RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Integrated Solutions for Positive Energy and Resilient Cities

D9.6

RESPONSE scalability and replicability evaluation toolkit - V1

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement nº 957751. The document represents the view of the author only and is his/her sole responsibility: it cannot be considered to reflect the views of the European Commission and/or the European Climate, Infrastructure and Environment Executive Agency (CINEA). The European Commission and the Agency do not accept responsibility for the use that may be made of the information it contains.

Document Control Sheet

Project Title	integRatEd Solutions for POsitive eNergy and resilient CitiEs - RESPONSE
Deliverable	D9.6. RESPONSE scalability and replicability evaluation toolkit - V1
Work package	WP9 Impact Measurement, Analysis and Benchmarking
Task	T9.6 Development of a toolkit for the evaluation of scalability and replicability of the RESPONSE City Information Platform
Number of pages	68
Dissemination level	PU
Main authors	Néstor Rodríguez Pérez (COMILLAS), Javier Matanza Domingo (COMILLAS), Gregorio López López (COMILLAS), Carlos Rodríguez Morcillo (COMILLAS)
Contributors	Mitra Hajigholi (eGAIN), Paola Astegiano (RINA-C), Federica Fuligni (RINA-C), Alessandro Venturini (RINA-C), Michele Valery (RINA-C)

Reviewers

Partner	Name	Contact information
eGAIN	Mitra Hajigholi	mitra.hajigholi@egain.io
EIFER	Monjur Murshed David Goujon	monjur.murshed@eifer.org david.goujon@eifer.org

Dissemination level codes

PU = Public, fully open, e.g., web

CO =Confidential, restricted under conditions set out in Model Grant Agreement

CI =Classified, information as referred to in Commission Decision 2001/844/EC.

Versioning and Contribution History

Version	Date	Author/Editor	Contributors	Description/Comments
V0	16 / 12 /2021	Néstor Rodríguez, Javier Matanza, Gregorio López, Carlos Rodríguez Morcillo (COMILLAS)	-	Table of content
V0.1	17 / 02 /2022	Néstor Rodríguez, Javier Matanza, Gregorio López, Carlos Rodríguez Morcillo (COMILLAS)	-	Complete draft of the deliverable.
V0.2	21 / 02 / 2022	-	Paola Astegiano, Federica Fuligni, Alessandro Venturini, Michele Valery (RINA-C)	Added comments.
V0.3	17 / 03 / 2022		Monjur Murshed (EIFER), David Goujon (EIFER), Mitra Hajigholi (eGAIN)	Review deliverable.
V0.4	25 / 03 / 2022	Néstor Rodríguez, Javier Matanza, Gregorio López, Carlos Rodríguez Morcillo (COMILLAS)	-	Modifications based on reviewers' comments
V1.0	29 / 03 / 2022	Néstor Rodríguez (COMILLAS)	-	Final version

Index

Index of Tables

Index of Figures

Glossary

Abbreviation	Full form
ADU	Application Data Unit
AI	Artificial Intelligence
AMI	Advanced Metering Infrastructure
API	Application Program Interface
BER	Bit Error Rate
BESS	Battery Energy Storage System
BMS	Battery Management System
CSV	Comma-Separated Values
DER	Distributed Energy Resources
EDF	Électricité de France
EMS	Energy Management System
EMPAIR	Equipement Modulaire de Protection des Accès Industriels Répartis
EV	Electric Vehicle
FC	Fellow City
GA	Grant Agreement
GPRS	General Packet Radio Service
GUI	Graphical User Interface
ICT	Information and Communication Technologies
IE	Innovative Element
IEEE	Institute of Electrical and Electronic Engineers
IP	Internet Protocol
IS	Integrated Solution
LAN	Local Area Network
LH city	Lighthouse city
MBAP	ModBus Application Protocol
OCCP	Open Charge Point Protocol
OSI	Open Systems Interconnection
OMS	Open Metering System

PDU	Protocol Data Unit
PEB	Positive Energy Block
PED	Positive Energy District
PLC	Programmable Logic Controller
PLC (in communications)	Power Line Communications
PV	(solar) Photovoltaics
RES	Renewable Energy Sources
SGAM	Smart Grid Architecture Model
SRA	Scalability and Replicability Analysis
SOAP	Simple Object Access Protocol
TA	Transformation Axis
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
V2G	Vehicle-to-Grid
VM	Virtual Machine

Executive Summary

The objective of T9.6 is to develop a software toolkit to be used to evaluate the scalability and replicability potential of some Innovative Elements (IEs) that will be implemented in the lighthouse (LH) cities of Dijon and Turku. This Scalability and Replicability Analysis (SRA) will be conducted in Task 9.7.

This deliverable, D9.6, focuses on determining the scope of the Information and Communication Technologies (ICT) scalability and replicability evaluation toolkit, on describing the ICT characteristics and requirements of the selected IEs, and on presenting the preliminary design of the toolkit. Deliverable D9.6 will be complemented by D9.13, which will provide a more complete description of the toolkit, possible SRA scenarios, and validation details.

To determine which IEs, among those where ICT may play a significant role, could be included under the scope of the toolkit, two main criteria were considered: the complexity of developing an accurate simulation model, and how significant the results of such a simulation could be for the quantitative SRA to be carried out in T9.7.

For the LH city of Dijon, the system included under the scope of the toolkit is part of IE 4.1.1 GENESYS tunnelling solution, implemented by EDF. The system to be simulated will use the Modbus/TCP protocol over Ethernet to communicate a control and monitoring device, called EMPAIR, with energy assets installed at a block level, such as Energy Management Systems (EMS), batteries, or solar PV. The simulation of the information exchanges in this system will provide relevant information to determine the operation limits imposed by ICT and how the system could be scaled and replicated while guaranteeing a good functioning.

For the LH city of Turku, the system included under the scope of the toolkit is the indoors condition monitoring system provided by eGAIN in IE 4.1.8. This system consists of wireless temperature and humidity sensors installed at dwellings that send measurements to a data collector installed at a block or building level. In this case, the communications would be done using the wireless M-Bus protocol. The simulation of these communications will provide insights about how a wireless sensing system implemented at dwellings could be leveraged in a Smart City so that more citizens could benefit from these solutions (e.g., to improve energy use while keeping the temperature comfort).

For both LH cities, Dijon and Turku, the overall approach to design the toolkit is the same. In both cases, the toolkit will consist of three main elements, as shown in Figure 1: the simulation model developed using the OMNeT++ simulation framework, a graphical user interface to easily configure and run simulation scenarios, and an analysis tool, consisting of a Jupyter Notebook, to easily generate some relevant charts and indicators.

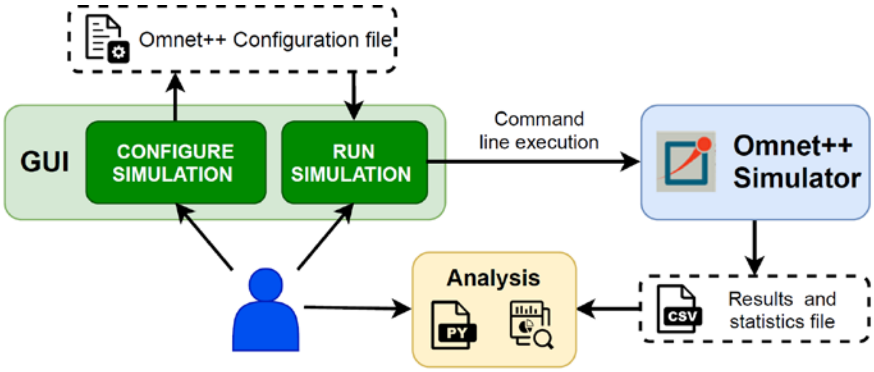

Figure 1 General design of the ICT toolkit to be used in the SRA for the LH cities

Based on this design, the toolkit is expected to be user-friendly, flexible and at the same time, provide a highly functional simulation of the systems under consideration, so that the analysts can conduct a thorough quantitative ICT SRA in the context of T9.7.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Chapter 1

Introduction

Chapter 1 - Introduction

1.1 RESPONSE project

RESPONSE is a 60-month project funded by the European Commission's Horizon 2020 Framework Programme (Grant Agreement nº 957751).

RESPONSE aims to achieve energy sustainability in smart cities by implementing intelligent and interconnected energy systems. RESPONSE supports the lighthouse (LH) cities of Dijon (FR) and Turku (FI), and the Fellow Cities (FC) of Brussels (BE), Zaragoza (ES), Botosani (RO), Ptolemaida (GR), Gabrovo (BU), and Severodonetsk (UA) to solve the energy trilemma (security, equity/affordability, and environmental sustainability) at building, block, and district levels.

The City Transformation Strategy adopted by RESPONSE consists of five complimentary Transformation Axes (TA) that group ten Integrated Solutions (IS), as shown by Figure 2.

Figure 2 Transformation Axes of RESPONSE grouping ten Integrated Solutions related to the energy trilemma.

1.2 Scope and objectives

The main objective of Task 9.6 is to develop a toolkit, mainly consisting of software elements, to be used in the Scalability and Replicability Analysis (SRA) of the Information and Communication Technologies (ICT) involved in some Innovative Elements (IEs) implemented in the LH cities. The SRA is carried out by Task 9.7.

The aim of an SRA is to extract conclusions, either quantitative or qualitative, about how a solution would perform under different scenarios. These scenarios may be related to the scalability of the solution: an increase in the size of the solution (e.g., larger area implementation) or in its density (e.g., involving more devices in the same area); or they may be related to the replicability of the solution, analysing its performance when it is implemented in a different environment or under different conditions.

The toolkit developed within this task will be mostly used in the scalability analysis of the ICT. As it is not cost-effective to carry out such analysis in a laboratory with real hardware, the toolkit consists of simulation models that replicate the ICT characteristics of an IE, increasing the variety of scenarios that can be studied.

However, some IEs implemented in the LH cities, due to their ICT characteristics, may be difficult to be modelled with accuracy (e.g., when the communications are through the Internet) or do not have a dedicated ICT infrastructure at all.

The purpose of this deliverable (D9.6 RESPONSE scalability and replicability evaluation toolkit - V1) is to present which IEs will be considered for the toolkit and describe their ICT characteristics and requirements. Furthermore, the software tools used in the toolkit development are introduced, and a preliminary design of the toolkit is presented, as well as some scalability and replicability parameters for the SRA in T9.7.

The complete description of the ICT toolkit, possible SRA scenarios, and unitary validation tests will be under the scope of the second deliverable of the task (D9.13 RESPONSE scalability and replicability evaluation toolkit - V2).

1.3 Relation to other RESPONSE activities and WPs

The IEs of RESPONSE that are more ICT-focused are those included in TA#4 "Integrated and Interconnected City Ecosystems". Therefore, the information presented in D3.1 "Master City plans for TA#4 Integrated and Interconnected City Ecosystems" was relevant for an initial identification of IEs to be considered for the toolkit.

The main outcome of T9.6 is an ICT toolkit to conduct an SRA in T9.7 "Replicability and Scalability of the demonstrated TAs on a European level". Therefore, T9.7 is highly related and, to some extent, dependent on the activities developed within T9.6.

1.4 Structure of the deliverable

The core of this deliverable consists of two main chapters that describe the ICT scalability and replicability evaluation toolkit that is being developed in T9.6:

- Chapter 2 – Dijon ICT Toolkit.
- Chapter 3 – Turku ICT Toolkit.

Both chapters follow the same structure.

First, the **scope** of the toolkit for the LH city is determined. That is to say, which IEs or systems will be possible to simulate with the toolkit. The IEs implemented within TA#4, together with other IEs where ICT may play an important role, are evaluated considering their simulation feasibility and the value that a simulation-based SRA could provide.

Once the scope of the toolkit for the LH city is defined, the ICT **architectures and technologies** of the IEs or systems under scope are described. This includes a general topology of the communications network implemented, information about what types of devices are involved, and a summarised but complete description of the specific communication protocol or technology that will be simulated by the toolkit.

After this, the systems under scope are **mapped to a modified version of the Smart Grid Architecture Model (SGAM).** The analysis of this SGAM-like mapping allows the formulation of key scalability/replicability questions that should be considered when designing and developing the toolkit, as it would be used during the SRA in T9.7 to try to answer these questions.

After the formulation of the main scalability/replicability questions, the **SRA parameters** are defined. SRA parameters are those simulation parameters that represent communication network characteristics that may be changed in order to analyse the network's performance under different scenarios. Following the SRA parameters definition, the ICT **characteristics and requirements** considered for the simulation of the communication system are detailed. These characteristics also include some reasonable assumptions and estimations because the final details of some implementations are still unknown at the moment of writing this deliverable.

Finally, the chapter describing the ICT toolkit for each LH city is concluded with the information about the **toolkit design**. For both LH cities, this section is divided into three subsections about the main elements that will form the toolkit: the OMNeT++ simulator, the graphical user interface, and the analysis tool.

After Chapters 2 and 3, Chapter 4 provides the **conclusions** drawn from this deliverable and explains how the SRA in T9.7 will take full advantage of the ICT toolkit developed in T9.6.

1.5 Methodology

The development of the toolkit consists of seven subtasks:

1. **Identify the simulation-based ICT SRA scope.** This subtask is about identifying which IEs can be included in the scope of the simulation-based ICT SRA. Although the information collected by T3.1 was a good starting point, the LH city managers and relevant partners were contacted in order to provide some more ICT details (e.g., ICT topology and technologies used), so that the inclusion of an IE in the scope of the toolkit could be better assessed.
2. **Determine the characteristics and requirements of devices and ICT links.** Once an IE was considered appropriate to be included in the toolkit, a more detailed description of the characteristics and requirements of the devices and communication links was needed in order to develop a simulation model, baseline scenarios, and to set the boundary conditions for scalability and replicability. As some characteristics and requirements were not known yet at the time of asking, the information was completed with reasonable assumptions.
3. **Identify critical links and nodes.** This subtask is about performing a first high-level qualitative analysis of the information collected in the previous subtask, so that those communication links or nodes that can constitute a risk for scalability are specially considered when designing SRA scenarios.
4. **Toolkit design and software to be used in the development.** This subtask is about how the toolkit is conceived so that the ICT characteristics and requirements can be properly simulated, and what software is used for that purpose. The software to be used in the analysis of results is also included.
5. **Toolkit development.** This subtask is just about developing the toolkit following the defined design.
6. **Unitary tests of the toolkit (validation).** To check that the simulation models work well when changing simulation parameter and that their results can be properly analysed.
7. **Baseline scenario for SRA.** This subtask is highly related to T9.7. It is about defining a first set of baseline scenarios for the different simulation models that form the SRA toolkit.

As commented in section 1.2, this deliverable mainly focuses on subtasks 1-4. A breakdown of the methodology followed for this deliverable is presented by Figure 3.

