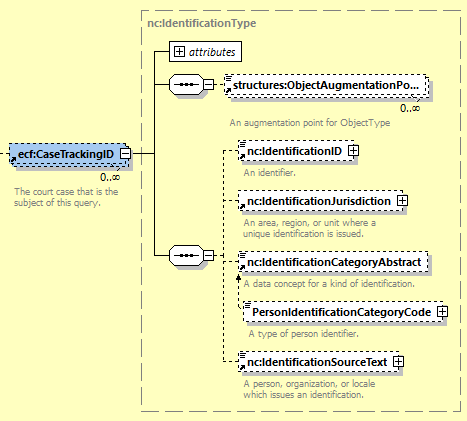
**ECF5 Spec Feedback and Considerations – 22**

This document provides additional ECF-5 feedback, questions, and commentary. This feedback is based on review of the Electronic Court Filing Version 5.0 Working Draft 28 (WD28), unless otherwise noted. This document does not represent a comprehensive review of WD28 but instead only raises issues uncovered during a limited review time interval. Additional review of WD28, or other future work draft(s), is anticipated as additional time permits.

1. **CaseTrackingID**

The revisions, in section 6.2.2 Case Identifiers. appear to be correctly capture the agreements from the recent conference call (02-13-2018). The narrative and the example both reflect the newly included element nc:IdentificationCategoryDescriptionText.

Unfortunately, this element is not available in ecf:CaseTrackingID (as shown below):



1. **CaseTrackingID**

ecf:CaseTrackingID should not be mandatory in GetCaseRequestMessage or in GetDocumentRequestMessage or in GetFilingListRequestMessage.

When requesting case information (or case documents), a requester may identify the case by case number (i.e. j:CaseNumberText) or by CaseTrackingID. A requester should not be required to know and provide both a case number and a case tracking ID. If both are provided then they MUST specify the same case.

1. **Cardinality for j:CaseLineageCase**

In ECF 4.01, the cardinality for j:CaseLineageCase, contained within j:CaseAugmentation, is 0, unbounded.

In ECF 5, the cardinality for j:CaseLineageCase, contained within j:CaseAugmentation, is 0, 1.

This needs to be corrected and returned to maxOccurs=unbounded.

It is not uncommon for a case, especially an appellate case, to have multiple case lineage predecessors or descendants.

1. **SystemOperationModeCode Validation**

In csprd01 in cbrn:MessageStatus (e.g. RecordDocketingResponse), the value for cbrn:SystemOperationModeCode is forced to be from a constrained list (i.e. ‘Exercise’, ‘Ops’, ‘Other’, ‘Test’, or ‘Unknown’).

This validation does not occur in WD27 and WD28. Why not?

The schema cbrncl.xsd in csprd01 contains the following which is absent from WD27 and WD28:

<xs:enumeration value="Exercise">

<xs:annotation>

<xs:documentation>The system is in use by an exercise.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Ops">

<xs:annotation>

<xs:documentation>The system is in live operational use.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Other">

<xs:annotation>

<xs:documentation>The system is in an unspecified operating mode. A description of this model needs to be provided in the element SystemOperatingModeText.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Test">

<xs:annotation>

<xs:documentation>The system is in test operations.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Unknown">

<xs:annotation>

<xs:documentation>The operating mode of the system is unknown.</xs:documentation>

</xs:annotation>

</xs:enumeration>

1. **MessageStatusCode**

In csprd01 in cbrn:MessageStatus (e.g. RecordDocketingResponse), the value for cbrn:MessageStatusCode is forced to be from a constrained list (i.e. 'ActivityCodeFailure', 'DeviceError', 'DuplicateMessage', 'ErrorAcknowledgment', 'InvalidSchema', 'MessageError', 'Other', 'Success', 'SystemError', or 'UnknownError').

This validation does not occur in WD27 or WD28. Why not?

