Jul 1st, 12:00 AM

Creating robust sensor networks - architecture and infrastructure

Gerald Schimak
Thomas Usländer
D. Havlik
R. M. Argent

Follow this and additional works at: https://scholarsarchive.byu.edu/iemssconference


This Event is brought to you for free and open access by the Civil and Environmental Engineering at BYU ScholarsArchive. It has been accepted for inclusion in International Congress on Environmental Modelling and Software by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.
Creating robust sensor networks - architecture and infrastructure

G. Schimak\textsuperscript{a}, Th. Usländer\textsuperscript{b}, D. Havlik\textsuperscript{a} and R. Argent\textsuperscript{c}

\textsuperscript{a} Austrian Research Center GmbH – ARC, A-2444 Seibersdorf, Austria (gerald.schimak@arcs.ac.at; denis.havlik@arcs.ac.at)
\textsuperscript{b} Fraunhofer IITB, Germany (uslaender@iitb.fraunhofer.de)
\textsuperscript{c} The University of Melbourne, Australia (r.argent@unimelb.edu.au)

Abstract: Significant unregulated development is occurring in sensor devices, protocols and sensor networks especially in the areas of air and water quality monitoring and management where the focus has been on human health and exposure impacts. Satellite technology is well established as a means of deploying sensors. However, for monitoring of air and water we lack the interconnectivity of both in-situ and remote sensing devices that link to interoperable information systems. One reason for this deficiency is the lack of a sensor network infrastructure generic enough to support this interoperability and interconnectivity.

This workshop will explore the architectural aspects of creating robust sensor networks including the aspects of the processing and visualisation of sensor-related information. The discussion will take into account topics like interoperability, self-organisation, network management and sensor service architectures. Candidate open geospatial architectures specified by European initiatives and research projects are taken as examples.

Keywords: Sensor Networks; Service Infrastructure, Architecture.

1. INTRODUCTION

Sensors provide some of the basic input data for environmental monitoring as well as for risk management of natural and man-made hazards. In this paper the word ‘sensors’ covers everything from remote sensing satellites, which provide valuable images of large regions, to single instruments providing highly-detailed point-based information installed on the Earth’s surface.

Because data from sensors play such an important part in improving understanding of our environment considerable investment has been made in the development and maintenance of highly-sophisticated sensor networks.

In spite of the ubiquitous need for information from sensor networks, the use of the data gleaned from such networks is hampered in many ways.

(1) Information about the presence and capabilities of sensor networks operating in a region is often difficult to obtain due to a lack of easily available and usable meta-information.

(2) Once sensor networks are identified their data is often not straightforward to access due to a lack of interoperability between data acquisition and dissemination systems.

(3) Incompatibilities between sensors and the systems that support them often limit the ability to transfer and share data. The current situation can lead to a lack of efficiency and under-use of available data which can severely compromise important applications such as risk mitigation.
2. CREATING ROBUST SENSOR NETWORKS - CHALLENGES

2.1 Generic (Architectural) Requirements

One major challenge when specifying and developing new sensor networks is to select the right architecture at the beginning of the project. The European research project ORCHESTRA (http://www.eu-orchestra.org) has provided a generic toolbox of utilities and a knowledge base for generic risk management applications upon which future-oriented service-oriented architectures can be developed.

The architectural principles that guided the specification of the Reference Model for the ORCHESTRA Architecture (see Usländer [RM-OA, 2007]) have to be adapted and extended for applications that involve service networks, especially for those that include the access to sensors and sensor-related information. We propose to consider these principles when discussing about robust sensor network architectures and infrastructures:

- **Rigorous Definition and Use of Concepts and Standards:** A Sensor Service Architecture should make rigorous use of proven concepts and standards in order to decrease the dependence on vendor-specific solutions, help ensure the openness of service network and support the evolutionary development process of an architecture.

- **Loosely Coupled Components:** The components involved in a service network should be loosely coupled, where loose coupling implies the use of mediation to permit existing components to be interconnected without changes to the basic system architecture.

