

WASHINGTON STATE ADMINISTRATIVE  
OFFICE OF THE COURTS

---

OXCI Electronic Filing Manager Implementation Project  
Post-Implementation Review

November 23, 2004

*This report documents the findings and lessons learned of the courts and vendors participating in this project as summarized by MTG Management Consultants L.L.C., and edited by the Washington Administrative Office of the Courts.*

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION .....	2
A. BACKGROUND .....	2
B. DOCUMENT ORGANIZATION .....	2
II. PROJECT SUMMARY .....	5
A. PROJECT SYNOPSIS.....	5
B. PROJECT DELIVERABLES.....	8
III. LESSONS LEARNED.....	11
A. COURT ELECTRONIC FILING LESSONS LEARNED.....	11
B. OPEN SOURCE SOFTWARE LESSONS LEARNED.....	14
C. JUSTICE XML LESSONS LEARNED .....	15
D. WEB SERVICES LESSONS LEARNED.....	17
IV. FUTURE WORK.....	21
A. IMPLEMENTATION OF THE EFM IN COURTS.....	21
B. DEVELOPMENT OF THE LEGALXML ELECTRONIC COURT FILING BLUE SPECIFICATION.....	21
C. DEVELOPMENT OF AN IMPLEMENTATION OF COURT FILING BLUE.....	22

I. INTRODUCTION

## I. INTRODUCTION

The state courts of Washington and Georgia are leading the Open XML Court Interface (OXCI) consortium of state courts in the development of an open source court electronic filing application. This middleware application is an open source implementation of an Electronic Filing Manager (EFM), that will eventually conform to the Organization for the Advancement of Structured Information Standards (OASIS) LegalXML Court Filing “Blue” standard. This document is a post-implementation review of the OXCI EFM development project, including a number of lessons learned and opportunities for future work.

### A. BACKGROUND

As the repositories of legal documents, courts are universally challenged with managing ever-increasing amounts of paper. Although electronic filing is widely viewed as a critical part of a solution to the courts’ document management problems, very few courts have the financial and technical resources to develop electronic filing systems on their own. To that end, a number of state courts together have founded the OXCI consortium in order to collaboratively produce an EFM middleware application. The OXCI EFM is intended for use by both courts and vendors as a common framework for their electronic filing implementations.

The OXCI consortium of state courts has produced a middleware implementation for electronic filing for use within all levels of state courts for the receipt, transmission, and validation of electronic filings, court orders, and associated data. The middleware provides a uniform open source implementation of an EFM, extending the LegalXML Electronic Court Filing Technical Committee (TC) standards of OASIS. This middleware has been made available nationally to courts and other vendors as the bridge between an electronic filing front-end application (provided by the court, a vendor, or a service provider) and the court’s case management system (CMS) and document management system (DMS).

### B. DOCUMENT ORGANIZATION

This document is the post-implementation review and the last of the project management deliverables for the OXCI EFM Implementation Project. The remainder of the document is organized as follows:

- Section II reviews the performance of the project with regard to the original schedule and objectives.

- Section III discusses a number of lessons learned in the project.
- Section IV describes several opportunities for future work to build on the project.

II. PROJECT SUMMARY

## II. PROJECT SUMMARY

The schedule and objectives of the project were originally defined in the Project Work Plan. This section summarizes the history and status of the project from the following perspectives:

- Project Synopsis
- Project Deliverables

### A. PROJECT SYNOPSIS

The OXCI EFM development project was officially launched on December 8, 2003, with a meeting in Olympia, Washington, of representatives from the Georgia Administrative Office of the Courts (AOC) and Washington AOC as the sponsoring courts; MTG Management Consultants, L.L.C., as the project manager; counterclaim, Inc., as the development team; and Mr. John Greacen as a member of the OXCI Steering Committee. At that meeting, the participants approved the Project Charter, including the following project description:

The OXCI consortium of state courts intends to produce a middleware implementation for electronic filing for use within all levels of state courts for the receipt, transmission, and validation of electronic filings, court orders, and associated data. The middleware will provide a uniform open source implementation of an EFM, extending the LegalXML Electronic Court Filing Technical Committee standards of the Organization for the Advancement of Structured Information Standards (hereinafter "OASIS"). It is intended that this middleware will be made available nationally to courts and other vendors as the bridge between an electronic filing front-end application (provided by the court, a vendor, or a service provider) and the court's case management system (hereinafter "CMS") and document management system (hereinafter "DMS").

