EU-CIRCLE

A pan-European framework for strengthening Critical Infrastructure resilience to climate change

D7.1 Demonstrable Deployment of Integrated SimICI

Contractual Delivery Date: 30/11/2016	**Actual Delivery Date**: 26/01/2017
Type: Other / Software	**Version**: v0.9

Dissemination Level: Public Deliverable

Statement

This document accompanies the first software release of the SimICI reference testbed and describes the selected architecture, methodology, and functional application perspectives. SimIC is a multi-user web-based application providing an Simulated Network of Interconnected Critical Infrastructures around which technical and non-technical users may collaborate in order to explore scenarios; to investigate, in conjunction with CIRP, and test intervention strategies, and to define additional analysis prototypes that will help support the strengthening of Critical Infrastructure resilience to climate change.

EU-CIRCLE is a project that has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 653824. Please see http://www.eu-circle.eu/for more information.

Preparation Slip			
	Name	Partner	Date
From	David Prior	XUV	26/01/2017
Reviewer	A. Kostaridis / L. Perlepes	STWS	27/01/2017
Reviewer			
For delivery	David Prior	XUV	

Document Log			
Issue	Date	Comment	Author / Organization
V0.1	05/12/2016	Document Structure and Indexes	D. Prior / XUV
V0.2	16/12/2016	Background	D. Prior / XUV
V0.3	16/12/2016	System Architecture	D. Prior / XUV
V0.4	16/12/2016	Cascade Engine	R. Vaughan / XUV
V0.5	23/12/2016	User Experience	R. Vaughan / XUV, D. Prior / XUV
V0.6	23/12/2016	GIS Services	D. Prior / XUV
V0.7	16/01/2017	Future Work and Data	R. Vaughan / XUV, D. Prior / XUV
V0.8	22/01/2017	Updates Pursuant to Project Discussions	D. Prior / XUV
V0.9	26/01/2017	Internal Review and Updates	R. Vaughan / XUV, D. Prior / XUV
V1.0		Final Version	D. Prior / XUV

Abbreviations List	
Term	Description
2.5D	Two-and-one-half Dimensional (Also called ‘isometric’)
2D	Two Dimensional
3D	Three Dimensional
AOI	Area of Interest
API	Application Programming Interface
CEF	Chameleon Enterprise Framework
CI	Critical Infrastructure
CIRP	Critical Infrastructure Resilience Platform
CORDEX	Coordinated Regional Climate Downscaling Experiment
CSS	Cascading Style Sheets
FWI	Fire Weather Index
GIS	Geographic Information System
GUI	Graphical User Interface
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
JSONP	JSON with Padding
M	Month
NetCDF	Network Common Data Format
ODbL	Open Database Licence
OGC	Open Geospatial Consortium
OSM	Open Street Map
REST	Representational State Transfer
SDK	Software Development Kit
SimICI	Simulated Network of Interconnected Critical Infrastructures
SQL	Structured Query Language

SRS	Spatial Reference System
UX	User Experience
WCS	Web Coverage Service
WFS(-T)	Web Feature Service (-Transactional)
WKT	Well Known Type
WMS(-T)	Web Mapping Service (-Transactional)
WPS	Web Processing Service
XUV	Xuvasi Ltd

Executive Summary

EU-CIRCLE‘s scope is to derive an innovative framework supporting resilience of the interconnected European Critical Infrastructure to climate pressures as the increasingly dependent, interdependent and interconnected nature of CI networks exposes previously unseen risks, new vulnerabilities, and opportunities for disruption of those networks.

This document accompanies the first release of the integrated SimICI environment that provides an innovative collaborative environment, delivered through standard web application interfaces, within which technical and non-technical users may collaborate in order to explore scenarios; to investigate, in conjunction with CIRP, and test intervention strategies, and to define additional analysis prototypes that will help support the resilience of the interconnected European Critical Infrastructure.

SimICI has been designed and implemented as a ‘single-window’ web-based application that provides a series of intuitive interfaces through which the EU-CIRCLE ‘virtual city’ dataset and associated hazards and propagation models may be manipulated. SimICI integrates geospatial data services, a workflow engine supporting impact and intervention strategy assessment, and the ability to rapidly prototype new analyses for the EU-CIRCLE CIRP platform. SimICI is, and will continue to be, populated with the data comprising the EU-CIRCLE ‘virtual city’ dataset and other assets, hazards, impacts, and propagation models as arise from the EU-CIRCLE project as a whole.

The approach to SimICI has been to leverage open-source software and open standards, data formats, and protocols wherever possible. This obviates third-party licence costs, maximises the potential to engage with the open-source community, and provides maximum flexibility and extensibility in the capability provided, within and beyond the extent of the EU-CIRCLE project.

SimICI has not only been designed as an integrated capability in itself but also, and in keeping with the testbed objective for SimICI within EU-CIRCLE, as a capability that can be readily integrated with the CIRP platform as and when required.

Integration with CIRP occurs on two primary levels. Firstly, data integration, through the medium of the Shapefile format, which allows round-trip communication of assets, networks, and impacts between CIRP and SimICI. Secondly, process integration in which SimICI may be employed to prototype new analyses that may be readily implemented in the CIRP platform.

From an administrative perspective, SimICI has been designed to require as light a touch as possible. Administration in SimICI is primarily concerned with setting the data and service endpoints that the SimICI application must link with and, additionally, user and group management functions.

The SimICI Administration and User Guide documents are in process and will be released, at version 1, in M20. Further updates to these documents will be issued in M25 and M33 as SimICI is populated with data and services derived from the EU-CIRCLE research activities and scenarios.

Contents

Figures

Tables

1 Introduction

This document accompanies D7.1, a deliverable of type "OTHER", in this case the first software release of the Simulated Network of Interconnected Critical Infrastructures (SimICI). SimICI serves as a testbed for all developments, integrations, and evaluations performed during the EU- CIRCLE project and provides a reference environment for further exploitation at the end of the project. As SimICI leverages open-source resources, SimICI will also serve as the primary enabler for an outreach programme aimed at maximising awareness, exploitation, and ecosystem-led extension of resilience strategies for climate impacts in the open-source community.

D7.1 is the first deliverable of Work Package (WP) 7 of the EU-CIRCLE project.

The objectives of WP7 are:

1. Establish a controlled testbed environment (SimICI) for Simulating Interconnected (Critical) Infrastructures, Climate Hazards, Effects, and Risk/Impact Propagation.
2. Leverage open-source software and open-standards/data formats to maximise the application space for SimICI and to provide maximum flexibility and extensibility in the capability provided for use within and beyond SimICI.
3. Build within SimICI the outputs of the EU-CIRCLE project in relation to Climate Hazards, Critical Infrastructure Assets and Networks, and Critical Infrastructure Resilience and provide a clearly defined Application Programming Interface (API) through which SimICI may be integrated with the core EU- CIRCLE Framework.

WP7 (SimICI) is led by XUV. Work commenced in M3 and will continue through M36. To date, work on SimICI has focused on the concept, development, and integration of the SimICI testbed environment with the EU-CIRCLE CIRP platform under implementation in WP5. As part of the development strand, a virtual city reference dataset has been produced (first edition released by NCSRD M17) and integrated into SimICI.

Work to date has been focused around Tasks 7.1 (Implement Scenario Supports) and Task 7.3 (Reference Scenario Builds). D7.1 is comprised of the SimICI deployment to http://simici.eu-circle.eu and this supporting document describing that architecture and implementation of that deployment. This document also forms the foundation of D7.3 (Administration and User Manuals for SimICI System). The deployment for D7.1 will be demonstrated and exercised during the M22 sessions in Cyprus.

The remainder of this document is structured as follows: the background to and evolution of the SimICI design is discussed in the following Section 2. Section 3 describes the architecture and integration strategy for SimICI. Section 4 details the GIS services provided by SimICI and the Rethymno 'virtual city' dataset integrated and exploited therein. Section 5 discusses the SimICI User Experience and functionality supported therein. Finally, planned future work is presented in Section 6.

2 Background

The SimICI environment must provide a controlled testbed for the **Sim**ulation of **I**nterconnected (**C**ritical) **I**nfrastructures, Climate Hazards, Effects, and Risk/Impact propagation. The expectation for SimICI is that it will serve as a testbed for all developments, integrations, and evaluations performed during the EU-CIRCLE project; provide a reference environment for further exploitation at the end of the project, and serve as the primary enabler for an outreach programme to the open-source community.

SimICI, therefore, must meet a number of core requirements and constraints:

1. Leverage open-source software components;
2. Leverage open standards / data formats;
3. Provide a clearly defined Application Programming Interface (API) through which SimICI may be integrated (specifically with the EU-CIRCLE CIRP platform but also elsewhere);
4. Permit the building and exploitation of Climate Hazards, Critical Infrastructure Assets, Critical Infrastructure Networks, and Critical Infrastructure Resilience entities and resources within SimICI, and
5. Be capable of use, exploitation, and extension by both EU-CIRCLE consortium members and those others who become engaged with EU-CIRCLE capabilities through outreach actions.

In addition to the core requirements and constraints, however, there are a number of underlying and associated requirements that SimICI should meet in order to maximise its exploitation potential:

1. Provide the ability for non-climate researchers and scientists (eg: Policy Makers, Response Managers, Insurers, etc.) to engage with and make pragmatic use of the outputs from EU-CIRCLE;
2. Provide the ability to leverage complex analyses, as designed and proven by experts, and their associated outputs contained within the EU-CIRCLE CIRP;
3. Permit the intuitive construction of additional Critical Infrastructure elements that may subsequently be exploited both within SimICI and EU-CIRCLE CIRP;
4. Permit the ability for SimICI users to prototype, and subsequently generate an automated specification for, new analyses to be included in the EU-CIRCLE CIRP platform;
5. Ensure that the mechanism employed for (4) above allows both specialist (eg: researchers and scientists) and end (eg: policy makers, response managers, etc.) user engagement such that it helps bring together the different actors with an interest in improved Critical Infrastructure Resilience;
6. Permit use of SimICI as a pure web-based application that requires only a modern web browser;
7. Permit multiple simultaneous users of the SimICI application to support active collaboration;
8. Permit the use of synthetic or real data as may be required by each SimICI user and use case, and
9. Provide clearly defined APIs, supporting open standards for data formats and protocols, for the various component boundaries within SimICI such that SimICI is formed from a set of components that may be orchestrated, integrated, and exploited in the widest possible application space.

With the above in mind, SimICI has evolved through a series of Design and Integration concepts during the first half of the EU-CIRCLE project (M1 – M18).

2.1 SimICI Design Concepts

At the EU-CIRCLE kick-off meeting (Athens, JUN15), the nature of the SimICI was discussed by the consortium members. With an emphasis on familiarity, usability, and flexibility, the suggestion was made that something akin to SimCity, the city simulation game, would be beneficial. By the time of the second EU-CIRCLE partners meeting (Cyprus, NOV15), that suggestion had been used as inspiration for a proof-of-concept design.

Figure 1: SimCity 4 City Scape - SimICI Inspiration (EU-CIRCLE Cyprus, NOV15)

The SimCity game, version 4 of which is illustrated in Figure 1, above, employs the conceit of user-led development of a city-scale environment. In SimCity, the underlying networks of a city (eg: water, electricity, transportation, and communications) are deployed in support of the development of city assets (eg: buildings, facilities, industry, and population). Each physical asset within a city is 'aware' of the set of resources and dependencies that it requires in order to (a) be instantiated and (b) operate.

For example, construction of a building asset requires the satisfaction of constraints with regard to both the raw materials to be used *and* the transportation infrastructure necessary to deliver those materials to the construction location. Once constructed, the successful operation of the building depends on the continued optimal connections to city infrastructure with regard to both utilities (eg: water, power, etc.) and supporting infrastructure (eg: roads, mass transit, etc.). The utility connections allow the building to function which the infrastructure connections allow the building to operate effectively by ensuring that employees can get to work, that supplies may be delivered, and that waste may be removed. Disaster scenarios, some of which – such as earthquake, tornado, and disease, are more real-world relevant than others, can then be applied to imbue damage to the city and its infrastructure.

The SimCity metaphor, therefore, is very relevant to the immediate goals for EU-CIRCLE and, moreover, provides ample indication as to where EU-CIRCLE outputs may be extended and exploited in the future.

