Electronic Information Environment (EIE)

Electronic Data Exchange Service Interaction Model

EIE Project

document identification	identifier du document
issue date	date de diffusion
8 October 2015	
version	version
1.6	
OPI	BPR
EIE Solution Office	
designator	désignation
EIE Project	
group / division	groupe / division
ADM(IM) / DGEAS	

The information being provided is to illustrate the model that exists for business processes and information exchange within the Performance Based Contracting (PBC) solution for the Department of National Defence. The information is provided to facilitate an understanding of the business architecture and the solution architecture that exist for the PBC program. The content is not intended to reflect the end state specifications for all of the PBC EIE related services.

Table of Contents

The information being provided is to illustrate the model that exists for business processes and information exchange within the Performance Based Contracting (PBC) solution for the Department of National Defence. The information is provided to facilitate an understanding of the business architecture and the solution architecture that exist for the PBC program. The content is not intended to reflect the end state specifications for all of the PBC EIE related services.

List of Tables

List of Figures

1. Introduction

1.1 Objective

The objective of this document is to define the Service Interaction Model for the Electronic Data Exchange (EDE) between Canada Department of National Defence (DND) and Industry partners participating in the In-Service-Support Contracting Framework (ISSCF) or Performance-Based Contracting (PBC) performance based model.

1.2 Scope

Canada DND and each Industry partner will have "front-line" (e.g. directly used by a maintenance technician) and "back-office" (e.g., used in supply chain planning) operational systems supporting ISSCF/PBC. The collection of operational systems support functions such as maintenance management, supply of parts, tools and equipment, configuration management, technical problem management, training, etc., in other words functions essential to the long term maintenance and sustainment of weapon systems.

The scope of the Electronic Data Exchange (EDE) is the collection of web services (and supporting infrastructure) which enable exchange of data between Industry and Canada DND's operational systems. Thus EDE components span application nodes, network zones and the Internet. The integration of backend operational systems which are internal to an Industry partner is not considered in the scope of EDE transaction context regardless of technology in use.

An "***element***" of EDE is an implementation of one or more web services executing on an application node or server. Elements of EDE reside at each Industry partner and at Canada DND. For EDE elements resident at an Industry partner, the Service Operation Model makes no assumptions about the architecture or technology used to implement the operational systems beyond their ability to support standards-based Web Services.

This document does not identify business services per se or define any business service. Existing services will be mentioned as examples.

1.3 Assumptions

The reader should be familiar with

- The objectives of the In-Service-Support Contracting Framework and its requirements for Electronic Information Exchange.
- Basic concepts of Service Oriented Architecture (SOA) and Web Services.
- Basic concepts of the subject matter, for example, purchase orders, work orders, etc.
- Some of the services identified in the EIE Business Process Catalogue.

1.4 Outline

This document contains:

- Overview of the Service Interaction Model and its main elements
- Web Service Standards which are essential to enable EDE services across a multi-Industry SOA ecosystem
- Definition of the Message Exchange Patterns which may be used in EIE
- Definition of the Security Operational Model governing message exchange
- Definition of specific message contents in all relevant exchange patterns
- Identification of the EDE Application Domain and the Message Exchange Pattern used in each domain
- Finally, a comprehensive set of references on SOA and Web Services is included.

2. Service Interaction Model

2.1 Business Processes and Web Services

ISSCF/PBC stipulates business processes for Canada DND and Industry with very specific integration points where business objects (e.g., Notifications, Purchase Orders) must be exchanged between Industry and Canada DND. EDE web services enable the exchange of information defined by the business processes based on relevant integration activities.

Most activities performed during business process involve updates to a Canada DND or Industry operational system. EDE defines a catalogue of services for integration of Canada DND and Industry activities as business processes are executed. The business objects exchanged through EDE are necessary for both Canada DND and Industry's operational systems to have the required information to support the various domains that make up ISSCF/PBC model and to fulfill their respective mandate of the ISSCF/PBC program.

Depending on the services being used an element of EDE (implementing some services) may be in the role of *Service Consumer* or *Service Provider* at different points in time.

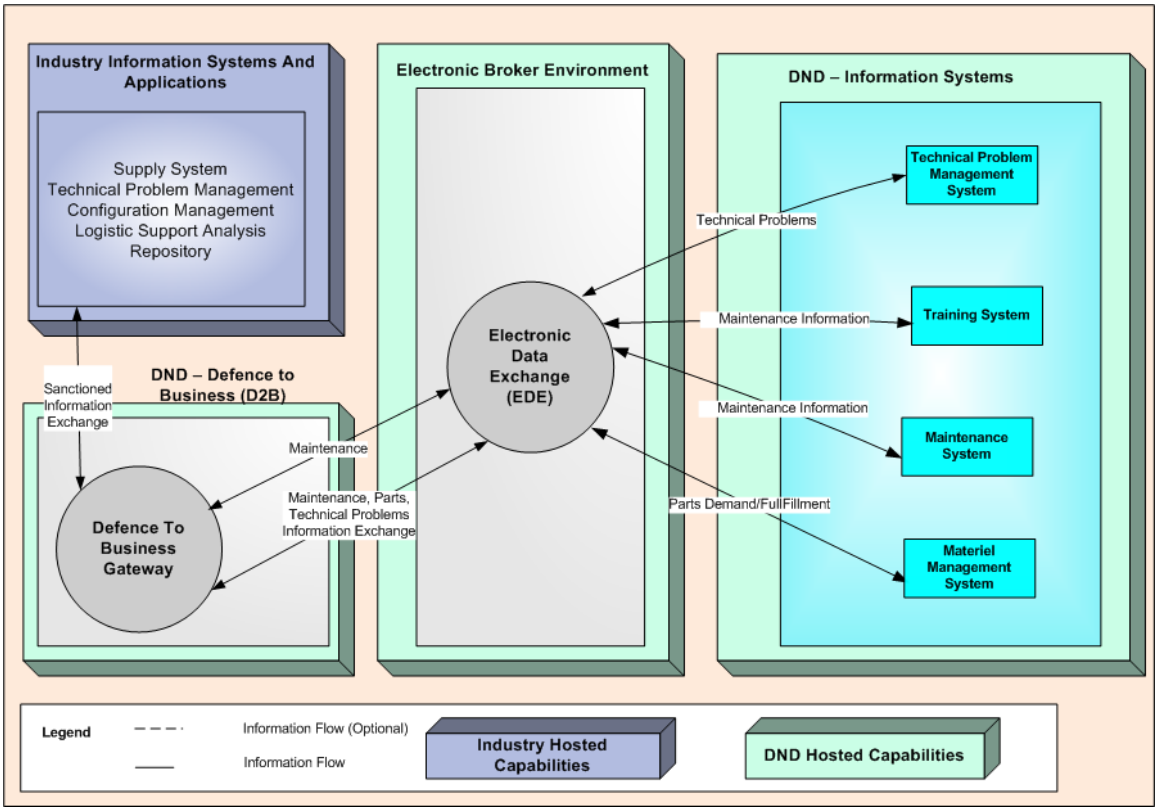


Figure 2-1 Electronic Information Environment (EIE) - System Context

2.2 Service Oriented Architectural Principles

EDE architecture is based on Service Oriented Architecture (SOA) principles. Each invocation of an operation of a service delivers a message between EDE Elements. Each message contains header information, security information, and one or more business objects (e.g., one purchase order, a list of Equipment Master Record (EMR) measurements).