The participants at the December 2003 meeting also approved the Project Work Plan, which provided a detailed list of the tasks and schedule of deliverables; agreed to use the <http://oxci.sourceforge.net> Web site to distribute the code and other deliverables; and agreed to use the Mozilla Public License (MPL) while the Georgia AOC looked into the options for licensing the OXCI EFM as an open source application.

In December 2003 and January 2004, the Georgia AOC and Washington AOC recruited the following four courts to be pilot sites for the OXCI EFM software:

- The Washington County, Georgia, Court.
- The Supreme Court of Georgia.
- The Chelan County, Washington, Court.
- The Lancaster County, Nebraska, Court.

In January and February 2004, MTG delivered the initial OXCI Architecture and OXCI XML Interface Specifications documents based on the following:

- The requirements for Court Filing Blue as defined by the LegalXML Electronic Court Filing TC in the December 2003 face-to-face meeting.
- A design meeting in January 2004 involving representatives from the Georgia AOC, Washington AOC, MTG, and counterclaim.

Based on the requirements defined in the OXCI Architecture and OXCI XML Interface Specifications, counterclaim delivered the Software Requirements and Software Design in February and March 2004.

Beginning in February 2004, the OXCI participants, including the pilot sites, began conference calls every 2 weeks to discuss a number of topics including project status, comments on the design deliverables, the development schedule, the requirements of the pilot sites, and ongoing requirement changes for the EFM. The key issue for each pilot site was the development of adapters to interface between the OXCI EFM and the CMS and DMS in each court. It was agreed that:

- The Georgia AOC would provide the adapters for the Washington County and Supreme Court pilot sites.
- The Nebraska AOC would provide the adapters for the Lancaster County pilot site.
- The Washington AOC would provide the adapter to the new Washington State CMS (Justice Information System [JIS]) for the Chelan County pilot site.
- counterclaim would provide the adapter to the Liberty DMS in the Chelan County pilot site.

In addition, MTG and counterclaim continued to meet approximately every week via conference calls to discuss project status and the design deliverables as they had informally since November 2003.



In March 2004, it was determined that the JIS would not be ready for testing with the OXCI EFM. As a result, the Washington AOC and MTG contracted with E-Filing.com (that had been working previously with the court in Chelan) in April 2004 to develop an adapter for the existing Washington State superior court CMS (Superior Court Management Information System [SCOMIS]).

In April 2004, the Washington AOC also extended the contract with MTG to provide Quality Assurance reviews of the code and documentation and to report defects back to counterclaim using the issue-tracking tools on the SourceForge.net site.

In May and June 2004, E-Filing.com identified a major architectural issue with the synchronous method in which the OXCI EFM was designed to interface with certain CMS and DMS adapters, including the SCOMIS adapter. These issues in the application architecture would make the EFM difficult to manage/implement as a standard across many jurisdictions. The frequency of OXCI conference calls was changed from every 2 weeks to weekly in order to resolve this and other issues related to the development of the adapters. The participants agreed to change the adapter interface to an asynchronous model, which forced a reimplementations of the CMS and DMS adapter interface in the EFM and, consequently, the need for a 3.1 release of the code in July 2004. An additional release in August 2004, Version 3.2, resolved other issues.

