

2010 NSDI CAP Final Report

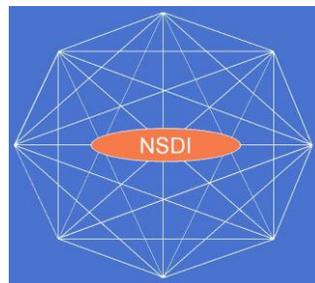
2010 NSDI Cooperative Agreement Program Proposal Category 2: Framework Data Exchange through Automated Geo-Synchronization

New Hampshire NHD GeoSynchronization Network

Prepared By:



Presented To:



February 9, 2012
Award Number G10AC00238

Organizations

Applicant Organization Information (Submitted on Behalf of Government Partner):

CubeWerx USA/ NHGS, 12052 Willowood Drive, Lake Ridge, VA 22192

Internet Address: <http://www.cubewerx.com>

Phone: (703) 628 8655, Email: jharrison@cubewerx.com

Government Partner:

New Hampshire Geological Survey, New Hampshire Dept. of Environmental Services (NH DES)

PO Box 95, 29 Hazen Drive, Concord, NH 03302-0095

Internet Address: <http://www.des.state.nh.us>

Principal Investigators:

Rick Chormann, Senior Hydrogeologist

Phone: (603) 271-1975, Email: rchormann@des.state.nh.us

CubeWerx USA, 12052 Willowood Drive, Lake Ridge, VA 22192

Internet Address: <http://www.cubewerx.com>

Phone: (703) 628 8655, Email: jharrison@cubewerx.com

Applicable Framework Themes: National Hydrography Dataset (NHD)

Geographic Scope or Area: New Hampshire, Regional Drainage Basins and Nationwide

Executive Summary

The New Hampshire Geological Survey and CubeWerx USA are pleased to submit this 2010 NSDI Cooperative Agreement Program (CAP) Category 2 Final Report for the Category 2, “Framework Data Exchange through Automated Geo-Synchronization” effort. Since kickoff the project developed and deployed a federated approach for exchanging framework data between state and national data stores using Geo-Synchronization and common National Hydrography Dataset (NHD) data models, services and applications. The project adopted a realistic approach to bridge the gap between current NHD production operations and the vision of the CAP grant. Specifically, it deployed new Geo-Synchronization Services (GSS) with the capability to transform Data Delivery Format (DDF) XML produced by current NHD ArcGIS tools into Web Feature Service Transactions (WFS-T) – providing the capability to support current and future NHD operations simultaneously. The project team also developed a simple web-based application to allow users to upload DDF files produced by current tools to the Geo-Synchronization Service (GSS) - further enabling support to current operations. To support future operations, a complete set of web tools for publishing and reviewing NHD edits from online users was developed and deployed – providing the potential for ‘crowdsourced’ updates of selected hydrography data using Geo-Synchronization. In support of the project the team configured and deployed multiple WFS and Web Map Services (WMS) with state and federal Framework NHD to support operations testing and deployment. Finally, the project coordinated the technical approach by engaging experts from the NHD community and conducted training on the new NHD Geo-Synchronization tools. The results of the project are a logical extension of NH DES’s stewardship agreement with USGS for developing and maintaining the state’s hydrography layer, and will enhance current stewardship practices to enable Geo-Synchronization and more participation in NHD.

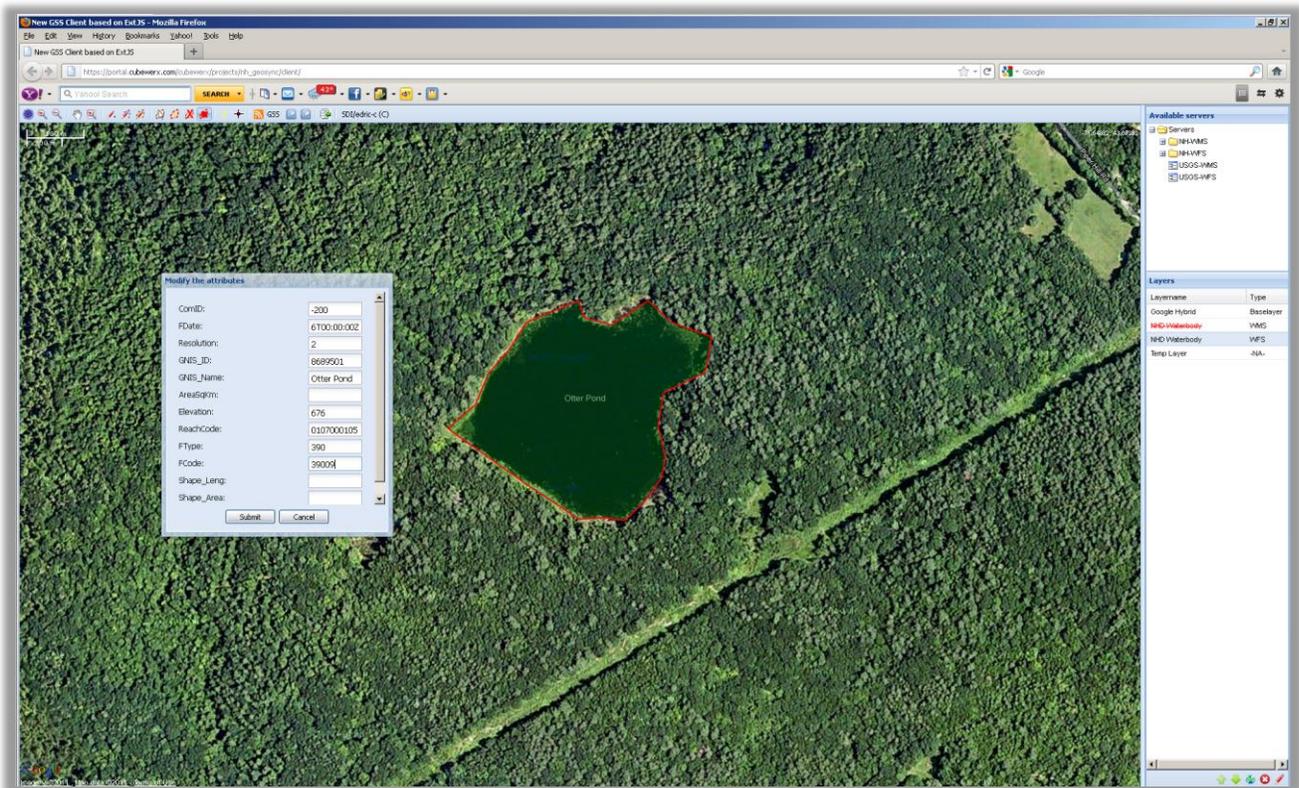


Figure 1 – This project developed NHD Geo-Synchronization tools that support both current and future NHD operations – including easy-to-use web tools for future ‘crowdsourced’ updates of selected hydrography data.

Project Narrative

The goal of this project, the New Hampshire NHD Geo-Synchronization Network, is to help transform the hydrography data theme of the NSDI into an agile, responsive framework driven by collaborative data maintenance partnerships at local, state and federal levels (Figure 2). To achieve this objective the project developed and deployed Geo-Synchronization Services (GSS), multiple Web Feature Services (WFS), Web Map Services (WMS), Identity Management Services and easy-to-use Web-based applications to synchronize NHD updates into an authoritative State data layer - which may then be provided for integration at national levels.

As a key information resource, NHD is a priority for the 2010 CAP and this project is providing tools and processes that can serve as models for other regions. The project built on CubeWerx experience developing the draft NHD Geography Markup Language (GML) Framework Schema¹ referenced in the grant solicitation. An example of this information displayed as a Web Map Service (WMS) in Google Earth is shown below (Figure 3).

¹ <http://frameworkwfs.usgs.gov/>

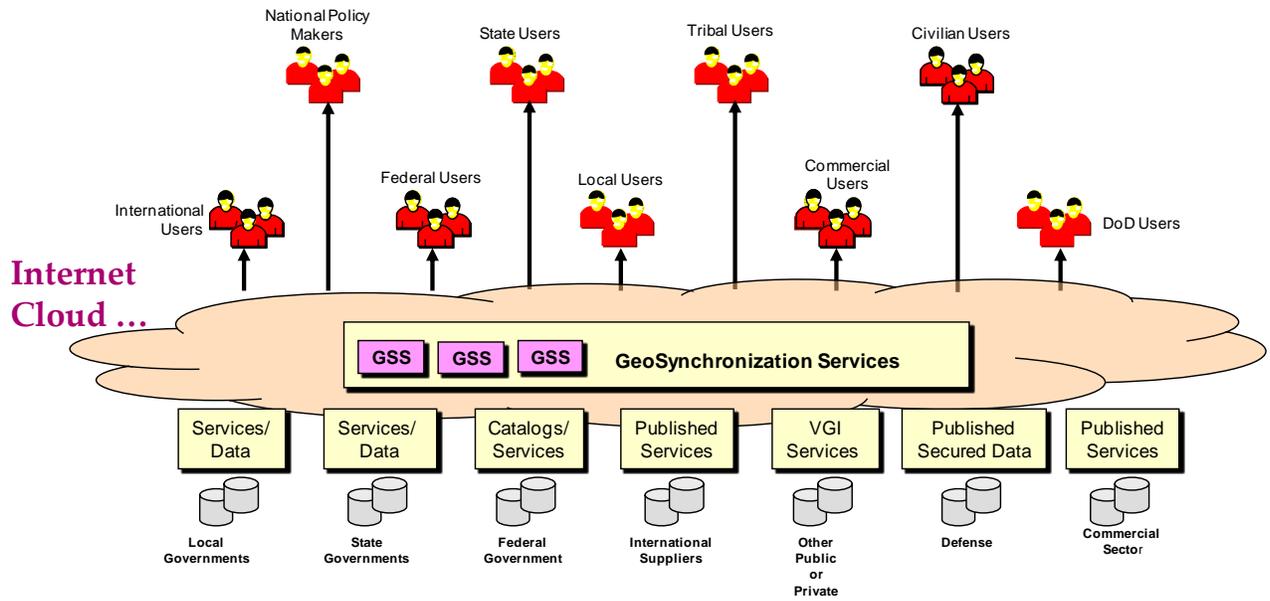


Figure 2 – The New Hampshire NHD Geo-Synchronization project advances the goals of Collaborative Spatial Data Infrastructure (SDI) based on open standards

