

D5.4 – Phase I Experimental validation – Lab Trials Executive Summary

Deliverable Type *	: PU
Nature of Deliverable **	: O (Demo)
Version***	: Released
Created	: 20 November 2009
Contributing Workpackages	: WP5
Editor	: Bart Dhoedt (IBBT)
Contributors/Author(s)	: IBBT(L. Deboosere, F. Azmat Ali, B. Vankeirsbilck, P. Simoens, B. Dhoedt), Prologue (A. Taguengayte), NTUK (B. Lecroart, F. Fok), IMEC (R. Torrea Duran), IT (B.Joveski, M. Mitrea, F. Prêteux) DTAG (F-J. Westphal, M. Kind)
File Name	: [MobiThin_D5.4_WP5_30June09_IBBT_V1.0]

Abstract:

This deliverable describes lab trials carried out to validate the architecture and selected components of phase 1 of the MobiThin project. These trials include the optimization of the image transmission, based on MPEG BiFS/Laser content description, the evaluation of a thin client viewer deployed on a mobile phone, and demonstration of protocol adaptivity.

This deliverable comes in parallel with D5.3, describing the validation work carried out based on emulations.

“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”

The **MOBITHIN Project Consortium** groups the following Organizations:

Interdisciplinary Institute for BroadBand Technology vzw	IBBT vzw	BE
Deutsche Telekom AG	DTAG	DE
Prologue Software	Prologue	FR
Interuniversitair Micro-Electronica Centrum vzw	IMEC vzw	BE
NEC Technologies (UK) Ltd	NTUK	UK
Groupe des Ecoles des Télécommunications	GET	FR
JCP-Consult SAS	JCP	FR

Contents

1.	Executive summary
2.	Introduction
2.1	Project context
2.2	Relation with D5.1 and D5.3
3.	Scope of the deliverable
4.	Experimental reports on Lab Trial
4.1	“We have a problem”
4.1.1	Software/hardware architecture
4.1.2	Lab trial set-up (hardware description – characteristics)
4.1.3	Results and interpretation
4.1.4	Conclusions – requirements met
4.2	MPEG based image transmission
4.2.1	Software/hardware architecture
4.2.2	Lab trial set-up (hardware description – characteristics)
4.2.3	Results and interpretation
4.2.4	Conclusions – Requirements met
4.3	Mobile Device Integration of a Thin Client
4.3.1	Software/hardware architecture
4.3.2	Ocean software architecture
4.3.3	Lab trial set-up
4.3.4	Description of experiments performed and the results
4.3.5	Conclusions – Requirements met
4.4	Protocol adaptivity – Hybrid Streaming
4.4.1	Software architecture
4.4.2	Lab trial set-up
4.4.3	Description of experiments performed
4.4.4	Results and interpretation
4.4.5	Conclusions – Requirements met
4.5	Protocol Adaptivity - A Scheduling Approach
4.5.1	Software architecture
4.5.2	A scheduling-based screen update transmission pattern
4.5.3	Lab trial set-up
4.5.4	Results and interpretation
4.5.5	Conclusions – Requirements met
5.	Overall conclusion
6.	References
Appendices	
Appendix 1 Detailed comparison between existing thin client protocols	
A1.1	Machine Specifications
A1.2	Reference Machine Computation Table
A1.3	Bandwidth Measurement Data
A1.4	CPU Load Measurement Data
A1.4.1	Raw Data
A1.4.2	Corrected Data
A1.5	Memory Footprint Measurement Data
Appendix 2 Power consumption of the Asus Eee PC	

1. EXECUTIVE SUMMARY

This report, together with its accompanying deliverable 5.3, brings together the validation results obtained in the first project phase of MobiThin. The results addressed are obtained through lab trials, i.e. validation work the “real” hardware is used in building the different Proof-of-Concept demonstrations. The results presented related to the characterization of existing thin client protocols (analyzing their shortcomings), to the MPEG based image transmission and user interaction, to porting the thin client viewer to a mobile platform and to assessing two different adaptivity mechanisms.

The characterization of existing thin client protocols reveals important shortcomings, especially when multimedia applications are used (e.g. browsing combined with viewing multimedia content). Issues both relating to client CPU load (upto 41%), bandwidth (up to 80 Mbps for full screen MPEG4 viewing) as well as interactivity are reported.

An MPEG BiFS¹ based image transmission subsystem has been realized, compatible with a MPEG client viewer. Important savings in bandwidth usage (typically a factor of 5) have been observed, whilst keeping the quality level and offering interactivity. Efforts are underway to standardize this approach in the MPEG community.

The project also succeeded in porting a thin client viewer to a mobile phone, hence showing that thin client viewers are a workable option for remotely accessing application through a resolution limited device. Conclusions regarding usability were drawn (pointing devices, resolution, selecting user sessions and establishing a session with the servers).

Two different approaches were presented and evaluated in this report:

1. The hybrid streaming protocol, switching dynamically between H.264 streaming and a pixel-based format, depending on the content shown. The protocol automatically chooses the optimal transmission mode, based on the content characteristics. The adaptivity succeeds in realizing the optimal operation of the individual modes, resulting in lower client side CPU consumption and bandwidth usage.
2. The scheduled update approach allows the client and server to agree on an optimal display update frequency, adapting this frequency to the current network conditions. It was shown that the bandwidth used by the protocol can be very well controlled by judiciously tuning this update frequency.

The lab trials reported here have been constructed in accordance with the architectural documents and PoC design documents. The results obtained allow to conclude that the approach taken on each sub-domain addressed is justified, and hence realizing the PoC's is an important step in building the MobiThin E2E system.

- End of Document -

¹ BiFS stands for Binary Format for Scenes, standardized as the part 11 of the MPEG-4 standard (ISO/IEC JTC1/SC29/WG11 14496-11); it allows heterogeneous multimedia content (graphics, video, audion, text, 3D, ...) to be represent into a unique scene.