# A modelling language for the resilience assessment of networked systems of systems

Roberto Filippini
*Institute for the Protection and Security of the Citizen, Joint Research Center of the European Commission, Ispra, Italy*

Andrés Silva
*Facultad de Informática, GIB Research Group, Universidad Politécnica de Madrid, Spain*

ABSTRACT: Systems of Systems (SoS) present challenging features and existing tools result often inadequate for their analysis, especially for heteregeneous networked infrastructures. Most accident scenarios in networked systems cannot be addressed by a simplistic black or white (i.e. functioning or failed) approach. Slow deviations from nominal operation conditions may cause degraded behaviours that suddenly end up into unexpected malfunctioning, with large portions of the network affected. In this paper,we present a language for modelling networked SoS. The language makes it possible to represent interdependencies of various natures, e.g. technical, organizational and human. The representation of interdependencies is based on control relationships that exchange physical quantities and related information. The language also makes it possible the identification of accident scenarios, by representing the propagation of failure events throughout the network. The results can be used for assessing the effectiveness of those mechanisms and measures that contribute to the overall resilience, both in qualitative and quantitative terms. The presented modelling methodology is general enough to be applied in combination with already existing system analysis techniques, such as risk assessment, dependability and performance evaluation.

## 1 INTRODUCTION

Civil and industrial installations do not work in isolation, and in many cases they form networks of Systems of Systems (SoS). Examples can be found in modern infrastructures such as power grids, ICT communications and transportation networks, civil emergency services and many other fields (Valerdi 2008, Maier 1998). What mostly distinguish a networked infrastructure from a complex system is the open architecture. The diverse elements interconnect as long as they possess the requisites of interoperability and they must also be able to adapt to the user's demand. Nonetheless, several problems of integration exist when dealing with systems that are heterogeneous and even more subtle issues come into play when considering hazards and accident scenarios.

These issues are recognized as difficult to model and often beyond the capabilities of traditional engineering tools, especially when they concern the analysis of interdependencies, vulnerabilities and resilience. Because of the diversity of the considered quantities, these are often addressed separately. Interdependencies analysis is addressed on the structural representation of the network (Rinaldi et al. 2001, Panzieri et al. 2008, Laprie et al. 2007), while vulnerabilities may be structural and behavioral at the same time (Egan 2007, Bompard et al. 2009, Ouyang et al. 2009, Johansson et al. 2010), and resilience is mostly behavioral (Madni et al. 2009, Hollnagel et al. 2006). This approach is effective to reduce the scope of the analysis, though it underestimates the existing mutual implications among the different quantities.

This paper presents a language for modeling the resilience of networks and infrastructures, the Infrastructure Resilience-oriented Modeling Language (I®ML). The language aims at broadening the scope of the representation to all players that may take a role in operation scenarios and are relevant to resilience. These players/components are called domains, and maintain their specificities (technical, organizational) without being an obstacle to the representation that, for this reason, is heterarchical andcross-sectoral.

The I®ML language is not specifically addressed to system design, rather it is conceived to support decision making and may be integrated into a risk assessment framework. The language is also accompanied by tools for the analysis of interdependencies, vulnerabilities and resilience. A case study is taken from the NIST smart grid in order show an example of model building with I®ML and further resilience analysis.

The paper is structured as it follows: section 2 provides an overview of the proposed language; section 3 introduces a few analysis insights; section 4 presents the case study, which is analyzed in section 5. Section 6 contains final remarks and conclusions.

## 2 I®ML: INFRASTRUCTURE RESILIENCE MODELLING LANGUAGE

The I®ML model of an SoS is heterarchical, with technical and organizational elements considered together. In I®ML functions/services are more important than the physical implementation and behavioral elements (the control relationships) are also included. These modelling features will be illustrated in this paper on a Smart Grid case study (NIST 2010). The I®ML model is shown in figure 1.

