



Project Acronym: **VICINITY**  
Project Full Title: **Open virtual neighbourhood network to connect intelligent buildings and smart objects**  
Grant Agreement: **688467**  
Project Duration: **48 months (01/01/2016 - 31/12/2019)**

## Deliverable D4.1

### Set of open sample VICINITY gateway adapters

Work Package: **WP4 – VICINITY Client Infrastructures Implementation**  
Task(s): **T4.1 – VICINITY Gateway Adapters**  
Lead Beneficiary: **GRN**  
Due Date: **30 June 2018 (M30)**  
Submission Date: **29 June 2018 (M30)**  
Deliverable Status: **Final**  
Deliverable Type<sup>1</sup>: **R/DEM**  
Dissemination Level<sup>2</sup>: **PU**  
File Name: **VICINITY\_D4.1\_Set\_of\_open\_sample\_VICINITY\_gateway\_adapters\_v1.0.docx**



*This project has received funding from the European Union's Horizon 2020 Research and innovation programme under Grant Agreement n°688467*

## VICINITY Consortium

No	Beneficiary		Country
1.	TU Kaiserslautern (Coordinator)	UNIKL	Germany
2.	ATOS SPAIN SA	ATOS	Spain
3.	Centre for Research and Technology Hellas	CERTH	Greece
4.	Aalborg University	AAU	Denmark
5.	GORENJE GOSPODINJSKI APARATI D.D.	GRN	Slovenia
6.	Hellenic Telecommunications Organization S.A.	OTE	Greece
7.	bAvenir s.r.o.	BVR	Slovakia
8.	Climate Associates Ltd	CAL	United Kingdom
9.	InterSoft A.S.	IS	Slovakia
10.	Universidad Politécnica de Madrid	UPM	Spain
11.	Gnomon Informatics S.A.	GNOMON	Greece
12.	Tiny Mesh AS	TINYM	Norway
13.	HAFENSTROM AS	ITS	Norway
14.	Enercouthim – Associação Empresarial de Energia Solar de Alcoutim	ENERC	Portugal
15.	Municipality of Pylaia-Hortiatis	MPH	Greece

**<sup>1</sup> Deliverable Type:**

R: Document, report (excluding the periodic and final reports)  
 DEM: Demonstrator, pilot, prototype, plan designs  
 DEC: Websites, patents filing, press & media actions, videos, etc.  
 OTHER: Software, technical diagram, etc.

**<sup>2</sup> Dissemination level:**

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.

**Disclaimer**

*This document reflects only the author's views and the European Union is not liable for any use that may be made of the information contained therein.*

## Authors List

Leading Author (Editor)				
Surname	First Name	Beneficiary	Contact email	
Bračko	Mihael	GRN	mihael.bracko@gorenje.si	
Co-authors (in alphabetic order)				
No	Surname	First Name	Beneficiary	Contact email
1.	Cimmino	Andrea	UPM	cimmino@fi.upm.es
2.	García-Castro	Raúl	UPM	rgarcia@fi.upm.es
3.	Guan	Yajuan	AAU	ygu@et.aau.dk
4.	Heinz	Christopher	UNIKL	heinz@cs.uni-kl.de
5.	Kostelnik	Peter	IS	peter.kostelnik@intersoft.sk
6.	Hovstø	Asbjørn	HITS	hovsto@online.no
7.	Koutli	Maria	CERTH	mkoutli@iti.gr
8.	Mazahir	Hashimli	TINYM	mazahir@tiny-mesh.com
9.	Oravec	Viktor	BVR	viktor.oravec@bavenir.eu
10.	Poveda-Villalón	María	UPM	mpoveda@fi.upm.es
11.	Serena	Fernando	UPM	fserena@fi.upm.es
12.	Sundvor	Mariann	TINYM	mariann@tiny-mesh.com

## Reviewers List

List of Reviewers (in alphabetic order)				
No	Surname	First Name	Beneficiary	Contact email
1.	Kostelnik	Peter	IS	peter.kostelnik@intersoft.sk
2.	Poveda	Maria	UPM	mpoveda@fi.upm.es
3.	Savaghebi	Mehdi	AAU	mes@et.aau.dk

## Revision Control

Version	Date	Status	Modifications made by
0.1	27 March 2018 (M27)	TOC	Mihael Bračko (GRN)
0.2.2	10 May 2018 (M29)	Initial Draft	Mihael Bračko (GRN)
0.2.3	25 May 2018 (M29)	Draft with some contributions	Mihael Bračko (GRN)
0.2.4	13 June 2018 (M30)	Draft with all contributions	Contributors to D4.1
0.2.5	18 June 2018 (M30)	Draft changes to new form, content is moved to GitHub	Mihael Bračko (GRN)
0.3	22 June 2018 (M30)	Draft updated according to reviewers recommendations	Mihael Bračko (GRN)
0.4	28 June 2018 (M30)	HITS removed as reviewer, Draft updated	Jure Ernst (GRN)
1.0	29 June 2018 (M30)	Submission to the EC	Christoph Grimm (UNIKL)

## Executive Summary

VICINITY adapters act as a gateway between the outer world and the VICINITY infrastructure, thus providing its main objective: Interoperability. An adapter consists of partners-specific infrastructure implementation on one side and a standard VICINITY interface on the other side.

