**ECF 5 Spec Feedback and Considerations –**

Jim Cabral responses in red

This document contains additional questions and commentary resulting from a review at the Electronic Court Filing Version 5.0 Working Draft 09.

1. **Reviewed Filing Lead/Connected Documents**

In the prior feedback document I raised an issue with regard to missing Filing Lead/Connected Documents in RecordDocketingMessage (Item #8). This item is copied below with response in red text:

The elements filing:FilingLeadDocument and filing:FilingConnectedDocument are not provided on docket:RecordDocketingMessage. These are needed.

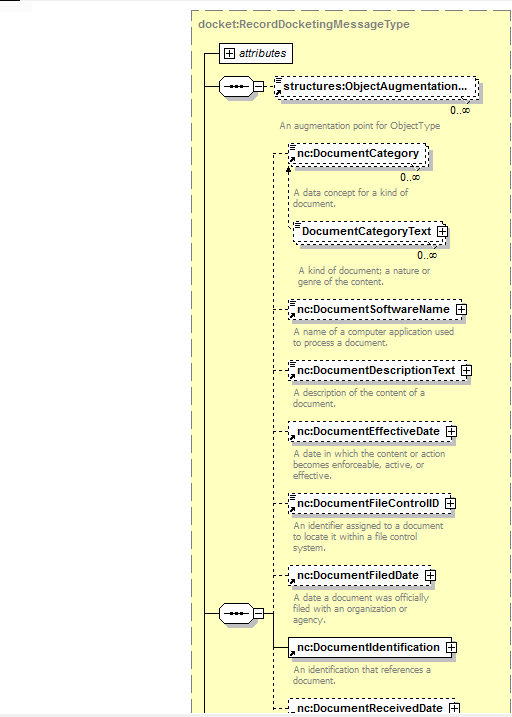
Perhaps you were thinking that these elements would be accessed through docket:CorrectedFiling ( the docket.xml example suggests that this may be the case). However, you had agreed to make docket:CorrectedFiling optional (i.e. “Modified the multiplicity to 0,1”) which means these document elements would not be available when docket:CorrectedFiling is not used.

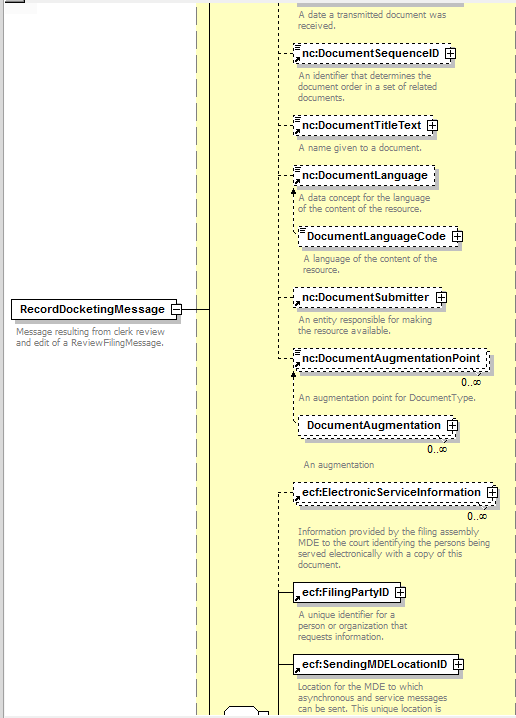
Also, even though you said that you had modified the multiplicity to 0,1, in work draft 08, docket:CorrectedFiling is still minOccurs = 1.

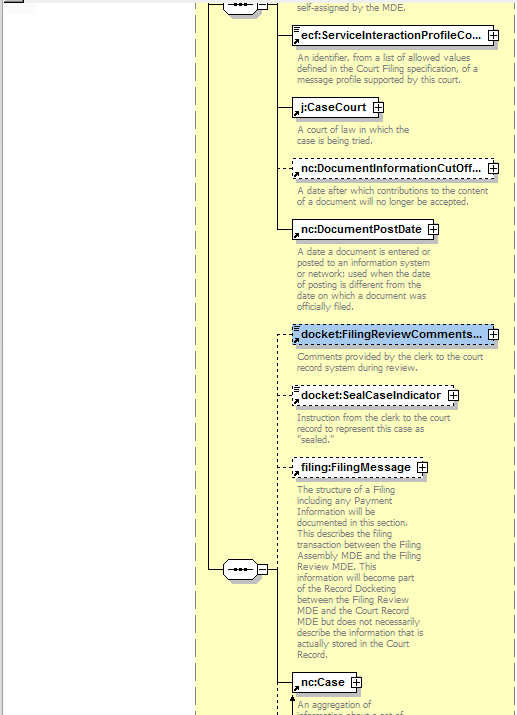
The ECF 5 ReviewDocketingMessage needs to include two versions of documents and case information – one pre-review and one post-review. The post-review documents are included in docket:ReviewedLeadDocument and docket:ReviewedConnectedDocument. I renamed docket:CorrectedFiling to filing:FilingMessage to provide the pre-review documents and case information. I fixed the cardinality and it is now 0,1.

