GAS INDUSTRY STANDARDS BOARD

1100 Louisiana, Suite 4925
Houston, Texas 77002
(713) 757-4175
(713) 757-2491 Fax
Email: gisb@aol.com
www.gisb.org

ELECTRONIC DELIVERY MECHANISM RELATED STANDARDS

Version ~~1.3~~ 1.4 July 31, 199~~8~~9

Special Thanks and Acknowledgments to:

GISB Member Companies for donating significant staff time to coordinate the publication of the ANSI ASC X12 guidelines.

FORESIGHT CORPORATION

For software used to develop the ANSI ASC X12 transaction sets.

GISB SUBCOMMITTEES

For support and materials describing the business practices, related data sets, data set organization, data elements and data element formats, implementation guides and mapping.

..

TABLE OF CONTENTS

VERSION NOTES

1.0 October 24, 1996

1.2 July 31, 1997

The following table shows a summary of requests and interpretations resulting in modifications to the Electronic Delivery Mechanism Related Standards. For full text of these modifications, refer to the Final Actions for Version 1.2 on GISB's home page.

Standard	Description	Request No.	Action
4.3.1	Modify standard	R97024	Modify standard
4.3.16	New Standard	R97023	Add standard

1.3 July 31, 1998

The following table shows a summary of requests and interpretations resulting in modifications to the Electronic Delivery Mechanism Related Standards. For full text of these modifications, refer to the Final Actions for Version 1.3 on GISB's home page.

Standard	Description	Request No.	Action
4.1.16	Principle	R97102/R97120	Add principle regarding Informational Postings Web Site information.
4.1.17	Principle	R97102/R97120	Add principle regarding Informational Postings Web Site information.
4.1.18	Principle	R97102/R97120	Add principle regarding Informational Postings Web Site information display.
4.1.19	Principle	R97102/R97120	Add principle regarding Informational Postings Web Site information download.
4.1.20	Principle	R97102/R97120	Add principle regarding Web site display.
4.1.21	Principle	R97102/R97120	Add principle regarding scrolling on Web sites.
4.2.1	Definition	R97102/R97120	Add definition for Informational Postings.
4.2.2	Definition	R97102/R97120	Add definition for Download.
4.2.3	Definition	R97102/R97120	Add definition for Display.
4.2.4	Definition	R97102/R97120	Add definition for Printing.
4.2.5	Definition	R97102/R97120	Add definition for Site Map.
4.2.6	Definition	R97102/R97120	Add definition for Central Address Repository.

Standard	Description	Request No.	Action
4.2.7	Definition	R97102/R97120	Add definition for Navigational Area.
4.2.8	Definition	R97102/R97120	Add definition for Content Area.
4.3.16	Standard	R97102/R97120	Revise standard regarding HTML / RTF formats.
4.3.17	Standard	R97102/R97120	Add standard regarding Informational Postings label.
4.3.18	Standard	R97102/R97120	Add standard regarding Central Address Repository.
4.3.19	Standard	R97102/R97120	Add standard regarding Central Address Repository.
4.3.20	Standard	R97102/R97120	Add standard regarding user ID or password.
4.3.21	Standard	R97102/R97120	Add standard regarding categories and labels for Informational Postings.
4.3.22	Standard	R97102/R97120	Add standard regarding navigational links.
4.3.23	Standard	R97102/R97120	Add standard regarding subcategories and labels for categories of Informational Postings.
4.3.24	Standard	R97102/R97120	Add standard regarding display of TSP identification on Informational Postings Web Site.
4.3.25	Standard	R97102/R97120	Add standard regarding the Site Map.
4.3.26	Standard	R97102/R97120	Add standard regarding search capability.
4.3.27	Standard	R97102/R97120	Add standard regarding Notices category.
4.3.28	Standard	R97102/R97120	Add standard regarding subcategories of Notices.
4.3.29	Standard	R97102/R97120	Add standard regarding labels in Notice Type column.
4.3.30	Standard	R97102/R97120	Add standard regarding display of links in Navigational Area.
4.3.31	Standard	R97102/R97120	Add standard regarding abbreviations used for Informational Postings.
4.3.32	Standard	R97102/R97120	Add standard regarding table of contents of the Tariff.
4.3.33	Standard	R97102/R97120	Add standard regarding “previous” and “next” links.
4.3.34	Standard	R97102/R97120	Add standard regarding columns not supported by TSP.
4.3.35	Standard	R97102/R97120	Add standard regarding display of Index of Customers.
7.3.24	Interpretation	C97010	Address contractual audit rights in relation to six-month time limit.
7.3.35	Interpretation	C97016	Address communication of notices.

Standard	Description	Request No.	Action
Tab 3, Executive Summary	Executive Summary	Minor Clarification & Correction	Replace “transaction set” with “EDI data” for clarity.
Tab 4, Business Process and Practices	Security	Minor Clarification & Correction	Add language to address RSA algorithm used for key generation.
Tab 6, Receiving Transactions	Writing the CGI Process	Minor Clarification & Correction	Add language regarding tag values in the HTTP header.
Tab 6, Security	Understanding PGP	Minor Clarification & Correction	Add language to address RSA algorithm used for key generation.
Tab 6, Security	Throughput Considerations	Minor Clarification & Correction	Add language regarding DNS.
Tab 6, Security	Security Requirements - PGP File Encryption	Minor Clarification & Correction	Add language to address RSA algorithm used for key generation.
Tab 6, Checklist of Testing Steps	Client/Browser	Minor Clarification & Correction	Add language to address RSA algorithm used for key generation.
Tab 7, Informational Postings Web Site	Appendices - Informational Postings	R97102/R97120	Add illustrations.

1.4 July 31, 1999

The following table shows a summary of requests and interpretations resulting in modifications to the Electronic Delivery Mechanism Related Standards.

Standard	Description	Request No.	Action
Table A	EDM Standard Error Messages	R97126	Add more error codes.
	Clean up 1.3 manual and relate information back to standards	R99036	Change wording in guidlines, add new guidelines

INTRODUCTION

The Gas Industry Standards Board (GISB) is a voluntary non-profit organization comprised of members from all aspects of the greater gas industry. GISB Standards are a product of the Gas Industry Standards Board. The GISB mission is to take the lead in developing and implementing standards across the industry to simplify and expand electronic communication, and to streamline business practices. This will lead to a seamless North American marketplace for natural gas, as recognized by its customers, the business community, industry participants and regulatory bodies.

The standards are written as 'minimums,' which industry participants are encouraged to exceed (if they are not doing so already) through provision of value-added services and customized arrangements. GISB defines 'exceed the minimum standard' to mean surpassing the standards without negative impact on contracting and non-contracting parties.

All of the standards have been adopted in the realization that as the industry evolves and uses the standards, additional and amended GISB standards will be necessary. Any industry participant seeking additional or amended standards (including principles, definitions, standards, data elements, process descriptions, technical implementation instructions) should submit a request to the GISB office, detailing the change, so that the appropriate process may take place to amend the standards.

TAB 1 Version Notes

Contains notes about this version, and, if appropriate, a brief summary of changes from the immediately preceding version.

TAB 2 Introduction

Provides a background statement about GISB's Mission and the underlying concepts behind the design and use of this guide.

TAB 3 Executive Summary

Provides a brief outline of the industry business situation which is the basis for development of this guide.

TAB 4 Business Process & Practices

Provides a brief overview of the business process and the GISB approved principles, definitions and standards related to the business process covered by this guide.

TAB 5 Related Standards

Provides a reference to any related standards.

TAB 6 Technical Implementation - Internet EDI/EDM

Provides an overview of the business process for Internet EDI/EDM.

Data Dictionary

Provides definition of the standard data elements and the usage requirements for each element.

EDI Tab

Batch Flow Diagram

Sending Transactions

Provides instructions to develop mechanisms for sending of GISB standard format data files.

Receiving Transactions

Provides instructions to develop mechanisms for receiving of GISB standard format data files.

Security

Provides guidelines for data privacy, data integrity, authentication and non-repudiation of inbound and outbound transactions.

Other Considerations

Provides information regarding error notification and testing. Includes a reference guide and examples for repudiation and validation.

TAB 7 Technical Implementation - Informational Postings Web Site

Executive Summary

The Gas Industry Standards Board (GISB) has developed standards for protocols to accomplish electronic commerce using the Internet. Technologies necessary for Internet Electronic Delivery Mechanism (EDM) to rapidly, reliably and safely move a EDI data across the Internet have been determined. Once received from a trading partner via the Internet, the EDI data is decrypted and moved through a translator or other appropriate processor for GISB standard file formats ~~such as X12~~ and forwarded to a back-end processing application. However, X12 translation and back-end processing are outside the Internet EDM scope. The scope of this document is concerned with the delivery from the output of one company's application to the input of another company's application.

This document is a high-level guide to implementing various technologies necessary to communicate transactions using the standard protocols. As such, this guide is not intended to be a comprehensive, in-depth manual. Wherever possible, this guide points to more in-depth material. The Reference section provides locations on the Internet to obtain more information as well as books and periodicals that have been recommended.

Open Standards

There are several major topic areas related to Internet Electronic Delivery Mechanism covered in this manual. When looking to implement Internet EDM, one should become familiar with the following components of the implementation:

Communications Protocols

Sending of Transactions

Receipt of Transactions

Security

The “open” standard technologies selected by GISB to address these areas are designed to provide flexibility and scalability. The specific implementation of the standards is dependent upon what fits the trading partner’s needs and available resources. A brief delineation of these components and their relationship to the model are covered at a high level in the Business Process and Practices (Business Process Description) section and in more detail in later sections of this manual.

Same Application Implementation For All Trading Partners

The basic assumption in designing and implementing the Internet EDM application is that it is not platform-specific. What is meant by this is that an organization’s Internet EDM application serves the role of communicating with all trading partners in the gas industry no matter what hardware, operating system and programming languages they use at their site. For this reason, testing with

other trading partners with a variety of platforms is very important in ensuring that your EDM application is compatible with a range of platforms used by various trading partners.

Testing With Gas Industry Internet EDM Participants

To provide a way for parties interested in Internet EDM testing to initiate testing relationships, the GISB home page will have a list of organizations willing to act as testing partners and their respective test coordinator. The FTTF meets on an intermittent basis by scheduled teleconference or in-person meetings to discuss issues, problems, further refinement of the standards. These discussions will provide a means to benchmark results and provide feedback to each other on possible enhancements to the participants' implementations. The FTTF realized that the technology being implemented is relatively new and all organizations can benefit from the sharing of research and technical information and the resolution of gas business issues integrated with the new technologies.

Importance of the Trading Partner Agreement When Using EDM

The expectations of who will perform what function and how it will be accomplished in Internet EDM should, at some level, be laid out in the trading partner agreement. This clarification in the agreement would help to expedite a smoother communication between the trading partners when first setting up their Internet EDM relationship. The newness of the Internet EDM standards and the various implementations of the applications between trading partners bring to the forefront a quandary of issues related to establishing the business rules associated with these standards. The specifications in the trading partner agreement should be tested before production implementation to formulate a solution to any problems revealed during testing well before reliance on the implementation.

Concerns About Future Reliability of the Public Internet

Continued monitoring of the Internet's viability as an infrastructure will take place. Increased traffic and potential lack of sufficient transmission capacity on the Internet is difficult to predict and quantify at this time. Concerns may be resolved by new Internet service providers and new communications technologies to compensate for the rapid growth of the Internet.

Year 2000 Compatibility

The Future Technology Task Force (FTTF) states there is no EDM standard that would preclude a company from implementing a Year 2000 compliant system.

Further Information

Please see the GISB home page at http://www.gisb.org/ for additional useful information on the implementation of Internet EDM.

Business Process and Practices

A. Overview

Where Internet EDM Fits in Gas Industry Commerce

The scope of Internet EDM is to address the communication of X12 or other GISB standard data format transactions between one trading partner's translator (or other appropriate processor for the data format) and another trading partner's translator or processor. Please refer to the diagram on the adjacent page during the following narrative as needed.

Business Reasons for Using Internet EDM

~~The question may be asked, what are the advantages of using Internet EDM to communicate our business transactions in GISB EDI standard data formats as opposed to using Value-added Networks (VANs). As an even broader question,~~ Why use EDI standard data formats for transactions ~~at all~~? With EDI, data already existing in your own computer applications can be used to build nominations and other gas industry transactions. Information from a service provider, such as scheduling, allocation, invoicing, can be mapped to a common format. This common format eliminates the need for the following as these additional steps leave room for errors, unnecessary intervention and complications in processing:

transfer data from a paper document to an application format input file at each trading partner site

if electronic files are used, mapping between various application data formats for each and every trading partner

A company that relies on computerized systems to conduct business and exchanges transactions with several trading partners can communicate those transactions more efficiently with EDI standard data formats and with Internet EDM as the communications mechanism. EDI employs standard data formats for all trading partners. By using the public Internet for transmission, a single connection is required, eliminating the complexity of different connection methods for different trading partners. ~~EDI using a VAN (Value-added Network) can rapidly become expensive if a significant volume of data is exchanged. VANs may impose charges based on number of transactions or number of characters sent, whereas, the public Internet does not impose transactions charges. In a VAN environment, transmission of transactions sent to trading partners who use a different VAN may be considerably delayed because of data transfer schedules between the VANs. The Internet EDM solution eliminates this delay because the transaction is sent directly to the trading partner's designated receipt site.~~

Roles in Electronic Commerce

In all electronic commerce, one party initiates, or sends, a transaction and the other party receives the transfer. In the Internet environment, the sender is referred to as the client and the receiver is referred to as the server. You should expect to act in both the client role and the server role during the electronic commerce process. Once a transaction set is successfully received for processing, the original receiving party switches to the client role to send a confirmation transaction back to the original sender's server. Therefore, it is essential that both the sending and receiving aspects of electronic commerce are addressed in your implementation.

