2GEFS-CCMS Implementation Guide - DRAFT

AUGUST 2010

ADMINISTRATIVE OFFICE OF THE COURTS
INFORMATION SERVICES DIVISION

Judicial Council of California
Administrative Office of the Courts
Information Services Division
455 Golden Gate Avenue
San Francisco, CA 94102-3688

Table of Contents

Revision History

This section includes this documents' revision history:

Version	Date	Editor	Changes	Sections
1.8	9/20/2009	Todd Vincent, <xmlLegal>	First externally published draft based on 2GEFS 02 and CCMS V3 Release 10.	All Sections.
2.0	3/15/2010	Todd Vincent, <xmlLegal>, Spring Snuffin	Changed fonts and styles to match AOC format.	All Sections.
2.1	8/23/2010	Todd Vincent, <xmlLegal>	Change to introduction to explain draft status and other minor change.	Introduction.

Introduction

This document is a technical guide explaining how the California Case Management System Version 3 (CCMS or CCMS V3) implements California's Second Generation E-Filling Specifications (2GEFS). Specifically, this document is a guideline of the implementation of 2GEFS Court Filing 02 (Filing XML), 2GEFS Court Policy 02 (Policy XML) in CCMS V3, and CCMS Case Type and Document Type Plug-In Schemas. This document is based on the CCMS implementation for Release 10.0.0 (November, 2009).

Additional lessons learned and changes occurred during the Superior Court of California, Orange County complex civil implementation during earlier in 2010. These lessons learned have not yet been incorporated into this document. This document will be re-published in the near future incorporating the lessons learned. The current CCMS V3 release is 10.0.2.

The *purpose* of this document is to provide Electronic Filing Service Providers' (EFSPs) technical staff with guidelines for implementing electronic filing to work with the statewide CCMS V3 system.

CCMS is a case management system (CMS) that includes "core" functionality plus Electronic Filing Manager (EFM) and Clerk Review components. This document is based on testing of the EFM and Clerk Review. Unless stated otherwise, when this document uses the terms *CCMS* or *CCMS implementation*, the term applies to the CCMS EFM and Clerk Review components, but not to the "core" case management system.

Audience

This document's *audience* is technical staff and software developers working to implement electronic filing applications. It is assumed that the reader has working knowledge of the following acronyms and terms: SOAP, WSDL, XML, XML Schema, HTTP, HTTPS, SMTP, URI, URL, 2GEFS, CMS, EFM, Clerk Review, Filing XML, Policy XML, Synchronous Confirmation XML, and Asynchronous Confirmation XML. These acronyms and terms are not explained in this document.[1]

Document Organization and Guidelines

The *document's organization* begins with Envelope XML and *transmission* and then follows the order of the elements in the Filing XML's schema. This includes an explanation of Case Type and Document Type Plug-In Schemas. The final sections include topics specific to Confirmation XML. Each section includes information about Filing XML, Confirmation

[1] Contact Christopher.Smith@jud.ca.gov if you require additional, background resources.

XML, and Policy XML as necessary. Certain Filing XML elements are not used in the CCMS implementation.[2] Discussion of these elements is omitted.

Matters deemed important are marked: **IMPORTANT**:. If you are reading this document quickly or are using it as a reference, please ensure that you understand sections marked as important.

The guidelines in this document are derived from (a) the 2GEFS documentation, (b) CCMS V3 R10 E-Filing functional and technical design documentation, and (b) the CCMS PRE-PAT and PAT testing activities.

The key words "must", "must not", "required", "shall", "shall not", "should", "should not", "recommended", "may", and "optional" in this document are to be interpreted as described in RFC 2119.[3]

Access to Resources

To obtain access to resources references in this document, you will need your <xmlLegal> user name and password at:

http://www.xmllegal.org/Password/

If you do not have an <xmlLegal> username and password, register as a "Guest" at the following address:

http://www.xmllegal.org/MembershipNew/

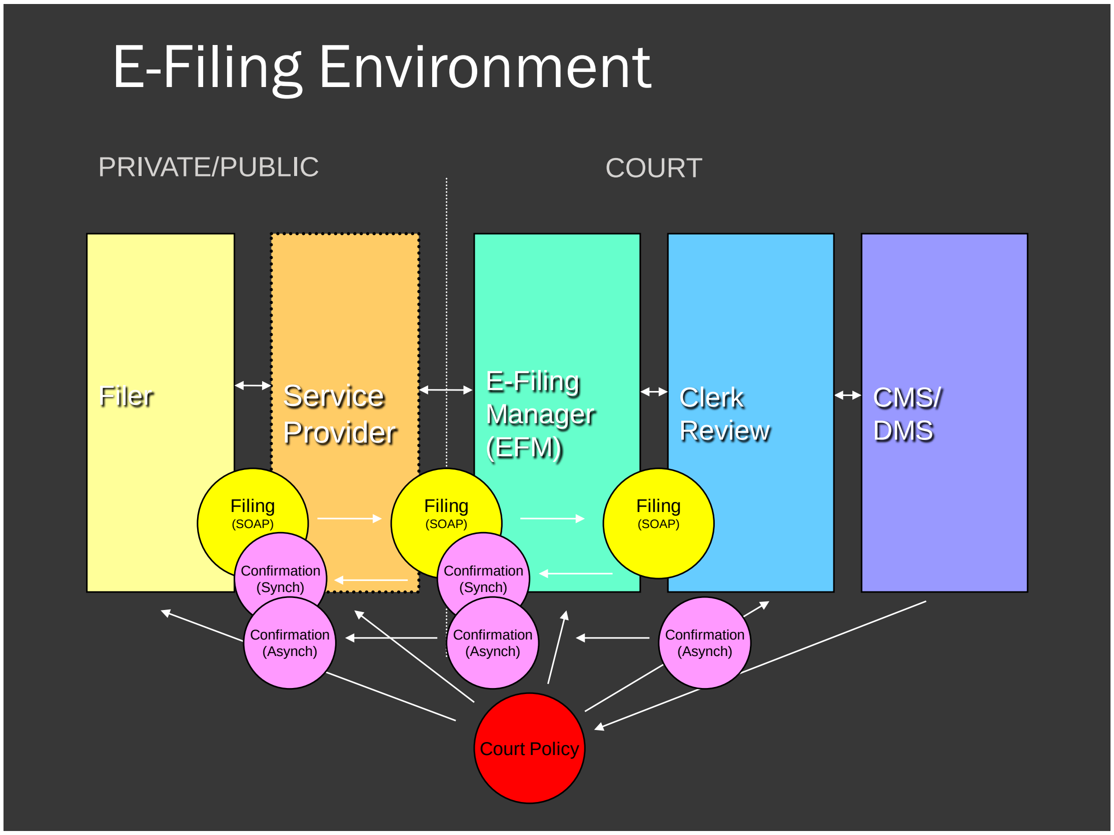
Contact Christopher.Smith@jud.ca.gov with comments about this document.

[2] For example, CCMS does not support 2GEFS elements such as **Filing:LegacyCase**, **Filing:Charges**, and **Filing:Service**.

[3] See http://www.ietf.org/rfc/rfc2119.txt.

1. Connectivity, Testing, Transmission, and Envelope XML

The following diagram illustrates the CCMS-2GEFS e-filing environment and the transactions that occur in it.



Please contact Christopher.Smith@jud.ca.gov if you require additional, background resources.

1.1. Use of SOAP, WSDL, and 2GEFS Envelope

The CCMS implementation requires the use of a SOAP 1.2 XML envelope for transmission of Filing XML to the court. The CCMS EFM SOAP service is defined by a WSDL file. The WSDL file defines a one method, one argument service. The input argument is defined as XML. The single output argument is likewise XML.

The XML input must be 2GEFS Enveloped Filing XML (Filing XML). Filing XML is defined as XML (a) that can be validated against the 2GEFS Envelope schema using the namespace below and (b) that uses the **Envelope:Body/Filing:Filing** node defined in the schema.

http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/Envelope/01/

The output XML is 2GEFS Enveloped Synchronous Confirmation XML (Synchronous Confirmation XML). Synchronous Confirmation XML is defined as XML (a) that can be validated against the 2GEFS Envelope schema using the same namespace, above, and (b) that use the **Envelope:Body/Confirmation:Confirmation** node defined in the schema

An example WSDL file is located at the following address:

http://www.xmllegal.org/Documents/2GEFS/CCMSEfmWebService_Ver0_1_2007_01_16.xml

The CCMS system is implemented in the California Courts Technology Center (CCTC) and in individual courts. The CCTC implementation supports multiple courts (e.g., Ventura, Sacramento, and San Joaquin). Individual courts support only one court (e.g., Orange, San Diego). For example, Sacramento County and Ventura County both operate from the same CCMS EFM located in CCTC, whereas Orange County operates from a local CCMS EFM implementation located in Orange County. The WSDL is the same for each implementation, except for the WSDL SOAP address.[4]

The SOAP envelope includes only a SOAP body. There is no SOAP header. See Section 2 for more information about the placement of the 2GEFS Header element.

1.2. No SOAP for Asynchronous Confirmations

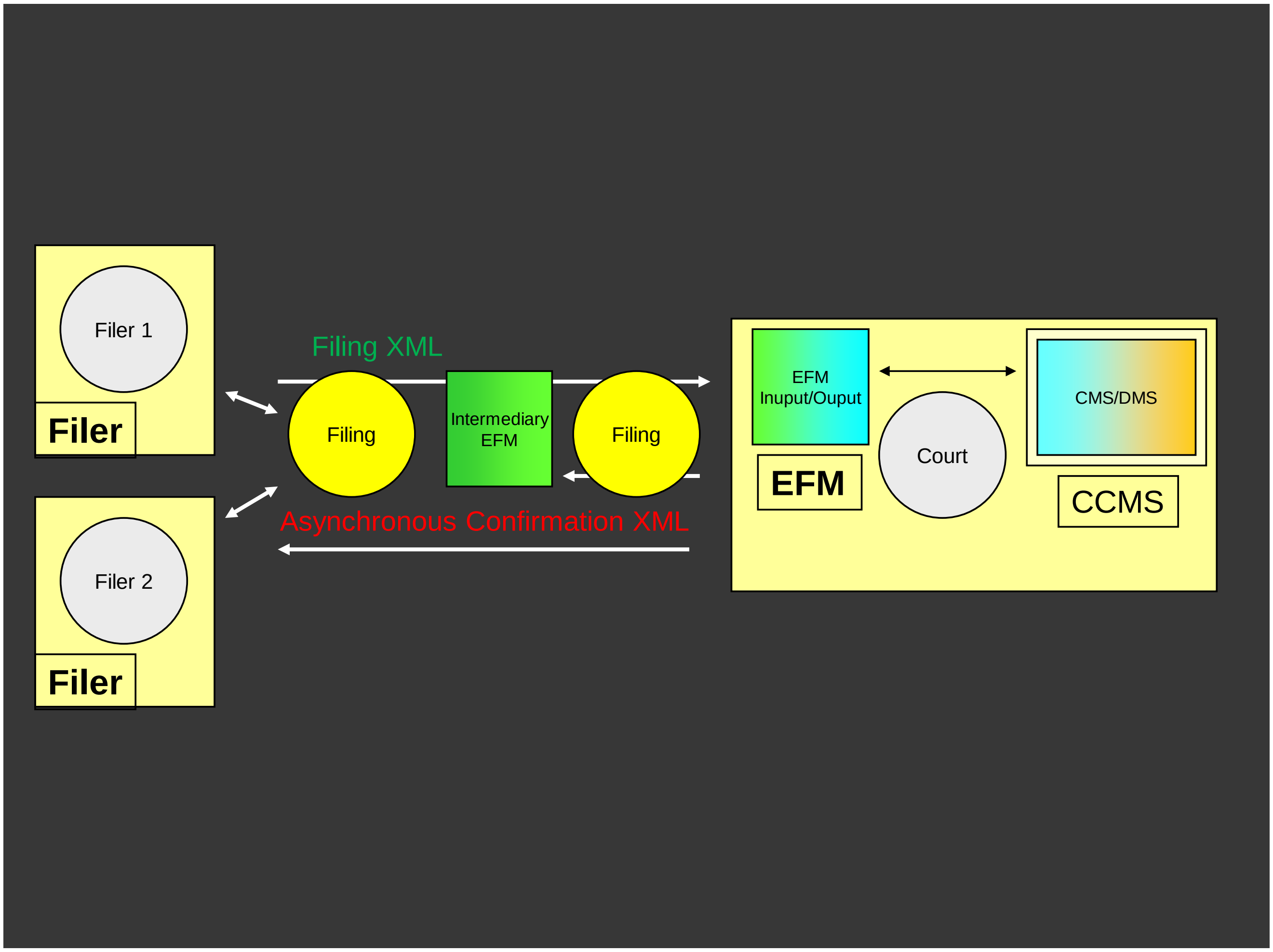
The CCMS implementation does not use SOAP to send Asynchronous Confirmation XML from the EFM to the EFSP. CCMS uses XML over HTTPS to deliver Asynchronous Confirmation XML to EFSP intermediary EFMs. Accordingly, EFSPs must implement XML over HTTPS to receive Asynchronous Confirmation XML. The EFSP receiving application must support HTTPS. The EFSP receiving application's URL must not be password protected. Finally, while the ReplyTo URL may be dynamic from an outgoing EFM perspective, the CCTC may require outbound IP filtering, which, in turn, may limit the EFSPs ability to vary ReplyTo URLs. See Section 2.3 for more information.