Each service is *atomic* at a technical level in that the message is successfully delivered or not and the successful delivery of one message does not depend on other messages (exceptions are noted later under the discussion of "Units of Work"). However, in the context of the business process it may be that the success or failure of a message affects use of other services. For example, if a Part Issue message fails there will not be a subsequent Part Receipt.

Each service is *idempotent* at a technical level in that message IDs are defined and tested for uniqueness. In the context of the business process, if duplicate business objects are sent through EIE Services – with distinct message IDs – then it is the responsibility of the operational systems to detect the duplication.

The following sections will also address SOA aspects including:

- Security;
- Quality of service requirements;
- Interaction context, including transaction and message correlation information;
- Failure conditions affecting message delivery and message processing.

2.3 Message Structure (Overview)

The key to message exchanges across system boundaries is to ensure that there is sufficient information packed into the respective segments of the message namely the Header and Body segments. Regardless of the technology in use, providing the right content/information can facilitate efficient message processing with minimal computing resources being employed by all of the nodes that participate in the processing of a message.

Every Message has a unique Message ID. The uniqueness is relative to the Service Consumer (the element which invokes the service and created the message).

2.4 Message Exchange Patterns

Two common SOA Message Exchange Patterns are defined for EDE. (Further details are provided below in Section 5).

2.4.1 Request Response Exchanges

In Request-response pattern, a Service Consumer creates a request message and sends it to a Service Provider. The Service Provider receives a request message and sends back a correlated response message. The response contains technical message processing detail level information in the body of the message; also known as "Synchronous".

In EDE the request-response pattern is used extensively, with an important caveat. The request message contains one or more business objects (e.g. a purchase order). The correlated response message pertains solely to the success or failure of initial processing of the request message and confirmation (or, in the case of failure, denial) that the Service Provider accepts custody of the message and responsibility to deliver business objects to a target operational system. The correlated response message never contains business objects (e.g. delivery information in response to a purchase order).

2.4.2 One-Way Exchanges

In One-way pattern Service Consumer creates a request message and sends it to the Service Provider. The Service Provider does not send internal processing details in a response stream; also known as "Asynchronous".

Currently EIE does not use the One-Way pattern between Canada DND and Industry partners.

2.5 "A-Periodic", "Periodic" and "Near Real Time"

There will be primarily three patterns of information exchange that will occur between Canada DND and Industry partners - thus by extension with the EDE and Canada DND application systems. These are categorized based on the frequency of occurrence and fulfillment period. They are namely:

A-periodic Information Exchange defined as messages that will be governed and managed as per operational requirements.

Periodic Information Exchange defined as messages that are published and distributed based on a predefined frequency of occurrence from frequency that can range from as low as, on an hourly to daily, weekly, monthly, quarterly or annually.

Near real-time Information Exchange defined as messages that are transmitted as the event or request occurs and fulfillment or a response is expected within a narrowly defined transactional context in terms of a response time.

The specific business domain and the associated periodicity will be defined in the business domain specific operational model.

2.6 Unit of Work

A certain class of EDE transactions need to occur with a degree of association across multiple independent services. These types of transactions participate in a unit of work construct that needs to be managed in its entirety from both a provider and consumer perspective.

2.6.1 Individual Messages

Each service within EDE can support the processing of a message with a single business object type. In this case the unit of work begins and end within the single invocation of the service and the message being delivered.

2.6.2 Group of Messages with Same Business Object Type

The services within EDE inherently can support receiving 1...N of individual messages of the same business object type either within multiple invocations of the service to deliver the same business object type. In this case the unit of work can span multiple invocations of the same service delivering one or more of the same business object type, which implies that the unit of work will fail or succeed in its entirety based on the segmentation strategy that was adopted by the data provider.

In the case of a unit of work spanning multiple service invocations, the unit of work shall have a unique identifier. To manage the unit of work each message within the unit will include the unit of work identifier to establish the context of where in the chain of messages does this delivery fall into.

2.6.3 Group of Messages with Multiple Business Object Types

There are instances of the business events in ISSCF/PBC which require a related set of business objects to be delivered and will be required to ensure that the multiple related business object types are handled within a unit of work that factors the integrity required across the multiple business object types.

The unit of work now spans across service boundaries. In order to support the cross over boundaries across service the Service Consumer will provide an explicit governance message that will contain the following:

- The Business Object types making up the unit of work;
- A record count of each business object type;
- A unit of work unique identifier.

For each service invocation within the unit of work the Service Consumer must include the unit of work unique identifier.

The Service Consumer – as the provider of business objects – will need to ensure that all of the declared number of business objects has been successfully delivered. The Service Provider – as

consumer of the business objects - will need to ensure that all of the business objects within the unit of work are managed as singular unit and delivered to the backend systems for processing.

3. Open Standards for Services within EDE

This section defines the open standards which apply to all EDE Services. This provides a common technical structure to ensure inter-operability. In order to be considered as an EDE Compliant Web service, a Service Provider and Service Consumer must comply with the criteria in all of the following areas.

We will use the standards described below to ensure common technical structure for the messages being exchanged that comply and remains the confines of the web services specifications listed below. EDE specific aspects of information will be injected into the respective segments of the message that will be specified in sections below.

Section Five provides the concrete design standards which will apply to the design of EDE services.

3.1 Security

Refer to Section 4 below for security model

3.2 XML and XML Schema

1) Must use XML 1.0 as defined by Extensible Mark-up Language (XML) Schema Definition (XSD) 1.0 (Part 1: Structures, Part 2: Data types, and Part 0: Primer) to formally and fully describe the structure and type of the XML elements used in Web Services-related XML documents. Encapsulating XML documents in a string of the payload does not meet the "fully described" criteria, and does not provide the necessary "self-descriptive" capabilities that facilitate the use of services. Interfaces that use Simple Object Access Protocol (SOAP) to transport XML documents based on a large variety of XML schemas, typically in content management and events, are exempted from this compliance criterion.
2) Must have the Target Namespace defined. Thus, XML Schemas can be used in a single XML document, or imported by a single XML schema file.
3) Must use the target Namespace element in Web Services Description Language (WSDL) and XML Schema for versioning of nontrivial changes.

