of the scenarios.

5. ARCHITECTURE

The architecture of the system is defined in D2.2.

In the Figure 2, the high-level MobiThin software architecture is illustrated as defined in D2.2. Six basic building blocks have been identified: the MobiThin Client process, the MobiThin Server Process, the Service Management Framework (SMF), the Operator Network Service, the Data Storage Service and the Application Delivery Service.

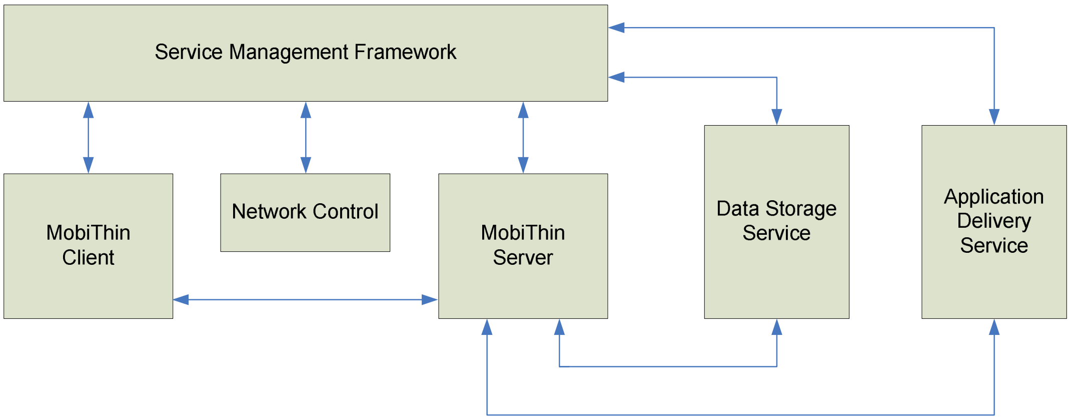


Figure 2. High-level Software Architecture

A high-level perspective, the functionality of the 6 basic blocks is summarized in the table below.

Basic Building Block	Description
MobiThin Client	This is the MobiThin software running on the mobile client device. When a user wants to start the thin client service, he starts the MobiThin client on his thin client device.
Service Management Framework (SMF)	When a user logs in, he has to identify himself to the SMF, which is responsible for managing the complete thin client service and guaranteeing the desired QoS to the users.
MobiThin Server	This is the MobiThin software running on the Thin Client Server selected by the SMF. As defined in Deliverable D2.1, all applications of the users are executed in their MobiThin Server. Audiovisual output from a user's applications is transported over the network to the user's device. Input from the user is transported in the opposite direction.
Network Control	This is the service of the network operator. The SMF will interact with the Network Control block to set the appropriate QoS classes for the traffic between the MobiThin Client and the MobiThin Server.
Data Storage Service	This service maintains the personal data of the users.
Application Delivery Service	For scalability reasons (not every application can be installed in every MobiThin Server), applications are delivered by an application delivery service to the MobiThin Server.

6. POCS DEMONSTRATIONS LISTS

In order to verify the scenarios identified in chapter 4, the following POC demonstrations have been identified. Each POC demonstration will attach to specific features or functions. This chapter presents the lists and main objectives of each POC demonstration.

- Integrated Demonstration
  - Aims to demonstrate major key features develop by the project in a single demonstration. It aims to be a first main step toward the development of future Thin Client service over wireless networks. Significant part of the Thin Client system will be demonstrated. The main feature demonstrated will be the dynamic protocol

adaptation to the network impairment, the infrastructure energy adaptation to the client needs, and the overall management of the Thin Client system.

- Multimedia Thin Client based on SIP
  - Aims to demonstrate an implementation of a Thin Client system supporting Multimedia services over a SIP based Service Delivery Platform infrastructure. This demonstration is a first step towards the realization of a Thin Client service over an IMS operated network using the full network QOS capabilities. It will be based on the IMS architecture study to be performed during phase 2.

- Effects of Constraints on Image Transmission
  - Aims to demonstrate the benefits of Mobithin Thin Client image transmission. A comparison of the state of the art image compression protocols (BiFS, LASer) used in a thin client environment will be performed with the legacy thin protocol VNC. The demonstration will attach to measure the benefits of each solution in term of bandwidth, resilience to the wireless-due packet losses, and CPU client consumption. The demonstration will be performed both in labs controlled network environment, and also on a real operated UMTS network.

- Link Optimization NS2 Emulation
  - Aims to perform measurement of the wireless link optimization developed during the project. The test environment uses a simulated SDR (Software Define Radio) platform based on NS2 simulator. This is an enhanced demonstration of the phase 1 Wireless medium optimization demonstration that works with "live" traffic from upper layers.

- Link Optimization Hardware Emulation (XMSF)
  - Aims to improve the accuracy of the precedent demonstration, by using a Cross Layer Simulation framework containing more precise models of the physical components. All those demonstration will allow measuring the energy and consumption gain of the wireless components due to the project developed optimizations of phase I. This simulator shows the low-level cross-layer gains, but without the flexibility of the NS2 simulator.

- Remote devices / Peripherals
  - Aims to demonstrate the capacity to remotely access and use devices connected to terminals, in a Thin Client server environment. The demonstration will first show the capability of accessing data from a remote a peripheral connected to a thin client. It will then show a demonstration of the use by a thin client, of a remote peripheral connected to another thin client.

7. POCS DEMONSTRATIONS DESCRIPTION

7.1 INTEGRATED DEMONSTRATION

7.1.1 Objectives

Demonstrate a complete End-to-End system on distributed infrastructure (Virtual Wall). The objective is to demonstrate some of the adaptivity mechanisms of the MobiThin system. This integrated demo will show some pure WP3 mechanisms, some pure WP4 mechanisms but also the inter-working between WP3 and WP4 to tackle certain problems.

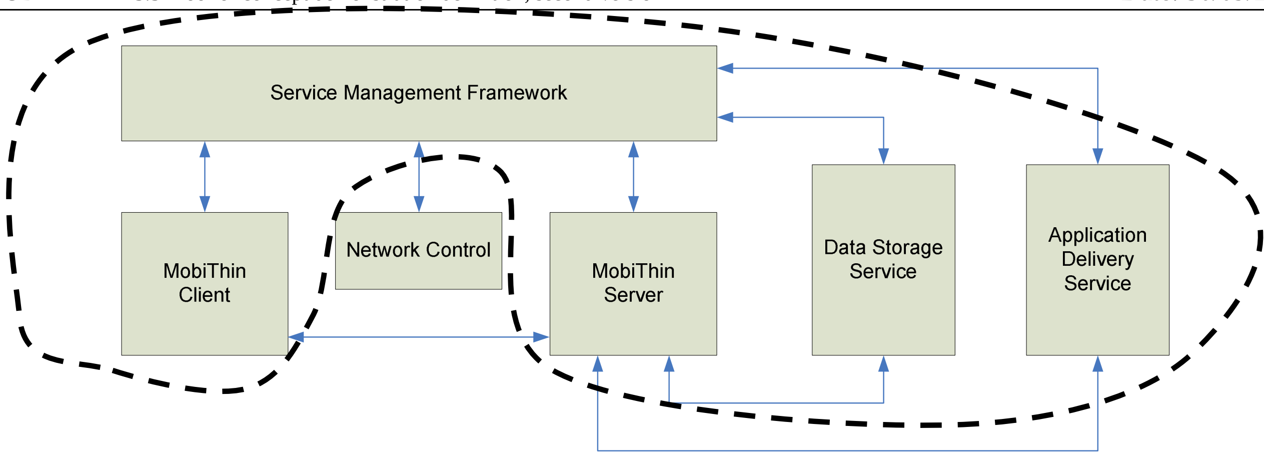
Relation to scenarios: The integrated demo impacts the scenarios in multiple ways.

