EU-CIRCLE

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

D7.2 Scenario Supports in the SimICI System

Contractual Delivery Date: 31/03/2017	**Actual Delivery Date**: 30/08/2017
Type: Report	**Version**: v1.0 FINAL

Dissemination Level: Public Deliverable

Statement
This document describes the supports for the EU-CIRCLE scenarios as enabled in the SimICI system. The requirements for SimICI have been modified by the introduction of the Asset Class Repository (ACR) and the ANDI methodology. The effect of this modification has been to obviate the majority of the previous content for this report and, accordingly, this report comprises new content that reflects the introduction of the ACR and the ANDI methodology. Development work is continuing to implement the extended requirements for SimICI and will not complete until the end of September 2017. To that end, this document is being released as a deliverable of the EU-CIRCLE project but contains forward looking statements reflecting the ongoing software development work for SimICI. This document should be read in conjunction with EU-CIRCLE deliverables D3.1[1], D5.4[2], and D7.4[3].

Preparation Slip			
	Name	Partner	Date
From	David Prior	XUV	25/08/2017
Reviewer	Leonidas Perlepes, Antonis Kostaridis	STWS	29/08/2017
Reviewer			
For delivery	David Prior	XUV	30/08/2017

Document Log			
Issue	Date	Comment	Author / Organization
V0.1	01/11/2016	Document Structure and Indexes	D. Prior / XUV
V0.2	31/01/2017	Initial Content per SimICI V1.0	R. Vaughan /XUV, D. Prior / XUV
V0.3	09/03/2017	Document Suspended at MTR	D. Prior / XUV
V0.4	28/04/2017	Document Suspended for Development of ANDI Methodology and ACR	D. Prior / XUV
V0.5	30/06/2017	Internal Review for ANDI Changes	R. Vaughan /XUV, D. Prior / XUV
V0.6	01/08/2017	Document Reopened and Recreated	D. Prior / XUV
V0.7	24/08/2017	New Base Content Added	R. Vaughan / XUV, D. Prior / XUV
V0.8	25/08/2017	Additional Content on ANDI and ACR	D. Prior / XUV
V0.9	25/08/2017	Released for Internal Review	R. Vaughan / XUV, D. Prior / XUV
V1.0	30/08/2017	Revised for Reviewers Comments	D. Prior / XUV

List of Abbreviations	
Term	**Description**
ACR	Asset Class Repository
ANDI	Assets, Networks, Dependencies, and Interconnections
API	Application Programming Interface
CI	Critical Infrastructure
CIRP	Critical Infrastructure Resilience Platform
D	Deliverable
GIS	Geographic Information System
GML	Geography Markup Language
GUID	Globally Unique IDentifier
HTML	HyperText Markup Language
HTTP(S)	HyperText Transfer Protocol (Secure)
JSON	JavaScript Object Notation
KML	Keyhole Markup Language
MTR	Mid-Term Review
NoN	Network of Networks
OGC	Open Geospatial Consortium
REST	REpresentational State Transfer
SDK	Software Development Kit
SimICI	Simulated Network of Interconnected Critical Infrastructures
URL	Uniform Resource Locator (An address string that references a thing on the Internet)
UI	User Interface
VDS	Virtual Data Set
WebDAV	Web Distributed Authoring and Versioning
WFS	Web Feature Service
XML	eXtensible Markup Language
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 EU-CIRCLE scenario supports available within the integrated SimICI system. SimICI 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 web-based application that provides a series of intuitive interfaces through which the EU-CIRCLE 'virtual city' dataset (VDS) 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 (See: EU-CIRCLE Deliverable 7.1[4]). SimICI is, and will continue to be, populated with the data comprising the EU-CIRCLE VDS and other assets, hazard 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 removes 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.

At the end of the current development activity, SimICI will include implementations of the Asset Class Repository (ACR) and the Assets, Networks, Dependencies, and Interconnections (ANDI) methodology. The ACR provides an active mechanism through which types of Critical Infrastructure (CI) assets may be defined and, subsequently, exploited. The ACR and its contents form, alongside the EU-CIRCLE VDS, part of the Open Data Output from the EU-CIRCLE project.

The ANDI methodology delivers a novel approach to operator-led definition of CI networks, and Networks of Networks (NoN), that insulates CI operators from the technical complexities of both Geographical Information Systems (GIS) and software development within SimICI. The ANDI methodology leverages the ACR to provide visual definition of a CI network or NoN that, once saved, automatically generates the required underlying GIS formatted data plus an active software representation of the network or NoN.

Outputs from the ANDI methodology are then exploitable by EU-CIRCLE CIRP for formal asset-wise analyses and by other applications, within or external to the EU-CIRCLE project, as may be required. The use of open-standards, protocols, and structures in SimICI reduces the integration burden of such other applications and helps broaden the range of potential applications for the EU-CIRCLE VDS, the ACR, and SimICI in general.

This document discusses the specific supports provided by SimICI to the EU-CIRCLE scenario use cases. The following sections discuss supports derived from the ANDI methodology; supports for active representation of CI networks; supports for the application of hazard impacts; support for the generation of metrics and indicators, and general supports for data provision.

Contents

Figures

1   Introduction

This document comprises deliverable D7.2 of the EU-CIRCLE project; providing details of supports for the EU-CIRCLE scenarios as provided by the SimICI software system. For a description of the SimICI system at Version 1.0, please refer to previous EU-CIRCLE deliverables D7.1[4] and D7.8[5][1].

The SimICI software system provides applications and methods that create **Sim**ulated (**I**nterconnected) **C**ritical **I**nfrastructure networks and Networks of Networks (NoN). As such, SimICI must provide supports for the representation of Critical Infrastructure (CI) assets, networks, interconnections, and dependencies. Moreover, SimICI must provide support for the application of impacts arising from climate hazard scenarios, as defined within EU-CIRCLE, such that the effect of those impacts can be simulated and assessed for the purposes of CI resilience.
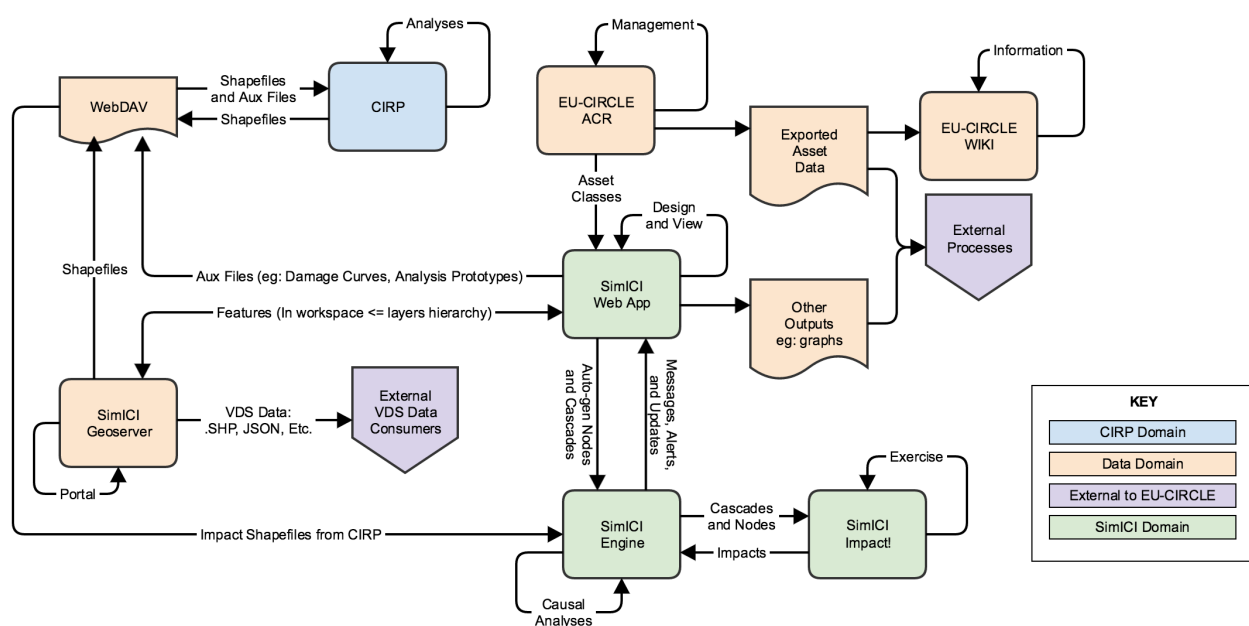
Figure 1: EU-CIRCLE Systems View (SimICI Perspective)

Figure 1, above, provides – from the perspective of SimICI - a Systems View of the EU-CIRCLE software and services infrastructure.

The systems outcomes of the EU-CIRCLE project span multiple domains:

- **The Critical Infrastructure Resilience Platform (CIRP) Domain:**
    - CIRP is the primary analytical platform for CI resilience in EU-CIRCLE
    - CIRP is responsible for asset-wise analyses of CI impacts due to climate hazards, producing modified Shapefiles as an output describing per asset impacts.
    - CIRP ingests CI assets described in Shapefile format, conducts analyses, and produces outputs that are also described in Shapefile format
    - Full details of CIRP may be found in the EU-CIRCLE WP5 deliverables

---

[1] It should, however, be noted that the change in requirements to incorporate ACR and the ANDI methodology in SimICI means that the SimICI system is under redevelopment at the time of writing and, therefore, that future deliverable D7.9[6] will provide the definitive description of the SimICI system as provided by the EU-CIRCLE project.