1.3. Intermediary EFM Required

EFSPs that electronically file into the CCTC are required to use an intermediary EFM to forward filings into the CCMS EFM and to receive asynchronous confirmations from the EFM. The

[4] CCMS EFM releases sometimes publish WSDL with a SOAP address that points to a host name local to the CCTC network that is not resolvable on the Internet. EFSPs will be supplied with an Internet-resolvable address to fetch the WSDL. Depending on the SOAP/WSDL development tools used, the EFSP may need to edit generated code to replace the WSDL host name with the resolvable host name.

following diagram illustrates the position of an intermediary EFM in a CCMS-2GEFS electronic filing environment.



Local courts may have different policies.

1.4. CCTC Connectivity Requirements

In addition to an Intermediary EFM, EFSPs that electronically file into the CCTC have additional connectivity requirements as defined in the “EFSP-CCTC-AOC Connectivity, Support, and Onboarding” document. This document defines requirements and policies for CCTC connectivity, including:

- Environments
- Ingress and Egress Connectivity Requirements
    - Connectivity, Ports, Protocols

  - Firewall
  - Authentication
  - SSL and Certificates
- Service Desk Support

Local court implementations may have different connectivity requirements and policies.

1.5. Legal Agreements

EFSPs must negotiate and sign service provider agreements with individual courts. CCTC courts will make a request on behalf of the EFSP to the Administrative Office of the Courts for connectivity into the CCTC. EFSPs connecting to the CCTC will be required to sign an agreement regarding CCTC connectivity with the California Judicial Council, Administrative Office of the Courts, in addition to the agreement with the court. Statewide CCTC connectivity agreements are undergoing internal review and may be provided as addendum to this or other documentation in the future. Contact Christopher.Smith@jud.ca.gov for more information.

1.6. Test and Production Environments

CCTC implementations use the following terminology to describe pre-production and production environments in which testing and production electronic filing takes place. Your AOC or CCTC representative will have additional information about the environment into which you will test.

- ***User Acceptance Testing (UAT)***: Pre-Production environment for testing.
- ***Localization Test (LT)***: Pre-Production environment for testing.
- ***Staging***: Pre-Production environment for testing.
- ***Production***: Live filings occur in the production environment.

Local court implementations may have different environments and naming conventions.

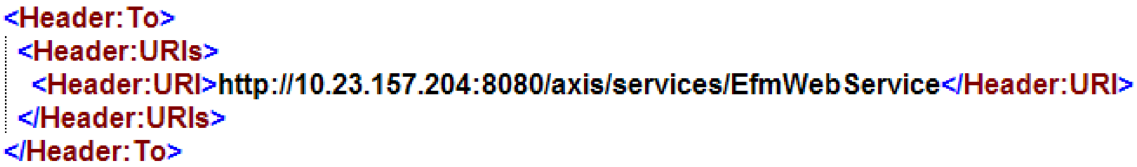
2. Header

The 2GEFS Header element is included within the 2GEFS Envelope. The 2GEFS Header element must not be placed within the SOAP Header element.[5] The CCMS construction is consistent with existing non-SOAP implementations that only use a 2GEFS Envelope.

2.1. Header To

CCMS implements the **Header:To** element in Filing XML, Synchronous Confirmation XML, and Asynchronous Confirmation XML according to the 2GEFS specifications.[6]

The value should be the URL of the CCMS EFM SOAP/WSDL service installed at the court. An example value would be:



```
<Header:To>
 <Header:URIs>
  <Header:URI>http://10.23.157.204:8080/axis/services/EfmWebService</Header:URI>
 </Header:URIs>
</Header:To>
```

2.2. Header From

IMPORTANT: CCMS requires **Header:From** information that provides contact information for the EFSP.

Filing applications are required to put the EFSP's name and contact details inside a **Header:From/Header:Organizations/Header:Organization** element. The information must include an EFSP organization name, contact person, contact phone number, and email address. The EFSP information is exposed in the CCMS Clerk Review interface so that court clerks can contact the EFSP with questions about the electronic filing, if necessary.

CCMS does not support the **Header:From/Header:People/Header:Person** element.

2.3. Header ReplyTo

IMPORTANT: CCMS currently requires one and only one **Header:ReplyTo/Header:URIs/Header:URI** element.

[5] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/01/Header/01/#Header.

[6] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/01/Header/01/#To.

CCMS will not send Asynchronous Confirmation XML to more than one **Header:URI** value. If more than one **Header:URI** value is included in the header, then CCMS will only use the first **Header:URI** value to send Asynchronous Confirmation XML. CCMS ignores other values. The **Header:URI** value must be an HTTPS value. See Section 2.2 for more information.

IMPORTANT: CCMS does not support SMTP or other types of **Header:ReplyTo** addresses.

2.4. Header MessageIdentifier

CCMS passes the **Header:MessageIdentifier** elements in the Synchronous Confirmation XML and the Asynchronous Confirmation XML as they appear in the Filing XML.

Filing applications must create **Header:MessageIdentifier** children elements in the same way as the **Filing:Key** element. See Section 3.1 for more information.

2.5. Header Date and Header Time

CCMS implements the **Header:Date** and **Header:Time** elements in Filing XML, Synchronous Confirmation XML, and Asynchronous Confirmation XML according to the 2GEFS specifications.[7] See Section 4 for more information about dates and times.

2.6. Header Other Elements

CCMS does not implement **Header:CC**, **Header:Credentials**, or **Header:Extensions**. You should not use these elements.

[7] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/01/Header/01/#Date and http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/01/Header/01/#Time.

3. Filing

3.1. Filing Key

CCMS implements the **Filing:Key** and **Confirmation:FilingKey** elements in Filing XML, Synchronous Confirmation XML, and Asynchronous Confirmation XML.

CCMS requires an *appropriate value* in the **Key:OrganizationKey** element.[8] The 2GEFS specification states:

> **[9]** **Key:OrganizationKey** is a unique identifier for the organization that is responsible for creating the XML. The unique identifier should be assigned by a policy authority in the jurisdiction in which the XML is being used. If no such policy authority exists to assign the identifier, then the organization should use its organization name, if its name is unique, or a domain name owned by the organization. For example, **Key:OrganizationKey** might be `FileDocs, Inc.` Used in combination with **Key:Application** and **Key:Version**, the **Key** element would identify the organization, application, and version that is responsible for the XML. For example, `FileDocs, Inc. E-Filing Manager 0.0.3`.

IMPORTANT: The following table includes the *organization key* value for the EFSPs that have either filed into the California AOC Test Suite, that are known to have a 2GEFS implementation (whether live or still in development), or that have requested an organization key. Services providers must use these values for the **Key:OrganizationKey** element:

Company Name	Organization Key
American Legal Net	ALN
MMC Group, L.P.	CaseFileXpress
Couterclaim.com	Counterclaim
DDS Legal Support Systems	DDSLegal

[8] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/BuildingBlocks/Primitives/Key/03/#OrganizationKey.

Essential Publishers	EssentialPublishers
Glotrans	Glotrans
First Legal	FirstLegal
Orange Legal Aid Society	ICAN
Intresys	EZLegalFile
Lawdex	Lawdex
Lexis-Nexis	Lexis
One Legal	OneLegal
WizNet	WizNet

If your company is not on the list and you require an *organization key*, contact Christopher.Smith@jud.ca.gov.

Filing applications must also conform to the 2GEFS requirements for the **Key:Application** and the **Key:Version** elements.[9] These values are useful when developers debug problems with Filing XML from multiple sources.

3.2. Sealed Filings

CCMS supports the **Filing:Sealed** element. The **Filing:Sealed** element is defined as a request by the filer to the court to seal the filing. This is a request by the filer only. The court is not required to grant the request.

When the **Filing:Sealed** element is set to `true`, the *lead document* is always visible to the court clerk. The attachments are visible only to a judicial officer.

[9] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/BuildingBlocks/Primitives/Key/03/#Application and http://www.xmllegal.org/Schema/Court/US/California/2GEFS/BuildingBlocks/Primitives/Key/03/#Version.

There are two scenarios in which a filing can be submitted as sealed (a) a new request and (b) a granted request. In both situations, the **Filing:LeadDocument** element must include a document that is not requested to be under seal or that is not under seal. The lead document may be a motion to seal (without confidential information in it) or an existing order to seal.

The **Filing:Attachments** elements must include the documents requested to be sealed or already ordered to be sealed. Keep in mind that if the court does not grant the request, the attachments may become visible to court staff other than a judicial officer.

4. Dates and Times

Dates only include the date, without the time zone offset, as shown below. Times always include the time zone offset, as shown below. This is the new rule for all incoming and outgoing dates and times.

```
<Confirmation:Date>2007-11-01</Confirmation:Date>
<Confirmation:Time>10:53:26-07:00</Confirmation:Time>
```

Daylight savings time must be considered when stating the time zone offset (e.g., -07:00).

4.1. Synchronous Confirmation Dates and Times

In the Synchronous Confirmation XML, CCMS returns the date and time at the moment of generation in the **Confirmation:Date** and **Confirmation:Time** elements. CCMS copies the **Filing:Date** and **Filing:Time** values from Filing XML into **Confirmation:FilingDate** and **Confirmation:FilingTime**.

4.2. Asynchronous Confirmation Dates and Times

CCMS returns the legally significant date and time of acceptance in the **Confirmation:FilingDate** and **Confirmation:FilingTime** elements. As a general rule, the *acceptance date* is the date the court received the filing from the service provider.

An exception to the general rule occurs when the filing is submitted after normal court business hours or on a holiday. According to California Rules of Court on Electronic Filing and Service (CRC 2.250-2.261), electronic filings submitted after the court's regular business hours "will be deemed to have been received on the next business day". As a result, the *acceptance date* for submissions after hours or on a holiday will be the next court business day.

CCMS also allows the court clerk to backdate a filing. If a filing is backdated, then the **Confirmation:FilingDate** and **Confirmation:FilingTime** values may be any value selected by the court clerk, except that it is not possible to backdate a filing prior to the case initiation date.

4.4. Dates and Times in PDF Confirmation of Receipt and Endorsed Document

All Asynchronous Confirmation XML includes a court initiated PDF confirmation of filing. The PDF confirmation of filing includes dates and times that mirror the dates and times in the Asynchronous Confirmation XML.

Asynchronous Confirmation XML includes electronically stamped (endorsed) PDF documents. Such documents are the originally filed PDF document, plus a graphical stamp flattened into the PDF. Some stamped documents include the *acceptance date*, which matches the **Confirmation:FilingDate** value.

4.5. Date and Time Matrix

The Filing 02 specification includes a date and time matrix that summarizes the expected values for dates and times in Filing XML and Confirmation XML. Use the following table as a reference.

Elements	Filing	Confirmation Synchronous (Received)	Confirmation Synchronous (Error)	Confirmation Asynchronous (Accepted)	Confirmation Asynchronous (Rejected)
Header:Date/Time	Moment of generation	Moment of generation	Moment of generation	Moment of generation	Moment of generation
Filing:Date/Time	Filer's assertion of when filed	N/A	N/A	N/A	N/A
Confirmation:Date/Time	N/A	Moment of generation	Moment of generation	Moment of generation	Moment of generation
Confirmation:FilingDate and **Confirmation:FilingTime**	N/A	Mirror Filer's assertion of when filed	If available, mirror Filer's assertion of when filed. Otherwise, moment of generation	Court's determination of when filed	Court's determination of when filed

Note that the legally significant acceptance date is the **Confirmation:FilingDate**. This date is derived from the date and time the court *received* the filing (see exceptions above). It is ***not*** the date and time that the court *processed* the filing, which is always later in time.

