**ECF5 Spec Feedback and Considerations – 13**

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 17.

1. **ReviewedLeadDocument and ReviewedConnectedDocument Corrections**

In ‘ECF 5 Spec Considerations – 10’, item 4 ‘Corrected Filing’, discusses at length, issues with correcting information, either case or document ,in clerk review.

For documents corrections, it pointed out that some elements that may be used to record what happened during clerk review (such as who, what, when, etc.) may also require correction from that provided in the Filing -Lead/Connected Document.

Working Draft 17 addresses this problem by adding two new Document Review elements; i.e. ecf:DocumentReviewer and ecf:DocumentReviewStatus. These two new elements adequately accommodate the need to record who, what and when information for clerk review.

The addition of these new elements requires modifications to the docket.xml example. The following revisions were made:

* ecf:ReviewedConnectedDocument/nc:DocumentSubmitter (lies 165 to 172) were removed, and the reviewing clerk information was placed into a newly added ecf:DocumentReviewer element.
* lines 159 to 164 (ecf:ReviewedConnectedDocument/nc:DocumentStatus) were removed – information moved to newly added ecf:DocumentReviewStatus element.
* Similar changes were also made to ecf:ReviewedLeadDocument.
* Various comment revisions for clarity.
* Added ecf:CaseRepresentedParty to j:CaseInitiatingAttorney to show plaintiff John W. Doe is represented by attorney Jane Doe, JD.

A revised docket.xml example is provided (as my-docket.xml).

This example replaced the previous example.

1. **Content Reference example**

Section 6.2.11.1 ‘Element Content References’ contains the following example:

<j:CaseInitiatingParty>

<nc:EntityPerson structures:id="Person1">

<nc:PersonName>

<nc:PersonGivenName>John</nc:PersonGivenName>

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

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Plaintiff</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

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

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</nc:EntityPerson>

</j:CaseInitiatingParty>

<j:CaseInitiatingAttorney>

<nc:RoleOfPerson structures:id="Person3">

<nc:PersonName>

<nc:PersonGivenName>Jack</nc:PersonGivenName>

<nc:PersonSurName>Jones</nc:PersonSurName>

</nc:PersonName>

</nc:RoleOfPerson>

<j:JudicialOfficialBarMembership>

<j:JudicialOfficialBarIdentification>

<nc:IdentificationID>100001</nc:IdentificationID>

</j:JudicialOfficialBarIdentification>

</j:JudicialOfficialBarMembership>

<ecf:CaseOfficialAugmentation>

<ecf:CaseRepresentedParty>

<nc:EntityPerson>

<ecf:PersonAugmentation>

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

<ecf:FilingPartyID>

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

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</nc:EntityPerson>

This example shows that plaintiff John Doe is represented by attorney Jack Jones. Even though the point of the example is to illustrate a content reference, I think it adds unnecessary confusion to label Jack Jones as both a ‘plaintiff’ and a ‘defendant’. It also begs the question, what is the purpose of ecf:CaseParticipantRoleCode in ecf:CaseRepresentedParty?

Updated the example to use “Plaintiff” in both cases.

1. **Normative Attorney to Party Association Method**

At the June 13, 2017 ECF Conference call it was agreed that although there are several ways that attorney/party representation relationships can be expressed or implied, a normative approach should be defined. It was also agreed that the normative approach was to use ecf:CaseRepresentedParty in an Attorney/CaseOfficial element. This approach was similar to that illustrated in ecf-v5.0-wd16.docx, section 6.2.11.1. It was also noted during the call that this example should be corrected to place ecf:CaseRepresentedParty into j:CaseInitiatingAttorney, as it is not necessary or even desirable to use both j:CaseInitiatingParty and j:CaseOfficial.

Whereas the example in section 6.2.11.1 has been revised and is structurally correct and consistent with the agreed changes from the 6-13-2017 conference call, this example does not define the illustrated approach for attorney/party representation relationships as ‘normative’. The specification does not state that the example is normative nor is it intended as an example of attorney/party representation relationship, as it is an example of Content References and Element References (although it can serve both illustration functions).

I continue to feel strongly that using ‘Reference Elements’ within ecf:CaseRepresentedParty instead of ‘Content References’ is a superior approach (see ECF5 Spec Considerations – 7.docx, item 1 ‘Attorney to party Association’).

To allow flexibility, the normative approach should permit implementers to choose to employ either ‘content references’ or reference elements’, when referring to the ecf:CaseRepresentedParty.