- **The Data Domain:**
  - o Encompasses:
    - The EU-CIRCLE Virtual City Dataset comprising representative CI assets geolocated in the Rethymno area of Crete; managed by the SimICI Geoserver
    - The EU-CIRCLE Wiki containing information relevant to the objectives of EU-CIRCLE; including schedules of CI assets and associated algorithms
    - The EU-CIRCLE ACR comprising a master set of CI Asset Classes detailing relevant attributes, inputs, outputs, and behaviours for each Asset Class
    - Other CI asset and network related data as may be required for the full exploitation of EU-CIRCLE project activities
  - o Supports the Open Data Pilot deliverables planned for EU-CIRCLE

- **The SimICI Domain:**
  - o Comprises the set of applications supporting the SimICI objectives within EU-CIRCLE
  - o Integrates with CIRP and Data Domains
  - o Delivers support for the presentation and execution of EU-CIRCLE scenarios
  - o Provides support to the external domain

- **The External Domain:**
  - o Reflects the ambition for EU-CIRCLE outputs to be widely exploited
  - o Supports the provision of EU-CIRCLE VDS data to external consumers as part of the Open Data Pilot deliverable
  - o Supports the provision of EU-CIRCLE ACR data to external consumers as part fo the Open Data Pilot deliverable
  - o Encompasses provisions to support the Outreach strategy to be developed under EU-CIRCLE Task 7.5

The Systems View provided in Figure 1 is colour-coded to reflect the domains described above. The Systems View also details the interconnections between the various aspects of the EU-CIRCLE systems, highlighting both key interaction points and the integration between the various components.

Figure 1 should be referenced when reading the remainder of this document.

The remainder of this document details SimICI supports to the EU-CIRCLE scenarios with regard to the ANDI methodology; the active simulation of CI networks; the application and causal assessment of effects due to climate hazard and other impacts; the generation of metrics and indicators, and general data services.

2   ANDI Supports

The Assets, Networks, Dependencies, and Interconnections methodology has been conceived and developed within EU-CIRCLE in response to a number of requirements related to reducing operator burden; abstracting and automating numerous processes that might otherwise delay CI resilience studies; addressing the representation of complex interdependencies, and enabling analysis of cascading effects – within and across CI networks – to a minimum of second and third order causality.

As noted previously, SimICI is currently undergoing extensive redevelopment in order to provide an implementation of the ANDI methodology. The remainder of this section contains forward-looking statements that will be validated with the release of future deliverable D7.9[6].

2.1   Thinking in Graphs

As has been observed in EU-CIRCLE to date, Critical Infrastructure (CI) is a complex network of networks characterised by interdependencies. An interdependency is a mutual relationship between two networks: meaning one network is influenced by another. In such a network of networks there is high potential for cascading failures (within a single network) and cascading effects (across network boundaries).

Examination of individual networks within the CI indicates that each such network may be considered as a graph: a structure used to model pairwise relationships between objects. A graph is comprised of nodes and edges that, mapped to a CI network, represent the components and connections within that network. Figure 2, below, illustrates a basic graph that represents nodes in two possible networks and the set of connections between them.
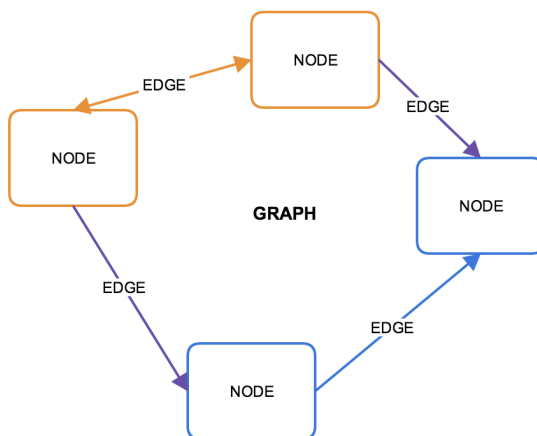

Figure 2: A Basic Interconnected Graph

Graphs may be undirected, meaning that all connections (edges) are bidirectional (ie: A -> B == B -> A), or directed, meaning that connections (edges) between the components (nodes) carry a declared direction (ie: A -> B != B -> A). The directed case does not preclude the existence of two edges between two nodes: one pointing in one direction, the other the inverse.

Graphs may be acyclic, meaning a graph within which there are no routes back over previously followed edges. While 'cyclic' is often used to mean any graph which is not acyclic, a case exists which is referred to as a cycle graph. A cycle graph is a graph that consists of a single cycle: some number of nodes

connected in a closed chain within which each node has exactly two edges. The directed and undirected cases continue to apply.

The provisions of graph thinking adapt well to both individual networks and the network of networks collectively comprising the CI. Graphs may be used to not only represent individual component networks but also the interactions (the dependencies and interconnections) between those components. Essentially, for the purposes of exploring the resilience and robustness of the CI, we can treat the CI as a complex graph of graphs.

Figure 3, below, illustrates the mapping between graph thinking and interconnected CI networks.
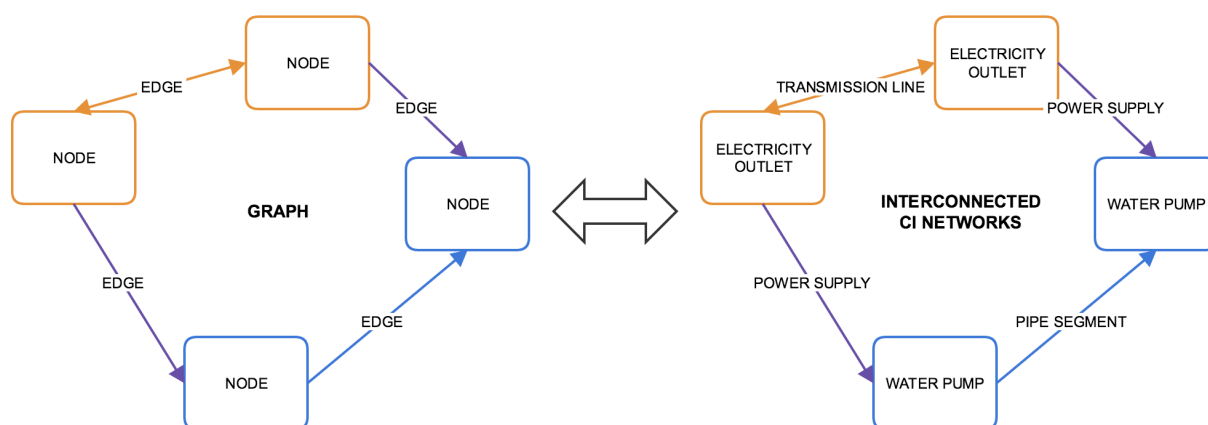

Figure 3: Mapping Graph Thinking to Interconnected CI Networks

The left hand side of Figure 3 illustrates two discrete networks and the interconnections between them. The orange network is unidirectional and cyclic while the blue network is directed and acyclic. The network of networks, coloured purple, is also directed and acyclic.

The right hand side of Figure 3 shows the same graph of graphs but with different references. The previously abstract graph of graphs is transformed into a representation of two component networks and the causal mesh created by their interaction[2].

For simplicity in EU-CIRCLE, we will simply refer to the representation, manipulation, and interrogation of nodes and edges from hereon in.

It is worth noting that the representation of CI as a complex graph of graphs permits useful mathematical analyses in relation to the resilience and robustness of the CI under examination. There are several metrics that may be used to characterise the structure of graphs but, for simple identification of critical nodes within a graph (and, therefore, critical assets within a CI network), centrality metrics offer immediate potential.

Key centrality metrics, which may be calculated from a graph representation of a network – even without the application of more complex preparation and analysis, are:

- Degree Centrality: The number of incident edges connected to each node;

---

[2] It should be noted that, obviously, the two component networks are incomplete and, depending on the real world network being represented, may in themselves be directed or undirected cycle graphs (eg: a water ring main). It should also be noted that, to adequately represent a real world network, the links between nodes as illustrated (eg: transmission line) should themselves be nodes, containing attributes and behaviours, that are connected via flows. For the purposes of the current discussion, however, the view in Figure 3 is sufficient.

- Closeness Centrality: A measure of distance (ie: shortest path) among nodes;

- Betweenness Centrality: The number of shortest paths through a node or edge;

- Eigenvector Centrality: The importance of a node based on its connections to nodes with high Degree Centrality.

Centrality metrics may, therefore, be used as a rapid approach to understanding key dependencies and associated susceptibility to binary (on/off) failure cases. This aids the understanding and appreciation of vulnerabilities at the network level even before calculating the outcome(s) of specific impact scenarios.

2.2 The Asset Class Repository

Work in EU-CIRCLE WP3 is developing a registry of CI assets and dependencies. Such a registry should represent asset classes: the prototype of an asset which may be instantiated one or more times in one or more networks. Asset classes may be thought of as templates that define the inputs, outputs, and behaviours of an asset and that describe the attributes of that asset. It is only once an instance of the class is created that the attributes are completed to reflect that specific instance.