5. Case

5.1. Initial Filings

An *initial filing* is Filing XML that uses the **Case:New** element and its children elements. Both 2GEFS and CCMS require the following elements to initiate a case:

- **Case:Category**
- **Case:Title**

CCMS also requires the **Case:Type** and **Case:JurisdictionalAmount** elements, which are optional in 2GEFS.[10] CCMS also supports the **Case:DataFile** element, described in Section 10.

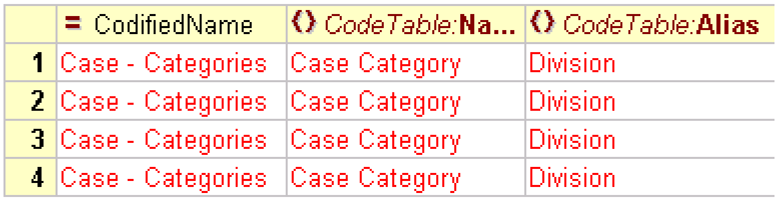
5.1.1. Case Category

The **Case:Category** element maps to the CCMS *case category*. CCMS currently supports four *case categories*, represented by the following values:

- `Civil - Limited`
- `Civil - Unlimited`
- `Probate`
- `Small Claims`

These values are mapped to a *court key* inside a court's Policy XML in a **Policy:Table** element. Depending on the CCMS *case categories* implemented by a court, there may be more than one **Policy:Table** element, since each CCMS case category must map to at least one Policy XML *court key* (and could map to more than one *court key*).

The following table shows the grid view of Policy XML for a court that supports more than one CCMS *case category*. The **CodeTable:Name** element value maps to the CCMS name. The **CodeTable:Mapping**, which is not shown in the diagram below, provides an XPath mapping to the Filing XML element (i.e., **Case:Category** in this situation).

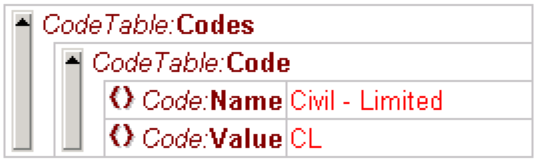


	= CodifiedName	CodeTable:Na...	CodeTable:Alias
1	Case - Categories	Case Category	Division
2	Case - Categories	Case Category	Division
3	Case - Categories	Case Category	Division
4	Case - Categories	Case Category	Division

[10] The optional elements **Case:InitiationReason**, **CaseNumberOfParties**, **Case:UnnamedParties**, **Case:DisputeDate**, **Case:Year**, **Case:Venue**, **Case:VenueZipCode**, **Case:ComplexLitigation**, and **Case:ComplexLitigationFactors** are not supported under the **Case:New** element.

Ignore the Division value in the **CodeTable:Alias**. The term is legacy terminology from Filing 01.

The following graphic shows the table information for a single *case category*.



CodeTable:**Codes**
CodeTable:**Code**

Code:**Name**	Civil - Limited
Code:**Value**	CL

IMPORTANT: Filing applications must use the **Code:Name** value to populate the Filing XML's **Case:Category** element.

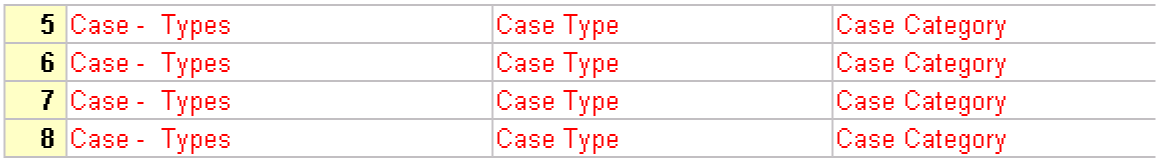
5.1.2. Case Type

The **Case:Type** element maps to the CCMS *case type*. CCMS support a number of *case types* for each CCMS *case category*. The list of *case types* for *civil limited* and *civil unlimited* are similar, but not the same. The list of *case types* for *probate* are not similar to *civil limited* or *civil unlimited*. There is only one *case type* for *small claims*, which is Small Claims.

The CCMS *case types* are standard statewide and do not vary from court-to-court in CCMS implementations.[11] The *case type* values are included in every court's Policy XML.

Like the CCMS *case category* values, the CCMS *case type* values are mapped to a *court key* inside a court's Policy XML in a **Policy:Table** element. Again, depending on the CCMS *case categories* implemented by a court, there may be more than one **Policy:Table** element for *case types*, since each CCMS case category must map to at least one Policy XML *court key* (and could map to more than one *court key*).

The following table shows the grid view of Policy XML for a court that supports more than one CCMS *case type*. The **CodeTable:Name** element value maps to the CCMS name. The **CodeTable:Mapping**, which is not shown in the diagram below, provides an XPath mapping to the Filing XML element (i.e., **Case:Type** in this situation).



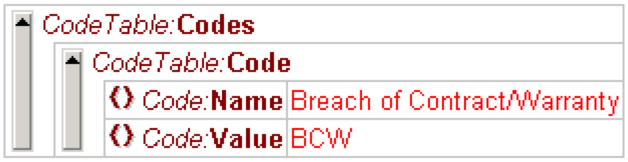
5	Case - Types	Case Type	Case Category
6	Case - Types	Case Type	Case Category
7	Case - Types	Case Type	Case Category
8	Case - Types	Case Type	Case Category

[11] This does not apply to non-CCMS implementations.

Ignore the `Case Category` value in the **CodeTable:Alias**. The term is legacy terminology from Filing 01.

CCMS creates *court keys* using a combination of the *case category* and *case type* values. As a result, each *case type* table will include only a single value, although it may map to more than one *court key* (because there can be multiple locations).

The following graphic illustrates a *case type* table.



CodeTable:**Codes**
CodeTable:**Code**
Code:**Name** Breach of Contract/Warranty
Code:**Value** BCW

IMPORTANT: Filing applications must use the **Code:Name** value (e.g. `Breach of Contract/Warranty`) to populate the Filing XML's **Case:Type** element.

5.1.3. Case Title

CCMS supports the **Case:Title** element as stated in the 2GEFS specifications.[12] CCMS exposes the submitted *case title* data in the Clerk Review interface. The clerk may change the *case title* or CCMS may automatically generate a case title. The system returns the court-accepted *case title*, which may be different than the submitted *case title*, in the Asynchronous Confirmation XML.

5.1.4. Case Jurisdictional Amount

CCMS supports the **Case:JurisdictionalAmount** element as stated in the 2GEFS specifications.[13] CCMS exposes the submitted *case jurisdictional amount* data in the Clerk Review interface. The clerk may change the *case jurisdictional amount*. The system returns the court-accepted *case jurisdictional amount*, which may be different from the submitted *case jurisdictional amount*, in the Asynchronous Confirmation XML.

5.2. Asynchronous Confirmation for Initial Flings

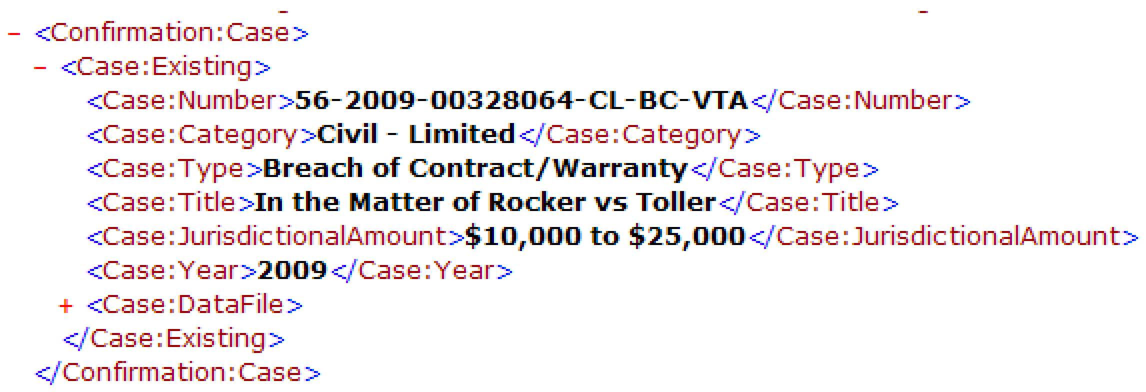
The CCMS Asynchronous Confirmation XML for ***accepted*** *initial filings* returns the following elements in the **Case:Existing** element:

- **Case:Number**
- **Case:Category**

[12] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/Case/01/#Title.
[13] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/Case/01/#JurisdictionalAmount.

- **Case:Type**
- **Case:Title**
- **Case:JurisdictionalAmount**
- **Case:Year**

The following diagram illustrates the returned values:

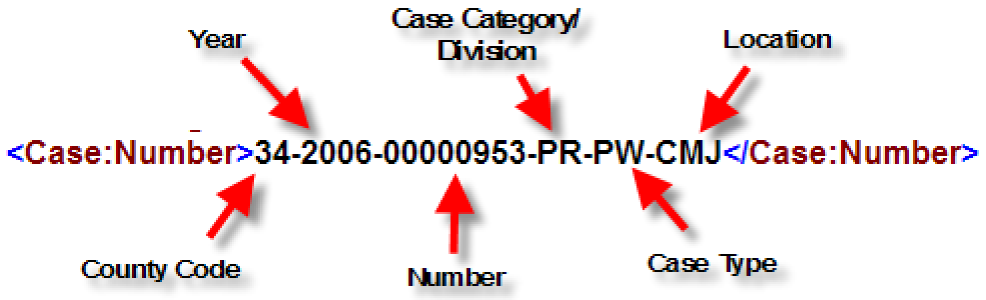


```
- <Confirmation:Case>
  - <Case:Existing>
      <Case:Number>56-2009-00328064-CL-BC-VTA</Case:Number>
      <Case:Category>Civil - Limited</Case:Category>
      <Case:Type>Breach of Contract/Warranty</Case:Type>
      <Case:Title>In the Matter of Rocker vs Toller</Case:Title>
      <Case:JurisdictionalAmount>$10,000 to $25,000</Case:JurisdictionalAmount>
      <Case:Year>2009</Case:Year>
    + <Case:DataFile>
    </Case:Existing>
  </Confirmation:Case>
```

The following sections describe the elements that CCMS returns.

5.2.1. Case Number

The **Case:Number** element includes a new *case number* assigned by CCMS to the case. The format for a CCMS *case number* is shown below:



The first part of the *case number* represents the *county code* in which the court resides. This value maps to the **CourtDetails:CourtCode** value in the court's Policy XML. The second part of the *case number* is the *year* in which the case is initiated. This maps to the **Case:Year** element returned in the Asynchronous Confirmation XML. The third part of the *case number* is a unique and sequential number assigned to cases in the court.

The fourth part is the abbreviated value for the CCMS *case category*. The fully-spelled value maps to the **Case:Category** element. The abbreviated value maps to the **Code:Value** element in the court's Policy XML along side the **Case:Name**, which includes the fully spelled value. See Section 5.1.1 for more information.

The fifth part is the abbreviated value for the CCMS *case type*. This value maps to the **Case:Type** element. See Section 5.1.2. for more information.

The sixth part is the abbreviated value for the court division's *location*. This maps to the value in the **CourtDetails:CourtLocation** element in Filing XML and Policy XML.

5.2.2. Case Category and Case Type

The CCMS *case category* and the CCMS *case type* values are editable in the CCMS Clerk Review. If the values change, then the corresponding *court key* will also change in the Asynchronous Confirmation XML. However, the *message identifier*, which includes a *court key* value, remains the same.

5.2.3. Case Title

The CCMS Clerk Review assigns the **Case:Title** value and may change the value submitted in the Filing XML. As a result, it is likely that the **Case:Title** value in the Asynchronous Confirmation XML will be different than the *case title* submitted by the filing application. If the *case title* changes, then this signifies to the filing application that the court has entered a different value into CCMS. The filing application should change the *case title* value in its system accordingly and submit to the court the court's *case title* value in future subsequent filings.

