**ECF5 Spec Feedback and Considerations – 2**

Jim Cabral’s responses in red

This document contains additional questions and commentary resulting from a review at the Electronic Court Filing Version 5.0 Working Draft 07.

1. **FilingConnectedDocument v. ConnectedDocument v. DocumentAssociation**

ecf:ConnectedDocument is now available in DocumentAugumentation.

The definition of ecf:ConnectedDocument is the same as the definition of filing:FilingConnectedDocument: “A document included in a Filing that supports the Document. This document is not separately entered on the dockets or register of actions.”

Also, nc:DocumentAssociation is provided (“An association between documents.”). nc:DocumentAssociation permits the defining of an nc:PrimaryDocument and nc:SecondaryDocument.

Why are there two or more different ways to identify connected documents and which should implementers use?

Section 6.3.1 filing:FilingMessage, in the second paragraph does mention connected and supporting documents with: “the message MUST include only one level of connected and supporting documents”. However, it does not clarify how this is to be done. The non-normative example uses filing:FilingConnectedDocument. Also, section 4.4 Attachments also implies the use of FilingLeadDocument and FilingConnectedDocument in the non-normative example. Additionally, section 6.2.4 DocumentIdentifers also implies the use of FilingConnectedDocument in its non-normative example. The example in D.1 Asynchronous operation input message also implies the use of FilingConnectedDocument and FilingLeadDocument (again this example is “informative” and not “normative”). The use of FilingLeadDocument is implied in that this element’s minOccurs is 1.

I suggest that we clearly state the rules, expectations, and standards in the main specification. We should make it simple and straightforward to figure out what to do and what not to do to be conformant. As such, we could add text to section 6.3.1 filing:FilingMessage (and also 6.3.3 docket:RecordDocketingMesage perhaps), such as:

Every submission MUST have at least one lead document which is described and referenced through <filing:FilingLeadDocument>. When a submission also includes connected documents, these MUST be described and referenced through <filing:FilingConnectedDocument>. Each connected document MUST include a <nc:DocumentAssociation> element that contains an <nc:PrimaryDocument> element that includes a structures:ref attribute that references the connected document’s lead document’s structures:id attribute value. Each connected document MUST also include the <nc:AssociationDescriptionText> element within <nc:DocumentAssociation> and this element MUST have the value ‘parent’. For both lead and connected documents, <ecf:ConnectedDocument> MUST NOT be used.

Example (non-normative)

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

<nc:DocumentCategoryText>Information</nc:DocumentCategoryText>

…

<nc:DocumentIdentification>

<nc:IdentificationID>1.1</nc:IdentificationID>

</nc:DocumentIdentification>

…

<ecf:DocumentAugmentation>

…

<nc:DocumentAssociation>

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

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

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

</filing:FilingConnectedDocument>

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

<nc:DocumentCategoryText>Apperance</nc:DocumentCategoryText>

…

<nc:DocumentIdentification>

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

</nc:DocumentIdentification>

…

</filing:FilingLeadDocument>

It would be preferred that elements which MUST not be used (e.g. (ecf:ConnectedDocument> ) are not included in schema (e.g. are absent from <DocumentAugmentation>).

Lead and connected documents should be included in filings in filing:FilingMessage/filing:FilingLeadDocument and filing:FilingMessage/filing:FilingConnectedDocument. ecf:ConnectedDocument is included as a way to REFERENCE a connected document from a court event (in j:CaseCourtEvent/ecf:CourtEventAugmentation/ecf:ConnectedDocument) or from a lead document (in nc:Document/ecf:DocumentAugmentation/ecf:ConnectedDocument).

To clarify, I propose the following changes and additional non-normative example to the relevant paragraph in Section 6.3.1:

A [filing:FilingMessage](file:///C:\Users\JamesECabral\AppData\Roaming\Microsoft\Word\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\AppData\Roaming\Microsoft\Word\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>

Should something be said about the use or non-use of <nc:SecondaryDocument>?

nc:DocumentAssociation is for relating a document to another document previously filed in the case. In this case, both nc:PrimaryDocument and nc:SecondaryDocument were intended to be REFERENCES. However, since the nc:SecondaryDocument will not be included within the filing, it should not be a REFERENCE. I’ll update the mapping.

1. **FilingPartyID**

What has happened to FilingPartyID? ECF 4 provided FilingPartyID in addition to FilingAttorneyID. ECF 5 only provided FilingAttorneyID.

The element ecf:AffectedPartyID is new in ECF 5; ecf:AffectedPartyID is defined as “Identifier recognized by the court as being unique within this case, and used to identify a party other than the filer who is affected by the document”.

Since the definition for ecf:AffectedPartyID ia very different than the definition for FilingPartyID (i.e. “ID recognized by the court as being unique within this case, and used to identify the party on whose behalf the document is being filed”), ecf:AffectedPartyID cannot be replacing FilingPartyID.

Note that section 6.3.1 filing:FilingMesage states: A filing:FilingMessage MUST express the name or names of the party or parties on whose behalf a document is filed, and the party whose document is the subject of a responsive document being submitted for filing.

