Needs and Architectural Strategies Related to Geospatial Information in Systems-of-Systems

Jakob Axelsson
School of Information, Design and Technology
Mälardalen University
Västerås, Sweden
jakob.axelsson@mdh.se

***Abstract*—This paper puts forward the hypothesis that all systems-of-systems (SoS) need to deal with geospatial information. It discusses some fundamental aspects of such geodata, including entities, coordinate systems, features, and representation. It then presents how geodata can be used for various purposes in SoS and suggests architectural strategies for handling geodata in this context, including the use of linked data to represent both geodata and other information; triple stores for databases; and cloud servers for executing geodata related constituent system functionality.**

Keywords—system-of-systems, interoperability, geospatial information, geodata, linked data.

I. INTRODUCTION

Systems-of-systems (SoS) are commonly described using Maier's five criteria, namely operational independence of constituent systems (CS); managerial independence of CS; evolutionary development; emergent behavior; and geographical distribution [1]. Of these, the first four have been widely studied in the literature on SoS, but the geographical distribution has received less attention. Maier points out that geographical distribution "means that the components can readily exchange only information and not substantial quantities of mass or energy." In essence, the geographical distribution is taken to primarily mean that SoS integration is a communication issue.

Although the aspect of integration through information exchanges is certainly a key aspect in SoS, we believe that there is more to the geographical distribution than that. In fact, we somewhat boldly put forward the following general hypothesis:

SoS always need to deal with geospatial information.

The intuitive argument for this is that since the CS are located in different positions, and since groups of CS will form constellations [2] that collaborate to deliver one of the capabilities of the SoS, there will always be situations where they need to exchange information about the different CS positions. They also need to treat information about the relations between those positions, meaning that there has to be a description of the geospatial environment where the CS are located. This may also include positions of other assets that the CS interact with when carrying out the mission.

A. Application areas

Some application areas where SoS are common illustrate how geospatial information (or *geodata*, for short) can be important:

- *Transportation*: The purpose of a logistics SoS, whether it is based on land, sea, air, or space, is to move assets between locations. A CS must then know the locations involved, as well as possible routes between them. They may also need to know the location of other CS which may be involved in the same logistics chain, to avoid interference.
- *Military*: The purpose of a military SoS is usually to either attack or defend an area or an asset, by combining the capabilities of different CS. The geographical position of the area or asset, as well as those of friendly and enemy forces, are crucial.
- *Disaster management*: In an SoS that handles the consequences of natural events, such as flooding or forest fires, the geographical location of the events is crucial. Also, assets that need to be protected, such as people's homes or critical infrastructure, have positions, as do the different CS that engage in the situation.
- *Surveillance*: An SoS which collects and fuses observations of the environment relies critically on geodata to describe where the observations occurred. This can apply on a global scale using satellites, such as in the Global Earth Observation System of Systems (GEOSS) [3], or very locally using e.g. formations of drones.
- *Construction and mining*: The building of houses and infrastructure involves substantial logistic SoS, but the activities also contribute in modifying the landscape above and underneath the surface. Information about the current state of a changing environment must be disseminated across the CS [4][5].
- *Agriculture*: An SoS related to farming needs detailed information on the condition of the soil at various locations, where seeds and plants have been put, and on the operations of the different CS, such as farming machines.
- *Financial*: A financial SoS, such as a payment or trading system, needs to be aware of the country in which transactions takes place, since different regulations apply in different parts of the world.

With that said, the representation and level of detail of the geodata will vary broadly between applications.

B. Overview of paper

Assuming that the above hypothesis applies, the important question becomes what strategies SoS engineering (SoSE) can use to deal with geodata. The main contribution of this paper is to provide an initial description of a relevant approach.

The remainder of this paper is structured as follows: In the next section, some related work is introduced. In Section III,

---

This research was funded by Vinnova (Sweden's innovation agency) under grant no. 2018-03244.

an overview of different kinds and aspects of geodata is presented. Then, in Section IV, various needs related to the use of geodata in SoS are discussed. Section V proposes some concrete architectural strategies for designing geodata mechanisms in SoS. The final section summarizes the conclusions and gives some indications of future work.