The standards adopted for Internet EDM, as with all GISB standards, should be adhered to by the trading parties as minimum standards. A trading party may offer additional functions or features as options but should not require their use. Such additional features or functions are termed “mutually agreed to” in that if both trading partners agree on the inclusion, the additional feature requirements will be met. However, if either trading party does not agree to the inclusion of additional features, then the partners must allow for transmission and receipt of data using the minimum standards.

The Trading Partner Agreement is a key reference in electronic commerce. It will define the “designated site” for each partner (see the Business Practices Subcommittee documentation), values used for variable parameters, and optional features that will be used by the partners.

Assess Your Capabilities

There are many questions that readers of this narrative may want answered to clarify the standards or at least provide options for their organization’s implementation of GISB Internet EDM standards. However, the best solution for a particular organization must be determined based on the assessment of specific needs and the resources available to that organization.

Depending on your situation, you may implement the complete solution with internal resources. Given the existence of in-house systems expertise, it should be possible to implement the technologies in this guide with little, if any, assistance. On the other hand, smaller organizations may want to use this guide to identify services that they will obtain from a third party.

As much as possible, the technologies chosen for most of the programs needed to implement Internet EDM could be acquired as “shrink-wrapped” software at low cost. Where commercial quality products that can just be “plugged in” do not exist, sample code has been identified. This sample code has the drawback of being unsupported. It is intended for companies that have technical expertise but need just some starter code from which to build their own versions.

A mixture of internal expertise and third-party services will be the likely approach of several organizations. To determine where you may require the services of a third party, you should assess your present capabilities. For example, a company may have experience with X12 translators, but little experience with Internet technology at this time.

In-house Implementation

If you are choosing to implement most or all of the required functionality internally, this document is particularly pertinent. The pilot test report, posted on GISB's home page, captures "lessons learned" from those companies that participated in the pilot project.

It was demonstrated throughout the pilot test that electronic commerce using the Internet can work. However, it is strongly encouraged that all parties to fully investigate the ramifications of introducing electronic commerce using the Internet. This includes ensuring that all customer data, internal data, and applications are secure from intruders or other parties not authorized for access.

Participation in electronic commerce over the Internet will involve hardware, software, and technical expertise. Hardware requirements may include a server to receive incoming EDI files, a firewall processor to block intruder access. Software will include operating software for the servers, including the firewall, programming languages which support Internet technologies, and encryption/decryption software to provide security during the transfer. Technical expertise may be involved in the development and maintenance of server applications to process incoming files as well as applications to initiate communication with the server of your trading partner.

The GISB home page contains the text of the pilot test report and reference material that parties may utilize in evaluating and choosing hardware and software.

Using a Third Party

There are many questions that readers of this narrative may want answered to clarify the standards or at least provide options for their organizations's implementation of GISB Internet EDM standards. However, the best solution for a particular organization must be determined based on the assessment of specific needs and the resources available to that organization

> It is expected that third-party providers will offer a variety of services from a full "turn key" solution to assistance only where you require it. Such assistance might include programming, system configuration and system administration. ~~as well as private communication links such as those provided by VANs.~~

EDM Network Connections

Trading partners should maintain redundant connections to the public Internet for EDM sites. These redundant connections should be topographically diverse (duality of) paths

to minimize the probability of a single point of failure. Three possible approaches to redundant connections are:

1). A high end approach involving two ISPs and two points of connectivity, each of which was identified by the same URL making the process of redundancy transparent to the sender.

2). Maintain different Internet connectivity URLs (presumably on topographically different ISPs). For this to result in communication redundancy, the sender should know of the existence of the secondary URL and have programming in place that will automatically switch batch-browser transmissions to the secondary URL when the primary URL is unavailable.

3). Maintain multiple connections to the same ISP. This involves only one URL but the presumption would be that the ISP would provide alternate diverse paths for the URL.

Receivers may maintain multiple URLs and, if such have been disclosed, the sender should attempt to use these during primary URL outages. The redundant public Internet connections can be through a single ISP or multiple ISPs. If multiple URLs are provided for EDM access, the following conditions should be met:

- The information provided by each URL should be exactly the same
- The trading partners should be informed of both URLs and their availability by system wide notice or by Trading Partner Agreement.
- The URLs should be identified as primary and secondary if either:
- There is a TSP connection speed difference between the URLs (The faster connection listed as primary)

or

- One URL is only available when the other is down (primary URL being the most available)
- The URLs should be listed as primary and alternate if:
- The URLs have the same TSP connection speed

and

- The URLs are customarily available simultaneously

Note: A URL is considered available (in the context of communication redundancy) if all the IP facilities are properly functioning up to and including the HTTP service. This would include any TSP provided facilities including firewalls, DNS servers, routers, hubs, LANs, etc. that are between the TSP's HTTP server and the ISP's point of presence.

Note: In this context redundancy refers to normal operations redundancy (as opposed to disaster recovery contingencies).

Private network connections to access GISB EDM sites may be at any point on the TSP's firewall boundary at the TSP's discretion on a nondiscriminatory basis. The specific type and speed of their connection should be mutually agreed. It is at the discretion of the TSP on how multiple private network connections should be managed. TSPs are not responsible for any additional security exposures when using private network connections.

TCP Communications

Principle: Transportation Service Provider EDM* implementations should minimize the number of ports required to be opened on the client side firewall.

Standard: TSPs should be limited to the GISB approved list of available TCP ports for EDM* implementations included in the appendix in the Electronic Delivery Mechanism Related Standards manual under Client Firewall.

Requirements for Service Provider EDM Implementations. Guidelines: Client Firewall Requirements for Service Provider EDM Implementations. FTTF recommendations include the potential modifications needed in the client-side firewall to allow for comminations with the various service providers' EDM* implementations. The following is a list of allowable TCP ports available for use by a Service Provider. Upon request, the TSP should indicate to their trading partners which specific tcp ports they will require to be opened to conduct electronic communication.

Allowable TCP Ports (not UDP ports)

HTTP 80, 5713, 6112, 6304, 8674, 7403
SSL 443
ICA 1494
RMI(Java) 1099-1100
Java Telnet 31415
TCP Optional 8001-8020**

There are other technologies available that would require additional ports to be opened, such as FTP, Telnet, and SMTP. If and when GISB approves such technologies, FTTF will modify this list of allowable ports accordingly. The client-side firewall implementation and client browser settings should permit the downloading and installation of GISB approved plug-ins and modules. Please refer to the GISB defined Minimum Technical Characteristics for Accessing Customer Activities Web Sites for the listing of plug-ins and modules.

These guidelines will be reviewed and updated by the Future Technology Task Force, at a minimum, by the spring of each year and presented to the GISB Executive Committee for adoption by the June meeting of that group.

*All GISB standard Internet communications
**The reservation of 20 optional ports was to provide room for implementations such as DCE, IIOP, and load balancing implementations. TSPs should endeavor to minimize the usage of these ports.

Major functions of the Internet EDM Model covered by the Standards

Communication Protocols

HTTP is the standard protocol and Post is the standard method by which transactions will be transmitted over the public Internet. The content type used to package the X12 or GISB standard format file and its related parameters for the HTTP request is multi part. This provides more flexibility in the coding of the messaging components in the application because of the way it handles the delimiting of data parts passed in the body of the form as the “package” is typically called in technology circles.

Sending Transactions (Client)

It is possible to send transactions using widely-available interactive web browsers. This may be appropriate for shippers who do not have a significant number of transactions to send each day.

It was determined that in order to provide the level of automation required by some organizations such as a large pipeline company to handle the volume of transactions and the level of interface needed for possibly many back-end process applications, a fully automated batch browser is a required component of the application. In this form, the batch browser can be an event-driven mechanism used to push the transaction from the sender’s previous processes (the back-end application, the translation, and the security process) across the Internet to the trading partner’s server site where receipt of the transaction is acknowledged. The automated batch browser would also better serve the logging function of transactions being sent.

Receipt of Transactions (Server)

The receipt of transactions in the multi part HTTP Post request would require some form of Common Gateway Interface (CGI) program in order to send back a response that would notify the batch browser that it has received the transaction and whether the file in its unprocessed form and its parameters were accepted as sent or rejected. This component of the application would be able to parse out the parameters and related file and determine if the appropriate parameters had been transmitted with the file, log the appropriate statistics including a time stamp about the file and parameters, store the file and send the response back to the batch browser with the time stamp and other required response elements. After the appropriate processes have taken place in the CGI, the file would then be forwarded to the security process, any translation necessary, and finally the back-end processor.

Security

Though many decisions as to overall security measures are left to each trading partner and their environment, several security measures were established as standards to ensure a minimum level of confidence in conducting business over the Internet and to provide some uniformity in the implementation of security. Four primary security aspects were considered as vital in providing the level of protection of transactions needed for gas industry commerce: data privacy, data integrity, authentication, and non-repudiation. The FTTF found that these concerns are addressed by the use of encryption and digital signature capability of the Pretty Good Privacy (PGP) security application. Any process used for encryption and decryption compatible with PGP 2.6 (using keys generated with the RSA algorithm) meets the minimum standard to be applied to files transmitted over the Internet. To prevent unwanted intruders from connecting to the Web sites, basic authentication is the required standard. Additional issues such as firewall security are discussed in the standards, but are considered implementation issues to be addressed by each organization.

B. Principles, Definitions and Standards

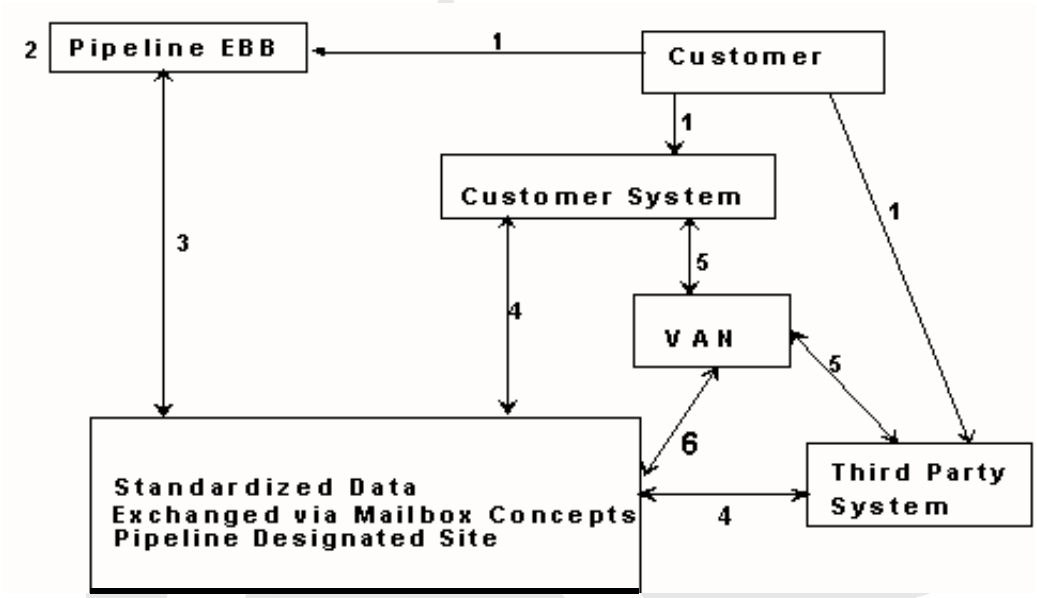
~~GISB has adopted the following principles:~~

~~4.1.1 The technology model and principles should be followed in implementing GISB's business standards electronically. The following schematic describes the EDM technology model that should exist post 4/1/97, that as agreed upon in the following standard is subject to validation:~~

~~**FUTURE TECHNOLOGY MODEL**~~

~~1. Technology and mechanisms that are at the sole discretion of the customer.~~

~~2. Technology and mechanisms that are at the sole discretion of the provider.~~



~~4.1.2 The Electronic Delivery Mechanism does not pick winners, rather it should create an environment where the marketplace can dictate a winner or winners.~~

~~4.1.3 The solutions should be cost effective, simple and economical.~~

~~4.1.4 The solutions should provide for a seamless marketplace for natural gas.~~

~~4.1.5 Data should be made available to all requesters in an accepted standard format comparable both in time and delivery mechanism.~~

~~4.1.6 Data providers (transportation service providers) should interface with third party vendors according to GISB standards.~~

~~4.1.7 Electronic communications between parties to the transaction should be done on a nondiscriminatory basis, whether through an agent or directly with any party to the transaction.~~

4.1.8 The same business result should occur regardless of the electronic delivery mechanism: this principle should guide the definition of the business process, data content of the transaction, and the timing of the transaction.

