



D41.1.2

4.1b MOBiNET Operating Procedures

Work package: 4.1
Version number: v1.3
Dissemination level: PP
Date: 29/07/2015



7th RTD Framework Programme
Directorate General for Communications Networks, Content & Technology
Cooperative Systems for energy efficient and sustainable mobility (FP7-ICT-2011-6.7)
Contract Type: Collaborative project
Grant agreement no.: 318485

Version Control

Version history			
Version	Date	Main author	Summary of changes
0.1	10-01-2014	Franc Buve, Maartje Stam	Initial version
0.2	28-01-2014	Franc Buve	Added commissioning of regions after internal review
0.3, 0.4, 0.5	31-01-2014	Franc Buve	Updated with doc templates in Annex
0.6	14-02-2014	Franc Buve	Clarified content of this version and the relationship with 4.1a. Added document templates in Annex.
0.8	19-02-2014	Franc Buve	Updated title and incorporated DLR review comments
0.9	03-04-2014	Franc Buve	Reformatted templates in Annex to improve readability and reduce space.
1.0	01-09-2014	Namik Erdogan	Updated for M22 delivery as internal report.
1.1 Draft	01-03-2015	Namik Erdogan	Updated for new lay-out and Swarco participation
1.1	02-06-2015	Namik Erdogan	Finalized document for V1.1
1.2	23-07-2015	Iñaki Etxaniz	Summary, updated components manuals
1.3	29-07-2015	Iñaki Etxaniz	Reorganization, Final corrections
	Name		Date
Prepared	Franc Buve, Namik Erdogan, Iñaki Etxaniz		23/07/2015
Reviewed	Henning Mosebach (DLR), Tobias Schlauch (DLR), Yanying Li (ERTICO)		27/07/2015
Authorised	Rasmus Lindholm		31/07/2015
Circulation			
Recipient		Date of submission	
European Commission		31/07/2015	
Project partners		31/07/2015	

Authors

I. Etxaniz (Tecnalia), N. Erdogan (CGI), F. Buve (CGI)

Table of contents

Abbreviations and definitions.....	11
Executive Summary.....	13
1 Introduction	14
1.1 Purpose and scope.....	14
1.2 Intended audience	14
1.3 Environment types.....	15
1.4 System overview	16
2 Referenced documents	19
3 Transition of applications into MOBiCENTRE	20
3.1 Design & develop phase (TEST ENV).....	20
3.2 Handover	21
3.3 Commissioning and approval (COM ENV)	22
3.4 Acceptance Review	22
3.5 Installation on production (PROD ENV).....	22
4 Transition of services into MOBiCENTRE.....	23
4.1 Design & develop phase.....	23
4.2 Test phase.....	24
4.3 Handover	24
4.4 Commissioning	24
4.5 Service Readiness Review (SRR).....	24
4.6 Installation on production.....	25
5 Transition of pilot sites or regions into MOBiCENTRE	26
5.1 Handover	26
5.2 Commissioning	27
5.3 Region Readiness Review (RRR)	27
5.4 Installation on production.....	27
6 MOBiCENTRE Operating Procedures	28
6.1 Login on the OAM.....	28
6.2 Accessing other VM's for installation	28
6.3 Login on the Components directly	28
6.4 Creating OAM User Account	29

6.5	Google Authenticator verification	29
6.6	Changes after the handover	29
6.7	Installing a component.....	29
6.8	Running/Stopping a component	30
6.9	Uninstalling a component	30
6.10	Monitoring a component	31
7	Annex: Component Operating Procedures	32
7.1	Service Directory	32
7.1.1	Component Description.....	32
7.1.1.1	Overview	32
7.1.1.2	Release Overview	32
7.1.1.3	Installation and Upgrade Notes	33
7.1.2	Installation	33
7.1.2.1	Pre-requisites	33
7.1.2.2	Installation Procedure.....	33
7.1.3	Pre-installation Tasks	33
7.1.4	Installation Procedure.....	33
7.1.5	Post-installation	33
7.1.6	Software Test Report	34
7.1.6.1	Insert Service Description	34
7.1.6.2	Lookup Service Description.....	36
7.1.6.3	Discover Service	38
7.1.6.4	Delete Service Description	40
7.1.7	Software User Manual.....	42
7.2	Communication Agent	42
7.2.1	Component Description.....	42
7.2.1.1	Overview	42
7.2.1.2	Release Overview	42
7.2.1.3	Installation and Upgrade Notes	42
7.2.2	Installation	43
7.2.2.1	Pre-requisites	43
7.2.2.2	Installation Procedure.....	43
7.2.2.3	Pre-installation Tasks	43
7.2.2.4	Installation Procedure.....	43
7.2.2.5	Post-installation	43
7.2.2.6	Uninstall or Roll-back	43
7.2.3	Software Test Report	44
7.2.3.1	Start CA.....	44

D4.1b Operating Procedures

7.2.3.2	Receive new position update from a new node.....	44
7.2.3.3	Receive a position update from a known node	45
7.2.3.4	Receive an dissemination request from a SP via the REST API	45
7.2.3.5	Detect node inside dissemination area	46
7.2.3.6	Detect vehicle outside dissemination area	46
7.2.3.7	Software User Manual	46
7.3	Dashboard	47
7.3.1	Component Description	47
7.3.1.1	Overview	47
7.3.1.2	Release Overview	47
7.3.1.3	Installation and Upgrade Notes	47
7.3.2	Installation	47
7.3.2.1	Pre-requisites	47
7.3.2.2	Pre-installation Tasks	48
7.3.2.3	Installation Procedure.....	48
7.3.2.4	Post-installation	48
7.3.2.5	Uninstall or Roll-back	48
7.3.3	Software Test Report	49
7.3.4	Software User Manual	51
7.4	Identity Manager	52
7.4.1	Component Description	52
7.4.1.1	Overview	52
7.4.1.2	Release Overview	52
7.4.1.3	Installation and Upgrade Notes	52
7.4.2	Installation	52
7.4.2.1	Pre-requisites	52
7.4.2.2	Pre-installation Tasks	53
7.4.2.3	Installation Procedure.....	53
7.4.3	Software Test Report	57
7.5	Service Directory Widgets	59
7.5.1	Component Description.....	59
7.5.1.1	Overview	59
7.5.1.2	Release Overview	59
7.5.1.3	Installation and Upgrade Notes	59
7.5.2	Installation	59
7.5.2.1	Pre-installation Tasks	59
7.5.2.2	Installation Procedure.....	60
7.5.2.3	Post-installation	60
7.5.2.4	Uninstall or Roll-back	61

D4.1b Operating Procedures

7.5.3	Software Test Report	61
7.6	Telematics Service Providers Manager	62
7.6.1	Component Description	62
7.6.1.1	Overview	62
7.6.1.2	Component Description	62
7.6.1.3	Release Overview	63
7.6.1.4	Installation and Upgrade Notes	64
7.6.2	Installation plan	64
7.6.2.1	Pre-requisites	64
7.6.2.2	DB Setup	64
7.6.2.3	Installation	67
7.6.2.4	Configuration	68
7.6.2.5	Uninstall.....	69
7.6.3	Operations User Manual	69
7.6.3.1	Files.....	69
7.6.3.2	Basic operational check	69
7.7	Data Quality Assessment	70
7.7.1	Component Description	70
7.7.1.1	Basic Information.....	70
7.7.1.2	Description	70
7.7.1.3	Release Overview	70
7.7.2	Architecture	71
7.7.2.1	Database Description	71
7.7.2.2	Interface Description	71
7.7.3	Installation Plan	71
7.7.3.1	Pre-requisites	71
7.7.3.2	Installation	72
7.7.3.3	Uninstall / Service Removal	72
7.7.3.4	Start/ Stop service	72
7.7.3.5	Service configuration.....	72
7.7.3.6	Files.....	72
7.7.3.7	Basic operational check	72
7.7.4	DQA API.....	72
7.7.5	Installing Python2.7	73
8	Annex: Document Templates	74
8.1	SRS Template	75
8.1.1	Scope	75
8.1.1.1	Identification	75

D4.1b Operating Procedures

8.1.1.2	System overview	75
8.1.1.3	Document overview	75
8.1.2	Referenced documents	75
8.1.3	Requirements	75
8.1.3.1	Required states and modes.	75
8.1.3.2	System capability requirements	75
8.1.3.3	(System capability)	76
8.1.3.4	(System capability)	76
8.1.4	System external interface requirements	76
8.1.5	Interface identification and diagrams.....	76
8.1.5.1	(Project-unique identifier of interface)	76
8.1.5.2	System internal interface requirements.....	78
8.1.5.3	System internal data requirements.....	78
8.1.5.4	Adaptation requirements	78
8.1.5.5	Safety requirements	78
8.1.5.6	Security and privacy requirements	78
8.1.5.7	System environment requirements.....	79
8.1.5.8	Computer resource requirements.....	79
8.1.5.9	Computer hardware requirements.....	79
8.1.5.10	Computer hardware resource utilization requirements	79
8.1.5.11	Computer software requirements.....	79
8.1.5.12	Computer communications requirements	79
8.1.5.13	System quality factors	79
8.1.5.14	Design and construction constraints	80
8.1.5.15	Personnel-related requirements.....	80
8.1.5.16	Training-related requirements.....	80
8.1.5.17	Logistics-related requirements	80
8.1.5.18	Other requirements	80
8.1.5.19	Packaging requirements	81
8.1.5.20	Precedence and criticality of requirements	81
8.1.6	Qualification provisions	81
8.1.7	Requirements traceability	81
8.1.8	Notes	82
8.2	DBDD Template	82
8.2.1	Scope	82
8.2.1.1	Identification	82
8.2.1.2	System overview	82
8.2.1.3	Document overview	82
8.2.2	Referenced documents	82

D4.1b Operating Procedures

8.2.3	Database-wide design decisions.....	82
8.2.4	Detailed design of the database.....	83
8.2.4.1	(Name of database design level).....	83
8.2.5	Detailed design of software units used for database access or manipulation..	84
8.2.5.1	(Project-unique identifier of a software unit, or designator for a group of software units)	84
8.2.6	Requirements traceability.....	86
8.2.7	Notes.....	86
8.3	SIP Template.....	86
8.3.1	Scope.....	86
8.3.1.1	Identification.....	86
8.3.1.2	System overview.....	86
8.3.1.3	Document overview.....	87
8.3.1.4	Relationship to other plans.....	87
8.3.2	Referenced documents.....	87
8.3.3	Installation Overview.....	87
8.3.3.1	Description.....	87
8.3.3.2	Contact point.....	87
8.3.3.3	Support materials.....	87
8.3.3.4	Training.....	87
8.3.3.5	Tasks.....	87
8.3.3.6	Personnel.....	88
8.3.3.7	Security and privacy.....	88
8.3.4	Site-specific information for software center operations staff.....	88
8.3.4.1	(Site name).....	88
8.3.4.2	Schedule.....	88
8.3.4.3	Software inventory.....	88
8.3.4.4	Facilities.....	89
8.3.4.5	Installation team.....	89
8.3.4.6	Installation procedures.....	89
8.3.4.7	Data update procedures.....	89
8.3.5	Site-specific information for software users.....	89
8.3.5.1	5.x (Site name).....	89
8.3.5.2	Schedule.....	90
8.3.5.3	Installation procedures.....	90
8.3.5.4	Data update procedures.....	90
8.3.6	Notes.....	90
8.4	STR Template.....	90
8.4.1	Scope.....	90

D4.1b Operating Procedures

8.4.1.1	Identification	90
8.4.1.2	System overview	91
8.4.1.3	Document overview	91
8.4.2	Referenced documents	91
8.4.3	Overview of test results	91
8.4.3.1	Overall assessment of the software tested.	91
8.4.3.2	Impact of test environment.	91
8.4.3.3	Recommended improvements.	91
8.4.4	Detailed test results.....	91
8.4.4.1	(Project-unique identifier of a test).	92
8.4.4.2	Summary of test results.....	92
8.4.4.3	Problems encountered.	92
8.4.4.4	Deviations from test cases/procedures.	92
8.4.5	Test log.....	93
8.4.6	Notes.....	93
8.5	SUM Template.....	93
8.5.1	Scope	93
8.5.1.1	Identification	93
8.5.1.2	System overview	93
8.5.1.3	Document overview	93
8.5.2	Referenced documents	93
8.5.3	Software summary	93
8.5.3.1	Software application.	93
8.5.3.2	Software inventory.....	94
8.5.3.3	Software environment.	94
8.5.3.4	Software organization and overview of operation.	94
8.5.3.5	Contingencies and alternate states and modes of operation.	95
8.5.3.6	Security and privacy.	95
8.5.3.7	Assistance and problem reporting.....	95
8.5.4	Access to the software.	95
8.5.4.1	First-time user of the software.....	95
8.5.4.2	Equipment familiarization.	95
8.5.4.3	Access control.	96
8.5.4.4	Installation and setup.	96
8.5.4.5	Initiating a session.....	96
8.5.4.6	Stopping and suspending work.	96
8.5.5	Processing reference guide.....	96
8.5.5.1	Capabilities.....	96
8.5.5.2	Conventions.	97

D4.1b Operating Procedures

8.5.5.3 Processing procedures.....	97
8.5.5.4 (Aspect of software use).....	97
8.5.5.5 (Aspect of software use).....	97
8.5.5.6 Related processing.....	97
8.5.5.7 Data backup.....	97
8.5.5.8 Recovery from errors, malfunctions, and emergencies.....	97
8.5.5.9 Messages.....	97
8.5.5.10 Quick-reference guide.....	98
8.5.6 Notes.....	98
8.6 SVD Template.....	98
8.6.1 Scope.....	98
8.6.1.1 Identification.....	98
8.6.1.2 System overview.....	98
8.6.1.3 Document overview.....	98
8.6.2 Referenced documents.....	98
8.6.3 Version Description.....	98
8.6.3.1 Inventory of materials released.....	98
8.6.3.2 Inventory of software contents.....	99
8.6.3.3 Changes installed.....	99
8.6.3.4 Adaptation data.....	99
8.6.3.5 Related documents.....	99
8.6.3.6 Installation instructions.....	99
8.6.3.7 Possible problems and known errors.....	99
8.6.4 Notes.....	99

Abbreviations and definitions

Abbreviation	Definition
C2C	Car to Car
CentOS	Community Enterprise Operating System
COM ENV	Commissioning Environment
CRM	Customer Relationship Management
DEV ENV	Development Environment
EWSP	European Wide Service Platform
IEEE	Institute of Electrical and Electronics Engineers
IOPS	Input/output Operations per Second
ITS	Intelligent Transport System
ITIL	Information Technology Infrastructure Library
IVE	In-Vehicle Equipment
J-STD-016	IEEE Standard for Information Technology--Software Life Cycle Processes--Software Development--
M2M	Machine to Machine
MDT	Mean Down Time
MTBF	Mean Time Between Failure
MTBM	Mean Time Between Maintenance
MTTM	Mean Time To Maintenance
MTTR	Mean Time To Repair
NLS	Native Language Support
OAM	Operation, Administration and Maintenance
OPS	Operations
PCI	Payment Card Industry
PCI DSS	Payment Card Industry Data Security Standards
PROD ENV	Production Environment
PSAP	Public Safety Answering Point
RHEL	Red Hat Enterprise Linux
RPM	Red Hat Package Manager

D4.1b Operating Procedures

SPx	Sub Project
SSH	Secure Shell Host
TEST ENV	Testing Environment
WP	Work Package
Terminology	Definition
Testing	The testing of services or applications to verify functional and non-functional behaviour in general.
Acceptance	The verification that a service or application meets the requirements to run in MOBiCENTRE. This does not verify functional behaviour.
Commissioning	The installation and configuration of a service, application or region prior to deployment into production, i.e. prior to going live.
Transitioning	The combination of Acceptance followed by Commissioning and Going live.

Executive Summary

This document is to provide operating procedures for the MOBiCENTRE, with respect to testing and commissioning: the transitioning of new applications, services and pilot sites (or regions) into MOBiNET

The document is intended to service providers that want to connect their service to MOBiCENTRE, to MOBiCENTRE suppliers that want to deliver a new MOBiCENTRE component, to MOBiCENTRE operators that need to accept and commission new services and applications, and to MOBiNET regions or pilot sites that want to be connected to MOBiCENTRE.

Chapter 1 contains an introduction, the document background and the system overview.

The transition of applications, services and pilot sites or regions into the MOBiCENTRE ecosystem make up the central part of the document, in chapters 3, 4 and 5 respectively. Here the phases of the workflows and their detailed steps are explained.

The operating procedures for the MOBiCENTRE platform are provided in chapter 6. This summarizes the usual or common tasks of the platform, and constitutes a basic handbook for the future operators of it. Also, the chapter includes some changes in the work procedures introduced after the handover of the work package 4 *Operation*, to facilitate the developers' task.

Specific component procedures are included as an annex- in chapter 7. This documentation is included as it is at the time of writing, but cannot be considered definitive. It is prone to change as the development progresses, and with the delivering of further releases or new components.

Chapter 8 summarizes and provides templates for the documentation described in chapters 3, 4 and 5. This documentation, to be provided through the work done in WP7 and WP3, is considered essential for the commissioning procedure to be performed correctly.

1 Introduction

1.1 Purpose and scope

This document 4.1b is the deliverable D41.1.2 of WP4.1 and describes the operating procedures for MOBiNET with respect to testing and commissioning: the transitioning of new applications, services and pilot sites (or regions) into MOBiNET.

At M15, an internal report was produced, being the stage of the project too early to describe the operations procedures of the MOBICENTRE platform and applications, which were still under development.

In M22 the operations procedures of the platform and applications were added to the document, and completed in further versions. The scope was then extended with a user manual for the operator, which includes accessing, security and monitoring aspects. The manual will also refer to the user manuals for the MOBICENTRE components, where instructions for installation, configuration, monitoring and security are described.

Instructions on how a future owner can operate the platform and its several environments are indeed included in the document. This can be considered as an initial operations handbook. Nevertheless, a more complete handbook is expected to be delivered towards the end of the project as part of the deliverable *D4.5 MOBiNET Platform Environment Guidelines*.

The fact that the handover of the platform operations -among CGI and Tecnalía- were done around M33 of the project has not changed, in essence, the transitioning procedures of MOBICENTRE. However, at the initiative of developers, some changes have been introduced in the way they access and operate to the Testing and Commissioning environments.

Note that requirements to and from service providers, component suppliers and pilot sites for the MOBiNET software environment are documented in deliverable 4.1a MOBiNET Software Environment (D41.1.1).

1.2 Intended audience

This document targets the following audiences:

1. Service providers (SP2¹, or WP7 in the rest of the document) that want to connect their service to MOBICENTRE;
2. MOBICENTRE suppliers (SP3², or WP3 in the rest of the document) that want to deliver a new (version of a) MOBICENTRE application;
3. MOBICENTRE operators (SP4³, or WP4 in the rest of the document) that need to accept and commission new services, applications or regions;

¹ SP2 in the former structure corresponds to *WP7-Service Development and Exploitation* in the new DoW

² SP3 in the former structure corresponds to *WP3-Architecture and Development* in the new DoW

³ SP4 in the former structure corresponds to *WP4-Operation* in the new DoW

4. MOBiNET regions or pilot sites (SP5⁴, or WP5 in the rest of the document) that want their region to be connected to MOBiCENTRE.

1.3 Environment types

WP4 provides four types of environments. A development and a test environment to support WP7 and WP3 development and an acceptance, commissioning and production environment:

DEV ENV	Development environment for WP7 and WP3. This is provided as an image that developers install locally. It is not hosted by WP4.
TEST ENV	Environment for testing of WP7 and WP3 deliveries. A MOBiCENTRE configuration with test stubs, that is configurable to the needs of WP7 and/or WP3. TEST ENV is controlled by WP4, but can be used by WP7 and WP3 to perform preliminary tests prior to installation on the COM ENV.
COM ENV	Environment to commission new or updated services, applications or regions prior to release into production. The COM ENV is a copy of the PROD ENV, with the exception that it is not connected to the live services, but to test instances provided by the service providers.
PROD ENV	The live production environment.