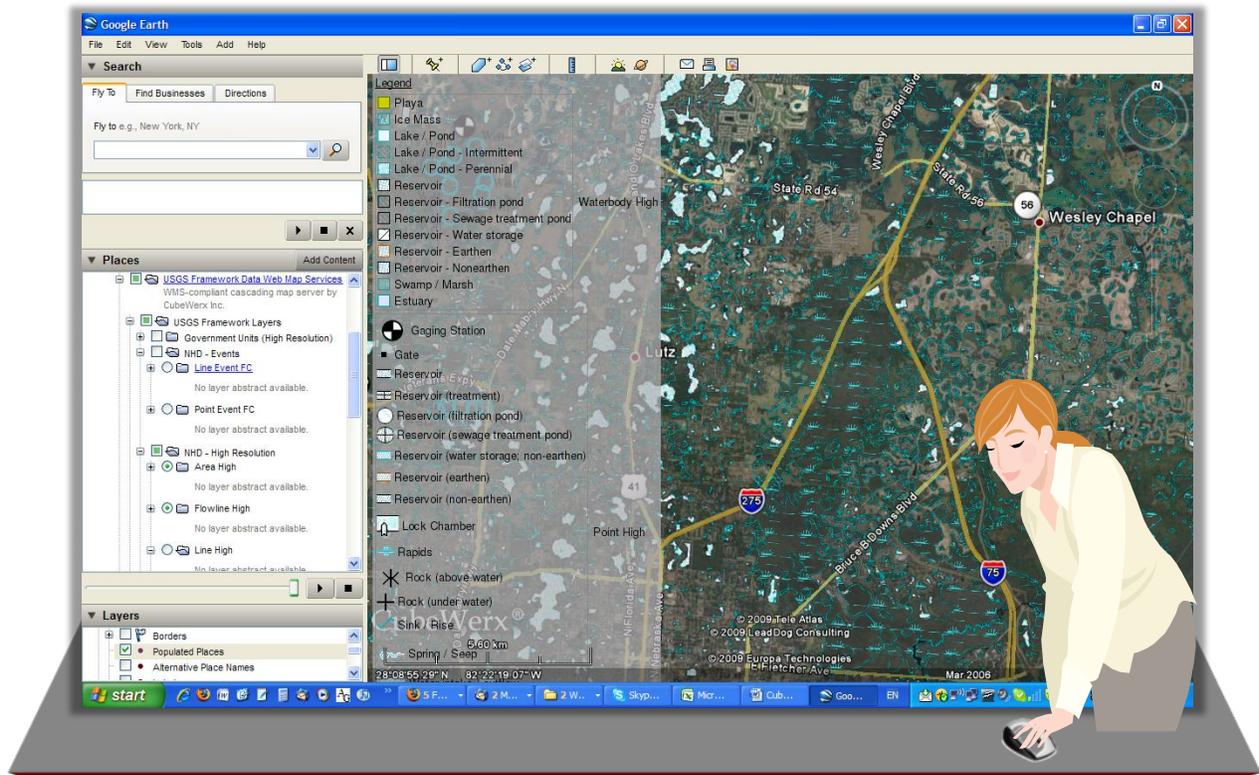


Figure 3 – NHD ANSI Framework information, shown as WMS in Google Earth

The project was based on the New Hampshire implementation of the NHD, a GIS comprised of spatial features that represent natural and man-made surface waters such as lakes, ponds, rivers, streams and wetlands. These features are mapped as a combination of stream centerlines and “artificial paths” through water bodies at a scale of 1:24,000.

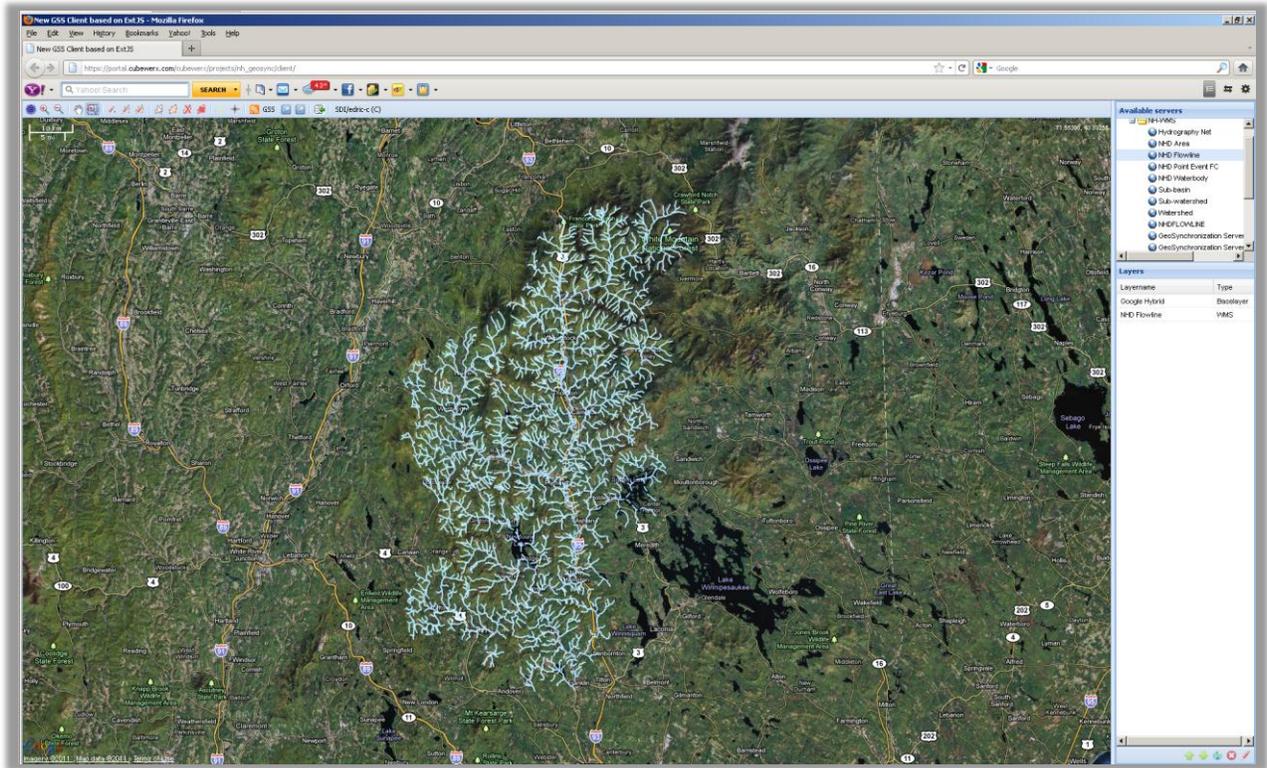


Figure 4 – The project was based on the New Hampshire implementation of the NHD

The effort acknowledged that the NSDI CAP seeks to migrate to a standards-based web services model to mitigate barriers associated with accessing and updating geospatial data. The project also recognized that the next phase of this transformation is to develop a network for collaborative data maintenance and integration of geospatial data into authoritative State data layers, which can then be provided for integration into The National Map. *However, to realize this vision the NSDI must bridge the gap between current and future operations on specific data themes like NHD (Figure 5). The project addressed these issues and provides a potential model for nationwide operations.*

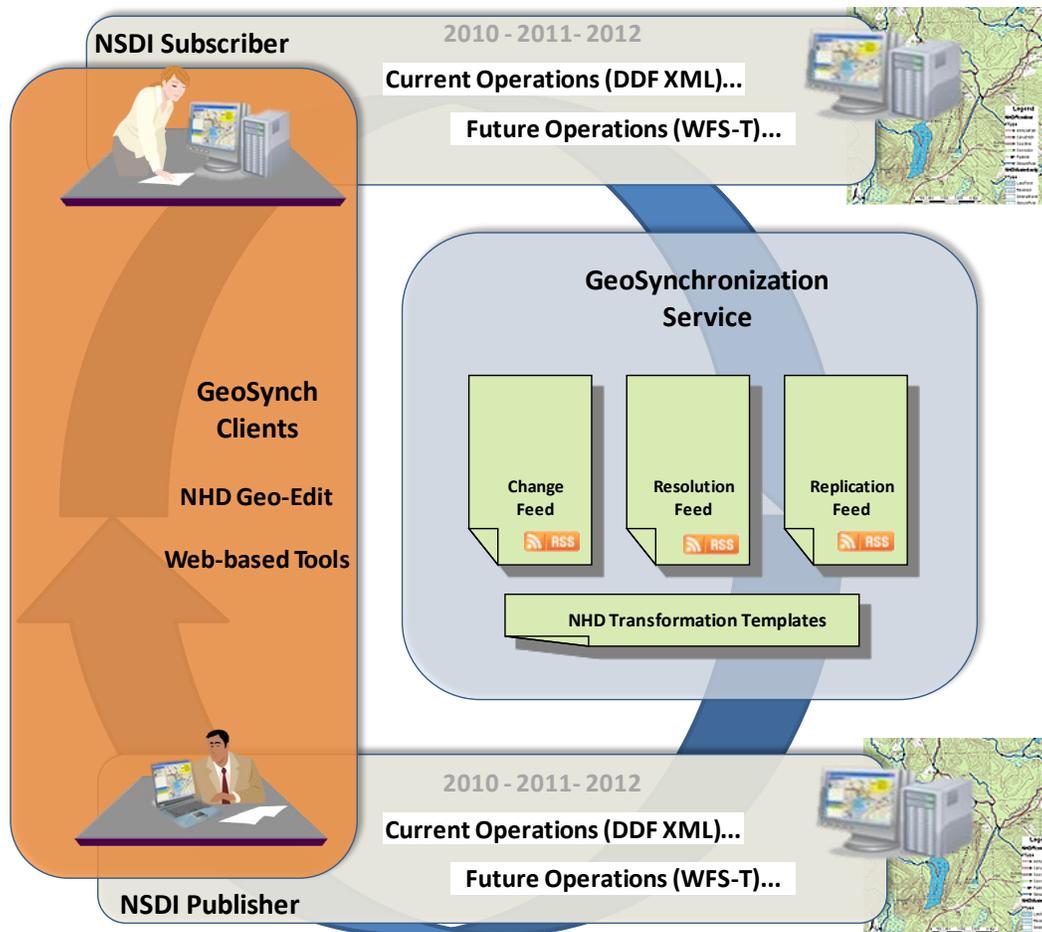


Figure 5 – This project developed Geo-Synchronization services and applications, and provides a model to bridge the gap between current and future NHD production operations.