In August 2004, the following initial set of lessons learned through the OXCI EFM development project were shared with the members of the Court Filing Blue Requirements Subcommittee and helped to shape the direction for the TC:

■ XML Interface Specifications Lessons Learned

- » Use GJXDM subset schemas rather than full schemas because extension through restriction is not supported by most object models.
- » Support versioning of the schemas and WSDL – the XML interface specifications and the underlying GJXDM will be updated periodically.
- » It isn't clear how to define interfaces using multipart MIME attachments in WSDL.
- » The Universal Business Language (UBL) provides a standard for describing payments in XML.

■ Architecture Lessons Learned

- » Some CMS and DMS adapters require asynchronous messaging which is not yet well defined in Web services. It is defined in the WS-Callback and SOAP-Conversation specifications but is not part of WS-I Basic Profile.
- » Most Java and .NET development tools have much better support for Web services than ebXML and the gap is increasing.

- » For good network security, the architecture should support deployment of the EFM outside the court's firewall.
- » The Internet Content Adaptation Protocol (ICAP) provides a standard for interfacing to virus-scanning engines.

In September 2004, the LegalXML Electronic Court Filing TC met in a 2-day, face-to-face meeting to resolve several architectural decisions related to the Court Filing Blue specification. Among other decisions, the TC decided to fundamentally change the way that the EFM interfaces with the court's back-end systems. Previous designs incorporated multiple interfaces between the EFM and the court's CMS, DMS, and financial systems. The current Court Filing Blue design describes the court's back-end systems as a single, integrated Electronic Court Record (ECR) system and defines a single interface between the EFM and the ECR system.

While the OXCI Architecture was designed to address the needs of the pilot installations, the application architecture underlying pilots is significantly different than that proposed for Court Filing Blue. For that reason, the Georgia AOC and Washington AOC agreed in October 2004 to stabilize the EFM code as a 3.3 release and halt further investment by OXCI in the EFM as currently designed and implemented. This included the cancellation of the pilot tests and related deliverables. The future opportunities for OXCI, described in Section IV, include options for refactoring the existing code base into a Court Filing Blue-compliant EFM, implementation of the 3.3 release in courts in the interim, and other opportunities for future work by OXCI members and the larger court filing community toward developing court filing standards and reference implementations.

## B. PROJECT DELIVERABLES

The OXCI Project Work Plan listed a number of deliverables to be developed by MTG and/or counterclaim as part of the project. Of the 18 deliverables listed in the original work plan, the following 14 have been completed and posted to the project Web site at <http://oxci.sourceforge.net>:

- Project management deliverables.
  - » Project Charter.
  - » Project Work Plan.
  - » Project Management Plan.
  - » Monthly Status Reports.
  - » Post-Implementation Review.

- Design deliverables.
  - » Revised Architecture Specification.
  - » Software Requirements.
  - » Design Document.
  - » Test Plan.
  - » XML Schemas.
- Development deliverables.
  - » Iteration Code and Preliminary Documentation (three iterations).
  - » Iteration Testing Reports (three iterations).
  - » Help Documentation and User Manuals.
- Testing and deployment deliverables.
  - » Pilot Test Plan.

In summary, all deliverables listed in the work plan related to project management, design, and development, as well as one testing and deployment deliverable, are available on the project Web site. In addition, the following deliverables beyond the scope of the original work plan were developed and posted to the project Web site:

- Quality Assurance Report.
- Three additional iterations of the code and documentation (3.1, 3.2, and 3.3).

Finally, as a result of the decision to cancel the pilot tests, the following four testing and deployment deliverables listed in the work plan were not developed:

- Pilot Test Report.
- Training Plan.
- Training Materials.
- Deployment Plan.

III. LESSONS LEARNED

### III. LESSONS LEARNED

The implementation experience from this project has generated a valuable set of lessons learned. This section is organized under the following headings:

- Court Electronic Filing Lessons Learned
- Open Source Software Lessons Learned
- Justice XML Lessons Learned
- Web Services Lessons Learned

The remainder of this section describes each of these lessons learned in detail.