4.1.9 Service providers should post clear and precise business processing rules at the designated site, or in writing, upon request.

4.1.10 There should be at least one standard (computer-to-computer exchange of transactional data) for data exchange format.

4.1.11 The proposed future technology model reflects a minimum standard capability for 4/1/97. This model represents an ongoing process and is subject to later revisions depending on the findings of the Future Technology Task Force.

4.1.12 Protocols and tools that parties elect to support should be "Internet-compatible".

4.1.13 Regarding the request that EBBs need to provide the ability to create and print specialized reports, the data should be made available so as to permit the users of the information to download the data to be used in their applications.

4.1.14 The industry should use standard policies and guidelines for testing new data sets. These guidelines are currently being developed using the GISB guideline adoption procedures (GAP).

4.1.15 The Gas Industry Standards Board should not set standards for site-level security. Individual organization security standards should be relied upon.

4.1.16 Informational Postings Web Sites should be easy to locate.

4.1.17 Information within an Informational Postings Web Site should be easy to locate.

4.1.18 Information across Informational Postings Web Sites should be consistently displayed.

4.1.19 Information across Informational Postings Web Sites should be easy to download.

4.1.20 Display space for content on Web sites should be maximized.

4.1.21 On the Web sites, the use of scrolling, especially left to right, should be minimized.

C. Definitions

GISB has adopted the following definitions to guide industry participants in their use of the standards set forth in section D, below.

4.2.1 "Informational Postings" is the term that identifies common information, which would include the five required postings under Standard 4.3.6.

4.2.2 "Download" is the term used to describe the retrieval of information from a Web site in a format suitable for storage.

4.2.3 "Display" is the term used to describe the typical visual presentation derived by a browser as a result of retrieval of information from a given URL.

4.2.4 "Printing" is the term used to describe the typical printed layout derived when a document is printed from a display tool (browser, word processor, etc.).

4.2.5 "Site Map" is the term used to describe a Web page of URL links, which resembles a table of contents or directory tree structure, of categories and subcategories of information.

4.2.6 "Central Address Repository" (CAR) is the term used to describe: 1) the Web site providing links to all Transportation Service Providers' Informational Postings, and 2) the entity administering and maintaining the above Web site and repository.

4.2.7 "Navigational Area" is the term used to describe the area on the left side of the browser display providing links to the Content Area and other navigational links.

4.2.8 "Content Area" is the term used to describe the area directly to the right of the Navigational Area of the browser display.

D. Standards

GISB has adopted the following standards:

4.3.1 By 4/1/97, all parties sending and receiving data should accept a TCP/IP connection. At a minimum, sending and receiving parties should designate an Internet address as a designated site for the receipt and delivery of GISB standardized data sets subject to the successful completion of pilot testing by 1/1/97 to ensure that security, performance (within GISB standard data transmission time), and reliability are acceptable. The GISB data file format should be utilized. The Future Technology Task Force should determine the direction of outstanding issues such as security, archiving, receipt notification, etc., by 7/1/96.

4.3.2 On time stamping, data leaves control of the originator by the same time (deadline), regardless of mechanism (3rd party service provider time stamp is acceptable) and 15 minutes of communication time should be available to allow accumulation of all transactions to the pipeline. A standard network protocol (TCP/IP) should be in service for direct connect to the pipeline designed site by 4/1/97.

4.3.3 Originating party is any system originating/creating the document reflecting the transaction to be submitted (this could also include a third-party service provider or a transportation service provider's EBB). Within the 15-minute window the transaction should be received by the receiving party. Errors in transmission shall be governed by the terms and conditions of the trading partner agreement between the parties. The receiving party may also waive the 15-minute window requirement at its own discretion.

4.3.4 Transactional data should be retained for at least 24 months for audit purposes.

This data retention requirement only applies to the ability to recover or regenerate electronic records for a period of two years and does not otherwise modify statutory, regulatory, or contractual record retention requirements.

4.3.5 Documents that are made available on the Transportation Service Provider's designated site should be downloadable on demand in a GISB specified electronic structure.

4.3.6 By August 1, 1997 Transportation Service Providers should establish a HTML page(s) accessible via the Internet's World Wide Web. The information that is currently provided should be posted as follows:

1) Notices (critical notices, operation notices, system wide notices, etc.)
2) FERC Order No 566 affiliated marketer information. (affiliate allocation log, 24 hr. discount postings, etc.)
3) Operationally available and unsubscribed capacity
4) Index of customers
5) Transportation Service Provider's tariff (Terms, conditions and rates), or general terms and conditions.

and

Transportation Service Providers should make all pertinent EBB functions and information available via the Internet or via the technology recommended by GISB within a reasonable amount of time after each such function or information has become standardized as appropriate by GISB.

and

Within a reasonable amount of time, all EBB information, functions and transactions should be achieved via one mode of communications. Information and functions should remain available through existing systems until one mode of communication is available. Implementation time lines for this activity would be determined during the 1997 annual planning activities held in 1996.

4.3.7 At a minimum, the designated site should be accessible via the public Internet. This specifically does not preclude location of the designated site on a private intranet as long as the designated site is accessible via the public Internet.

4.3.8 The minimum acceptable protocol should HTTP. All sending and receiving parties should be capable of sending and receiving using HTTP.

4.3.9 There is a time stamp (HTTP Time-stamp) that designates the time that a file is received at the designated site. The receiving party should generate a time-stamp upon successful receipt of the complete file and send as an immediate response to the sending party. The time-stamp should be generated by Common Gateway Interface (CGI) of the receiving party, prior to further processing by the CGI.

4.3.10 The time-stamp should be included in the HTTP response back to the sender of the original HTTP transaction. It is recommended that the server clock generating the time-stamp be synchronized with the National Institute of Standards and Technology (NIST) time in order to mitigate discrepancies between the clocks of the sender and receiver.

4.3.11 The HTTP response should be sent to the sending Internet Protocol (IP) address. Other response documents should be returned to the official designated site defined in the Trading Partner Agreement.

4.3.12 As a minimum, within a trading partner agreement, one designated site for receipt should be identified for each trading partner. That site should be identified by a specific Uniform Resource Locator, (URL). This does not preclude multiple designated sites being mutually agreed to between trading partners.

4.3.13 The sender should make three attempts to complete a unit of work. After three failed attempts, it should be considered a failure.

4.3.14 The roles of sender and receiver are defined in following table. The entire table defines a unit of work: [1]

Client (Sender)	Server (Receiver)	CGI (Receiver)
	Listen for Connect	
Connect	Accept Connection	
Write	Read	Start of Receipt
Write	Read	
EOF (send)	Read	End of Receipt
Read (HTTP response)	Write (HTTP response)	
Received		
EOF (HTTP response)		

4.3.15 Trading partners should implement all security features (secure authentication, integrity, privacy, and non-repudiation) using a file-based approach via a commercially available implementation of PGP 2.6 or greater (or compatible with PGP 2.6). Trading partners should also implement basic authentication. This should be regarded as an interim solution since this technology is not an open standard. This technology supports all of the above security features while providing independence of choice of Web servers and

[1] A unit of work consists of one complete HTTP transaction as defined in the technical specification of the HTTP protocol (Internet Engineering Task Force RFC 1945). The roles of sender and receiver are also defined in that document.

~~browsers. Encryption keys should be self-certified and the means of exchange should be specified in the trading partner agreement.~~

~~4.3.16 The documents identified in GISB Standard 4.3.6 should be made available in HTML or RTF format, except with respect to the Index of Customers document which may be displayed in HTML or RTF and which should be downloadable in a defined, tab-delimited ASCII text file, with provisions for title information and footnote capability, as set forth in Code of Federal Regulations Part 284, Section 223. (Reference Order Number 581, Docket No. RM 95-4-000, issued February 29, 1996, "Appendix A, Instruction Manual for Electronic Filing of the Index of Customers" issued with the above referenced order.)~~

~~4.3.17 "Informational Postings" should be the label used for navigation to or within the Web site.~~

~~4.3.18 Transportation Service Providers should provide and keep current to the Central Address Repository the addresses (URLs) for the following in a specified format and communication method(s):~~

~~Informational Postings~~
~~Affiliated Marketer Info.~~
~~Capacity~~
~~Index of Customers~~
~~Notices~~
~~Tariff~~
~~Downloads~~
~~Site Map~~

~~This specification and any changes to it should be subject to GISB approval.~~

~~4.3.19 The Central Address Repository should make available a consolidated repository of the Transportation Service Providers' current URLs listed in Standard 4.3.18 in two ways: 1) a vehicle to link to sites and categories, and 2) a downloadable list.~~

~~4.3.20 A user ID or password should not be required to access the Central Address Repository or the Transportation Service Provider's Informational Postings Web Site.~~

~~4.3.21 The categories and the labels for Informational Postings required under Standard 4.3.6 should be as follows:~~

~~Affiliated Marketer Info.~~
~~Capacity~~
~~Index of Customers~~
~~Notices~~
~~Tariff~~

~~These categories and labels should appear in the order specified above and before any others.~~

4.3.22 The following navigational links should appear last in the Navigational Area and be labeled as follows:

Downloads
Search
Site Map

4.3.23 The subcategories and labels for the categories of Informational Postings should be as follows:

CATEGORIES	SUBCATEGORIES
Affiliated Marketer Info.	Capacity Allocation Log (when applicable)
	Discount Offers
Capacity	Operationally Available
	Unsubscribed
Index of Customers	
Notices	Critical
	Non-Critical
Tariff	Title Page
	Table of Contents
	Preliminary Statement
	Map
	Currently Effective Rates
	Rate Schedules
	General Terms and Conditions
	Form of Service Agreement
	Entire Tariff
	Sheet Index

4.3.24 The Transportation Service Provider's Informational Postings Web Site should include the name, nickname, or name abbreviation of the Transportation Service Provider so that it will appear first in the browser title bar. Content Area documents should have a similar name when printed.

4.3.25 The Site Map should be provided in the Content Area and should include links to all levels of categories described in Standard 4.3.21 and Standard 4.3.23. Each level of category and subcategory should be indented to show its relationship and should be presented in text form to best utilize space.

4.3.26 Transportation Service Providers should provide search capability for a word or phrase within the text, headers, and footers of the entire tariff and within any of the following tariff subcategories: 1) Rate Schedules, 2) General Terms and Conditions, and 3) Form of Service Agreement. The results of the search should provide a list of links to the pages

containing the word or phrase. "Search" should appear as a link and be labeled as such, appearing immediately above the Site Map link.

4.3.27 The "Notices" category (as shown in the Navigational Area) should expand to a list of subcategories (in the Navigational Area) when clicked; there are no display requirements for the Content Area. Each of these subcategories, when clicked, should display a list of notices for that subcategory in the Content Area.

4.3.28 For the subcategories of Notices, the first column headings in the Content Area should be Notice Type, Posted Date/Time, Notice Effective Date/Time (and Notice End Date/Time, when applicable), Notice Identifier (optional*) and Subject, with the list sorted in reverse chronological order by Posted Date/Time.

* When used as a reference, the Notice Identifier should be displayed.

4.3.29 The words or labels that should appear in the "Notice Type" column in Standard 4.3.28 should be:

Words	Labels
Capacity Constraint	Cap. Constraint
Capacity Discount	Cap. Discount
Curtailment	Curtailment
Force Majeure	Force Majeure
Maintenance	Maintenance
Operational Flow Order	OFO
Press Release, Company News or Phone List	News, Phone List
Other	Other

4.3.30 The links to categories of Informational Postings should be displayed vertically on the left (Navigational Area) of the screen at all times.

4.3.31 With regard to Informational Postings, when using abbreviations to display column and field names, the following abbreviations should be used:

Available	Avail
Capacity	Cap
Date/Time	D/T
Description	Desc
Effective	Eff
Location	Loc
Quantity	Qty
Maximum Daily Quantity	MDQ
Maximum Storage Quantity	MSQ

4.3.32 Each line of the Table of Contents of the Tariff should provide a link to a corresponding sheet by clicking on the sheet number shown. The subcategories Currently Effective

Rates, Rate Schedules, General Terms and Conditions, and Form of Service Agreement should provide either a table of contents or a similar breakdown, when applicable, and a link function to a corresponding sheet. For example, if General Terms and Conditions has a separate table of contents, it should provide corresponding links.

4.3.33 For Tariff documents, "previous" and "next" links should be displayed at the top of each HTML document. If the "previous" and "next" links may scroll off the display, they should also be provided at the bottom of the HTML document.