- **Technology Independence:** A Sensor Service Architecture should be independent of technologies, their life-cycles and changes in sensor design as far as practically feasible. It should be possible to accommodate changes in technology (e.g. life-cycle of middleware technology) without changing the Sensor Service Architecture itself. The Sensor Service Architecture should be independent of specific implementation technologies (e.g. middleware, programming language, operating system). If possible, the Sensor Service Architecture should not be influenced by or deal with limitations of specific implementation technologies.

- **Evolutionary Development - Design for Change:** A Sensor Service Architecture should be adaptive and designed to evolve over time, i.e., it should be possible to develop and deploy the system in an evolutionary way. A Sensor Service Architecture should be able to cope with changes of user requirements, system requirements, organisational structures, information flows and information types in the source systems.

- **Component Architecture Independence:** A Sensor Service Architecture should be designed such that a service network and source systems (i.e. existing information systems, sensors and sensor networks) are architecturally decoupled. This means that a Sensor Service Architecture should not impose any architectural patterns on source systems for the purpose of having them collaborate in a Service Network, and no source system should impose architectural constraints on a Sensor Service Architecture.

- **Generic Infrastructure:** Services should be independent of the application domain. This means that Services should be designed in such a flexible and adaptable way so that the Services can be used across different thematic domains and in different organisational contexts. Also that the update of integrated
components (e.g. sensors, applications, systems, ontologies) have minimal impacts on the users of the Services.

In the following, the architectural approach of the European research project SANY (http://www.sany-ip.eu) is described as an example how to address such architectural principles in an open, geospatial service-oriented architecture.

### 2.2 SANY Sensor Service Architecture

The SANY project argues that the development of robust sensor networks encounters obstacles and challenges not only in the sensor domain but also in higher-level functional domains in order to further process sensor-related information and present it according to the end-user’s needs. These functional domains are illustrated in Figure 1 by Usländer [SANY, 2007].

![Figure 1: Functional Domains of the SANY Sensor Service Architecture](image)

The major component of the SANY Sensor Service Architecture is the ORCHESTRA Architecture which is described in Usländer [RM-OA, 2007] as an “open architecture that comprises the combined generic and platform-neutral specification of the information and service viewpoint as part of the ORCHESTRA Reference Model” (RM-OA). The SANY Sensor Service Architecture builds upon some essential features of the ORCHESTRA Architecture. These comprise:

- Rules and guidance on specifying information and service models in the context of international standards.
- Basic re-usable specification units for information models (e.g. pre-defined feature types) and service models (e.g. re-usable interfaces).
- A series of textual descriptions and formal specifications of generic ORCHESTRA Architecture Services that are application and technology independent.
The SANY project intends to re-use and apply these specific components in the context of the SANY Sensor Service Architecture to the extent that they enable the development of service networks based on sensor information and higher-level sensor information processing.

2.3 Functional domains

SANY services are designed to serve the needs of applications and users. Although there is no prescribed hierarchy of services, SANY services may be grouped into functional domains that match their design purpose. The SANY Sensor Service Architecture distinguishes between the following functional domains of SANY services (see Figure 1).

- **Sensor Domain**: SANY services in the sensor domain cope with the configuration and the management of individual sensor devices and their organization into sensor networks. Examples are services that support communication between the sensors themselves, e.g. a take-over service in case of an impending sensor battery failure. Services in this domain are abstractions from proprietary mechanisms and protocols of sensor networks.

- **Acquisition Domain**: SANY services in the acquisition domain deal with access to observations gathered by sensors. This includes other components in a SANY Sensor Network (e.g. a database or a model) that may offer their information in the same way (as observations) as sensors do. They explicitly deal with the gathering and management of information coming from the source system of type “sensor”. The information acquisition process may be organized in a hierarchical fashion by means of intermediate sensor service instances (e.g. using data loggers).

