**ECF 4.1 WD04 and Web Services SIP WD02 Feedback**

This document provides review feedback to the recently released ECF 4.1 WD04 specification and ECF Web Services SIP WD02 specification.

**ECF 4.1 WD04 Feedback**

1. Page 1, Additional artifacts – are the URL path names provided for ‘XML Schema’, ‘XML sample messages’, ‘Model and documentation’, ‘Genericode code lists’, and ‘Specification metadata’ correct?

Each URL path includes a ‘/errata02’ component. Since ECF 4.1 is new, it does not yet have errata as does ECF 4.01.

2. In Section 1.2, provide a definition for COSCA/NACM (perhaps in the Symbols and Abbreviations section). Persons outside the court/justice community may not know what this acronym represents.

3. Figure 1 Simple Message Stream & Figure 2 Message Stream with a Document in Multiple Attachments, still contain ‘4.0’ and not ‘4.1’ – could live with this if necessary. Perhaps the statement in section 1.3 ECF Version 4.1, that “all references in this document to ECF 4.0 apply to ECF 4.01 and 4.1 as well” is intended to cover this.

4. The section numbering for ‘Filer and Party Identifiers’ (was 3.3.1.8) got whacked and needs restoration.

5. In section 5.3 Supported Service Interaction Profiles, shouldn’t the new Web Services SIP also be included?

Note: I get a bit confused on Web Services SIP version numbering. In the latest version of the Web Services SIP specification document (WS-SIP v4.1? or is it ‘stage’ not ‘version’?) i.e., ecf-v4.1-webservices-wd02.docx, it states that the new specification replaces or supersedes:

* Web Services Messaging Profile 1.0 Specification
* Web Services Interaction Profile 1.1 Specification
* Web Services Interaction Profile 2.01 Specification

In the latest ECF 4.1 specification (ecf-v4.1-spec-wd04.docx), section 5.3 Supported Service Interaction Profiles, it lists both:

* Web Services Service Interaction Profile 2.0 Specification
* Web Services Service Interaction Profile 2.1 Specification

Is it correct that somehow WSSIP 2.01 is replaced/superseded, but WSSIP 2.0 is still supported and compatible with ECF 4.1?

6. In section 3.1 The Filing-Preparation-to-Docketing Process Model, the NotifyFilingReviewComplete is no longer bold (is no longer a required operation).

This revision is consistent with one of the goals of the ECF 4.1 specification, in that the cardinality for ecf:SendingMDELocationID and ecf:SendingMDEProfileCode in ecf:CaseFilingType has been relaxed (now optional) to enable MDEs to send request exchanges without requiring any asynchronous response.

This is further explained in the ECF 4.1 specification in section 3.3.1.6 MDE Identifiers by the inclusion of the new paragraph:

ECF 4.1 messages that support asynchronous responses include <SendingMDELocationID> and <SendingMDEProfileCode> to support the return of the asynchronous response to the sending MDE. If the <RequireAsynchronousResponsesIndicator> in the CourtPolicyResponseMessage is “true”, then both <SendingMDELocationID> and <SendingMDEProfileCode> MUST be included in all ECF 4.1 messages that include these elements.

The implication, of course, is that these two elements must be included and must also provide appropriate values (not included as empty elements).

There are only two ECF messages (i.e., CoreFilingMessage, via ReviewFiling & RecordDocketingMessage via RecordDocketing) for which an asynchronous response may be expected.

As such, should not NotifyDocketingComplete also be ‘not-bold’ (e.g., optional) in section 3.1?

7. While on the topic of required ECF operations, perhaps it is an appropriate moment to also consider the reason for RecordFiling is a required operation.

Certainly, in the logical work-flow sense, courts need to docket accepted e-filing submissions. However, in many (perhaps most) systems today, the docketing functions are not invoked by the sending of an ECF RecordFilingRequest. In some CMSs, the eFiling clerk review function is provided within the CMS, and upon submission acceptance, the court docketing functions are initiated without a formal XML based ECF RecordFiling operation employing an ECF approved SIP.

Can a court not be ECF compliant if they do not implement clerk review to docketing functions using a formal ECF RecordFiling, XML based, operation invocation method?