4.3.34 Columns that would contain data not supported by the Transportation Service Provider should be eliminated on display and left blank on download.

4.3.35 The header information should be displayed at the top before the columnar information. The column headings for the posting of "Index of Customers" should be displayed as follows:

- Rate Schedule
- Customer
- Contract Effective Date
- Contract Termination Date
- Maximum Daily Quantity
- Maximum Storage Quantity
- Rollover Period
- Footnotes (when applicable)

These columns should appear in this order from left to right. The data should be sorted in ascending order by rate schedule and then by customer name within rate schedule. Footnote text should be displayed below the columnar information.

E. Interpretations

GISB has adopted the following interpretations of standards that relate to Electronic Delivery Mechanism Related Standards implementation:

7.3.24 Does the language of Standard 2.3.14, 2.3.26, 3.3.15 and 4.3.4 mean that contractual audit rights are excluded from the six-month time limitation and that no statement adjustments can be made after the six-month period? In addition, is GISB recommending that audit rights be excluded from contracts or otherwise limited in contracts to a six-month period?

Interpretation:
Audit rights, to the extent they exist in a contract are contractual rights within the meaning of Standards 2.3.14, 2.3.26, 3.3.15, and 4.3.4. Further, the GISB standards make no finding or recommendation with respect to the advisability of including or excluding audit rights, specifying audit timing or specifying the timing of subsequent audit corrections in

~~a contract.~~

~~7.3.35 According to Standard 4.3.6, notices are now supposed to be posted on the Transportation Service Providers' (TSP) Web pages. Does this mean that a TSP is not required to provide any alternative form of communication for notices such as telephone or fax, particularly for those notices issued outside of business hours and on weekends?~~

~~According to GISB Standard 4.3.6, notices (critical notices, operation notices, system wide notices, etc.) are supposed to be posted on the Transportation Service Providers' (TSP) Web pages. Does this mean that a TSP is not required to provide any alternative form of communication for these specified notices?~~

~~Interpretation:~~
~~GISB Standard 4.3.6 does not specify any alternative means of notification aside from the Web page nor does it specify that the only means of notification is by means of the Web page. Alternative means of notification for particular information may be required by regulation, tariff or other GISB standards. For example notices pertaining to system wide events of both a critical and non-critical nature (GISB Standard 5.3.18) are implemented via both downloads (GISB Standard 5.4.16) and the Web pages (GISB Standard 4.3.6).~~

Related Standards

Common Codes

A decision made in 1993 by a FERC-established standards development group (EBB Working Group 5) resulted in a location coding system which cross-references proprietary point codes to a common industry-supported location code. This common location code, called the GRID Code, was developed based on the American Petroleum Institute (API) well code model. The FERC, in Order 563-A, directed the industry to establish any necessary relationships and to proceed with the implementation of the GRID Code. To achieve this implementation, in August 1994 trade associations representing three segments of the natural gas industry entered into an agreement with Petroleum Information Corporation (PI) to develop and maintain the PI *GRID* J Common Code database. As GISB prepared standards for capacity release (July 1995) and nominations (September 1995), GISB fully endorsed the use of the PI *GRID* J common codes.

However, after extensive consideration by GISB's Common Code Subcommittee, GISB adopted, on September 30, 1996, a new Common Code for Gas Transaction Points, the GISB/PI Data Reference Number (generally referred to as "DRN"). The DRN is a one-to-nine digit, non-intelligent number also assigned by PI, which has a one-to-one relationship with the PI GRID J Code. In 1998 the maintenance of DRNs was asssigned to IHS Energy Group. For more information, call GISB at (713) 757-4175 or access the GISB Web Page at www.gisb.org.

In keeping with the trends in other industries involved with EDI, EBB Working Group 5 recommended the acceptance of the D-U-N-S® Number as a common company identifier. This recommendation was also adopted in FERC Order 563-A. The D-U-N-S® Number is assigned to companies by the Dun & Bradstreet Corporation (D&B). Similarly, as GISB prepared standards for capacity release (July 1995) and nominations (September 1995), GISB fully endorsed the use of the D-U-N-S® Number common code.

However, after extensive consideration by GISB's Common Code Subcommittee, GISB, on December 10, 1996, did confirm the use of the D-U-N-S® Number, but with a major refinement: For GISB Common Code purposes, a legal entity will use one and only one D-U-N-S® number. Since D&B offers customers the option of carrying more than one D-U-N-S® number per legal entity, please refer to GISB's Web Page at www.gisb.org for directions on determining the one and only one D-U-N-S® number constituting the GISB Legal Entity Common Code.

In the datasets, an asterisk by a data element means that it is a "common code," so the field will reflect the industry-supported common code for location or company.

Model Trading Partner Agreement

In 1995, GISB drafted a Model Trading Partner Agreement (MTPA) for exchange of data within the gas industry. The GISB MTPA defines the relationship of the sender and receiver of GISB Standard ~~ASC X12~~ documents. This agreement represents a complete set of balanced terms which a company should accept whether it is sender or receiver of electronic documents. It has established all the data items necessary to exchange electronic documents in a step by step, fill in the blank model form. GISB endorses the use of the MTPA in order to minimize preparation, negotiation and review time. This will allow more time for implementation of electronic commerce. Copies of this agreement may be obtained from the GISB office.

Party Roles

In all of the transaction sets, there are multiple parties that may be involved in the transaction. There are the Transportation Service Provider (a.k.a. Pipeline or Transporter), the Service Requester (a.k.a. Shipper), Service Requester Agent (a.k.a. Shipper's Agent) and Third Party Service Provider (a.k.a. Third Party Agent). It is important to distinguish between the role of the Service Requester Agent and the Third Party Service Provider.

The Service Requester Agent is the party contractually authorized by the Service Requester to submit business transactions to the Transportation Service Provider on behalf of the Service Requester for a service requester contract. Once the Service Requester Agent is contractually authorized, the agent becomes the Service Requester for subsequent business transactions unless and until the agency relationship is terminated.

The Third Party Service Provider is the communications agent that the Service Requester or Service Requester Agent may subscribe to in order to send and receive transactions with the Transportation Service Provider.

It is possible that a single entity may, at times, provide the role of a Service Requester Agent for one party while providing the role of Third Party Service Provider for another party. Likewise, a single entity could be both Service Requester Agent and Third Party Service Provider for a single party.

In EDI implementation, the party that is authorized to send and receive transactions will be the party identified in the transmission envelope (ISA Header Segment). If the sending party is a Service Requester, Service Requester Agent or Third Party Service Provider, their appropriate identifiers will appear here. In all cases, the Transportation Service Provider, Service Requester and Service Requester Agent (if applicable) will be identified in the body of the transaction (N1 Name Segment).

ANSI ASC X12 Standards

The GISB standards reflect an industry utilization of the American National Standards Institute (ANSI) ASC X12 standards maintained by the Data Information Standards Association (DISA).

The technical implementation documents included in this manual reflect GISB's subset of the ANSI ASC X12 version 003040 standards. It is recommended that any industry participant who wishes to utilize the ANSI ASC X12 standards should also have a copy of the ANSI ASC X12 Standards Reference document for a full understanding of the X12 requirements. GISB members may purchase an ANSI reference document through GISB by contacting the GISB office. Non-GISB industry participants may purchase the reference document by contacting:

Manager of Publications
DISA
1800 Diagonal Rd, Suite 355
Alexandria, VA 22314-2852
705-548-7005

As a member of ANSI, GISB will utilize the ANSI ASC X12 standards and remain in full compliance. In all standards, occasions arise where the standard does not fully meet a need. GISB recognizes this and will add interim usages and code values when required. When GISB utilizes an interim solution, GISB will apply to ANSI and the appropriate ANSI organizations for acceptance of the interim solution. ANSI's final solution may provide a usage or code value different than the interim solution. GISB standards will be updated to reflect the final solution.

The architecture of ASC X12 is designed for end to end communications. The translator that generates the ASC X12 file and envelope will assign control numbers and counts that will appear within the ISA/IEA segments of the transaction and within the GS/GE segments of the transaction. These numbers and counts allow the translator to ensure that all of the segments in an envelope and all of the data elements in an envelope have been received and that the transmission was complete.

ISA contents

The ISA segment marks the beginning of an X12 document. It can be equated to an envelope that a paper document would come in via the mail. The envelope may contain one or more functional groups (defined by the GS segment) and one or more transaction sets.

The ISA is the interchange control segment to be utilized on all GISB X12 standards. The segment identifies the sender and receiver of the document. The Interchange Sender ID/Interchange Receiver ID is published by both the sender and receiver for other parties to use as the sender/receiver ID to route data to them. The sender must always code the sender's ID in the sender element and the designated receiver's ID in the receiver ID. Trading partners utilizing a password for their documents will use the Security Information element. The receiver of the document identifies a password for the sender to include in this element.

There are additional elements in the ISA segment. These elements are traditionally assigned by the sending party's translator. These elements inform the receiver of the date/time that the envelope was generated, the X12 version number being utilized, whether the transmission is for test or production purposes, and what characters were used to designate the end of a sub element, element or segment. Different characters must be chosen for the sub element, element and segment delimeters. These delimiting characters must never appear in the data.

GS contents

The GS segment indicates the beginning of a functional group and provides control information for the data that follows it. A functional group can be defined as a group of transactions related to one business application. Within a mailing envelope, there may be a bundle of information relating to imbalances and a bundle of information relating to measurement information. Each of these 'bundles' is sent within its own (or a separate) GS Functional Group Header and a GE Functional Group Trailer in the X12 environment. The sender of a transmission provides the Application Sender's Code that the receiver of the transmission will reflect back on acknowledging documents. The receiver of a transmission provides the Application Receiver's Code that the sender will include in the transmission for the receiver to utilize in routing to internal applications. Group Control Numbers are originated and maintained by the sender of the document.

997 Usage

The 997 Functional Acknowledgment is used to indicate the results of the syntactical analysis of the X12 documents. The documents include the transaction sets and functional groups with an ISA/IEA envelope. This standard covers all of the X12 and GISB standard criteria that the receiver of the document has incorporated into the receiver's translator. The translator may be set to accept all information into the receiver's application processing, it may be set to accept only ANSI ASC X12 compliant information into the receiver's application processing, or it may be set to accept only ANSI ASC X12 and GISB compliant information into the receiver's application processing. Compliance checking, in a translator, may be set to any of several levels. GISB recommends that compliance checking be set to the element level in the Functional Acknowledgement.

The 997 informs the originator of the transaction whether the translator accepted the file, accepted it with errors, or rejected it. When errors occur, the 997 tells the location and type of error that was encountered. Once a transaction passes the translater, the 997 is sent to the originator of the transaction and the data (if accepted) is passed on to the receiver's business application for processing.

Hypertext Transfer Protocol (HTTP)

The Hypertext Transfer Protocol (HTTP) is an application-level protocol with the lightness and speed necessary for distributed, collaborative, hypermedia information systems. It is a generic, stateless, object-oriented protocol which can be used for many tasks, such as name servers and distributed object management systems, through extension of its request methods (commands). A feature of HTTP is the typing of data representation, allowing systems to be built independently of the data being transferred.

HTTP has been in use by the World-Wide Web global information initiative since 1990. This specification reflects common usage of the protocol referred to as "HTTP/1.0".

TECHNICAL IMPLEMENTATION - INTERNET EDI/EDM

Technologies Selected by ~~the~~ GISB

The transport protocol for communication of future GISB transactions should be TCP/IP. In addition, standard Internet protocols should be chosen for specific tasks. Various Internet protocols were considered to accomplish to delivery of a transaction at the application protocol level. The Hyper-Text Transfer Protocol (HTTP) was chosen.

Practical information systems require more functionality than simple retrieval, including search, front-end update, and annotation. HTTP allows an open-ended set of methods to be used to indicate the purpose of a request. HTTP is also used as a generic protocol for communication between user agents and proxies/gateways to other Internet protocols, allowing basic hypermedia access to resources available from diverse applications and simplifying the implementation of user agents.

There are two primary Internet software components involved in Web communications. The first is called a browser and runs as client software. The second is called a Web server, or HTTP server and usually runs on a dedicated server computer.

The standard data elements, each with element name and description, have been defined in the Section "Data Dictionary For Internet EDM". The following two sections identify what is involved in sending and receiving transactions. After that comes a discussion regarding the securing of the transactions to be sent. The remaining sections cover considerations for other aspects of the overall process. While these were not the focus of the Internet EDM process as mentioned above, selected topics that may affect your overall implementation are discussed.