Despite this, however, the SimCity metaphor falls short in one critical aspect: SimCity is typically concerned with geotypical rather than geospecific city constructs. XUV has previously worked on the use of simulations as a means to engage multiple actors, with multiple perspectives, around a common city challenge. As a result of that work, XUV discovered that a geotypical city (ie: a city that is generic) is less engaging than a geospecific city (ie: a city that appears familiar). The difference between the two can be as simple as instantiating one or two landmark buildings within a city model. Those landmarks serve to give participants a sense of place and, in consequence, a sense of responsibility with regard to what they are engaged in.

For EU-CIRCLE, the requirement for geospecific constructs is, of course, more heavily reinforced: in order to accurately apply data and models to the challenges of Critical Infrastructure Resilience, it is necessary to ensure that those data and models are geo-correlated and, therefore, exploitable through common context.

Figure 2: OpenStreetMap 'Buildings' Detail - SimICI Inspiration (EU-CIRCLE Cyprus, NOV15)

The challenge for SimICI, therefore, was to locate a readily exploitable source of geospecific context against which the EU-CIRCLE objectives could be realised.

In keeping with the mandate to employ open-source materials, OpenStreetMap (OSM) (an openly licensed map of the world created by volunteers using local knowledge) was investigated. For OSM, openly licensed is subject to the terms of the Open Database Licence (ODbL): a copyleft, “Share Alike” licence agreement intended to allow users to freely share, modify, and use a database while maintaining those same freedoms for others. Under the terms of the ODbL, users are free to share (copy, distribute, and use), create works from, and modify, transform, and build upon the database to which the licence applies.

Such freedoms are preserved as long as users attribute any public use of the database (or of works produced from the database); preserve and keep intact those licences and notices associated with any redistribution of the original database (or works produced from it); offer any adapted database derived from the original database under the same ODbL as the original database, and ensure that any redistribution of the original or an adapted database continues to be offered in an open and accessible form. Compliance with the ODbL will, therefore, strongly support the open-source use, outreach, and further exploitation goals for EU-CIRCLE.

While OSM is most typically used to power the generation of map tiles (ie: the set of images used to underpin mapping applications), the OSM database contains additional information represented in a standard format and capable of being discretely exploited. One such set of additional information is known as the OSM Buildings layer. OSM Buildings comprises sets of structured geometries that represent individual real-world buildings across the globe. Extracting OSM Buildings, and rendering the geometries therein, generates a layer of buildings that may be overlaid on the underlying set of map tiles. An example of this, for Paris, is shown in Figure 2, above. Additional layers, representing different discrete and aggregated sets of assets, networks, risks, or impacts may be both rendered within the same map client *and* programmatically manipulated by shared external services.

The OSM data model (the table and field structure that supports the OSM database) is also readily adaptable to the registration and persistence of additional structures that are relevant to the Critical Infrastructure Resilience sector. For example, using the OSM concept of nodes (geometric points) and ways (links between points), it is as easy to represent an electricity network as it is a road network (after all, both are specific instances of a route-flow graph). Moreover, because of the underlying geo-correlation of nodes contained within the data model, the co-presence of different node types (perhaps representing assets in different networks) allows for ready extrapolation of critical interdependencies and, therefore, the external calculation of the potential for cascading effects due to failure in any part of the interdependency matrix.

The extensibility of the data model, coupled with the core OSM concept of volunteer maintenance through the direct editing and augmentation of the OSM database, therefore makes the OSM capability highly relevant to the SimICI objectives and requirements. However, for the purpose of ensuring that EU-CIRCLE modifications to the OSM database do not bleed through to and contaminate the primary OSM database, it will be necessary to ensure that EU-CIRCLE specific extensions are implemented alongside a discrete, but regularly updated, instance of the primary OSM database.

The use of the OSM data model provides support for the creation of additional Critical Infrastructure elements within SimICI (Core Requirement 3; Underlying Requirement 4). Indeed, given that OSM is a user editable dataset, OSM provides its own tools through which additional nodes and ways may be created. The OSM editing tools are, however, implemented as part of the overarching OSM application and it is not the objective of SimICI to provide a simple replication of OSM functionality nor to require that SimICI users must move between applications in order to achieve their objectives.

To that end, SimICI will exploit the underlying editing capabilities of the OSM data model *but* will integrate the design of such edits, using well known geometry types (eg: Points, Lines, Polylines, Polygons), into the main SimICI interface. Having identified this objective, the next challenge for SimICI was to prove the concept of editing OSM compatible data structures in a web browser. While numerous options exist that support the viewing of OSM layers in browser-based map clients (eg: Leaflet, ViziCities, Cesium, Terria – to name several open-source options), the options for *editing* an underlying OSM data model are far fewer and, in the main, closed source.

A proof-of-concept web application was developed that directly addressed the requirements of the editing challenge by fusing a city simulator user interface (allowing SimCity-like actions towards the construction of a city (offices, factories, leisure complexes, residences) and its networks (power and transportation)) with an underlying adaptor that was capable of bi-directional data exchange with the OSM data model.

Figure 3: Constructing Virtual Cities in a Web Browser - SimICI Proof of Concept (EU-CIRCLE Cyprus, NOV15)

Figure 3, above, illustrates the SimICI proof-of-concept that was (remotely) demonstrated[1] at the second EU-CIRCLE partners meeting (Cyprus NOV15) and which met with approval from all partners. This proof demonstrated that it is possible to extend existing geospecific datasets, create new geospecific datasets, and even to generate entirely synthetic geotypical datasets for the purposes of representing Critical Infrastructure components and the environment in which they operate.

From the second EU-CIRCLE partners meeting (Cyprus NOV15) in M6 on, the emphasis in WP7 switched from explorations and proofs of capability to the implementation and deployment of the SimICI environment. This activity has embraced the iterative and final outputs, to date, of EU-CIRCLE WP1, WP2, WP3, and WP4 and factored for observations arising from the case study discussions that took place during the EU-CIRCLE Consolidation Workshop (Milan MAY16) in M12.

The final 'as-designed' architecture of the SimICI environment is described in the following section regarding integration with the EU-CIRCLE CIRP platform.

[1] The SimICI proof-of-concept described by Figure 3, above, was also put on display for delegate engagement at the UK Transport Security Expo 2015 (DEC15). During the course of the Expo, enthusiastic reactions and favourable comments were received from a wide cross-section of transport security industry representatives who recognised the potential benefits of such an interface. While this was not a formal dissemination action, it provided external validation of the SimICI concept and approach to user experience.

3 Architecture

Core Requirement 3 specifies that SimICI must provide a clearly defined Application Programming Interface (API) through which SimICI may be integrated with the core EU-CIRCLE Framework: CIRP. This requirement is expressed as having the possibility of exploitation by EU-CIRCLE CIRP. However, an increasing awareness of what CIRP does, how CIRP is implemented, and whom the intended audience for CIRP is has spawned a number of underlying requirements (specifically Underlying Requirements 1 through 7) that make integration between SimICI and EU-CIRCLE CIRP increasingly desirable.

At the time of the second EU-CIRCLE partners meeting (Cyprus NOV15), a basic concept for SimICI integration had been developed per Figure 4, below.

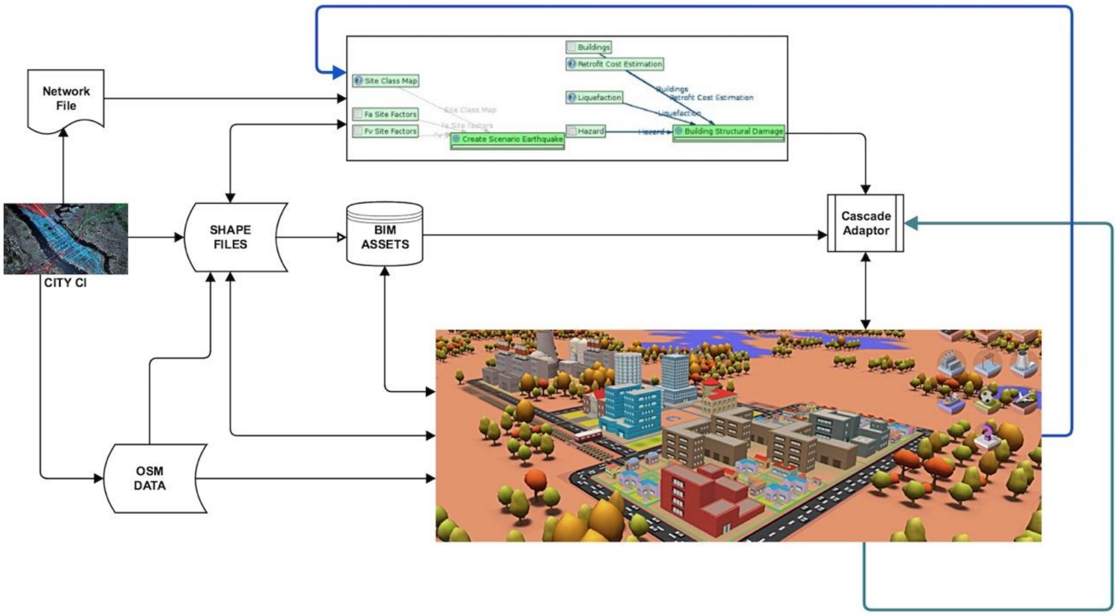


Figure 4: Base Concept for SimICI Integration (EU-CIRCLE Cyprus, NOV15)

In Figure 4, the web application surface for SimICI – as indicated by the User Experience (UX) proof-of-concept image – is shown drawing on datasets held in the OSM data model, assorted shapefiles, and a Building Information Management (BIM) asset database. The datasets as shown provide the environment and the Critical Infrastructure for a specific Area of Interest (AOI). Shapefiles are shown as being shared resources between SimICI and EU-CIRCLE CIRP.

Figure 4 shows a direct integration of SimICI with EU-CIRCLE CIRP via a Cascade Adaptor. Cascade is XUVs core platform technology that provides the means to define and execute user-defined analytical pipelines driven by any set of data sources and populated by a wide variety of algorithms and functions. At the M6 point at which Figure 4 was generated, the expectation was that the ERGO application framework and/or the SATWAYS Chameleon Enterprise Framework (CEF), on which EU-CIRCLE CIRP is being built, would expose a remote Application Programming Interface (API) that could be exploited by Cascade and, therefore, provide an active link between SimICI and EU-CIRCLE CIRP.

An additional direct integration, shown by the blue line in Figure 4, indicates that user-defined outputs (eg: specific algorithms, functional analyses, etc.) from SimICI would flow back as resources for exploitation within EU-CIRCLE CIRP. This mechanism was put in place to support the prototyping of analyses in SimICI that could then be sent to EU-CIRCLE CIRP where a complete implementation of that analysis, created to production quality and backed by scientific rigour, would then be implemented prior to release as a formal analysis. Once released for use within EU-CIRCLE CIRP, the formal analysis would then be accessible via the Cascade Adaptor route.

The M6 transition to the development and implementation of SimICI stimulated deeper analysis of the integration requirements, informed through reference to the ERGO source code that is exploited within EU-CIRCLE CIRP. As a result of that process, the Architecture and Integration model for SimICI was refined and evolved to that presented in Figure 5, below.

