**ECF5 Spec Feedback and Considerations – 20**

This document provides additional feedback, questions, and commentary resulting from a review at the Electronic Court Filing Version 5.0 Public Review Draft 1, unless otherwise noted.

1. **Feedback 18 (#4 Filing and Message Identifier)**

Issues raised in prior feedback documents were discussed at the face-to-face meeting in Salt Lake City. The meeting minutes capture the result for the item labeled ‘Feedback 18, #4 Filing and Message Identifiers’:

* + - Decisions:
			* Each MDE gets an opportunity to provide an Identifier in DocumentType (via DocumentIdentification with prescribed nc:IdentificationCategory values that will be defined in a genericcode list) - Completed
			* The originator of a message should get its MessageID back in all synchronous and asynchronous operations - Completed

These decisions are captured in Electronic Court Filing Version 5.0 Public Review Draft 1.

Even with these decisions, some filing and message identifier ‘entanglements’ remain:

1. **Message Identifiers**

The specification allows for message identifiers but does not require them, except in the circumstance where a response (either synchronous or asynchronous) is either required or optionally employed (section 6.2.4). In practice, this is most of the time.

Message identifiers are issued and posted to the message XML by the MDE that sends the message (see section 6.2.4). For example, the FAMDE would assign a message identifier to a GetCaseRequest, and this message identifier would be retuned on the corresponding GetCaseResponse. An FAMDE would also assign a message identifier to a ReviewFilingRequest which would be returned on the corresponding synchronous MessageStatus response.

Message Identifiers and Filing Identifiers are similar, but different things (see definitions, section 1.1). Their purpose, and the way in which they are to be used is also similar, but different.

A Filing Identifier is assigned by the FRMDE upon receipt of a ReviewFilingRequest (RvFR) in the ReviewFiling operation. This Filing Identifier value is returned to the FAMDE in the synchronous MessageStatus response message (see section 6.1.4). In this circumstance, both the original RvFR Message Identifier and Filing Identifier issued by the FRMDE must be returned to the FAMDE on the synchronous response MessageStatus message.

To support this (i.e. returning both a message identifier and a filing identifier to the FAMDE), ecf:MessageStatusAugmentation must be modified to permit multiple nc:DocumentIdentification elements.

Here is an example of this scenario:

1. A FilingMessage is sent to the FRMDE, from the FAMDE, as part of a ReviewFilingRequest. The FilingMessage contains a message identifier issued by the FAMDE (e.g. ‘FA0001’). The XML snippet below shows the FilingMessage with the message identifier:

<filing:FilingMessage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:structures="http://release.niem.gov/niem/structures/4.0/" xmlns:nc="http://release.niem.gov/niem/niem-core/4.0/" xmlns:j="http://release.niem.gov/niem/domains/jxdm/6.0/" xmlns:cbrn="http://release.niem.gov/niem/domains/cbrn/4.0/" xmlns:biom="http://release.niem.gov/niem/domains/biometrics/3.2/" xmlns:ecf="https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/ecf" xmlns:filing="https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/filing" xsi:schemaLocation="https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/filing ../schema/filing.xsd">

 <nc:DocumentIdentification>

 <nc:IdentificationID>FA00001</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 <ecf:DocumentFiler>

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

 </ecf:DocumentFiler>

 <ecf:SendingMDELocationID>

 <nc:IdentificationID>http://example.com/efsp1</nc:IdentificationID>

 </ecf:SendingMDELocationID>

1. Upon receipt of the ReviewFilingRequest, the FRMDE issues a filing identifier (e.g. ‘FR00099’) which travels along with all subsequent exchanges in the e-filing process.
2. The FRMDE responds to the FAMDE by returning a StatusMessage that includes the FRMDE issued filing identifier. Per section 6.2.4, the message identifier sent to the FRMDE from the FAMDE must also be returned as a correlation identifier. The XML snippet below shows this using the anticipated and necessary cardinality revision:

<cbrn:MessageStatus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:structures="http://release.niem.gov/niem/structures/4.0/" xmlns:nc="http://release.niem.gov/niem/niem-core/4.0/" xmlns:ecf="https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/ecf" xmlns:j="http://release.niem.gov/niem/domains/jxdm/6.0/" xmlns:cbrn="http://release.niem.gov/niem/domains/cbrn/4.0/" xsi:schemaLocation="http://release.niem.gov/niem/domains/cbrn/4.0/ ../schema/niem/domains/cbrn/4.0/cbrn.xsd http://release.niem.gov/niem/niem-core/4.0/ ../schema/niem/niem-core/4.0/niem-core.xsd https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/ecf ../schema/ecf.xsd" cbrn:systemSimulatedIndicator="false">

 <cbrn:SystemEventDateTime>2017-01-07T13:47:42.0Z</cbrn:SystemEventDateTime>

 . . .

