2008 NSDI Cooperative Agreement Program Category

Category 2: Best Practices in Geospatial Service-Oriented Architecture (SOA)

The Polis Center
We bring things into perspective.™

Geospatial Service-Oriented Architecture for Flood Inundation Mapping and Hazard Assessment

Final Report

Presented to:

NSDI

September 29, 2009

NSDI Cooperative Agreements Program
Category 2: Best Practices in Geospatial Service-Oriented Architecture (SOA)

Final Report

Date: September 29, 2009

Agreement Number: 08HQAG0026

Project Title: Geospatial Service-Oriented Architecture for Flood Inundation Mapping and Hazard Assessment Final Report

Organization: Indiana University
620 Union Drive, Room 618
Indianapolis, IN 46202-5167
http://www.iupui.edu

Principal Investigator: David J Bodenhamer

V. 317-274-2455
F. 317-278-1830
Intu100@iupui.edu

Collaborating Organizations: Indiana Department of Homeland Security (IDHS), National Weather Service Central Region (NWS), the USGS Indiana Water Science Center, the Indiana University Community Grids Lab, and The Polis Center IUPUI.

Executive Summary

The primary objective of this project is to create a composite application using multiple Web services to generate flood inundation geospatial grids and estimate flood-related property losses. These Web services are self-contained, well-defined functions encapsulated as reusable software components. This application process generates flood maps for FEMA's building loss, analysis as well serves flood maps near-real-time over the Web. The secondary outcome of this project is to document best practices in geospatial service-oriented architecture (SOA).

Flood inundation grids are generated by using the U.S. Geological Survey's (USGS) Multi-Dimensional Surface-Water Modeling System (MD_SWMS) tool, which provides simulation programs for a variety of environmental and hydraulic models. One such hydraulic model that was used in this project is FaSTMECH, a two-dimensional hydraulic model. To achieve the objectives of the project, we developed an SOA Wrapper to the FaSTMECH model, which allowed us to invoke FaSTMECH processes using Web Service Description Language (WSDL) Application Programming Interfaces (APIs) and SOAP-based XML messages.

Building loss estimations are performed using a loss estimation calculation Web service developed for this project. This service overlays the flood inundation grids, generated from the FaSTMECH model, with parcel-level property data and calculates percentage damage to properties using Federal Insurance Agency (FIA) property damage curves. The loss estimation calculation service is compound; it uses two other data services—a parcels centroids service and a property assessment value service.

The execution of the overall flood inundation and loss estimation business process was performed by a custom execution application. This application chains Web services and executes processes asynchronously. We also tested service chaining using a Business Process Execution Language (BPEL) based module.

Finally, for controlling the process and visualizing process outputs we developed a graphic user interface (GUI) using Adobe Flex framework.

We have successfully developed, tested, and deployed these services and the process application.

Project Narrative

The following is a list of project activities we performed.

- **Use case development:** We met with USGS, NWS, and IDHS stakeholders to elicit requirements and create use cases. We developed two use cases: 1) Response planning for after NWS issues a flood watch or warning; 2) Development of mitigation strategies for "what if" scenarios for reducing flood risk.
- **Web service development:** We developed the following Web services.
  - **NWS Real-Time River Data Monitoring Service:** The real-time river data monitoring service constantly monitors the NWS real-time forecast of the Nora station and begins recording both the flow gauge and the river stage data up to six days into the future once a pre-defined flood condition is met.
  - **CGNS Input Service:** During a flood study, the CGNS input process service infuses such information as initial conditions into the pre-calibrated regional model represented by a CGNS file. The updated CGNS file is in turn fed to the flood simulation service as the input to perform the FaSTMECH simulation, which stores computation results by once again updating the given CGNS file.
  - **FaSTMECH flood inundation simulation execution Web service:** Currently with this service we have the capability of providing a pre-defined CGNS file to simulate inundation and generate flood depth grids and velocity vectors. To generate predefined CGNS files, we use an interactive standalone application. We are still working on a method for automating the creation of the CGNS file to simulate the model with variable model parameters in a service process. We also will be developing methods to extract flood depth grid data and velocity vectors from CGNS files.
  - **CGNS Output Service:** The CGNS output process service parses the FaSTMECH simulation results and generates curvilinear grids in ASCII files.
  - **Parcel building points and property value data services:** We have developed these two support services for the Marion County geographic area in Indiana.
  - **Property loss estimation Web service:** This service overlays parcel point data on the flood depth grid file from FaSTMECH Web service. It also selects the flooded parcels and calculates the percentage property damage using the FIA depth-damage functions. Depth-damage functions are plots of floodwater depth versus percent damage, plotted for a variety of building types and occupancies. The percentage property damage is then used with property assessment data to calculate the total losses to the property.
- **Business process development:** Web services may be chained and coordinated through aggregating clients. These clients may be either formal (workflow engines) or informal (scripts and mash-ups). We evaluated both approaches; and then developed a custom .NET based

service chaining and process execution application and an XML based BPEL module to orchestrate a composite application to create the workflows of the two use cases.

- **Graphic User Interface:** We have developed a graphic user interface (GUI) using Adobe Flex frame work using Flex Builder IDE. This GUI allows users to execute the modeling and data processing workflow and also allows Mashups with any Web 2.0 compliant geospatial Web services such as Marion County parcel maps and demographic maps from Social Assets and Vulnerabilities Indicators (SAVI – http://www.savi.org) community information system.
- **Virtualization deployment:** The FaSTMECH, CGNS input and output, and river monitoring services are all deployed on the Indiana University gateway hosting service. This service provides virtual machine hosting, real-time redundancy and failover of hardware between the Bloomington and Indianapolis campuses, and mounting of the Data Capacitor (a very large wide-area research network).
- **Testing:** We have tested individual services and operation APIs. We developed unit tests for new services developed in this project to validate that they were consuming and producing correct data sets. These were used to validate the deployment of developed services into the production pipeline on the gateway hosting service. We have tested the complete workflows for the two use cases, and we also validated our pipeline against a historical flood.

Figure 1: Collaborations

Figure 1 depicts the collaboration partners, which include state, federal, and regional groups.

