

**Electronic Court Filing Service
*Version 4.01***

 **Springboard**

**Quality Assurance Review**

***Version 1.0.1***

 **October 2013**

Table of Contents

[**Introduction** 3](#_Toc371390470)

[**Scope** 3](#_Toc371390471)

[**Issues** 4](#_Toc371390472)

[**I-1. Package Structure** 4](#_Toc371390473)

[**I-2. WSDL Validation** 4](#_Toc371390474)

[**I-3. Document Reference** 5](#_Toc371390475)

[**I-4. Schema Encoding** 5](#_Toc371390476)

[**I-5. Juvenile Reference** 6](#_Toc371390477)

[**I-6. Inconsistent Reliable Messaging Standards** 6](#_Toc371390478)

[**I-7. Invalid Sample** 6](#_Toc371390479)

[**I-8. Incorrect Parameter Reference** 7](#_Toc371390480)

[**I-9. Requirement Typo** 7](#_Toc371390481)

[**I-10. Error Handling** 8](#_Toc371390482)

[**Observations** 9](#_Toc371390483)

[**O-1. Optional Interoperability?** 9](#_Toc371390484)

[**O-2. Must and/or May?** 9](#_Toc371390485)

[**O-3. Simplify Requirements** 10](#_Toc371390486)

[**O-4. Binary Encoding** 10](#_Toc371390487)

[**O-5. Substitution groups** 10](#_Toc371390488)

[**O-6. GRA Reference** 11](#_Toc371390489)

[**O-7. WS-Addressing Not Specified** 11](#_Toc371390490)

# **Introduction**

The Springboard Quality Assurance (QA) Review serves as an independent assessment of an interoperability specification. This assessment is typically conducted as part of a Springboard certification initiative in conjunction with the development of the conformance package. The review addresses all topics related to specification conformance, including how features are classified and the distinction between normative requirements and informative explanations. As stated in the World Wide Web Consortium (W3C) QA Specification Guidelines, “clear presentation of what is meant by conformance to the specification is ultimately crucial to successful interoperability of implementations” (<http://www.w3.org/TR/qaframe-spec/>). The Springboard QA Review aims to enhance overall specification clarity while maintaining focus on the testability of the conformance requirements. Thus, the goal of the review is to help improve the precision of the specification, to make the documentation easier to interpret without ambiguity and clearer as to what is required in order to conform.

# **Scope**

This QA review is being conducted as part of the Springboard Electronic Court Filing (ECF) Initiative, which is based on the Organization for the Advancement of Structured Information Standards (OASIS) ECF Standard. The review has been conducted for the following particular specifications:

Electronic Court Filing Version 4.01 (ECF v4.01)

<http://docs.oasis-open.org/legalxml-courtfiling/specs/ecf/v4.01/ecf-v4.01-spec/os/>

Electronic Court Filing 4.0 Web Services Service Interaction Profile Version 2.01 (ECF WS SIP 2.01)

<http://www.oasis-open.org/committees/download.php/43733/ecf-v4.0-webservices-v2.01-csprd02.zip>

The feedback below has been broken into two categories, Issues and Observations, based on the perceived impact on testability and conformance assertion.

# **Issues**

### **I-1. Package Structure**

**Issue Details:**

* The WSDL files provided in the ECF v4.0 WS SIP 2.01 package are not clearly linked to the schema files contained in the ECF v4.01 package.
* The WS SIP document has 2 loose references to the WSDL - XSD relationships:
	+ On page 1 of the ECF v4.0 WS SIP 2.01 document, the “Related work:” section provides a reference to the overall Electronic Court Filing Version 4.0 specification. While the underlying link opens a specification that contains schema files, the link is not obvious. In addition, the link is *not* available if reviewing the “document” in the PDF version.
	+ On page 8 of the ECF v4.0 WS SIP 2.01 document, the “Message” section states that “Information transmitted between MDEs that consists of a well-formed XML document that is valid against one of the defined message structure schemas in the ECF 4.0 specification.”

**Suggestion:**

* Provide the schemas in the ECF v4.0 WS SIP 2.01 package so that an implementer is able to open the WSDL and validate without having to first search for the associated schemas and build a structure consistent with the WSDL ’s relative schema paths.
* Alternatively, embed a metadata or readme file in the ECF v4.0 WS SIP 2.01 package that accompanies the WSDL files and clearly directs the reader/implementer to the schemas.