II. RELATED WORK

This section reviews some of the related work on the use of geodata in SoS. One of the most ambitious examples is the Global Earth Observation System of Systems (GEOSS), which integrates geospatial data from many sources around the world. When it comes to the SoS aspects of this, Butterfield et al. present the architectural approach behind GEOSS with a focus on how SoSE is applied to this problem [3]. Christian describes how the GEOSS architecture is used in a specific component called the clearing house, which is a cross-cutting catalogue of the registered services [6]. Khalsa et al. describe an interoperability test scenario of GEOSS and discuss the need for mediating services [7].

Other applications of geodata in SoS include precision agriculture, where Delgado et al. discuss how to make geodata accessible using WebGIS, which is an architectural pattern for implementing geodata using web technology [8]. Lenk argues that geospatial and geodetic engineering should be considered an important specialty engineering discipline in SoSE [9]. He reviews the various specialty engineering fields mentioned in the INCOSE handbook and discusses how they relate to geodata. Chaves et al. present an approach for building SoS based on sensor network, where an ontology-driven semantic registry is used. It is exemplified with a crisis management scenario [10]. Lee and Sekar present an SoS architecture for creating and managing geospatial datasets [11]. Key concerns include scalability, extensibility, evolvability, and efficiency. Snyder et al. address the collection of high-fidelity raster terrain models using aerial lidar scanners for use in distributed testing of SoS for military applications [12].

III. FUNDAMENTALS OF GEODATA

The use of geodata has been studied extensively in the field of geographic information systems (GIS), resulting in a large number of concepts, principles, and standards that are commonly used for cartography, geography, remote sensing, and surveying. Here, only a brief overview of some basic ideas will be given as a basis for the later discussion on its usage in SoS.

Geodata can come in many different forms depending on what physical entities it describes, and how it is represented. This section gives an overview of some key aspects with relevance to SoS.

A. *Physical entities*

Geodata is used to describe a certain part of the world we live in, and there are two key kinds of subjects, namely assets and environment, where spatial information can be used:

1) Assets. Delimited objects, such as man-made entities and easily distinguishable parts of the environment can be treated as assets. This category includes the CS of an SoS. Often, assets can be broken down into parts that do not overlap spatially.

2) Environment. Some parts of the physical environment are not easily distinguishable, but must be included in an overall environment description. This applies in particular to the topography of the landscape. For instance, it is quite hard to say where a mountain begins or ends, but it gradually morphs into the surrounding plains. Subsets of the environment can still be viewed, but it is an arbitrary choice where to put the limits.

B. *Coordinate systems*

Since all geodata refers to positions, it is a key question to be able to describe positions clearly and numerically, in order to say if two positions are in fact the same, or what the distance is between them. For this, coordinate systems are used which relate a position to a chosen reference point. There is an infinite number of possible coordinate systems, and depending on how the data is to be used, one or the other may be more appropriate. Therefore, all usage of geodata must clearly state what coordinate system is applied. Some examples of useful coordinate systems include:

1) Geodetic. Gives coordinates in terms of latitude and longitude positions on the surface of a reference ellipsoid, and possibly also vertical position above or below the surface. Depending on the choice of ellipsoid, different places on earth will end up with different positions. A common global geodetic coordinate system is WGS84, used by e.g. GPS. However, the shape of the planet is not a perfect ellipsoid, and therefore local deviations can be large, which sometimes motivates the usage of other reference ellipsoids that better match reality in a region.

2) Planar. In traditional maps, it is common to map the positions on an ellipsoid to a planar representation that allows two-dimensional presentation. Approximating a part of the ellipsoid as a plane usually works well for small areas, but leads to distortions that become larger as the area grows. We also tend to act as if living in a planar world, and the language for expressing many decisions by a CS is planar, such as for a vehicle to move in a certain direction.