 <ecf:MessageStatusAugmentation>

 <ecf:ServiceRecipientID>

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

 </ecf:ServiceRecipientID>

 <nc:DocumentIdentification>

 <nc:IdentificationID>FA0001</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 <nc:DocumentIdentification>

 <nc:IdentificationID>FR00099</nc:IdentificationID>

 <nc:IdentificationSourceText>FilingReview MDE</nc:IdentificationSourceText>

 </nc:DocumentIdentification>

 </ecf:MessageStatusAugmentation>

</cbrn:MessageStatus>

The use of the same element, nc:DocumentIdentification, for both message and filing IDs may be confusing. Even when the nc:IdentificationSourceText is provided, it may not be clear whether in ID is a message or filing ID. Perhaps we should use a separate element for message ID. cbrn:MessageStatus includes cbrn:MessageID with the definition “An identifier associated with a message content. There is no required format for the ID value.” and that we could insert into ecf:DocumentAugmentation for other messages. Let’s discuss with the TC.

1. **Message Identifier Clarification**

There appears to be errors/contradictions in section 6.2.4 Message and Filing Identifiers.

This section states:

Message identifiers are assigned by the MDE sending each message and MUST be returned to the originating MDE in any synchronous and asynchronous responses to that message.

Then beneath this, it also provides that the nc:DocumentIdentification/nc:IdentificationSourceText should specify the originating MDE for the identifier.

‘Originating MDE’ is understood to mean the MDE that issued and posted the message identifier to the message.

First, in the statement “The originating MDE of each identifier SHOULD be identified with the nc:DocumentIdentification/nc:IdentificationSourceText element and the IdentificationCategory.gc code list.” Is seems that the word ‘and’ should be replaced with ‘in’.

For the FilingMessage, the Filing Identifier is issued by the FRMDE upon receipt from the FAMDE. This can be gleaned from the first bullet beneath ’intended usage is described as follows’. This bullet states: “In … filing:FilingMessage … the message identifier MUST be assigned by the Filing Review MDE …” This statement is inaccurate. The word “message’ should be replaced by the word “filing”, as “In … filing:FilingMessage … the filing identifier MUST be assigned by the Filing Review MDE …” (also see section 6.1.4 which also refers to this as “the filing identifier”).

This bullet could be rewritten as follows:

A filing identifier is issued by the Filing Review MDE upon receipt of a valid ReviewFilingRquest to identify a unique filing in the court. This filing identifier MUST be present in each docket:RecordDocketingMessage, reviewfilingcallback:NotifyFilingReviewCompleteMessage, docketcallback:NotifyDocketingCompleteMesage, cancel:CancelFilingMessage and MUST also be present in each synchronous cbrn:MessageStatus response to these messages. This filing identifier MUST be present in each filing:FilingMessage, except in the ReviewFilingRequest. The filing identifier MUST also be provided in the filingstatusrequest:GetFilingStatusRequestMessage and the filingstatusresponse:GetFilingStatusResponseMessage.

 These changes will be included in WD28.

1. **Non-Normative asynchronous response message identifier example**

I suggest that the example in section 6.2.4 be extended to also include nc:IdentificationSourceText, such as:

<filing:FilingMessage ... >

 …

 <nc:DocumentIdentification>

 <nc:IdentificationID>cf42805c-5e4d-4ba3-850a-c9c635c255b5</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 …

</filing:FilingMessage>

These changes will be included in WD28.

1. **NotifyDocumentStampInformation**

The ‘Filing and Service Process’ diagram (Figure 1) does not include NotifyDocumentStampInformation.

1. **Typographical Issues**

Section 2.1 Relationship to Prior Specifications

1. Currently:

• New Document Stamp and operations support retrieval of case information required for stamping.

Should this be:

New Document Stamp operations that support retrieval of case information required for stamping

1. Currently:

• Deprecated content references (e.g. referring to related entities with common identifiers) in favor or element references (e.g. referring to related elements with structures:ref attributes) as described in Reference Rules.

