**ECF5 Feedback 24 – Reviewed Documents**

An ECF TC task team met telephonically to review a Civil Complaint use case message example set, provided in Working Draft 29 (WD29). When reviewing the RecordDocketingRequest example (i.e. civil-complaint-003-RecordDocketingRequest-01.xml) there was discussion and debate regarding the appropriate action to take for various elements within ecf:ReviewedConnectedDocument and ecf:ReviewedLeadDocument, for circumstances when there were no document information changes made in the clerk review function, as well as circumstances where document information had been changed in clerk review.

The team sought to both clarify the mark-up in these circumstances, as well as to simplify the mark-up if possible. It was suggested that there may need to be some redesign to resolve these issues and that the proposal included in feedback #10 should be reviewed as a starting point for any redesign consideration.

This issue arises because of section 6.4.3 docket:RecordDocketingMessage, which provides that if a clerk made any modifications to the original filing information, then the modified information SHOULD be included in … ecf:ReviewedLeadDocument and ecf:ReviewedConnectedDocument.

The Civil Complaint RecordDocketingRequest example reviewed by the task team included an exhibit document that was accepted in clerk review and did not have any document changes made during clerk review (xml excerpt provided below). So, per 6.4.3, it should not be necessary to repeat ecf:FilingConnectedDocument information into ecf:ReviewedConnectedDocument since no document information changes were made. Nevertheless, clerk review results for the document must be included.

Below is an excerpt for the reviewed exhibit document from civil-complaint-003-RecordDocketingRequest-01.xml as reviewed by the task team:

Example 1:

<ecf:ReviewedConnectedDocument structures:id="ReviewedDocument3"> <!-- Exhibit -->

<nc:DocumentIdentification>

<!-- Does this element repeat the value from FilingConnectedDocument? -->

<!-- In ECF 4, this element was used as a content reference to identify the FilingConectedDocument being reviewed. -->

<!-- This function is now performed by nc:DocumentAssociation -->

<!-- At least one nc:DocumentIdentification is required so either the FilingConnectedDocument value is repeated or a new value is assigned. -->

<!-- This example shows repeating the FilingConnectedDocument value -->

<nc:IdentificationID>1</nc:IdentificationID>

<nc:IdentificationSourceText>FilingAssembly MDE</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<ecf:DocumentAugmentation>

<!-- Since the DocumentFiler is not revised from FilingConnectedDocument then should this element be present ? -->

<!-- Per the total replacement methodology, it should be absent. -->

<!--

<ecf:DocumentFiler>

<nc:EntityPerson structures:ref="Person1" xsi:nil="true"/> --> <!-- Plaintiff John W Doe -->

<!-- </ecf:DocumentFiler> -->

<nc:DocumentAssociation>

<nc:PrimaryDocument structures:ref="Document2" xsi:nil="true"/> <!-- Exhibit -->

<ecf:DocumentAssociationAugmentation>

<ecf:DocumentRelatedCode>reviewed</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

<ecf:ReviewedDocumentAugmentation>

<ecf:DocumentReviewStatus>

<nc:StatusText>accepted</nc:StatusText>

<nc:StatusDate>

<!-- Date and Time that acceptance disposition was applied - should be earlier or same as nc:DocumentPostDate -->

<nc:DateTime>2008-07-07T14:17:31.0Z</nc:DateTime>

</nc:StatusDate>

</ecf:DocumentReviewStatus>

<ecf:DocumentReviewer>

<nc:EntityPerson structures:id="Clerk1">

<nc:PersonName>

<nc:PersonGivenName>Mary</nc:PersonGivenName>

<nc:PersonSurName>Smith</nc:PersonSurName>

</nc:PersonName>

</nc:EntityPerson>

</ecf:DocumentReviewer>

</ecf:ReviewedDocumentAugmentation>

</ecf:ReviewedConnectedDocument>

1. **Clerk Review Document Association**

After a review of Feedback #10, it was recognized that the incorrect nc:DocumentAssociation element within the reviewed connected document element had been used in the example message. In the XML RecordDocketingRequest excerpt above (i.e. Example 1), it can be seen that nc:DocumentAssociation had been used to reference the FilingConnectedDocument reviewed as the ReviewedConnectedDocument. Although this ‘reviewed’ association is necessary, the wrong nc:DocumentAssociation element had been used. The nc:DocumentAssociation element within ecf:ReviewedDocumentAugmentation should be used instead, and not the nc:DocumentAssociation within ecf:DocumentAugmentation.