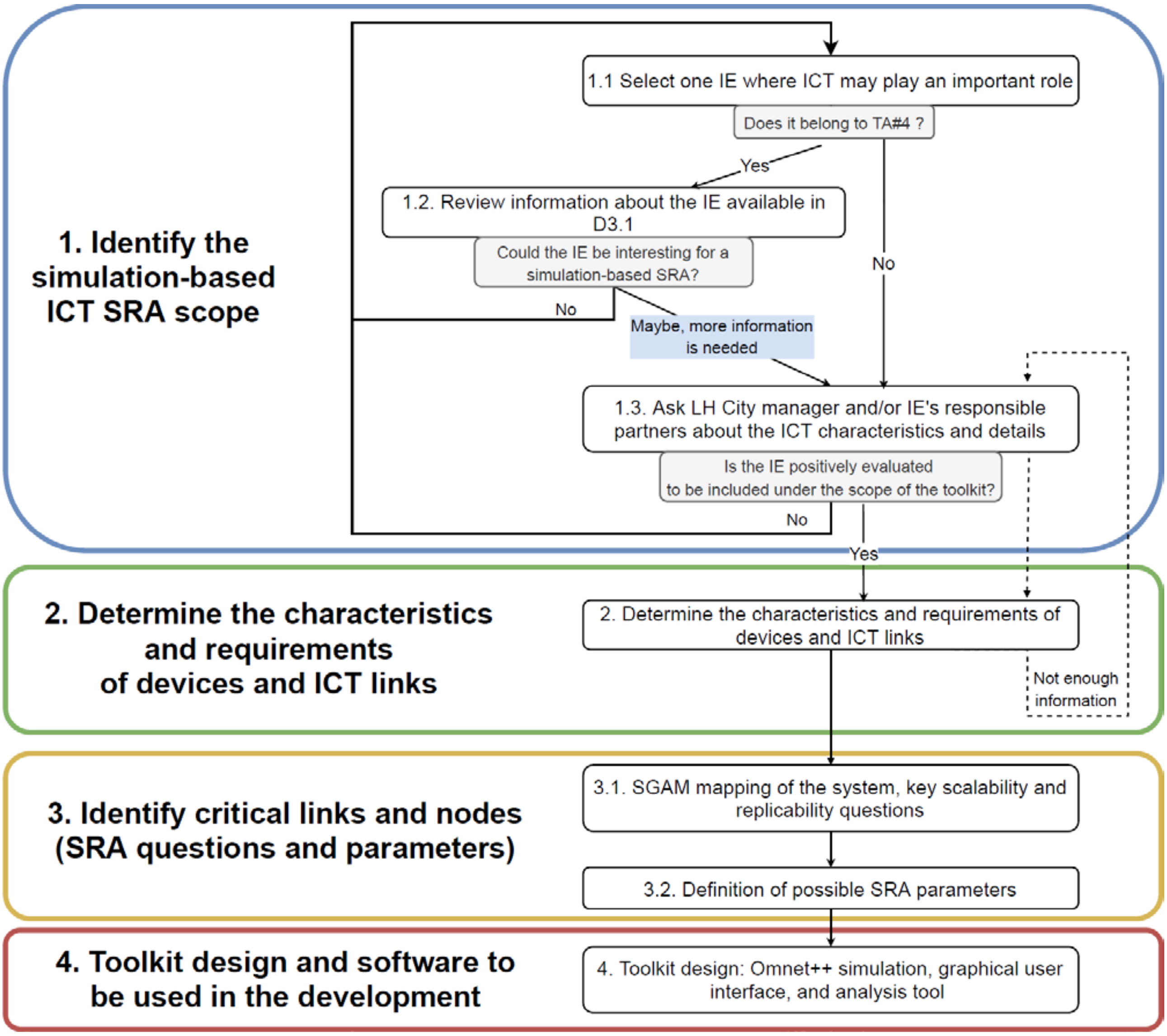

Figure 3 Methodology applied for this deliverable.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Chapter 2

Dijon ICT Toolkit

Chapter 2 – Dijon ICT Toolkit

2.1 Scope

Of all the IEs implemented in the LH city of Dijon, those grouped under TA#4 (Integrated and Interconnected City Ecosystems) are the ones that are more ICT-related, as their objective is to facilitate the integration of IEs' systems to complement and optimize the wider energy system. Table 1 shows the IEs demonstrated in Dijon under the IS 4.1 and IS4.2 of the TA#4 considered in D3.1 (Tryferidis et al., 2021).

Table 1 Innovative Elements demonstrated in Dijon under IS 4.1 and IS 4.2

IS 4.1 City information Platform-enable innovations
IE 4.1.1 GENESYS tunnelling solution
IE 4.1.3 PEB Multi-Energy Dashboard
Merged Solution: Energy-Climate Platform IE 4.1.2 Shared data-lake IE 4.1.4 Online computation of advanced energy and climate indicators IE 4.1.5 Energy-Climate dashboard
IS 4.2 E-mobility integration into the Grid and City Planning
IE 4.2.1 EV charging points with PV, shading, and battery featuring smart-charging and V2G operation
Merged Solution: Electric Mobility Tooling and Studies IE 4.2.2 Smart-charging infrastructure deployment planning IE 4.2.3 Geographic visualization tool for enhanced decision-making

However, not all the IEs shown in Table 1 would require a simulation-based approach (quantitative approach) for the SRA; for some of them, due to their characteristics, the simulation results could be irrelevant or unreliable. This is the case for the merged solutions of Dijon; the "Energy-Climate Platform" and the "Electric Mobility Tooling and Studies" are solutions that are mainly software-related and whose communications are assumed to be mostly through Web Services on the Internet. Both characteristics are complex to replicate accurately in a simulation environment, so these two solutions are discarded for the toolkit.

IE 4.1.1 GENESYS tunnelling solution will be implemented in Dijon at a district level by EDF. It consists of a software element, called eCore, which supports multiple communication protocols (e.g., IEC 61850, Modbus, HTTPS, etc.); and a hardware element, called EMPAIR (in French, *Equipement Modulaire de Protection des Accès Industriels Répartis*). In Dijon, an EMPAIR device will be installed, including an eCore instance, to control and monitor the self-consumption energy system by communicating with the assets (e.g., batteries, solar PV, etc.) through a LAN.

IE 4.1.3 PEB Multi-Energy Dashboard will be implemented in Dijon to measure the energy efficiency at a PEB/building level. It will be based on the deployment of the i-Board web platform, which will show real-time data to building operators and managers. The data from smart sensors, according to D3.1, would be collected by a data concentrator through wireless or wired Local Area Network (LAN) communications. However, the specific ICT details of this implementation for Dijon are still to be defined.

IE 4.2.1 EV charging points with PV, shading, and battery featuring smart-charging and V2G operation will implement EV charging points coupled with solar PV and a stationary battery. For smart-charging and Vehicle-To-Grid (V2G) operation, the charging stations would communicate with DREEV platform using the Open Charge Point Protocol (OCPP), according to D3.1. The DREEV platform is a software element in charge of establishing the power setpoints of the V2G charging points.

Considering the characteristics of the different IEs, IE 4.1.1 GENESYS tunnelling solution may be the one where a simulation-based SRA will provide more insightful results. IE 4.1.3 is discarded for the toolkit given that its specific deployment characteristics are still in progress. Regarding IE 4.2.1, its ICT scalability would be deeply related to the scalability of the DREEV platform, which is not hardware-based (software element). The scalability of software, when hosted in the cloud, can be assumed to be guaranteed.

From the scalability point of view, the EMPAIR could be a potential information bottleneck (many devices exchanging information with the same device) that may affect the scalability of the solution. Furthermore, the scalability of EMPAIR's communications is deeply related to the scalability of the connected IEs, as GENESYS is involved in the control and monitoring of these systems.

Considering this, the ICT SRA toolkit for Dijon focuses on IE 4.1.1, as it is ICT-related to multiple IEs and its SRA results can be of great interest for the LH cities and FCs. Table 2 summarises the evaluation of the different IEs for the scope of the Dijon ICT SRA toolkit.

Table 2 Summary of the evaluation of the IEs for the scope of the Dijon ICT SRA toolkit.

IS 4.1 City information Platform-enable innovations	Consideration for ICT Toolkit
IE 4.1.1 GENESYS tunnelling solution	**Included. Focused on communications between EMPAIR device and energy assets**
IE 4.1.3 PEB Multi-Energy Dashboard	**Not included.** • Definition of the ICT details is still in progress
Merged Solution: Energy-Climate Platform IE 4.1.2 Shared data-lake IE 4.1.4 Online computation of advanced energy and climate indicators IE 4.1.5 Energy-Climate dashboard	**Not included.** • Software-related elements. • Internet communications (inaccurate simulations)
IS 4.2 E-mobility integration into the Grid and City Planning	
IE 4.2.1 EV charging points with PV, shading, and battery featuring smart-charging and V2G operation	**Not included.** • Scalability related to the DREEV platform, which is a software element assumed to be in a cloud server.
Merged Solution: Electric Mobility Tooling and Studies IE 4.2.2 Smart-charging infrastructure deployment planning IE 4.2.3 Geographic visualization tool for enhanced decision-making	**Not included.** • Software-related elements. • Internet communications (inaccurate simulations)

2.2 Architectures and technologies

Figure 4 shows the architecture to be implemented in IE 4.1.1 to control and monitor the self-consumption system at a district level, which consists of IEs considered under IS 1.1 (RES generation), IS 3.1 (Electricity storage), and IS 3.2 (Heat storage).

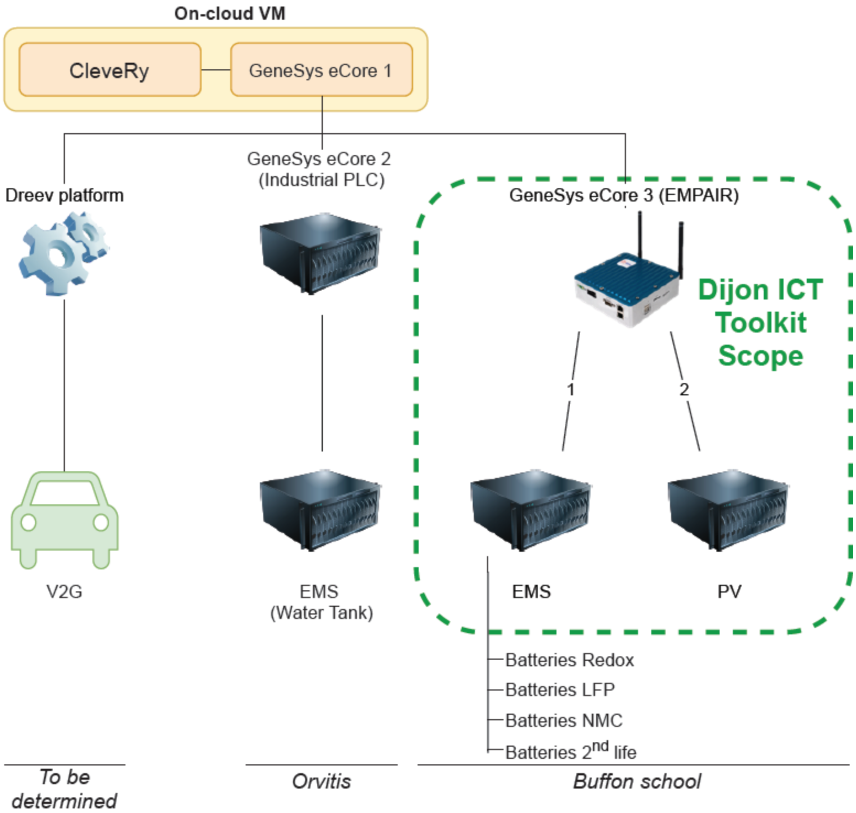

Figure 4 Self-consumption control system architecture for IE 4.1.1

As shown by Figure 4, and as described in section 2.1, the ICT toolkit developed for Dijon will focus on the ICT network of the EMPAIR device. The EMPAIR is in charge of sending control and monitoring commands to two systems that are being demonstrated within the project: the EMS in charge of four types of batteries, and the PV control system (in all likelihood, a PV data logger). These two systems will be located at Buffon School (PEB2_B4) and, following a star topology, connected to the EMPAIR device through Ethernet cables. Therefore, a LAN is implemented, and the communication protocol used at the application layer to exchange information is the Modbus over TCP protocol (Modbus/TCP).

Following subsections briefly describe the technologies involved in the communication network, as they will be modelled for the simulation toolkit.

2.2.1 Ethernet

The Ethernet standard (IEEE 802.3 standard[1]) defines the cabling and signalling characteristics at the physical layer and the data frames at the link layer for LAN operation.

Although the Ethernet standard allows the use of coaxial, twisted pair, or fibre optic cables, only the use of twisted pair cables is considered for the development of the toolkit. More specifically, the cable will be assumed to have the characteristics of an Ethernet CAT5E cable: 1000 Mbps of speed and a frequency of 100 MHz.

2.2.2 Modbus/TCP

Modbus/TCP (Modbus Organization, 2012) is an application-layer communication protocol for client-server communications between devices. Modbus can be implemented on the TCP/IP stack over Ethernet, as shown in Figure 5.

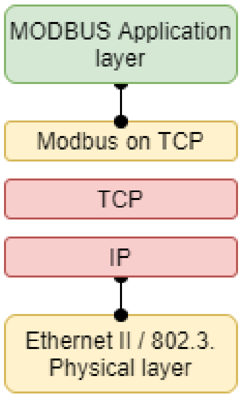

Figure 5 Modbus TCP/IP Communication Stack

The Modbus protocol defines a protocol data unit (PDU) that contains a function code and the data sent. As it is a client-server protocol, the client indicates to the server (request), through the function code, what kind of action must be performed, and provides additional information in the data attached. Then, the server processes the client's request, performs the required action, and sends a response back to the client. Figure 6 shows a simplified diagram of Modbus transactions between client and server.

[1] https://www.ieee802.org/3/

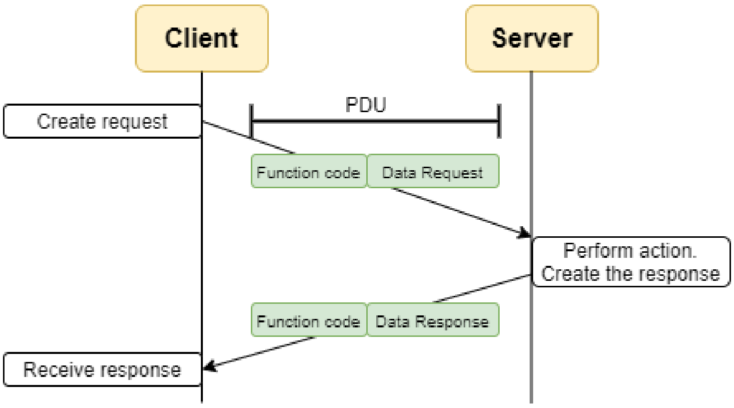

Figure 6 Simplified Modbus transaction between client and server

Modbus defines up to 21 function codes. For each function code, the PDU of a request and a response is specified and must be ≤ 253 bytes. For example, for function code 03, “Read Holding Registers”, the request and response PDU sizes would be as shown by Table 3.

Table 3 Request and Response PDU for Read Holding Registers Modbus function code

Request		Response	
Function code	1 Byte	**Function code**	1 Byte
Starting Address	2 Bytes	**Byte count**	1 Byte
Quantity of Registers (N)	2 Bytes (Max. 125 registers)	**Register value**	N x 2 Bytes

To be transmitted, a Modbus Application Protocol (MBAP) header (7 bytes) must be added to the PDU, constituting the Application Data Unit (ADU). As the Modbus protocol is implemented over TCP and Ethernet, working at different OSI model layers, the final message that will be transmitted through the Ethernet cable will contain additional valuable information for each of the layers as shown by Figure 7: an Ethernet header (14 bytes), an IP header (20 bytes), a TCP header (20 bytes), and checksum information (4 bytes).

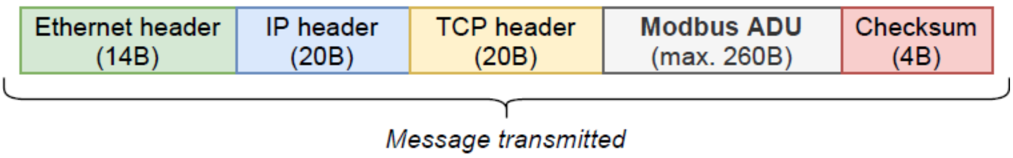

Figure 7 Packets that form a message transmitted with Modbus/TCP over Ethernet

2.3 SGAM mapping

In order to fully understand the IE 4.1.1 EMPAIR subsystem, and to develop a toolkit that is able to simulate interesting and reasonable scalability scenarios, the subsystem was mapped to a modified version of the Smart

Grid Architecture Model (SGAM) that better reflects the energy domains of a smart city. In this modified version, the traditional smart grid domains (bulk generation, transmission, distribution, Distributed Energy Resources (DER), and customer premises) have been replaced by two energy-related smart city domains: the Positive Energy District (PED) domain and the Positive Energy Block (PEB) domain.