### **I-2. WSDL Validation**

**Issue Details:**

* The schemas included in the ECF v4.01 package have an inconsistency in the UBL schema namespace/reference which prevents complete validation of the WSDL.
* The ECF-4.0-FeesCalculationResponseMessage.xsd file contains a schema import that links the "urn:oasis:names:specification:ubl:schema:xsd:CommonAggregateComponents-2" namespace with version 2.0 of the associated schema.
* The ECF-4.0-PaymentMessage.xsd and the ECF-4.0-PaymentReceiptMessage.xsd schemas assign the same namespace to version 2.1 of the associated schema.

**Suggestion:**

* Verify that the correct version of the schema files is referenced for the "urn:oasis:names:specification:ubl:schema:xsd:CommonAggregateComponents-2" namespace.
* Ensure the WSDL and associated schemas validate.

### **I-3. Document Reference**

**Issue Details:**

* The ECF v4.0 WS SIP 2.01 document references an older version (4.0) of the ECF specification.
* On page 1 of the ECF v4.0 WS SIP 2.01 document, the “Related work:” section provides a reference to the overall Electronic Court Filing Version 4.0 specification
* The latest version of the ECF specification, however, appears to be version 4.01.

**Suggestion:**

* Provide clarification regarding which version(s) of the ECF specification are acceptable for the ECF v4.0 WS SIP 2.01.

### **I-4. Schema Encoding**

**Issue Details:**

* The constraint schemas included in the ECF v4.01 package do not open in <oXygen/> XML Developer v14.1.
* Upon opening the files, <oXygen/> XML Developer reported “Bidirectional Text Detected” and that “The document <http://docs.oasis-open.org/legalxml-courtfiling/specs/ecf/v4.0/ecf-v4.0-spec/xsd/constraint/niem/ansi-nist/2.0/ansi-nist.xsd> contains characters from the BIDI Unicode range or its orientation was changed.”
* After further inspection, we discovered that:
	+ Two of the constraint schemas have been properly encoded with UTF-16 as shown in the following screen snippet from a binary editor:



* + The remaining constraint schemas have **not** been properly encoded with UTF-16 as shown in the following screen snippet from a binary editor:



 **Suggestion:**

* Since UTF-16 encoding does not seem necessary, we suggest updating the constraint schemas to match the UTF-8 character encoding used by the other NIEM Schemas.

### **I-5. Juvenile Reference**

**Issue Details:**

* The “ECF-4.0-JuvenileCase.xsd” schema does not define an nc:case substitution group
* The highlighted text in the following example is only missing from the Juvenile Case schema:

<xsd:element name="AppellateCase" type="AppellateCaseType" substitutionGroup="nc:Case">

**Suggestion:**

* Update the “ECF-4.0-JuvenileCase.xsd” schema file to include the substitutionGroup reference

### **I-6. Inconsistent Reliable Messaging Standards**

**Issue Details:**

* The ECF v4.0 WS SIP 2.01 document includes references to WS-Reliability, WS-ReliableMessaging 1.0 and WS-ReliableMessaging 1.1
	+ - The Introduction section states “… adds supports for message splitting and assembly through inclusion of **WS-Reliable Messaging 1.0**”
		- Section 1.2.9 states “The **WS-Reliability 1.1** ([WS-RM 1.1]) specification complements [WS-I BP 1.1] and defines a set of extensions for exchanging SOAP messages …”
		- Section 2.14 states “… a sending MDE MAY include reliability extensions to the SOAP envelope as defined in the [**WS-RM 1.1**] specification”
		- Section 2.15 states “**WS-Reliable Messaging** defines mechanisms by which messages MAY be split into multiple pieces …”

**Suggestion:**

* Consistently identify and specify the required OASIS WS-ReliableMessaging standard or just reference the corresponding Web Services Profile, which specifies the reliable messaging standard(s).

### **I-7. Invalid Sample**

**Issue Details:**

* ECF-4.0-CoreFilingMessage-BankruptcyNotice.xml sample is invalid
	+ - The content of element 'nc:PersonWeightMeasure' is not complete. One of '{"http://niem.gov/niem/niem-core/2.0":MeasureValue}' is expected.
		- The content of element 'nc:PersonWeightMeasure' is not complete. One of '{"http://niem.gov/niem/niem-core/2.0":MeasureValue}' is expected.
		- Invalid content was found starting with element '{"urn:oasis:names:tc:legalxml-courtfiling:schema:xsd:CommonTypes-4.0":CaseParticipantRoleCode}'. One of '{"http://niem.gov/niem/niem-core/2.0":EntityRepresentation}' is expected.
		- The content of element 'bankruptcy:Debtor' is not complete. One of '{"http://niem.gov/niem/niem-core/2.0":EntityRepresentation}' is expected.