- It demonstrates how a user can access his user session from different terminals, as mentioned in "Bringing the hospital to the patient's home" and "Mobile Thin Office"
- It demonstrates how the complete MobiThin system cooperates to guarantee sufficient QoS, adapted to the available network resources. This contributes to realizing the scenarios "Getting In Touch With Paris" and "Mobile Thin Office". It demonstrates how the MobiThin management framework dynamically allocates sufficient resources, e.g. when a resource-intensive application is started. This is related to "Bringing the Hospital to the Patient's Home" and "Keep Scoring at School"

7.1.2 Description of the test environment

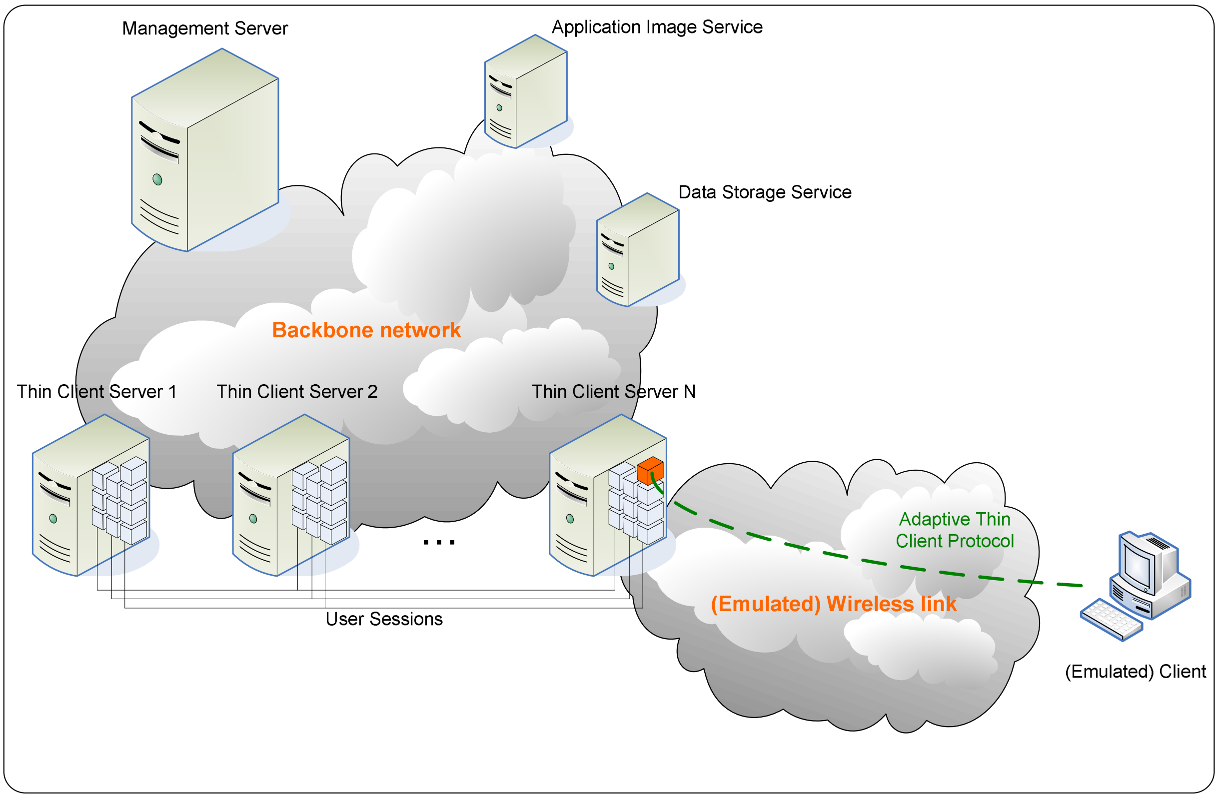
7.1.2.1 LOGICAL VIEW

The next figure identifies the architecture blocks relative to the demonstration.



All of the MobiThin components with the exception of the network control will be involved in this demonstration.

7.1.2.2 PHYSICAL VIEW



7.1.3 Experiments to be performed

First of all, this integrated demo aims to show a client can connect to the MobiThin system and is automatically redirected to an appropriate server able to host the user's session. The other aim of this demonstration includes showing the adaptivity of the MobiThin system.

We can distinguish three adaptivity parts in this demonstration:

- Pure WP3 adaptations
- Pure WP4 adaptations
- Adaptations in which both WP3 and WP4 are involved

7.1.3.1 END-TO-END SYSTEM

Some possible actions that can be demonstrated:

Nb	Scenario description	Expected result	Infeed	Comments
01	User logs in to the MobiThin system	The user's VM gets started and MobiThin viewer is started on the client	Hosting infrastructure client device Network impairment node	Try to show graphs of monitoring, resource reservations, etc.
02	The user wants to switch devices (for example because his device is running out of battery). Therefore, he logs off his session and chooses to "suspend"	Session is stored to be resumed later.	Idem Nb01	
03	The user logs in with his new device	Same as Nb.01, but now the VM is resumed	idem Nb01	The user gets the same session as before (but screen adapted to the capabilities of the new device)
04	The users finishes his session	The business component can "charge" the user for this session based on logging information	idem Nb01	

7.1.3.2 PURE WP3 ADAPTATIONS

In the demonstration, some of the adaptation mechanisms developed within WP3 will be shown. These mechanisms are related to the dynamic adaptation of the thin client protocol characteristics to the current status of the device, network and server.

Some possibilities:

- **Adapt network bandwidth usage in downstream direction**

Nb	Scenario description	Expected result	Infeed	Comments
01	The available network bandwidth is increased	The thin client protocol increases the rate of display updates	Impairment node + WP4 monitoring framework	Try to show graphs of monitoring and frame rate.
02	The available network bandwidth is decreased	The thin client protocol decreases the rate of display updates	Impairment node + WP4 monitoring framework	Try to show graphs of monitoring and frame rate.

- **Adapt network bandwidth usage in upstream direction**

Nb	Scenario description	Expected result	Infeed	Comments
01	The current network delay is increased.	User event transmission mechanism is adapted.	Impairment node + WP4 monitoring framework	Try to show graphs of monitoring and user event buffering.
02	The upstream bandwidth availability is varied.	User event transmission mechanism is adapted accordingly.	Impairment node + WP4 monitoring framework	Try to show graphs of monitoring and user event buffering.

- **Adapt encoding to available decoding resources at the client**

Nb	Scenario description	Expected result	Infeed	Comments
01	Client network load is increased	Server adapts	Load	Try to show graphs of total

		encoding parameters to a less resource-demanding process.	generator on client + WP4 monitoring framework	client load

7.1.3.3 PURE WP4 ADAPTATIONS

In the demonstration, some of the following pure WP4 adaptations can be shown:

- ***Shutdown TCS for maintenance***

Nb	Scenario description	Expected result	Infeed	Comments
01	Administrator chooses to shutdown a certain TCS for maintenance.	The SMF should identify the server(s) onto which migrating the active VMs. The SMF should then proceed and perform the required migration.	All WP4 components	The VMs running on the original TCS need to be re-distributed to run on other available servers. In order for each VM to be able to migrate, a decision resulting from collaboration between different SMF components must have been taken. This decision should take into account the currently available resources, the capacity of each server and the predictions of possible future load resulting from the data collected by the profiling manager.
02	When all VMs are migrated, the TCS shuts down and maintenance operations can be executed	Once the VMs have been migrated, the server should be removed from the pool of running servers	All WP4 components	The status of the server should be updated and the reason for the shutdown should be registered by the monitoring framework. All SMF components running on the TCS should cleanly disconnect established communications with the other SMF components and also with the external world, particularly with data storage servers.