The Modbus/TCP Ethernet LAN modelled for the toolkit would be placed at a PEB level (Buffon school) and would cover the field (battery and solar PV controllers) and station (EMPAIR device) zones of the SGAM, as shown by Figure 8. The on-cloud GeneSys eCore 1, which is out of the scope of the toolkit, is in charge of managing the EMPAIR and, therefore, is at the PED domain.

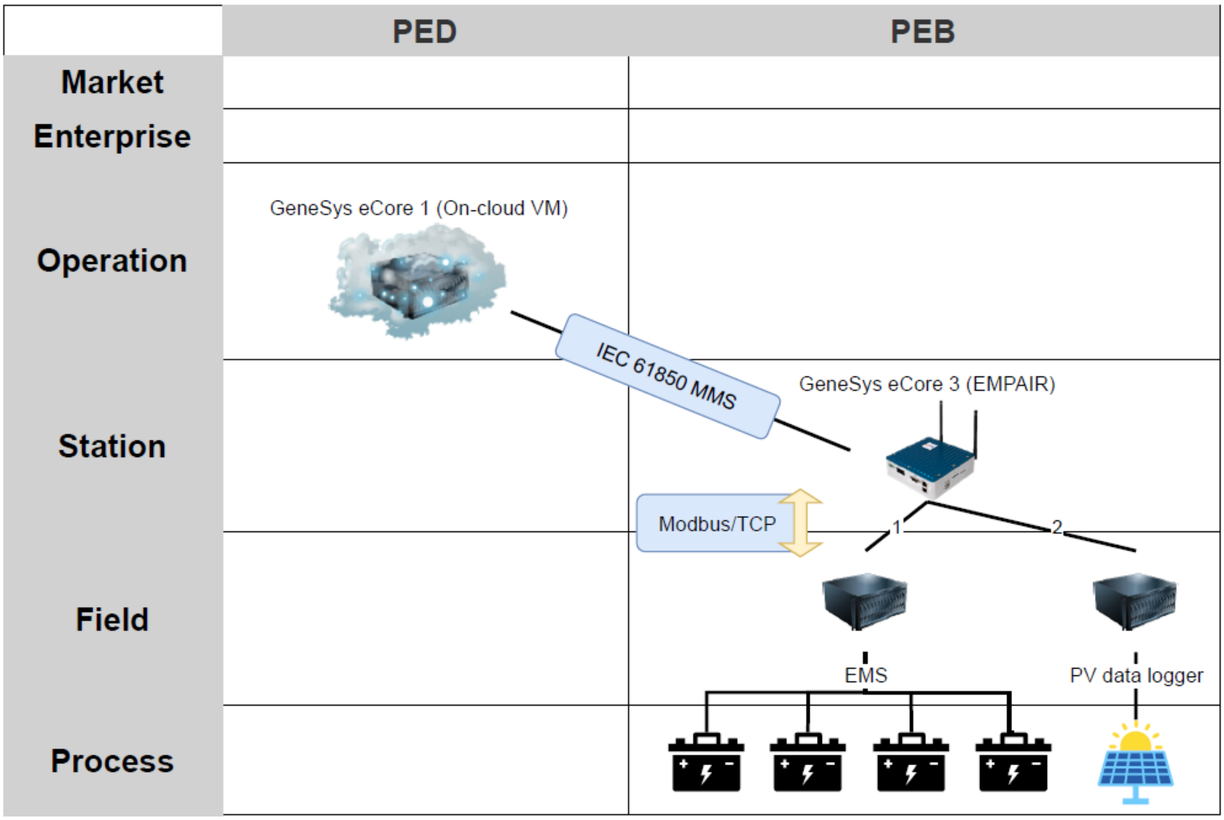

Figure 8 Mapping of the IE 4.1.1 under the toolkit scope to the six SGAM zones and PED/PEB domains. Component-communication layers.

By analysing Figure 8, some scalability questions can be initially formulated so that the scalability parameters can be better defined in section 2.4:

- **What would be the effect of placing the EMPAIR device in the PED domain?** This would mean increasing the size of the LAN or, in other words, increasing the distance (i.e., Ethernet cable lengths) between the connected devices. There may be a maximum distance under which the operational requirements can no longer be satisfied.
- **What would be the effect of increasing the number of devices connected to the EMPAIR?** This question could also be studied in combination with the previous one. When placed at a PEB level, the

results would show the maximum number of devices that can be controlled within a building; when placed at a PED level, the operational contour defined by distance and number of devices could be obtained.

The SGAM mapping in Figure 8 is complemented by Figure 9 with a summary of the network regarding the five interoperability layers of SGAM.

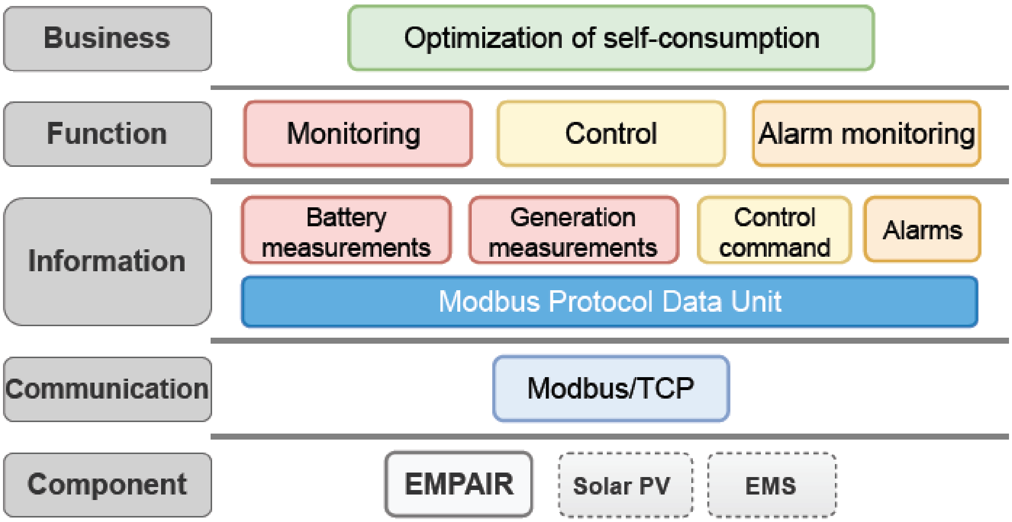

Figure 9 Mapping of IE 4.1.1 to the five interoperability layers of SGAM

At the business layer, the objective of IE 4.1.1. is to optimize the self-consumption operation of the PED. To achieve this optimization, three tasks are carried out at the function layer as shown by Figure 9:

- **Monitoring.** This function is about retrieving data (battery and generation measurements in the information layer) from the EMS and solar PV through the EMPAIR.
- **Control.** This function is about sending control commands to the EMS through the EMPAIR.
- **Alarm monitoring.** This function is about monitoring determined alarm parameters from the devices so that possible problems or the unavailability of the resources can be considered by the optimization process.

Therefore, the simulation model developed for the toolkit should include these three information exchange functions. Details about how these are modelled are described in section 2.5.

2.4 SRA parameters

In order to answer the scalability questions presented in section 2.3, and to obtain other insightful results in the ICT SRA of T9.7, it has to be possible to change some parameters in the simulation model to easily reproduce and analyse different scalability and replicability scenarios.

The definition of potential SRA parameters is of great importance for the development of the toolkit, as they must be taken into consideration during the toolkit design and development to guarantee some flexibility for the SRA. It will be possible to modify the following SRA parameters in the Dijon ICT SRA toolkit:

- **Application layer protocol.** Modbus/TCP is not the only industrial protocol that can run over Ethernet. A possible alternative would be the PROFINET protocol, which is briefly described in Appendix A. It could be interesting for the SRA to simulate the operation with both alternatives and analyse the differences in performance for the same scenario. This parameter would be related to the replicability of the system.
- **Nº of devices connected.** The real implementation would only connect the EMPAIR to the EMS and PV data logger in the Buffon school. However, the SRA will consider scenarios where the number of devices will vary in order to assess the scalability of the system.
- **Type of devices.** It may be interesting to assess the performance of the system when the type of device is not equally assigned among the total number of connected devices.
- **Distance of devices.** To assess the scalability, some SRA scenarios may need to vary the Ethernet cable length of the devices connected.
- **Bit Error Rate (BER).** The BER is the number of wrong bits divided by the total number of bit transmitted. The ideal approach would be to use an Ethernet cable model that provides the BER given the length of the cable. However, it should be possible to define specific BER values to replicate other conditions.
- **Processing delay.** Real devices do not send the response to a Modbus request immediately; it takes them some processing time. Since it is not possible to know the average processing time of the real hardware that will be implemented, different values may be assumed during the development of the SRA.

Table 4 summarises these potential SRA parameters, indicating the type of analysis to which they would contribute (scalability, replicability, or both), and proposing a baseline value (i.e., “default” value defined in the simulation model). These SRA parameters listed, however, do not necessarily mean that they all will be considered by the SRA scenarios in T9.7, but that it will be possible to easily modify these parameters in the graphical user interface of the (see Section 2.6.2) ICT toolkit if necessary.

Table 4 Potential SRA parameters to develop different scenarios for IE 4.1.1 in Dijon

Parameter	Type	Proposed baseline
Application layer protocol	Replicability	Modbus/TCP. Alternative: PROFINET
Nº of devices connected	Scalability	Two: EMS and PV data logger
Type of devices	Replicability / Scalability	EMS – PV data logger
Distance of devices	Scalability	20 meters
Bit Error Rate (BER)	Replicability / Scalability	0 or cable-based
Processing delay	Replicability	9 ms

2.5 Characteristics and requirements

The three functional tasks (monitoring, control, and alarm monitoring) in which the EMPAIR device is involved in IE 4.1.1 translate into three different types of information exchanges. To model each information exchange, and considering the device involved, characteristics such as the frequency and the amount of information exchanged have to be taken into consideration:

- **Battery measurements.** This information object is considered to be retrieved once per hour from the EMS. The Modbus function code would be 03 "Read Holding Registers". The basic amount of information transmitted is assumed to be three float measurements per battery type and, in Modbus/TCP, each float value is represented by two registers. This means that for the EMS that manages four types of batteries, the total amount of registers would be 24, and, using the Read Holding Registers function, this information would take 48 bytes in the message (2 bytes per register).
- **Alarms.** This information object is considered to be retrieved once per minute from all the devices connected downstream to the EMPAIR (EMS and PV data logger). For this information object, the Modbus function code would be 01 "Read Coils". The basic amount of information transmitted is assumed to be one Boolean value per battery type and PV data logger, in Modbus/TCP, a Boolean value is represented by one coil (one bit). This means that in one byte, eight alarm values can be transmitted (one byte = eight bits).
- **Generation measurements.** This information object is considered to be retrieved once per minute from the PV data logger connected to the EMPAIR. The Modbus function code would be 03 "Read Holding Registers". The basic amount of information transmitted is assumed to be one float measurement, taking 4 bytes in the message.
- **Control command.** This information object is considered to be sent from the EMPAIR to the EMS. In this case, the Modbus function code would be 16 "Write Multiple Registers" and the amount of

information is assumed to be constant to four float commands, regardless of the number of battery types. This means that for the EMS, the total amount of registers would be eight, taking 16 bytes in the message (2 bytes per register).

Table 5 summarises the characteristics and assumptions made for each information exchange. Links ID corresponds to those indicated both in Figure 4 and Figure 8.

Table 5 ICT Characteristics and assumptions of IE 4.1.1 for the ICT SRA toolkit of Dijon

Link ID	Source	Receiver	Protocol	Information object	Frequency of exchange	Message size (Bytes)	Communication technology
1	EMS (4 battery types)	GENESYS eCore 3 (EMPAIR)	Modbus/TCP	Battery measurements	1/h	3 float x 4 batteries = 12 float = 24 registers = 48 bytes	Ethernet
				Alarms	1/min	1 Boolean x 4 batteries = 4 Booleans = 4 coils = 1 byte	
	GENESYS eCore 3 (EMPAIR)	EMS (4 battery types)		Control command	1/h	4 float = 8 registers = 16 bytes	
2	PV data logger	GENESYS eCore 3 (EMPAIR)	Modbus/TCP	Generation measurements	1/min	1 float = 2 registers = 4 bytes	Ethernet
				Alarms	1/min	1 Boolean = 1 coil = 1 byte	

Based on the previous characteristics, and considering that the Modbus/TCP protocol is a client-server protocol, the following baseline **operational requirements** must be taken into consideration when modelling the communications network:

1. The EMPAIR device (Modbus client) should be able to request the alarm values to all the devices (Modbus servers) in less or equal to one minute.
2. The client can only establish a Modbus/TCP connection with one server at a time. This means that, once the client is connected to a server to request the alarm values, it should also evaluate if it is time to request battery measurements, generation measurements, or to write control command, depending on the type of server to which it is connected at that moment. If the client has to request all types of information objects within the same connection, the order will be: alarms, battery/generation measurements, and, finally, control commands.
3. Therefore, the additional delays caused by non-alarm information exchanges must be considered in order to comply with requirement 1.
4. Once the client has established connection with every server in less or equal to one minute, the client must repeat the process automatically.

5. The cable will be assumed to have the characteristics of an Ethernet CAT5E cable: 1000 Mbps of speed and a frequency of 100 MHz.
6. The servers will be connected to the client following a STAR topology.

The PROFINET version of the network would implement the same characteristics that are presented in Table 5 for Modbus/TCP. However, since PROFINET is a provider/consumer protocol running over UDP (only the non-real-time channel defined in the protocol will be simulated), the EMPAIR does not need to request the information: each device in the network will autonomously send the information objects to the specific receiver at the defined frequency (e.g., every minute, every hour, etc.).

2.6 Toolkit design

The ICT toolkit will be developed using free open-source programs (e.g., OMNeT++ simulator) and widely used programming languages (e.g., C++ and Python). The operating system selected for the development environment was Ubuntu 20.04.1, which is GNU/Linux based.

The ICT toolkit for the LH city of Dijon will consist of three main software elements, as shown in Figure 10:

- The **OMNeT++ simulator.** It is the main element, since the simulation model will be developed using this framework. Based on a configuration file, the OMNeT++ simulation framework will run the communication network model and generate a large comma-separated values (CSV) file with statistics and other information to be analysed.
- A **Graphical User Interface (GUI)**. Through a simple interface, a user who has not been directly involved in the development of the simulation model could create a configuration file to run one or more scenarios. The user would be able to run a configuration file, whether recently created or an existent one, without the need of having previously initiated the OMNeT++ simulation framework.
- **Analysis tool.** Developed to obtain relevant indicators and graphs from the results and statistics CSV file.

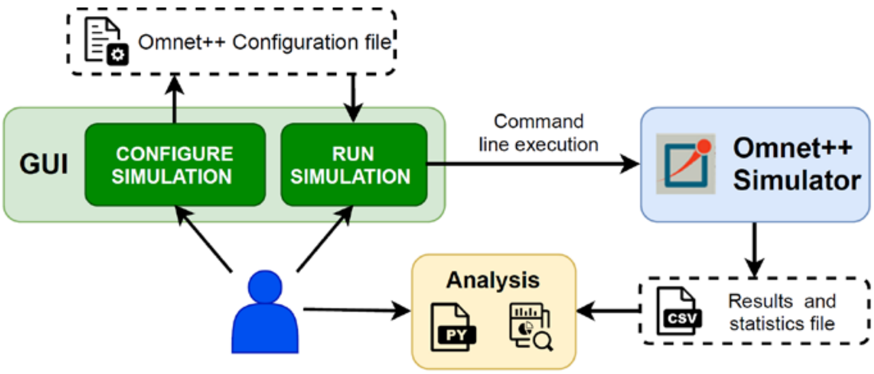

Figure 10 General design of the ICT toolkit to be used in the SRA for the LH cities

Following subsections provide a more in-depth description of these software elements and the technologies, models, and libraries involved in each of them.