The key elements in any I®ML model are "domains" and "resources". The concept of "domain" comes from Jackson's Problem Frames (Jackson 2000). In I®ML, a domain is a set of phenomena that represents a technological or organizational component in the real world. There are also "resources", that are special kind of domains that represent goods, materials and information either produced or consumed. They relate to each other via control relationships, which support the exchange of information among the elements and pursue a particular goal. A control relationship is represented by an oriented arc with a filled "o" in the controller side and an empty "o" in the controlled side. For instance, in figure 1 there is a control relationship that ties together the "Customer Energy mgmt. System (5)" (controller domain) and the "Customer Appliances and Equipment (3)" (controlled domain). The relationship may also be unidirectional, e.g. a domain can read the state of another domain,but can not change it. This is expressed with an arrow (for example, in figure 1, the "meter (8)" is read, but not changed, by the "Customer Energy mgmt System (5)"). Connections among domains and resources express the fact that a domainconsumes, or provides, a resource.

A set of tightly connected domains constitutes a system and every system provides a service. Sys-

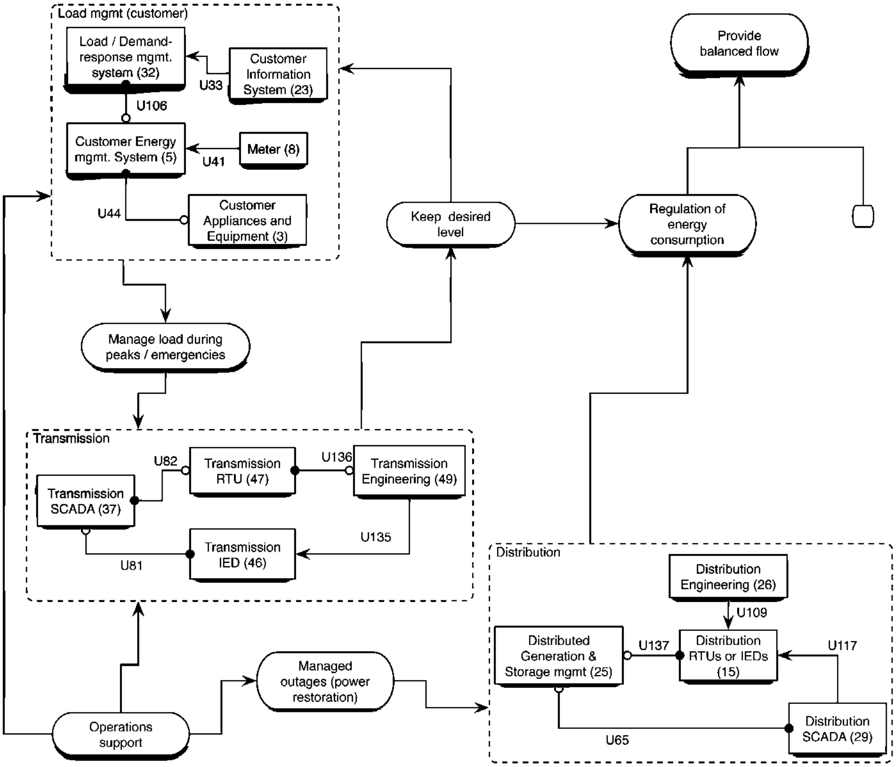
Load mgmt (customer)
Load / Demand-response mgmt. system (32)
Customer Information System (23)
U33
U106
Customer Energy mgmt. System (5)
Meter (8)
U41
U44
Customer Appliances and Equipment (3)
Provide balanced flow
Keep desired level
Regulation of energy consumption
Manage load during peaks / emergencies
Transmission
U82
Transmission RTU (47)
U136
Transmission Engineering (49)
Transmission SCADA (37)
U135
Transmission IED (46)
U81
Distribution
Distribution Engineering (26)
U109
Distributed Generation & Storage mgmt (25)
U137
Distribution RTUs or IEDs (15)
U117
Managed outages (power restoration)
Distribution SCADA (29)
U65
Operations support

Figure 1. I®ML partial representation of the Smart Grid (NIST).

tems are delimited by dashed lines, and an arrow links the system with the service provided. For example, in figure 1 the "transmission" system provides the "keep desired level" service. The figure also shows how this system relies on other services ("manage load during peaks and emergencies" and "operations support"). A service can also relate to another service, for example "regulation of energy consumption" and "provide balanced flow". For a detailed description of the language, see (Silva & Filippini 2011).

The idea of "domain", despite its apparent abstractness, is actually what will allow I®ML to model cross-sector issues, and resilience is cross-sector. This approach worked very well in Jackson's Problem Frames (Jackson 2000). For design, more accurate, sector-specific models are required. However, I®ML is not a design-oriented model and its purpose is to support resilience analysis.

