# PHINMS
# Acronyms & Glossary List

**Version: 1.0.3**

**Prepared by:**
**U.S. Department of Health & Human Services**

**Date: March 27, 2008**

## VERSION HISTORY

| VERSION # | IMPLEMENTER | DATE | EXPLANATION |
|---|---|---|---|
| 1.0.0 | Wendy Fama | 03-19-07 | Create version 1.0.0. |
| 1.0.0 | Rajeev Seenappa | 06-14-07 | Updates. |
| 1.0.0 | Wendy Fama | 06-14-07 | Updates. |
| 1.0.0 | Wendy Fama | 06-20-07 | Added terms and acronyms. |
| 1.0.1 | Wendy Fama | 07-05-07 | Added terms and acronyms. |
| 1.0.2 | Wendy Fama | 08-08-07 | Added acronyms. |
| 1.0.3 | Wendy Fama | 03-27-08 | Updated with Content Sensitive Help definitions. |
| | | | |

## EXECUTIVE SUMMARY

The Public Health Information Network Messaging System (PHINMS) is a secure and reliable data transport system built to open standards. PHINMS is the primary data transport mechanism for the Centers for Disease Control and Prevention (CDC) Public Health Information Network (PHIN).

This document lists the acronyms and common terms with definitions used by PHINMS.

## TABLE OF CONTENTS

## 1.0 ACROYNMS

| | |
|---|---|
| AH | Authentication Header |
| AJAX | AJAX Asynchronous Javascript and XML |
| AJP | Apache JServ Protocol |
| API | Application Program Interface |
| ATO | Authority to Operate |
| | |
| B2B | Business to Business |
| BA | Basic Authentication |
| BSIO | Business Services Improvement Office |
| BPSS | Business Process Specification Schema |
| | |
| CA | Certificate Authority |
| C&A | Certification and Accreditation |
| CCB | Configuration Change Board |
| CDC | Centers for Disease Control and Prevention |
| CDS | Common Data Store |
| CSC | Customer Service Center |
| CM | Configuration Management |
| CMVP | Cryptographic Module Validation Program |
| COBRA | Common Object Request Broker Architecture |
| COTS | Commercial Off-the-Shelf |
| CPA | Collaboration Protocol Agreement |
| CPP | Collaboration Protocol Profile |
| CPS | Certification Practice Statement |
| CRA | Countermeasure Response Administration |
| CSE | Communications Security Establishment |
| | |
| DMZ | De-Militarized Zone |
| DN | Distinguished Name |
| DNS | Domain Name System |
| DPiT | Data Provisioning Info Technology |
| DSN | Data Source Name |
| | |
| ebMS | ebXML Messaging Services |
| E-SIGN | Electronic Signatures in Global and National Commerce Act |
| ebXML | Electronic Business Extensible Markup Language |
| EJB | Enterprise Java Beans |
| ELR | Electronic Lab Reports |
| ErrorQ | Error Queue |
| ESP | Encapsulating Security Payload |

| | |
|---|---|
| FAQs | Frequently Asked Questions |
| FIPS | Federal Information Processing Standard |
| FTP | File Transfer Protocol |
| GA | General Availability |
| GB | Gigabyte |
| GIS | Geographic Information System |
| GPEA | Government Paperwork Elimination Act |
| GUI | Graphical User Interface |
| HCN | Healthcare Collaborative Network |
| HIPPAA | Health Insurance Portability and Accountability Act |
| HL7 | Health Level Seven |
| HSM | Hardware-Based Security Modules |
| HSQL | HyperXtreme Structured Query Language |
| HSQLDB | HyperXtreme Structured Query Language Database |
| HTML | Hyper Text Markup Language |
| HTTP | Hyper Text Transfer Protocol |
| HTTPS | Hyper Text Transfer Protocol over Secure Sockets Layer (SSL) |
| IBMJCE | International Business Networks Java Cryptography Extension |
| IIS | Internet Information Server |
| IMAP | Internet Message Access Protocol |
| IP | Internet Protocol |
| IPSEC | Internet Protocol Security |
| ISAKMP | Internet Security Association and Key Management Protocol |
| ISAPI | Internet Server Application Programming Interface |
| ITL | Information Technology Laboratory |
| J2EE | Java 2 Platform, Enterprise Edition |
| J2SE | Java 2 Platform Standard Edition |
| JAX | Java API for XML |
| JAX-WS | Java API for XML Web Services |
| JDBC | Java Database Connectivity |
| JDK | Java Development Kit |
| JMS | Java Messaging Service |
| JMSQ | Java Messaging Service Queue |
| JSP | Java Server Pages |
| JVM | Java Virtual Memory |
| KB | Kilobyte |
| LAN | Local Area Network |
| LDAP | Lightweight Directory Access Protocol |
| LDF | Locally Defined Fields |
| LRN | Laboratory Response Network |

