



BJA Bureau of
Justice Assistance

U.S. Department of Justice
Office of Justice Programs
Bureau of Justice Assistance



United States
Department of Justice

Building Exchange Content Using the Global Justice XML Data Model:

***A User Guide for
Practitioners and Developers***



Developed in partnership with SEARCH,
The National Consortium for Justice Information and Statistics

* * *

This document was prepared under the leadership, guidance, and funding of the Bureau of Justice Assistance (BJA), Office of Justice Programs, U.S. Department of Justice in collaboration with SEARCH, The National Consortium for Justice Information and Statistics. This project was supported by Award No. 2003-DD-BX-K014, awarded by the Bureau of Justice Assistance.



U. S. Department Of Justice

Bureau of Justice Assistance

June 2005

Dear Colleague:

The U.S. Department of Justice Global Justice Information Sharing Initiative and the resulting product, the Global Justice XML Data Model (GJXDM), represent an unprecedented government paradigm. Not only is GJXDM the result of a true partnership among local, state, and federal practitioners, but it is also the result of an extraordinary collaboration with industry and the private sector. Most important, GJXDM was developed from the “ground up” by local users.

Building GJXDM on this local vision is critical to the effectiveness and success of the initiative. While I am currently the Director of the Office of Justice Programs’ Bureau of Justice Assistance, my nearly 20 years of justice experience at the state and local level reinforce two axioms: that local control must be at the core of every new policy and product, and collaborations provide insight and a network for long-lasting change. Once a vision, Global Justice XML is now a reality in Pennsylvania’s Justice Network (JNET) system; in Maricopa County, Arizona; in Los Angeles’s and Southern California’s regional information sharing; in the Wisconsin Justice Information Sharing Initiative (WIJIS) Gateway project; in my home state of Ohio; and in many other jurisdictions across America.

In March 2005, this information-sharing initiative expanded to a new collaboration between the Department of Justice and the U.S. Department of Homeland Security. Recognizing that critical data regarding offenders and crimes is often *local* information, local agencies have begun adopting the model and can now share information more effectively than ever before. Placing the right information in the right hands at the right time is a powerful resource in the collective fight against terrorism and crime.

From the novice user to the experienced technology practitioner, the Global Justice XML *User Guide* provides clear and consistent guidelines for GJXDM implementation, as well as reference architecture and best practices that have evolved since GJXDM’s release in 2004. Still in the early stages of information sharing, we are nonetheless “ahead of the curve” and setting a new course for future generations.

Domingo S. Herraiz
Director
Bureau of Justice Assistance

Contents

Acknowledgments	iv
------------------------------	----

Executive Summary	v
--------------------------------	---

Introduction: What This <i>User Guide</i> Is and Why It Is Important Now	1
---	---

Module 001: Background/Overview of Global Justice XML Data Model Development	
Part 1 General Overview	4
Part 2 Enterprise Architectural Overview	14

Module 002: General Object-oriented and XML Concepts	
Part 1 Overview of the Global Justice XML	
Data Model Framework	17
Part 2 Properties and Types	25
Part 3 XML Schema Definition Standard	30
Part 4 Global Justice XML Data Model	
Naming Conventions	32
Part 5 Namespaces	36
Part 6 Metadata	40

Module 003: Global Justice XML Data Model Content	
Part 1 Architecture of the Global Justice	
XML Data Model	50
Part 2 Global Justice XML Data Dictionary	
Core Components	52
Part 3 Metadata in the Global Justice XML	
Data Model	59
Part 4 Code Tables and Enumerations in	
the Global Justice XML Data Model	63
Part 5 XML Schema Elements Versus	
Attributes	67
Part 6 Global Justice XML Data Model	
DocumentType	70

Module 004: Using the Global Justice XML Data Model To Describe Real-life Justice Data	
Part 1 Exchange Document Development	
Process	74
Part 2 Exchange Document Project	
Inception	76
Part 3 Domain Modeling	79

Part 4 Domain-Global Justice XML	
Data Model Mapping	86
Part 5 Schema Building	89
Part 6 Building Schema-valid Instances	
and Packaging	103

Module 005: Global Justice XML Data Model Use Cases for Justice and Public Safety	
Part 1 Field Interview Report	104
Part 2 Incident Report	109
Part 3 Sentencing Order	118
Part 4 Uniform Rap Sheet	121

Module 006: Global Justice XML Data Model Funding and Conformance	
Part 1 Funding Requirements: U.S. Department	
of Justice and U.S. Department of Homeland	
Security Special Conditions	125
Part 2 Global Justice XML Data Model	
Conformance Guidelines	127
Part 3 Global Justice XML Data Model	
Conformance Guide for the Practitioner	129

Appendix 1: Global Justice XML Data Model Feedback, Release Process, and Documentation	131
---	-----

Appendix 2: Global Justice XML Data Model Tools and Resources	136
--	-----

Appendix 3: Glossary of Justice and Information Technology Terms	141
---	-----

Appendix 4: Global Justice XML Data Model Partners	154
---	-----

Acknowledgments

The Bureau of Justice Assistance (BJA) would like to thank the following members of SEARCH, The National Consortium for Justice Information and Statistics, for their dedication in producing this document:

Francis X. Aumand III, Chair
Ronald P. Hawley, Executive Director
Kelly J. Harris, Deputy Executive Director
Catherine Plummer, Justice Information Systems Specialist
Twyla R. Putt, Manager, Corporate Communications
Jane L. Bassett, Publishing Specialist

This *User Guide* would not have been possible without the significant contribution of the engineers at the Georgia Tech Research Institute, who developed much of the technical content for this document.

BJA also acknowledges members of the Global XML Structure Task Force, who continue to guide the development of the Global Justice eXtensible Markup Language Data Model (GJXDM).

Last, but not least, BJA extends its thanks to the members of the GJXDM *User Guide* Review Committee, a small cadre of justice practitioners with direct experience in the implementation of GJXDM. Their contribution to the successful completion of this guide is very much appreciated.

GJXDM *User Guide* Review Committee

Scott Came

Justice Integration Solutions

Jeff Harmon

Maximus

Gerry Coleman

Wisconsin Department of Justice

Ken Bouche

Illinois State Police

Pam Scanlon

Automated Regional Justice Information System (ARJIS)

Steve Prisoc

New Mexico Supreme Court

John Ruegg

Los Angeles County (California) Information Systems Advisory Body (ISAB)

Robin Gibson

Missouri Office of the State Court Administrator

Mark Kindl

Georgia Tech Research Institute (GTRI)

Joe Mierwa

VisionAir

Winfield Wagner

Crossflo

Jim Cabral and Ellen Perry

MTG Management Consultants

Paul Embley

Global XML Structure Task Force (XSTF)

Executive Summary

Accurate and germane sharing of information across jurisdictions is a critical issue for justice and public safety. Although there has been significant progress in the field of information technology, the lack of *standards* for exchanging justice data has not only been a major obstacle to, but also the principal reason for, the high costs involved with justice information exchange. The development of the *Global Justice XML Data Model* (GJXDM) represents a significant achievement in the process of developing standards for sharing justice information.

Integral to the success of the Global Justice XML standards effort is the promulgation of written guidelines to facilitate GJXDM implementation by the justice community. In FY 2004, the Bureau of Justice Assistance (BJA) awarded SEARCH with a grant to provide Global Justice XML technology assistance, including the development of a *User Guide* to support the implementation of GJXDM.

Since the inception of this project, BJA has recognized the importance of including input from practitioners and developers currently implementing justice exchanges using the GJXDM components. With that objective in mind, BJA, with support from SEARCH, established a GJXDM User Guide Review Committee, comprised of a focused group of subject-matter and industry experts who have helped to determine the outline of *User Guide* content, reviewed *User Guide* chapters as they were developed, and provided written material and comments throughout the process. The Review Committee was integral to the development of this document.

Through its experience in providing technical assistance over the past year, BJA discovered that agencies and jurisdictions throughout the nation are at varying stages of readiness to implement GJXDM. Some justice agencies are just beginning the process of evaluating and exploring XML as a technology. They need to begin to develop a local knowledge base about GJXDM and its impact on supporting interoperability among justice, public safety, and related systems. Other justice participants have fully embraced XML and related

technologies, and have incorporated them into their systems' architecture. However, they may still need instruction and examples defining information exchange packages, queries, and messages using GJXDM in a form that will ensure interoperability among the justice and public safety community. In developing the outline for the *User Guide*, BJA realized that the guide could not focus on only one audience; it needed to speak to users covering a wide range of technical proficiency and readiness for implementation.

For the new user, this *User Guide* provides a background and overview of the development of GJXDM, a general enterprise architectural overview, and a baseline set of technical concepts derived from training material and documentation developed by the Georgia Tech Research Institute (GTRI), which played the principal role in developing the technical architecture for GJXDM. Terminology and concepts presented in module 001 are explained in more technical detail as the modules advance, so that the new or nontechnical user can also advance, incrementally.

For the more experienced or technical user, the *User Guide* moves from the technical framework and takes the reader further into a methodology for defining the business requirements of the information exchange, as well as an information exchange package (IEP) development process guideline. Several projects undertaken in 2004 to build reference GJXDM exchange content enabled the development of this clear process guideline. These projects, which many GJXDM partners have joined (noted in appendix 4), have been supported by BJA. The experience gained during these projects has helped develop a practical, clinical approach to justice information sharing using GJXDM, derived from the technical foundation.

Many justice participants may not be technical, but every reader should benefit from several domain-specific Use Cases (presented in module 005), which illustrate how the process presented in module 004 has resulted in successful information exchange development.

Nontechnical participants also will benefit by this opportunity to start down the path of discovery of

the GJXDM standard. Policymakers should consider incorporating this information into their information technology planning and funding instruments. This *User Guide* provides guidance on the topic of compliance with new special conditions language affecting grantees of the U.S. Department of Justice and the U.S. Department of Homeland Security.

Finally, this *User Guide* provides information about GJXDM tools, resources, partners, terminology, and documentation.

BJA Seeks Reader Input. Readers of this *User Guide* are valued critics and commentators. BJA values reader opinions regarding GJXDM, and wants input on what it is doing well, what it could do better, and what additional areas readers would like guidance on. Please send comments to catherine.plummer@search.org.

This *User Guide* is available at http://it.ojp.gov/documents/200506_gjxdm_building_exch_content_user_guide.pdf.

Information Exchange Package Documentations Provide Baseline Models for Building XML Schemas

Most justice practitioners, including law enforcement officers and information technology professionals, clearly understand the nation's need to share information among justice and public safety agencies at all levels of government. Many embrace the notion of interoperability for both voice and data exchange and understand the utility of XML as a universal translator. The reality is, however, that building XML schemas (data exchange instructions) that conform to GJXDM has proven to be a challenge for many of those who have attempted this work.

Reference *Information Exchange Package Documentations* (IEPDs), formerly known as *Reference Exchange Documents*, can significantly help local law enforcement agencies by providing good baseline models for GJXDM-conformant information exchange. BJA, with support from SEARCH and the U.S. Department of Justice's Office of Community Oriented Policing Services (COPS), has developed a first set of reference IEPDs for law enforcement, including an Incident Report, a Field Interview Report, a Booking

Report, and a Charging Document. Some of these are included in module 005.

The law enforcement IEPDs have been developed collaboratively by public-sector subject-matter experts and technical developers, with open-source, nonproprietary tools. This subject-matter expertise was critically important: It supported the association of GJXDM components and the correct use of object inheritance, thereby providing domain models that represent the business requirements of the exchange *before* the GJXDM mapping and schema creation commenced.

The publication of ubiquitous law enforcement IEPDs provides, for the first time, tangible models and GJXDM content that can be leveraged by local law enforcement agencies — whether large or small, urban or rural, local, county, state, tribal, or federal, to begin on the path of data interoperability to support information sharing about crimes and offenders throughout the nation.

INTRODUCTION:

What This *User Guide* Is and Why It Is Important Now

The Justice/Public Safety Information-sharing Landscape

In the terrible days following the 9/11 terrorist attacks, the entire nation became more conscious of the critical importance of sharing criminal justice and public safety information and intelligence among all branches and levels of government.

Timely and accurate information is needed not only to defend the homeland from terrorist attacks, but also to support the ability of local, state, tribal, and federal agencies to respond to and manage criminal investigations, incidents, and natural disasters. Today, the safe, reliable, and timely sharing of information across jurisdictions is recognized as the most critical and fundamental issue facing those entrusted with protecting the security of the homeland.

Information, or data, as it is known today, can often be completely different in every single system in use. Each software application has its own database schema (design and instructions) and business logic that may mean nothing to an external system. Because of this, a lot of effort has been put into translating the data either into a common format or to another form that the external system could understand. These translation layers, or *interfaces*, often have been costly and difficult to implement. Moreover, many interfaces between systems have been developed solely for the use of the exchanging systems, in a proprietary format, and are not able to be reused by anyone else.

According to the U.S. Department of Justice, during the past 30 years, this *lack of standards* for linking justice information systems has been responsible for a substantial part of the high costs involved with information exchange and has contributed significantly to the challenges of sharing information among justice agencies. Wouldn't it make more sense if all these systems spoke the same language?

Why XML Matters

XML is a universal language for information exchange and is used to transport data from system to system regardless of the type of database or application. Enabling the easy sharing of data provides a direct and significant benefit to justice and public safety. For example, agencies will no longer need to change a legacy case management system to exchange data with another agency's system.

Efforts Made to Create Data-sharing Standards

Like many other communities within the public sector, the justice and public safety domain has been addressing the need to develop common data-sharing standards and to create a common language for information exchange. This is what XML (**eXtensible Markup Language**) is all about.

XML is the universal language for data description. *What this means is that data from any database or application can be described in one universal format.* XML allows the structure and meaning of data to be defined through simple but carefully defined syntax rules, thereby providing a common framework for cross-platform or cross-system data exchange. XML can act as a *universal translator* among all disparate information systems.

XML finally makes it possible to share data easily by providing a translation layer at each agency system. XML makes this process simple to implement through an **XML StyleSheet Transformation** (XSLT), which is an XML document that maintains a data-mapping scheme to transform one set of data into another through this common exchange format.

Global Justice XML Data Model: Key to Standards Efforts

However, another important key to establishing an XML standard for justice and public safety nationwide is that the exchange partners agree on a common XML *vocabulary* to more clearly represent the information to be shared.

This vocabulary exists today. It is called the **Global Justice XML Data Model** or **GJXDM**.

The Bureau of Justice Assistance (BJA) has supported the development of the Global Justice XML Data Model as a collaborative effort among local, state, tribal, and federal visionaries. The Georgia Tech Research Institute (GTRI) played an important role in developing the academic foundation and the engineering for GJXDM.

Through the collaborative leadership of these many partners since the initial release of the GJXDM standard in January 2004, BJA has supported the efforts of practitioners and developers working together to build exchange content constructed from this common vocabulary. The experience gained during this process has allowed BJA to develop a practical, clinical approach to justice information sharing using GJXDM, derived from the technical foundation. Added to this was the value of *real-world experience* gained by building and implementing justice exchanges. This *User Guide* is an effort to lay out the experience of the past year (2004) and to walk the reader through an explicit, step-by-step process of how to develop exchange content that will support the overarching goal of justice data interoperability.

What This Guide Is: Approach, Assumptions, Goals

Modules 001 and 002 of this *User Guide* provide a background and overview of the development of GJXDM, an enterprise architectural overview, and a baseline set of technical concepts derived from training material and documentation developed by GTRI. The *User Guide* modules build upon each other. For example,

This *User Guide* is an effort to lay out the experience of the past year (2004) and to walk the reader through an explicit, step-by-step process of how to develop exchange content that will support the overarching goal of justice data interoperability.