## 3 SYSTEM RESILIENCE ANALYSIS AND I®ML

### 3.1 *Modeling and analysis framework*

The I®ML model is part of a modeling and analysis framework specially conceived to assess resilience in networked SoS. This framework includes the following stages:

1. I®ML model construction,
2. Interdependency analysis, with the Goal Dependency Structure,
3. Generation of resilience scenarios and
4. Resilience analysis.

The I®ML model was already introduced in the previous section. The other stages of the framework are described in this section.

### 3.2 *The goal dependency structure*

A Goal Dependency Structure (GDS) represents the interdependencies, expressed in term of goals, among the components of an SoS, for an assumed nominal operational scenario. Any I®ML model can be transformed in a GDS. The GDS of figure 2 is derived from the I®ML in figure 1. The passages necessary to transform the I®ML model into the goal dependency structure are here omitted. A detailed explanation can be found in (Silva & Filippini 2011).

The services that are provided by a system, are here represented with large ovals and further specified with the internal control goals that must be satisfied in order to provide the service. Systems are represented with dashed squares, but the GDS can beexplored ignoring them, as they serve just to label the related goals. In this way, dashed lines delimit the goals related to the delivery of a service. The goals refer to the control relationships that are established among domains in the I®ML model.

There are two kinds of relationships in a dependency structure:

1. *Depends-on* relationship: expresses that a service depends on another service, or that a goal depends on a service, and they are represented with arrows. For instance, the service "regulation of energy consumption" depends on "keep desired level". Reading backwards, if "keep desired level" is not achieved, this will impact on "regulation of energy consumption".
2. *Consists-of* relationship: expresses which system goals, within a system, are needed to provide a service. Conversely, it expresses that a system-provided service is carried out if some system-internal goals are achieved, at varying levels of quality. It is said, then, that the service *consists-of* those goals. Graphically, it is expressed with a line ended by a diamond. For instance, the "manage load during peaks" service consists-of the goals labelled as U106, U33 (mediated by the "load management system" which, as said, can be ignored as it is just shown for the purpose of illustrating the origin of the goals) and U44, U41 (mediated by the "customer energy management" box).

### 3.3 *Resilience scenarios and resilience analysis*

A "resilience scenario" is the evolution in time of the off-nominal set-up. In this respect, it is the dynamic representation of what can go wrong. Dynamics are driven by events, either undesired or generated by the response of the systems. Each resilience scenario is a particular sequence of events that goes from the non-achievement of a goal to the final end state. This can be the recovery back to the nominal operation scenario or a non-recoverable scenario. Indeed, the process of restoration of services in the network is crucial and it may happen that it cannot be completed.

The generation of resilience scenarios consists of the definition of the off-nominal scenario set up, from which the resilience scenarios are generated. An off-nominal set-up consists of: 1) one goal that is assumed to be non achieved, and 2) the set of goals that are potentially affected. The off-nominal set up is obtained by exploration of the goal-dependency structure, in an inductive way: e.g. the non achievement of goal A affects goal B and C, and so forth. It is a static representation of what can go wrong, and to which extent the undesired event can propagate. This is not a combinatorial technique of system analysis, like reliability graphs