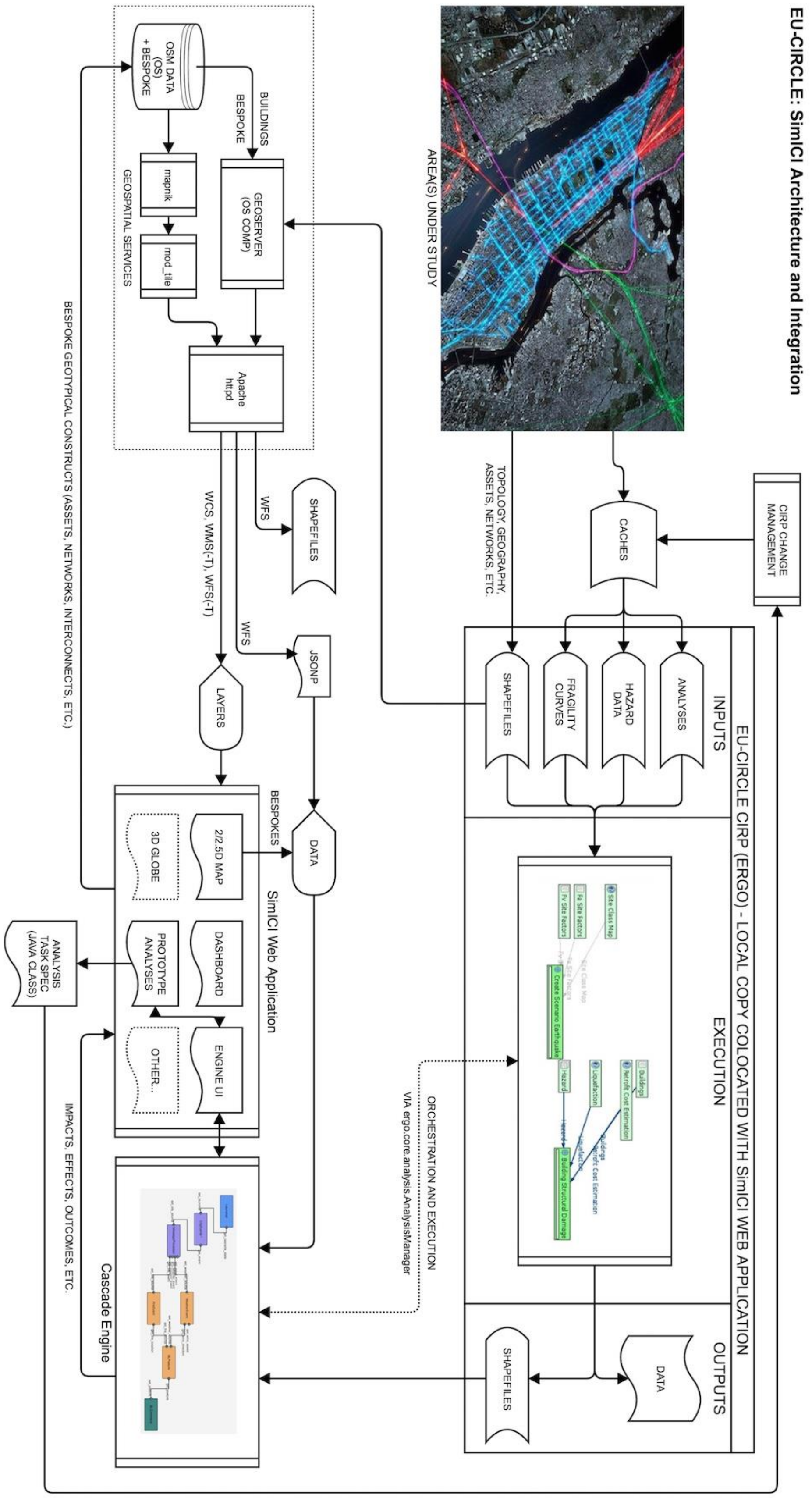


Figure 5: EU-CIRCLE SimICI Architecture and Integration

Figure 5, above, presents the final 'as-designed' Architecture and Integration model for SimICI. This model delivers services that meet all of the Core and Underlying Requirements for SimICI within the EU-CIRCLE context. As noted, the final model is an extension of and enhancement to the original concept model developed and disseminated in M6 (per Figure 4, previously).

The final model, per Figure 5, is split horizontally by aspect: the core EU-CIRCLE CIRP Framework is presented in the top half of the model and the SimICI components are presented in the bottom half. There are four integration touchpoints between the two aspects.

It is key to note that both aspects of the Architecture and Integration model are driven by the selection of a region or area that acts as the AOI for analysis. In the CIRP aspect, the AOI is defined by the creation of a scenario that is populated by the selection of relevant geographic data as held in a local or cached shapefile. In the SimICI aspect, the AOI is defined by the viewport of the map client that then sets a geographic Bounding Box[2] to be used in the selection of map tiles, features, and other geometry or associated data as may be relevant to the selected AOI.

The Rethymno-based dataset (first edition M17), comprising the Virtual City reference implementation, provides a ready populated AOI against which the geographic and geometric services of SimICI will be tested and evaluated. It is anticipated that similar data will be released for use in the M22 Cyprus Workshop and that that data, specific to the Cyprus AOI, will provide further opportunity for testing and evaluation of the SimICI services.

With regard to the CIRP aspect of Figure 5, this is discussed in extensive detail in D5.1 (CIRP Detailed Design Document) as released M10. The CIRP aspect is provided as part of Figure 5 in order to detail and explain the integration touchpoints (See later in this section). It should be noted, however, that for reasons of access to CIRP components from within SimICI, the latest release of EU-CIRCLE CIRP will be collocated with the SimICI application. This is discussed in more detail below.

3.1 SimICI Components

Regarding the SimICI components of the Architecture and Integration model, the following discusses those in sequence from left to right across the bottom half of Figure 5:

1. ***Geospatial Services:***
    a. Provides a complete geospatial stack built entirely from open source components and leveraging open standards and data formats.
    b. Comprises:
        i. A PostgreSQL database, with PostGIS extensions, populated by the OSM data model plus extensions to that model to support bespoke elements as created by SimICI users. The core OSM data model is populated by the OSM data feed, configured for regular updates. The bespoke extensions are populated by a thin API that is leveraged from within SimICI for the purposes of data persistence.
        ii. The Mapnik map tile generator that produces cached and dynamic instances of base map tiles for use in map clients.
        iii. An instance of Geoserver, an Open Geospatial Consortium (OGC) compliant instance of open standards such as Web Mapping Service (WMS), Web Feature Service (WFS), and Web Coverage Service (WCS), for the sharing of geospatial data. Geoserver is configured to serve geospatial data from both the OSM data

[2] NB: SimICI is predicated on the EPSG:4326 coordinate reference system on the WGS84 reference ellipsoid. This is also known, in shorthand, as the WGS84 datum. Geographic and geometric data held in other projections will be re-projected to the WGS84 datum at the point of ingestion. Obviously WGS84 datum source materials are preferred!

model *and* any shapefile sources as may be uploaded (via a RESTful interface) to the Geoserver instance.

iv. The Apache httpd (with mod_tile extension to serve Mapnik generated tiles) and Tomcat web servers for the exposure of geospatial services to map clients and other consumers of SimICI geospatial services.

c. Delivers:

i. Base map tiles for consumption by map clients.

ii. Overlay layers as derived from the OSM data model, the bespoke extensions to that data model, and the shapefile sources registered in Geoserver. Overlays may be delivered to clients in any of WMS(-T) or WFS(-T) formats to suit the client's objective.

iii. Map features, delivered through WFS, in shapefile format. Intended to support consumers of shapefile resources (eg: EU-CIRCLE CIRP) generated from within SimICI.

iv. Map features, delivered through WFS, in JavaScript Object Notation (JSON) and/or JSON with Padding (JSONP) format. Map features delivered through this route are intended to be processed on the client rather than simply displayed.

2. ***Data Exchange Services:***

a. Derived primarily from the delivery routes of the Geospatial Services component.

b. Key Services:

i. Shapefile formatted WFS outputs for consumption by EU-CIRCLE CIRP and other shapefile consumers as may be employed against SimICI. Typically employed to feed SimICI generated data, such as bespoke assets, to EU-CIRCLE CIRP analyses for processing.

ii. Base map, WMS(-T), and WFS(-T) overlay layers for consumption by the SimICI web application and other OGC compliant layer consumers. Typically employed to provide mapping interfaces as a key aspect of the SimICI User Experience.

iii. JSON(P) formatted WFS outputs for consumption by the SimICI web application, the SimICI Cascade Engine, and other consumers of JSON formatted geospatial data. Typically employed to provide geo-correlated feature data to the SimICI web application in order to generate interactive resources that are able to be programmatically manipulated by SimICI.

iv. Ingestion of shapefiles output from EU-CIRCLE CIRP analyses into the SimICI Cascade Engine and, subsequently, the SimICI web application. Typically employed to incorporate the results of an EU-CIRCLE CIRP analyses within a SimICI session.

v. Generation of specifications and prototype implementation code from within SimICI for transfer to the EU-CIRCLE CIRP change management process. This will be a file output that can be passed to EU-CIRCLE CIRP developers who will then implement a formal analysis, with all required extension endpoint registrations and supports, as part of EU-CIRCLE CIRP.

3. ***SimICI Web Application:***

a. Standards compliant HTML5/CSS3 web application forming the primary User Experience through which SimICI will be exercised.

b. Delivered as a standard web application, supported by most modern web browsers, that supports multiple simultaneous users. User authentication is by user id/password pair providing access to the SimICI web application within which group-user-role based permissioning is used to control user access to the various functions available within SimICI.

c. Provides:

   i. A 2/2.5D, OGC compliant web map environment within which geospatial data, as employed within SimICI, can be visualised, interrogated, manipulated, and extended to suit the purposes of each user. Contains standard web map controls (eg: zoom, layers, etc.) alongside auxiliary controls for geometry management (ie: drawing on the map), time series visualisation, searching and filtering, and user defaults management.

   ii. A 3D globe web map supporting enhanced visualisation of SimICI geospatial data and providing perspective support to those SimICI users generating bespoke geospatial elements.

   iii. A remote interface to the SimICI Cascade Engine allowing user-defined workflows (eg: what-if analysis) to be designed and executed from within SimICI.

   iv. An extended version of (ii) above that, through the use of a Cascade Engine node type that accepts, validates, and executes user-definable implementations, supports the definition and evaluation of prototype analyses within SimICI prior to the implementation of a formal analysis in EU-CIRCLE CIRP.

   v. An impact control view that is designed to allow user control over impacts and other parameters as may be required to stimulate a workflow within the SimICI Cascade Engine. For example, as part of a prototype analysis, a user may want to explore ranges of outcomes arising from, say, an increase in mean air temperature. Using impact control, such events may be dynamically injected into a SimICI Cascade Engine workflow in order to support such exploration.

   vi. A dashboard visualisation that provides insight into the data points output from any running Cascade Engine workflow that a given user has the right to observe. For example, a Cascade Engine workflow may be established to perform a what-if analysis. The set of data points output by software ‘sensors’ as defined within each node of that workflow is then available for visualisation in dashboard form. The use of dashboards can provide a more intuitive interface for certain classes of users and, moreover, provides evidentiary stimulus for discussions around intervention and mitigation strategies, development policies, and other aspects that EU-CIRCLE is expected to inform.

d. Delivered as a modular application with the option to extend the UX through the implementation of straightforward functional additions that leverage the core services within the SimICI web application.

4. *SimICI Cascade Engine:*

a. SimICI contains an instance of Cascade, XUV’s core platform technology that provides the means to define and execute user-defined analytical pipelines driven by any set of data sources and populated by a wide variety of algorithms and functions.

b. In the context of SimICI, the Cascade Engine component essentially provides a minimal version of the main EU-CIRCLE CIRP framework. The intention is not to create two competing applications but, rather, to provide collaborative exploration and rapid prototyping capabilities, both of which are EU-CIRCLE CIRP compliant, within the SimICI environment.

c. The SimICI Cascade Engine component provides a foundation stone for the integration of additional tools, services, and capabilities as may be introduced by the Critical Infrastructure and open source communities once EU-CIRCLE outreach activities commence.

d. The SimICI Cascade Engine component is designed to support the execution of multiple simultaneous workflows, each of which may be under the control of a specific user or group. This supports the collaborative, multi-user focus of the SimICI web application and provides for rapid set-up and tear-down of workflows, including prototype analyses for later exploitation within the core EU-CIRCLE CIRP framework.

e. Outputs from a SimICI Cascade Engine workflow are accessible via either (a) the SimICI web application mapping interfaces (eg: calculated outcomes from the workflow impact geospatial elements displayed on the map, causing a change in appearance or other alert or alarm) or, in the case of data drawn from sensors within the workflow itself, (b) the SimICI dashboard view.

f. As noted previously, the SimICI Cascade Engine contains a special instance of a Cascade node that is able to accept user input describing its core behaviour; validate that input, and execute the defined core behaviour as part of a larger Cascade workflow. This allows for new, refined, or extended algorithms to be evaluated and approved prior to generation of an analysis task specification for formal implementation as part of EU-CIRCLE CIRP.