- **Mediation and Processing Domain**: SANY services in the mediation and processing domain are not specific to the sensor service architecture. They are specified independently of the fact that the information may stem from the source system of type “sensor”. They mediate access from the application domain (see below) to the underlying information sources. They provide generic or thematic processing capabilities such as fusion of information (from sensors and other information sources), the management of models and the access to model results. In addition, service support for the discovery of sensors, data and services, naming resolution or service chaining are grouped in the mediation and processing domain.

- **Application Domain**: Based on services within the acquisition and processing domain, SANY services in the application domain support the rendering of information in the form of maps, diagrams and reports directly to the end-user in the user domain.

- **User Domain**: The functionality of the user domain is to support the system interface to the end user. SANY will not specify dedicated services to support the user interface. However, when building real systems and applications, such functionality is essential. SANY considers this functionality to be specified in a dedicated implementation architecture that also may take proprietary components and products into account.
3. USER REQUIREMENTS

Beyond the architectural requirements of SANY - the workshop will also foster discussion concerning user requirements within various environmental domain(s). In the SANY project users from different environmental monitoring domains (air pollution risks, marine risks, geo-hazards) have identified their most important needs. An analysis of the various use cases has led to a set of requirements that have been grouped into functional blocks as illustrated in Figure 2 and listed below [Watson/SANY 2007].

![Figure 2: User requirement groups for sensor (service) networks](image)

Here, the term “system” stands for the entirety of a sensor service network including its architecture, its hardware and software components.

3.1 Sensor Network

- **Plug & measure** refers to the capability of being able to add a new sensor node without a manual re-configuration of the sensor network or sensor node.
- **Dependability** is required to provide data access and management services, in order to cope with the dynamic availability of possibly redundant sensor data sources, especially in the case of mobile sensors.
- **Sensor network management.** Of particular interest here is the localisation of sensor nodes, e.g. for the planning and management of their deployment or the configuration of the measurement frequency in order to optimise network and battery load.
- **Deployment of mobile ad hoc sensor clusters.** Especially in the case of biological and chemical hazards, the responsible administrative authority needs to monitor air pollution or water quality in order to quickly assess environmental risk. Unfortunately, appropriate sensors are often not available at the sites they are needed.
- **Self-validation of sensor nodes** with regard to its residual battery life and measurement capability (needed for re-calibration or maintenance) is important for the assessment of the node deployment and data quality.
- **Battery life-time optimisation** through selective data transmission is a useful function for assessing information via an interface to sensor nodes, often with self-diagnosis capability. It should support the ability to automatically select between alternate data transmission routes and/or choose the frequency of data transmissions, if the residual battery level of a sensor is too low.
3.2 Data and Information

- **Data sources** not only include sensors and databases of archived data, but also data obtained from a laboratory analysis of samples, or data entered manually by humans. Data sources may also be results of fusion services.

- **Spatial and temporal metadata.** The system should be capable of adding spatial metadata (sensor location information) and temporal metadata (sensor measurement time information) to any measurement data with/without this metadata from outset.

- **Rapid access to data.** The system should be capable of accessing data in sufficiently short time.

- **Data type.** The system should operate with different data types (e.g. fields, coverages) associated with measurement data like:
  - a single sensor measurement observation represented by a single value and unit located at a fixed location and detected at a particular time (time, date and year). For critical evaluation situations it has to be distinguished between a measurement result with a sensor internal clock time stamp or the nearly adequate particular time when the result has been received by the system.
  - time series of sensor measurement observations represented by a time series of triples \{value and unit, feature of interest, time\}. The time representation may be either absolute time (time, date and year) or a relative time representation (e.g. every day at midday starting from a certain date).

- **Geographic objects of several types.** The system should be capable of handling geographic objects of several types (e.g. the types specified by the ISO 19107 and ISO 19123 such as MultiPoint, MultiLineString, MultiPolygon, LineStrings, Polygons, MultiCurve, MultiSurface).

- **3D fields.** The system should be capable of operating with three-dimensional measurement data (e.g. components of vector fields etc.) of an observed property (e.g. wind speed, water current etc.)

