

**EVEREST****EVEREST IST-2002-001858****D12*****Implemented Testbed: Subsystems & Modules*****Contractual Date of Delivery to the CEC: 31.12.2004****Actual Date of Delivery to the CEC: 14.01.2005****Editor: Antoni Gelonch and Ramón Ferrus (UPC)****Author(s): see list****Participant(s): UPC, KCL, PTIN, TID, TEL****Workpackage: WP4****Est. person months: 40****Security: PU****Nature: Report****Version: 01****Total number of pages: 159****Abstract:**

This deliverable provides the complete description of the Everest Testbed architecture and the details of the subsystems and modules implemented. The aspects that the EVEREST testbed will try to address are all them related with the provision of the E2E QoS. Basically, two main blocs has been considered, one related with the QoS emulation introduced in the Radio Access part and the other one related with QoS management performed in the Core Network. In the EVEREST Tesbed up to three different access technologies (UTRA, GERAN and WLAN) has been considered and all the elements related with the Common Radio Resource Management has been included. The Core Network side has been driven by an implementation based in Policy Based Management of a Diffserv domain where the Bandwith Broker (BB) module is the main element.

In addition some extension related with the architecture of the QoS management, like Wireless QoS Broker, has been incorporated coming from discussion performed in WP3.

**Keyword list: Emulation, Testbed, UMTS, GERAN, WLAN, Diffserv Domain, BB, Policy Based Management.**

## **DISCLAIMER**

The work associated with this report has been carried out in accordance with the highest technical standards and the EVEREST partners have endeavoured to achieve the degree of accuracy and reliability appropriate to the work in question. However since the partners have no control over the use to which the information contained within the report is to be put by any other party, any other such party shall be deemed to satisfied itself as to the suitability and reliability of the information in relation to any particular use, purpose or application.

Under no circumstances will any of the partners, their servants, employees or agents accept any liability whatsoever arising out of any error or inaccuracy contained in this report (or any further consolidation, summary, publication or dissemination of the information contained within this report) and/or the connected work and disclaim all liability for any loss, damage, expenses, claims or infringement of third party rights.

**DOCUMENT HISTORY**

Date	Version	Status	Comments
8.11.04	1.0	Int	ToC 1 <sup>st</sup> version
23.11.04	2.0	Int	First Partners contribution
10.01.05	10.0	Int	New version
12.01.05	11.0	Int	TID final revision
14.01.05	1.0	Apr	Approved by PCC

## **Authors List**

APARICIO PEÑA, Maria Teresa (TID)  
CABRAL PINTO, Filipe (PTIN)  
CASADEVALL PALACIO, Fernando (UPC)  
FERRÚS FERRER, Ramón (UPC)  
GELONCH BOSCH, Antoni (UPC)  
GONZÁLEZ SANCHÍS, Hector (TID)  
NAFISI, Nima (KCL)  
REBELO, João (PTIN)  
REVES BALLESTER, Xavier (UPC)  
VEGA NOVELLA, Avelina (TID)

## EXECUTIVE SUMMARY

This deliverable aims at providing a detailed description of the implemented Everest testbed, starting from its architecture and going down to the details of the implementation and/or configuration of the different modules and tools. In addition it has been included in this document the progress in the definition of the procedures to be shown and managed by the EVEREST testbed at the end of the project. This must be understood as a powerful tool to progress in the complete EVEREST testbed definition.

Due to the nature of the EVEREST testbed and its high requirements in terms of flexibility and management features some of the currently developed modules still present some inconsistency that will be properly corrected during the integration phase and along the progress in the procedures definition.

Therefore, this document describes the actual status of the EVEREST implemented testbed. It initiates its description starting with the EVEREST testbed architecture, as already introduced in D06, and including the description of the Software Environment as the framework that gives support to the complete testbed. Related to the emulation of the radio access part, the document describes how the multiple heterogeneous access networks emulators considered in EVEREST (UTRAN, GERAN, WLAN) are integrated in a common framework. This integration imposes a set of design constraints to be followed in the implementation of each emulator. In particular, switching functions needed for the integration of the emulators, and that constitute the basis of what in the document is referred to as the RAN skeleton, are deployed so that that from real applications can be directed to the proper emulator. Then, the description of each RAN emulator is addressed, showing the implementation details of the correspondent modules and also the emulation approach followed in each one of the different RATs. We should not forget the description of the Wireless QoS Broker element as a QoS manager of the Radio Access part. Later on it is addressed the implementation issues related with the CN sided specially in reference to the Diffserv implementation approach followed, the use of the Bandwidth Broker as a CN QoS manager and the Mobility aspects. Next point has been the description of the solution adopted for the Network and UE QoS managers that has been based in the solution adopted for the BB. At this point has been included the current description of the Procedures to be managed in the complete testbed under the proof of concept approach that can provide a better understanding of the complete EVEREST testbed framework. And finally the installation and configuration details of the Applications included in the EVEREST testbed have been addressed as well as the Testbed Management Capabilities, as a tool to configure properly the testbed and obtain results from it, and the Remote Testbed Capabilities, as an extension of the previous one tool.