The result of the project is the ability to support both current and future NHD operations using Geo-Synchronization. For example, in current operations, changes are made to a local copy of NHD, implemented in geodatabases using the Geo-Edit tool in ESRI ArcGIS. The output of this tool is an NHD Data Delivery Format (DDF) XML file that describes the changes made. The methodology to support this “current” NHD environment through Geo-Synchronization was described in the following Use Case, developed at the start of the project:

Use Case 1 – Enhancing Current NHD Operations with GeoSynchronization:

- 1) A user in NH with Geo-Edit publishes a DDF to the GSS using a web form. The web form not only uploads the DDF file to the GSS, but also converts it to a WFS Transaction using XSLT.*
- 2) The custodian at NH (i.e. the subscriber), using a web client, reads the proposed change from the change feed.*

3) The web client displays the proposed change so the custodian can determine if the change is acceptable. The custodian can optionally modify the proposed change using the web client.

4) If the custodian rejects the change the “reject” option is selected and an entry is put into the resolution feed to notify the user that their change request has been rejected.

5) If the custodian accepts the change the “accept” option is selected and the web client posts the WFS transaction to the NH WFS-T on ArcGIS. The web client will put an entry into the resolution feed to notify the user that their change has been accepted, and create an entry in the replication feed containing the DDF.

NOTE: The web client can also create an entry in the replication feed containing the WFS transaction ... for third parties wishing to replicate via WFS.

6) USGS can access the DDF from the replication feed and apply it to their NHD.

To support this ‘current operations’ Use Case the project developed and delivered the capability to transform NHD Data Delivery Format (DDF) XML produced by ArcGIS geo-edit tools into Web Feature Service Transactions (WFS-T) used in Geo-Synchronization. For example, a user on ArcGIS can work with NHD and create a DDF file. Then, using a web based client developed in this project they can select and upload the DDF to the Geo-Synchronization Service (Figure 6). The data is then translated into WFS Transactions, goes to Review and if accepted gets applied to New Hampshire WFS, also to USGS WFS data (Figure 7).

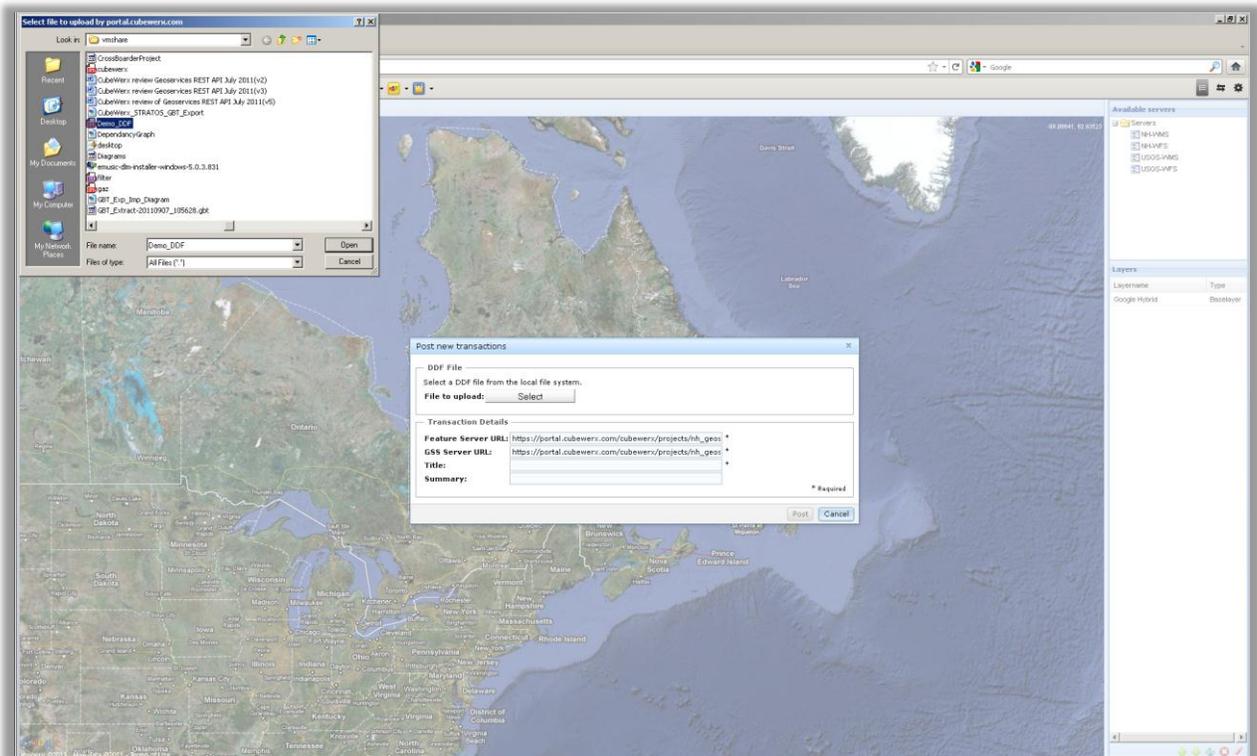


Figure 6 – To support ‘current operations’ the project successfully developed and delivered the capability to upload and transform NHD Data Delivery Format (DDF) XML produced by ArcGIS geo-edit tools into Web Feature Service Transactions (WFS-T) used in Geo-Synchronization. Uploading a DDF is shown above.

This approach allows NHGS and other organizations in the NSDI to leverage current NHD business processes and Geo-Synchronization without disruption.

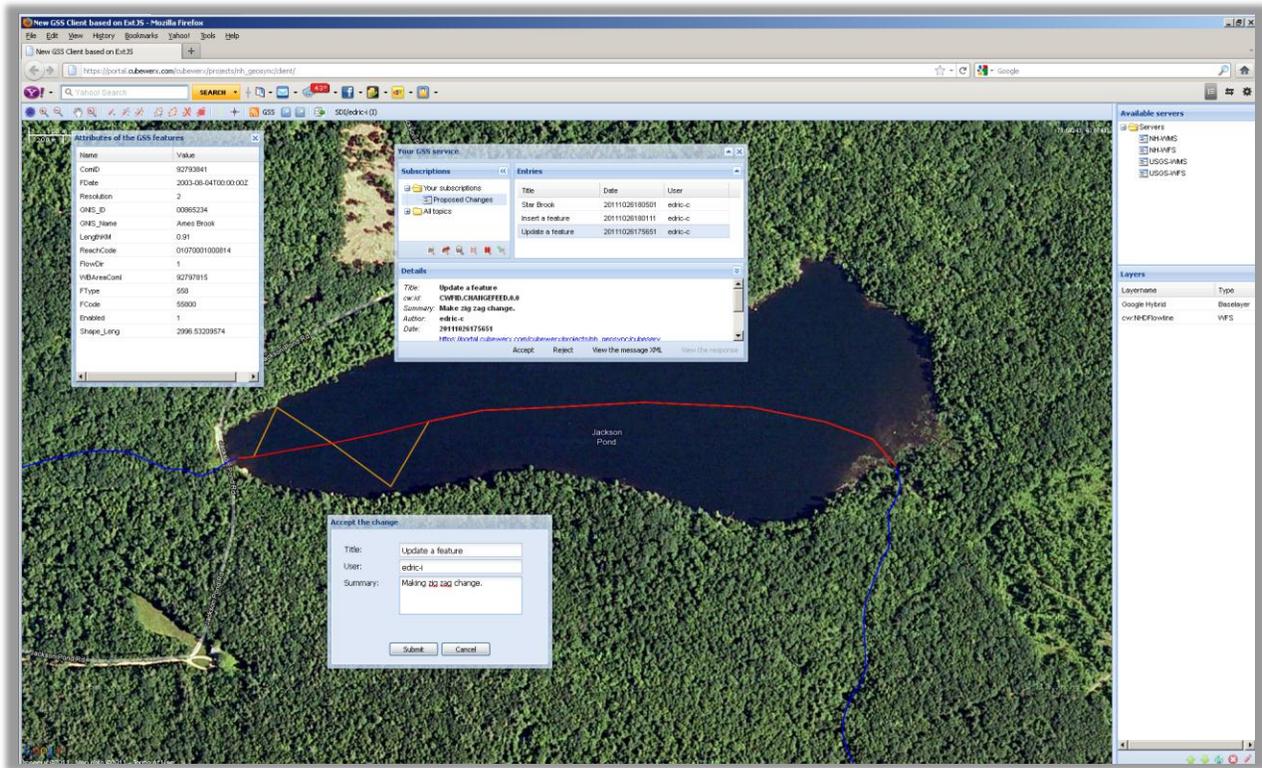


Figure 7 – Once uploaded and transformed by the GSS the DDF data may be reviewed like any other proposed WFS update, and sent to state and national level WFS via the Geo-Synchronization Service.

In addition to support for current operations the project developed tools and capability to support future Geo-Synchronization operations - where changes are made to NHD with easy-to-use interoperable Web-based tools, and direct WFS Transactions are shared to describe the changes. This Use Case is summarized below:

Use Case 2 - Enhancing Future NHD Operations with GeoSynchronization:

- 1) Field User reads NHD data via NH WFS-T with an OpenLayers web tool developed by CubeWerx.*
- 2) Field User publishes updates to the GeoSynchronization server's change feed using the web applications.*
- 3) The custodian at NH (i.e. the subscriber), using a web client, reads the proposed change from the change feed.*
- 4) The web client will display the proposed change so the custodian can determine if the change is acceptable or not. The custodian can optionally modify the proposed change using the web client.*
- 5) If the custodian rejects the change an entry is put into the resolution feed to notify the user that their change request has been rejected.*

6) If the custodian accepts the change the web client will post a WFS transaction to the NH WFS-T on ArcGIS. The web client will put an entry into the resolution feed to notify the user that their change has been accepted. The web client will create an entry in the replication feed containing the WFS transaction.

7) USGS can access the WFS transaction from the replication feed and apply it to their NHD WFS server. Optionally, the DDF can also be maintained in the replication feed of the GeoSynchronization server and used by USGS.

