

Developing and Implementing an Exchange Network Node

Version 1.1

Environmental Information

exchange
Network

ACKNOWLEDGEMENTS

This document was developed with invaluable input and support from the following individuals:

Alesia Whitney-Knight	Iowa Department of Natural Resources
Cheryl Franklin	Indiana Department of Environmental Management
Carmel Rubin	Maine Department of Environmental Protection
David Blocher	Maine Department of Environmental Protection
David H. Ellis	Maine Department of Environmental Protection
Dennis Burling	Nebraska Department of Environmental Quality
Glen Carr	Oregon Department of Environmental Quality
Leslie Brennan	New York Department of Conservation
Mary Blakeslee	Environmental Council of States
Mitch West	Oregon Department of Environmental Quality
Robert E. Williams	Maine Department of Environmental Protection
Tom Aten	Wisconsin Department of Natural Resources

Prepared By



4000 Kruse Way Place
Building 2, Suite 160
Lake Oswego, OR 97035
(503) 675-7833

<http://www.windsorsolutions.com/>

Table of Contents

INTRODUCTION	1
HOW TO USE THIS DOCUMENT	3
GETTING STARTED.....	4
1. <i>Research the Network</i>	4
2. <i>Consult Mentoring States</i>	5
3. <i>Planning What Data to Exchange</i>	5
4. <i>Securing Agency Commitment</i>	8
DON'T REINVENT THE WHEEL.....	10
IMPLEMENTING YOUR NODE	12
1. <i>Determine the Technical Architecture</i>	12
2. <i>Determine Network Node Capabilities</i>	12
3. <i>Develop and Deploy Network Node for Testing</i>	13
4. <i>Test Network Node Components</i>	14
5. <i>Implement Network Node in Production Environment</i>	15
6. <i>Ongoing Maintenance and Support</i>	15
IMPLEMENTING A DATA EXCHANGE.....	16
1. <i>Design the Exchange</i>	16
2. <i>Develop the Exchange Services / Procedures</i>	20
3. <i>Test the Exchange</i>	21
4. <i>Implement the Exchange</i>	21
5. <i>Develop Trading Partner Agreement</i>	21
DECIDING ON YOUR TECHNICAL ARCHITECTURE	22
1. <i>Physical Components of Node Architecture</i>	22
2. <i>Logical Components of the Node Architecture</i>	25
3. <i>Management, Support and Extensibility Tools</i>	25
4. <i>Node Functional Capabilities</i>	26
REFERENCE SUMMARY	28

THIS PAGE INTENTIONALLY LEFT BLANK

Introduction

The National Environmental Information Exchange Network (Network) is an innovative approach for the exchange of environmental data between the US EPA, States, and Tribes. The Network promotes access to, and exchange of, quality environmental data while reducing burden and increasing efficiency of data exchanges. States, Tribes, and the US EPA expect the Network to become the preferred method for routine inter-governmental transfers of environmental data. Figure 1 illustrates the conceptual structure of the Network.

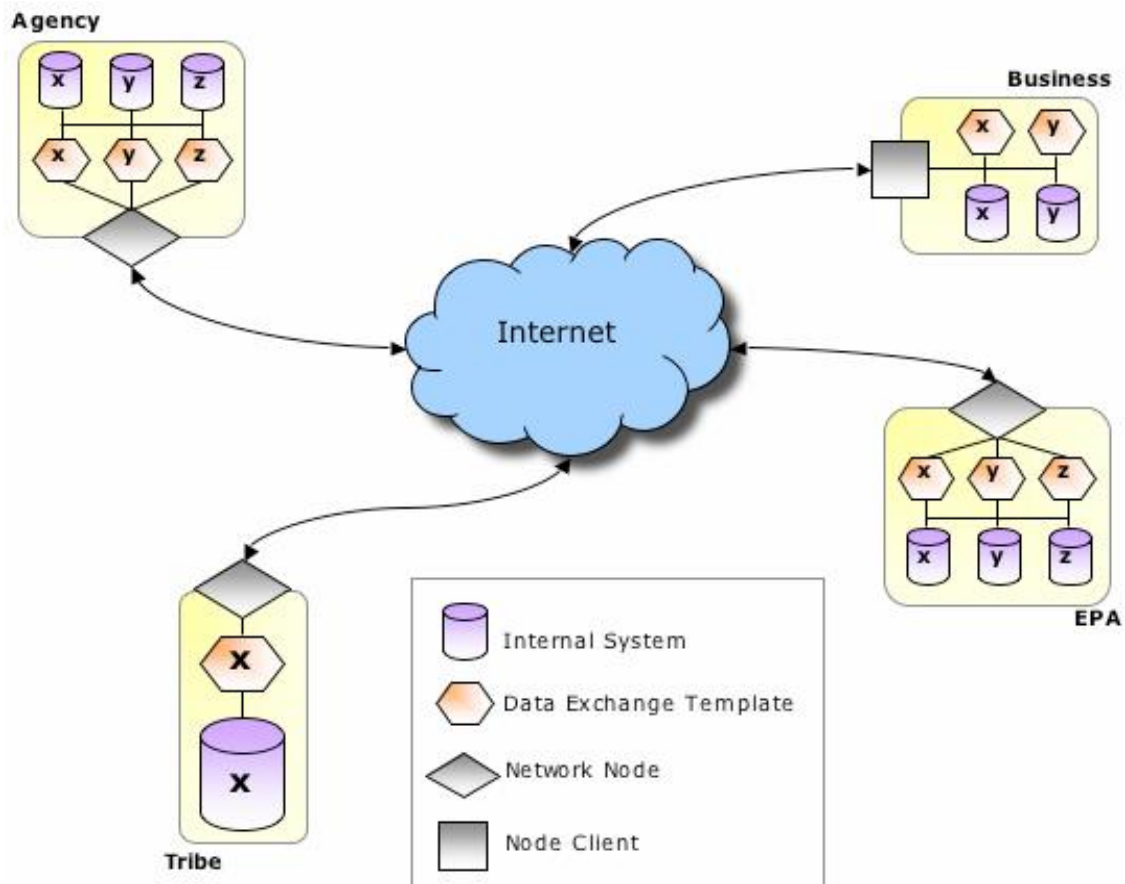


Figure 1: Exchange Network Overview

Since the initial Network design was articulated, significant progress has been made towards implementing the fundamental Network concepts. The US EPA has established their Central Data Exchange (CDX) Network Node, and many States and other Partners have also implemented Network Nodes of their own or Node "clients" which allow these Partners to initiate requests to other Network Partners but not to respond directly to external requests. States and the US EPA have already established a number of information exchanges over the

***'Node':** A Node is a common Web service made externally available over the Internet by the Partner.*

Network, using Node client or Node-to-Node communications.

The fundamental Network concepts are documented in two core documents¹:

- *Blueprint for a National Environmental Information Exchange Network (Blueprint)*, and
- *Implementation Plan for the National Environmental Information Exchange Network (Implementation Plan)*.

These documents introduce the objectives for the Network and describe the generalized architecture. This architecture is founded on the deployment of Nodes by Network Partners. Subsequent design and implementation activities have complemented these basic concepts with the following additional detailed specification documents:

- *Network Node Functional Specification version 1.1*,
- *Network Node Protocol Specification version 1.1*, and
- *Node Implementation Guide 1.0*.

The purpose of this document is to provide practical, step-by-step help for managing the deployment of a Network Node and the sharing of environmental data with one or more Partners. The intended audience for this document includes:

- Partners who have not yet implemented a Node;
- Partners who have implemented a Node, but are interested in alternative approaches;
- Partners looking to advance their Node implementation (for example expanding their Node administration functionality).

'Partner': a Federal, State, Tribal or Local agency that wishes to share data using the exchange Network.

¹ Many of these documents can be found at www.exchangenetwork.net/node

How to Use this Document

This guide was compiled based on input from a variety of States that have already implemented their Nodes and at least one exchange of data. These States represented a diversity of geography, technology and covered the spectrum of early implementers to more recent implementers. The experience they have gained (both good and bad) provides very valuable insight into the business and technical issues that a new Partner may experience during initiation into the Network.

Throughout the document the reader will notice italicized quotations inserted within the text. These quotes were provided by the representatives of the contributing States.

The document is organized into the following sections, which generally follow the process required to implement a Node and an exchange of data.