3.3 Web Services Description Language (WSDL) and SOAP

1) Web services deployed within EIE must be described using WSDL 1.1.
2) WSDL binding definitions must be limited to SOAP bindings using literal encoding with Document style.
3) Must be SOAP 1.1 Document style with literal encoding.

4) SOAP Messages with Attachments for SOAP Messages must be used to carry binary objects (e.g., JPEG, MPEG, MP3, etc.) and complied with "WS-I Attachments Profile Version 1.0".

3.4 SOAP Binding and Transport

1) SOAP Binding Style: Document Style (payload in the body of the SOAP message).
2) Encode Style: Literal (Using an XML schema to provide a description and constraint for the XML content).
3) SOAP with Attachments (MTOM/XOP) may be used.
4) For external interfaces, HTTP and HTTPS 1.1 are the approved transport protocols to maximize interoperability with external Service Requesters.
5) Optionally JMS 1.1 is also approved transport protocols for external partners.
6) The value of SOAP Action attribute in HTTP(S) header field must be either absent or its value is set as a quoted empty string. The SOAP Action header is purely a hint to processors. All vital information regarding the intent of a message is carried in the Envelope.

3.5 Quality of Service

Quality of Service is established with the following required standards:

1) WS-Security 1.1

Quality of Service may be supplemented with the following standards:

1) WS- RM 1.1
2) JMS 1.1
3) WS-Conversation
4) WS-Trust
5) WS-Federation.

3.6 Web Services Interoperability (WS-I)

Web Services Interoperability (WS-I) implementation guidelines: "WS-I Basic Profile Version 1.1" of WS-I provides detailed clarifications and amendments of key Web Services specifications to promote interoperability. Its recommendations must be followed unless they conflict with any other standard adopted for use within the architecture.

3.7 Summary of SOA Standards

Summary of EIE Web Service and SOA Standards is provided in Table 3-1.

Table 3-1 Adopted Open Standards for Services

Subject	Required Standard	Optional Standard
Transport	HTTP/S 1.1 JMS 1.1	
Message Format and Protocols	SOAP 1.1/1.2 MTOM/XOP	SOAP with Attachments
Identity and Location	WS-Addressing 1.0	
Quality Of Service	WS-Security 1.1	WS- RM 1.1 JMS 1.1 WS-Conversation WS-Trust WS-Federation
Service Definition	WSDL 1.1 XML 1.0	
WS – Interoperability (WS-I)	Basic Profile 1.1	

4. Security

EDE is built on a multi-layer security model that is distributed across multiple network zones and interacts with both DND and Industry point of presence, thus a tiered approach to security has been undertaken that is built on the models adopted.

In following the model prescribed below EDE security framework supports the entire required element of security using an open standards based support.

4.1 Security Standards

1) Use only HTTPS (SSL 3.0) for exchanging security sensitive messages over SOAP between DND and external trading partners
2) Use of the following mechanisms is required to meet DND security standards (e.g., when Web services requests and responses traverse between DND's Defence Wide Area Network (DWAN) and GPNET network security Zones), and to satisfy Web Service Provider or Requester authentication needs:
    - HTTPS (SSL 3.0) with the "mutual authentication".
    - WS-Security X.509 Certificate Token Profile is allowed if it conforms to "WS-I Basic Security Profile Version 1.1" to authenticate the Service Requester.
    - JMS/MQ over Secure Socket Layer (SSL).

A combination of the security standards identified here must be used for all security needs.

4.2 Transport Layer Security

The transport layer security will always be enabled using SSL with the authentication model being set to "Mutual". We will use X.509 certificates to enable the authentication model. Certificates that are used for transport will only be used in that context.

Canada DND will provide its own self-signed certificates and will be issued by DND Certificate authority. Industry should provide a verifiable certificate that is signed by a known and authorized Certificate authority.

Certificates expiry notification windows will have to be established and agreed to by both industry and DND in advance of transitioning into operations and during the in-service-support phase for renewals of the same.

4.3 Security Across Domains

Each technical environment (Development, Integration, Test, Partner, Quality Assurance (QA) and Prod) that are exposed to Industry by Canada will have its own distinct certificates issued by

DND to industry. Industry will have to comply with this requirement since we want to ensure that only authorized environments are allowed to access DND's equivalent environments.

Industry will access the exposed environment based on the DND published Uniform Resource Locators (URLs) that are discoverable via the internet for the various environments; for example, https://dev.eie.isscf.forces.gc.ca/. The specific URLs for the environment will be provided on commencement of development and testing activities with an industry partner.

Note: DND have two separate networks that will be used for interaction with Industry they are as follows:

- Test and Development Centre (TDC);
- Production Network – Defence Wide Area Network (DWAN).

Canada DND will require information to register specific access points originating from industry that will access the Canada DND environments.

4.4 WS-Security

The use of WS-Security in the context of EDE message exchange is to support the following:

- An authorization model to ensure that the service requestor is authorized to access the specific service.
- Since we will be using X.509 certificates the message can be signed using the private key. By signing of the message to assert identity of the requestor and using the Binary security token to transmit the public key of the certificate in the message as per the WS-Security schema standard we can authenticate the system providing the message.
- Using the signature we can ensure that the message digest that is generated and included in WS-security can be used to validate that the message that was sent and the one that was received has not been tampered with during transit.

Canada will provide Public Key Infrastructure (PKI) certificates for use with WS-Security to each industry partner. Canada will sign the messages using the private key of the certificate that was issued to the specific industry partner.

The certificates will be assigned to a specific Industry partner, business domain and technical environment. A certificate applies to a single combination. For example a certificate issued to Industry "A" for the supply domain and the QA environment cannot be used by any other Industry or by "A" for maintenance history, or by "A" for supply services in production.

Canada requires that the industry partner will always use distinct certificates for Transport layer and WS-Security layer. It is up to the industry partner's discretion if they want to further segregate WS-Security certificates by business domains.

5. Defining Web Services

The model applies to service provider-to-service consumer communication for applications and components deployed on any EDE infrastructure subject to the EDE Standards defined in Section 3.

The EDE will support two patterns for message communication as described below:

5.1 DND Internal SOA Repository

This section is included for future reference.

DND Internal SOA Repository will be used to register all Web services that are exposed to industry partners.

Web service artefacts must be attached to the service entry of the DND Internal SOA Repository.