GJXDM User Guide Modules

Module 001: Background/Overview of Global Justice XML Data Model Development

Module 002: General Object-oriented and XML Concepts

Module 003: Global Justice XML Data Model Content

Module 004: Using the Global Justice XML Data Model to Describe Real-life Justice Data

Module 005: Global Justice XML Data Model Use Cases for Justice and Public Safety

Module 006: Global Justice XML Data Model Funding and Conformance

terminology and concepts presented in module 001 are explained in more technical detail as the modules advance.

This *User Guide* is *not* a primer on XML or data processing, and it assumes that the reader has a basic knowledge of data exchange concepts. There are many other good resources to help the less technical reader, many of which are listed in appendix 2. The goal of this document is to provide the justice practitioner, as well as the developer, with clear and precise guidance on how to build conformant justice exchange content using GJXDM.

The *User Guide* moves from the technical framework and takes the reader further into:

- A methodology for defining the business requirements of the information exchange.
- Domain model naming and design rules.
- GJXDM mapping rules and a template mapping spreadsheet.

-
- A complete set of naming and design rules for the exchange schema set and instance.
 - An information exchange package development process guideline.
 - Customization for local requirements (help is provided with this).

This *User Guide* also offers sample justice exchange content, such as a national model for incident reporting created by experts in local law enforcement and statistical crime reporting, including Uniform Crime Reporting (UCR), National Incident Based Reporting (NIBRS), and the FBI Law Enforcement National Data Exchange system (N-DEx).

Finally, this *User Guide* provides guidance to justice practitioners on the topic of compliance with new special conditions language that affects grantees of the U.S. Department of Justice and the U.S. Department of Homeland Security. It supports the public sector's obligation to ensure that agency exchange content—whether developed in-house or with vendor support—is truly GJXDM-conformant and can be consumed and understood by all exchange partners.

MODULE 001

Background/Overview of Global Justice XML Data Model Development

PART 1 General Overview**What is Justice XML?**

XML (eXtensible Markup Language) is a structured language for describing information being sent electronically by one entity to another. XML Schema defines the rules and constraints for the characteristics of the data, such as structure, relationships, allowable values, and data types.

XML is:

- In a text format, readable by both machines and humans.
- License-free.
- Platform-independent.
- Well-supported by industry.

XML specifications are guided by the [World Wide Web Consortium \(W3C\)](#) standards.

The Justice XML model, illustrated in figure 1.1.1, is premised on XML, but provides XML

Key Terms

XML: A simple, very flexible text format derived from Standardized General Markup Language (SGML). Originally designed to meet the challenges of large-scale electronic publishing, XML is now playing an increasingly important role in the exchange of a wide variety of data on the web and elsewhere.

XML Schema: A W3C-recommended language that defines the structure, content, and semantics of XML documents.

W3C is the international standards body for interoperable technologies (specifications, guidelines, software, and tools) that support the development of the World Wide Web to its full potential, including XML.

For definitions of key justice and information technology (IT) terms, see appendix 3.

tag names and other structure for data that are constrained to meet the specific information exchange requirements of justice and public safety. In other words, Justice XML *extends* XML to provide a concise and defined standard for sharing justice and public safety information throughout the nation, regardless of whether the justice agency or branch sharing the information is local, state, tribal, or federal and regardless of whether the information is exchanged horizontally or vertically within the justice system.

How Did the Justice XML Initiative Get Started, and Why is it Needed?

Since the initiation of justice systems integration in the United States, practitioners have generally worked with vendors to develop unique and proprietary solutions to their individual information-sharing needs, either within one agency or, collectively, within a specific jurisdiction. These technical solutions—while solving immediate information-



figure 1.1.1

sharing objectives—have, unfortunately, created independent systems of information-sharing capability with a limited ability to share information among systems throughout the nation.

The development of the Global Justice XML Data Model (GJXDM), which incorporates a comprehensive Global Justice XML Data Dictionary version 3.0 (currently GJXDD 3.0.2), represents a significant change in the way practitioners will develop their information-sharing systems. GJXDM provides a common language with which justice entities can describe, structure, and share information on criminal justice matters and offenders within a locality, the state, among the states, or with federal or tribal entities.

Whose Efforts Helped Create GJXDM?

Justice practitioners and the private sector have been working for a number of years in the United States to develop standards for justice data exchange through various coordinated and collaborative efforts, including an early grassroots effort known as LegalXML.¹ The achievements of LegalXML include initial drafts of XML schemas for a standard arrest warrant, incident report, prosecution charging document, and sentencing order, as well as an XML standard for electronic court filing.

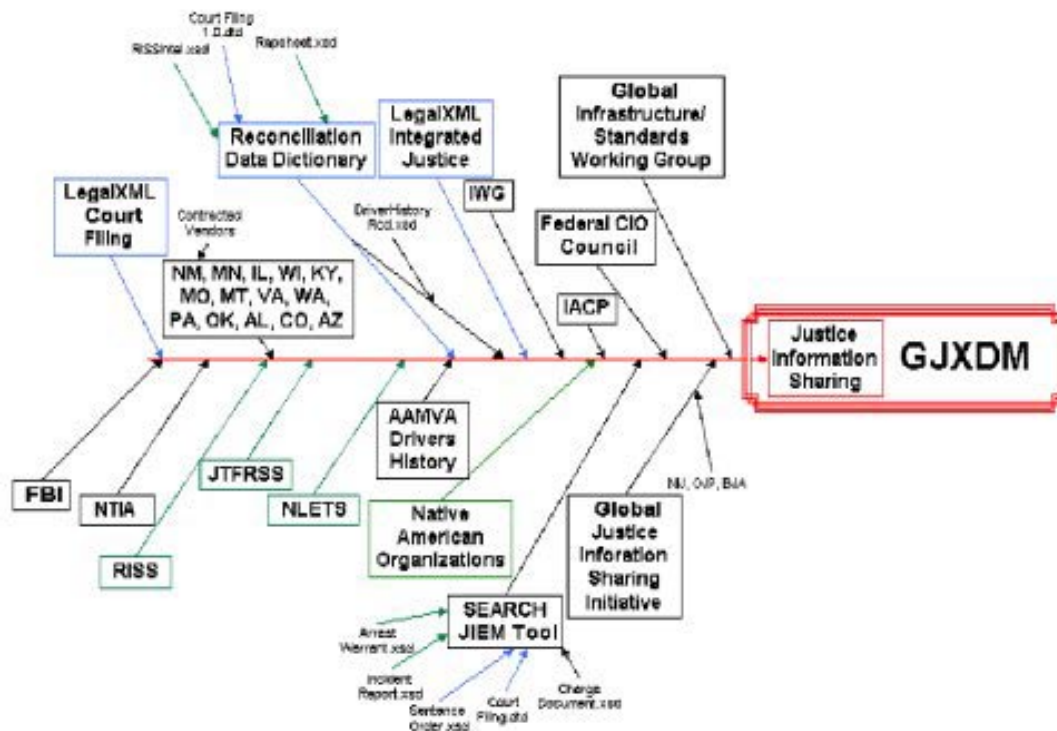
Figure 1.1.2 on page 6 illustrates some of the additional sources that were significant to the core objective of extending and promoting XML standards in justice and public safety information exchange at the local, state, and federal levels. Input from these parallel Justice XML initiatives provided important source requirements, which supported the development of GJXDM as it exists today.

What is the Global Justice XML Data Model?

A data reference model for the exchange of information within the justice and public safety communities. It is a body of rules and concepts that provides a common language with which these agencies can describe, structure, and share information on justice/public safety matters and offenders. It integrates the data names and structures from the **Global Justice XML Data Dictionary** into a database. From this database, an XML schema specification can be generated that consistently represents the semantics and structure of common data elements and types required to exchange information within the justice and public safety communities.

¹ Legal XML has today evolved into a Member Section of the Organization for the Advancement of Structured Information Standards (OASIS), an international XML standards body.

Initiatives Supporting Development of GJXDM



Timeline

1998... 1999... 2000... 2001... 2002... 2003... January 15, 2004

AAMVA	American Association of Motor Vehicle Administrators
BJA	Bureau of Justice Assistance
FBI	Federal Bureau of Investigation
JIEM	Justice Information Exchange Model
JTFRSS	Joint Task Force on Rap Sheet Standardization
IACP	International Association of Chiefs of Police
IWG	Industry Working Group
NIJ	National Institute of Justice
NLETS	The International Justice and Public Safety Information Sharing Network
NTIA	National Telecommunications and Information Administration
OJP	Office of Justice Programs
RISS	Regional Information Sharing System
SEARCH	The National Consortium for Justice Information and Statistics

figure 1.1.2

In 2001, a formal effort was undertaken to reconcile several of the XML specifications developed by justice practitioners. The U.S. Department of Justice (DOJ) funded meetings of the following organizational representatives:

- The Joint Task Force on Rap Sheet Standardization, which developed an Interstate Criminal History Transmission Specification using XML.
- The Institute for Intergovernmental Research (IIR), which developed the Regional Information Sharing System.
- The LegalXML Court Filing Workgroup, which developed an Electronic Court Filing Standard using XML.
- The American Association of Motor Vehicle Administrators, which was working to develop Driver and Vehicle transactions in XML.

These meetings resulted in a reconciliation of these various XML data standards into a common Justice XML Data Dictionary (JXDD) (see [Lessons Learned in Reconciling Three Justice XML Development Efforts](#)).² The first JXDD (Version 1.0) depicted the elements in a Microsoft Access database. Subsequent versions (2.0 and 2.1) of JXDD represented the 1.0 elements as XML schema.

These early efforts were critical to the development of the current GJXDM—an effort undertaken as part of the DOJ’s Global Justice Information Sharing Initiative supported by the XML Structure Task Force (which will be discussed in more detail later).

■ **Global Justice Information Sharing Initiative**

Since 1998, active practitioners from local, state, tribal, federal, and international justice entities have been participating in DOJ’s Global Justice Information Sharing Initiative (Global), an advisory committee to the nation’s highest-

The Global Advisory Committee (GAC) is an advisory committee to the nation’s highest-ranking law enforcement officer, the U.S. Attorney General. Global’s mission—the efficient sharing of data among justice entities—is at the very heart of modern public safety and law enforcement.

ranking law enforcement officer, the U.S. Attorney General. These justice professionals have been working collaboratively to address policy, connectivity, and jurisdictional issues that have historically impeded effective *justice information sharing*.

DOJ’s Global is comprised of justice agency executives and policymakers; justice system planners and managers; justice information practitioners; and, most important, end users. This last group is vital because it distinguishes Global as an entity whose members are actively dedicated to the issue of information sharing precisely because they continue to be producers, consumers, and administrators of critical justice and public safety information.

DOJ’s Global is a “group of groups” that represents many independent organizations spanning the entire spectrum of law enforcement, judicial, correctional, and related domains. Members participate in Global out of a shared belief that, together, they can bring about positive change in interorganizational, enterprisewide communication and data sharing.

DOJ’s Global membership reflects the tenet that the entire justice community, as an enterprise, must be involved in collaborative information exchange, despite the challenge afforded by organizations in disparate levels and branches of government. Global members represent the following justice and public safety constituencies:

- Law enforcement.
- Prosecutors.

² The “lessons learned” report was developed by the XML subgroup of the Global Advisory Committee’s Infrastructure/Standards Working Group.

- Public defenders.
- Courts.
- Corrections.
- Probation and parole.
- Other agencies directly involved in the justice process.

It is important to keep in mind that, in the United States, justice is predominantly a function of state and local government. Global has supported, and continues to support, the objectives of justice information sharing by helping federal—and, more important, local, state, and tribal—justice practitioners create common information-sharing and interoperability standards.

DOJ's Global initiatives and activities, especially data exchange standards development, are derived from actual user requirements and have been driven from the "bottom up." The initiatives are based on the business problems of the different justice disciplines and are not focused solely on any national perspective. This unique approach has provided an opportunity for national organizations to assist and support the process of sharing critical justice information where that information originates: at the local, state, and tribal levels.

■ **Global XML Structure Task Force**

In early 2002, DOJ formed a group called the XML Structure Task Force (XSTF) under Global's Infrastructure/Standards Working Group to develop an object-oriented XML data model for justice information sharing, deriving from the early efforts described above.

XSTF is comprised of justice practitioners and industry representatives from various justice communities of interest and includes representation from local, state, and federal law enforcement, courts, corrections, probation and parole, and transportation agencies; the Federal Bureau of Investigation; SEARCH, The National Consortium for Justice Information and Statistics;³ the U.S. Chief Information Officers Council; and the Integrated Justice Information

Key Terms

Object-oriented Programming (OOP)

combines data structures and functions (computer directions) to create "objects," making it easier to maintain and modify software.

Community of Interest is a group of professionals informally bound to one another through exposure to a common class of problems, and common pursuit of solutions, thereby themselves embodying a store of knowledge. The justice and public safety domain is considered a community of interest.

Systems (IJIS) Institute (a consortium of private-sector companies involved in justice and public safety). XSTF's contribution has been supported by development staff, notably research scientists from Georgia Tech Research Institute (GTRI) and the National Telecommunication and Information Administration. GTRI developed the technical concepts, using XML best practices and standards in the design and implementation of GJXDM.

This XSTF effort has provided a framework within which a productive relationship has developed among practitioners, industry, and development staff. There were several key elements to that relationship that enabled success:

- The development staff had no product at stake in the effort, enabling them to really listen to the requirements of the practitioners and look at those requirements impartially.
- Practitioners and industry came to the table willing to compromise with each other and to consider technical trade-offs the developers provided.

³ SEARCH is a nonprofit membership organization, based in Sacramento, California, that is dedicated to improving the quality of justice and public safety through the use, management, and exchange of information; application of new technologies; and responsible law and policy, while safeguarding security and privacy. See www.search.org.

Primary goal of XSTF in designing for the Data Model:

Develop a common set of reusable, extendible XML data components that could be combined in justice documents, transactions, and messages that are consistently structured to support interoperability among justice and public safety systems nationwide.

- The practitioners were involved with key decision points throughout the process to ensure that the product met their needs and to enable practitioner buy-in.

These factors created an open atmosphere for XSTF's contribution that allowed all the participants to concentrate on achieving a common Justice XML vocabulary that would most effectively meet the needs of each justice domain.

XSTF's success has been enabled by structuring the work into stages:

- First, XSTF worked to collect and analyze user requirements derived from actual justice agency information exchanges.
- Second, an architecture for Justice XML elements was developed, which distributed components of information exchange into domain-specific categories for review and analysis by domain experts.
- Third, data types and elements were parsed into groups organized under a focused set of core objects.
- Last, tools and examples are being developed to effectuate the benefits of using GJXDM in justice-oriented document schema development efforts.

What Does GJXDM Contain?

GJXDM 3.0, current release 3.0.2, developed by XSTF, is the result of an analysis of more than 16,000 justice and public safety-related data elements collected from more than 30 unique sources across the justice domain.

As illustrated in figure 1.1.3 on page 10, these source data were analyzed and organized into groups of common data elements, then organized and reduced to approximately 2,200 properties and 550 types that were then incorporated into approximately 300 core data types, or reusable components. (Note: These concepts are defined and explained in greater detail in module 002.)

Key Terms

Architecture: The design of a system. It may refer to either hardware or software or a combination of both. The software architecture of a program or computing system is the structure or structures of the system. This structure includes software components, the externally visible properties of those components, the relationships among them, and the constraints on their use.