To support this ‘future operations’ Use Case the project developed and delivered a Web-based GSS Client (Figure 8). This client included easy-to-use web tools for publishing and reviewing NHD edits from online users - providing the potential for ‘crowdsourced’ updates of selected hydrography data using GeoSynchronization. The main components of the Web-based GSS Client include:

- Display map for WMS and WFS with scale-sensitive display for performance, and geometry and feature editing and publishing capability
- RSS Feed Manager to access Change, Resolution, Replication Feeds, also Topics
- Reviewer tools to accept or reject proposed NHD updates
- Server manager panel to manage user roles and access to OGC WMS and WFS via GSS
- DDF management component (described above)

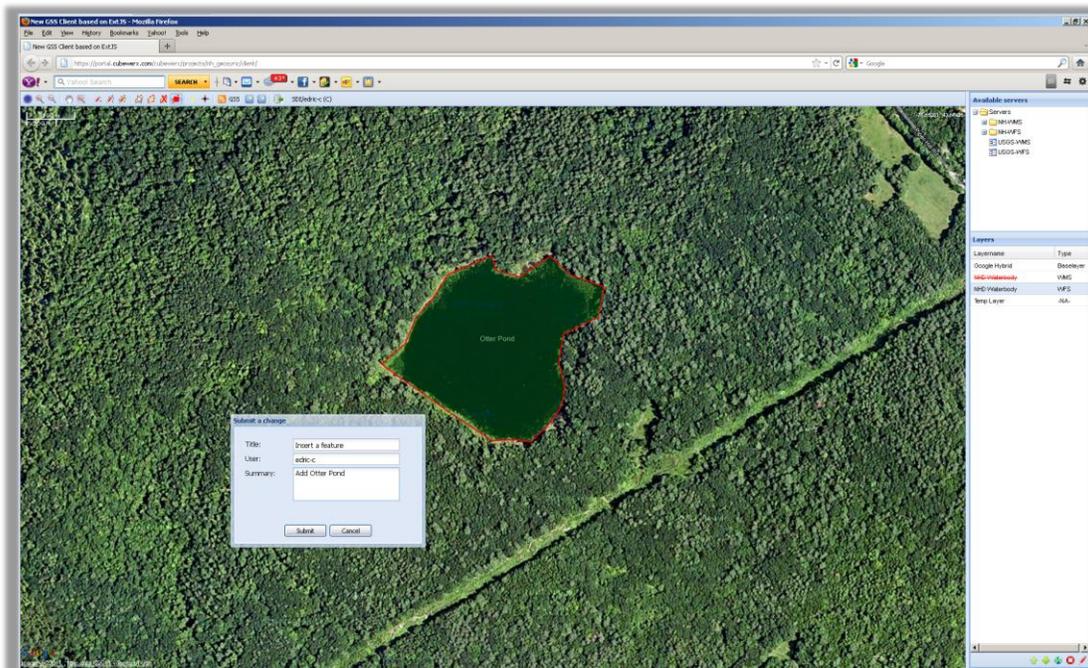


Figure 8 – ‘GeoSync’ web tools developed in this project will allow anyone to contribute potential NHD updates – such as ‘Otter Pond’ above.

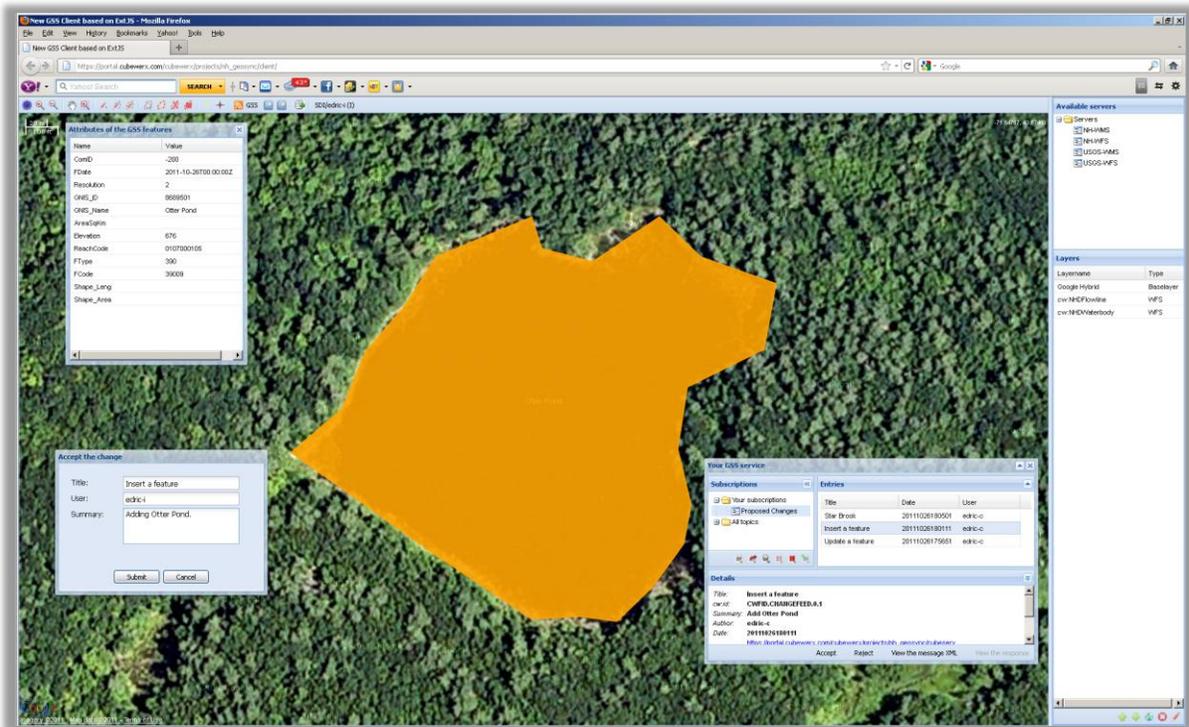


Figure 9 – Updates such as ‘Otter Pond’ can be reviewed before being sent to state and national level WFS

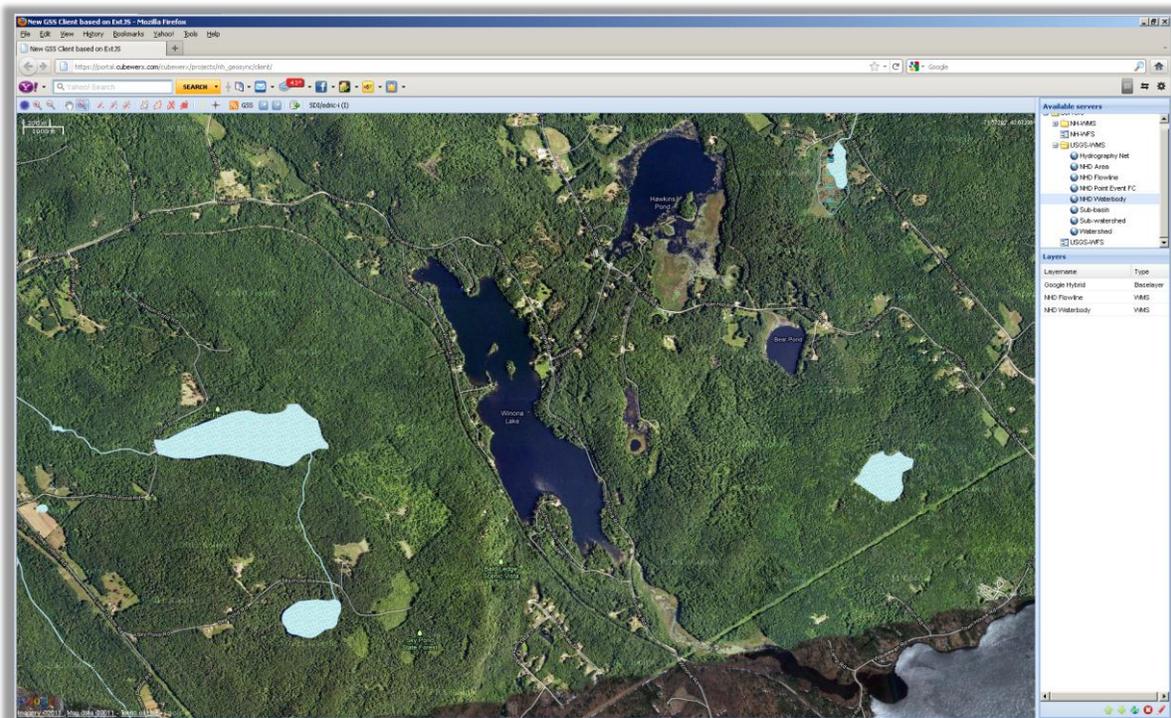


Figure 10 – The project architecture supports both DDF and Web-based edits for NHD Geo-Synchronization. Edits are made available as both WFS and WMS services.

NOTE - The Web-based GSS Client is context sensitive, which means that different tools are presented to the user based on their Geo-Synchronization role (i.e. Data Publishers, Reviewers or Data Managers). Roles are determined by the Identity Management Service deployed for the project (see Appendix A).

Finally, the project configured and deployed multiple WFS and Web Map Services (WMS) with state and federal Framework NHD to support operations testing and deployment. The WFS deployed provided full support to Transactions and the WMS provided fast navigation into NHD data through the use of scale-dependent display of the data.

“Most Crowdsourcable” NHD Features

The Web-based GSS Client developed during this project is based on capability refined and tested in early 2010 as part of the Open Geospatial Consortium (OGC) OWS-7 testbed². As described above, this client focuses on providing simple tools to allow publishing and review of NHD feature types. Specifically, during the project the following features were identified as the having the most potential for broad community engagement and update using GeoSynchronization (i.e. the “most crowdsourcable”):

- NHDArea
- NHDWaterbody
- NHD Flowlines
- NHDEvent
- Geonames associated with all of the above feature types

As such, the intent of the Web-based GSS Client application is to allow more people contribute to NHD. The project team assessed that these people will most likely be “professionals” (for example, people doing environmental assessments etc.). The capability has been designed to support updates to both new features and unnamed features, or features that have been so significantly modified since the NHD was created.