5.2.4. Jurisdictional Amount

The CCMS Clerk Review allows the clerk to edit the *jurisdictional amount* value. As a result, the value in the Asynchronous Confirmation XML may be different than the *jurisdictional amount* submitted by the filing application. The filing application should change the *jurisdictional amount* value in its system accordingly and submit to the court the court's *jurisdictional amount* value in future subsequent filings.

5.2.5. Case Year

CCMS returns the **Case:Year** element with a value equal to the year in which the case is initiated.

5.2.6. Case Type Plug-Ins

See section 10 for information about case type plug-ins.

5.3. Subsequent Filings

A *subsequent filing* is Filing XML that uses the **Case:Existing** element and its children elements. CCMS requires the following element in a *subsequent filing*:

- **Case:Number**

The **Case:Title** element is supported and encouraged, but optional, in both CCMS and 2GEFS.

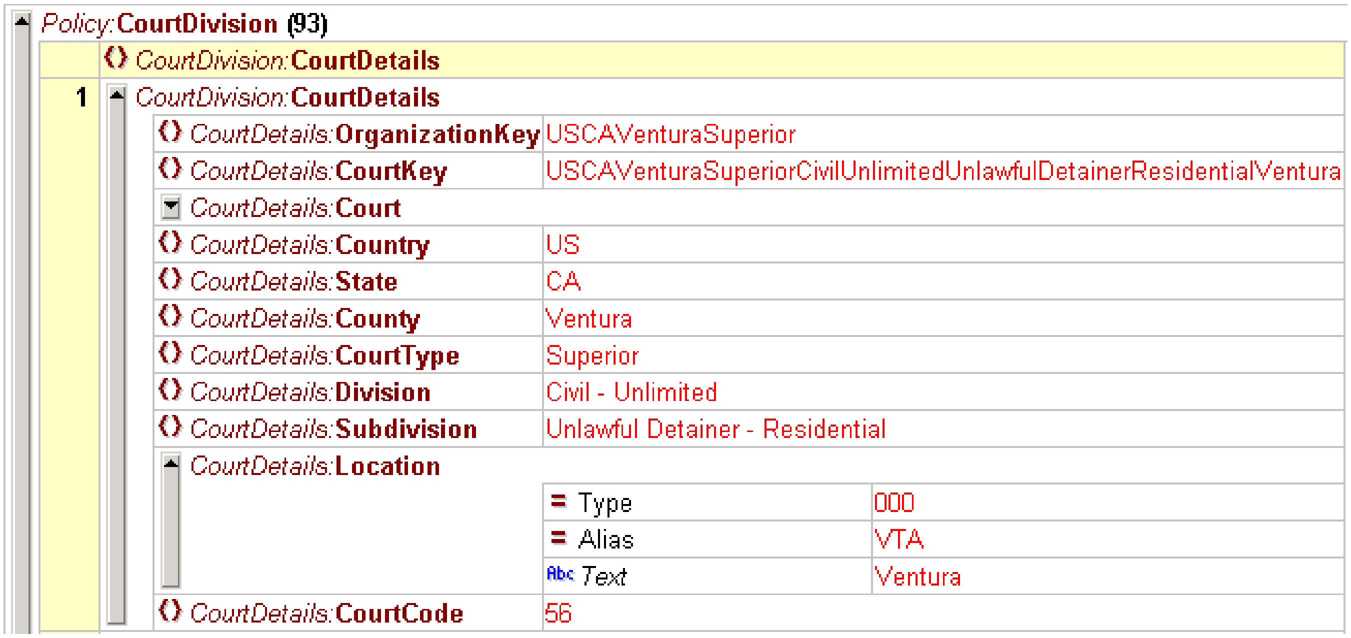
5.4. Asynchronous Confirmation for Subsequent Filings

Asynchronous Confirmation XML for *subsequent filings* returns the same elements returned in initial filings. See Section 5.2 for more information.

6. Court Details

CCMS implements the **Filing:CourtDetails** element and its children elements in Filing XML, Synchronous Confirmation XML, and Asynchronous Confirmation XML according to the 2GEFS specifications.[14]

The following illustrates the court details for a single division of a court:



Policy:CourtDivision (93)

CourtDivision:CourtDetails

1 CourtDivision:CourtDetails

CourtDetails:OrganizationKey	USCAVenturaSuperior	
CourtDetails:CourtKey	USCAVenturaSuperiorCivilUnlimitedUnlawfulDetainerResidentialVentura	
CourtDetails:Court		
CourtDetails:Country	US	
CourtDetails:State	CA	
CourtDetails:County	Ventura	
CourtDetails:CourtType	Superior	
CourtDetails:Division	Civil - Unlimited	
CourtDetails:Subdivision	Unlawful Detainer - Residential	
CourtDetails:Location		
	= Type	000
	= Alias	VTA
	Abc Text	Ventura
CourtDetails:CourtCode	56	

Each court is identified by an *organization key* and also by a *court code*. Each court division is identified by a *court key*.

In CCMS policies the *court key* is derived by concatenating the *organization key* and the *division*, *subdivision*, and *location* of the court, with spaces and special characters removed.

The **CourtDetails:Division** elements maps to the CCMS *case category* and the 2GEFS **Case:Category**.

The **CourtDetails:Subdivision** element maps to the CCMS *case type* and the 2GEFS **Case:Type** element.

In Asynchronous Confirmation XML, if the court details values related to the submitted *court key* change, then the **Confirmation:CourtDetails** values, including the *court key* also change. However, the **Header:MessageIdentifier** stays the same and includes the original

[14] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/#CourtDetails and http://www.xmllegal.org/Schema/Court/US/California/2GEFS/BuildingBlocks/Primitives/CourtDetails/03/.

court key. For example, if the court changes the *case category* from `Civil - Limited` to `Civil - Unlimited`, the Asynchronous Confirmation XML will include a different court key value and different a different value for **CourtDetails:Division**.

7. Parties

This section explains how to convey information to CCMS about *filers* and *parties* in Filing XML. This section also explains the information that is returned by CCMS to filing applications in Asynchronous Confirmation XML. In this section, the term *filer* is defined as a *person* or *organization* included in the **Filing:Filers** element. The term *parties* is defined as *people* and *organizations* included in the **Filing:People** and **Filng:Organizations** elements.[15]

Information that can be passed to CCMS about *filers* and other *parties* includes:

- Names
- Alternate Names (aliases)
- Addresses
- Phone Numbers
- Email Addresses
- Roles
- Identifiers

Descriptions of people are not supported by CCMS.

7.1. Filers and Parties

7.1.1. Filers and Parties Generally

2GEFS supports sending *filer* and *party* information in three elements:

- **Filing:Filers**
- **Filing:People**
- **Filing:Organizations**

The general 2GEFS rule is that the **Filing:Filers** element must include at least one *person* or *organization* that is responsible for submitting the filing.[16] Beyond the general rule, 2GEFS is flexible about the *role* of the *filer*, stating:

> **[30]** Specific courts and jurisdictions should define business rules for this element. There are a number of options for the meaning: (a) **Filing:Filer** could be the exact person who files the document, such as a legal secretary or a lawyer, (b) **Filing:Filer** could be a lawyer, even though the actual person who clicks the button and files the

[15] Note that the CCMS rules allow for other *filers* to be derived from the **Filing:People** and **Filing:Organizations** elements, as described in Section 7.1. CCMS uses the term "participants" or "case participants" to encompass people and organizations associated with a case.

[16] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/#Filer.

document is a lawyer's legal secretary, (c) **Filing:Filer** could be the firm for which a lawyer works, even though a lawyer or a lawyer's legal secretary actually clicks the button and files the document or (d) **Filing:Filer** could be the party on whose behalf a lawyer or law firm files a document.[17]

IMPORTANT: CCMS only recognizes only ***one*** *person* or *organization* in the **Filing:Filer** element. Other *filers*, *attorneys*, and *parties* must be included **Filing:People** and **Filing:Organizations** elements. A *filer* is a named party (e.g., person or organization named on the face of a filed document in the caption). A *filer* is the same as a *filed by* party, documented below. While there can be more than one *filer* associated with a filing, application must only use one *filer* in the **Filing:Filer** element. It does not matter which *filer* appears in the **Filing:Filer** element.

For example, if `Mary Kay` (`Plaintiff`) were represented by `John Jones` (`Attorney`) and `Mary Kay` were suing `ABC, Inc.` (`Defendant`), then `Mary Kay` would be included in the **Filing:Filer** element, `John Jones` would be included in the **Filing:People** element, and `ABC, Inc.` would be included in the **Filing:Organizations** element.

IMPORTANT: CCMS requires that all *filers*, *people*, and *organizations* included in the Filing XML be parties, or representatives of parties, to the case. It is not possible to include information about other non-parties. See Section 2.2. for more information about sending service provider name and contact information.

7.1.2 Filed By and Refers To

IMPORTANT: On *initial filings*, the CCMS implementation requires that the filer associate every party (e.g., **Filing:Filers**, **Filing:People**, **Filing:Organzation**) to the *lead document* using the **Document:FiledBy** or **Document:RefersTo** elements. Failure to include this information on a lead document will result either in the filing being rejected at the EFM or the party will not appear to the clerk in the CCMS application.

For *subsequent filings*, the **Document:FiledBy** or **Document:RefersTo** elements are optional, but preferred.

The **Document:FiledBy** element must be populated with the **Role:Qualfier** value of *filers* and *parties* that are responsible for filing the document. For example, if the plaintiff files the document, then the **Document:FiledBy** element must be populated with the plaintiff's **Role:Qualifier** value and, if applicable, the plaintiff's attorney's **Role:Qualifier** value.

[17] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/#Filer.

The **Document:RefersTo** element must be populated with the **Role:Qualifier** of the parties to which the document refers. For example, if the `Plaintiff` files a document and is the *filed by party*, then the `Defendant` would be the *refers to party*. Similarly, if the `Defendant` filed the document, then the `Defendant` would be the *filed by party* and the `Plaintiff` would be the *refers to party*.

On an *initial filing*, CCMS assumes that any attorneys included in **Filing:People** or **Filing:Organization** elements are associated with the party filing the document. On *initial filing*, do not include attorneys who are not filing the document, even if they are known. For example, if the `Plaintiff` files a document then include the `Plaintiff’s Attorney`, but do not include the `Defendant’s Attorney`.

In an *initial filing*, the CCMS Clerk Review reads the *role* of the ***first*** **Filing:Filer** person or organization. If there are other roles that match the *first filer’s role*, then the other parties are deemed to be CCMS *filed by parties* as well for all of the documents in the filing. In CCMS, the filer, plus other parties with the filer’s role, are deemed to be the *filed by parties* with respect to all documents in the filing. For example, if the *first filer’s role* is `Plaintiff`, then other `Plaintiffs` in the **Filing:People** and **Filing:Organizatios** elements are also deemed to be CCMS *filed by parties* for all of the documents in the filing.

For *subsequent filings*, do not include parties in a filing that are known to be associated with the case, ***unless*** (a) the party is needed as a *filed by* or *refers to party* or (b) the filer wishes to update contact details (such as address, phone, email address) for an existing party. In situation (a), include only the name, role, and CCMS ID of the party (i.e., do not include contact details). (See next section for more information about the CCMS ID.)

7.1.3. CCMS ID

The **Person** and **Organization** elements include attributes ***PersonKey*** and ***OrganizationKey***, respectively. In CCMS, these attributes must be used to convey the *CCMS ID*. The *CCMS ID* is a unique identifier generated by CCMS to identify a party within the context of a case. CCMS IDs are unique within a single court. CCMS IDs are not unique across courts.

It is possible in CCMS, and desired by the court, for a single CCMS ID to represent a single party on multiple cases. For example, it is possible and preferred for `Jason Jones` to have a single CCMS ID used for every case in which `Jason Jones` appears (assuming, of course, that Jason Jones is the same person).

It is also possible in CCMS for the same party to have multiple CCMS IDs. This occurs when the party appears in multiple cases and when the filer or the clerk is unable to associate the party with an existing CCMS ID. This is a practical necessity, but it not desired by the court, because it leads to duplicate parties in the courts CMS (e.g., `Jason Jones` might have 20 CCMS ID, when he is the same person).

Service providers should mitigate the latter issue by using existing CCMS IDs whenever possible. The next sections provide guidelines for using the CCMS ID.