Error! Reference source not found. below depicts the transition of WP3 applications and WP7 services into production by promoting from TEST ENV up to PROD ENV. MOBiCENTRE provides a production-like test environment, TEST ENV, for testing by WP7 and WP3. TEST ENV is under control of WP4, but the test activities in that environment are the responsibility of WP7 or WP3 respectively.

New services from WP7 do not run within MOBiCENTRE, but in order to demonstrate that they interface correctly with MOBiCENTRE, they must show a test report before they are accepted into the COM ENV. WP4 will configure new WP7 services in the COM ENV for use at specific sites. When a new service is ready to go live, the new configuration settings to enable that service will be promoted to PROD ENV.

⁴ SP5 in the former structure corresponds to *WP5-Validation* in the new DoW

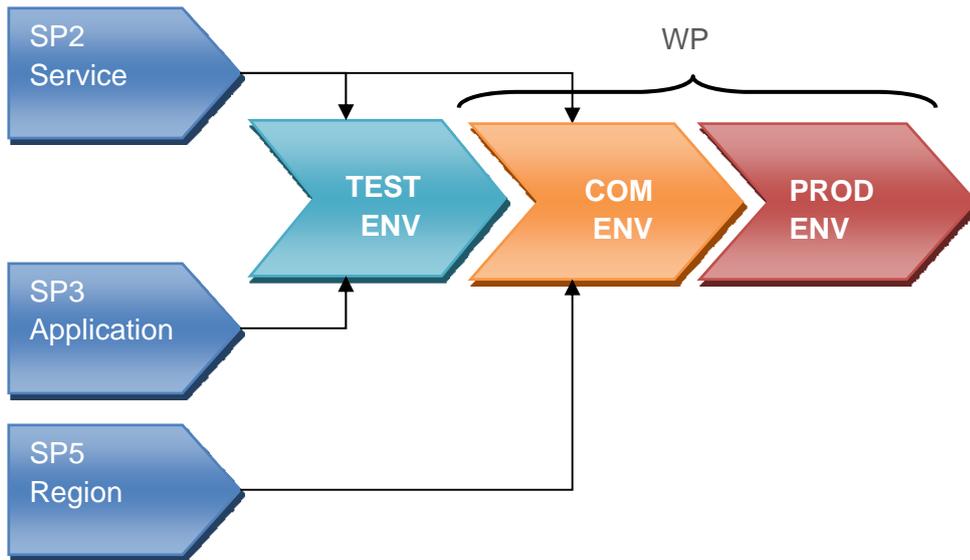


Figure 1 Transition process to production

1.4 System overview

WP4 provides and operates the environment that hosts the MOBiCENTRE applications.

As the Billing component is going to be hosted on Swarco-Mizar premises, the table below presents the responsibilities of the partners in the technical realisation and commissioning of the MOBiCENTRE.

TECHNICAL REALISATION

Task	CGI/Tecnia	Swarco	Component suppliers
VPN connection Swarco-CGI/Tecnia	Lan2lan VPN port (*, **)	Lan2lan VPN port (*, **)	
Windows environment		Create environment (TEST, COM, PROD)	
Billing component		Create component	
Interfacing between components			(TEST, COM, PROD) Provide interface description for components

(*) Security standards need to be agreed upon.

(**) One lan-to-lan VPN connection per environment. Making 3 lan-to-lan VPN tunnels between CGI and Swarco.

COMMISSIONING

D4.1b Operating Procedures

Steps	Environment	Component supplier	CGI/Tecnalia	Swarco
Deployment	Test	X		X
Testing & bug solving	Test	X		X
Test reporting	-	X		X
Deployment & validation	Commissioning		For the rest of components (*)	For Billing component
Non-functional integrated testing	Commissioning		For all components (**)	
Reporting			Overall report including chapter provided by Swarco	Only deployment reporting chapter for Billing component

(*) Only feasible if correct documentation is provided by component supplier

(**) When Use Cases provided, performed by WP5

WP5 arranges that existing roadside and central systems are adapted to connect to WP7 services and provides end user devices that can run a MOBiAGENT. Figure 2 shows the context of MOBiCENTRE within MOBiNET. Service providers connect their service to MOBiCENTRE and provide a specific app in the MOBiNET App Store to access their service. The service is not hosted within MOBiCENTRE. WP7 is responsible for development of services.

WP3 develops the applications that are run in MOBiCENTRE (such as service directory, identity management, billing) and the MOBiAGENT that is running on the (mobile) device.

WP4 provides and operates the environment that hosts the MOBiCENTRE applications.

WP5 arranges that existing roadside and central systems are adapted to connect to WP7 services and provides end user devices that can run a MOBiAGENT.

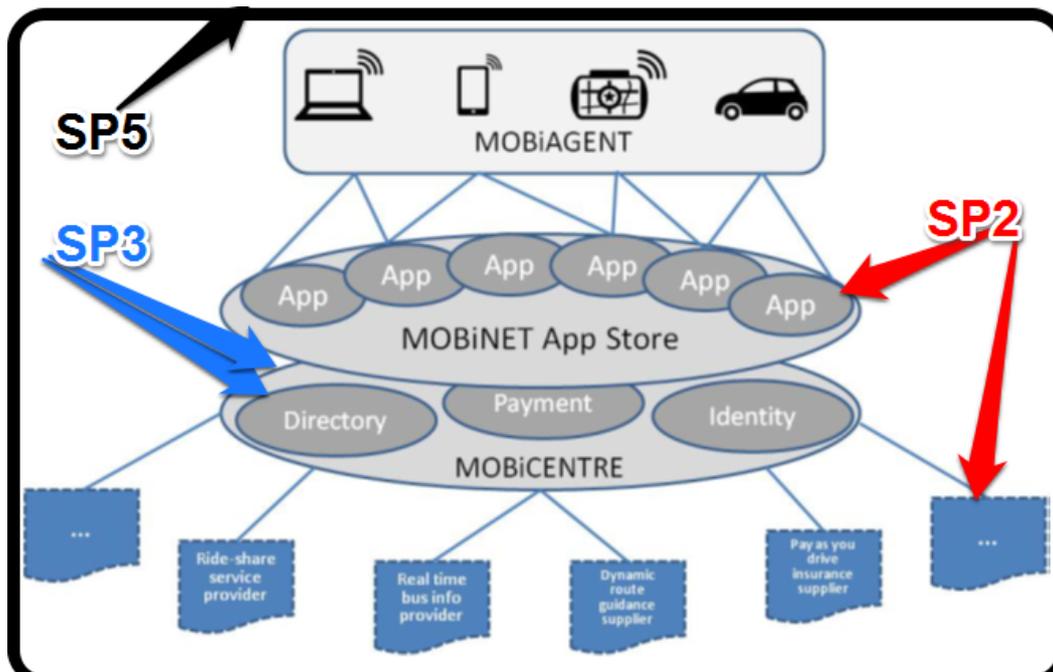


Figure 2 Context of MOBICENTRE

2 Referenced documents

Ref	Title
D31.2	Initial Concept and Architecture
D41.1.1	4.1a MOBiCENTRE Software Requirements
D32.4.1	Service directory concept
IR35.11	User Application Environment development
D37.161	MOBiNET Dashboard development
D33.7.1	Development of MOBiNET Identity Manager

3 Transition of applications into MOBiCENTRE

The figure below shows the process as foreseen for the parallel workflows of WP3 and WP4. The upper blue arrow shows the flow for WP3, and the lower blue arrow the flow for WP4.

The main responsibility or action holder is shown in the yellow blocks. However this process is a joint action, so the grey blocks show the parallel activities in the ‘other’ work package.

In between the arrows, there are feedback loops and decision moments (GO/ NO GO) for further action.

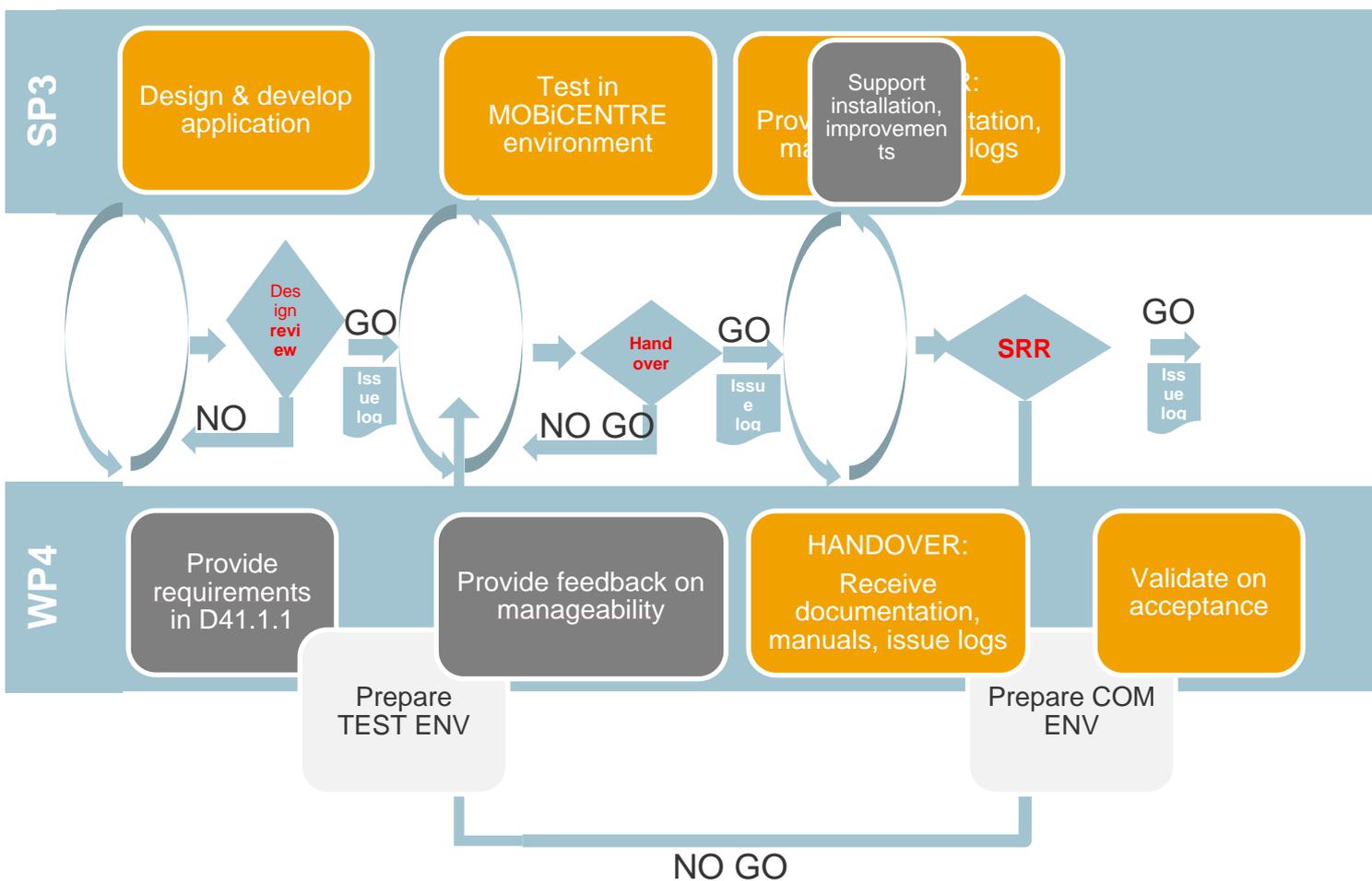


Figure 3 Transition of applications

3.1 Design & develop phase (TEST ENV)

In the design phase, WP3 is designing the services and its components. WP4 is already involved and preparing the acceptance criteria for the MOBiCENTRE. By going back and forth, the service centre can adapt to the needs of the applications, or the applications can adapt to the requirements of MOBiCENTRE in this early stage. For each application a design review should be organized to be sure

the requirements are met, on paper. WP4 shall review the design of every application that will be installed. Prior to the design phase WP4 will have prepared the D(evelopment) and T(est) environments according to the specifications in D41.1.1 (supporting task in light grey).

During the software development phase, WP4 is involved by providing an environment for running newly built components and testing them. This directly opens the opportunity for WP4 to give advice and feedback on the manageability aspects of the service, as reflected in the second dark grey block in the WP4 arrow.

3.2 Handover

Next step is the handover of the software to WP4. This is the official moment to deliver a package to WP4 which can be tested on the Commissioning environment. The WP4 leader checks all documentation, including installation manuals and issue logs, and the software. As soon as he gives a “GO”, he is the one who needs to come to action and install and test the service in the MOBiCENTRE COM ENV environment

WP4 requires a set of documentation to be able to commission and operate a component. The documentation templates are based on the J-STD-016 standard.

The following set of documents is required for any application running in MOBiCENTRE. An SSS is optional. It may be needed if the system consists of multiple software components. A DBDD is only needed if the component uses a database.

Doc	Document type	Content
SSS	System/Subsystem Specification	System and interface requirements, quality factors. Needed if the application consists of multiple software components.
SRS	Software Requirements Specification	Software modules and interface requirements, quality factors.
DBDD	Database Design Description	Database design and schemas. Only needed if the component uses a database.
SIP	Software Installation Plan	Plan and instructions for installation and configuration of the software.
STR	Software Test Report	Results of prior module and system tests in TEST ENV.
SUM	Software User Manual	User manual to operate the system/component.
SVD	Software Version Description	Release notes with list of files, solved issues and known issues.

3.3 Commissioning and approval (COM ENV)

After Handover the WP4 leader is the action holder to install the application in the COM ENV environment, which he prepared in the meantime. WP4 will perform the commissioning together with a developer from WP3. WP4 will not focus in functional testing, but will verify the application against the requirements stated in [D41.1a]. This focuses on non-functional aspects, like:

- (Automatic) Installation
- Configuration
- Logging
- Manageability
- Scalability

3.4 Acceptance Review

When commissioning on COM ENV has finished, the WP4 leader plans a meeting with all involved stakeholders to go through the findings. The participants in this meeting are:

- WP4.1 leader (chair and commissioning manager),
- WP4.2 leader (MOBiCENTRE operations manager),
- WP3 leader (MOBiCENTRE supplier),
- WP3 application's developer (MOBiCENTRE supplier technical specialist).

The outcome of the review will be a Go/No-go decision for installation into PROD ENV. The WP4.2 operations manager is authorized to reject the application and send it back to the building phase for improvement if it does not meet MOBiCENTRE requirements.

3.5 Installation on production (PROD ENV)

After a Go in the COM ENV, control is handed over to WP4.2 to activate the new application in PROD ENV and go live.

4 Transition of services into MOBiCENTRE

The figure below shows the process as foreseen for the parallel workflows of WP3 and WP4. The upper blue arrow shows the flow for WP3, and the lower blue arrow the flow for WP4.

The main responsibility or action holder is shown in the yellow blocks. However this process is a joint action, so the grey blocks show the parallel activities in the 'other' workpackage.

In between the arrows, there are feedback loops and decision moments (GO/ NO GO) for further action.

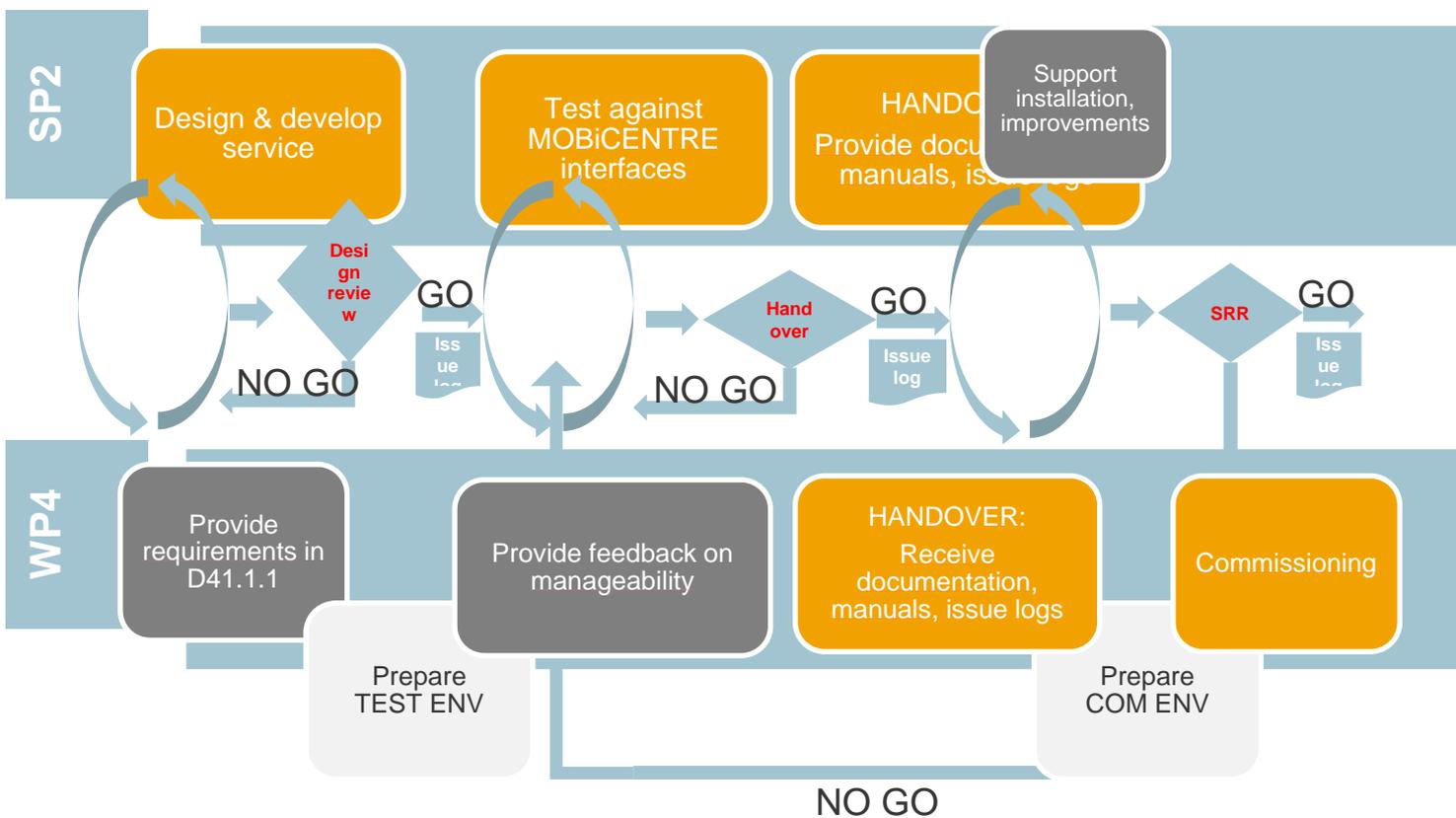


Figure 4 Transition of services

4.1 Design & develop phase

In the design phase, WP7 will prepare a design that will document which interfaces and applications of MOBiCENTRE this service will use and what the estimated quantitative usage of those interfaces will be. WP7 shall organize a design review with WP4 to be sure that the requirements to connect to MOBiCENTRE are met on paper.

4.2 Test phase

After to the design phase WP4 will prepare the COM environments for use by WP7 to test the interfaces of the service with MOBiCENTRE.

4.3 Handover

Next step is the handover of the service to WP4. This is the official moment to deliver a package to WP4 which can be commissioned in the commissioning environment. The WP4 leader checks all documentation, including installation manuals, acceptance report and issue logs. As soon as he gives a GO, he is the one who needs to come to action and install and test the service in the MOBiCENTRE COM ENV environment

WP4 requires a set of documentation to be able to commission and operate a component. The documentation templates are based on the J-STD-016 standard.

