

# State of Arizona Multi-Vendor Electronic Case Filing Specification

## Introduction

This document describes in a summary format the application of LegalXML/Electronic Court Filing (ECF) 3.0 in the Arizona Courts' environment based on the pilot implementation in the Clerk of the Superior Court in Maricopa County, as documented in the Court Policy, High Level Design Document, and Filing Review MDE Web Service documents located at Technical Advisory Council (TAC) meeting web site:

<http://www.supreme.state.az.us/tac/Archives/FY07/060804/mat060804.htm>.

## Scope of Document

This document applies only to public to court and attorney to court electronic filing of documents related to court cases. It does not apply to any court-to-court filings or records transfers.

## Background

The Arizona courts' standard for multi-provider electronic case filing is LegalXML Electronic Court Filing (ECF) 3.0, modified as described within this document. ECF is based on national information exchange standards utilizing relevant portions on the Global Justice XML Data Model (GJXDM). The complete ECF 3.0 specification is available from <http://docs.oasis-open.org/legalxml-courtfilling/specs/ecf/v3.0/ecf-v3.0-spec-cd01.zip>. ECF 3.0 is a technical architecture that defines a set of functional features needed to accomplish electronic filing in a court, defining both normative (required) and non-normative (optional) business processes. The use of ECF 3.0 standardizes the data descriptions, definitions, and transport methods used by all vendors and other filers when communicating electronic filings to the courts.

ECF 3.0 defines and describes:

1. Message structures or specific actions in three types:
  - a. Core messages which apply to all case classifications at all courts,
  - b. Case-classification-specific messages that apply to a single supported type, and
  - c. Court-specific messages that apply only to a specific court to the exclusion of all others.
2. Messaging profile specifications defining communications infrastructures within which electronic filing transactions can take place.
3. Document signature profile specifications that define mechanisms for stating or assuring that a person signed a particular document.
4. Court Policy which specifies information about a court's practices including local variations – both human-readable and machine-readable documents are required.

When the Clerk of the Superior Court (COSC) in Maricopa County initially implemented ECF 3.0, it did so with the requirement of interfacing to existing systems. Many of the components needed to support E-Filing had already been deployed in the COSC. They did not want to duplicate or change how these systems functioned; they were looking for a common means of describing the

data as well as providing a standard method to exchange data. Also, the data transport method had to be available for both internal and external use and access.