- ***Energy-efficiency in the datacenter***:

Nb	Scenario description	Expected result	Infeed	Comments
01	Average load on the TCS servers drops. Some TCS can be powered off to save energy (after migrating the active VMs to other TCS)	The SMF should identify the server(s) to power off and the servers onto which migrating the active VMs. The SMF should then proceed and perform the required actions (VMs migration and TCS power off).	All WP4 components	Under certain circumstances like low average load of certain TCS servers, the SMF self logistics manager might decide to re-distribute connected users on reduced number of servers and power off the other servers to save energy. As for the "maintenance" scenario described above, the decision will be taken based on the data provided by all the other SMF components.
02	Average load on the TCS servers raises	The SMF should start additional TCS server(s).	All WP4 components	Depending on the load conditions, the SMF might either migrate a number of already running sessions to the

				new servers, or simply redirect new connections to those servers.

7.1.3.4 WP3/WP4 ADAPTATIONS

In the demonstration, we will show the hierarchical organization of the SLM infrastructure. If a problem cannot be solved at a specific architectural level, the SLM at a higher level is contacted.

The following scenario is envisaged.

Nb	Scenario description	Expected result	Infeed	Comments
01	The user is running an office application	Sufficient QoE	hosting infrastructure client device	
02	The user starts a resource-intensive application (e.g. he wants to take a screenshot of a 3D model to paste in his document).	Due to changed characteristics of the graphics, MobiThin protocol changes the encoding mechanism	load generator	
03	The monitoring notices there are not enough resources available.	The MobiThin protocol must resort to a suboptimal graphical quality.	monitoring framework	
04	The monitoring notices the QoE is not high enough	The SLM@TCS is notified the US runs out of resources	idem Nb03	
05	The TCS does not have resources available to enlarge the user's VM	The SLM@MS is notified the TCS runs out of resources	idem Nb03	
06	The MS decides to migrate sessions from the TCS to another TCS	The user's VM gets the resources it needed and the QoE raises.	graphical interface to demonstrate the process	

7.1.3.5 ADD-ONS

There are some features in MobiThin that could be demonstrated when time allows to.

A possibility:

Nb	Scenario description	Expected result	Infeed	Comments
01	The user clicks on a (new) application in the AIS@US component	The application is automatically retrieved from the N/A AIS without the user noticing	AIS	
02	The application is started and the user can see the output	Sufficient QoE	client device	

7.2 MULTIMEDIA THIN CLIENT BASED ON SIP

7.2.1 Objectives

The purpose of this proof of concept is to demonstrate a thin client system using the SIP signaling protocol. The demonstration will consist in establishing a thin client session between a client and a server, and then of establishing multiple traffic channels for different types of media traffic like, video or audio. This demonstration will be based on the IMS architecture that will be defined during the phase 2 of MobiThin.

The demonstration expects to use a SIP infrastructure service platform.

The use of a real IMS service platform such as the Open-IMS platform (http://www.openimscore.org/), along with an IMS client will be investigated. Depending on the result of investigation, the demonstration could be done on a real IMS environment.

This demonstration will be a main step toward the realization of IMS based thin client services on an operated network such as a 3GPP network.

Relation to scenarios: The POC will allow demonstrating part of the D2.1 scenario where video with audio capability were identified as key requirements (Scenario 1, 2 and 3).

During the system review phase performed during the phase 1, the following points where identified:

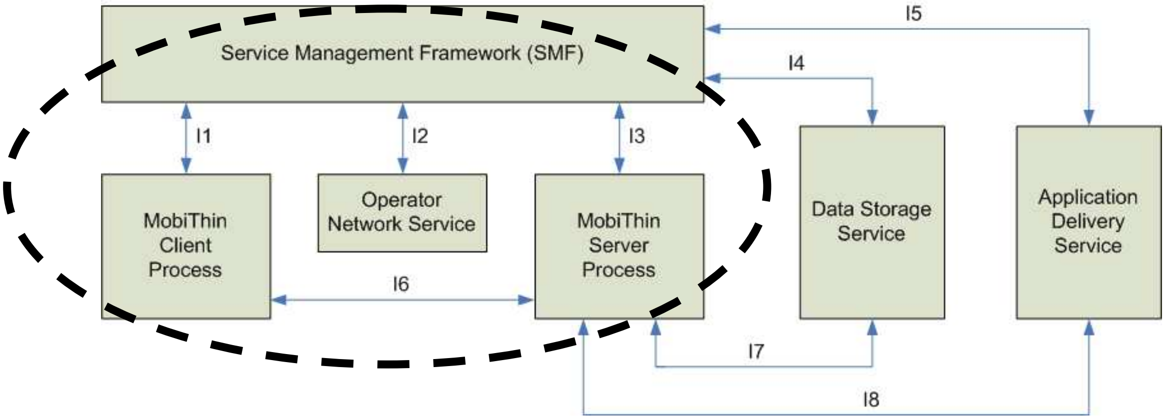
- New use cases including audio flow would be needed.
- The use of standard protocol and interfaces is expected for deploying the MobiThin services.

This proof of concept will fulfill those requirements. SIP is one of the key protocols for Multimedia services, and is also the foundation protocol of the IMS. The support of IMS over 3GPP network will allow using the QOS capabilities of the underlying network, and in particular of the radio-access network, to improve the user's Quality of Experience of thin client.

7.2.2 Description of the test environment

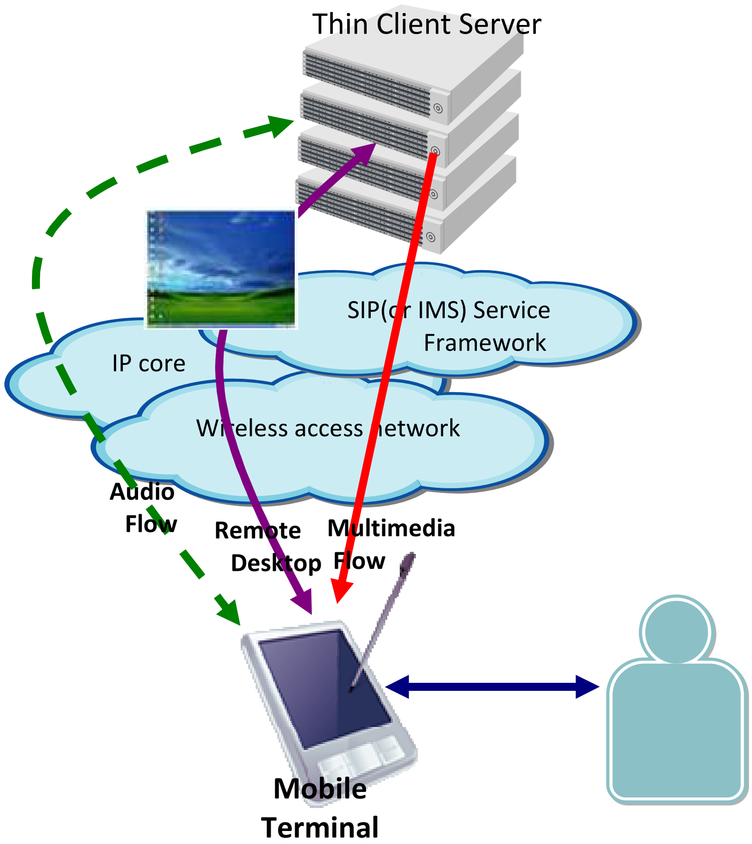
7.2.2.1 LOGICAL VIEW

The next figure identifies the architecture blocks relative to the demonstration.