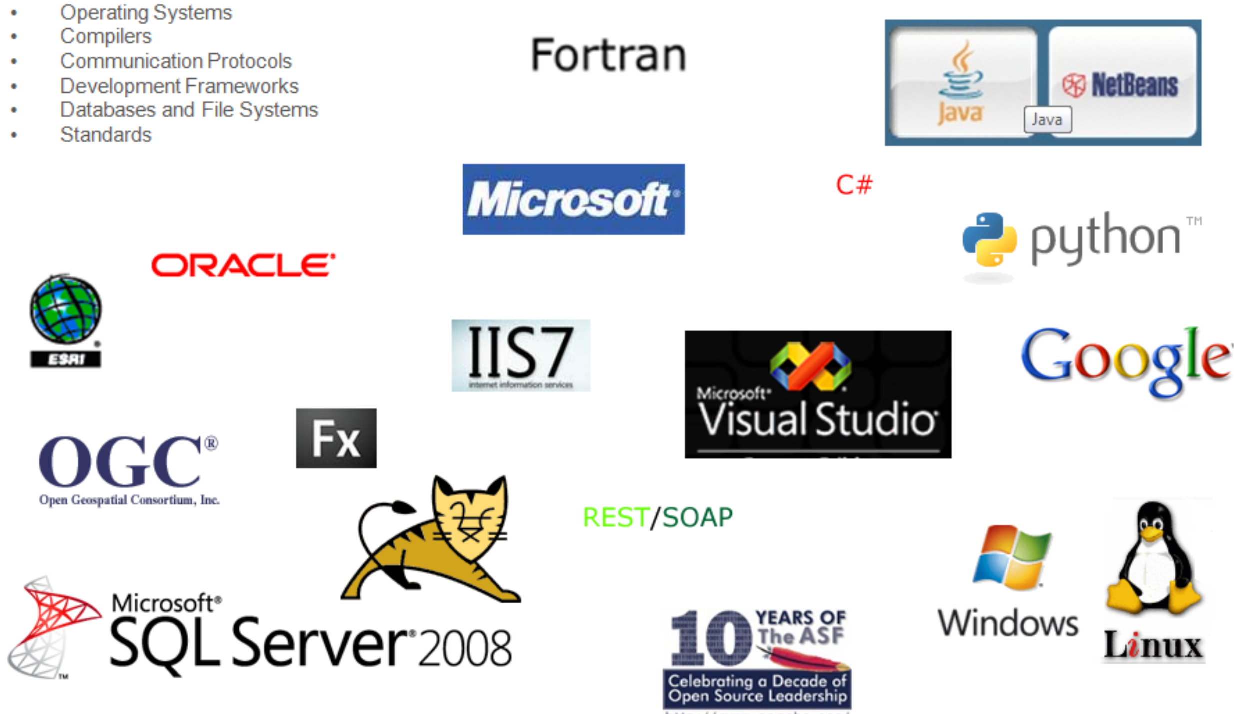

Figure 2: Heterogeneous system: Tools that were used to build various components of our open network service approach. Encapsulating services behind well-defined interfaces enables this diversity.

Figure 2 depicts the heterogeneous nature of the service solution. The current solution uses multiple operating systems, program code compilers, wire communication protocols (REST and WSDL/SOAP style services), development frameworks, databases, files systems, and standards.

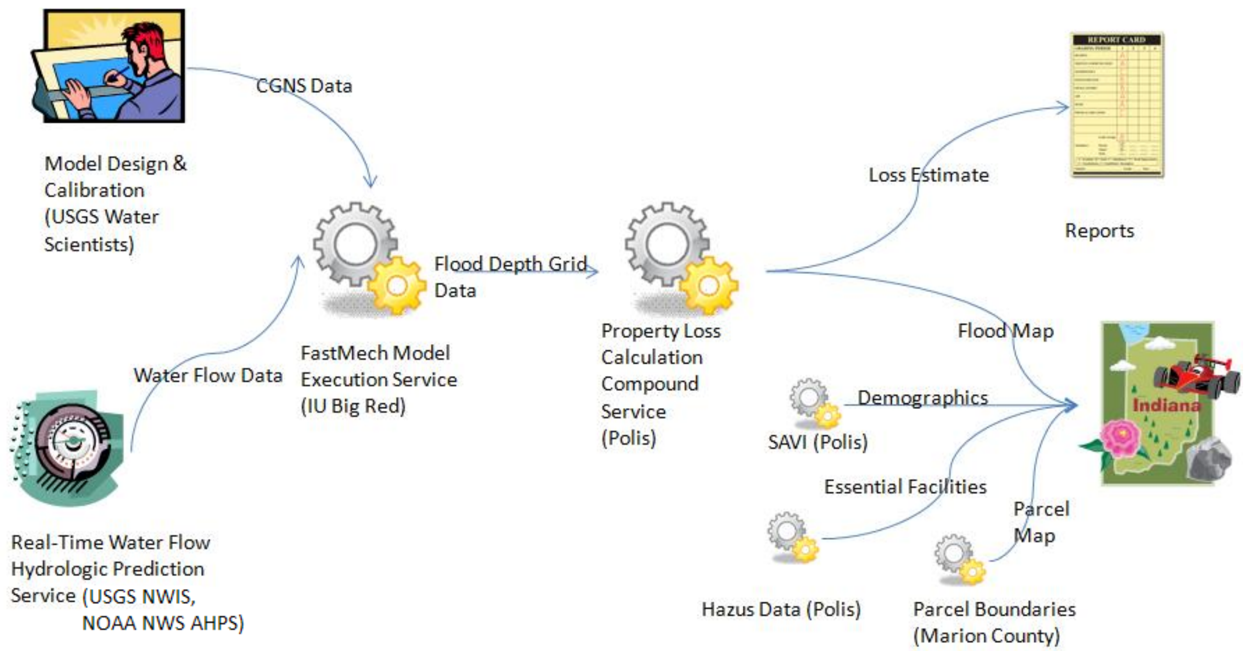

Figure 3: System Diagram (see text).

Figure 3 depicts the distributed processes, inputs and outputs between the people, sensors, and Web services. In this system, people, sensors, and Web services participate in achieving business workflows. Complex tasks such as building models and testing parameters or interpreting maps and reports are performed by experts. Real-time water flow data is generated and sent by sensors and distributed via a Web service. Model execution and data calculations are performed by Web services as well. You will also notice that these Web services are distributed nationwide. Real-time water flow gauge data is received from USGS NWIS (National Water Information System) and NOAA NWS AHPS (Advanced Hydrologic Prediction Service), FaSTMECH model execution services is hosted at Indiana University, and property loss calculation compound service is hosted at Polis. Also the graphic user interface mashup integrates other freely available services such as Marion County parcel maps and demographic maps from SAVI. All services encapsulate well-defined functions and are easily used to compose workflows.