#### A. COURT ELECTRONIC FILING LESSONS LEARNED

Over the course of the project, we have discovered a number of gaps in the electronic filing requirements developed by the LegalXML Electronic Court Filing TC. These gaps, including the current approach and schedule for addressing them, include:

- *The coordination between the EFM and the CMS and DMS adapters can be complex.* The TC has decided to model the set of court back-end systems as an integrated ECR system and define a single interface between the EFM and ECR. However, the TC has left open the possibility of modeling the interactions within the ECR in a future extension to Court Filing Blue.
- *The relationship between electronic court filing and electronic service (e-service) needs to be carefully defined.* The TC has decided that e-service is within the scope of Court Filing Blue. The Court Filing Blue Requirements Subcommittee is currently developing use cases to describe the relationship between electronic court filing and e-service.
- *The requirements for court policy are not well defined.* The TC has decided that each component of the Court Filing Blue architecture needs to provide a policy interface. The Requirements Subcommittee is currently tasked with developing the requirements of the policy interfaces.

The following paragraphs discuss these issues and other lessons learned in detail.

1. Some CMS and DMS adapters require asynchronous messaging.

The CMS and DMS adapters receive filings from the EFM, enter them into the CMS or DMS, and return a confirmation to the EFM that the filings were docketed in the CMS or DMS. The original architecture assumed incorrectly that the CMS and DMS adapters would be able to automatically enter the filings into the CMS or DMS and return a confirmation to the EFM synchronously, or within seconds of receiving the filing. In fact, some CMSs and DMSs require additional input from the clerk and cannot accept filings completely automatically. In these cases, it may be minutes to hours between the time the EFM sends the filings to the adapter and the time when the adapter can return a confirmation to the EFM. Therefore, the most flexible approach is to design the interfaces between the EFM and the adapters as asynchronous interfaces. Fortunately, this does not preclude the adapters from sending immediate confirmations if entry into the specific CMS or DMS can be completely automated.

2. Courts have different methods for representing the disposition of electronic filings.

In the case of paper filings, most courts apply an inked stamp to each document indicating that it was accepted by the clerk and the date it was accepted. However, courts differ in how they represent the disposition of electronic filings. Some courts require a visual stamp applied to imaged documents to replicate the stamp applied to paper documents. Other courts apply a digital signature to the filing. After considerable discussion during the project, the decision was made not to implement visual stamping in the EFM and, instead, to capture the filing disposition and disposition date in XML. However, the EFM does not currently protect the integrity of the filing and disposition with a digital signature.

3. The coordination between the EFM and the CMS and DMS adapters can be complex.

The original architecture positioned the EFM as the manager of the interactions with each of the court back-end systems. However, the coordination of the interactions with the back-end systems proved to be more complex than originally expected, due to conflicting requirements from participating courts. In some cases, information originating in the CMS needed to be captured in the DMS. Also in some cases, courts insisted on adding a visual stamp to each imaged document, and the simplest way to implement the visual stamp was through the CMS adapter. These complexities led to several design changes:

- The original design in which the EFM would submit the original filing to both the CMS and the DMS, first with synchronous confirmations from both and, in a later version, with asynchronous replies.

- A modified design in which the EFM would submit the original filing to the CMS, the CMS would process the filing and return the modified filing to the EFM, and the EFM would submit the modified filing to the DMS.

In the end, the choice was made to stay with the original design. Unfortunately, this issue required a lot of development time to resolve. Some courts were still not satisfied with the final decision. Effective progress on the project halted when it became clear that stable requirements from the pilot participants in this area would not be achievable in a reasonable time frame.

4. The relationship between electronic court filing and e-service needs to be carefully defined.

Some of the courts agreed to participate in the project with the expectation that e-service of documents was intrinsically part of any electronic court filing solution. In fact, outside of the filing confirmation from the EFM to the electronic filing service provider (EFSP) submitting a filing, e-service was always out of scope. However, this confusion arose from the fact that there is a relationship between court filing and e-service. We recommend that future implementations of court filing explicitly define how e-service relates to the scope of the implementation.