3.2 SimICI-CIRP Integration

As noted, there are four integration touchpoints (between SimICI and EU-CIRCLE CIRP). These are:

1. **Shapefiles from CIRP ingested into SimICI Geospatial Services:**
    a. Re-use, via importation, of shapefiles previously defined for use in CIRP.
    b. May be delivered from the EU-CIRCLE CIRP cache or directly uploaded into SimICI.
2. **Shapefiles from SimICI Geospatial Services for use in CIRP:**
    a. Conversely, the provision of shapefiles derived from data generated for within SimICI.
    b. May be delivered to the EU-CIRCLE CIRP cache or manually transferred.
3. **Analysis Task Specifications from SimICI to CIRP:**
    a. As described above, output of a file containing the specification and prototype implementation of an analysis as defined within SimICI.
    b. The file shall then be routed via EU-CIRCLE CIRP change management for formal implementation, quality assurance, and inclusion within the EU-CIRCLE CIRP framework.
4. **Orchestration and Execution of CIRP Analyses from within SimICI (Stretch):**
    a. Despite expectations leading to the base concept architecture produced at M6 (and provided in Figure 4, previously), there is no readily exploitable Application

Programming Interface (API) that would permit a CIRP analysis task to be remotely orchestrated and executed.

b. Despite this, and after additional investigation against the ERGO framework source code, XUV have identified a mechanism that may support the required functionality despite the lack of API access. It is stressed that this is a stretch goal, albeit one that would represent a solid advance in the state of the art.

c. It is proposed that a SimICI Cascade Engine node be developed that is capable of orchestrating and executing any available basic analysis held within the collocated instance of the EU-CIRCLE CIRP application. To do this, we will leverage the ERGO AnalysisManager (edu.illinois.ncsa.ergo.core.analysis.AnalysisManager) class and, through its executeAnalysis method, the underlying OgreScriptBaseTask class that is the basic object beneath the general purpose AnalysisBaseTask upon which all ERGO, and therefore CIRP, analysis tasks are built.

d. In order to address this stretch goal, XUV will acquire an instance of the EU-CIRCLE CIRP source code from STWYS (or work with a STWYS technical point of contact to ensure correct versioning of source, libraries, and other aspects across the technical partners). This instance will, as mentioned previously, be collocated with SimICI and provide an entry point for the SimICI Cascade Engine to leverage and exploit EU-CIRCLE resources as part of any what-if, refinement, or prototyping activities.

The SimICI Environment will continue to evolve and expand as additional outputs from the EU-CIRCLE project are implemented within it; as more widespread testing and evaluation activities – up to and including the EU-CIRCLE case studies of WP6 – take place, and as the EU-CIRCLE outreach program commences.

4 GIS and Data

As described in Section 3, Architecture, previously, SimICI integrates a number of technologies that collectively provide GIS services to SimICI. Those technologies have been selected from open-source providers and integrated to address the design considerations expressed in Section 2, Background, previously.

SimICI also contains the reference dataset for interconnected critical infrastructures as represented by the current Rethymno-based 'virtual city'. It is expected that (a) that dataset will continue to evolve through the remainder of the EU-CIRCLE project and (b) that additional reference sites will be integrated into SimICI as and when required.

4.1 GIS Services

Figure 6, below, illustrates the GIS Services model deployed for, and integrated with, SimICI.

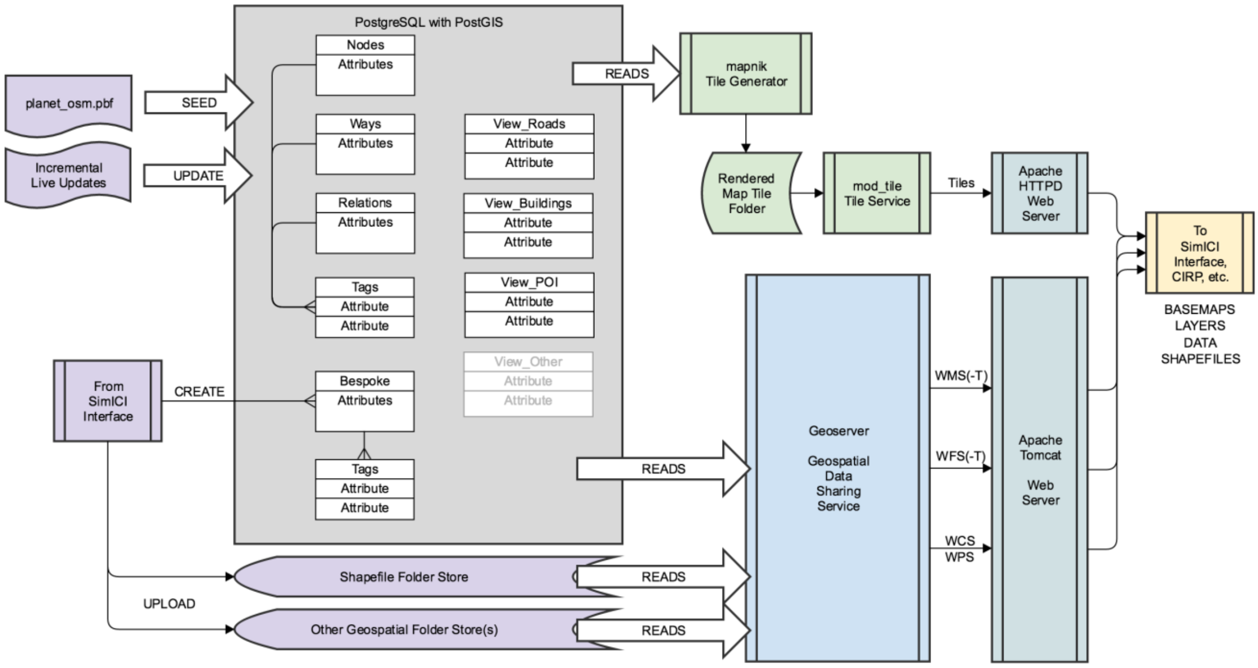


Figure 6: SimICI GIS Services and Data Model

The GIS data lifecycle runs from left to right through Figure 6. In the initial instance, the GIS data is initialised from the crowdsourced Open Street Map database through ingestion of the OSM 'planet' data file into a spatially enabled SQL database. The resulting local copy of the OSM data is then kept up to date through regular subscription to and ingestion of OSM updates. The OSM data covers the whole world: allowing SimICI to be exercised in any Area of Interest.

The local SQL database is populated with the standard OSM data model, the utility of which is described in Section 2, previously. As also described in Section 2, the standard OSM data model is extended with the addition of a 'Bespoke' table (with associated Tags) that acts as a repository for custom artefacts (eg: CI assets, buildings, networks, etc.) that are generated within SimICI. This additional table provides the opportunity to generate truly virtual assets and networks that seemingly exist within fully geo-correlated environments that may then be exploited as test cases within SimICI and, through the data integration pathways, within CIRP.

The resulting SimICI data model is then extended with views on the OSM and bespoke data that provide discrete overlays for buildings, roads, points of interest, and so on. These views are used, amongst other purposes, to drive discrete overlays than may then be differentially rendered in the SimICI user interfaces.

For example, a discrete region of buildings may be requested from SimICI – using the industry standard zoom/x/y naming convention or through the use of a Bounding Box query – and rendered as flat shapes, indicating coverage, within the SimICI 2D view but as extruded shapes, both indicating coverage and providing familiar context, in the SimICI 3D view. This approach means that a single data source may be exploited in numerous ways: removing duplication and decreasing GIS data management overheads.

The local SimICI database, which is primarily used to generate base layer map tiles, is supplemented by local repositories for Shapefiles (ie: data that may be originated within external GIS systems and that are capable of being used within CIRP) and other relevant data objects as may be required (eg: NetCDF files representing EU-CORDEX data points). Population of these repositories is, as with the generation of custom artefacts in the SimICI GIS database, controlled through the main SimICI interface. This abstracts away the requirement for dedicated data management services in the GIS component of SimICI and supports the integrated component approach.

The set of SimICI data, as held in both the database and the folder repositories, is then processed for exploitation through two routes.

In the case of base layer map tiles, the OSM data is rendered into a pyramid of tiles each 256 pixels wide. This pyramid represents the entire planet across 20 levels of zoom. At zoom level 0 within this pyramid, the entire globe is depicted on a single map tile (approximately 1:500 million scale) while, at level 19, the surface of the globe is depicted in 274,877,906,944 tiles (approximately 1:1,000 scale). This provides for a maximum resolution of 29.8cm per pixel (approximately 10m per cm of map) which supports highly granular placement and scaling of critical infrastructure resources in SimICI.

For SimICI, zoom levels 0 through 15 are pre-rendered and stored in a tile repository folder. Additional zoom levels, and updated tiles, are dynamically rendered and, once rendered, are cached in the tile repository folder for future use until such time as they are updated. The tile repository folder is then served, via the mod_tile plugin, through the standard Apache HTTP web server. Tiles are retrieved through standard calls using the zoom/x/y.png nomenclature that directly references the tile pyramid.

For overlay layers (ie: those layers that may be rendered on top of the base map and which typically represent specific features within the map), exploitation is achieved through the open-source Geoserver platform for sharing spatial data.

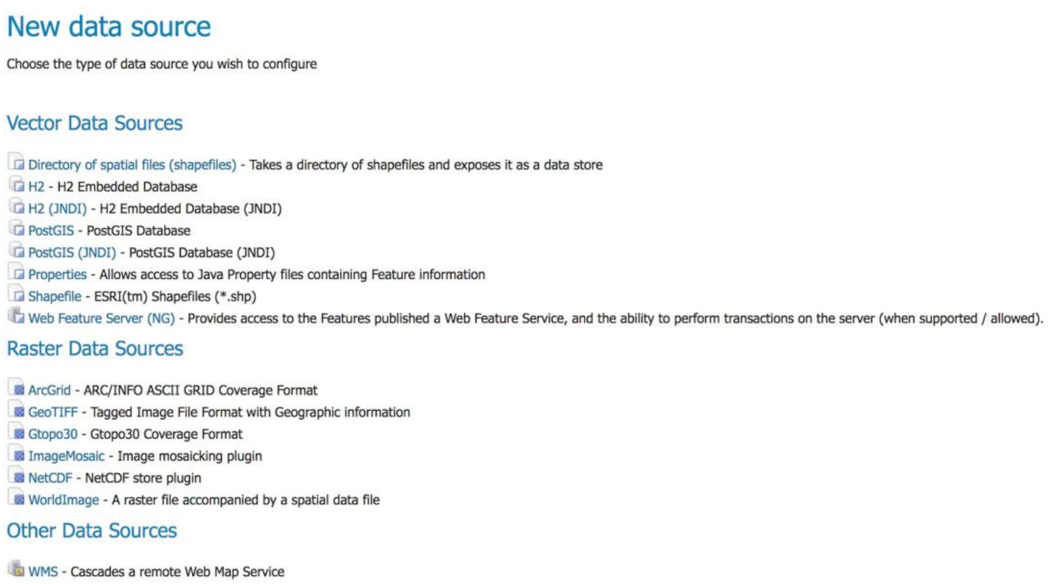
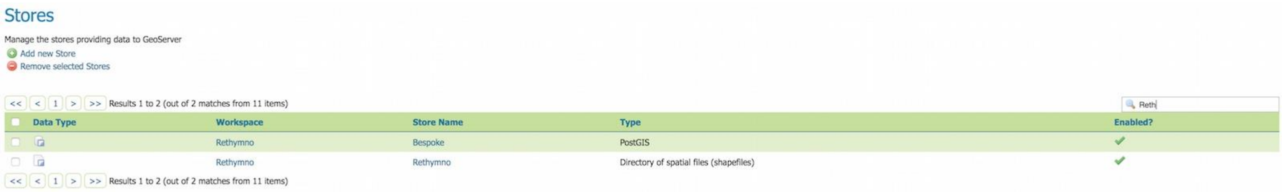


Figure 7: SimICI GIS Services - Supported Data Stores

Figure 7, above, indicates the default set of data sources exploitable by Geoserver version 2.9.3 (as deployed for SimICI); augmented with the addition of support for NetCDF formatted raster data files as will be used within EU-CIRCLE.

A data source is connected for use with Geoserver through the definition of a data store. A data store is simply a single point of reference for the connectivity protocol, including any authentication, that exposes the contents of a data source. For example, for the Shapefile and other file repositories, the data store will simply define the folder in which the data resides. For a PostGIS enabled database, the data store specifies the network location of the database server, the database containing the spatial data, and the authentication credentials to be used when accessing that database.


Stores

Manage the stores providing data to GeoServer

Add new Store

Remove selected Stores

<< < 1 > >> Results 1 to 2 (out of 2 matches from 11 items)

Data Type	Workspace	Store Name	Type	Enabled?
	Rethymno	Bespoke	PostGIS	✓
	Rethymno	Rethymno	Directory of spatial files (shapefiles)	✓

<< < 1 > >> Results 1 to 2 (out of 2 matches from 11 items)

Figure 8: SimICI Data Stores

Figure 8, above, details the data stores enabled in SimICI for the Rethymno ‘virtual city’ dataset. As can be seen, one store is defined for the custom artefacts stored in the Bespoke table of the SimICI databse and another is defined for the Shapefile repository folder.