| | |
|---|---|
| MB | megabit |
| MMC | Microsoft Management Console |
| MTDC | Mid-Tier Data Center |
| MTOM | Message Transmission Optimization Mechanism |
| NBS | NEDSS Based System |
| NCPHI | National Center for Public Health Information |
| NEDSS | National Electronic Disease Surveillance System |
| NHSN | National Healthcare Safety Network |
| NIST | National Institute for Standards and Technology |
| NND | Nationally Notifiable Disease |
| NNDM | Nationally Notifiable Disease Manager |
| NPP | NEDSS PAMS Platform |
| NVLAP | National Voluntary Laboratory Accreditation Program |
| NYC | New York City |
| NYS | New York State |
| OASIS | Organization for the Advancement of Structured Information Standards |
| ODBC | Open Database Connectivity |
| OID | Object Identifier |
| OMB | Office of Management and Budget |
| OMG | Object Management Group |
| OMS | Outbreak Management System |
| PAMS | Program Area Modules |
| PC | Personal Computer |
| PDW | Preparedness Data Warehouse |
| PEMS | Program Evaluation & Monitoring System |
| PFS | perfect forward secrecy |
| PHIN | Public Health Information Network |
| PHINMS | Public Health Information Network Messaging System |
| PKCS | Public-Key Cryptography Standards |
| PKI | Public Key Infrastructure |
| PM | Project Manager |
| POP | Post Office Protocol |
| PS | Product Support |
| PSK | Pre-Shared Key |
| PTP | Point-to-Point |
| QA | Quality Assurance |
| R&D | Research and Development |
| RAD | Rapid Application Development |
| RC | Release Candidate |
| RDBMS | Relational Database Management System |
| RNR | Route-not-Read |

| | |
|---|---|
| RPC | Remote Procedure Call |
| RT | Real Time |
| SAML | Security Assertion Markup Language |
| SCM | Software Configuration Management |
| SDK | Software Development Kit |
| SDN | Secure Data Network |
| SMTP | Simple Mail Transport Protocol |
| SOA | Service-Oriented Architecture |
| SOAP | Simple Object Access Protocol |
| SP | Service Pack |
| SQL | Structured Query Language |
| SRP | Security Rollup Package |
| SSL | Secure Socket Layer |
| SSO | Single Sign-On |
| StatusQ | Status Queue |
| STP | Secure Transport Protocol |
| TCP | Transmission Control Protocol |
| TLS | Transport Layer Security |
| TransportQ | Transport Queue |
| UAT | User Acceptance Test |
| UDDI | Universal Description, Discovery, and Integration |
| UDP | User Datagram Protocol |
| UID | User Identifier |
| UNIX | Universal Network Information Exchange |
| UML | Unified Modeling Language |
| URI | Uniform Resource Identifier |
| URL | Uniform Resource Locator |
| VPN | Virtual Private Network |
| W3C | World-Wide-Web Consortium |
| WAR | Web Archive |
| WorkerQ | Worker Queue |
| WS | Web Services |
| WSA | Web Service Adapter |
| WSDL | Web Service Description Language |
| WSRP | Web services for Remote Portlets |
| XACML | eXtensible Access Control Markup Language |
| XML | eXtensible Markup Language |
| XMLDSIG | eXtensible Markup Language Digital Signature |
| XMLENC | eXtensible Markup Language Encryption |