I recommend the following revision in the specification document to better identify the correct nc:DocumentAssociation element to use for reviewed documents:

Currently section 6.1.6 includes:

In the RecordDocketingMessage, ecf:ReviewedLeadDocument MUST reference filing:FilingLeadDocument and ecf:ReviewedConnectedDocument MUST reference filing:FilingConnectedDocument using nc:DocumentAssociation/nc:PrimaryDocument, and ecf:DocumentRelatedCode with a value of “reviewed”.

Revised to include the highlighted text:

In the RecordDocketingMessage, ecf:ReviewedLeadDocument MUST reference filing:FilingLeadDocument and ecf:ReviewedConnectedDocument MUST reference filing:FilingConnectedDocument using ecf:ReviewedDocumentAugmentation/nc:DocumentAssociation/nc:PrimaryDocument, and ecf:DocumentRelatedCode with a value of “reviewed”.

The corrected excerpt from above (i.e. Example 1) becomes:

Example 2:

<ecf:ReviewedConnectedDocument structures:id="ReviewedDocument3"> <!-- Exhibit -->

<nc:DocumentIdentification/>

<!-- The element ecf:ReviewedDocumentAugmentation below is necessary for each reviewed document. -->

<ecf:ReviewedDocumentAugmentation>

<ecf:DocumentReviewStatus>

<!-- nc:StatusText should be valid in FilingStatusCode.gc -->

<nc:StatusText>accepted</nc:StatusText>

<nc:StatusDate>

<!-- Date and Time that acceptance disposition was applied - should be earlier or same as nc:DocumentPostDate -->

<nc:DateTime>2008-07-07T14:17:31.0Z</nc:DateTime>

</nc:StatusDate>

</ecf:DocumentReviewStatus>

<ecf:DocumentReviewer>

<!-- the clerk who reviewed this document and applied the disposition above. -->

<nc:EntityPerson structures:id="ReviewingClerk1">

<nc:PersonName>

<nc:PersonGivenName>Mary</nc:PersonGivenName>

<nc:PersonSurName>Smith</nc:PersonSurName>

</nc:PersonName>

</nc:EntityPerson>

</ecf:DocumentReviewer>

<!-- This DocumentAssociation relates this ReviewedConnectedDocument to the FilingConnectedDocument provided in filing:FilingMessage below. -->

<nc:DocumentAssociation>

<nc:PrimaryDocument structures:ref="Document2" xsi:nil="true"/> <!-- Exhibit in filing:FilingMessage-->

<ecf:DocumentAssociationAugmentation>

<ecf:DocumentRelatedCode>reviewed</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:ReviewedDocumentAugmentation>

</ecf:ReviewedConnectedDocument>

A look back at the ‘docket.xml’ example (from the examples folder) revealed that it too has the same problem as above where the wrong instance of nc:DocumentAssociation is being used. The docket.xml example has been corrected, for this problem, and has also been modified to properly reflect the use of the ‘total replication method’ as agreed to in the Salt Lake City face-to-face meeting and referenced in 6.4.3.

1. **Nil-able Complex Element**

However, when updating the docket.xml example to include document information replication from FilingLeadDocument to ReviewedLeadDocument, a bit of a problem surfaced with regard to the use of ecf:DocumentRendition. However, there is an easy fix for this problem (i.e. two small edits to ecf.xsd).

So, when replicating information, such as document information, per 6.4.3 it is very useful to leverage structures:ref attributes to reference existing information that has not been revised without verbosely copying the information to another location within the message. Here’s an illustration from the docket.xml example:

In the docket.xml example, decedent information is provided in the filing:FilingMessage/nc:Case/civil:CaseAugmentation/civil:DecedentEstateCase element as shown below:

Example 3:

<civil:DecedentEstateCase>

<civil:Decedent structures:id="decedent1">

<nc:PersonBirthDate>

<nc:Date>1900-01-01</nc:Date>

</nc:PersonBirthDate>

<nc:PersonName>

<nc:PersonGivenName>Janice</nc:PersonGivenName>

<nc:PersonMiddleName>H</nc:PersonMiddleName>

<nc:PersonFullName>Doe</nc:PersonFullName>

</nc:PersonName>

</civil:Decedent>

<nc:PersonDeathDate>

<nc:Date>2017-01-01</nc:Date>

</nc:PersonDeathDate>

</civil:DecedentEstateCase>

In the docket:CorrectedCase/civil:CaseAugmentation, then the following can be used to convey that civil:Decedent information (i.e. nc:PersonBirthDate and nc:PersonName) are not revised in CorrectedCase:

Example 4:

<civil:DecedentEstateCase>

<civil:Decedent structures:ref="decedent1" xsi:nil="true"/>

<nc:PersonDeathDate>

<nc:Date>2017-01-01</nc:Date>

</nc:PersonDeathDate>

</civil:DecedentEstateCase>

This same approach can be applied with documents to reference document information and without needing to copy the information.

