Data Exchange Standards

Adopted May 2016
Version 1.0

Introduction

The exchange of court data represents an extremely dynamic challenge for all involved. The demands of efficiency, timeliness, accuracy and confidentiality combine to impose significant, often conflicting, demands on the exchange process. Traditionally, these challenges have been met locally with solutions targeted to the specific court data management system involved. However, if the court system is to keep pace with the evolving information age, a more global solution is required. The task of this specification is to define a sufficiently rigorous mechanism to standardize the transfer of data between two or more data systems while remaining flexible enough to tailor the exchange particulars required to the specific needs of those systems.

For the purpose of this standard, interaction is being considered between the following entities:

- Clerks of court case maintenance/management systems and supporting systems (referred to as clerk "CMS").
- Circuit court judicial viewer and/or Court Application Processing Systems (referred to as JV).
- State level Judicial Data Management Services system (referred to as "JDMS").

The decentralized nature of the relationships between county and circuit, circuit and state and county and state and the variety of data management solutions deployed guarantees that the transfer of data between various entities within and outside of the court system is a complex matter. Multiple counties may maintain individual CMS systems or may share the same CMS system. Similarly, circuits may share a single JV system among multiple counties within their jurisdiction or deploy individual JV system as appropriate. Consequently, this standard must define a data transfer mechanism that satisfies the need to efficiently and effectively exchange data between court partners and that is independent to the complex relationships mentioned above while simultaneously guaranteeing the highest levels of security, resilience and privacy of the data contained and shared among these systems.

However, it is not possible to compose a standard describing a limitless set of possible interactions. The intent of this standard is to define the mechanism by which a data transfer event is initiated and completed and to provide a description of the data package that is exchanged. It is not concerned with what must happen to a particular piece of data once it is received. Those details are left to the consuming system.

This Data Exchange Standard incorporates other existing, non-proprietary standards and specifications wherever possible. In particular, this standard has dependencies on the [ECF] (See Appendix A), [NIEM] (See Appendix A), [FIPS 180-2] (See Appendix B), and the World Wide Web Consortium (W3C) (See Appendix A). The terminology used in this standard to describe the components of the Data Exchange architecture conforms to a Service Oriented Architecture (SOA) (See Appendix B and C).

Governance

Once the standard is approved, the Data Exchange Workgroup will schedule quarterly conference calls with at least one meeting in-person annually.

Changes to these standards must be approved by the FCTC based on recommendations of the Data Exchange Workgroup before implementation. Requests for changes to these standards will be submitted to the Data Exchange Workgroup via the Office of the State Courts Administrator ("OSCA") and reviewed at the next scheduled meeting and a recommendation will be made to the FCTC.

Volusia County completed a pilot project testing the data exchanges. The documentation can be accessed via http://app02.clerk.org/menu/ccis/.

Nonconformance to these standards, once adopted, may be referred to the FCTC Compliance Subcommittee.

Data Exchange Security

As noted in the Introduction section, version 1.0 of these standards will cover the exchange of data between local Case Maintenance Systems ("CMS"), Judicial Viewer ("JV") and state level Judicial Data Management Services ("JDMS") systems and may include interactions with other state level systems such as the Comprehensive Case Information System ("CCIS") as appropriate. Subsequent versions of this standard may expand upon and include data exchange between additional systems or stakeholders.

The Data Security Model should contain the following elements:

- Data Storage Encryption: All data stored electronically in locations other than those where the systems are located must also be encrypted, (e.g., an offsite backup facility). This also applies to any data extracted from the CMS with the intention of performing bulk transfers into other systems.
- Workstation Security: All end user workstations or devices must maintain an up-to-date, industry standard anti-malware system to protect the information being consumed by the end user. This may be exempted only in the event that a business case has been developed showing that the end device cannot be kept current. In this event, the organization providing the data must be notified prior to the exchange.
- Mobile devices: No data may reside in mobile devices beyond the current session. If such a device is deployed or used for the "consumption" of information, a VPN solution must be deployed and managed by the courts.
- Cleaning Hard Disks: If at any moment a portable Hard Disk Drive or similar technology is used to transfer data among systems, the storage device must be sanitized using the DoD 5220.22-M approach.
- Firewalls: Firewalls are required when data must transport through an external network to reach its destination. This will be through a firewall specific source and destination (IP

and port) defined in the firewall to prevent unintentional access to source/destination servers.