2.6.1 OMNeT++ Simulator

OMNeT++ is a discrete event simulator extensively used to build communication network simulations. It is open-source and freely distributed under the Academic Public License.

In OMNeT++, networks and devices are defined in NEtwork Description (NED) language, and their functioning is programmed in C++.

An OMNeT++ implementation of the application layer protocols considered in section 2.4 (Modbus/TCP and PROFINET) has not been found in any of the open-source libraries that extend the framework. However, the INET Framework[2] provides open-source OMNeT++ models for wired and wireless networks, and modules to be used as a basis to develop application layer protocols running over TCP (Modbus/TCP) and UDP (PROFINET). INET also provides a detailed Ethernet model to simulate the physical layer of communications.

To model the Modbus/TCP network, both the EMPAIR and the different types of devices connected (EMS, BMS, and PV data logger) are modelled using the already-defined "Standard Host" model of INET. The difference will be in the application each "Standard Host" will use:

- For the EMPAIR, the application is a basic Modbus client application, which was obtained by modifying the TCP basic client application of INET to replicate the characteristics described in section 2.5 and the basic characteristics of Modbus/TCP function codes.
- For the devices connected to the EMPAIR, the application is a basic Modbus server application, which was obtained by modifying the TCP generic server application of INET. Since the server only has to answer to client's request, only the changes to guarantee compatibility with the Modbus/TCP client are necessary in this case.

To model the PROFINET alternative to the Modbus/TCP network, the "Standard Host" model of INET is also used, only changing the application implemented so that it replicates the basic functioning of PROFINET. Therefore, in this case the application developed for all the devices (EMPAIR and connected devices) is a basic PROFINET UDP application which was obtained by modifying the basic UDP application of INET. Since, in PROFINET, the devices can be simultaneously providers and consumers, the application was programmed to behave different depending on the type of device, which should be specified in the configuration file for each device in the network.

[2] http://www.inet.omnetpp.org

From the network's topology perspective, the module representing the EMPAIR would be connected to an Ethernet switch module (defined in INET) which would be connected, through Ethernet cables, to the different devices, following a STAR topology, as shown in Figure 11.

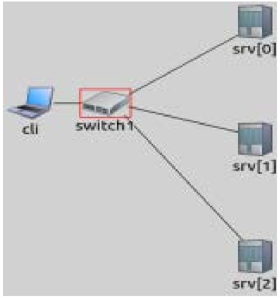

Figure 11 Screenshot of the baseline communication network in OMNeT++

2.6.2 Graphical User Interface

As previously introduced in Figure 10, the GUI consists of two windows: the "configuration" window, and the "run simulation" window. The GUI is developed in Python.

2.6.2.1 Modbus/TCP Simulation Configuration window

In this window, which is shown in Figure 12, the user can introduce a set of parameters for the Modbus/TCP simulation and generate an .INI configuration file that can be read by OMNeT++. By clicking on "Run" in the top bar, the run simulation window would emerge.

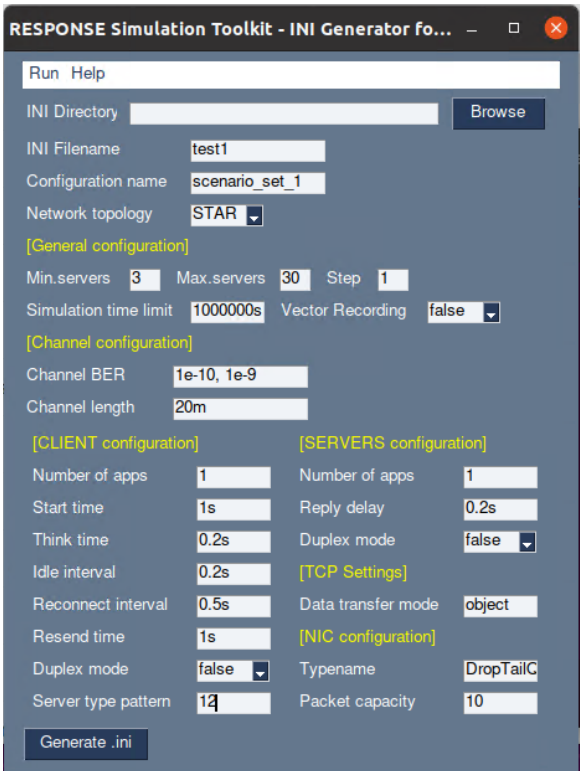

Figure 12 Screenshot of the window to create a configuration file for the Modbus/TCP simulation. Draft version.

The simulation parameters and fields that the user would be able to modify are the following:

- **INI Directory:** The path where the .INI configuration file will be saved when pressing the "Generate .ini" button at the lower left corner of the window.
- **INI Filename:** The name of the .INI configuration file used to save it.
- **Configuration name:** The configuration needs a name to be distinguished from others and to execute the simulation. For example, it could be "scenarios_set_1" or, more descriptive, "IncreasingBER".
- **Network topology:** If necessary, the user would be able to change the topology of the network from STAR to BUS. This parameter was not included as a relevant SRA parameter in section 2.4 due to the significant increase in the number of possible scenarios that considering both topologies would cause.
- **General configuration:**
    - **Min. Servers:** Minimum number of servers (i.e., devices connected to the EMPAIR) for a set of scenarios.
    - **Max. Servers:** Maximum number of servers (i.e., devices connected to the EMPAIR) for a set of scenarios. If it is equal to "Min. Servers", all the scenarios in the configuration file will have the same number of servers.
    - **Step:** Step to be considered when iterating from the "Min. Servers" to the "Max. Servers". For example, if "Min. Servers" =2, "Max. Servers" = 10 and "Step" =2, the OMNeT++ would run simulations where, while keeping every other parameter the same, the number of servers would be 2 in the first simulation run, 4 in the second, 6 in the third... until 10 in the fifth simulation run. Therefore, this is an automated way of varying the number of devices connected, which was an SRA parameter defined in section 2.4.
    - **Simulation time limit:** Simulation time in seconds that, once reached, will stop the execution of the simulation.
    - **Vector recording:** In case the user wants to obtain a CSV file that stores the vectors of variable indicators. By default, it is set to *false*, since statistics would be simpler to analyse.
- **Channel configuration:**
    - **Channel BER:** BER for the Ethernet cables connecting the different devices. Multiple BER values can be set, separated by commas. This would increase the number of possible scenarios. For example, following with the example previously described in "Step", if "BER" = 1e-10, 1e-9, there would be two scenarios where the number of servers would be 2: one

where "BER" = 1e-10, and another one where it is 1e-9. Therefore, in this example, the number of scenarios would double. This was considered an SRA parameter in section 2.4.

- **Channel length:** To define the length of the Ethernet cable (in meters) between the switch and all the connected devices (except the EMPAIR, which would be connected very close to the switch). This was considered an SRA parameter in section 2.4.

- **Client configuration:** This would be the configuration of the EMPAIR device.
  - **Number of apps:** The number of basic Modbus client applications (see subsection 2.6.1) that the device would contain. During the SRA, it should be always set to one.
  - **Start time:** Once the simulation has started, the simulation time in seconds at which the client will try to establish its first connection with a server.
  - **Think time:** Time gap in seconds between requests to the same server. It is related to the processing delay the device would experiment, which was considered an SRA parameter in section 2.4.
  - **Idle interval:** Time gap in seconds between sessions (i.e., between the closure of a connection and the starting of a new one with a different server). It is related to the processing delay the device would experiment, which was considered an SRA parameter in section 2.4.
  - **Reconnect interval:** Time in seconds before trying again to establish a connection with a server when there has been a failed attempt.
  - **Resend time:** Time in seconds that the client waits for a server's response before resending the request.
  - **Duplex mode:** To set bidirectional communications for the client. In case of Modbus/TCP, it must be set to false.
  - **Server type pattern:** To define the order and proportions of the different types of devices considered for the toolkit, which are related to the "Type of devices" SRA parameter defined in section 2.4. The equivalences are:
    - **1:** PV data logger type.
    - **2:** BMS (4 battery types).

    The pattern set will be used by the simulation model to know in which order the servers must be placed and to let the client know the type of server it is connecting to, so that the characteristics collected in Table 5 can be applied correctly during the simulation.

For example, if “Server type pattern” = 122, this would mean that, in a scenario with 10 devices connected, the servers would be placed in this order: 1221221221.

- **Servers’ configuration:**
    - **Number of apps:** Similar to client configuration. The number of generic Modbus server applications (see subsection 2.6.1) that the device would contain. During the SRA, it should be always set to one.
    - **Reply delay:** Time gap in seconds between receiving a client’s request and sending the response to that request. It is related to the processing delay the device would experiment, which was considered an SRA parameter in section 2.4.
    - **Duplex mode:** To set bidirectional communications for the server. In case of Modbus/TCP, it must be set to false.
- **TCP settings:**
    - **Data transfer mode:** This parameter was included because it is necessary to correctly configure the simulation. Nevertheless, it should not be changed from its default value, “object”. It defines how the data are represented in TCP packets: by encapsulated C++ packet objects (used in this case), by raw bytes, or by its byte count.
- **NIC configuration:**
    - **Typename:** The queue type implemented to manage packets. It should not be changed from its default value, “DropTailQueue”.
    - **Packet capacity:** The “DropTailQueue” requires defining the size of the buffer so that when it is surpassed, further packets are dropped. It should not be changed from its default value, 10.

It must be taken into consideration that the toolkit is still being developed and that the interface described in this subsection is not definitive.

2.6.2.2 Run simulation window

The run simulation window, shown in Figure 13, provides a simple interface to run any OMNeT++ configuration file with the OMNeT++ framework.

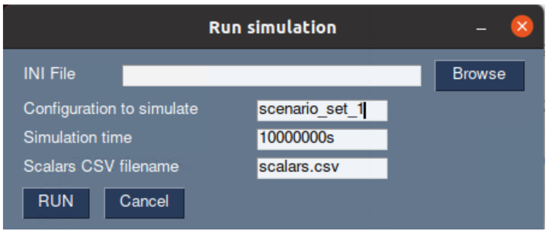

Figure 13 Screenshot of the window to run an OMNeT++ simulation using a specific configuration file. Draft version.

The fields that the user should introduce are the following:

- **INI file:** Through a system explorer, the user must select the .INI configuration file that will be read by OMNeT++.
- **Configuration to simulate:** The user must specify the configuration included in the .INI that must be executed.
- **Simulation time:** In case the user wants to modify the simulation time set in the configuration file.
- **Scalars CSV filename:** The user can specify the name of the CSV file with simulation results that will be generated. This file is automatically saved in the "Results" folder of the OMNeT++ project.

2.6.3 Analysis tool

The analysis tool to process the simulation results will just consist of a Jupyter Notebook[3], using Python as programming language. The main Python libraries that will be used to clean and analyse the results are Pandas[4], PivotTablejs[5], MatplotLib[6], and Seaborn[7].

Through this analysis tool, different charts that may be relevant for the SRA can be obtained. For example:

- **Number of messages lost VS number of servers**. For each type of request (read measurements, read alarm, or send control command), the number of requests lost during transmission could be represented in relation to the number of servers connected to the EMPAIR.
- **Round-trip time of request-response VS BER**. For each type of request, the amount of time between sending the request and receiving a response (round-trip time) can be measured and represented in relation to the channel BER. The representation of the statistical distribution could be interesting in this case (e.g., through a boxplot chart).
- **Round-trip time of request-response VS number of servers.** In the same way, the round-trip time could be represented in relation to the number of servers.

[3] https://jupyter.org/
[4] https://pandas.pydata.org/
[5] https://pivottable.js.org/examples/
[6] https://matplotlib.org/
[7] https://seaborn.pydata.org/

- **Average polling loop time VS number of servers.** As commented in section 2.5, an operational requirement is that the EMPAIR polls all the devices in under a minute (to obtain alarm values every minute for each device). Therefore, the average time that it takes the EMPAIR to poll all the servers can be obtained and represented in relation to the number of servers.

This analysis tool is not integrated in the GUI described in the previous subsection. The Jupyter Notebook is a GUI-like tool where results can be easily visualized by just running the code for each .CSV file of results. It also provides more flexibility to the scalability and replicability analysts, since it can be easily modified (e.g., to add a chart not considered in the previous list), and analysts can include formatted text using markdown language. This way, for example, there could be a notebook per scenario where interesting insights and conclusions could be directly written for each chart, easing the process of writing the SRA report in T9.7.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Chapter 3

Turku ICT Toolkit

Chapter 3 – Turku ICT Toolkit

3.1 Scope

As in the case of Dijon, the most ICT-related IEs implemented in Turku would be mainly those grouped under TA#4. Table 6 shows the IEs demonstrated in Turku under TA#4 considered in D3.1 (Tryferidis et al., 2021)

Table 6 Innovative Elements demonstrated in Turku under IS 4.1 and IS 4.2

IS 4.1 City information Platform-enable innovations
IE 4.1.6 Smart City Knowledge Graph AI
IE 4.1.7 Air-quality Journey planner application for cyclists and pedestrians
IE 4.1.8 District heating, cooling, and flexibility control, situational awareness and anomaly detection
IE 4.1.9 Automated driving and vehicle-to-vehicle communication of robot cars via 5G
IE 4.1.10 5G Smart City lighting poles
IS 4.2 E-mobility integration into the Grid and City Planning
IE 4.2.4 Fast V2G charging station
IE 4.2.5 Light Electric Vehicle charging Hubs
IE 4.2.6 EV sharing scheme

Similar to Dijon, not all the IEs shown in Table 6 are appropriate for a simulation-based analysis to assess their scalability and replicability potential.

IE 4.1.6 Smart City Knowledge Graph AI, mainly consists of a graph database where real-world concepts and smart city data are represented by interconnected nodes. This way, the solution will integrate data from public sources and from RESPONSE participant systems (e.g., data from IE 4.1.8, 4.1.9, 4.1.10 and other smart building data) to be used as input for Artificial Intelligence (AI) algorithms, such as the journey planner in IE 4.1.7 (Deliverable 3.1). The Smart City Knowledge Graph AI will be a cloud-based Software-as-a-Service (SaaS), and data ingestion and access would be done through standardised Application Programming Interfaces (APIs) on the Internet. This means that, from the communication point of view, this solution is quite complex to be replicated by a simulation model (e.g., simulation of internet delays). Furthermore, a quantitative SRA of this solution in T9.7 would not be of much utility, since cloud-based solutions usually have great scalability capabilities. Likewise, the journey planner in IE 4.1.7 would not be appropriate for a simulation-based SRA. Therefore, both IE 4.1.6 and 4.1.7 were not included under the scope of the Turku ICT SRA toolkit.

IE 4.1.8 District heating, cooling, flexibility control, situational awareness and anomaly detection would involve different systems implemented during the project, such as the smart city knowledge graph previously described, the indoor conditions monitoring system provided by eGAIN, the building automation system for the operation of Tyyssija building thermal energy system (provided by HögforsGST), and the VTT optimisation algorithm. Figure 14 shows a first draft of the ICT topology for IE 4.1.8. Communication links with a fifth system is included, the FiksuHub, which is not included in the project's budget and whose feasibility is still being studied by Turku partners, so the topology shown in Figure 14 may change during the development of the project.