The following set of documents is required for any service using MOBiCENTRE.

Doc	Document type	Content
SIP	Service Installation Plan	Plan and instructions for configuration of the service.
STR	Service Test Report	Results of prior testing of the service in TEST ENV.
SUM	Service User Manual	User manual to operate the system/component. (Only required when some or all of the service is hosted within MOBiCENTRE).
SVD	Service Version Description	Release notes with list of files, solved issues and known issues.

4.4 Commissioning

After Handover the WP4 leader is the action holder to install the application in the COM ENV environment, which he has prepared in the meantime. WP4 will configure the service in MOBiCENTRE according to instructions in the Software Installation Plan and Software User Manual. WP4 will verify that the service is ready to move to the production environment.

4.5 Service Readiness Review (SRR)

When commissioning has finished, the WP4 leader plans a meeting with all involved stakeholders to go through the findings. The participants in this meeting are:

- WP4.1 leader (chair),
- WP4.2 leader (operations manager),
- WP7 service developer,

D4.1b Operating Procedures

- WP7 leader,
- the MOBiNET business owner.

The outcome of the SRR will be a Go/No-go decision for installation into PROD ENV. The MOBiNET Business Owner and WP4.2 Operations Manager need to accept the readiness of the service and are authorized to reject it and send it back to the building phase for improvement.

4.6 Installation on production

When the SRR decides on a Go for installation into PROD ENV, control is handed over to WP4.2 to activate the new service in PROD ENV and go live.

5 Transition of pilot sites or regions into MOBICENTRE

The figure below shows the process as foreseen for the parallel workflows of WP5 and WP4. The upper blue arrow shows the flow for WP5, and the lower blue arrow the flow for WP4.

The main responsibility or action holder is shown in the yellow blocks. However this process is a joint action, so the grey blocks show the parallel activities in the 'other' workpackage.

In between the arrows, there are feedback loops and decision moments (GO/ NO GO) for further action.

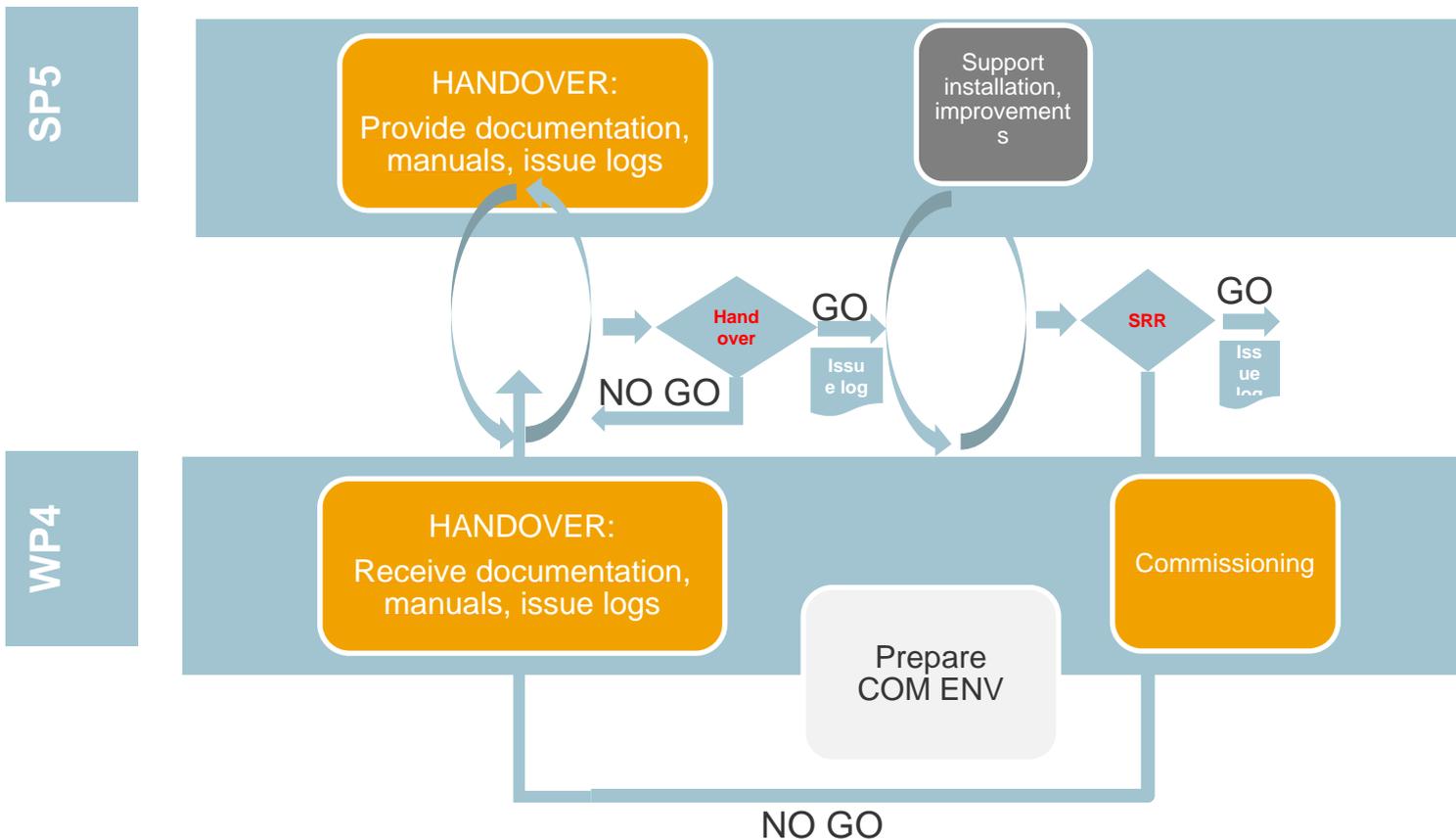


Figure 5 Transition of regions

5.1 Handover

First step is the handover of documentation by WP5 of the region to WP4. This is the official moment to deliver a package to WP4 which can be commissioned in the commissioning environment. The WP4 leader checks all documentation. As soon as he gives a GO, he is the one who needs to come to action and commission the region in the COM ENV environment

WP4 requires a set of documentation to be able to commission and operate a component. The documentation templates are based on the J-STD-016 standard.

The following set of documents is required for any service using MOBICENTRE.

Doc	Document type	Content
RIP	Region Installation Plan	Plan and instructions for configuration of the region.

5.2 Commissioning

WP4 will configure the region in MOBICENTRE according to instructions in the Region Installation Plan and Region User Manual. WP4 will verify that the region is ready to move to the production environment.

5.3 Region Readiness Review (RRR)

When commissioning has finished, the WP4 leader plans a meeting with all involved stakeholders to go through the findings. The participants in this meeting are:

- WP4.1 leader (chair),
- WP4.2 leader (operations manager),
- WP5 test site leader,
- the MOBINET business owner.

The outcome of the SRR will be a Go/No-go decision for installation into PROD ENV. The MOBINET Business Owner and WP4.2 Operations Manager need to accept the readiness of the region and are authorized to reject it

5.4 Installation on production

When the SRR decides on a Go for installation into PROD ENV, control is handed over to WP4.2 to activate the new region in PROD ENV and go live.

6 MOBICENTRE Operating Procedures

This section will include the operating procedures for starting up and installing the components on the MOBICENTRE environment. After the handover of the WP4-Operations from CGI to Tecnalia, and at the request of developers, some changes have been introduced in the accessing and operating procedures for them, which are described in the point 6.6 Changes after the handover.

6.1 Login on the OAM

Prerequisite for the instructions below is that the user should have Admin rights, or root rights, and some tooling (e.g. Putty to access the environment through a Secure Shell Host (SSH)). Furthermore, each user needs to have a Google Authenticator verification code.

The MOBICENTRE Environment is constructed such that the Admin User can access all components, virtual machines from the Operations, Administration and Maintenance (OAM) Server.

Open an SSH to the OAM Server with Admin username:
SSH Adminusername@OAM.com1.Mobinet .EU (for accessing the OAM on the commissioning environment). Note that only whitelisted IPs and users with Admin rights will get access to this server.

Once accessed, provide the Google Authenticator verification code followed by the user password.

The Admin User is now logged in into the OAM Server to perform the required operational, administrative or maintenance actions.

6.2 Accessing other VM's for installation

Once logged in on the OAM as described in section 6.1, one can log in into the different VM. This can be performed using SSH to the server-names. Google Authenticator verification code is not required in this case.

As the components are provided as RPMs, using the command line “yum install ...” suffice. For the App Directory it will look like: “yum install mobinet-app-directory”. See for all other components the provided installation manuals as embedded in section 7Annex: Component Operating Procedures.

Accessing the VMs from the OAM is for installation and administrative purposes. Logging in to the component for testing purposes, see section 6.3.

6.3 Login on the Components directly

Users can access the components using the URLs provided for these components and using a Username-Password for login if necessary.

For example, for the Dashboard on the Test Environment <http://dashboard.test1.mobinet.eu:8080>.
Respectively on the COM Environment <http://dashboard.com1.mobinet.eu:8080>.

6.4 Creating OAM User Account

Only roles requiring administrator rights will be granted access to the OAM Server. The OAM Super-user will add the user to the Admin user-list. A Google Authenticator barcode will be created and provided as an encrypted file to the user.

Access to the OAM server will be limited with only WP4 members responsible for commissioning and operating the system as the OAM server access will allow user to add/remove/modify installations on other servers.

6.5 Google Authenticator verification

Each user granted access to the MobiCentre is given a QR-Code for Google Authenticator.

To use this QR-Code, the user needs to install the Google Authenticator on his/her smartphone.

For Android Smartphones, Google Authenticator can be downloaded from Google Play Store.

iPhone users can download Google Authenticator from the Apple App-store.

Windows phone users can download Google Authenticator from the Windows store.

Google Authenticator is a token generator installed on the personal smartphone and act as an extra security layer for logging in into the MOBiCENTRE environment. In case username and password have been compromised, Google verification code will block unauthorized access.

6.6 Changes after the handover

As explained before, the aim of these changes has been to facilitate the daily work of developers in the TEST environment, providing more flexibility and easier access. The access to the COM environment continues to be more restrictive, given the nature of this environment, more prone to production than to testing processes.

A summary of the changes follow:

- Allowed direct access via SSH to TEST servers, and not only to intermediate (OAM) server. This access is based on user/password provided for each developer.
- Allowed file transfer (sftp / scp) to TEST servers
- Provided rights to kill/start/stop own services and processes in TEST servers.
- Provided access to JBoss and its console in TEST server.
- Provided access to the PostgreSQL database using the command line tool (psql) in TEST server.
- Provide read access to the log files in COM environment.

6.7 Installing a component

Installation of components is largely based on the use of RPMs. These RPM files have to be first placed in a repository, where the system is then able to find them. The instructions for this process are as follows:

In the TEST environment, a repository, called mobinet, was created in **OAM** node (in the path `"/opt/mobinet"`)

To add a new rpm to the Mobinet repository, a developer should:

1. Copy the rpm to the **"incoming_package"** directory located in the Home directory of each developer (e.g. `home/dev-lars/incoming_package`) in **OAM** node.
2. To load this rpm to the repository, run:

```
cd
```

```
sudo mobinet-add-rpm
```

3. After that, you could run (from any node in TEST environment, where you want to install the component):

```
yum install rpmname
```

In the COM environment, where the developers have not rights for this, is the administrator who has to upload the rpm and the copy it to the repository, with the instruction:

```
create repo opt/mobinet
```

Take into account that the component may need specific pre-requisites to be installed properly. For example, it may need a database with some tables and some user to be installed beforehand.

6.8 Running/Stopping a component

How a component should be started depends on the specific component instructions. Some components start automatically after installation, while others have to be started manually. The procedure here describes is a generic one. For detailed instructions, you should go to the component manual (see Annex: Component Operating Procedures)

For starting the component (e.g. the Service Directory), use the start-up script inside the `/etc/init.d/` folder:

```
mobinet-service-directory start or  
service mobinet-service-directory start
```

In order to modify the configuration properly, you have to stop the component first:

```
mobinet-service-directory stop or  
mobinet-service-directory stop
```

6.9 Uninstalling a component

First stop the service

```
service mobinet-service-directory stop
```

And then remove it:

```
yum remove mobinet-service-directory
```

6.10 Monitoring a component

How to monitor if a component is running properly is also dependent on the component itself. Some basic instructions are the following:

Locate or check if the service is running in the system, with the instruction:

```
ps -ef
```

Check the status of the service by its name (running, stopped, etc.)

```
service mobinet-service-directory status
```

Check the logs of the component (developers are allowed to read them in COM environment) in:

```
var/log/mobinet/service-directory/
```

7 Annex: Component Operating Procedures

For the commissioning it is fundamental to know how to install the components on the COM environment. Here for the provided documentation of the components are used. For installation manuals, please see following documentation included here in as an annex. This is also available in the Emdesk wiki, where it is updated by developers with every new release, allowing a more agile and easy access during the commissioning and maintenance work.

Any operations instructions for the components on the commissioning environment are included in these manuals. Findings regarding the provided installation manuals and installation of the components will be reported in the commissioning reports per release.

7.1 Service Directory

7.1.1 Component Description

7.1.1.1 Overview

Component:	Service Directory
Current version:	1.2-1
Work Package:	WP32
Contact Person:	Lars Mikkelsen [lmm@es.aau.dk]

Service directory consists of the components Dispatcher, Lookup indexer, Geo discovery, and storage. These components together comprises the service directory and offers service description management functionalities along with possibilities for lookup and discovery, based on service type and geographical coverage area of the service. This is offered through a REST interface.

7.1.1.2 Release Overview

Release number	Remarks / Release notes
1.1-5	The initial release.
1.2-1	MOBiCENTRE Release 1.1, bug-fixes

7.1.1.3 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.1.2 Installation

7.1.2.1 Pre-requisites

- System environment variables: NONE
- Java 7
- PostgreSQL 9.2
 - With predefined users
 - App directory: user:mobinet01 password:mobinet01
 - Service directory: user:mobinet02 password:mobinet02
 - DNS name for database: db

7.1.2.2 Installation Procedure

7.1.3 Pre-installation Tasks

- Ensure that the postgresQL server is setup correctly and started

7.1.4 Installation Procedure

Step	Task
1	yum install mobinet-service-directory

7.1.5 Post-installation

The service directory will start after the installation. In order to modify the config properly you have to stop it:

“mobinet-service-directory stop”

The config files are located in /root/loTArelease/config/.

For starting the Service Directory again, please use the start-up script inside the /etc/init.d/ folder:

- “mobinet-service-directory start”

Files:

- Install path: /opt/mobinet-service-directory/
- Log is placed in: /var/log/mobinet/servicedirectory
- Configuration files are placed in: /root/loTArelease/

Basic operational check:

- Insert service description by using a REST client (e.g. <http://code.fosshub.com/WizToolsorg-RESTClient/downloads>)
 - Select POST as HTTP Method
 - Insert <http://HOST:8090/iota/services/> as URL where host is the address of where the directory is installed
 - As request body insert the content of a service description, including insert request
 - The response should contain the service ID of the inserted service
- Lookup a service description by using a REST client
 - Select GET as HTTP Method
 - Insert <http://HOST:8090/iota/services/SERVICEID> as URL where HOST is the address of where the directory is installed and SERVICEID is the id of the service just inserted in above
 - The response should contain the service description

1.1.1 Uninstall or Roll-back

Stop the service directory:

```
“mobinet-service-directory stop”
“yum remove mobinet-service-directory”
```

7.1.6 Software Test Report

7.1.6.1 Insert Service Description

Test:	Insert service description
Requirement:	MP-84, (App directory: MP-39), (From service description: MP-85, MP-86, MP-87, MP-89, MP-90, MP-91)
Component:	Service Directory
Description:	<p>Test Summary</p> <p><i>With this test we are try to insert a Service Description inside the Service Repository and check if the POST Operation on</i></p> <p>Variances</p> <p>None</p> <p>Assessment</p> <p><i>Operation: POST Request</i></p> <p><i>Resource: http://localhost:8090/iota/services</i></p>

Body:

```

<?xml version="1.0" encoding="UTF-8"?>
<si:insertServiceRequest
  xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd"
  xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#">
<si:serviceDescription>
  <rdf:RDF>
    <sm:ResourceService rdf:ID="test_service_1">
      <sm:hasServiceEndpoint>
        <sm:RestfulServiceEndpoint>
          <sm:endpointPath rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >MOTESENSOR/IRIS_MOTE_02-00-00-00-27-F4-DC-E1/humidity</sm:endpointPath>
          <sm:supportsMethod
            rdf:resource="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#Read"/>
          <sm:endpointHost rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >localhost</sm:endpointHost>
          <sm:endpointPort rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
            >8554</sm:endpointPort>
          <sm:endpointProtocol rdf:datatype="http://www.w3.org/2001/XMLSchema#string"
            >http</sm:endpointProtocol>
        </sm:RestfulServiceEndpoint>
      </sm:hasServiceEndpoint>
      <sm:hasServiceArea>
        <sm:RectangularArea>
          <sm:NE_latitude
            rdf:datatype="http://www.w3.org/2001/XMLSchema#float">49.64197916173861</sm:NE_latitude>
          <sm:NE_longitude
            rdf:datatype="http://www.w3.org/2001/XMLSchema#float">14.403409211173315</sm:NE_longitude>
          <sm:SW_latitude
            rdf:datatype="http://www.w3.org/2001/XMLSchema#float">46.37015996342953</sm:SW_latitude>
          <sm:SW_longitude
            rdf:datatype="http://www.w3.org/2001/XMLSchema#float">12.27086910199098</sm:SW_longitude>
        </sm:RectangularArea>
      </sm:hasServiceArea>
      <sm:hasOutput
        rdf:datatype="http://purl.oclc.org/NET/ssnx/qu/quantity#QuantityKind">temperature</sm:hasOutput>
      <sm:hasServiceType>thermometer</sm:hasServiceType>
    </sm:ResourceService>
  </rdf:RDF>
</si:serviceDescription>
</si:insertServiceRequest>