Core Data Type: A basic business data item that describes common concepts used in general business activities.

Data Element: A basic unit of data having a meaning and distinct units or values. A component of data definition; a data "cell" into which data items (actual values) can be placed; the lowest level of physical representation of data.

Property: A characteristic common to all members of an object class.

Type: A description of a class of objects that share the same operations, abstract attributes, relationships, and semantics.

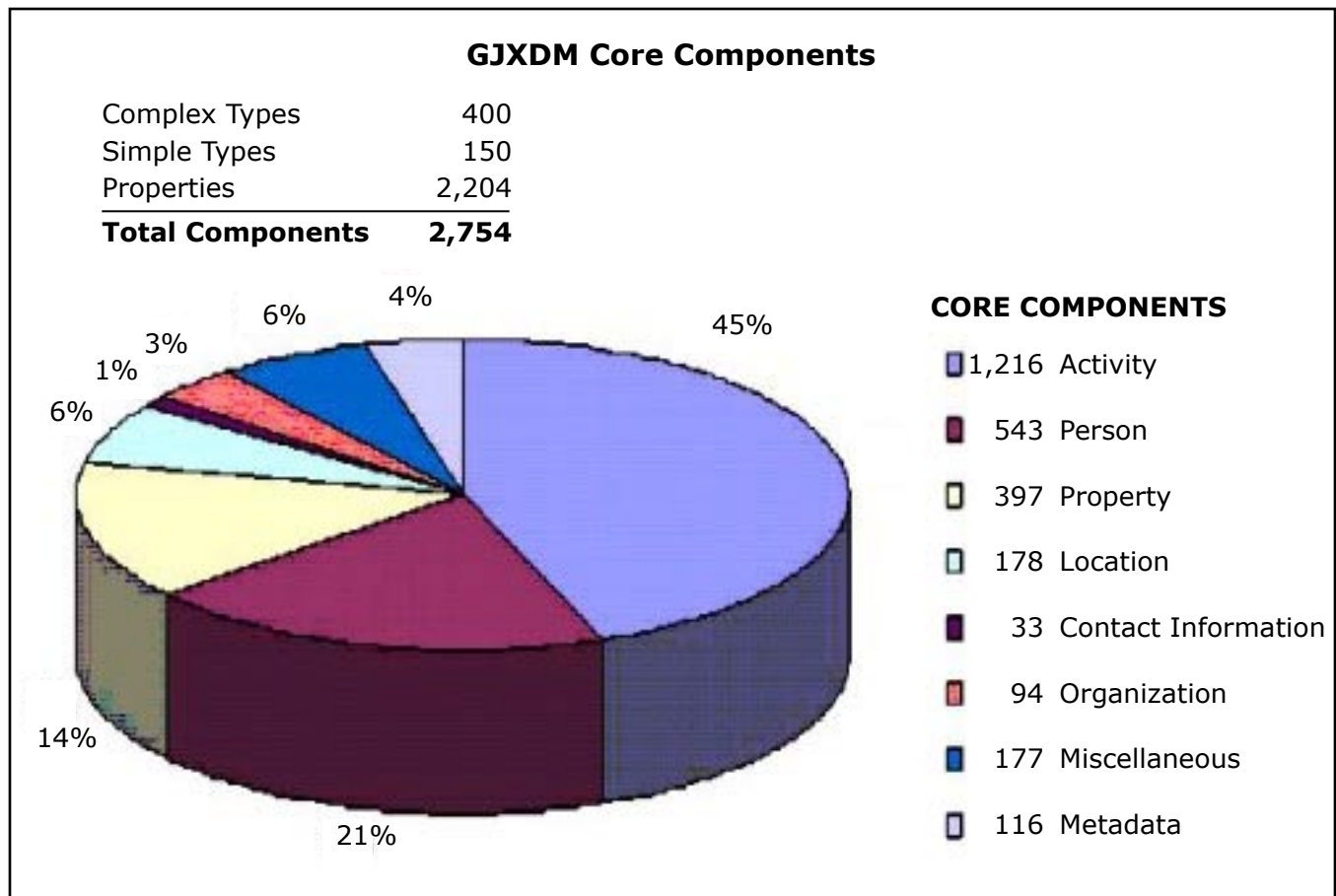


figure 1.1.3

These GJXDM core components have inherent qualities that will enable access from multiple sources, as well as reuse in multiple applications.

The standardization of these core components provides significant potential for increased interoperability among and between justice and public safety information systems. Why? Standardization in this manner provides each of us with functionally equivalent or interchangeable components of the system or process in which they are used, regardless of our individual system differences.

Key Terms

Component: A software object, meant to interact with other components, encapsulating certain functionality or a set of functionalities. A component has a clearly defined interface and conforms to a prescribed behavior common to all components within an architecture.

Interoperable: Functionally equivalent or interchangeable components of the system or process in which they are used.

How Do the Various Parts of GJXDM Relate To One Another?

There has been some confusion among users regarding what is described as the data model (GJXDM) versus the data dictionary (GJXDD). One way to describe the Justice XML architecture is that the data dictionary (GJXDD) is a well-defined vocabulary of data names and structures assembled in an object-oriented data model (GJXDM) from which consistent XML schemas are generated to be used as templates for valid XML instances that carry data payload (actual data content). GJXDM refers to the body of concepts and rules that underlie the data dictionary and the schema. The rules behind the data model are fundamental to the structure of the data dictionary and are used to generate the XML schemas.

Key Terms

Resource Description Framework (RDF) and Web Ontology Language (OWL):

Semantic Web standards that provide a framework for asset management, enterprise integration, and the sharing and reuse of data on the web.

Figure 1.1.4 illustrates the composition of GJXDM.

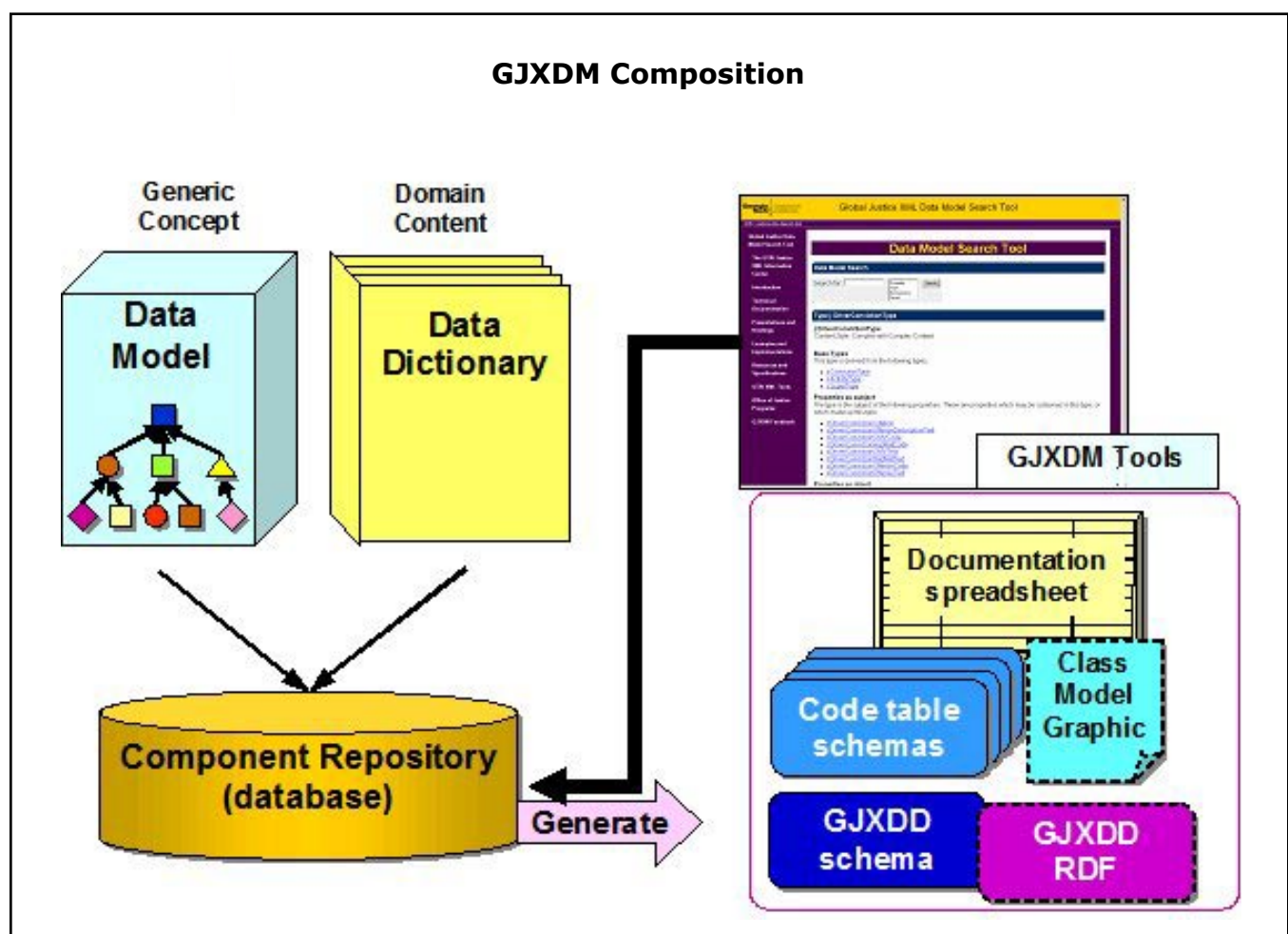


figure 1.1.4

As shown in figure 1.1.5, the GJXDM spreadsheet provides all of the element names, organized hierarchically under the core components (Property, Person, Organization, etc.) with hyperlinks to related elements. The spreadsheet also provides information identifying the type of data being represented (date, integer, Boolean, string, etc.) and a precise, context-rich definition of each dictionary component. The definitions were developed and refined over many meetings and conference calls among XSTF members and GTRI and represent a commitment to provide reusable components that mean the same thing to all of us.

- It is recommended that new users acquaint themselves with the [GJXDD documentation spreadsheet](#), which is provided in a Microsoft Excel® format for easy navigation. The spreadsheet, as shown in figure 1.1.5, is a tangible representation of the data dictionary.



Future products that will be able to be generated from the same component repository include a GJXDD in RDF format and a full class model graphic representation (see figure 1.1.4 on page 11). GJXDM Tools, such as the Justice Information Exchange Model (JIEM) Modeling Tool⁴ and the [Schema Subset Generation Tool](#) also are under continued development to help practitioners and developers use the GJXDM products more efficiently.

What Was the Design Criteria for GJXDM?

XSTF's primary goal has been to develop a common set of reusable, extendible XML data components that could be combined in justice documents, transactions, and messages that are consistently structured to support interoperability among justice systems nationwide. With this goal in mind, XSTF used the following design criteria in the development of GJXDM:

- GJXDM should be constructed from actual functional requirements, reference documents, use cases, and business context components.
- An object-oriented data model, named types, and extensions are best suited to the goals of justice information exchange.
- The composition of the data dictionary should be over-inclusive and optional to allow users to pick and choose appropriate building blocks for their data exchanges.
- GJXDD element and attribute tag names should be based on relevant international standards for electronic data exchange, especially ISO 11179-Specification & Standardization of Data Elements.⁵ Additional source standards include, but are not limited to:

- W3C XML Schema Specification and RDF Schema Specification.
- The Federal Chief Information Officers (CIO) Council Draft Federal XML Schema Developer's Guide.
- The United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT) Electronic Business XML (ebXML) Core Components Technical Specification 1.9.
- Dublin Core Metadata for Documents.
- U.S. Department of Defense 5015.2-STD, "Design Criteria Standard for Electronic Records Management Software Applications."
- Intelligence Community Metadata Language.
- The OASIS XML Common Biometrics Format Committee.
- The Accredited Standards Committee (ASC) X12 Reference Model for XML Design.
- XSTF should define a reference architecture and namespaces for the GJXDM schema specification. GJXDM will continually evolve, so the data model must facilitate change and extension as required.
- Extension methods should minimize impact on prior schema and code investments by practitioners and developers.
- GJXDM must provide migration paths for evolution to new technologies, such as RDF OWL.

GJXDM will provide a mechanism through which XSTF can continue to define standards for justice data exchange with a high degree of granularity. GJXDM will allow practitioners to exchange information in a way that was previously unattainable.

It is truly exciting to see all of these pieces suddenly coming together as the technology provides a way to accomplish what previously was only conceptualized.

⁴ The SEARCH JIEM® project is funded by BJA, under Cooperative Agreement No. 98-DD-BX-0066. The data collected in the JIEM Modeling Tool during its initial development has served, along with other practitioner-developed data exchange specifications, as the foundation for the data exchange components, which have been constructed to provide a standard GJXDM.

⁵ ISO refers to the International Organization for Standardization. See module 002, part 4: *Global Justice XML Data Model Naming Conventions*.

PART 2 Enterprise Architectural Overview

GJXDM is the result of a collaborative effort of numerous agencies from all levels of the justice and public safety domain, and it has been the intent of DOJ's Global to harmonize its effort with federal strategies, including the Office of Management and Budget's *Federal Enterprise Architecture* (FEA).

FEA is constructed through a collection of interrelated "reference models" designed to facilitate cross-agency analysis of services provided to citizens and to identify duplicative investments, gaps, and opportunities for collaboration within and across federal agencies.

Reference Models released by the Federal Enterprise Architecture Planning Management Office (FEAPMO)⁶, and illustrated in figure 1.2.1, include:

- The **Performance Reference Model** (PRM), a standardized framework to characterize the performance of information technology (IT) initiatives and their contribution to program performance. PRM can help produce enhanced IT performance information to improve strategic and daily decisionmaking;

improve the alignment and contribution of IT to outputs and outcomes, thereby creating a clear "line of sight" to results; and identify performance improvement opportunities across traditional agency boundaries.

- The **Business Reference Model** (BRM), which is a function-driven framework for describing the business operations of the federal government independent of the agencies that perform them. BRM provides an organized, hierarchical construct for describing the federal government's day-to-day business operations.
- The **Data and Information Reference Model** (DRM) helps to describe the types of interactions and information exchanges that occur between the federal government and its various constituencies. It will categorize the government's information along general content areas specific to BRM subfunctions and decompose those content areas into greater levels of detail, ultimately to data components that are common to many business processes or activities. DRM will establish a commonly understood classification for federal data and enable information sharing between agencies. A common data classification model will streamline the processes associated with information exchange,

⁶ See www.whitehouse.gov/omb/egov.

FEA Reference Models



figure 1.2.1

both within the federal government and between the government and its external stakeholders.

- The **Service Component Reference Model** (SRM) is a business- and performance-driven functional framework that classifies service components with respect to how they support business and/or performance objectives. SRM is intended for use to support the discovery of governmentwide business and application service components in IT investments and assets.
- The **Technical Reference Model** (TRM), which is a component-driven, technical framework used to identify the standards, specifications, and technologies that support and enable the delivery of service components and capabilities. TRM provides a foundation to support the construction, delivery, and exchange of business and application or service components that may be used and leveraged in a Component-based or Service-oriented Architecture.

Taken together, the FEA reference models create a comprehensive governmentwide framework to guide agency IT investment activities, identify opportunities to collaborate on and consolidate initiatives, and integrate government activities at the local, state, tribal, and federal levels.