3) Local. Within an asset, it is often natural to use a local coordinate system, that indicates the position of different parts relative to a reference position. In particular, as the whole asset moves around, the geodetic or planar positions of the parts will vary, but the positions in the local coordinate system remain the same.

C. *Features*

Once assets and environmental areas have been identified and positioned, it is necessary to associate more information to them. This can include attaching labels to them, such as the name of a city or a lake. However, many other features are possible, and can include cultural aspects such as administrative boundaries; socioeconomic data such as demographics; environmental data such as soil material and climate; hydrographic data of oceans, lakes, and rivers; and elevation data of terrain.

D. *Representation*

The geodata needs to be encoded in some form, and here, different levels of abstraction can be chosen.

1) Topology. The simplest way to describe places and interesting relations between them is as a logical graph, where the positions are nodes and the relations are links. For example, this can often be a sufficient representation in logistics, where destinations are the nodes and the roads between them are links [13]. It is often not interesting to know the exact location of every bend on the road, but it suffices to represent it with a few features such as distance. Also, the exact location of a node is not necessarily important, but it can be enough to know how to get there from a given place.

2) Vector. Geodata elements can be specified using points, lines, and polygons in space, or collections of such elements. This can describe the location of an item of interest (point), a connection between two locations such as a segment of a road (line), and an area such as an administrative region or the surface of a lake. Also, surfaces can be represented by triangular irregular networks or other types of meshes specifid by vectors.

3) Raster. Features of a surface can be described using matrices giving the value of a feature at regular intervals in a plane. For instance, a raster can contain information about terrain elevation in cells of a certain size, the usage of land, etc. Raster data is often the output of remote sensing systems, using cameras, radars, or lidars.

E. Standards

Within the GIS community, many different formats have been developed for storing and transferring geodata files. This includes dealing with different coordinate systems; standardized ontologies for describing features; and usage of vector and raster representations. Currently there appears to be some convergence towards the ISO standard Geography Markup Language (GML), which is an XML based format defined by the Open Geospatial Consortium (OGC) [14]. Nevertheless, many other formats remain in use and there are software libraries that perform conversions between them.

GML also includes the possibility to define application schemas that extend the vocabulary for specific application domains. There is ongoing work to provide such schemas for infrastructure (referred to as LandInfra in its conceptual model, and InfraGML in its GML encoding), thereby providing a richer vocabulary for describing elements related to infrastructures.

When it comes to assets, such as buildings, vehicles, etc., their geometry and features have typically been described using CAD systems rather than GIS, and this makes it complicated when representing environments that contain a mixture of the two. For instance, a road environment contains the topography of the landscape, the layout of man-made objects such as the road and bridges, and the vehicles. In the CAD world, a widely used ISO standard is the Industry Foundation Classes (IFC) [15].

A major difference between geodata such as GML and CAD data like IFC is that the former primarily uses some kind of global coordinate system, whereas the latter uses local coordinates.

IV. GEOGRAPHICAL INFORMATION IN SOS

Based on the description of geodata in general from the previous section, some aspects related to its usage in SoS will now be discussed.

A. Usage

In an SoS, there are a number of uses of geodata, related to both development and operation.

1) World model creation. In general, each CS must contain a model of the world around it in one form or another to support its decision making, and this must also include geodata. Often, at least the environmental data will be initialized based on existing map sources, and the ability to load and select relevant map data is thus needed. Other aspects of the world model are updated continuously as the CS receives information through its sensors or through communication from other CS.

2) Communication. CS need to exchange geodata between each other. This includes positions of itself and other assets, but also sensory information in raster or processed form, that provides updates to geometric information about the environment. It is thus necessary to ensure geodata interoperability within the SoS.

3) Simulation. Evaluating the effectiveness of an SoS is difficult since many of the effects are only appreciated at scale, and thus a smaller prototype may not be representative. Therefore, larger scale simulations are often needed during development, and to make them realistic, it is necessary to create simulation models of the environment and assets, against which the CS and possible mediating systems in the SoS can test their functionality.

