FFI Forsvarets forskningsinstitutt

Towards operational agility using service oriented integration of prototype and legacy systems

Authors:

Frank T. Johnsen, Trude H. Bloebaum, Ketil Lund, and Espen Skjervold

Norwegian Defence Research Establishment (FFI)

Outline

- Motivation / SOA – Web services
- Experiment
  - Components
  - Information infrastructure
  - Lessons learned
- Conclusion / future work

Motivation

- The NATO Network Enabled Capability (NNEC) Feasibility Study
  - Presents a discussion of technology, focusing on the needs of future interoperable military communications.
  - Identifies the Service Oriented Architecture (SOA) concept and Web services technology as the key enablers for NNEC.

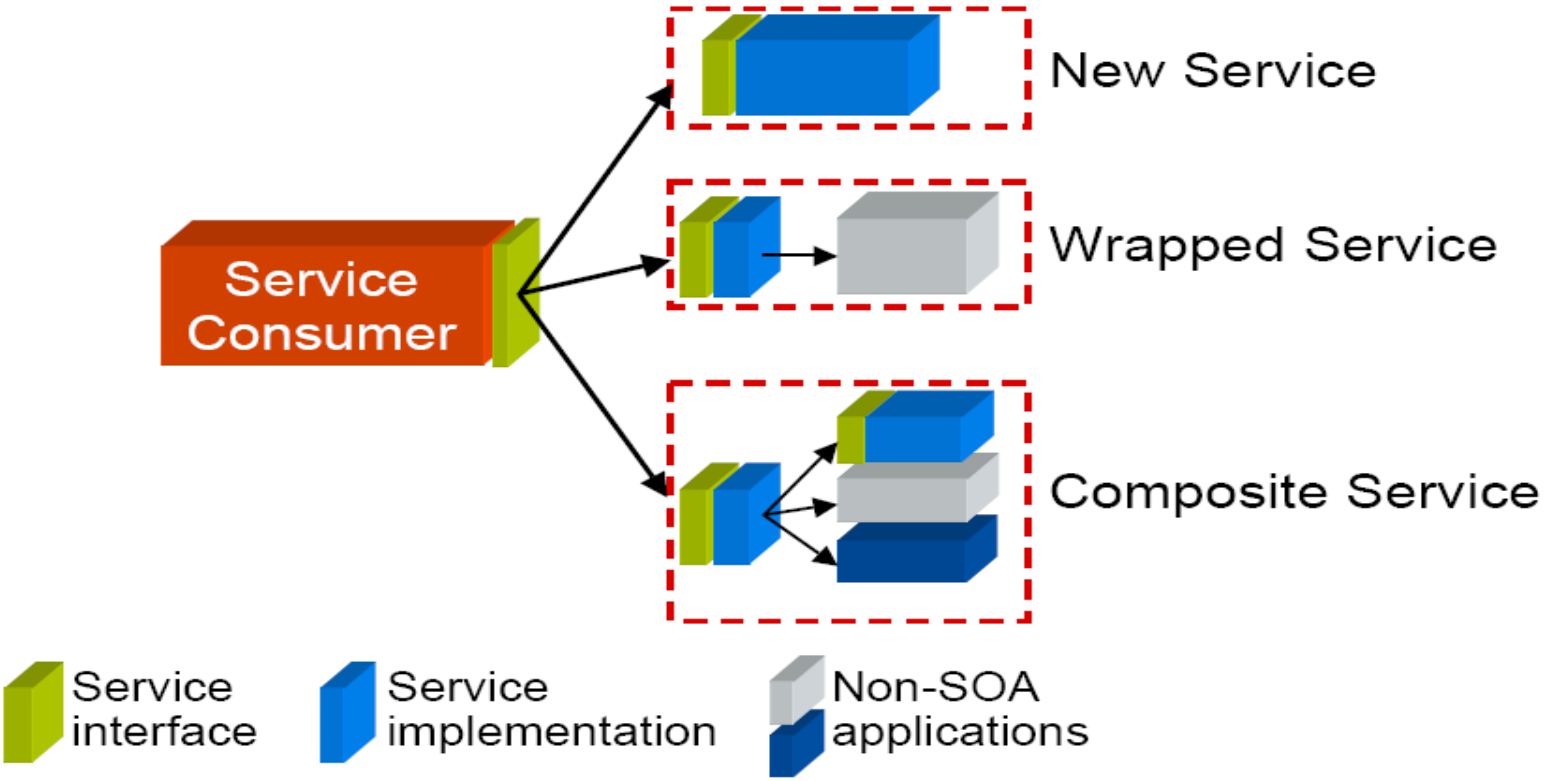
- Our focus
  - Interconnecting prototype solutions and legacy systems.
  - Using a combination of established standards and bespoke solutions.