**In summary, the document defines the EVEREST complete testbed framework just before starting its integration process, its architecture, the approach followed to solve the different issues and the progress in the definition of the procedures that must be available in the final EVEREST testbed. Nevertheless, some refinements in the current implementation of the QoS management entities and CRRM functions may be required according to expected results coming from the progress in WP3 activities.**

## Table of Contents

<b>EXECUTIVE SUMMARY</b> .....	<b>V</b>
<b>1 INTRODUCTION</b> .....	<b>1</b>
1.1 PURPOSE OF THE DOCUMENT.....	1
<b>2 TESTBED ARCHITECTURE</b> .....	<b>1</b>
2.1 FUNCTIONAL TESTBED ARCHITECTURE .....	1
2.1.1 <i>Radio Access Domain</i> .....	1
2.1.2 <i>Core Network Domain</i> .....	3
2.2 SOFTWARE ENVIRONMENT .....	4
2.2.1 <i>Operating System</i> .....	4
2.2.2 <i>Communications Manager</i> .....	4
<b>3 RADIO ACCESS NETWORKS EMULATION</b> .....	<b>4</b>
3.1 RADIO ACCESS NETWORKS EMULATION REQUIREMENTS .....	4
3.2 HARDWARE INFRASTRUCTURE .....	7
3.2.1 <i>Description of Testbed Machines</i> .....	7
3.2.2 <i>Communications Equipments and Network Connectivity</i> .....	7
3.3 SKELETON FOR MULTIPLE RAN EMULATION SUPPORT (UPC) .....	8
3.3.1 <i>Logical architecture</i> .....	8
3.3.2 <i>Functional architecture</i> .....	9
3.3.2.1 Packet routing .....	10
3.3.2.2 Packet processing .....	11
3.3.2.3 RAN module software structure .....	13
3.4 UTRAN EMULATOR .....	14
3.4.1 <i>Emulation Model</i> .....	14
3.4.1.1 Radio Access Bearer Emulation.....	14
3.4.1.2 Radio Resource Management Emulation.....	17
3.4.1.3 Support for the Propagation Scenario and mobility models.....	19
3.4.2 <i>Implementation Details</i> .....	20
3.5 GERAN EMULATOR.....	23
3.5.1 <i>Emulation Model</i> .....	23
3.5.1.1 User Data processing.....	23
3.5.1.2 Statistical data management .....	24
3.5.1.3 RRM algorithm .....	24
3.5.1.4 BLER vs C/I.....	26
3.6 WLAN EMULATOR .....	27
3.6.1 <i>Emulation Model</i> .....	27
3.6.1.1 Main Functionalities .....	28
3.6.1.2 User Data Processing .....	28
3.6.1.3 System Level Emulation.....	29
3.6.1.3.1 Error Behaviour .....	29
3.6.1.3.2 Multiple Stations Performance.....	29
3.6.2 <i>Implementation Details</i> .....	31
3.7 WIRELESS QOS BROKER .....	31
<b>4 CORE NETWORK</b> .....	<b>34</b>
4.1 HARDWARE INFRASTRUCTURE .....	34
4.2 DIFFSERV SUPPORT .....	34
4.2.1 <i>Kernel configurations</i> .....	35
4.2.2 <i>DiffServ edge router configuration</i> .....	35
4.2.3 <i>DiffServ core router configuration</i> .....	37
4.3 BANDWIDTH BROKER .....	38
4.3.1 <i>Implementation details</i> .....	38
4.3.2 <i>Implementation extension</i> .....	43
4.4 MOBILITY MANAGEMENT SOFTWARE.....	44
<b>5 NETWORK AND UE QOS MANAGERS</b> .....	<b>46</b>
5.1 QOSCLIENT.....	46
5.2 MPDP .....	47