The project has implemented a vast set of different adapters to many other platforms. In the deliverable, we give an overview the available adapters that are provided by the partners of the VICINITY consortium.

The available adapter implementations provide a vast basis of templates and examples for future development of VICINITY adapters by any future potential partner of VICINITY.

## Table of Contents

Executive Summary .....	5
1. Introduction.....	9
1.1. Adapter placement within VICINITY infrastructure .....	9
1.2. Objectives in Work Package WP4 and Task T4.1 .....	10
2. VICINITY adapter development state.....	11
2.1. Greek Use Case VICINITY adapters – Building and Health Domain (CERTH, GNOMON, OTE).....	11
2.1.1. MPH Use Case Adapters .....	11
2.1.2. CERTH-ITI Test lab .....	12
2.2. UNIKL VICINITY adapters .....	12
2.3. TINYM VICINITY adapter.....	12
2.4. HITS VICINITY adapter .....	13
2.5. AAU VICINITY adapter .....	13
2.6. GRN VICINITY adapter .....	13
2.7. ENERC VICINITY adapters.....	13
3. Ontology updates for VICINITY adapters.....	14
4. Conclusions .....	15
References.....	16

## List of Figures

Figure 1 VICINITY Node Architecture.....	10
--	----

## List of Definitions & Abbreviations

Abbreviation	Definition
EC	European Commission
EU	European Union
IoT	Internet of Things
P2P	Peer to Peer
JSON	JavaScript Object Notation



## 1. Introduction

### Context in VICINITY

This deliverable is built on top of the finding and the conclusions of deliverables:

- D1.1: VICINITY Communication Server
- D3.2: VICINITY Neighbourhood Manager
- D3.3: Open VICINITY Gateway API

Finally, this deliverable will provide the ground work for deliverables:

- D4.2: VICINITY Agent and Auto-Discovery platform

### Technical context

Manufacturers use private, proprietary communication solutions which are usually also copy-righted. Many of them also joined one or more existing IoT initiatives and implemented their protocols. Yet the number of various IoT initiatives rises from year to year, so does also various communication techniques and protocols.

A VICINITY adapter represents a two-way gateway between manufacturers infrastructure in VICINITY infrastructure. It could also be used as a translation layer from various IoT initiatives to and from VICINITY infrastructure. As such it serves as an entry point to VICINITY infrastructure.

Since manufacturers are using proprietary communication solutions, it is their task to develop their own VICINITY adapter in order to enter the world of VICINITY. Actual implementations could be implemented either as a pure software solution (as standalone services, simple code extensions, etc.) or as an additional piece of hardware, with dedicated software implementation.

### 1.1. Adapter placement within VICINITY infrastructure

According the VICINITY overall architecture [4], a VICINITY Adapter is one of the VICINITY Node components which ensures the translation of a local infrastructure to the common VICINITY format. The VICINITY Adapter communicates directly with VICINITY Agent which provided IoT object discovery functionality and handles the communication with VICINITY Gateway API (Figure 1).