- User Credentials: When credentials (passwords) are necessary to access or transmit data among systems, the password should be a complex (upper, lower, numeric, and special character) combination password no shorter than 8 characters and renewable every 90 days. Provisions should be taken to deny the reuse of the previous 5 passwords.
- Security Updates: To mitigate vulnerabilities at the host and PC level, systems **must** have security updates applied frequently (preferably via automatic update); checks to ensure any system is not vulnerable should be performed before bringing it into production.

Transport

All data transport should be secured and encrypted in compliance with ECF 4.0.1, Section 5, Service Interaction Profiles, as augmented below. See Appendix B – [FIPS-180-2] and Appendix C).

- Data Exchange Protocol: Enhanced transport requirements shall be Secure HTTP (HTTPS) that consists of the standard HyperText Transfer Protocol (HTTP) layered on top of a secure Transport Level Security (TLS) session. To maximize security, any public-facing interface should be registered with a Certificate Authority ("CA); either a commercial service, or maintained via the State Courts System. For the best security, 2048 bit (or more) key lengths should be used. For closed data center environments where communications occur between trusted servers, TCP may also be used (See Appendix A.).
- Web Services: To ease implementation, the use of the Web Services Description Language ("WSDL") is strongly recommended, as it helps automate the creation of compliant interfaces for clients by providing a machine-readable description of the web service.

Data transport includes the transfer of data to state and other repositories. For example, AOSC09-30, Statewide Standards for Electronic Access to the Courts, identifies the capability to transfer case and court activity data, both as single records and in bulk, to state level data repositories as an essential capability of court data management systems. Transfers may be made for a wide variety of purposes including routine activity reporting, program and performance monitoring, resource allocation, court operations management and data warehousing. The transfers may use a wide variety of data exchange scenarios, e.g., a data transfer initiated by a local data provider to a receiving state repository in response to changes within the underlying data being reported (event-push), or a transfer where the request originates from the repository to the local data management system (timed-pull). Consequently, the general web services capability established at either end of the data transfer should be capable of handling both types of transactions. The specific strategy, event-push or timed-pull, should be identified by the entity originating the transaction as part of the data request package definition.

It is recommended that data transfer occur using the lowest level, stable technology suitable for the task, in conformance to this standards document. However, it may be necessary to define alternate data transfer mechanisms, such as FTP or FTPS, in order to maintain compatibility with legacy reporting systems or when reporting is of sufficiently short term or is of such a nature as to not justify the cost to develop a web services solution. Suitability of alternate transfer mechanisms should be determined by the entity originating the reporting requirement and

approved by this standard's governing body.

While this data transfer standard is comprehensive, not all elements defined for a data request package may be applicable to a given exchange scenario. Since the data request may involve a large number of agencies, the entity originating the request should define a data transfer package description document detailing the format and content of the data being transferred and identifying the appropriate auditing and tracking elements as provided in this standard. This information may be included as part of the integration kit discussed below. If necessary to ensure data transfer integrity, the service enabling the specific data transfer should provide for immediate, synchronous response to, for example, allow a service to initiate a transfer and the receiving service to signal success or failure of transfer. (See Appendix C).

Data Transfer Framework

The court system is adopting an enterprise standard for data management. Conformance to this standard requires the use of a SOA as the foundation for all data transfer. This approach views data exchange not as a series of isolated data projects with each exchange subject to separate and unconnected rules. It is expected that data exchange projects can be built from a set of reusable modular components that can be mixed and matched as needed to provide the necessary functionality. The data exchange mechanism defined in this standard can serve as an architecture for data transfer in that the mechanism is capable of exchanging data between two end points.

The data transfer can be broken down in to three types of information:

- Metadata describing the data being transferred.
- Sufficient tracking and auditing information to ensure reliable transmission, receipt, and messaging.
- The actual data to be transferred.