Justice Enterprise Architecture

DOJ's Global, through its committee, the Global XML Structure Task Force, has been employing business-process and data-modeling techniques that are not only consistent with FEA guidelines, but especially designed to integrate the core data exchange of local, state, and tribal justice entities.

BJA, with support from [SEARCH](#), has developed a Justice Information Exchange Model Modeling Tool, with the data collected from participating jurisdictions provided as a baseline Justice Reference Model for subsequent jurisdictions.

Justice Data Model development began in March 2001, with Global's XML Reconciliation Project, and has evolved to now provide the Global Justice XML Data Model. Each of these projects has already proven to be an effective means of leveraging technology to serve the needs of justice, public safety, and homeland security. Figure 1.2.2 on page 16 illustrates the concept of a **Justice Enterprise Architecture**, derived from the more general FEA model.

The strategy has been to develop a group of modules connected by the common thread of XML. XML is for the transport of data from system to system; no one will have to change a legacy case management or records management system. The XML data standards described here are proposed for the exchange information payload. Application program transformations can be easily written via XML StyleSheet Transformations to export to or import from any automated system today.

DOJ's Office of Justice Programs is providing local, state, tribal, and federal justice agencies and branches with real tools, developed by justice practitioners and based on comprehensive user requirements, to leverage current technology and enable justice data exchange throughout the nation.

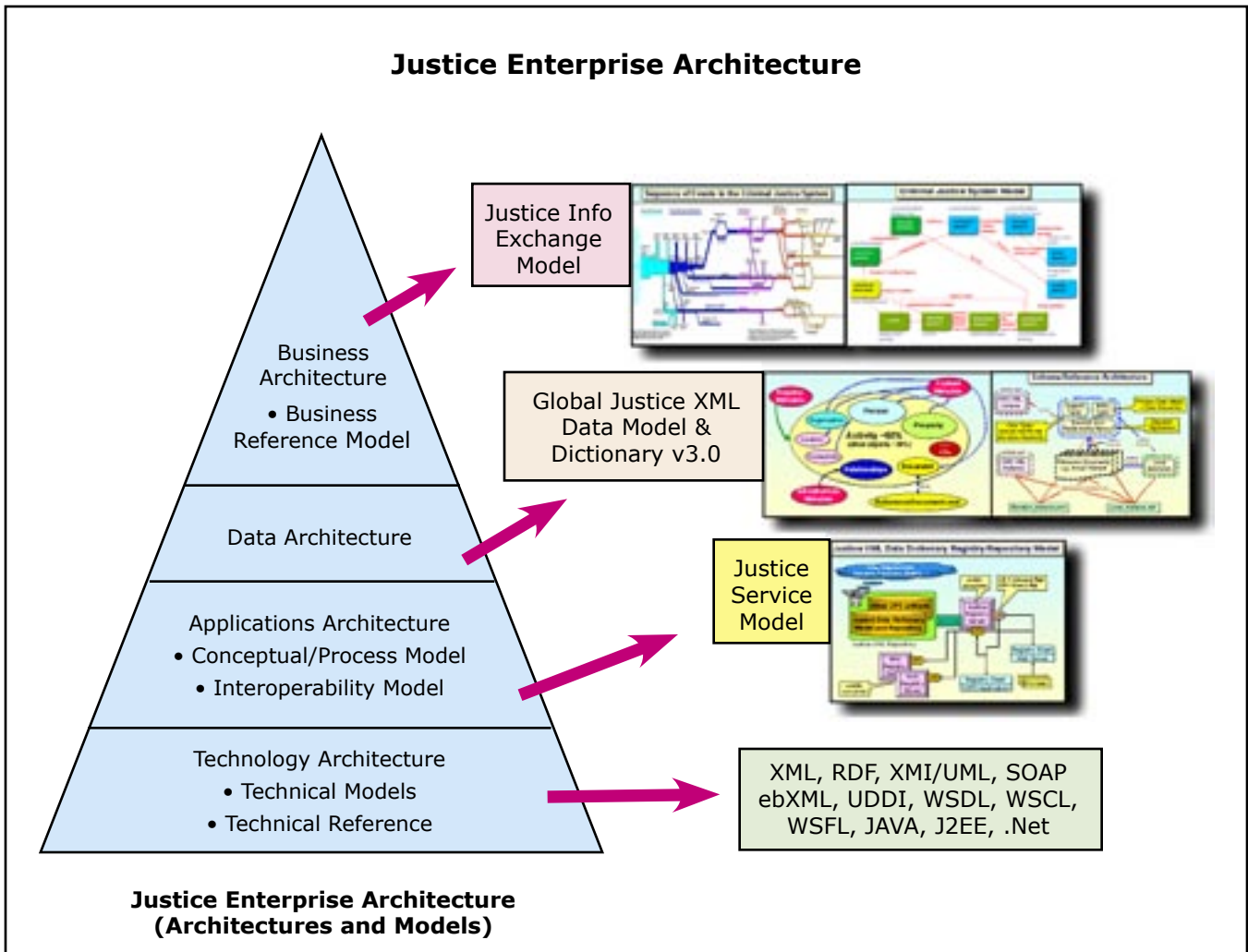


figure 1.2.2

MODULE 002

General Object-oriented and XML Concepts

PART 1 Overview of the Global Justice XML Data Model Framework**How is the Global Justice XML Data Model Constructed?****■ An Introduction to Object-oriented Concepts**

As noted in module 001, the data model is the body of concepts and rules that underlie the Justice XML Data Dictionary and schema. The rules behind the data model are fundamental to the structure of the data dictionary and are used to generate the eXtensible Markup Language (XML) schemas. To understand the rules, which are laid out explicitly in the following modules, it is helpful to develop at least a rudimentary understanding of object-oriented programming, which is the foundation for the development of the data model.

Object-oriented Programming (OOP) has been around since the early 1960s. Yet today, most legacy justice systems are not primarily object-oriented, but rather relational or flat-file databases. Over the past two decades, the emergence of the World Wide Web has led to a broad acceptance of markup languages, which are a universal syntax for marking up document content or other information. Today, where there are legacy systems involved in data exchange, a current trend is to wrap the legacy data in object wrappers.

The primary reason that information technology has shifted toward objects is the advantage of component reuse. Component reuse is one of the core objectives of the Justice XML initiative, as well as the Federal Enterprise Architecture.⁷ The FEA Data and Information Reference Model objectives include:

- Horizontal and vertical information sharing between business lines.

- Business-focused data standardization that can be categorized for re-use.
- Re-use and integration of data, as opposed to duplication.
- Facilitation of cross-agency information exchanges.
- Development of a consistent means to categorize and classify data, among others.

With these objectives in mind, the U.S. Department of Justice's (DOJ) XML Structure Task Force (XSTF) made an early key decision to pursue an *object-oriented approach* in creating a data model for justice information sharing.

So what exactly is an object? This can be a complex question, especially for individuals with a relational database

Object: An entity that contains both data and behavior.

view of information sharing, because now they have to learn an entirely new way of thinking about data. But it also can be a simple question because most people already think in terms of objects without realizing it. For example, look at a person. The person is actually seen as an object, with *attributes* such as height, weight, eye color, hair color, gender, etc. It is also possible to identify *behaviors* associated with a person, such as a certain way of walking or speaking. So, in its basic definition, an object is an entity that contains both data and behavior.

In OOP, a *class* is described as a blueprint for an object, defining all of the attributes and behaviors associated with that object. OOP not only facilitates component reuse, but also provides a more efficient design mechanism by organizing classes and factoring in commonalities of various classes. *Inheritance* is the primary means of providing this function.

⁷ See module 001, part 2.

Inheritance allows a class to inherit the attributes of another class. For example, if one has a Dog Class and a Cat Class, each may have an attribute for eye color. In an OOP design, the eye color attribute could be abstracted up to a super class called "Mammal," along with any other common attributes. Both the Dog and Cat Class would inherit from the Mammal Super Class. The super class contains all of the attributes that are common to classes that inherit from it, and it is not necessary to duplicate them down the inheritance tree for each specific sort of mammal. The Dog and Cat Classes both inherit all common attributes and behaviors from the Mammal Super Class, such as "warm-blooded" or "hair color."

Inheritance provides a design advantage, in that if you are designing a Cat Class, the Mammal Super Class already provides most of the attributes needed to describe a cat. We can then define the Cat Class as a subclass that contains additional attributes that pertain solely to a cat. It may be necessary to abstract the Cat Class further, that is, to define subclasses for Persian cats or Siamese cats, etc.

■ ***Is-a Relationships and Has-a Relationships***

In the Mammal Super Class example, the Dog and Cat Classes inherit directly from the Mammal Super Class. This relationship is referred to as an "Is-a" relationship because a cat is a mammal. When a subclass inherits from a super class, it has all of the characteristics of the super class. Therefore, a dog and cat are considered *extensions* of mammal.

It also is natural to think of objects as containing other objects. For example, a computer, which can be considered an object by itself, also may contain a hard drive and video card, which also are considered valid objects. It is possible to open up the computer and remove the hard drive, so both the computer and drive are considered objects. In this manner, objects are sometimes built, or *composed*, from other objects. This is called

composition. A composition relationship is described as a "Has-a" relationship.

■ ***Expressing Object-oriented Concepts in XML***

One of the benefits of XML over other data formats is that it closely resembles the structure of object data, although the terminology is a little different. Because much of today's design and development is object-based, it's no surprise that when the data is being exchanged today, it is most often represented as XML.

In its development of a data model using XML for the exchange of justice information, the primary goal of BJA's XSTF was to provide a solid *infrastructure* for storing and managing relationships between objects, or in XML terminology, what is referred to as *data entities*.

In the Global Justice XML Data Model (GJXDM), these data entities, their composition, and the relationships between them have been defined precisely and flexibly using two kinds of components: **types** and **properties**. An understanding of the concept of types and properties is the baseline for developing an understanding of GJXDM and is described in module 002, part 2.

The basic concepts and terminology to keep in mind are the following:

- XML *types* define the data structure.
- XML *elements* define data semantics.
- XML *schema* is the specification or the structure for the documents. For example, see figure 2.1.1.

Schema
<pre><xsd:complexType name="PersonNameType"> <xsd:sequence> <element name="PersonGivenName" type="TextType" / > <element name="PersonSurName" type="TextType" / > </xsd:sequence> </xsd:complexType> <xsd:element name="PersonName" type="PersonNameType" / ></pre>

figure 2.1.1

- The XML *instance* defines the payload, as shown in figure 2.1.2.

Instance
<pre><PersonName> <PersonGivenName>John</PersonGivenName> <PersonSurName>Smith</PersonSurName> </PersonName></pre>

figure 2.1.2

Global Justice XML Data Model Structure and Relationships

The entire data model is based on the paradigm that figure 2.1.3 represents:

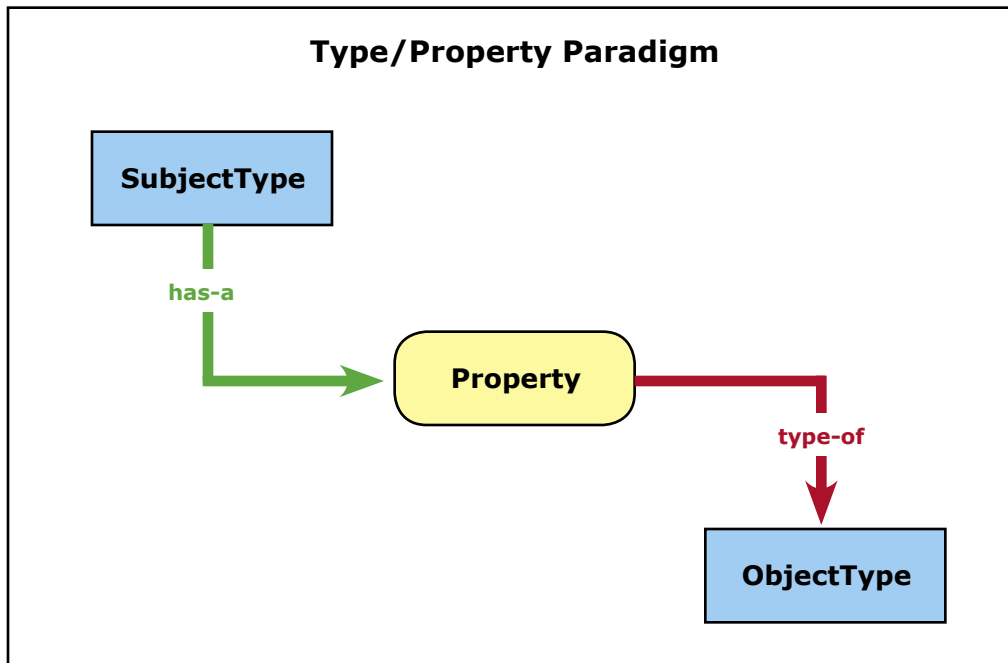


figure 2.1.3

Each property, if complex, has components that each follow some type, and that type may itself be complex. The model cascades all the way down in this manner until one gets to the smallest components, which are simple types, things like strings, text, integers, etc.

The whole model is rooted at SuperType, the most generic object of all. The SuperType is basically any thing. Using a SuperType as a root of the model enables the association of certain attributes common to all of the types. In other words, all types in the model can inherit basic attributes such as "@source Text" (the name or identification of an information resource from which the content came) or "@reportedDate" (the date information was observed, measured, identified, or became known). This is commonly known as **metadata**.

Derived from the SuperType are other types, such as PersonType or ActivityType. Each of these types has properties of its own, and the important objective of component reuse

is achieved by using the OOP concept of inheritance. The model was developed this way to ensure consistency. This structure also will support future technology, such as Resource Description Framework.

GJXDM Object Model Example

The following GJXDM Object Model example illustrates the object-oriented concepts of inheritance and "is-a" and "has-a" relationships.

In figure 2.1.4 on page 21, the SuperType, from which all of the GJXDM types derive, is broken down into types (Subclasses) that inherit elements or attributes from the types above in the hierarchy.

Each SubType has elements or attributes that are specifically associated with that SubType, as illustrated earlier with cat being derived from mammal, and then abstracting SubTypes of cat, such as Siamese.

In figure 2.1.4, the core objects are shown as deriving from the SuperType, therefore, PersonType, OrganizationType, PropertyType, LocationType, and ActivityType would each inherit properties from the SuperType, such as @reportedDate or @sourceText. Each Subclass would have additional pertinent properties, such as PersonName and PersonBirthDate, inherent to PersonType.