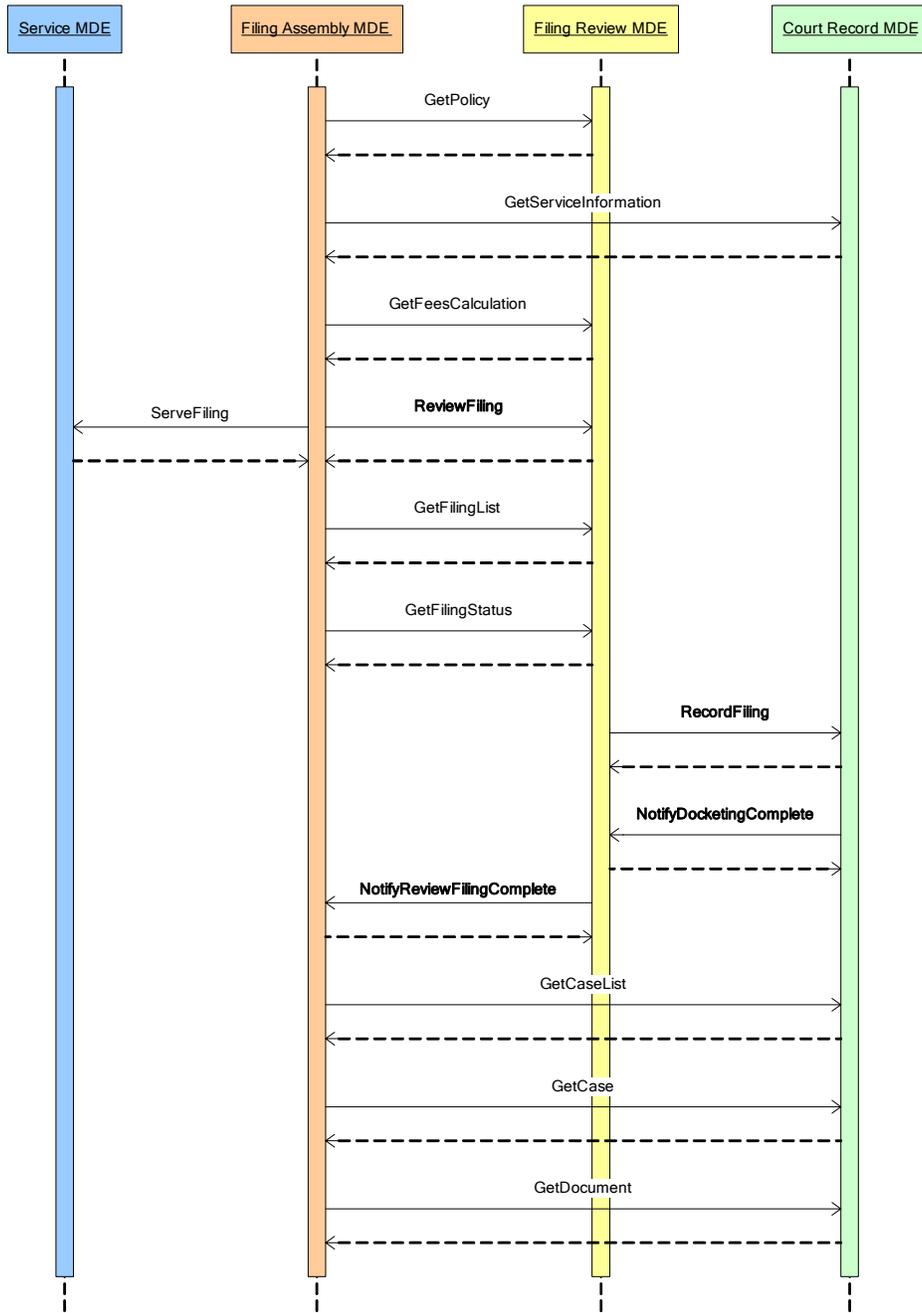
## Architecture Detail

The ECF 3.0 architecture consists of four Major Design Elements (MDEs) which support operations and messages. An MDE is a logical grouping of operations, such as the operations involved in creating a filing or the operations involved in receiving and recording a filing; that is, incorporating the constituent documents in a court document management system. A message is the data exchanged between MDEs in the form of an XML document that may include one or more additional binary attachments. These messages contain the information to be filed with the court.

The ECF 3.0 specification defines four MDEs, as follow:

- **Filing Assembly MDE** – enables a filer to create a filing message for submission to a court and for service on other parties in the case, returning the response from the court to the filer.
- **Filing Review MDE** – enables a court to receive and review a filing message and prepare the contents for recording in its case management and document management systems, sending a response concerning the filing to the Filing Assembly MDE. The Filing Review MDE also enables filers to obtain court-specific policies regarding electronic filing and to check on the status of a filing.
- **Court Record MDE** – enables a court to record electronic documents and docket entries in its case management and document management systems and returns the results to the Filing Review MDE. The Court Record MDE also enables filers to obtain service information for all parties in a case, to obtain information about cases maintained in the court's docket, register of actions and calendars, and to access documents maintained in the court's electronic records.
- **Service MDE** – enables a party to receive service electronically FROM other parties in the case. Note that service TO other parties in the case is performed by the Filing Assembly MDE.

Figure 1, below, graphically displays all MDE types contained in ECF 3.0, the methods supported by each MDE, and the methods used by each MDE to send and receive information. Bold items in the diagram are required by ECF 3.0 for compliance, but not all are currently allowed in Arizona implementations.



**Figure 1. Filing Preparation to Docketing Cycle**

(Diagram has been extracted from the ECF 3.0 specification.)