Data Dictionary For Internet EDM

Business Name	Definition	Format	Usage*	Condition
from**	the party sending the transaction	Common Code Identifier ~~format~~;Alphanumeric 13 bytes maximum	in Request; M	used in file transmittal; displayed in HTTP response; and, used in posting back decryption-related errors
input-data	the filename for the transaction data set transmitted	including drive letter and directory name with filename if needed	in Request; M	~~used in file transmittal of any 10 HPDRs; and, used for posting back all transaction value pairs for a transmittal that had decryption-related errors~~ Used for documentation purposes only.
input-format	descriptor of the data format used for the file transmitted	X12~~,~~ FF error	in Request; M	"X12",FF, or other GISB standard format indicator used in file transmittal; "error" used in posting back any ~~decryption-related~~ processing errors
request-status	status describing success or failure of transmission at recipient server	ok; EEDM###:error description; WEDM###:warning description. see Table A, "Internet EDM Standard Error Codes and Messages"	in Response; M	"ok" is returned if all is fine with the CGI processing; error messages/warnings and their related descriptions are returned if problems were encountered in ~~CGI processing or in the decryption process~~ processing.
server-id	uniquely identifies the server and CGI processing the transaction	*domainname* or *hostname.domainname*; no embedded spaces allowed	in Response; M	displayed in the HTTP response ~~and posted back for any decryption-related errors~~
time-c	the time file transfer is complete at the server	*yyyymmddhhmmss*	in Response; M	displayed in the HTTP response~~ and posted back for any decryption-related errors~~
to**	the party the transaction was sent to	Common Code Identifier ~~format~~;Alphanumeric 13 bytes maximum	in Request; M	used in file transmittal and displayed in HTTP response ~~and posted back for any decryption-related errors~~
transaction-set	name of the document type being sent	8 character code; examples are: G811TSIN, G820PYRM, G860PDAL, G811IMBL, G865ALLC, etc.; please refer to ~~GISB Implementation Standards~~ the table in Tab6 titled GISB Transaction Codes.	in Request; MA	used in file transmittal

trans-id	sequential number assigned to the transaction by the server CGI upon processing before being passed to the decryption process	integer up to 15 characters in length	in Response; M	displayed in the HTTP response ~~and posted back for any decryption-related errors~~

*The **Usage** column defines whether the element appears in the HTTP Request (Client-generated) or the HTTP Response (Server-generated), the order in which the element appears in the data stream, and whether the field is Mandatory (M) or Mutually-Agreed-To (MA).

** Common Code Identifier

This Section Contains **Page**

Batch Flow Diagram

SENDING TRANSACTIONS

General Flow

Sending a batch EDM
1. Open HTTP connection
2. Check connection status. If in error requeue file according to GISB standards (this check should be performed here and throughout the following processes)
3. Post
   A. Authentication (password must be uuencoded)
   B. Send multipart form
   C. Receive HTTP response data
4. Check connection status. If in error requeue file according to GISB standards
5. Check HTTP status code (200 is good, less than 300 may be acceptable). If status is not successful requeue file according to GISB standards
6. Close connection - wait for other end to close in a reasonable time
7. Parse HTTP response data elements
8. If request-status ok, then log success
9. If request-status error, then log error
10. If no valid request-status requeue file according to GISB standards
11. Remove file from sending queue when successful or when failed completely

HTTP Post

Most people think of the Web as the process of using a browser to fetch, or download, documents, not upload them. Indeed, this capability is most prevalent. HTML pages, text files, and other documents can be retrieved by a browser using HTTP, FTP, or other protocols. However Web browsers allow the user to input data to a server using HTML forms. Data is entered into the fields of the form and is transmitted to the server by pressing a pushbutton or hitting the enter key.

The HTTP protocol has two methods for transmitting a request to a server. Both methods return a response to the client, which may be a document retrieved from the server. Both methods can be used to transmit form data. The GET method is the simplest and is used for requests that pass a small amount of information. Data passed with the GET method must be translated into a special format known as “URL encoding.” Furthermore, the data stream transmitted by the GET method has a limit of 1024 characters. The POST method, on the other hand, allows the upload of complete datasets without special encoding. It is this method which will be used to send GISB standard format transactions and receive the response from the server.

Using an Interactive Browser

When most of us think of Web surfing, we think of using an interractive browser. When you enter an HTTP Uniform Resource Locator (URL), the browser opens the HTML document identified by the URL. Basically, a URL is an “address” of an HTML document on a Web server. For purposes of GISB standards Uniform Resource Locator (URL) is as defined by the Internet Engineering Task Force (IETF).

In order to use an interactive browser to upload data, an HTML document must be created for that function. The HTML document can reside on either the server to which you are uploading or the client’s system. The “form” feature of HTML allows that within an HTML document, a form can be created which allows the client to type in any necessary data elements, such as to, from, and input format and then specify a file to be uploaded from the PC. Some type of “Send” button would be on the form and when selected, the form would cause an HTTP POST to be issued, thereby uploading the file. Below is an example of an HTML document with a form which specifies the POST method and contains the required data elements.

An HTML form like that described here could be used with any retail browser that supports multipart POST with a file upload. When choosing a packaged browser, it is mandatory that it supports multipart encoding.

Sample of HTML document with a form to perform a multipart post using an interactive browser:

```
<HTML>
<HEAD>
<TITLE>GISB File Upload</TITLE>
<H1><CENTER>GISB File Upload</CENTER></H1>
</HEAD>
<HR>
<BODY>
<form ENCTYPE="multipart/form-data" ACTION="http://www.target.server/cgi-bin/upload.exe"
METHOD=POST>
Enter Common Code Identifier for From and To
From: <input TYPE="text" NAME="from" SIZE=20 VALUE=""><br>
To: <input TYPE="text" NAME="to" SIZE=20 VALUE=""><br>
Format of this file: <input TYPE="text" NAME="input-format" SIZE=6 VALUE="X12"><br>
Send this file: <INPUT NAME="input-data" TYPE="FILE"><br>
<input TYPE="submit" VALUE="Send File"><br>
</form>
</BODY>
</HTML>
```

The non-bolded text in this example is the basic HTML required for a document and allows your page to show a title in the title bar. The bolded text is the form within the document and is described in more detail.

The important characteristics of the form within the HTML document are:

ENCTYPE= specifies the encoding type. The “multipart/form-data” encoding type is identified as the standard encoding methodology.

- ACTION= specifies the URL that will receive the uploaded data. The Trading Partner Agreement identifies the URLs for both parties.
- METHOD= specifies the HTTP protocol method. “POST” has been defined as the GISB standard method.
- <input ...> Five input areas are specified on this form: from, to, file format, file name, “Send File” button.

NOTE: This document often refers to “multipart POST” which implies the encoding type and method as described in this example.

When a user selects the “Send File” button, the browser will take the values entered in the input fields and reformat them according to the encoding type into a data stream. For the file identified for upload, the file is opened and its contents are included in the data stream, rather than the file’s name. The data stream is then sent to the URL specified by **ACTION=**. The URL will indicate an HTTP server script or program written to receive the data.

For a smaller site only performing a few transactions or file transfers this manual process would be viable as a primary transmission tool. This method could also be considered a back-up method to any batch or automated process that may be implemented. If the client provides its own form, the form can be copied for each trading partner. The only change to the HTML would be to modify the URL shown for the **ACTION=** attribute.

Using a Batch Browser

For companies that have automated much of their back-end process and prefer to avoid unnecessary human involvement, a so-called "batch browser" is needed. This browser needs to be capable of program-based or script-based initiation. At this time, there are few off-the-shelf batch browsers which use the POST method. Most packaged batch browsers use the GET method.

However, a batch browser can be created using custom programming. The batch browser will be coded to perform all of the same formatting that the interactive browser performed to send a data stream which conforms to the HTTP protocol. A batch browser must be coded as a sockets program. See Section “Writing a Batch Browser”.

A sockets program can be written with various programming languages which offer the required library to achieve this function.

Authentication

HTTP basic authentication includes a userid and password. Interactive browsers include a basic authentication feature which automatically prompts for userid and password. In a batch browser, the authentication must be specifically coded. The userid and password are to be UUEncoded within the document header. UUEncoding utilities are readily available on the Internet as either public domain software or commercial libraries.

Server Response

The receiving server will send an HTTP response to the client before dropping the client's connection. The response returned from the Web server will contain timestamps that include a timestamp recorded when the final byte from the file upload is received and stored. This timestamp is the official timestamp regarding transaction turnaround deadlines defined in GISB standards. This timestamp and all other pertinent file transmittal information should be logged when the posted file is stored on the receiving server as well as logged by the client . Likewise, any errors or warnings should be logged at both the server and client.

Throughput Considerations

The performance of the batch browser is one component critical in meeting deadlines. It is conceivable that it may be called many times for a busy site (such as a pipeline sending quick responses). It should therefore utilize whatever performance techniques that are possible. For example, it may be desirable to write a multithreaded version which can handle a certain number of requests simultaneously with a single copy of the program.

HTTP Request Data Elements

Required Data Elements (listed in the required order)

Data Element Name	Description
from	Common Code Identifier of sending/client company
to	Common Code Identifier of receiving/server company
input-format	Descriptor of the data format within the input data set.
input-data	The properly formatted file of electronic commerce data.

Mutually Agreed Upon Data Elements

Data Element Name	Description
transaction-set	Descriptor of the transaction types included in the input-data. The values used must be from the unique 8-character names defined in the Implementation Standards. See ~~GISB Standards~~ the following table for the various transaction types and their corresponding 8-character names.

Processing of all HTTP data elements should be case insensitive.

GISB Transaction Codes

Writing a Batch Browser

A batch browser needs to simulate the actions of an interactive browser. As stated earlier, the interactive browser will take the HTML form and reformat the information according to the HTTP protocol before it sends the data stream to the HTTP server. The reformatting involves adding a header and placing field delimiters around the data items. A batch browser needs to produce the same kind of data stream and therefore, writing a batch browser requires some specific knowledge of the HTTP protocol. See the GISB home page for sources of HTTP protocol information.

First, consider the header:

Sample of a typical header sent to the HTTP server

```
POST c:\execute HTTP/1.0
Referer: http://www.get.a.life/upl.htm
Connection: Keep-Alive
User-Agent: brow v0.1 XYZ Corp.
Host: localhost
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Content-type: multipart/form-data; boundary=---------------------------87453838942833
Content-Length: 5379
```

This information is documentary in purpose. The parts that are important are:

The first line: *POST c:\execute HTTP/1.0* indicating that the POST method is used and which program to call.

The content type line:

Content-type: multipart/form-data; boundary=---------------------------87453838942833

The content-type element indicates that the encoding method is multipart. It also identifies the character string used as the boundary. The boundary will appear between each field as a delimiter. In this example, the boundary is comprised of 27 hyphen characters followed by a number.

The boundary can be any character string that you choose except that it is required that it will not to occur anywhere else in the form or in the transaction being sent. This is usually accomplished by using either the system clock or a random number so that even if by some remote chance the string appears in the document it would not appear in any re-transmission of the file. It is strongly recommended that a relatively long string be used as a boundary. The boundary when used as a separator requires two hyphen characters appended to the front of the string as you can note by the lines between the data fields in the example. The last boundary required in the form is two hyphen characters appended to the back of the separator boundary, this is used to indicate to the server program that this is the end of the data.

The content length:

Content-Length: 5379

The content-length value should match the number of bytes contained in the entity body including the characters in the boundary lines, variable content, blank lines, etc. In essence, it tells the server how much is going to come after this point.

In this example, the data portion, or body, sent to the server program is as follows and assumes only required data elements are sent (not mutually agreed data elements):

```
----------------------------87453838942833
Content-Disposition: form-data; name="from"

123456789
----------------------------87453838942833
Content-Disposition: form-data; name="to"

234567890
----------------------------87453838942833
Content-Disposition: form-data; name="input-format"

X12
----------------------------87453838942833
Content-Disposition: form-data; name="input-data"; filename="c:\temp\smallnom.bin"
Content-Type: application/octet-stream

ISA~00~          ~01~AAA6300300~14~1234567890000  ~14~2345678900000

... more data from the X12 file…

IEA~1~000003616
----------------------------87453838942833--
```

The important characteristics of the above stream are:

- The boundary string appears at the beginning of each data field in the body.

- For each body data field, two identifiers define the contents of the data field. The Content-disposition identifier defines that "form-data" is contained in the element. The name identifier defines the name of the data element. These data element names must match the name specified by GISB. The name identifier is not completely relevant since the fields should be present in the correct order but this field should be checked to verify the validity of the form content.

- The actual data value of the field is always preceded by a line termination. This is typically used as a marker for the server program to indicate that a data value will follow. For example, note the blank line preceding "X12" in the above sample. In most programming libraries and commercial products the starting delimiter is "\r\n\r\n" (c notation).

- The data field containing the ~~X12~~GISB standard file has two extra identifiers: first the name of the file sent from the source computer, *filename="c:\temp\smallnom.bin"*, and second a content type identifier on a separate line. This line should always be shown as:

"Content-Type: application/octet-stream" . This indicates that the content of the file should be treated as binary and not converted in any manner.

- After the contents of the last data field, the boundary appears again as the last item of the form with the required two hyphen characters following the boundary at the end of the form to indicate the end of the data.