Again, I am providing suggested text for the specification document. This suggestion is from ECF5 Spec Considerations – 7.docx, with some updated revisions:

**Attorney Party Relationships**

The relationship of a case party or parties to the attorney or attorneys representing the party or parties in a law suit MUST be done using a j:CaseOfficialType element (e.g. j:CaseDefenseAttorney, j:CaseInitiatingAttorney, j:CaseProsecutionAttorney, j:CaseRespondentAttorney, j:CaseOtherEntityAttorney, and j:CaseOfficial ) within j:CaseAugmentation.

The attorney in the relationship MUST be identified using nc:RoleOfPerson. When identifying the attorney using nc:RoleOfPerson reference, then two methods are supported. One and only one of these two methods MUST be used. The two methods are:

1. The attorney is identified using the structures:ref attribute on nc:RoleOfPerson. This is the preferred and recommended method.
2. The attorney details are elaborated within nc:RoleOfPerson.

When using the first method, the value of structures:ref MUST correspond to a person type entity with a matching value for structures:id. The nc:RoleOfPerson element MUST not have any sub-element content.

When using the second method, the structures:ref attribute in nc:RoleOfPerspon MUST not be used. The attorney person details MUST be elaborated with sub-elements of nc:RoleOfPerson.

The element j:JudicialOfficialBarMembership and j:JudicialOfficialRegistrationIdentification MAY only be used within j:CaseOfficial, when they have not been used within the attorney person elaboration.

(Not sure what to say about j:CaseOfficialCaseIdentification – seems like the rule above should apply here too).

The attorney SHOULD be provided with an identifier in ecf:CaseOfficialAugmentation/ecf:AttorneyID/nc:IdentificationID.

To identify the parties represented by the attorney, ecf:CaseOfficialAugmentation MUST be substituted for j:CaseOfficialAugmentationPoint. An ecf:CaseRepresentedParty element MUST be used within ecf:CaseOfficialAugmentation to identify each party represented by the attorney. If the attorney represents more than one party on the case, then multiple ecf:CaseRepresentedParty elements should appear. When an attorney represents more than one party on a case, multiple j:CaseOfficialType elements SHOULD not be used as an alternative to using multiple ecf:CaseRepresentedParty elements within a single j:CaseOfficialType element.

The ecf:CaseRepresentedParty element MUST contain a valid entity type element which has been substituted for nc:EntityRepresentation. The appropriate element will depend upon the nature of the case party; whether it is a person party, an organization party, or a property/item party.

When identifying the party represented by the attorney, there are two methods supported. One and only one of these two methods MUST be used. The two methods are:

1. The party is identified using the structures:ref attribute on the substituted entity type element. This is the preferred and recommended method.
2. The party entity details are elaborated within the substituted entity type element (e.g. EntityItem, EntityOrganization or EntityPerson).

When using the first method, the value of structures:ref MUST refer to an entity type element with a corresponding entity type and which has a structures:id attribute with a matching value. The entity type element substituted for nc:EntityRepresentation MUST not have any sub-element content. The element ecf:CaseParticipantRoleCode MUST not be used when using the first method.

When using the second method, the structures:ref attribute in the substituted entity type element MUST not be used. The entity details MUST be elaborated using sub-elements of the entity type element substituted for nc:EntityRepresentation. The ecf:CaseParticipantRoleCode within ecf:CaseOfficialAugmentation MAY only be used if the entity is an item type entity. For organization and person type entities, use the ecf:CaseParticipantRoleCode available within the entity augmentation.

It seems to me, that if a reader were looking in the specification for how to define attorney/party representation relationships, the section on ‘Identifier Rules’ may not be the section you would look to. Perhaps it should be a section on ‘Identifiers and Associations’.

I added the following to 6.2.11.2

The relationship of an attorney to the party being represented SHOULD be defined using a reference element in ecf:CaseOfficialAugmentation/ecf:CaseRepresentedParty.

1. **Multiple CaseParticipantRoleCodes**

At present ECF 4.01 allows an entity to have multiple CaseParticipantRoleCodes.