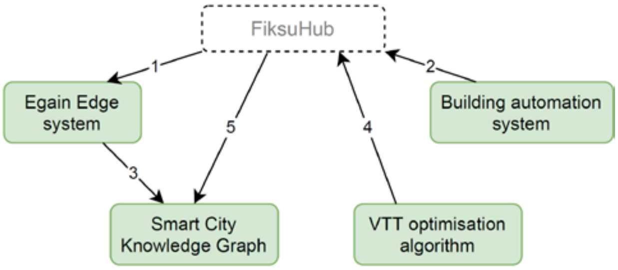

Figure 14 Draft of the ICT topology for IE 4.1.8

The estimated characteristics of the communication links in Figure 14 are shown by Table 13 in Appendix C. Due to the current state of this IE, some aspects, such as the protocol for link 2 or the frequency of information exchange for links 1, 2, 3, and 5, could not be estimated and are yet to be defined. However, the information available for the rest of the links can be used to determine its adequacy for a quantitative analysis. Starting with links 1, 3 and 5, an HTTPS-based REST API through the Internet would be used. As previously commented when describing IE 4.1.6, the Internet delays affecting the communications through an HTTPS-based REST API would be extremely complex to simulate with accuracy, despite the frequency of information exchange set could probably pose a challenge under certain conditions. For link 4, in addition to be using communications through the Internet, the frequency of information exchange set is quite affordable for the communication technology used, not posing an immediate threat to its scalability.

Therefore, the communications between system in IE 4.1.8 were not included under the scope of the Turku ICT SRA toolkit. However, it was noticed that the eGAIN's indoor conditions monitoring system, involved in this innovative element as shown in Figure 14, could be appropriate to be included in the simulation toolkit.

The **eGAIN's indoor conditions monitoring system** would consist of three elements connected as shown in Figure 15: the Edge Cloud where the data is stored and communicated to the relevant systems as previously indicated in Figure 14; the Edge Hub would be placed at a building level to act as an interface between the Edge Cloud and the sensors; and the Edge Sense devices, which are the IoT sensors installed in dwellings to monitor indoor conditions.

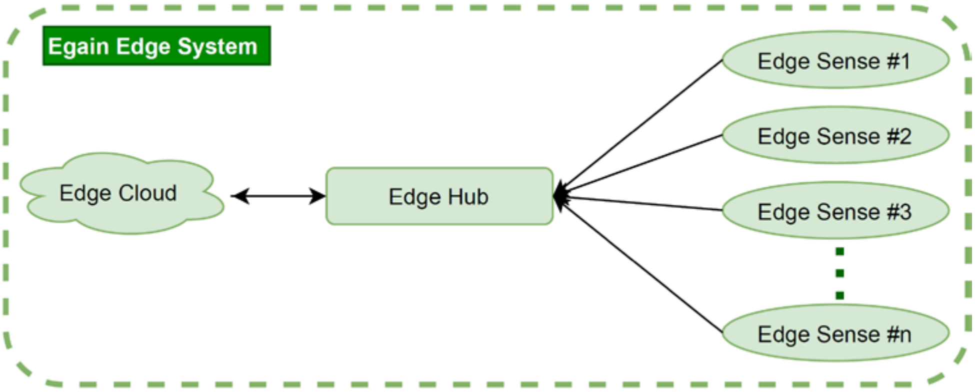

Figure 15 ICT topology of the devices forming the eGAIN Edge System.

To communicate with the Cloud, the Edge Hub uses communications through the Internet. However, to collect the data from the IoT sensors, the wireless M-BUS protocol would be implemented. In contrast to Internet communications, this type of communication can be simulated with more accuracy. In addition to this, since the Edge Hub will be collecting data from tens of devices, it is a potential information bottleneck that may have an impact on the scalability of the solution if the number of sensors or their distance to the Hub increases.

Based on this information, the system formed by the Edge Hub and the Edge Sense devices, which is involved in IE 4.1.8 (see Figure 14), was considered appropriate and interesting for a future simulation-based SRA and, therefore, included under the scope of the Turku ICT SRA toolkit. More details about this system and its model for the toolkit are provided in following sections of this chapter.

IE 4.1.9, Automated driving and vehicle-to-vehicle communication of robot cars via 5G, aims to demonstrate the application of 5G on automated driving in Turku's PED area. As shown by Figure 16, Robot cars would exchange information between them and with the interchange node, which would communicate with Turku City Data to enrich the Smart City Knowledge Graph platform (IE 4.1.6).

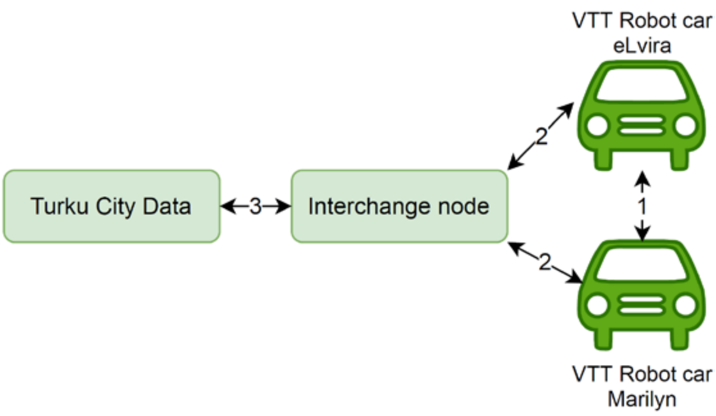

Figure 16 Draft of the ICT topology for IE 4.1.9

The estimated ICT characteristics for this IE are shown by Table 14 in Appendix C. Based on these characteristics, IE 4.1.9 would not be appropriate for a simulation-based SRA (although it could be for a qualitative SRA) due to two main reasons:

- The use of 5G technology to connect robot cars and the interchange node. The main characteristics of 5G are very low latencies, large data volumes, reliability and adequacy for devices that are moving. This means that to reach the operational limit of the communications, in all likelihood the parameters that configure the scenario would be unreasonable and unfeasible in a real implementation. An SRA should only analyse scenarios that are possible in practice. In addition to this, the scalability and replicability of these communications depend on the extension of the 5G infrastructure implemented.
- The communications between robot cars through short-range communications (ITS-G5) would not really face a scalability challenge. The reason is that these communications would only be established between vehicles that are close to one another, such as in the same block or crossroad. This means that the number of simultaneous communications established by one robot car with other vehicles is very limited by the number of such vehicles inside its communication radius. In practice, this number may never be high enough to saturate the communication link.

Therefore, due to the previous reasons, IE 4.1.9 was not included under the scope of the Turku ICT SRA toolkit.

IE 4.1.10, 5G Smart City lighting poles, was also discarded for the Turku ICT SRA toolkit as the main purpose of this IE is not to produce or receive information, but to provide a telecommunications infrastructure (5G) for different functions monitored in the project, such as IE 4.1.9.

Regarding the IEs under IS 4.2 (E-mobility integration into the Grid and City Planning), these have been discarded for the ICT toolkit for different reasons. IE 4.2.4 Fast V2G charging station would implement ISO 15118-20 and the new version 2.0.1 of the OCPP communication standards to communicate between V2G points and a local controller. However, according to D3.1, the implementation of these standards is a bit uncertain. IE 4.2.5 Light Electric Vehicle charging Hubs is not expected to face a scalability challenge regarding local communications between charging points and a controller: the number of charging point per hub is usually reduced and they are not always being used at the same time. Finally, IE 4.2.6 EV sharing scheme will be implemented using a booking app, so it would not be appropriate for a simulation-based SRA.

In addition to the IEs shown by Table 6 at the beginning of this section, the possibility of including the ICT involved in the operational optimisation of the energy operations in Tyyssija building, the LVDC microgrid, and the AC-connected elements in Aitiopaikka building was considered:

- **Operational optimisation of the energy operations in Tyyssija building.** This solution is a combination of different IEs: smart district heating substation (IE 2.1.8), VTT optimisation algorithm (IE 2.2.5), dynamic district heating tariffs (IE 2.2.7), and Ferroamp EMS (IE 2.2.8). Figure 26 shows a draft of the ICT topology for the operational optimisation of the energy operations in Tyyssija building, and Table 15 in Appendix C shows the estimated characteristics of the communication links in such topology.

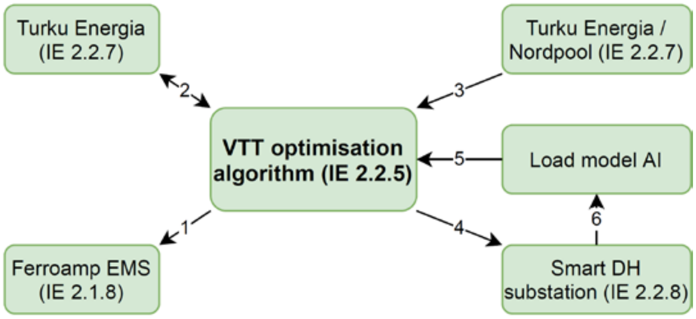

Figure 17 Draft of the ICT topology for the operational optimisation of the energy operations in Tyyssija building in Turku

The use of HTTPS-based REST API through the Internet makes the communication network shown in Figure 17 difficult to be simulated with accuracy. In addition to this, the frequency of information exchanges presented in Table 15 in Appendix C are not expected to put the communication links under stress, leaving much room for scalability. Therefore, this solution was not included under the scope of the Turku ICT SRA toolkit.

- **LVDC microgrid.** The LVDC microgrid implemented in Turku will involve multiple systems and devices (e.g., EMS, V2G chargers, inverters, etc.) connected in the Tyyssija and retrofitted buildings. These systems and devices will be connected not only from the electrical point of view but from the communications perspective so that the microgrid can be operated in a safe, reliable, and efficient manner.

  Usually, devices used in power systems, such as inverters or V2G charges, do not use Internet-based communications to exchange monitoring and control information. This is the main reason why this solution could be very attractive for a simulation-based ICT SRA, and thus be included in the simulation toolkit developed. However, at the moment of publishing this deliverable, there are still many uncertainties about the ICT characteristics of this IE, since many key system components have not been selected yet by the responsible partners.

  Figure 18 shows the estimated ICT topology for the LVDC microgrid. Information such as the frequency of information exchange and the size of the messages is still unknown; other characteristics, such as the communication protocol and technology implemented are estimations:

- The communication technology that will probably be used in the blue links in Figure 18 (links 1, 2, 22-24) is Ethernet. In the case of link 1, the communication protocol is still unknown, although HTTPS, MODBUS TCP, and MQTT are being considered. For links 22-24, the communication protocol, in all likelihood, will be HTTPS used in a REST API.
- The green links in Figure 18 (links 3-20) would communicate elements provided by partner Ferroamp through narrowband Power Line Communications (PLC).
- The orange link in Figure 18 (link 21) connecting the V2G system with Ferroamp EMS is estimated to be MODBUS TCP over Ethernet. However, this is still to be determined.

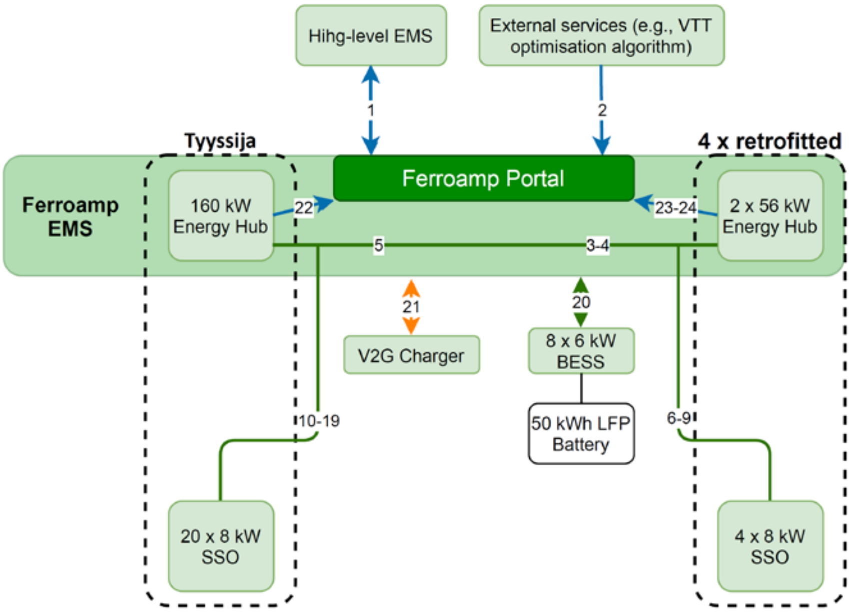

Figure 18 Draft of the ICT topology for the LVDC microgrid implemented for the Tyyssija and retrofitted buildings

Some of the protocols estimated to be used in the communications of some of the elements of this solution are appropriate for a simulation-based ICT SRA. In addition to this, it is considered that such analysis could be very interesting and some synergies with the simulation model developed for Dijon could appear (e.g., use of the same communication protocol, Modbus TCP). However, the number of uncertainties at this stage is still high to determine if any part of this solution will be included under the scope of the Turku ICT SRA toolkit. For this reason, its possible inclusion will be evaluated when more accurate information is available and, if finally included, described in D9.13.

- **AC-connected elements in Aitiopaikka building.** Elements installed in Aitiopaikka building, such as the second-life Battery Energy Storage System (BESS) and the fast V2G charging station, will need to communicate with local generation equipment (i.e., solar PV inverters) to operate.

  Figure 19 shows the estimated ICT topology for these elements. As in the case of the LVDC microgrid, the information about the ICT involved is still an estimation. Communications with the EMS of the second-life BESS (links 1-6) will probably implement the MODBUS RTU communication protocol, using a RS-485 serial interface. However, this is still to be determined.

Regarding the communications between the V2G charging points and the charging service provider, they will probably implement HTTPS through an Ethernet connection.

As in the case of the LVDC microgrid, the possible inclusion of any part of this solution in the scope of the Turku ICT SRA toolkit will be evaluated when more accurate information is available and, if finally included, described in the second deliverable of this task, D9.13.

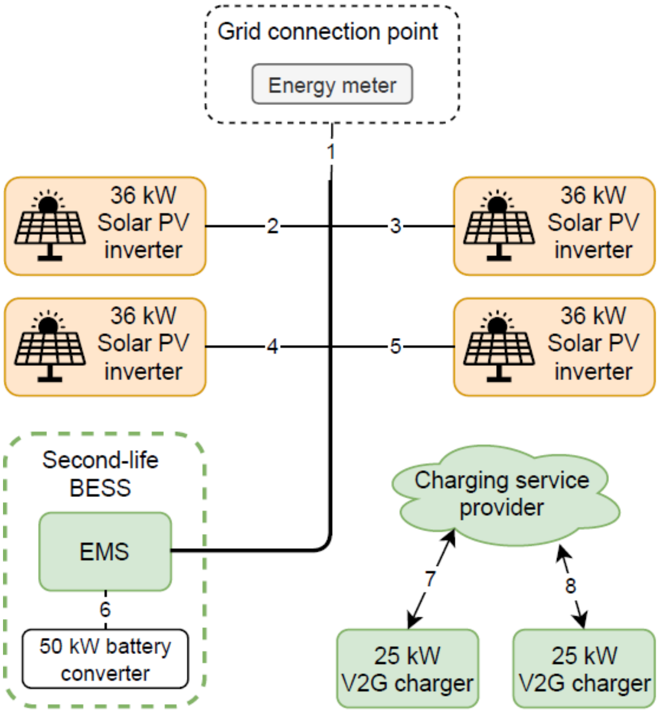

Figure 19 Draft of the ICT topology for the AC-connected elements in Aitiopaikka building

In summary, the ICT SRA toolkit for Turku focuses on the eGAIN's indoor conditions monitoring system, which is involved in IE 4.1.8, as shown in Figure 14. The inclusion of ICT systems involved in the LVDC microgrid and the AC-connected elements in Aitiopaikka building will be evaluated when more accurate information is available. Table 7 summarises the evaluation of the different IEs for the scope of the Turku ICT SRA toolkit.

Table 7 Summary of the evaluation of the IEs for the scope of the Turku ICT SRA toolkit.