Getting Started

Explains the first steps to take, such as where to learn about the Network, how to determine what data the agency should share, and how to solicit agency buy-in.

Don't Reinvent The Wheel

By joining the Network at this stage in its rollout, your organization can benefit from the investments made by the early implementers. This section explains the potential for reuse of Node technologies from other Partners.

Implementing Your Node

This section describes the typical tasks and issues that normally occur during Node implementation based on the experience of others.

Implementing a Data Exchange

Similarly, for the development of the initial data exchange, this section provides a breakdown of the tasks that other Partners have typically experienced to develop an exchange, along with some tips that may simplify the process.

Deciding on a Technical Architecture

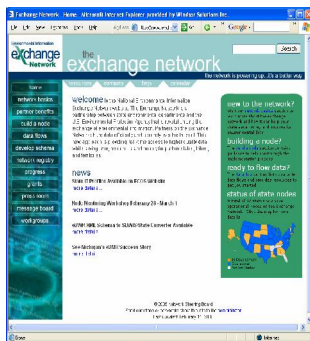
This section provides some insights into the types of technology decisions needed, which is one of the earliest tasks to perform. The section has been placed at the end of the document because the intended audience is technology specialists, which may represent only a subset of the readers.

Getting Started

There are a number of steps that a new Network Partner should undertake when considering the development of a Node. These steps will help the Partner to prepare for the required direct and indirect activities. This section describes these preliminary steps.

1. Research the Network

The first step for a Partner considering participation in the Network or the development of a Node should be to research the extensive information resources that are already available on the Exchange Network Web site and at other locations. While not an exhaustive list, the following key resources are currently available:

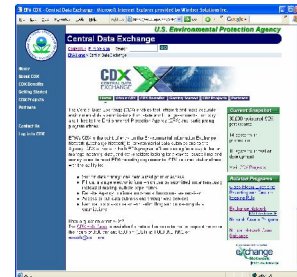


Exchange Network Web site

- The primary resource about the Exchange Network.
- Includes background information, FAQs, funding information, and much more.
- The entry point to the “Registry” of technical documentation.
- Includes a message board to help share knowledge.
- www.exchangenetwork.net

US EPA’s CDX Web site

- For information about exchanging data with the US EPA.
- Help Desk Phone: 888-890-1995
- Help Desk E-mail: nodehelpdesk@csc.com
- www.epa.gov/cdx
- Exchange Network grant links



Environmental Data Standards Council Web site

- For information about data standards that are incorporated into existing or future exchanges.
- Can be used to provide additional details (e.g., definitions) regarding specific elements within XML Schema.
- www.envdatastandards.net

ECOS Web site State Profiles

- To identify peer States and what they've achieved with the Network. Provides a profile of each State's exchange capabilities, and contact information.
- www.ecos.org/content/state/profile/?id=wi (e.g., for Wisconsin).



2. Consult Mentoring States

A group of States who led the way during the initial pilot implementation of the Network has offered to mentor other Partners as they join the Network. These States are: Maine, New Hampshire, Delaware, Nebraska, Mississippi, Oregon, Utah and New Mexico, and all are willing to offer assistance in a variety of ways to new Network Partners.

The Network is a rapidly maturing initiative, and as such the preparation and dissemination of important information tends to lag behind the innovation. It is highly recommended that you not rely solely on published information regarding, for example, any one exchange, availability of Nodes and other software components (referred to as Demonstrated Node Configurations or 'DNCs'), or technical guidance documentation. It is advisable to use the community of ECOS, the US EPA, mentor States and other Partners to get the most up-to-date, or as yet unpublished information. If in doubt, pick up the phone, send an e-mail or post a message on the Network message boards (on the Exchange Network Web site) to request additional information. The US EPA CDX team is another extremely valuable resource in this regard.

“Talk with the CDX Help Desk to find out what is available on their end. They are the ‘constant’ regarding working on the Node. Other states can only help so much, and in some respects, we had a different technical infrastructure. This is where the experience and expertise of the Help Desk can better assist you.”

3. Planning What Data to Exchange

Prior to building a business case for Network participation, the Partner should determine which Network exchange or ‘flow’ should be pursued with the initial implementation of their Node. This is important for Node validation purposes and business case preparation although not technically essential. An analogy might be to contrast implementing a Node without an exchange to purchasing a mobile phone without a service plan.

3.1. Factors that determine the first exchange

Deciding on the appropriate initial data exchange to be implemented will depend on a variety of factors, including:

- The Partner's priorities for data sharing with the US EPA or other Partners. For example, some Partners have found that pre-existing regional initiatives provide pressing drivers for exchange of data; other Partners instead focus on modernizing their legacy State/US EPA data management processes.

‘Exchange’: the sharing of a specific type of data between two or more Partners; for example, sharing of water quality data. This is often also referred to as a ‘flow’.

- The data management authority of the Partner. What data does the Partner ‘own’ that may be usefully shared with other Partners?
- The exchanges that are currently supported by the Network infrastructure. It is recommended to deploy an existing, proven type of exchange for the initial sharing of data. Ideally, this exchange should have a Flow Configuration Document (FCD) available to explain the operational requirements of the data exchange.
- The status of agency information systems. The Partner must ensure that the system that will provide a source of the data is stable and has adequate data quality (completeness, accuracy, QA/QC procedures). Furthermore, the agency will need to have a good institutional knowledge regarding the database structure and organization so that the extraction of that data is realistic without a significant learning curve, or the risk of too much trial and error. A detailed review of any candidate system is recommended to determine the suitability as a data exchange.
- The scope and complexity of the initial exchange should be limited. Some data exchanges are far more elaborate than others, so it is recommended that a new Partner consider all of the pertinent issues when making an exchange selection, to minimize the number of issues that may be encountered.

3.2. Some Suggested Exchanges

3.2.1. Facility Registry System (FRS)

As an example, many States have chosen the exchange of facility identification data to the US EPA’s FRS system as their initial data exchange for the following reasons:

- This is the most mature exchange currently implemented on the Network. A Flow Configuration Document (FCD) and template Trading Partner Agreement (TPA) is available for a Partner to use.
- The US EPA presents this data to the public via the Envirofacts Web site (www.epa.gov/enviro/index.html), including data from several US EPA system sources. The US EPA and States see the value of providing this integrated (i.e., multi-media) data directly from the State, because they typically have more accurate data, and local knowledge to ensure appropriate data integration.
- Integrated facility identification data is readily available in many States. Even if not all facility data has been reconciled by the state, FRS is still able to receive partial data as long as a complete data set is available for at least one ‘media’ or ‘program’ type.
- The US EPA’s CDX is a proven Node and has resources dedicated to helping Partners test their exchange (i.e., a separate test environment and the CDX Help Desk).
- The FRS exchange is not required by regulation and so the timing of the initial implementation is typically not ‘mission critical’.
- If a State does already have an integrated facility system, there is a good chance that the US EPA also manages some facility data that the State would like to ‘cross-reference’ with the State facility data (e.g., facilities managed within US EPA systems such as TRIS, RCRAInfo and PCS). By ensuring that FRS is up-to-date with State data, the State would also be able to make use of the US EPA-‘owned’ facility data that FRS also contains.

3.2.2. Safe Drinking Water Information System (SDWIS)

As another example, if an agency is already using SDWIS/State and FedRep, then this may also support a relatively straightforward initial exchange of drinking water quality data. The FedRep software outputs XML and performs all of the federal data edits necessary, and therefore reduces many of the risks associated with an agency's first exchange of data.

To provide some insight into the mechanics of a data exchange, Figure 2 illustrates a typical data exchange process between a Partner and the US EPA. It should be noted that this represents only one type of exchange process and that a variety of other exchange mechanisms are also possible.