- Our goal
  - To give a practical demonstration of the benefits of service oriented system integration.

Building services

A service is a mechanism to enable access to resources, using a prescribed interface and is exercised as specified by the service description.

(OASIS: Reference Model for Service Oriented Architecture 1.0).



Gartner Research "Service-Oriented Architecture Under the Magnifying Glass" by Yefim Natis, Application Integration & Web Service, Summit 2005, April 18-20, 2005

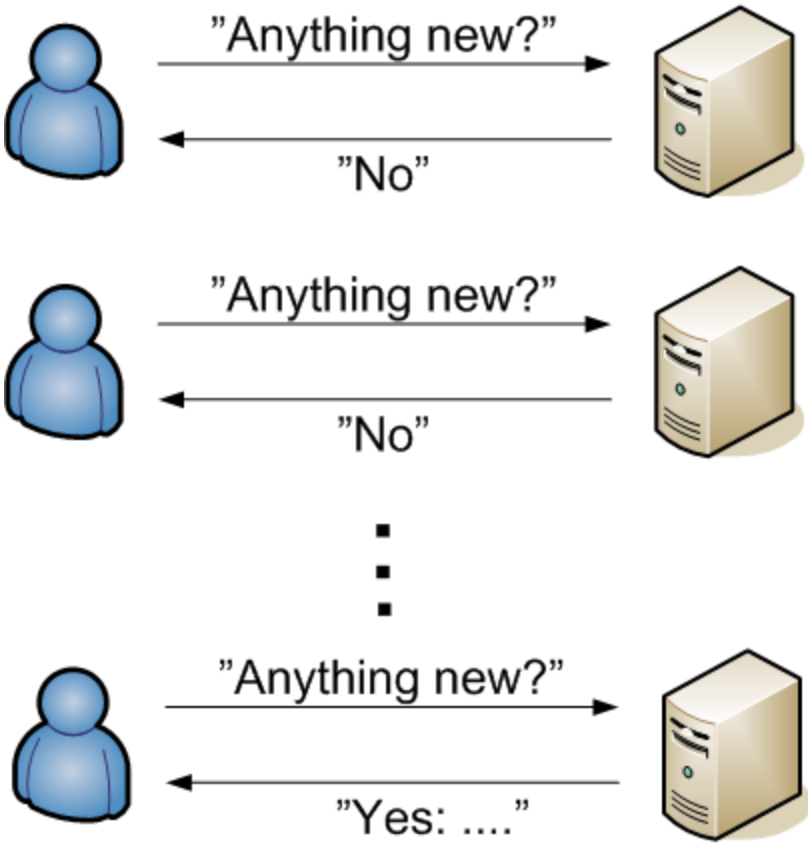
Gartner

Web services: Implementing a SOA

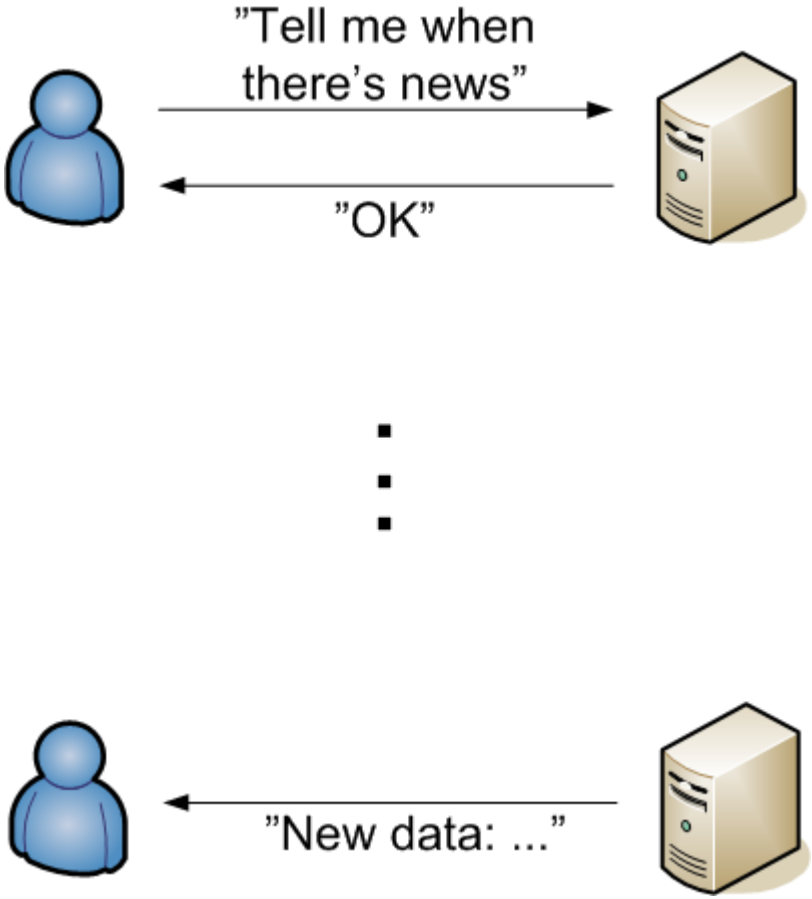
- The W3C definition of a Web service:
  - “A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.”
- Thus, we see that there are three central standards:
  - WSDL (service interface)
  - SOAP (messaging)
  - XML (encoding)
- Web services support
  - Synchronous request/response communication.
  - Asynchronous publish/subscribe communication.