<b>6</b>	<b>SUPPORTED PROCEDURES</b>	<b>48</b>
6.1	CONNECTION ESTABLISHMENT WITH QoS NEGOTIATION	48
6.1.1	Objective	48
6.1.2	Involved entities, interfaces and protocols	48
6.1.3	Procedure description	49
6.1.3.1	Reference scenario	49
6.1.3.2	Message Chart	50
6.1.3.3	Expected Results	50
6.2	E2E QoS RE-NEGOTIATION	50
6.2.1	Objective	50
6.2.2	Involved entities, interfaces and protocols	50
6.2.3	Procedure description	51
6.2.3.1	Reference scenario	51
6.2.3.2	Message Chart	51
6.2.3.3	Expected Results	51
6.3	RAT SWITCHING	52
6.3.1	Objective	52
6.3.2	Involved entities, interfaces and protocols	52
6.3.3	Procedure description	53
6.3.3.1	Reference scenario	53
6.3.3.2	Message Chart	53
6.3.3.3	Expected Results	54
6.4	CN MOBILITY MANAGEMENT AND QoS INTERACTIONS	54
6.4.1	Objective	54
6.4.2	Involved entities, interfaces and protocols	54
6.4.3	Procedure description	54
6.4.3.1	Reference scenario	55
6.4.3.2	Message Chart	55
6.4.3.3	Expected Results	55
6.5	COMMON RADIO RESOURCE MANAGEMENT OPERATION	56
6.6	IMPACT OF CRRM AND QoS MANAGEMENT ON APPLICATIONS	56
6.6.1	Objective description	56
6.6.2	Quantitative Quality Measurements	56
6.6.2.1	Relevant QoS traffic parameters	57
6.6.2.2	Capture Tools	57
6.6.2.2.1	Oreneta Tool	57
6.6.2.2.1.1	Capturing Scheme	57
6.6.3	Subjective Quality (MOS) Measurement	58
6.6.4	Objective Evaluation of QoS MOS	59
6.6.4.1	Audio: PESQ tools	59
6.6.4.1.1	Audio Used Tools	60
6.6.4.2	Video: PSNR based tool	60
6.6.4.2.1	Video used Tools	61
6.6.5	Involve entities, interfaces and protocols	61
6.6.6	Procedure description	61
6.6.6.1	Reference scenario	62
6.6.6.2	Message Chart	62
6.6.6.3	Expected Results	62
<b>7</b>	<b>APPLICATIONS IN THE TESTBED</b>	<b>63</b>
7.1	VIDEO-TELEPHONY	63
7.1.1	VIC	63
7.1.1.1	Overview	63
7.1.1.2	The Codecs	64
7.1.1.3	Installing VIC	64
7.1.1.3.1	Preparing the system	65
7.1.1.3.1.1	Installing the PWC	65
7.1.1.3.1.2	Installing the PWCX module	65
7.1.1.3.1.3	Installing the ALSA Sound drivers	65
7.1.1.3.2	Installing VIC	66
7.1.2	RAT	66
7.1.2.1	Overview	66
7.1.2.2	The Codecs	67
7.1.2.3	Installing RAT	67

7.2	VIDEO-STREAMING .....	67
7.2.1	<i>Darwin Streaming Server</i> .....	68
7.2.1.1	Overview .....	68
7.2.1.2	Installing Darwin Streaming Server .....	68
7.2.2	<i>MPEG4IP</i> .....	68
7.2.2.1	Overview .....	68
7.2.2.2	Installing MPEG4IP .....	69
7.2.2.2.1	Setting-Up the tools .....	69
7.2.2.2.1.1	nasm-0.98.38.tar.gz .....	69
7.2.2.2.1.2	faac-1.24.tar.gz .....	69
7.2.2.2.1.3	lame-3.96-1.1.fc2.fr.i386.rpm .....	69
7.2.2.2.1.4	SDL-1.2.7-1.i386.rpm .....	69
7.2.2.2.2	Installing MPEG4IP .....	70
7.3	WEB BROWSING .....	70
7.3.1	<i>Apache HTTP Server</i> .....	70
7.3.1.1	Overview .....	70
7.3.1.2	Installing Apache HTTP Server .....	70
7.3.1.2.1	Quickconfig .....	70
7.3.1.2.2	Proconfig - Installing the Apache 1.3 HTTP server with APACI .....	70
7.3.1.2.2.1	Requirements .....	70
7.3.1.2.2.2	Configuring the source tree .....	71
7.3.1.2.2.3	Building the package .....	71
7.3.1.2.2.4	Installing the package .....	71
7.3.1.2.2.5	Testing the package .....	71
7.3.2	<i>Mozilla Web Browser</i> .....	71
7.3.2.1	Overview .....	71
7.3.2.2	Installing Mozilla Web Browser .....	72
7.4	E-MAIL .....	72
7.4.1	<i>Qmail</i> .....	72
7.4.1.1	Overview .....	72
7.4.1.2	Installing qmail .....	72
7.4.2	<i>Mozilla Mail</i> .....	73
<b>8</b>	<b>TESTBED MANAGEMENT CAPABILITIES .....</b>	<b>73</b>
8.1	SOFTWARE MODULES MANAGED BY THE AGMT .....	74
8.2	CONTROL OF THE EXECUTION FLOW .....	76
8.3	CONFIGURATION OF ALL THE MODULES .....	76
8.3.1	<i>Format of the initialisation files</i> .....	77
8.4	TRACE GENERATION AND OFF-LINE ANALYSIS .....	77
8.4.1	<i>Log file format description</i> .....	77
8.5	REAL TIME DATA VISUALISATION AND MODIFICATION .....	79
<b>9</b>	<b>REMOTE TESTBED CAPABILITIES .....</b>	<b>83</b>
9.1	USAGES .....	83
9.2	REMOTE TESTBED MANAGEMENT & MONITORING .....	84
9.2.1	<i>Evaluation of remote execution tools</i> .....	84
9.2.1.1	Telnet .....	84
9.2.1.2	SSH .....	84
9.2.1.3	VNC .....	85
9.2.2	<i>Implementation Details</i> .....	86
9.3	REMOTE EXECUTION OF APPLICATIONS .....	87
<b>10</b>	<b>CONCLUSIONS .....</b>	<b>89</b>
	<b>REFERENCES .....</b>	<b>89</b>
	<b>ACRONYMS .....</b>	<b>91</b>
	<b>ANNEX A: .....</b>	<b>93</b>
A.1	ORENETA TOOL .....	93
A.1.1	<i>Capturing Scheme</i> .....	93
A.2	INSTALLATION GUIDE .....	93
A.3	USER GUIDE .....	94
A.4	ORENETA MODIFICATIONS .....	95