Project Tasks

The project was conducted as a series of collaborative tasks to develop and deploy the Geo-Synchronization capability, supporting WFS and WMS and engage and train community stakeholders:

- **Development, Deployment and Sustainment** – These tasks developed, deployed and sustained services and applications in New Hampshire to synchronize NHD updates into an authoritative State data layer, which can then be provided for integration at national levels. Services and applications deployed included WFS, WMS, GSS and Web-based client tools to support both current and future NHD operations simultaneously.
- **Community Engagement** – These tasks defined and executed communication and outreach activities to NSDI stakeholders – including the conduct of NHD orientation and training sessions in New Hampshire held to obtain a better understanding of NHD data production processes and editing rules and educate users on the tools being made available by the project.

² <http://www.youtube.com/watch?v=9ysWfbr4RSc>

- **Project Coordination** – This task enabled coordination between project staff, participants and the NSDI community. Specific activities include management, documentation, meetings and maintaining online collaborative environments.

This approach allowed stakeholders to implement GeoSynchronization with minimal change to existing technologies and planned development.

In summary, progress on this project advances a realistic approach to bridge the gap between current NHD production operations and the FGDC vision of developing, deploying and sustaining a federation for exchanging local, state and national framework data using GeoSynchronization services.

Feedback on Cooperative Agreements Program

The NSDI Cooperative Agreements Program provided an opportunity for the project team to work with NSDI practitioners from New Hampshire and other organizations on an initiative that has real world implications in much of the U.S NSDI. 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 services and simulate deployment scenarios with other team members, an effort that came out of discussions and regular teleconferences with the CAP government team. The program's mixture of federal government, enterprise, and local government teams was also very beneficial. Overall, the program is making very good progress towards promoting key aspects of realizing NSDI services online. Continued emphasis needs to be placed on promoting an online infrastructure of standards-based location content across the nation that can flexibly support operational requirements, and governance of resulting standards for information sharing. With the progress on Geo-Synchronization exemplified by efforts such as those outlined in this report we can identify few technical impediments to advancing such an infrastructure. However, we suspect funding issues are holding back development of this online infrastructure – which is difficult to understand since geo-synchronization offers one of the most compelling methods to save government money by turning geospatial stovepipes into interactive community data networks.

Weaknesses: Although not a weakness, additional external Federal engagement (i.e. outside FGDC) in project continuation and partnering efforts should continue to be encouraged. This is occurring but agencies such as USDOT and others can benefit from CAP solutions such as those developed in this project - and should continue to engage more in the process. Specifically, the CAP needs to have continued strong liaison in operational aspects of these agencies and with other state, federal, and commercial interests.

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 greatly facilitated collaborations.

Appendix A - Technical Architecture

The technical approach for this project is based on the Open Geospatial Consortium (OGC) draft specification for GeoSynchronization Services (GSS) developed and authored by CubeWerx and The Carbon Project, and the CubeSERV[®] CubeWerx GSS³. The GSS uses the ATOM Publishing Protocol to connect to a WFS-T and messages based on Atom format encodings. GSS is a powerful capability that is agnostic to the underlying database; it simply interfaces with the WFS-T service - and includes the ability to transform DDF XML produced by current NHD geo-edit tools to WFS feature transactions to support current and future NHD operations simultaneously.

The GSS for this project uses Atom Publishing Protocol and Publish/Subscribe interfaces to:

- ✓ advertise changes made to a geospatial database
- ✓ create and edit GeoRSS feeds in support of data currency and replication
- ✓ alert subscribers to the service of proposed changes, resolutions to those changes, and replication transactions available
- ✓ perform queries on the feeds

In addition, the GSS included capabilities to bridge the gap between the current needs of state data production organizations and the geo-synchronization vision outlined by the FGDC, including:

- ✓ capability to transform Data Delivery Format XML produced by current NHD geo-edit tools to Web Feature Service Transactions (WFS-T) in order to support current and future NHD operations simultaneously

The GSS uses *feeds*, managed by a web service to support data currency & replication capabilities in a federated environment (Figure 11). The GSS uses a “Publish-Subscribe” model where feeds are published by one data source and subscribed to by another data source. Publishers adds entries to feeds for changes made to a database, and Subscribers receives alerts when new transactions have occurred, retrieves contents, and applies them to the database. The web service is implemented according to the OGC draft specification for GeoSynchronization Services (GSS).

³ <http://www.cubewerx.com/products/gss>

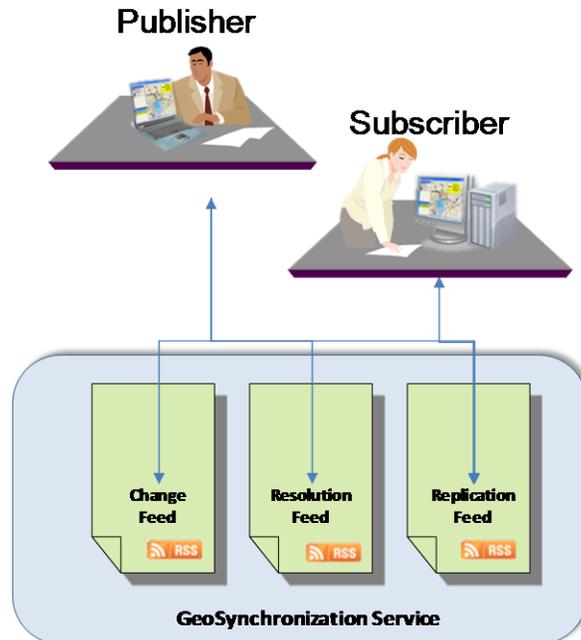


Figure 11 – A GSS uses *feeds*, managed by a web service called a GeoSynchronization Service

In the deployment, three major roles are supported by the GSS (Figure 12) –

- **NHD Data Collector (Publisher)** – This role is in charge of making changes to content. This person or system generates feature changes and submits them for review in the **Change GeoRSS Feed**. The changes include adding new features, deleting features, updating existing features. The changes are then submitted to the GSS and may be transformed from NHD DDF as needed. When a change is accepted or rejected the Collector is notified via a Resolution GeoRSS feed. In case of a rejection the Collector can correct and adjust according to the Integrator comments.
- **NHD Data Integrator (Reviewer)** - This role reviews, accepts or rejects updates proposed by Collectors. The Integrator submits the approvals or rejections to the GSS using a standard GeoRSS-type message and the **Resolution Feed**. Following acceptance of the proposed updates, the WFS database is updated by the integrator and the interaction tools automatically submits an RSS feed with encoded feature transactions DDF XML so other organizations can monitor the progress. These transactions or XML files may also be used for data replication by other collaborators.
- **NHD Data Follower** – This role follows accepted changes via the **Replication Feed**. A Follower subscribed to these event notifications will receive the appropriate updates in the form of RSS or GeoRSS entries.

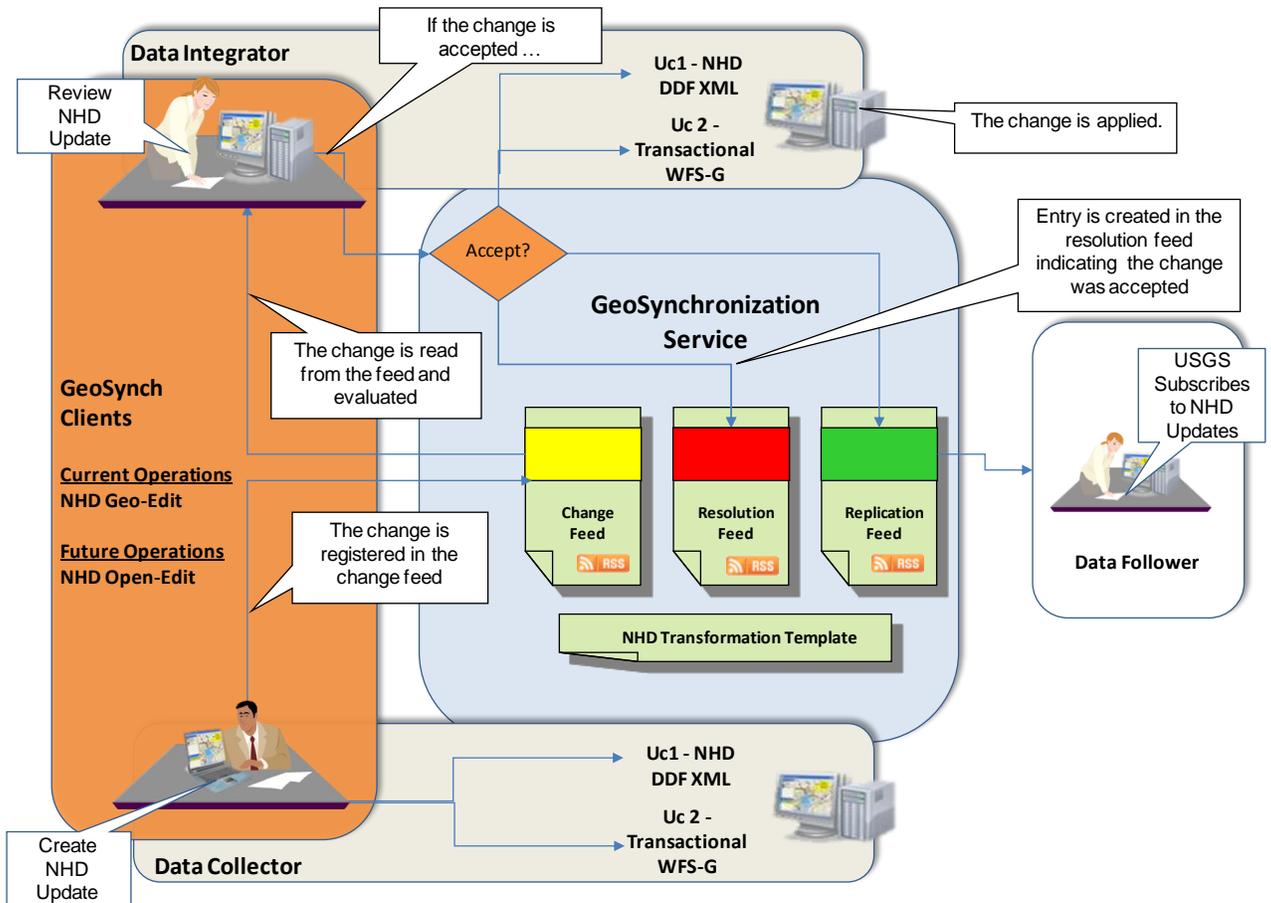


Figure 12 – The proposed Geo-Synchronization capability will facilitate maintenance and update of NHD into authoritative State data layers, which can then be provided for integration into the NSDI.