Communication paradigms

Request/response



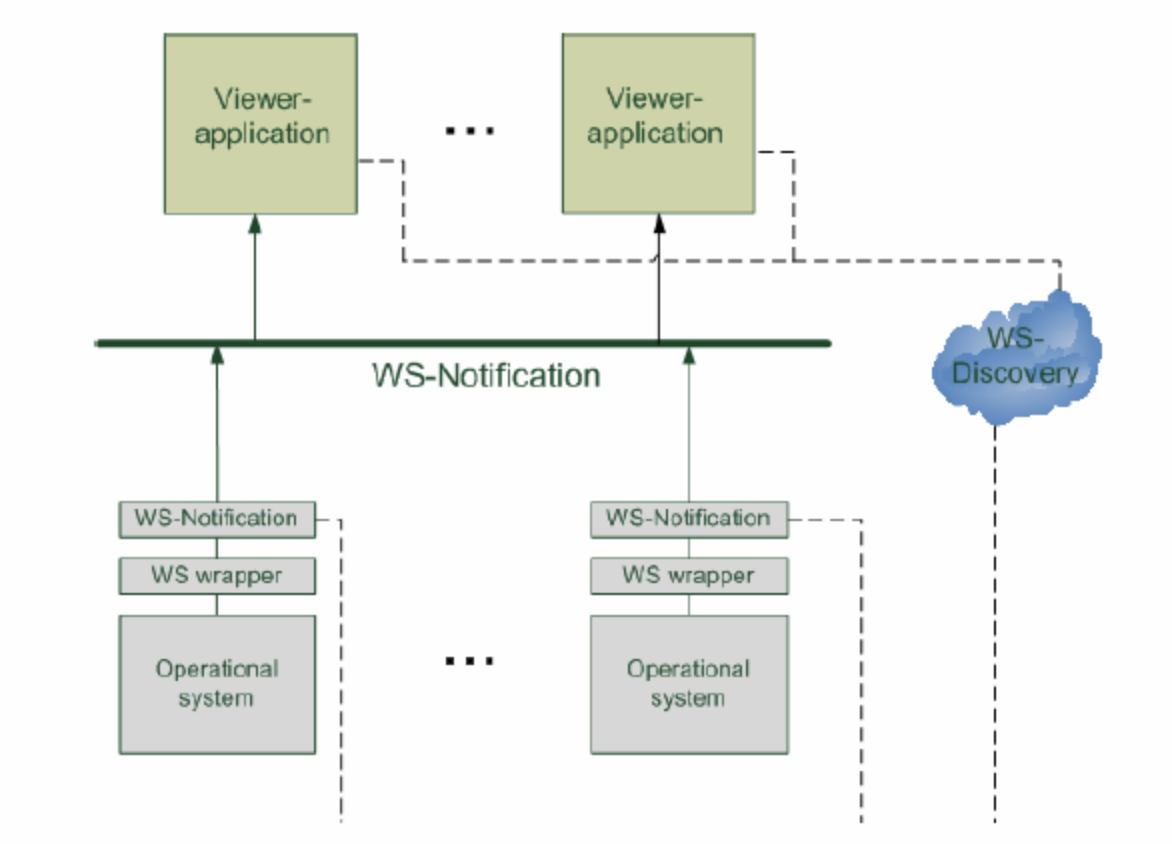
Publish/subscribe



Experiment components

- Two main types of services involved:
  - Core services (infrastructure components)
    - Service discovery, publish/subscribe, mediation, collaboration, etc.
  - Functional services (information services)
    - The functional services were
      - NFFI
      - incident reporting

Experiment components



- We leverage existing standards: WS-Notification and WS-Discovery.