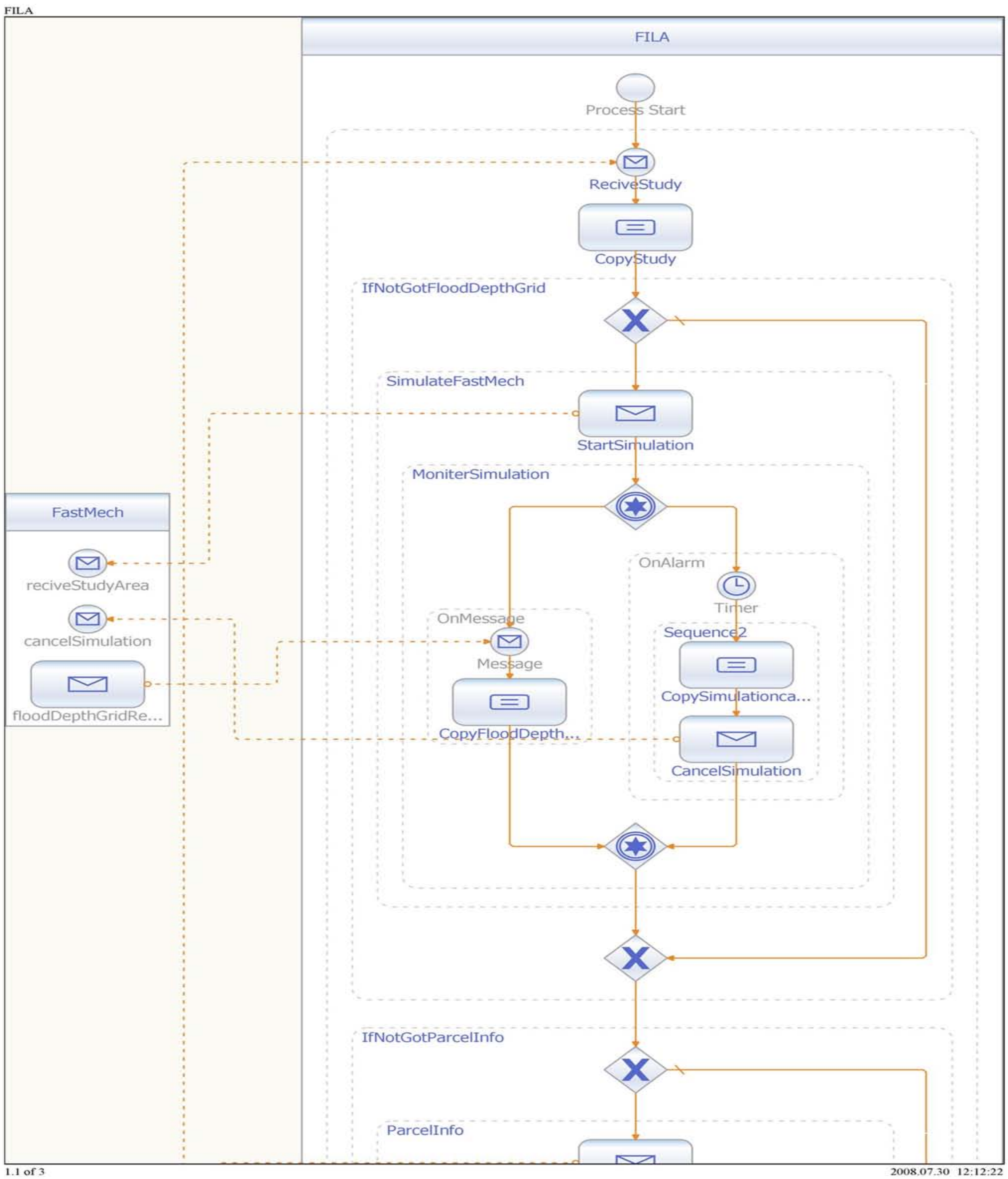

Figure 4: BPEL Process (see text).

Figure 4 depicts a partial view of the BPEL process. BPEL process orchestrates the workflow by passing messages between services and executing the service functions. This BPEL was designed using SUN Microsystems NetBeans 6.0 Integrated Development Environment (IDE).