5. For good network security, the architecture should support deployment of the EFM outside the court's firewall.

Although the EFM restricts access to certain EFSPs, there are still a number of concerns with granting access through a court's firewall to an EFM deployed on an internal network. The LegalXML Electronic Court Filing TC reviewed the OXCI Architecture and suggested that the architecture support deployment of the EFM outside the court's firewall. Although there are no technical issues in the current architecture that would prevent a deployment of the OXCI EFM outside the firewall, there may be some security concerns with allowing an EFM outside the firewall to connect directly to a CMS, DMS, or database inside the firewall. One possible solution may be to break up the EFM into two systems. One system outside the firewall would simply control access, receive filings, and forward them to a second system inside the firewall. The inside system would then provide the clerk review function and connect to the back-end systems.

6. The requirements for court policy are not well defined.

The LegalXML Electronic Court Policy Subcommittee created a Court Policy Interface Requirements document for use with the Court Filing 1.x specifications, and the Nebraska AOC submitted a Court Policy Document Type Definition (DTD) for consideration by the TC. However, these specifications have not been updated to reflect the needs of Court Filing Blue. In addition, the decision by the TC that each component of the Court Filing Blue architecture (which includes the Filing Assembly Component, the Filing Review Component, and the ECR system) needs to provide

a policy interface will fundamentally change the requirements of the policy interfaces. The Requirements Subcommittee is currently tasked with developing the requirements of the policy interfaces.

7. Early reference implementations can be extremely valuable to a standards development process.

The OXCI EFM is intended to be a reference implementation of the Court Filing Blue specification, which are still being developed. By developing the OXCI EFM based on the requirements of Court Filing Blue, we have been able to learn the implementation lessons described in this section early enough in the standards development process to affect the standards themselves. This approach will also help the early adopters of the Court Filing Blue standard by providing a reference implementation on which to build or compare against other implementations.

## B. OPEN SOURCE SOFTWARE LESSONS LEARNED

We also learned the following lessons regarding development of open source software:

1. As with all projects, independent Software Quality Assurance should be included early in an open source project.

The original RFP and contracts did not include an independent Software Quality Assurance (SQA) testing component. However, once the development team started releasing the first iteration of the EFM, it became evident that, even though the code was available publicly as open source, no one outside the development team was performing any structured testing of the code and documentation. The SQA component was eventually assigned to MTG, but by that time most of the development schedule time had already passed, which resulted in reporting most software defects late in development. In addition, several of the defects identified through SQA were related to specific tools used by local architectures. These persistent issues with tool implementations caused continuous delays in the final phase of the project.

2. Developers of open source applications should select a licensing model as early as possible.

Intellectual property (IP) issues are inherent to any software development project. However, the IP issues in this project took an exceptionally long time to resolve because of a combination of complex factors. First, the legal staff involved were not well acquainted with the nuances of open source licensing models which were made even more complex by the fact that the EFM was a modification of an existing product, counterclaim's OpenEFM, which came with its own licensing model. These issues were resolved by an agreement to make the OXCI EFM a separate and distinct product from



the OpenEFM. Second, the state attorney general was concerned with hosting the project on SourceForge.net where California rather than Georgia laws apply. This issue was resolved in collaboration with the SourceForge legal staff, with a special agreement for state governments. Third, the attorney general was also concerned about the fact that this was a multistate effort due to the restrictions that have to be placed on interstate contracts. This issue was resolved by stating that the states were contractually independent of one another. Finally, the legal staff needed some time to explore the potential liabilities of electronic filing. Given all these issues, there was fortunately agreement early on in the project to use the MPL in the interim until the final IP issues were resolved.

3. With some caveats, open source is a good approach for software development in government, particularly for applications for more than one jurisdiction.