5.2 Message Definition

All EDE Messages will conform to the following design standards:

1) A Metadata portion will be present that will contain a subset of information that uniquely identifies the associated payload of the schema. Any other relevant elements that are required to support transformation and routing needs will also be included as required, and will be identified as per the exchange type.
2) The payload content nomenclature will always be represented using business terms and will be agnostic of the system supporting the implementation of the same. In other words, it should be a canonical schema that is built from the logical representation of the information domain from which it is sourced.
3) XML Namespaces will be used to support consistency of definitions and providing for flexibility and reuse to build new schemas using import mechanisms as opposed to redefining the element each time.
4) XML namespace will also be used for element that are dependent on open standards and will be defined using the same open standards specification.
5) XML namespace for DND domain specific standard will take the following form.

```
//URL//PrimaryEnablingFunctionalUnit/Version/BusinessDomain/SubBusinessDomain.
```

For example:

```
//eie.isscf.forces.gc.ca/EDE-Canada/2010/08/20/Maintenance/EMR/service
```

5.3 WSDL Definition

All EDE Services will conform to the following design standards using six major elements as follows:

1) Types, which provides data type definitions used to describe the messages being exchanged.
2) Message, which represents an abstract definition of the business data being transmitted.
3) Port Type, which is a set of business services functions/operations. Each operation refers to an input message and output messages.
4) Binding, this specifies concrete protocol and data format specifications for the operations and messages defined by a particular portType.
5) Port, which specifies an address for a binding, thus defining a single communication endpoint.
6) Service, which is used to aggregate a set of related ports.

Services which use multiple operations will use the soapAction field to uniquely identify each operation.

5.4 Message Processing for Message Exchange Patterns

5.4.1 One-way Exchange Pattern

In the One-way pattern, the Service Provider receives a message but does not send internal processing details in a response stream. When using HTTP/HTTPs protocol, the HTTP response stream does not contain any processing level information beyond those required by the transport protocol. This interaction model is meant for delivery of messages where technical processing of the message payload can be conducted with a purely asynchronous response. However, technical processing of the message should occur within the boundaries of the security aspect of the message which will be defined using WS-Security constraints.

As per WSDL Specification, the grammar of one-way operation contains only an **input** element which specifies the message format.

Further processing by the Service Provider does not impact the One-way message.

5.4.2 Request-response

In the Request-response pattern, the endpoint receives a message and sends back a correlated response. The response contains technical message processing detail level information in the body of the message.

The transport layer dictates how the response is transmitted. When a message is transmitted over HTTP/HTTPs transport protocol without WS-Addressing, the processing level response (SOAP output or fault) is sent over HTTP response stream in the same HTTP session, in which, the endpoint received the request.

However when using WS-Addressing over HTTP/HTTPs, the processing level response is sent in a separate message to the endpoints as declared in the WS-Addressing construct. Similarly, when using JMS transport, the endpoint that will be defined for the response will be to the reply Queue as a new message with the correlation association to the original request.

As per WSDL Specification, the grammar of the request-response operation contains **input**, **output** and the **fault** elements. The input and output element specifies the message format of request and response, respectively. The optional fault elements specify the message format for any error that may be output as a result of the operation, beyond those specific to the protocol.

5.4.3 Phases of Processing for Request-Response Exchange Pattern

5.4.3.1 Technical Delivery Phase

5.4.3.1.1 Phase 1 – Type 1 Failures/Errors

In EDE, the processing in request-response operations prior to sending a response is limited to taking ownership of a "valid and technically compliant" message. The Service Provider is expected to perform the following validation steps prior to sending the response:

1) Security functions such as Authentication, Authorization, and Audit etc.;
2) XML Schema level compliance;
3) Message level validation such as duplicate message validation, expiry time based validation;
4) Validation of the message header contents;
5) Optional persisting of the message to guarantee recovery in case of system level failures.

If the request message successfully passes the above validations, a positive response as defined in the output element is sent back to the Service Consumer. The Service Consumer now has the received the assurance that the message has been delivered successfully and accepted by the Service Provider for the further processing.

In case of failure, a negative response as defined in the fault element is sent back. It is important to note that some of the above mentioned functions can be executed at the protocol/transport level in which case, the negative response can be protocol specific. For example, authentication can be implemented as Mutual SSL function, and in case of an authentication failure, there is no SOAP fault response since SSL connection will not have been established.

The error conditions and the associated handling fall into a category of Type 1 errors and will be referenced as such with service specification documents.

5.4.3.1.2 Phase 2 - Type 2 Failures/Errors

There is a second phase of technical processing that will occur for messages accepted by EDE which are then dispatched to Canada's business systems to process.

During the technical phase of the processing by the business systems errors can be observed on the data that was received. These types of errors are categorized as 'Type 2" errors. These errors will be reported back to the data provider using a distinct error reporting mechanism that will be inherent for every service that is contained within ISSCF/PBC catalogue of services. In other words every service will expose an error reporting operation that can be invoked at any time after a technical delivery of a message is complete.

These errors will be reported using the structure as described in each service specification where error operation is defined.

The error reporting service will be hosted by the service provider who is initiating the information exchange. The service consumer of the information will invoke the error reporting based on the conditions described above.

5.4.3.2 Business Processing Phase

In the business process phase within certain business domain the natural lifecycle of business events ensures that the business processing phase occurs and both the consumer and provider know the roles and responsibility to fulfill the required business events.

However in other business domains namely Maintenance History and Master Data and Engineering change the business processing phase could occur after some lapse time after the technical delivery phase has been completed. It is now important for both the provider and the consumer to know if the business processing phase has been executed.

The results of the execution whether successful or not should be communicated using the exposed service operations to the provider of the information within the constraints of the host systems. The degree to which business processing can occur will need to be bounded and defined and understood by both the service provider and consumer. In order to ensure the results that are shared can be interpreted in a meaningful manner and is constrained based on the prior knowledge of the information that was shared.

In order to support this information exchange the data provider who generated the data and initiated the transfer will expose an "Acknowledgement" operation that can be invoked by the service consumer who received the information.

Note: The choice of exposing an acknowledgement service will be determined by each provider's acknowledgement needs for the data that was transferred.

5.4.4 Retry Processing Phase

In a message exchange scenario the initiating party will transmit a message to a designated end-point and based on the pattern of "Request/Response" pattern expects an "acknowledgement"

response. But in some instances the end-point would not respond with the usual technical acknowledgement, in these circumstances the following protocol would be followed:

- The initiating party would wait for the Acknowledgment Time Interval (ACK_TIME_INTERVAL) to expire;
- Initiate subsequent message transfer with all of the information remaining exactly the same, this process will be repeated as per the retry time interval parameter (RETRY_TIME_INTERVAL) and based on the number of retry attempts parameter (NUMBER OF RETRIES) as defined in the Service Specifications for each service/ operation.