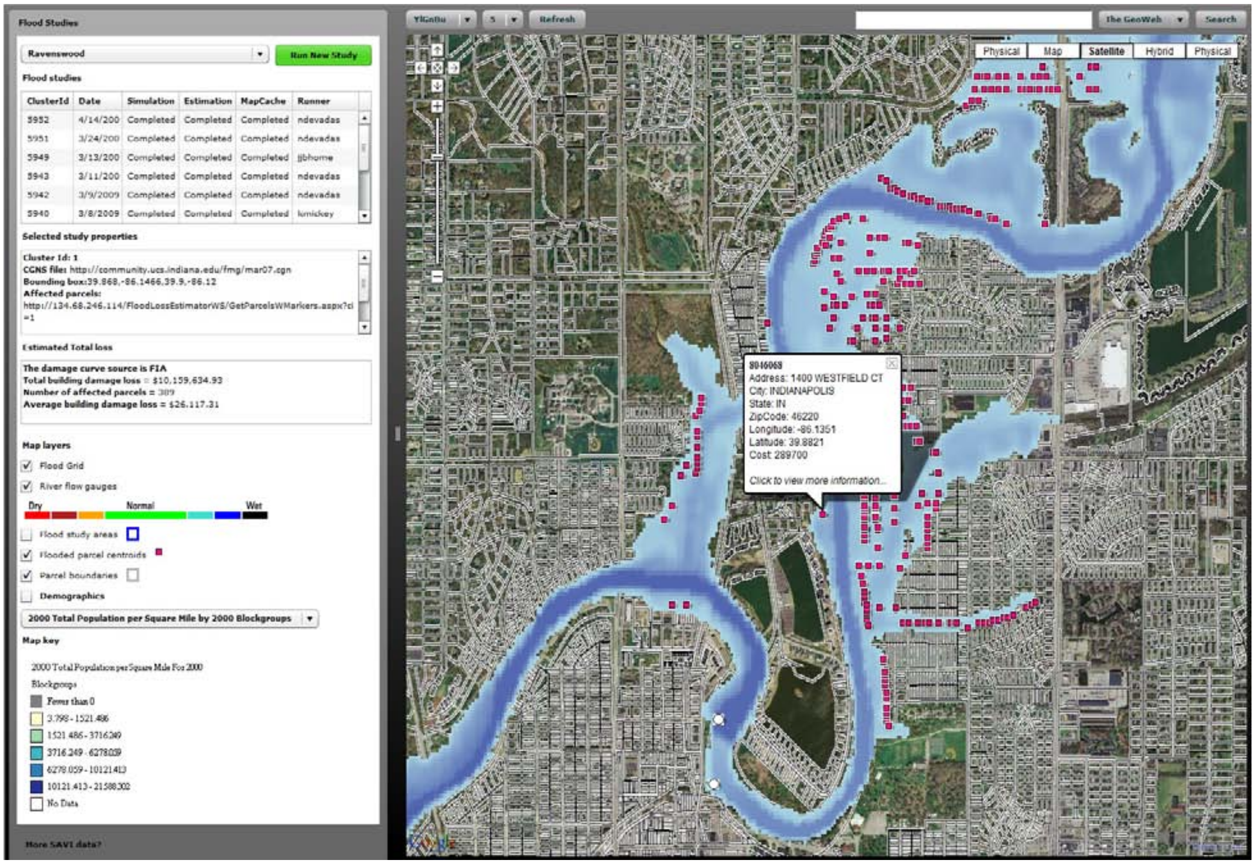

Figure 5: Graphic User Interface (see text).

Figure 5 depicts the graphic user interface (GUI). This application is developed using Adobe Flex SDK. It consists of two vertical panes. The left pane contains controls to start a flood inundation study and process output reports. The user interface invokes the process execution application asynchronously. Status of the process is continuously monitored and reported in GUI controls. The right pane of the GUI contains a control to map geospatial layers, primarily flood-related, like the flood depth GRID and flooded parcel centroids. The GUI also mash-ups freely available contextual geospatial layers such as parcel boundaries from county servers and demographic data layers from SAVI community information system and river flow gauge points from USGS NWIS and NOAA NWS AHPS.

SOA Definitions and Approach

Definitions

Please refer to appendix A for acronyms and definitions.

Approach

The FaSTMECH application is written in FORTRAN programming language. To run as a back-end service, we decoupled the solver portion of the code from its user interface. We also ported the code to run on Linux for our convenience. The FaSTMECH service was developed under the Open Grid Computing Environments (OGCE) Job Submission Service framework using Java, which provides a set of Axis2 Web services to execute and query status of the FaSTMECH application. FaSTMECH runs on computing resources managed by Condor system (www.cs.wisc.edu/condor/).

The rest of the Web services were developed in VB.NET and compiled against Microsoft CLR 3.5. These Web services are deployed under Internet Information Server (IIS) 6.0. Custom service chaining and process execution application was developed using VB.NET.

For BPEL modeling, we used Sun's NetBeans IDE. The BPEL module was deployed under Sun GlassFish Enterprise Server v3.

Requirements and Process Definition

To elicit requirements of the project, we met with meteorologists, hydrologists, and mitigation planners. After these meetings, we developed use cases and process flows of the project. The following sections describe the use cases and data inputs and outputs.

Usage Scenario:

Composition of a Geospatial Service-Oriented Architecture for Flood Inundation Mapping and Hazard Assessment

Authors:

- **Neil Devadasan, The Polis Center, IUPUI**
- **John Buechler, The Polis Center, IUPUI**
- **Marlon Pierce, Community Grids Lab, Indiana University**
- **Sally Pavlow, National Weather Service**
- **Scott Morlock, USGS Indiana Water Science Center**
- **Moon Kim, USGS Indiana Water Science Center**

Domain: Hydrology, Geospatial SOA, Flood Hazard Damage Assessment, GIS Visualization, Near Real-Time Flood Inundation

Description

The National Weather Service (NWS) and the U.S. Geological Survey work closely with state agencies—for example, the Indiana State Emergency Management Agency (Indiana Department of Homeland Security, IDHS)—when natural hazards such as severe storms and flooding occur. The State Emergency Operations Centers are manned twenty-four hours per day by staff from a variety of state agencies during these events. The most frequent severe weather-related disasters involve riverine flooding. The NWS provides flood warnings and forecasts before and during flood disasters. The USGS provides flood information to the NWS for its use in flood warnings and forecasts; the information is provided in near real-time from a network of stream gages. Stream gages provide water level (stage) data and volumetric streamflow data. IDHS coordinates emergency response and recovery activities using the best available information from the NWS and USGS on the extent of the flooding. The severe June 2008 floods starkly illustrated this risk and the necessity for interagency collaboration.

The Federal Emergency Management Agency (FEMA) supports the activities of the Indiana Department of Homeland Security (IDHS) and local governments that can prevent or reduce significant losses resulting from disasters. Recognizing that flooding is the most widespread and significant natural hazard in Indiana and that major flooding occurs within Indiana almost every year, IDHS sees a critical need to assess the potential severity and extent of floods in the update of hazard mitigation plan risk assessments. The updated risk assessments will be used to improve the mitigation strategies for communities.

Issues

- Tools do not readily exist to visualize the projected extent and depth of the flooding or estimate the damage to the affected structures in near real time.
- Web tools for disseminating information between agencies and to the general public need to be improved.
- There currently is limited ability to run extensive validation scenarios of the models using historical data.
- Flood forecasts need to be automated and event-driven using open Web architectures.

Actors & Goals

- Meteorologists, The National Weather Service (NWS)
- Hydrologists, The National Weather Service (NWS)
- Hydrologists, The U.S. Geological Survey (USGS)
- Hydrologists, The Indiana Department of Natural Resources (IDNR)
- Hydrologists, The Army Corps of Engineers (USACE)
- Hazard Analysts, Federal Emergency Management Agency (FEMA)
- Hazard Analysts, Indiana Department of Homeland Security (IDHS)
- Hazard Analysts, Indiana Department of Natural Resources (IDNR)
- Environmentalists, Environmental Protection Agency (EPA)
- FaSTMECH Simulation Service
- Assessment Data Service
- Parcel Data Service
- Loss Estimation Service
- Web Map Service
- GeoRSS and KML Forecast Streams

The goal is to develop an SOA base composite application to assist the following components:

- Hazard response planning
- Developing mitigation strategies for “what if” scenarios for reducing flood risk

Stakeholders & Interests

The Indiana Department of Homeland Security (IDHS) is responsible for supporting activities that can prevent or reduce the significant losses that result from disasters. Recognizing that flooding is the most widespread and significant natural hazard in Indiana, IDHS sees a critical need to apply computer-based geospatial, hydrologic, and river hydraulic flood modeling tools with HAZUS-MH risk analysis software for the purpose of assessing the frequency and severity of floods.