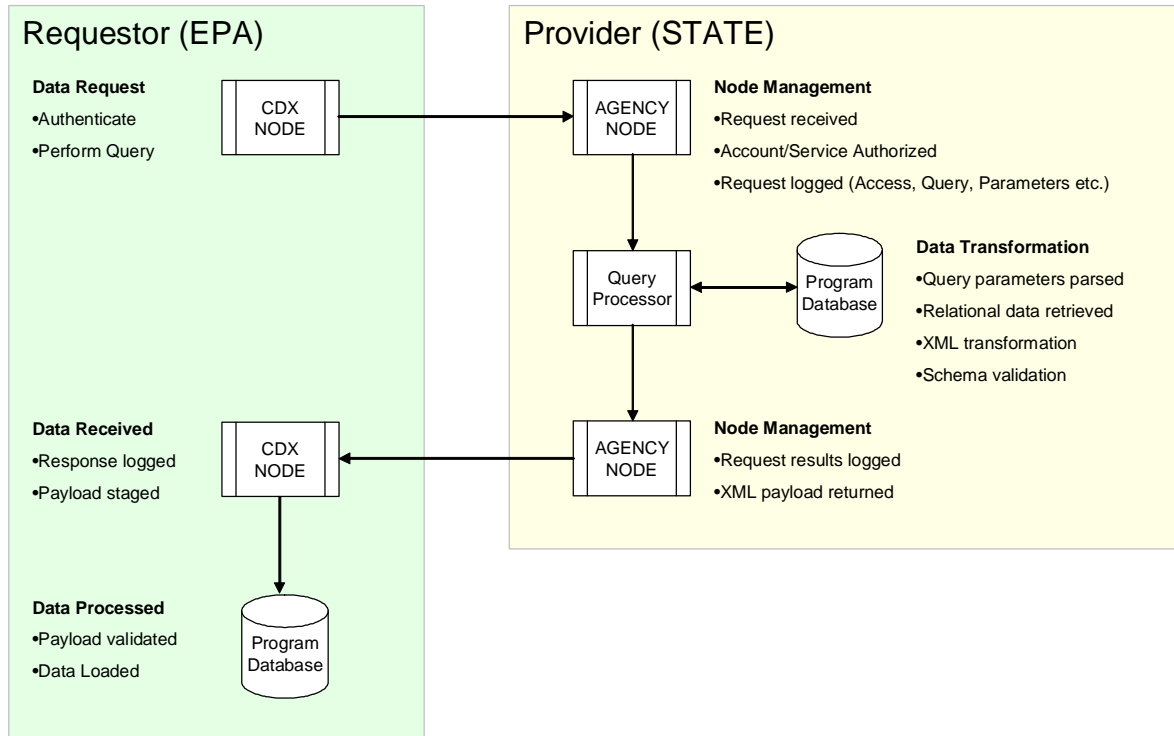


Figure 2: Example State-EPA data exchange process

3.3. Initial Exchanges to Avoid

Based on the experience of some of the States that have already developed a number of exchanges, these are some examples of exchanges that may be best to avoid for an initial implementation:

- An exchange to support a Partner's information system that is undergoing significant re-engineering.
- An exchange with a Partner that is only required very infrequently (e.g., annually), and thus, depending upon the timing, may be at a time when the Partner is not in a 'testing mode'.
- An exchange where there is no proven exchange already occurring. The added burden and risk of designing the necessary XML Schema and the exchange process could take away from the effort required to successfully implement and test a new Node.

4. Securing Agency Commitment

As the Network is extended and matures, the availability of shared data should increase dramatically, enabling Partners to better serve their stakeholders through enhanced access to data.

However, the short term benefits of participation are often not so clear. Participation in the Network will require resource investments both in the short and long term, and the agency must be prepared for the initial ‘start up’ investment of agency resources. Securing the commitment of agency management to participate in the Network is an important step.

An important characteristic of the ‘Network’: If the number of network Partners and the exchanges they support grow linearly, then the benefit will grow exponentially.

4.1. Presenting the case for the Network

Partners have identified a number of important values to Network implementation, including the following:

- Ongoing Exchange Network grant fund availability can help balance out the initial investment required. The provided funds will be used to initiate the exchange, but can also be used to improve data quality and associated system functionality, which adds value for the program.
- There are clear benefits to programs that replace existing arduous ‘batch data translation’ processes with automated processes. For example, the manual effort required to extract, validate and reformulate data into legacy file formats is both time consuming and unreliable, the Network infrastructure is intended to help mitigate many of these issues.
- An investment in the Network is also an investment in the broader electronic exchange of data, including ‘electronic reporting’. This is something that every agency sees as a growing demand from the regulated community. This initiative will help put in place a rigorous infrastructure that will help support the agencies’ future needs for Web-based reporting.

“Our statement to the agency has been: ‘the exchange process will be seamless to your daily activities and allow you to focus on your work and data entry in the office, not here and at EPA as well’.”

4.2. Securing Sponsorship

To participate in the Network, an agency will need involvement from technical and programmatic staff. Sharing data is not just a technical challenge, and programmatic staff has the knowledge needed to ensure that agency data is shared appropriately. Therefore, senior-level sponsorship is strongly encouraged to ensure the organization’s commitment to Network participation.

Partners should consider the appointment of a staff person with specific responsibility for Node implementation. This person would ideally have the following characteristics:

- A person who has the trust and respect of their peers and superiors, and has access to, and the trust of, technical staff.
- A person who controls funding resources and who is upstream in the reporting line of all staff that will be asked or told to assist with the implementation of the Node and exchange(s).

- A person with an organization-wide vision of how the Network can help both environmental exchanges as well as other agency data exchanges.
- A person who is able to articulate the vision of the Network in easily understood terms.
- A person who has an active interest about what benefits can be attained through technology.
- A person who is also involved with other cross-program initiatives.

Don't Reinvent the Wheel

The Network relies on Web services technology, a technology that is relatively new and still maturing. The basic capabilities of a Node, and the way data can be exchanged has been precisely dictated by the Network guidance documentation. The early implementers of the Network expended significant effort in creating and troubleshooting the technology required to implement a Node and exchange data.

“I think it would be foolish to start from scratch, since so many of the issues have been resolved.”

From the inception of the Network, it has been anticipated that much of the early research and development for building Nodes would be reused amongst many Partners.

States that participated in the initial Node development projects have reported costs/effort in excess of \$100,000 or 1 FTE year to develop their Nodes from scratch. Some experienced developing more than five versions of their Node to account for the steep learning curve required for implementing the new Web services technology and in

response the lessons gained through testing and enhancement.

To facilitate the reuse of Nodes, the Network promotes the development and sharing of source code (known as DNCs – Documented Node Configurations). Several DNCs are currently available on the Exchange Network Web site. During 2004, some States found that by reusing existing, proven Node software they could implement their Node at a fraction of the resources required by the early implementers (for example, the effort required was counted in days or weeks, rather than months or years).

Although reuse of an existing Node has in some cases been proven to be a viable and economic choice, there are some important points to note. Although some Partners have been able to implement a Node with only minor configuration changes, others have found that their agency's unique infrastructure has been at odds with that assumed by the Node they are deploying. For example, differing versions of database, operating system and/or development tools may require some challenging rework. Furthermore, if an agency assumes the design of a Node developed by others, there will be a learning curve associated with developing the in-house knowledge needed to maintain and enhance that Node in the future.

“Each Node development project must use the Node models and specifications in order to interact with the Network. As much as possible, Partners should use existing applications to streamline their development but must be prepared to adapt to their environment.”

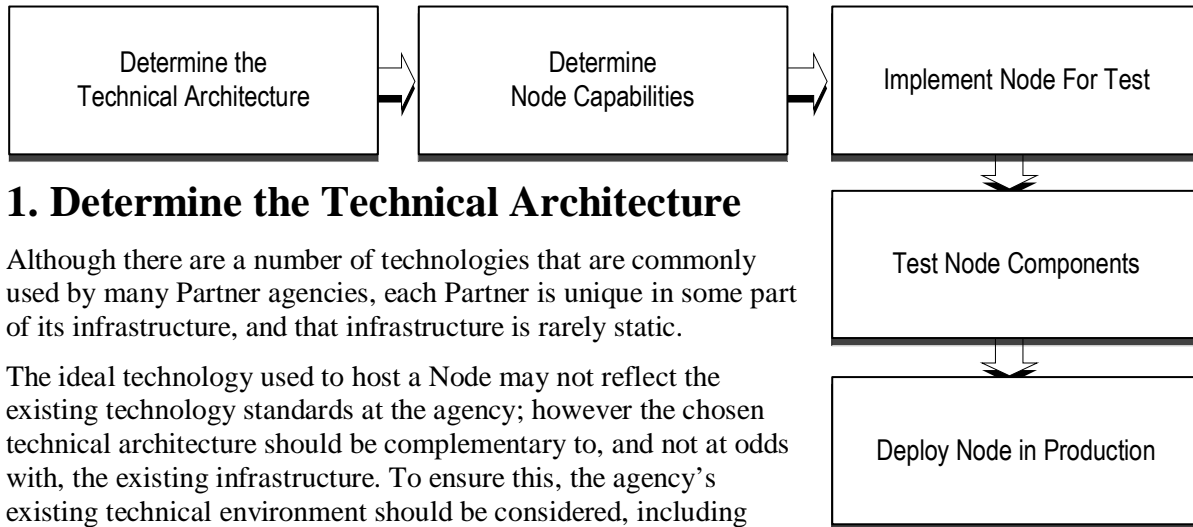
In addition to the reuse of basic Node technology, some Partners are also beginning to reuse certain data exchanges previously developed by other Partners. The format and process of an exchange is dictated by the agreed upon specification components such as the FCD TPA, and therefore lends itself to sharing. However, unlike a Node which must perform identically in all cases in order to be compliant with the Network specifications, an exchange must be specifically customized to the Partner's source system that provides the data for exchange. In some cases these systems may be identical across Partners (e.g., SDWIS State, widely used commercial systems, e.g. for managing monitoring data). Even if the source system is unique, however, the process of formulating XML and responding to a variety of defined exchange data services presents the opportunity for sharing some of the investment.