5.4.5 Dead Message Phase

In instances where all attempts to deliver the message and receive a technical acknowledgment has failed. The message will be categorized as a "Dead Message" and the flagged as such by the service consumer within their system boundaries.

The dead messages will then be collected on a periodic basis and a subset of metadata from each message will be constructed into "Dead Message" information exchange and dispatched via the "Dead Message" service.

The dead message will be sent to service provider to whom the service consumer could not successfully deliver the messages prior to being categorized as "Dead message".

The responsibility lies with both the service consumer and provider to notify each other that the message have not been successfully exchanged.

Note: Dead Message Service is itself a web services. Thus if the entire web services infrastructure is non-responsive then alternate channels of delivery will need to be used to communicate the messages that went dead.

A standalone service specification for "Dead Message" is available.

5.4.6 Dead Message Resolution Phase

Here is the sequence under which Dead message processing should occur:

1) SC identifies unfulfilled requests (Message State of DEAD).
2) SC notifies SP
    a. Grouped by Exchange Type Unfulfilled
    b. Identifies the Context (aka message state)
    c. Error details received from SP system when attempts were made to connect (such as Connection refused/Connection time-out. [This will help the SP to establish the probable cause of the error])
    d. Violation against established SLA.

3) SC formats the DEAD Message content as per DEAD message service definition.
4) SC uses a dedicated channel similar to other Services model (Web Service based).
5) SC invokes SP exposed DEAD Message and delivers the message and receives an ACK from the SP within the same session.
6) SC will mark within their Dead Message Queue that all of the dead message information has been sent to SP for action.

SP Providers Internal SOPs will be followed:

1) SP will determine the course of action based on the nature of the error details as provided by SC in the Dead Message report.
2) Based on the type of transaction and as identified within the Service Specification the appropriate e-processing model will apply.
3) SP will work internal to their systems to establish processing states for message records that were received via the dead message service.

The specific handling of messages that have been deemed dead by both the message's original sender and its recipient (a.k.a. **discordant** messages) will be determined as per the following criteria:

- Business Domain the message belonged to
- Business event that generated the message
- Business event that consumes the message
- The state of business event after generating the message
- The state of the business event that consumed the generated message.

All of these criteria considered holistically will determine the process model for handling such messages.

Broadly speaking, Supply oriented business events and associated messages occur in near real-time, so the ability to rectify discordant message is more immediate and will be dealt with the participation of both the service provider and service consumer organization operational staff.

In the case of Maintenance history, since the data is released after the business process has occurred; the window to rectify are typically longer and lends itself to tracing and resolving the discordance which may result in either the consumer or provider agreeing to manually rectify the discordance internal to their respective system boundaries.

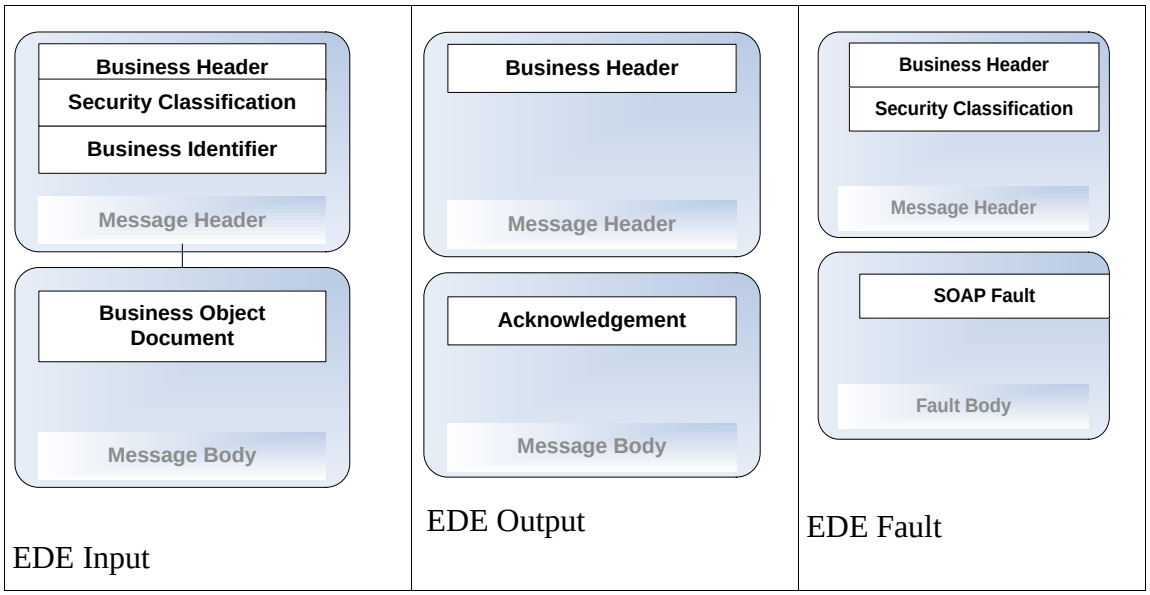
In the case of Master Data, a model similar to Maintenance history may be undertaken in that the discordant data can be traced prior to determining the rectification mechanism.

6. EDE Message Definitions

6.1 Overview

A message in the context of EDE SOAP Web Services is defined as:

- A SOAP XML document with
  - Soap Envelope, SOAP Header and SOAP Body.
- The EIE specific message content includes:
  - Message Header – A Business Header used and Security Markings;
  - Message Body – One or more Business Objects of the same type.



It is important to note that SOAP output and the SOAP Fault is communicated only in request-response model. In both request-response and one-way models, the SOAP input structure is the same.

6.2 Message Header

Message header contains elements which are processed at the mediation layer to facilitate the following:

- Establishing and validating the identity and the context of the message;

- Establishing and validating the security classification of the message;
- Establishing the identity of the business objects in the payload of the message for audit purposes; and
- Quick and efficient routing decisions through the mediation layer.

The preference is to place the message header in the SOAP header section of the message. In cases where it cannot be achieved due to technical limitations, the message header can be placed in the SOAP body of the message. The next sections describe each element in the message header in detail.

6.2.1 Business Header

Business Header establishes the identity, context and the time stamps of a message. Fields contained in Business Header are provided in Table 6-1 below. Known values for certain attributes are described in the following paragraphs.