## Arizona’s Approach – Using ECF 3.0

Using ECF 3.0 standardizes the data descriptions, definitions, and transport methods used by all vendors and other filers when communicating with the courts. ECF 3.0 defines 4 MDEs, however, Arizona has elected to make some prohibited, some required, and some allowed, as noted below. Each of the MDEs is designed and developed as a Web Service. A Web Service is a program that runs on a Web Server like Microsoft IIS (ASP, ASPX), Apache, WebSphere, etc., which performs a requested action, does not have user interface (UI), and uses the HTTP protocol as the principal communications transport protocol. In addition, a Web Service uses an application-level protocol called SOAP (Simple Object Access Protocol) to package its data requests and responses. For more information about SOAP, please see: <http://www.w3.org/TR/soap>.

Arizona has decided to develop and implement the MDEs as follows:

Filing Assembly MDE	ALLOWED
Filing Review MDE	REQUIRED
Court Record MDE	REQUIRED
Service MDE	PROHIBITED since vendors provide this in the Arizona approach

Each MDE has many methods described in the specification. Arizona chooses to implement a “required” set as the multi-vendor implementation. This does not mean that the remainder will never be implemented, only that they are not needed at the present time to support the vendors and implementations. The standard will be expanded over time to describe increased functionality and this document will be updated to reflect that expansion.

### **Filing Assembly MDE**

The Filing Assembly MDE enables the submission of a filing message to the court as well as a returning response from the court to the filer. Use of the Filing Assembly MDE is not required at this time.

#### ***Filing Assembly MDE Operations***

Operation Name	Operation Purpose	AZ Disposition
NotifyReviewFilingComplete	Informs the Filing Assembly MDE of the result of a filing submitted in the ReviewFiling operation. The result will indicate whether the filing was accepted or rejected.	Disallowed

Although the operation NotifyReviewFilingComplete is disallowed, notification via e-mail is required at this time. E-mail notification must include the following data items:

**Filing Date:** MM/DD/YYYY HH:MM:SS AM/PM  
**Filing ID:** nnnn  
**Case Number:** alphanumeric  
**Filed By:** Registered filer name, full address, phone, e-mail address  
**Authorizing Attorney:** Bar number, (state abbrev), e-mail address  
**Document(s) Filed:**  
 1. Title: x / Type: [Alpha Document Type] example Motion To Quash  
**Document Status:** Accepted/Rejected  
**Clerk of Court Address:** full address  
**eFiling Support Phone:** nnn-xxx-xxxx  
**Clerk of Court Web Site:** full URL

## **Filing Review MDE**

The Filing Review MDE is Webservice designed to electronically file court documents. To accomplish this, it will accept uploaded documents related to the Court Filing and save the XML that describes the filing and the related documents.

The Filing Review MDE, at a minimum, supports subsequent filings for case types CV (Civil) and CR (Criminal). Courts are allowed to implement additional case types as well as initial filings.

Arizona's version uses the SOAP DIME (Direct Internet Message Encapsulation) protocol to upload the file in chunks. The benefit of allowing the "Chunked" transfer method is there is no limit to the number of documents or to the size of the documents that can be uploaded for the E-filing.

The Filing ReviewMDE Webservice also will accept documents as SOAP attachments. The "Attached" transfer method is limited to **16 MB** for all documents within the E-filing submission.

The FilingReviewMDE Webservice allows either or both file transfer methods to be used within an E-filing submission. Providing both "Chunked" and "Attached" transfer capabilities avoids size limitations imposed by the current Web Server applications.

### ***Filing Review MDE Operations***

<b>Operation Name</b>	<b>Operation Purpose</b>	<b>AZ Disposition</b>
Initialize	Used with the "Chunked" transfer method to initialize a buffered upload. <u>Not</u> part of the ECF 3.0 specification.	Allowed
AppendChunk	Used with the "Chunked" transfer method to append a DIME attachment to a SOAP message. <u>Not</u> part of the ECF 3.0 specification.	Allowed
RemoveInstanceManager	Used with the "Chunked" transfer method to remove the upload instance manager from session state. <u>Not</u> part of the ECF 3.0 specification.	Allowed