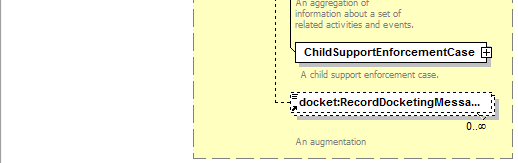
**Further feedback**:

It is correct that two versions of documents and case information (both pre-review and post-review) are needed. You have stated that the post-review documents are in the ReviewDocketingMessage as docket:ReviewedLeadDocument and docket:ReviewedCorrectedDocument. However, in wd09, I cannot find these elements in docket.xsd (i.e. ReviewDocketingMessage). Here is what I see using XMLSpy:





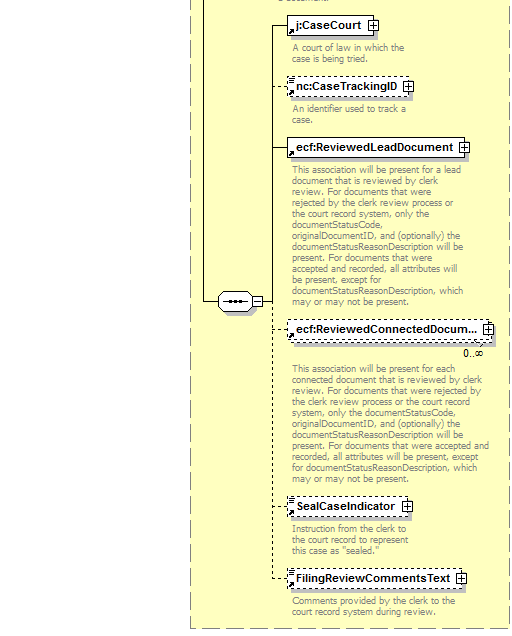




I do see where you renamed docket:CorrectedFiling as filing:FilingMessage, but I do not see docket:ReviewedLeadDocument and docket:ReviewedCorrectedDocument.

Am I missing something or looking in the wrong place? I would expect to find these elements similarly as to their current (ECF4) locations in ECF-4.0-RecordDocketingMessage.xsd, as shown below:

I was mistaken. ecf:ReviewedLeadDocument and ecf:ReviewedConnectedDocument are actually mapped to the callback messages, docketcallback:NotifyDocketingCompleteMessage and filingreviewcallback:NotifyFilingReviewCompleteMessage.



Also, I do not think renaming docket:CorrectedFiling as filing:FilingMessage is a correct remedy. Perhaps both are needed.

Section 4.1 Messages shows that the RecordDocketing operation takes two input messages, docket:RecordDocketingMessage and optionally, payment:PaymentMessage. In ECF4 (Appendix C.3.1) the RecordFiling operation required two messages; RecordDocketingMessage and CoreFilingMessage.

In ECF5, filing:FilingMessage replaces CoreFilingMessage.

So it seems there are two possible approaches in ECF5: (1) modify RecordFiling to take three messages (i.e. RecordDocketingMessage (with CorrectedFiling), FilingMessage, and optionally PaymentMessage), or (2) include both CorrectedFiling and FilingMessage within RecordDocketingMessage). The first option is most similar to ECF4 and therefore may be easier for implementers to understand.

If implemented as option 1 (most similar to ECF4), then the CorrectedFiling element need not contain filing:FilingConnectedDocument and filing:FilingLeadDocument as these pre-review elements are available in filing:FilingMessage.

Let’s discuss these 2 options with the TC.

1. **More on ID/IDREF**

This item continues the concern raised in the prior feedback document as item #4 ID/IDREF Reference Hell.