Experiment component: WS-Notification

- WS-Notification
  - OASIS standard.
  - There are three parts in the specification:
    - WS-BaseNotification, WS-BrokeredNotification, and WS-Topics.
    - The specifications standardize the syntax and semantics of the message exchanges that establish and manage subscriptions.

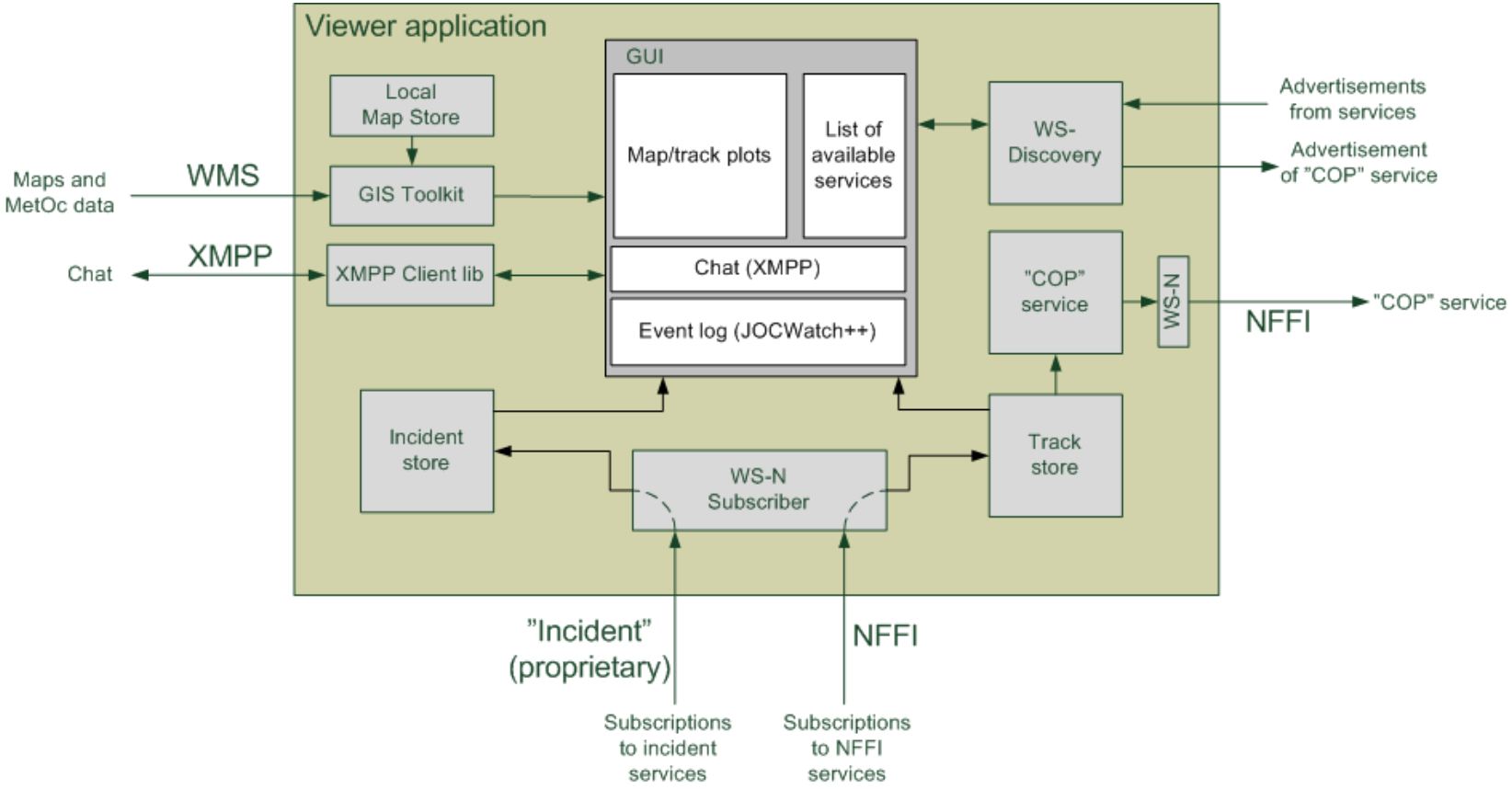
- We used a freely available implementation of WS-Notification
  - http://www.extreme.indiana.edu/xgws/messenger/