7.1.3.1. Initial Filings and CCMS ID

On *initial filings*, the `Plaintiff` may not have a CCMS ID (or the CCMS ID may not be known to the service provider). In this situation, the service provider may send the `Plaintiff` without a CCMS ID. On the other hand, the `Attorney` representing the `Plaintiff`, especially if he/she is an existing service provider customer, should have a CCMS ID. Service providers are encouraged to store and re-use CCMS IDs from existing cases for use on initial filings of new cases. This is true for any party for which a CCMS ID is known.

7.1.3.2. Subsequent Filings and CCMS ID

If a CCMS ID does not exist for a party, CCMS either (a) allows the court to match the party with the CCMS ID of an existing party in CCMS (e.g., from another case) or (b) generate a new ID. In both situations, CCMS sends the resulting CCMS ID to the service provider in the Asynchronous Confirmation XML. Service providers must record and use the CCMS ID for subsequent filings in the same case.

7.1.3.3. CCMS IDs in Different Courts

CCMS can harmonize CCMS IDs across cases in the same court. However, CCMS does not harmonize CCMS IDs across courts. As a result, it is likely that `Ventura` and `Sacramento` will have different CCMS ID for the same person or organization. Do not use a CCMS ID from one court in a filing to another court.

7.2. Roles and Role Associations

The **Role:Name** element is required for all *people* and *organizations* in the **Filng:Filer**, **Filing:People**, and **Filing:Organizations** elements.

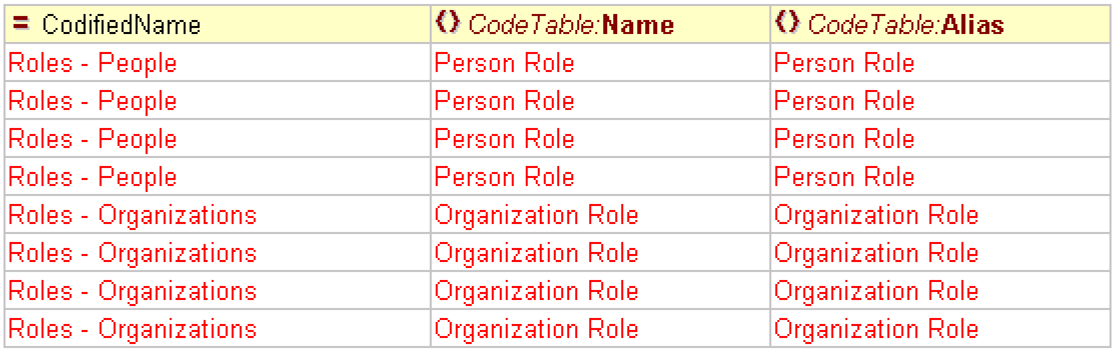
CCMS *role* values are standard statewide and do not vary from court-to-court in CCMS implementations.[18] *Roles* vary based on the CCMS *case category*. (See Section 5.1.1 for more information.) *Role* values are the same for *civil unlimited* and *civil limited*. *Role* values are different for *probate* and *small claims*. *Role* values are the same for *people* and

[18] This does not apply to non-CCMS implementations.

organizations, regardless of the CCMS *case category*. *Role* values are included in every court's Policy XML.

The CCMS *role* values are mapped to a *court key* inside a court's Policy XML in a **Policy:Table** element. Depending on the CCMS *case categories* implemented by a court, there may be more than one **Policy:Table** element for *roles*, since each CCMS *case category* and *case type* combination must map to at least one Policy XML *court key* (and could map to more than one *court key*).[19]

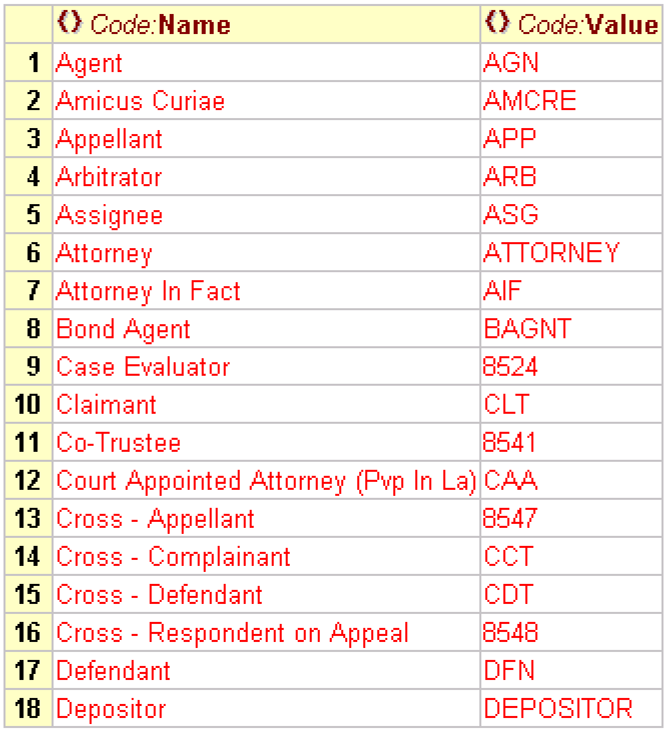
The following table shows the grid view of Policy XML for a court that supports more than one CCMS *case category*. The **CodeTable:Name** element value maps to the CCMS name. The **CodeTable:Alias** element value maps to the 2GEFS name. The **CodeTable:Mapping**, which is not shown in the diagram below, provides an XPath mapping to the Filing XML element (i.e., **Role:Name** in this situation).



= CodifiedName	CodeTable:**Name**	CodeTable:**Alias**
Roles - People	Person Role	Person Role
Roles - People	Person Role	Person Role
Roles - People	Person Role	Person Role
Roles - People	Person Role	Person Role
Roles - Organizations	Organization Role	Organization Role
Roles - Organizations	Organization Role	Organization Role
Roles - Organizations	Organization Role	Organization Role
Roles - Organizations	Organization Role	Organization Role

The following graphic shows partial table information for *roles* for a *case category*.

[19] For additional information, please submit questions to the author.



	Code:Name	Code:Value
1	Agent	AGN
2	Amicus Curiae	AMCRE
3	Appellant	APP
4	Arbitrator	ARB
5	Assignee	ASG
6	Attorney	ATTORNEY
7	Attorney In Fact	AIF
8	Bond Agent	BAGNT
9	Case Evaluator	8524
10	Claimant	CLT
11	Co-Trustee	8541
12	Court Appointed Attorney (Pvp In La)	CAA
13	Cross - Appellant	8547
14	Cross - Complainant	CCT
15	Cross - Defendant	CDT
16	Cross - Respondent on Appeal	8548
17	Defendant	DFN
18	Depositor	DEPOSITOR

IMPORTANT: Filing applications must use the **Code:Name** value (e.g. `Agent`) to populate the Filing XML's **Role:Name** element.

CCMS electronic filing does not support *role associations*.[20] The 2GEFS **Role:Associations** and **Role:Association** elements are not supported.

7.3. Person Names

CCMS supports the following 2GEFS elements for the *person names* of *filers* and *parties*:

- **Person:FirstName**
- **Person:MiddleName**
- **Person:LastName**

CCMS requires **Person:LastName**. CCMS uses and, in some situations, requires the ***PersonKey*** attribute. See Section 7.1.3 for more information.

CCMS does not support the following 2GEFS elements:

- **Person:Title**
- **Person:Salutation**
- **Person:FullName**
- **Person:Suffix**

[20] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/BuildingBlocks/Primitives/Role/02/#Associations.

- **Person:Designation**

Note that the **Person:FullName** element is defined as *mixed content* with **Person:FirstName**, **Person:MiddleName**, and **Person:LastName** elements as optional children elements. CCMS does not support text content in the **Person:FullName** construct without the children elements present.

The CCMS Clerk Review exposes the **Person:FirstName** , **Person:MiddleName**, and **Person:LastName** (*name information*) values submitted in the Filing XML. The *name information* is editable in the CCMS Clerk Review. As a result, it is possible for the *name information* in the Asynchronous Confirmation XML to be different than the *name information* submitted by the filing application. If the *name information* changes, then this signifies to the filing application that the court has entered different values into CCMS. The filing application must change the *name information* values in the service provider system accordingly.

7.4. Organization Names

CCMS supports the following 2GEFS element and attribute for the *organization names* of *filers* and *parties*:

- **Organization:FullName**
- **Organization:Name/@*OrganizationType***

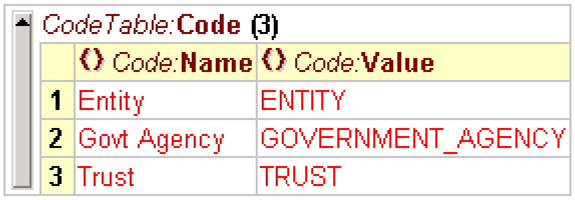
CCMS uses and, in some situations, requires the ***OrganizationKey*** attribute. See Section 7.1.3, above.

CCMS does not support the following 2GEFS elements:

- **Organization:Name**
- **Organization:AbbreviatedName**
- **Organization:Acronym**

Note that the **Organization:Name** element is defined as *mixed content* with **Organization:FullName**, **Organization:AbbreviatedName**, and **Organization:Acronym** elements as optional children elements. CCMS does not support text content in the **Organization:Name** construct without the children element **Organization:FullName** present.

CCMS requires the attribute **Organization:Name/@*OrganizationType***. The values for the attribute are included in a court's Policy XML in the `Organization Type` table. These values are standard statewide for CCMS implementations and do not vary from court-to-court. The following graphic shows the values in Policy XML grid view:



CodeTable:Code (3)

	Code:Name	Code:Value
1	Entity	ENTITY
2	Govt Agency	GOVERNMENT_AGENCY
3	Trust	TRUST

IMPORTANT: Filing applications must use the **Code:Name** value (e.g. Entity) to populate the Filing XML's **Organization:Name/@*OrganizationType*** attribute.

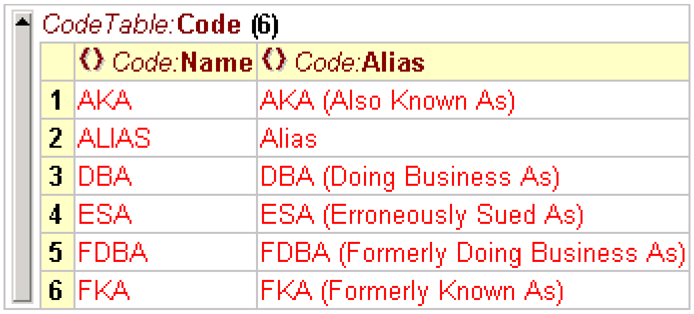
7.5. Waiver of Fees for Government Filers

CCMS Clerk Review supports the waiver of fees for government filers. To avoid the assessment of fees for a government filer, filing applications must use the Govt Agency value in the ***OrganizationType*** attribute in **Filing:Filer/Filing:Organization** element.

7.6. Alternate Names (Aliases)

CCMS electronic filing supports the 2GEFS **Person:AlternateNames** or **Organization:AlternateNames** elements. The **Person:AlternateName** and the **Organization:AlernateName** are supported in the same way as the **Person:Name** and **Organizaton:Name**, respectively. The difference is that the ***PersonAlternateName*** and ***OrganizationAlternateName*** attributes on the **AlternateName** elements must include a value from the Policy XML tables Person Alternate Names and Organization Alternate Names, respectively.

The following illustrate the Person Alternate Names table in Policy XML:



CodeTable:Code (6)

	Code:Name	Code:Alias
1	AKA	AKA (Also Known As)
2	ALIAS	Alias
3	DBA	DBA (Doing Business As)
4	ESA	ESA (Erroneously Sued As)
5	FDBA	FDBA (Formerly Doing Business As)
6	FKA	FKA (Formerly Known As)

7.7. Addresses

CCMS electronic filing supports the following 2GEFS elements and attributes for *addresses*:

2GEFS	CCMS Support	Notes

@AddressType	Required	See Schema for values.
Line	Not Supported	
Descriptor	Supported	
BuildingName	Not Supported	
StreetNumber	Supported	
StreetName	Supported	
StreetSuffix	Supported	See Policy XML for values.
Directional	Supported	
ApartmentNumber	Depreciated	Map to CCMS Unit Number
UnitNumber	Supported	Map to CCMS Unit Number
Suite	Not Supported	
POBox	Not Supported	
Suburb	Not Supported	
City	Supported	
State	Supported	See Policy XML for values.
County	Not Supported	
PostalCode	Supported	

Country	Supported	See Policy XML for values.