IS 4.1 City information Platform-enable innovations	Consideration for ICT Toolkit
IE 4.1.6 Smart City Knowledge Graph AI	**Not included.** • Cloud-based (scalable). • Internet communications (inaccurate simulations)
IE 4.1.7 Air-quality Journey planner application for cyclists and pedestrians	**Not included.** • Mobile app. • Internet communications (inaccurate simulations)
IE 4.1.8 District heating, cooling, and flexibility control, situational awareness and anomaly detection	**Only eGAIN's indoor condition monitoring system** • In general, communications between systems in IE 4.1.8 are based on HTTPS REST APIs (internet communications). The frequency of some information exchanges will not pose a scalability challenge. Not considered for the toolkit. • eGAIN's system based on local wireless M-Bus communications. Scalability challenge: one hub collecting data from several sensors frequently. Considered for the toolkit.
IE 4.1.9 Automated driving and vehicle-to-vehicle communication of robot cars via 5G	**Not included.** • Use of 5G technology. Very high ICT scalability expected. • Short-range communications (ITS-G5) would not face a scalability challenge.
IE 4.1.10 5G Smart City lighting poles	**Not included.** • It just provides a 5G telecommunications infrastructure for other solutions.
IS 4.2 E-mobility integration into the Grid and City Planning	
IE 4.2.4 Fast V2G charging station	**Not included.** • Communication standards under development. • Implementation details still in an uncertain state.
IE 4.2.5 Light Electric Vehicle charging Hubs	**Not included.** • Communications will not face a scalability challenge: reduced number of chargers, which are usually not used simultaneously.
IE 4.2.6 EV sharing scheme	**Not included.** • Mobile app.

3.2 Architectures and technologies

Figure 20 shows the ICT topology of the system included under the Turku ICT SRA toolkit scope: the eGAIN's indoor conditions monitoring system, which is involved in IE 4.1.8.

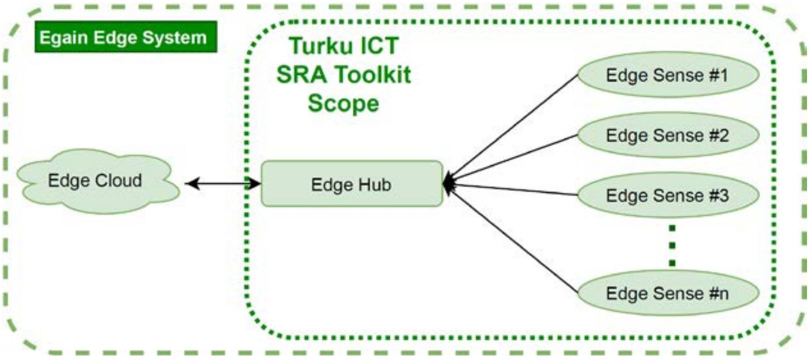

Figure 20 ICT topology of the eGAIN's indoor conditions monitoring system

The simulation model will focus on the communications between the sensors (Edge Sense devices) and the Edge Hub through wireless M-Bus. Both the Edge Hub and the Edge Sense devices are products commercialised by eGAIN:

- The **Edge Hub** device is a building access point with Global System Mobile (GSM) and wireless M-Bus connectivity, which allow it to collect data from wireless M-Bus sensors and interact with cloud-based services. It implements high performance filtering to avoid disturbances caused by mobile phones (eGAIN, 2020a).
- The **Edge Sense** device is a wireless sensor installed in apartments to measure temperature and humidity. It sends the encrypted information about these parameters several times per hour to the Edge Hub using wireless M-Bus (eGAIN, 2020b).

Following subsection briefly describes the wireless M-Bus protocol involved in the eGAIN's indoor conditions monitoring system, as it will be replicated in the simulation toolkit.

3.2.1 Wireless M-Bus

M-Bus or Meter Bus is a lightweight communication protocol for Advance Metering Infrastructures (AMI) that describes the Open Metering System (OMS) requirements for communications between a meter device (e.g., sensor) and "other" device (e.g., data concentrator) (OMS, 2016).

The wireless M-Bus protocol stack is shown by Figure 21. It must be remarked that, although layers 3-6 of the OSI model are not defined and implemented explicitly, the standard EN 13757-3:2013, which defines the application layer, also includes the definition of the transport layer. The conditions under which the transport layer should be applied are described in the OMS specification (OMS, 2016)

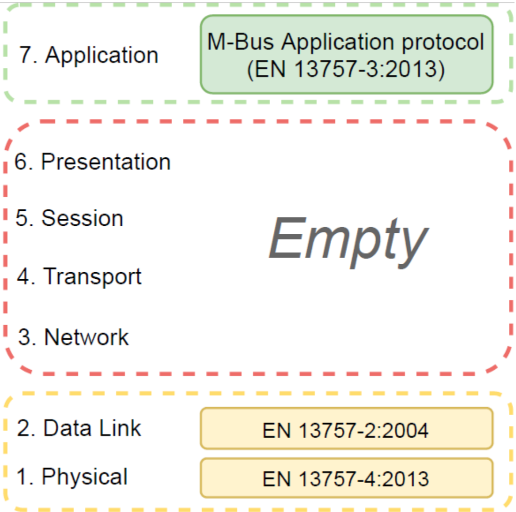

Figure 21 Wireless M-Bus protocol stack

A wireless M-Bus network will follow a star topology where a unique device collects the data from several sensors or meters. Up to six different transfer modes are defined by EN 13757-4:2013 for M-Bus: S-Stationary, T-Frequent Transmit, R-Frequent Receive, C-Compact, N-Narrowband VHF, and F-Frequent Receive and Transmit. For some of these modes, some sub-modes are also defined depending on whether the communication is unidirectional or bidirectional. Table 8 summarises the characteristics of the wireless M-Bus transfer modes.

Table 8 Summary of wireless M-Bus transfer modes.[8]

Mode	Frequency band	Bit rate	Brief description of use
S-Stationary	868 MHz	16,384 kbps	Meters send data several times a day to a stationary/mobile concentrator.
T-Frequent Transmit	868 MHz	66,67 kbps	Meters send the data every few seconds to a walk-by or drive-by data collector
R-Frequent Receive	868 MHz	2,4 kbps	Each meter sends the data to the collector in a different frequency channel to avoid interferences
C-Compact	868 MHz	50 or 100 kbps	Similar to T, but it takes less energy to transmit the same information
N-Narrowband VHF	169 MHz	2,4; 4,8; 6,4; or 19,2 kbps	For a long range, narrowband system
F-Frequent Receive and Transmit	433 MHz	2,4 kbps	Bidirectional communication with NRZ encoding

The structure of a frame transmitted through wireless M-Bus is shown in Figure 22. It basically consists of a preamble (header + sync) and a payload that depend on the packet format implemented, A or B, defined by

[8] Sources: (Mohan, 2019; STMicroelectronics, 2015; Zeman et al., 2017)

the EN 13757-4:2013. Format A can be used in any of the wireless M-Bus modes listed by Table 8, whereas format B can only be optionally used in modes C, N, and F (Lavra, 2016).

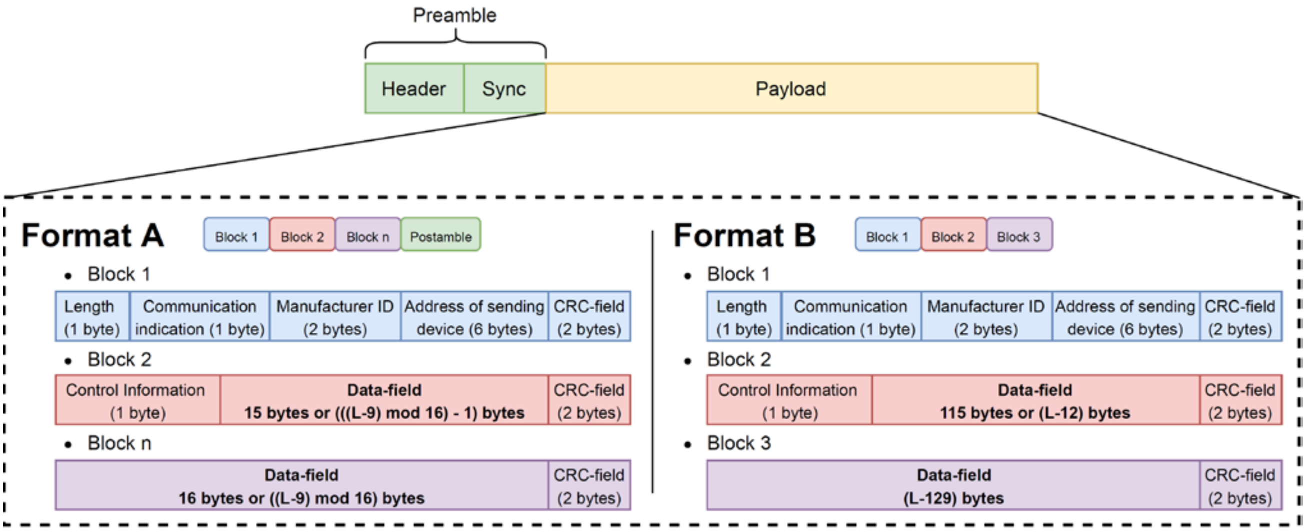

Figure 22 Packets that form a message transmitted with wireless M-Bus. Breakdown of frame format A and B.

As Figure 22 shows, the main difference between format A and B is in the data field. In addition to this, format A implements a postamble, consisting of a short bit sequence, for modes S, T, and R2. Table 9 shows the length of the preamble depending on which mode and format are used in the wireless M-Bus implementation (Lavra, 2016).

Table 9 Length of the preamble depending on which mode and format are implemented

Mode	Format A	Format B
S	6 bytes if short preamble 72 bytes if long preamble	
T	6 bytes	
R2	12 bytes	
C	8 bytes	8 bytes
N	Configurable length (EN 13757-4 sets it to 2 bytes)	4 or 8 bytes (depending on modulation)
F	12 bytes	12 bytes

3.3 SGAM mapping

As in the case of Dijon (section 2.3), the eGAIN's indoor conditions monitoring system considered for the Turku ICT SRA toolkit was mapped to PED/PEB-modified version of the SGAM (Figure 23).

The wireless M-Bus data collection system that would be placed at PEB level, more specifically, at Turku's cluster B4 (100 dwellings), according to the GA. The Edge Hub would be placed at a building/block level (station zone), since the coverage for wireless M-Bus communications is expected to be between 0.5-1km. Each dwelling in the cluster would have an Edge Sense device installed (process zone).

The Edge Cloud, which is out of the scope of the toolkit, is in charge of the communication with the different Edge Hubs installed and analyse the data collected by these. Therefore, as shown by Figure 23, the Edge Cloud would be working at a PED level.

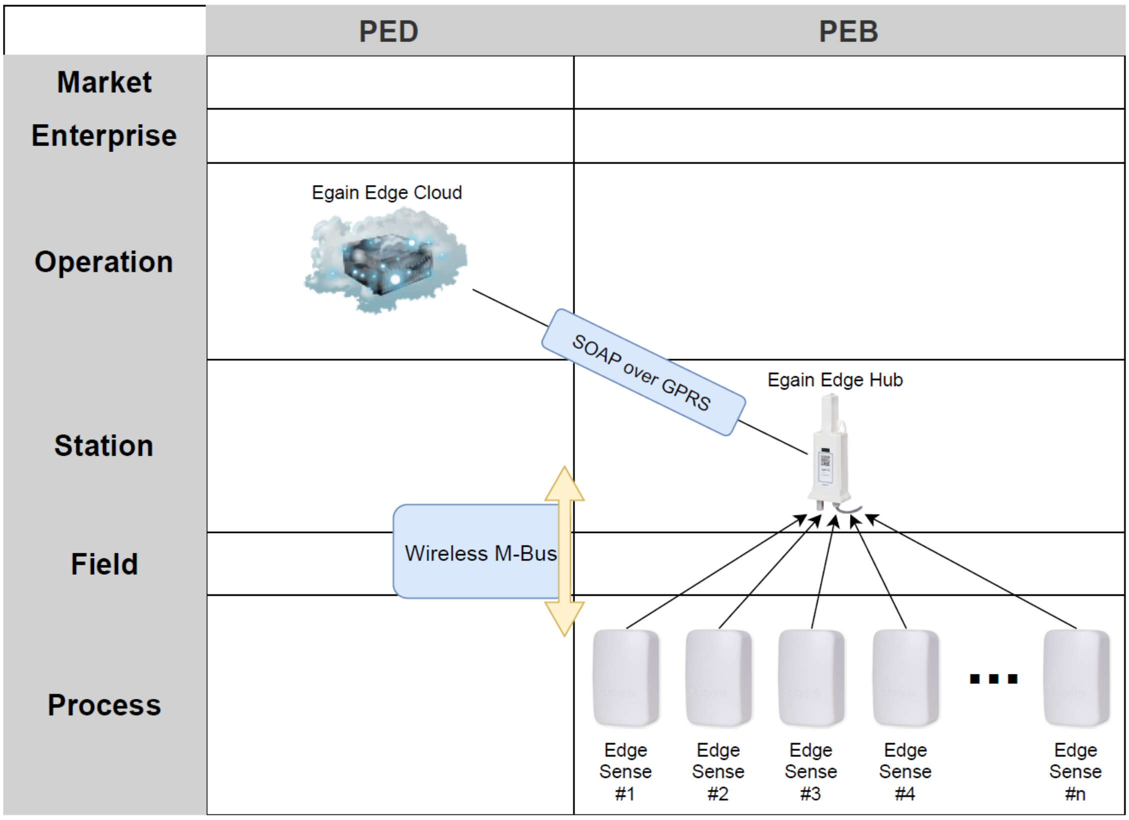

Figure 23 Mapping of the eGAIN's indoor conditions monitoring system under the toolkit scope to the six SGAM zones and PED/PEB domains. Component-communication layers.

As similarly done with IE 4.1.1 in section 2.3, the analysis of Figure 23 allows the formulation of similar questions regarding scalability and replicability for the definition of parameters in the following section. However, in this case, it must be noted that the system uses a wireless protocol, instead of a wired one:

- **What would be the effect of placing the Edge Hub in the PED domain?** This would mean increasing the distance between the Edge Sense devices and the Edge Hub. Since the communications are wireless, the maximum distance is expected to be between 0.5 and 1 km, as previously commented.
- **What would be the effect of increasing the number of Edge Sense devices connected to the same Edge Hub?** Since modifying the distance will be very limited by the wireless communication, increasing the number of sensors connected to a single Edge Hub could pose a significant challenge: the wireless medium is shared by all the sensors and all of them need to send their measurements at a minimum interval.

  By determining the maximum number of devices that can be connected to the Edge Hub, the placement of this data collector could be optimised at a PED level in the future, so that all the blocks can be covered, and, this way, optimise the investment.

The SGAM mapping in Figure 23 is complemented by Figure 24 with a summary of the network regarding the five interoperability layers of SGAM.

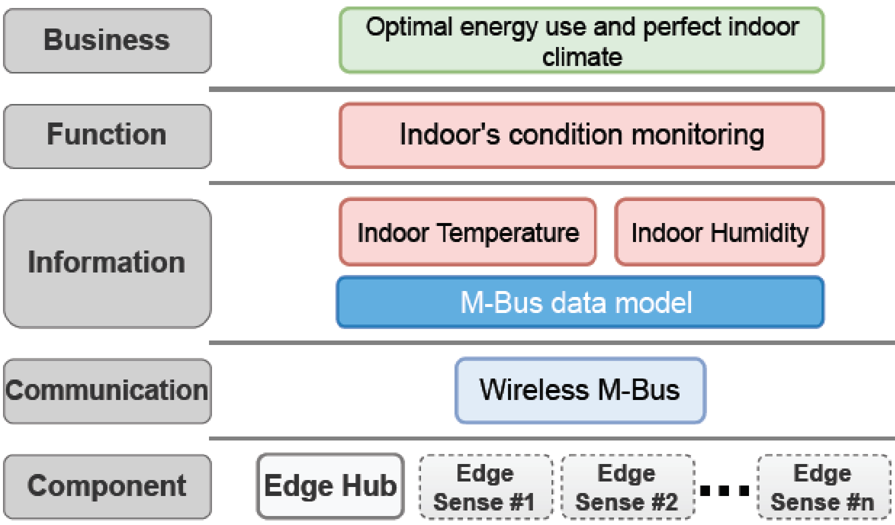

Figure 24 Mapping of the eGAIN's indoor conditions monitoring system to the five interoperability layers of SGAM.