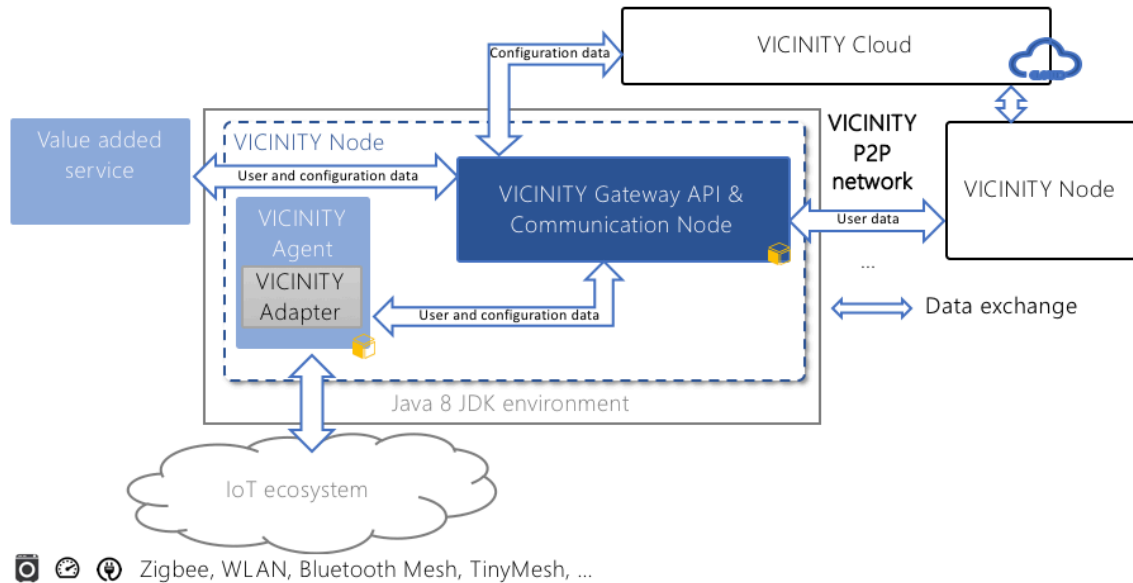


Figure 1 VICINITY Node Architecture

The VICINITY Adapter interface towards the VICINITY Agent includes following capabilities:

- IoT object discovery – which expose description of each IoT object accessible through the VICINITY Adapter;
- Provides consuming and exposing services for IoT objects properties and actions accessible through the VICINITY Adapter;
- Provides services to consume events by the VICINITY Adapter from the VICINITY Peer-to-peer network.

Detail description of these interfaces is described in: <https://github.com/vicinityh2020/vicinity-agent/blob/master/docs/ADAPTER.md>.

The IoT object discovery and exposing services are necessary to provide semantic discovery services through Gateway API Services in VICINITY Gateway API to distribute interoperable queries over different IoT objects in P2P network.

## 1.2. Objectives in Work Package WP4 and Task T4.1

In the rest of this deliverable, some general directions on how to build typical VICINITY adapter will be provided. Each partner, tackling with VICINITY adapter will provide a link to their VICINITY adapter sources, which in case of open source licensing, will be also available to a broad spectrum of future partners of VICINITY, who will support VICINITY with their solution portfolio.

In terms of T4.1, this deliverable provides a framework for implementing various spectrum of VICINITY adapters.

## 2. VICINITY adapter development state

This chapter contains links to actual implementations of VICINITY adapters, provided by consortium partners.

### 2.1. Greek Use Case VICINITY adapters – Building and Health Domain (CERTH, GNOMON, OTE)

#### 2.1.1. MPH Use Case Adapters

---

Vicinity Adapter based on IoTivity Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-IoTivity-MPHPilot/blob/master/README.md>

Vicinity Adapter based on Node Red Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-NodeRed-MPHPilot/blob/master/README.md>

Vicinity Adapter based on EHealthPass Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-EHealthPass-MPHPilot/blob/master/README.md>

Five Vicinity Adapters for the respective Value-Added Services (incorporated):

- [https://github.com/vicinityh2020/vicinity-VAS-GDPR\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-VAS-GDPR_with_adapter/blob/master/Adapter_README.md)
- [https://github.com/vicinityh2020/vicinity-VAS-IndividualStatistics\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-VAS-IndividualStatistics_with_adapter/blob/master/Adapter_README.md)
- [https://github.com/vicinityh2020/vicinity-VAS-AbnormalDetection\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-VAS-AbnormalDetection_with_adapter/blob/master/Adapter_README.md)
- [https://github.com/vicinityh2020/vicinity-VAS-UrbanMarathon\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-VAS-UrbanMarathon_with_adapter/blob/master/Adapter_README.md)
- [https://github.com/vicinityh2020/vicinity-adapter-VAS-AggregatedStatistics\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-adapter-VAS-AggregatedStatistics_with_adapter/blob/master/Adapter_README.md)

### 2.1.2. CERTH-ITI Test lab

---

Vicinity Adapter based on IoTivity Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-IoTivity/blob/master/README.md>

Vicinity Adapter based on SiteWhere Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-SiteWhere/blob/master/README.md>

Vicinity Adapter based on LinkSmart Platform:

- <https://github.com/vicinityh2020/vicinity-adapter-CERTH-LinkSmart/blob/master/README.md>

Vicinity Adapter for Value Added Service:

- [https://github.com/vicinityh2020/vicinity-VAS-DRFlexibility\\_with\\_adapter/blob/master/Adapter\\_README.md](https://github.com/vicinityh2020/vicinity-VAS-DRFlexibility_with_adapter/blob/master/Adapter_README.md)

## 2.2. UNIKL VICINITY adapters

UNIKL implemented two distinct adapters. These adapters and accompanied documentation could be obtained from:

- VICINITY OpenHAB adapter:  
<https://github.com/vicinityh2020/vicinity-adapter-openhab>
- VICINITY adapter based on Eclipse Kura IoT gateway framework:  
<https://github.com/vicinityh2020/Vicinity-Adapter-Kura-UNIKL>

## 2.3. TINYM VICINITY adapter

Adapter and documentation are available at:

- <https://github.com/vicinityh2020/vicinity-adapter-tinymesh>

## 2.4. HITS VICINITY adapter

Adapter and documentation are available at:

- <https://github.com/vicinityh2020/vicinity-adapter-hits>

## 2.5. AAU VICINITY adapter

Adapter and documentation are available at:

- <https://github.com/vicinityh2020/LabVIEW-adapter/>

## 2.6. GRN VICINITY adapter

Adapter and documentation are available at:

- <https://github.com/vicinityh2020/vicinity-adapter-GRN>

## 2.7. ENERC VICINITY adapters

Adapters and documentation are available at:

- <https://github.com/vicinityh2020/vicinity-ENERC-UVMeterKZ>
- <https://github.com/vicinityh2020/vicinity-ENERC-weatherstation>
- <https://github.com/vicinityh2020/vicinity-ENERC-serinussensors>
- <https://github.com/vicinityh2020/vicinity-ENERC-energomonitorThermosense>
- <https://github.com/vicinityh2020/vicinity-ENERC-energomonitorAirsenseCanary>
- <https://github.com/vicinityh2020/vicinity-ENERC-energomonitorPowersense>
- <https://github.com/vicinityh2020/vicinity-ENERC-energomonitorOptosense>
- <https://github.com/vicinityh2020/vicinity-ENERC-energomonitorRelaysense>
- <https://github.com/vicinityh2020/vicinity-ENERC-iotsensWatchmeter>
- <https://github.com/vicinityh2020/vicinity-ENERC-iotsensLuminosity>
- <https://github.com/vicinityh2020/vicinity-ENERC-nodered>

### 3. Ontology updates for VICINITY adapters

This section describes the evolution of the VICINITY ontology network<sup>1</sup> according to the VICINITY client adapters needs.<sup>2</sup> As already described in “D2.2. Detailed Specification of the Semantic Model” **Fehler! Verweisquelle konnte nicht gefunden werden.**, the VICINITY ontology network is designed following a modular approach. In such document, the general process being followed to develop the ontology network is also described.

The main update in the VICINITY ontology network is related to its modularization. In the previous version of the network three modules were defined, namely:

- Core: <http://iot.linkeddata.es/def/core/>
- WoT: <http://iot.linkeddata.es/def/wot/>
- WoT mappings: <http://iot.linkeddata.es/def/wot-mappings/>

In the updated version of the ontology network, there is a total of five modules as shown in **Fehler! Verweisquelle konnte nicht gefunden werden.**. The two new modules are:

- Adapters: <http://iot.linkeddata.es/def/adapters/>
- Datatypes: <http://iot.linkeddata.es/def/datatypes/>

The adapters module has been created to decouple from the generic part of the ontology related to the VICINITY platform (i.e., the core module) the particular characteristics of those devices that come from specific pilot scenarios. In this sense, the main goal is to provide support for third parties to reuse the core definition of the ontology independently of the particular devices and properties used in the VICINITY use cases, therefore in a more lightweight fashion. In any case, third parties will still be able to reuse the VICINITY adapters module if needed.

The datatypes module has been developed to support the specification of JSON schemas or datatypes in which interaction patterns from devices and services offer and accept data.

More information about ontology updates is available at GitHub:

- <https://github.com/mariapoveda/vicinity-ontology-adapters>
- <https://github.com/mariapoveda/vicinity-ontology-datatypes>

<sup>1</sup> <http://vicinity.iot.linkeddata.es/>

<sup>2</sup> It is worth noting that other updates of the ontology network will be reported in the corresponding deliverables.

## 4. Conclusions

Deliverable D4.1. brings an overview on VICINITY Adapters and various types of implementations, provided by consortium partners. Its sources accompanied by the documentation can be obtained from GitHub repositories, using links provided in this deliverable. In addition, the updates of the VICINITY ontology according to the use case adapters requirements has been reported. As such, provided adapters could serve as a starting point for any further adapter implementation.

## References

- [1] <http://www.vicinity-h2020.eu>
- [2] ICT 30 – 2015: Internet of Things and Platforms for Connected Smart Objects - <http://ec.europa.eu/research/participants/portal/desktop/en/opportunities/h2020/topics/914-ict-30-2015.html>
- [3] VICINITY project. D2.2 Detailed Specification of the Semantic Model. August 2017.
- [4] D1.6 VICINITY Architectural Design