The integration toolkit discussed below will contain sufficient information to describe the data exchange. While some of the data needs can vary widely between jurisdictions, there are many types of common data exchanged, across all entities within the state. As specific data exchanges are defined and appropriate integration kits built, it is planned that these standards will be expanded with a library of namespaces, XML Schemas, Major Design Elements (MDEs), and data dictionaries for common data exchanges (See Appendix C). This library will further help standardize data exchange within the court system and simplify implementation of new exchanges across the state. Data Exchange Content Models will be developed to facilitate this standardization (See Appendix C and D.) In the context of web services, Major Design Elements (MDEs) are the conceptual representation of the exchange (See Appendix C) exposing a canonical set of core capabilities (See Appendix F). The Data Exchange architecture is divided into two principal elements:

- Core specifications that define the MIDEs and the operations and messages that are exchanged between the MDEs.

- Service Interaction Profiles (See Appendix C) that are specifications that describe the communication infrastructures that deliver the messages between MDEs. Any Data Exchange MDEs will follow these two principal elements as formulated in the ECF 4.0.1 (or current) standard for data exchange. In addition, the data transfer framework components of:
    - Metadata description.
    - Audit and tracking information.
    - Data content are to be constrained through the use of namespaces and XML Schema Definition ("XSD") files.

Multiple namespaces can be included in one or more XML Schema Definition files that includes all necessary constraints that are specific to the particular data transfer. The Data Exchange XML schemas are implementations of the data exchange content models (See Appendix C and D). They are the only normative representations of the messages.

Integration Toolkit

An integration toolkit should be provided for any implementation purposes. This toolkit consists of detailed documentation identifying:

- A plain language name for the integration toolkit.
- A Universally Unique Identifier ("UUID") for the integration toolkit (mandatory element) – A UUID for the integration toolkit as agreed upon by the entities involved.
- A UUID for other existing or new data exchange specifications – This UUID allows versioning of the specification and promotes controlled upgrades and modifications between different data systems.
- A clear plain language description of the contents of the data being transferred including appropriate references to specifications if necessary.
- Example XML requests and responses, data dictionary (including the detailed description / format of each data element or attribute), references to appropriate business rules, relevant standards and definitions, XML schema definition files, theory of operation, Major Design Elements – (MDEs, and sample usage cases for each MDE (See Appendix C).

Conformance

Conformance to this standard does not apply to existing systems that are technically incapable, or it is cost prohibitive to conforming to this standard and data exchanges.

Appendix A

Symbols and Abbreviations

The key symbols and abbreviations used in this standard include:

ECF

Electronic Court Filing

IEPD

Information Exchange Package Documentation

MDE

Major Design Element (See Appendix C)

NIEM

National Information Exchange Model

OASIS

Organization for the Advancement of Structured Information Standards, *a non-profit consortium for open standards*

SOAP

Simple Object Access Protocol

TCP

Transmission Control Protocol

XML

eXtensible Markup Language

W3C

World Wide Web Consortium

WSDL

Web Services Description Language

WS-I

Web Services Interoperability Organization

Appendix B

Normative References

[ECF Specification]

Electronic Court Filing Version 4.01, https://www.oasis-open.org/standards/, OASIS, May 2013.

[FIPS 180-2]

Secure Hash Standard, http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf, National Institute for Standards and Technology, August 2002.

[Genericode]

A. B. Coates, *Code List Representation (Genericode) 1.0*, http://docs.oasis-open.org/codelist/ns/genericode/1.0/, OASIS Committee Specification, December 28, 2007

[NIEM]

National Information Exchange Model 2.0, http://niem.gov, US DOJ and DHS, 2007.

[NIEM Guide]

NIEM Implementation Guidelines, http://www.niem.gov/implementationguide.php, US DOJ and DHS, 2007.

[NIEM Techniques]

*Techniques for Building and Extending NIEM***,** http://www.niem.gov/topicIndex.php?topic=techPDF, Georgia Tech Research Institute, August 2007.