## 2.0 COMMON TERMS

| TERM | DEFINITION |
|---|---|
| Acknowledge Folder | The location used to store messages received from the Receiver. |
| Acknowledge Request | The Message Receiver sends the Message Sender an acknowledgement after the Message Receiver receives the message. |
| Acknowledge to File | The status of messages attempted to be sent or successful sent from the Acknowledgement folder. |
| Action | The name of the one (Action) of a two-part combination (Service/Action pair) which establishes the address of the WorkerQ where the message is received. |
| Application Error Code | Indicates the application which received the message experienced some kind of error while trying to process the message. The error codes vary depending on the type of application used. PHINMS does not determine the error codes. |
| Application Level Caching | A process which uses the combination of Message ID, Record ID, and PartyID to determine if a message has been sent. |
| Application Response | The application sends a response to the Message Sender after processing the message. The response contains the application status and/or error codes. |
| Application Status | This field is populated with the status of the PHINMS application. |
| Archive Log | Indicates whether to Archive Log files when the configured size limit has been reached. When the log is archived and a new log is created when the size is reached as indicated in the Max Log Size field. The log is deleted and a new log is created when the size is reached as indicated in the Max Log Size field when the Archive Log is not selected. |
| Arguments | A field not used specifically by PHINMS but a third party application. PHINMS provides this additional metadata field. The third party can provide specific processing instructions to the receiving application. |
| Authentication Type | Identifies what kind of security measures are used to verify identity. |
| Basic Authentication Index Page | The validation of the User's access allowing log in to an index page utilizing the User Name and Password. |
| Basic Authentication Password | The confidential sequence of characters required by the Basic Authentication Index Page. |
| Basic Authentication User Name | The unique name or log on used in conjunction with the password required by the Basic Authentication Index Page. |

| TERM | DEFINITION |
|---|---|
| Cache Entry Max Age | The maximum age in hours entries are cached in the database. The entry is deleted when the age indicated has been reached. |
| Certification URL | The uniform resource locator (URL) of the recipient's public key. |
| Chunk Size | The size of the increments to be sent. |
| Chunking Repository | The directory where received message chunks are stored. |
| Client | Also referred to as a Sender. |
| Connection Timeout | The number of seconds the Message Sender waits before timing out the attempt to connect to the SSL. |
| Console | The PHINMS GUI interface which manipulates the Core Transport. |
| Core Transport | The method PHINMS uses to send and receive messages securely and reliably over the internet. |
| CPA | A Collaboration Protocol Agreement is a necessary agreement between two messaging partners stored in an .xml file. |
| CPA Location | The relative path name to the directory storing the Collaboration Protocol Agreement. |
| Creation Time | The time stamp which records were created in universal time coordinated (UTC) format. |
| Custom Authentication Login Page | The validation of the User's access which allows a user to log into a customized index page utilizing the User Name and Password. |
| Custom Authentication Public Parameters | The relative path name of the Message Sender's Key Store which includes the certificate name and extension. |
| Custom Authentication Secret Parameters | A secured word or string of characters assigned to the Sender's Key Store Private Key. |
| Data Read Timeout | The configured number of seconds the Message Sender waits to receive a response from the Message Receiver before the Message Sender times out. |
| Database Driver | The type of Java Database Connectivity translator between the device and the PHINMS application. |
| Database ID | The unique user defined name of the PHINMS database connection pool. |
| Database Password | The confidential sequence of characters created on the database associated with the User Name to gain access to the database. |
| Database Polling Interval | The number of seconds between polls. |