Should be:

• Deprecated content references (e.g. referring to related entities with common identifiers) in favor of element references (e.g. referring to related elements with the structures:ref attributes) as described in Reference Rules

1. Section 3.2.2 The Scheduling Process, contains an extra unneeded space following the word “Process” and before the comma:



These changes will be included in WD28.

1. **CaseNumberText in DocumentStampInformationMessage and other messages**

The DocumentStampInformationMessage is designed to be used for both case initiation filings and subsequent filings. In a case initiation filing, the case number may be unknown when the document stamp request is submitted (and may be provided in the response). However, in a subsequent filing circumstance, the case number should be known and should be permitted on the request message. Currently only nc:CaseTrackingID is provided.

Also, it seems reasonable to add j:CaseNumberText to the GetDocumentRequestMessage, the GetFilingListRequestMessage, and the GetFilingStatusRequestMessage so that case and document list information can be requested by court and case number.

CaseNumberText should also be included in GetDocumentRequestResponse.

These messages each need a case identifier known by the both the sending and received MDEs, either CaseTrackingID or CaseNumberText, but not both. Let’s discuss with the TC which is best for these messages.

1. **CaseTrackingID cardinality on GetCaseRequest**

Why is the cardinality maxOccurs for nc:CaseTrackingID on the GetCaseRequestMessage two (2)?

Should this be one (1) instead as it is in ECF 4.01 CaseQueryMessage?

 The cardinality for caserequest:GetCaseRequestMessage/nc:CaseTrackingID will be set to 1,1 in WD28.

1. **Multiple Document Identifiers**

The term ‘document identifier’ is not defined in the glossary in section 1.1.

The element definition for nc:DocumentIdentifcation is “an identification that references a document.” This is a bit wide open.

The cardinality for nc:DocumentIdentification is maxOccurs unbounded. This permits multiple document identifiers to be associated with a single document. The cardinality for nc:DocumentIdentification is established based on the need for multiple identifiers for messages (see Feedback 5, #11 for business use case).

However, section 6.2.5 it states that “document identifiers are assigned by the MDE sending each message and MUST be returned to the originating MDE in any synchronous and asynchronous responses to that message.”

The minOccurs for nc:DocumentIdentification is one, therefore the specification requires at least one document identifier. A typical e-filing process begins in the FAMDE, therefore the first, and perhaps only document identifier for a document is typically assigned by the FAMDE. Note: documents may be added at other points in the process, such as clerk review; as such then the contributing MDE would assign the minimal one document identifier.

It is not clear whether there is any expectation for additional document identifiers to be added to a document as it moves from MDE to MDE. For example, in the ReviewFilingRequest (RvFR) a document identifier is added in the FilingMessage by the FAMDE. Since the FilingMessage is also subsequently contained within the RecordDocketingRequest sent by the FRMDE, one may interpret the specification (6.2.5) to mean that the FRMDE must now add an additional document identifier when it sends the message to the CRMDE. If done in this way, then the document would have two document identifiers, one assigned by the FAMDE, and another one assigned by the FRMDE.

However, this is not what I would expect, therefore I think this interpretation of 6.2.5 may not be correct (even if it is not an unreasonable interpretation given the wording).

If it is correct that the FRMDE adds an additional document identifier to the document in the RecordDocketingMessage, then is it added in the FilingMessage (e.g. to FilingLeadDocument) or is it introduced into the RecordDocketingMessage (e.g. to ReviewedLeadDocument)?

So, can this be cleaned up a bit?

First, I will presume that the interpretation offered above is not correct and that a document need only be assigned a single document identifier by the MDE that first introduces it into the transaction and not each time it is sent from one MDE to another.

Perhaps the following rewording will help:

Documents are elements derived from nc:DocumentType other than the messages identified in the previous section. Document identifiers are assigned to a document by the MDE that initially introduces the document into the transaction. The document identifier MUST be returned to the originating MDE in any synchronous and asynchronous responses to that message. Document identifiers include the following:

This change will be made in WD28.

Since section 6.2.5 requires that document identifiers are returned to the originating MDE on any synchronous and asynchronous response message, and since MessageStatus is one such response message, this is further reason to change the cardinality for nc:DocumentIdentification in MessageStatusAugmentation (also see #2 above).