This was straight forward in ECF 4.0 when cardinalities were generally more liberal. You just listed as many codes as needed as shown in the following example:

<ecf:CaseParticipant>

<ecf:CaseParticipant>

<ecf:EntityPerson s:id="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

</ecf:EntityPerson>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

</ecf:CaseParticipant>

</ecf:CaseParticipant>

In ECF 4.01, the cardinality was limited to only one. However, it was stated that multiple CaseParticipantRoleCodes could still be had by using multiple ecf:CaseParticipant elements. The above ECF 4.0 example must be expressed as:

<ecf:CaseParticipant>

<ecf:CaseParticipant>

<ecf:EntityPerson s:id="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

</ecf:EntityPerson>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

</ecf:CaseParticipant>

</ecf:CaseParticipant>

<ecf:CaseParticipant>

<ecf:CaseParticipant>

<ecf:EntityPersonReference s:ref="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

</ecf:CaseParticipant>

</ecf:CaseParticipant>

In retrospect, the method we were instructed to use in ECF 4.01 (i.e. applying multiple CaseParticipantRoleCodes to an entity through the use of multiple CaseParticipant elements) is not valid in ECF 5. We now know that NIEM 3.2 Rule 12-2 (e.g. element with structures:ref does not have content) would forbid this approach.

In ECF 5, applying multiple ecf:CaseParticipantRoleCodes to an entity is even more obtuse than in ECF 4.01. One approach that can be used when the entity is a case party with attorney representation is illustrated in the example for section 6.2.11.1 ‘Element Content References’ (shown above in item 2).

Other approaches are possible. The Samuel Adams example from ECF 4 above could be done as shown below:

<j:CaseOtherEntity>

<ecf:EntityPerson structures:id="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

<j:CaseOtherEntity>

<ecf:EntityPerson>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

Being forced to use Content References rather than Reference Elements makes this decidedly more difficult to process.

Here’s another ECF 5 approach to accomplish the Samuel Adams example:

<j:CaseOtherEntity>

<ecf:EntityPerson structures:id="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

The multiple cardinality approach in ECF 4.0 is still the most elegant and practical way to achieve multiple CaseParticipantRoleCodes for an entity.

However, the example above is not the only way you can do this in ECF 5!

It could also be done like this:

<j:CaseOtherEntity>

<ecf:EntityPerson structures:id="Person1">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

<j:CaseOtherEntity>

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

<nc:PersonName>

<nc:PersonGivenName>Elizabeth</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Elizabeth Adams</nc:PersonFullName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Wife</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>17</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

…

<nc:PersonAssociation>

<nc:Person>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</nc:Person>

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

<ecf:PersonAssociationAugmentation>

<ecf:EntityAssociationTypeCode>spouse</ecf:EntityAssociationTypeCode>

</ecf:PersonAssociationAugmentation>

</nc:PersonAssociation>

Having so many different ways to do one thing makes it difficult for implementers and does not propmote interoperability.

So for those of you who like puzzles, how many and what CaseParticipantRoleCodes does Samuel Adams have in the ECF 5 XML snippet below?

<j:CaseOtherEntity>

<ecf:EntityPerson structures:id="ia7bcafff-e6f4-400c-9d36-fc46473c199a">

<nc:PersonName>

<nc:PersonGivenName>Samuel</nc:PersonGivenName>

<nc:PersonSurName>Adams</nc:PersonSurName>

<nc:PersonFullName>Samuel Adams</nc:PersonFullName>

</nc:PersonName>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Brewer</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

<j:CaseOtherEntity>

<ecf:EntityPerson>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Patriot</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Delegate</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</ecf:EntityPerson>

</j:CaseOtherEntity>

…

<j:CaseInitiatingAttorney>

<nc:RoleOfPerson structures:id="Person3">

<nc:PersonName>

<nc:PersonFullName>Jane Doe, Esq.</nc:PersonFullName>

</nc:PersonName>

</nc:RoleOfPerson>

<ecf:CaseOfficialAugmentation>

<ecf:AttorneyID>

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

</ecf:AttorneyID>

<ecf:CaseParticipantRoleCode>Attorney</ecf:CaseParticipantRoleCode>

<ecf:CaseRepresentedParty>

<nc:EntityPerson>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>TaxCollector</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Father</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</nc:EntityPerson>

</ecf:CaseRepresentedParty>

</ecf:CaseOfficialAugmentation>

</j:CaseInitiatingAttorney>

…

<nc:PersonAssociation>

<nc:Person>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Revolutionary</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Colonist</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</nc:Person>

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

<ecf:PersonAssociationAugmentation>

<ecf:EntityAssociationTypeCode>spouse</ecf:EntityAssociationTypeCode>

</ecf:PersonAssociationAugmentation>

</nc:PersonAssociation>

…

<payment:Payer>

<ecf:PersonAugmentation>

<ecf:CaseParticipantRoleCode>Bostonian</ecf:CaseParticipantRoleCode>

<ecf:FilingPartyID>

<nc:IdentificationID>16</nc:IdentificationID>

</ecf:FilingPartyID>

</ecf:PersonAugmentation>

</payment:Payer>

The answer is 8 – Brewer, Patriot, Delegate, TaxCollector, Father, Revolutionary, Colonist, and Bostonian.