For example, the asset class for WATER PUMP may contain the following:

- INPUTS:
    - Water Flow in Litres/Minute (typically from a PIPE SEGMENT asset)
    - Power in Watts/Hour (typically from an ELECTRICITY network asset)
    - Degradation as a percentage (typically as a damage function output)

- OUTPUTS:
    - Water Flow in Litres/Minute (typically to a PIPE SEGMENT asset)
    - Effectiveness as a percentage (used to drive a resilience metric)

- BEHAVIOURS:
    - Output Flow Calculation (stimulated when inputs change)
    - Effectiveness Calculation (stimulated when inputs change)

- ATTRIBUTES:
    - GUID: Empty Text (Globally Unique Identifier)
    - Name: Empty Text
    - Geometry Type: Point
    - Geometry: Empty Array
    - Max Flow: Empty Number (for Litres/Minute)

Once an instance of the class is created, the attributes will be populated similar to:

- ATTRIBUTES:
    - GUID: g2398-hgfsk-92385-jsh89
    - Name: WATER_PUMP_1
    - Geometry Type: Point

o   Geometry: [18.07221, 42.65539] (Longitude, Latitude)

o   Max Flow: 1200

Populating the attributes at instance time creates a node that reflects a real-world physical asset.

From the above, it is clear that, without an interactive digital asset registry that can be actively exploited, the manual creation of network representations could be extremely time-consuming!

To that end, it has been suggested and agreed that the EU-CIRCLE Repository be enacted in software as an adjunct to SimICI. The Repository will contain the asset class definitions for those assets identified in the WP3 activity and will be exploitable within SimICI for the definition of assets, networks, dependencies, and interconnections (ANDI). This repository will be known as the Asset Class Repository (ACR).

In order to generate the EU-CIRCLE ACR and to support the ANDI methodology, it will be necessary to define the asset classes for the various types of CI assets used in the EU-CIRCLE scenarios. It should be noted that an asset class should not reflect a concrete real world instance of the asset but, instead, should describe the common aspects of every real world instance of the asset.

An asset class, in ACR, is more than just an asset type. An asset class must contain – at a minimum – the INPUTS, OUTPUTS, and ATTRIBUTES for each asset.

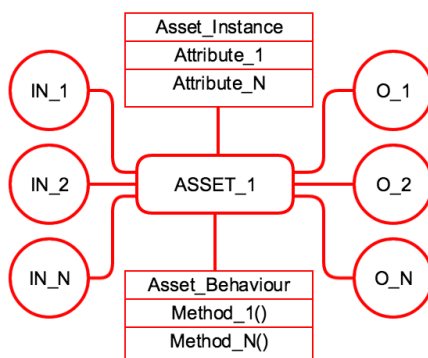Figure 4, below, illustrates what is required to define an asset class.

Figure 4: An Asset Class Template (Example)

Asset BEHAVIOURS are optional if, and only if, there is no intention to exercise the interconnections and dependencies in the CI network (of networks). BEHAVIOURS are calculations that, subject to a change in INPUTS to an asset, modify one or more OUTPUTS from that asset. It is this stimulated, dynamic modification that exposes the impact of both cascading failures and cascading effects.

As a contrived example, think of a plant as an asset class. The definition for that class might look something like:

•   ATTRIBUTES:

o   Name:

o   Genus:

o   Geometry Type: Point

o   Geometry: []

- o   Required Water: litres/day

- o   Required Sunshine: hours/day

- o   Required Fertiliser: g/week

- • INPUTS:

  - o   Water: litres/day

  - o   Sunlight: hours/day

  - o   Fertiliser: g/week

- • OUTPUTS:

  - o   Growth Rate: cm/week

  - o   Crop Yield: %age

  - o   Health: %age

- • BEHAVIOURS:

  - o   Growth_Rate():

    ((Water / Required Water) * (Sunshine / Required Sunshine))

  - o   Crop_Yield():

    (Growth Rate * (Fertiliser / Required Fertiliser))

When defining an asset class, the key is to start with the ATTRIBUTES, then the INPUTS and OUTPUTS, before considering the BEHAVIOURS. For some assets, the BEHAVIOURS will be obvious and there may be industry standard calculations that can be employed. For other assets, BEHAVIOURS will be less obvious and may require assumptions.

It is not necessary to have a fully accurate or complete asset class definition in the first instance but, of course, the more detailed you can make it the more useful the class will be. It should also be considered that, after a full consideration of the INPUTS to an asset class, there may be auxiliary OUTPUTS that can be calculated.

Such auxiliary OUTPUTS may, for example, provide support for the calculation of metrics and indicators derived from the stimulated behavior of an asset such that higher order network metrics may be calculated from the ANDI methodology.

2.3   Instancing an Asset Class

With ACR implemented and CI asset classes registered, it is necessary to consider how the ACR may be exploited. Exploitation of the ACR will occur as part of the Open Data Pilot but, for the immediate purposes of the EU-CIRCLE project, as a key enabler for the ANDI methodology as implemented in SimICI.

For this purpose, asset classes held within ACR – reflecting the generic aspects of particular asset types – must be used instanced to reflect real world physical CI assets. Figure 5, below, illustrates the SimICI mechanism that allows a generic asset class to be instanced and its attributes populated such that the instance represents the real world asset.
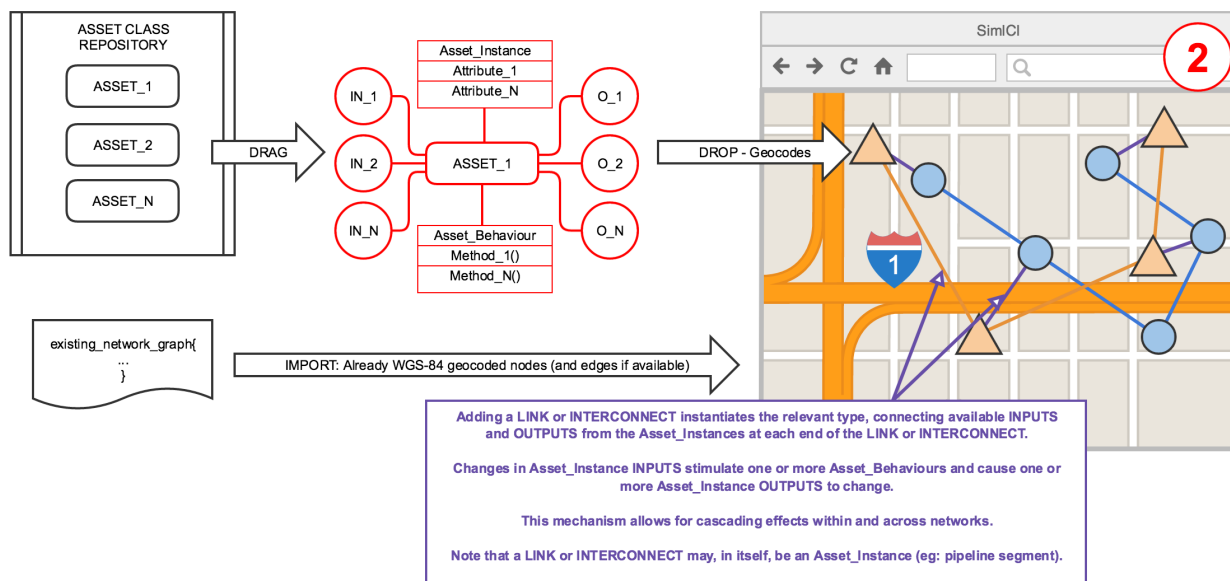
Figure 5: Instancing an Asset Class

At the left hand side of Figure 5, an indication of the Asset Class Repository is shown. This may appear as a scrollable list of classes that may be dragged onto a SimICI map. The act of dragging a class to the map creates an instance of the class with an assigned GUID. The act of dropping it onto the map immediately geocodes the instance – completing the geometry attributes of the asset class definition - and opens a dialog box requesting the input of additional attributes. Moving the dropped instance on the map updates it's latitude and longitude accordingly.

The example described above is, patently, for an asset with Point geometry (eg: a WATER PUMP is bounded to a specific point). A similar process is applicable for assets with (Poly)line geometry (eg: a PIPE SEGMENT has start, end, and perhaps intermediate points that form a line) and Polygon geometry. The handling of geometry, allowing assets to mapped to their real world locations, simply requires that the SimICI map understands the relevant geometry as declared for each asset class and provides the relevant drawing tools for that geometry.

With a set of now geolocated assets on the map, interconnections and dependencies can be added. This can be enabled by drawing a connector between any two assets, opening a dialog box to define the specific output and input that the connection refers to. It is the setting of these connections, not the addition of network conduit assets to the map, that creates the graph (of graphs).

Figure 5 also contains a reference to the direct importation of existing network graphs. It is assumed that some models for CI networks will already exist and that the owners of those models will not wish to recreate that for the purposes of EU-CIRCLE. To that end, it is suggested that the ability to import a standard network representation be provided in SimICI. This will require only that any existing models be generated in a standard format (eg: a JSON structure to be defined by EU-CIRCLE) that refers to the asset classes defined by EU-CIRCLE.

For example, in the case of an existing water network model, the output from that model should be generated with reference to the EU-CIRCLE water asset classes and with as many of the attributes for each class instance as possible populated. At a minimum, the expectation is that any existing models to be imported should provide at least the name and geometry for each asset within that network.

