

# Milestone 1 Report



**Submitted To:** Program Manager  
GeoConnections  
Victoria, BC, Canada

**Submitted By:** Jody Garnett  
Brent Owens  
Refractions Research Inc.  
Suite 400, 1207 Douglas Street  
Victoria, BC, V8W-2E7  
jgarnett@refractions.net  
Phone: (250) 885-0632  
Fax: (250) 383-2140

# TABLE OF CONTENTS

---

MILESTONE 1 REPORT.....	1
TABLE OF CONTENTS.....	2
1 SUMMARY OF WORK ACCOMPLISHED.....	3
2 ENCOUNTERED PROBLEMS AND SOLUTIONS .....	5
2.1 WEB FEATURE SERVER LOCKING MODEL.....	5
2.2 LACK OF CONCRETE VALIDATION EXAMPLES.....	5
2.3 GEOTOOLS2 DEVELOPMENT ENVIRONMENT .....	5
2.4 WEB FEATURE SERVER PARTIAL SUCCESS MODEL .....	5
3 WORK PLAN .....	6

# 1 SUMMARY OF WORK ACCOMPLISHED

---

The goal of this project is to build a Validating Web Feature Server (VWFS) by adding extra functionality to a Web Feature Server (WFS). A Web Feature Server is an application that delivers information about spatial objects across the Web.

The WFS that we will be using is called GeoServer. GeoServer is an open source implementation of the Open GIS Consortium's Web Feature Server Specification. GeoServer allows information about spatial data to be sent to and received from a Geotools2 Data Source. Our project will add a level of validation to GeoServer to maintain spatial database integrity.

For our Validating Web Feature Server, all required documents and research have been completed to meet the first milestone. The documents are:

## Phase 1.1

- Report on academic references
- Report on online references
- Report on XML tools and XML technologies
- Report on MRSMS tools and required feature set

## Phase 1.2

- Design document for plug-in API
- Design document for validation language processor
- Design document for validation language

## Phase 2.1

- Report on transactional integrity issues and data source differences
- Design for transactional WFS implementation

## Phase 2.2

- Report on design lessons from initial implementation

We have accomplished a full design of the Validating Web Feature Server and implementation of strong transaction support.

Our design has covered:

- Integration strategy: how and where we will incorporate validation into GeoServer.
- The validation processor that performs the validation on features.
- Definition of the Plug-In Validation API.
- Definition of the Validation Language used with the Validation Plug-Ins.
- The addition of strong transaction support to Geotools2 for PostGIS, Oracle, and ArcSDE.

For the first milestone we have implemented:

- 
- A Validation Language designed with the XML Schema specification.
- A Table locking procedure for a PostGIS data source.
- A transactional data source to allow strong transaction support in Geotools2.

## 2 ENCOUNTERED PROBLEMS AND SOLUTIONS

---

Below is a list of problems encountered and our solutions to overcome them.

### ***2.1 Web Feature Server locking model***

The Web Feature Server specification's locking model represents a compromise between row locking and full long-term transaction support (row versioning). The locking model defines timed feature locks that are persistent across transactions.

Although this simple locking model will be successful, initial feedback has been negative.

Even though this is a valid complaint, we have decided to follow the specification.

### ***2.2 Lack of concrete validation examples***

Our search for online and academic references resulted in very few concrete validation examples. Much of what we found was specific to a limited domain or exceeded the capabilities of existing toolsets. We were disappointed in the lack of publicly available validation tests.

In response we obtained advice from our contacts in IMB and MSRM.

### ***2.3 Geotools2 development environment***

The GeoTools2 development team is using alpha-phase software, called Maven, for its build process. We have run into delays on several occasions due to problems in Maven and its use for building Geotools2.

The Geotools2 team has been very supportive in fixing problems and has considered replacing Maven.

### ***2.4 Web Feature Server partial success model***

The WFS specification's partial success model is provided for data sources with limited transaction support. This mechanism looked like an opportunity for supporting partial acceptance for feature validation.

The design for partial validation success proved complex. Our contacts at IMB and MSRM assured us that this was unnecessary. Therefore we have decided not to pursue this idea at this time.

### 3 WORK PLAN

Our work plan for the next milestone:

Calendar Month	Phase Completion	Event	Specific Tasks
<b>October 2003</b>	Oct 3, 2003	Phase 1.3	<ol style="list-style-type: none"> <li>1) Add data validation hooks to the GeoServer core.</li> <li>2) Test data validation modules using test data and known degenerate cases, stress validation modules.</li> <li>3) Test data validation hooks in GeoServer.</li> <li>4) Report on design and recommend changes to existing code and to constraint language on the basis of current implementation.</li> </ol>
<b>November 2003</b>	Nov 3, 2003	Phase 2.3	<ol style="list-style-type: none"> <li>1) Implement design changes described in previous phase.</li> <li>2) Implement ArcSDE datasource driver based on model investigated in the previous phase.</li> <li>3) Implement Oracle Spatial datasource based on model investigated in the previous phase.</li> <li>4) Test concurrency strength of transaction implementation. Stress test to expose any limitations in scalability or maximum concurrency. Resolve any race conditions and document locking semantics.</li> <li>5) Report on design implications of any implementation difficulties encountered during this phase. Recommend any changes to the design required.</li> </ol>

<b>November 2003</b>	Nov 21, 2003	Phase 2.4	<ol style="list-style-type: none"> <li>1) Stress test all data sources, and gather metrics on expected performance under various loads.</li> <li>2) Ensure all in-code documentation is complete for all drivers and up-to-date with current design.</li> <li>3) Document issues involved in implementing transaction data sources. Table layouts, expected behavior, heuristics for freeing unused locks, etc.</li> </ol>
<b>December 2003</b>	Dec 3, 2003	Overhead	Reports and documents
<b>December 2003</b>	Dec 5, 2003	Finish	Milestone 2 complete.