Because the EFM needs to be adapted to support the business processes and systems of many courts, open source was a better option than trying to license the EFM to each court. By releasing the EFM as open source, OXCI allows any court or vendor to adapt it for its own needs.

However, open source projects succeed when someone contributes the initial codebase and there is an energized community of developers to extend and maintain it. Although counterclaim was contracted to provide the initial codebase and the courts participating in the project have provided developers to extend it to support each court, so far there has been little interest by other courts and vendors in extending or maintaining the code. In addition, no court, company, or individual has volunteered to manage the codebase beyond the scope of the current project. To succeed in the long term, the OXCI Steering Committee must designate an entity to manage the code base and must find a way to encourage courts and the larger community to use and maintain the EFM.

### C. JUSTICE XML LESSONS LEARNED

Over the course of the project, we learned the following lessons regarding the use of the GJDXM 3.0:

1. Implementers of applications using the GJXDM should use subset schemas rather than full schemas because extension through restriction is not supported by most object models.

For a number of reasons, most importantly the fact that the subset schema generator tool was not yet available, the original OXCI XML Interface Specifications imported the full Global Justice XML Data Model (GJXDM) schema. The OXCI schemas then used type restriction to add constraints (such as required elements) and type extension to add more elements specific to court filing. The

OXCI schemas also included provisions for local courts to add more elements and constraints through type extension/restriction and type substitution.

counterclaim and MTG successfully validated the GJXDM schemas, the OXCI schemas, and the OXCI example instances using XMLSpy and the Xerces-based validation tool provided by the Georgia Tech Research Institute (GTRI). counterclaim attempted to generate Java objects from the XML schemas, using the Java Architecture for XML Binding (JAXB) reference implementation, JaxMe, an open source implementation of JAXB, Castor, and also several Windows code generation tools, including Liquid XML and Breeze XML Binder. All of these products failed to generate code from the full GJXDM schemas, due to the schemas' complexity and use of features not supported by the tools. Some of these tools even failed to validate the schemas.

The issue was the fact that none of the tools implement the full World Wide Web Consortium (W3C) XML Schema specification. Specifically, the code generation tools we tested did not support type restriction. This turned out to be a well-known limitation as Mr. Jim Beard from counterclaim corresponded with other developers who experienced problems implementing this feature in their applications.

We eventually decided to use subset schemas rather than importing the full GJXDM as recommended by GTRI. However, at the time, there were not yet any documented procedures or examples for manually developing subset schemas, and the subset schema generator tool was not yet available. In the end, we used a tool developed by MTG to generate the subset schemas, and counterclaim was successful in generating Java objects from the subset schemas.

2. Court filing implementers should consider using constraint schemas to enforce constraints on specific instances of the GJXDM.

Constraints refer to the restrictions on minimum or maximum occurrences, or the allowable values of an element. Although using subset schemas solved the problem of generating Java objects from the GJXDM, the subset schemas were still rather loosely constrained. This is due to the fact that the constraints on GJXDM objects in subset schemas are defined globally (e.g., every element based on the same type has the same constraints). The result of these loose constraints is that we were forced to define some constraints in the OXCI XML Interface Specifications that we were not able to describe and enforce in the actual schema.

Although they were not recommended when the OXCI XML Interface Specifications were originally developed, constraint schemas are now a GJXDM-compliant method of defining local constraints for each element, including different constraints for elements that are based on the same GJXDM type. If we developed a new version of the OXCI XML Interface Specifications, we would strongly consider using constraint schemas.