4) Digital twins. During operation, a CS may combine its world model data describing the current state of the environment with dynamic models of the elements in the world model. In this way, a digital twin of the relevant parts of the world is created, which it can use during its operational planning in order to evaluate alternative courses of action. Note that the time available for operational decisions is usually much shorter than for engineering decisions, and the dynamic models used in the digital twins thus typically have to be simpler than the ones used during development to ensure that many alternatives can be evaluated. On the other hand, the simulation time span is usually also shorter, and each CS only needs to include parts of the complete SoS which reduces the simulation effort.

5) Visualization. Most SoS are socio-technical, and many of the CS interact with human operators. To give the human users a sufficient situational awareness, visualization of geodata is often necessary.

B. Hierarchical views

Many SoS contain a certain level of hierarchy among their CS, as witnessed by the directed archetype often encountered in military applications [1] as well as the hierarchical centralized architecture pattern [16]. In a hierarchical setting, the following aspects related to geodata should be noted:

- *Abstraction.* CS high up in the hierarchy will use geodata features at a high level of abstraction, whereas CS further down will use more concrete details. For instance, in an agricultural SoS, the highest levels may only need to know if the land is used for farming or forestry, whereas a lower level CS may need to know the details of soil composition.
- *Resolution.* The level of detail needed is higher the further down the hierarchy one moves. As an example, in a logistics example, the higher levels may only need a coarse map with distances between destinations for route planning, whereas the lower

levels require the exact curvature and inclination of roads to execute vehicle control.

- *Coverage*. The higher up the hierarchy, the larger geographical area is usually considered. This is intuitive, since a high-level commander has the role to coordinate resources at lower levels, and thus the union of the areas relevant to all the resources have to be considered.

As information flows up and down the hierarchy, there is a need to translate information between the different levels of abstraction, resolution, and area sizes.

The differences in abstraction, resolution and coverage apply to both environmental data and asset data. It is common that geographical data has a large coverage with low resolution whereas CAD models typically have a limited coverage with much more details.

C. *Temporal aspects*

Things change over time, both as a result of the SoS operation and due to external factors, and the geodata collected in the world models need to reflect this. It is quite clear that assets including CS change, both by moving around and sometimes also changing their geometry. As an example, a pile of rock material used in a road construction SoS may shrink or grow over time as material is added from crushing or removed to be put into use.

Often, the environment is treated as more static, and a map is used to reflect reality with the assumption that it remains valid during the operation of the SoS. However, in some applications, the environment changes and it is crucial to keep the model of it updated. Continuing with the road works example, the building of the road includes removing a lot of soil to create a stable surface for the roadbed, and this is a clear modification of the environment. In a military application, an artillery attack may substantially damage an area, making it difficult to traverse with vehicles. In a disaster management scenario, flooding may cause the water level, and hence the boundary, of a lake to change continuously. This means that some SoS need to have mechanisms in place to exchange substantial amounts of geodata about the environment and include that data in the CS world models.

The distributed nature of an SoS poses further challenges when it comes to temporal data management in general. A CS will have data in its world models that describe the world at different times, since various data points get updated at varying frequencies and there are also communication delays. A world model will thus never describe the world as it is right now but will be a mix of evidence about how various parts of it looked at different times in the past.

So far, the focus has been on data at the current time, but often a CS also needs to deal with past and future data to identify trends, improvement and prognostics. Sometimes this is stored as a time series, possibly smoothing the data to reduce resolution, and at other times aggregated values such as averages over time suffice.

D. *Relating geodata to other information*

An SoS deals with other kinds of data which have relations to geodata. Basic ontologies for SoS typically include concepts such as goals; missions and tasks; capabilities; physical environment, and observations thereof [17][18]. In many applications, all these relate to geodata. For instance, in mining a goal is to extract valuable material from a certain space underground; a capability of a mining load-haul-dump machine CS is to move a certain amount of material per hour in spaces of a certain confinement; and the physical environment is the underground shafts at a given point in time. It is thus necessary to relate concepts of these other ontologies to both positions (including coordinate systems), and features (such as different classes of objects in the environment, and their geometries.)