At the business layer, the objective of the indoor conditions monitoring system is to provide the data necessary to optimise the energy use of the buildings while keeping the dwelling's indoor climate at a perfect level. At the function layer, the main objective of the system considered is, therefore, to monitor the indoors condition, which is represented by two parameters: the indoor temperature and the indoor humidity.

As shown in Figure 23 and Figure 24, there would be only one unidirectional information exchange taking place: the Edge Sense devices sending temperature and humidity information to the Edge Hub through wireless M-Bus. Details about how this information exchange is modelled is described in section 3.5.

3.4 SRA parameters

The SRA parameters that will be possible to modify in the Turku ICT SRA toolkit (simulation of eGAIN indoors condition monitoring system) are the following:

- **Nº of devices connected.** Since the monitoring system will be implemented in Turku's cluster B4, which has 100 dwellings, the initial number of sensors could be set to 100 if simulation tests show that a single Edge Hub can cope with this number. Based on this, the SRA will consider scenarios where the number of devices will vary in order to assess the scalability of the system.
- **Distance of devices.** To assess the scalability, some SRA scenarios may need to vary the distance at which the sensors are place with respect to the Edge Hub. Since wireless communications have a determined coverage radius, the analysis of how the network performs depending on the average distance of devices and the number of devices could be interesting for the SRA.
- **Bit Error Rate (BER).** The BER is the number of wrong bits divided by the total number of bit transmitted. Despite the Edge Hub has advanced filtering, it should be possible to define specific BER values to replicate errors due to interferences.
- **Number of retransmissions.** The M-Bus specification (OMS, 2016) sets that every message should be transmitted at least twice within the data update period in order to have a 95% probability of reception by the data collector. This parameter may affect the number of sensors that can be connected to the Edge Hub, so the SRA may consider keeping this parameter constant to simplify the analysis process. Nevertheless, the possibility of modifying it will be considered in the design of the simulation model.
- **Information size.** By modifying this parameter, it will be possible to assess the performance of the wireless M-Bus network when the amount of information transmitted increases or decreases for the same number of sensors. The baseline should be the actual size of the information, which is estimated to be 1 kB.

Table 10 summarises these potential SRA parameters, indicating the type of analysis to which they would contribute (scalability, replicability, or both), and proposing a baseline value (i.e., "default" value defined in the simulation model). These SRA parameters listed, however, do not necessarily mean that they all will be considered by the SRA scenarios in T9.7, but that it will be possible to easily modify these parameters in the ICT toolkit if necessary.

Table 10 Potential SRA parameters to develop different scenarios for the indoor condition monitoring system in Turku

Parameter	Type	Proposed baseline
Nº of devices connected	Scalability	100 (if feasible)
Distance of devices	Scalability	20 meters
Bit Error Rate (BER)	Replicability / Scalability	0
Number of transmissions	Replicability / Scalability	2 (minimum for 95% probability of reception)
Information size	Replicability / Scalability	1 kB

3.5 Characteristics and requirements

The unique functional task (indoors condition monitoring) considered for the Edge Hub would only involve one type of information exchange which will receive the generic name of “Indoor measurements”. The main characteristics of this information object would be:

- It would contain both the temperature and humidity measurements. The information size per message was estimated by eGAIN to be 1kB.
- It would be sent, in a unidirectional way, by the Edge Sense to the Edge Hub.
- New measurements would be taken every 15 minutes. However, the transmission of this information would be more often. The wireless M-Bus specification sets that, for a 95% probability of reception, each datagram should be transmitted at least twice within the update period (15 minutes). It also sets the minimum time delay between successive transmissions (from meters to data collectors) for modes S (1.8 seconds), T (0.72 seconds), and C (0,72 seconds) (OMS, 2016).

Table 11 summarises the characteristics and assumptions made for the “Indoor measurements” information exchange.

Table 11 ICT Characteristics and assumptions of the indoors condition monitoring system for the ICT SRA toolkit of Turku

Source	Receiver	Protocol	Information object	Frequency of exchange	Message size (Bytes)	Communication technology
Edge Sense	Edge Hub	Wireless M-Bus	Indoor measurements	According to specification	1kB	Wireless M-Bus

In addition to the previous information, to replicate the functioning of the Edge Hub and the Edge Sense, the technical sheets of these devices (eGAIN, 2020a, 2020b) provide their wireless M-Bus characteristics to be considered during the development of the simulation tool. These characteristics are shown by Table 12.

Table 12 Wireless M-Bus characteristics of the eGAIN Edge Hub and eGAIN Edge Sense devices. Source: (eGAIN, 2020a, 2020b)

<table>
<tr><th colspan="2">Edge Hub</th><th colspan="2">Edge Sense</th></tr>
<tr><td>Standards</td><td>EN 13757-3/4:2013, OMS 4.0.2</td><td>Standards</td><td>EN 13757-3/4:2013, OMS 4.0.2</td></tr>
<tr><td>Modes</td><td>C, S, and T-mode</td><td>Modes</td><td>S-mode and C-mode</td></tr>
<tr><td>Frequency</td><td>868.3 and 868.95 MHz</td><td>Intervals</td><td>S = 5 minutes. C = 60 seconds</td></tr>
<tr><td>Sensitivity</td><td>-112 dBm S, -109 dBm T/C</td><td>Frequency</td><td>868.3 and 868.95 MHz</td></tr>
<tr><td>Receiver Class</td><td>2</td><td>RF-power</td><td>< 14 dBm</td></tr>
<tr><td rowspan="2">Antenna</td><td rowspan="2">External</td><td>Encryption</td><td>AES128 according to mode 5</td></tr>
<tr><td>Antenna</td><td>Dual Internal Diversity</td></tr>
</table>

Based on these characteristics, the wireless M-Bus communications are assumed to be using the S-mode, since the sensors would be sending data several times a day to a stationary Edge Hub (see Table 8). This means that the frame format implemented would be format A, and the time interval between retransmissions would be 5 minutes.

Regarding the topology of the communication network, a wireless M-Bus system can only implement the star topology.

3.6 Toolkit design

The approach that will be followed for the development of the Turku ICT SRA toolkit is the same as the followed in the case of Dijon (section 2.6). That is to say, the ICT toolkit for the LH city of Turku will consist of an OMNeT++ simulation model, a GUI to configure simulation scenarios, and an analysis tool, as was shown by Figure 10.

3.6.1 OMNeT++ Simulator

An OMNeT++ implementation of the wireless M-Bus protocol used by the eGAIN indoors condition monitoring system has not been found in any of the open-source libraries that extend the framework. However, the INET Framework provides open-source OMNeT++ models for radio-frequency communications (environment, basic nodes, etc.) that can be used as a basis to replicate wireless M-Bus communications.

To model both types of devices (data collector and sensor), the basis will be a basic wireless node model that implements narrowband IEEE 802.15.4 communications. As described in subsection 3.2.1, the wireless M-Bus implements its own protocol at the physical, link, and application layer. Therefore, the necessary modifications and additions to replicate the functioning of wireless M-Bus will have to be evaluated during the development of the model, since a highly detailed replication of the protocol would not be strictly necessary to obtain meaningful results and conduct an insightful analysis.

3.6.2 Graphical User Interface

The GUI for the Turku ICT SRA toolkit, as in the case of Dijon, will consist of two windows: the "configuration" window, and the "run simulation" window. This last window would be exactly the same as the one previously presented in subsection 2.6.2.2. The GUI is developed in Python.

3.6.2.1 Wireless M-Bus Simulation Configuration window

In this window, which is shown by Figure 25, the user can introduce a set of parameters for the wireless M-Bus simulation and generate an .INI configuration file that can be read by OMNeT++. By clicking on "Run" in the top bar, the run simulation window would emerge.

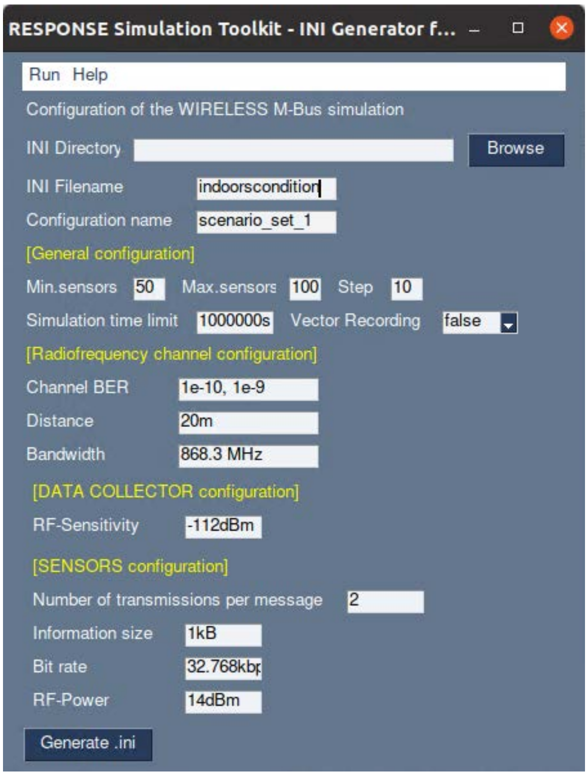

Figure 25 Screenshot of the window to create a configuration file for the wireless M-Bus simulation. Draft version.

The default values set for the radio-frequency parameters shown by Figure 25 mostly correspond to the wireless M-Bus S-mode, which was the mode assumed to be implemented by the monitoring system. The description of the simulation parameters that the user would be able to modify is provided below:

- **INI Directory:** The path where the .INI configuration file will be saved when pressing the "Generate .ini" button at the lower left corner of the window.
- **INI Filename:** The name of the .INI configuration file used to save it.
- **Configuration name:** The configuration needs a name to be distinguished from others and to execute the simulation. For example, it could be "scenarios_set_1" or, more descriptive, "IncreasingBER".

- **General configuration:**
    - **Min. Sensors:** Minimum number of sensors (i.e., Edge Sense devices) for a set of scenarios.
    - **Max. Sensors:** Maximum number of sensors (i.e., Edge Sense devices) for a set of scenarios. If it is equal to "Min. Sensors", all the scenarios in the configuration file will have the same number of sensors.
    - **Step:** Step to be considered when iterating from the "Min. Sensors" to the "Max. Sensors". This is an automated way of varying the number of sensors connected, which was an SRA parameter defined in section 3.4.
    - **Simulation time limit:** Simulation time in seconds that, once reached, will stop the execution of the simulation.
    - **Vector recording:** In case the user wants to obtain a CSV file that stores the vectors of variable indicators. By default, it is set to false, since statistics would be simpler to analyse.
- **Radiofrequency channel configuration:**
    - **Channel BER:** BER for the wireless M-Bus communications. Multiple BER values can be set, separated by commas. This was considered an SRA parameter in section 3.4.
    - **Distance:** To define the distance (in meters) between the data collector (Edge Hub) and all the connected sensors (Edge Sense devices). This was considered an SRA parameter in section 3.4.
    - **Bandwidth:** The frequency range used by the communication signal. The default value is set to 868.3 MHz, which corresponds to wireless M-Bus S-mode.
- **Data collector configuration:** This would be the configuration of the Edge Hub device.
    - **RF-Sensitivity:** This parameter sets the minimum strength of a signal to be detected by the data collector. It is defined by the Edge Hub's technical characteristics (Table 12).
- **Sensor's configuration:**
    - **Number of transmissions per message:** The number of times the same information is transmitted by the sensor. This was considered an SRA parameter in section 3.4.
    - **Information size:** The size of the information to be transmitted. It was initially estimated to be 1 kB. This was considered an SRA parameter in section 3.4.
    - **Bit rate:** Number of bits transmitted per second. The default value is set to 16,384 kbps, which corresponds to wireless M-Bus S-mode.

- **RF-Power:** This parameter set the radiofrequency power with which the sensors would transmit the information. It is defined by the Edge Sense's technical characteristics (Table 12).

It must be taken into consideration that the toolkit is still being developed and that the interface described in this subsection is not definitive.

3.6.2.2 Run simulation window

The run simulation window for the Turku ICT toolkit would be the same as for Dijon's. A screenshot of this window (Figure 13), together with the description of the information that it requires, can be found in subsection 2.6.2.2.

3.6.3 Analysis tool

The analysis tool to process the simulation results will just consist of a Jupyter Notebook, using Python as programming language. The main Python libraries that will be used to clean and analyse the results are Pandas, PivotTablejs, MatplotLib, and Seaborn.

Through this Jupyter Notebook, different charts and indicators that may be relevant for the SRA can be automatically obtained by just running the code inside. For example:

- **Unique data received / unique data sent ratio (Delivery ratio).** Since the same information is being transmitted a few times to guarantee delivery, this indicator would allow to see if the multiple-transmissions mechanisms are effective. This means that, within the simulation model, the data collector must count the reception of the same information only once. The delivery ratio could be calculated to generate useful charts.
- **Delivery ratio VS number of sensors.** To analyse the influence of the number of sensors in the delivery ratio defined before, so that the maximum number of sensors that could be connected, while guaranteeing the good functioning of the monitoring system, can be determined.
- **Delivery ratio VS number of transmissions.** The wireless M-Bus sets that every piece of information should be transmitted at least twice within the data update period in order to have a 95% probability of reception. It would be interesting to explore the relationship between this probability and the number of transmissions for a given number of sensors, since it may affect its scalability potential.

As in the case of Dijon, this analysis tool is not integrated in the GUI described in the previous subsection so that the analysts have more flexibility when conducting the SRA in T9.7.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Chapter 4

Conclusions

Chapter 4 – Conclusions

The objective of T9.6 is to set the foundations for the ICT Scalability and Replicability Analysis (SRA) toolkit, to be used in task T9.7. The present document, D9.6, has focused on determining which IEs are appropriate for the ICT scalability and replicability evaluation toolkit, on describing their characteristics and requirements, and on presenting the design of the toolkit.

For this deliverable, the methodology adopted consists of four steps. First, the simulation-based ICT SRA scope was identified, selecting the systems or IEs that would be included. Then, for the selected ICT systems, the characteristics and requirements of devices and links were determined. Once this information was analysed, critical links and nodes were identified by mapping the ICT systems into an adapted version of SGAM that considers smart city energy domains (i.e., PED, PEB), and which enabled the definition of possible SRA parameters and questions. Finally, the toolkit design and software to be used in the development was presented for each LH city.

The core of the ICT toolkit will be the simulation models of the communication systems implemented in the selected IEs. Two main criteria for the selection of IEs have been evaluated: the complexity of developing an accurate simulation model, and how significant the results of such a simulation could be for the quantitative SRA to be carried out in T9.7. In general, those IEs whose communications are expected to be through the Internet were discarded for the simulation toolkit, as it would imply simulating a too simple model of such a large network. This type of simulation would heavily depend on the statistical distribution used to model the Internet delay, so the simulation results would not be very insightful nor reliable for the SRA.

Based on these criteria, for Dijon, the toolkit will focus on IE 4.1.1 GENESYS tunnelling solution, more specifically on the communications between the EMPAIR device and the different energy assets installed at a building level (e.g., EMS, solar PV data loggers) for the optimization of self-consumption. For Turku, the toolkit will focus on part of the eGAIN’s indoor conditions monitoring system: the communications between a data collector at a building level and several sensors in apartments. This system is one of the systems involved in IE 4.1.8 (District heating, cooling, and flexibility control, situational awareness and anomaly detection). Regarding the discarded IEs, the ICT information collected for these may be equally valuable for a qualitative SRA in T9.7.