D. WEB SERVICES LESSONS LEARNED

We have discovered a number of gaps related the implementation of messaging using Web services. These gaps, including the current approach and schedule for addressing them, include:

- *Asynchronous messaging is not yet well defined in Web services.* Although several vendor-specific solutions exist, there is no clear direction or schedule for when this issue will be resolved.
- *Most Java and .NET development tools have much better support for Web services than the Electronic Business XML (ebXML) Messaging Service 2.0 specification, and the gap is increasing.* However, ebXML Messaging Service (ebMS) 3.0 which incorporates many of the latest Web services standards, is scheduled for release in November 2004.
- *The Multipurpose Internet Mail Extensions (MIME) specification is not well supported by the .NET development tools.* The Simple Object Access Protocol (SOAP) Message Transmission and Optimization Mechanism (MTOM) should replace both MIME and Direct Internet Message Encapsulation (DIME). MTOM should be finalized in December 2004 and be supported by both Java and .NET development tools in early 2005.
- *It is not clear how to define interfaces using multipart MIME attachments in Web Services Description Language (WSDL).* However, the MTOM and WSDL 2.0 specifications should provide more guidance on describing attachments in WSDL. WSDL 2.0 is scheduled to be finalized between March and June 2005.

These gaps and other lessons learned related to Web services are described in detail below.

1. Asynchronous messaging is not yet well defined in Web services.

As we described above in subsection III.A., Court Electronic Filing Lessons Learned, we discovered a requirement for asynchronous messaging, or callbacks, between the EFM and the CMS and DMS adapters. Unfortunately, there is not yet a standard method for implementing asynchronous messaging using Web services. Specifically, although there are some vendor-specific solutions for supporting asynchronous messaging such as WS-Callback and SOAP-Conversation, it is not defined within the WS-I Basic Profile 1.0 or any other well-supported Web service specification. The result of this issue was that counterclaim had to develop an application-specific method for supporting callbacks. Hopefully, future versions of the WS-I Basic Profile and related specifications will provide more support for asynchronous messaging.

2. Most Java and .NET development tools have much better support for Web services than the ebXML Messaging Service 2.0 specification, and the gap is increasing.

The OXCI Architecture defined the ebMS as the messaging interface between the EFSP and the EFM, and Web services as the messaging interface between the EFM and the CMS and DMS adapters. This design decision to use ebMS for the public interface of the EFM was made based on the fact that ebMS provided a more complete specification and proven interoperability between various ebMS Message Service Handler (MSH) products. However, in the last year, the emergence of new Web service specifications, particularly the WS-I Basic Profile, has matched and surpassed ebMS 2.0 in support by software development tools, including both Java and .NET tools. In fact, even the new ebMS 3.0 specification will incorporate many of the latest Web service specifications.

3. The MIME specification is not well supported by the .NET development tools.

The OXCI Architecture uses the SOAP with Attachments (SwA) and MIME specifications to attach payments, lead documents, and other documents to the filing. Java Web service frameworks include support for MIME. However, .NET development tools, such as Visual Studio .NET, include support for DIME, a competing specification advocated by Microsoft, rather than MIME. The use of base64 encoding of binary documents rather than the use of attachments is another alternative which was included in the LegalXML Electronic Court Filing 1.x specifications. Base64 encoding is supported by both Java and .NET tools and may be useful in the present, low filing volume environment. However, the issues with base64 encoding include message size overhead and the performance impact of parsing XML with large sections of embedded binary data. Currently, one option for .NET developers implementing the OXCI specifications is to use third-party MIME generation and parsing tools, although there are currently no free tools to provide this feature. In the future, the SOAP MTOM should replace both MIME and DIME and be well supported by both Java and .NET development tools.

4. It is not clear how to define interfaces using multipart MIME attachments in WSDL.

Although the SwA specification is used in many Web service implementations, it is still not clear how to define a Web service using multiple multipart MIME attachments in WSDL. In particular, the fact that WSDL defines a fixed number of attachments in each message conflicted with the requirement to support an undefined number of attachments per filing. Given this issue, we were forced to limit the number of attachments in each filing. In the future, the MTOM and WSDL 2.0 specifications should provide more guidance on describing attachments in WSDL.