But this also leads to a further problem. Some of the nc:DocumentIdentifcation/nc:IdentifierID elements in MessageStatus will identify messages and others will identify documents (and may also include a filing identifier). How will each be distinguished?

For example, if ReveiwFilingRequest from the example in #2 above contained two FilingLeadDocuments with document identifiers ‘LD0001’ and ‘LD0002’, then the MessageStatus message returned to the FAMDE from the FRMDE would include:

xmlns:structures="http://release.niem.gov/niem/structures/4.0/" xmlns:nc="http://release.niem.gov/niem/niem-core/4.0/" xmlns:ecf="https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/ecf" xmlns:j="http://release.niem.gov/niem/domains/jxdm/6.0/" xmlns:cbrn="http://release.niem.gov/niem/domains/cbrn/4.0/" xsi:schemaLocation="http://release.niem.gov/niem/domains/cbrn/4.0/ ../schema/niem/domains/cbrn/4.0/cbrn.xsd http://release.niem.gov/niem/niem-core/4.0/ ../schema/niem/niem-core/4.0/niem-core.xsd https://docs.oasis-open.org/legalxml-courtfiling/ns/v5.0/ecf ../schema/ecf.xsd" cbrn:systemSimulatedIndicator="false">

 <cbrn:SystemEventDateTime>2017-01-07T13:47:42.0Z</cbrn:SystemEventDateTime>

 . . .

 <ecf:MessageStatusAugmentation>

 <ecf:ServiceRecipientID>

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

 </ecf:ServiceRecipientID>

 <nc:DocumentIdentification>

 <nc:IdentificationID>FA0001</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 <nc:DocumentIdentification>

 <nc:IdentificationID>FR00099</nc:IdentificationID>

 <nc:IdentificationSourceText>FilingReview MDE</nc:IdentificationSourceText>

 </nc:DocumentIdentification>

 <nc:DocumentIdentification>

 <nc:IdentificationID>LD0001</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 <nc:DocumentIdentification>

 <nc:IdentificationID>LD0002</nc:IdentificationID>

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

 </nc:DocumentIdentification>

 </ecf:MessageStatusAugmentation>

</cbrn:MessageStatus>

Note that the nc:DocumentIdentifcation elements may appear in any order.

What is the purpose of the 6.2.4 requirement that “Document identifiers … MUST be returned to the originating MDE in any synchronous and asynchronous responses …”?

It does make sense to have a requirement that all documents submitted by the FAMDE to the FRMDE in the ReviewFilingRequest are accounted for by the conclusion of the e-filing transaction. The transaction would conclude upon receipt by the FAMDE of the last NotfyFilingReviewComplete message for the transaction (i.e. with the matching filing identifier) in which all documents originally submitted by the FAMDE had been dispositioned or otherwise accounted for (e.g. in the circumstance of an approved cancellation request). In this context, the document identifiers originally provided by the FAMDE should be returned to the FAMDE in the NotifyFilingReviewComplete message.

It also makes sense that the response message to the GetDocumentRequestMessage return the document identifiers (i.e. nc:DocumentFileControlID) for requested documents provided.

Are there any other exchanges where it would be important or beneficial to require the return of document identifers?

The requirement that the document identifier be returned to the originating MDE is necessary so that the originating MDE can receive the disposition of the document. In my opinion, we should not try to define all potential uses of that disposition by the originating MDE.

1. **PaymentMessage missing in NotifyDocketingCompleteRequest**

The PaymentMessage has not been included in the NotifyDocketingCompleteRequest.

Also, something else seems to be amiss. XMLSpy shows that NotifyDocketingCompleteRequestType is composed of reviewfilingcallback:NotifyFilingReviewCompleteMessage (as shown in the XML fragment below):

 <xs:complexType name="NotifyDocketingCompleteRequestType">

 <xs:sequence>

 <xs:element ref="reviewfilingcallback:NotifyFilingReviewCompleteMessage"/>

 </xs:sequence>

</xs:complexType>

Shouldn’t this be docketcallback:NotifyDocketingCompleteMessage instead?

The NotifyDocketingCompleteMessage has an appropriate element definition for this context:

The message returned from the Court Record MDE to the Filing Review MDE when the functions of entering information onto the docket or register of actions and commiting a filed document(s) to the official court record have been completed, conveying the results of those functions.

Note however that the above description contains a spelling error; should be ‘committing’.

These 3 errors will be fixed in WD28.