Although the specifications for multipart POST include several variations on this method, the GISB standards do not include implementing them at this time. The most significant of these variations is to send several files in a single post. Additionally, sending a single file split into more than one post is not expected by the HTTP server.

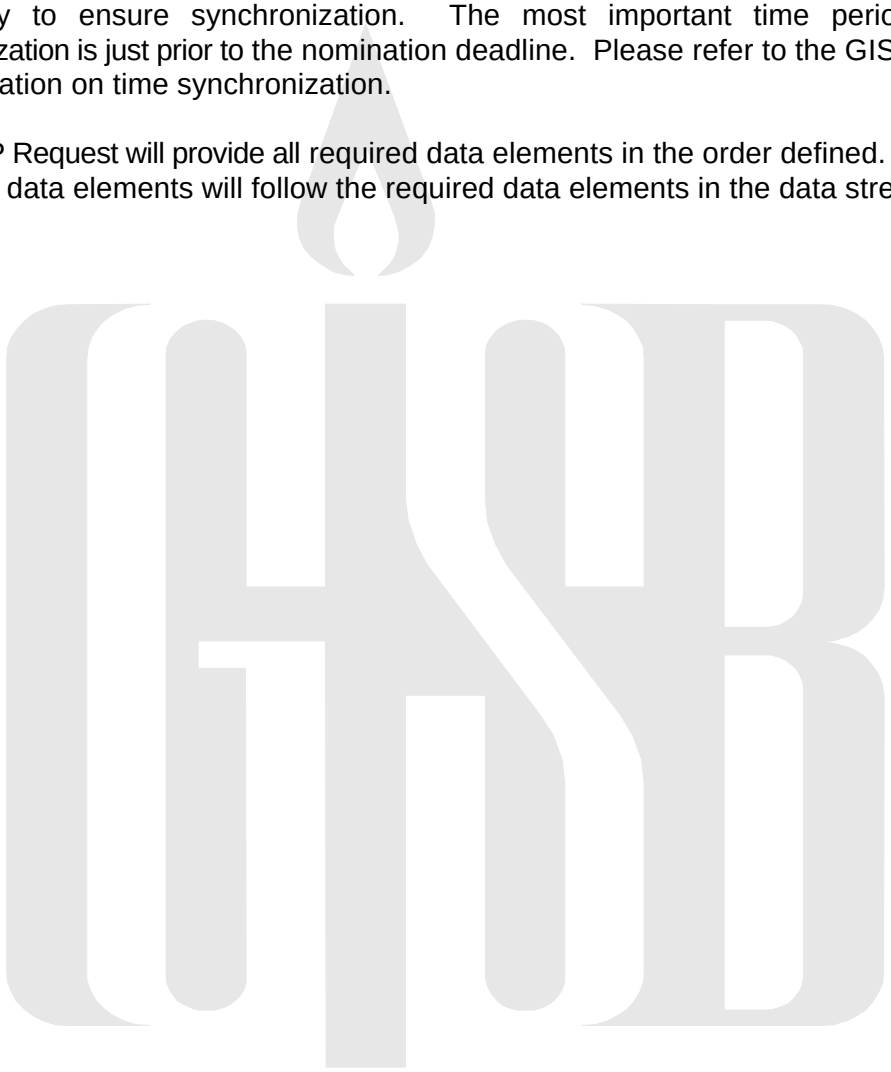
The output from the browser is important to the understanding of the processing needed by the server script or program which must interpret the result. The complete data stream from the browser will look like:

```
POST c:\execute HTTP/1.0
Referer: http://www.get.a.life/upl.htm
Connection: Keep-Alive
User-Agent: brow v0.1 XYZ Corp.
Host: localhost
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Content-type: multipart/form-data; boundary=---------------------------87453838942833
Content-Length: 5379

-----------------------------87453838942833
Content-Disposition: form-data; name="from"

123456789
-----------------------------87453838942833
Content-Disposition: form-data; name="to"

234567890
-----------------------------87453838942833
Content-Disposition: form-data; name="input-format"

X12
-----------------------------87453838942833
Content-Disposition: form-data; name="input-data"; filename="c:\temp\smallnom.bin"
Content-Type: application/octet-stream

ISA~00~          ~01~AAA6300300~14~1234567890000  ~14~2345678900000

... more data from the X12 file…

IEA~1~000003616
-----------------------------87453838942833--
```

Client Specifications

Each client should be synchronized to Central Time (Central Standard / Central Daylight) available at any of the sites on a synchronized network of atomic clocks. Each trading party should observe the client clock over a period of time to determine the amount of "drift" occurring throughout the day. The client should be synchronized as many times per day as necessary to ensure synchronization. The most important time period to ensure synchronization is just prior to the nomination deadline. Please refer to the GISB home page for information on time synchronization.

The HTTP Request will provide all required data elements in the order defined. Any mutually agreed to data elements will follow the required data elements in the data stream.



RECEIVING TRANSACTIONS

General Flow

1. Parse multi-part form
2. Validate HTTP request data elements
3. If HTTP request data elements in error, return appropriate standard error code in the HTTP response data elements
4. Save data
5. Create time stamp
6. Return HTTP response data elements back to server
7. Close connection
8. Log final results
9. Route data file to the next process based upon input format

Using a Web Server

As was stated above, the protocol HTTP using the POST method as the means to upload a transaction is the standard. On the receiving side of this HTTP request is the Web server, the second primary component in Web technology. However, the Web server does not actually save the uploaded file. Instead, it hands this responsibility over to a special program which, in effect, extends the Web server's functionality with custom programming. This special program is known as a Common Gateway Interface (CGI) program. Besides storing the file, the CGI program has the task of parsing the incoming HTTP message, noting the time so to create the timestamp, and creating an HTML response to the sending browser.

The GISB standard places no particular requirements on the vendor for the Web server. Most commercially available Web servers will provide the needed functionality. However, please refer to comments regarding performance under "Throughput Considerations" later in this section. While the current approach to security does not require a Secure Sockets Layer (SSL) or Secure Hyper Text Transfer Protocol (S-HTTP) capable server, one of these may be a requirement in the future. Determine whether the product you are considering provides a secure version capable of either SSL or S-HTTP. (Unfortunately, it is too early to predict which of these, if either, will prevail as an emerging standard.)

Another capability you may wish to consider when choosing a Web server is whether it supports Binary Gateway Interface (BGI) capability. Specifically, this is the capability to run Dynamic Link Library (DLL) equivalents of CGI applications. Some vendors call this capability Internet Server Application Programming Interface (ISAPI) while others call it Netscape Application Programming Interface (NSAPI).

The CGI Process

A CGI (or BGI) program must be able to parse the multipart form. It accomplishes this by finding the boundary string in the Content-Type header and scanning for its occurrences further within the uploaded stream. Upon finding these boundary strings, the program must

next determine the content-disposition for each data element. This allows detection of the required text elements as well as the GISB standard format file.

The CGI program is not concerned with the content of the GISB standard format data. In fact, the standard format file will be encrypted (see the Security section). The CGI will merely accept the standard format data and store it as a file. The CGI will use the Content-Length to determine how much data to expect in the body.

Throughput Considerations

It is critical that the Web server and the associated CGI programs perform efficiently. This is particularly true for pipelines which may expect to see a large number of nomination transactions come in close to the deadline. For the greatest possible throughput, the Web server should be multithreaded. The CGI program should be multithreaded as well or be small and efficient as is possible with a C program. BGI programming may provide even better performance. It is also suggested that a Web server and operating system be chosen that allow for scaling to a more powerful computer (possibly multi-CPU). Transaction volumes are likely to be light at first but may become heavy rather quickly.

Writing the CGI Process

A CGI process is the executable program or module that is called by the HTTP server when it is identified by a POST or GET operation. (In this case we are only concerned with POST method operations.)

When the HTTP server receives a POST it will first read the header and populate environment variables before calling the CGI. A sample header is shown below.

```
POST c:\execute HTTP/1.0
Referer: http://www.get.a.life/upl.htm
Connection: Keep-Alive
User-Agent: brow v0.1 XYZ Corp.
Host: localhost
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Content-type: multipart/form-data; boundary=---------------------------87453838942833
Content-Length: 5379
```

The important point to note is that you will not specifically code the step of reading the header and populating the environment variables, the HTTP server performs it for you. The variables populated are usually listed with the HTTP server documentation.

After reading this header the server will buffer the remaining data transmitted and then call the CGI process specified in the POST statement. Do not assume that the CGI process is called as soon as the header is read. The more common implementations will buffer the entire transmission before calling the CGI. You may want to check your server implementation if this characteristic is important to you.

The called CGI process will have the following stream available in the standard input (stdin) and most of the header data available in environment variables.

```
----------------------------87453838942833
Content-Disposition: form-data; name="from"

123456789
----------------------------87453838942833
Content-Disposition: form-data; name="to"

234567890
----------------------------87453838942833
Content-Disposition: form-data; name="input-format"

X12
----------------------------87453838942833
Content-Disposition: form-data; name="input-data"; filename="c:\temp\smallnom.bin"
Content-Type: application/octet-stream

ISA~00~          ~01~AAA6300300~14~1234567890000  ~14~2345678900000

... more data from the X12 file…

IEA~1~000003616
----------------------------87453838942833--
```

This process should check for basic validity in the environment variables and the data stream. It will parse the variables/data from the format. The data validations should include:

- The "REQUEST_METHOD" environment variable is "POST".
- The "CONTENT_TYPE" enviroment variable should be "multipart/form-data" and a boundary, which is unique in that it cannot appear anywhere in the transaction being sent (see above stream for an example).

The input stream should be in binary mode to accommodate encrypted files.

- Each data element is be preceded by the boundary with the required two hyphen characters appearing before it.
- Each data element should contain the correct name on the *Content-Disposition* line.
- Each data element should have \r\n\r\n (c notation) before the start of the data.
- In the receiving program, all tag values in the HTTP header should be evaluated in a case insensitive manner.

Finding the end of the stream using both content length and the boundary end mark (the boundary with two required hyphen characters in front and behind) is usually the best method to detect improperly formatted input.

Immediately after the CGI validates (as above), parses, and saves the data, the CGI should record the time and construct a response described in the following section. This response is usually sent from the CGI by writing to the standard output (stdout) of the CGI process.

URL/CGI Implementation Guidelines

GISB standard 4.3.12 states

> *"As a minimum, with a trading partner agreement, one designated site for receipt should be identified for each trading partner. That site should be identified by a specific Uniform Resource Locator (URL). This does not preclude multiple designated sites being mutually agreed to between trading partners.*"

This standard specifies that each company must offer at least one URL (URL is a one-to-one association with CGI) to accept ~~EDI~~EDM files. However, a maximum number of URLs per company is *not* included so that companies that wish to offer additional URLs will not be held back from doing so. Though companies are free to construct an ~~EDI~~EDM Web site with multiple "single-purpose" URLs, GISB recommends the use of one "general-purpose" URL.

Error notifications include errors that occur some time after the HTTP response is sent (such as a file decryption error) as well as errors on the ~~X12~~ data transactions. A general-purpose URL would handle all error notifications.

Companies that wish to offer multiple URLs must negotiate additional URLs with their trading partners. All URLs that will be required for use in the ~~EDI~~EDM process must be agreed to and defined in the Trading Partner Agreement (TPA) signed by both companies. An example of a company that would define multiple URLs in the TPA is a company that comes to agreement with its partners that all nominations-related transactions are sent to a URL offered by an out-sourcing vendor. All other transactions are sent to a URL offered on its own Web server.

A company can also offer additional URLs which have a special purpose without defining the URL in a TPA. Such additional URLs would be a way of offering additional customer service. The trading partners would have the option of using the additional URL. An example of a company that offers a URL for additional customer service is a company that offers a URL to accept capacity release information requests with immediate turnaround while the general-purpose URL is set up to postpone all capacity release information requests until 4 p.m. that day. This company wishes to keep its primary Web server available for nominations requests while other information requests are handled on a secondary Web server.

To those companies who wish to offer multiple URLs, GISB strongly recommends that you divide URL usage along transactional grouping lines, such as nominations or capacity release. Create groupings that are likely to correlate to business functions in a company within the gas

industry. Do not divide URL usage along an arbitrary internally-understood group such as region of the country. Remember that the intent of not specifying a maximum number of URLs is to allow companies the freedom to offer services, not to further complicate the ~~EDI~~EDM process.
Some companies have raised a question of offering a "default" URL. The default URL would be used when the trading partner was not able to determine the proper URL from the trading partner agreement. GISB does not recommend that any company offer a default URL. When situations arise where the TPA does not fully define the appropriate URL, the partners should communicate the situation, agree to the appropriate URL usage, and revise the TPA.

Server Specifications

The HTTP server should be synchronized to Central Time (Central Standard / Central Daylight) available at any of the sites on a synchronized network of atomic clocks. Each trading party should observe the server clock over a period of time to determine the amount of "drift" occurring throughout the day. The server should be synchronized as many times per day as necessary to ensure synchronization. The most important time period to ensure synchronization is just prior to the nomination deadline. Please refer to the GISB home page for references on public sites for synchronization.