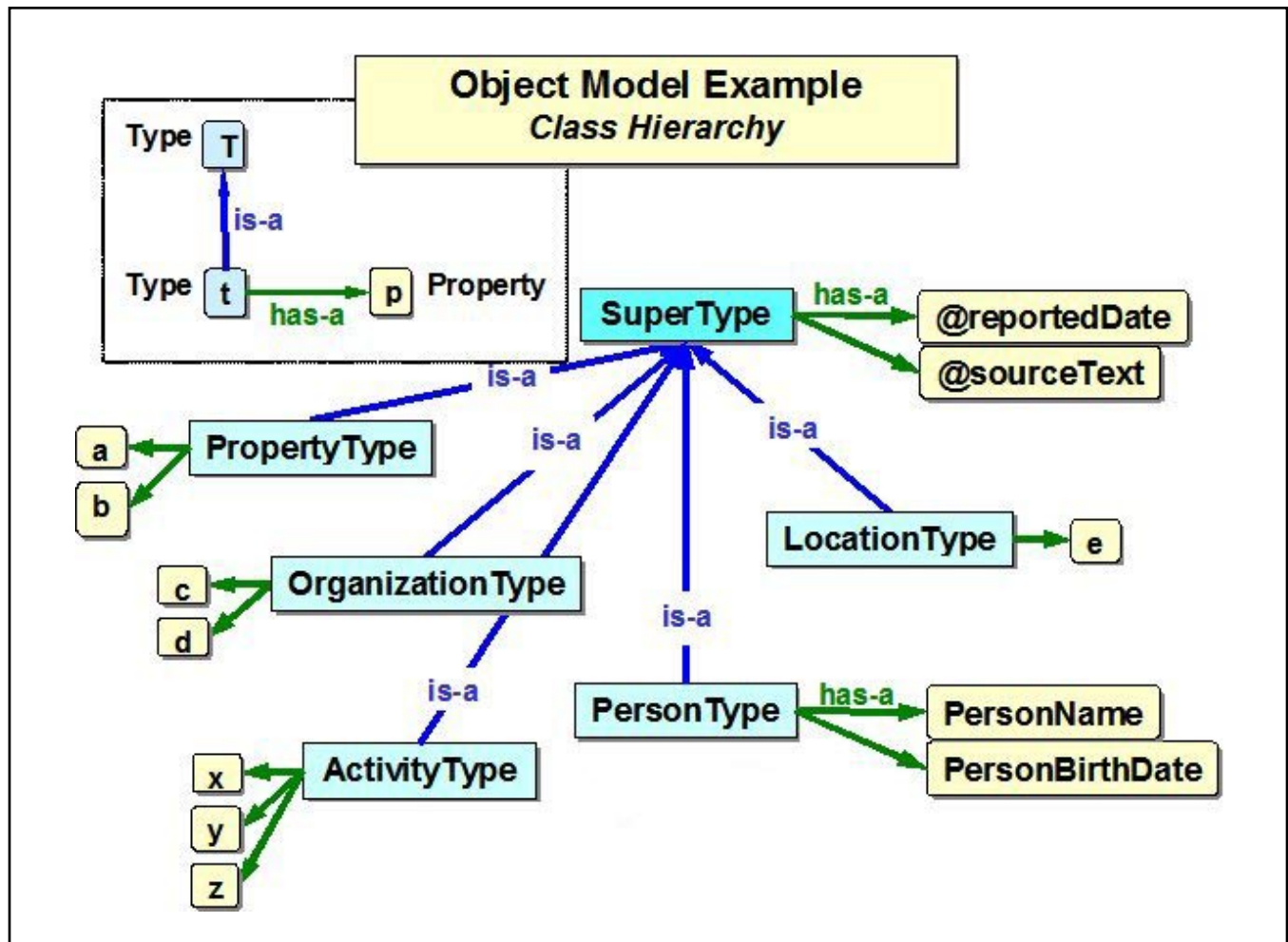


figure 2.1.4

When members of BJA's XSTF developed the core object of a Person, in-depth discussions occurred regarding which elements or attributes should be classified under what categories of persons generally participating or not participating in the justice system.

One extensive discussion revolved around whether or not a *JudicialOfficerType* should inherit *PersonBiometricDetails*, such as a *PersonFingerprint* property from the SuperType (*PersonType*). Obviously, judges have fingerprints, but the discussion focused on whether or not a judge's fingerprints would ever be exchanged in the context of justice- and public safety-related information, notwithstanding the applicability of fingerprint information to employment or human resources sorts of background checks.

Ultimately, BJA's XSTF concluded that with recent technological advances in biometrics, especially with regard to driver's licenses incorporating biometrics such as thumbprints in many jurisdictions, the core Person object should contain all elements and attributes normally associated with any ordinary *driver*,

and the only subclasses that would be abstracted are those with elements specifically related to a particular context within the justice domain.

Figure 2.1.5 further illustrates the conceptual framework of the GJXDM Object Model and how inheritance works. The SuperType, from which all of the types are derived, has properties that are inherited by all SubTypes. Thus, *PersonType* inherits @reportedDate and @sourceText from the SuperType, but has additional properties, such as *PersonName* and *PersonBirthDate*.

SubjectType and *JudicialOfficerType* are both extended from *PersonType*, and thus inherit from the SuperType (@reportedDate, @sourceText) and *PersonType* (*PersonName*, *PersonBirthDate*), while each have some additional properties that are relevant only to the specific SubType, such as *SubjectCriminalTraitDetails* (a set of identifying characteristics a subject has with regard to illegal activity) and *JudicialOfficialBarID* (an identifier assigned to a judicial official after meeting the requirement to practice law in a region).

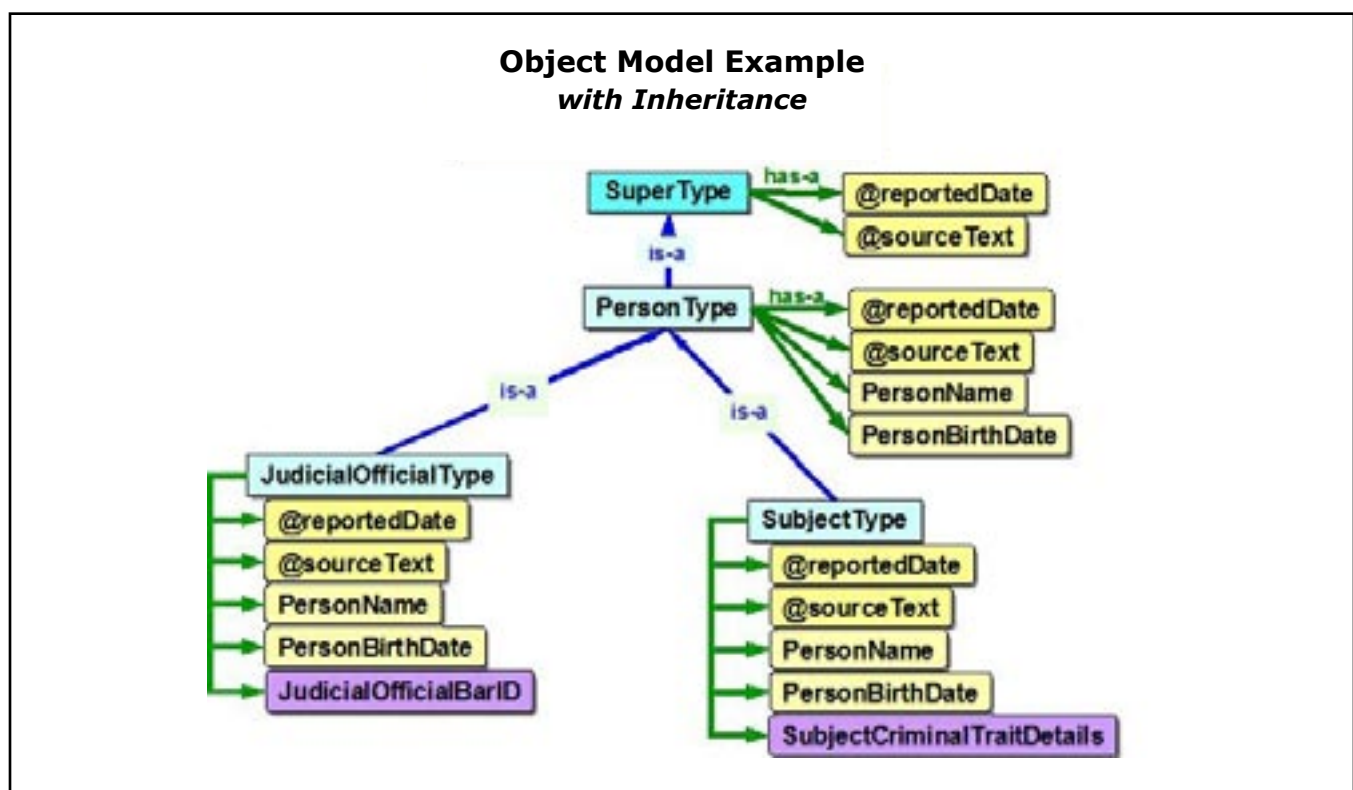


figure 2.1.5

BJA's XSTF has abstracted the PersonType into several specific SubTypes, which are shown in figure 2.1.6.

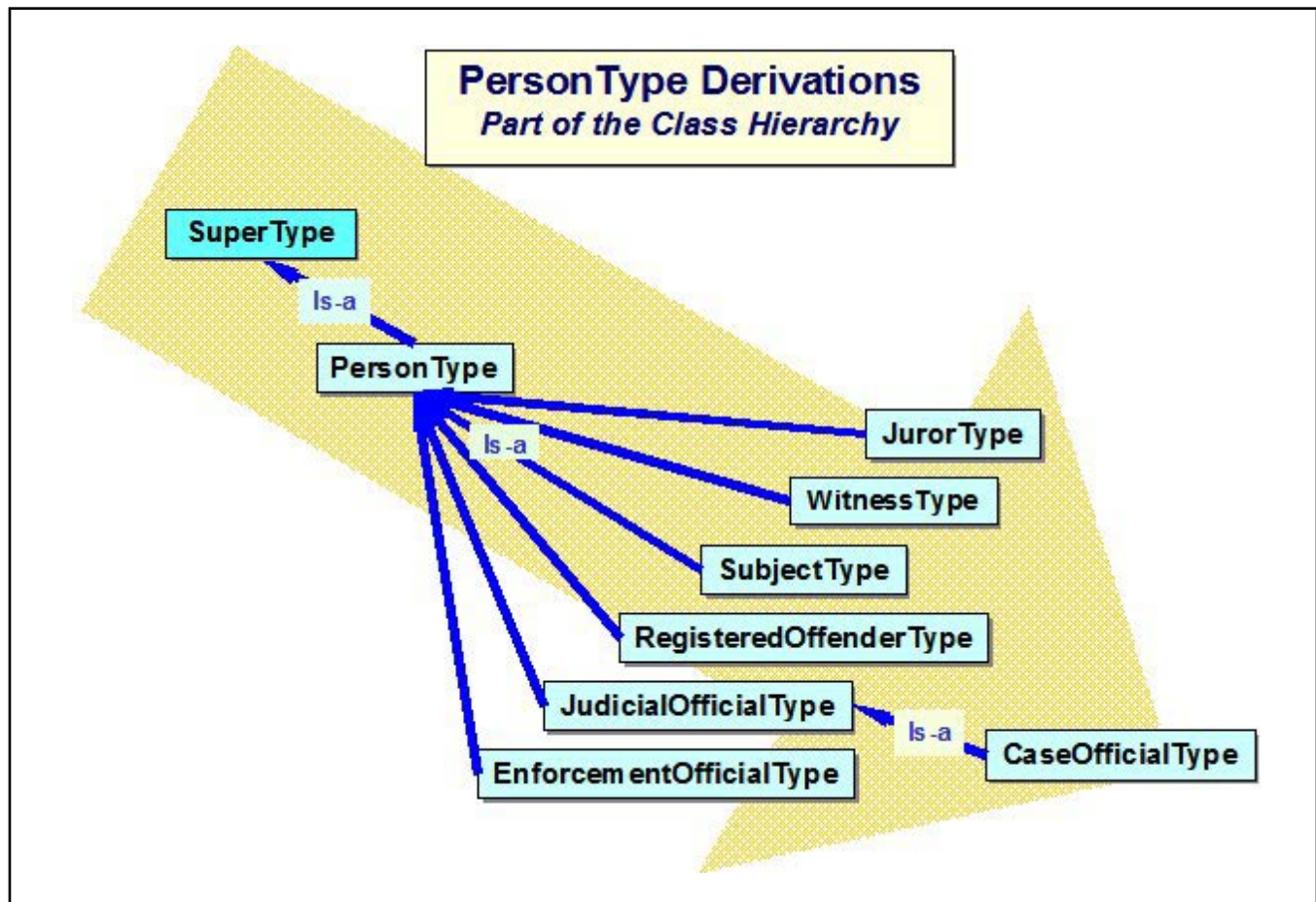


figure 2.1.6

Each SubType provides additional properties specific to the role the person plays in the business processes of justice and public safety. For example, the SubjectType provides properties associated only with a person who is the subject of a criminal justice or justice matter, whether the matter is being investigated or has progressed in the system as a court proceeding. SubjectType is the type for which BJA's XSTF has associated the highest number of elements and attributes in connection with the exchange of justice- and public safety-related data.

XSTF has spent much of its effort in defining the class hierarchy of the object model represented by GJXDM. From approximately 33 source specifications,⁸ which were parsed

by the Georgia Tech Research Institute (GTRI) into more than 16,000 elements, XSTF worked through the process of analyzing, combining, and restructuring elements that are exchanged by jurisdictions throughout the nation into the hierarchical model of more than 2,700 components that comprise GJXDM today.

⁸ Source specifications include, but are not limited to, the Global Reconciliation Data Dictionary (RDD) v1.0.0; Joint Task Force on Rap Sheet Standardization Rap Sheet v2.2; RISSIntel v2.0; LegalXML CourtFiling v1.1; AAMVA Driver History v1.02; NIJ InfoTech v2.0; Los Angeles County (CA) Incident Report schema; SEARCH JIEM data sets; OASIS LegalXML schemas: Arrest Warrant, Charging Document, Sentencing Order, and Uniform Incident Report; Minnesota CriMNet v1.0; National Center for State Courts data element specifications (civil, criminal, juvenile); Maricopa County (AZ) ICJIS Data Dictionary v1.3; Criminal Information Sharing Alliance (CISA) Southwest Border States Data Dictionary (TX, AZ, NM); FBI NCIC 2000 Data Dictionary and Code Tables; and NIBRS Incident Report.

GJXDM Structure

When describing GJXDM, the term “structure” is used often, but what is really meant by that? In this case, ***structure refers to the representation of data dictionary entities in XML Schema***. The key factors to remember are:

- Data model types *become* XML Schema types.
- Properties *become* XML elements and attributes.
- Data model type inheritance *becomes* XML Schema type inheritance.
- Duplication of entities is avoided by using references.

A more comprehensive discussion of the GJXDM structure is provided elsewhere in this *User Guide*. The important concept here is the transition of the underlying data model hierarchy to the representation of the objects in the XML data dictionary (GJXDD) as elements and attributes.

A critical objective of BJA’s XSTF was to define the data elements and attributes as nonambiguously as possible. Take a look at the [GJXDD spreadsheet](#) and note comprehensive definitions for virtually every component in the dictionary.

XSTF is working continuously to provide more complete definitions and examples for all xsd: string elements, as well as identifying and collecting **Keywords** and **Context** definitions to be implemented in the data dictionary. These definitions and examples will help further the utility of the model for justice data exchange. Only when the *semantic meaning* of the data being exchanged is clearly agreed upon will an important goal be accomplished: The accurate information sharing necessary to support decisionmakers in the justice system.

PART 2 Properties and Types

XML Properties and Types

The Global Justice XML Data Model consists of **Properties** and **Types**. *Types* represent real-world entities, such as persons or vehicles (remember objects). When an object is created, it is described as being *instantiated*. A *Property* associates specific characteristics with an *instance* of the Type.

Each property has three components:

1. The **property name** is a unique label applied to the property. Property names are unique within the data dictionary.
2. The **subject type** is the type to which the property applies. For example, with the property Name, if a person has a name, then the subject type of the property Name would be PersonType.
3. The **object type** is the type of the value of the property. For example, if a name is a string, then the object type of the property Name would be StringType.

■ Relationship Between Properties and Types

Figure 2.2.1 illustrates the relationship between properties and types.