It should be noted that Figure 8 shows a data store listing filtered for Rethymno. The data store representing the views into the OSM data are therefore not displayed in this image. Section 5, pursuant, contains user interface screenshots that illustrate the combination of real-world OSM data as the base map with overlays derived from the Rethymno ‘virtual city’ dataset.

Once a data store is enabled, layers may then be generated from the information held within that store.


Layers

Manage the layers being published by GeoServer

Add a new layer

Remove selected layers

<< < 1 > >> Results 1 to 25 (out of 17 matches from 36 items)

Type	Title	Name	Store	Enabled	Native SRS
	oilmarineterm	Rethymno:oilmarineterm	Rethymno	✓	EPSG:4326
	oilpipel	Rethymno:oilpipel	Rethymno	✓	EPSG:4326
	oilpumbST	Rethymno:oilpumbST	Rethymno	✓	EPSG:4326
	oilstorageF	Rethymno:oilstorageF	Rethymno	✓	EPSG:4326
	photoVolt	Rethymno:photoVolt	Rethymno	✓	EPSG:4326
	portpoly2	Rethymno:portpoly2	Rethymno	✓	EPSG:4326
	stepd_sub150_20	Rethymno:stepd_sub150_20	Rethymno	✓	EPSG:4326
	stepd_sub20_230v	Rethymno:stepd_sub20_230v	Rethymno	✓	EPSG:4326
	stepd_subUn150_20v	Rethymno:stepd_subUn150_20v	Rethymno	✓	EPSG:4326
	StepUP_sub150	Rethymno:StepUP_sub150	Rethymno	✓	EPSG:4326
	trHvolt_tower500	Rethymno:trHvolt_tower500	Rethymno	✓	EPSG:4326
	voltHilineb	Rethymno:voltHilineb	Rethymno	✓	EPSG:4326
	voltHilineov500m	Rethymno:voltHilineov500m	Rethymno	✓	EPSG:4326
	voltHilineov500mP	Rethymno:voltHilineov500mP	Rethymno	✓	EPSG:4326
	voltlineb	Rethymno:voltlineb	Rethymno	✓	EPSG:4326
	voltlineo1	Rethymno:voltlineo1	Rethymno	✓	EPSG:4326
	windGen	Rethymno:windGen	Rethymno	✓	EPSG:4326

<< < 1 > >> Results 1 to 25 (out of 17 matches from 36 items)

Figure 9: Rethymno Virtual City Data (Shapefiles) as GIS Layers

Figure 9, above, details the layers currently generated in SimICI for the Rethymno ‘virtual city’ dataset. As can be seen, these layers are all generated from the Rethymno Shapefile repository data store. Due to filtering as previously, layers generated from the views on the OSM data are not visible in this image.

It is worth noting the ‘Type’ column at the far left of Figure 9. This column pictorially represents the nature of the data contained within each layer definition. From this, we observe that the available layers are of the well known geometry types Point, Polyline, and Polygon. It is also worth noting the ‘Native SRS’ column at the far right of Figure 9. While this shows that the data forming the Rethymno ‘virtual city’ is all in the WGS84 datum (EPSG:4326), it should be observed that different native Spatial Reference Systems may be used in the source data and, subsequently, reprojected as required.

Once a layer is defined within Geoserver, it is then available to clients exploiting that Geoserver as a service endpoint. Geoserver is deployed as an application within the standard Apache Tomcat web application container and is accessible through a well documented RESTful interface employing standard HTTP verbs.

Geoserver supports - and, indeed, is the reference implementation for - a number of Open Geospatial Consortium standards. Those OGC standards include the Web Mapping Service (ie: the request and delivery of pre-rendered map data in image format) and the Web Feature Service (ie: the request and delivery of map feature data). The latter standard supports the delivery of features which, to draw an analogy, may be thought of as the 'source code behind a map'.

Features are made available through WFS for the purposes of spatial-analysis or, in the case of the WFS-T variant, for editing. For the purposes of SimICI, feature layers are requested and subsequently manipulated in JSONP format and, for the purposes of supporting data integration with CIRP, in Shapefile format. In the former case, feature data is programmatically manipulated within the SimICI application. In the latter case, requested Shapefiles may be downloaded and subsequently transferred to CIRP for use in a CIRP scenario.

Reference to the preceding text indicates that there is a fair degree of complexity involved in exposing data through the SimICI GIS Services component. This is recognised and, in order to simplify the use of SimICI, the design abstracts away the Geoserver processes (with the exception of setting up the initial data stores). This abstraction is achieved through the integration of the Geoserver REST API, allowing data upload and editing in addition to map and feature requests, into the main SimICI application.

As a result, for example, an additional Shapefile may be uploaded through the SimICI application and will then automatically migrate to the Geoserver data store and have a layer generated from it. This technique removes the complexity, and associated psychological nervousness, from the SimICI user community and directly supports the principle of a 'single-window' application as a result.

It should be noted that the REST API is also exploited by the SimICI services that support the generation of custom artefacts for use within any given AOI. Such artefacts, declared visually in SimICI, are written back to the Bespoke table in the SimICI database from where, as required, they may be exploited either programmatically (in future exercises with SimICI) or, using the Shapefile delivery format, in CIRP via the data integration pathway.

At the time of writing, SimICI is populated with a complete and regularly updating instance of the OSM dataset and with the Rethymno 'virtual city' dataset held in Shapefiles. There are currently no custom artefacts or other data files. This is due to the newness of SimICI and will resolve over the remainder of the EU-CIRCLE project as assets, networks, hazards, impacts, and other resources are defined for exploitation by EU-CIRCLE.

4.2 SimICI Data – Rethymno 'Virtual City'

NCSRD, as the lead for Task 7.3, have generated an initial 'virtual city' dataset containing interconnected critical infrastructure assets and networks. The dataset is based in the real-world environment of Rethymno and has been ingested for use within SimICI.

4.2.1 Definition of Reference Data Sets

A number of data were developed for the establishment of the SimICI data set (based on the Greek city of Rethymno – Crete and its wider area) which is analysed below. Some of the data, for the city of Rethymno, were partially provided by Associate Professor Dr. Christos Makropoulos, Department of Water Resources and Environmental Engineering, School of Civil Engineering, NTUA, after personal communication.

Data were created from Open Sources used for the collection of data (e.g Open Greek Geodata, CORINE Land Cover) while the elaboration of data and the construction of the virtual datasets were based on bibliography (Makropoulos C. et al., 2015[3]) and available statistics by Hellenic Statistical Authority. The

[3] Makropoulos C.; Tsoukala V.; Belibbasakis K.; Lykou A.; Chondros M.; Gourgoura P.; Nikolopoulos D. (2014). Managing flood risk in

dimensioning of the constructed Infrastructure Networks was performed based on real data available in bibliography.

4.2.2 Main Infrastructure

Table 1, below, presents the main infrastructure elements which have been constructed:

Name	EU-CIRCLE Use	Source	Short description
Drinking water network	Main infrastructure	virtual	The drinking water network was created (not a hydraulic equivalent of the actual network) based on the road network. A water treatment plant and pumping stations were added trying to depict assets of drinking water infrastructure
Wastewater network	Main infrastructure	virtual	A virtual wastewater network was created (not a hydraulic equivalent of the actual network) based on the road network. A wastewater treatment plant, pumping stations and control vaults were added trying to depict assets of the wastewater infrastructure
Electricity network	Main infrastructure	virtual	The virtual Electricity Network was created based on Road Network and the spatial distribution of the basic city components and services.
Oil network	Main infrastructure	virtual	A virtual Oil Network was created based on the spatial distribution of relevant city components and services (e.g port)
Road network	Main infrastructure	Open Street Map	National and Major road network.
Govermental buildings and Health Services	Main infrastructure	virtual	

Table 1: Main infrastructure elements of the Virtual Environment Datasets

coastal cities through an integrated modelling framework supporting stakeholders' involvement: the case of Rethymno, Crete. 36th IAHR Worl Congrees (IAHR 2015), 28 June-3 July. 2015, The Hague, the Netherlands

4.2.3 Auxiliary Data

Auxiliary spatial data which will be used as basic inputs for the implementation of the Reference Scenarios have been organized as part of the Virtual Environment Datasets. These are described in Table 2, below:

Name	EU-CIRCLE Use	Source of acquisition	Short description
Buildings	**Auxiliary data**	virtual	Buildings for Residential, Commercial use.
Topography	**Auxiliary data**	National Geodatabase (http://geodata.gov.gr)	A DTM of 50m spatial analysis has been created for the wildland part of the site area. In addition, the percentage of Slope and Aspect layers were created from this DTM in order to be used as inputs to the fire Risk and behaviour modelling.
Vegetation and Fuel map	**Auxiliary data**	Corine Land Cover 2012	CLC 2012 was reclassified for the creation of Forest Fuel map of a raster format and spatial analysis of 50m, for the wildland area

Table 2: Auxiliary data of the Virtual Environment Datasets

The geoview of the cumulative dataset resulting from Tables 1 and 2, with all the Infrastructure and Auxiliary elements of the Virtual City is presented in Figure 6, below.

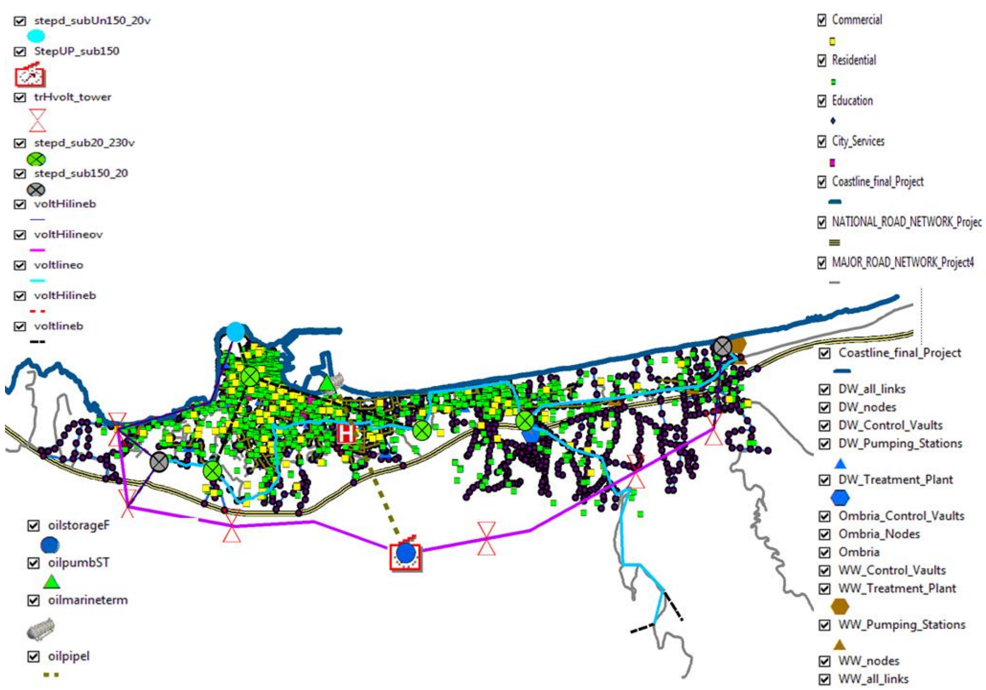


Figure 10: Geoview of the cumulative datasets comprising the Virtual City

4.2.4 **Build of Reference Scenarios**

SimICI will support the build of up to 5 reference scenarios (eg: flood, forest fire, drought, ports, and transportation) to be exercised in evaluation of the EU-CIRCLE outputs. With regard to this aspect, NCSRD has been conducting work in relation to the collection, creation and elaboration of the required datasets (infrastructure, assets, auxiliary data) for the implementation of the Reference Scenarios.

Data produced to date exists in the NetCDF format which, while a well known format, nevertheless requires defined Spatial Reference System information to be configured before it can be exploited through the Geoserver component. This work remains ongoing but, for the purposes of this document we provide details of the data generated to date.

4.2.5 **Definition of Weather Scenarios**

Meteorological data from CORDEX data base for three climatic scenarios (RCP2.6, RCP4.5, RCP8.5) have been acquired and elaborated for the definition of the weather scenarios for the Reference Scenarios.