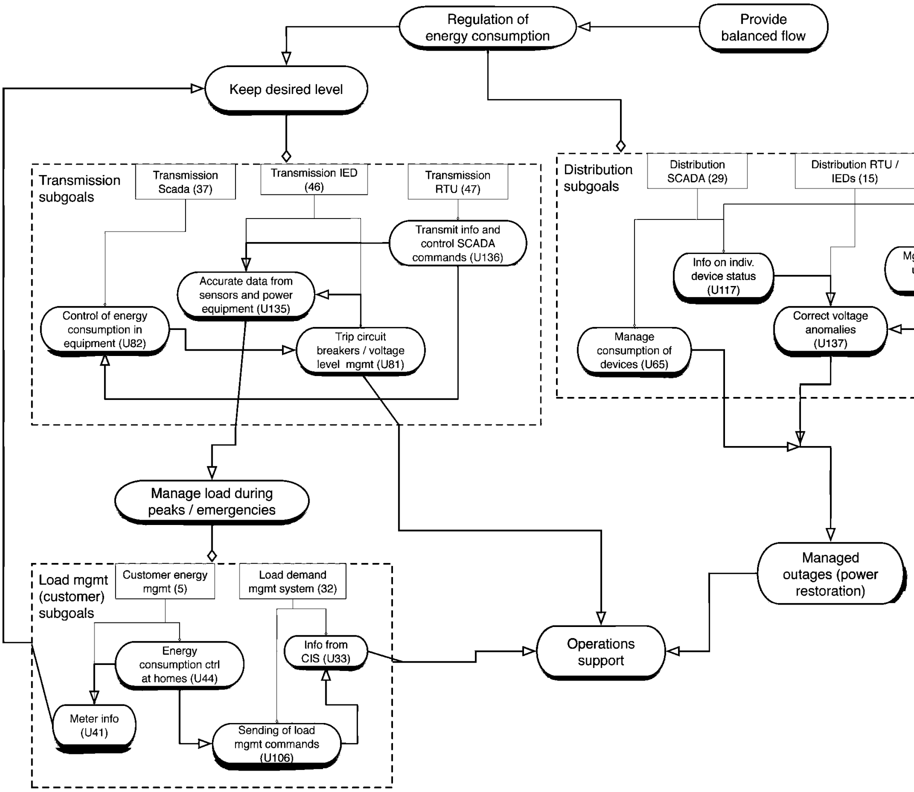
Regulation of energy consumption
Provide balanced flow
Keep desired level
Transmission subgoals
Transmission Scada (37)
Transmission IED (46)
Transmission RTU (47)
Transmit info and control SCADA commands (U136)
Accurate data from sensors and power equipment (U135)
Control of energy consumption in equipment (U82)
Trip circuit breakers / voltage level mgmt (U81)
Distribution subgoals
Distribution SCADA (29)
Distribution RTU / IEDs (15)
Info on indiv. device status (U117)
Mgmt of load upgrades (U109)
Correct voltage anomalies (U137)
Manage consumption of devices (U65)
Manage load during peaks / emergencies
Managed outages (power restoration)
Load mgmt (customer) subgoals
Customer energy mgmt (5)
Load demand mgmt system (32)
Energy consumption ctrl at homes (U44)
Info from CIS (U33)
Meter info (U41)
Sending of load mgmt commands (U106)
Operations support

Figure 2. Goal Dependency Structure.

or fault trees, as the non achievement of a goal only returns those goals that are potentially affected due to the propagation of an undesired event.

The identification of resilience scenarios is of inductive nature and systematic. The analyst has at his disposal a finite number of options, that define the "what-if" in terms of system response, and the "what-happens-next" to those goals that are inthe neighborhood. The non achievement of a goal implies that the system is challenged to survive. Given this challenge, the system may resist, react or break down. We provide a systematic approach for the identification of the resilience scenarios from an assumed non-achieved goal. The resilience measures counteract different events in a diverse way. (i) Preventive measures implement robustness to perturbations and tolerance to faults. They are able to locally contain the effect of perturbation/fault, up to certain threshold, thus avoiding the propagation. (ii) Reactive measures counteract misbehaviors by controls in combination with fault management systems (for reconfiguration and recovery). As opposed to preventive measures, reactive measures cannot impair that a perturbation (or fault) may propagate. Sometimes they are the cause of the propagation.

The generation of resilience scenarios from the GDS, for the given goal, ends when all the goals of the off-nominal set-up have been visited. Clearly, the number of resilience scenarios generated may be very large. Still, it is important that the generation process is exhaustive. Pruning can be done, for instance, by clustering scenarios that develop in a similar way and/or terminate with the same consequences, or by prioritizing scenarios with respect to a judgment criteria, e.g. based on risk, performance, etc.

In the last stage of the framework, a resilience analysis is conducted on the resilience scenarios. There are different approaches and techniques that may be applied to the purpose. It will depend on the type of analysis required, e.g. probabilistic or deterministic, and the quantity of interest, e.g. performance, risk, etc. The idea is to consider this final step as an interface with existing analysis and simulation tools. For instance, Bayesian networks (Ben-Gal 2007) can be mapped to a dependency structure.