However, to use this technique for complex elements that contain mandatory child elements, the complex element must be both ‘nillable’ and provide the xsi:nil attribute. Unfortunately ecf:DocumentRendition does not. Adding nil-support to ecf:DocumentRendition makes the handling the problem of document information changes and addition of new document renditions in clerk review much more manageable. Adding nil-support for ecf:DocumentRendition is easy and can be achieved with the two highlighted edits to ecf.xsd shown below:

Example 5:

<xs:complexType name="DocumentRenditionType">

<xs:annotation>

<xs:documentation>Descriptors for a rendition of a Document. This is meant to include all the information about the document that is needed to enter it into the Document Management System.</xs:documentation>

</xs:annotation>

<xs:complexContent>

<xs:extension base="nc:DocumentType">

<xs:sequence>

<xs:element ref="cbrn:MultimediaDataMIMEKindText" minOccurs="0" maxOccurs="1"/>

<xs:element ref="ecf:ColorRelevantIndicator" minOccurs="0" maxOccurs="1"/>

<xs:element ref="ecf:DocumentRenditionHash" minOccurs="0" maxOccurs="1"/>

<xs:element ref="ecf:DocumentSignature" minOccurs="0" maxOccurs="1"/>

<xs:element ref="ecf:RedactedIndicator" minOccurs="0" maxOccurs="1"/>

<xs:element ref="nc:Attachment" minOccurs="1" maxOccurs="1"/>

</xs:sequence>

<xs:attribute ref="xsi:nil"/>

</xs:extension>

</xs:complexContent>

<!--Genericode code list MultimediaDataMIMEKindText.gc-->

</xs:complexType>

AND:

Example 6:

<xs:element name="DocumentRendition" type="ecf:DocumentRenditionType" nillable="true">

<xs:annotation>

<xs:documentation>Descriptors for a rendition of a Document. This is meant to include all the information about the document that is needed to enter it into the Document Management System.</xs:documentation>

</xs:annotation>

</xs:element>

With this nil-support added to ecf.xsd for ecf:DocumentRendition, then for the lead document modifications in docket.xsd, referencing the original rendition is as compact as:

Example 7:

<ecf:DocumentRendition structures:ref="OriginalLeadRendition" xsi:nil="true" />

This is very helpful in handling clerk review circumstances where document stamping has occurred (as well as any other scenario in which additional document renditions have been included).

1. **Document Information Changes and Removal**

Separating clerk review results from reviewed document information as suggested in Feedback 10, #4 would provide considerable improvement.

Since clerk review results must be included using ecf:ReviewedDocumentAugmentation within ecf:ReviewedConnectedDocument, the ecf:ReviewedConnectedDocument element cannot be left empty (e.g. xsi:nil=“true” cannot be used). Since ecf:ReviewedConnectedDocument must be provided, and since nc:DocumentIdentification is a required element, then this presents a mark-up conundrum. How does a message consumer understand the presence of nc:DocumentIdentification even when this element is left empty (i.e. as in <nc:DocumentIdentification/>). One possible interpretation is that ‘no document information has been changed’, the other interpretation is that ‘all document information has been revised (and cleared)’. As seen in the examples below, the ‘no changes’ permutation is exactly the same as the ‘all information cleared’ permutation:

Example 8- No Changes to Document Information

<ecf:ReviewedConnectedDocument structures:id="ReviewedDocument3"> <!-- Exhibit -->

<nc:DocumentIdentification/>

<ecf:ReviewedDocumentAugmentation> ... </ecf:ReviewedDocumentAugmentation>

</ecf:ReviewedConnectedDocument>

Example 9 – All Document Information Cleared

<ecf:ReviewedConnectedDocument structures:id="ReviewedDocument3"> <!-- Exhibit -->

<nc:DocumentIdentification/>

<ecf:ReviewedDocumentAugmentation> ... </ecf:ReviewedDocumentAugmentation>

</ecf:ReviewedConnectedDocument>

When replicating information, such as document information, per 6.4.3, it is very useful to leverage structures:ref attributes to reference existing information that has not been revised without verbosely copying the information to another location within the message. Below an illustration from an earlier version (i.e. WD 14) of the docket.xml example:

Example 10:

<ecf:ReviewedConnectedDocument structures:ref="Document2" xsi:nil="true"/>

<ecf:ReviewedLeadDocument structures:ref="Document1" xsi:nil="true"/>

This earlier WD14 version of the docket.xml example should be understood as communicating that there were no changes to document information, by referencing the filing document information using structures:ref. This both clean and concise.