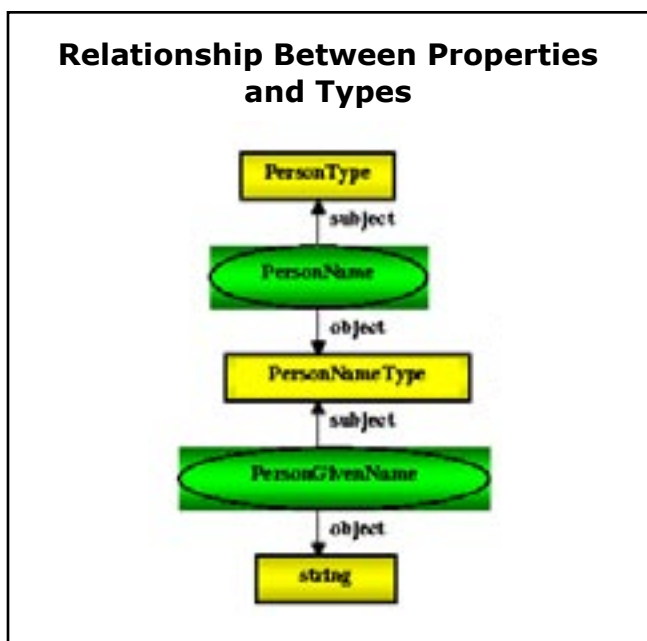


figure 2.2.1

Figure 2.2.2 illustrates the concept of Type "has-a" property, property of "of-type" type.

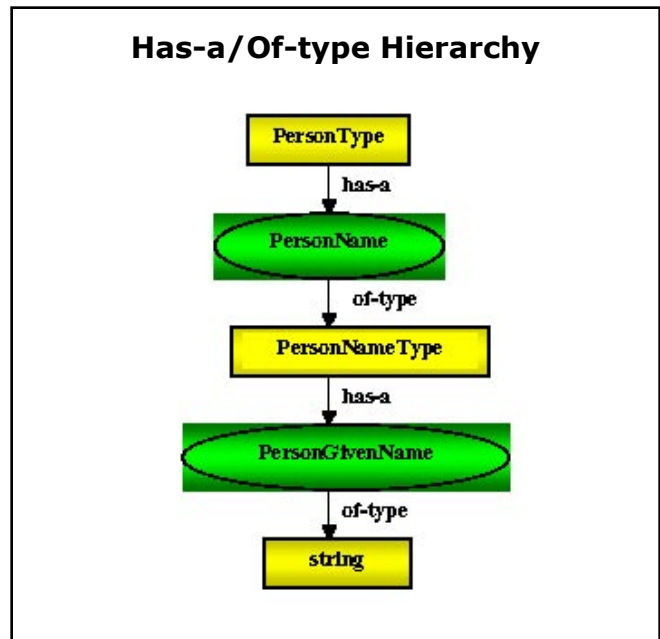


figure 2.2.2

In figure 2.2.2, the *subject arcs* from figure 2.1.1 have changed into *has-a arcs* and also have changed directions. The *object arcs* changed into *of-type arcs* and remained directed the same way as before. This illustrates how the hierarchy can be followed down the chain:

- PersonType has property PersonName.
- PersonName is of type PersonNameType.
- PersonNameType has property PersonGivenName.
- PersonGivenName is of type string.

Note that the *of-type arc* is equivalent to the *object arc* in figure 2.2.1. However, the *has-a arc* is not equivalent to the *subject arc*.

■ Schema

Figure 2.2.3 represents a schema, which says the following:

- A Person *has* a pet, which is a Dog.
- A Person and a Dog each *have names*, and the name is a string.

Note that the PersonType and DogType subjects would be instantiated as a Person of type PersonType, etc.

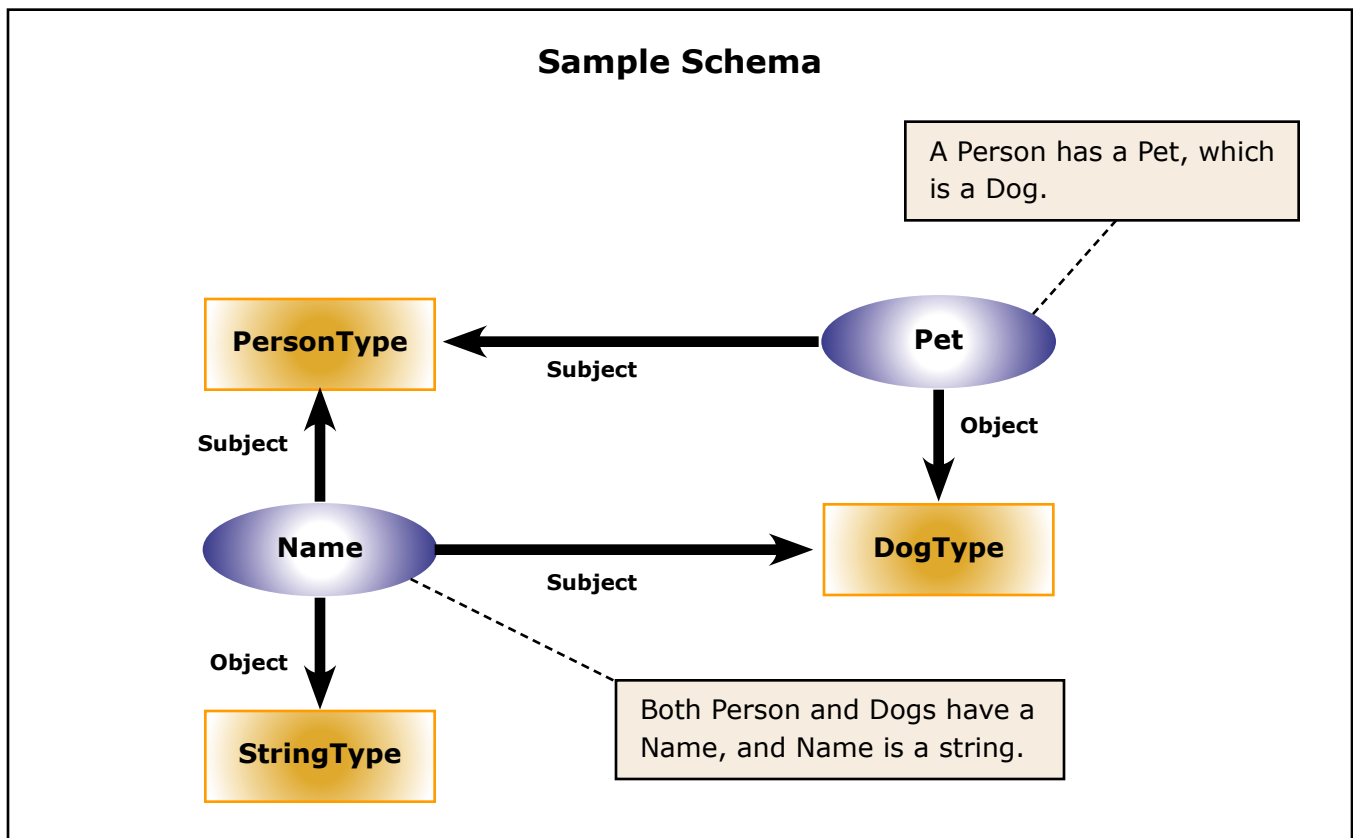


figure 2.2.3

■ Instance

Figure 2.2.4 illustrates a sample *instance*, using the schema previously presented, and makes the following statements:

- John has a pet dog named Abby.
- John, Bob, and Dave all share a pet dog named Mulder.

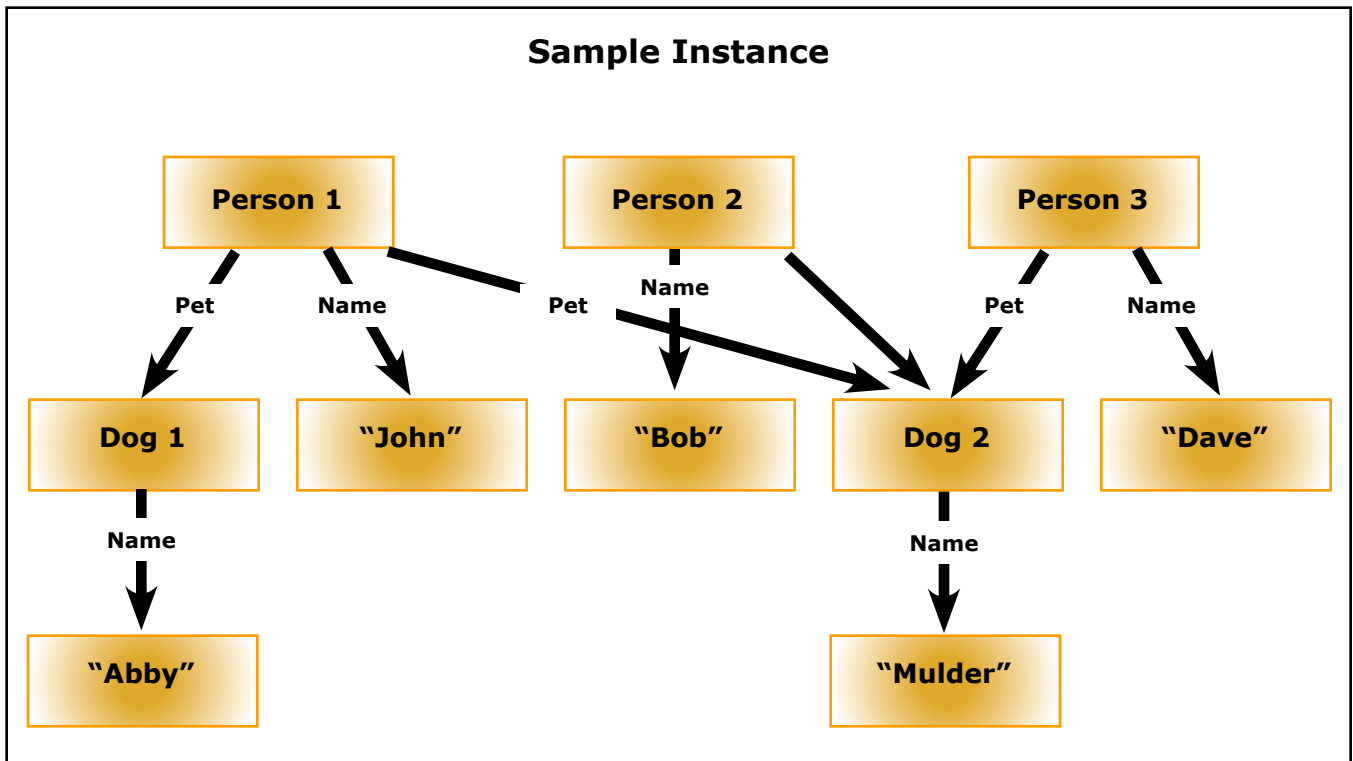


figure 2.2.4

Multiple Subjects

If the **PersonName** **property** has these **subject types**: **PersonType**, **DriversLicenseType**, or **TaxRecordType**, then each of those types either **HAVE**, **CONTAIN**, or are **RELATED TO** a **PersonName**, as shown in figure 2.2.5.

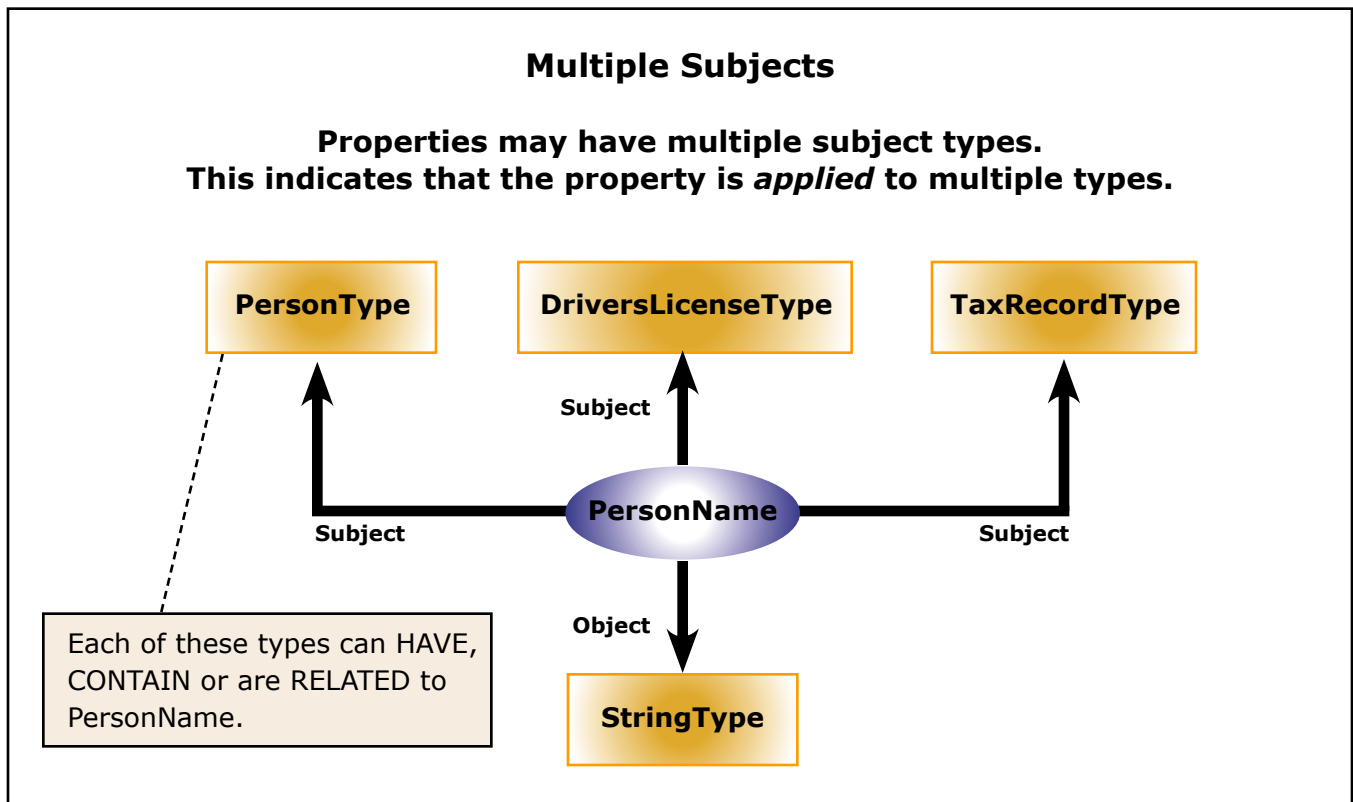


figure 2.2.5

Multiple Objects

In the example illustrated in figure 2.2.6, if the EyeColor property has object types EyeColorCodeType and StringType, then the property may have a *value* of either type.