This project leverages resources from a variety of agencies including IDHS, the Indiana Department of Natural Resources (IDNR), the USGS, The Polis Center, the National Weather Service (NWS), and the U.S. Army Corps of Engineers (USACE). Flood mitigation is a focus activity of the Indiana Silver Jackets, an inter-agency natural hazard mitigation team. Using the technology to expand mitigation efforts highlights the collaboration of the Silver Jackets member agencies (FEMA, USACE, USGS, NWS, IDHS, IDNR, Department of Housing and Urban Development, and U.S. Department of Agriculture Natural Resources Conservation Service).

Within the last year, new computer resources with the potential to substantially improve the accuracy of flood risk assessments have become available, including high-resolution digital ground elevation models equivalent to contour intervals of one or two feet, and an update to FEMA’s HAZUS-MH risk assessment and hazard mitigation planning software program that allows users to easily import a flood surface to perform loss estimates, detailed digital local building inventory data, and state-of-the-art river flood hydraulic programs developed by the U.S. Geological Survey (USGS). These new computer resources, in combination with other new hazard science information, would produce detailed estimates of flood damages.

Appendix B lists use cases.

SOA Deployments and Acceptance

FaSTMECH flood inundation simulation execution Web service: The core FaSTMECH FORTRAN computation program was wrapped under the Open Grid Computing Environments (OGCE) Job Submission Service framework using Java, and deployed under an Apache Tomcat Web server running on the virtual Red Hat Enterprise Linux 4 environment hosted by the TeraGrid Gateway Hosting Service at Indiana University Bloomington. Computation job scheduling and available resources are managed by the Condor system running on the same virtual host, which can adapt and scale dynamically as the underneath hardware system expands. In addition, deployed on the same host is the real-time NWS river data monitoring service written in Python and input/output CGNS file processing services written in C++.

Parcel centroids and property value data services and flood loss estimation and loss assessment Web service: These services are developed using Windows Communication Foundation (WCF) and deployed on a Windows 2003 server running IIS 6.0. Service output content are managed by a SQL Server 2005 database and deployed on a Windows 2003 server. During the test runs the system had shown acceptable performance; as the services are modular, it can be scaled up to respond to higher demands.

BPEL Process: BPEL orchestration is executed with Sun Java Application Server (GlassFish Enterprise Server v3). We used Sun's NetBeans BPEL designer.

Process Execution .NET Application: This is a .NET based execution process. Due to some difficulties of seamlessly integrating J2EE Web services and WCF Web services we had to develop an API based .NET executable. This executable chains the Web services asynchronously to achieve the workflow.

Graphic User Interface: User interface (UI) was developed using Adobe Flex SDK. .

The Flex application is deployed in an IIS 6.0 server as a flash movie. It can be run under Flash Player 9.0 or 10.0.

Best Practices

Summarize applied standards and specifications: We developed most of the Web services using WSDL and SOAP XML formats. Additionally a few services were developed using the Representational State Transfer (REST) approach. Data produced by services is exchanged through URLs. The real-time flood monitoring service consumes RSS feeds from the NWS. Maps were generated with KML. Overlays in the GUI (such as parcel information) were obtained from OGC-compatible Web Feature and Web Map Services provided by Marion County, IN. The data calculations were performed with loss estimation tables from HAZUS-MH.

Document all monitoring or testing performed on the service to measure the quality and availability of the service. We developed test suites for selected services using JUnit. These were used to verify services operated in the correct fashion when transferred from development to production environments. We also validated the entire workflow against historical flood information. Flood modeling and related services are run on the hardware-redundant and mirrored gateway hosting service. We ported the FaSTMECH service to Linux (from Windows) in order to potentially make use of the NSF TeraGrid.

Identify any optimization applied to improve quality of service. We maintained a cache of the NWS real-time river monitoring data (updated hourly) to improve the responsiveness of the system. This allows us to also maintain data archives that can be used to simulate historical incidents.

Identify changes in approach taken in the deployment and acceptance phase that differed from the original design. The original design called for pulling real-time river data from the local (Indianapolis) USGS office's database. This was not feasible because of security and privacy issues; therefore, it was replaced with a service that pulls data directly from the NWS feeds. We also learned that the FaSTMECH parameters were only valid for very high river stages (300 cubic meter/second discharge rates), so we developed a trigger capability in the real-time data monitoring service so that signals are only produced when minimum conditions are met.

Identify any service patterns that might be applicable to other services. Wrapping applications is a common task for many scientific gateways. The issue is that the solver service (FaSTMECH in our case) may take several minutes to hours to complete. This is commonly handled using a callback pattern coupled to either a polling or event notification system. The callback pattern means the service returns a message immediately to the client (usually with a ticket, or receipt) instead of blocking until completion. The state of the execution can then be communicated using either a poll (the client periodically checks the service for state updates) or an event push (the service sends a message to the client). We implement tickets using unique ID (UID) strings. The polling (or pull) method is the most common since it is compatible with HTTP and browser clients (AJAX, for example, is commonly used to poll). However, polling introduces unnecessary network traffic and so should be carefully tested for scaling. Event-based messages are most commonly used to communicate state changes between backend services.

Describe the selection and performance of service binding patterns (GET, POST, SOAP, other) used in the solution. We implemented all services with SOAP messages; the river monitoring service consumes RSS and generates SOAP. This choice was made in part to leverage pre-existing services. We did not explicitly measure performance of the protocol, as the computation step (FaSTMECH) was significantly longer than any network I/O. Our services were implemented with Apache Axis 2's Java framework and .NET, which have been benchmarked extensively.

Describe how common vocabularies, ontologies, and data structures were supported by this project. We use CGNS as a common data format. CGNS is a standard format for computational fluid dynamics. In the GUI we also incorporate OGC-compatible data formats (KML, GML).