## 4 CASE STUDY: THE SMART GRID

### 4.1 *Scope and assumptions*

The presented example is taken from the "NISTIR 7628" documents on the Smart Grid. Those documents were prepared by the National Institute of Standards and Technology. In particular, the source material for this case study comes from the document "Guidelines for Smart Grid Cyber Security: Vol. 1" (NIST 2010), which includes a representation of the different high-level actors in the Smart Grid. Schemes and models in the NIST documents (NIST 2010) are complex, widely encompassing and represented with a particular, boxes-and-arrows, adhoc notation. This is an issue that pervades many representations of critical infrastructures, among other problems (Hollnagel et al. 2006, Egan 2007, Silva & Filippini 2011). In the reference documents there are different areas of concern and within each area there are actors that collaborate and communicate by several means and through a variety of interfaces. The NIST framework identifies seven areas[1] within the Smart Grid: **(1) Transmission, (2) Distribution, (3) Operations, (4) Bulk generation, (5) Markets, (6) Customer and (7) Service.**

For our modeling purposes, we have restricted the study to three areas of the Smart Grid: transmission, distribution and customer areas. Within the chosen areas we also made a selection of the core actors, whose joint collaborative goal is to avoid energydisruptions and provide a balanced energy flow. Those actors will be considered as I®ML domains, and grouped here into systems.

### 4.2 *Description of the Smart Grid I®ML model*

The I®ML model in figure 1 represents the three areas of the NIST Smart Grid model that are within the scope of the case studied. Names and identifiers of the different elements (actors and interfaces) are taken directly from the NIST document for the purpose of cross-referencing. A detailed description of the systems, the domains involved and the service provided follows.

The *load management (customer) system* provides the "manage load during peaks/emergencies" service. The system relies on the services "operations support" and "keep desired level". This system is composed by five domains:

**LMS/DRMS (#32)**: Load and Demand-Response Management System. Issues load management commands to customer appliances. It also issues signals aimed at increasing or decreasing customer loads. It controls the "customer energy management system" (domain #5) with the goal of sending local management commands to the customer site (U106). For taking the correct decision, it depends on the information provided by the Customer Information System CIS (#23).

**CIS (#23)**: Customer Information System. It consists of a set of applications for managing the company relationships with its customers. It must provide customer pricing and load information to the LMS/DRMS (#32).

**Meter (#8)**: Devices for measuring point-of-sale usage. Data are feed for the decision-making process carried out by the EMS (#5).

**EMS (#5)**: Customer Energy Management System. From the data provided by the meters (#8, U41) and the commands sent by the LMS/DRMS (#32, U106), this domain will issue those decisions aimed to efficiently control energy consumption (U44) at the customer end (#3).

**Customer end (#3)**: Customer Appliances and Equipment. They are customer appliances (toasters and televisions, for instance) which perform a specific function for the customer. Their energy consumption is controlled (U44) by the customer energy management system EMS (#5).

The *transmission system* provides the "keep desired level" service, where "desired level" refers to voltage "within the parameters established by the company". It relies on the services "manage loads during peaks/emergencies" and "operations support". The system is composed by the following domains:

**Transmission IED (#46)**: Transmission Intelligent Electronic Devices. Reads information from the Transmission Engineering equipment (#49). Its goal (U81) is to maintain the voltage level by sensing anomalies and sending voltage management commands (or tripping circuit breakers). It depends on data read from the Transmission Engineering status (U135)

**Transmission SCADA (#37)**: Transmission Supervisory Control and Data Acquisition. Transmits device status and sends commands to the Transmission RTU (#47) for controlling power system equipments and manage energy consumption (U82).

**Transmission RTU (#47)**: Transmission Remote Terminal Unit. Cooperates with SCADA (#37) by sending status information on equipment and transmitting the control commands (received from SCADA) to field equipment (U136) at Transmission Engineering.

**Transmission Engineering (#49)**: It consists of equipment between conductor lines, designed

[1] Actually, the NIST document calls "domains" to those areas, but that term conflicts with "domain" in I®ML . In this paper, the "domains" correspond to the actors in the NIST document.

for more than 345,000 volts. Its status is monitored by the Transmission IED (#46).

Finally, the *distribution system* provides the "regulation of energy consumption" service and relies on the service "managed outages". The system is composed by the following domains:

**Distribution Engineering (#26)**: Domain for planning and managing the design or upgrade of the distribution system (addition of new customers, reconfiguration of capital investments, etc.). The relevancy for this model stands in the fact that it isDistribution Engineering who informs the Distribution RTUs/IEDs (#15) about reconfigurations (U109).