Exchange reuse is an area that is still evolving, but some States have already been able to reuse a Node as well significant components of another Partner's exchange infrastructure which has

accelerated their deployment and reduced the up front investment needed, allowing them to focus more on mapping their existing systems for the exchange, and resolving the data issues that they identify.

Implementing Your Node

Regardless of whether the Partner chooses to implement a solution previously developed by another Partner, or to develop a custom solution, the basic activities that must be conducted are essentially the same. This section describes the typical and generalized activities required to implement a Node, based on the lessons learned by both early and recent participants in the Network.



1. Determine the Technical Architecture

Although there are a number of technologies that are commonly used by many Partner agencies, each Partner is unique in some part of its infrastructure, and that infrastructure is rarely static.

The ideal technology used to host a Node may not reflect the existing technology standards at the agency; however the chosen technical architecture should be complementary to, and not at odds with, the existing infrastructure. To ensure this, the agency's existing technical environment should be considered, including network topology and application architecture.

A Node must be accessible from the outside world via the Internet, and yet it must serve up data that originates from secure, production servers. Therefore, agreement should be reached at an early stage with State-wide and/or agency network support, application architecture, and security staff as to the technical architecture that will be used for the Node.

There are a variety of technical architecture considerations and decisions that will need to be made before you are able to implement a Node, and these are detailed in the section of this document titled "Deciding on Your Technical Architecture."

Figure 3:
Implementing Your Node

2. Determine Network Node Capabilities

The minimum functionality of the Node Web service is specified in the *Network Node Functional Specification v.1.1* and *Network Node Protocol v.1.1* documents available at the Exchange Network Web site. However, beyond the basic Web service interface, there are other design decisions that should be made related to how the Node operations will be managed, how service requests will be satisfied, and how the Node will be administered. For example, these are some of the capabilities that Partners have supported within the design of their Nodes:

- The ability to troubleshoot an exchange, for example:
 - Reviewing details of a recently received or submitted XML document. This can be achieved by configuring the Node to store a copy of each payload which passes in or out of the Node.
 - Viewing a log to determine if a request had been correctly submitted by another Partner.

- Viewing a log to determine if a data services is being used with inappropriate parameters (e.g., use of wildcards not supported).
- The ability to monitor the overall use of the Node by other Partners, including frequency of access by each Partner, which services are most / least used.
- The ability to administer security that determines which Partners have access to which services (if used as an alternative to the US EPA's central NAAS -Network Authentication and Authorization Services).
- The ability to set up schedules for certain data exchanges (e.g., some exchanges are so large that they should only be supported outside of office hours).
- The ability to establish email notifications so that agency staff are notified when different Node functions are invoked.
- The ability to perform costing estimates which can determine an appropriate response to overly-intensive data requests.

Many of the capabilities that early Node implementers have added to their Nodes are not critical for an initial deployment, but as the use of the Node expands are likely to be a growing need, and so should be considered early on to ensure that the Node is designed in such a way to support its future growth.

A more comprehensive list of capabilities that other Partners have implemented is described in the section of this document titled “*Deciding on Your Technical Architecture.*”

3. Develop and Deploy Network Node for Testing

If the agency has chosen to reuse an existing Node, then little or no customization may be required. If the agency has decided to develop a Node from scratch, then:

1. The basic Node Web service interface will be developed to process and manage the incoming Network requests. It must be made compatible with the Web Service Definition Language (WDSL) description for the services defined in the Exchange Network Functional Specification.
2. Depending on the design of the Node, the necessary administration functions must be developed, including functionality to manage user accounts, transactions, documents, logs, and so on.

The required security identification certificates should be purchased and installed on the application server that will host the Node Web services. This will allow the Node to support the Secure Socket Layer (SSL) authentication mechanism that is required by the Network. The US EPA is also able to provide a certificate for free, however some States have decided to procure their own and use it for other purposes beyond just the Node.

For any exchange of data with the US EPA, or if the Network Authentication and Authorization Service (NAAS) is intended to be the sole authentication service used, the Partner must establish one or more accounts with NAAS. Once a “master” account has been established, this account may be

“If a Documented Node Configuration (DNC) has been selected, it is very important to read all the DNC instructions before starting to implement the Node. Read all the documentation relevant to installing and configuring the environment for the Node, i.e., the database and the application server. Careful and complete reading of this information will answer most of your questions and allow you to avoid unnecessarily boxing yourself into a corner and needing others’ help to get you out.”

used to create additional “operator” accounts and also to allow access to the Partner’s Node by other registered NAAS users. Please refer to the *Node Administrator’s Guide to Network Security* and the *Exchange Network Security Policy* documents which are provided by the US EPA CDX by request only.

4. Test Network Node Components

The Node components should be installed on a dedicated test environment and tested using the US EPA CDX test client² and/or other testing application. These can be located via: www.exchangenetwork.net/node/dev_toolbox. These tools will test basic Node operation and functions. One can also use some of these testing tools to access another Partner’s Node, if duly authorized, to see what types of response are received from that Node, and may provide a useful comparison.

The CDX Help Desk has assisted in testing many of the implemented Nodes, and it is recommended that you contact them to also test connectivity and functionality of your Node. Depending on the exchange deployed in your environment the CDX Help Desk may also be able to test the US EPA’s post-submission processing.

If possible, it is also suggested that one should try to test a Node with Partners other than the US EPA, as an added validation of the Node’s interoperability. This step is especially important if you are planning on inter-agency data exchanges.

4.1. Test compatibility with the specifications defined by the WSDL file

This test is best accomplished using the CDX Node Test utility version 1.1³.

The most common issues experienced during this stage of testing are:

- Lack of physical connectivity to the Internet due to either firewall configuration and/or security constraints;
- Incorrect implementation of the WSDL resulting, for example, from manual Web services interface generation;
- Incorrect configuration or lack of support for NAAS security.

4.2. Test capability to respond to and process predefined data queries

Once the basic specification compatibility has been verified, the Node should be tested for appropriate responses to incoming requests. Depending on the exchange supported by a Node this test may include either a multi-step processing scenario like Solicit, GetStatus, Download, or simple Query. Note that while the initial exchange may not require the Query primitive method, it is important to assure the Node is capable of processing such requests.

The most common issues experienced during this stage of testing are:

- Inappropriate configuration of the Node to support a particular query. Sometime this may be a result of lack of connectivity to the data source required to fulfill this request;

² The EPA test tool requires the support of the Facility Identification exchange. The Submit, Solicit and Query tests are performed using that exchange.

³ The EPA Test utility can be access at <https://test.epacdxnode.net/test/>. Earlier implementations of the Node Test Utility are no longer recommended for use.

- The server fails to respond when processing a data request with a specific combination of parameters due to either inappropriately sized hosting environment or data queries in need of optimization;
- Incompatibilities between heterogeneous Nodes may cause errors. While Web services technology is an open standard, it is possible that Nodes utilizing different platforms and technologies may have trouble communicating. An example of this would be that Nodes built using Microsoft .NET technology may require the use of a special output filter to communicate with the US EPA CDX Java Node.