| TERM | DEFINITION |
|---|---|
| Database Type | The type of database used to store PHINMS messages such as MS Access, MySQL, HSQL, etc. |
| Database URL | The Uniform Resource Locator (URL) of the database connection. |
| Database URL Prefix | The database driver portion of the database's Uniform Resource Locator connection. |
| Database URL Suffix | The server and user name portion of the database's Uniform Resource Locator connection. |
| Database User | The database user's name. |
| Database User | The database's user identification. |
| Delayed Retry | Automatically retries sending failed records at intervals specified by the configured value in the Max Delayed Retries field. |
| Delayed Retry Interval | The number of seconds configured before failed messages are re-queued to be sent out. |
| Delete Records | The amount of records to be deleted determined by the administrator. |
| Delete Start Time | Identifies when the deletion process will begin according to the date and time determined by the administrator. |
| Deletion Interval | The frequency to perform the deletion process determined by the administrator. |
| Destination File Name | The name of the payload file when it is stored on the Receiver. |
| Documentation | Applicable PHINMS documentation used to reference implementation, technical guidance, processes, etc. See also the term for Web Information. |
| Domain Name | An added identifier maintained for future PHINMS compatibility unique to an organization. |
| Dual Sender | The installation of multiple PHINMS instances which can be hosted on the same machine. |
| Duplicate Message Detection | The approach used by the PHINMS Receiver to detect duplicate received messages under the Multi-Transport Queue. This duplicate message detection is beyond the persistent cache's duplicate message detection and is database neutral. |
| ebXML Acknowledge Request | The Message Receiver sends the Message Sender an acknowledgement after the receiving the message. |
| ebXML Signed Acknowledge | The digitally signed acknowledgement which comes from the Message Receiver to the Message Sender. |

| TERM | DEFINITION |
|---|---|
| ebXML Synchronous Reply | Selected when the Message Sender wants to wait for a reply from the Message Receiver for the previous message before attempting to send the next message. |
| Enable Chunking | When this feature is enabled, PHINMS will send a large file in smaller sizes depending on the maximum file size configured. |
| Encryption | The value is Yes if the payload is unintelligible to unauthorized parties. |
| End Point | A final destination for the message. |
| Entry Age | The maximum time in hours when records are cached. |
| Error Code | Used to identify the status of the sent message processed which is either success or failed at the Receiver. |
| Error Message | This field is populated identifying a failed response when trying to process a message |
| ErrorQ | A PHINMS database table which stores error messages of the delivery failure status. |
| File Name | The nomenclature of the outgoing message being sent. |
| Folder Based Polling | Messages which are sent from a folder instead of a database. |
| From PartyID | The unique identifier for each instance of the PHINMS Sender of the message. |
| From User | The Route-not-Read user name used by the Poller to identify where the message originated. |
| High Priority | Checked when the message needs to be sent immediately. |
| In Use | An indication the database is currently running or available. |
| Incoming Directory | A directory where messages are stored if the incoming message is not placed in the database. |
| Install Wizard | The PHINMS install wizard provides upgrades, fresh installs, and different Solaris, Linux, and Windows builds at the operating system level. The application sever level contains JBoss, Web Logic, and Tomcat. |
| Instance | Also referred to as a PHINMS installation or node. |
| Java | A general purpose, high-level, object-oriented, cross-platform programming language. |
| JBoss | Open source Java 2 enterprise application server used for developing and deploying applications. |