If existing models are not natively capable of, or extensible to, providing references to EU-CIRCLE asset classes and (at least some of the required) attributes, then the outputs from those models will require intermediate manipulation prior to import into SimICI.

Regardless of whether the network graph within SimICI is drawn natively or imported and updated, SimICI shall be responsible for the persistence of the graph such that it may be exploited and reused as required. The persistence mechanism will leverage the geospatial services already provided within SimICI and will permit the exploitation of defined graphs in a variety of formats (ie: automated generation of Shapefiles, graph files, and other formats as may be required for exploitation in EU-CIRCLE and elsewhere). This, therefore, allows an asset network to be visually designed in SimICI and subsequently exploited in CIRP: without the user having had to contend with the conversion of the network design into Shapefile (or other) format.

2.4   Creating the Causal Mesh

The previous section referred to the setting of interconnections and dependencies between assets; noting that it was these links the formed the edges of the graph. We refer to the collection of interconnections and dependencies, within and across individual CI networks, as the Causal Mesh.

A Causal Mesh is, quite simply, the topology representing causality within a graph (of graphs).

Topology, here, refers to the connections between assets and provides a mechanism to understand propagation, of either material or effect, between those assets. This permits the understanding and analysis of both cascading failures (within a network) and cascading effects (across networks in the network of networks).

The Causal Mesh is, as noted in the previous section and referenced in the call out in Figure 6, not the same as the map of physical world conduits (eg: electricity cables, pipe segments, fibre optic lines, etc. which are asset classes in themselves) but, instead, provides an overlay to the physical map that details the relationship(s) between assets. This is why, previously, reference has been made to both WATER PUMP and PIPE SEGMENT assets as nodes and, subsequently, the relationship between those assets as edges.

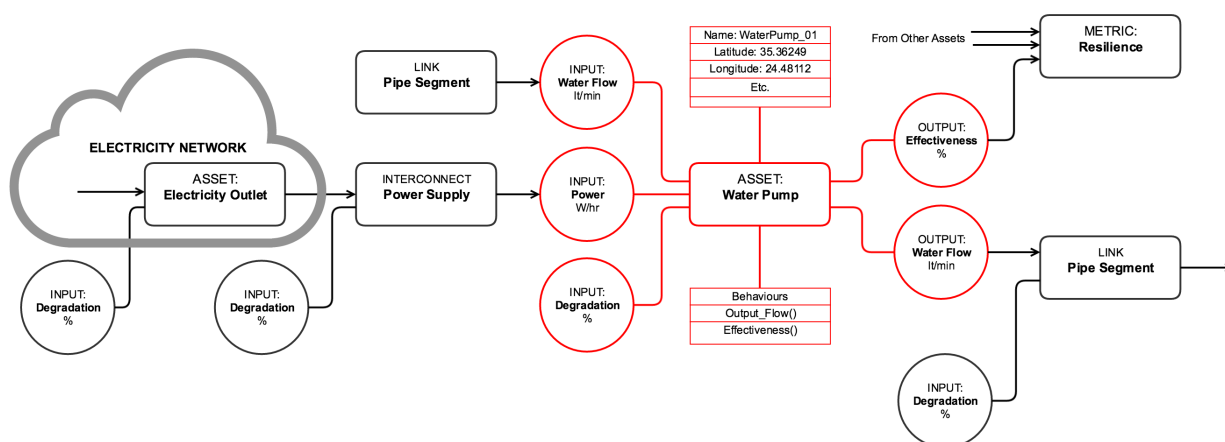It is hoped that Figure 6, below, makes this distinction clearer!

Figure 6: Modelling Dependencies and Interconnections

Figure 6, above, drills down into an exemplar WATER PUMP asset: an instance of the WATER PUMP asset class. This asset sits between two PIPE SEGMENT assets, with OUTPUTS and INPUTS connected to reflect water flow in the water network.

In this example, the upstream PIPE SEGMENT feeds water to the pump at a certain rate and, subject to other dependencies, the WATER PUMP feeds water out to the downstream PIPE SEGMENT at another certain rate. Cascading failures are visible here when, for some reason, the upstream flow rate declines: a change in the input to the WATER PUMP will stimulate its Output_Flow() behaviour to recalculate its output flow rate and, therefore, its Effectiveness() metric. For example, even if the pump is 100% operational and efficient, its maximum output flow rate will only be equal to its input flow rate.

For cascading effects, an incident of which may explain the upstream decline in flow rate, the Causal Mesh of the WATER PUMP is expanded to include the input of power to run the pump. This input comes from the electricity network graph allowing cascading effects to become visible as any impact to power provision to the WATER PUMP will, again, stimulate its Output_Flow() behaviour to recalculate flow and, subsequently, its Effectivess() metric.

This approach is repeatable at scale and across all interconnections and dependencies.

Referring back to Figure 6, attention is drawn to two other aspects contained within the Causal Mesh.

Those are:

- A Degradation Input to each asset:

    - Based on the idea of damage curve outputs from CIRP, these are decimal percentages that are applied to degrade the performance of any asset.

    - A fully operational asset has Degradation = 0.0

    - A fully degraded (ie: failed) asset has Degradation = 1.0

    - Degradations can be applied from a CIRP output or directly from the IMPACT! console within SimICI. This latter use case is for 'what-if' analyses.

- An Effectiveness Output from each asset:

    - Suggested in support of the calculation of Resilience metrics.

    - Currently a simple percentage effective calculation.

    - On a per asset basis, for any given range of input influences, this can be used to define a damage curve that can be exported for use within CIRP.

In addition to asset class specific attributes, it should be noted that additional inputs and outputs, along with associated behaviours, can be defined for any asset class. It is the pragmatic definition of the complete asset that will facilitate creation of an EU-CIRCLE Repository that is both comprehensive and useful.

2.5   The ANDI Methodology in SimICI

The preceding sections detail a number of techniques that, cumulatively, provide for a methodology for the representation, manipulation, and interrogation of CI ANDI in SimICI.

That methodology is described in Figure 7, below.

Figure 7 details six steps in the methodology, collectively enabled by the EU-CIRCLE technology backplane of CIRP, SimICI, and their respective supports.

The methodology is enumerated as:

1. The user opens SimICI and navigates to the Area of Interest (AOI). The SimICI map draws down map tiles and existing features relevant to the AOI to provide a canvas for user interaction.

2. The user visually defines, or imports and manipulates, a CI asset network; then defines the associated causal mesh. If one or more other CI asset networks are already defined and visible in the area of interest, the causal mesh at the network of networks level may also be defined.

3. When the user saves their work, a SimICI server process will take the network (of networks) graph and persist it both as geospatial features (in the case of assets) and other forms as may be required to support additional exploitation (eg: causal mesh analysis in the SimICI Engine, graph processing in an external tool, etc.).

4. Persisted features, most typically of Point and (Poly)Line geometry, will be available for use in CIRP through export to WebDAV from the SimICI Geoserver. This allows for the transparent generation of Shapefile formatted outputs representing any asset network, defined within SimICI, that requires analysis in CIRP. If damage curves have been generated for the assets in that network (in Step 5), those are also available for use in CIRP. Further definition of the data provision service that allows the Shapefiles and any other data to be accessible from CIRP will be required but, at this time, the most likely provision will be via a central WebDAV server in the EU-CIRCLE technology backplane.

5. Available graph (of graphs) data, as persisted in Step 3, will be accessible to the SimICI Engine application. This will allow the user-defined graph (of graphs), from Step 2, to be dynamically regenerated as a SimICI Engine cascade for the purposes of interrogation, analysis, and – where required – the generation of damage curves[3], etc. Once created as a cascade, the graph (of graphs) may then be stimulated by damage inputs from CIRP or by direct input from the SimICI Impact! console. As the model is stimulated, so cascading failures and effects – along with associated metrics and indicators - are calculated in real time by the SimICI Engine.

6. Dynamic changes to the graph, from interaction in Step 5, are despatched in real time to the main SimICI map view. This provides support for end-user / operator focused workshops that allow for the interactive exploration of resilience and robustness within and across CI component networks. A typical use case here may incorporate collective decision making from the workshop process being used to directly stimulate the graph (of graphs) from the SimICI Impact! console.

The above represents the standard use case for the CI ANDI methodology as implemented in SimICI in support of the representation and exercise of assets, networks, dependencies, and interconnections in the EU-CIRCLE scenarios. Careful design of asset classes and resulting graph (of graphs) definitions should support additional analyses that help to inform awareness of both network robustness and mitigation strategies. An example of such an additional analysis is the exploitation of graph centrality metrics as discussed previously.