The schema cbrncl.xsd in csprd01 contains the following which is absent from WD27 and WD28:

<xs:restriction base="xs:token">

<xs:enumeration value="ActivityCodeFailure">

<xs:annotation>

<xs:documentation>The message was successfully received by not successfully processed due to an activity code error.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="DataError">

<xs:annotation>

<xs:documentation>The message was successfully received by not successfully processed due to a data error.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="DeviceError">

<xs:annotation>

<xs:documentation>The message was successfully received by not successfully processed due to a device error.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="DuplicateMessage">

<xs:annotation>

<xs:documentation>The message was successfully received but not processed since it is a duplicate of a message already processed.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="ErrorAcknowledgement">

<xs:annotation>

<xs:documentation>Acknowledgement of receipt of an error message.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="InvalidSchema">

<xs:annotation>

<xs:documentation>The message was received, but was not successfully processed due to an invalid schema.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="MessageError">

<xs:annotation>

<xs:documentation>The message was received, but was not successfully processed due to an invalid message error (invalid Message Type, encoding, format, etc.)</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Other">

<xs:annotation>

<xs:documentation>The message status does not fit any known category.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="Success">

<xs:annotation>

<xs:documentation>The message was sucessfully received and accepted.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="SystemError">

<xs:annotation>

<xs:documentation>The message was successfully received by not successfully processed due to a system error.</xs:documentation>

</xs:annotation>

</xs:enumeration>

<xs:enumeration value="UnknownError">

<xs:annotation>

<xs:documentation>The message was not successfully received and/or processed due to an unknown error.</xs:documentation>

</xs:annotation>

</xs:enumeration>

</xs:restriction>

1. **ecf:CourtEventTypeCode**

This element appears in caserequest:GetCaseRequestMessage, Its current definition is as follows:

“Filter criterion indicating that only calendar entries of the specified type are being requested.”

In caserequest.GetCaseRequestMessage, the element ecf:CourtEventTypeCode is a companion element to element caserequest:IncludeCalendarEventIndicator.

For example, if caserequest:IncludeCalendarEventIndicator is ‘true’, then ecf:CourtEventTypeCode may be used to further ‘reduce’ the calendar events to only those that are of the indicated type (see definition above).

Why does it not have a name that better fits its purpose (e.g. filtering calendar events)?

In the GetCaseResponse, which element is used to send back calendar event information? Is it j:CaseCourtEvent?

If so, then which element is used to return docket events?

1. **MessageStatus in GetCaseResponse**

The element cbrn:MessageStatus is contained, and is optional, within caseresponse:GetCaseResponseMessage.

First off, this is very unusual to embed cbrn:MessageStatus within a response message, unless the response message is composed solely of cbrn:MessageStatus. I believe caseresponse:GetCaseResponseMessage is the only instance of a ‘hybridized’ response message that embeds cbrn:MessageStatus.

So, first, is it correct for cbrn:MessageSttaus to be included within caseresponse:GetCaseResponseMessage?

If so, then when should cbrn:MessageStatus be used in caseresponse:GetCaseResponseMessage and when should it not be used?

1. **Cardinality of ecf:ConnectedDocument**

The cardinality for ecf:ConnectedDocument in CourtEventAugmentation should be unbounded (e.g. see GetCaseResponse). Often a single event will involve multiple documents.

Also, the name of the element may be misleading. Whereas it is true that the document is connected to the court event, it may be a lead document and not a connected document (e.g. as in FilingConnectedDocument).

1. **ConnectedDocument ‘awkwardness’**

In addition to the cardinality addressed above, it is ‘awkward’ and ‘unnatural’ to express the parent/child (e.g. lead/connect) relationship between documents using nc:DocumentAssociation within ecf:ConnectedDocument.

Consider a circumstance where a lead document (e.g. a complaint) has been filed with two connected documents (e.g. exhibit-1 and exhibit-2).

Describing this parent-child document relationship in a GatCaseResponse using ecf:ConnectedDocument would need to be done as shown below, assuming that the cardinality is revised as suggested in item #8:

<ecf:ConnectedDocument> <!—Exhibit-1 -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387761</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 1</nc:DocumentTitleText>

<ecf:DocumentAugmentation>

<nc:DocumentAssociation>

<nc:PrimaryDocument> <!-- parent Complaint document -->

<nc:DocumentIdentification>

<nc:IdentificationID>1234567889</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Complaint</nc:DocumentTitleText>

</nc:PrimaryDocument>

<ecf:DocumentAssociationAugmentation>

<!-- Value must be from DocumentRelatedCode.gc -->

<ecf:DocumentRelatedCode>parent</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

</ecf:ConnectedDocument>

<ecf:ConnectedDocument> <!—Exhibit-2 -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387762</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 2</nc:DocumentTitleText>

<ecf:DocumentAugmentation>

<nc:DocumentAssociation>

<nc:PrimaryDocument> <!-- parent Complaint document -->

<nc:DocumentIdentification>

<nc:IdentificationID>1234567889</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Complaint</nc:DocumentTitleText>

</nc:PrimaryDocument>

<ecf:DocumentAssociationAugmentation>

<!-- Value must be from DocumentRelatedCode.gc -->

<ecf:DocumentRelatedCode>parent</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

</ecf:ConnectedDocument>

This approach is not only ‘awkward’ and ‘unnatural’ it also requires that the parent document is redundantly described.