| TERM | DEFINITION |
|---|---|
| JDBC Driver | The driver name which the Transports Queue database table uses. |
| JMS Message Handler | The Java Message Service Message Handler is a Java Message Service client application polling (surveying) the Worker Queue at a configurable time interval which sends the payload and metadata information to a Java Message Service Queue. |
| JMS Queue | A Java Message Service Queue is a storage area where messages are stored by the PHINMS Receiver, it contains messages ready to be sent and have been sent with different processing status. As the name Queue suggests, the messages are ready to be sent in order. Processing status of the Queue is changed when the message has been sent. |
| Key Store | The full path name of the Message Sender's Key Store including the certificate name and extension. |
| Key Store Location | The relative path name of the Message Sender's Key Store including the certificate name and extension. |
| Key Store Password | A secured word or string of characters assigned to the Sender's Key Store Private Key. |
| Last Update time | A time stamp indicating the latest modification to the message. |
| LDAP Cache | When selected, the public keys are retrieved from the stored Lightweight Directory Access Protocol (LDAP) searches. |
| LDAP Cache Path | The location of the Lightweight Directory Access Protocol (LDAP) cache. |
| LDAP Cache Timeout | Configured hours to wait before refreshing the Lightweight Directory Access Protocol (LDAP) cache. |
| LDAP Key Retrieval | Configured when the Lightweight Directory Access Protocol (LDAP) search is used to retrieve the public key of the recipient to encrypt the message. |
| LDAP Proxy | Use when PHINMS clients are unable to send Lightweight Directory Access Protocol (LDAP) lookup requests to directory.verisign.com over port 389. |
| LDAP Proxy Action | The ebXML action performed when forwarding the Lightweight Directory Access Protocol (LDAP) request. |
| LDAP Proxy Route | The route used when the proxy server sends the Lightweight Directory Access Protocol (LDAP) request. |
| LDAP Proxy Service | The ebXML service forwarding the Lightweight Directory Access Protocol (LDAP) request. |

| TERM | DEFINITION |
|---|---|
| Location | The area in the directory the acknowledgement, outgoing, and processed folders are created to store for PHINMS messages used with the Folder Based Polling feature. |
| Log Directory | The relative path where PHINMS stores the Sender log files. |
| Log Level | The amount of detail written to a log file. The lowest level of detail is “None” and the highest level of detail is “Messages”. All details are written to the log file including the contents of the message when the “Message” level is selected. |
| Log Location | The directory the log files are stored. |
| Master Password | The password used to decrypt the password file with the PBE utility. |
| Max Cache Entries | The maximum number of entries which can be stored in the persistent cache. The entries are deleted when the maximum number has been cached. |
| Max Cache Size | The maximum number of bytes which can be stored in persistent cache. The entries are deleted when the maximum number of bytes has been cached. |
| Max Chunk Age | The age of the cache before it is cleared. When receiving a chunk from a Sender, this setting is used to determine the hours the chunk should remain on the system before deleting. |
| Max Delayed Retries | The maximum number of times the Message Sender will automatically retry sending failed records. |
| Max Last Update | The frequency to check and process PHINMS messages from the outgoing folder. |
| Max Log Size | An indicator used to determine when the configured size indicated has been reached; the current log will be deleted or archived depending on whether the Archive Log check box is selected. A new log file will be created when the Maximum Log Size has been reached. |
| Max Multi-Block Size | The chunk size configured to send large messages. |
| Max Resend Attempts | The maximum number of times the Message Sender will automatically retry sending failed records. |
| Max Retry Attempts | The maximum number of retry attempts before marking a message permanently failed. |
| Maximum Threads | The total number of Threads which can be connected. |
| Message ID | An assigned unique identifier created by the application for each message. |

| TERM | DEFINITION |
|---|---|
| Message Recipient | A field used for the Sender to identify a receiver of a Message. |
| Message Table Name | The table name of the Transport Queue. |
| Multi-Block | Used to breaks up large messages into chunks before sending. |
| Multi-Block Directory | The location of stored message chunks. |
| Multi-Threading | The number of threads or connections which can run at the same time without interfering with each other. |
| Name | The identity of a regional file which stores processed messages. |
| Node | Also referred to as a PHINMS installation or instance. |
| Origin | The name of the Message Poller. |
| Outgoing Directory | A folder which contains files queued for sending to an application or server. |
| Outgoing Folder | A folder used to store messages to be sent. |
| PartyID | A unique identifier for each instance of the PHINMS Sender used when sending messages to the Centers for Disease Control and Prevention (CDC). The PHINMS Help Desk assigns the PartyID. The PartyID value must be the same as the Message Receiver's PartyID in the Collaboration Protocol Agreement. |
| Password | The challenge phrase created when enrolling for a SDN Digital Certificate. Also used to refer to the login phrase used to open the PHINMS console. |
| Payload File | File name of an outgoing message relative to a local directory such as myinputs.txt. |
| Payload to Disk | Select when the incoming payload needs to be written to disk, if not selected, the payload will be written to the database. |
| Persistent Cache | Detects duplicate messages. |
| PHINMS | CDC's implementation of Secure and Reliable Messaging System |
| PHINMS Software | An application which provides language neutral queue-based interfaces for sending and receiving secure messages. |
| Poll Directory | The relative path where outgoing payload files to be polled are stored. |
| Poll Mode | A select loop and an outgoing directory which is continually polled. |
| Polling Destination | The intermediary server used for Route-not-Read. |
| Polling Interval | The number of seconds between polls. |