5. UBL provides a standard for describing payments in XML.

One of the common requirements of any electronic commerce transaction is the ability to describe payments. Rather than defining a payment XML structure, the OXCI XML Interface Specifications leverage UBL, which describes robust structures for both payments and payment requests.

6. ICAP provides a standard for interfacing to external virus-scanning engines.

An important functional requirement of the EFM that emerged during the project was the capability to scan submitted documents for viruses. Because the EFM is a Java 2 Platform, Enterprise Edition (J2EE) application and there are no well-supported Java-based antivirus solutions, the decision was made to use an external virus-scanning engine. Although almost all antivirus-scanning application programming interfaces (APIs) are proprietary, we discovered that two vendors, Symantec Corporation and Trend Micro, Inc., provided support for ICAP. Although ICAP was originally designed to be an open standard for interfaces between Internet caching engines, we have demonstrated that it also works well as a standard interface for antivirus-scanning.

IV. FUTURE WORK

#### IV. FUTURE WORK

There are a number of opportunities to continue the work begun on this project both within the courts participating in OXCI and in the larger court filing community. Some of these initiatives include:

- Implementation of the EFM by Courts
- Development of the LegalXML Electronic Court Filing Blue Specification
- Development of an Implementation of Court Filing Blue

The remainder of this section describes each of these initiatives and how they relate to the OXCI EFM Implementation Project.

##### A. IMPLEMENTATION OF THE EFM IN COURTS

While the Court Filing Blue specification is being developed, some courts may decide to implement electronic filing in the interim. The OXCI specifications and EFM provide a framework for these implementations that should migrate to the final specifications more easily than an implementation based on the Court Filing 1.1 or other specifications. The main issue for any court implementing the OXCI EFM will be the development of adapters for the court's CMS, DMS, and payment systems. The developers should refer to the OXCI Developer's Guide for guidance on how to create adapters compatible with the OXCI EFM.

##### B. DEVELOPMENT OF THE LEGALXML ELECTRONIC COURT FILING BLUE SPECIFICATION

The LegalXML Electronic Court Filing TC plans to release a draft of the Court Filing Blue specification in December 2004 or early 2005. The TC created a Requirements Subcommittee, which is developing both functional and nonfunctional requirements for Court Filing Blue as follows:

- Use cases which describe the sequence of events that occur when users interact with the court filing system.
- Message types that describe the XML structures for each interaction in the use cases.

- Nonfunctional requirements that require essential technical features of the court filing system.
- ebXML and Web service messaging profiles which describe architectures that support the nonfunctional requirements.

The lessons learned from the OXCI EFM development project have been invaluable in defining all aspects of Court Filing Blue. In particular, the OXCI XML Interface Specifications will provide an excellent basis for the Court Filing Blue message types. In addition, the experiences implementing both ebXML and Web services in the OXCI project have helped to focus those profiles in Court Filing Blue on compatibility with existing tools.

### C. DEVELOPMENT OF AN IMPLEMENTATION OF COURT FILING BLUE

The OXCI EFM was intended to eventually provide a reference implementation of the Court Filing Blue specification. Once the draft Court Filing Blue specification is available, the OXCI Steering Committee should decide how to best provide that implementation. Based on the requirements and design of Court Filing Blue, OXCI might weigh the following options:

- Updating the current OXCI EFM to conform to Court Filing Blue.
- Replacing certain components of the current OXCI EFM with new components.
- Developing a completely new implementation of Court Filing Blue.

In order to select the appropriate option, the OXCI Steering Committee should consider the following issues:

- The quality of the current EFM framework, code, and documentation.
- The percentage of the existing EFM code that could be reused in the new implementation.
- The level of interest by both courts and vendors in using the existing EFM.
- The long-term strategy for maintaining the EFM.
- The opportunities for partnering with vendors or other organizations to develop and maintain the EFM.

Whichever option is selected, the experience developing the current OXCI EFM will be very useful in the development of a reference implementation of Court Filing Blue.