Identify measures that could be/were taken to improve the scalability and extensibility of this service. The FaSTMECH service currently runs on a single node on IU’s Gateway Hosting Service, which limits the number of jobs that can run simultaneously. The OGCE Job Management service, however, can be used to submit FaSTMECH to the NSF TeraGrid, which makes over 100,000 computing cores available. This service could be used to greatly expand the computational resources available. The CGNS files are specialized to this specific application (that is, for manipulating FaSTMECH parameters). It may be possible to develop a more generalized version of this service.

Document the results of applications using this service, including any current project experience with service chaining, composition, or orchestration. As described in separate sections, our project integrates multiple services into a workflow. The key problem with workflow operations is always the adequate communication of failure conditions to down-chain services.

Identify any emergent technical requirements that should be considered in future deployments based on the experience of this project. Several of our services (particularly FaSTMECH) would benefit from Cloud Computing infrastructure systems such as Amazon EC2. These would provide additional, expandable infrastructure to handle variable systems loads. One could expect, for example, much higher loads during training sessions, as well as when weather and hydrological conditions make floods likely. Related Cloud

Computing runtime engines such as Apache Hadoop and Microsoft Dryad can be used to manage complicated calibration scenarios. These are currently executed by hand but could be replaced by (for example) genetic algorithms that efficiently investigate the complicated parameter space of the solver.

Describe the ease and issues of re-deployment of this service to other host sites (government, commercial, ISP). The FaSTMECH core solver was ported to Linux, so both Linux and Windows server environments can be used. The services were also developed on virtual machines, so it should be possible to distribute virtual machine images containing our service stack. One issue is that FasTMECH is a third-party application, developed by the USGS, so redistribution and licensing issues need to be discussed with the developers. Most services were developed in Java, and we put effort into build systems (using Apache Maven) and source code management (SVN on SourceForge), so we believe the software stack can be ported to new hosting environments.

Identify preferred measures to assure service availability and reliability (redundancy, failover approach, hosting requirements, synchronization approach, and service recovery time). We host services on the the IU gateway hosting service, as described previously. This provides fault tolerance and redundancy. We also use caching to maintain a local copy of third-party data. This improves performance but also shields against network and service outages at the NWS.

Identify any prospective service level agreements emerging from the deployment of this SOA. Since the FaSTMECH service may need several hours to run certain scenarios, it is important that privileged users (IDHS, for example) be given high priority queues, especially during periods of peak usage. This may include the ability to disable access for lower priority users. The simulation results themselves are also sensitive, since they calculate flood damage to real land parcels. Since the model results are only valid for calibrated floods, simulation runs must be made available only to authorized users. The ability to launch new simulations should ultimately be restricted to privileged users; public users should be given access only to approved simulation runs. This is necessary because of real estate and insurance issues.

Project Management

The composite application has been demonstrated to multiple federal, state, and local entities. The application is available to Silver Jackets participants and will remains active for the NWS flood forecast gage in the Ravenswood area of Indianapolis. The application will continue to be evaluated during upcoming floods. The Silver Jackets members are pursuing funding to extend the application to an additional 20 NWS Indiana forecast points. This follow-up phase will require a software component to manage the inputs and outputs from the 20 stream gages within different watersheds across Indiana.

Feedback on Cooperative Agreements Program

The NSDI Cooperative Agreements Program provided an opportunity for Indiana University computer science researchers to work with applied science researchers and water science practitioners on an initiative that has real world implications annually in much of the U.S. The program provides for the injection of new technologies and approaches into the geospatial community. The grant provided both research challenges and important collaboration experiences.

Strengths: the program reviews and funding decisions were made very quickly. We were also pleased to have the opportunity to prototype some cloud infrastructure, a side project that came out of discussions and regular teleconferences. The program's mixture of government, enterprise, and academic teams was also very beneficial.

Weaknesses: the original nine-month time frame seems a little short for most projects. A twelve-month program (with same funding amount) would be easier to manage.

The team had no program management concerns. The team received prompt responses to questions. Additionally the program management team’s format for meetings and communications facilitated collaborations.

Appendix A: List of Acronyms and Definitions

AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
BPEL	Business Process Execution Language
CS-W	Web Catalog Service
EAI	Enterprise Application Integration
ESB	Enterprise Service Bus
ETL	Extraction, Transformation, and Loading
FGDC	Federal Geographic Data Committee
FIA	Federal Insurance Agency
GeoDRM	Geospatial Digital Rights Management
GIS	Geographic Information System
GOS	Geospatial One-Stop
GUI	Graphic User Interface
HTTP	Hypertext Transport Protocol
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IIS	Internet Information Server
ISO	International Organization for Standardization
IT	Information Technology
JDK	Java Development Kit
KML	Keyhole Markup Language
OASIS	Organization for the Advancement of Structured Information Standards
OGC	Open Geospatial Consortium, Inc.
RDBMS	Relational Database Management System
REST	Representational State Transfer
SDK	Software Development Kit
SOA	Service-Oriented Architecture
SSL	Secure Sockets Layer
UDDI	Universal Description, Discovery, and Integration
UI	User Interface
VB	Visual Basic
WCF	Windows Communication Foundation
WCS	Web Coverage Service
WFS	Web Feature Service

WMS	Web Map Service
WS	Web Services
WS-BPEL	Web Services Business Process Execution Language
WSDL	Web Services Description Language
WSDM	Web Services Distributed Management
WS-I	Web Services Interoperability Organization
XACML	Extensible Access Control Markup Language
XML	Extensible Markup Language
XPath	XML Path Language

Appendix B: Use Cases

Use Case: Response planning after NWS issues a flood watch or warning

Issues

IDHS is concerned about flooding.

Requirements

No specific requirements.

Actors/Roles

- *Available Services:*
    1. FaSTMECH Simulation:
        - Input: DEM, USGS real-time stream gage data, NWS five-day flood forecast data
        - Output: Flood depth GRID ASCII file
        - Preconditions: Calibration of the surface model
        - Effects: Event Raised => flood_depth_grid_generated
    2. Parcel Data Service:
        - Input: Flood depth GRID ASCII file
        - Output: Parcel information
        - Preconditions: True
        - Effects: Event Raised => parcel_information_for_grid_generated
    3. Assessment Data Service:
        - Input: Parcel information for the grid
        - Output: Property assessment information
        - Effects: Event Raised => property_assessment_information_generated
    4. Loss Estimation Service:
        - Input: Property assessment information, HAZUS-MH damage curves
        - Output: Parcel information
        - Preconditions: HAZUS-MH damage curves calibration
        - Effects: Event Raised => damage_estimations_generated
    5. Web Map E-Service:
        - Input: Property assessment information, parcel centroids, GRID file
        - Output: Map of flood inundation and affected parcel centroids and affected parcel information with an Arial photo background.