| TERM | DEFINITION |
|---|---|
| Pool Size | The number of database connections established by the Receiver. |
| Priority | An integer indicating the request's precedence. |
| Process ID | The unique identification of a series of actions the transportation of the message is performing. |
| Process Status | The current condition of the message at the Sender. The status identifiers are queued, attempted, sent, and done. |
| Processed Folder | A regional file which store successfully sent messages. |
| Protocol | Use either HTTP or HTTPS to send messages. |
| Proxy Host | The host name or the IP address of the proxy server. |
| Proxy Password | The password for the proxy user. |
| Proxy Port | The port number used by the proxy. |
| Proxy User | The name of the proxy user. |
| Public Key LDAP Address | The Lightweight Directory Access Protocol (LDAP) address for the LDAP Directory Server. |
| Public Key LDAP Base DN | The Lightweight Directory Access Protocol (LDAP) Base Distinguished Name of the public key such as an organization. |
| Public Key LDAP DN | The Lightweight Directory Access Protocol (LDAP) Distinguished Name of the public key such as a common name. |
| Queue ID | A unique identifier assigned to each Worker Queue which is associate with the table name. |
| Queue Map | The Queue Map indicates which database and table to store incoming messages. |
| Queue Map ID | The unique ID created by the user referenced by the service map entry. |
| Received Time | The time stamp which records when the message was received. |
| Receiver | A server which is used to send and/or receive messages. |
| Receiver Logs | Stored information on the status of received messages. |
| Recipient | A field identifying the addressee of the destination. |
| Record ID | An accumulating auto field value assigning a unique numeric number to a message. The Record ID is the table's unique key. |
| Resend Delay Interval | The number of seconds the system will wait before failed records are re-queued and sent again. |
| Resend Failed Messages | Activates the option to send unsuccessful sent messages again. |

| TERM | DEFINITION |
|---|---|
| Response Arguments | Used in the Route-not-Read scenario to convey arguments being sent by the Sender to the Receiver. |
| Response File Name | The field is the response file name used in Route-not-Read. |
| Response Local File Name | The response to a poll type request which may contain a payload file. The payload file is written to a local folder under a unique file name used in Route-not-Read. |
| Response Message ID | The Message ID of the received message in the Route-not-Read scenario. |
| Response Message Origin | The PartyID of the user which originally created the message used in a Route-not-Read scenario. |
| Response Message Signature | The PartyID of the user which originally signed the message used in a Route-not-Read scenario. |
| Response to Database | An acknowledgement which indicates an entry was made into the database. |
| Retention Period | The period of time during records are determined to be retained before final disposition or deletion. |
| Route | A path created and stored in the Route Map. The Route includes the To PartyID, Path, Host, Port, Protocol, and Authentication Type assigned to a specific recipient. |
| Route Map | Used to store the recipient's attributes, such as the URL, transport protocol, and authentication type. |
| Route Map Location | The relative path name of the Route Map. |
| Route Name | The name of the route which maps to the CPA files. |
| RouteInfo | A configuration file which maps to the uniform resource locator (URL) of the message Receiver. It contains the address of the Receiver. |
| Route-not-Read | The process used by the Message Sender which an intermediary server receives its messages allowing the Message Sender to later retrieve the messages by the polling server. |
| SDN Challenge Phrase | Formatted security password with special characters created by the user based upon parameters set by the Secure Data Network (SDN). |
| SDN Login Page | The Centers for Disease Control and Prevention's (CDC) secured environment used to request the SND Challenge Phrase. |
| Sender | A server used to send messages. |
| Sender Logs | Information stored on the status of sent messages. |