The HTTP server will provide an HTTP response to the client according to GISB standards.

- All data element names of the HTTP request and response fields will be in lower case. Note that the GISB standard format file contained in the request and response may follow a different standard.

Carriage returns and line feeds will be ignored in all files.

A field delimiter of "*" will be used in the HTTP response. Please refrain from displaying a "*" anywhere else in the response so as not to confuse programs that need to parse on this basis.

No spaces should surround the equal sign or the field delimiter.

- The required data elements must appear first in the response.

Additional information can be included after the required elements at the server's discretion.

- The HTTP response must be enveloped by opening and closing HTML tags at a minimum.

The HTTP response must be no more than 2048 characters.

- The first occurrence of the field name within the response will contain the value.

If an HTML response is given, all data must be presented in a user-readable fashion. For example, if the required machine-readable fields are embedded in comments, another

representation of these fields must be presented to the user. Processing of all HTTP data elements should be case insensitive.

The HTTP Server should be configured as port 80. If port 80 is not available, use one of the five recommended alternate ports: 5713, 6112, 6304, 6874, 7403.

HTTP Response Data Elements

Required Data Elements (listed in the required order)

Data Element Name	Description
time-c	the time of transfer completion at the server. The format will be *yyyymmddhhmmss*.
request-status	a text status indicator by the server. The only defined value at this time is “ok” for a successful transfer. The server should supply a descriptive indication of the error detected following the standards for error codes and messages presented in Table A, “Internet EDM Standard Error Codes and Messages”.
server-id	a *domainname* or *hostname.domainname* uniquely identifying the server associated with the CGI that received and processed the file.
trans-id	a number (integer) up to 15 characters in length uniquely identifying the received transaction file at the server. The trans-id will uniquely identify the file only at the receiving server. A client may receive non-unique trans-ids across multiple servers.

Processing of all HTTP data elements should be case insensitive.

Samples of HTTP Response Required Data Elements:

successful, plain text format:

```
<html>
time-c=19960123203618*
request-status=ok*
server-id=coolhost*
trans-id=232323897*
</html>
```

or

error, plain text format:

```
<html>
time-c=19960619082855*
request-status=EEDM106:  Invalid To Common Code Identifier
server-id=coolhost*
trans-id=234423897*
</html>
```

or

warning, plain text format:

```
<html>
time-c=19960123203618*
request-status=WEDM100:  Transaction Set Sent, Not Mutually Agreed*
server-id=coolhost*
trans-id=532323897*
</html>
```

or, as a more elaborate response to a successful transmittal,

HTML format (this example is for a successful transmittal):

```
<html>
<head>
<title>Upload OK</title>
</head>
<!-- time-c=19960123203618*-->_
<!-- request-status=ok* -->
<!-- server-id=coolhost* -->
<!-- trans-id=232323897*-->
<h1>Upload OK </h1><br>
<body>
<B>File Saved at (time-c): </B>19960123203618<br>
<B>Status (request-status): </B>ok<br>
<B>Server (server-id): </B>coolhost<br>
<B>Transaction ID (trans-id): </B>232323897<br>
</body>
</html>
```

Using a Service Provider for Web Hosting

If you do not wish to install and maintain a Web server, you may wish to contact an Internet Service Provider (ISP) to provide the hosting service for you. Consider the following when selecting an ISP for Web hosting:

- limit on storage space for receiving files
- ability to meet GISB standards for HTTP response
- accommodation for CGI to meet GISB standards for validation and processing

SECURITY

Security Concepts

The security requirements include the current four primary security aspects: data privacy, data integrity, authentication, and non-repudiation.

- Data privacy: unauthorized parties cannot decipher the content of the data.
- Data integrity: unauthorized parties cannot modify or corrupt the data.
- Authentication: the receiver is certain of the identity of the sender.
- Non-repudiation: the sender cannot deny ownership of the transaction if it was sent with his/her digital signature.

In general, these needs are met by using the Basic Authentication capability of the Web server and the encryption and digital signature capability of the PGP security application for securing transactions.

Understanding PGP

Pretty Good Privacy (PGP) is the name of the chosen security application. See the GISB home page for information on software packages to implement the PGP security application. PGP utilizes a public key/private key pair to accomplish secure file transfers. The private key must be known only to the company which generated it. The public key counterpart is shared with trading partners.

Each company must generate its public key and private key pair. The RSA key generation algorithm should be chosen for versions of PGP which offer alternatives. The public keys will be distributed using a secure method (eg., courier mail) to the company's trading partners. You must use the utmost care in protecting your private key. If it is compromised, the security is broken. It is recommended that a key size of 1024 be chosen when generating the key pair. This provides a significantly secure transaction.

When a company wishes to send transactions to its trading partner, it will use the partner's public key to encrypt the file. Encryption provides data privacy. Only the private key counterpart can decrypt this file. Hence, the need to guard your private key.

When the sending party encrypts the file, it also uses its own private key to "sign" the transaction. The receiving party can use the sender's public key to verify the signature. The digital signature provides non-repudiation.

Encryption / Digital Signature

Encryption and signatures are applied to files already translated to a GISB standard data format. (Use of internal encryption such as X12.58 encryption is outside the scope of GISB encryption standards but does not conflict with PGP.).

Encryption and signatures can be accomplished manually for each file using the on-line PGP software, or in an automated (or "batch") fashion using programs to encrypt and sign. Whether encrypting in a manual or automated fashion, it is essential that the correct public key of the trading partner be used to encrypt and just as essential that the correct sender's own private key be used to digitally sign the file.

Decryption / Signature Verification

After a transaction is received and processed by the CGI program, it is ready to be decrypted and have its signature verified. PGP will utilize the appropriate key pair when encrypting, signing, and decrypting if given the correct userID in the key ring identifying the trading partner. Upon request for signature verification, the PGP software will return a human-readable company name.

It is recommended that all implementors create a process where the name is used to look up the ID of the company in a database table. If the ID is passed along with the decrypted file, a process could be created to verify that the company which sent the transaction corresponds to the company identified within the file, once the data has been translated.

Throughput Considerations

Encryption, digital signing, decryption and signature verification are all very CPU intensive. It is not recommended that decryption or signature verification be performed within the CGI that receives and processes the file. In fact, it would not be a good idea to have these steps performed on the same computer that is attempting to receive transactions at a time close to a deadline. Therefore, it is recommended that the secured or to-be-secured transaction be passed to a separate computer for security processing. This "passing" would likely be accomplished by using the File Transfer Protocol (FTP). The security processing computer should be optimized for CPU and memory.

Implementers of Internet EDM sites should review and evaluate Domain Name Server (DNS) cache refresh intervals so as to ensure trading partner address changes are recognized on a timely basis. A refresh interval of 24 hours or less is common.

Because decryption and signature verification are not handled at the time the file is received, the sender will get an HTTP response of successful transfer but doesn't know if the file can be decrypted by the receiver. Guidelines for communicating the status of the decryption step have been developed. See Section "Sending Error Notification Transactions" and Table A, "Internet EDM Standard Error Codes and Messages".

Security Requirements

Basic Authentication

Basic authentication, also known as realm one security, has been defined as one of the security standards for transmission on the Internet. The userid and password will be assigned by the server party according to site standards. The trading party agreement must identify the userid and password for this security as well as procedures for changing the password, if applicable.

PGP File Encryption

File encryption of the EDI file is also selected as a security standard for transmission on the Internet. The encryption software employed is required to be compatible with PGP 2.6 or greater (using keys generated with the RSA algorithm). Those companies who wish to conduct business across the Internet in an unsecure fashion may do so by mutual agreement.

General Security Recommendations

Firewall

A firewall is one or more computers running special software which is designed to provide control of communications between two networks. Its purpose is to limit the types of services between these two networks. Often, a company's connection to the Internet is intended to provide several other services to its employees who are connected by an internal network such as a Local Area Network or Wide Area Network (LAN or WAN). Examples of these services include access to the World Wide Web, use of e-mail, use of file transfer capabilities and publishing content intended for viewing by the external world on a Web server. In addition, the internal network will likely have connections to host computers which provide internal services such as file and print sharing, fax and database capabilities. So that availability of these services and confidential internal data are not compromised by unwelcome intruders from the Internet, there should exist a protective mechanism between the internal network and the public Internet, the firewall.

There are two general mechanisms employed by firewalls to provide this control: packet filtering and proxy services. Packet filtering examines important components of the messages such as the address of the sending and target computers and the designator (port number) for a specific application running on the target computer. By doing this, it can prevent access to specific computers or programs on those computers. It can also reject messages from certain computers. Proxy servers have various capabilities. They can act as relay agents that can examine attempted use of certain features within an application thus limiting access to these features. They can also hide (by substituting its own address) the internal addresses of clients communicating with external hosts. This hiding makes it difficult for potential attackers to focus on specific internal hosts.

Because firewalls are designed to deal with a broad set of security issues, which may vary at each organization, and are not specific to the use of HTTP, this guide does not attempt to provide specific implementation information. Deciding on a specific firewall architecture, organizational security policies, and choosing between numerous products may require outside resources to address these issues.

SENDING ERROR NOTIFICATION TRANSACTIONS

Error Notification

When a client sends a file to a server, the server responds to the receipt of the file. Though the file may be received correctly, some further processing must be done, such as decryption and X12 translation. The decryption step which will have a pass/fail status and then the X12 general translation step which will have a pass/fail status. The X12 general translation is merely the check that the file meets the X12 standards and has not been corrupted. Further translation and processing of specific transactions and elements is outside the Internet EDM scope.

When a file passes the decryption step and passes the general translation step, no notifying communication is sent back to the client. However, if either the decryption step or the general translation step fails, an error notification must be sent to the client.

In general, this standard format for error notification applies to the posting of an error message after sender's session has been disconnected. This error notification has the potential of occurring only after the original HTTP Response is returned with an "ok" or a warning (WEDM999 format) for the request-status value, not an error (EEDM999).

Error Notification Data Elements

The data elements for the error notification are the same as those described in Section "Sending Transactions", with the exception of the "input-format" and "input-data" elements. The file containing the data elements for error notification should not be encrypted.

Required Data Elements for Error Notification (listed in the required order)

Data Element Name	Description
from	Common Code Identifier of sending/client company, the server company which detected the error
to	Common Code Identifier of receiving/server company, the client company which sent the data set in error
input-format	"error"

input-data	A text block containing the following items: orig-from — The "from" value from the original transmission orig-to — The "to" value from the original transmission. orig-input-format — The "input-format" value from the original transmission. resp-time-c — The "time-c" value from the original response. resp-server-id — The "server-id" value from the original response. resp-trans-id — The "trans-id" value from the original response. request-status — The new status of the transaction based on some process beyond CGI such as decryption; see Table A, "Internet EDM Standard Error Codes and Messages". comments — Any comments the original receiving server wishes to include.

Processing of all HTTP data elements should be case insensitive.

Mutually Agreed Upon Data Elements for Error Notification

none defined at this time

Error Notification "input-data" Element Specifications:

The file containing the data elements for error notification should not be encrypted.

All data element names will be in lower case in the Error Notification.

Carriage returns and line feeds will be ignored in all files.

A field delimiter of "*" will be used in the Error Notification. Please refrain from displaying a "*" anywhere else in the error notification so as not to confuse programs that need to parse on this basis.

No spaces should surround the equal sign or the field delimiter.

The required data elements must appear first in the response.

Additional information can be included after the required elements at the server's discretion.

The entire error notification must be no more than 2048 characters.

The first occurrence of the field name within the response will contain the value.

If an HTML response is given, all data must be presented in a user-readable fashion. For example, if the required machine-readable fields are embedded in comments, another representation of these fields must be presented to the user.

Error Notification Example:

```
POST c:\execute HTTP/1.0
Referer: http://www.get.a.life/upl.htm
Connection: Keep-Alive
User-Agent: brow v0.1 XYZ Corp.
Host: localhost
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Content-type: multipart/form-data; boundary=----------87453838942833
Content-Length: 1958

----------87453838942833
Content-Disposition: form-data; name="from"

234567890
----------87453838942833
Content-Disposition: form-data; name="to"

123456789
----------87453838942833
Content-Disposition: form-data; name="input-format"

error
----------87453838942833
Content-Disposition: form-data; name="input-data"; filename="c:\temp\error.not"
Content-Type: application/octet-stream

orig-from=123456789*
orig-to=234567890*
orig-input-format=X12*
resp-time-c=19960619102855*
resp-server-id=coolhost*
resp-trans-id=234423897*
request-status=EEDM601:  Public Key Invalid*
comments=Please contact 1-800-555-1212 for correct public key*
----------87453838942833--
```

Pre-validation before Decryption

Proper trappng of the range of decryption process errors listed in Table A (Internet EDM Standard Error Messages and Codes) may require program code which is external to the decryption algorithm. Some versions of the PGP software do not explicitly discriminate between EEDM601, EEDM602, EEDM603, and EEDM699 type errors. Under such a circumstance, files inbound to the decryption process should be preprocessed to trap the errors not identified by the PGP version being used. For example, searching the file for the text strings "BEGIN PGP MESSAGE" and "END PGP MESSAGE" can quickly identify "EEDM602 File not encrypted" and "EEDM603 Encrypted file truncated" type errors when the implimented PGP version only identifies decryption success, invalid public key (EEDM601), and decryption failure (EEDM699).