Operation Name	Operation Purpose	AZ Disposition
ReviewFiling	Main method used when uploading an E-Filing using “Chunked” and “Attached” transfers	Required
GetPolicy	Used to obtain the machine readable “Court Policy” describing the features supported by this implementation, the court’s code lists, and any other information a Filing Assembly MDE would need to know in order to electronically file successfully into that court	Required
GetFilingStatus	Returns the Filing status for the requested Filing ID	Required
VerifyFileHash	Can be used to verify an uploaded file has not been altered during the upload process. Though <u>not</u> part of ECF 3.0, use of SHA-2 is consistent with ECF 3.0	Allowed, use of SHA-2 required
NotifyDocketingComplete	Though required by ECF 3.0, specifies action by the CMS	Disallowed
GetFilingList	Returns a list of filings that match specified criteria such as filer ID, case number, or filing date.	Disallowed
GetFeesCalculation	Will be required before some types of initial filings are allowed	Disallowed

The development and implementation of all the above methods will be revisited and expanded as the need arises.

## **Court Record MDE**

The Court Record MDE Webservice enables the court to record electronic documents and docket entries in its case management system and document management system, not to provide case and document information electronically to E-filing vendors and their users.

### ***Court Record MDE Operations***

Operation Name	Operation Purpose	AZ Disposition
GetCase	Returns the Case information using the specified case tracking ID. The response includes the list of documents related to the Case	Required
GetDocument	Returns the requested document from the Case encoded in Base64. This encoding insures the proper transfer of the document to the requestor. The returned document must	Required

Operation Name	Operation Purpose	AZ Disposition
	be un-encoded in order for it to be viewable	
RecordFiling	Performs necessary business functions in the courts' case and document management systems such as docketing and registering documents.	Required
GetCaseList	Queries and returns a listing of cases that satisfy specified criteria such as case number, case participant, etc.	Disallowed
GetServiceInformation	Queries and returns service information for all case parties and participants in a case.	Disallowed

The development and implementation of all the above methods will be revisited and expanded as the need arises.

### **Service MDE**

The Service MDE enables a party to receive service electronically from other parties in a case. It is prohibited at this time to allow vendors to develop their own approaches in Arizona. The court is not in the business of primary service.

#### ***Service MDE Operations***

Operation Name	Operation Purpose	AZ Disposition
ServeFiling	Serves the filing to other parties in a case.	Disallowed

### **Service Interaction Profiles**

SIP Name	SIP Purpose	AZ Disposition
Web Services Profile	Defines a transmission system based on web services utilizing SOAP 1.1 over HTTP 1.1.	Required
Portable Media Profile	Defines a transmission system which utilizes portable media (e.g., CD, DVD, USB drive, etc.) to deliver the filing in electronic format to the court apart from any network connection. This SIP is intended for supplemental use only.	Allowed

### **Document Signature Profiles**

Signature Profile Name	Signature Profile Purpose	AZ Disposition
------------------------	---------------------------	----------------

<b>Signature Profile Name</b>	<b>Signature Profile Purpose</b>	<b>AZ Disposition</b>
Null Signature	Indicates the absence of a document signature.	Required
XML Document Signature	Defines a digital signature encoded in W3C XML Signature Syntax.	Disallowed
Application-Specific Document Signature	Permits applications to use innate methods for signing documents. This specification defines a mechanism for embedding an application-specific binary signature within a document. This profile supports the native capabilities in document formats such as Microsoft Word and the Adobe Portable Document Format (PDF) for describing and embedding signatures.	Disallowed
Proxy Document Signature	This specification defines a mechanism for indicating documents that are digitally signed by a court filing infrastructure component on behalf of an authenticated filer. An example could be a server that exposes its asymmetric key and digital certificate to authenticated users for signing documents.	Disallowed
Symmetric Key Document Signature	This specification defines a mechanism for indicating documents that are digitally signed by a trusted entity on behalf of the signer using a symmetric key known only to the trusted entity.	Disallowed

## **Messages and Schemas**

List – include all extensions. (This portion of the specification is still in development.)

## **Code Lists**

List – identify mandatory lists and/or values in lists. (This portion of the specification is still in development.)

## **Change Process**

Notice of intention to implement any function beyond that mandated in this document must be addressed to the Technical Advisory Council of the Commission on Technology. Revisions to this document will be approved by TAC prior to publication.