For both LH cities, the simulation framework used is OMNeT++. Since this simulation framework is a bit complex to use, the ICT SRA toolkit design presented in this deliverable includes a Graphical User Interface to make it user-friendly and flexible.

The simulation results that will be provided by the toolkit, consisting of a single CSV file per simulation execution, can be easily and rapidly analysed with the Jupyter Notebook that will be provided as an analysis tool within the developed toolkit. Some of the charts that will be possible to automatically generate through this analysis tool have been mentioned in this document. Despite this, it may be necessary to include some other charts when the toolkit is tested in the second phase of the task, which will be described in D9.13. Nevertheless, the SRA analysts will still have the possibility and flexibility of including additional metrics and visualizations by using the Python programming language, which is widely used in data science.

For the second version of this deliverable, D9.13, the focus will be on completing the description of the ICT toolkit, proposing SRA scenarios to be considered in T9.7, and describing the unitary tests carried out to validate the toolkit.

Regarding the upcoming T9.7, the analysts involved in the SRA (D9.7 and D9.14, RESPONSE ICT-SRA and Replicability/Scalability of the demonstrated TAs at EU level) will benefit of the aforementioned graphical user interface that will allow them to easily modify simulation parameters, to configure multiple scalability and replicability scenarios, and to run such scenarios as a batch process, instead of one-by-one. All this, without the actual need of having an advanced knowledge of the simulation framework, C++, or NED language.

In summary, the ICT SRA toolkit presented in this deliverable will play an important role in the quantitative SRA that will be conducted in the upcoming T9.7. All the information provided in this deliverable aims at achieving a toolkit that will provide the necessary data to analyse, in a quantitative and thorough way, the impact of ICT on the scalability and replicability potential of some solutions that belong to the Smart City environments implemented by RESPONSE.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Bibliography

Bibliography

eGAIN. (2020a). *eGAIN Edge Hub Technical Sheet*. eGAIN Group. https://wwweGAINio.cdn.triggerfish.cloud/uploads/2020/06/eGAIN-techspecs_913-edge-hub_version-a_2020-1.pdf

eGAIN. (2020b). *eGAIN Sense Technical Sheet*. eGAIN Group. https://wwweGAINio.cdn.triggerfish.cloud/uploads/2020/06/eGAIN-techspecs_eGAIN-sense-908_version-a_2020-1.pdf

Lavra, F. (2016). *Telit Wireless M-Bus 2013 Part 4 User Guide*. Telit Communications.

Modbus Organization. (2012). *Modbus application protocol specification V1.1b3*.

Mohan, V. (2019). *An Introduction to Wireless M-Bus*. Silicon Labs.

OMS. (2016). *Open Metering System Specification Vol.2*.

PROFINET University. (2019, April 17). *PROFINET Communication Channels*. https://profinetuniversity.com/profinet-basics/profinet-communication-channels/

PROFINET University. (2020, June 2). *PROFINET Head to Head—Modbus TCP*. https://profinetuniversity.com/profinet-features/profinet-head-to-head-modbus-tcp/

STMicroelectronics. (2015). *WM-Bus firmware stack overview*.

Tryferidis, T., Tsompanidou, E., Boutopoulou, Z., & Ioannidis, D. (2021). *Master City plans for TA#4 Integrated and Interconnected City Ecosystems v1.0* (D3.1; p. 296). H2020 RESPONSE.

Zeman, K., Masek, P., Krejčí, J., Ometov, A., Hosek, J., Andreev, S., & Kröpfl, F. (2017). Wireless M-Bus in Industrial IoT: Technology Overview and Prototype Implementation. *European Wireless 2017: 23th European Wireless Conference*, 1–6.

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

Appendixes

Appendixes

Appendix A. PROFINET protocol

PROFINET[9] is an open Industrial Ethernet standard that defines an application-layer communication protocol that follows a provider/consumer model. This means that devices can be simultaneously providers and consumers of data, using full-duplex Ethernet (PROFINET University, 2020).

PROFINET can use three different channels, with a different OSI protocol stack (Figure 26), depending on the needs of the data exchange taking place (PROFINET University, 2019): non-real-time channel (NRT), real-time (RT) channel, and isochronous real-time (IRT) channel.

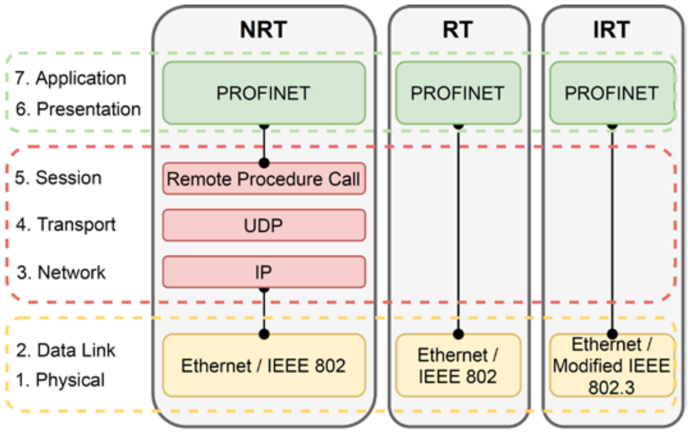

Figure 26 PROFINET OSI Protocol stack for each channel

- **NRT channel.** The communications through this channel have higher latency and jitter, since it uses all the OSI layers. Therefore, the NRT channel is appropriate for no time-sensitive communications. This would be PROFINET channel considered as an alternative for the Modbus/TCP implementation in the Dijon ICT toolkit. Figure 27 depicts the packets that would form a message transmitted with PROFINET's NRT channel (PROFINET University, 2019).

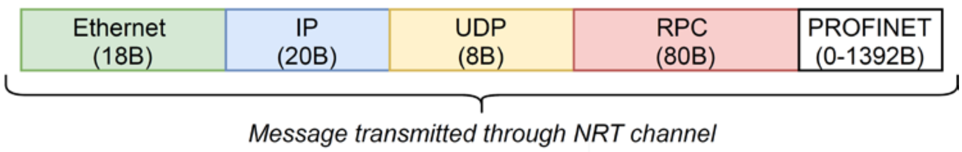

Figure 27 Packets that form a message transmitted with PROFINET's NRT channel

- **RT channel**. This channel removes the steps in the Network, Transport, and session layers. As a result, latency and jitter decreases at the expense of not having an IP address to route frames between LANs. Figure 28 depicts the packets that would form a message transmitted with PROFINET's RT channel (PROFINET University, 2019).

[9] https://us.profinet.com/

- **IRT channel.** This channel only differs from the RT channel in the physical layer, where a modified version of IEEE 802.3 is used. Figure 28 depicts the packets that would form a message transmitted with PROFINET's IRT channel (PROFINET University, 2019).

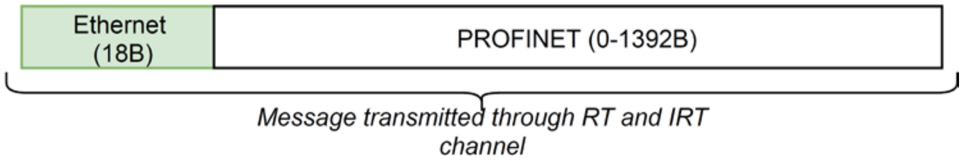

Figure 28 Packets that form a message transmitted with PROFINET's RT and IRT channel

Appendix B. PROFINET Simulation Configuration GUI

In this window, which is shown in Figure 29, the user can introduce a set of parameters for the PROFINET simulation of the toolkit for Dijon and generate an .INI configuration file that can be read by OMNeT++. By clicking on "Run" in the top bar, the run simulation window would emerge.

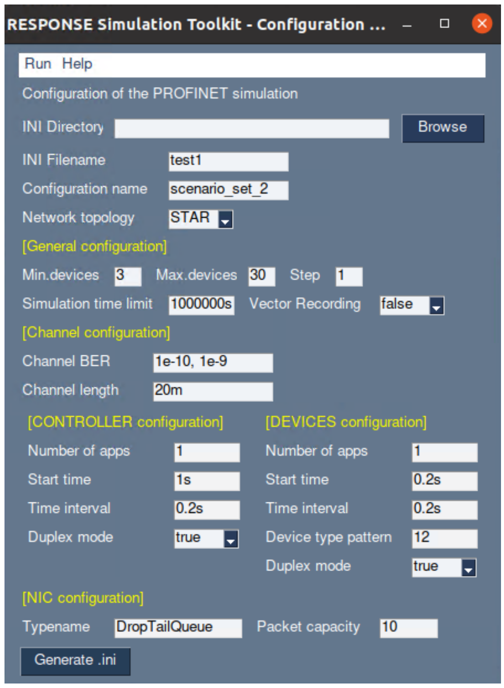

Figure 29 Screenshot of the window to create a configuration file for PROFINET simulation (toolkit for Dijon). Draft version.

The simulation parameters and fields that the user would be able to modify are the following:

- **INI Directory:** The path where the .INI configuration file will be saved when pressing the "Generate .ini" button at the lower left corner of the window.
- **INI Filename:** The name of the .INI configuration file used to save it.
- **Configuration name:** The configuration needs a name to be distinguished from others and to execute the simulation.

- **Network topology:** If necessary, the user would be able to change the topology of the network from STAR to BUS.
- **General configuration:**
    - **Min. Devices:** Minimum number of devices connected to the EMPAIR for a set of scenarios.
    - **Max. Devices:** Maximum number of devices connected to the EMPAIR for a set of scenarios. If it is equal to “Min. Devices”, all the scenarios in the configuration file will have the same number of devices.
    - **Step:** Step to be considered when iterating from the “Min. Devices” to the “Max. Devices”.
    - **Simulation time limit:** Simulation time in seconds that, once reached, will stop the execution of the simulation.
    - **Vector recording:** In case the user wants to obtain a CSV file that stores the vectors of variable indicators. By default, it is set to false, since statistics would be simpler to analyse.
- **Channel configuration:**
    - **Channel BER:** BER for the Ethernet cables connecting the different devices. Multiple BER values can be set, separated by commas.
    - **Channel length:** To define the length of the Ethernet cable (in meters) between the switch and all the connected devices (except the EMPAIR, which would be connected very close to the switch).
- **Controller configuration:** This would be the configuration of the EMPAIR device.
    - **Number of apps:** The number of basic PROFINET UDP applications (see subsection 2.6.1) that the controller would contain. During the SRA, it should be always set to one.
    - **Start time:** Once the simulation has started, the simulation time in seconds at which the controller will start operating.
    - **Time interval:** Time gap in seconds between sending messages.
    - **Duplex mode:** To set bidirectional communications for the controller. In case of PROFINET, as the same device can be simultaneously a provider and producer of data, it must be set to true.
- **Device’s configuration:**
    - **Number of apps:** Similar to controller configuration. The number of basic PROFINET UDP applications (see subsection 2.6.1) that the device would contain. During the SRA, it should be always set to one.

- **Start time:** Once the simulation has started, the simulation time in seconds at which the device will start operating.
- **Time interval:** Time gap in seconds between sending messages.
- **Device type pattern:** To define the order and proportions of the different types of devices considered for the toolkit, which are related to the "Type of devices" SRA parameter defined in section 2.4. The equivalences are:
  - **1:** PV data logger type.
  - **2:** BMS (4 battery types).

  The pattern set will be used by the simulation model to know in which order the devices must be placed and to let the controller know the type of server it is connecting to, so that the characteristics collected in Table 5 can be applied correctly during the simulation.
- **Duplex mode:** To set bidirectional communications for the devices. In case of PROFINET, as the same device can be simultaneously a provider and a producer of data, it must be set to true.

- **NIC configuration:**
  - **Typename:** The queue type implemented to manage packets. It should not be changed from its default value, "DropTailQueue".
  - **Packet capacity:** The "DropTailQueue" requires defining the size of the buffer so that when it is surpassed, further packets are dropped. It should not be changed from its default value, 10.

It must be taken into consideration that the toolkit is still being developed and that the interface described in this appendix may not be definitive.

Appendix C. ICT characteristics for Turku's IEs

Table 13 Estimated ICT characteristics for IE 4.1.8

Link ID	Source	Receiver	Protocol	Information object	Frequency of exchange	Communication technology
1	FiksuHub	eGAIN Edge system	HTTPS	Energy consumption information Indoor condition information?	*1/h or 1/m* *1/h or 1/m*	REST API
2	Building automation system	FiksuHub	*To be defined*	Energy consumption information Indoor condition information?	*1/h or 1/m* *1/h or 1/m*	*To be defined*
3	eGAIN Edge system	Smart City Knowledge Graph	HTTPS	Anomaly alarm	*1/m (to be defined)*	REST API
4	VTT optimisation algorithm	FiksuHub	HTTPS	DH sell, hourly DH buy, hourly DC sell, hourly DC buy, hourly Heat pump operation plan, hourly	1/day 1/day 1/day 1/day 1/day	REST API REST API REST API REST API REST API
5	FiksuHub	Smart City Knowledge Graph	HTTPS	Energy production, consumption information	*1/m or 1/h*	REST API

Table 14 Estimated ICT characteristics for IE 4.1.9

Link ID	Source	Receiver	Protocol	Information object	Frequency of exchange	Message size (Bytes)	Communication technology
1	*VTT robot car (eLvira)* *VTT robot car (Marilyn)*	*VTT robot car (Marilyn)* *VTT robot car (eLvira)*	*TCP Publish-Subscribe*	*VTT robot car data* *VTT robot car data*	*10/s* *10/s*	*100 B* *100 B*	*Short-range communications (ITS-G5)*
2	*VTT robot car* *Interchange node*	*Interchange node* *VTT robot car*	*TCP Publish-Subscribe*	*VTT robot car data* *Traffic information* *Other vehicle data*	*1/s* *1/s* *1/s*	*100 B* *100 B* *100 B*	*Cellular communications 5G*
3	*Turku City Data* *Interchange node*	*Interchange node* *Turku City Data*	*HTTPS*	*Traffic information* *Other vehicle data* *VTT robot car data*	*1/s* *1/s* *1/s*	*100 B* *100 B* *100 B*	*REST API*

Table 15 Estimated ICT characteristics for the operational optimisation of the energy operations in Tyyssija building in Turku

Link ID	Source	Receiver	Protocol	Information object	Frequency of exchange	Communication technology
1	VTT optimisation algorithm	Ferroamp EMS	HTTPS	Heat pump electricity load 24h ahead, hourly	1/day	REST API
2	VTT optimisation algorithm	Turku Energia	HTTPS	DH buy forecast, hourly DH sell forecast, hourly DC sell forecast, hourly waste heat buy forecast, hourly	1/day	REST API
	Turku Energia	VTT optimisation algorithm	HTTPS	DH supply temperature set point PED DH return temperature forecast, hourly DH price (sell&buy), hourly DC price (sell&buy), hourly DC waste heat price, hourly		
3	Turku Energia / Nordpool	VTT optimisation algorithm	HTTPS	Nordpool spot prices of PED electricity, hourly	1/day	REST API
4	VTT optimisation algorithm	HögforstGST	HTTPS	DH sell, hourly DH buy, hourly DC sell, hourly DC buy, hourly Heat pump operation plan, hourly	1/day	REST API
5	Load model AI	VTT optimisation algorithm	HTTPS	Supply temperature, hourly Return temperature, hourly Tyyssija load, hourly	1/day	REST API
6	HögforsGST	Load model AI	HTTPS	Supply temperature, hourly Return temperature, hourly Tout, hourly	2/month	REST API

RESPONSE

Integrated Solutions for Positive Energy and Resilient Cities

This project has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement nº 957751. The document represents the view of the author only and is his/her sole responsibility: it cannot be considered to reflect the views of the European Commission and/or the European Climate, Infrastructure and Environment Executive Agency (CINEA). The European Commission and the Agency do not accept responsibility for the use that may be made of the information it contains.