Hopefully, this little puzzle illustrates how out of control this appears to be! It seems to be neither elegant nor practical to have to search multiple nooks and crannies in the XML to assemble the full participation picture. I think we need to discuss this with the TC.

We can discuss this at the next TC meeting but I just think this is an example of the flexibility of the specification.

1. **CaseParticipantRoleCodes gc**

In the newly added section 6.4 Case Participant Rules. It states:

The [CaseParticipantRoleCode.gc](file:///C:\Users\JamesECabral\OneDrive\xml\ecf5\schema\CaseParticipantRoleCode.gc) code list defines the allowed values for ecf:CaseParticipantRoleCode and includes columns indicating which code values are valid in combination with each role element. If ecf:CaseParticipantRoleCode is provided, the code value MUST be in the [CaseParticipantRoleCode.gc](file:///C:\Users\JamesECabral\OneDrive\xml\ecf5\schema\CaseParticipantRoleCode.gc) code list and the code list column matching the role element MUST have the value “true”.

I am having trouble verifying this.

I can see that the schema folder includes three CaseParticipantRoleCode gc files: CaseParticipantRoleCode.gc, CaseParticipantRoleCode-noroles.gc and CaseParticipantRoleCode-roles.gc.

Of these three gc files, the one that appears to “includes columns indicating which code values are valid in combination with each role element“ is CaseParticipantRoleCode-roles.gc and not CaseParticipantRoleCode gc.

CaseParticipantRoleCode-roles.gc has been copied to CaseParticipantRoleCode.gc. The other version is an artifact of the schema generation toolset we are using for ECF 5 which cannot add the additional columns required for this code list. CaseParticipantRoleCode-roles.gc is a manually edited version that adds the columns. When we regenerate the schemas, we will need to manually replace the auto-generated version with this manually edited version each time.

1. **Appendix F**

At the June 13, 2017 TC conference call, it was agreed to include additional NIEM reference guidance in an appendices. This has been added as Appendix F. It was also requested that this appendix be referred to from within section 6.2.11.2. This has been accommodated with:

Future non-normative guidance regarding the use of references is provided in Appendix F.

But why the word ‘future’?

Isn’t [NIEM NDR] already in effect?

“Future” has been replaced with “Additional”

1. **Payment Message Corrections**

Now that clerk review corrections to case and reviewed document information has been addressed, we still need to discuss how changes to the Payment Message due to clerk review corrections are accommodated.

First, consider that corrections in clerk review can have an impact on fees charged/due. For example, if there is a fee for filing a specific document type, but in clerk review, the clerk determines that the document has been mistyped, and corrects the document type, the corrected type may have no fees, or different fees. This clerk review document type correction will have Payment message implications.

In these cases, the fees and total amount may change but the payment method will not change. Do we still need the version of the Payment message prior to clerk review?

1. **Attorney Identifier**

Section 6.2.8 Filer and Party Identifiers only references ecf:FilingPartyID/nc:IdentificationID and ecf:FilingAttorneyID/nc:IdentificationID. There is no mention of ecf:AttorneyID/nc:IdentificationID.

Added ecf:AttorneyID

1. **Attorney elements vs. Attorney Reference elements**

In Section 6.2.8 Filer and Party Identifiers, immediately beneath the non-normative example, the paragraph begins with:

“Attorney elements, e.g. ecf:FilingAttorneyID and ecf:AttorneyID, …”

To me, Attorney elements are j:CaseDefenseAttorney, j:CaseInitiatingAttorney, j:CaseProsecutionAttorney, j:CaseRespondentAttorney, j:CaseOtherEntityAttorney, and j:CaseOfficial.

I suggest that ecf:FilingAttorneyID and ecf:AttorneyID are instead Attorney Reference elements.