**Distribution RTUs/IEDs (#15)**: Distribution Remote Terminal Unit and Intelligent Electronic Devices. Cooperates with the Distribution Generation (#25) with the goal (U137) of level maintenance by issuing control commands to correct frequency and voltage anomalies.

**Distribution SCADA (#29)**: Distribution Supervisory Control and Data Acquisition. It acquires and transmits distribution device status (U117) and manages consumption by controlling (U65) the Distribution Generation and Storage Management (#25) domain.

**Distr. Generation and Storage Mgmt (#25)**: Generation of electricity from small energy sources. Reduces the amount of energy lost in transmission, as it is generated near where it will be used, helping to regulate energy consumption (overall goal of the distribution system).

### 4.3 *Derived goal dependency structure*

The GDS derived from the I®ML model is shown in figure 2. Control relationships in I®ML becomes goals in the GDS. For instance, the "transmission" system subgoals correspond to the control goals of the domains "transmission SCADA" (#37), "transmission IED"(#46) and "transmission RTU" (#47). For the sake of convenience, the goals have been labelled with the name of the connection they correspond to. In this way, the GDS shows that "transmission SCADA" needs the U82 goalto be achieved, "Transmission IED" needs the U135 and U81 goals, etc. The U81 control goal depends also on the external "operations support" service, which is provided by another system that has been left out of the scope of this analysis.

The GDS shows several interesting dependencies. For instance, it shows that "meter info" has a dependency on the service Òmaintenance of the desired levelÓ. This is due to the fact that modern meters are connected to the electrical line just as any other electrical appliance, not via an independent line. Hence, if the level is not appropriate, the meters may malfunction and the returned info will become inaccurate.ÊExperts will be able to quantify the consequences of this scenario. In any case IRML helps to identify that loop, and forces analyst to consider it. On another side, we can see that some distribution system subgoals depends on the "managed outages" service. Managed outages (or power restoration, in other words) is one of the services that "distribution" relies on (distribution relies on power to be restored). In addition, "accurate data from sensors" in the transmission system has a dependency relationship with the "manage load during peaks" service. This is because the sensors that communicate the transmission system status are necessary in order to manage voltage (or trip circuit breakers). The load is managed at the customer side, as fluctuations can lead to inaccurate readings. It could happen that a quick peak makes the distribution system decide that some action is to be taken, erroneously.

## 5 ANALYSIS OF RESILIENCE SCENARIOS

A key feature of a Goal Dependency Structure (GDS) is that, by exploring it, it is possible to identify the goals that are reachable from a given goal (and that depend on it), under "nominal operation conditions". The exploration is facilitated by the fact that the GDS is an oriented graph. The relationship *depends on* has the dependent goal on the tail place and because of that, the visit must be performed backwards the direction specified by the arcs. When a service goal is encountered, this is replaced with its depending goals.

The GDS provides the necessary information in order to perform a resilience analysis. Starting from the assumption that one goal is not achieved, it is possible to evaluate the potential impact onto the remaining goals. Each goal of the GDS is given a set of resilience measures, which are associated to the non achievement of that goal, either because of internal or external causes. The effect (of the non achievement of a goal) may remain confined, or it propagates to the next goal. A resilience scenario is complete when a terminal event is reached. This may be a recovery action (back to nominal conditions) or a non recoverable failure.

An example of generation of resilience scenario is analyzed in the case that the meter info goal U41 (g1) is not achieved. Six goals depend on g1: "energy consumption U44" (g2), "accurate data from sensors and power equipments U135" (g3), "trip circuit breakers U137" (g4), "control of energy consumption U82" (g5), "transmit info and controls U136" (g6), "regulation of energy consumption" (g7). The service "provide balanced flow" is obviously concerned too, though it is not included in the example. In order to understand to which extent these goals are affected, one has to consider the resilience measures in place. In the case studied, we assumed that every goal may have two measures: detection of the perturbing event (d) and recovery (r). Detection and recovery are abstractions of actual technical or organizational measures.