The main components involved in this demonstration are the Mobithin Client and Server, the SMF and possibly the operator network services

7.2.2.2 PHYSICAL VIEW



- Mobile Terminal:
    - The mobile terminal software will be composed of an application, a SIP stack and a thin client viewer.
    - The mobile terminal will have a WiFi network interface

- Network:
    - The network will have a SIP service framework composed of SIP proxies and servers.
    - A WiFi access point will be available

- The Thin Client server runs in a PC
    - The thin client software will be composed of a thin client server application, a SIP stack and a thin client protocol stack.

7.2.3 Experiments to be performed

Nb	Scenario description	Expected result	Infeed	Comments
1	Establishment of a thin client session	Negotiation of the thin client session parameters. SIP session established. Thin Client protocol is started between the client and the server. User of the mobile device is able to control its remote PC	SIP based Thin Client application, and SIP stack ported in on the mobile terminal SIP based thin client server and SIP stack ported in the thin client server Test platform : SIP or IMS Based service platform	
2	Scenario 2 + Establishment of an audio channel for supporting the downlink and then the uplink audio between the terminal and the server	SIP signaling is used to negotiate the real time RTP traffic flow that will transport the audio. RTP flow is exchanged between the client and the server	Same as 1 with addition of the support of the audio channel on the terminal and on the server	There is a risk to have hardware or software limitation of the server and more likely on the terminal platform for multimedia services.
3	Scenario 3 + Establishment of video channel for supporting the downlink and then the uplink audio between the terminal and the server	SIP signaling is used to negotiate the streaming RTP traffic flow that will transport the video. Video is transported to the client and displayed in the client. Low bandwidth is used compared to traditional thin client	Same as 1 with addition of the support of the video channel on the terminal and on the server	There is a risk to have hardware or software limitation of the server and more likely on the terminal platform for multimedia services.

7.3 Effects of Constraints on Image Transmission

This demonstration consists of two parts. In the first part, the intelligence and efficiency of the MobiThin image transmission scheme will be shown under controlled network circumstances. In the second part, connection will be made via UMTS over the Internet, to demonstrate that the MobiThin protocol works well in real life outside the laboratory too.

7.3.1 Objectives

This demonstration is devoted to the optimization of image transmission under the joint constraints of the Client, Server and network. In this respect, we shall extend the components and functionalities already ensured in wired PC environments so as to meet the requirements of the mobile and thin clients. Three studies will be carried out:

- A compression algorithm comparison (BiFS vs. LASeR vs. VNC);
- An evaluation of the wireless network impact on MobiThin solution for image transmission (bandwidth, package loss);
- An identification of the thin terminal requirements (CPU, RAM, OS, display).

Relation to scenarios:

- this first part of the demo is transversal with regard to all uses cases, i.e. it can be transparently involved in all of them;
- it is likely to impact more in the use cases where heterogeneous content is to be transmitted "Getting in touch with Paris" and "Keep scoring at school"

The second part of this demonstration has the objective:

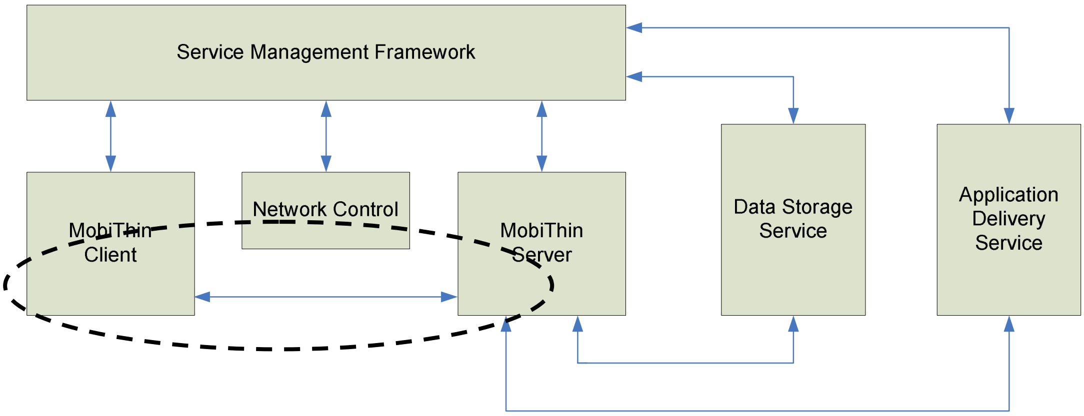
- to show that MobiThin has practical value, in that it works comfortably on a reasonably sized mobile device, over a 3G network which is highly bandwidth constrained.

Relation to scenarios: This second part of the demo relates to all scenarios described in D2.1, where the wireless communication can happen over a 3G network.

7.3.2 Description of the test environment

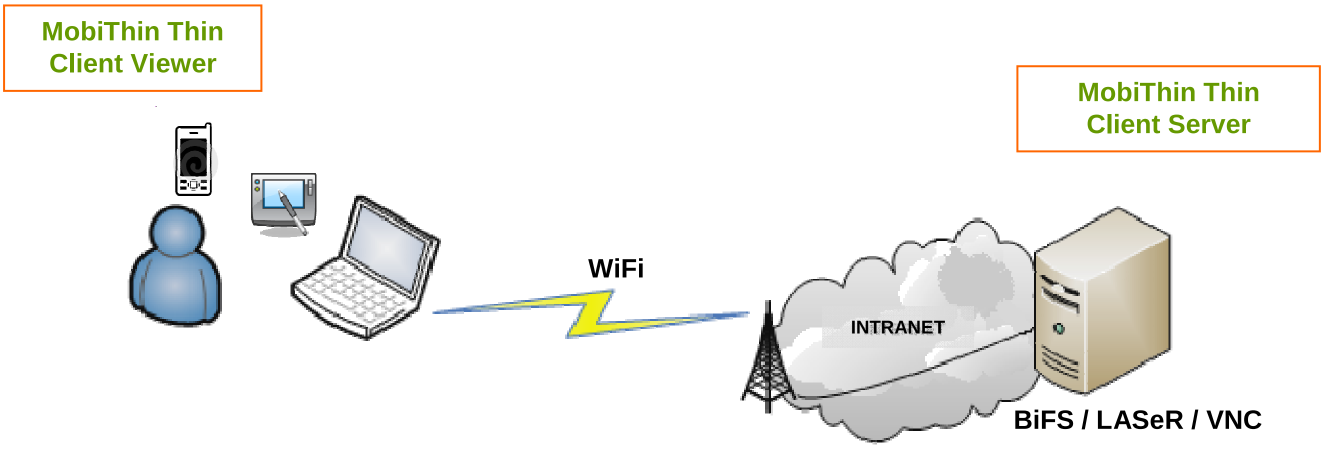
7.3.2.1 LOGICAL VIEW

The next figure identifies the architecture blocks relative to the demonstration.



The main architectural components involved in this demonstration are the MobiThin Client and Server, here considered with respect to their mutual interaction with the link optimization.