---

[3] Because Asset Classes in the ACR contain behaviour definitions, it is possible to exercise those behaviours against a range of inputs, applied through the Impact! application or directly within the cascade itself. This, in turn, generates a series of outputs due to the behaviours which can be persisted and exploited as damage curves. Once work on the SimICI extensions completes, all such outputs will be available for examination and, where required, subsequent extension and/or modification.
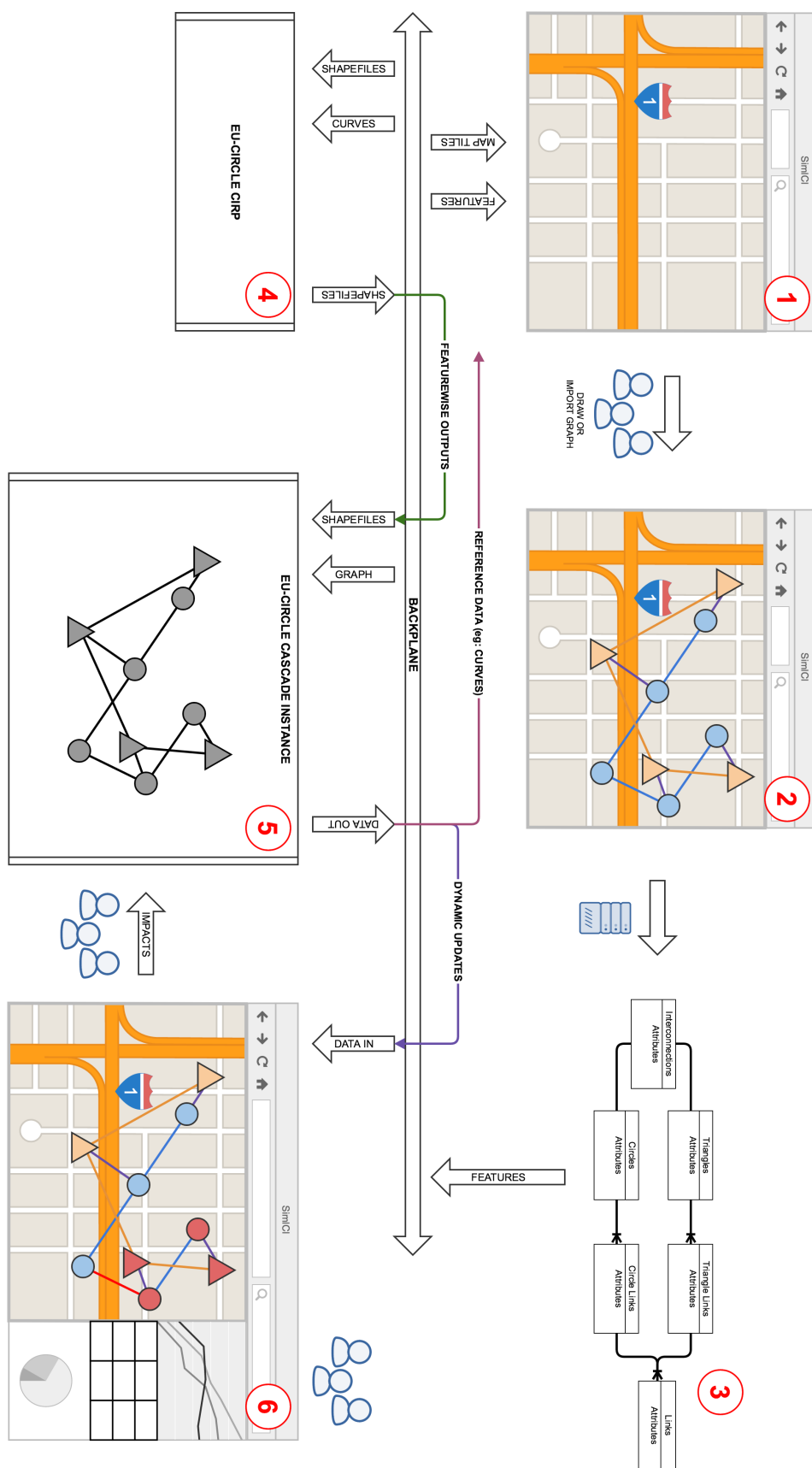
Figure 7: The ANDI Methodology in EU-CIRCLE

3   Active Supports

In the previous section, step 5 of the ANDI methodology described how the graph (of graphs) created from the ACR within SimICI is made accessible to the SimICI Engine such that a simulation model can be automatically generated and executed. That process turns a visually described network into an active piece of software.

In other words, a map view similar to that shown in Figure 8, below, can be turned from a static view into an active simulation that is capable of being stimulated, interrogated, and analysed for CI resilience purposes.
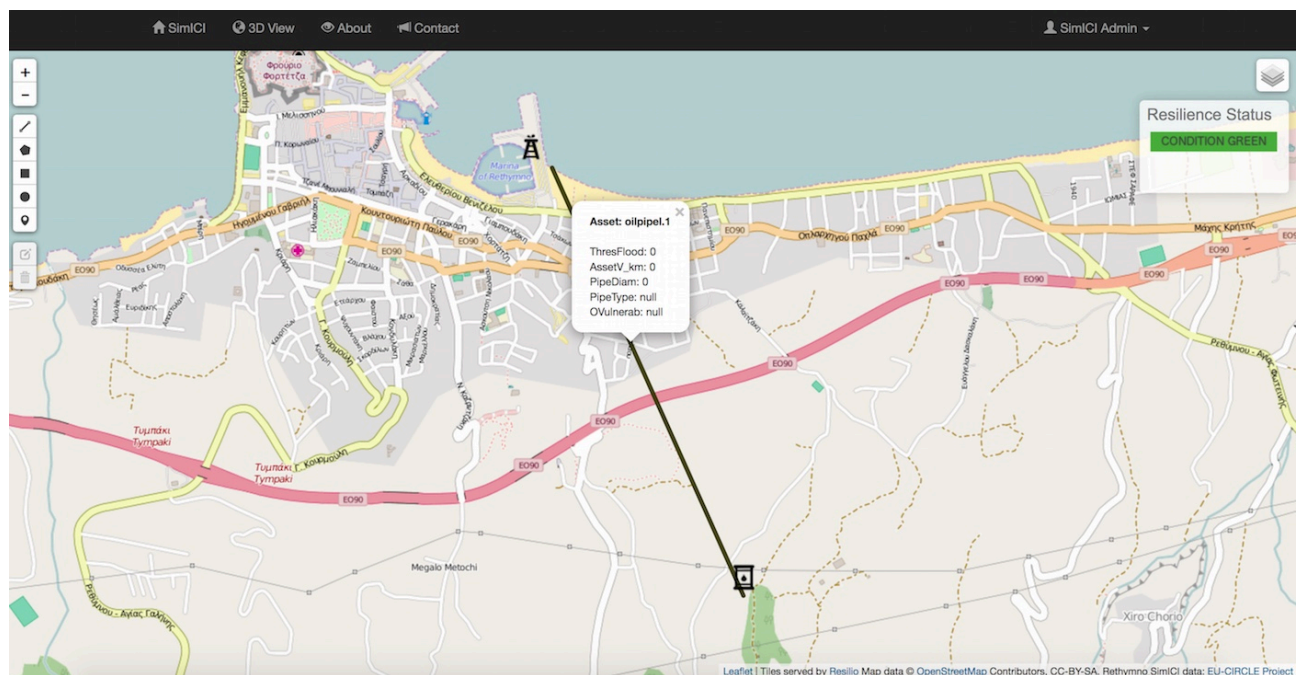
Figure 8: Rethymno Oil Network in the EU-CIRCLE VDS

Figure 8 shows the Oil Network contained within the EU-CIRCLE VDS, visualized in version 1.0 of the SimICI Mapterface. Essentially, this is three assets – an Oil Pumping Station, an Oil Pipeline, and an Oil Storage Facility – connected by a flow of oil.

In version 1.0 of SimICI, the definition of specific assets is the responsibility of an external party who generates a geospatial Shapefile that must be uploaded and published from the SimICI Geoserver by an Administrator. See D7.3[7] and D7.8[5] for more details.

With the implementation of the ACR and the ANDI methodology in SimICI, this mandrolic process is no longer required. Instead, the SimICI user simply drags and drops the relevant asset classes onto the map, completes any required specific attributes for the resulting instances, and declares the interconnections between the asset instances. Saving the new network design then automatically generates feature data entries in the SimICI Geoserver **and** passes the design to the SimICI Engine for conversion into an active simulation.

In version 1.0 of SimICI, the conversion process required that (a) SimICI Engine nodes (self-contained software representations of an asset) had been previously developed and published by an Administrator or Developer and (b) that the SimICI operator manually design the cascade (simulation)