[Namespaces]

T. Bray, *Namespaces in XML*, http://www.w3.org/TR/1999/REC-xml-names-19990114, January 14, 1999.

[RFC2046]

N. Freed, *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*, http://www.ietf.org/rfc/rfc2046.txt, IETF RFC 2046, November 1996.

[RFC2119]

S. Bradner, *Key words for use in RFCs to Indicate Requirement Levels*, http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.

[RFC4122]

Leach, et al., *A Universally Unique IDentifier (UUID) URN Namespace*, http://www.ietf.org/rfc/rfc4122.txt, IETF RFC 4112, July 2005.

[Schema Part 1]

H. S. Thompson, D. Beech. M. Maloney, N. Mendelsohn, *XML Schema Part 1: Structures Second Edition*, http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/, W3C Recommendation, October 28, 2004.

[Schema Part 2]

P. Biron, A. Malhotra, *XML Schema Part 2: Datatypes Second Edition*, http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/, W3C Recommendation, October 28, 2004

[SOA-RM]

MacKenzie, et al., *Reference Model for Service Oriented Architecture 1.0*, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=soa-rm, OASIS Public Review Draft 1.0, February 10, 2006.

[UBL]

Universal Business Language Version 2.1 30 May 2011. 30 May 2011. Committee Specification Draft 02 / Public Review Draft 02. http://docs.oasisopen.org/ubl/prd2-UBL-2.1/UBL-2.1.html J. Bozak, T. McGrath, G. K. Holman (editors), *Universal Business Language 2.0*, OASIS Standard, December 12, 2006.

[XML 1.0]

T. Bray, *Extensible Markup Language (XML) 1.0 (Third Edition),* http://www.w3.org/TR/REC-xml/REC-XML-20040204, W3C Recommendation, February 4, 2004.

[XMLENC]

D. Eastlake, J. Reagle, *XML Encryption Syntax and Processing*, http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/, W3C Recommendation, December 2002.

Appendix C

Terms and Definitions

The key terms used in this standard include:

Attachment

Information transmitted between MDEs that is of an arbitrary format, and is related to the message(s) in the transmission in a manner defined in the standard. An attachment may be in XML format, non-XML text format, encoded binary format, or un-encoded binary format. (See the terms Message and Major Design Element (MDE) in Appendix C).

Callback message

A message transmission returned by some operations sometime after the operation was invoked (asynchronously). (See the terms Message and Message Transmission in Appendix C).

Content Model

Describes the information components used in the messages defined. The data exchange content models will be the result of a detailed analysis of the data requirements to support the particular data exchange. (See Appendix D).

Core Messages

Defined by the core specifications which define the MDEs and the operations and messages that are exchanged between MDEs. These are required messages for the particular MDEs. (See the terms Message and Major Design Element (MDE) in Appendix C).

Major Design Element (MDE)

A logical grouping of operations representing a significant business process supported by the standard. Each MDE operation receives one or more messages, returns a synchronous response message, and optionally sends an asynchronous response message back to the original sender. (See the terms Message and Synchronous Response in Appendix C).

Message

Information transmitted between MDEs that consists of a well-formed XML document that is valid against one of the defined message structure XML schemas. A message may be related to one or more attachments in a manner defined in the standard. (See the term Attachment in Appendix C).

Message Transmission

The sending of one or more messages and associated attachments to an MDE. (See the terms Attachment and Message in Appendix C) Each transmission must invoke or respond to an operation on the receiving MDE, as defined in the standard. (See Receiving MDE in Appendix C).

Operation (or MDE Operation)

A function provided by an MDE upon receipt of one or more messages. The function provided by the operation represents a significant step in the business process. A sender

invokes an operation on an MDE by transmitting a set of messages to that MDE, addressed to that operation. An operation will have an operation signature. (See the terms Message, Operation Signature, and Major Design Element (MDE) in Appendix C).

Operation signature

A definition of the input message(s) and synchronous response message associated with an operation. Each message is given a name and a type by the operation. The type is defined by a single one of the message structures defined. (See the terms Message and Synchronous Response in Appendix C).

Receiving MDE

The MDE that receives the request with the operation invocation performs the operation and sends the response. (See the terms Major Design Element (MDE) and Operation in Appendix C).