Corrected to “Attorney identifier elements”

1. **CorrectedCase Integrity Rules**

Some integrity rules were first suggested in ‘ECF5 Spec Considerations 6’, item 2, such as:

CorrectedCase must be the same case type and use the same augmentation point type as FilingMessage/nc:Case.

The above example may not be such a good integrity rule, since it seems to say that the case type cannot be changed in clerk review.

However, I do think that how to use and how to understand the usage of CorrectedCase should be more specifically described in the specification. Presently it only gets the following mention in Section 6.3.3:

If the clerk made any modifications to the original filing case information, then the modified case information SHOULD be included in the docket:CorrectedCase element.

The specification does not go on to say, or even suggest, how this should be done. It may be more challenging than simply stated.

Whereas the docket.xml example may provide some additional (non-normative) guidance, this still leaves a lot of room for differing interpretations and approaches. This derides from interoperability.

Writing up this additional specification guidance will not be easy. So if the TC agrees that this should be done, then it will need to be assigned to someone (perhaps me) to draft the write-up for review. Many elements of this guidance may already be included in ‘ECF5 Spec Considerations – 10’, item 4 Corrected Filing.

I’m not sure this detail is warranted. I think providing the non-normative example (docket.xml) should be sufficient.

1. **Corrected civil-ReviewFilingRequest-01.xml**

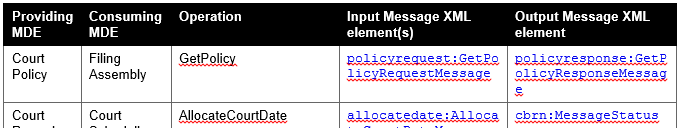
I corrected this example by removing the sub-elements from beneath payment:Payer. This sub-element content would violate NIEM Rule 12-2 since payment:Payer has the structures:ref attribute. Correction provided as an attachment.

Replaced the example.

1. **Policy Response**

Section 6.1.1 GetPolicy, states “an MDE (typically, a Filing Assembly MDE) MAY obtain a court’s machine-readable court policy, …”.

The table in section 4.1 Message lists the Filing Assembly MDE exclusively as a GetPolicy operation consumer.



How should the table in section 4.1 be interpreted? Does this table only list allowed MDE Provider/Operation/Consumers triads or only ‘typical’ triads?

The policy response example (policyresponse.xml) contains:

<policyresponse:MajorDesignElementTypeCode>CourtRecord</policyresponse:MajorDesignElementTypeCode>

<policyresponse:SupportedOperations>

<policyresponse:OperationNameCode>AllocateCourtDate</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>DocumentStampInformation</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>GetCase</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>GetCaseList</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>GetDocument</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>GetServiceInformation</policyresponse:OperationNameCode>

<policyresponse:OperationNameCode>RecordDocketing</policyresponse:OperationNameCode>

</policyresponse:SupportedOperations>

If the FAMDE is the only MDE recipient of policyresponse:GetPolicyResponseMessage (see table in Section 4.1 Messages), is there any reason to list operations that are not consumed by the FAMDE, such as AllocateCourtDate (consumed by CSMDE), DocumentStampInformation (consumed by FRMDE), and RecordDocketing (consumed by FRMDE)?

Or is it now true in ECF5 that MDEs other than FAMDE will utilize GetPolicy to see if operations are supported, such as the FRMDE using GetPolicy to see if DocumentStampInformation supported? If so, would this be ‘typical’?

If the new normal is that GetPolicy is typically used by MDEs other than FA, then the table in section 4.1 may need some additional GetPolicy triad entries.

Updated the preface before the table in 4.1 to say:

The following table lists each ECF 5.0 operation, the MDEs that MUST provide and MUST consume the operation, and the input and output XML messages that define the data content exchanged. Other MDEs MAY also consume the operation.

1. **Just saying**

It seems to me, that if you (e.g. FAMDE) receive a policyresponse:GetPolicyResponseMessage that does not contain:

<policyresponse:MajorDesignElement>

<policyresponse:MajorDesignElementLocationID>

<nc:IdentificationID>https://efilingmanager.com:8000</nc:IdentificationID>

</policyresponse:MajorDesignElementLocationID>

<policyresponse:MajorDesignElementTypeCode>CourtPolicy</policyresponse:MajorDesignElementTypeCode>

<policyresponse:SupportedOperations>

<policyresponse:OperationNameCode>GetPolicy</policyresponse:OperationNameCode>

</policyresponse:SupportedOperations>

…

Then something is very wrong!