E. *Geodata operations*

Based on the characterization of geodata above, the main operations needed for geodata in a CS can be summarized as the following basic operations:

- *Receive*. A CS needs to be able to accept geodata from other systems, and there is thus a need for interoperability on at least a semantic level with respect to geodata.
- *Send*: A CS needs to send geodata to other CS as agreed in the conditions for participating in the SoS.
- *Store*: A CS needs to be able to update stored geodata in its world model with new data, which could for instance contain changes that have occurred in the real world over time.
- *Retrieve*: A CS needs to search the geodata in its world model for information based on both features of objects, and for specific regions and specific points in time.
- *Transform*: A CS needs to aggregate geodata to get it on the desired level of abstraction, resolution, and area coverage. Other kinds of processing may also be necessary depending on the application.
- *Curate*: A CS needs to keep track of the validity of its data and adjust when necessary. This could include the removal of data that is now so old that it is unlikely to be correct; data that comes from an unreliable source or is contradicted by other data; or data that relates to a region where the CS is no longer operating.

V. ARCHITECTURAL STRATEGIES FOR GEODATA IN SOSE

To deal efficiently with geodata in SoS, some guidance is needed. In this section, we will describe some of the strategies we have used for handling geodata in conjunction with other complex data in SoS. The focus is primarily architectural, and deals with interoperability, storage, and allocation. Figure 1 gives an overview of the strategies that are discussed below.

A. *Interoperability*

In SoS engineering, a key concern related to geodata is to specify the required interoperability, which often goes beyond the syntactic and semantic levels. Another is to ensure that each CS receives and provides the information necessary to carry out its role in a way that is sufficient to make the SoS effective.

As proposed in [19], semantic web based concepts such as linked data and ontologies within the Resource Description Framework (RDF) provide a generic basis for interoperability in SoS. It has also been shown how other information related to SoS operations, as described in Section IV.D above, can be represented using linked data [18].

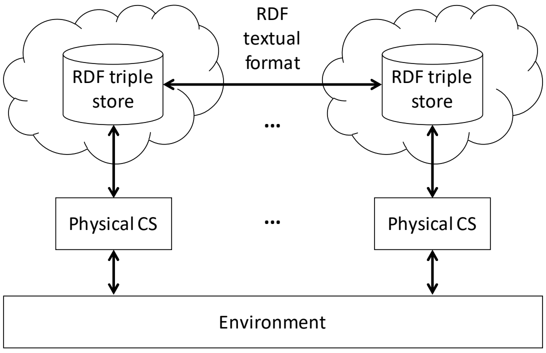

Fig. 1. Illustration of the architectural strategies.

1) Overview of RDF. In RDF, a key element is resources, which provide unique identifiers to all the things represented. Those identifiers are strings with a syntax very similar to the strings used for web addresses. This allows unique identifiers to be generated in a distributed way by means of convention. The information about the resources is expressed as triples on the format subject – predicate – object (where the predicate is sometimes also called a property). The subject and predicate are always resource identifiers, and the object may be either a resource identifier or literal data (e.g. strings, numbers).

To provide semantics to the linked data, a number of predefined resource identifiers exist that define common concepts from description logic. With these, it is possible to define classes of resources and subclass relations between classes; declare that a resource is an instance of a certain class; define properties of relations, such as the domain and range; etc. This ontological information is also represented as linked data triples, and it allows logical reasoning to derive further information from a set of triples.

2) Using RDF for geodata. It seems reasonable to use linked data also for geospatial information due to its strength for providing interoperability in general. Luckily, this is to some extent supported by the existing standards for geodata. Version 1.0 of GML (see Section III.E) had an official schema definition in RDF, but this appears to have been dropped in subsequent versions. However, according to [20], the structure of later GML versions is still based on the same principles as version 1.0, which implies that a translation to RDF is still feasible. For assets, the IFC standard (see Section III.E) seems appropriate, and it has official RDF schemas which can be readily used.