Sending MDE

The MDE that sends the request including the operation invocation and receives the response with the results of the operation. (See the terms Major Design Element (MDE) and Operation in Appendix C).

Service Interaction Profiles

Specifications that describe communication infrastructures that deliver messages between MDEs. (See the terms Message and Major Design Element (MDE) in Appendix C).

Service Oriented Architecture

A design pattern based on distinct pieces of software providing application functionality as services to other applications via a protocol. It is independent of any vendor, product, or technology. The W3C defines it as a set of components which can be invoked, and whose interface descriptions can be published and discovered.

Synchronous response

A message transmission returned immediately (synchronously) as the result of an operation. Every operation has a synchronous response. (See the terms Message and Message Transmission in Appendix C).

Appendix D

Data Exchange Content Models

Data exchange content models describe the information components used in all of the messages defined (See the term Message in Appendix C). The data exchange content models will be the result of a detailed analysis of the data requirements to support the particular data exchange. During the modeling process, common items of data will be identified by a process of normalization to identify aggregates based on functional dependency. Where appropriate, these will be generalized so that they can be re-used to support the various messages. The data exchange content models will be used for the following purposes:

- Facilitate the identification of the reusable components, i.e., the data structures that are common across the Data Exchange messages (See Appendix E).
- Aid in understanding the information requirements of the total scenario.
- The source from which the object classes are derived and documented in the Data Exchange XML Schemas (See the normative references for Schema Part 1 and Schema Part 2 in Appendix B).

To facilitate comprehension, several particular data exchange content model diagrams will be developed. Each diagram will represent a logical grouping of components and display both the attributes and object classes belonging to the components in the grouping. The scope of each diagram will be arbitrary and will not hold any significance beyond the diagrams.

Appendix E

Data Exchange Messages

The key principles that shall guide the design of the Data Exchange message structures are:

- Interoperability – The Data Exchange message structures shall provide a means for exchanging data among all types of court information systems.
- Completeness – The Data Exchange message structures format shall provide for all the elements for the particular data exchange.
- Simple implementation – The design should foster rapid implementation.
- Simple XML and portable structure – The core messages in a data exchange will be formatted as XML documents (See Appendix C).
- Familiarity – The data elements and code values shall be meaningful.
- Interdisciplinary utility – The design should be usable by a broad range of court related applications.

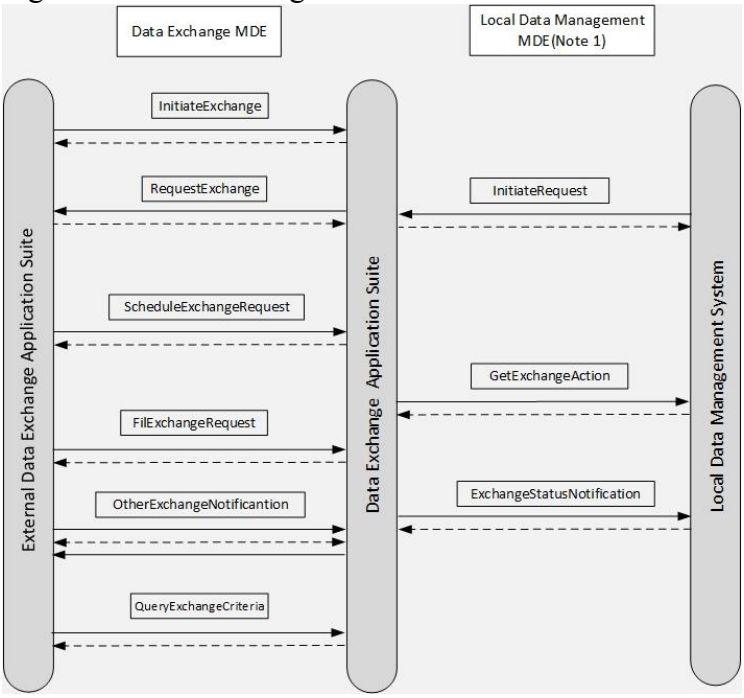
(See the term Message in Appendix C)

Appendix F

Data Exchange Capability Model

This data exchange standard advances a common set of exchange capabilities that should be built upon to define a specific data exchange. The below general methods describe a minimal set of capabilities that each exchange must implement. However, implementation details are left to the individual exchange which need not define methods with these specific names. Refer to Figure 1. for a representative diagram.