CHECKLIST OF TESTING STEPS

Purpose

Preliminary steps in testing are helpful before the full batch browser and server applications are completed. This checklist is intended to provide a series of small achievements leading up to the complete solution.

Client/Browser

NOTE: Throughout all transfer tests, compare files stored on the server against the source file to ensure that the file transferred intact. While transferring to another company's server, you may have to contact that company to send the file back to you so that you can perform the compare.

1. Install an interactive browser. Identify an existing Web server from among GISB compliant servers offering interactive upload for test. See the GISB home page for a list of organizations willing to act as testing partners. These organizations should have a URL complete with the CGI program name to which a tester may send test files. File content does not need to be X12 or other GISB standard format to accomplish this step in testing.

2. Develop or acquire a batch browser that uses multipart for the encoding methodology. Transfer the same test file as in step 1 to the URL not requiring Realm One security.

3. Add Realm One security to your file transfer, and change the URL to the secure URL. Continue transfer tests with your batch browser.

4. Acquire and install PGP software. Generate your public and private key pair. Make sure to choose the RSA key generation algorithm. Download the test server's test public key. Encrypt your data file using this key. Modify your file transfer to send the encrypted file. Continue transfer tests. Request that the test server contact decrypt your file.

HTTP Server and CGI

1. Install Web server. Establish an Internet connection to your server. Ensure that you have ample storage space for transferred files. Ensure that permissions are granted to the directories.

2. As an optional preliminary step, acquire or develop an HTML page for interactive file upload (sample code is earlier in this document). Test interactive file upload to your own server using an interactive browser.

3. Acquire or develop a CGI program to receive file transfers and process according to GISB standards. Test transfers to your CGI using your batch browser.

4. Transfer a X12 or other GISB standard format dataset to your server and process it through your translator or other appropriate processes.

5. Copy the CGI to a "secure" directory where Realm One security, or basic authentication, is enabled. Using your batch browser, transfer to both URLs, with and without authentication. Thoroughly test using the incorrect userid and password against the secure directory.

6. Generate a second public/private key pair. Use the second key to encrypt a file and transfer the file to your server. Decrypt the file.

7. Once your site security is established, contact a trading partner to test transfers against your server.

8. Test with various file sizes to ensure that your CGI can process small and large files.

9. Request that several other trading partners and/or several clients within your own company transfer concurrently to ensure that your server can withstand the load.

10. Test application with various simulated errors in both file transfers and in PGP decryption.

FREQUENTLY ASKED QUESTIONS

As an end user, do I need a continuously connected internet Web server to participate in the Internet EDM in the gas industry, or can I just use a dial-up connection to my ISP and my favorite shrink-wrapped browser software?

An interactive browser connection is not enough to actively participate in the system. It is not necessary to have a private Web server, you can use a service, however the system requires that you have access to a permanent internet connection which is capable of both sending and receiving files (with CGI or BGI) without operator intervention.

If we use ANSI X12.58 encryption do we still need to use PGP encryption?

Both encryption methods are supported and do not conflict with each other. The use of PGP and X12.58 encryption must be specified in the Trading Partner Agreement.

~~**Will pipelines continue to support existing trading systems beyond the normal transition period allowed for implementation of the new internet-based system?**~~

~~Pipelines will continue to support existing systems as long the existing systems are specified in GISB Standards. Existing systems could remain long term as primary backup to the new system. Electronic Commerce, or EDI, is encouraged as an efficient and effective method of conducting business. The Internet EDM is a means of communication that standardizes the transfer of EDI transactions. However, pipelines are under no obligation to discontinue existing proprietary EBB systems and will determine how long to maintain those systems based on customer needs.~~

TABLE A - Internet EDM Standard Error Codes and Messages

These errors and warnings are strictly related to problems found in the recipient CGI or decryption levels of processing before translation. Errors and warnings generated by the client batch browser are assumed to be documented at the client site to distinguish them from problems occurring in the recipient CGI or decryption. Numbering schemes and descriptions should aid in this distinction.

Note: For HTTP error codes see the GISB home page for information sources.

EEDM### standard error format with ### representing a numeric value further processing will not take place

WEDM### standard warning format with ### representing a numeric value further processing will take place

The string for the error or warning should appear in the following format:

Validation Code:Description;supplemental message to be defined by the issuing site up to 80 characters

Internet EDM Standard Error Codes and Messages

Validation Code	Description	Data Element	Required vs. Mutually Agreed
EEDM100	Missing from Common Code Identifier	from	required
EEDM101	Missing to Common Code Identifier	to	required
EEDM102	Missing input format	input-format	required
EEDM103	Missing data file	input-data	required
EEDM104	Missing transaction set	transaction-set	mutually agreed
EEDM105	Invalid from Common Code Identifier	from	required
EEDM106	Invalid to Common Code Identifier	to	required
EEDM107	Invalid input format	input-format	required
EEDM108	Invalid transaction set	transaction-set	mutually agreed
EEDM109	No parameters supplied	parameter string	required
EEDM601	Public key invalid	file itself	required - security
EEDM602	File not encrypted	file itself	required - security
EEDM603	Encrypted file truncated	file itself	required - security
EEDM604	Encrypted file not signed or signature not matched		
EEDM699	Decryption Error		required for general decryption errors not specifically identified by PGP messages or exit codes

Validation Code	Description	Data Element	Required vs. Mutually Agreed
EEDM701	EDM party not associated with EDI party		Check for association between EDM tags and ISA sender and receiver failed. Optional message at receiver's option.
EEDM702	Data structure error		X12 compliance error. Optional message at receiver's option.
EEDM703	Data set exchange not established for Trading Partner		Data set exchange not established for Trading Partner. Optional message at receiver's option.
EEDM980	System error - sender		Optional message.
EEDM981	System error - receiver		Optional message.
EEDM901	System unavailable due to scheduled outage - Transaction rejected		Optional message as a courtsey.
EEDM999	System error		required for general system errors to indicate severe errors in processing at the receiving site
WEDM100	Transaction set sent not mutually agreed	transaction-set	mutually agreed
WEDM901	System unavailable due to scheduled outage - Transaction queud		Optional message as a courtsey.

APPENDIX A - Reference Guide

CGI

An excellent source on CGI is a book entitled “Special Edition Using CGI” by Jeffrey Dwight and Michael Erwin.

Firewall Security

An excellent source which covers this topic in detail is a book entitled "Firewalls and Internet Security: Repelling the Wily Hacker" by William Cheswick and Steven Bellovin.

GISB

GISB Web Site: (http://www.gisb.org) Primary reference for natural gas industry standards

General GISB FTTF Reference Page: (http://www.gisb.org/fttf.htm). This location provides pointers to samples and further documentation.

HTTP

As of this printing (July 1998), there are two versions of HTTP (1.0 and 1.1) that are recognized as standards. The GISB EDM architecture is based on HTTP 1.0, and all implementations should be compatible with this version. All of the HTTP functions required by GISB EDM are expected to be fully compatible with HTTP 1.1 servers and should work without changes.

W3C WorldWide Web Consortium. All aspects of HTTP, HTML, and other Web-related topics:
http://www.w3.org/pub/WWW/

General information regarding HTTP with basic terminology included:
http://www.w3.org/pub/WWW/Protocols/HTTP/1.0/spec.html

Syntax information for multipart can be found in RFC1341 secton 7.2. ~~here: http://unix1.sncc.lsu.edu/internet/guides/www-docs/WWW/Protocols/rfc1341/7_2_Multipart.html~~

HTML

Before April 24, 1998, the recommended standard from the WorldWide Web Consortium was HTML 3.2. The specification for this standard can be found at:
http://www.w3.org/pub/WWW/TR/REC-html32.html

Effective April 24, 1998, the WorldWide Web Consortium has made a recommendation for HTML 4.0. Information on HTML 4.0 may be found at http://www.w3.org/TR/REC-html40/.

http://www.ncsa.uiuc.edu/General/Internet/WWW/HTMLPrimer.html

http://www.interlink-2000.com/guide-to-publishing-html.html

Special Edition Using HTML, Second Edition, Mark Brown, John Jung, and Tom Savola, Que Corporation, 1996.

PGP Software

PGP is available for a variety of operating systems and platforms. For more information contact Network Associates (http://www.nai.com)

~~available for the following operating systems:~~
- ~~Windows~~
- ~~Macintosh~~
- ~~MS-DOS~~
- ~~UNIX (platforms):~~
  - ~~SunOS 4.1.x (SPARC)~~
  - ~~Solaris 2.3, 2.4~~
  - ~~IBM RS/6000 AIX~~
  - ~~HP 9000 Series 700/800 UX~~
  - ~~SCO 386/486 UNIX~~
  - ~~SGI IRIX~~
  - ~~BSD/OS~~
  - ~~DEC Alpha OSF/1~~
  - ~~VAX/VMS~~
  - ~~VMS Alpha~~
  - ~~DG UX AviiON (88/OPEN)~~

Time Synchronization

Testing has shown that the clocks on all computer systems drift. It has also been surprising to see just how much they do. Time synchronization is required to assure that all trading partners transaction times are accurate. Time accuracy is dependent on how much a system's clock drifts, how frequently it is resynchronized and the accuracy of the source used for synchronization.

Authoritative time synchronization is now being provided by governmental agencies around the world based on a synchronized network of atomic clocks. In the United States this includes the U. S. Naval Observatory and the National Institute of Standards and Technology.

A easy way to obtain the current time is from the U. S. Naval Observatory's Web site at http://tycho.usno.navy.mil/cgi-bin/timer.pl. The output from this page can easily be edited and reformatted to set a local system's time. Commercial, shareware and public domain packages

are also available to synchronize system times. Among them are NTP (which is an internet standard), internet daytime, nisttime / usnotime.

Further information on time synchronization may be found at the following Web sites:

http://www.eecis.udel.edu/~mills/ntp/test.html

http://tycho.usno.navy.mil/ntp.html

http://www.ccd.bnl.gov/xntp

http://www.txdirect.net/users/sfisher/clock.html

http://www.is.co.za/resources/ftpsite/tucows/softsync.html

Appendix B - Repudiation and Validation Examples

Repudiation and Validation examples:

When a transaction file is received using the EDM mechanism there are a couple of questions that typically must be answered:

1.) Is the HTTP sender (from) valid to send to the HTTP 'to' party?
2.) Does the HTTP sender match the party who encrypted and signed the file?
3.) Does the HTTP sender match the sender within the file?
4.) Is that sender with the data valid to 'speak' for the parties transacting business?

Is the HTTP sender (from) valid to send to the HTTP 'to' party?

The first validation, determining that a party is a valid sender must be done during CGI execution. This is simply a 'look up' verification that the Common Code Identifier 'from' is recognized as a valid sender.

Does the HTTP sender match the party who encrypted and signed the file?

The next validation, determining that the HTTP sender is the same as the signer, requires that the following information be available:

1.) The 'from' common code identifier (9 digit D-U-N-S® Number). This is the second field in the HTTP post message sent to the CGI. This information must be preserved from that earlier process and passed to the 'post-CGI' process.

2.) The Pretty Good Privacy (PGP) User ID associated with that same party

To compare these items a 'table' would most likely be established that would allow the post-CGI process to identify that there is a correlation between these identifiers. The origin of the 'from' identifier is the HTTP POST 'from' field. The origin of the PGP user ID is the decryption process. The PGP User ID of the signer is a byproduct of file decryption on a signed file. If PGP is executed from the command line the output would be presented in a format like:

Good signature from user "ENRON CORP".
Signature made 1997/05/13 19:30 GMT
Plaintext filename: test3

If PGP is executed using a program interface the User ID that signed the file will be provided in a buffer. Comparing this buffer to the expected User ID would serve to verify this value.

Does the HTTP sender match the sender within the file?

The data file itself indicates (in the case of x12 data) the sender and the intended recipient within the ISA segment. Although this may be the same (D-U-N-S® Number) as the 'from' data these fields are not standardized. This may require the use of a 'table' to relate these identifiers.

Consider also that, although it is strongly recommended that only a single ISA be contained within a file, that the process should account for the possibility of several ISA segments. This comparison will ensure that the parties used during translation are in fact the parties that sent, encrypted and signed the data.

Is that sender with the data valid to 'speak' for the parties transacting business?

This last validation is listed here only to complete the chain of identity. The process that would evaluate this relationship would typically be the business application. Since we have checked the identity through each step of this process this is the point at which the identity of the sender would finally be verified as having a business relationship to conduct the business specified.