- Additional output: KML and GeoRSS feeds of output.
- Effects: Queriable Web site with maps and tables

- *End-users and their roles*:
  1. Meteorologists
     - Meteorologists model weather events which provide input data to develop stream flow estimates.
  2. Hydrologists
     - Hydrologists generate flood forecast information and also capture real-time stream gage data. Then hydrologists perform the inundation modeling.
  3. Hazard Analysts
     - Hazard analysts use the output of the model to determine the extent and severity of flooding.
  4. Environmentalists
     - Environmentalists use the output of the model to determine potential environmental impacts of the flooding, including hazardous materials locations within the flooded area.
  5. General Public
     - The general public checks Web sites for latest flood forecasting results.

Assumptions

We assume the following:

- Web-accessible data sources such as stream gages will be reliable (i.e. provide accurate and timely data) and robust.
- The flood modeling codes give reliable forecasts (has been validated).
- The workflow described above can be executed in a timely fashion.

Scenario/Steps

1. NWS issues a flood watch or warning; IDHS is concerned about flooding.
2. Based upon USGS streamflow-gaging station data, or NWS flood prediction data, an approximate two-year (50 percent) flood is reached.
3. USGS FaSTMECH flood inundation Web service is automatically activated.
4. Upon receiving flood_depth_grid, parcel data service is activated.
5. Upon receiving parcel_information_for_grid_generated, assessment data service is activated.
6. Upon receiving property_assessment_information_generated, loss estimation service is activated.
7. Upon receiving damage_estimations_generated, maps are generated by Web map e-service and published for analyzing.

Reasoning Techniques Required

- Advance planning techniques
- Data integration techniques
- Error detection and validation of results (i.e. avoid or minimize human errors that may result in incorrect forecasts).

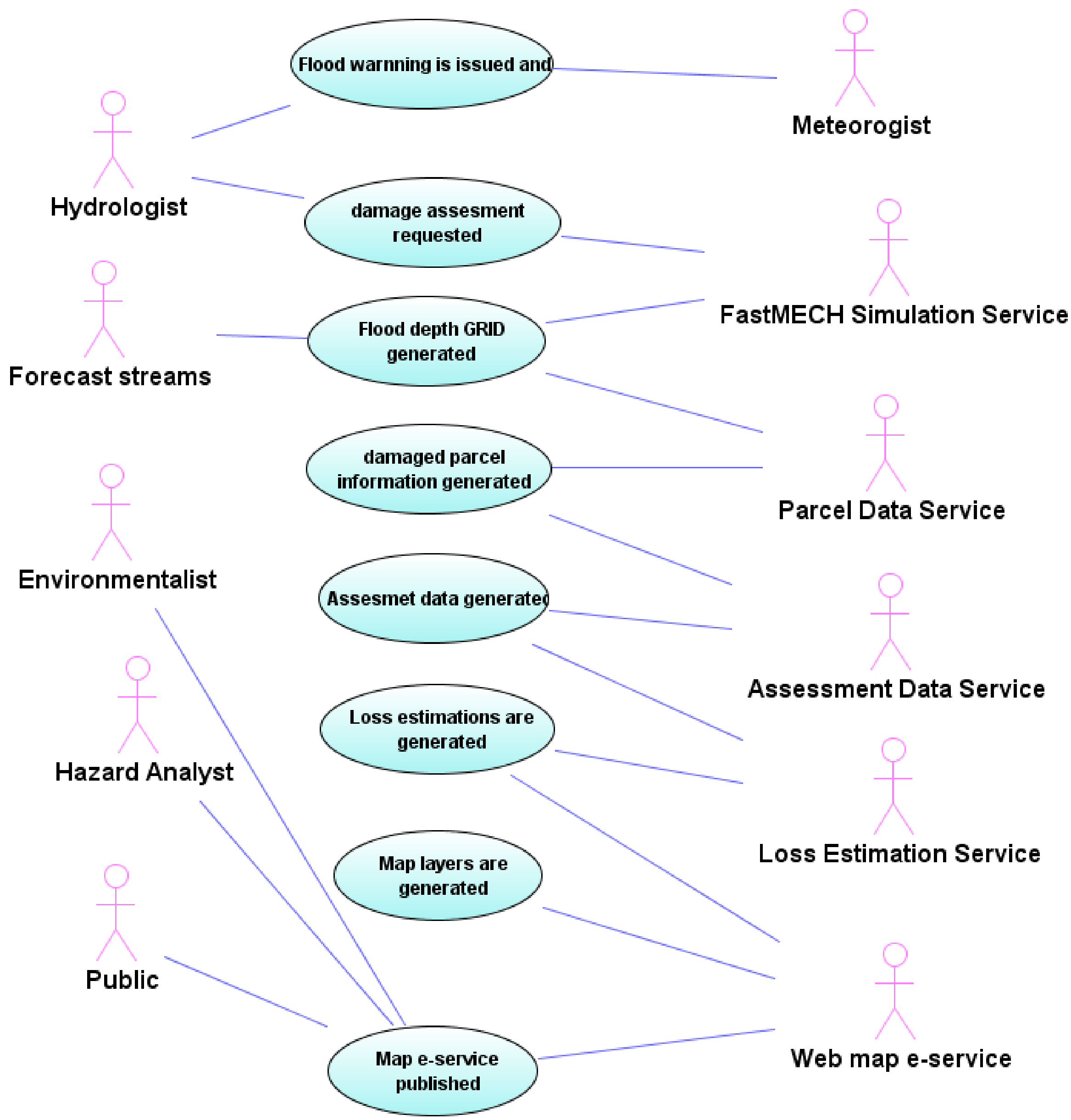

Figure 6: UML Use case diagram

Use Case: Developing mitigation strategies for “what if” scenarios for reducing the flood risk.

Issues

FEMA, IDHS, and the local community would like to assess flood risk for several flood frequencies and then develop mitigation strategies. As part of this effort, the community would develop an annualized estimate of flood risk.

Requirements

No specific requirements.