that reflected the network (of networks) in the AOI. The manually designed cascade that makes the Rethymno Oil Network, as shown in Figure 8 previously, active is illustrated in Figure 9, below.
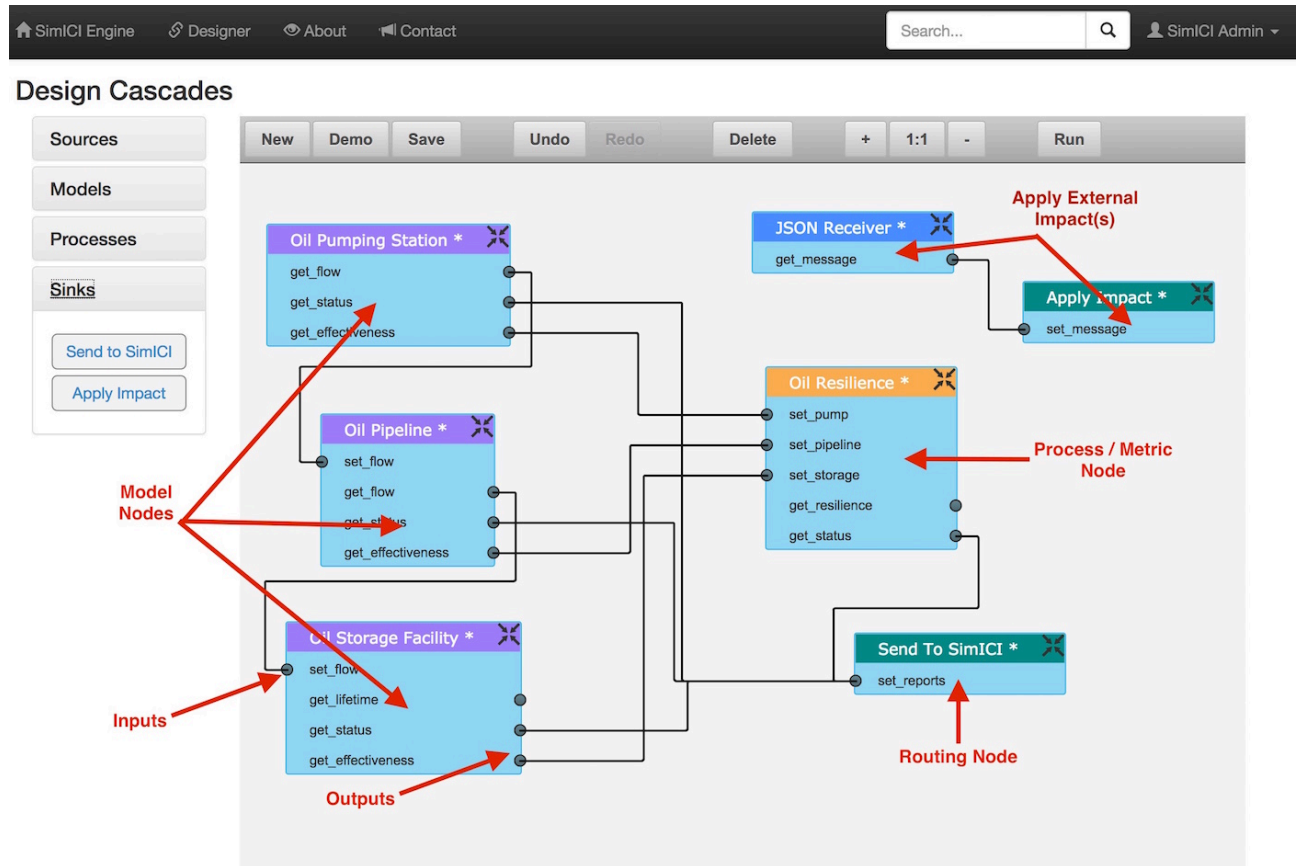

Figure 9: The Oil Network Cascade in the SimICI Engine

With the implementation of ACR in SimICI, the act of saving an asset class definition automatically generates a SimICI Engine node for that asset class. With the implementation of the ANDI methodology, where the network design based around instances of asset classes held within the ACR is sent to the SimICI Engine for auto-generation of the cascade, there is no longer a requirement for manual design of cascades.

Implementing ACR and the ANDI methodology, therefore, dramatically reduces the SimICI user burden. In addition, this implementation removes the potential for human error with regard to the definition of cascades that accurately reflect the network and reduces round-trip engineering time between the definition of a network on the SimICI map and the availability of an active simulation to support the exploitation of that network in EU-CIRCLE.
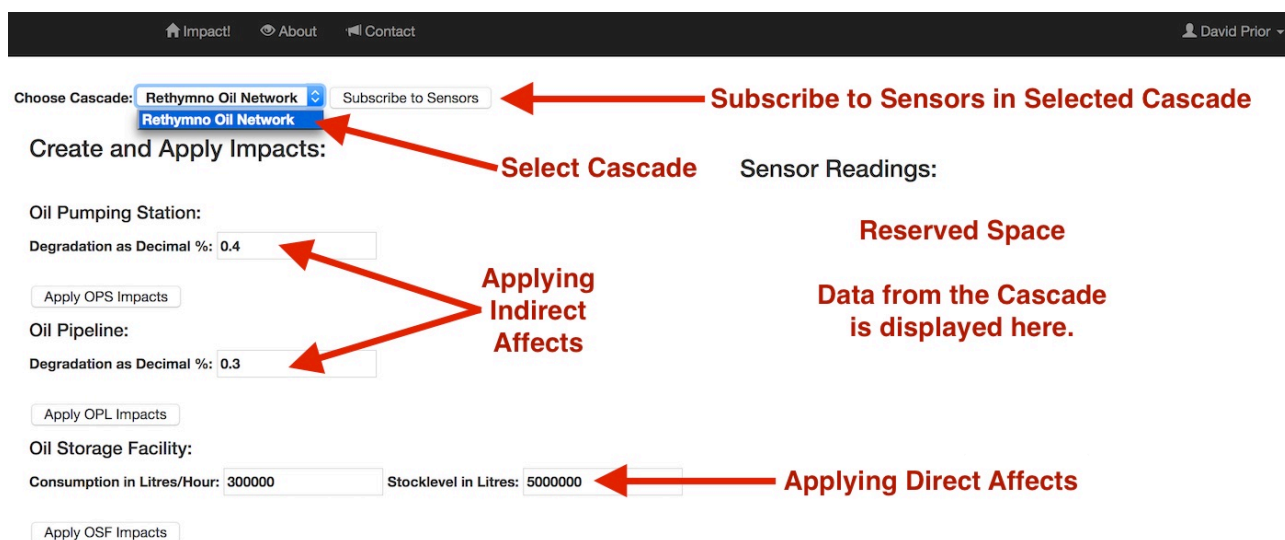
The resulting SimICI Engine cascade provides an active simulation of the network (of networks) that can be used to explore, interrogate, analyse, and otherwise exercise the network in support of CI resilience and other analysis as may be required.

As noted previously, SimICI Engine cascades may be stimulated by inputs derived from CIRP or directly through the SimICI Impact! application. Regardless of the source of stimulus, a cascade will dynamically react to received stimuli; propagating effects through networks and across dependencies in order to assess the effects of the stimulus.

4   Hazard Impact Supports

The previous section referred to SimICI Engine cascades being able to be stimulated by inputs derived from CIRP or directly through the SimICI Impact! application. For the purposes of visually describing how a stimulus – representing an impact due to climate or other hazard or affect – is applied to a cascade and what the effect of doing so is, the following discusses the use of the SimICI Impact! client.

The SimICI Impact! application is provided as a means of dynamically connecting to a running cascade and applying affects and influences[4] to it in order to stimulate the cascade into action. Logging in to the Impact! client endpoint[5] presents the Impact! main screen as shown in Figure 10, below.

Figure 10: The SimICI Impact! Application

The Impact! client identifies available cascades within the SimICI Engine and provides a selector from which the user chooses the cascade they want to influence. With the cascade selected, a press of the 'Subscribe to Sensors' button will read out the available nodes and associated impact points from the selected cascade as well as subscribe to all of the internal sensors available from all of the nodes within that cascade.

As shown in Figure 10, the available impact points for the Rethymno Oil Network cascade are listed on the left hand side of the screen while space is reserved on the right of the screen for the sensor data outputs to be received from the cascade.

It should be noted that different types of affect can be applied. For the Oil Pumping Station and the Oil Pipeline, indirect affects are available. These are, essentially, calculated or estimated inputs that have a more generic impact (eg: degradation is not a known physical input to an asset but can be estimated based on the current context in which that asset is operating).

For the Oil Storage Facility, on the other hand, direct impacts – correlated with actual or forecast data – can be applied. Here, for example, there is the potential to vary demand and reserves in order to determine the useful lifetime of the storage. Factoring those impacts with variable input flow

---

[4] The Impact! application is intended for use in consultant-led activities, such as workshops, within which operators express their collective subjective opinion as to what an impact may be, due to any cause. That subjective impact may then be applied to the CI network in order to validate or otherwise test operator assumptions.
[5] See D7.3/D7.8 for details of application endpoints in SimICI.

(replenishment rates) due to impacts elsewhere in the network then helps with resilience analysis under different conditions.

4.1 Applying Hazard Impacts (Rethymno Oil Network example)

The following describes the use of the SimICI Impact! application to interact with a running cascade such that impacts can be applied and the effects of those impacts observed. As previously, the network under consideration is the Rethymno Oil Network from the EU-CIRCLE VDS (illustrated in Figure 8, previously).

At this point in the process of supporting an EU-CIRCLE scenario, the following items are available within the SimICI domain:

1. A 2D map displaying the geography of the real world assets in the Oil Network

2. A cascade representing the Oil Network, and its causal mesh, as running software with the capability of sending updates to the 2D Map

3. The ability to directly affect Oil Network assets from an Impact! client connected to the cascade

In order to exercise the network, perhaps for reasons of a resilience workshop with a client looking to make a balance of investment decision, an impact must be applied.

Figure 11: Applying an Impact to a CI Asset

In Figure 11, an indirect impact is applied to the Oil Pumping Station that assumes degradation of the overall capabilities of that asset by 30%. As soon as the 'Apply OPS Impacts' button is pressed, the impact is sent to the Oil Pumping Station model in the cascade. That stimulus triggers a recalculation of the flow rate from the Oil Pumping Station as a result of the degradation which, in turn, propagates through the cascade to determine the cascading effect on Oil Pipeline capacity and, subsequently, the fill rate of the Oil Storage Facility.