```

Summary of results**Normal Response:**

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<si:insertServiceResponse
  xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#"

```

	<pre> xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:serviceId>test_service_1rt</si:serviceId> </si:insertServiceResponse> Anomalies: Service Already in the Storage <?xml version="1.0" encoding="UTF-8" standalone="yes"?><si:insertServiceResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:errorCode> <si:code>490</si:code> <si:reasonPhrase>WARNING</si:reasonPhrase> <si:details xsi:type="xs:string" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">SERVICE ID ALREADY IN THE STORAGE</si:details> </si:errorCode> </si:insertServiceResponse> Anomalies: Service Description not well formatted, XML error <?xml version="1.0" encoding="UTF-8" standalone="yes"?> <si:insertServiceResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:errorCode> <si:code>500</si:code> <si:reasonPhrase>INTERNAL ERROR</si:reasonPhrase> <si:details xsi:type="xs:string" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">Parser Exception!</si:details> </si:errorCode> </si:insertServiceResponse> </pre>
Status:	Passed

7.1.6.2 Lookup Service Description

Test:	Lookup Service Description
Requirement :	MP-83 (App directory MP-43)

Component:	Service Directory
Description:	<p>Test Summary</p> <p><i>With this operation we are trying to retrieve the Service Description previous inserted using a GET operation</i></p> <p>Variances</p> <p>None</p> <p>Assessment</p> <p><i>Operation: GET Request</i></p> <p><i>Resource: http://localhost:8090/iota/services/test_service_1</i></p> <p>Summary of Results</p> <p>Normal Response:</p> <pre><si:lookupServiceResponse xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:serviceDescription> <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"> <sm:ResourceService xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" rdf:ID="test_service_1"> <sm:hasServiceEndpoint> <sm:RestfulServiceEndpoint> <sm:endpointPath rdf:datatype="http://www.w3.org/2001/XMLSchema#string"> /MOTESENSOR/IRIS_MOTE_02-00-00-00-27-F4-DC-E1/humidity </sm:endpointPath> <sm:supportsMethod rdf:resource="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#Read"/> <sm:endpointHost rdf:datatype="http://www.w3.org/2001/XMLSchema#string">localhost</sm:endpointHost> <sm:endpointPort rdf:datatype="http://www.w3.org/2001/XMLSchema#int">8554</sm:endpointPort> <sm:endpointProtocol rdf:datatype="http://www.w3.org/2001/XMLSchema#string">http</sm:endpointProtocol> </sm:RestfulServiceEndpoint> </sm:hasServiceEndpoint> <sm:hasServiceArea> <sm:RectangularArea> <sm:NE_latitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">49.64197916173861</sm:NE_latitude> <sm:NE_longitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">14.403409211173315</sm:NE_longitude> <sm:SW_latitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">46.37015996342953</sm:SW_latitude> <sm:SW_longitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">12.27086910199098</sm:SW_longitude></pre>

D4.1b Operating Procedures

	<pre> </sm:RectangularArea> </sm:hasServiceArea> <sm:hasOutput rdf:datatype="http://purl.oclc.org/NET/ssnx/qu/quantity#QuantityKind">tempera ture</sm:hasOutput> <sm:hasServiceType>thermometer</sm:hasServiceType> </sm:ResourceService> </rdf:RDF> </si:serviceDescription> </si:lookupServiceResponse> Anomalies: Service Description not found <si:lookupoServiceResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntity Model.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="ht tp://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#"xmlns:rm="http://www.surrey.ac.uk/c csr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/Assoc iationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax- ns#"xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:errorCode> <si:code>404</si:code> <si:reasonPhrase>SERVICE ID NOT FOUND</si:reasonPhrase> </si:errorCode> </si:lookupoServiceResponse> </pre>
Status:	Passed

7.1.6.3 Discover Service

Test:	Discover Service
Requirement :	MP-83 (App directory MP-43)
Component:	Service directory
Description:	<p>Test Summary</p> <p><i>With this operation we are trying to discover a Service based on a combination of Service Type and Service Aera.</i></p> <p>Variances</p> <p>None</p> <p>Assessment</p>

Operation: POST request

Resource: <http://localhost:8090/iota/services/discover>

Body:

```
<?xml version="1.0" encoding="UTF-8"?>
<si:discoverServicesRequest
xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd">
<si:serviceSpecification>
    <si:sparql><![CDATA[
        PREFIX sm:<http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#>
        SELECT ?x
        WHERE {
            ?x sm:hasServiceType thermometer
        }]]>
    </si:sparql>
    <si:geographicScope>
        <si:rectangularGeographicScope>
            <si:NE_Latitude>49.644998</si:NE_Latitude>
            <si:NE_Longitude>16.194846</si:NE_Longitude>
            <si:SW_Latitude>46.368296</si:SW_Latitude>
            <si:SW_Longitude>11.272971</si:SW_Longitude>
        </si:rectangularGeographicScope>
    </si:geographicScope>
</si:serviceSpecification>
</si:discoverServicesRequest>
```

Summary of Results

Normal Response:

```
<si:lookupServiceResponse xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd"
xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd">
<si:serviceDescription>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<sm:ResourceService xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#"
rdf:ID="test_service_1">
<sm:hasServiceEndpoint>
<sm:RestfulServiceEndpoint>
<sm:endpointPath rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
/MOTESENSOR/IRIS_MOTE_02-00-00-00-27-F4-DC-E1/humidity
</sm:endpointPath>
<sm:supportsMethod rdf:resource="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#
Read"/>
<sm:endpointHost rdf:datatype="http://www.w3.org/2001/XMLSchema#string">localhost</sm:
endpointHost>
<sm:endpointPort rdf:datatype="http://www.w3.org/2001/XMLSchema#int">8554</sm:endpoint
```

D4.1b Operating Procedures

	<pre> Port> <sm:endpointProtocol rdf:datatype="http://www.w3.org/2001/XMLSchema#string">http</sm:endpointProtocol> </sm:RestfulServiceEndpoint> </sm:hasServiceEndpoint> <sm:hasServiceArea> <sm:RectangularArea> <sm:NE_latitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">49.64197916173861</sm:NE_latitude> <sm:NE_longitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">14.403409211173315</sm:NE_longitude> <sm:SW_latitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">46.37015996342953</sm:SW_latitude> <sm:SW_longitude rdf:datatype="http://www.w3.org/2001/XMLSchema#float">12.27086910199098</sm:SW_longitude> </sm:RectangularArea> </sm:hasServiceArea> <sm:hasOutput rdf:datatype="http://purl.oclc.org/NET/ssnx/qu/quantity#QuantityKind">temperature</sm:hasOutput> <sm:hasServiceType>thermometer</sm:hasServiceType> </sm:ResourceService> </rdf:RDF> </si:serviceDescription> </si:lookupServiceResponse> Anomalies: Service Id not Found <?xml version="1.0" encoding="UTF-8" standalone="yes"?> <si:discoverServicesResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/loTServiceResolutionInterface.xsd"> <si:errorCode> <si:code>404</si:code> <si:reasonPhrase>SERVICE ID NOT FOUND</si:reasonPhrase> </si:errorCode> </si:discoverServicesResponse> </pre>
Status:	Passed

7.1.6.4 Delete Service Description

Test:	Delete Service Description
Requirement:	MP-96

Component:	Service Directory
Description:	<p>Test Summary</p> <p><i>With this test we are trying to delete a Service Description from the Service Directory.</i></p> <p>Variances</p> <p>None</p> <p>Assessment</p> <p><i>Operation: DELETE request</i></p> <p><i>Resource: http://localhost:8090/iota/services/test_service_1</i></p> <p>Summary of Results</p> <p>Normal Response:</p> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <si:deleteServiceResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/IoTServiceResolutionInterface.xsd"> <si:statusCode> <si:code>200</si:code> <si:reasonPhrase>OK</si:reasonPhrase> </si:statusCode> </si:deleteServiceResponse></pre> <p>Anomalies: Service Id Not Found</p> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <si:deleteServiceResponse xmlns:vm="http://www.surrey.ac.uk/ccsr/ontologies/VirtualEntityModel.owl#" xmlns:ai="http://www.neclab.eu/schema/VEResolutionInterface.xsd" xmlns:sm="http://www.surrey.ac.uk/ccsr/ontologies/ServiceModel.owl#" xmlns:rm="http://www.surrey.ac.uk/ccsr/ontologies/ResourceModel.owl#" xmlns:am="http://www.surrey.ac.uk/ccsr/ontologies/AssociationModel.owl#" xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:si="http://www.neclab.eu/schema/IoTServiceResolutionInterface.xsd"> <si:statusCode> <si:code>404</si:code> <si:reasonPhrase>SERVICE ID NOT FOUND</si:reasonPhrase> </si:statusCode> </si:deleteServiceResponse></pre>
Status:	Passed

7.1.7 Software User Manual

Starting the Service Directory requires to start four components. To start /stop all the components you should run “/etc/init.d/mobinet-service-directory start/stop”:

Please refer to the Mobinet_Service_Directory_Interface_ver2.0.docx where the API are described for using the Service Directory and the App Directory. The document could be found in the Mobinet SVN: https://svn.dlr.de/mobinet/service_directory/trunk/service_directory_docs/Mobinet_Service_Directory_Interface_ver2.0.docx

7.2 Communication Agent

7.2.1 Component Description

7.2.1.1 Overview

Component:	Communications Agent
Current version:	0.0.3
Work Package:	WP35
Contact Person:	Nikola Zahariev [Nikola.Zahariev@neclab.eu]

7.2.1.2 Release Overview

Release number	Remarks / Release notes
0.0.3	The initial release.

7.2.1.3 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.2.2 Installation

7.2.2.1 Pre-requisites

- A mainstream OS supporting Java RT above v1.5, preferably v1.7 or v1.8

7.2.2.2 Installation Procedure

7.2.2.3 Pre-installation Tasks

None.

7.2.2.4 Installation Procedure

This section must be as clear and explicit as possible listing all the steps, in chronological order, the integrator must follow to complete the installation procedure.

Step	Task
1	Install the provided RPM packages using the yum package management tool.

7.2.2.5 Post-installation

- No files are modified during the installation. However, new files are added during the installation process.
- A logging configuration file logback.xml is located in /opt/mobinet-ca/etc. This is a standard configuration file of the Logback logging framework.
- Upon installation of the CA, one can check the logging file of the component at: /var/log/mobinet-ca/ca.log or any other location configured in the logback.xml configuration file (see previous point). In there a log entry stating that the component is listening for position updates on port 2001 and an HTTP server started on port 7180 should be listed. Additionally, one can check if ports 2001 and 7180 have open sockets (netstat).
- The configuration file /opt/mobinet-ca/etc/ca.config.properties can be used to configure different port numbers. Any changes to this file require a restart of the component.

7.2.2.6 Uninstall or Roll-back

The CA component can be uninstalled by using the yum package management tool. During the installation process all files created during the installation are deleted except for the logging file in /var/log/mobinet-ca/.

7.2.3 Software Test Report

7.2.3.1 Start CA

Test:	Start CA
Requirement:	
Component	CA
Description	Start the CA and observe the log file <code>/var/log/mobinet/ca/ca.log</code> . One should see an entries that the component is listening for Position updates on port 2001 and HTTP server for requests on port 80.
Status:	Passed

7.2.3.2 Receive new position update from a new node

Test:	Receive a new position update from a new node
Requirement:	
Component	CA
Description	When a new position entry arrives, it should processed correctly. The following log entry should be observed: Position entry with StationId {} scheduled! Position update entry with StationId {} added
Status:	Passed

7.2.3.3 Receive a position update from a known node

Test:	Receive a new position update from already know node
Requirement:	
Component	CA
Description	<p>When a new position entry arrives from a node already known, the following log entry should be observed:</p> <p>Position Update with StationId {} already exists, removing it.</p> <p>Position entry with StationId {} scheduled!</p> <p>Position update entry with StationId {} added</p>
Status:	Passed

7.2.3.4 Receive an dissemination request from a SP via the REST API

Test:	Receive an HTTP request from GLOSA/SP
Requirement:	
Component	CA
Description	<p>Receive a request from a Service Provider, e.g. GLOSA. A log entry should be logged:</p> <p>Received an HTTP request.</p>
Status:	Passed

7.2.3.5 Detect node inside dissemination area

Test:	Detect vehicle inside Dissemination Area
Requirement:	
Component	CA
Description	<p>When a matching node is found in the dissemination area of the SP, send the message to this node. Log entry:</p> <pre>Vehicle INSIDE dissemination area {}</pre> <pre>Sent packet to {}</pre>
Status:	Passed

7.2.3.6 Detect vehicle outside dissemination area

Test:	Detect vehicle outside Dissemination Area
Requirement:	
Component	CA
Description	<p>When a node is detected to be outside the dissemination area, the following log entry should be observed:</p> <pre>Vehicle OUTSIDE relevance area {}</pre>
Status:	Passed

7.2.3.7 Software User Manual

The CA can be started and stopped with the command:

```
# /etc/init.d/mobinet-ca {start|stop|restart}
```

A typical workflow looks as follows:

- Vehicles/nodes send position updates to the CA, which stores them
- A service provider request the CA to send information to vehicles/nodes within a certain geographical area by using predefined REST API
- CA sends messages to vehicles/nodes

7.3 Dashboard

7.3.1 Component Description

7.3.1.1 Overview

Component:	Dashboard
Current version:	1.1-4
Work Package:	WP37
Contact Person:	Benjamin Hebgen [Benjamin.Hebgen@neclab.eu]

7.3.1.2 Release Overview

Release number	Remarks / Release notes
1.0-12	The initial release.
1.1-4	MOBiCENTRE Release 1.1, bug-fixes

7.3.1.3 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.3.2 Installation

7.3.2.1 Pre-requisites

Prerequisites are of the order of

- System Environment Variables: CATALINA_HOME needs to be set to /opt/mobinet-dashboard/apache-rave-0.23 however the installer takes care of this

7.3.2.2 Pre-installation Tasks

None

7.3.2.3 Installation Procedure

Step	Task
1	yum install mobinet-dashboard

7.3.2.4 Post-installation

The dashboard will start after the installation. In order to modify the config properly you have to stop it with “/etc/init.d/mobinet-dashboard stop”. You can start it again with “/etc/init.d/mobinet-dashboard start”

In /opt/mobinet-dashboard/ modify the following the following files and exchange dashboard.test1.mobinet.eu with the correct FQDN

- apache-rave-0.23\webapps\portal\WEB-INF\classes\portal.properties (4 hits)
 - o Line 28: portal.opensocial_engine.root=dashboard.test1.mobinet.eu:8080
 - o Line 76: provider.wookie.wookieServerUrl=http://dashboard.test1.mobinet.eu:8080/wookie
 - o Line 101:
portal.mail.service.baseurl=http://dashboard.test1.mobinet.eu:8080/portal/app/changepassword/
 - o Line 108: portal.mail.service.loginpage=http://dashboard.test1.mobinet.eu:8080/portal/
- apache-rave-0.23\webapps\ROOT\WEB-INF\classes\config\oauth.json (3 hits)
 - o Line 21: "http://dashboard.test1.mobinet.eu:8080/gadgets/oauth.xml" : {
 - o Line 28: "http://dashboard.test1.mobinet.eu:8080/gadgets/shindigoauth.xml" : {
 - o Line 30: "consumer_key" :
"http://dashboard.test1.mobinet.eu:8080/gadgets/shindigoauth.xml",
- apache-rave-0.23\webapps\ROOT\WEB-INF\classes\rave.shindig.properties (1 hit)
 - o Line 30: shindig.host=dashboard.test1.mobinet.eu
- apache-rave-0.23\webapps\wookie\WEB-INF\classes\policies (1 hit)
 - o Line 37: * http://dashboard.test1.mobinet.eu:8080 ALLOW

7.3.2.5 Uninstall or Roll-back

Execute “/etc/init.d/mobinet-dashboard stop”

Yum remove mobinet-dashboard

7.3.3 Software Test Report

Test:	Login to Dashboard w/o OpenId
Requirement:	Login to Dashboard
Component:	Dashboard
Description:	<p>Start one of supported internet browsers and open the URL of Dashboard.</p> <p>Verify that Login page is shown.</p> <p>Enter an invalid user / pass</p> <p>Verify that the accessed denied page is shown</p> <p>Enter "canonical" as username and password</p> <p>Verify that you gained access to the dashboard</p> <p>Verify that you have access to the administrative area in the top right corner</p>
Status:	PASSED

Test:	Login to Dashboard w OpenId
Requirement:	Login to Dashboard
Component:	Dashboard
Description:	<p>Start one of supported internet browsers and open the URL of Dashboard.</p> <p>Verify that Login page is shown.</p> <p>Select corresponding Mobinet OpenID provider.</p> <p>Enter invalid username/password combination for the user.</p> <p>Verify that invalid message was shown.</p> <p>Enter valid username/password for the user.</p> <p>Verify that user was able to login and is redirected to the Dashboard.</p>
Status:	PASSED

D4.1b Operating Procedures

Test:	Check general widget loading
Requirement:	Login to Dashboard , Start javascript console, Widgets work only in Firefox properly at the moment
Component:	Dashboard
Description:	<p>Login to the Dashboard</p> <p>Start the javascript console and go to the network tab.</p> <p>Reload the page and check the network tab for any missing sources.</p>
Status:	PASSED

Test:	Check widget interaction
Requirement:	Login to Dashboard , Start javascript console, Widgets work only in Firefox properly at the moment
Component:	Dashboard
Description:	<p>Login to the Dashboard</p> <p>Start the javascript console and go to the network tab.</p> <p>Reload the page and check the network tab for any missing sources.</p> <p>Enter something in the service directory search widget and press search.</p> <p>Check if the service details widget is modified</p> <p>Check the network console for pubsub messages between the widgets.</p>
Status:	PASSED

Test:	Check opensocial widget loading
Requirement:	Login to Dashboard , Start javascript console
Component:	Dashboard
Description:	Login to the Dashboard

	<p>As canonical/canoical</p> <p>Create a new working are by clicking on the plus sign.</p> <p>Call it test.</p> <p>Go to the widget store and add the activity stream</p> <p>If it loaded properly you'll see a widget</p> <p>With Working on Rave...</p> <p>Ate a good sandwich!</p> <p>Traffic is bad right now...</p> <p>On it.</p> <p>Remove the working area by clicking on the small arrow next to the name and select delete working area.</p>
Status:	PASSED

Apache Rave and it's subcomponents are tested by their project. Visit the following websites for further information about tests and current problems.

Rave JIRA:

<https://issues.apache.org/jira/browse/RAVE/?selectedTab=com.atlassian.jira.jira-projects-plugin:summary-panel>

Shindig JIRA:

<https://issues.apache.org/jira/browse/SHINDIG/?selectedTab=com.atlassian.jira.jira-projects-plugin:summary-panel>

Wookie JIRA:

<https://issues.apache.org/jira/browse/WOOKIE/?selectedTab=com.atlassian.jira.jira-projects-plugin:summary-panel>

7.3.4 Software User Manual

- To start and stop execute /etc/init.d/mobinet-dashboard stop/start
- Once started go to <http://<hostname>:8080>.
- Login with user/pass: canonical/canonical or a valid openID account
- For further information about rave visit <http://rave.apache.org/documentation/index.html>

7.4 Identity Manager

7.4.1 Component Description

7.4.1.1 Overview

Component:	Identity Manager
Current version:	1.0.0
Work Package:	WP33
Contact Person:	Enzo Contini [enzo.contini@telecomitalia.it]

7.4.1.2 Release Overview

Release number	Remarks / Release notes
1.0.0	The initial release.

7.4.1.3 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.4.2 Installation

7.4.2.1 Pre-requisites

The *Identity Manager* module needs an application server (e.g. *JBoss*) installed on an Operating System (e.g. *CentOS*): moreover it has *OpenLDAP* as required package.

Here there is an estimation of the resources, even though they depends obviously on the number of simultaneously use of MIM sub modules:

- Network traffic income/outgoing (Mbps) < 10 kB/s
- Ports internally (inbound/outbound, TCP/UDP): 389 (*OpenLADP*)

In the project *svn repository* section for the *Identity Manager* module (https://svn.dlr.de/mobinet/identity_manager/trunk) there is the *rpms* directory with inside the *RPMS*,

SOURCES and SPECS files: in particular, a tag was created for RELEASE1 version tested in Helmond test1 environment (https://svn.dlr.de/mobinet/identity_manager/tags/REL1.0).

7.4.2.2 Pre-installation Tasks

OpenLDAP installation is required. The *mobinet-ldap-1.0.0-tilab.1.el6.x86_64.rpm* anyway has *OpenLDAP* as required package.

7.4.2.3 Installation Procedure

Step	Task
1	<p>Cleanup the yum cache since the cache would hide the IDM RPMs:</p> <pre>sudo /usr/bin/yum clean metadata</pre>
2	<p>Verify if <i>OpenLDAP</i> is already installed and maybe install it, even though the IDM RPM should require its installation so it should install it if not already installed.</p> <p>Anyway, if needed, to have a clean new <i>OpenLDAP</i> installation, the following procedure should be performed:</p> <pre>#remove previous installation yum remove openldap-servers openldap-clients rm -rf /etc/openldap/slapd.d/ rm -rf /etc/openldap/slapd.conf/ rm -rf /etc/openldap/schema/ rm -rf /var/lib/ldap/ #install yum openldap openldap-servers openldap-clients</pre>
3	<p>Install <i>mobinet-ldap</i> and <i>mobinet-identitymgr</i> RPMs:</p> <pre>sudo /usr/bin/yum install mobinet-ldap mobinet-identitymgr</pre> <p>In particular, the <i>mobinet-ldap</i> RPM provide the <i>OpenLDAP</i> installation and in <i>/opt/mobinet-ldap</i> the script to properly configure it. The <i>mobinet-identitymgr</i> RPM deploy in JBoss three JEE6 components are deployed (<i>LDAP EJB3.1 BE JEE6</i> component; <i>User Management FE JEE6</i> component, <i>MOBiNET-OpenID FE JEE6</i> component): moreover, the <i>MOBiNET-LoginSelector</i> additional component is also deployed, because it can be useful to test the <i>MOBiNET-OpenID Server</i>.</p>

4	<p>Run the script <i>mobinet-ldap-installer.sh</i>, provided by the <i>mobinet-ldap.rpm</i>, to properly configure LDAP:</p> <pre>sudo /opt/mobinet-ldap/mobinet-ldap-installer.sh</pre> <p>P.S. By making the files in <i>mobinet-ldap</i> owned by user <i>identitymgr</i> and group <i>identitymgr</i>, with mode 664, root privileges for modifying the configuration are not required anymore. The user and group may be assumed to be present.</p> <p>NOTE: be careful to see if some errors are displayed during the running of this script.</p> <p>At the end, you will be requested to “<i>Insert ldapdump_YYYYMMDD.ldif filename</i>”: for the actual installation, the filename is <i>ldapdump_20140506.ldif</i> (anyway, you can find the latest ldap dump file in the <i>/opt/mobinet-ldap</i>).</p> <p>P.S. We changed the <i>mobinet-ldap-installer.sh</i> in order that it can be executed passing the proper parameters (“<i>-f [dump file name]</i>” “<i>-n [yes/no]</i>” <i>yes for new installation, no otherwise</i>”). Only if one or all of these parameters are missing, the installer will ask for the missing user input.</p>
5	<p>Add to JBoss <i>standalone.xml</i> file the lines described in <i>standaloneIdentityMgr_stub.xml</i> file (the <i>mobinet-identitymgr.rpm</i> installation put in <i>/opt/jboss/jboss-eap-6.2/standalone/configuration</i>, the same directory where there is the JBoss <i>standalone.xml</i> that should be modified). See comments inside <i>standaloneIdentityMgr_stub.xml</i> file for more information: the proper URL address must be set in some parameters, depending where the <i>identitymgr</i> is installed.</p> <p>P.S. we add in the <i>mobinet-identitymgr-1.0.0.zip</i> not only the <i>standaloneIdentityMgr_stub.xml</i> but also a <i>standaloneExample.xml</i> that is the complete <i>standalone.xml</i> as it is now on the <i>identitymgr.test1.mobinet.eu</i>; however this is only an example (as it was already provided in the installation document) because some URL must be changed depending on the target machine and some other fields may change depending on the Jboss installation/version and that it is why I leave the <i>standaloneIdentityMgr_stub.xml</i> that must be the reference (it says what to delete and what to add from the pre-existing <i>standalone.xml</i> file).</p> <p>Anyway, also at the end of this document as an example, there is the full <i>standalone.xml</i> file as it is in the Helmond <i>test1</i> installation, with the added lines highlighted in yellow (in that case, the appropriate URL of the server was <i>identitymgr.test1.mobinet.eu</i>).</p>
	<p>Once deployed in JBoss all the relevant modules, the following URL must be used:</p> <p>http://hostname:80/MOBI-NET-IdentityManager to access the <i>User Management</i> site</p> <p>http:// hostname/MOBI-NET-OpenID/OpenId</p>

<p>to be used (e.g. from the Dashboard) to contact the <i>Mobinet OpenID Server</i> (http:// hostname/MOBiNET-LoginSelector to access the example login selector page, usable to test an <i>OpenID Server</i>)</p> <p>The predefined user that has the role of <i>PlatformAdmin</i> is:</p> <p>Username: m0binet</p> <p>Password: m0binet! (this default psw could be changed by that user from his profile page)</p>
--

NOTE: because JBoss on the VM must be accessible with his public URL (e.g. `identitymgr.test1.mobinet.eu`) even from that

machine, in *etc/hosts/* of that VM there is the need to set the public address and not the local one. Another possibility is to set in the iptable a nat redirect to the 80 port to the port used by JBoss (50080 in test1 Helmond JBoss) redirecting all the traffic coming from the private IP on 80 port to that JBoss configured port (NAT on iptables).

e.g. in the case of Helmond's test1 installation:

```
iptables -t nat -A OUTPUT --destination 172.20.33.3 -p tcp --dport 80 -j REDIRECT --to-ports 50080
```

```
iptables -t nat -A PREROUTING --destination 172.20.33.3 -p tcp --dport 80 -j REDIRECT --to-ports 50080
```

=====

The MOBiNET OpenID Server can be tested also using our test client (<http://identitymgr.test1.mobinet.eu/MOBiNET-LoginSelector>)

In the text field for the OpenId Identifier, you could insert <http://identitymgr.test1.mobinet.eu/MOBiNET-OpenID/OpenId> that is the OP identifier , not just the user Identifier. This is not managed yet.

If you should retrieve some attributes from the MOBiNET *OpenID Server*, you have to modify the file "*applicationContext-security.xml*" for the portal application:

```
<security:attribute-exchange identifier-match="http://identitymgr.test1.mobinet.eu/MOBiNET-OpenID/OpenId">
  <security:openid-attribute name="email" type="http://schema.openid.net/contact/email" required="true" />
  <security:openid-attribute name="firstname" type="http://axschema.org/namePerson/first"
required="true"/>
  <security:openid-attribute name="lastname" type="http://axschema.org/namePerson/last"
required="true"/>
</security:attribute-exchange>
```

=====

For your convenience, some accounts are already created in the LDAP, but except the PlatformAdmin one, the others could be deleted in next releases (the deletion of a user is not implemented yet): note that at least the *Platform Admin* user should be pre-defined in the platform, because it cannot be created by anyone, even though obviously he can change his profile (and then his password) ones logged].

Identity	Role	Party	Username	Password
DN: uid=m0binet,ou=people,dc=MOBiNET,dc=telecomitalia,dc=it	Platform Admin	All	m0binet	m0binet
DN: uid=jocondor,ou=people,dc=MOBiNET,dc=telecomitalia,dc=it	BU Party Admin	TI	jocondor enzoc test1admin	!condor1 contini1! test1ad
DN: uid= ammoby,ou=people,dc=MOBiNET,dc=telecomitalia,dc=it	BU Party User	TI	ammoby	!ammoby1
DN: uid=jocker,ou=people,dc=MOBiNET,dc=telecomitalia,dc=it	BU Party User	Ertico	jocker PartyProva1	jojo!1 m0binet
DN: uid= luigi,ou=people,dc=MOBiNET,dc=telecomitalia,dc=it	BU Party User	Ertico	luigi	grossi1!

- 1) **LDAPxMOBiNET.ear** => LDAP EJB3.1 BE JEE6 component. It handles the operations on OpenLdap (MOBiNET tree). This component if deployed creates a connection pool to the OpenLdap server and offers a service for other components (MOBiNET-IdentityManager and MOBiNET-OpenID) that have to access to the directory. It is configured for a directory server on the same machine (localhost), but is possible to set another IP address setting a binding on the *standalone.xml* of JBoss instance:

```
<simple name="java:global/custom/ldapserver/address" value="x.x.x.x"/>
```

Anyway, if the *LDAP directory server* is unavailable and/or the related service is not running, the two components that should use that service will run *inMemory* mode, using some cabled users previously defined.

- **MOBiNET-OpenID.war** => LDAP version of the OpenID Server. The component implements the server side of the OpenId 2.0 protocol for authentication request from a Relying Party. It uses, if available, the service exposed by the *LDAPxMOBiNET* component, for accessing to the directory server: if this service is unavailable, the component run *inMemory* mode and, in this case, the credentials to be used are: username = *m0binet* psw = *m0binet*
- **MOBiNET-IdentityManager.war** => LDAP version of the User Identity Management site. It uses, if available, the service exposed by the *LDAPxMOBiNET* component, for accessing to the

directory server: if this service is unavailable, the component run *inMemory* mode and, in this case, the credentials to be used are one of the three cabled (one for each role) described above. When logged in the *User Identity Management* site, different functions can be allowed depending on the role of that specific user (e.g. only the *Platform Admin* user can register a new *Party Admin*). It allows the following features for all the registered users: Login, ChangePassword, UpdateProfile, Logout and for users with a specific role, to create new MOBiNET users.

7.4.3 Software Test Report

The *MOBiNET-LoginSelector* site was used to test the *MOBiNET OpenID Server*. Selecting the MOBiNET icon button the proper login page (created by that server) is shown to the user and then the response (together with the requested identity requested information) is returned to the relying party and to the browser.

Related to the *User Identity Management site*, all the relevant trees, for Release 1 integration, were tested as follows.

Test:	Login with OPEN-ID
Requirement:	MP-54 Login to Dashboard
Component:	Identity Manager
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of Dashboard. • Verify that Login page is shown. • Select corresponding Mobinet <i>OpenID</i> provider. • Enter invalid username/password combination for the user. • Verify that invalid message was shown. • Enter valid username/password for the user. • Verify that user was able to login and is redirected to the Dashboard.
Status:	PASSED

Test:	PartyAdmin Users registration
Requirement:	MP-55 Register users
Component:	Identity Manager
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of IdentityManager. • Login as <i>PlatformAdmin</i> (default username/password <i>m0binet/m0binet!</i>).

D4.1b Operating Procedures

	<ul style="list-style-type: none"> • Select <i>Admins</i> tab and press New button. • Enter relevant attributes for the new user. • Confirm the user creation. • Log out as a <i>PlatformAdmin</i> and log in as the just created <i>PartyAdmin</i> user. • Verify that user was able to log in and can modify some of its attributes.
Status:	PASSED

Test:	Party/Developer Users registration
Requirement:	MP-55 Register users
Component:	Identity Manager
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of IdentityManager. • Login as <i>PartyAdmin</i> of a party (he must be previously created by the <i>PlatformAdmin</i>). • Select <i>PartyAccounts</i> tab and press New button. • Enter relevant attributes for the user, choosing if he is a <i>PartyUser</i> or a <i>Developer</i>. • Confirm the user creation. • Log out as a <i>PartyAdmin</i> and log in as the new created Party user. • Verify that user was able to login and can modify some of its attributes.
Status:	PASSED

Test:	End-Users registration
Requirement:	MP-79 Create new MOBiNET account
Component:	Identity Manager
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of IdentityManager. • In the <i>Login</i> page select the <i>New End-user</i> link • Enter relevant attributes for the user. • Confirm the user creation. • Verify that user was able to login and can modify some of its attributes.

Status:	PASSED
---------	--------

7.5 Service Directory Widgets

7.5.1 Component Description

7.5.1.1 Overview

Component:	Service/App Directory Widgets
Current version:	1.0-4
Work Package:	WP32
Contact Person:	Lars Mikkelsen [Imm@es.aau.dk]

The widgets offer a graphical front end to the service and app directories, to be used via the Dashboard. This is done by using the REST interface offered by the directories.

7.5.1.2 Release Overview

Release number	Remarks / Release notes
1.0-4	The initial release.

7.5.1.3 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.5.2 Installation

- Dashboard must be installed

7.5.2.1 Pre-installation Tasks

- Start Dashboard

7.5.2.2 Installation Procedure

Step	Task
1	yum install mobinet-widgets-app-service-directory

7.5.2.3 Post-installation

Files:

- Install path: /opt/mobinet-dashboard/mobinet-dashboard-[version]/apache-rave-0.23/webapps/ROOT/gadgets/
- Service directory base URL defined in: /opt/mobinet-dashboard/mobinet-dashboard-[version]/apache-rave-0.23/webapps/ROOT/gadgets/serviceLibs/restmanager.js
 - The variable “var baseUrl”
 - <http://servicedirectory.com1.mobinet.eu:8090/iota/services>
 - Change “com1” to “test1” to use test environment
- App directory base URL defined in: /opt/mobinet-dashboard/mobinet-dashboard-[version]/apache-rave-0.23/webapps/ROOT/gadgets/appLibs/restmanager.js
 - The variable “var baseUrl”
 - <http://appdirectory.com1.mobinet.eu:8094/iota/services>
 - Change “com1” to “test1” to use test environment

Basic operational check:

- Open Dashboard in Firefox Browser and log in
- If the widgets does not appear on a page then do the following steps to add them
- Select Widget Store
- Add Service Directory widgets:
 - Search for service
 - Add Service Deletion, Service Details, Service Discover, Service Search, Service Uploader
- Add App Directory widgets:
 - Search for app
 - Add App Deletion, App Details, App Discover, App Search, App Uploader
- In case the widgets mentioned does not appear when searching
 - Select add Add new widget
 - Under OpenSocial insert
<http://localhost:8080/gadgets/WIDGETNAME/WIDGETNAME.xml>
 - Where WIDGETNAME is:
 - serviceDelete
 - serviceDetails
 - serviceDiscovery
 - serviceSearch
 - serviceUpload
 - appDelete

- appDetails
 - appDiscovery
 - appSearch
 - appUpload
- Select Back to Rave
 - Verify that the widgets are added to a page

7.5.2.4 Uninstall or Roll-back

yum remove mobinet-service-app-directory-widgets

7.5.3 Software Test Report

Test:	Connect to Service Directory
Requirement:	
Component:	Dashboard / Service Directory
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of Dashboard. • Login to the Dashboard • Open the “Service-Directory-Widgets” page • Enter “mobinet” in the lookup widget • Verify that the result is shown in the result-widget
Status:	PASSED

Test:	Connect to App Directory
Requirement:	
Component:	Dashboard / App Directory
Description:	<ul style="list-style-type: none"> • Start one of supported internet browsers and open the URL of Dashboard. • Login to the Dashboard • Open the “App-Directory-Widgets” page • Enter “mobinet” in the lookup widget • Verify that the result is shown in the result-widget
Status:	PASSED

7.6 Telematics Service Providers Manager

7.6.1 Component Description

7.6.1.1 Overview

Component:	TSP Manager
Current version:	1.0-2.1
Work Package:	WP32
Contact Person:	Cercato Pierandrea [pierandrea.cercato@allianz.it]

The TSP Manager is responsible for collecting telematics data from TSPs (Telematic Service Providers) and distributing it to registered users like insurance providers or traffic advice systems. It handles the communication between these parties and defines data formats to enable a consistent and common understanding of the telematics data.

7.6.1.2 Component Description

The overall responsibilities of the TSP Manager are:

- All the communications between TSPs (Telematic Service Providers) and IP (Insurance Providers);
- Communication between IP and TSPs when there is a change of IP; and,
- Communication between different TSPs for delivering data from a vehicle to the selected IP;

MOBiNET will be notified by the new IP, chosen by the end consumer. At this point MOBiNET will do the following:

- Inform the previous IP that it will no longer receive data from the TSP in charge of the previous customer's on-board device
- Notify the two TSPs involved, the one of the previous IP and the one of the current IP

This will allow the correct handling of the UBI data flow. The information retrieved from the TSP of the previous IP will first pass through MOBiNET and then through the new IP TSP in order to reach the new IP.

Main use cases can be divided in two categories, the ones belonging to data provisioning on IP switch, and the ones belonging to Telematics data transfer (when the switch is done).

Principio del formulario

Final del formulario

1.1.2 Database Description

The following database table will be used by the TSP Manager:

Table Name	Description
ip_provider	Basic information on the registered IPs (e.g. name, server IP and port...)
ip_service_provider	Services provided of the registered IPs (e.g. description,dataendpointurl, serviceendpointurl ...)
tsp_provider	Basic information on the registered TSPs (e.g. name, server IP and port...)
tsp_service_provider	Services provided of the registered TSPs (e.g. description,dataendpointurl, serviceendpointurl ...)
contract	Information on the contracts (e.g. id,start/enddate, vehicle info...)
customer	Information on the customer (e.g. name, surname...)
contract_customer	Link between contracts and customers
contract_history	Link between IPs and a contract to trace its history

Principio del formulario

Final del formulario

1.1.3 Interface Description

The TSP Manager offers an external SOAP interface for other external components/services. This interface is used by IP to trigger the UBI IP switch.

A JDBC interface is used for communication with the database.

Interface Name	Port	Type	Description
JDBC	5432	TCP (internal)	Communication with the PostgreSQL Database
SOAP	8380	TCP (external)	Communication with external components/services

7.6.1.3 Release Overview

Release number	Remarks / Release notes
1.0-2.1	Fix RPMs issues

1.0-2	Fix RPMs issues
1.0-1	The initial release.

7.6.1.4 Installation and Upgrade Notes

If applicable, instructions on how to install or update from previous version.

7.6.2 Installation plan

7.6.2.1 Pre-requisites

- System environment variables: NONE
- Java 7
- PostgreSQL 9.2
 - With predefined users: user:mobinet03 password:mobinet03
 - DNS name for database: db

7.6.2.2 DB Setup

```
psql -d mobinetdb_phase1 -U mobinet03 -h db -a -f tspmanager_DDL.sql
```

tspmanager_DDL.sql

```
CREATE SCHEMA tspmanager AUTHORIZATION mobinet03;
DROP TABLE IF EXISTS tspmanager.tsp_provider;
CREATE TABLE tspmanager.tsp_provider
(
  name character varying(255) NOT NULL,
  surname character varying(255) NOT NULL,
  phone character varying(255) NOT NULL,
  email character varying(255) NOT NULL,
  openidurl character varying(255) NOT NULL,
  party character varying(255) NOT NULL,
  legalid character varying(255),
  address character varying(255),
  zipcode character varying(255),
  city character varying(255),
  province character varying(255),
  country character varying(255),
  CONSTRAINT tsp_provider_pk PRIMARY KEY (openidurl),
  CONSTRAINT tsp_email_uk UNIQUE (email)
);
ALTER TABLE tspmanager.tsp_provider OWNER TO mobinet03;
```

D4.1b Operating Procedures

```
DROP TABLE IF EXISTS tspmanager.tsp_service_provider;
CREATE TABLE tspmanager.tsp_service_provider
(
  openidurl character varying(255) NOT NULL,
  name character varying(255) NOT NULL,
  description character varying(255) NOT NULL,
  dataendpointurl character varying(255) NOT NULL,
  serviceendpointurl character varying(255) NOT NULL,
  CONSTRAINT tsp_service_provider_pk PRIMARY KEY (openidurl, name),
  CONSTRAINT tsp_provider_fk_001 FOREIGN KEY (openidurl)
    REFERENCES tspmanager.tsp_provider (openidurl) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
  CONSTRAINT tsp_name_uk UNIQUE (name)
);
ALTER TABLE tspmanager.tsp_service_provider OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.ip_provider;
CREATE TABLE tspmanager.ip_provider
(
  name character varying(255) NOT NULL,
  surname character varying(255) NOT NULL,
  phone character varying(255) NOT NULL,
  email character varying(255) NOT NULL,
  openidurl character varying(255) NOT NULL,
  party character varying(255) NOT NULL,
  legalid character varying(255),
  address character varying(255),
  zipcode character varying(255),
  city character varying(255),
  province character varying(255),
  country character varying(255),
  CONSTRAINT ip_provider_pk PRIMARY KEY (openidurl),
  CONSTRAINT email_uk UNIQUE (email)
);
ALTER TABLE tspmanager.ip_provider OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.ip_service_provider;
CREATE TABLE tspmanager.ip_service_provider
(
  openidurl character varying(255) NOT NULL,
```

D4.1b Operating Procedures