For better conformance with the NIEM Naming and Design Rules, ecf:FilingPartyID is renamed ecf:FilerIdentification in ECF 5. However, I now see that there are other NIEM and ECF elements of type IdentificationType that end with “ID” (e.g. ecf:FilingAttorneyID, nc:CaseTrackingID). Therefore, I’ll update the mapping to use ecf:FilingPartyID again.

1. **Typo – 6.2.8 Filer and Party Identifiers** - Identifiers for filers and parties to a case, including person, sorganizations and property, MUST be unique within a case and will be generated by the court in response to a ReviewFiling operation. The following is a non-normative example of an identifier for filer number 100:

Fixed

1. **Typo** - Section 6.3.4 serveprocess:ServeProcessMesssage – typo, “Altherativel” should be “Alternatively”.

Fixed

1. **Policyresponse.xml error:** Missing ‘S’ in highlighted text.

<policyresponse:CodeListExtension>

<nc:DocumentIdentification>

<nc:IdentificationID>ignatureProfileCode</nc:IdentificationID>

</nc:DocumentIdentification>

<policyresponse:ExtensionCanonicalURI>https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/SignatureProfileCode</policyresponse:ExtensionCanonicalURI>

<policyresponse:ExtensionCanonicalVersionURI>https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/SignatureProfileCode/2017-02-04</policyresponse:ExtensionCanonicalVersionURI>

<policyresponse:ExtensionLocationURI>https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/SignatureProfileCode</policyresponse:ExtensionLocationURI>

</policyresponse:CodeListExtension>

Fixed

1. **MIME Type**

Many examples include:

<cbrn:MIMEContentCode>pdf</cbrn:MIMEContentCode>

Or:

<nc:BinaryFormatText>pdf</nc:BinaryFormatText>

Note however that ‘pdf’ is not a valid MIME type. The correct MIME type is:

application/pdf

Certainly cbrn:MIMEContentCode (“A MIME content type of a data file”) should only contain a valid MIME type code as published by the Internet Assigned Numbers Authority (IANA.org). The alternative may be to only use codes valid in cbrncl:MIMEContentCodeSimpleType (in which the ‘code’ ‘pdf’ has the definition ‘application/pdf’). Howver, one serious short coming of this code table is that it does not contain all or even enough MIME types. Of the three we allow in Arizona, only one is listed in this table (i.e. application/pdf). We also allow filers to submit docx and odt files (MIME types application/vnd.openxmlformats-officedocument.wordprocessingml.document, and application/vnd.oasis.opendocument.text respectively).

nc:BinaryFormatText (“A file format or content type of a binary file.”) does not appear to have any suggested or required code list in NIEM. However, for interoperability, this should be a standardized value such as a MIME type.

cbrn:MIMEContentCode is a complex type which gets its values from cbrncl:MIMEContentCodeSimpleType. However, you are correct that cbrncl:MIMEContentCodeSimpleType appears to be outdated. I propose we just use nc:BinaryFormatText instead and use the default values from the ECF 4 code list BinaryDescriptionText.gc in the new ECF 5 code list BinaryFormatText.gc and add “text/html” and “application/json”. Are there any MIME codes from https://www.iana.org/assignments/media-types/media-types.xhtml that we should add?

1. **Mandatory Augmentation?**

Section 5.2.1 Court-Specific Augmentations states:

“Court-specific augmentations MUST extend one of the following ECF or NIEM messages or augmentable elements by substituting for the associated augmentation point.”

Is this a mistake or are implementers required to extend (through augmentation) at least one message or element?

It does not seem quite right that an implementation should not be required to extend, but it seems instead it should state that if an implementation extends the schema, it must be done using augmentations.

The one possible exception to mandatory court-specific augmentation appears to perhaps be the use of nc:CaseAugmentationPoint to substitute a case-type specific case (e.g. CivilCase, CriminalCase, etc.). However this appears to be optional as well; see section 4.1 Messages, bullet 3 (beneath the table) which states: “Optionally, one or more case type augmentations, as defined in Section 4.2, that include information appropriate to a filing in a specific case type.”

I changed that paragraph to:

Court-specific augmentations MAY extend any of the following ECF or NIEM messages or augmentable elements by substituting a court-specific element for the associated augmentation point.

Note: it’s still not clear why ChildSupportEnforcementCase is implemented by substitution for nc:Case (ecf-4 style), whereas appellate case, bankruptcy case, citation case, civil case, criminal case domestic case, and juvenile case are provided by substitution for nc:CaseAugmentationPoint.

In ECF 5, cyfs:ChildSupportEnforcementCase is contained within domestic:CaseAugmentation. While the NIEM allows it to substitute for nc:Case, it is not intended to be used that way in ECF 5.

1. **Court Policy MDE – optional or mandatory?**

Section 3.2.1 The Filing and Service Process lists the GetPolicy operation as optional. Since the GetPolicy operation is optional, doesn’t this make the Court Policy MDE optional as well?

Or is the Court Policy MDE not considered optional, because section 5 Court Policy, requires (i.e. ‘MUST have’) both human-readable and machine-readable court policies (even though machine-readable policy need not be provided by GetCourtPolicy and may be provided by ‘some other means’)?