- **Images (e.g. IR/UV, SLAR, INSAR).** As an example, such image data may be delivered by an earth observation sensor mounted on an aircraft or on a satellite with a certain spatial resolution.

- **Maps.** The system should be capable of handling a wide spectrum of maps (topography, roads, land usage etc).

3.3 Data Quality

- **Data quality.** To assess the quality (e.g. uncertainty) of measured data information about the quality of its measurement (e.g. accuracy, tolerance, resolution, drift) should be available. The system should be capable of accessing this sensor data quality assurance information.

- **Sensor level spatial and temporal uncertainty.** To ensure a previously defined user data quality target level, the system should provide details of data quality levels according to:
  - available sensor data sources which may vary by location,
  - estimated data statistical uncertainties derived from past data (e.g. model temperature predictions at a given location depends on the distance to available sensor values and on topological conditions e.g. located on mountain), and
  - the time period and frequency of measurement observations (limited e.g. by a certain availability frequency of EO satellite images of an area)

- **Certification of data and its propagation.** The system should be capable of supporting the process to formally certify the data quality, e.g. by instructing a certified laboratory per email with an investigation of new microbial sampling at a specific beach.
• **Associate data quality with the measurement context (validation).** The system should be capable of associating and storing data quality together with measurement context as meta-information. The system should be capable of visualizing this meta-information in its spatial and temporal context. This feature may be used as a tool for quality validation and as an aid to proper placement of sensors.

3.4 **Security**

- **Confidentiality of data** - UAA (User Management, Authentication, Authorisation). The system should provide several security related services:
  - Authentication Service (verification of user identities, support for multiple authentication mechanisms)
  - Authorisation Service (support of roles, authorisation for services authorisation on sensor level, etc)
  - User Management Service (storage and management of user profiles)
- **Monitoring Service** (checks the availability of services and sensors)
- **Billing, encryption, track data ownership.** The provision of sensor data requires a significant investment in equipment and in supporting services. These services normally cannot be offered free of charge, therefore a billing service is necessary. Another security topic is to protect measurement data against manipulation. In order to ensure data authenticity encryption may be necessary at different operational levels (e.g. during transmission, storage in databases etc.). It may be a security problem to allow certain users of the system to track sensor data over a period of time (e.g. a shipping company does not want competitors to know the exact routes of their ships).

3.5 **Processing and Fusion**

- **Interfaces for data processing services development.** The system should provide general data processing services e.g. merging data, extracting relevant data for reports. In general, the system should be capable of handling meta-information. This includes the storage of intermediate information in order that overall services (and service chains) can execute with acceptable speed.
- **Image analysis and feature extraction.** The system should be capable of processing images and extracting features (e.g. such as a road or a watercourse).
- **Different time steps, resolution.** The system should be capable of handling and adjusting different temporal and/or spatial resolutions of sensor data, depending on the sensors used and if values are missing.
- **Improving spatial resolution** if the spatial resolution of different sensor data is too heterogeneous or if values are missing.
- **Fusion of measurements.** The system should be capable of processing and merging measurements of the same phenomenon using different sensor platforms. The information about the source should be stored in meta-information. Data gathered at different times but generated by the same equipment should be processed using a fusion service.
- **Library of algorithms as ((statistical) processes) services.** The system should be capable of providing statistics libraries e.g. for interpolation and merging services.
- **Workflow.** The system should be capable of performing service chaining according to workflows. The output produced by a model service may be connected to the input of sequential process.
- **Visualisation.** In many situations several forms of visualisation are prescribed, e.g. maps, diagrams. To display them in a flexible manner styled layers should be capable of being output separately.
3.6 Events, Alerts and Alarms

- **Threshold surpasses detection.** The system should be capable of detecting threshold surpassing. For instance in the field of air quality measurement, pollution of the environment can be asserted if data values obtained from sensors reach defined threshold levels.