5. Implement Network Node in Production Environment

Finally, the Node components will be installed in the Partner's production environment (assuming this is separate from the initial testing environment).

If possible, it is recommended that the agency maintain both production and test Node implementation. This will ensure a more stable production environment as the Partner develops and tests enhancements to the Node, and adds new exchanges to the Node (both of which can cause Node interruption until they have been fully proven during testing). If dual environments are to be employed, it is important to make sure they are configured the same wherever possible; as other Partner's have found that minor differences can cause unexpected issues.

6. Ongoing Maintenance and Support

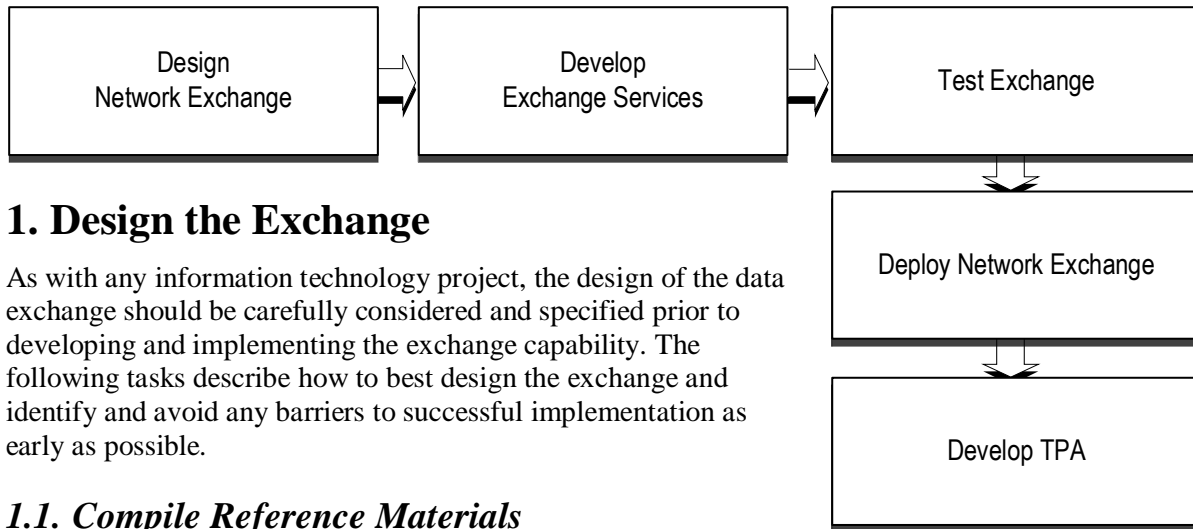
Although a Node is intended to be self-sufficient by automating the process of data exchange, it is still necessary to dedicate some support staff time to monitor its operation. This will require training to ensure that the assigned person knows how to evaluate the status of the Node and its operations.

As a basis for dedicating resources for this role, a sample of Partners has reported between ¼ and 1½ of an FTE being required to support the ongoing operation of their Node⁴. This is partly dependent upon whether the Partner uses internal staff or contractors to perform upgrades to the Node.

⁴ This estimated effort is solely for the ongoing support of the Node, and excludes any effort related to setting up and supporting new exchanges of data through the Node.

Implementing a Data Exchange

In an ideal situation, a Partner will pursue the development of an initial data exchange on the Network in parallel with the implementation of the Node. This will help with Node validation as well as in business case preparation. This section discusses the high-level steps involved in designing and implementing a Network data exchange. The following tasks outline a generic, but proven, approach to achieve this. Some of these tasks may overlap, for example the development of a Trading Partner Agreement (TPA) is a task that should ideally be performed in parallel to each of the other tasks to implement an exchange, even though it will probably not be completed until the exchange has been finally implemented. Figure 4 illustrates the general steps in the process.



1. Design the Exchange

As with any information technology project, the design of the data exchange should be carefully considered and specified prior to developing and implementing the exchange capability. The following tasks describe how to best design the exchange and identify and avoid any barriers to successful implementation as early as possible.

1.1. Compile Reference Materials

The Exchange Network Web site provides important reference materials for each existing exchange (in the ‘Network Registry’ section of the site). The most recent versions of the materials summarized in the following table (if available) should be compiled and reviewed.

Once again, it also recommended that the Partner considering the exchange also contact other Partners who have previously implemented the same exchange. This may provide some additional insights, and may also provide updates on the status of the exchange, for example whether there are updates to the FCD or XML Schema being considered.

Figure 4:
Implementing a Data Exchange

	Document Type	Acronym	Purpose
Documents	Data Exchange Template	DET	A spreadsheet outlining each data element within the Schema along with definitions, validation rules, and example content. This is a more human readable version of the XML Schema.
	Example XML Instance document		An example XML document that can greatly help understand what the intended 'output' of the Node should be when servicing a request. For example, for a Facility Identification exchange, one could get an XML file from an existing partner that includes data for a few of their Facilities.
	Flow Configuration Document	FCD	Describes HOW the exchange of data should occur. This should be the first document to consider.

	Document Type	Acronym	Purpose
	Step by Step Guide		A document that summarizes the steps required to initiate the exchange. These exchange-specific documents are being developed to help explain any requirements that are unique to each type of exchange.
	Template Trading Partner Agreement	Template TPA	A template document that specifically developed for this type of Network exchange. If unavailable, an example TPA should be attained as an alternative.
Technical Files	XML Schema		The formal definition of the structure and format of the data.
	Data Services XML File		This file formally specifies the data services described in the FCD. This may be used during testing of the exchange using a Node Client so that the Client can provide a more intuitive user interface to your Node's data services.

1.2. Determine Exchange Approach

The exchange developer(s) will work with program system representatives for the source data system(s) that will support the data exchange and determine how the exchange will be achieved.

1.2.1. Determine Data Services to be supported

If a Flow Configuration Document (FCD) has been developed for the selected exchange, this will provide much of the design information needed for the data exchange. This document describes the types of data services that the Partner must provide to support the exchange of data. In some cases the FCD may include alternative or optional data services. Exchange developer(s) will review this document and in conjunction with the program representatives, identify the appropriate data services to be implemented.

Many exchanges are based on the concept of a Partner making their data available to any other Partner to “come and get it,” based on a few commonly used criteria (e.g., facility name, analyte name). Some exchanges provide the option to exchange only incremental changes to data (i.e., transactional data sets, which only include new, modified or deleted data items). This approach allows for much smaller data sets to be exchanged, but also requires that the source information system keeps a detailed audit trail of such changes to the data. Many Partners have found that their systems do not support this reliably, so it is important to ensure that the implications are clearly understood before this exchange is attempted.

“Deletions were not handled until after the flow had been in ‘production’ for over a year.”

As a general rule, it is important to consider how subsequent exchanges of data will be supported, not just the initial exchange of data.

1.2.2. Map agency databases and identify potential issues

The exchange developer(s) should investigate the structure and content of the existing system database(s) and map them to the XML Schema for each data exchange. If a DET is available (in the Registry) this will provide the ideal format for mapping the two systems. If you can obtain an example XML document from another Partner, this is highly recommended. It will give you a real-life example of what the end result of the exchange should look like.

This mapping exercise is intended to provide a specification of the data extraction process, and also to help identify any issues (e.g., data incompatibility, ambiguity of intent within the schema). Any

mapping issues that are identified should be discussed and resolved with the Partner program system staff and the relevant US EPA or Partner staff. Some issues may be easily resolved, while others could require that the program system be modified to include additional data points or controls.

“Our biggest problem was not knowing what the (XML) package should look like.”

The following table provides an excerpt of a possible mapping document. The left side includes information about each of the data elements within the XML Schema (this can be extracted from the DET if one is available), and the right side presents the database fields in the source system that supplies this data, along with comments or the logic required to translate the data. If a DET is not available, one should review any available XML Schema

documentation, and refer to any applicable data standards at the EDSC Web site to provide a better insight into the intent of the schema.