The Court Policy MDE is optional. Section 3.2 includes the following:

An ECF 5.0-conformant implementation may implement one or more of the MDEs defined in the specification but a complete ECF 5.0 system MUST include at least one each of the Filing Assembly, Filing Review and Court Record MDEs.

1. **GetPolicy variations?**

The GetPolicy request example provided (i.e. policyrequest.xml) includes both a j:CourtName value and an ecf:CaseTypeCode value. This implies that the policy request is only for criminal case policy from ‘King County Superior Court’. There are two possible locations in the specification where one might expect to find information that may clarify this question, specifically section 5 Court Policy, or in the Business Rules section (i.e. section 6) under subsection 6.1.1 GetPolicy. However, the presumed behavior is not described in either section.

Is this ‘silence’ by design and the usage pattern implied in the example XML only just a suggestion or hint? Or is there a behavior that implementers can rely upon (e.g. interoperate)?

I changed 6.1.1 as follows:

An MDE (typically, a Filing Assembly MDE) MAY obtain a court’s machine-readable court policy, optionally filtered by specific case type, at any time by invoking the GetPolicy operation on the Filing Review MDE with the identifier for the court. The Court Policy MDE returns the machine-readable court policy in a synchronous response. The contents of machine-readable court policy is described in Section 5.2. This step may be omitted if the requesting MDE already has the current court policy.

1. Identifiers (section 6.2)

All Identifiers addressed in section 6.2 are accompanied with non-normative illustrative example. This is good. However, only one of these, specifically 6.2.1 Attachment Identifier, can be understood to define which element is the normative element to contain the identifier:

“Attachment identifiers (nc:BinaryURI) MUST be unique within a message transmission.”

I think the above is reasonably read to understand that it is normative to use nc:BinaryURI as the ‘Attachment Identifier’.

This same level of clarity/specificity is not achieved with the other identifiers. Through examples, each ‘suggests’ an element that may be appropriate, but does not require the use of the suggested element.

For example, I have always thought that it was well understood, and universally accepted that nc:CaseTrackingID was the element to use for Case Identifiers. The case identifier was typically understood to be a case number. However, since this is not normative, it would be equally compliant to place the case number into some other element, such as nc:ActivityIdentification/nc:IdentificationID.

To promote interoperability, I believe the very element that should be used for any given identifier should be established in a normative manner.

I updated each identifier definition to clearly indicate the label. E.g.

Attachment identifiers, labeled by nc:BinaryURI, MUST be unique within a message transmission.

1. Section 6.2.8 Filer and Party Identifiers, states:

“Attorney elements MAY reference the parties they represent with party identifiers.”

The term “attorney elements” is understood to be one of the several elements that are used to define an attorney, such as j:CaseOfficial, j:CaseProsecutionAttorney, j:CaseRespondentAttorney, or j:CaseInitiatingAttorney.

To describe attorney party representation relationships, ecf:CaseRepresentedParty elements are available (within CaseOfficialAugmentation). Ecf:CaseRepresentedParty provides nc:EntityRepresentation to identify the party through substitution with EntityOrganization or EntityPerson.

EntityPerson does provide access to the ‘Filer/Party Identifier’ (e.g. the non-normative acf:FilerIdentification/nc:IdentificationID) through EntityPerson/PersonAugmentation/ecf:FilerIdentification/cn:IdentificationID. However, EntityOrganization does not provide access to ecf:FilerIdentification (not even from within OrganizationAugmentation).

I updated the mapping to include ecf:FilingPartyID in ecf:OrganizationAugmentation.

Property type parties cannot even be associated with attorneys through ecf:CaseRepresentedParty because EntityItem is not provided as a substitute for nc:EntityRepresentation.

In NIEM 3, nc:Entity can be represented as nc:EntityPersons or nc:EntityOrganization - nc:EntityItem does not exist. I propose that Property parties be represented as Organizations in ECF 5. Alternatively, we could define ecf:EntityItem to substitute for nc:EntityRepresentation.

1. What’s really normative?

The description of the ‘first’ CaseAugmentation element in nc:Case which is permitted as an nc:CaseAugmentationPoint substitute is:

“Information needed to initiate a court case. The presence of caseTrackingIdentifer is the indicator of whether this is an existing case (present) or new case (absent). For existing cases, caseTrackingIdentifer and shortCaseTitle are mandatory; caseTypeCode is not allowed. For new cases, shortCaseTitle and caseTypeCode are mandatory; caseTrackingIdentifier is not allowed.”

Is this language normative?

If so, where does one look for all the normative parts of the specification?

Note that ecf:CaseShortTitleText is not minOccurs=1.

Also note that the description for ecf:ShortTitletext is inconsistent with CaseAugmentation above. The ecf:CaseShortTitleText includes:

“No title exists when the message is initiating a new case”, whereas above is says “for new cases, shortCaseTitle and caseTypeCode are mandatory.”

I updated the definition to remove the normative detail. The new definition is

Information needed to initiate a court case.

1. xxx