The resilience scenarios generated from goal g1 are shown in Figure 3. The first scenario (S1) corresponds to the (event) goal g1 non achieved, detected and recovered. If goal g1 is not achieved (-), propagation is possible up to goal g6, as goals 2–5 are here assumed to be passive (see definition in the following paragraphs). Again two possibilities exist. The non achievement of goal g5 is detected at g6 and recovery is performed, which corresponds to scenario S2. The effectmay also propagate to g7, which is the worst case. The SCADA will generate inconsistent controls, thus affecting the regulation of energy consumption goal g7. The goal g7 may detect the off-nominal conditions, stop propagation and recover, giving origin to scenario S3. But it is also possible that propagation affects next

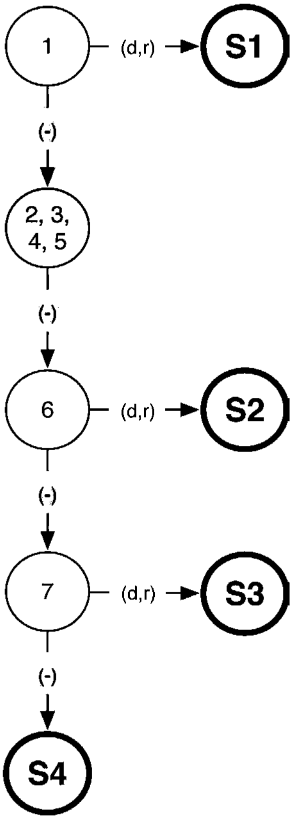

Figure 3. Resilience scenarios generated by goal g1 non achieved (meter info).

goal, leading to scenario S4. This is the last generated scenario, for the given example. The analysis may continue if other components outside of the scope are included.

In total, four resilience scenarios are identified from the non achievement of goal g1, three of which lead to the successful recovery back to nominal conditions. It is important to remark that the later the detection and the bigger the time to recover, and this impacts onto the duration of the service outage and the consequences. Early (local) detection is also the most effective measures to contain a fault and avoid propagation, though it might not be always possible. When the propagation affects a large number of goals, for instance in S3, the recovery process will be longer and more complex. Recovery time also matters as it is not excluded that during the recovery process, other goals will be affected.

The set of resilience measures per goal depends on the incoming arcs (i.e. the neighbor goal dependencies). As it was shown in the example, they can be active or passive with respect to the dependent goal and perturbing event. A goal is defined "active"with respect to a dependent goal, if the perturbing event is locally detectable and counteractable (e.g. g6, g7). The goal is defined "passive" with respect to a dependent goal if the perturbing event is not detectable and counteractable (e.g. g2, g3, g4 and g5). This distinction is important and enriches the expressiveness of the model.

There are other elements relevant to the analysis. The larger the set of reachable goals, the more critical the goal is. This information is clearly related to the network vulnerability. If combined with the identified resilience scenarios, it can assess the potential risk, in a qualitative way.

The example aimed at explaining how to build a resilience scenario from the GDS. The number of goals is here low, and the dependencies can be easy identified from the GDS, as well as the resilience scenarios. For more complex case studies, some guidance from experts is envisaged in order to avoid state explosion problems. Experts know the systems and may judge if the possible alterations in the nominal scenario are (quantitatively or qualitatively) sufficient to cause any consequence. The space of search can be reduced by eliminating those scenarios that are either non significant or very unlikely to happen.

## 6 CONCLUSIONS

This paper presented the IRML language for the representation of networked infrastructures and their resilience analysis. A case study from the NIST smart grid served to explain the language and the analysis tools. Analogies and differences exist with respect to other methodologies of system analysis. The identification of the resilience measures for each non achieved goal is in good analogy with the FMEA (Hoyland & Rausand 1994). In FMEA a component fault can be tolerated, detected, diagnosed and recovered, as well as in IRML a non achieved goal can be counteracted by the ability of the system of resisting, detecting and recovery. The analogy also exists with the Probabilistic Risk Assessment framework, and in particular with the construction of the event sequence diagram (Fullwood 2000), which resembles to the generation of resilience scenarios from an assumed non-achieved goal (the triggering event).

Among the differences and introduced innovations, this language is conceived to describe the behavior of a networked system at higher level. The scope of the representation includes components and players that are heterogeneous and cross-sector. The goaldependency structure is another important innovation. Although resembling to a logic tree-structure, it cannot be compared to any of the existing combinatorial system analysis techniques (i.e reliability graph, fault tree) (Hoyland & Rausand 1994). The non achievement of a goal in the GDS discloses a number of scenarios in which the system is challenged to resist, react, recover, in a word showing its resilience features (Madni et al. 2009). The fact that there is no stringent logic, makes the search space (of what can go wrong) to be very large. This will require rules forbounding the generation of scenarios only to those that are realistic and significant.