IMPORTANT: CCMS requires the ***AddressType*** attribute. 2GEFS enumerates values for the ***AddressType*** attribute in the schema. These values map to CCMS values as follows:

2GEFS	CCMS
Home	Home
Work	Work
Mailing	Mailing
Other	Other

IMPORTANT: The **Address:Line** element is defined as *mixed content* with a number of children elements defined as optional elements. CCMS does not support text content in the **Address:Line** construct without the children elements present.

Values for **Address:StreetSuffix**, **Address:State**, and **Address:Country** are defined in Policy XML. These values are standard statewide for CCMS implementations and do not vary from court-to-court. There is a new attribute on the **Address:State** element that allows an application to define a province. If a province is used, then do not use the Policy XML values, which are only U.S. states.

IMPORTANT: The CCMS application does not support military addresses.

7.8. Phone Numbers

CCMS electronic filing supports the following 2GEFS elements and attributes for *phone numbers*:

2GEFS	**CCMS Support**	**Notes**
PhoneType	Required	See Schema for values.

CountryCode	Not Supported	
AreaCode	Supported	
Number	Supported	
Extension	Supported	

2GEFS enumerates values for the ***PhoneType*** attribute in the schema. These values map to CCMS values as follows:

2GEFS	CCMS Support
Home	Home
Work	Work
Mobile	Cell
Pager	Pager
Fax	Fax
Other	Not Supported

7.9. Email Addresses

CCMS supports the receipt of an email address for *filers* and *parties*.

7.10. Descriptions and Identifiers

CCMS electronic filing support certains 2GEFS **Person:Identifiers** and **Organization:Identifiers** elements. CCMS does not support **Person:Descriptions**. The following identifiers are supported.

Participant	Name	Standard or Extension

Person and Organization	CCMS ID	See PersonKey and OrganizationKey in section 7.1.3.
Person	Bar Number	Standard, Identifier
Person	Bar Number State	Standard, Identifier
Person	Social Security Number	Standard, Identifier
Organization	Federal Tax ID	Standard, Identifier
Organization	Franchise Site Number	Extension, Identifier

The tables and mapping for these identifiers and descriptions are included in Policy XML in the following tables:

- Person Identifiers
- Organization Identifiers

8. Documents

This section explains how to convey documents and information about documents to CCMS. CCMS only supports Adobe PDF documents. All other document formats will be rejected.

CCMS currently supports the following elements and attribute in the 2GEFS Document schema:

- ***@DocumentSource***
- **Document:Key**
- **Document:Title**
- **Document:Type**
- **Document:Code**
- **Document:File**
- **Document:File/*@MIMEType***
- **Document:FilingStatus**
- **Document:DataFile**

8.1. Document Source

Filing applications must use the value `Filer` in the ***DocumentSource*** attribute. This signifies that the filer is responsible for filing the document. In Asynchronous Confirmation XML, the CCMS uses the value `Filer` for documents originally filed and the value `Court` for documents created and sent back by the court.

8.2. Document Title

CCMS requires the 2GEFS optional **Document:Title** element. CCMS reads the **Document:Title** element and displays the value in the Clerk Review interface.

It is possible for the clerk to edit the **Document:Title** value. As a result, the Asynchronous Confirmation XML may include a different document title value as compared to the value submitted in the filing.

8.3. Document Type, Code, Generic Type

CCMS requires the optional 2GEFS **Document:Type** element.

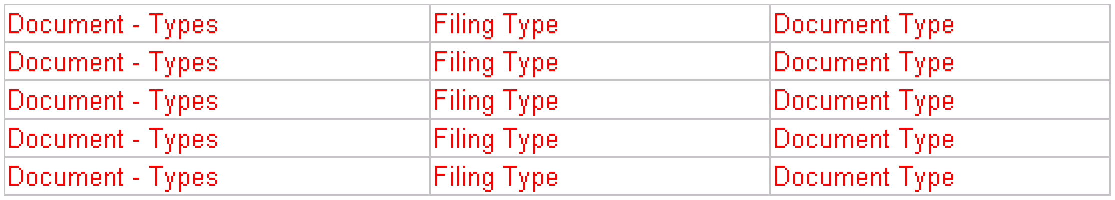
Unlike many other values in CCMS, the CCMS *document types* [21] are ***not*** standard statewide and ***vary*** from court-to-court in CCMS implementations. Like other CCMS values, *document types* vary based on the CCMS *case category* and may vary by *case type*. *Document type*

[21] CCMS uses the term *filing name* or *filing document name* for the 2GEFS term *document type*. The term *document type* is used in this document to avoid confusion with the 2GEFS term for *filing*.

values are different for *civil unlimited*, *civil limited*, *probate*, and *small claims*. *Document type* values are included in every court's Policy XML.

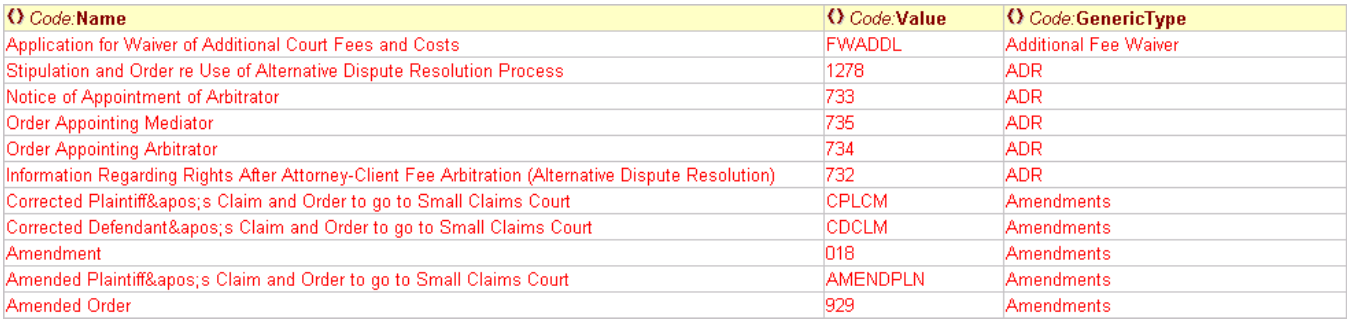
The CCMS *document type* values are mapped to a *court key* inside a court's Policy XML in a **Policy:Table** element. Depending on the CCMS *case categories* and *case types* implemented by a court, there may be more than one **Policy:Table** element for *document type*, since each CCMS *case category* and *case type* maps to at least one Policy XML *court key* (and could map to more than one *court key*).

The following table shows the grid view of Policy XML for a court that supports more than one CCMS *case category* and *case type*. The **CodeTable:Name** element value maps to the CCMS name. The **CodeTable:Alias** element value maps to the 2GEFS name. The **CodeTable:Mapping**, which is not shown in the diagram below, provides an XPath mapping to the Filing XML elements (i.e., **Document:Type** and **Document:GenericType** in this situation).



Document - Types	Filing Type	Document Type
Document - Types	Filing Type	Document Type
Document - Types	Filing Type	Document Type
Document - Types	Filing Type	Document Type
Document - Types	Filing Type	Document Type

The following graphic shows partial table information for *document types* for a *case category* and *case type*.



⟨⟩ Code:Name	⟨⟩ Code:Value	⟨⟩ Code:GenericType
Application for Waiver of Additional Court Fees and Costs	FWADDL	Additional Fee Waiver
Stipulation and Order re Use of Alternative Dispute Resolution Process	1278	ADR
Notice of Appointment of Arbitrator	733	ADR
Order Appointing Mediator	735	ADR
Order Appointing Arbitrator	734	ADR
Information Regarding Rights After Attorney-Client Fee Arbitration (Alternative Dispute Resolution)	732	ADR
Corrected Plaintiff&apos;s Claim and Order to go to Small Claims Court	CPLCM	Amendments
Corrected Defendant&apos;s Claim and Order to go to Small Claims Court	CDCLM	Amendments
Amendment	018	Amendments
Amended Plaintiff&apos;s Claim and Order to go to Small Claims Court	AMENDPLN	Amendments
Amended Order	929	Amendments

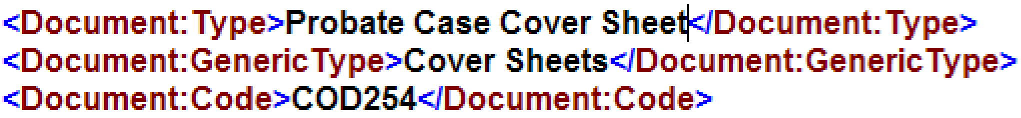
IMPORTANT: Unlike other CCMS Policy XML tables, notice that the table also includes a **Code:GenericType** element. In the 2GEFS, Policy XML maps to Filing XML documents as follows:

Policy XML	Filing XML
Code:Name	**Document:Type**

Code:Value	**Document:Code**
Code:GenericType	**Document:GenericType**

IMPORTANT: Filing applications must use the **Code:Name** value (e.g. `Fee Waiver – Application for Waiver of Additional Court Fees`) to populate the Filing XML's **Document:Type** element and the **Code:Value** (e..g, FWADDL) in the **Document:Code** element.

For example, the CCMS supports the following:



```
<Document:Type>Probate Case Cover Sheet</Document:Type>
<Document:GenericType>Cover Sheets</Document:GenericType>
<Document:Code>COD254</Document:Code>
```

The *generic type* information is useful to filing application because, among other things, CCMS often contains several hundred *document types* for a given *case category* and *case type* combination. The *generic type* allows a filing application to understand the CCMS groupings (called *filing types*). CCMS *filing types* (aka 2GEFS *generic types* for *document types*) are standard statewide and do not vary from court-to-court in CCMS implementations.

CCMS *filing types* vary by CCMS *case category*. For example, the *filing types* for *probate* are different than the *filing types* for *small claims*.

The CCMS *filing types* `Case Initiation` and `First Paper` are significant. The `Case Initiation` *filing types* must be used for the *lead document* of *initial filings*. The `First Paper` filing types must be used for *subsequent filings* when a party files a document for the first time into the case.

It is possible for the clerk to change the values for **Document:Type**, **Document:GenericType**, and **Document:Code**. The CCMS Asynchronous Confirmation XML reflects the changes made by the clerk.

8.4. Lead Documents and Attachments

CCMS requires a single **Filing:LeadDocument** element and supports **Filing:Coverssheet** and multiple **Filing:Attachment** elements. This section explains details about how documents are processed. See Section 9 for more information about the relationship of documents and fees.

8.4.1. Document Key

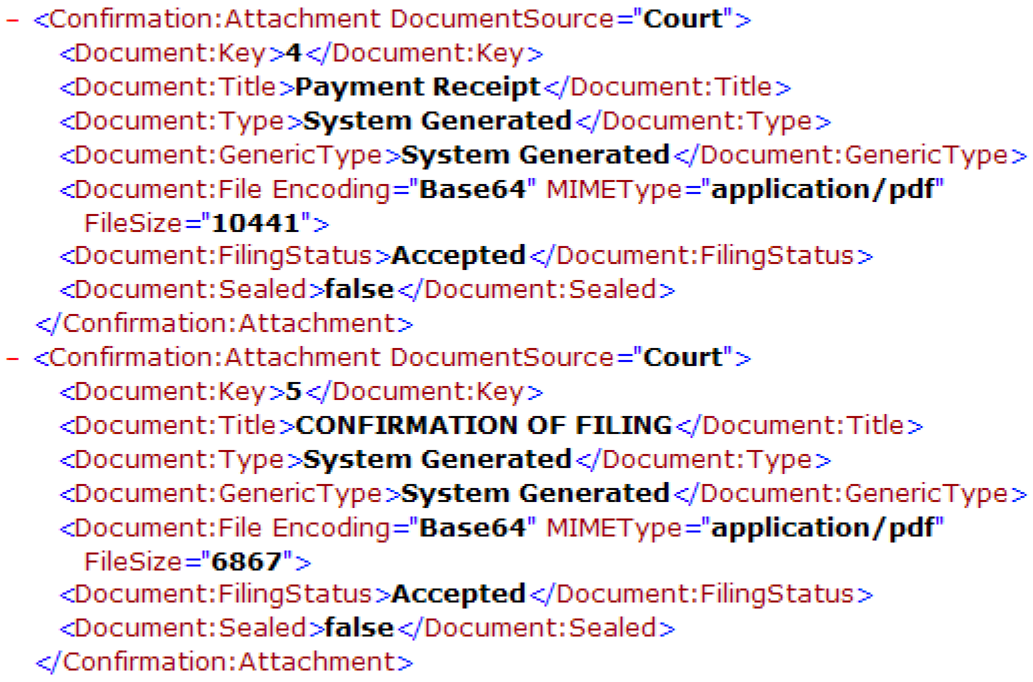
IMPORTANT: 2GEFS requires a **Document:Key** element value.[22] CCMS reads the filing application's **Document:Key** value and returns it in Asynchronous Confirmation XML. CCMS mirrors the filing application's **Document:Key** in the Asynchronous Confirmation XML **Document:Key** element for each lead document and attachment.