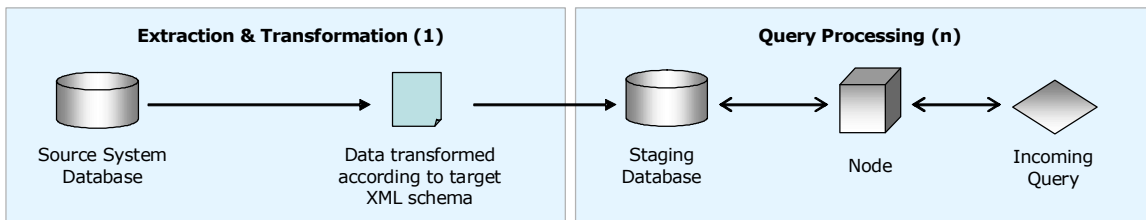
XML Schema					Source System		
Data Element	Description	XML Tag	Required (Y/N)	Data Type	Table	Column Name	Comments or Join Information
Environmental Information System Abbreviate Name	The abbreviated name that represents the name of an information management system for an environmental program. <i>Example Values :</i> TRIS RCRAInfo PCS	InformationSystemAcronymName	N	A(20)			Always 'FIS'
Environmental Information System Identification Number	The identification number, such as the permit number, assigned by an information management system that represents a facility site, waste site, operable unit, or other feature tracked by that Environmental Information System.	InformationSystemIdentifier	N	A(30)	authorizations / program district identifier	prog id / prog district identifier	max(prog id) from authorizations where auth status code in ('2', '3', '4', '5', '7', 'd')
Environmental Interest Type	The environmental permit or regulatory program that applies to the facility site. <i>Example Values :</i> TRI Reporter NPDES Major Air Synthetic Minor Air Minor TSD LQG	EnvironmentallInterestTypeText	N	A(60)	authorization type / program district type	prog district type desc	for authorization: if article > " then auth type desc + ' under ' + article + ' ' + title else auth type desc

1.2.3. Define the data extraction procedures from the source systems

There are a few alternative ways that data can be extracted from source systems to respond to exchange requests. The choice of which to use is dependent upon the architecture chosen, but is also dependent upon the nature of the exchange being implemented. The following three options outline some typical scenarios that have been used by some Partners.

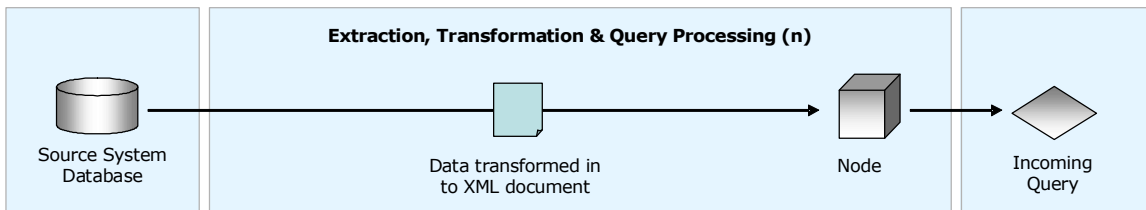
Fully Disconnected Model

The fully disconnected model is recommended for more complex or high-volume data exchanges where the isolation of demanding or frequent Node queries from the internal production system is most important. This model allows for data to be pre-filtered and de-normalized allowing maximum performance during repetitive or similar queries. The bulk of the transformation work is performed one time (e.g., nightly) whereas the queries against the ‘pre-shaped’ data is performed multiple (n) times. Two additional advantages are that this model eliminates the load on the production database and increases security, since the node does not connect directly to the source system.



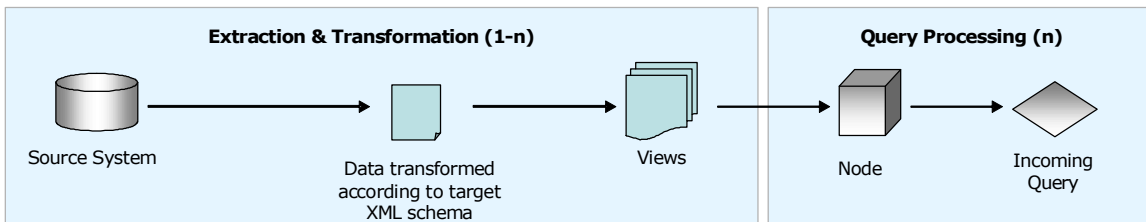
Real-time Data Model

A real-time transformation process directly accesses the production source system and therefore guarantees the most recent data for all queries. This model is best suited for undemanding queries against data sets where even multiple (n) concurrent queries would not compromise the performance of the source systems. In this model the data is transformed during each query.



Snapshot-based Model

The snapshot approach to data extraction is a hybrid between the fully disconnected and real-time extraction models. It uses a concept of a snapshot of data (currently supported in Oracle but the next release of SQL Server will support this feature too). The source production data is transformed into a schema-like model via frequent refreshes of ‘materialized views’ of the source data. This model provides the performance of a fully disconnected model along with the timeliness that is closer to that a real time model.



1.2.4. Resolve Issues with Trading Partner(s)

Some exchanges are fairly new, and in these cases there will quite likely be some items that are not clearly documented or are ambiguous within the XML Schema, FCD, etc. Any issues identified should be discussed with the trading Partner to ensure that the data is being exchanged in a mutually

acceptable way. Where the resolution is specific to the agency, these should be documented so that they can be incorporated into the Trading Partner Agreement (TPA). Example issues include:

- Mapping of look up code values.
- Ambiguous data elements that need clarification of their definition and ultimate use.
- How to handle data quality issues (e.g., bad historical data – should it be excluded or included?).

If the resolution to any issue results in a necessary compromise, then that decision should be included in the TPA.

1.3. Prepare for Implementation

Once the approach for the exchange has been determined, the project manager should develop or refine the plan for implementation and coordinate with each of the stakeholders to ensure their involvement is timed appropriately. Example stakeholders and involvement are:

- Source system representatives:
 - Support internal testing of the data exchange;
 - Resolve data issues (e.g., data elements that do not directly map to permitted values as specified in the XML schema).
- The US EPA / CDX:
 - Provide NAAS security.
- Trading Partner(s)
 - Support testing of the exchange (e.g., providing a sample XML document, invoking a test request from the Node);
 - Co-refining the TPA document.

2. Develop the Exchange Services / Procedures

With the approach for the exchange defined and the other stakeholders prepared, the automated procedure to exchange the data can be developed, tested and implemented.

2.1. Develop Exchange Services

The necessary data exchange service request components must be developed in the selected target technical environment. This functionality will reside behind the basic Node Web service interface and will be developed to process and manage the incoming requests to the Node.

Components will be developed to manage the extraction of information from the source systems. Depending on the exchange design agreed on during the previous task, these components may be database level or middle-tier components.

2.2. Modify source system(s)

If the design of the exchange calls for modifications to be made to the source system (e.g., adding additional data validation rules, tracking additional data items) then these changes must be applied. Depending upon the situation, this task may be unnecessary, or could require extensive effort.

3. Test the Exchange

The selected data exchange should be tested to evaluate responses to incoming service requests, the exchange operations and the correctness of the returned XML documents. Test extracts should be submitted to the US EPA or the appropriate exchange Partner for testing and evaluation for completeness and accuracy. It is often useful to initially prepare an example XML document and e-mail it to the recipient to review and test, prior to using the Node for the exchange. This way any issues associated with the data can be resolved without having to wait until the Node has been fully configured to share the data automatically.

Some common issues that may be encountered during the testing of the exchange are:

- Some data records are duplicated or missing due to incorrect extraction of the data. The XML Schema may not identify this issue.
- Look up codes may be incorrectly being assigned due to a failing in the cross reference to existing system codes.
- Formulation of some larger XML documents can consume significant server resources. Testing should include the worst case scenario to ensure the software and hardware is capable of supporting such heavy payloads.

4. Implement the Exchange

Following testing of the Node and data exchange components, corrections should be made to the developed components as necessary.

The new exchange components will then be installed in the Partner's production environment, according to the defined technical architecture.

5. Develop Trading Partner Agreement

A Partner may be required, or feel it necessary to develop a TPA. This is typically done jointly with the Trading Partner as a way to manage each party's commitments to the data exchange.

In some cases this may be a more unilateral document that provides a context for any Partner that wishes to access and use the data that is being made available. This work should take account of the published TPA development guidelines (available on the Exchange Network Web site), and may be further informed by any template TPA that may have been developed for the data exchange.

“We clearly specified what we intended to do and have subsequently used the TPA to back up our work.”