Due to the open nature of RDF, connecting different vocabularies is fully supported. In an SoS specific vocabulary, which e.g. defines ontological concepts for missions, the part of the mission that relates to geographical points or areas are simply expressed using the terminology defined in the ontologies for geodata. Similarly, exchange of data about asset geometries refer to the vocabulary used in IFC but with reference to its position in a global coordinate system.

To send and receive information which includes geodata, the internal representation of RDF just has to be transferred into one of the standardized textual formats and sent over a network as a text message.

B. Data stores

Both GML and IFC are primarily intended as data transfer formats, that allow data to be exported from one tool and imported into another. Therefore, it makes sense for them to use document-oriented formats like XML. However, in an operational SoS, partial data is repeatedly exchanged between CS, which need to store them internally and make operations on them.

Geospatial information is often very large, and efficient storage is imperative. There exist many RDF database solutions, so-called triple stores, both commercial and open source, that can be used for this purpose. Some of them support not only queries based on the triples, but also extended queries using the language GeoSPARQL [21], that allows geographical boundaries to be used as search criteria. In this way, it becomes possible to query for, e.g., all assets within a specific geographical area. The databases also allow insertion and removal of triple sets, so when a message containing RDF data on a textual format arrives, it is simply uploaded to the database and the new triples are added. Curation can be achieved through queries that identify outdated triples which can be removed.

C. Allocation

When a physical system is adapted to become a CS of a particular SoS, several different patterns exist for how those adaptations are allocated [5]:

- *Embedded*, where the added functionality is integrated in an on-board system of the CS;
- *Disconnected*, where the functionality provides an interface to an operator, which in turn controls the CS;
- *Remote*, where the functionality is allocated on a server, with a private connection to CS; and
- *Split*, which is a combination of embedded and remote.

The remote pattern is getting increasingly popular in many domains, not the least transportation, due to many benefits including security. For dealing with geodata, it also has apparent benefits. On a server, possibly cloud based, dealing with large data sets is normally not an issue, and communication between CS can take place over high-speed Internet connections with limited risks of interruption.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we started off by hypothesizing that Maier's fifth characteristic of SoS, which describes geographical distribution, implies that SoS always need to handle geospatial information. A number of application examples were put forward to justify this claim. After reviewing some fundamentals of geodata, it was discussed how it can be used in SoS. This led to the identification of some architectural strategies, such as the use of linked data to achieve interoperability, where the geodata is conforming to standards such as GML and IFC; include triple stores in the CS for storing linked data; and allocate the geodata to servers rather than embedded in the physical CS.

The architectural strategies should be taken as an initial guideline, and depending on specific application requirements, it can be motivated to deviate from them in a concrete case. However, we still believe that they serve as a good starting point for architecting SoS that are aware of geodata.

Our ongoing and future work aims at detailing these architectural strategies further, which is done as part of investigating larger and more complex application scenarios with a focus on the construction domain [4][5].

Another area which we will pursue is to extend the usage of the geodata based models into distributed digital twins, which can be used by the CS in order to plan their actions and support operational decision making.

REFERENCES

[1] M. W. Maier, “Architecting Principles for Systems-of-Systems,” INCOSE Int. Symp., vol. 6, no. 1, pp. 565–573, Jul. 1996.

[2] J. Axelsson, “A Refined Terminology on System-of-Systems Substructure and Constituent System States,” in IEEE Systems of Systems Conference, 2019, pp. 31–36.

[3] M. L. Butterfield, J. S. Pearlman and S. C. Vickroy, "A System-of-Systems Engineering GEOSS: Architectural Approach," in IEEE Systems Journal, vol. 2, no. 3, pp. 321-332, Sept. 2008.