1. **Case Type and Classification**

ECF provides ecf:CaseCategoryCode (“The type of trial in the lower court (e.g. bench, jury)”. This may be the same or similar to the NCSC’s Manner of Disposition (e.g. Jury Trial, Bench/Non-Jury Trial, Non-Trial Disposition).

NIEM provides nc:CaseCategoryText (“a kind of case”), nc:CaseSubCategoryText (“a more specific classification of a kind of case”), and nc:CaseGeneralCategoryText (“a broadly defined kind of case”).

ECF provides support for the seven National Center for State Courts (NCSC) case types (i.e. criminal, violations (e.g. citation), juvenile, domestic, civil, bankruptcy, and appellate). More recently, additional types of cases have been suggested (e.g. auxiliary, and administrative – see ‘[So You Think You Know What a Case Is](http://www.ncsc.org/sitecore/content/microsites/future-trends-2013/home/monthly-trends-articles/so-you-think-you-know-what-a-case-is.aspx)?’)

ECF also provides ecf:CaseTypeCode (“Indicates whether the e-filing system supports electronic filing of a certain case type”). A companion CaseTypeCode.gc provides the seven NCSC case types.

The NCSC further breaks down case types into finer categories; for example, civil is further classified as Automobile Tort, Intentional Tort, Malpractice – Medical, Malpractice – Other, Premises Liability, Product Liability, Slander/Libel/Defamation, Tort-Other, Buyer Plaintiff, Employment Dispute – Discrimination, Employment Dispute – Other, Fraud, Landlord/Tenant Dispute – Unlawful Detainer, Landlord/Tenant Dispute – Other, Mortgage Foreclosure, Seller Plaintiff (Debt Collection), Contract – Other, Eminent Domain, Real Property – Other, Small Claims, Guardianship – Adult, Guardianship – Juvenile, Guardianship – Unknown, Conservatorship/Trusteeship, Probate/Wills/Intestate, Probate/Estate – Other, Mental Health, Appeal from Administrative Agency, Appeal from Limited Jurisdiction Court, Civil Appeals – Other, Habeas Corpus, Non-Domestic Relations Restraining, Tax, Writ, and Civil – Other.

In Arizona, we use a variety of Case Management Systems across the various jurisdictions from limited jurisdiction to the Supreme Court. These CMSs encompass those that have been locally developed to vendor purchased solutions. We have generally learned that three levels of case classification are necessary and sufficient to adequately describe the kind of case. We have extended the ECF 4.01 specification to use the three NIEM provided case classification elements for this purpose (i.e. nc:CaseGeneralCategoryText, nc:CaseCategoryText, and nc:CaseSubcategoryText). This is especially important for case initiation submissions.

Why was nc:CaseCategoryText removed in ECF 5?

Is ecf:CaseCategoryCode intended to replace nc:CaseCategoryText even though they are semantically different?

Can the three NIEM elements (nc:CaseGeneralCategoryText, nc:CaseCategoryText, and nc:CaseSubcategoryText) be provided for all case types?

If ecf:CaseCategoryCode was not added as a replacement, can it be renamed to avoid confusion with nc:CaseCategoryText, such as ecf:CaseProceedingsTypeCode or other (open to suggestion)?

NIEM includes the following:

|  |  |  |
| --- | --- | --- |
| Name | Definition | Example Content |
| nc:CaseCategoryText | A kind of case. | law enforcement, prosecution, court, public safety, department of human services, department of health |
| nc:CaseGeneralCategoryText | A broadly defined kind of case | criminal, civil, juvenile, family |
| nc:CaseSubCategoryText | A more specific classification of a kind of case. | employment (to further classify a civil case), homicide (to further classify a felony case) |

 In ECF, we need to provide at least two classifications:

1. a standardized and simple list of case types to identify the high-level case type and enable our case-type augmentations. Currently, that is provided by the ecf:CaseTypeCode element. If the TC prefers, we could replace this by nc:CaseGeneralCategoryText. The code list is provided in the specification and based on the NCSC statistical model. Let’s discuss.
2. A court-specific, and potentially hierarchical, list of case types to support the court divisions and work flows. Currently, that is provided by ecf:CaseCategoryCode and the court-provided Genericode list which could include multiple columns and a hierarchy if needed. Based on the example content, neither nc:CaseCategoryText not nc:CaseSubCategoryText seem like a good fit for this.