<b>ANNEX B</b> .....	<b>97</b>
B.1 INTRODUCTION TO CM .....	97
<i>B.1.1 Overview</i> .....	97
B.2 CM FUNCTIONS .....	97
<i>B.2.1 Inter-process Communication</i> .....	97
<i>B.2.2 Configuration Parameters</i> .....	98
<i>B.2.3 Logs and Monitoring</i> .....	98
<i>B.2.4 Execution of the Application</i> .....	98
<i>B.2.5 Timing control</i> .....	98
B.3 CM ARCHITECTURE OVERVIEW .....	98
<i>B.3.1 CM SHELL</i> .....	99
B.3.1.1 SHELL Commands .....	100
B.3.1.2 Connection Configuration File Format.....	100
B.3.1.3 Scenario Configuration File.....	102
B.3.1.4 Launching SHELL .....	103
<i>B.3.2 CM RUNTBD</i> .....	104
B.3.2.1 CM CDAEMON .....	105
B.3.2.2 CM ICONSOLE .....	105
B.3.2.3 CM BRIDGEM .....	105
B.3.2.4 CM STATSD .....	105
B.3.2.5 CM SYNC.....	107
B.3.2.6 Launching SYNC.....	108
B.4 APPLICATION SOFTWARE MODULES STRUCTURE.....	109
<i>B.4.1 Definition of Module</i> .....	109
<i>B.4.2 Considerations on Using System Calls</i> .....	109
<i>B.4.3 Timing Procedures</i> .....	110
<i>B.4.4 Code Generation Guidelines</i> .....	110
<i>B.4.5 Modules Initialisation File Format</i> .....	112
B.5 CM API FUNCTIONS .....	113
<i>B.5.1 Communication and Control Functions</i> .....	114
B.5.1.1 Description of control functions .....	114
B.5.1.2 Description of communication functions.....	115
<i>B.5.2 Initialisation functions</i> .....	117
B.5.2.1 Description of initialisation functions .....	117
<i>B.5.3 Log file functions</i> .....	117
B.5.3.1 Description of log functions .....	117
<i>B.5.4 Statistics Functions</i> .....	118
B.5.4.1 Description of statistics functions .....	118
B.5.4.2 Use of Statistics Functions.....	120
<i>B.5.5 Other Functions</i> .....	122
B.6 COMPILATION AND EXECUTION OF CM .....	123
<i>B.6.1 Compilation</i> .....	123
B.6.1.1 Setting Time Slot Length.....	124
<i>B.6.2 Execution</i> .....	125
<i>B.6.3 Using CM with modules</i> .....	125
B.7 MODULE EXAMPLES .....	125
<i>B.7.1 Example #1</i> .....	125
B.7.1.1 Running Software.....	126
B.7.1.2 Operation and Code Overview .....	128
<i>B.7.2 Example #2</i> .....	131
B.7.2.1 Running Software.....	132
B.7.2.2 Code Overview.....	133
B.7.2.3 Operation .....	135
<i>B.7.3 Example #3</i> .....	137
B.8 ADVANCED GRAPHICAL MANAGEMENT TOOL.....	139
<i>B.8.1 Introduction</i> .....	139
<i>B.8.2 AGMT Overview</i> .....	139
B.8.2.1 The Scenario File.....	140
B.8.2.2 Flow Execution Control .....	141
<i>B.8.3 Using AGMT and CM: example #4</i> .....	142