```
name character varying(255) NOT NULL,
description character varying(255) NOT NULL,
dataendpointurl character varying(255) NOT NULL,
serviceendpointurl character varying(255) NOT NULL,
CONSTRAINT service_ip_provider_pk PRIMARY KEY (openidurl, name),
CONSTRAINT ip_provider_fk_001 FOREIGN KEY (openidurl)
    REFERENCES tspmanager.ip_provider (openidurl) MATCH SIMPLE
    ON UPDATE NO ACTION ON DELETE NO ACTION,
CONSTRAINT name_uk UNIQUE (name)
);
ALTER TABLE tspmanager.ip_service_provider OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.contract;
CREATE TABLE tspmanager.contract
(
    unique_id serial NOT NULL,
    ip character varying(255) NOT NULL,
    tsp character varying(255) NOT NULL,
    begin_date date NOT NULL,
    end_date date NOT NULL,
    vehicle_plate character varying(255),
    vehicle_brand character varying(255),
    vehicle_model character varying(255),
    ip_contract_uid character varying(255),
    vehicle_vin character varying(255),
    CONSTRAINT contract_pk PRIMARY KEY (unique_id),
    CONSTRAINT contract_ip_fkey FOREIGN KEY (ip)
        REFERENCES tspmanager.ip_provider (openidurl) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT contract_tsp_fkey FOREIGN KEY (tsp)
        REFERENCES tspmanager.tsp_provider (openidurl) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
);
ALTER TABLE tspmanager.contract OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.customer;
CREATE TABLE tspmanager.customer
(
    unique_id serial NOT NULL,
    name character varying(255),
    surname character varying(255),
```

```
    legacy_id character varying(255),
    CONSTRAINT customer_pk PRIMARY KEY (unique_id)
);
ALTER TABLE tspmanager.customer OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.contract_customer;
CREATE TABLE tspmanager.contract_customer
(
    contract integer NOT NULL,
    customer integer NOT NULL,
    CONSTRAINT contr_cust_fk001 FOREIGN KEY (contract)
        REFERENCES tspmanager.contract (unique_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT contr_cust_fk002 FOREIGN KEY (customer)
        REFERENCES tspmanager.customer (unique_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
);
ALTER TABLE tspmanager.contract_customer OWNER TO mobinet03;

DROP TABLE IF EXISTS tspmanager.contract_history;
CREATE TABLE tspmanager.contract_history
(
    contract integer NOT NULL,
    ip character varying(255) NOT NULL,
    tsp character varying(255) NOT NULL,
    insert_date timestamp with time zone NOT NULL DEFAULT now(),
    id serial NOT NULL,
    ip_contract_uid character varying(255),
    CONSTRAINT contr_hist_fk_001 FOREIGN KEY (contract)
        REFERENCES tspmanager.contract (unique_id) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT contr_hist_ip_fk_002 FOREIGN KEY (ip)
        REFERENCES tspmanager.ip_provider (openidurl) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION,
    CONSTRAINT contr_hist_tsp_fk_002 FOREIGN KEY (tsp)
        REFERENCES tspmanager.tsp_provider (openidurl) MATCH SIMPLE
        ON UPDATE NO ACTION ON DELETE NO ACTION
);
ALTER TABLE tspmanager.contract_history OWNER TO mobinet03;
```

7.6.2.3 Installation

```
yum install TSPManager
```

7.6.2.4 Configuration

Modify the following parameter under the HTTP and HTTPS transport receiver configurations in the `/opt/mobinet-tspmanager/wso2esb-4.8.1/repository/conf/axis2/axis2.xml` file.

Find the `WSDLEPRPrefix` parameter and enter as value the hostname of the environment (e.g. `tspm.test2.mobinet.eu` for test2, `tspm.com2.mobinet.eu` for com2)

```
<parameter name="WSDLEPRPrefix" locked="false">tspm.test2.mobinet.eu</parameter>
```

The following sample HTTP and HTTPS configuration shows how to listen the hostname `tspm.test2.mobinet.eu`.

axis2.xml

```
<transportReceiver name="http"
class="org.apache.synapse.transport.passthru.PassThroughHttpListener">
    <parameter name="port" locked="false">8380</parameter>
    <parameter name="non-blocking" locked="false">true</parameter>
    <parameter name="WSDLEPRPrefix"
locked="false">http://tspm.test1.mobinet.eu</parameter>
    <parameter name="httpGetProcessor"
locked="false">org.wso2.carbon.transport.nhttp.api.PassThroughNHttpGetProcessor</parameter>
</transportReceiver>

<transportReceiver name="https"
class="org.apache.synapse.transport.passthru.PassThroughHttpSSLListener">
    <parameter name="port" locked="false">8243</parameter>
    <parameter name="non-blocking" locked="false">true</parameter>
    <parameter name="WSDLEPRPrefix"
locked="false">https://tspm.test1.mobinet.eu</parameter>
    <parameter name="httpGetProcessor"
locked="false">org.wso2.carbon.transport.nhttp.api.PassThroughNHttpGetProcessor</parameter>

    <parameter name="keystore" locked="false">
        <KeyStore>
            <Location>repository/resources/security/wso2carbon.jks</Location>
            <Type>JKS</Type>
            <Password>wso2carbon</Password>
            <KeyPassword>wso2carbon</KeyPassword>
        </KeyStore>
    </parameter>
    <parameter name="truststore" locked="false">
        <TrustStore>
```

```
truststore.jks</Location>      <Location>repository/resources/security/client-
                                <Type>JKS</Type>
                                <Password>wso2carbon</Password>
                                </TrustStore>
                                </parameter>
                                <!--<parameter name="SSLVerifyClient">require</parameter>
                                supports optional|require or defaults to none -->
                                </transportReceiver>
```

Start the TSP Manager:

```
/etc/init.d/mobinet-tspmanager start
```

7.6.2.5 Uninstall

Stop the TSP Manager:

```
/etc/init.d/mobinet-tspmanager stop
```

Remove it:

```
yum remove TSPManager
```

7.6.3 Operations User Manual

To start /stop the TSP Manager run the following:

```
/etc/init.d/mobinet-tspmanager start|stop
```

To verify the status of the TSP Manager run the following:

```
/etc/init.d/mobinet-tspmanager status
```

7.6.3.1 Files

- Install path: /opt/mobinet-tspmanager/
- Log is placed in: /opt/mobinet-tspmanager/wso2esb-4.8.1/repository/logs/
- Configuration files are placed in: /opt/mobinet-tspmanager/wso2esb-4.8.1/repository/conf

7.6.3.2 Basic operational check

Web console is available at the following URL: <https://<ip-address>:9443/carbon using credentials admin/admin>

7.7 Data Quality Assessment

7.7.1 Component Description

7.7.1.1 Basic Information

Component:	Data Quality Assessment (DQA)
Current version:	-
Work Package:	WP3.11
Contact Person:	Francesco Alesiani[francesco.alesiani@neclab.eu]; Phone: +49 (0)6221 4342 253

7.7.1.2 Description

Data Quality Assessment tool is meant to support the evaluation of traffic data set using a standardized service. The user is able to get indicators associated to traffic data by using the REST API offered by MOBINET.

The current version of the service focused on the analysis of fixed sensors (e.g.: loop counters). These indicators gives indication of presence of:

1. Type 1: Zero occupancy measures
2. Type 2: Non-zero occupancy with zero flow
3. Type 3: High occupancy
4. Type 4: Constant measure

These indicators represent a way to first assess of the presence of not consistent measures using a standardized services. The service will evolve to include also indicators for floating vehicle data (FVD).

Generally speaking we design indicators to assess the quality of incoming data on two distinct aspects: Accuracy and Value. The first one addresses the problem of determining the level of reliability of the measurements obtained by each sensor (e.g.: difference between the obtained measurements and the real ones; anomalous sensor detections). On the other hand, the second axis is focused on evaluating how representative such data source is regarding the coverage provided on the human mobility activities on a given urban area. These data quality assessment tools provide input to the development of the certification framework.

7.7.1.3 Release Overview

Release number	Remarks / Release notes
----------------	-------------------------

1.1-7	The initial release.

7.7.2 Architecture

The service is composed of a frontend that is responsible of handling REST request. The DQA library computes the Quality indicators and manages the interface with the internal structures.

The DQA service is composed of some entities. The main entity is the measure. The Assessment is the function that act on the measure. Summary entity provide summary information on the data quality indicators. Measures are store in separate databases. The access to the databases is defined by the role entity. Each user can be associated to one or more database via the role entity. The following figure shows the REST API entities and actions of the DQA service.

7.7.2.1 Database Description

The following database table will be used by the Service Directory:

Table Name	Description
users	Store the user information, namely the user name and the authentication parameters
databases	Stores the databases connected to the service
roles	Store the connection between users and databases
measure	Store the actual measure. This table is actually distributed in each database

7.7.2.2 Interface Description

The DQA offers an REST-API as an external interface for other components or services.

Interface Name	Port	Type	Description
REST-API	8082	TCP (external)	access to the REST API

7.7.3 Installation Plan

7.7.3.1 Pre-requisites

D4.1b Operating Procedures

- Python 2.7
- python libraries: random, logging, sys, os,datetime,time, sqlite3,json,hashlib,string, math, csv, signal, BaseHTTPServer, ssl, cgi, urlparse, ConfigParser, httplib

7.7.3.2 Installation

```
yum install mobinet-dqa
```

7.7.3.3 Uninstall / Service Removal

```
yum remove mobinet-dqa
```

7.7.3.4 Start/ Stop service

```
/etc/init.d/mobinet-dqa start|stop
```

7.7.3.5 Service configuration

To Start and stop the service it is possible to use the command

```
/etc/init.d/mobinet-dqa start|stop
```

The service can be configured by editing the configuration file dqa.cnf

The databased are located in the “db” folder. Each database has a separate file. There is a common configuration file with the user information.

7.7.3.6 Files

- Install path: /opt/mobinet-dqa/
- Log is placed in: /opt/mobinet-dqa/log/
- Databases are placed in: /opt/mobinet-dqa/db/
- Configuration files is located in: /opt/mobinet-dqa/dqa.cnf

7.7.3.7 Basic operational check

- Insert/Get/Remove Measure
- Get the assessment result
- Add User, Database and Role

7.7.4 DQA API

General

```
http://[ip address]:8082//dataquality/assessment
http://[ip address]:8082//dataquality/measure
http://[ip address]:8082//dataquality/summary
http://[ip address]:8082//dataquality/user
http://[ip address]:8082//dataquality/db
http://[ip address]:8082//dataquality/role
```

Commissioning environment host: <http://???com1.mobinet.eu:8082//dataquality>

Test Environment host: <http://???test1.mobinet.eu:8082//dataquality>

7.7.5 Installing Python2.7

Install pre-requisite packages

```
yum groupinstall "Development tools"
yum install zlib-devel bzip2-devel openssl-devel ncurses-devel sqlite-devel xz-libs
gcc
cd /opt
wget --no-check-certificate https://www.python.org/ftp/python/2.7.10/Python-
2.7.10.tar.xz
tar xf Python-2.7.10.tar.xz
cd Python-2.7.10
./configure --prefix=/usr/local
make && make altinstall
wget https://bitbucket.org/pypa/setuptools/raw/bootstrap/ez_setup.py
/usr/local/bin/python2.7 ez_setup.py
/usr/local/bin/easy_install-2.7 pip
```

verify the different version of Python

```
ls -ltr /usr/bin/python*
ls -ltr /usr/local/bin/python*
echo $PATH
```

if local is not present, it is possible to add as follow