Actors/Roles

- *Available Services:*
  1. FaSTMECH Simulation:
     - Input: DEM, user-supplied discharges for specific gauge locations
     - Output: Flood depth GRID ASCII file
     - Preconditions: Calibration of the surface model
     - Effects: Event Raised => flood_depth_grid_generated
  2. Parcel Data Service:
     - Input: Flood depth GRID ASCII file
     - Output: Parcel information
     - Preconditions: True
     - Effects: Event Raised => parcel_information_for_grid_generated
  3. Assessment Data Service:
     - Input: Parcel information for the grid
     - Output: Property assessment information
     - Effects: Event Raised => property_assessment_information_generated
  4. Loss Estimation Service:
     - Input: Property assessment information, HAZUS-MH damage curves
     - Output: Parcel information
     - Preconditions: HAZUS-MH damage curves calibration
     - Effects: Event Raised => damage_estimations_generated
  5. Web Map E-Service:
     - Input: Property assessment information, parcel centroids, GRID file
     - Output: Map of flood inundation and affected parcel centroids and affected parcel information with an Ariel photo background
     - Effects: Queriable Web site with maps and tables
- *End-users and their roles*:
  1. Meteorologists
     - None
  2. Hydrologists
     - The hydrologists provide discharge data for each theoretical flooding event.
  3. Hazard Analysts
     - The hazard analysts use the output of the model to analyze the risk from each flood scenario.
  4. Environmentalists
     - The environmentalists use the output of the model to determine potential environmental risks from each flood scenario.
  5. Local Community
     - The local community desires to prepare a detailed flood risk assessment.

Goals/Context

Assumptions

1. Accurate historical data (both inputs and outputs) are available.
2. Estimated losses from accurate sources.

Scenario/Steps

1. The local community working with IDHS desires to prepare a detailed flood risk assessment.
2. The local community provides flood scenarios to the IDHS, USGS, and/or other partner agencies.
3. USGS provides historical streamflow and flood frequency data.
4. USGS FaSTMECH flood inundation Web service is activated.
5. Upon receiving flood_depth_grid_generated, parcel data service is activated.
6. Upon receiving parcel_information_for_grid_generated, assessment data service is activated.
7. Upon receiving property_assessment_information_generated, loss estimation service is activated.
8. Upon receiving damage_estimations_generated, maps are generated by Web map e-service and published for analyzing.
9. IDHS and local community analyze the data, if necessary change the flood discharges and reiterate simulation.

Reasoning Techniques Required

- Advance planning techniques
- Data integration techniques

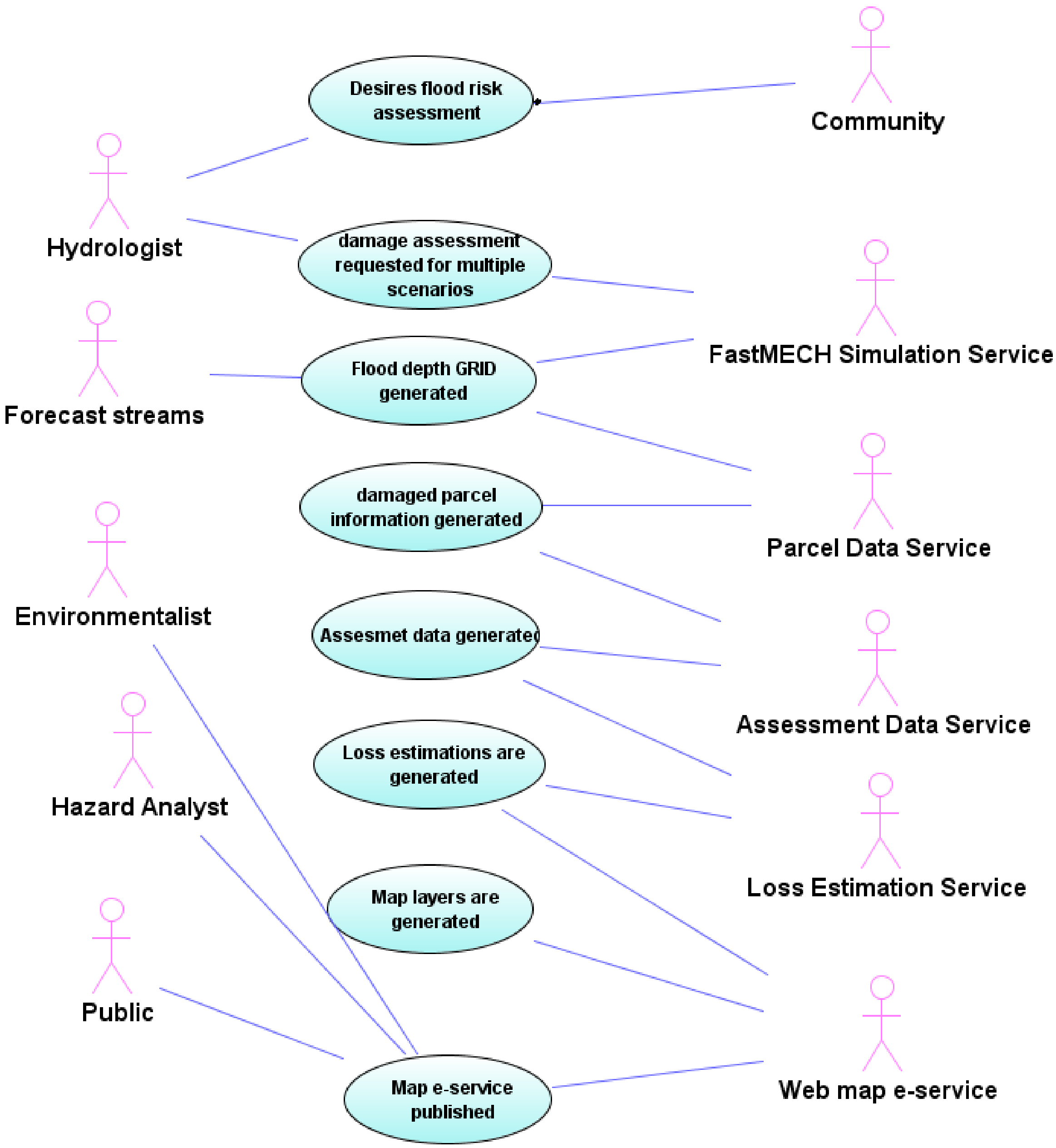

Figure 7: UML Use case diagram