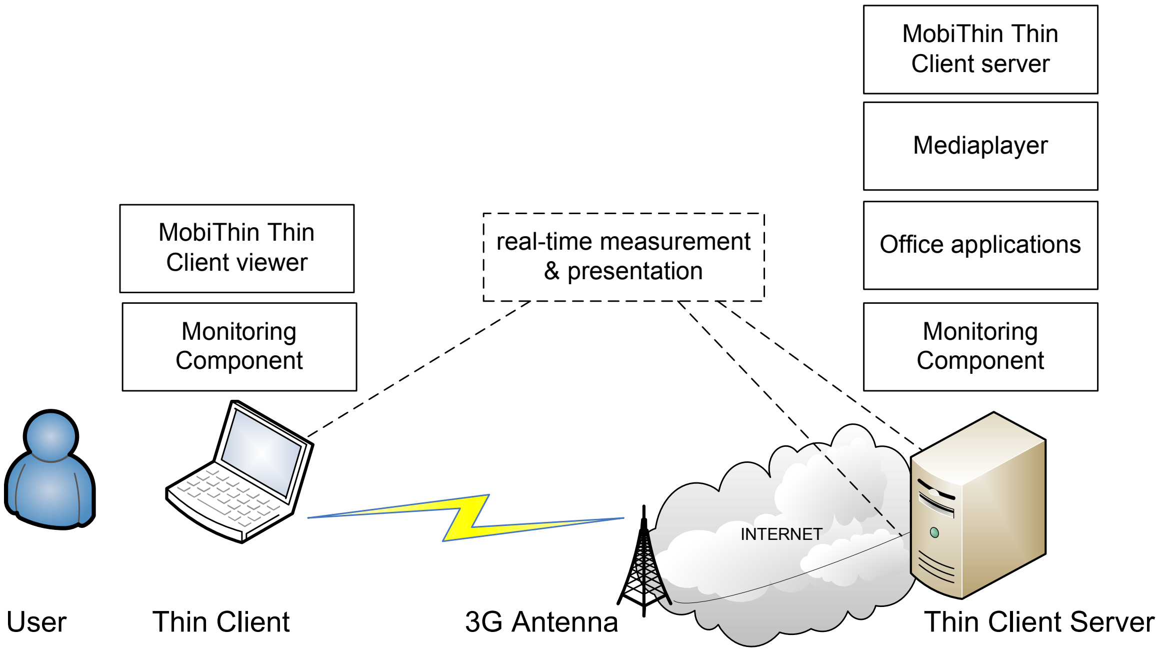
7.3.2.2 PHYSICAL VIEW



This demo will interpret information measured/monitored from the following MobiThin components:

- *The MobiThin Client Server*: the investigation will be focused on the compression (format conversion and transmission) for the content to be displayed (text/image/graphics/video);
- *The MobiThin Client Viewer*: the terminal type and the player requirements will be assessed;
- *The Network*: the network traffic (both uplink and downlink) will be monitored.

For the second part of this demonstration, the physical set-up requires some more work, since we will use an UMTS network that connects to a MobiThin server over the Internet.



- A 3G subscription is available through which the Thin Client is connected to the internet.
- A thin client server is publicly available over the internet.
- The Thin Client Server has office applications and media applications available, as well as components to monitor the server.
- The Thin Client has the MobiThin Client Viewer installed, as well as monitoring components.
- The outgoing link of the Thin Client Server is monitored.

7.3.3 Experiments to be performed

For this demonstration, we will reconsider the corresponding experiments described in D5.1, and we will adapt them to cope with light terminals and wireless networks:

	Scenario description	Expected result	Infeed	Comments
01	Comparative study of the BiFS / LASeR / VNC compression on different type of content	To chose for each type of content the most compact representation	Components developed within the scope of WP3.2 (X11 to BiFS/LASeR converter, etc…) - monitoring software tools	The type of terminal to be used will be decided later;
02	Evaluation of the wireless network impact on the image transmission	To asses the client experience allowed by a wireless network	Same as above	Same as above
03	Identification of the thin terminal requirements	To identify the class of terminal on which the MobiThin Viewer can run	Same as above	Same as above

For the second part of the demonstration, we will execute the same applications as in the “We have a problem” and “Protocol Adaptivity” demonstrations described in D5.1, as we want to show that the adaptive protocol is able to cope with a highly constraining network.

Nb	Scenario description	Expected result	Infeed	Comments
01	MobiThin Thin Client Protocol and	Sufficient QoE	Components	Somehow present a figure with

	office application		developed within the scope of WP3	the used bandwidth, if at all possible
02	MobiThin Thin Client Protocol and multimedia application	Sufficient QoE	Same as above	

7.4 Link Optimization NS2 Emulation

In this section we will describe the Link Optimization Emulation demonstrations. We divide it in 2 parts, one concerning NS2 emulation and another one concerning Hardware emulation. For the first one, we will extend the experiments done in phase I to show the gains of an integral cross-layer approach. For the second one, we will use a framework called Cross-Layer Simulation Framework (XMSF). The main difference with the NS2 simulator is that the XMSF is more accurate at hardware level because it uses very precise models of the physical components of the IMEC Software Defined Radio (SDR) platform, which makes it very close to a real hardware implementation. In contrast, NS2 works on packet level that allows a higher degree of flexibility in the functionality and it works with "live" traffic from upper layers. Following is a short description of both simulators.

7.4.1 Objectives

The objectives of this demonstration are:

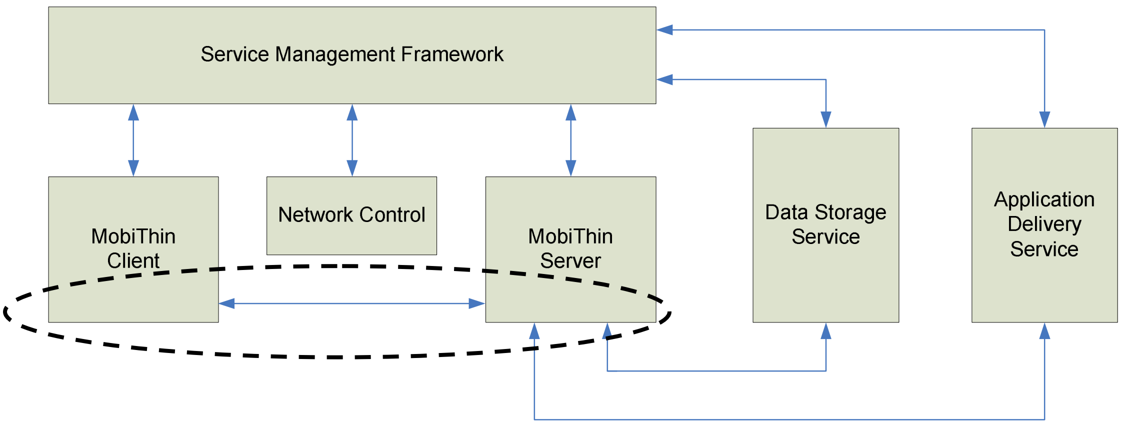
- To characterize a thin client system over a WLAN, 802.11e and LTE in terms of
  - Energy consumption
  - MAC-level QoS metrics
- To demonstrate an integral cross-layer approach (upper and lower layers) and show the gains in terms of energy consumption and QoS with respect to a state-of-the-art radio link control.

Relation to scenarios: This proof-of-concept demonstrates the benefits of the wireless protocols optimization. Those optimizations will benefit to all the scenarios defined in D2.1

7.4.2 Description of the test environment

7.4.2.1 Logical view

The next figure identifies the architecture blocks relative to the demonstration.