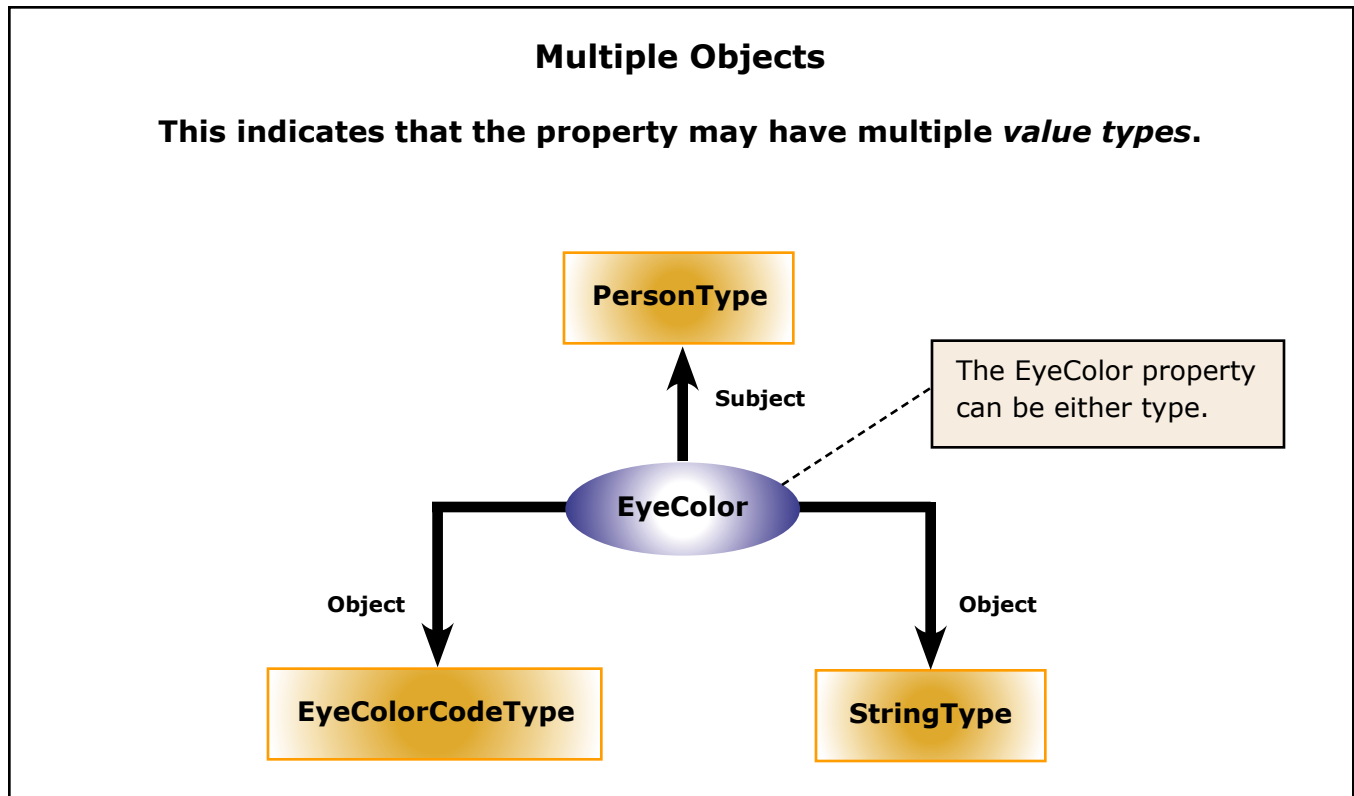


figure 2.2.6

PART 3 XML Schema Definition Standard

XML may be the key to system interoperability, but successful information sharing also requires precise definition of the XML data's content (semantics). XML identifies different structures or data containers by assigning data "tags" to define both the name of a data element (syntax) and the format of the data within that element.

XML, in most simple terms, is a language used to delimit or tag portions of content for a document. *Document* is being used here in very general terms. Document may refer to an electronic file, a message or transaction, a collection of database records, a data stream, etc. Unlike Hypertext Markup Language (HTML), which tags a document primarily with markup used for *presentation* (e.g., font size, typeface, headings, paragraphs, tables, etc.), XML tags a document based on its *content*, (e.g., address, street, city, state, ZIP code, etc.). In fact, XML represents the hierarchical content of a document, where the elements in the example—street, city, state, and ZIP code—are children of a parent structure, address.

The earliest standard for describing an XML document was the **Document Type Definition** (DTD). A DTD described the structure of a document, with the focus on how to structure text within the document.

However, DTD did not provide a way to describe elements exchanged among systems or databases, such as identifying *data types*, defining *validation ranges*, or defining how to *encapsulate elements with namespaces*. XML is *extensible* in the sense that custom tags may be defined to represent the contents of a specific document. The World Wide Web Consortium (W3C), with the goal of exchanging semantically precise data among systems, went on to develop a more robust standard. Today, the primary mechanism for defining data and structure is the **XML Schema Definition** (XSD).

Key Terms

HTML: A language that enables a document to be "tagged," primarily with markup used for presentation (e.g., font size, typeface, and headings).

DTD: A W3C-recommended document-centric XML schema language.

XSD: A W3C-recommended specification for defining the structure, content, and semantics of XML documents. Defines a richer set of datatypes than the DTD. XML Schemas support namespaces.

Data Type: A specification of the permissible content for a class of objects, where the content can be composed of one or more literal values (e.g., positive integer), or any complex data structure (e.g., hierarchy of child elements within an XML core component).

Element: The fundamental building block of an XML document. XML elements can contain other elements and/or text data. XML elements are composed of a start tag, content, and end tag.

Attribute: A characteristic of an object or entity. An object's attributes are said to describe the object and are often specified in terms of their physical traits, such as size, shape, weight, and color, etc., for real-world objects. Objects in cyberspace might have attributes describing size, type of encoding, network address, etc.

XML schemas express shared vocabularies and allow computers to follow precise business rules. In XML Schema, an **element** refers to the construct of tag, data, and end tag. Elements are combined to form data containers or objects. An **attribute** is a name and value pair that can be associated with an XML tag to provide additional information about the tag. When a schema is created, users are able to define what the XML data should look like and also *validate* the XML data transmitted or received.

An XML schema defines and dictates what content is permitted in an XML document, called the *instance*. Schemas may be quite general, but they also may be very exact and sophisticated in defining the content of an XML document. With its ability to represent hierarchical relationships and its extensibility, XML provides a powerful method for representing complex collections of data. With its flexibility and specificity, it becomes very important to be able to automatically determine—via validation—if the contents of an XML document are acceptable and in their proper order/relationship, which is also accomplished using XML schema.

XSD provides the highest degree of data exchange functionality. XSD provides a way to define the structure, content, and semantics of XML documents.

XSD also specifies *type* information for elements; defines primitive types, such as *decimal*, *string*, and *time*; and allows extension of the type system by defining unique *simple* and *complex types*.

XSD was approved as a [W3C Recommendation](#) on May 2, 2001. Documentation on the XSD standard is available on the W3C site, including a primer and specifications for structures and data types.

Namespace: The solution to naming conflicts in XML. Using XML namespaces can help alleviate issues that arise where XML elements and attributes use identical names. XML namespaces help to identify and resolve conflicts between elements that have the same name but mean different things. A namespace is a domain that contains a set of XML element names.

Primitive Types, as distinct from composite types, are datatypes provided by a programming language as basic building blocks. Typical primitive types include Character, Integer, Boolean, String, Reference, etc.

PART 4 Global Justice XML Data Model Naming Conventions

When designing GJXDM, BJA's XSTF was faced with the challenge of creating a set of specifications that would clearly define the structure of the justice-related data contained therein. Many decisions, such as whether to use an object-oriented model, had to be made. One of the most important factors was how to implement an appropriate and *nonambiguous* naming scheme.

Following extensive research and analysis, XSTF has defined elements in the data dictionary in accordance with the most appropriate published and established standards for data exchange, including:

- Standard specifications from public standards organizations.
- Specifications from government bodies.
- Preexisting data systems.
- *De facto* standards and common usages by the community.

Source Standards

The primary standard utilized by the XSTF is *ISO 11179*. The International Organization for Standardization (ISO) is the world's largest developer of standards. ISO is a network of the national standards institutes of 147 countries and occupies a special position between the public and private sectors. Many of ISO's member institutes are part of the governmental structure of their countries or are mandated by their government. Other members have their roots uniquely in the private sector, having been set up by national partnerships of industry associations. Therefore, ISO is able to act as a bridging organization in which a consensus can be reached on solutions that meet both the requirements of business and the broader needs of the public. The ISO 11179 specification ". . . describes the standardizing and registering of data elements to make the data understandable and sharable. . . ." and,



thus, is a core standard guiding the selection of data tag names used in GJXDM.

ISO 11179 provides guidelines for the naming and definition of data elements, as well as information about the metadata captured about data elements. Part 5 of the ISO 11179 Standard establishes a methodology for naming items in data dictionaries. Names in GJXDM have been constructed using the object class, term property, and representation terms, as defined by ISO 11179.

The standard tag names have been constructed in upper camel case (UpperCamelCase), with no spaces, dashes, underscores, or other punctuation. Camel casing is called such because the "hump," or uppercase letters, appear in the middle of the word. Java and many other OOP languages use camel casing.

Some of the additional standards XSTF uses include, but are but are not limited to:

- Organization for the Advancement of Structured Information Standards (OASIS) Electronic Business XML (ebXML).
- [Federal XML Developer's Guide.](#)
- [U.S. National Archives and Records Administration \(NARA\) Records Management.](#)
- [Department of Defense 5015.2-STD Design Criteria Standard for Electronic Records Management.](#)

Refer to the [Office of Justice Programs' \(OJP\) Information Technology \(IT\) Initiatives web site](#) for a comprehensive list of all source standards used in GJXDM's development.

Components of Global Justice XML Data Model Tag Names

In accordance with ISO 11179, each GJXDM item name consists of several parts, which are illustrated in figure 2.4.1:

- **Object class term.** The object class term represents the specific real-world object to which the property is applicable.
- **Property term.** The property term is a plain-language summary of the quality that the property represents. Example property terms would include *Hair Color*, *Hair Appearance*, and *Tax Identifier*.
- **Representation term.** The representation term describes, from a very high level, the *form* of the data represented. This term is taken from a list of ebXML representation terms, which is included in the following section on **ebXML Representation Terms**. *Representation terms not on the list are not valid.* The point of the representation term is not to specify the exact type of the represented value. It is instead intended to give the person reading the property name a clue as to what the property is for and what it might mean.

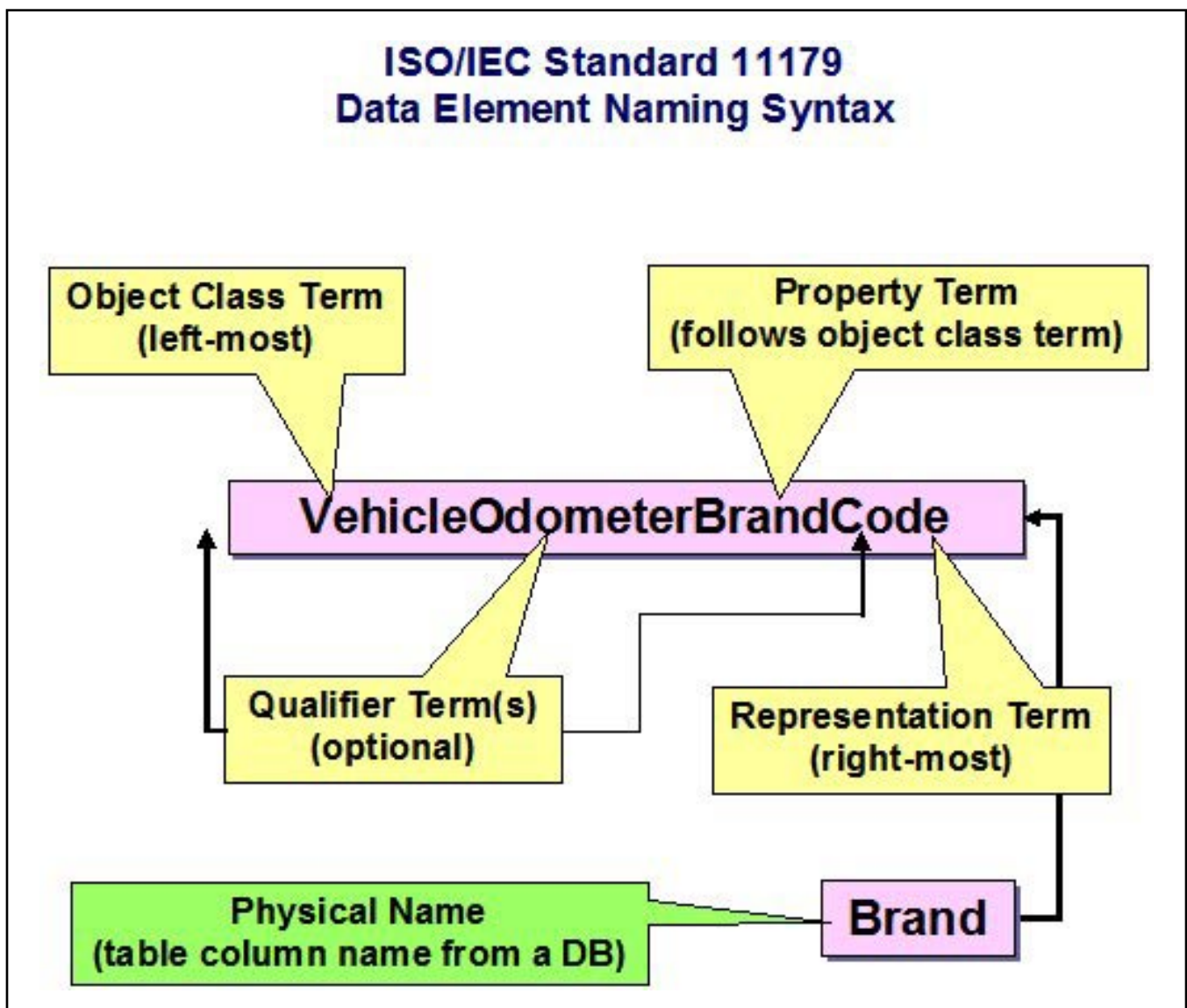


figure 2.4.1

The ISO 11179 specification assumes that instances of fields are *local* to a specific object class in the database. For example, in a law enforcement record management system, the last name contained in an enforcement official type would be different than the last name of a subject type.

The GJXDM Data Dictionary is defined to provide properties with *global* meaning, that can be reused as subjects of many types throughout the data dictionary. To make the 11179 specification relevant and useful to this data dictionary, XSTF uses the object class term to refer to real-world objects instead of specific object types in a database. This permits *reuse* and global meaning while still ensuring a useful context for understanding the applicability of a property.

Another core standard the XSTF uses is the OASIS ebXML standard.

ebXML Representation Terms

GJXDM representation terms are based on the ebXML representation terms from the *Core Components Technical Specification, Part 1*, version 1.8, dated February 8, 2002. These clearly lay out a concise set of representation terms, in line with the ISO 11179 standard, and are provided here:

- **Amount.** A number of monetary units specified in a currency where the unit of currency is explicit or implied.
- **Code.** A character string (letters, figures, or symbols) that for brevity and/or language independence may be used to represent or replace a definitive value or text of an attribute. Codes usually are maintained in code lists per attribute type (e.g., color).
- **Date.** A day within a particular calendar year (ISO 8601).
- **Date Time.** A particular point in the progression of time (ISO 8601).
- **Graphic.** A diagram, graph, mathematical curve, or similar representation.
- **Identifier.** A character string used to establish the identity of and distinguish uniquely one instance of an object within an identification scheme from all other objects within the same scheme. [Note: Type shall not be used when a person or an object is identified by its name. In this case, the Representation Term "Name" shall be used.]
- **Indicator.** A list of two, and only two, values that indicate a condition such as on/off, true/false, etc. (synonym: "Boolean").
- **Measure.** A numeric value determined by measuring an object. Measures are specified with a unit of measure. The applicable unit of measure is taken from UN/ECE Rec. 20.
- **Name.** A word or phrase that constitutes the distinctive designation of a person, place, thing, or concept.
- **Percent.** A rate expressed in hundredths between two values that have the same unit of measure.
- **Picture.** A visual representation of a person, object, or scene.
- **Quantity.** A number of nonmonetary units. It is associated with the indication of objects. Quantities need to be specified with a unit of quantity.
- **Rate.** A quantity or amount measured with respect to another measured quantity or amount or a fixed [garbled in the PDF] or appropriate charge, cost, or value (e.g., U.S. dollars per hour, U.S. dollars per Euro, kilometer per liter, etc.).
- **Text.** A character string generally in the form of words of a language.
- **Time.** The time within a (not specified) day (ISO 8601).
- **Numeric.** Information that is assigned or is determined by calculation, counting, or sequencing. It does not require a unit of quantity or a unit of measure.