| TERM | DEFINITION |
|---|---|
| Sent Time | The time stamp which records when the message was sent. |
| Service | The name of the one (Service) of a two-part combination (Service/Action pair) which establishes the address of the WorkerQ where the message is received. |
| Service/Action Pair | Attributes which are based upon the ebXML transport specification. |
| Signature | When Yes is selected, the extensible markup language (ebXML) signature is applied to the payload. |
| Signature Required | The setting which indicates whether a digital signature is required on the incoming payload. The payload will fail when a required digital signature is missing. |
| Signed Acknowledgement | The digitally signed acknowledgement which comes from the Message Receiver to the Message Sender. |
| Signing Certificate Location | The relative path which signed certificates is stored. |
| Synchronous Message Handler | A message handler which processes incoming messages at the same time and responds. |
| Synchronous Message Handler Read Timeout | The amount of time the Receiver should timeout after trying to connect to a Message Handler when doing a synchronous call from a Sender. |
| Synchronous Reply | When the Message Sender waits for a reply from the Message Receiver for the previous message before the Message Sender attempts to send the next message. |
| System Level | System Methodologies and practices which enable quick integration of newly designed components. Additionally, this area will have integration points through out design, development, and deployment. This non-tangible component area will cultivate and insure interoperability and standards compliance for the PHINMS Product Level. |
| Table ID | The PHINMS identification tag within the .xml file in relation to the external database table name. |
| Table Name | The database table name which the Service/Action pair is linked. |
| Text Payload | Selecting this option allows payloads to be received in text format, if not selected, the payloads will be written to the database as binary. |
| Transport Error Code | The error code describing the transport failure. The error codes are Security Failure, Delivery Failure, Not Supported, Unknown, and No Such Service. |

| TERM | DEFINITION |
|---|---|
| Transport Status | The existing condition of the received message either being a transport success, attempted, or transport failure. |
| TransportQ | A relational database table which interfaces between the application creating the message and PHINMS. The database table where the meta-base information and data is stored. |
| Trusted Certificate | Consists of a public key and a private key. Contains Intermediate or Root certificate authority from the trusted source. |
| Trusted Store Location | The relative path of the Message Sender's Trusted Store. |
| Trusted Store Password | A secured word or string of characters for the Trusted Store file. |
| Type | Identifies the approach (Worker Queue, Servlet, or Error Queue) used to route a message. Worker Queues are used to make connections to the database. Servlet are used to interface with third-party applications. Error Queues interface with an error log or error databases to track errors. |
| UDDI | An agreement to operate solutions conforming to a specification for how to build a registry of business services and how to connect to them. |
| URL | Denotes the Domain Name System and the path of the Receiver. |
| Usage | The defined purpose for the database connection. |
| Use Persistent Cache | Indicates whether to use a database to store message information which prevents duplicate messages from being written to the service. The message information is automatically cached in the memory of the Message Receiver when the Use Persistent Cache check box is not selected. |
| Use Web Proxy | Used when the PHINMS client wants to send HTTP requests through a proxy server. |
| User Name | A unique name or logon used in conjunction with the password to gain access to the system. |
| Web Information | PHINMS intranet, internet, online help, and applicable documentation used to reference implementation, technical guidance, processes, etc. |
| Web Services Adapter | Web service components which allow entries into the Transport and Worker Queues via Web Services Clients. |
| WorkerQ | A relational database table which interfaces between the application receiving the message and PHINMS. The database table which stores meta-base information and the payload. |