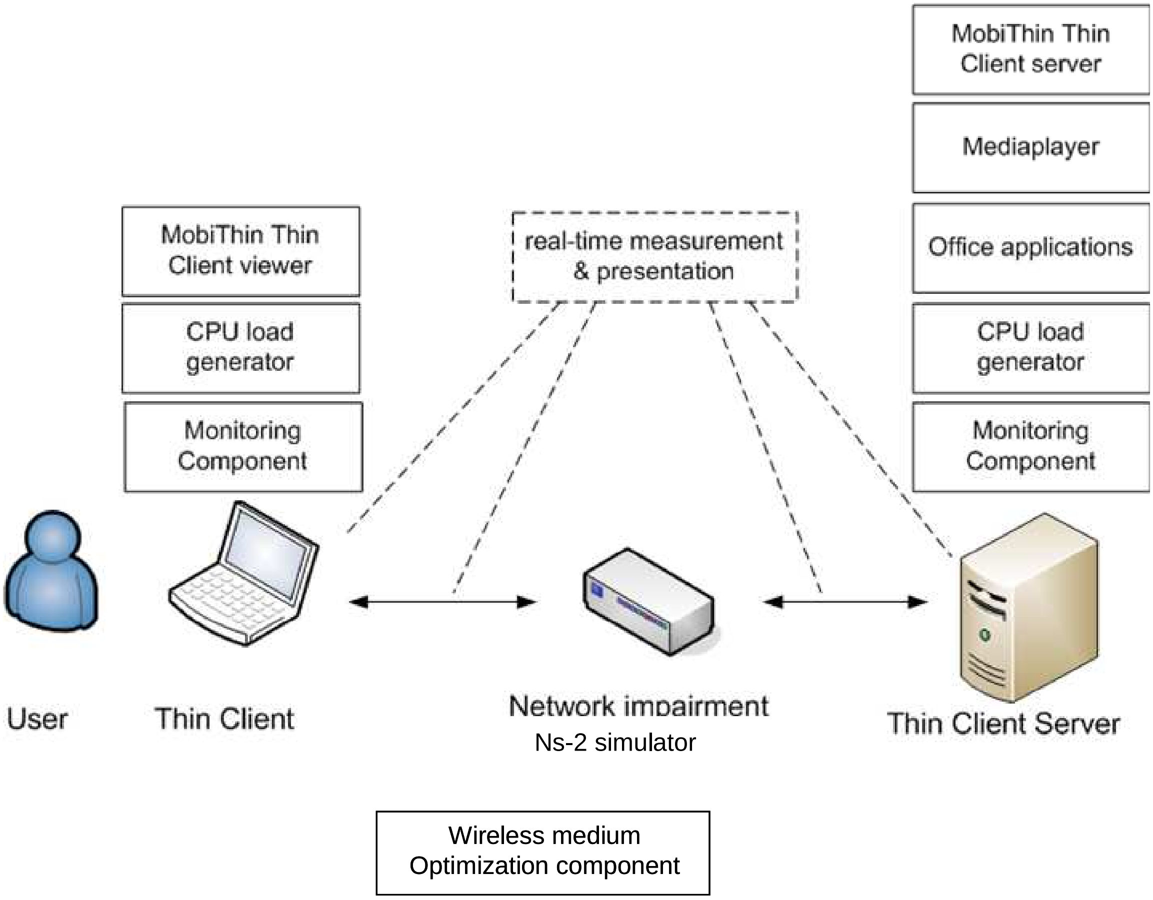


The main components involved in this demonstration are the Mobithin Client and Server and the wireless network simulated between them.

7.4.2.2 Physical view

The demonstration will be compounded as it was described in D5.1:

- 3 desktop PCs
- The thin client protocol developed as part of WP3.1 with updates from an overall cross-layer integration.
- The emulated wireless network in NS-2 including radio-level energy modeling, the wireless medium optimization solution developed in the context of WP3.1, and further enhancements from an overall cross-layer integration.



7.4.3 Experiments to be performed

For this demonstration, we will reconsider the corresponding experiments described in D5.1, but with the corresponding updates from the simulator (overall cross-layer integration).

Nb	Scenario description	Expected result	Infeed	Comments
01	Simulate different thin client types of traffic for different radio link condition and different configuration of the thin client adaptive protocol for WLAN, and 802.11e	Evaluate energy consumption at the thin client device, evaluate MAC-level QoS metrics, Evaluate subjective perception of QoE	- Components developed within the scope of WP3.1, WP3.2 - IMEC simulator - IBBT client/Server software	-
02	Same as 01 with wireless medium optimization including an overall cross-layer integration (upper and lower layers)	Evaluate energy consumption at the thin client device, evaluate MAC-level QoS metrics, Evaluate subjective perception of QoE	- Components developed within the scope of WP3.1 - IMEC simulator - IBBT client/Server software	Agreement on the interfacing between lower and upper layers

7.5 Link Optimization Hardware Emulation (XMSF)

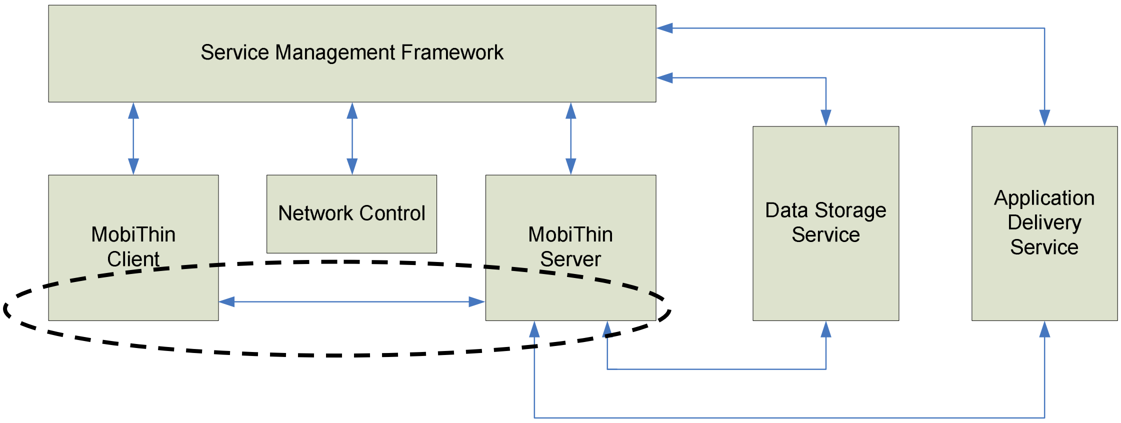
7.5.1 Objectives

As a parallel track to the Link Optimization NS2 Emulation, we plan to show the gains achieved by a cross-layer approach under a more hardware-oriented simulator (only lower layers). This low-level framework is called Cross Layer Simulation Framework (XMSF). It can show the cross-layer energy gains of the IMEC software defined radio (SDR) platform. The main difference with the NS2 simulator is that the XMSF is more accurate at hardware level (IMEC SDR platform specific), while NS2 works on packet level.

7.5.2 Description of the test environment

7.5.2.1 LOGICAL VIEW

The next figure identifies the architecture blocks relative to the demonstration.