Table 6-1 Fields in Business Header

Attribute	Description	Service Consumer	Service Provider
Exchange Type	An enumeration of the various exchange types (a.k.a. business messages) that will be transacted between a specific service provider and a consumer. The enumeration will jointly be defined based on a mutually defined set of exchange types.	Defined in request	Replayed In response
Industry Identifier	Industry Partner in an ISSCF/PBC relationship with Canada DND as identified within Canada Maintenance Management System An enumerated list identifying the Industry Identifier defined during ICD development.	Populated as per information available within consumer's information systems.	Preserved and relayed back in a response

Attribute	Description	Service Consumer	Service Provider
Fleet Identifier	A unique Identifier of a weapon system as defined within Canada Maintenance and Management System. This is mainly relevant when the same Industry partner is in an ISSCF/PBC relationship with Canada DND for more than one fleet/weapon system. An enumerated list identifying the Fleet defined during ICD development.	Populated as per information available within consumer's information systems.	Preserved and relayed back in a response
Message Id	Unique identifier that is generated by the message creator - Service Consumer for a request message Within EDE, a Message ID will not be re-used – except for message re-tries after a recoverable error. A message retry will use the same message ID as the prior attempt(s).	Defined during creation of a message instance	Unique identifier that is generated by the message creator - Service Consumer for a SOAP output response message or a SOAP fault message
Generation Timestamp	A timestamp that is generated by the message creator - Service Consumer for a request message, Service Provider for a response message. The timestamp is indicative of when the message was created within their respective system boundaries[1]. The timestamp will not change on during a retry cycle.	Defined during Initiation of a message instance	Defined during Initiation of message response

[1] Comparing request and response timestamps must take into account possible clock drift between Service Consumer and Service Provider.

Attribute	Description	Service Consumer	Service Provider
Unit of Work Identifier	Unique Identifier that is generated by the Service Consumer which is the creator of the unit of work.	Defined during creation of a unit of work. Passed as input to subsequent services in the same unit of work.	
Correlation Id	A Service Provider will populate this attribute with the Message ID of the message to which they are correlating (either technical or business response (as applicable)) **(See Below)**		Populated with the value from the from request message header's - Message Id attribute. When responding to a message in SOAP output message or a SOAP fault message

6.2.1.1 Rules for Service Consumer (SC) Responsibilities to Populate

1) Populate the attributes listed above where the source of the information is the SC.
2) Adhere to the mandatory nature of the related attributes, which implies that it will have been populated as per the specification rules of the message being transacted.

6.2.1.2 Rules for Service Provider (SP) Responsibilities to Populate

1) Populate the attributes listed above for which the SP is the only source.
2) Adhere to the mandatory nature of the related attributes, which implies that it will have to be resent with a response message or a related business transaction as required and thus complying with the specification rules of the messages being transacted.

6.2.1.3 Example Values for Header Attributes

The following table gives a sample of values for some of the attributes used in the Business Header. Fleet-specific values will be determined during implementation of the ICD (Interface Control Documents).

Exchange Type	Notification and its associated sub messages Work Order and its associated sub messages Equipment Master Record

	Part Demand Part Demand Response Part Issue Part Receipt Part Return Part Return Receipt Note: Specific values will be be provided by Canada prior to implementation of the interfaces by industry partner.
Industry Identifier	0001 - LMA 0002 - BOEING Note: These are sample values. Specific values will be provided by Canada prior to implementation of the interfaces by industry partner.
Fleet Identifier	0001 - C130J 0002 - CH147 Note: These are sample values. Specific values will be provided by Canada prior to implementation of the interfaces by industry partner

6.2.1.4 Correlation ID

A service provider will populate this attribute with a Message ID against which they are providing a business response. Population of correlation ID is conditional, based upon the domain specific message operating model. Correlation ID will be populated in the following situations:

- Within the initial response message to an originating message request. For example, a Part Demand technicalacknowledgement message will populate the correlation ID with the message ID from the originating Part Demand message.
- Within a Technical Error message for an EDE generated technical error generating from an original request message. For example, a Part Demand message may generate a technical error if undeliverable to the Industry partner. The Technical Error message will populate the correlation ID with the message ID from the originating Part Demand message.
- Correlation Id will be populated in a response message for request-response message regardless of the protocol in use (HTTP or JMS).

Correlation ID MAY or MAY NOT be populated in the following conditions (depending upon partner system ability to provide the correlation ID):

- Within the subsequent business response message (after the initial response) to an originating business request. For example, EDD may be updated in a subsequent Part

Demand Response message after a significant period of time. The correlation ID may or may not be populated, depending upon availability in the partner system.

6.2.2 Security Classification

Security Classification markings are mandatory for EDE Business messages and applies to the message payload as a whole. Some uses of Security Classification:

1) Supports restricting access to message content based upon security levels;
2) Human oriented processes can interact based upon sensitivity of the message payload;
3) Supports storage and routing of content based upon security markings.

6.2.2.1 Security Classification Logical Structure

The following table gives a sample of values for some of the attributes used in the Security Classification block. Fleet-specific values will be defined prior to the implementation of the ICD (Interface Control Documents) by industry partner.

Attribute	Description	Service Consumer	Service Provider
Classification Type	An enumeration of the classification type which will apply to the business payload. enumeration evaluates to: • Top Secret • Secret • Unclassified	Populates as per payload classification.	Handles message in a method commensurate with its classification type.
Designation Type	An enumeration of the designation type which will apply to the business payload. Enumeration evaluates to: • Protected A • Protected B • Protected C • CH147-Boeing • C130J-LMA	Populated as per payload designation.	Handles message in a method commensurate with its designation type

Attribute	Description	Service Consumer	Service Provider
Releasable Type	An enumeration of the releasable type which will apply to the business payload. Enumeration evaluates to: • Canada only • USA only • Public • Industry X[2]	Populated as per information available within consumers information systems.	Handles and routes message in a method commensurate with its releasable type

Note: Actual values will be provided by Canada prior to implementation of the interfaces.

6.3 Business Identifier

Business identifiers provide a unique identity to an instance of a business object that is being exchanged. Listed below are some definitive uses of Business Identifier:

1. Supports relaying the specific business information to all layers of the solution stack;
2. Human oriented processes can observe the state of a specific business transaction in their native form;
3. Technical oriented process both system and human can now intelligently be able to support processing based on business identity;
4. Supports providing specific business context regardless of when an acknowledgement or response is being provided to a service consumer;
5. Always provide status in the context of a business transaction context with a specific Business Identifier association, thus enabling a clear concise view of the business transaction;
6. Is always associated with an underpinning technical reference thus giving bidirectional navigation.

6.3.1 Business Identifier Logical Structure

The logical structure for a Business Identifier will vary dependent upon the Business Objects it is representing. The construct of a business identifier for each business object must be agreed upon with the Business Message owner.

[2] Where X – represents the releasability to a particular industry community/entity

A business identifier is always a pre-defined subset of elements from the business objects in a message. For example, the Business Identifier for a Notification will be different from that of a Work Order. These business identifiers will be defined within the schema definition using an attribute attached to the element. When a business identifier is required by the service provider or service consumer they will be able to define an appropriate XSLT function to extract the business identifier from the business object(s) in a message. The extracted elements and its value can then be included in the appropriate message and transmitted as required.