Meteorological data for the years 2006-2050 for three climatic scenarios were acquired from CORDEX database for several points within the area of interest. As a first step of the definition of the Weather scenarios to be used in EU-CIRCLE, Meteorological Time series were created for these points for each climatic scenario. Time Series diagrams of the daily mean values of precipitation, temperature and wind for a representative point for the climatic scenarios RCP2.6, RCP4.5 and RCP8.5 are presented in the following charts (Figures 11, 12, and 13):

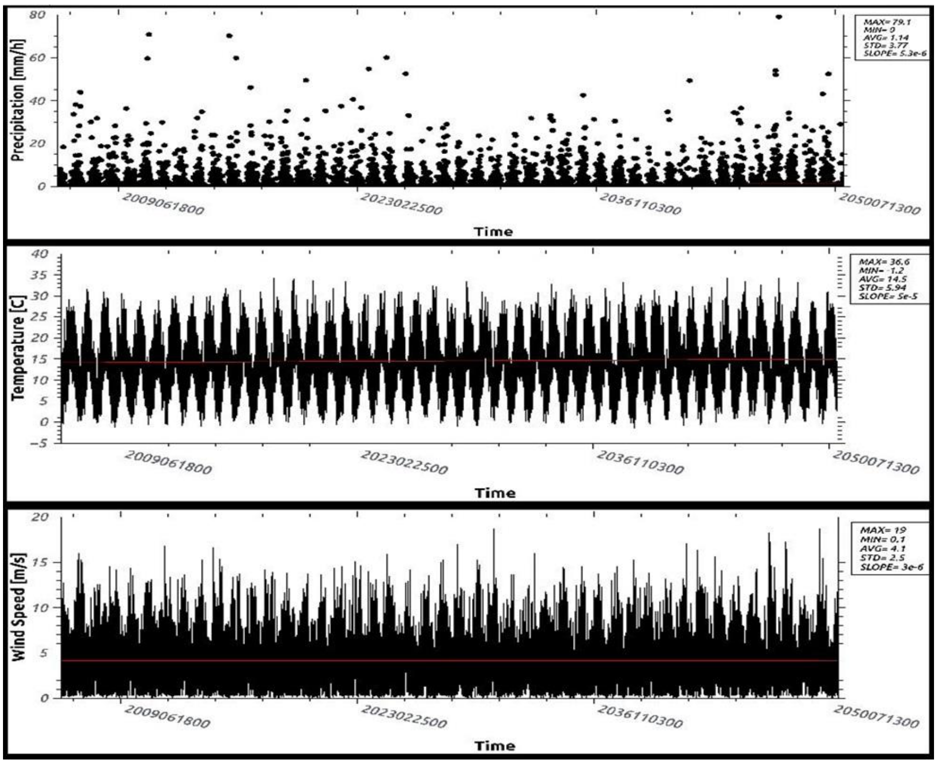


Figure 11: Time Series diagrams of the daily values of precipitation and 3-hour interval temperature and wind values for a representative point, for the climatic scenario RCP2.6

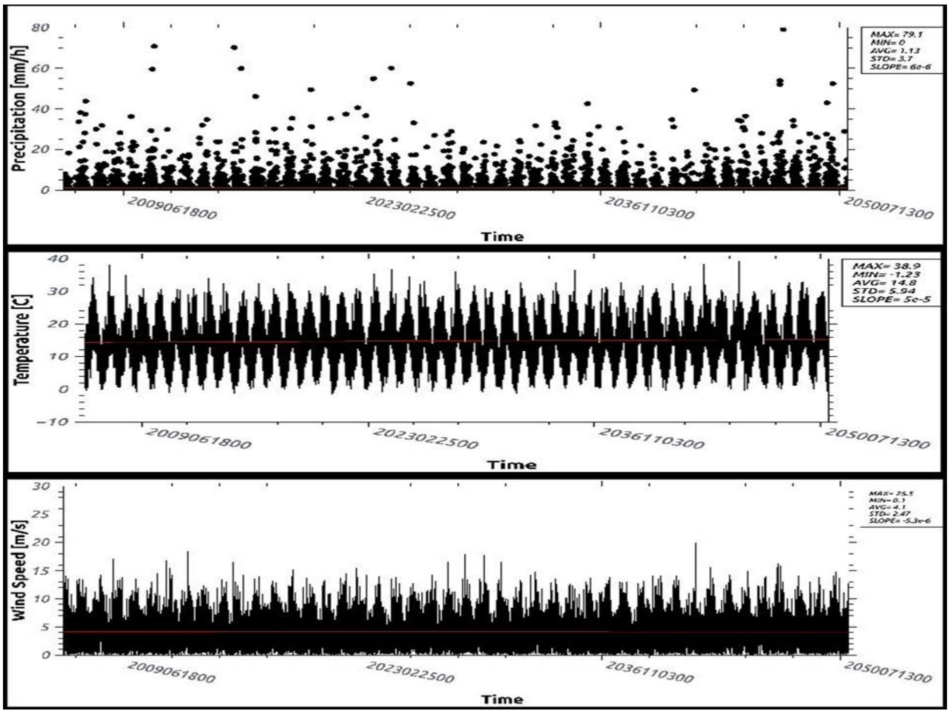


Figure 12: Time Series diagrams of the daily values of precipitation and 3-hour interval temperature and wind values for a representative point, for the climatic scenario RCP4.5

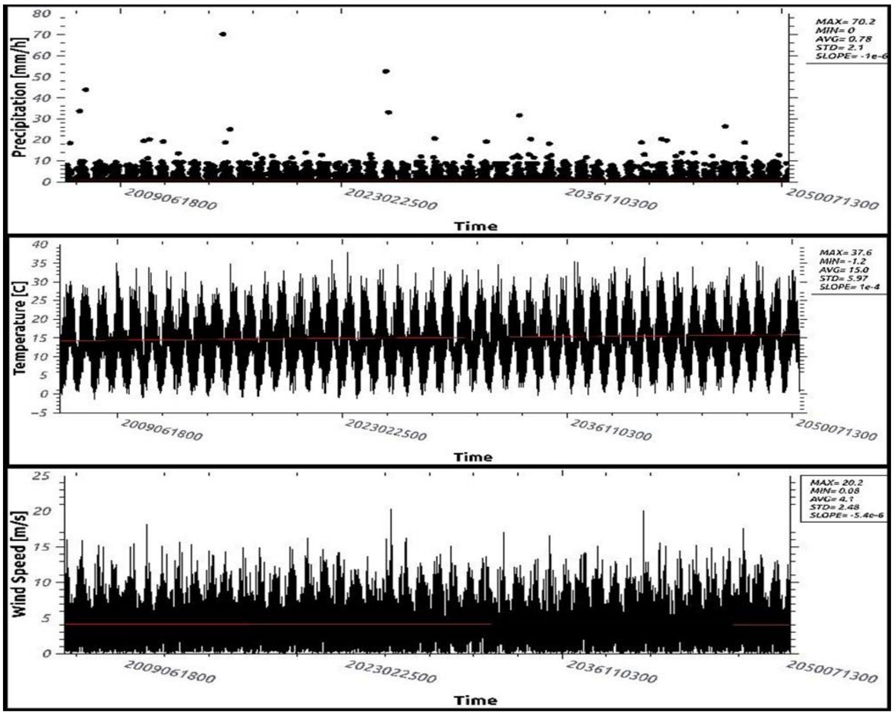


Figure 13: Time Series diagrams of the daily values of precipitation and 3-hour interval temperature and wind values for a representative point, for the climatic scenario RCP8.5

4.2.6 Floods

One of the objectives of the Flood Reference Scenario is the impact assessment of rainfall events for the Virtual City. As a first step for this Reference Scenario, the results of flood simulations were acquired (Figure 14). These results are the product of coupled hydraulic models for 1D/2D simulation for the urban area that enable hazard assessment from flooding derived from rivers, urban, drainage and sewer systems (Makropoulos et al. 2014[4]). The rainfall event used for the flood simulation was that of November 10th, 1999 rainfall event, which has been estimated to have a return period of 100 years (T = 100).

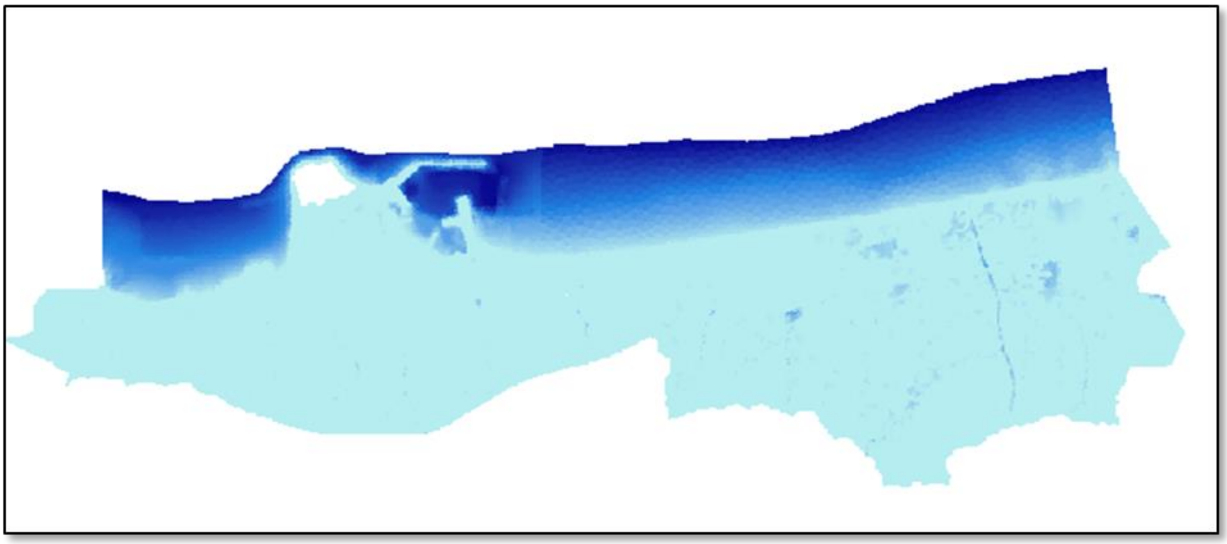


Figure 14: Flood simulation of the rainfall event of the November 10th , 1999

(Lykou and Makropoulos, 2015[5])

4.2.7 Forest Fires

The Forest Fires Reference Scenario will be based on the calculation and study of the Canadian Fire Weather Index and the Fire Behaviour estimation using climatic forecast data. The work performed so far for this Reference Scenario has been concerned with the calculation of daily and weekly FWI maps for the whole of Greece - in which Rethymno region belongs - for the Summer of 2016, using seasonal weather forecasts. The weekly maps presenting FWI classes for four weeks of Summer 2016 are presented in Figure 15.

[4] Makropoulos C.; Tsoukala V.; Belibbasakis K.; Lykou A.; Chondros M.; Gourgoura P.; Nikolopouloe case of Rethymno, Crete. 36s D. (2014). Managing flood risk in coastal cities through an integrated modelling framework supporting stakeholders' involvement: the case of Rethymno, Crete. 36th IAHR World Congress (IAHR 2015), 28 June-3 July. 2015
, The Hague, the Netherlands

[5] Lykou Archontia and Makropoulos Christos, Technical report on the city of Rethymno flood modelling of the rainfall event of the No

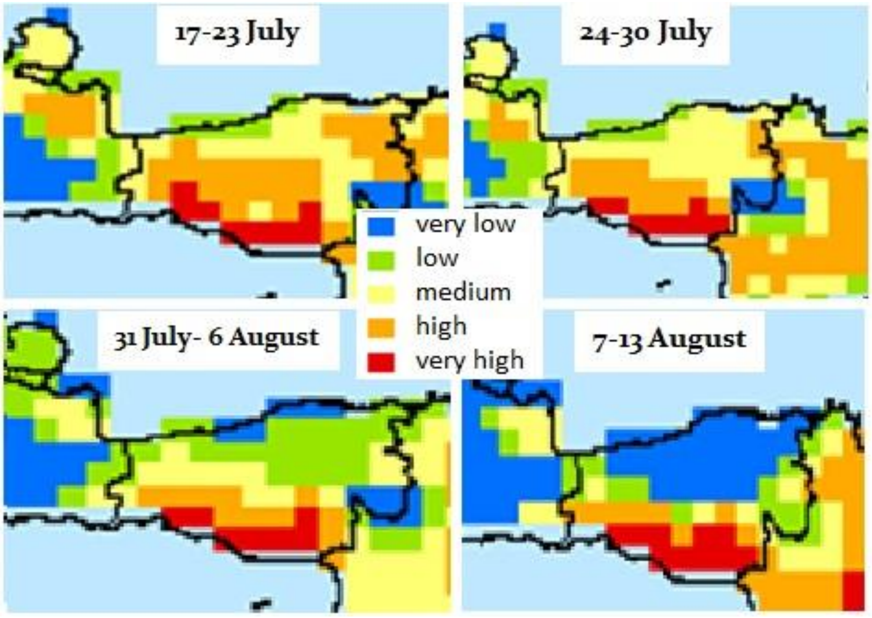


Figure 15: Weekly maps of FWI classes for Summer 2016, using seasonal weather forecasts

4.3 Integration of Rethymno 'Virtual City' Dataset in SimICI

As stated above, the Rethymno 'virtual city' dataset has been integrated into SimICI as:

1. Critical Infrastructure Assets and Networks:
    a. Shapefile formatted data, held in the SimICI GIS Shapefile repository
    b. Shapefile repository exposed as the 'Rethymno' data store
    c. Layers generated fro all current 'virtual city' Shapefiles
2. Custom Artefacts:
    a. None created to date.
3. Reference Scenario Supports:
    a. Data produced in NetCDF format (ongoing work)
    b. NetCDF support added to and enabled in SimICI GIS Services (Geoserver)
    c. Additional Spatial Reference System definitions required

5 The SimICI User Experience

SimICI is implemented as a 'single-window', multi-user web application that provides intuitive interfaces through which technical and non-technical users can collaborate in order to explore scenarios, investigate risks and intervention strategies, and perform both output validation and prototyping in relation to CIRP processes.