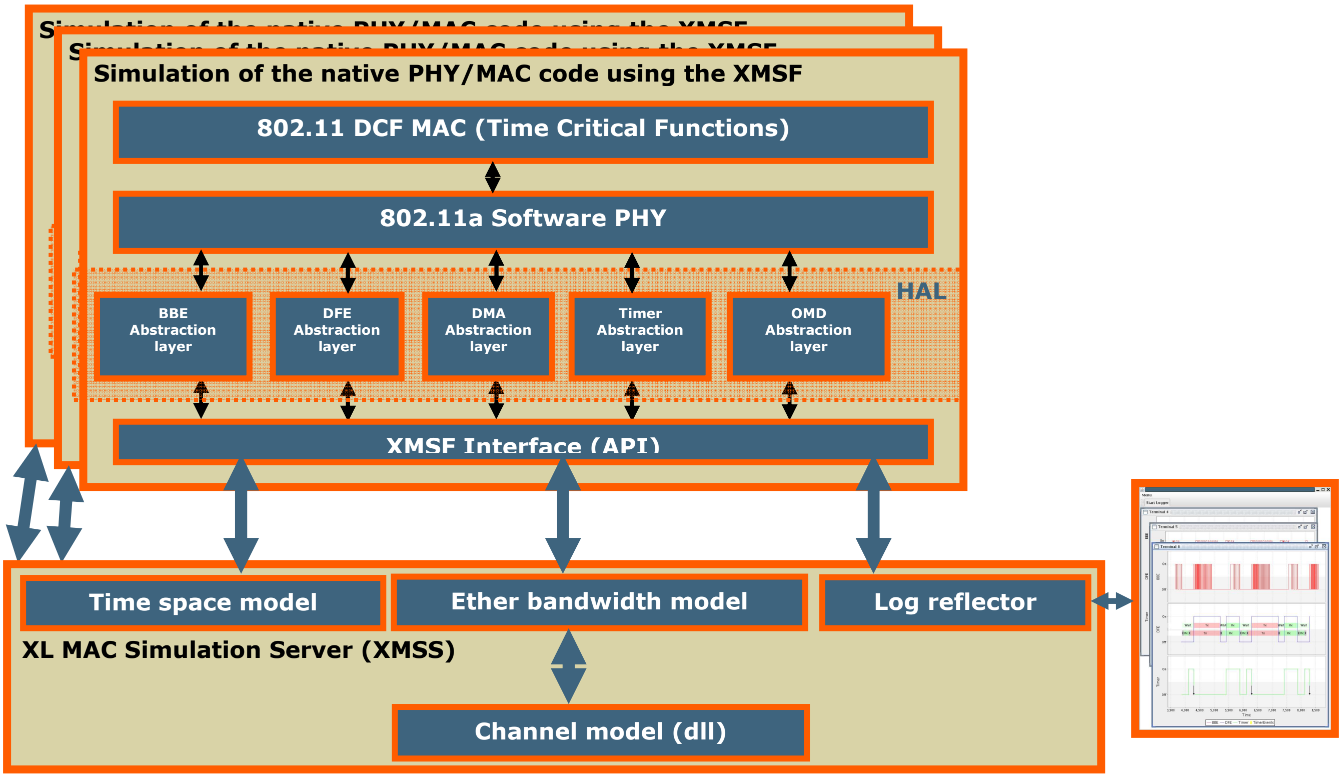
The main components involved in this demonstration are the Mobithin Client and Server and the wireless network layer simulated between them.

7.5.2.2 PHYSICAL VIEW

The XMSF is a set of tools that allows simulating the control software for the IMEC SDR. The control software is the software that runs on the control processor (in this case, an ARM9 embedded processor) and is written in ANSI C. It implements the data path that forms the 802.11a physical layer and some time critical functions of the 802.11 MAC layer. Besides that, also the Cross Layer functionality has been mapped on this ARM processor. While, NS2 simulator permits more flexibility at the functionality level, the physical component models of the XMSF offer a more close-to-hardware view of the platform.

When using the XMSF, the ANSI C embedded control software is compiled and ran on a host system. All the hardware accelerators are modeled using simulation. This means that no actual data processing is performed when simulating control software on the XMSF but that very detailed timing information can be obtained and simulated.

Using the XMSF, several simulated terminals can be connected together through a channel model. This channel model can corrupt packets. This way, the PHY and MAC functionality can be validated at network level using a realistic scenario.



The hardware communication, modeling, and simulation are done through the hardware abstraction layer (HAL) that communicates with all the platform components following an event-based simulation.
Using the XMSF logging capabilities several aspects can be visualized and steered through the Log reflector:

- The channel state between all the simulated terminals, the packet error rate (PER)
- The energy consumption of the major components of the IMEC SDR: the front end, the inner modem, the outer modem.
- The state of all these major components: on/off, receiving/transmitting, modulation, coding rate.
- The state of the PHY layer control software: Rx/Tx, carrier sensing, transient modes.
- The state of the MAC layer control software: Network Allocation Vector (NAV), SIFS timings, back off, acknowledgements, header information.
- The state of the Cross layer functionality: Pareto point selection (modulation, code rate, power amplifier settings), channel state.

7.5.3 Experiments to be performed

For this demonstration, we will reconsider part of the experiments done in phase I for the Link Optimization NS2 Emulation (7.4).

Nb	Scenario description	Expected result	Infeed	Comments
01	Setup the platform and verification of the basic functionalities	A reliable environment setup	- IMEC XMSF simulator	-
02	Simulate different thin client types of traffic for different radio link condition and different configuration of the thin client adaptive protocol for WLAN	Evaluate energy consumption at the thin client device	- Components developed within the scope of WP3.1, WP3.2 - IMEC XMSF simulator	-
03	Same as 02 with wireless (MAC/PHY) medium optimization	Evaluate energy consumption at the thin client device	- Components developed within the scope of WP3.1, WP3.2 - IMEC XMSF	-

			simulator	

7.6 REMOTE DEVICES / PERIPHERALS

7.6.1 Objectives

This demonstration aims to show the ability to remotely access and use devices connected to terminals, in thin client server environment. From the point of view of data exchange, two scenarios will be demonstrated:

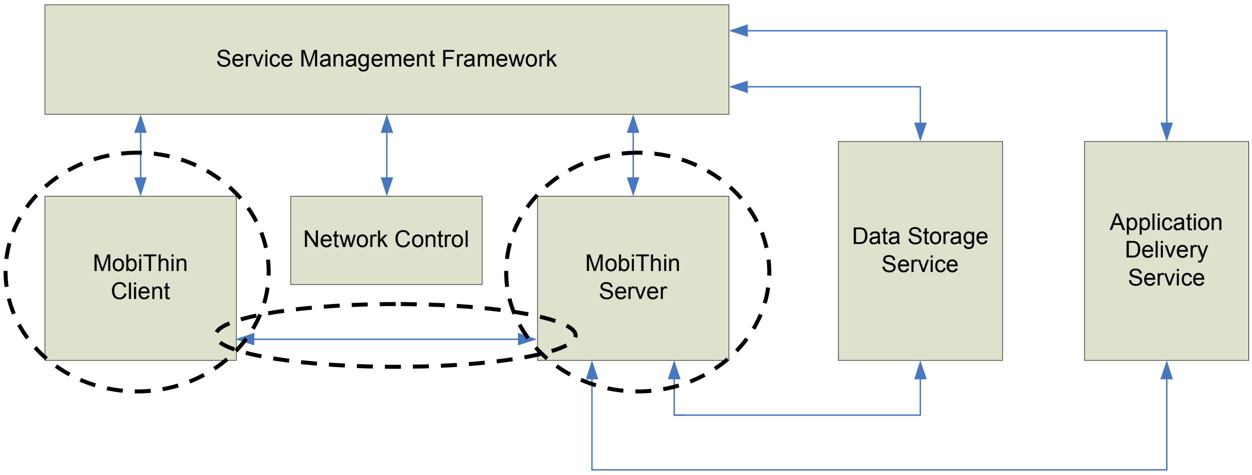
- Data exchange between a thin client server and a remote device connected to a terminal. The goal of this scenario is to prove that any remote device exported to a thin client server becomes usable by any authorized user connected to the thin client server.
- Data exchange between a remote device connected to a terminal and another remote device connected to another terminal. In this scenario data exchange between the two end points will be done through the thin client server. The goal of this scenario is to demonstrate that the exchange of data is not limited to "only between terminals and the thin client server" but is rather possible between terminals, with the thin client server as a "redirector" and as a "controller".