Thank you for the reference to section 12.2 in NIEM 3 Naming and Design Rules. There are several rules contained within this section.

Rule 12-2 Element with structures:ref does not have content – An element that has attribute structures:ref MUST Not have element or text content.

This rule appears to only allow the stuctures:ref attribute to be used on elements that do not have any content; example:

<nc:DocumentSubmitter>

<nc:EntityPerson structures:ref="Person2" xsi:nil="true"/>

</nc:DocumentSubmitter>

This rule alone would eliminate a lot of the possible misuse illustrated previously.

Rule 12-3 Attribute structures:ref must reference strcutures:id – The value of an attribute structures:ref MUST match the value of an attribute structures:id of some element in the XML document.

No surprises here; hopefully this is a well understood rule.

Rule 12-4 Linked elements have same validation root – Every element that has attribute structures:ref MUST have a referencing element validation root that is equal to the referenced element validation root.

This rule is not so easy to understand as stated.

First, the term ‘validation root’ must be understood. ‘Validation root’ is defined as: “This item, that is the element information item at which –assessment- began, is called the validation root.”

Rule 12-5 Attribute structures:ref references element of correct type – Every element that has an attribute structures:ref MUST have a referencing element type definition that is validly derived from the referenced element type definition.

“This rule requires that the type of the element information item pointed to by a structures:ref attribute must be of (or derived from) the type that is specified by the element declaration of the reference element.”

Rule 12-5 appears to be the same or similar to the rule that I suggested:

“When using structures:ref attribute, the element containing the destination structures:id attribute must be of the same element type (e.g. element nc:Person could contain a structures:ref that pointed to another nc:Person element or could also point to an nc:RoleOfPerson element since both nc:Person and nc:RoleOfPerson are of type nc:PersonType). Note – if this rule were adopted, then ecf:PayerNameText in PaymentMessage would need to be revised to be nc:PersonType.”

If my interpretation of 12-5 is correct, then ecf:PayerNameText would need to be revised to be of nc:PersonType.

Furthermore, corrections may need to also be made to examples provided. For instance, in the civil.xml example, contains:

<nc:DocumentSubmitter>

<nc:EntityPerson structures:ref="Person2" xsi:nil="true"/>

</nc:DocumentSubmitter>

And also contains (abbreviated):

<j:CaseRespondentParty structures:id="Person2">

<nc:EntityPerson>

<nc:PersonName>

<nc:PersonGivenName>Jane</nc:PersonGivenName>

<nc:PersonMiddleName>Q</nc:PersonMiddleName>

<nc:PersonSurName>Doe</nc:PersonSurName>

<nc:PersonMaidenName>Smith</nc:PersonMaidenName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Defendant</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>10</nc:IdentificationID>

</ecf:FilingPartyID>

<nc:ContactInformationAssociation>

</ecf:PersonAugmentation>

</nc:EntityPerson>

</j:CaseRespondentParty>

This is a violation of rule 12-5. Element j:CaseRespondentParty is type nc:EntityType, whereas nc:EntityPerson is type nc:PersonType. Therefore, per Rule 12-5, nc:EntityPerson cannot have a structures:ref attribute value that matches the j:CaseRespondentParty structures:id attribute value.

This example could be corrected by moving the structures:id attribute from j:CaseRespondentparty to nc:EntityPerson as shown below:

<j:CaseRespondentParty>

<nc:EntityPerson structures:id="Person2">

<nc:PersonName>

<nc:PersonGivenName>Jane</nc:PersonGivenName>

<nc:PersonMiddleName>Q</nc:PersonMiddleName>

<nc:PersonSurName>Doe</nc:PersonSurName>

<nc:PersonMaidenName>Smith</nc:PersonMaidenName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Defendant</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>10</nc:IdentificationID>

</ecf:FilingPartyID>

<nc:ContactInformationAssociation>

</ecf:PersonAugmentation>

</nc:EntityPerson>

</j:CaseRespondentParty>

This mistake occurred in multiple examples and has been corrected in each.

These NIEM rules do not however prevent all misuses of ID/IDREF. We may still want to impose our own additional ECF rules such as:

* Specify that circular references are not permitted (this would also require defining exactly what it means to be circular).
* Define in the specification, standardized id/ref relationships and what they mean. For example, when an element of nc:PersonType references another element of nc:PersonType (using structures:ref), then this means the very same person is being referred to..
* Provide guidance on non-specification standardized id/ref relationships.
* Require non-specification standardized id/ref relationships to be defined in court policy.