Court generated documents (identified by the ***DocumentSource*** attribute, described above) include a **Document:Key** value generated by CCMS.

8.4.2. Court Initiated PDF Confirmation of Filing and Other Documents

All Asynchronous Confirmation XML includes a court initiated PDF confirmation of filing. This additional PDF document will always be included as a **Confirmation:Attachment** element after other attachments that were submitted in the original filing. The **Document:Title** value for the PDF confirmation of filing is CONFIRMATION OF FILING. The **Document:Key** value for the PDF confirmation will be a string generated by the court unique within the Confirmation XML, but otherwise with no meaning to CCMS.

CCMS also returns a Payment Receipt if fees and payments are associated with the filing. The following illustrates the confirmation of filing and payment receipt:



```
- <Confirmation:Attachment DocumentSource="Court">
    <Document:Key>4</Document:Key>
    <Document:Title>Payment Receipt</Document:Title>
    <Document:Type>System Generated</Document:Type>
    <Document:GenericType>System Generated</Document:GenericType>
    <Document:File Encoding="Base64" MIMEType="application/pdf"
      FileSize="10441">
    <Document:FilingStatus>Accepted</Document:FilingStatus>
    <Document:Sealed>false</Document:Sealed>
  </Confirmation:Attachment>
- <Confirmation:Attachment DocumentSource="Court">
    <Document:Key>5</Document:Key>
    <Document:Title>CONFIRMATION OF FILING</Document:Title>
    <Document:Type>System Generated</Document:Type>
    <Document:GenericType>System Generated</Document:GenericType>
    <Document:File Encoding="Base64" MIMEType="application/pdf"
      FileSize="6867">
    <Document:FilingStatus>Accepted</Document:FilingStatus>
    <Document:Sealed>false</Document:Sealed>
  </Confirmation:Attachment>
```

[22] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/02/Document/01/#Key.

It is possible for the court clerk to return other court initiated documents. These documents will be sent as attachments after attachments that were submitted in the original filing.

8.4.3. Partial Acceptance and Rejection

IMPORTANT: CCMS supports *partial acceptance* and *partial rejection*.

2GEFS requires the **Document:FilingStatus** element in situations where there has been a partial acceptance and rejection of documents in a filing. 2GEFS does not require **Document:FilingStatus** element in situations where there has been a full acceptance. CCMS, however, includes **Document:FilingStatus** in all situations.

8.4.4. Splitting and Merging Documents

CCMS does not split or merge documents. If two documents are improperly merged or if a document is improperly split into two documents, then the clerk will reject the fling.

8.5. Coversheet

CCMS supports the **Filing:Coversheet** element. Civil case cover sheets and other similar document types should be submitted in the **Filing:Coversheet** element.

8.6. Fee Waivers

CCMS supports applications for fee waivers. To submit an application for fee waiver in an *initial filing*, the filer must file a fee waiver as an *attachment* using a document type from the `Initial Fee Waiver` document generic type in Policy XML. Using a document type from the `Additional Fee Waiver` document generic type in Policy XML will not result in a request for fee waiver.

Filers may also file fee waivers in *subsequent filings* using a document type from the `Initial Fee Waiver` document generic type in Policy XML. Unlike initial filings, in *subsequent filings*, the fee waiver document may be either the *lead document* or an *attachment*.

8.7. Document Type Plug-Ins

See section 10 for information about document type plug-ins.

9. Fees and Payments

9.1. Payments

2GEFS supports two payment methods (a) *offline ACH payments* and (b) *credit card payments*. Under the *ACH payment* method, service providers submit filings to CCMS. When the filing is accepted, the service provider receives an asynchronous confirmation. After receipt of the asynchronous confirmation, the service provider must pay the court the fees due for the filing.

Using ACH, service providers deposit funds into the court's bank account in amounts that represent a lump sum payment for a single day or a range of days (i.e., there are not be a separate deposit for each transaction). For example, if 10 filings, each with a $100 fee, are accepted on January 10, 2007, then the service provider is responsible for depositing a lump sum of $1,000 into the court's account for that day's transactions.

In addition to the ACH deposit, the service provider is responsible for submitting a *reconciliation report* to the court that accounts for the transactions represented by the lump sum deposit.

Under the *credit card payment* method, the service provider receives a credit card number from the filer (or may also use the service provider's credit card) and passes the credit card information to CCMS in Filing XML. CCMS, thereafter, uses the credit card information to authorize and settle funds from the credit card. Under this method, funds are deposited directly into the court's bank account. As a result, unlike ACH, the service provider is not required to submit *reconciliation reports* to the court. Only the last four digits of the credit card number are returned in both Synchronous Confirmatoin XML and Asynchronous Confirmation XML.

9.2 Fees

CCMS accepts multiple fee-generating documents per Filing XML. There should be one **Filing:Fee** element in the **Filing:Fees** element for every fee-generating document in the Filing XML.

CCMS Policy XML includes a fee schedule in the **Policy:Fees** element. Each CCMS fee in the schedule is associated with at least one *court key* value.[23] For CCMS Policy XML that supports more than one CCMS *case category* and *case type*, it is possible that fee names will be duplicated. Fee names that are duplicated are associated with different *court keys* because

[23] Some fees are associated with multiple court keys.

the fees are usually associated with different CCMS *case categories* and *case types*. This is done by design and is consistent with the 2GEFS specification.

Some fees have the same name and the same association to court keys, but differ by amount. The reason for this difference is usually because one fee amount applies to first papers while another fee amount does not. For example, the fee for filing a motion that is a party's first paper might be $100, whereas the fee for filing a motion that is not a party's first paper might be $40.

In addition to a *court key* association, each fee is associated with one or more *document types*. This *fee-to-document-type association* allows a filing application to create a fee calculator. The fee calculator mirrors the fee calculation logic in CCMS, thus making the determination of fees more accurate.

For example, suppose the filer elects to file a `Petition for Change of Name`. The *document type* `Petition for Change of Name` might be associated in the Policy XML fee schedule with the fee `26 - Complaint or other 1st paper`, at a cost of `$320`. The *fee-to-document-type association* is the same association in CCMS. As a result, it is possible to calculate the same fees in the filing application as the fees calculated in CCMS. This logic, however, does not take into consideration first papers.

Some CCMS *document types* are associated with more than one fee *name*. In most cases, the fee name includes in it a human-readable *jurisdictional amount*, such as `Less Than $10,000`. In this situation, by knowing the *document type*, it is possible for an application to narrow the list of fees to a handful of fees (usually two or three). The filer must know the *jurisdictional amount* and select the appropriate fee from the narrowed fee selection.

CCMS displays the fees calculated and submitted in Filing XML to the clerk. At the same time, CCMS calculates fees for the fee generating documents in the filing. The clerk uses the fees from Filing XML only for information. The CCMS-calculated fees are the fees charged to the filer and/or the service provider.

The fees returned in Asynchronous Confirmation XML are the fees ultimately accepted and charged by the court. When using ACH, service providers must use the fee information in Asynchronous Confirmation XML to calculate the lump sum amount to be deposited into the court's account using ACH, as described in the Section 9.1.

When using credit cards, CCMS supports the **Payment:NotToExceed** element. This element allows the filer to authorize a credit card charge that is up to, but not above the amount specified. This provides a buffer in situations where the CCMS-calculated fees exceed the filer-calculated fees.

10. Case and Document Type Plug-Ins

10.1. Case Type Plug-Ins

There are 16 case type plug-in schemas. Instances based on these schemas are included in the **Case:DataFile** element on case initiation for certain case categories and case type combinations. The case category/type combinations are identified in Policy XML using a court key. The plug-in schema's namespace is associated with the court key in Policy XML tables.

The case type plug-in schemas are available at:

http://www.xmllegal.org/Schema/Court/US/California/2GEFS/CCMS/DataFiles/Civil/Civil.zip

http://www.xmllegal.org/Schema/Court/US/California/2GEFS/CCMS/DataFiles/Probate/Probate.zip

10.2. Document Type Plug-Ins

There are eight document type plug-in schemas. Instances based on these schemas are included in the **Document:DataFile** element based on the case categories, case type, and filing document type (2GEFS "generic type") combinations. The case category/type combinations are identified in Policy XML using a court key. The plug-in schema's namespace is associated with the court key and the filing document type (2GEFS "generic type") in a Policy XML table.

The document type plug-in schemas are available at:

http://www.xmllegal.org/Schema/Court/US/California/2GEFS/CCMS/DataFiles/Documents/Documents.zip

10.3. General Processing Rules for Plug-Ins

The processing rules for plug-in schemas are generally the same for both case and document type plug-ins. Applications should identify the plug-in schema in Policy XML using a *court key*. For document type plug-in, the Policy XML table includes the filing name (described in Section 8.3), which must also be used to identify the schema. The schema itself is identified by a namespace. Each of the plug-in schemas has a unique namespace. Application should use the namespace to determine the XML appropriate for the **DataFile** element.

The court clerk may change the submitted data during clerk review. The Asynchronous Confirmation XML includes the data file and any information accepted into CCMS.

11. Asynchronous Confirmations

This section discusses elements in *synchronous confirmations* and *asynchronous confirmations* that are unique to Confirmation XML (i.e., elements that are not included in Filing XML and, therefore, are not mirrored from Filing XML). Elements that have an equivalent in Filing XML and Confirmation XML are discussed in other sections of this document.

11.1. Confirmation Filing Status

CCMS supports the following **Confirmation:FilingStatus** values:

- `Received` (*synchronous confirmation* only)
- `Error` (*synchronous confirmation* only)
- `Accepted` (*asynchronous confirmation* only)
- `Rejected` (*asynchronous confirmation* only)
- `Partial` (*asynchronous confirmation* only)
- `Updated` (*asynchronous confirmation* only)

Generally, CCMS supports only one Synchronous Confirmation XML and one Asynchronous Confirmation XML per Filing XML. Generally, there is never more than one *asynchronous confirmation*, except that it is possible in CCMS to (a) resend an existing asynchronous confirmation and (b) update an asynchronous confirmation.

Resending an asynchronous confirmation might occur when there has been an error sending the original or when there is otherwise a need for the service provider to receive the asynchronous confirmation again. The service provider should contact the court if it wishes to have an asynchronous confirmation resent. A resent asynchronous confirmation is exactly the same as the original.

Updating an asynchronous confirmation might occur when the court determines that some clerical error has taken place, such as mistyping a phone number or email address. In this situation, the clerk may resent the asynchronous confirmation with the updated information. Updated asynchronous confirmations should not change legally significant substantive information, such as whether the filing is accepted or rejected or, in a partial accept/rejection situation, whether one document is accepted or rejected.

11.2. Confirmation Number

2GEFS requires *confirmation numbers* to be *unique* and *sequential* within a court's electronic filing system.[24] CCMS *confirmation numbers* are *unique*, but are ***not*** *sequential*.

CCMS creates *unique confirmation numbers* by prefixing the CCMS internal, unique *transaction number* with the text CMSASYN_and CMSSYNC_. For example, the following values would be used in a *synchronous confirmation* and an *asynchronous confirmation*, respectively, for the same Filing XML:

- CMSASYN_411682
- CMSSYNC_411682

The latter portion of the *confirmation number* is the CCMS *transaction number* and it will always be the same for the *synchronous confirmation* and the *asynchronous confirmation* for the same *filing*.

11.3. Confirmation Dates and Times

The dates and times in Synchronous Confirmation XML generated by CCMS are a mirror of the dates and times in the Filing XML. These dates and time are not legally significant dates and times, since clerk review has not yet occurred. This is consistent with the 2GEFS specification.