Relation to scenarios: The POC will allow demonstrating parts of scenario number 1 "Bringing the hospital to the patient's home" and parts of scenario number 2 "Getting in touch with Paris" described in deliverable D2.1.

7.6.2 Description of the test environment

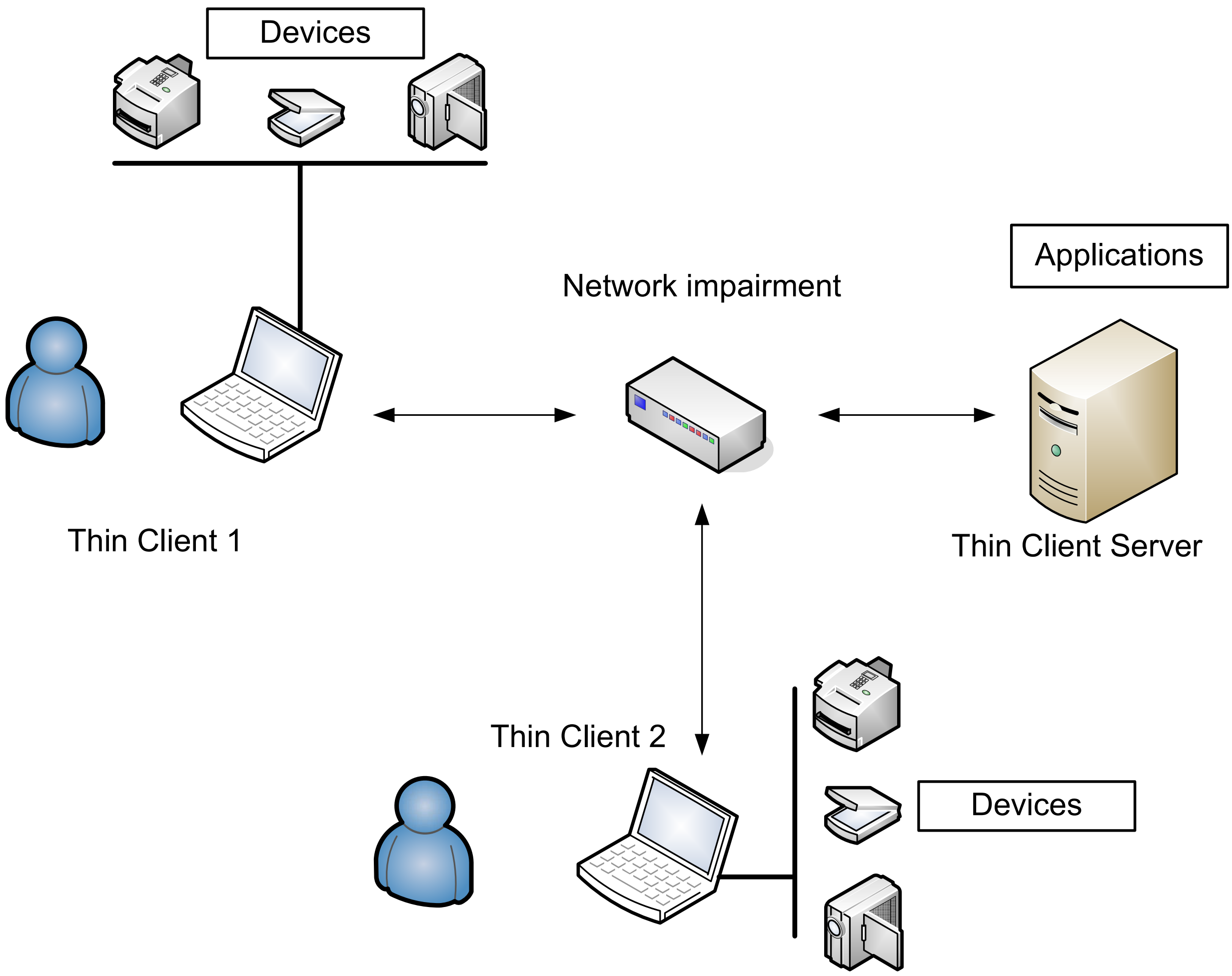
7.6.2.1 LOGICAL VIEW

The next figure identifies the architecture blocks relative to the demonstration.



The main components involved in this demonstration are the Mobithin Client and Server.

7.6.2.2 PHYSICAL VIEW



- Thin clients are connected to the server with Ethernet cables
- Thin client 1 has its own physical usb devices
- Thin client 2 has its own physical usb devices
- The MobiThin thin client server has required usb device drivers and applications for managing remote peripherals (eg: konqueror, Gimp, mplayer)

7.6.3 Experiments to be performed

Nb	Scenario description	Expected result	Infeed	Comments
01	User on client 1 connects and shares a physical usb device on thin client 1. The physical device might be a usb stick, a printer, a scanner, a web camera, etc…	The device should be seen as a shared device.	Virtual device drivers and their associated tools, on the thin client.	Device drivers loading and device sharing scenario
02	User on client 1 connects to the thin client server then imports the shared device on thin client 1 into the thin client server.	The shared device on thin client 1 should be visible to appropriate applications running on the thin client server.	Virtual device drivers and their associated tools on the thin client server.	e.g. usb stick shared on client 1 might be visible within “Konqueror” application as a usb storage device.
03	User on client 2 connects to the thin client server and runs an application that needs to connect to the device shared on thin client 1.	The application run by user 2 should be able to connect to the shared device on thin client 1.	Virtual device drivers, their associated tools and end-user applications.	e.g. usb stick shared on client 1 might be visible within “Konqueror” application as a usb storage device launched by user 2.

04	User on client 2 starts downloading data from the device shared on thin client 1 to the thin client server.	Data Download success.	Same as in scenario # 03	e.g. open a document on a usb stick shared on client 1.
05	User on client 2 starts uploading data from the thin client server to the device shared on thin client 1.	Data upload success.	Same as in scenario # 03	e.g. copy a file from the thin client server to a usb stick shared on client 1.
06	User on client 1 and user on client 2 connect and share a usb stick on their respective client.	The devices should be seen as a shared device.	Same as in scenario # 01.	The goal is to exchange data between devices connected to terminals, through the thin client server.
07	User on client 1 and user on client 2 import their respective shared usb stick devices into the thin client server.	The shared device on each thin client should be visible to appropriate applications running on the thin client server.	Same as in scenario # 02	e.g. Both usb sticks should be visible within "Konqueror" application as usb storage devices.
08	User on client 1 copies a file from his usb stick to the usb stick connected to thin client 2.	The file copy should succeed.	Same as in scenario # 03.	
09	User on client 1 copies a file from the usb stick connected to thin client 2 to his usb stick.	The file copy should succeed.	Same as in scenario # 03.	A good scenario would consist of copying the same file as in scenario 8 and compare with the original one.

- End of document -