SimICI presents as a web application, accessible through any modern web browser, that contains a number of 'pages' linked through a menu bar. Each page is tailored to a different objective and, moreover, access to each page can be controlled on a group-user-role permissions basis. Thus, with SimICI, it is possible to have multiple users connected to the system and, through permissions control, permit one user to define and execute impact tests while all other users are restricted to observing the results of the impact tests in either 2D or 3D views. Equally, it is possible to have multiple users connected to the system, each of whom has full access and each of whom is exploring their own scenarios and objectives.

This approach gives SimICI flexibility in relation to its modes of use. The same implementation may be used to lead a workshop, support an individual line of enquiry, or – in an extreme context – to design a future environment and test it for exposure to risks due to climate hazards. Moreover, any or all of those uses may take place simultaneously.

As noted previously in this document, the SimICI UX is designed to abstract away underlying complexity: allowing the user to focus on the task at hand rather than on acquiring, normalising, and exposing data and/or functions for use within SimICI. This abstraction, which is present in relation to both the SimICI GIS Services and the Xuvasi Cascade components of SimICI, works to flatten the well-observed 'Bathtub Curve' effect whereby the time available to perform an analytical task is often but a small proportion of the total time allocated to a task (the balance being consumed by preparation and reporting requirements).

From an administrative perspective, SimICI requires configuration with regard to the GIS Services and Cascade endpoints that it is to communicate with (for data and services respectively). Thereafter, other than user account management, there is no additional administrative load.

Once the endpoints are configured, using a simple web-based form that is available to the SimICI administrative roles, the SimICI application brokers communication with those endpoints through standard, documented RESTful APIs. The SimICI application is aware of each user's context, in relation to SimICI at least, and ensures that state transfers between interfaces maintain continuity from a user's perspective. That is, that switching from a 2D view to a 3D view will maintain the location, zoom level, and visible data from one view to another. Similarly, if a user switches from a 2D view to a dashboard and then back again, the state of the 2D view will be preserved.

Individual user preferences (eg: geographic area of interest, zoom level, data layers, filters, etc.) are stored against each user's profile in the SimICI database. Users have control over their default preferences and may switch preferences as often as they wish. Users are provided with a 'My Account' style menu option within the SimICI menu bar. Selecting this drops down a menu that is tailored to the role(s) that the current has. For example, a user with the administrator role will see the 'System Settings' option while a user with the publisher role will see the 'Upload Shapefile' option.

While the formal Administration and User Manuals are a current work in progress, which will go into more detail on the functionality of each SimICI interface, the following sections provide a brief overview of each interface and its purpose.

5.1 2D Interface

Figure 16 provides a screenshot of the 2D interface contained within SimICI.

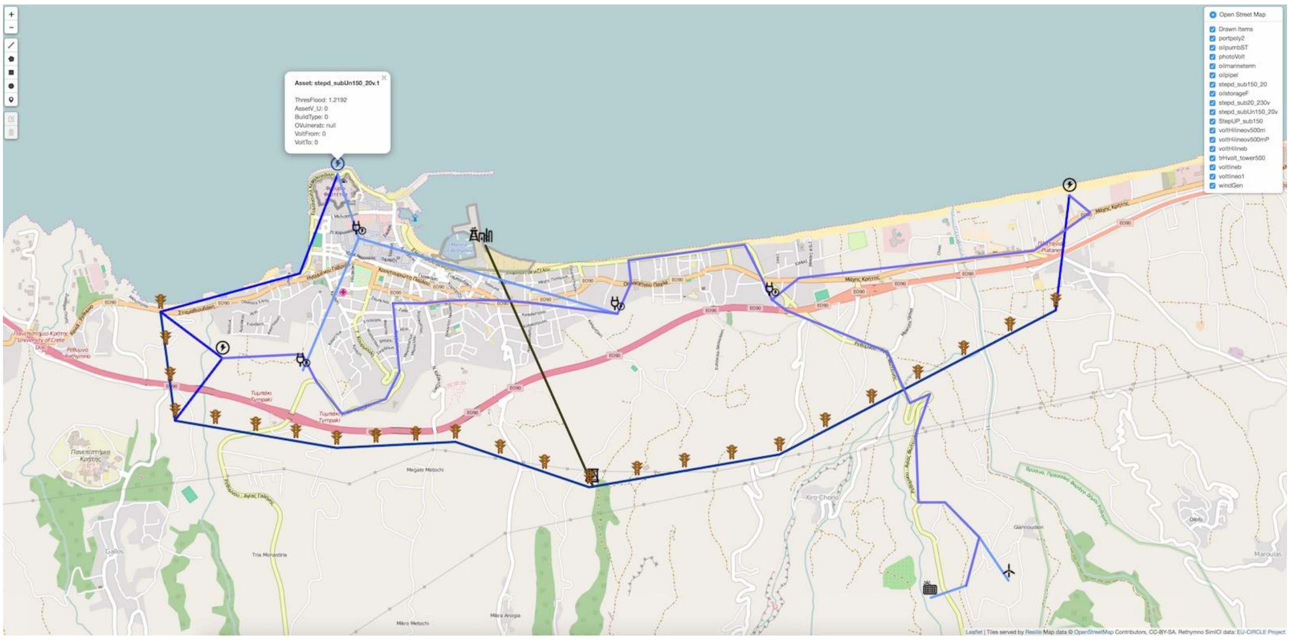


Figure 16: SimICI 2D Interface

The 2D interface is intended for plan view style activities and, essentially, provides a cut-down GIS system as the means to do this. As shown, the 2D interface is built from a base map, drawn from the SimICI instance of the OSM database. The base map is then augmented with the set of layers relevant to the area of interest (in this case, the Rethymno 'virtual city' dataset).

Layers may be switched on or off as required using the layer switcher tool shown in the top right of Figure 16. This layer list is dynamically populated when the 2D interface is first launched and, thereafter, is driven by the user's context preferences as stored within SimICI.

Layers are styled for presentation purposes. In Figure 16, the different instances of the polyline and polygon well known types are differently styled and, in the same manner, each different class of point types is associated with a specific icon that represents the real-world asset at that point. Each feature within a layer is, as shown in Figure 16, associated with a clickable popup that provides information on both the attributes of a feature and its current state in the context and environment it is in.

The 2D interface also supports the drawing of new geometries, again using the polyline, polygon, and point well known types. For polygons, the drawing of rectangular, circular, or irregular polygons is supported. The drawing tool is shown at the top left of Figure 16 and permits the creation of new geometries that, on creation, prompt the user for additional information: including whether or not the geometry should be saved as a custom artefact. If saved as a custom artefact, the new geometry is then transparently written to the underlying SimICI GIS Services and stored, with its attributes, in the Bespoke table. In subsequent uses, the custom artefact will be displayed according to its class style (eg: house, pumping station, etc.).

The 2D interface exploits the JSONP output from the SimICI GIS Services in order to support the programmatic manipulation of items. Such manipulation may include simple user interaction, editing of the item's attributes (eg: moving an asset upwards or to a different location), or dynamic updates to the item's visible state as the result of an impact assessment being run in the Cascade Engine component of SimICI.

Additional tools and controls are available for the 2D interface. These include time-series playback of data resources (for example, those arising from a WMS(-T) GIS service) that allow controlled exploration of changing risk over time; side-by-side split maps allowing before and after event perspectives to be viewed and analysed, and embeddable data visualisation services that allow abstract contextual data to be associated with, and visualised against, any specific feature.

Additional tools will be added to the 2D interface as more detail on the reference scenarios is developed within EU-CIRCLE.

5.2 3D Interface

In contrast, Figure 17, below, illustrates the SimICI 3D interface.

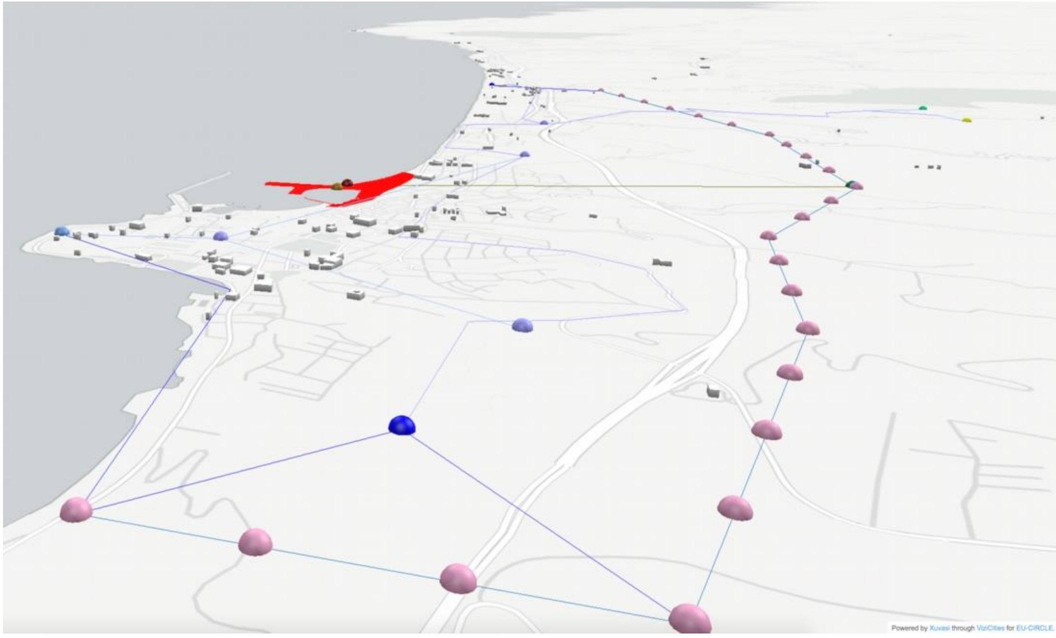


Figure 17: SimICI 3D Interface

The SimICI 3D view is primarily intended for use as a visualisation tool: one with the benefit of height in order that, for example, flood risk zones may be readily identified. In Figure 17, the 3D interface is shown as being built on top of a base map, again derived from OSM data, with feature layers rendered according to class style and well known type.

Figure 17 is shown populated with the OSM-derived buildings layer (in which crowdsourced building geometry from OSM is extruded and rendered in multi-face grey). Custom artefacts representing buildings, as may be created in the SimICI 2D interface, will be extruded and rendered in the same manner. As with the feature layers in the 2D interface, each feature is capable of being manipulated programmatically and, therefore, may change colour or representation subject to impacts being determined and applied in SimICI.

Figure 17 also shows the rendered layers for the Rethymno 'virtual city' dataset. As previously, each layer is distinct and each feature is (a) styled according to its class and (b) able to be programmatically manipulated. It will be obvious from Figure 17 that the icons for Point features as used in the 2D interface are not currently carried through to the 3D view. This is because the 3D view is a rendered scene graph within which flat 2D icons look rather pitiful.