The approach includes an Identity Management Service to provide role-based access control (see Figure 13).

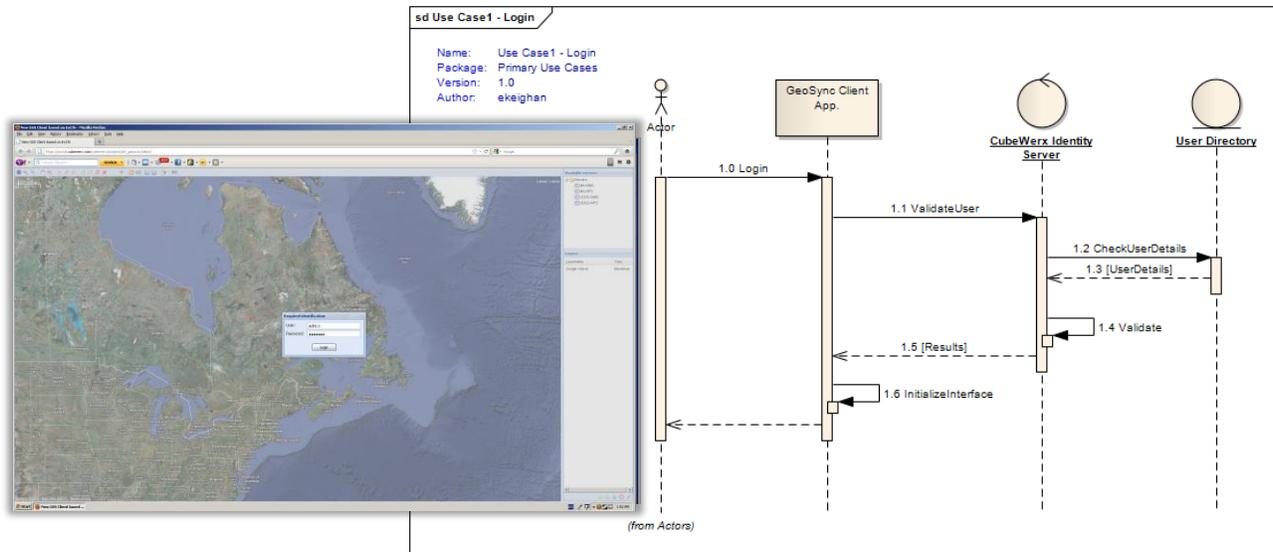


Figure 13 – Our deployment includes Role-based Access Control for security.

The NHHD GSS includes the capability to AccessGeographicFeatures, EditGeographicFeatures, ApplyFeatureChanges and other functions (Figure 14). Finally, the GSS includes the capability to validate proposed Changes (Figure 15). This workflow includes the ability to interact with the GSS and AccessProposedChangeFeed, RequestChangeFeed, ValidateProposedChange, SelectFeaturesFromWFS, AccessProposedChange, PerformUpdateTransaction, SubmitResolutionFeed, SubmitReplicationFeed, RefuseProposedChange, and SubmitResolutionFeed.

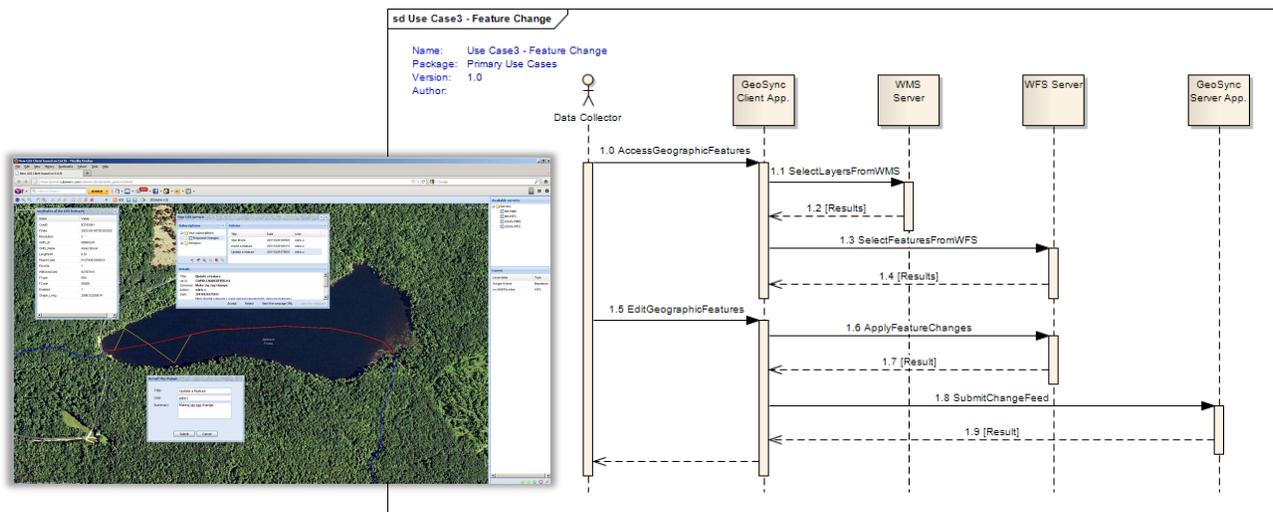


Figure 14 - The New Hampshire GSS supports publishing of collaborative updates and changes

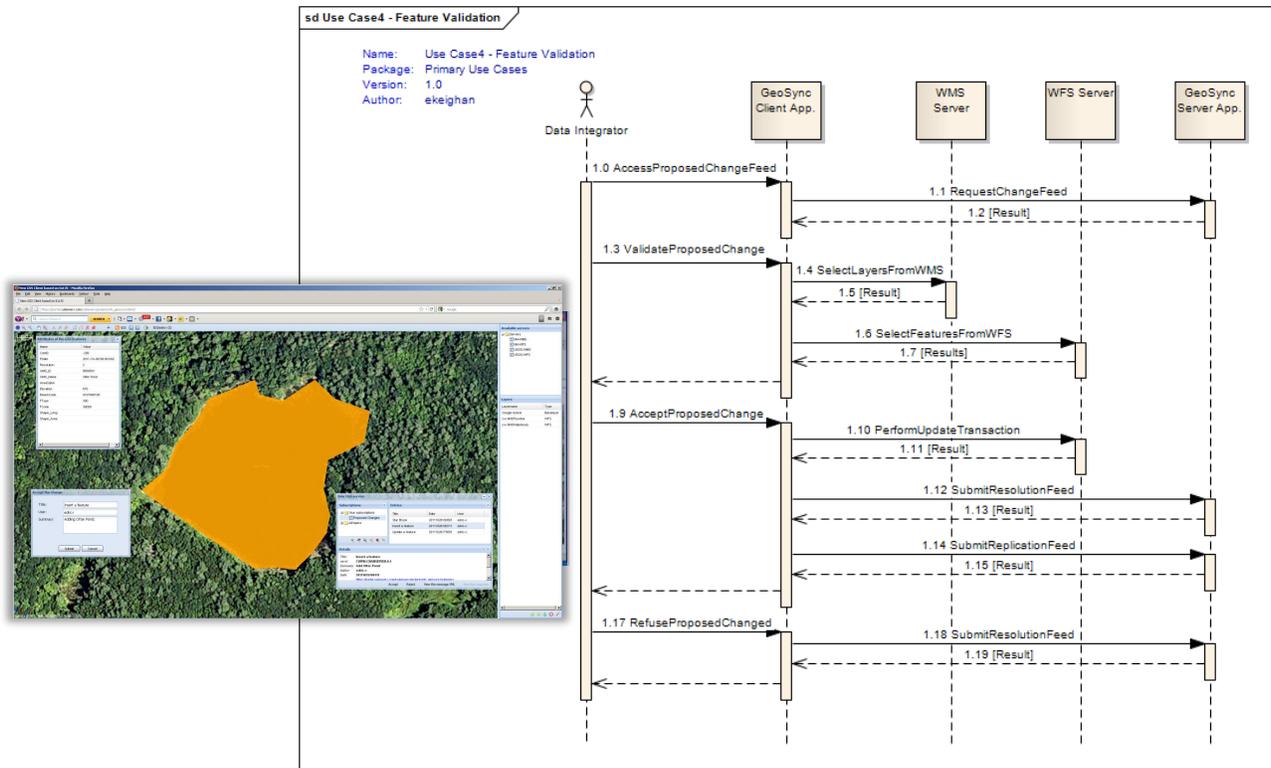


Figure 15 - The New Hampshire GSS includes the ability to support validation of proposed data

GeoSynchronization Test Environment

To support project development a GeoSynchronization test environment consisting of WFS, GSS, application clients and NHD data was established. This test environment follows the basic architecture of a collaborative SDI based on GeoSynchronization and includes roles for publishing, reviewing and following proposed NHD changes (Figure 16).

As part of the test environment, the project team identified and configured machines to host Framework NHD and to host the NH NHD data. A GSS was developed and deployed to support geosynchronization between feature types from the USGS machine and the NH machine.