```
export PATH="/usr/local/bin:$PATH"
```

To install the setuptools

```
# downloading the installation file using wget:
wget --no-check-certificate
https://pypi.python.org/packages/source/s/setuptools/setuptools-18.0.tar.gz
# Extract the files from the archive:
tar -xvf setuptools-18.0.tar.gz
# Enter the new directory:
cd setuptools-18.0
# Install setuptools
python2.7 setup.py install
```

8 Annex: Document Templates

This Annex lists the documents that are required for applications, services and regions when they are commissioned into MOBICENTRE. The actual templates are included in the next pages and can be used as a basis when preparing such a document.

Document templates for WP3 applications

Doc	Document type	Content
SSS	System/Subsystem Specification	System and interface requirements, quality factors. Needed if the application consists of multiple software components.
SRS	Software Requirements Specification	Software modules and interface requirements, quality factors.
DBDD	Database Design Description	Database design and schemas. Only needed if the component uses a database.
SIP	Software Installation Plan	Plan and instructions for installation and configuration of the software.
STR	Software Test Report	Results of prior module and system tests in TEST ENV.
SUM	Software User Manual	User manual to operate the system/component.
SVD	Software Version Description	Release notes with list of files, solved issues and known issues.

Document templates for WP7 services

Doc	Document type	Content
SIP	Service Installation Plan	Plan and instructions for configuration of the service.
STR	Service Test Report	Results of prior testing of the service in TEST ENV.
SUM	Service User Manual	User manual to operate the system/component.
SVD	Service Version Description	Release notes with list of files, solved issues and known issues.

Document templates for WP5 regions

Doc	Document type	Content
RIP	Region Installation Plan	Plan and instructions for configuration of the region.

8.1 SRS Template

8.1.1 Scope

8.1.1.1 Identification

This paragraph shall contain a full identification of the system to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

8.1.1.2 System overview

This paragraph shall briefly state the purpose of the system to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

8.1.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.1.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.1.3 Requirements

This section shall be divided into the following paragraphs to specify the system requirements, that is, those characteristics of the system that are conditions for its acceptance. Each requirement shall be assigned a project-unique identifier to support testing and traceability and shall be stated in such a way that an objective test can be defined for it.

Each requirement shall be annotated with associated qualification method(s) (see section 4) and, for subsystems, traceability to system requirements (see section 5.a), if not provided in those sections. The degree of detail to be provided shall be guided by the following rule: Include those characteristics of the system that are conditions for system acceptance; defer to design descriptions those characteristics that the acquirer is willing to leave up to the developer. If there are no requirements in a given paragraph, the paragraph shall so state. If a given requirement fits into more than one paragraph, it may be stated once and referenced from the other paragraphs.

8.1.3.1 Required states and modes.

If the system is required to operate in more than one state or mode having requirements distinct from other states or modes, this paragraph shall identify and define each state and mode. Examples of states and modes include: idle, ready, active, postuse analysis, training, degraded, emergency, backup, wartime, peacetime. The distinction between states and modes is arbitrary. A system may be described in terms of states only, modes only, states within modes, modes within states, or any other scheme that is useful. If no states or modes are required, this paragraph shall so state, without the need to create artificial distinctions. If states and/or modes are required, each requirement or group of requirements in this specification shall be correlated to the states and modes. The correlation may be indicated by a table or other method in this paragraph, in an appendix referenced from this paragraph, or by annotation of the requirements in the paragraphs where they appear.

8.1.3.2 System capability requirements

This paragraph shall be divided into subparagraphs to itemize the requirements associated with each capability of the system. A "capability" is defined as a group of related requirements. The word "capability" may be replaced with "function," "subject," "object," or other term useful for presenting the requirements.

D4.1b Operating Procedures

8.1.3.3 (System capability)

This paragraph shall identify a required system capability and shall itemize the requirements associated with the capability. If the capability can be more clearly specified by dividing it into constituent capabilities, the constituent capabilities shall be specified in subparagraphs. The requirements shall specify required behavior of the system and shall include applicable parameters, such as response times, throughput times, other timing constraints, sequencing, accuracy, capacities (how much/how many), priorities, continuous operation requirements, and allowable deviations based on operating conditions. The requirements shall include, as applicable, required behavior under unexpected, unallowed, or "out of bounds" conditions, requirements for error handling, and any provisions to be incorporated into the system to provide continuity of operations in the event of emergencies. Paragraph 3.3.x of this DID provides a list of topics to be considered when specifying requirements regarding inputs the system must accept and outputs it must produce.

8.1.3.4 (System capability)

There shall be sufficient 3.2.x subsections to cover all system capabilities.

8.1.4 System external interface requirements

This paragraph shall be divided into subparagraphs to specify the requirements, if any, for the system's external interfaces. This paragraph may reference one or more Interface Requirements Specifications (IRs) or other documents containing these requirements.

8.1.5 Interface identification and diagrams

This paragraph shall identify the required external interfaces of the system. The identification of each interface shall include a project-unique identifier and shall designate the interfacing entities (systems, configuration items, users, etc.) by name, number, version, and documentation references, as applicable. The identification shall state which entities have fixed interface characteristics (and therefore impose interface requirements on interfacing entities) and which are being developed or modified (thus having interface requirements imposed on them). One or more interface diagrams shall be provided to depict the interfaces.

8.1.5.1 (Project-unique identifier of interface)

This paragraph (beginning with 3.3.2) shall identify a system external interface by project-unique identifier, shall briefly identify the interfacing entities, and shall be divided into subparagraphs as needed to state the requirements imposed on the system to achieve the interface. Interface characteristics of the other entities involved in the interface shall be stated as assumptions or as "When [the entity not covered] does this, the system shall..." not as requirements on the other entities. This paragraph may reference other documents (such as data dictionaries, standards for communication protocols, and standards for user interfaces) in place of stating the information here. The requirements shall include the following, as applicable, presented in any order suited to the requirements, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

- a. Priority that the system must assign the interface
- b. Requirements on the type of interface (such as real-time data transfer, storage-and retrieval of data, etc.) to be implemented
- c. Required characteristics of individual data elements that the system must provide, store, send, access, receive, etc., such as:
 - 1) Names/identifiers
 - a) Project-unique identifier
 - b) Non-technical (natural-language) name
 - c) DoD standard data element name
 - d) Technical name (e.g., variable or field name in code or database)

D4.1b Operating Procedures

e) Abbreviation or synonymous names

- 2) Data type (alphanumeric, integer, etc.)
- 3) Size and format (such as length and punctuation of a character string)
- 4) Units of measurement (such as meters, dollars, nanoseconds)
- 5) Range or enumeration of possible values (such as 0-99)
- 6) Accuracy (how correct) and precision (number of significant digits)
- 7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply
- 8) Security and privacy constraints
- 9) Sources (setting/sending entities) and recipients (using/receiving entities)

d. Required characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the system must provide, store, send, access, receive, etc., such as:

- 1) Names/identifiers
 - a) Project-unique identifier
 - b) Non-technical (natural language) name
 - c) Technical name (e.g., record or data structure name in code or database)
 - d) Abbreviations or synonymous names
- 2) Data elements in the assembly and their structure (number, order, grouping)
- 3) Medium (such as disk) and structure of data elements/assemblies on the medium
- 4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)
- 5) Relationships among assemblies, such as sorting/access characteristics
- 6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply
- 7) Security and privacy constraints
- 8) Sources (setting/sending entities) and recipients (using/receiving entities)

e. Required characteristics of communication methods that the system must use for the interface, such as:

- 1) Project-unique identifier(s)
- 2) Communication links/bands/frequencies/media and their characteristics
- 3) Message formatting
- 4) Flow control (such as sequence numbering and buffer allocation)
- 5) Data transfer rate, whether periodic/aperiodic, and interval between transfers
- 6) Routing, addressing, and naming conventions
- 7) Transmission services, including priority and grade

D4.1b Operating Procedures

8) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing

f. Required characteristics of protocols the system must use for the interface, such as:

- 1) Project-unique identifier(s)
- 2) Priority/layer of the protocol
- 3) Packeting, including fragmentation and reassembly, routing, and addressing
- 4) Legality checks, error control, and recovery procedures
- 5) Synchronization, including connection establishment, maintenance, termination
- 6) Status, identification, and any other reporting features

g. Other required characteristics, such as physical compatibility of the interfacing entities (dimensions, tolerances, loads, plug compatibility, etc.), voltages, etc.

8.1.5.2 System internal interface requirements

This paragraph shall specify the requirements, if any, imposed on interfaces internal to the system. If all internal interfaces are left to the design or to requirement specifications for system components, this fact shall be so stated. If such requirements are to be imposed, paragraph 3.3 of this DID provides a list of topics to be considered.

8.1.5.3 System internal data requirements

This paragraph shall specify the requirements, if any, imposed on data internal to the system. Included shall be requirements, if any, on databases and data files to be included in the system. If all decisions about internal data are left to the design or to requirements specifications for system components, this fact shall be so stated. If such requirements are to be imposed, paragraphs 3.3.x.c and 3.3.x.d of this DID provide a list of topics to be considered.

8.1.5.4 Adaptation requirements

This paragraph shall specify the requirements, if any, concerning installation-dependent data that the system is required to provide (such as site-dependent latitude and longitude or site-dependent state tax codes) and operational parameters that the system is required to use that may vary according to operational needs (such as parameters indicating operation-dependent targeting constants or data recording).

8.1.5.5 Safety requirements

This paragraph shall specify the system requirements, if any, concerned with preventing or minimizing unintended hazards to personnel, property, and the physical environment. Examples include restricting the use of dangerous materials; classifying explosives for purposes of shipping, handling, and storing; abort/escape provisions from enclosures; gas detection and warning devices; grounding of electrical systems; decontamination; and explosion proofing. This paragraph shall include the system requirements, if any, for nuclear components, including, as applicable, requirements for component design, prevention of inadvertent detonation, and compliance with nuclear safety rules.

8.1.5.6 Security and privacy requirements

This paragraph shall specify the system requirements, if any, concerned with maintaining security and privacy. The requirements shall include, as applicable, the security/privacy environment in which the system must operate, the type and degree of security or privacy to be provided, the security/privacy risks the system must withstand, required safeguards to reduce those risks, the security/privacy policy that must be met, the security/privacy accountability the system must provide, and the criteria that must be met for security/privacy certification/accreditation.

8.1.5.7 System environment requirements

This paragraph shall specify the requirements, if any, regarding the environment in which the system must operate. Examples for a software system are the computer hardware and operating system on which the software must run. (Additional requirements concerning computer resources are given in the next paragraph). Examples for a hardware-software system include the environmental conditions that the system must withstand during transportation, storage, and operation, such as conditions in the natural environment (wind, rain, temperature, geographic location), the induced environment (motion, shock, noise, electromagnetic radiation), and environments due to enemy action (explosions, radiation).

8.1.5.8 Computer resource requirements

This paragraph shall be divided into the following subparagraphs. Depending upon the nature of the system, the computer resources covered in these subparagraphs may constitute the environment of the system (as for a software system) or components of the system (as for a hardware-software system).

8.1.5.9 Computer hardware requirements

This paragraph shall specify the requirements, if any, regarding computer hardware that must be used by, or incorporated into, the system. The requirements shall include, as applicable, number of each type of equipment, type, size, capacity, and other required characteristics of processors, memory, input/output devices, auxiliary storage, communications/network equipment, and other required equipment.

8.1.5.10 Computer hardware resource utilization requirements

This paragraph shall specify the requirements, if any, on the system's computer hardware resource utilization, such as maximum allowable use of processor capacity, memory capacity, input/output device capacity, auxiliary storage device capacity, and communications/network equipment capacity. The requirements (stated, for example, as percentages of the capacity of each computer hardware resource) shall include the conditions, if any, under which the resource utilization is to be measured.

8.1.5.11 Computer software requirements

This paragraph shall specify the requirements, if any, regarding computer software that must be used by, or incorporated into, the system. Examples include operating systems, database management systems, communications/network software, utility software, input and equipment simulators, test software, and manufacturing software. The correct nomenclature, version, and documentation references of each such software item shall be provided.

8.1.5.12 Computer communications requirements

This paragraph shall specify the additional requirements, if any, concerning the computer communications that must be used by, or incorporated into, the system. Examples include geographic locations to be linked; configuration and network topology; transmission techniques; data transfer rates; gateways; required system use times; type and volume of data to be transmitted/received; time boundaries for transmission/reception/response; peak volumes of data; and diagnostic features.

8.1.5.13 System quality factors

This paragraph shall specify the requirements, if any, pertaining to system quality factors. Examples include quantitative requirements concerning system functionality (the ability to perform all required functions), reliability (the ability to perform with correct, consistent results -- such as mean time between failure for equipment), maintainability (the ability to be easily serviced, repaired, or corrected), availability (the ability to be accessed and operated when needed), flexibility (the ability to be easily adapted to changing requirements), portability of software (the

D4.1b Operating Procedures

ability to be easily modified for a new environment), reusability (the ability to be used in multiple applications), testability (the ability to be easily and thoroughly tested), usability (the ability to be easily learned and used), and other attributes.

8.1.5.14 Design and construction constraints

This paragraph shall specify the requirements, if any, that constrain the design and construction of the system. For hardware-software systems, this paragraph shall include the physical requirements imposed on the system. These requirements may be specified by reference to appropriate commercial or military standards and specifications.

Examples include requirements concerning:

- a. Use of a particular system architecture or requirements on the architecture, such as required subsystems; use of standard, military, or existing components; or use of Government/acquirer-furnished property (equipment, information, or software)
- b. Use of particular design or construction standards; use of particular data standards; use of a particular programming language; workmanship requirements and production techniques
- c. Physical characteristics of the system (such as weight limits, dimensional limits, color, protective coatings); interchangeability of parts; ability to be transported from one location to another; ability to be carried or set up by one, or a given number of, persons
- d. Materials that can and cannot be used; requirements on the handling of toxic materials; limits on the electromagnetic radiation that the system is permitted to generate
- e. Use of nameplates, part marking, serial and lot number marking, and other identifying markings
- f. Flexibility and expandability that must be provided to support anticipated areas of growth or changes in technology, threat, or mission

8.1.5.15 Personnel-related requirements

This paragraph shall specify the system requirements, if any, included to accommodate the number, skill levels, duty cycles, training needs, or other information about the personnel who will use or support the system. Examples include requirements for the number of work stations to be provided and for built-in help and training features. Also included shall be the human factors engineering requirements, if any, imposed on the system. These requirements shall include, as applicable, considerations for the capabilities and limitations of humans, foreseeable human errors under both normal and extreme conditions, and specific areas where the effects of human error would be particularly serious. Examples include requirements for adjustable-height work stations, color and duration of error messages, physical placement of critical indicators or buttons, and use of auditory signals.

8.1.5.16 Training-related requirements

This paragraph shall specify the system requirements, if any, pertaining to training. Examples include training devices and training materials to be included in the system.

8.1.5.17 Logistics-related requirements

This paragraph shall specify the system requirements, if any, concerned with logistics considerations. These considerations may include: system maintenance, software support, system transportation modes, supply-system requirements, impact on existing facilities, and impact on existing equipment.

8.1.5.18 Other requirements

This paragraph shall specify additional system requirements, if any, not covered in the previous paragraphs. Examples include requirements for system documentation, such as specifications, drawings, technical manuals, test plans and procedures, and installation instruction data, if not covered in other contractual documents.

D4.1b Operating Procedures

8.1.5.19 Packaging requirements

This section shall specify the requirements, if any, for packaging, labeling, and handling the system and its components for delivery. Applicable military specifications and standards may be referenced if appropriate.

8.1.5.20 Precedence and criticality of requirements

This paragraph shall specify, if applicable, the order of precedence, criticality, or assigned weights indicating the relative importance of the requirements in this specification. Examples include identifying those requirements deemed critical to safety, to security, or to privacy for purposes of singling them out for special treatment.

If all requirements have equal weight, this paragraph shall so state.

8.1.6 Qualification provisions

This section shall define a set of qualification methods and shall specify for each requirement in Section 3 the method(s) to be used to ensure that the requirement has been met. A table may be used to present this information, or each requirement in section 8.1.3 may be annotated with the method(s) to be used. Qualification methods may include:

- a. Demonstration: The operation of the system, or a part of the system, that relies on observable functional operation not requiring the use of instrumentation, special test equipment, or subsequent analysis.
- b. Test: The operation of the system, or a part of the system, using instrumentation or other special test equipment to collect data for later analysis.
- c. Analysis: The processing of accumulated data obtained from other qualification methods.

Examples are reduction, interpolation, or extrapolation of test results.

- d. Inspection: The visual examination of system components, documentation, etc.
- e. Special qualification methods. Any special qualification methods for the system, such as special tools, techniques, procedures, facilities, acceptance limits, use of standard samples, preproduction or periodic production samples, pilot models, or pilot lots.

8.1.7 Requirements traceability

For system-level specifications, this paragraph does not apply.

For subsystem-level specifications, this paragraph shall contain:

- a. Traceability from each subsystem requirement in this specification to the system requirements it addresses. (Alternatively, this traceability may be provided by annotating each requirement in Section 3.)

Note: Each level of system refinement may result in requirements not directly traceable to higher-level requirements. For example, a system architectural design that creates two subsystems may result in requirements about how the subsystems will interface, even though these interfaces are not covered in system requirements. Such requirements may be traced to a general requirement such as "system implementation" or to the system design decisions that resulted in their generation.

- b. Traceability from each system requirement that has been allocated to the subsystem covered by this specification to the subsystem requirements that address it. All system requirements allocated to the subsystem shall be accounted for. Those that trace to subsystem requirements contained in Interface Requirements Specifications (IRs) shall reference those IRs.

D4.1b Operating Procedures

8.1.8 Notes

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall contain an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

8.2 DBDD Template

8.2.1 Scope

8.2.1.1 Identification

This paragraph shall contain a full identification of the database to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

After tailoring, this might be a part of the Software Design Description Document.

8.2.1.2 System overview

This paragraph shall briefly state the purpose of the system and the Software to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

8.2.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.2.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.2.3 Database-wide design decisions

This section shall be divided into paragraphs as needed to present database-wide design decisions, that is, decisions about the database's behavioral design (how it will behave, from a user's point of view, in meeting its requirements, ignoring internal implementation) and other decisions affecting further design of the database. If all such decisions are explicit in the system or CSCI requirements, this section shall so state. Design decisions that respond to requirements designated critical, such as those for safety, security, or privacy, shall be placed in separate subparagraphs. If a design decision depends upon system states or modes, this dependency shall be indicated. If some or all of the design decisions are described in the documentation of a custom or commercial database management system (DBMS), they may be referenced from this section. Design conventions needed to understand the design shall be presented or referenced. Examples of database-wide design decisions are the following:

a. Design decisions regarding queries or other inputs the database will accept and outputs (displays, reports, messages, responses, etc.) it will produce, including interfaces with other systems, HWCIs, CSCIs, and users (5.x.d of this DID identifies topics to be considered in this description). If part or all of this information is given in Interface Design Descriptions (IDDs), they may be referenced.

b. Design decisions on database behavior in response to each input or query, including actions, response times and other performance characteristics, selected equations/algorithms/rules, disposition, and handling of unallowed inputs

D4.1b Operating Procedures

- c. Design decisions on how databases/data files will appear to the user (4.x of this DID identifies topics to be considered in this description)
- d. Design decisions on the database management system to be used (including name, version/release) and the type of flexibility to be built into the database for adapting to changing requirements
- e. Design decisions on the levels and types of availability, security, privacy, and continuity of operations to be offered by the database
- f. Design decisions on database distribution (such as client/server), master database file updates and maintenance, including maintaining consistency, establishing/reestablishing and maintaining synchronization, enforcing integrity and business rules
- g. Design decisions on backup and restoration including data and process distribution strategies, permissible actions during backup and restoration, and special considerations for new or non-standard technologies such as video and sound
- h. Design decisions on repacking, sorting, indexing, synchronization, and consistency including automated disk management and space reclamation considerations, optimizing strategies and considerations, storage and size considerations, and population of the database and capture of legacy data

8.2.4 Detailed design of the database.

This section shall be divided into paragraphs as needed to describe the detailed design of the database. The number of levels of design and the names of those levels shall be based on the design methodology used. Examples of database design levels include conceptual, internal, logical, and physical. If part or all of the design depends upon system states or modes, this dependency shall be indicated. Design conventions needed to understand the design shall be presented or referenced.

Note: This DID uses the term "data element assembly" to mean any entity, relation, schema, field, table, array, etc., that has structure (number/order/grouping of data elements) at a given design level (e.g., conceptual, internal, logical, physical) and the term "data element" to mean any relation, attribute, field, cell, data element, etc. that does not have structure at that level.

8.2.4.1 (Name of database design level).

This paragraph (starting at 4.1 and expanding till 4.x as much as needed) shall identify a database design level and shall describe the data elements and data element assemblies of the database in the terminology of the selected design method. The information shall include the following, as applicable, presented in any order suited to the information to be provided:

- a. Characteristics of individual data elements in the database design, such as:

- 1) Names/identifiers

- a) Project-unique identifier
- b) Non-technical (natural-language) name
- c) DoD standard data element name
- d) Technical name (e.g., field name in the database)
- e) Abbreviation or synonymous names

- 2) Data type (alphanumeric, integer, etc.)

D4.1b Operating Procedures

- 3) Size and format (such as length and punctuation of a character string)
- 4) Units of measurement (such as meters, dollars, nanoseconds)
- 5) Range or enumeration of possible values (such as 0-99)
- 6) Accuracy (how correct) and precision (number of significant digits)
- 7) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the data element may be updated and whether business rules apply
- 8) Security and privacy constraints
- 9) Sources (setting/sending entities) and recipients (using/receiving entities)

b. Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) in the database design, such as:

- 1) Names/identifiers
 - a) Project-unique identifier
 - b) Non-technical (natural language) name
 - c) Technical name (e.g., record or data structure name in code or database)
 - d) Abbreviations or synonymous names
- 2) Data elements in the assembly and their structure (number, order, grouping)
- 3) Medium (such as disk) and structure of data elements/assemblies on the medium
- 4) Visual and auditory characteristics of displays and other outputs (such as colors, layouts, fonts, icons and other display elements, beeps, lights)
- 5) Relationships among assemblies, such as sorting/access characteristics
- 6) Priority, timing, frequency, volume, sequencing, and other constraints, such as whether the assembly may be updated and whether business rules apply
- 7) Security and privacy constraints
- 8) Sources (setting/sending entities) and recipients (using/receiving entities)

8.2.5 Detailed design of software units used for database access or manipulation

This section shall be divided into the following paragraphs to describe each software unit used for database access or manipulation. If part or all of this information is provided elsewhere, such as in a Software Design Description (SDD), the SDD for a customized DBMS, or the user manual of a commercial DBMS, that information may be referenced rather than repeated here. If part or all of the design depends upon system states or modes, this dependency shall be indicated. If design information falls into more than one paragraph, it may be presented once and referenced from the other paragraphs. Design conventions needed to understand the design shall be presented or referenced.

8.2.5.1 (Project-unique identifier of a software unit, or designator for a group of software units)

This paragraph (starting at 5.1 and expanding till 5.x as much as needed) shall identify a software unit by project-unique identifier and shall describe the unit. The description shall include the following information, as applicable. Alternatively, this paragraph may designate a group of software units and identify and describe the software units in subparagraphs. Software units that contain other software units may reference the descriptions of those units rather than repeating information.

D4.1b Operating Procedures

- a. Unit design decisions, if any, such as algorithms to be used, if not previously selected
- b. Any constraints, limitations, or unusual features in the design of the software unit
- c. The programming language to be used and rationale for its use if other than the specified CSCI language

d. If the software unit consists of or contains procedural commands (such as menu selections in a database management system (DBMS) for defining forms and reports, online DBMS queries for database access and manipulation, input to a graphical user interface (GUI) builder for automated code generation, commands to the operating system, or shell scripts), a list of the procedural commands and reference to user manuals or other documents that explain them