Unfortunately, since clerk review results need to also be included (using ecf:ReviewedDocumentAugmentation) the reviewed document elements cannot be left empty (i.e. cannot include xsi:nil=true”). When not empty, then the nc:DocumentIdentification element must appear.

One solution might be to make nc:DocumentIdentification optional within ecf:ReviewedConnectedDocument and ecf:ReviewedLeadDocument (note that nc:DocumentIdentification is option in ECF 4.01). However helpful, this does not really address the true underlying problem.

There exists a fundamental inconsistency when indicating that ‘no changes’ were made in clerk review. For case, the absence of docket:CorrectedCase signals ‘no change’ to case information. For payments, the absence of a PaymentMessage with payment:CorrectedPaymentIndicator set to ‘true’ signals ‘no changes’ to payment information. Neither of these require complete information replication to indicate ‘no changes’ to relevant information.

Additionally, for case and payment there exists a method by which removal of all information can be indicated. For case, providing an empty docket:CorrectedCase element would indicate that all case information has been cleared. For payment, signaling the removal of all payment information would be done by including an ‘empty’ corrected version of the PaymentMessage (i.e. with payment:CorrectedPaymentIndicator set to ‘true’). Note however that neither docket:CorrectedCase nor payment:PaymentMessage provide the xsi:nil attribute, and both contain mandatory child-elements (i.e. CaseAugmentation for CorrectedCase, and cac:Address and payment:Payer for PaymentMessage (these sub-elements may be empty).

But with reviewed documents, ‘absence’ (e.g. of child document elements) signals change (i.e. information removal) not that the information is unchanged. As such, and without a design revision, reviewed document information must always be replicated in the review document element.

I feel that this is not right and changes to or removal of document information should follow the same or similar methods as changes to or removal of case and payment information.

An alternative solution could consider adding a new ‘outer’ element for a reviewed document, such as docket:DocumentReview. The docket:DocumentReview element would contain either ecf:ReviewedConnectedDocument or ecf:ReviewedLeadDocument as each is currently defined, but absent ecf:ReviewedDocumentAugmentation. The ecf:ReviewedDocumentAugmentation child elements could be packaged into a new docket:ReviewResults element. The docket:ReviewResults element would be a child element of docket:DocumentReview. A structure such as this would allow:

Example 11:

<docket:DocumentReview>

<ecf:ReviewedConnectedDocument structures:ref="Document2" xsi:nil="true"/>

<docket:ReviewResults>

<ecf:DocumentReviewStatus>

<nc:StatusText>accepted</nc:StatusText>

<nc:StatusDate>

<nc:DateTime>2008-07-07T13:47:42.0Z</nc:DateTime>

</nc:StatusDate>

</ecf:DocumentReviewStatus>

<ecf:DocumentReviewer>

<nc:EntityPerson structures:id="Clerk1">

<nc:PersonName>

<nc:PersonGivenName>Mary</nc:PersonGivenName>

<nc:PersonSurName>Smith</nc:PersonSurName>

</nc:PersonName>

</nc:EntityPerson>

</ecf:DocumentReviewer>

<nc:DocumentAssociation>

<nc:PrimaryDocument structures:ref="Document2" xsi:nil="true"/>

<ecf:DocumentAssociationAugmentation>

<ecf:DocumentRelatedCode>reviewed</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</docket:ReviewResults>

</docket:DocumentReview>

The option illustrated above allows for either:

<ecf:ReviewedConnectedDocument structures:ref="Document2" xsi:nil="true"/>

or the complete absence of ecf:ReviewedConnectedDocument, to be understood as no changes to document information in the reviewed document, whereas either:

<ecf:ReviewedConnectedDocument xsi:nil="true"/>

or:

<ecf:ReviewedConnectedDocument>

<nc:DocumentIdentification/>

<ecf:ReviewedConnectedDocument/>

would be understood to mean that all the document information has been removed during review (not a very likely circumstance, but possible I suppose).

**Summary:**

XML markup for ecf:ReviewedConnectedDocument and ecf:ReviewedLeadDocument requires the inclusion of ecf:ReviewedDocumentAugmentation to communicate clerk review results. The original filing document that has been reviewed must be identified using nc:DocumentAssociation within ecf:ReviewedDocumentAugmentation. Section 6.1.6 should be enhanced to clearly identify the appropriate nc:DocumentAssociation element for this purpose.

Complex elements which contain mandatory child elements, such as ecf:DocumentRendition should be made ‘nillable’.

To provide a consistent way to identify document information changes or removal in clerk review, a redesign is recommended. The redesign recommendation separates clerk review results for the reviewed document from document information.

The nc:DocumentIdentification element should be optional for ecf:ReviewedConnectedDocument and ecf:ReviewedLeadDocument.