Experiment component: WS-Discovery

- WS-Discovery
  - OASIS standard.
  - Builds on the SOAP-over-UDP standard to provide decentralized service discovery.
  - Better suited to dynamic networks than the registries, but requires IP multicast support.

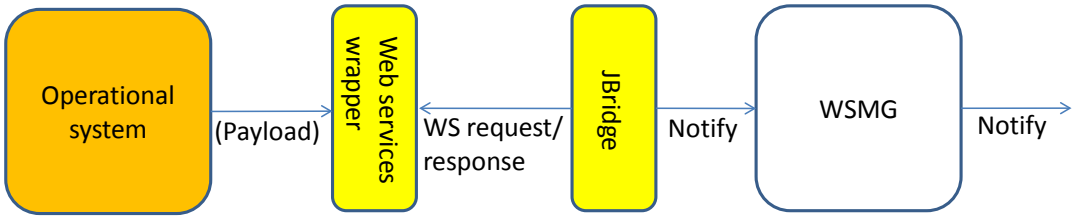
- WS-Discovery generates a lot of network traffic, but the overhead can be reduced by using compression.
  - The implementation is available from
    - http://code.google.com/p/java-ws-discovery/
  - In the experiment we used WS-Discovery with EXI compression
    - Described in [Johnsen and Hafsøe, “Adapting WS-Discovery for use in tactical networks”, ICCRTS 2011].

Experiment component: SOA viewer



Experiment component: Bespoke solutions

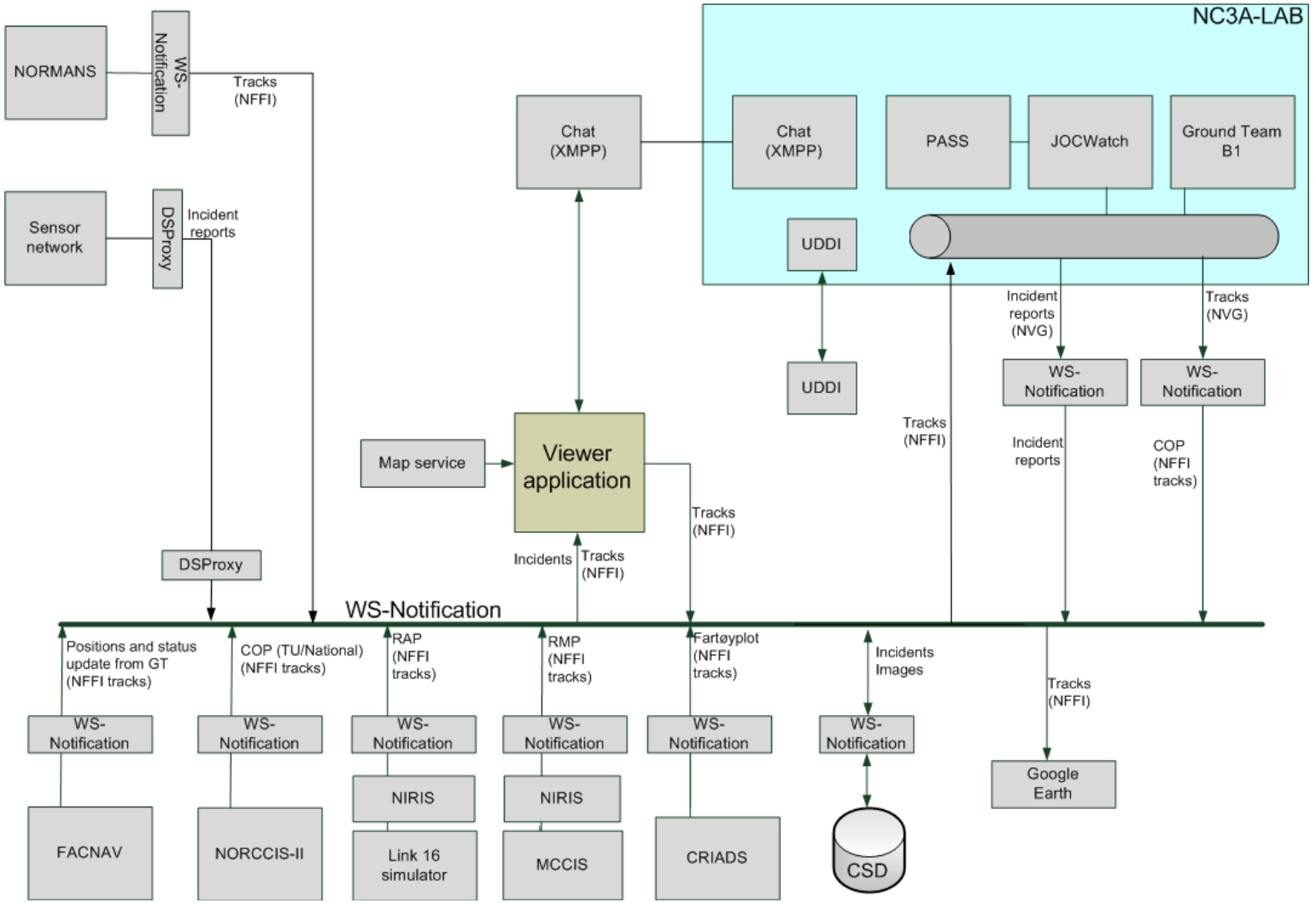
- Delay and Disruption tolerant SOAP Proxy (DSProxy)
  - Developed in-house, provides Web services support across disadvantaged grids.
  - In the experiment we used the DSProxy to run Web services across the MRR VHF radio.
- JBridge
  - Developed in-house for publish/subscribe-enabling request/response services.