6.3.2 Business Identifier Usage

The use of a business identifier depends on the message type. The Business Identifier is used in:

- The Output Message to clearly identify for which business object the status applies;
- The Fault Message to clearly identify for which business object the reported fault applies.

Some examples of candidate business identifiers are shown below to present the reader with a perspective of what elements can be considered for business identifiers.

Note: Actual definitions will be reflected in the specific business message schema definition.

Message Type	**Business Identifier Attributes**
Part Demand Part Issue Part Receipt	Customer ID PO Number PO Line Number
Notification	Customer ID Notification ID
EMR	Customer ID CAGE Manufacturer Part Number (MPN) Serial Number

Note: The Unit of Work identifier may be added to the business identifier.

6.4 Message Body

6.4.1 Business Object Data

Business Object Data (BOD) contains the actual business objects communicated. They are often referred to as the business payload of a message.

In instances where systems have restriction with regards to the size limit for message being transmitted, the systems should support the maximum size of a single business object. A single business object will not be broken into fragments, thus ensuring a single business object will always be transmitted as a complete business object. For example a Purchase Order, with many Line items, is a single business object which will always be exchanged in a single message. On the other hand a set of Notifications sent according to the Periodic Model may be split between one or more service invocations and hence one or more messages.

6.4.2 Acknowledgement

The Acknowledgement block is communicated in the message body of the SOAP output. It is used in communicating successful receipt and validation of the message. Any errors in receipt or validation are communicated via SOAP Fault block in the case on request/response style of message.

6.4.3 Fault Message

Faults will be returned in the same HTTP session as the incoming message, using the <SOAP: Fault> construct in accordance with the WS Basic Profile 1.1 standard;

```
<SOAP: Fault>
   <faultcode>
   <faultstring>
   <faultactor>
   <detail>
</SOAP:Fault>
```

The error message with its business context will be represented as below, within the <detail> element of a SOAP: Fault.

6.4.3.1 Usage and Structure

An error message is generated by the provider if the provider environment is unable to accept the incoming message. An error message will be generated under the following conditions:

- requested service is unavailable;
- Supplied User Credentials are rejected;
- Supplied User Credentials do not match <BizHeader> information. Specifically, Exchange Type, Industry and Fleet;
- XML Message structure fails validation for attribute, type, or mandatory value.

6.4.3.2 Error Message Attributes

The Error message will support reporting of more than one error against more than one Business Identifier, and will be composed of the following attributes.

Attribute	Description	Service Consumer	Service Provider
Error Type	An enumeration of the error type which will apply to the fault payload. enumeration evaluates to: • Technical • Business		Populates with "Technical" value.
Business Identifier	As described above. Actual Business Identifier attributes will vary according to message type:		Populated as per message type, and business payload record generating the error.
Error Code	An enumerated list as currently exists is defined below: • Unauthenticated User • Unauthorized Request • Tampered Message • Invalid certificate • Expired certificate • Message Header Error • Malformed Message • Transformation Error • Service Unavailable • Routing Error • Business Error • Backend Error • Other **Note:** Actual values will be defined prior to implementation of Interfaces and will be defined by Canada		Populated as per type of error being reported.
Short Description	An optional field that may be used to qualify the error		Optional

Attribute	Description	Service Consumer	Service Provider
Error Message	A description of the error. Level of detail reported is dependent upon the type of error. Whenever possible, errors are reported down to the attribute level.		Mandatory. Populated as per the type of error being reported.

7. Note: Please refer to the Error Processing ModelOperational Model by Domains

The various business domains supported by EDE vary in their information exchange requirements thus require different runtime operational models. The specific operational runtime model for a business domain will specify:

- Message Exchange Pattern: Request-Response or One-Way;
- Message Flow Periodicity: Periodic, A-periodic, Near-Real-Time;
- Specific rules for setting Security Classification;
- Units of Work: Single Message, multiple messages – same type, multiple messages multiple types;
- Steps performed during Technical Validation Phase;
- Business Identifiers.

7.1 Master Data and Engineering Change Information

In both of these instances industry provides the information based on a specific set of business conditions. The information exchange requirement is not frequent, however when information exchange occurs there is typically a strong inter-relationship across multiple business object type messages. In order to process these business object type messages all of the constituent related business object type needs to be provided to Canada. A declaration by type of business object type and the quantitative count of the number distinct business object items for each type of business object type will have to be declared as well.

On complete receipts of the entire constituent business message Canada will process the information within its business systems.

The specific operational model for this domain will be discussed in the **Master Data and Engineering Change Service Operational Model,** an appendix to this document.

Note: Not all ISSCF/PBC platforms will use the operational model described above for this domain. The respective platform authority within Canada will declare their intent to their industry partner as to the particular operational model that they will be using for the above domain.

7.2 Maintenance History

Maintenance history data is generated during the conduct of first and second level maintenance and supported by Canada Maintenance Management Systems.

The information will be released on a periodic basis and will be dispatched to industry after being subject to a period of latency.

Industry will receive the information and will process it into its systems. There are no real-time requirements for information to be exchanged. However, in order to assure that Industry has received the message and it has passed all of the technical compliance processing phase, we will use the "Request/Response" model for delivery of the message.

The specific details for this information exchange domain will be discussed in the **Maintenance History Service Operational Model,** an appendix to this document.

7.3 Materiel Management (Supply)

The transactions that make up the supply interaction with industry are categorized into the real-time transaction type. The immediacy nature of the information by both Canada and Industry requires the interaction to be complete in order to ensure the respective processing by the back end systems can occur.

Thus all of the transactions for supply will follow the "Request/Response" pattern of communication.

The specific details for this information exchange domain will be discussed in the **Materiel Management Service Operational Model**, an appendix to this document.

7.4 Technical Problem Management (TPM)

The TPM transactions have the same characteristic as the supply transaction in that the Canada and Industry need to ensure that the specific transaction request is acknowledge and is being fulfilled.

Thus all of the transactions for TPM will follow the "Request/Response" pattern of communication.

The specific details for this information exchange domain will be discussed in the **Technical Problem Management Operational Model,** an appendix to this document.

7.5 Technical Services

These are transactions that are initiated by the infrastructure and are intended to communicate the failure of certain transactions. The information provided by these transactions should enable either Canada or Industry to take the appropriate remedial measures to rectify the cause of the errors.

Currently there is only one such service titled: **"Dead Message Service"** which enables both organizations to report the failure of processing of a particular respect that was initiated based on the specific non-functional characteristic that has been established for the particular service.