- **Alert algorithms.** Typically, an alert will result if a number of data observations for pre-selected parameters no longer fall within a prescribed range (data plots outside the desirable operating range of values). Alerts cause application level procedures and / or workflows to be executed in response to the event.

- **Interfaces for alarm management.** Alarms provide communication procedures associated with the current environment to warn about an imminent hazard and to initiate emergency procedures (such as evacuation).

3.7 Tracing

- **Tracing requirements** address the need to document what information sources were used as a basis for the decisions taken and the decision making process itself. The purpose is to be able to provide a retrospective justification for decisions made, these decisions may later be contested by affected parties to the decision.

3.8 Models

- The system should provide in general
  - a model service catalogue, where all available services may be selected by the user
  - an execution management service that handles input/output and, (optionally if expedient for system performance) data by reference
  - the capability of using the output of models and fusion algorithms as sensor values without changing the system.

In addition to basic support for models and model wrapping (the process of integrating the models into the SANY service network), model-specific functionality is required in the following areas:

  - Gather applicable source impact models
  - Domain skills compilation
  - Library of models as processes
  - Library of geo-statistical analysis as processes
  - Predictive models for adaptive sampling
  - Spill advection & dispersion modelling
  - Forecasting risks (water quality, bathing water, beach closed)
  - Model long-term degradation of ecosystems
  - Improved soil models

3.9 Decision Support

- **Providing supporting information for decisions.** The types of decisions to be taken will dictate the need for modelling, data fusion and visualization of model results and other information. Auxiliary methods may be required to compute utility functions and to undertake multi-criteria analyses.

3.10 System User Interface

- **User registration.** The system should offer the user a dialog window to login normally with user name and password. When entering the password the input window should not display the characters in a readable form.
• **Message to the user.** The system should be capable of sending a message to the user in a separate window. This window must be visible to the user (e.g. it pops into the foreground).

• **Sending form(s) to the user.** The system should be capable of sending a form or several forms to the user to and to encourage them fill them out with the required information.

• **Sending message that authorisation failed.** If an authorisation has failed (e.g. due to insufficient access rights), the user should be informed by a message.

• **Sending message that authorisation failed and waits for a new input.** If an authorisation has failed (e.g. due to a wrong password), the user should be informed by a message and is requested to repeat the login procedure.

4. **CURRENT EXAMPLES FROM ONGOING PROJECTS**

4.1 **ORCHESTRA and SANY**

The European Commission (EC) is funding a number of Integrated Projects within the Sixth Framework Programme concerned with improving the accessibility of data and services for risk management. Two of these projects, ‘Open Architecture and Spatial Data Infrastructure for Risk Management’ (ORCHESTRA, http://www.eu-orchestra.org/) and ‘Sensors Anywhere’ (SANY, http://sany-ip.eu/), were introduced earlier and some pertinent features of these projects that relate to sensor networks were briefly discussed.

These projects have developed an open distributed information technology architecture and have implemented web services for the access and use of the data emanating, for example, from sensor networks. These developments are based on existing data and service specifications proposed by international standardisation organisations such as ISO and the Open Geospatial Consortium (OGC).

The projects seek to develop the principal features of the EC INSPIRE initiative (http://inspire.jrc.it), which was launched in 2001 and began this year to implement elements relevant to risk management.

Thanks to the open nature of the architecture and the services being developed within these projects, the technology can be implemented by any interested party and can be accessed via the web by all potential users of the software. The architecture is based on a service-oriented approach that makes use of Internet-based applications (web services) whose inputs and outputs conform to recognized standards.

The benefit of this philosophy is that it is expected to favour the emergence of an operational market for risk management services in Europe. It eliminates the need to replace or radically alter the multitude of already operational IT systems in Europe, drastically lowering costs for users [Klopfer/Kannellopoulos 2008]. Furthermore, it allows users and stakeholders to achieve interoperability while using the system most adequate to their needs, budgets, culture etc. (i.e. it has flexibility).