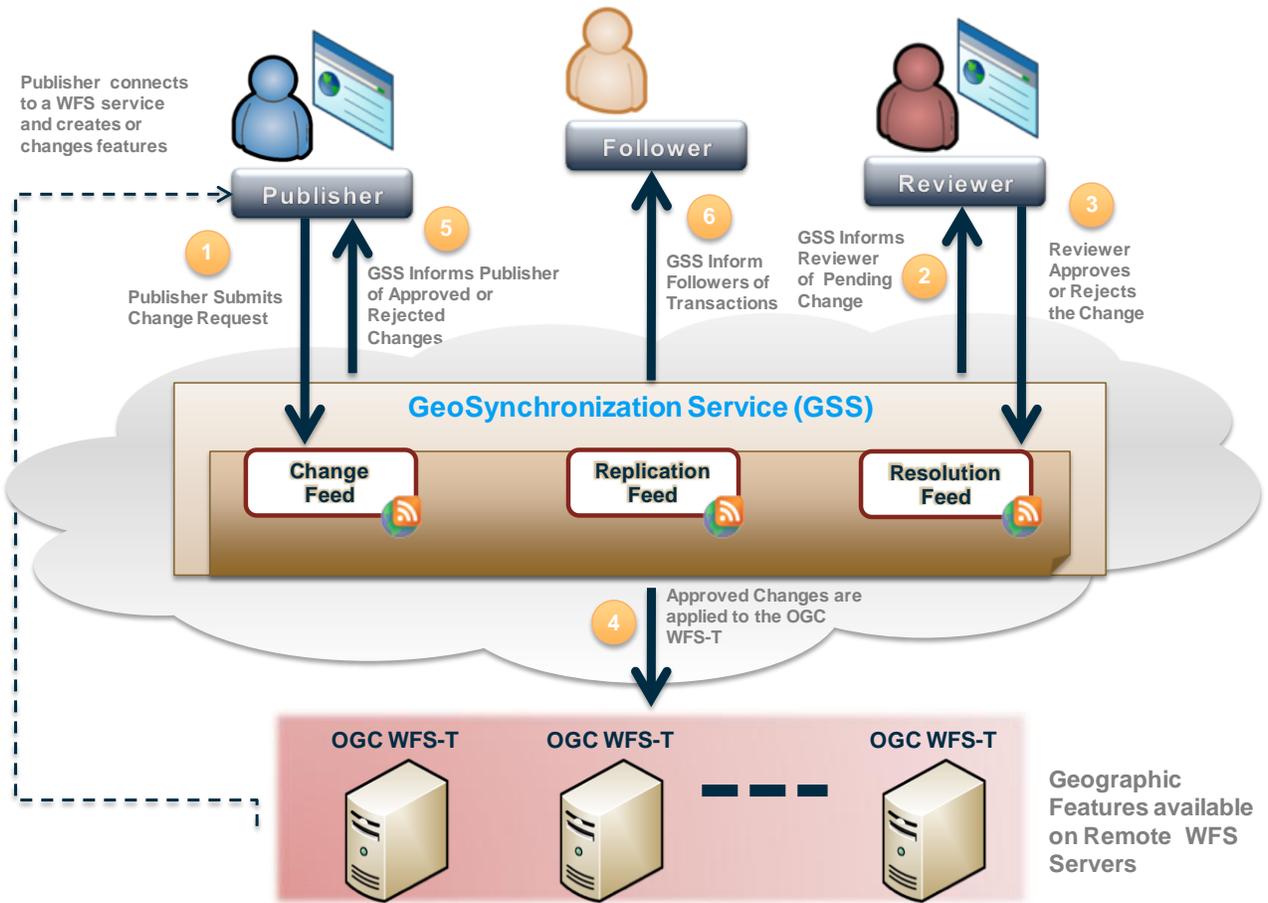


Figure 16 – Overview Architecture for GeoSynchronization Test Environment

Appendix B: Converting DDF XML to GeoSynchronizable WFS Transactions

Real-world deployment of GeoSynchronization for NHD requires converting DDF XML (see Appendix C) into WFS transactions. In our approach, DDF is converted to WFS transactions using XSLT (XSL Transformations) - a declarative language that transforms XML documents into other XML documents. We use XSLT to provide a bridge from current production operations to future operations using WFS-T.

To convert NHD DDF XML we note the format contains three classes of operations:

FEATURE MANAGEMENT,
RELATIONSHIP MANAGEMENT and
METADATA MANEGEMENT.

The key transformation parameters for NHD DDF to WFS-T are summarized below:

FEATURE MANAGEMENT:

These commands are used to Insert, Update or Delete features and map directly into the following WFS Transaction actions:

CreateFeature -> WFS Insert into the feature table
DeleteFeature -> WFS Delete from the feature table
ModifyFeature -> WFS Update to the feature table

RELATIONSHIP MANAGEMENT:

These commands are used to create relationships between features. They map to WFS Transaction action into the NHDVerticalRelationship tables.

CreateRelationship ->
WFS Insert into NHDVerticalRelationship or NHDVerticalRelationship
DeleteRelationship ->
WFS Delete from NHDVerticalRelationship or NHDVerticalRelationship
ModifyRelationship ->
WFS Update on NHDVerticalRelationship or NHDVerticalRelationship

METADATA MANAGEMENT:

This command creates metadata about features and organizations that are inserting/updating/deleting data. There is one command that maps to a WFS insert.

CreateMetadata -> WFS Insert into the NHDMetadata and NHDSourceCitation tables

All operations are based on object id which maps directions into a WFS filter. Here is an example:

```
<DeleteFeature>  
<ID>113794781</ID>
```

```
</DeleteFeature>
```

Which becomes:

```
<Delete>  
  <Filter>  
    <FeatureId fid="113794781"/>  
  </Filter>  
</Delete>
```

To date the project team has focused on writing an XSLT to transform (not transpose) the DDF to a geosynchronizable WFS transaction. *We were originally targeting the ANSI framework schema as the transactional schemas (since this is what the grant proposal indicated) but this approach has been adjusted to additionally support the Geodatabase NHD schema. This approach will provide even more opportunity to allow users to leverage current business processes without disruption.*

The DDF assessment was ongoing throughout the project and appears to be easily transformable into WFS transactions. During this process we identified and forward multiple questions to NHD experts. For example:

- What does a ModifyVAA that does not seem to modify any properties (except maybe FDATE) mean? Do empty elements mean that the corresponding table property be set to NULL or just not changed?
- Does the <ID> element map to the OBJECTID or some other NHD identifier?
- What does a negative id mean on the CreateFeature operation? That the id will be assigned upon creation and how is that done?
- What does the element Dimensionality mean? In cases it is 3 other 4.
- Does HYC = Hydrographic Category?

In addition, other questions were forward to NHD experts as part of Community Outreach activities. These questions were reviewed during email exchanges and telecons (ongoing). In addition, the questions were presented during monthly group telecons with other GeoSynchronization project in the 2010 NSDI CAP.

Finally, an HTML web form was written that will allow a user to upload a DDF file to the GSS. In the process of being uploaded it will be transformed into a WFS transaction (see previous sections).

Appendix C: Data Delivery Format XML

NHDGDB Input XML File Specification

All data transactions shall be in Extensible Language (XML) format, as defined in The NHDGDB Input XML File Specification below.

XML Overview

XML stands for **EX**tensible **M**arkup **L**anguage. The XML file format is required to load data to the NHD geodatabase. It is a markup language much like HTML that was designed to describe data. The syntax rules of XML are very simple and very strict. Below is a sample XML file.

```
<?xml version="1.0"?>
<Transaction>
<GDBVersionInformation>NHD20040728</GDBVersionInformation>
<ProcessingOrganization>USGSMCMC</ProcessingOrganization>
<Resolution>High</Resolution>
<CreateMetadata>
  <ID>-2</ID>
  <Attribute>
    <POD>Deleted overlapping reach in NHDFlowline to join with
    adjacent subbasin.</POD>
    <PDA>20040727</PDA>
  </Attribute>
</CreateMetadata>
<DeleteFeature>
  <ID>113794769</ID>
</DeleteFeature>
</Transaction>
```

The first line in the document - the XML declaration - defines the XML version. The next line describes the root element of the document. The next lines describe the 5 child elements of the root. And finally the last line defines the end of the root element.

All XML elements must have a closing tag. XML tags are case sensitive. All XML documents must contain a single tag pair to define a root element - <Transaction></Transaction> for NHDGDB XML files.

Required Elements of a NHDGDB XML File

Every XML file must start with the XML declaration line: <?xml version="1.0"?>

The second line of the XML file must be the opening tag of the root element: <Transaction>

The last line of the XML file must be the closing tag of the root element: </Transaction>

Every XML file must contain the following child elements:

GDBVersionInformation

Opening/Closing Tag: <GDBVersionInformation></GDBVersionInformation>

Description: The version of the GDB the updates will be applied to. If you are loading only new data then the version will be NewLoad. If you are loading updates to existing data the version is the version found in the NHDProcessingParameters table in the distributed personal geodatabase.

Example:

<GDBVersionInformation>NHD20040728</GDBVersionInformation>

Processing Organization

Opening/Closing Tag: <ProcessingOrganization></ProcessingOrganization>

Description: The processing organization loading the data. This must be the same as the processing organization that allocated the reach codes used in the load file.

Example:

<ProcessingOrganization>USGSRMMC</ProcessingOrganization>

Resolution

Opening/Closing Tag: <Resolution></Resolution>

Description: The resolution of the data being loaded. Valid values are Local, High or Medium.

Example:

<Resolution>High</Resolution>

CreateMetadata

Opening/Closing Tag: <CreateMetadata></CreateMetadata>

Description: This transaction contains the data that was in the CMDI transaction of the FCP file. The attributes are loaded to the NHDMetadata and NHDSourceCitation tables. The CreateMetadata element contains 2 child elements - <ID></ID> and <Attribute></Attribute>. The ID element contains a numeric identifier that is unique across all CreateMetadata transactions. If there are multiple CreateMetadata transactions in the XML file this identifier is used in the CreateRelationship transactions that specify Feature-to-Metadata relationships. This identifier is re-assigned when it is loaded into the geodatabase and stored in the NHDMetadata table as the DUUID.

Attributes:

XML Tag	Field Name
AAR	AttributeAccuracyReport
LCR	LogicalConsistencyReport
COR	CompletenessReport
HOR	HorizPositionalAccuracyReport
VOR	VertPositionalAccuracyReport
POD	ProcessDescription
PDA	ProcessDate
MED	MetadataDate
MSN	MetadataStandardName
MSV	MetadataStandardVersion
DSC	DatasetCredit
COO	ContactOrganization
ADT	AddressType
ADD	Address
CIT	City
STP	StateOrProvince
PSC	PostalCode
CVT	ContactVoiceTelephone
COI	ContactInstructions
SourceCitation	--

There can be multiple SourceCitation elements. Each element is stored as a record in the NHDSourceCitation table. The DUUID column of the NHDSourceCitation record will be the DUUID assigned to the NHDMetadata record. The SourceCitation element can have the following child elements:

XML Tag	Field Name
TIT	Title
SCA	SourceCitationAbbreviation
ORG	Originator
PUD	PublicationDate
BED	BeginningDate
END	EndingDate
SRC	SourceContribution
SSD	SourceScaleDenominator
TSM	TypeOfSourceMedia
CAD	CalendarDate
SCR	SourceCurrentnessReference

The ProcessDescription attribute (POD) is the only required attribute in the CreateMetadata transaction. If an attribute is null it does not need to be included in the XML file.