[4] J. Axelsson, J. Fröberg, and P. Eriksson, “Towards a System-of-Systems for Improved Road Construction Efficiency Using Lean and Industry 4.0,” in 13th Annual Conference on System of Systems Engineering, 2018, pp. 576–582.

[5] J. Axelsson, J. Fröberg, and P. Eriksson, “Architecting systems-of-systems and their constituents: A case study applying Industry 4.0 in the construction domain,” Syst. Eng., vol. 22, no. 6, pp. 455–470, Nov. 2019.

[6] E. J. Christian, "GEOSS Architecture Principles and the GEOSS Clearinghouse," in IEEE Systems Journal, vol. 2, no. 3, pp. 333-337, Sept. 2008.

[7] S. J. S. Khalsa, S. Nativi and G. N. Geller, "The GEOSS Interoperability Process Pilot Project (IP3)," in IEEE Transactions on Geoscience and Remote Sensing, vol. 47, no. 1, pp. 80-91, Jan. 2009.

[8] J. A. Delgado, N. M. Short, D. P. Roberts and B. Vandenberg. “Big Data Analysis for Sustainable Agriculture on a Geospatial Cloud Framework,” in Froniers in Sustainable Food Systems, vol. 3, p. 54, 2019.

[9] U. Lenk, "Geospatial Geodetic Engineering Contributions to Specialty Engineering Activities or a dedicated Specialty Engineering Discipline on their own in System of Systems Engineering?," 13th Annual Conference on System of Systems Engineering, Paris, 2018, pp. 291-296.

[10] F. Chaves, J. Moßgraber, M. Schenk and U. Bügel, "Semantic Registries for Heterogeneous Sensor Networks: Bridging the Semantic Gap for Collaborative Crises Management," 24th International Workshop on Database and Expert Systems Applications, Los Alamitos, CA, 2013, pp. 118-122.

[11] A. H. Lee and A. Sekar, “A Distributed Software Architecture for a Rapid Geospatial Dataset Generation System,” Proc. 16th IASTED International Conference on Parallel and Distributed Computing and Systems, Cambridge, MA, 2004.

[12] G. Snyder, N. Gorkavyi, D. Lashlee and M. Lorenzo, “High fidelity terrain models and geospatial datasets for use in distributed test environments,” Proc. SPIE 5805, Enabling Technologies for Simulation Science IX, May, 2005.

[13] J. Axelsson, “An initial analysis of operational emergent properties in a platooning system-of-systems,” in 12th Annual IEEE International Systems Conference, 2018, pp. 1–8.

[14] ISO. Geographic information – Geography Markup Language (GML), Std. 19136:2007.

[15] ISO. Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries — Part 1: Data schema, Std. 16739-1:2018.

[16] C. Ingram, R. Payne, and J. Fitzgerald, “Architectural Modelling Patterns for Systems of Systems,” in INCOSE International Symposium, 2015, pp. 1177–1192.

[17] Y. Baek, J. Song, Y. Shin, S. Park, and D. Bae, “A Meta-Model for Representing System-of-Systems Ontologies,” in IEEE/ACM 6th International Workshop on Software Engineering for Systems-of-Systems (SESoS), 2018, pp. 1–7.

[18] J. Axelsson, “Experiences of Using Linked Data and Ontologies for Operational Data Sharing in Systems-of-Systems,” in IEEE Systems Conference, 2019, pp. 396–403.

[19] J. Axelsson, “Achieving System-of-Systems Interoperability Levels Using Linked Data and Ontologies,” to appear in Proc. 30th INCOSE International Symposium2, 2020.

[20] L. van den Brink, P. Janssen, W. Quak and J. Stoter, ”Linking spatial data: automated conversion of geo-information models and GML data to RDF,” International Journal of Spatial Data Infrastructures Research, no. 9, pp. 59-85, 2014.

[21] Open Geospatial Consortium. OGC GeoSPARQL – A Geographic Query Language for RDF Data. OGC Standard, 2012.