The propagation of the impact through the cascade changes the majority of the run time sensors inside the cascade: causing each sensor to report data back to the Impact! client (the right hand side of Figure 11). The whole process – from application of impact through to sensor data being presented on the screen – takes almost less time than releasing the mouse button that triggered it.

Figure 12, below, shows the status page for the target cascade in the SimICI Engine and confirms that the impact was received and applied to the cascade. As expected, an impact message was received: declaring that a degradation of 30% should be applied to the asset 'oilpumbST.1' (the asset identifier for the Oil Pumping Station as drawn from the EU-CIRCLE VDS). On receipt of that message, the cascade is stimulated and reacts to the impact on one of its models. This triggers propagation of effects and consequently a dynamic response in the causal mesh described by the cascade.

Figure 12: Oil Network Cascade Processing Impact

Back in the 2D Mapterface, as shown in Figure 13, below, the degraded asset has been visually highlighted and a SimICI Alarm message, reflecting an impact on the resilience of the Oil Network, has been displayed in the message box.
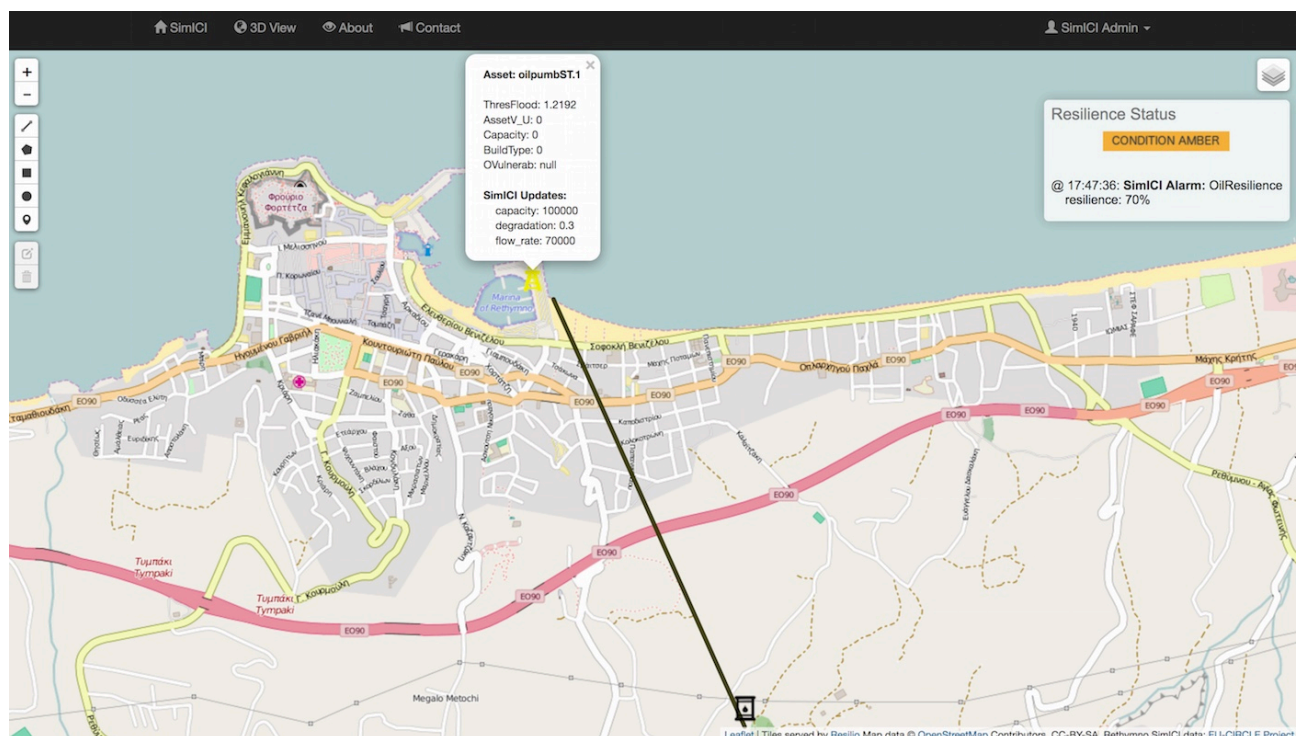

Figure 13: Visualisation of Effects due to Impact in SimICI Mapterface

Impacts from climate hazards, and other causes, may be applied to different assets within the network: creating a cumulative impact due to multiple failure cases at different points in one or across several networks. To demonstrate this, Figure 14 adds a second impact to the Oil Network cascade.

Figure 14: Adding a Second Impact

This time, as shown in Figure 14, a direct impact is applied to the Oil Storage Facility that increases consumption from it a hundredfold: from the default 30,000 litres per hour to a massive 3,000,000 litres/hour. This, for example, may occur where other Oil Storage Facilities have been removed from the network due to a pipe failure elsewhere and all demand has been switched to the single tank in our cascade.

As previously, the sensors inside the cascade have reported changes due to the new impact; showing not only the change in consumption rates but also the dramatic impact on the performance of the Oil Storage Facility. Not only does the facility now only have a single hour lifetime (against current stock levels): it will never be able refill! In consequence, it should be noted that the (basic multiplier) Oil Resilience metric has collapsed from 70% due to the previous impact to just 7% as a result of the cumulative effect of both impacts.

It should also be noted that, while this section is detailing a scenario solely concerned with the VDS Oil Network, exactly the same techniques can be applied to scenarios containing multiple CI networks in Network of Networks (NoN) form. In such scenarios, impacts can be applied at any point in any network and the propagation of effects both within and across the NoN can be observed due to the definition of interdependencies through the ANDI methodology.

As previously, the application of the second impact can be observed through reference to the cascade results screen in the SimICI Engine. Figure 15, below, shows the receipt of the second impact to be applied to the asset 'oilstorageF.1'.

Figure 15: Processing the Second Impact

And, as previously, the cumulative effects are also visible in the SimICI Mapterface for the VDS Oil Network per Figure 16, below.
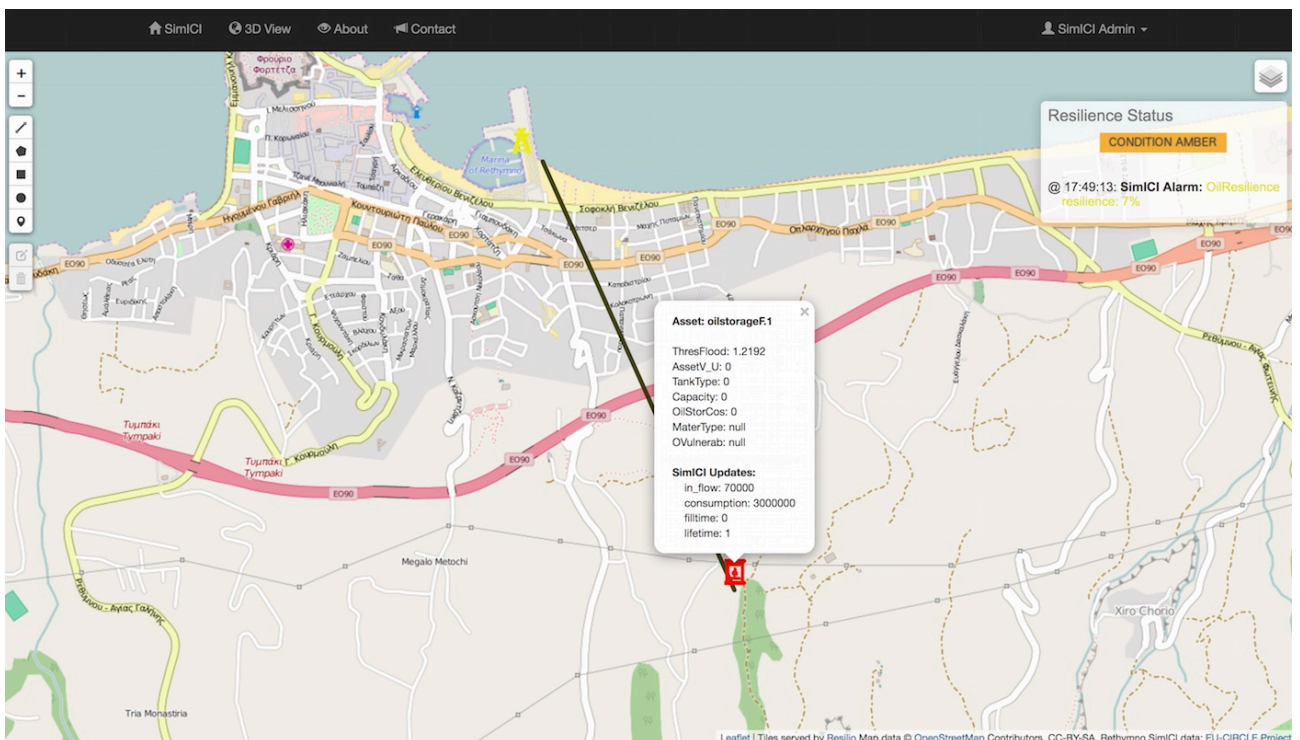
Figure 16: Visualising the Cumulative Effect of Impacts

While the example describing hazard impact supports in SimICI has focused on the use of the SimICI Impact! application to deliver stimuli, exactly the same outcomes ensue if stimuli are derived from a CIRP output file and applied programmatically to the cascade. The use of automated feature data production through the ANDI methodology, generating shapefiles that can be consumed by CIRP analyses, ensures that the output data returned **from** CIRP preserves the asset names and allows CIRP derived impacts to be applied to the correct assets within the SimICI environment.