**Suggestion:**

* Ensure that all sample XML instance documents are valid.

### **I-8. Incorrect Parameter Reference**

**Issue Details:**

* The provided operations tables in sections C.1.1, C.2.1 and C.3.1 each contain one instance of an operation that is defined with two “Parameters”. The table entries, however, do not clearly identify whether the parameters are both optional, both required or conditional.
* After reviewing the web service definition, we discovered that the actual base parameters are defined in the WSDL and extend the “ElectronicFilingMessageType” type. As such, the parameters column, depicted below, does not seem to properly represent the actual web service definition

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Called By** | **Output** | **Parameters** |
| ReviewFiling | Filing Assembly MDE | xsd/message/ECF-4.0-MessageReceiptMessage.xsd : MessageReceiptMessage | xsd/message/ECF-4.0-CoreFilingMessage.xsd : CoreFilingMessage |
| xsd/message/ECF-4.0-PaymentMessage.xsd : PaymentMessage |

**Suggestion:**

* Update the tables in sections C.1.1, C.2.1 and C.3.1 to properly reflect the base types.
* Additionally, consider moving the full type declaration from the WSDL to a message schema.

### **I-9. Requirement Typo**

**Issue Details:**

* The table in section “C.2.1 Provided Operations” has the following typo in the Output column:

|  |  |  |  |
| --- | --- | --- | --- |
| **Operation** | **Called By** | **Output** | **Parameters** |
| GetPolicy | Filing Assembly MDE | xsd/message/ECF-4.0-CourtPolicyQueryMessage.xsd : CourtPolicyReponseMessage | xsd/message/ECF-4.0-CourtPolicyQueryMessage.xsd : CourtPolicyQueryMessage |

**Suggestion:**

* Correct the typo so that the schema file name and type name match

### **I-10. Error Handling**

**Issue Details:**

* Section 2.4 of the ECF WS SIP states:
	+ - The response to a request for an operation not supported by the court MUST be reported using the ECF 4.0 <ErrorCode> element in the core message and MAY also include a SOAPFault in the SOAP envelope.
* The requirement seems to have two issues:
	+ - First, in the event that a web service has not implemented an operation, the client will receive a standard SOAP fault, not an application-level error from the server.
		- Also, a service cannot generate two synchronous responses to the same message, so either a response with the <ErrorCode> element or a SOAP fault will be sent.

**Suggestion:**

* The ECF WS SIP should be revised to clearly define SOAP faults as the mechanism used to communicate error information between web service exchange partners.
	+ - All SOAP-based web service applications must be able to handle standard SOAP faults, which will be the default mechanism used to communicate *non-application* errors. Therefore, using the same method to send *application* specific errors, allows interface systems to handle all errors, *infrastructure* as well as *application*, in a common way.
		- The Error element (and associated type) defined in the ECF-4.0-CommonTypes schema will not be utilized by systems that implement interfaces consistent with the ECF WS SIP. The ECF-4.0-CommonTypes schema will not need to change, since the WS SIP interfaces will simply ignore any Error elements found in the message responses.

# **Observations**

### **O-1. Optional Interoperability?**

**Comments:**

* The ECF Core Specification (Section 5.1) and the ECF WS SIP define numerous optional (May/Should) requirements for several foundational aspects of the service interaction.  As such, the specifications seem to defer core decision points to the system owners/developers, thereby diminishing the overall interoperability of the system.
* Example: [ *two sites both adhere to the specification, but cannot communicate* ]
	+ - Site A decides they want to leverage the optional Message Integrity [XMLSIG] and Message Confidentiality [XMLENC] parts of the specification to securely send/receive message data.
		- Site B, on the other hand, builds their system with only the base requirements.
		- Site A sends an encrypted message to Site B, but Site B is not able to handle the message due to the confidentiality encoding [XMLENC] of the message.