Perhaps the solution is to add a new DocumentRelatedCode.gc row for ‘child’ (currently this gc file provides: ‘parent’, ‘prior-related’, and ‘reviewed’).

If ‘child’ were to be added, then the above example for a single complaint with two connected exhibits would be:

<ecf:ConnectedDocument> <!— Complaint-->

<nc:DocumentIdentification>

<nc:IdentificationID>1234567889</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Complaint</nc:DocumentTitleText>

<ecf:DocumentAugmentation>

<nc:DocumentAssociation>

<nc:PrimaryDocument> <!-- child Exhibit-1 document -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387761</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 1</nc:DocumentTitleText>

</nc:PrimaryDocument>

<ecf:DocumentAssociationAugmentation>

<!-- Value must be from DocumentRelatedCode.gc -->

<ecf:DocumentRelatedCode>child</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

<ecf:DocumentAugmentation>

<nc:DocumentAssociation>

<nc:PrimaryDocument> <!-- child Exhibit-2 document -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387762</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 2</nc:DocumentTitleText>

</nc:PrimaryDocument>

<ecf:DocumentAssociationAugmentation>

<!-- Value must be from DocumentRelatedCode.gc -->

<ecf:DocumentRelatedCode>child</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

</ecf:ConnectedDocument>

Not only is this much more natural and less awkward, but the lead document (i.e. complaint) is only expressed once.

Note that to provide this solution also requires consideration for adding an nc:SecondaryDocument element which would be used in place of the nc:PrimaryDocument element. If nc:SecondaryDocumnet is added with an unbounded maxOccurs, then the above example could perhpas be further simplified as:

<ecf:ConnectedDocument> <!— Complaint-->

<nc:DocumentIdentification>

<nc:IdentificationID>1234567889</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Complaint</nc:DocumentTitleText>

<ecf:DocumentAugmentation>

<nc:DocumentAssociation>

<nc:SecondaryDocument> <!-- child Exhibit-1 document -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387761</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 1</nc:DocumentTitleText>

</nc:SecondaryDocument>

<nc:SecondaryDocument> <!-- child Exhibit-2 document -->

<nc:DocumentIdentification>

<nc:IdentificationID>8690056387762</nc:IdentificationID>

<nc:IdentificationSourceText>DocumentRenditionID</nc:IdentificationSourceText>

</nc:DocumentIdentification>

<nc:DocumentTitleText>Plaintiff's Exhibit 2</nc:DocumentTitleText>

</nc:SecondaryDocument>

<ecf:DocumentAssociationAugmentation>

<!-- Value must be from DocumentRelatedCode.gc -->

<ecf:DocumentRelatedCode>child</ecf:DocumentRelatedCode>

</ecf:DocumentAssociationAugmentation>

</nc:DocumentAssociation>

</ecf:DocumentAugmentation>

</ecf:DocumentAugmentation>

</ecf:ConnectedDocument>

1. **CaseOfficial**

Element j:CaseOfficial appears in both j:CaseAugmentation and ecf:CaseAugmentation. Their content is the same in both (i.e. j:CaseOfficialType). Why is it duplicated?

1. **Civil Complaint Use Case examples**

Also provided, in a separate zip file (i.e. civil-complaint.zip), are 19 xml instance document examples that collectively comprise a use case for a round-trip for a simple civil complaint case initiation filing.

This use case collection is based on Barbara Holmes Use case #300 – Case Initiation Civil Complaint, alternative 1 (self-represented plaintiff)

All xml instance examples are annotated using xml comments. Some annotations simply provide additional clarifying information. Other annotations raise considerations or seek answers to questions.

The ECF TC members are encouraged to carefully review these use case examples and to review the considerations raised. Additional review/work sessions may be appropriate to review and address these questions and considerations.

1. xxx