The specification makes it clear which operations are required. The policy needs to conform with the specification.

1. **ServeFiling optional**

The ServeFiling operation is optional, however the sequence diagram in section 3.2.1 shows otherwise.

Updated the diagram

1. **Policy Requester Filer ID**

The GetPolicyRequest message contains a mandatory ecf:FilingPartyID (“A unique identifier for a person or organization that requests information.”).

Of course the policy requesting entity may not be a case party; it could be an attorney, or some other participant. So perhaps this element should be ecf:RequesterID instead.

By whatever element name, would this element be considered a ‘Filer or Party’ Identifier as referenced in section 6.2.8 Filer and Party Identifiers?

If so, does the requirement that the identifier “MUST be unique within a case and will be generated by the court in response to a ReviewFiling operation” make any sense or even apply?

If ‘generated by the court in response to a ReviewFiling operation’, then would it not be unreasonable to expect that the very first time an entity accessed an ECF compliant efiling implementation, with the intent to submit an efiling or otherwise, and began the process by requesting the Court Policy for this efiling implementation, that the requester’s identifier element is empty?

I suspect that portions of your (J. Cabral) response in ‘ECF5 Spec Consideration – 6’, item 7 is relevant here too. Particularly number 4. “Filing party and attorney IDs are unique within the e-filing system. In practice, these will need to be assigned by an identity management system (not the EFSPs) that is outside the scope of the ECF specification. If there is a single EFM, the EFM will likely be the source of identity for filers and service recipients.”

Revised 6.2.8 to

Identifiers for filers and parties to a case, including person, organizations and property, labeled as ecf:FilingPartyID/nc:IdentificationID, ecf:AttorneyID/nc:IdentificationID, or ecf:FilingAttorneyID/nc:IdentificationID, MUST be unique within a case.

1. **Lest We Forget**

As referred to above, a lengthy response was provided to an even lengthier item 7 in ‘ECF5 Spec Consideration – 6’. The suggestions in the response are still pending. The full response is repeated below:

There is a lot to unpack here. It seems like we need to start with some assumptions that should probably be in the specification. Here are my suggestions:

1. We need common definitions for “filer”, “party” (or, if you prefer “participant”), “attorney”, “service recipient” and possibly “submitter” that we use consistently throughout the specification.
2. Every filing is associated with a filing party and, unless they are filing pro se, a filing attorney. There are no “litigant identifiers”.
3. The filing party and attorney are both legally responsible for submitting the filing regardless if someone else submits the filing on their behalf.
4. Filing party and attorney IDs are unique within the e-filing system. In practice, these will need to be assigned by an identity management system (not the EFSPs) that is outside the scope of the ECF specification. If there is a single EFM, the EFM will likely be the source of identity for filers and service recipients.

We should discuss and agree on assumptions with the TC. Based on your feedback, I expect some dissent. Once we agree to the assumptions, we should consider the following changes to the specification.

1. Rewrite 6.2.8 completely to provide the definitions and clarify the assumptions.
2. In the schema, provide a reference from the filing party and attorney IDs to the entities (usually person objects), rather than reference from one ID to another ID as you suggested.
3. Improve the non-normative examples to more clearly demonstrate how the specification is intended to be used.

This issue was discussed with the task team on May 23 and the changes listed above were already made to the specification.

1. **Judge ID**

A ‘Filer’ is defined in section 1.1 ‘Terminology’ as:

An attorney, judicial official or a *pro se* (self-represented) litigant acting as an individual who assembles and submits one or more filings (combinations of data and documents).

The term ‘judicial officer’ is understood to include judges, such as the case judge.

Section 6.2.8 Filer and Party Identifiers specifies “Identifiers for filers and parties to a case“ and specifically prescribes ecf:FilingPartyID and ecf:FilingAttorneyID. The case judge is not a party on the case, and not all judges are licensed to practice law (e.g. are not all attorneys). So when a case judges files on the case (such as an order), what ‘Filer’ ID should be used?

Perhaps instead of coining a new ecf:JudgeID or ecf:FilingJudgeID element, we can eliminate ecf:FilingPartyID, ecf:FilingAttorneyID, and ecf:AttorneyID with just a single element such as ecf:FilingEntityID or even just ecf:EntityID.

The judge is obviously not represented by an attorney and should just use ecf:FilingPartyID. We could discuss your idea of consolidating these IDs on a future TC call.

1. **xxx**