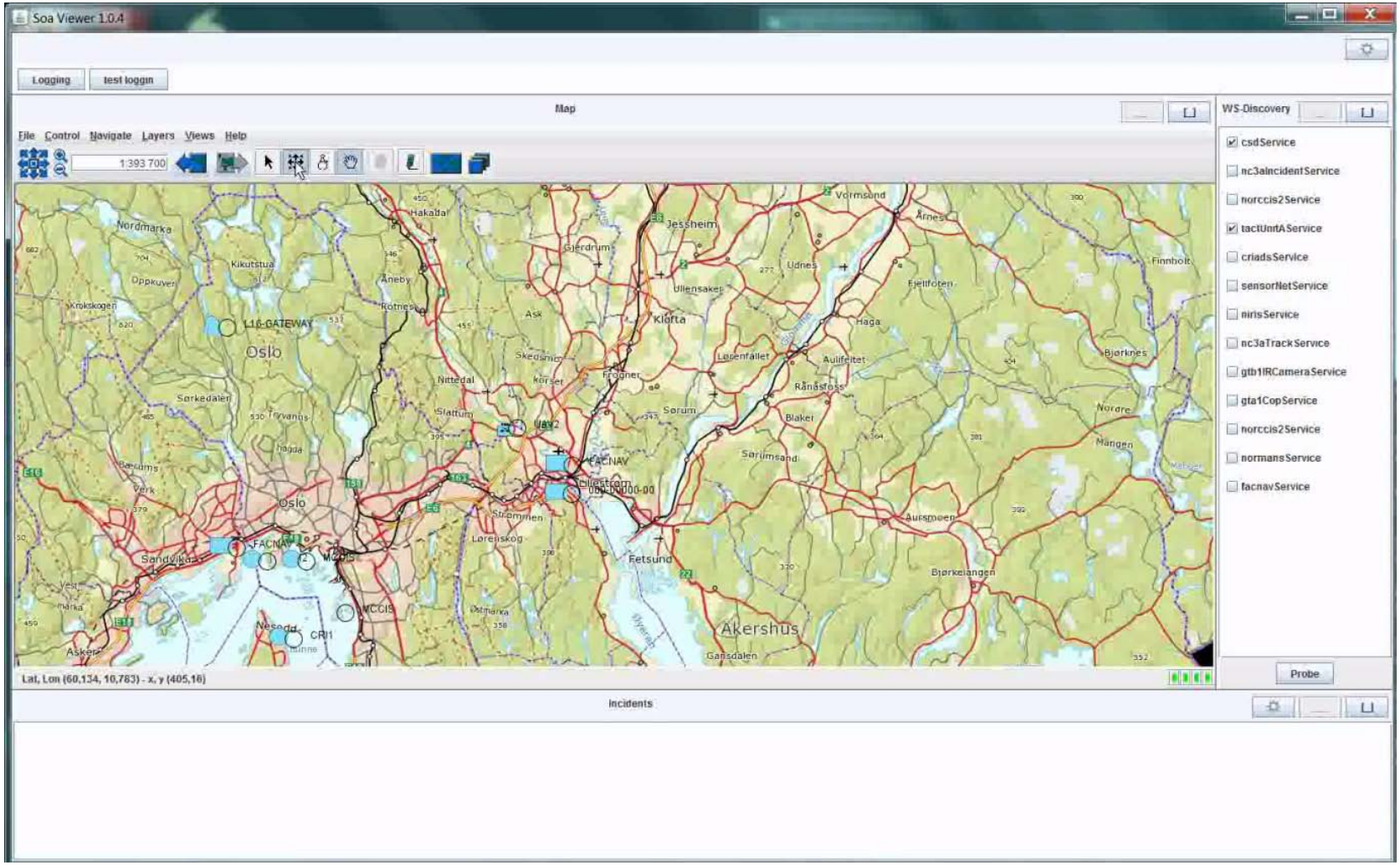
Experiment component: Systems

- Experimental
  - Wireless sensor network
  - Norwegian Modular Network Soldier (NORMANS)
- Operational
  - CRIADS
  - NORCCIS II
  - NIRIS
    - TDL
    - MCCIS
  - CSD
  - FACNAV

Experiment information infrastructure



SOA viewer screenshot



Lessons learned: WS-Notification

- WSMG
  - Academic project, support and further development is uncertain.
  - WSMG displayed shortcomings
    - Added proprietary tags in the notification messages, requires adaption in the receivers.
    - No support for XML in the payload.
    - Unable to deliver any notifications if one subscriber is unavailable.
  - We advise against using it even in a test environment if you need to disseminate XML payloads.
    - Workaround – Base64 encode the payload, but that requires adaption in the receivers.

Lessons learned: Wrappers, JBridge and DSProxy

- Wrappers
  - The principle of wrapping existing software with a Web service front-end works very well.
  - The complexity of the wrapper is dependent on the type of connection it has to the actual operational system.
- JBridge
  - A viable approach to publish/subscribe-enable request/response services.
  - This component can be eliminated by integrated publishing functionality directly into the Web services wrapper.
- DSProxy
  - MRR represents a challenge when using Web services due to low bandwidth and high delays.
  - The DSProxy successfully disseminated information from the sensor network across the MRR link.

Lessons learned: Service discovery

- WS-Discovery