Deciding on Your Technical Architecture

This section describes a number of potentially useful considerations when a Partner begins the process of determining a suitable technical architecture. The technical architecture defines the technology considerations and decisions that will guide many aspects of design, development and deployment of a Node.

The capabilities of a Node itself are based on a series of well-defined functional specifications, and so the architecture chosen should be compatible with these specifications as well as the standards and existing infrastructure at the agency. It is appropriate to define the desired technical architecture prior to either identifying a Node implementation to reuse or beginning development of a custom solution.

The aspects of the Node Technical Architecture that are particularly important are:

- Definition of the physical environment including the identification of physical hardware, hosting platform and network components.
- Identification of logical components including middleware, applications and frameworks necessary to support the Node specifications while adhering to any agency-wide standards.
- Selection of the necessary development, management and support tools, which may be used to enhance and extend Node functionality.
- The types of more advanced Node capabilities that are not as critical for an initial Node implementation, but have been found to be important for longer term use, and would impact the architecture chosen for the Node.

1. Physical Components of Node Architecture

1.1. Network Topology

One of the most critical decisions to consider prior to deployment of the Node is the physical location of the Node's external interfaces (services) and the supporting components (RDBMS, Content Management, LDAP, etc.). This decision is typically dictated by the existing agency-wide methodology, which is likely to fall within one of three scenarios: DMZ-centric, DMZ-light and Proxy-centric.

1.1.1. DMZ-Centric Network Topology

The DMZ-centric approach tries to isolate the Node and its dependant components into the agency network's "Demilitarized Zone" (DMZ). Partners that employ this approach either host the Node on existing hardware already deployed in the DMZ or by deploying the Node to its own independent hosting environment.

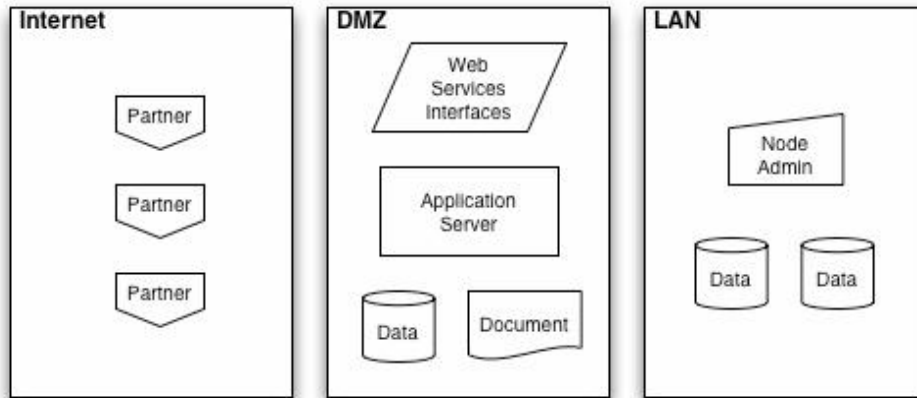


Figure 5: DMZ-Centric Topology

1.1.2. DMZ-Light Network Topology

The DMZ-Light approach also uses the agency DMZ, but only to host the actual Web services interfaces. The application server and its supporting components are hosted from the internal network.

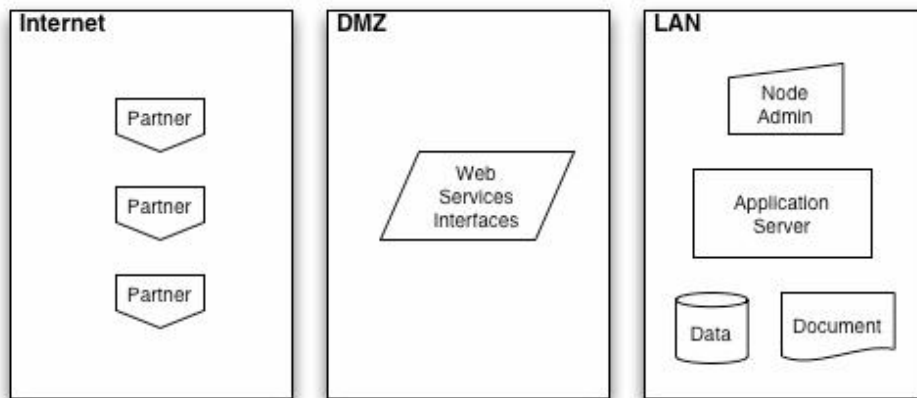


Figure 6: DMZ-Light Topology

1.1.3. Proxy-Centric Network Topology

A proxy-centric solution utilizes a proxy server as a virtual host for the Node while relaying all requests to an internal server where such requests are processed. This approach is likely to be the preferred solution in environments where reversed-proxy solutions represent standard operating procedures for hosting Web applications.

In all scenarios, the supporting components such as the Node hosting database and/or exchange data providers are almost always hosted from a trusted network protected by firewalls where no direct access from the outside world is allowed.

“Originally we were just out there on our own, exposed to the world. We are now behind the state firewall for our node and our data.”

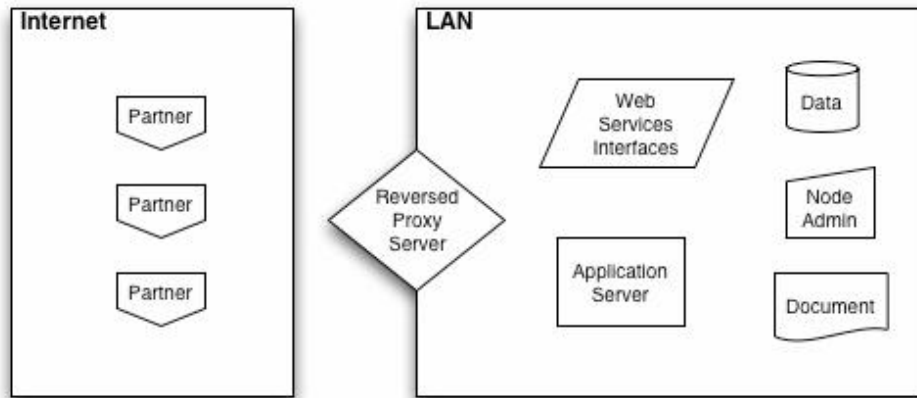


Figure 7: Proxy-Centric Topology

1.2. Dedicated vs. Shared Server

Based on the States that have already implemented Nodes, there is no definitive answer to whether the Node should reside on a shared or dedicated server. The decision to either co-host the Node with existing applications or isolate it to its own environment should be based on factors such as cost, selected architecture (i.e., will this be a thin-Node, or will it also perform XML formulation), and the existing load on the server.

Microsoft Server-based Nodes seem to have generally been hosted on a dedicated server (possible due to Microsoft's recommendation that Web services and Web sites should be kept apart), whereas UNIX based Nodes seem to have generally been co-hosted on servers along with other applications.

1.3. Separate Testing Environment

To assure maximum stability in their production environment as well as to assure maximum testing it is recommended that Partners establish a production-mirrored test environment where the new exchanges and/or Node enhancements can be tested prior to deployment in production.

This may not be of great importance during the initial deployment, and if the Node server is dedicated only to the Node, but once the Node is up and running, it is intended to be accessible to other Partners on a 24x7 basis, and the risk to stability caused by using the production Node for testing of enhancement, fixes or new exchanges should be avoided if possible.

1.4. Staged RDBMS

Due to the demands of some exchange database queries it is recommended that a Partner employ a staged database environment which will allow the exchange data to be pre-filtered and de-normalized. This will ensure the maximum performance in fulfilling incoming data requests without impacting the internal production systems that provide the data. This approach may be more challenging for any real-time systems (e.g., Air Quality Monitoring data), as a real-time replication process would be needed to stage the data.

"We are still working on "query costing" so we can reject those that might impact systems."

2. Logical Components of the Node Architecture

With the increasing use of Web services there are a number of platforms and software packages capable of supporting the Node specifications.

While it is possible to develop a Node using many of these solutions, in general there appear to be two primary middleware platforms used in for implementations of a Node; Microsoft's .NET and Sun Microsystems's Java. A Partner's choice between these two or other alternatives will be influenced by compatibility with existing architecture and the experience of support staff.

"We used the agency's preferred technology mostly for simplicity and cost reasons."