If we agree to provide guidance on use of IDs and IDREFs above and beyond the NDR, we’ll need someone to draft the language carefully to ensure it doesn’t conflict with the NDR. Let’s discuss with the TC.

1. **Connected Documents**

I had previously raised issues regarding a normative approach to identifying and associating lead and connected documents. This resulted in a revision to section 6.3.1 filing:FilingMessage:

A [filing:FilingMessage](file:///C:\Users\JamesECabral\OneDrive\xml\ecf5\schema\filing.xsd) MAY NOT include documents for transactions such as the payment of a criminal fine. If a [filing:FilingMessage](file:///C:\Users\JamesECabral\OneDrive\xml\ecf5\schema\filing.xsd) includes documents, the lead documents MUST be included in filing:FilingLeadDocument elements and the message MUST include only one level of connected and supporting documents in filing:FilingConnectedDocument elements and referenced in filing:FilingLeadDocument with the ecf:ConnectedDocument element. The following non-normative example includes a single lead document and single connected document:

<filing:FilingMessage>

<filing:FilingConnectedDocument structures:id=”Document2”>

…

</filing:FilingConnectedDocument>

<filing:FilingLeadDocument structures:id=”Document1”>

…

<ecf:ConnectedDocument structures:ref=”Document2”/>

…

</filing:FilingLeadDocument>

…

</filing:FilingMessage>

After reviewing the FilingMessage example in wd09 (i.e. appellate.xml, civil.xml, citation.xml, criminal.xml, domestic.xml, and juvenile.xml), none of these examples is employing ecf:ConnectedDocument within filing:FilingLeadDocument to identify connected documents. Most, but not all (appellate.xml is the exception here) use nc:DocumentAssociation within filing:FilingConnectedDocument to reference the parent FilingLeadDocument (appellate.xml is just plain wrong either way).

Was the adjustment made previously (and shown above) an ‘overcorrection’ in that nc:DocumentAssociation within filing:FilingConnectedDocument should be used and not ecf:ConnectedDocument in filing:FilingLeadDocument?

Note: the niem-mapping.csv spreadsheet ‘suggests’ that nc:DocumentAssociation (line 437) is the correct replacement.

You are correct. I corrected 6.3.1 to reference nc:DocumentAssociation rather than ecf:ConnectedDocument (including the non-normative example) and updated appellate.xsd.

1. **Document Associations**

Assuming that the answer to item 3 above is that an ‘overcorrection’ had been made:

The manner in which documents are associated, particularly the way that a Filing Connected Document is associated with its parent Filing Lead Document is changed in ECF5.

ECF4: the s:ref attribute on the ecf:ParentDocumentReference element within FilingConnectedDocument identifies the parent FilingLeadDocument.

<ecf:DocumentMetadata>

<j:RegisterActionDescriptionText/>

<ecf:ParentDocumentReference s:ref="\_123456ABC"/>

<ecf:FilingAttorneyID/>

<ecf:FilingPartyID/>

</ecf:DocumentMetadata>

ECF5: the structures:ref attribute on the nc:PrimaryDocument element along with the nc:AssociationDescriptionText within filing:FilingConnectedDocument identifies the FilingLeadDocument.

<nc:DocumentAssociation>

<nc:AssociationDescriptionText>parent</nc:AssociationDescriptionText>

<nc:PrimaryDocument structures:ref="Document1" xsi:nil="true"/>

</nc:DocumentAssociation>

In ECF4 the fact that the FilingLeadDocument is the ‘parent’ of the FilingConnectedDocument is connoted by ecf:ParentDocumentReference (context is ‘built-into’ the element name and definition). In ECF5 the nature of the association must be expressed in nc:AssociationDescriptionText. As such, and for interoperability, it seems that a standardized set of allowable values should be specified for nc:AssociationDescriptionText. Is “parent” the only allowable value? Should there be a code list? Should nc:AssociationDescriptionText be required (i.e. minOccurs = 1)?

Actually, the instances should be using ecf:DocumentAssociationAugmentation/ecf:DocumentRelatedCode instead of nc:AssociationDescriptionText. I changed the cardinality of both ecf:DocumentAssociationAugmentation and ecf:DocumentRelatedCode to 1,1 and fixed the instances.

1. xxx