5   Support for Metrics and Indicators

As indicated in Figure 9, SimICI includes support for metric and indicators that are not necessarily directly connected to physical aspects of the CI assets under study.

There are a number of mechanisms through which this support may be leveraged.

1. Within an asset class definition in the ACR, an asset behavior representing an indicator may be defined that calculates one or more metrics and indicators to drive a corresponding number of outputs. The actual values of the indicators so defined, changing as they are stimulated by direct impacts on the asset and cascading effects due to impacts elsewhere in the network, are reported from the cascade in the SimICI Engine.

2. One or more higher order metrics and indicators, for example: relating to an entire network or to a CI construct within an AOI, can be defined as a pseudo-asset in the ACR. Such an indicator would then have to be instanced in a SimICI design and connected to the relevant inputs arising from other instanced assets in that design. The results of the indicator would again be reported from the resulting cascade running in the SimICI Engine.

3. As the output of the ANDI process is, essentially, a graph representing assets in a network (of networks), that graph can be made available for alternative processing. An example of such alternative processing might include the calculation of graph centrality metrics (as described earlier in this document) to provide a naïve graph perspective on the relative importance of individual assets within a network (of networks). This is referred to as a naïve graph as the results will be based on the structure of the graph and will not factor for any specific influencing factors due to the purpose of the network that the graph represents.

Regardless of the mechanism used, the key point is that the ANDI methodology as implemented in SimICI supports the application of metrics and other indicators in relation to the CI assets and networks defined by both the EU-CIRCLE VDS and the wider EU-CIRCLE scenarios.

5.1   Extensibility in support of future metrics

A major consideration with regard to the ANDI methodology and the ACR relates to the ability to extend asset classes with additional attributes in support of either (a) additional data that has been previously unavailable or (b) the addition of new metrics and indicators to support revised or novel algorithms for CI network assessments.

The EU-CIRCLE ACR is, essentially, a database of master class definitions where a master class is the definition of what an asset looks like in terms of its attributes, inputs, outputs, and behaviours. The ACR is designed to allow a user to define a new master class or edit an existing one. A key aspect of that process is defining the attributes and inputs of the class. It is at this stage that any relevant attribute or input, as may be required to drive a physical behavior or an indicator, can be added. Once completed, the master class definition can then be saved and a SimICI Engine node generated to match it.

If a new input or attribute is required, perhaps to support a new algorithm as suggested above, then the approach to take in ACR is to open the existing master class definition, make the necessary modifications, and save it as a new version of the master class definition. Alternatively, a completely new master class can be saved with a different name.

In either case, the effect is to ensure that previously defined master classes, and their associated nodes, which may already be used in existing network designs are not affected by the enhancement being made.

6 Support for Data Integration Services

From the earliest design concepts, SimICI has contained strong support for Data Integration Services. Design and architectural decisions have been made with Data Integration Services at the forefront of the decision process and, as Figure 1 illustrated previously, SimICI makes great use of Data Integration in support of the CIRP, Data, and External domains it coexists with.

The most basic support for Data Integration lies in the use of Shapefiles, via a WebDAV resource, to facilitates the exchange of data between CIRP and SimICI. It should be noted that this mechanism is not restricted to CIRP integration as any application with access to the WebDAV resource and the ability to process the Shapefile format may also make use of this service.

The Shapefiles used to send data from SimICI to CIRP are generated – manually or programmatically - by the SimICI Geoserver. The SimICI Geoserver, through an open standards API, also provides support for access to data in different formats. As all data generated through the ANDI methodology in SimICI is persisted in the SimICI Geoserver, this means that whatever is put in at the SimICI frontend is persisted and made available not only to EU-CIRCLE systems but also to external parties outside of the EU-CIRCLE project. It is this mechanism that will be exploited in support of the Open Data Pilot and associated community outreach activities.

Figure 17, below, illustrates the use of the SimICI Geoserver API to retrieve the EU-CIRCLE VDS Rethymno Drinking Water Network feature data in GeoJSON format.
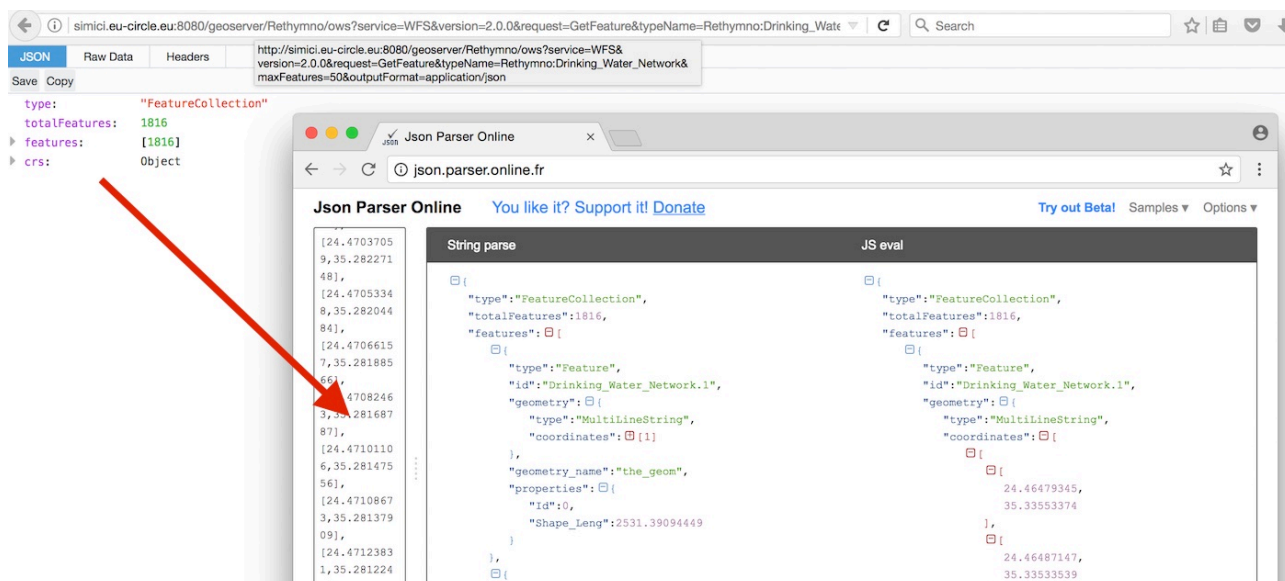

Figure 17: Data Services from the SimICI Geoserver (Example)

The approach used in Figure 17 would, typically, be employed within a software application in order to provide that application with structured, machine readable data that could then be processed by the application. The same approach, using a subtly different API call, may be used to provide data from EU-CIRCLE in KML (Google Earth), GML (eXtensible Markup Language (XML) standard maintained by the Open Geospatial Consortium (OGC)), or other useful formats.

As mentioned in the previous section, the graph data representing a CI network (of networks) may also be directly exported for alternative processing that is concerned with graph analytics or related topics. At a later stage, however, it is possible that the main SimICI web application would look to integrate graph analytics as a native capability.

Finally, with regard to the EU-CIRCLE Wiki, the ACR database should be used as the master source for all asset classes addressed by EU-CIRCLE. This use would ensure that changes made in the ACR automatically propagated to the EU-CIRCLE Wiki such that (a) the Wiki contained accurate and up-to-date information as to the extent of asset types addressed by EU-CIRCLE and (b) the ACR becomes elf-documenting via the Wiki.

As the EU-CIRCLE ACR is to form part of the Open Data Pilot deliverables, it is strongly anticipated that third parties will make additions to the ACR and, therefore, the self-documenting capability will be necessary in order to understand exactly how the ACR is evolving.

It should also be noted that the ACR asset data used to update the EU-CIRCLE Wiki could also be used to provide support to external software (for example, in providing an index to the contents of the SimICI Geoserver.

7   In Closing

This document has discussed the full range of scenario supports available from the SimICI system.

SimICI offers demonstrable support to all aspects of the EU-CIRCLE project objectives and provides a full range of services across the EU-CIRCLE Systems View.

It is reiterated that this report contains a number of forward looking statements reflecting the fact that SimICI is under redevelopment at the time of writing. The redevelopment will conclude by the end of September 2017 and the resulting final capabilities of SimICI will be documented in the future deliverable D7.9[6], due Month 33, which will detail the full capabilities of the SimICI within EU-CIRCLE.

8   Bibliography

[1] EU-CIRCLE Deliverable 3.1: Registry with CI assets, and interconnections (EUC)

[2] EU-CIRCLE Deliverable 5.4: Final Integrated CIRP Release (Document) (STWS)

[3] EU-CIRCLE Deliverable 7.4: Demonstrable EU-CIRCLE Scenarios in SimICI (NCSRD)

[4] EU-CIRCLE Deliverable 7.1: Demonstrable Deployment of Integrated SimICI (Document) (XUV)

[5] EU-CIRCLE Deliverable 7.8: Administration and User Manuals for SimICI System V2 (XUV)

[6] EU-CIRCLE Deliverable 7.9: Administration and User Manuals for SimICI System Final (XUV – Future)

[7] EU-CIRCLE Deliverable 7.3: Administration and User Manuals for SimICI System (XUV)

** ENDS **