The choice of middleware and hosting platform are closely related and often dictate a particular solution for one or the other; for example, the Microsoft architecture virtually requires the use of IIS when developing Web services using .NET technology. In contrast, the choice of Java as the middleware allows greater flexibility in choosing both the application server as well as the hosting platform. For example, it is perfectly feasible that a Java-based Node be hosted

from Oracle Application Server, Tomcat/Axis or IBM WebSphere on the Windows, Linux or UNIX platforms.

For a database platform, there are even more choices. Though the most commonly used solutions are Oracle and SQL Server, other Partners have utilized DB2, MySQL, PostgreSQL or even MS Access. It is important to realize that middleware and database vendors are constantly enhancing their products and with the popularity of XML and Web services, their related capabilities are being frequently upgraded.

Application and database server choices are often driven by cost. When making these choices, Partners should consider not only the purchasing and deployment costs, but also the ongoing support and Node maintenance costs.

3. Management, Support and Extensibility Tools

3.1. Node Administration User Interface

While some Node deployments support a fully integrated GUI-based management and testing environment, many installations require the management to be performed through the editing of configuration files. In the latter case, if the staff that are tasked with the oversight and management of the Node do not have direct access to the hosting environment, then they must rely on the help of network administrators.

In contrast, GUI based tools that manage the configuration in a central location (database or LDAP) allow the Node administrator to manage the Node configuration and monitor its activity without having a physical access to the hosting environment. This approach also allows for more customizable Node monitoring and configuration; depending on the role of an individual user within the agency, he or she may have rights for monitoring or management of a particular exchange.

3.2. Other Tools

The developer tool of choice for XML manipulation and validation is Altova XMLSpy®.⁵ XMLSpy® provides a number of useful features, including XML validation, design of XML schemas and transformation style sheets.

⁵ XMLSpy® is a product of Altova. For more information about XMLSpy® see <http://www.altova.com/>

Alternatively, some vendor-specific solutions may provide a somewhat integrated environment or utilities to support the XML schema mapping and validation.

It is also worth mentioning that some preliminary studies are being conducted to evaluate other products like the US EPA CDX Schematron tool to perform more contextual validation of XML files, for example to validate lookup values that reside outside of the XML schema as well as more complex business rules.

4. Node Functional Capabilities

While there are a variety of implemented Node architectures with unique capabilities, there are some universal aspects of functionality that are important to the long-term scalability and flexibility of the Node. This section outlines these capabilities.

Separation of the Node and the individual exchange implementations.

One consideration for the Node architecture is the independence of the Node implementation from that of the individual exchanges. As new exchanges are added the already deployed Node infrastructure should use the new exchange extensions. This loosely coupled approach to the Node architecture allows for additions and modification to the data exchanges without the need for disruption and risk of alteration (e.g. code change and recompilation) to the Node itself.

This approach minimizes the necessary testing and allows for division of labor when developing new exchanges as the developers need to know only the interfaces required by the Node and not the Node architecture itself.

Support for an XML Document Header File on an exchange-by-exchange basis.

The Header File was developed to provide additional meta-data to the specific exchange and its payloads. The use of the Header File allows a sender and recipient to identify the particular payload during transport as well as at its processing destination. The introduction of the Header File after the Node functional specification documents were developed has created some confusion although its use is now required by several data exchanges.

While the use of Header File is nearly standard, its contents and usage vary by exchange. Ideally, a Node should be capable of designating the use and the content of the Header parameters for each exchange. This capability will support a wider spectrum of exchanges without source code modifications.

Support of authentication and authorization through both NAAS and Local Security.

Any Node exchanging data with the US EPA is required to use NAAS for its authentication. While the use of NAAS for authentication purposes is common, few Nodes rely on NAAS as their means of authorizing incoming requests. While the use of NAAS solely for authentication may be adequate if an agency only exchanges data with the US EPA, as the list of participating Partners becomes larger the need for centralized NAAS authorization will be more important.

Furthermore, the Node architecture should consider support for a local security model. Many Partners already have a standard means of authentication across multiple applications to facilitate a single-sign-on and would like to be able to leverage that same metaphor for data exchanges that are more on a local level. That approach allows for a closer integration into the existing security model without the need for replication of existent accounts on the national level.

Persistent attachment management.

The way that attachment management is implemented is not defined by the Exchange Network Specifications; however, a Node must support persistent attachments (resulting from the payload on

the Submit or content generated by the internal processes). While it is feasible to save attachments on the server that hosts the Node, this approach may not scale well and may compromise the security of the server.

A more scalable and secure alternative to local attachment storage is file management based on either binary storage in a database or a dedicated file management solution. A distributed model of attachment management allows for linear scalability. Should the Node's external interfaces need to be distributed across multiple servers (clustering) each one of these Nodes could have stateless access to the internally stored attachments.

Support for both incoming and outgoing data exchange.

While the Node architecture in its original deployment was somewhat US EPA-centric with the unidirectional data exchanges (State to US EPA), recently it has become more important for Partner Nodes to also be able to process incoming data payloads and integrate them in to their internal data stores.

Depending on the physical architecture of an individual Node, the process of integrating incoming data payloads into the internal data stores may be handled in a variety of ways. However, differentiation of incoming versus outgoing data is becoming more important, especially if the Node architecture is to be used to support electronic reporting of the reporting community.

Secured Sockets Layer (SSL) support through the use of certificates.

A node must utilize SSL technology in order to be Exchange Network compliant. The certificates issued by the US EPA that are made available to individual States are sufficient for the current data exchanges with the US EPA. However, as these certificates are self-signed (the US EPA at that point is acting as a Certificate Authority) they may not be recognized by all Partners, for example if a Node is accessed via the regulated community or external commercial applications.

While the issue can be easily resolved if encountered using a Web browser (the browser would prompt the user to accept the particular certificate), when dealing with machine-to-machine communication there is just no option for manual intervention. A variety of workarounds have been developed to deal with this issue however, when possible, the Node should be hosted from an environment where the certificate is issued by a well-known party and its full path is recognized by all common browsers.

Reference Summary

Please note that all URLs are subject to change. In the event that a resource can not be located at the address listed, please check the exchange network website located at www.exchangenetwork.net.

Exchange Network Fundamentals

Blueprint for a National Environmental Information Exchange Network

http://www.exchangenetwork.net/basics/blueprint_report.pdf

Implementation Plan

http://www.exchangenetwork.net/basics/imp_plan_feb2002.pdf

Exchange Network Grant Guidance

<http://www.epa.gov/neeengprg/index.html>

Exchange Network Frequently Asked Questions (FAQ)

<http://test.epacdxnode.net/faq/>

Node Building Resources

Exchange Network Node Building Home Page

<http://www.exchangenetwork.net/node/index.htm>

Node Functional Specification 1.1

http://www.exchangenetwork.net/node/dev_toolbox/node_functional_spec_v1.1.pdf

Network Exchange Protocol 1.1

http://www.exchangenetwork.net/node/dev_toolbox/network_exchange_protocol_v1.1.pdf

Exchange Network Node Implementation Guide 1.0

http://www.exchangenetwork.net/node/dev_toolbox/implementation_guide_v1.0_032504.pdf

Understanding Exchange Network Security

<http://test.epacdxnode.net/faq/ch02.html>

Data Exchange Resources

Data Exchange Home Page

<http://www.exchangenetwork.net/flow/index.htm>

Trading Partner Agreement Best Practices

http://www.exchangenetwork.net/flow/TPA_Final_Report_Best_Practices.pdf

Facility Identification (FRS) Data Exchange FCD

http://www.exchangenetwork.net/flow/cross/frs_fcd_v1_061804.doc

Facility Identification (FRS) Model TPA

http://www.exchangenetwork.net/flow/cross/frs_tpa_padep_2002.pdf

Getting Assistance

US EPA CDX/Exchange Network Help Desk

<http://www.epa.gov/cdx>

phone: 1-888-890-1995

Email: nodehelpdesk@csc.com

Node Mentoring Contacts

http://www.exchangenetwork.net/node/mentoring/node_mentoring_services_v1.0.doc

Exchange Network Message Board

<http://www.websitetoolbox.com/tool/mb/exnet>

Online Test Tools and Utilities

Node Developer Toolbox

http://www.exchangenetwork.net/node/dev_toolbox/index.htm

Network Authentication and Authorization Service (NAAS)

<http://naas.epacdxnode.net/>

Exchange Network Document Validation Service

<http://tools.epacdxnode.net/>