If RecordFiling becomes an optional operation, then section 3.2.6 RecordFiling will need to be updated.

8. Again, in section 3.1, line 606, it says: “at any point after the NotifyFilingReviewComplete operation, if the case is acceptable, a party MAY access information through the following operations:”.

Since NotifyDocketingComplete may now be optional (per Court Policy), perhaps this statement should be reworded.

From a practical perspective, for subsequent filing submissions, a party need not wait for clerk review results to get case, case list or document information. Waiting until after clerk review may only be important when the party wants the GetCaseList, GetCase or GetDocument response to also include information and documents from within the recently provided and recently clerk reviewed submission. Furthermore, if the party wants their recent submission included in the GetCaseList, GetCase and/or GetDocument response, they must surely need to await court docketing completion, not just clerk review completion.

Consider the following rewording:

At any point after the completion of the ReviewFiling and RecordDocketing operations, if the filing submission is accepted, a party MAY access information that includes their newly accepted filing through the following operations:

9. 3.2.8 NotifyFilingReviewComplete – if NotifyFilingReviewComplete is now optional (per Court Policy) then the word “MUST” (line 687) requires revision or needs to be qualified, such as:

If the clerk rejects the filings or the Filing Review MDE receives the Notify Docketing Complete message, the Filing Review MDE either MUST invoke the NotifyFilingReviewComplete operation on the Filing Assembly MDE as a callback message to the ReviewFiling operation to indicate whether the filing was accepted and docketed by the clerk and court record system, or if the Court Policy does not mandate the NotifyFilingReviewComplete operation, and the Review Filing Request provided SendingMDELocationID and SendingMDEPorfileCode, then the Filing Review MAY invoke NotifyFilingReviewComplete.

10. 3.2.7 NotifyDocketingComplete – If NotifyDocketingComplete is now optional, then this section may need similar treatment as above for NotifyFilingReviewComplete.

**ECF WebServices SIP WD02 Feedback**

1. Page 1, “This specification is related to:” includes WSDL documents: ECF-4.1-WebServicesProfile-Definitions.wsdl, ECF-4.1-WebServicesProfile-ImplementationExample.wsdl.

Is it not true that there is no longer any ECF-4.x-WebServicesProfile-Definitions.wsdl in this newest WSSIP (replaced by CourtRecordMDE.wsdl, FilingAssemblyMDE.wsdl, FilingReviewMDE.wsdl and ServiceMDE.wsdl)?

Similarly, has not ECF-4.x-WebServicesProfile-ImplementationExample.wsdl been superseded by the 4 WSDL files in the ‘examples’ folder?

2. The reference to the citation for the specification on page 2 (Citation Format) still refers to Working Draft 01 even though this is now working draft 02. Not a big deal, as long as it is correct in the final version.

3. In section 2.5 Request and Operation Invocation is states that “each message transmission MUST identify the operation being invoked …”. However, just above in 2.4 Operation Addressing is states “each message transmission MUST either identify the operation or operations being invoked or …”.

Is it possible to invoke multiple operations in a single transmission or should the word ‘operation’ only be singular?

If multiple operation invocations in a single transmission are possible, how would this be done? Examples? How might this impact section 2.5 Request and Operation invocation that only uses the word ‘operation’ in the singular?

4. Missing ‘space’ character in section 3 Service Definitions between ‘WSDL’ and ‘file’.

5. As of ECF 4.1 WD04 and the newly revised wrappers.xsd, the example in section D.1 Operation Invocation is out-of-date. The current example shows a two-layered SOAP request (i.e., ReviewFilingRequestMessage, and CoreFilingMessage/PaymentMessage).

The example should be updated to show a three-layered SOAP request with ReviewFiling as the Operation Name element, ReviewFilingRequest as the Container element, and CoreFilingMessage/PaymentMessage as the ECF Message elements.

6. The example in D.3 Asynchronous Response requires updating to show a three-layered SOAP request (e.g., NotifyFilingReviewComplete, NotifyFilingReviewCompleteRequest, and ReviewFilingCallbackMessage/PaymentReceiptMessage).