  - WS-Discovery works fine over medium bandwidth radios like the Kongsberg WM600, while it is not advisable, even with compression, on radios with very low bandwidth, such as the MRR.
  - Provides run-time service discovery and requires IP multicast support.
  - A general problem is that seen from a WS-Discovery point of view, all publish/subscribe (WS-Notification) services are equal.
    - Consequently, the information about which topics each service published on had to be distributed beforehand.
- It is clear that when using publish/subscribe for information dissemination, it is necessary to focus on how to search for and discover available topics.
  - Not covered by the standards, requires further research.

Lessons learned: Format translation

- In the experiment we had to handle a number of different formats coming from the different operational systems.
- A challenge when integrating information from different formats is loss of information.
  - For instance, when translating from NVG to NFFI information is lost.
- Another challenge is lack of information.
  - When translating from OTH gold to NFFI, it is necessary to add information, since OTH gold is a much simpler format.

Conclusion

- The experiment was executed at FFI in June 2011 in cooperation with the NC3A.
  - As a demonstration of the benefits of SOA we consider the effort a success.
- Recent work
  - NATO RTO/IST-090 experiment in autumn 2011
    - Subset of this experiment, substituting WSMG for Apache ServiceMix
- Ongoing work
  - CoNSIS June 2012
    - Experiments with Web services over IPv6.