4.2 **Architectural Requirements put on Sensor Networks from initiatives like GMES**

The ‘Global Monitoring for Environment and Security’ (GMES) represents a concerted effort to bring data and information providers together with users, so they can better understand each other and make environmental and security-related information available to the people who need it through enhanced or new services.

For example, the GMES initiative (http://www.gmes.info) as illustrated in Figure 3 below focuses on four major components:
The final report for the GMES initial period [GMES FR, 2004] provides the following key architectural requirements on pages 25-26:

“For GMES to become a success, the architecture needs to facilitate the integration of standalone data and information elements. It should allow to the selection and aggregation of information from heterogeneous sources and should provide the capability to translate data and information between the various sources in real time. This applies as much to the incorporation of socio-economic data and information, as well as products derived from the space and in situ observing networks.

GMES must therefore provide a structured framework for data integration and information management, i.e., a European shared information capacity. The following key architectural and user-oriented requirements will therefore drive the implementation of GMES:

• Openness, based on agreed open standards, facilitating seamless communication and interoperability, i.e. the ability of different devices or systems (usually from different vendors) to work together, as well as enabling user service autonomy;

• Federated architecture, enabling systems to grow and evolve;

• Simplicity of architecture (e.g. modularity of components), to break the complexity barrier, systems must be made easier to design, administer and use;

• Self-configuration, programmability, scalability (e.g. to handle various levels of operational load and external conditions);

• Dependability, i.e. the system's resilience to security threats or breakdown;

• User-friendliness of services and interfaces, e.g. in the handling of user request services, access control, workflow management, delivery management, visualisation, data extraction (e.g. “multilinguality”), multiuser sessions, administration;

• Data security, protection of provider and user data against alteration, theft and misuse;

• Quality of service;

• Ubiquity of access, including global reach.”

In a overview paper prepared by DG [GMES, 2005] on Data Integration and Information Management Capacity, provisional and important functional requirements are listed. The paper suggests that the approach needed to create an efficient data integration and information management component is to use a “system of systems” design as explained below:
"The GMES information and services infrastructure goes well beyond a simple exchange of data. It addresses the need to integrate business and workflow processes which span across boundaries, be they political, administrative or thematic. It addresses the main issues of heterogeneity and fragmentation of the information: "heterogeneity" meaning that the same information is often represented differently, "fragmentation" meaning that needed information is spread over multiple locations.

The approach is to seamlessly integrate existing systems into a "system of systems" perspective. System of systems is an emerging design and development method of complex systems build from large scale component systems. The subsystems that comprise the system of systems are generally built by different organisations, having different goals, are very often built to different quality standards, and are managed independently.

Reusing existing legacy systems in a dependable fashion without the need for extensive re-engineering is a key problem currently faced by industry. System of systems can be seen as new systems linking data, services and workflows to produce new data, new services together with metadata (information about the information products generated). This approach, which was a research topic in the last decade, is now becoming sufficiently mature. In the short term, robust system architectures will be developed and tested allowing the exchange of data and services that are well identified: This is the syntactic interoperability phase. In the longer term, we expect to achieve significant semantic interoperability, which will allow cross-system search for data and services. It could be dubbed the "GMES Google" since it will work for data and services in the way web search engine works for web pages.

A number of non technical obstacles should however be addressed to eventually reduce the complexity of the implementation, such as trusted electronic billing principles between data providers and services providers to partly overcome the hurdles created by different data policies. Mechanism should also be agreed upon to manage access privilege across institutional borders in a practical, transparent and secure way.

This approach will allow sharing and efficient management of information that is consistent across organisations, borders and thematic domains such as from land use and mapping to risk management and security."

The suggested system of systems approach is still regarded today as the best way forward as indicated in the following excerpt of the report of Joint Operability Workshop held in April 2007:

"There was broad agreement that any proposed solution for a single information space for the environment must allow for multiple architectures and hence be a "system of systems". Further comments on this topic included:

- Hierarchical architecture is required to cope with services at various levels: EO ground segment services, GMES core services, GMES downstream services etc.
- SOA is well suited for deployment of system of systems.
- Significant investments have been made already by ESA, ECMWF\(^2\) etc. Therefore it is unrealistic that everybody will adopt a single architecture. This leads to a system of interoperable systems which may be based (internally) on heterogeneous technologies.