Example:

```

<CreateMetadata>
<ID>1</ID>
<Attribute>
  <MSV>FGDC-STD-001-1998</MSV>
  <MSN>FGDC Content Standard For Digital Geospatial Metadata, ver.2</MSN>
  <MED>20040512</MED>
  <COO>U.S. Geological Survey</COO>
  <COI>Monday-Friday, 7AM-3PM CST</COI>
  <CVT>(573) 308-3647</CVT>
  <PSC>65401</PSC>
  <STP>Missouri</STP>
  <CIT>Rolla</CIT>
  <ADD>Mid-Continent Mapping Center - 1400 Independence Road</ADD>
  <ADT>Mailing and Physical Address</ADT>
  <POD>Create high-resolution NHD from revised DLG data and U.S. Forest Service Cartographic
  Feature Files (CFFs).</POD>
  <PDA>20040420</PDA>
  <SourceCitation>
    <SCA>NHD basic features</SCA>
    <SCR>Date the revision process step was completed</SCR>
    <CAD>20040405</CAD>
    <TSM>online</TSM>
    <SSD>24000</SSD>
    <SRC> spatial and attribute information</SRC>
    <END>Not Applicable</END>
    <BED>Not Applicable</BED>
    <PUD>unpublished materials</PUD>
    <TIT>NHD basic features</TIT>
    <ORG> U.S. Geological Survey </ORG>
  </SourceCitation>

```

```

<VOR>Statements of vertical positional accuracy for elevation of
water surfaces are based on accuracy statements made for U.S.
Geological Survey topographic quadrangle maps...</VOR>
<HOR>Statements of horizontal positional accuracy are based on accuracy statements made for U.S.
Geological Survey topographic quadrangle maps. ...</HOR>

<COR>The completeness of the data reflects the published USGS
topographic quadrangle and/or the U.S. Forest Service Primary Base
Series (PBS) map...</COR>
<LCR>For DLG data, points, nodes, lines, and areas conform to
topological rules...</LCR>
</Attribute>
</CreateMetadata>

```

Other Elements of a NHDGDB XML File

The other elements that may be included in the NHDGDB XML file are used to create, modify or delete features and reaches and to create, modify, or delete relationships.

CreateFeature

Opening/Closing Tag: <CreateFeature></CreateFeature>

Description: This transaction is used to create a feature in a feature class or create a reach in the NHDReachcodeComId table. It contains five child elements:

- 1) ID – a numeric identifier that is unique across all features in the XML file. This identifier will be reassigned a unique comid when the feature is loaded to the geodatabase.
- 2) FeatureType – the feature type of the feature
- 3) Dimensionality – a number representing the feature’s dimension: 2 for point, 3 for line, 4 for polygons, or 5 for reaches
- 4) Coordinate – This element contains the collection of points that define the geometry of the feature. Each point is a child element of the Coordinate element. The format for the Point element is

```
<Point X="-89.3687125" Y="46.8583277"/>
```
- 5) Attribute – This element contains the attributes for the feature or reach. Note that not all attributes in the list below will apply to every feature type. There are several attributes that are not stored as fields in the feature class. These attributes along with the feature type are used to derive the FCODE attribute.

XML Tag	Field Name
RSL	Resolution
GID	GNIS_ID
NAM	GNIS_NAME
FLD	FlowDir
ELE	Elevation
RCH	Reachcode
OWT	WBAreaComId
ICS	(used to derive fcode)
RET	(used to derive fcode)
COM	(used to derive fcode)
HYC	(used to derive fcode)
STG	(used to derive fcode)
PIT	(used to derive fcode)
RTS	(used to derive fcode)
SZT	(used to derive fcode)

OPS	(used to derive fcode)
POA	(used to derive fcode)
CDY	(used to derive fcode)
CGC	(used to derive fcode)

Each of the 5 child elements listed above is required to create a feature. The Coordinate element does not apply to reaches. There are no required attributes. If an attribute is null it does not need to be included in the XML file. If there are no attributes for the feature, the Attribute element can be empty –

```
<Attribute></Attribute>
```

Examples:

```
<CreateFeature>
<ID>2741</ID>
<FeatureType>460</FeatureType>
<Dimensionality>3</Dimensionality>
<Coordinate>
  <Point X="-89.357778" Y="46.8515609"/>
  <Point X="-89.3578384" Y="46.8515731"/>
  <Point X="-89.3578863" Y="46.8515721"/>
  <Point X="-89.3579428" Y="46.85158"/>
  <Point X="-89.3580227" Y="46.8515783"/>
  <Point X="-89.3619394" Y="46.852555"/>
</Coordinate>
<Attribute>
  <HYC>289</HYC>
  <FLD>1</FLD>
  <RSL>2</RSL>
</Attribute>
</CreateFeature>
```

```
<CreateFeature>
<ID>2</ID>
<FeatureType>557</FeatureType>
<Dimensionality>5</Dimensionality>
<Attribute>
  <RCH>04020101000002</RCH>
</Attribute>
</CreateFeature>
```

ModifyFeature

Opening/Closing Tag: <ModifyFeature></ModifyFeature>

Description: This transaction is used to modify an existing feature in a feature class or modify a reach. The ModifyFeature transaction contains the same elements as the CreateFeature transaction. The ID element is the comid of the feature or reach to modify. The Coordinate element only needs to be provided for a feature if the geometry of that feature has changed.

Example:

```
<ModifyFeature>
<ID>49081000</ID>
<FeatureType>334</FeatureType>
<Dimensionality>3</Dimensionality>
<Attribute>
  <GID>01552301</GID>
  <NAM>New River</NAM>
</Attribute>
```

</ModifyFeature>

DeleteFeature

Opening/Closing Tag: <DeleteFeature></DeleteFeature>

Description: This transaction is used to delete a feature from a feature class or delete a reach from the NHDReachcodeComid table. It contains one child element (ID) that specifies the comid of the feature or reach to be deleted. When a feature is deleted, all vertical relationships for that feature are also deleted. When a reach is deleted, the reachcode, GNIS_id, and GNIS_name attributes of all features related to the reach are deleted (set to null).

Example:

```
<DeleteFeature>
  <ID>113794781</ID>
</DeleteFeature>
```

CreateRelationship

Opening/Closing Tag: <CreateRelationship></CreateRelationship>

Description: This transaction is used to create composed-of relationships, vertical relationships, or feature-to-metadata relationships. It contains 4 required child elements.

- 1) ID – a numeric identifier
- 2) Object1 – For composed-of relationships this is the comid or temporary id of the reach. For vertical relationships this is the comid or temporary id of the 'abovecomid'. For feature-to-metadata relationships this is the temporary duuid found in the CreateMetadata transaction.
- 3) Object2 - For composed-of relationships this is the comid or temporary id of the basic feature. For vertical relationships this is the comid or temporary id of the 'belowcomid'. For feature-to-metadata relationships this is the temporary id of the feature.
- 4) TypeRelationship – specifies the relationship type: 1 for composed-of relationships, 5 for vertical relationships, and 7 for feature-to-metadata relationships.

The NHDGDB load software does not load flow relationships into the geodatabase. Creating a composed-of relationship will cause the feature to have the same reachcode attribute as the reach. If the reach has the GNIS_ID and GNIS_NAME attributes populated, the GNIS attributes of the feature will be updated to be the same as the GNIS attributes of the reach.

Creating a vertical relationship adds a record to the NHDVerticalRelationship table.

Creating a feature-to-metadata relationship adds a record to the NHDFeatureToMetadata table.

Example:

```
<CreateRelationship>
<ID>1</ID>
<Object1>6</Object1>
<Object2>2737</Object2>
<TypeRelationship>1</TypeRelationship>
</CreateRelationship>
```

ModifyRelationship

Opening/Closing Tag: <ModifyRelationship></ModifyRelationship>

Description: This transaction is used to modify composed-of relationships. This transaction can not be used to modify vertical relationships or feature-to-metadata relationships. It contains 4 child elements.

- 1) ID – the comid of the feature in the composed-of relationship
- 2) Object1 – the comid of the reach
- 3) Object2 – the comid of the feature
- 4) TypeRelationship – specifies the relationship type (1). This element is not required.

Modifying a composed-of relationship will cause the feature to have the same reachcode attribute as the reach. If the reach has the GNIS_ID and GNIS_NAME attributes populated, the GNIS attributes of the feature will be updated to be the same as the GNIS attributes of the reach.

Example:

```
<ModifyRelationship>
<ID>54487893</ID>
<Object1>54488305</Object1>
<Object2>54487893</Object2>
<TypeRelationship>1</TypeRelationship>
</ModifyRelationship>
```

DeleteRelationship

Opening/Closing Tag: <DeleteRelationship></DeleteRelationship>

Description: This transaction is used to delete composed-of relationships and vertical relationships. It contains 1 child element - ID. When deleting a composed-of relationship, ID is the comid of the feature in the composed-of relationship. When deleting a vertical relationship, the ID is the comid of the vertical relationship. Deleting a composed-of relationship causes the reachcode, gnis_id, and gnis_name attributes of the feature to be set to null.

Example:

```
<DeleteRelationship>
  <ID>36475343</ID>
</DeleteRelationship>
```

Other Notes:

With XML white space inside elements is preserved. For example, if the process description in your XML file contains leading spaces, there will be leading spaces in the process description in the geodatabase.

Indentation of child elements is not required, but may be useful for readability. For example, the following transactions are considered the same:

```
<ModifyFeature>
  <ID>49081000</ID>
  <FeatureType>334</FeatureType>
  <Dimensionality>3</Dimensionality>
  <Attribute>
    <GID>01552301</GID>
    <NAM>New River</NAM>
  </Attribute>
</ModifyFeature>
```

```
<ModifyFeature>
<ID>49081000</ID>
<FeatureType>334</FeatureType>
<Dimensionality>3</Dimensionality>
<Attribute>
<GID>01552301</GID>
<NAM>New River</NAM>
</Attribute>
</ModifyFeature>
```