Figure 1. Data Exchange MDE Reference



InitiateExchange

The Data Exchange MDE must allow for an external data source to initiate a data exchange at any time. The initiation action for this method includes the direct transfer of data from the external data source to the Data Exchange MDE as part of the Initiate Exchange message. The Data Exchange MDE must respond synchronously with a message denoting receipt of the data or failure of the transfer. Failure messages must include a reason for failure if such reason is identifiable by the Data Exchange MDE.

RequestExchange

The Data Exchange MDE may request an exchange of data from another Data Exchange MDE. The receiving MDE must respond synchronously with the data requested, an error message, or by invoking the ScheduleExchangeRequest operation on the consuming Data Exchange MDE to schedule a date/time when the request will be filled. The RequestExchange message must include a unique identifier for the request that must be used through subsequent operations.

ScheduleExchangeRequest

The Data Exchange MDE may satisfy a RequestExchange action by scheduling a date and time when the requested data will be provided. Messages must use the unique identifier established during the original RequestExchange operation.

FillExchangeRequest

The Data Exchange MDE must resolve a ScheduleExchangeRequest operation by providing the data originally requested by invoking the FillExhangeRequest operation on the requesting Data Exchange MDE. The FillExchangeRequest must use the unique identifier associated with the original RequestExchange operation. The message must contain the data requested. The Data Exchange MDE must respond synchronously with a message denoting receipt of data or failure of transfer. Failure messages must include a reason for failure if such reason is identifiable by the Data Exchange MDE.

OtherExchangeNotification
The Data Exchange MDE must define a capability to establish arbitrary data exchanges. The complexity of court data exchange will necessitate specialized exchanges between local data providers. The OtherExchangeNotification operation should provide a mechanisms for meeting this local exchange need through the appropriate message namespaces while remaining compliant with this specification.

QueryExchangeCriteria
A Local Data Exchange MDE may obtain the necessary exchange criteria parameters from a Data Exchange MDE by invoking the QueryExchangeCriteria operation. The invocation of the QueryExchangeCriteria must include a specific exchange UUID for which to receive criteria as the exchange of different data products may imposed different limitations. The Data Exchange MDE returns a machine readable WSDL describing specific limitation associated.

The following methods should not be exposed for general consumption. They are intended to provide management capabilities to local and/or internal data management systems authorized to interact with a specific instance of a Data Exchange MDE. In particular, the implementation details of the Local Data Management MDE is left to the specific jurisdiction. While it is expected that the accepted method of interaction with the Data Exchange MDE is via a web services protocol, the interaction between the Local Data Management MDE need not be constructed as a web service. The intent of this element of the diagram is to illustrate functionality that the Data Exchange MDE needs to define. For example, the Data Exchange MDE must have functionality to enable local, authorized data management system to initiate a request for data via the Data Exchange MDE. However, while the request for data may be accomplished via web services, the initiation could be accomplished by different means such as another web service, a locally defined message queue or even a simple set of scheduled jobs.

InitiateRequest
The Local Data Management MDE may invoke this operation on the Data Exchange MDE to retrieve data from an external data provider. The Data Exchange MDE must respond synchronously reporting the date/time that the data was requested (via the RequestExchange operation) and the unique identifier for the request. The Data Exchange MDE must respond asynchronously with the requested data, the date/time the data is scheduled to be provided or an error message indicating failure of the data transfer.

GetExchangeAction

The Data Exchange MDE may invoke the GetExchangeAction on the local data management MDE if that system provides for it. The Local Data Management MDE must respond synchronously with a method, location or mechanism to store or process the data received the the Data Exchange MDE.

ExchangeStatusNotification

The Data Exchange MDE must define a capability to exchange status and other relevant information with the Local Data Management MDE through appropriate messages and namespaces.