8. Appendix Domain Oriented Operational Model

Note: Only applicable operational model will be made available to industry partner based on the adoption by the platform authority within Canada - DND

8.1 Master Data and Engineering Change Information

See the Master Data and Engineering Change Service Operational model.

8.2 Maintenance History

See the Maintenance History Service Operational Model.

8.3 Materiel Management (Supply)

See the Materiel Management Service Operational Model.

8.4 Technical Problem Management

The Technical Problem Management Operational model will be published in the next release of this document.

8.5 Error Processing Model

The runtime error processing model for messages and the specific error codes that will be used are defined in the following document.

See“Error Model”

9. Action Indicator Usage

Action Indicator (AI) is an attribute that will be defined in XML schemas and the associated complex type within where the use of AI will aid the data consumer to apply the appropriate processing as per the AI values.

The specific enumerated values are defined in the associated XML schemas. They are defined as an enumeration as follows:

- 1 – Add
- 2 – Edit
- 3 – Delete
- 4 - Snapshot

Principle:

- The data source provider will determine the "Action" context (Add, Edit, and Delete).
- The data provider make this determination based on what it knows with regards to the data namely:
    1. Creation disposition
    2. Removal disposition of/from the business object
    3. Whether it was shared externally
    4. Whether it was changed or modified since being shared
- The data consumer will use the disposition of the "Action" and applies its own processing rule against the data set it is receiving and conducive to its usage of the information within its information systems.

Maintenance History Data Originated From Canada

Based on the above principles Canada will determine the action indicator rules as follows:

Starting with the lifecycle of the Business Object and its constituent parts:

Action = Create

- Primary Containment Business Object is created
- Primary Containment Business Object and Associated Child Records that are part of the Primary Business Objects are being created

Action = Edit

- Previously created Primary Containment Business Object has been modified
- Previously created Primary Containment Business Object and Associated Child Records are modified

NOTE**:** A modification on a child record is marked specific to the child record itself.

- Multiple independent child record could have various statuses (Create, Edit, Delete) within a record type grouping, but with the governing Primary Containment Business Object having an action indicator of "Edit"

Action = Delete

- Previously transmitted Primary Containment Business Object and Associated Child Records record have been marked for removal (a logical delete)
- Previously transmitted Child Records are being removed from the associated business object

Action = Snapshot

- Indicates that this record is a snapshot of the data that was previously shared and that this is the current view of this data in the source system
- Typically used to transmit the end state view of the data in the source system
- Currently applies only to maintenance history data and is applied to the last extract of the maintenance record.

Similarly, Industry will determine its disposition for the Action Indicator based on its knowledge of the data as it relates: **Master Data in the Context of Engineering Change.**

10. Service Non Functional Attributes

Term	Definition
ISSCF	In-Service Support Contract Framework
PBC	Performance Based Contracting
ACK Time Interval	The length of time for the service provider to provide a response within the same HTTP/HTTPS session. If no response received within this time period, Canada System will retry the service call.
Retry Time Interval	Maximum period for retrying a service call. If no success within this period, revert to Dead Message Handling. Alternatively, can use number of retries parameter.
Number of Retries	The Maximum numbers of attempts to retry a service call. If there is no success after this number of attempts, revert to Dead Message Handling. Alternatively, we can use the retry time interval parameter.
Time to Live Span	Maximum length of time EDE will wait for Industry to respond with a business acknowledgement message after an initial request was initiated.
Dead Message Handling	Out-of-Band (specific to the service in question) processing requirements when a message cannot be processed. Dead Message Handling Service will be used to communicate the nature of the issue being encountered.

11. Definitions, Acronyms, Abbreviations

Term	Description
AI	Action Indicator
BOD	Business Object Data
CMMS	Canada Maintenance Management System
DND	Department of National Defence
EDE	Electronic Data Exchange
EIE	Electronic Information Environment
EMR	Equipment Master Record
ISSCF	In Service Support Contracting Framework
MPN	Manufacturer Part Number
PBC	Performance Based Contracting
PKI	Public Key Infrastructure
QA	Quality Assurance
SC	Service Consumer
SOA	Service Oriented Architecture
SP	Service Provider
SSL	Secure Socket Layer
TDC	Test and Development Centre
TPM	Technical Problem Management
WS	Weapon System
WSDL	Web Services Description Language
WS-I	Web Services Interoperability
XML	Extensible Mark-up Language
XSD	XML Schema Definition

12. External References

1	IBM DeveloperWorks SOA and Web Services: http://www.ibm.com/developerworks/webservices/
2	Web Services Interoperability Organization: http://www.ws-i.org/
3	XML Schema Definition (XSD) 1.0: http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/ and http://www.w3.org/TR/2001/REC-xmlschema-0-20010502/ http://www.w3.org/TR/2007/WD-xmlschema11-1-20070830/ http://www.w3.org/TR/2006/WD-xmlschema11-1-20060831/
4	SOAP 1.1 Specification: http://www.w3.org/TR/2000/NOTE-SOAP-20000508/
5	WSDL 1.1 Specification: http://www.w3.org/TR/2001/NOTE-wsdl-20010315
6	WS-I Basic Profile (BP) Version 1.0, April 16, 2004: http://www.ws-i.org/Profiles/BasicProfile-1.0-2004-04-16.html
7	WS-I Basic Profile (BP) Version 1.1, Final Material, April 10, 2006: http://www.ws-i.org/Profiles/BasicProfile-1.1.html
8	WS-I Simple Soap Binding Profile (SSBP) Version 1.0, August 24, 2004: http://www.ws-i.org/Profiles/SimpleSoapBindingProfile-1.0-2004-08-24.html
9	WS-I Attachments Profile Version 1.0, Final Material, April 20, 2006: http://www.ws-i.org/Profiles/AttachmentsProfile-1.0.html
10	Web Services Security: SOAP Message Security 1.0 (WS-Security 2004), OASIS Standard 200401, March 2004: http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf
11	Related Information “Which style of WSDL should I use?”: Russell Butek, October 31, 2003, http://www.ibm.com/developerworks/webservices/library/ws-whichwsdl/

13. Document History

Revision No.	Description	Date
Draft - 0.8	Release to Industry	09 June 2011
1.0	Release to Boeing	21 November 2011
1.1	Added process for handling dead message	07 December 2011
1.2	Updated section 8.1	27 January 2012
1.3	Updated Section 6.2	15 June 2012
1.4	Updated to remove reference EIE. Added exception notes for use of Operational Model. Included reference to Error Processing Model appendix	03 Aug 2012
1.5	Removed PROTECTED-A markings from document and add proviso to page footer.	10 June 2013
1.6	Reference PBC for Navy	8 October 2015