In Asynchronous Confirmation XML, CCMS returns the legally significant date and time of acceptance in the **Confirmaiton:FilingDate** and **Confirmation:FilingTime** elements. As a general rule, the *acceptance date* will be the date the court received the filing from the service provider. An exception to the general rule occurs when the filing is submitted after normal court business hours. According to California Court Rules, electronic filings submitted after the court's regular business hours will be deemed to have been received on the next business day. As a result, the *acceptance date* for submissions after hours will be the next business day.

See Section 5 more information about dates and times.

11.4. Confirmation Filing Key

The **Confirmation:FilngKey** is a mirror of the values in the **Filing:Key**. CCMS implements the **Confirmation:FilingKey** consistent with the 2GEFS specification.

[24] See http://www.xmllegal.org/Schema/Court/US/California/2GEFS/Filing/01/Confirmation/01/#Number.

See Section 4 for more information.

12. Error Messages and Rejections

CCMS generates *errors* and *rejections* for *synchronous confirmations*. CCMS generates only *rejections* for *asynchronous confirmations*. An *error* is Confirmation XML with a **Confirmation:ErrorStatus** equal to `Error`. A rejection is a Confirmation XML with a **Confirmatoin:ErrorStatus** equal to `Rejected`.

12.1. Synchronous Confirmations

CCMS generates *errors* in *synchronous confirmations* only when the Filing XML is invalid against the schema. CCMS generates *rejections* when the Filing XML is valid, but when there is some element or value in the Filing XML that CCMS cannot accept.

As published by the CCMS product team, the following is list of EFM validations that should cause a *rejection* in a *synchronous confirmation*:

#	Name	Description	Filing XML	Policy XML	Error Message
1	XML	Well-Formed XML			SOAP Error Message
2	Court Code	The Court code must exist in the Court Policy file	CourtDetails:CourtCode	CourtDetails:Court Code	The CourtDetails:CourtCode value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
3	Service Provider	The Service Provider must exist in the Court Policy File	Filing:Key/Key:OrganizationKey	Policy:CodeTable/ CodeTable:Codes /CodeTable:Code	The Filing:Key/Key: OrganizationKey must include a registered service provider identifier. The incorect submitted value is “<value>”.
4	Header URI	The URI tag must not be null, and it must be pointing to a valid web address	Header:To/Header:URIs/Header:URI	Code Validation	The Header:To/.../Header:URI element value is missing or is not a valid web address.
5	Header ReplyTo	The ReplyTo tag must not be null, and it must be pointing to a valid web address	Header:ReplyTo/Header:URIs/Header:URI	Code Validation	The Header:ReplyTo/.../Header:URI element value is missing or is not a valid web address.
6	Case Type	The Case Type must be listed in the E-Filing XML.	Case:Type	Policy:CodeTable/ CodeTable:Name of Case Type	The Case:Type value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
7	Case Category	The Case Category must be listed in the E-Filing XML.	Case:Category	Policy:CodeTable/ CodeTable:Name of Case Category	The Case:Category value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
8	Case Type/Case Category Relationship	There must be a parent-child relationship between the selected Case	Case:Type and Case:Category Relationship	Code Validation (Both CodeTables for Case Type and Case Category must share a	The Case:Type value is not correctly associated with the Case Category value. The incorrect submitted values are: Case:Category: "<value>" and Case:Type: "<value>"..

		Type and the selected Case Category.		common CodeTable:CourtKey value)	
9	Person Role	Any people on the case must have a role that is listed in the Court Policy file.	Person:Roles/Person:Role/Role:Name	Policy:CodeTable/CodeTable:Name of Person Role	The Person:Role value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>"
10	Organization Role	Any organizations on the case must have a role that is listed in the Court Policy file.	Organization:Roles/Organization:Role/Role:Name	Policy:CodeTable/CodeTable:Name of Organization Role	The Organization:Role value is either absent or does not match a value in Policy XML The incorrect submitted value is "<value>"
11	Number of Lead Documents	There must be one and only one lead document.	NA	Code Validation	Only one Filing:LeadDocument element per filing is allowed. The multiple lead documents sent are “<x1>”,”<x2>”…..
12	Document Key	The Document Key should not be greater than 30 characters .	NA	Code Validation	The Document:Key cannot be greater than 30 characters. The incorrect submitted value is "<value>".
13	Lead Document Title	The lead document title must be 250 characters or less.	Filing:LeadDocument/Document:Title	Code Validation	The Lead Document Document:Title value cannot be greater than to 250 characters. The incorrect submitted value is "<value>".
14	Lead Document File Format	The lead document must be a PDF file.	Filing:LeadDocument/Document:File	Code Validation	Only PDF file type is allowed for the Lead document file type. Please correct the MIMEType attribute value on the Filing:LeadDocument element. The incorrect submitted value is "<value>".
15	Lead Document Filing Type	The lead document must have a filing type that is listed in the Court Policy file.	Filing:LeadDocument/Document:Generic Type	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document – within that element, Code:GenericType	The Document:Generic Type is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
16	Lead Document Filing Name	The lead document must have a filing name that is listed in the Court Policy file.	Filing:LeadDocument/Document:Type	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document	The Document:Type value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
17	Lead Document Filing Type and Filing Name Relationship	There must be a parent-child relationship between the lead document filing type and lead document filing name.	NA	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document. That value and its Code:GenericType value must	The Filing:LeadDocument Document:Type is not correctly associated with the Lead Document Document:GenericType. The incorrect submitted values are: Document:Type: "<value>" and Document:Generic Type: "<value>".

				match	
18	Lead document – case initiating document?	If the E-Filing is an initial filing, the lead document must be a case initiating document. If the E-Filing is a subsequent filing, the lead document must not be a case initiating document.	Filing:LeadDocument/Document:Generic Type	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document – within that element, Code:GenericType must be Case Initiation	For Initial filings - “The Lead Document must be a case initiating document. The incorrect submitted value is "<value>".” For subsequent filings - “The Lead Document may not be a case initiating document. The incorrect submitted value is "<value>".”
19	Attachments title	The attachment(s) document title must be 250 characters or less.	Filing:Attachments/Filing:Attachment/Document:Title	Code Validation	The Filing:Attachment Document:Title cannot be greater than to 250 characters. The incorrect submitted value is "<value>".
20	Attachments file format	The attachment(s) must be a PDF file.	Filing:Attachments/Filing:Attachment/Document:File	Code Validation	Only PDF file type is allowed for the Lead document file type. Please correct the MIMEType attribute value on the Filing:LeadDocument element. The incorrect submitted value is "<value>".
21	Attachment filing type	The attachment(s) must have a filing type that is listed in the Court Policy file.	Filing:Attachments/Filing:Attachment/Document:GenericType	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document – within that element, Code:GenericType	The Document:Generic Type value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
22	Attachments filing name	The attachment(s) must have a filing name that is listed in the Court Policy file.	Filing:Attachments/Filing:Attachment/Document:Type	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document	The Document:Type value is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
23	Attachments filing type and filing name relationship	There must be a parent-child relationship between the attachment filing type and attachment filing name.	NA	In Policy:CodeTable/CodeTable:Name of Filing Type, CodeTable:Code will refer to a single document. That value and its Code:GenericType value must match	The Document:Type value is not correctly associated with the Document:Generic Type value. The incorrect submitted values are: Document:Type: "<value>" and Document:GenericType: "<value>".
24	Credit card number	The credit card number must be a valid credit card number, according to the appropriate credit card	Payment:CreditCard/CreditCard:Number	Code Validation	The CreditCard:Number value is either absent or is not a valid credit card number. The incorrect submitted value ends with the last four digits XXXX.

		algorithm.			
25	Credit card type	The credit card type must be Visa, Mastercard, Discover or American Express.	NA	Code Validation	The CreditCard:Number value is a valid credit card number, but it is not a type that of card that the court supports.
26	Credit Card Security Code	The credit card security code must not be null	Payment:CreditCard/CreditCard:Security Code	Code Validation	CreditCard:SecurityCode value is missing.
27	Credit Card Security Code	The credit card security code must be a number.	Payment:CreditCard/CreditCard:Security Code	Code Validation	CreditCard:SecurityCode value must be a 3 or 4 digit numeric value. The incorrect submitted value is "<value>".
28	Credit Card Card Holder	The credit card holder address code must be sent in the incoming Filing XML.	CardHolder:Person/Person:Addresses/Person:Address	Code Validation	CreditCard:Card Holder address is not valid or missing. The incorrect submitted value is "<value>".
29	Credit Card Number	The encrypted credit card number must be a valid encrypted number	Payment:CreditCard/CreditCard:Number	Code Validation	Failed to encrypt credit card number.
30	Case Title	The Case Title should not exceed 250 characters	Case:Title	Code Validation	Case:Title cannot exceed 250 characters. The incorrect submitted value is "<value>".
31	Document Key	The Document Key should be unique within the E-Filing XML	NA	Code Validation	Document:Key is not unique for Document:Type “<value1>” and “<value2>”
33	Payment Total	The payment total must be a number.	Payment:Amount/Payment:Total	Code Validation	Payment:Total value should only contain numeric values
34	Not to Exceed Value	The ‘Not to Exceed’ value must be a number.	Payment:Amount/Payment:NotToExceed	Code Validation	Payment:NotToExceed value should only contain numeric values
35	State for Driver License	The “State” is a mandatory tag for the Driver License	Filing:Person/Person:Identifiers/Person:Identifier/Person:DriversLicense/Person:DriversLicenseState	Code Validation	The State attribute is required for the Person:DriversLicense element. The attribute value is missing from the Person:DriversLicense: “<value>”.
36	Court Location	The Court Location must exist in the Court Policy file	Filing:CourtDetails/CourtDetails:Location	Policy:CourtDetails/CourtDetails:CourtLocation	CourtDetails :Location is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".
37	Court Location	The Court Location must exist in the E-filing XML file	Filing:CourtDetails/CourtDetails:Location	Policy:CourtDetails/CourtDetails:CourtLocation	CourtDetails:Location is either absent or does not match a value in Policy XML. The incorrect submitted value is "<value>".

38	Person Name	Person Name(First Name, Middle Name or Last Name)	Filing:Filers/Filing:Filer/Filing:Person	Code Validation	Person FirstName/Middle Name/LastName exceeds 40 characters. The incorrect submitted value is "<value>".
39	Person Key	The PersonKey should be anumeric value	Filing:Filers/Filing:Filer/Filing:Person	Code Validation	The PersonKey attribute value should have only numeric values. The incorrect submitted value is "<value>".
40	Organization Type	OrganizationType value can not be PERSON	Filing:Filers/Filing:Filer/Filing:Organization	Code Validation	Attribute OrganizationType cannot have value PERSON.
41	Organization Name	Organization Name value should not exceed 250 characters	Filing:Filers/Filing:Filer/Filing:Organization	Code Validation	Organization:Name value cannot exceed 250 characters. The incorrect submitted value is "<value>".
42	Organization Key	The OrganizationKey should be anumeric value	Filing:Filers/Filing:Filer/Filing:Organization	Code Validation	The OrganizationKey attribute value should have only numeric values. The incorrect submitted value is "<value>".
43	Data File	DataFile is required for the specific case type	Filing:LeadDocuments/Filing:LeadDocument/Document:DataFile	Code Validation	A DataFile Extension is required for the specified Case Type “<value>”.
44	Data File	DataFile is not supported for the specific case type and courtkey	Filing:LeadDocuments/Filing:LeadDocument/Document:DataFile	Code Validation	The XML within the Document:DataFile element is not supported for the specific Case Type and Court Key. The offending values are: Case:Type: "<value>", CourtDetails:CourtKey: "<value>"; and Namespace "value".
45	Data File	Namespace for the data file is missing	Filing:LeadDocuments/Filing:LeadDocument/Document:DataFile	Code Validation	The XML within the Case:DataFile element is missing a namespace
46	Data File	Amended Number in the Data File should not be greater than 999	Filing:LeadDocuments/Filing:LeadDocument/Document:DataFile	Code Validation	Amended Number value in the Document:DataFile should not be greater than Number 999.

The *rejection messages* are included in the **Confirmation:Message** element. The messages are descriptive, human-readable text.

12.2. Asynchronous Confirmations

The CCMS Clerk Review allows a court clerk to *reject* a filing for various business reasons. The Clerk Review includes a drop down list that includes standard reasons for rejection as well as a free-form text box that allow the clerk to type an optional, non-standard rejection reasons. These rejection reasons will be included in the **Confirmation:Message** element, which is consistent with the 2GEFS specifications.