e. If the software unit contains, receives, or outputs data, a description of its inputs, outputs, and other data elements and data element assemblies, as applicable. Data local to the software unit shall be described separately from data input to or output from the software unit. Interface characteristics may be provided here or by referencing Interface Design Descriptions (IDDs). If a given interfacing entity is not covered by this DBDD (for example, an external system) but its interface characteristics need to be mentioned to describe software units that are, these characteristics shall be stated as assumptions or as "When [the entity not covered] does this, [the software unit] will...." This paragraph may reference other documents (such as data dictionaries, standards for protocols, and standards for user interfaces) in place of stating the information here. The design description shall include the following, as applicable, presented in any order suited to the information to be provided, and shall note any differences in these characteristics from the point of view of the interfacing entities (such as different expectations about the size, frequency, or other characteristics of data elements):

- 1) Project-unique identifier for the interface
- 2) Identification of the interfacing entities (software units, configuration items, users, etc.) by name, number, version, and documentation references, as applicable
- 3) Priority assigned to the interface by the interfacing entity(ies)
- 4) Type of interface (such as real-time data transfer, storage-and-retrieval of data, etc.) to be implemented
- 5) Characteristics of individual data elements that the interfacing entity(ies) will provide, store, send, access, receive, etc. Paragraph 4.x.a of this DID identifies topics to be covered in this description.
- 6) Characteristics of data element assemblies (records, messages, files, arrays, displays, reports, etc.) that the interfacing entity(ies) will provide, store, send, access, receive, etc. Paragraph 4.x.b of this DID identifies topics to be covered in this description.
- 7) Characteristics of communication methods that the interfacing entity(ies) will use for the interface, such as:
 - a) Project-unique identifier(s)
 - b) Communication links/bands/frequencies/media and their characteristics
 - c) Message formatting
 - d) Flow control (such as sequence numbering and buffer allocation)
 - e) Data transfer rate, whether periodic/aperiodic, and interval between transfers
 - f) Routing, addressing, and naming conventions
 - g) Transmission services, including priority and grade
 - h) Safety/security/privacy considerations, such as encryption, user authentication, compartmentalization, and auditing
- 8) Characteristics of protocols the interfacing entity(ies) will use for the interface, such as:
 - a) Project-unique identifier(s)
 - b) Priority/layer of the protocol
 - c) Packeting, including fragmentation and reassembly, routing, and addressing
 - d) Legality checks, error control, and recovery procedures
 - e) Synchronization, including connection establishment, maintenance, termination

D4.1b Operating Procedures

f) Status, identification, and any other reporting features

9) Other characteristics, such as physical compatibility of the interfacing entity(ies) (dimensions, tolerances, loads, voltages, plug compatibility, etc.)

f. If the software unit contains logic, the logic to be used by the software unit, including, as applicable:

- 1) Conditions in effect within the software unit when its execution is initiated
- 2) Conditions under which control is passed to other software units
- 3) Response and response time to each input, including data conversion, renaming, and data transfer operations
- 4) Sequence of operations and dynamically controlled sequencing during the software unit's operation, including:
 - a) The method for sequence control
 - b) The logic and input conditions of that method, such as timing variations, priority assignments
 - c) Data transfer in and out of memory
 - d) The sensing of discrete input signals, and timing relationships between interrupt operations within the software unit
- 5) Exception and error handling

8.2.6 Requirements traceability

This section shall contain:

- a. Traceability from each database or other software unit covered by this DBDD to the system or CSCI requirements it addresses.
- b. Traceability from each system or CSCI requirement that has been allocated to a database or other software unit covered in this DBDD to the database or other software units that address it.

8.2.7 Notes.

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

8.3 SIP Template

8.3.1 Scope

8.3.1.1 Identification

This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

8.3.1.2 System overview

This paragraph shall briefly state the purpose of the system and the Software to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

D4.1b Operating Procedures

8.3.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.3.1.4 Relationship to other plans.

This paragraph shall describe the relationship, if any, of the SIP to other project management plans.

8.3.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.3.3 Installation Overview

This section shall be divided into the following paragraphs to provide an overview of the installation process.

8.3.3.1 Description.

This paragraph shall provide a general description of the installation process to provide a frame of reference for the remainder of the document. A list of sites for software installation, the schedule dates, and the method of installation shall be included.

8.3.3.2 Contact point.

This paragraph shall provide the organizational name, office symbol/code, and telephone number of a point of contact for questions relating to this installation.

8.3.3.3 Support materials.

This paragraph shall list the type, source, and quantity of support materials needed for the installation. Included shall be items such as magnetic tapes, disk packs, computer printer paper, and special forms.

8.3.3.4 Training.

This paragraph shall describe the developer's plans for training personnel who will operate and/or use the software installed at user sites. Included shall be the delineation between general orientation, classroom training, and "hands-on" training.

8.3.3.5 Tasks.

This paragraph shall list and describe in general terms each task involved in the software installation. Each task description shall identify the organization that will accomplish the task, usually either the user, computer operations, or the developer. The task list shall include such items as:

- a. Providing overall planning, coordination, and preparation for installation
- b. Providing personnel for the installation team

D4.1b Operating Procedures

- c. Arranging lodging, transportation, and office facilities for the installation team
- d. Ensuring that all manuals applicable to the installation are available when needed
- e. Ensuring that all other prerequisites have been fulfilled prior to the installation
- f. Planning and conducting training activities
- g. Providing students for the training
- h. Providing computer support and technical assistance for the installation
- i. Providing for conversion from the current system

8.3.3.6 Personnel.

This paragraph shall describe the number, type, and skill level of the personnel needed during the installation period, including the need for multishift operation, clerical support, etc.

8.3.3.7 Security and privacy.

This paragraph shall contain an overview of the security and privacy considerations associated with the system.

8.3.4 Site-specific information for software center operations staff.

This section applies if the software will be installed in computer center(s) or other centralized or networked software installations for users to access via terminals or using batch inputs/outputs. If this type of installation does not apply, this section shall contain the words "Not applicable."

For each site name, a new subsection needs to be created including all the sub-subsections identified below from 4.1.1 till 4.1.6. General lay-out shall consist of 4.x.1 till 4.x.6.

8.3.4.1 (Site name).

This paragraph shall identify a site or set of sites and shall be divided into the following subparagraphs to discuss those sites. Multiple sites may be discussed together when the information for those sites is generally the same.

8.3.4.2 Schedule.

This paragraph shall present a schedule of tasks to be accomplished during installation. It shall depict the tasks in chronological order with beginning and ending dates of each task and supporting narrative as necessary.

8.3.4.3 Software inventory.

This paragraph shall provide an inventory of the software needed to support the installation. The software shall be identified by name, identification number, version number, release number, configuration, and security classification, as applicable. This paragraph shall indicate whether the software is expected to be on site or will be delivered for the installation and shall identify any software to be used only to facilitate the installation process.

D4.1b Operating Procedures

8.3.4.4 Facilities.

This paragraph shall detail the physical facilities and accommodations needed during the installation period. This description shall include the following, as applicable: a. Classroom, work space, and training aids needed, specifying hours per day, number of days, and shifts

- b. Hardware that must be operational and available
- c. Transportation and lodging for the installation team

8.3.4.5 Installation team.

This paragraph shall describe the composition of the installation team. Each team member's tasks shall be defined.

8.3.4.6 Installation procedures.

This paragraph shall provide step-by-step procedures for accomplishing the installation. References may be made to other documents, such as operator manuals. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable. The procedures shall include the following, as applicable:

- a. Installing the software
- b. Checking out the software once installed
- c. Initializing databases and other software with site-specific data
- d. Conversion from the current system, possibly involving running in parallel
- e. Dry run of the procedures in operator and user manuals

8.3.4.7 Data update procedures.

This paragraph shall present the data update procedures to be followed during the installation period. When the data update procedures are the same as normal updating or processing procedures, reference may be made to other documents, such as operator manuals.

8.3.5 Site-specific information for software users.

This section shall provide installation planning pertinent to users of the software. When more than one type of user is involved, for example, users at different positions, performing different functions, or in different organizations, a separate section (Sections 5 through n) may be written for each type of user and the section titles modified to reflect each user.

For each site name, a new subsection needs to be created including all the sub-subsections identified below from 5.1.1 till 5.1.3. General lay-out shall consist of 5.x.1 till 5.x.3.

8.3.5.1 5.x (Site name).

This paragraph shall identify a site or set of sites and shall be divided into the following subparagraphs to discuss those sites. Multiple sites may be discussed together when the information for those sites is generally the same.

D4.1b Operating Procedures

8.3.5.2 Schedule.

This paragraph shall present a schedule of tasks to be accomplished by the user during installation. It shall depict the tasks in chronological order including beginning and ending dates for each task and supporting narrative as necessary.

8.3.5.3 Installation procedures.

This paragraph shall provide step-by-step procedures for accomplishing the installation. Reference may be made to other documents, such as user manuals. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable. The procedures shall include the following, as applicable:

- a. Performing the tasks under 4.x.5 if not performed by operations staff
- b. Initializing user-specific data
- c. Setting up queries and other user inputs
- d. Performing sample processing
- e. Generating sample reports
- f. Conversion from the current system, possibly involving running in parallel
- g. Dry run of procedures in user manuals

8.3.5.4 Data update procedures.

This paragraph shall be divided into subparagraphs to present the user's data update procedures to be followed during the installation period. When update procedures are the same as normal processing, reference may be made to other documents, such as user manuals, and to Section 4 of this document

8.3.6 Notes.

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

8.4 STR Template

8.4.1 Scope

8.4.1.1 Identification

This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

D4.1b Operating Procedures

8.4.1.2 System overview

This paragraph shall briefly state the purpose of the system and the Software to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

8.4.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.4.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.4.3 Overview of test results

This section shall be divided into the following paragraphs to provide an overview of test results.

8.4.3.1 Overall assessment of the software tested.

This paragraph shall:

- a. Provide an overall assessment of the software as demonstrated by the test results in this report
- b. Identify any remaining deficiencies, limitations, or constraints that were detected by the testing performed. Problem/change reports may be used to provide deficiency information.
- c. For each remaining deficiency, limitation, or constraint, describe:
 - 1) Its impact on software and system performance, including identification of requirements not met
 - 2) The impact on software and system design to correct it
 - 3) A recommended solution/approach for correcting it

8.4.3.2 Impact of test environment.

This paragraph shall provide an assessment of the manner in which the test environment may be different from the operational environment and the effect of this difference on the test results.

8.4.3.3 Recommended improvements.

This paragraph shall provide any recommended improvements in the design, operation, or testing of the software tested. A discussion of each recommendation and its impact on the software may be provided. If no recommended improvements are provided, this paragraph shall state "None."

8.4.4 Detailed test results.

This section shall be divided into the following paragraphs to describe the detailed results for each test. Note: The word "test" means a related collection of test cases.

D4.1b Operating Procedures

This section shall be divided in the number of subsections required to cover each test identifier. Each subsection shall be constructed as provided below.

8.4.4.1 (Project-unique identifier of a test).

This paragraph shall identify a test by project-unique identifier and shall be divided into the following subparagraphs to describe the test results.

This subsection shall be expanded as much as required to cover all the tests results as per below. Each subsection shall contain the 3 sub-subsections as identified below, e.g. 4.1.1 till 4.1.3.

8.4.4.2 Summary of test results.

This paragraph shall summarize the results of the test. The summary shall include, possibly in a table, the completion status of each test case associated with the test (for example, "all results as expected," "problems encountered," "deviations required"). When the completion status is not "as expected," this paragraph shall reference the following paragraphs for details.

8.4.4.3 Problems encountered.

This paragraph shall be divided into subparagraphs that identify each test case in which one or more problems occurred.

This sub-subsection shall include as much paragraphs as required, e.g. 4.1.2.1, 4.1.1.2, till 4.1.2.x...

(Project-unique identifier of a test case).

This paragraph shall identify by projectunique identifier a test case in which one or more problems occurred, and shall provide:

- a. A brief description of the problem(s) that occurred
- b. Identification of the test procedure step(s) in which they occurred
- c. Reference(s) to the associated problem/change report(s) and backup data, as applicable
- d. The number of times the procedure or step was repeated in attempting to correct the problem(s) and the outcome of each attempt
- e. Back-up points or test steps where tests were resumed for retesting

8.4.4.4 Deviations from test cases/procedures.

This paragraph shall be divided into subparagraphs that identify each test case in which deviations from test case/test procedures occurred.

This sub-subsection shall include as much paragraphs as required, e.g. 4.1.3.1, 4.1.3.2, till 4.1.3.x...

(Project-unique identifier of a test case).

This paragraph shall identify by projectunique identifier a test case in which one or more deviations occurred, and shall provide:

- a. A description of the deviation(s) (for example, test case run in which the deviation occurred and nature of the deviation, such as substitution of required equipment, procedural steps not followed, schedule deviations). (Red-lined test procedures may be used to show the deviations)
- b. The rationale for the deviation(s)
- c. An assessment of the deviations' impact on the validity of the test case

8.4.5 Test log.

This section shall present, possibly in a figure or appendix, a chronological record of the test events covered by this report. This test log shall include:

- a. The date(s), time(s), and location(s) of the tests performed
- b. The hardware and software configurations used for each test including, as applicable, part/model/serial number, manufacturer, revision level, and calibration date of all hardware, and version number and name for the software components used
- c. The date and time of each test-related activity, the identity of the individual(s) who performed the activity, and the identities of witnesses, as applicable

8.4.6 Notes.

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

8.5 SUM Template

8.5.1 Scope

8.5.1.1 Identification

This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s).

8.5.1.2 System overview

This paragraph shall briefly state the purpose of the system and the Software to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

8.5.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.5.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.5.3 Software summary

This section shall be divided into the following paragraphs.

8.5.3.1 Software application.

This paragraph shall provide a brief description of the intended uses

D4.1b Operating Procedures

of the software. Capabilities, operating improvements, and benefits expected from its use shall be described.

8.5.3.2 Software inventory.

This paragraph shall identify all software files, including databases and data files, that must be installed for the software to operate. The identification shall include security and privacy considerations for each file and identification of the software necessary to continue or resume operation in case of an emergency.

8.5.3.3 Software environment.

This paragraph shall identify the hardware, software, manual operations, and other resources needed for a user to install and run the software. Included, as applicable, shall be identification of:

- a. Computer equipment that must be present, including amount of memory needed, amount of auxiliary storage needed, and peripheral equipment such as printers and other input/output devices
- b. Communications equipment that must be present
- c. Other software that must be present, such as operating systems, databases, data files, utilities, and other supporting systems
- d. Forms, procedures, or other manual operations that must be present
- e. Other facilities, equipment, or resources that must be present

8.5.3.4 Software organization and overview of operation.

This paragraph shall provide a brief description of the organization and operation of the software from the user's point of view. The description shall include, as applicable:

- a. Logical components of the software, from the user's point of view, and an overview of the purpose/operation of each component
- b. Performance characteristics that can be expected by the user, such as:
 - 1) Types, volumes, rate of inputs accepted
 - 2) Types, volume, accuracy, rate of outputs that the software can produce
 - 3) Typical response time and factors that affect it

D4.1b Operating Procedures

- 4) Typical processing time and factors that affect it
- 5) Limitations, such as number of events that can be tracked
- 6) Error rate that can be expected
- 7) Reliability that can be expected

c. Relationship of the functions performed by the software with interfacing systems, organizations, or positions

d. Supervisory controls that can be implemented (such as passwords) to manage the software

8.5.3.5 Contingencies and alternate states and modes of operation.

This paragraph shall explain the differences in what the user will be able to do with the software at times of emergency and in various states and modes of operation, if applicable.

8.5.3.6 Security and privacy.

This paragraph shall contain an overview of the security and privacy considerations associated with the software. A warning shall be included regarding making unauthorized copies of software or documents, if applicable.

8.5.3.7 Assistance and problem reporting.

This paragraph shall identify points of contact and procedures to be followed to obtain assistance and report problems encountered in using the software.

8.5.4 Access to the software.

This section shall contain step-by-step procedures oriented to the first time/occasional user. Enough detail shall be presented so that the user can reliably access the software before learning the details of its functional capabilities. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable.

8.5.4.1 First-time user of the software.

This paragraph shall be divided into the following subparagraphs.

8.5.4.2 Equipment familiarization.

This paragraph shall describe the following as appropriate:

- a. Procedures for turning on power and making adjustments
- b. Dimensions and capabilities of the visual display screen
- c. Appearance of the cursor, how to identify an active cursor if more than one cursor can

D4.1b Operating Procedures

appear, how to position a cursor, and how to use a cursor

d. Keyboard layout and role of different types of keys and pointing devices

e. Procedures for turning power off if special sequencing of operations is needed

8.5.4.3 Access control.

This paragraph shall present an overview of the access and security features of the software that are visible to the user. The following items shall be included, as applicable:

a. How and from whom to obtain a password

b. How to add, delete, or change passwords under user control

c. Security and privacy considerations pertaining to the storage and marking of output reports and other media that the user will generate

8.5.4.4 Installation and setup.

This paragraph shall describe any procedures that the user must perform to be identified or authorized to access or install software on the equipment, to perform the installation, to configure the software, to delete or overwrite former files or data, and to enter parameters for software operation.

8.5.4.5 Initiating a session.

This paragraph shall provide step-by-step procedures for beginning work, including any options available. A checklist for problem determination shall be included in case difficulties are encountered.

8.5.4.6 Stopping and suspending work.

This paragraph shall describe how the user can cease or interrupt use of the software and how to determine whether normal termination or cessation has occurred.

8.5.5 Processing reference guide.

This section shall provide the user with procedures for using the software. If procedures are complicated or extensive, additional Sections 6, 7, ... may be added in the same paragraph structure as this section and with titles meaningful to the sections selected.

The organization of the document will depend on the characteristics of the software being documented. For example, one approach is to base the sections on the organizations in which users work, their assigned positions, their work sites, or the tasks they must perform. For other software, it may be more appropriate to have Section 5 be a guide to menus, Section 6 be a guide to the command language used, and Section 7 be a guide to functions. Detailed procedures are intended to be presented in subparagraphs of paragraph 5.3. Depending on the design of the software, the subparagraphs might be organized on a function-by-function, menu-by-menu, transaction-by-transaction, or other basis. Safety precautions, marked by WARNING or CAUTION, shall be included where applicable.

8.5.5.1 Capabilities.

This paragraph shall briefly describe the interrelationships of the transactions, menus, functions, or other processes in order to provide an overview of the use of the software.

D4.1b Operating Procedures

8.5.5.2 Conventions.

This paragraph shall describe any conventions used by the software, such as the use of colors in displays, the use of audible alarms, the use of abbreviated vocabulary, and the use of rules for assigning names or codes.

8.5.5.3 Processing procedures.

This paragraph shall explain the organization of subsequent paragraphs, e.g., by function, by menu, by screen. Any necessary order in which procedures must be accomplished shall be described.

8.5.5.4 (Aspect of software use).

The title of this paragraph shall identify the function, menu, transaction, or other process being described. This paragraph shall describe and give options and examples, as applicable, of menus, graphical icons, data entry forms, user inputs, inputs from other software or hardware that may affect the software's interface with the user, outputs, diagnostic or error messages or alarms, and help facilities that can provide on-line descriptive or tutorial information. The format for presenting this information can be adapted to the particular characteristics of the software, but a consistent style of presentation shall be used, i.e., the descriptions of menus shall be consistent, the descriptions of transactions shall be consistent among themselves.

8.5.5.5 (Aspect of software use).

Section 5.3 shall be expanded as much as required to cover all the aspects of software use.

8.5.5.6 Related processing.

This paragraph shall identify and describe any related batch, offline, or background processing performed by the software that is not invoked directly by the user and is not described in paragraph 5.3. Any user responsibilities to support this processing shall be specified.

8.5.5.7 Data backup.

This paragraph shall describe procedures for creating and retaining backup

data that can be used to replace primary copies of data in event of errors, defects, malfunctions, or accidents.

8.5.5.8 Recovery from errors, malfunctions, and emergencies.

This paragraph shall present detailed procedures for restart or recovery from errors or malfunctions occurring during processing and for ensuring continuity of operations in the event of emergencies.

8.5.5.9 Messages.

This paragraph shall list, or refer to an appendix that lists, all error messages, diagnostic messages, and information messages that can occur while accomplishing any of the user's functions. The meaning of each message and the action that should be taken after each such message shall be identified and described.

8.5.5.10 Quick-reference guide.

If appropriate to the software, this paragraph shall provide or reference a quick-reference card or page for using the software. This quick-reference guide shall summarize, as applicable, frequently-used function keys, control sequences, formats, commands, or other aspects of software use.

8.5.6 Notes.

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

8.6 SVD Template

8.6.1 Scope

8.6.1.1 Identification

This paragraph shall contain a full identification of the system and the software to which this document applies, including, as applicable, identification number(s), title(s), abbreviation(s), version number(s), and release number(s). It shall also identify the intended recipients of the SVD to the extent that this identification affects the contents of the software released (for example, source code may not be released to all recipients.)

8.6.1.2 System overview

This paragraph shall briefly state the purpose of the system and the Software to which this document applies. It shall describe the general nature of the system; summarize the history of system development, operation, and maintenance; identify the project sponsor, acquirer, user, developer, and support agencies; identify current and planned operating sites; and list other relevant documents.

8.6.1.3 Document overview

This paragraph shall summarize the purpose and contents of this document and shall describe any security or privacy considerations associated with its use.

8.6.2 Referenced documents

This section shall list the number, title, revision, and date of all documents referenced in this document. This section shall also identify the source for all documents not available through normal Government stocking activities.

8.6.3 Version Description

This section shall be divide into the following paragraphs.

8.6.3.1 Inventory of materials released.

This paragraph shall list by identifying numbers, titles, abbreviations, dates, version numbers, and release numbers, as applicable, all physical media (for example, listings, tapes, disks) and associated documentation that make up the software version being released. It shall include applicable security and privacy considerations for these items, safeguards for handling them, such as concerns for static and magnetic fields, and instructions and restrictions regarding duplication and license provisions.

D4.1b Operating Procedures

8.6.3.2 Inventory of software contents.

This paragraph shall list by identifying numbers, titles, abbreviations, dates, version numbers, and release numbers, as applicable, all computer files that make up the software version being released. Any applicable security and privacy considerations shall be included.

8.6.3.3 Changes installed.

This paragraph shall contain a list of all changes incorporated into the software version since the previous version. If change classes have been used, such as the Class I/Class II changes in MIL-STD-973, the changes shall be separated into these classes.

This paragraph shall identify, as applicable, the problem reports, change proposals, and change notices associated with each change and the effects, if any, of each change on system operation and on interfaces with other hardware and software. This paragraph does not apply to the initial software version.

8.6.3.4 Adaptation data.

This paragraph shall identify or reference all unique-to-site data contained in the software version. For software versions after the first, this paragraph shall describe changes made to the adaptation data.

8.6.3.5 Related documents.

This paragraph shall list by identifying numbers, titles, abbreviations, dates, version numbers, and release numbers, as applicable, all documents pertinent to the software version being released but not included in the release.

8.6.3.6 Installation instructions.

This paragraph shall provide or reference the following information, as applicable:

- a. Instructions for installing the software version
- b. Identification of other changes that have to be installed for this version to be used, including site-unique adaptation data not included in the software version
- c. Security, privacy, or safety precautions relevant to the installation
- d. Procedures for determining whether the version has been installed properly
- e. A point of contact to be consulted if there are problems or questions with the installation

8.6.3.7 Possible problems and known errors.

This paragraph shall identify any possible problems or known errors with the software version at the time of release, any steps being taken to resolve the problems or errors, and instructions (either directly or by reference) for recognizing, avoiding, correcting, or otherwise handling each one. The information presented shall be appropriate to the intended recipient of the SVD (for example, a user agency may need advice on avoiding errors, a support agency on correcting them).

8.6.4 Notes.

This section shall contain any general information that aids in understanding this document (e.g., background information, glossary, rationale). This section shall include an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.