Once the set of CI asset types catalogued by EU-CIRCLE is complete, 3D icons will be located – or created – for each asset class in order that the 3D interface is presented in a more compelling and user-friendly format. It is also a stretch goal of SimICI that, in keeping with the original Design Concepts around SimCity, it will be possible to drag and drop a 3D icon in the 3D interface and then, by setting a few parameters, have that new feature representation drive the generation of an associated custom artefact.

5.3 Cascade Engine Interface

Figure 18, below, illustrates the Cascade Engine interface within SimICI. It should be noted that, because there are no current Cascade node definitions within SimICI, the node repository and designer window are shown as an interface example only. As EU-CIRCLE progresses specifically in relation to the scenario supports Task in WP7, so the node repository will be populated and the Cascade Engine will become useful.

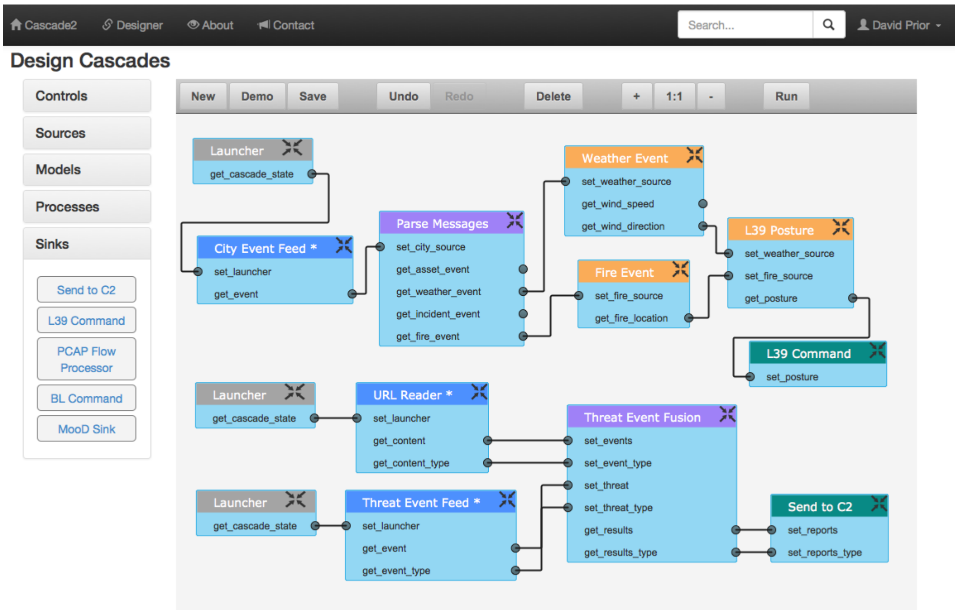


Figure 18: SimICI 'Cascade' Interface (Example)

The Cascade Engine is a drag-and-drop tool that allows workflows to be quickly assembled, tested, and evaluated. SimICI integrates the Cascade Engine as a means of supporting the definition and execution of evaluation scenarios that can be played out in SimICI. The objective of such evaluation scenarios may be to review or otherwise exploit the outputs of a detailed CIRP analysis; to explore the efficacy of a variety of intervention strategies in response to a risk identified by a CIRP analysis, or to explore the potential of a new analysis through rapid prototyping prior to triggering a CIRP change request to add that analysis.

The Cascade Engine interface in SimICI is a split view presenting the set of available nodes (functions) on the left hand sidebar and a blank designer workspace on the right. To set up a workflow, the user simply drags and drops the required nodes onto the designer workspace and then connects the relevant inputs and outputs that define the flow of both data and control. Where available, parameters – such as 'data source to access', 'target for result', etc. – may be set on the nodes by double-clicking.

The resulting cascade (small 'c') may then be launched. To do this, the user simply presses 'Run' in the menu bar and the newly designed cascade is despatched to the Cascade Engine where it is dynamically assembled, compiled, and executed. The execution of the cascade is then under the user's control and it may be stopped, edited, deleted, or restarted as the user requires.

Outputs from the Cascade Engine are, in SimICI, always routed back to the SimICI API in order that impacts on features, whether in 2D or 3D view, may be calculated and the representation of that feature modified accordingly. For example, a building that comes under threat from a flood may turn red and, in the example of interconnected infrastructure, the CI assets within that building may – as a result of the threat event arising, assuming that there is a Cascade Engine workflow watching for that event – begun a decay countdown representing survivability against water penetration. If the countdown expires for a CI asset, eg:

energy substation, then a failure event might be triggered for that asset with the resulting cascading effects that would engender.

As noted, while the Cascade Engine is fully integrated into SimICI, the node repository is currently empty. As the EU-CIRCLE project progresses, so the repository will be populated to, at a minimum, reflect the requirements of the reference scenarios.

5.4 Dashboard Interface

As the node repository for the SimICI Cascade Engine is currently empty, there are no actual Dashboards available to illustrate the function and purpose of the Dashboard interface. Figure 19, however, provides an example of a Dashboard interface from a previous Cyber Resilience research project.

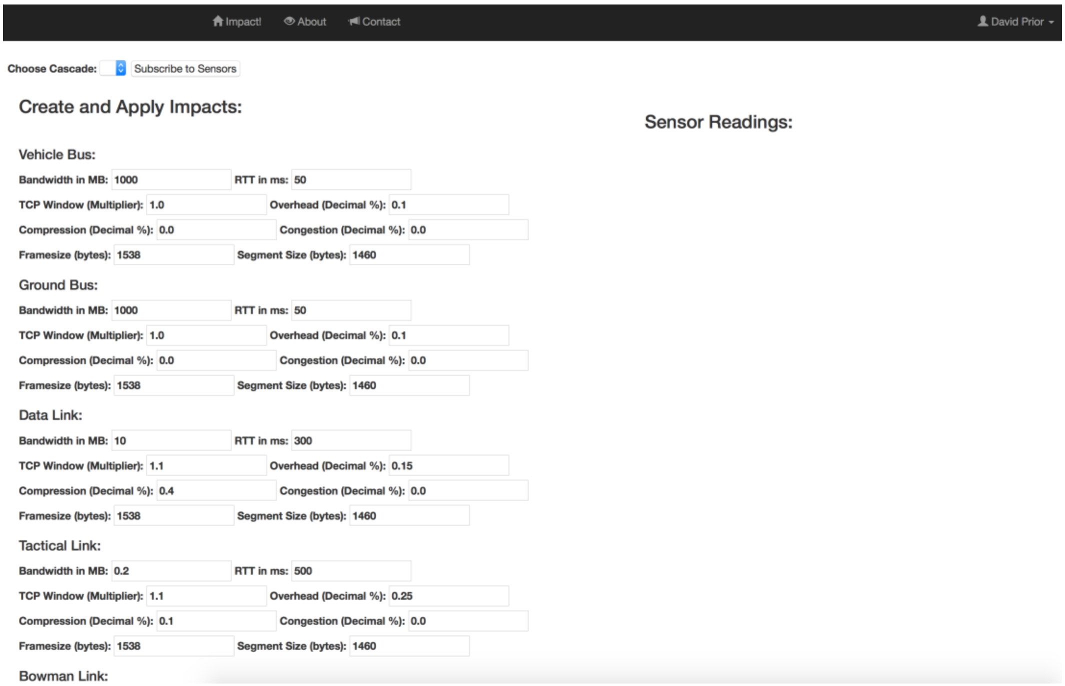


Figure 19: SimICI Dashboard Interface (Example)

Figure 19 depicts a Dashboard split into two halves. On the left hand side, there is an enumeration of key components (asset classes) against which events can be defined that are then injected into a running cascade as launched from the Cascade Engine interface. This can be done in real time, while the cascade is running, and results in the stimulation of one or more nodes within the running cascade.

In the dashboard provided in Figure 19, for example, the injection of an impact on a networking component will have an immediate impact on the efficiency and effectiveness of the entire network chain. In order to measure that impact, the right hand side of the Dashboard is used to display state messages, graphs, sparklines, or other visualisations of how the cascade is performing in the face of external impacts.

The right hand side of the dashboard is populated by the set of sensors contained within the running cascade to which the Dashboard has been attached (ref: Attachment options, top left of Figure 19). Sensors, in Cascade Engine terms, are essentially observers and reporters embedded inside a Cascade Engine node. As the state of a node, or any of its attributes, changes, so that change can be reported by a sensor assigned to watch that state. The sensor outputs from each node flow up to the master sensor list held by each cascade where, as Figure 19 indicates, they may be subscribed to by clients such as the SimICI dashboard interface.

The SimICI Dashboard interface, therefore, provides the opportunity to look inside a running cascade in order to understand how risks and threats are impacting the workflow. Additionally, if required by the question under analysis and if permitted by the cascade designer, the SimICI Dashboard may then also be employed as a means of influencing a scenario under evaluation within SimICI.

5.5 Prototyping Interface

The SimICI Prototyping Interface, while not yet fully scoped, is anticipated to be a semi-technical interface that allows a user to visually design bespoke behaviour that can (a) be executed by a special-to-type node within a cascade run from within SimICI and (b) be used to 'write out' a function specification that represents a new analysis to be implemented within SimICI.

In order to support this interface, additional work is required. This is noted in Section 6, Future Work.

5.6 Interface Extensibility

SimICI is a standard web application written to HTML/CSS3 standards and employing JavaScript for client-side functionality. The SimICI framework is designed as a container for any number of different interfaces and, therefore, provides a set of master services to all interfaces contained within it.

Those master services include user authentication and role-based access control, system-wide settings and configuration, user profile management, and – of course – the SimICI API through which data, events, and other actions are exchanged.

The container model, as used in SimICI, therefore lends itself very well to the addition of new interfaces that simply 'plug in' to the framework and appear as an option to the user. Discrete functional interfaces, as yet unknown as requirements, may therefore be developed further down the line and integrated into today's SimICI environment.

This potential supports the ambition to exploit SimICI as the mechanism for outreach to the open-source and perhaps, non-traditional resilience community. The potential for extension is also key to increasing the attractiveness of SimICI as a 'leave behind' reference implementation of EU-CIRCLE outputs.

6 Future Work

Future Work for SimICI is summarized in the following tables.

Short-Term Activities:

Planned Task	**Description**
Expose NetCDF format from within GSI Services	Determine and define the Spatial Reference System to be commonly used across NetCDF producers and consumers. Add definition to Geoserver.
Add base populations to Cascade Engine and Dashboard to support Mid-Term Review demo.	Select and implement (even if not 100% defined) Cascade Engine nodes that allow the full scope of SimICI to be demonstrated at the Mid-Term Review demo.
Finalise Version 1 of the Administration and User Manuals for SimICI.	Finalise and publish Administration and User manuals at Version 1.0

Medium-Term Activities:

Planned Task	**Description**
Add CI assets, networks, etc. to SimICI	Based on the outputs from WP2,3, and 4, continue to update the matrix of required representations; implement those representations, and deploy them for use.
Cascade Engine node for prototype analysis	Finalise definition of, implement, and test the special-to-task Cascade node and the associated interface that will allow for visual design of custom behaviours. Define CIRP AnalysisTask template that can be written out by the prototyping node / interface.
3D icons for 3D interface and drag-and-drop 'city builder' functionality.	Locate, acquire, or build 3D icons for the various CI asset classes. Integrate rendering of those icons into the 3D interface scene graph when features are drawn. Provide 'building box' tool that allows a 3D icon to be dropped into the 3D view and then quickly configured to represent a custom CI asset.
Direct Integration between SimICI and CIRP. (Stretch Goal)	Explore the work required to exploit the CIRP hooks previously identified as means of allowing CIRP analyses to be set up and executed from outside of CIRP. Assess the level of complexity and make a build or abandon decision by M30.

In addition to the Future Work identified above, the remainder of the WP7 Tasking will continue in parallel.

7 Conclusions

The document accompanies the first release of the SimICI environment and describes the background and architecture evolution of SimICI to this D7.1 release. In addition, the document has described the underlying components comprising SimICI and the integration between them.

The document also described the GIS Services component and the Rethymno 'virtual city' dataset that has been ingested and exposed through that component. SimICI interfaces, leveraging the underlying GIS resources, have been illustrated and Future Work identified.

SimICI has the potential to allow technical and non-technical users to collaborate through a web-based application that supports multi-tenanted interaction under permissions based controls. This collaboration may be employed in support of scenario validation, intervention strategy definition, or in the development of prototypes for new, more rigorous analyses in CIRP.

** ENDS **