### **O-2. Must and/or May?**

**Comments:**

* The specification documents seem to provide contradictory guidance regarding some of the normative (Must/Required) and non-normative (May/Should) requirements.
* Section 5.1 “Service Interaction Profile Requirements” of the ECF Core Specification indicates that a SIP “***SHOULD*** provide”:
	+ - Message non-repudiation
		- Message integrity
		- Message confidentiality
		- Message authentication
		- Message transmission reliability
* Sections 1.2.5 - 1.2.8 of the ECF WS SIP, on the other hand, seem to indicate that many of those optional features are actually required for compliance.
	+ - Example (from section 1.2.8):

The following options in [WS-I BSP 1.0] are ***REQUIRED*** for compliance with this web services service interaction profile:

* + - * E0002 - Security Tokens - Security tokens MUST be specified in additional security token profiles. (NOTE: This will be determined in Court Policy)
			* R3103 - A SIGNATURE MUST be a Detached Signature as defined by the XML Signature specification.
* Sections 2.10 - 2.14 of the ECF WS SIP seem to agree with Section 5.1 of the ECF Core Spec that the capabilities are ***OPTIONAL***, while *contradicting* sections 1.2.5 - 1.2.8 of the ECF WS SIP.

### **O-3. Simplify Requirements**

**Comments:**

* The ECF WS SIP provides requirement detail that, at times, seems to repeat, or delve into the details of, other specifications. In order to simplify the requirement definitions, the specification could identify requirements only at the higher level.
* Example:
	+ - WS SIP calls for the use of MTOM, but then specifies deeper level requirements such as using **[RFC2045]** for themessage delimiter and unique “Content-ID**.**
		- Since RFC 2045 is included in the MTOM specification, the WS SIP does not need to repeat the requirement as the supplemental detail requires the reader to dig further into the specs to confirm that MTOM indeed implements RFC 2045.
* The ECF WS SIP would be more straightforward and easier for the reader to read/ developer to implement, if the “nested” requirements were not explicitly identified.

### **O-4. Binary Encoding**

**Comments:**

* The ECF WS SIP identifies the use of MTOM for all request/response message encoding. We recommend consideration of the following aspects of the binary encoding decision:
	+ - Consistency:
			* Since MTOM is a SOAP-based solution, an alternative binary encoding/storage mechanism must be provided for any non-soap SIP.  As such, the core data specification must provide another method for encoding binary data.
		- Performance:
			* MTOM encoding has more overhead than text/xml encoding for simple text-based messages and even small (< 5 KB) binaries.  Therefore, the composition of message data is an important factor in determining the encoding mechanism.
* NIEM compliant base64 encoding, which is built into the XML specification, would provide a consistent mechanism for binary encoding and transport across all SIPs.

### **O-5. Substitution groups**

**Comments:**

* Substitution groups provide message exchange systems with a high degree of flexibility and extensibility.  Substitution groups, however, also greatly increase the complexity of a system.
* In many cases, the complexity of substitution groups far outweighs any practical, real-world benefits.  As such, implementation teams (analysts, developers, tests) must accommodate the difficulties of substitution groups without realizing the extensibility benefits.
* As an example, automated schema validation becomes more difficult with substitution groups. Further, test tools, such as soapUI, do not fully support service that use substitution groups.
* Since a substitution group is really just another way to write a <choice>, we recommend consideration of moving from substitution groups to choices.

### **O-6. GRA Reference**

**Comments:**

* The ECF v4.0 WS SIP enumerates the older JRA WS SIP v1.1 as a Non-Normative requirement rather than referencing the current GRA Web-Services Service Interaction Profile v1.3 specification.
* In addition, to referencing an older specification, the reference also incorrectly connotes the JRA rather than GRA and provides a broken link to the specification.
* The following depicts the reference in question:

[JRA WS-SIP]

*Global Justice Reference Architecture Web Services Service Interaction Profile 1.1*, <https://it.ojp.gov/process_links.jsp?link_id=5800>, Global Infrastructure/Standards Working Group, August 1, 2007

* Note: The ECF v4.0 WS SIP Sections 1.5 “Normative References” and 1.6 “Non-Normative References” seem to provide confusing, if not contradictory, information.

### **O-7. WS-Addressing Not Specified**

**Comments:**

* SIP does not overtly specify the use of WS-Addressing although other normative standards in the Profile require the use of WS-Addressing
* Challenge - Source and destination identifiers are included in the SOAP body. Under these circumstance when message level encryption is used for the SOAP body, any exchange other than a direct source to destination exchange would not be defined for the Profile
* Recommendation – Consider a formal binding for address-related information between ECF and WS-Addressing, e.g. specific requirement to duplication destination in the WS-Addressing To element