The set of resilience scenarios is the final outcome of the IRML modeling framework. This is also the input to further system analysis, which we envisage as one of the next research topics. The resilience scenarios, analyzed qualitatively in the IRML framework, could be analyzed quantitatively by existing system analysis tools, by simulations or even reproducing them by experiments, thus integrating the diverse methodologies into a comprehensive modeling framework.

## ACKNOWLEDGEMENTS:

Dr. Silva participation is partially funded by the DICODE project (FP7-ICT-20095257184).

## REFERENCES

I. Ben-Gal (2007). Bayesian Networks. in *Encyclopedia of Statistics in Quality and Reliability*. Ruggeri, Fabrizio, Kennett, Ron, Faltin, Frederick (editors). John Wiley & Sons.

E. Bompard, R. Napoli, F. Xue (2009) Analysis of Structural Vulnerabilities in Power Transmission Grids. *International Journal of Critical Infrastructure Protection 2*, 5–12.

M.J. Egan (2007). Anticipating Future Vulnerability: Defining Characteristics of Increasingly Critical Infrastructure-like Systems. *Journal of Contingencies and Crisis Management 15(1)*, 4–17.

A. Fritzon & K. Ljungkvist & A. Boin & M. Rhinard (2007). Protecting Europe's Critical Infrastructures: Problems and Prospects. *Journal of Contingencies and Crisis Management 15*, 30–41.

R.R. Fullwood (2000) *Probabilistic safety assessment in the chemical and nuclear industries*. Butterworth-Heinemann.

E. Hollnagel, D.W. Woods, N. Leveson (Editors) (2006) *Resilience Engineering: Concepts And Precepts.* Ashgate.

A. Hoyland and M. Rausand (1994) *System Reliability Theory: Models and Statistical Methods*. Wiley.

M. Jackson (2000) *Problem Frames: Analysing and Structuring Software Development Problems*. Addison-Wesley.

J. Johansson, H. Hassel (2010) An approach for modelling interdependent infrastructures in the context of vulnerability analysis. *Reliability Engineering and System Safety 95(12)*, 1335–1344.

J.C. Laprie, K. Kanoun, M. Kaaniche (2007) Modelling Interdependencies between the Electricity and Information Infrastructures. *Lectures Notes in Computer Science. Computer Safety Reliability and Security 4680*, 54–67.

N. Leveson (2004). A new accident model for engineering safer systems. *Safety Science 42(4)*, 237–270.

A.M. Madni and S. Jackson (2000). Towards a conceptual framework for resilience engineering *IEEE Systems Journal 3(2)*, 181–191.

M. Maier (1998). Architecting principles for system-of-systems. *Systems Engineering 1(4)*, 267–284.

National Institute of Standards and Technology (2010). *Guidelines for Smart Grid Cyber Security: Vol. 1, Smart Grid Cyber Security Strategy, Architecture and High-Level Requirements*. Available at http://www.nist.gov/smartgrid.

M. Ouyang, L. Hong, M. Zi-Jun, Y. Ming-Hui, Q. Fei (2009) A Methodological Approach to Analyze Vulnerability of Interdependent Infrastructures. *Simulation Modeling Practice and Theory 17(5)*, 817–828.

S. Panzieri, R. Setola (2008). Failure Propagation in Critical Interdependent Infrastructures. *International Journal in Modeling identification and Control 3(1)*, 69–78.

S.M. Rinaldi, J.P. Peeremboom, T.K. Kelly (2001) Identifying, Understanding and Analyzing Critical Infrastructure Interdependencies. *IEEE Control Systems Magazine 21(6)*, 11–25.

A. Silva & R. Filippini (2011). Infrastructure (Resilience-oriented) Modelling Language: IRML. A proposal for modelling infrastructures and their connections. *JRC Scientific and Technical Reports*. JRC63302. JRC of the European Commission.

R. Valerdi et al. (2008) A Research Agenda for System of Systems Engineering. *International Journal of System of Systems Engineering 1*, 171–188.