---

1 In the SANY Service Architecture therefore called “meta-information”.
2 European Centre for Medium range Weather Forecasting
An exhaustive list of use cases is unfeasible – the number of possibilities is too large and is changing rapidly. Thus, a flexible architecture is needed which also leads to the concept of a system of systems. The fully top-down approach and pan-European architecture that was anticipated as recently as five years ago appear now to not be feasible. The concept is being replaced by pan-European interoperability in a system of systems-like architecture.

Many systems are legacy systems in this field as satellites have a relatively short life and scientific advances in EO are rapid. It is totally unfeasible to convert all existing data so they all have the same data and metadata formats. Any viable solution must be based on fully utilising legacy systems. Harmonised, “on the fly” access to legacy datasets is required.

Access to data from heterogeneous missions must be seamless and transparent.”

As recommended in the following extract of the overview paper [GMES, 2005] on Data Integration and Information Management Capacity, the technology of choice to be used to implement the GMES architecture is SOA: “The underlying architecture for ensuring interoperable GMES services is the Service Oriented Architecture (SOA)”.

Modern system architecture for building environmental sensor networks are based on a less tightly coupled collaboration of distributed services. For this, the term SOA is becoming widely used, but there is little agreement in the manner by which it is used. The World Wide Web Consortium (W3C) for example refers to SOA as ‘A set of components which can be invoked, and whose interface descriptions can be published and discovered’. This broad definition highlights a key aspect of an SOA: components (e.g. functionalities) can be discovered and invoked dynamically. This type of architecture is different from “hard wiring” of business processes, which make changes according to the recognition of new circumstances very difficult. SOAs are inherently more flexible and adaptable than most alternate approaches.

A service-oriented architecture is also proposed for the following reasons:

- A SOA can represent business functionality remaining implementation- and vendor-neutral while providing standards-based shared services.
- Existing legacy systems can be extended with service interfaces, and, in that way, become part of a SOA.
- By using inherent service discovery and access flexibility (see above), a SOA enables GMES/INSPIRE service providers to be more agile and to respond more quickly to changing business opportunities and changing requirements.
- A set of common generic services can provide standard, domain-independent functionality (for discovery, search, navigation, data access, authorisation etc.) which only needs to be implemented once.
- Sharing of services — no need to “re-invent the wheel”
- Loose coupling — ability to update applications with minimal effect on services that invoke them
- Location transparency — ability to re-host applications with minimal effect on services that invoke them

GMES applications are built on top of a joint, adopted infrastructure that will be interoperable and much easier to integrate into multi-purpose, cross-application operations.”

The adoption of SOA architecture based on open standards for its services is a good path to take in order to provide flexibility, stability, and durability while preventing vendor lock-in. Example of open standards are the OGC standards for Web services.
The relevance of these two major European initiatives, INSPIRE and GMES, for the set-up and design of robust sensor networks and higher-level service infrastructures will be a topic for the discussion at the workshop, too.

ACKNOWLEDGEMENTS

The authors wish to thank the SANY and ORCHESTRA Consortium for their valuable input. Special thanks to the architectural working group (SP2) of SANY for providing most of the requirements input listed above.

SANY and ORCHESTRA are Integrated Projects co-funded by the Information Society and Media DG of the European Commission within the RTD activities of the Thematic Priority Information Society Technologies.

REFERENCES

GMES, GMES reflection paper on Data Integration and Information Management Capacity, DG-INFSO, Draft 6, July 2005
Usländer, T. and SANY-Consortium, SANY-Deliverable, D2.3.1 Sensor Service Architecture V0 (D2.3.1 Sensor Service Architecture V0 _Doc. V1.0_.pdf), 2007 (http://sany-ip.eu/publications/1957).