OASIS ebXML also specifies the following aggregate representation terms:

- **Detail.** This term is not being used in GJXDD.
- **Content.** This term is not being used in GJXDD.
- **Type.** XSTF is using the term "Type" in the data dictionary, but not in the ebXML context of naming properties. In GJXDD, types and properties are maintained as distinct entities. A type is defined as an independently instantiable collection of properties. (Remember, when an object is created, this is described as being *instantiated*. A property associates specific characteristics with an *instance* of the type.) GJXDM types are bound to properties, either as subjects (type "has-a" property), or as objects (property has value of some type).

Additional Representation Type

XSTF has added the following aggregate representation type:

- **Group** - This has been defined as a collection of properties that are closely associated in some way. A type having a property that is a group is exactly equivalent to the type having all of the properties that are included in that group.

PART 5 Namespaces

Namespaces are the solution to naming conflicts (also known as collisions) in XML. Using XML namespaces can help alleviate issues that arise where XML elements and attributes use identical names.

XML namespaces help to identify and resolve conflicts between elements that have the same name but mean different things. **An XML namespace is a domain that contains a set of element declarations and type definitions.**

An analogy is women's clothing sizes. A size eight is not always the same size eight, because women's clothing is designed in junior, misses, and petites. These various size eights, which may represent different specifications, can exist in several domains.

Internet e-mail addresses have had to solve a similar issue. Each address must be unique. However, many names, such as "John Smith," are ubiquitous. So some information must be added to John Smith to set it apart from every other John Smith, as shown in figure 2.5.1.

Adding Information to E-mail Addresses

Why are all the world's e-mail addresses for "John Smith" unique?

John.Smith@gatech.edu

John.Smith@yahoo.com

John.Smith@earthlink.com

John.Smith1@earthlink.com

figure 2.5.1

XML uses namespaces to uniquely identify the origin and definition of elements.

Each XML schema provides a unique identification (ID) in the form of a **Unified Resource Identifier (URI)**, which points to an external file that defines the namespace.

The URI may either be a **Universal Resource Locator (URL)**, which points to a web server, or a **Universal Resource Name (URN)**, which names a resource but does not specify a de-referenceable network object.

The URI is defined in the format illustrated in figure 2.5.2.

Unified Resource Identifier in XML Schema

Protocol part	Host part	Path part
http://	dir.yahoo.com	/News_and_Media/Weather/By_Region/U_S__States/
urn:	justicexml.ojp.gov	:jdd:3.0

figure 2.5.2

- The **protocol part** either specifies a real network transfer protocol, such as HTTP or FTP for a URL, or the string "urn:" for a URN.
- The **host part** specifies a registered host name, resolvable through Domain Name Service (DNS).
- The **path part** specifies a unique ID on that host, and its meaning is under the control of the parties that control the host.

GJXDM 3.0 Namespace

Figure 2.5.3 illustrates the GJXDM 3.0 namespace, compared to the State of Georgia Department of Motor Vehicles namespace.

Namespace Comparison

GJXDM v3.0 namespace	=	"http://www.it.ojp.gov/jxdm/3.0"
Georgia DMV namespace	=	"http://www.dmv.state.ga.us/dmvns"
In a schema	xmlns:jxdm	= "http://www.it.ojp.gov/jxdm/3.0"
In a schema	xmlns:gadmvm	= "http://www.dmv.state.ga.us/dmvns"
Refer to components	<jxdm:VehicleBrand>	<gadmvm:VehicleBrand>

figure 2.5.3

Note that we can use both the GJXDM VehicleBrand element and the VehicleBrand from the State of Georgia in the same schema because both jxdm: and gadmvm: are replaced by the full namespace. In XML schema, the "xxx:" designation is simply a shorthand for the full namespace.

Figure 2.5.4 demonstrates a schema with element names (VehicleBrand) that look the same but are actually different components because their namespaces are different.

Schema With Different Components

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
  targetNamespace="http://www.it.ojp.gov/jxdm/reg/1.1"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:reg="http://www.it.ojp.gov/jxdm/reg/1.1"
  xmlns:jxdm="http://www.it.ojp.gov/jxdm/3.0"
  xmlns:gadmvm="http://www.dmv.ga.state.us/dmvns">
  <xsd:import
    namespace="http://www.it.ojp.gov/jxdm/3.0"
    schemaLocation="jxdm/3.0/jxdm.xsd" />
  <xsd:element name="reg:Registration">
    ...
  <xsd:element name="jxdm:VehicleBrand">
    ...
  <xsd:element name="gadmvm:VehicleBrand">
    ...
</xsd:schema>
```

Different Components

figure 2.5.4

Namespaces and Versioning

Namespaces can also be used for version control to ensure that changes to component names and definitions between versions do not collide or confuse, as shown in figure 2.5.5.

Namespaces for Version Control

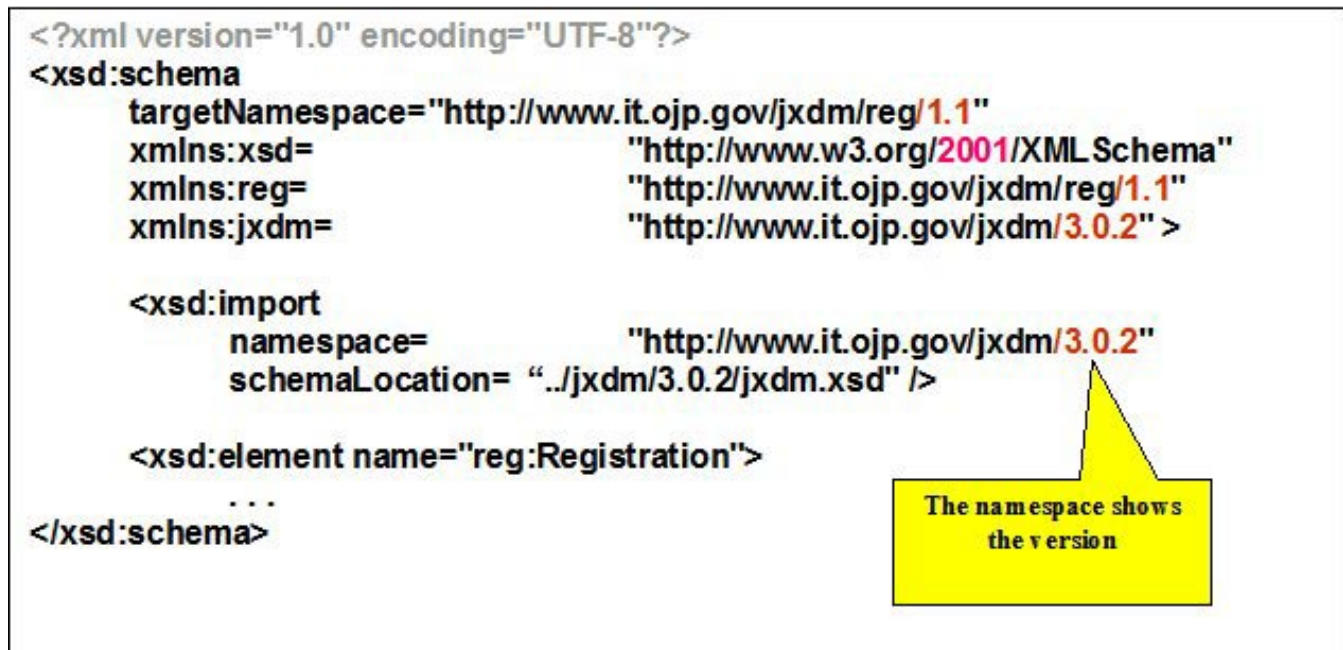


figure 2.5.5

The objective is to support a potential requirement for something from a previous version. Because the versions are kept in separate namespaces, this can be done without the risk of a name clash. This is consistent with the W3C schema standard, which allows the use of namespaces to incorporate multiple versions, which also is illustrated in figure 2.5.5.

The [current operational release of GJXDM](#) (Version 3.0.2) is available on the OJP IT Initiatives web site. Please refer to this site for up-to-date technical release notes and examples of using namespace.

PART 6 Metadata

Metadata: Data About Data

Metadata, or data about data, defines information that supports the actual content of data instances. These data may include details about the origination of data, who authenticated it, or when it was created. For the sake of this discussion, information that helps specify approximate information—such as the specification of ranges, precision, and patterns, as well as units of measure—is included.

GJXDM metadata methodology uses the same basic Types/Property described earlier, but requires a bit of rigor when creating definitions. Failure to keep in mind the difference between simple and complex types will result in unexpected XML structures.

As an example, consider the data structure describing a Visitor in figure 2.6.1.

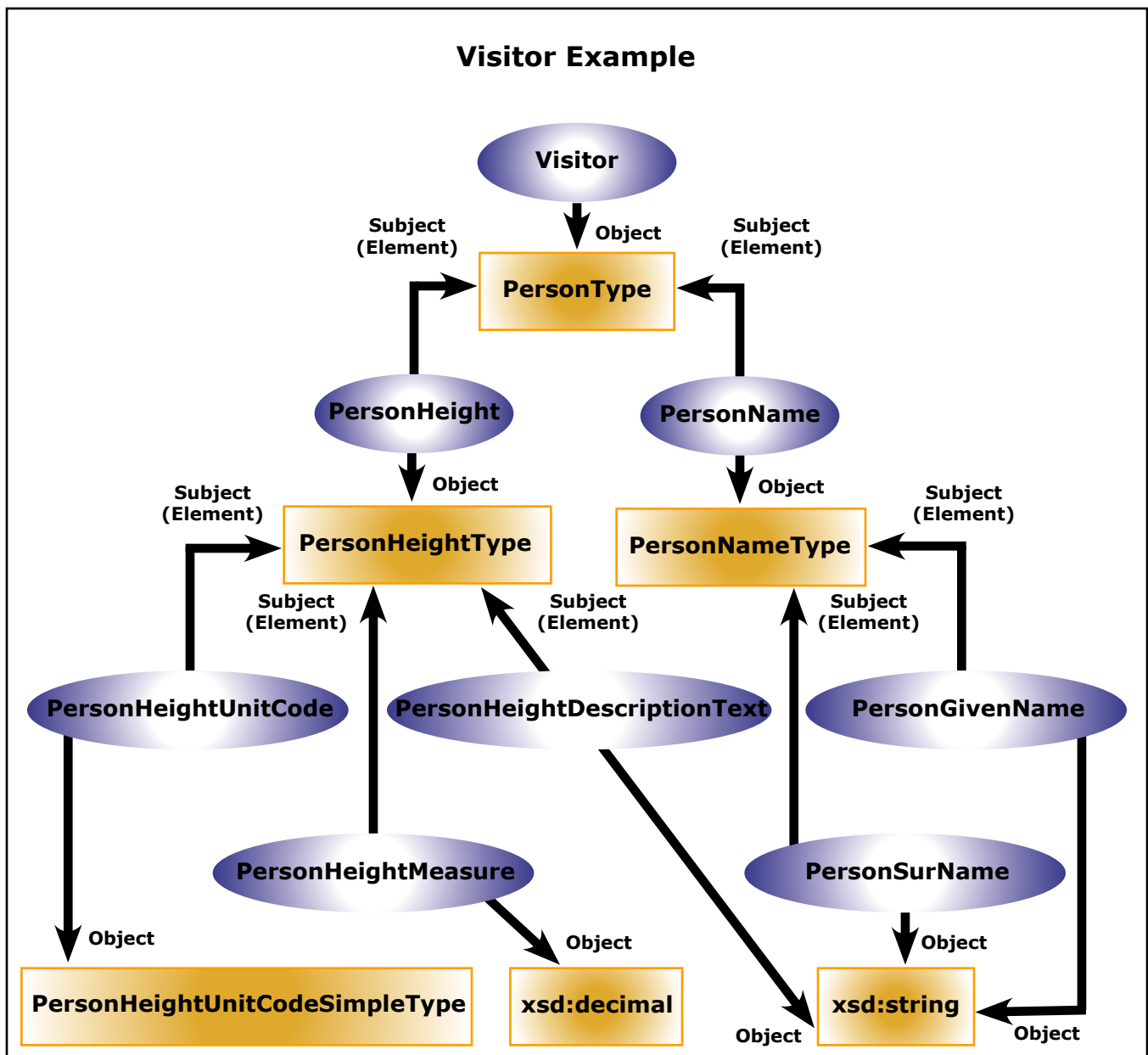


figure 2.6.1

In XML Schema terms, the representation of that data model would appear as shown in the Visitor XSD Schema example below. Note that the use of the schema as depicted below mirrors how this data would actually be defined in the GJXDD XSD. Also note that the namespace references have been omitted for simplicity.

```

<xsd:complexType name="PersonType">
  <xsd:complexContent mixed="false">
    <xsd:sequence>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonName>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonHeight>
    </xsd:sequence>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="PersonNameType">
  <xsd:complexContent mixed="false">
    <xsd:sequence>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonSurName>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonGivenName>
    </xsd:sequence>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="PersonHeightType">
  <xsd:complexContent mixed="false">
    <xsd:sequence>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonHeightUnitCode>
      <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonHeightMeasure>
    <xsd:element minOccurs="0" maxOccurs="unbounded" ref=PersonHeightDescriptionText>
  </xsd:sequence>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="PersonName" type="PersonNameType"/>
<xsd:element name="PersonHeight" type="PersonHeightType"/>
<xsd:element name="PersonSurName" type="xsd:string"/>
<xsd:element name="PersonGivenName" type="xsd:string"/>
<xsd:element name="PersonHeightMeasureCode" type="PersonHeightUnitCodeSimpleType"/>
<xsd:element name="PersonHeightMeasure" type="xsd:decimal"/>
<xsd:element name="PersonHeightDescriptionText" type="xsd:string"/>
<xsd:element name="Visitor" type="PersonType"/>

```

An XML representation conforming to Visitor XSD would provide an instance, such as the one illustrated below.

```
<Visitor>
  <PersonName>
    <PersonGivenName>John</PersonGivenName>
    <PersonSurName> Smith </PersonSurName>
  </PersonName>
  <PersonHeight>
    <PersonHeightMeasure>75</PersonHeightMeasure>
    <PersonHeightUnitCode>Inches</PersonHeightUnitCode>
    <PersonHeightDescriptionText>Witnessed October 19,1994</PersonHeightDescriptionText>
  </PersonHeight>
</Visitor>
```

In looking at this representation, however, PersonHeightMeasure and PersonHeightUnitCode seem somewhat complicated. With a little bit of thought, a standard length measurement type can be defined that can be reused throughout the dictionary, as illustrated in figure 2.6.2. The new object is used as the object type of PersonHeightMeasure. This new type will have a simple type of decimal to store the value of the length and a unit (an attribute that tells if the measure is in inches or centimeters).

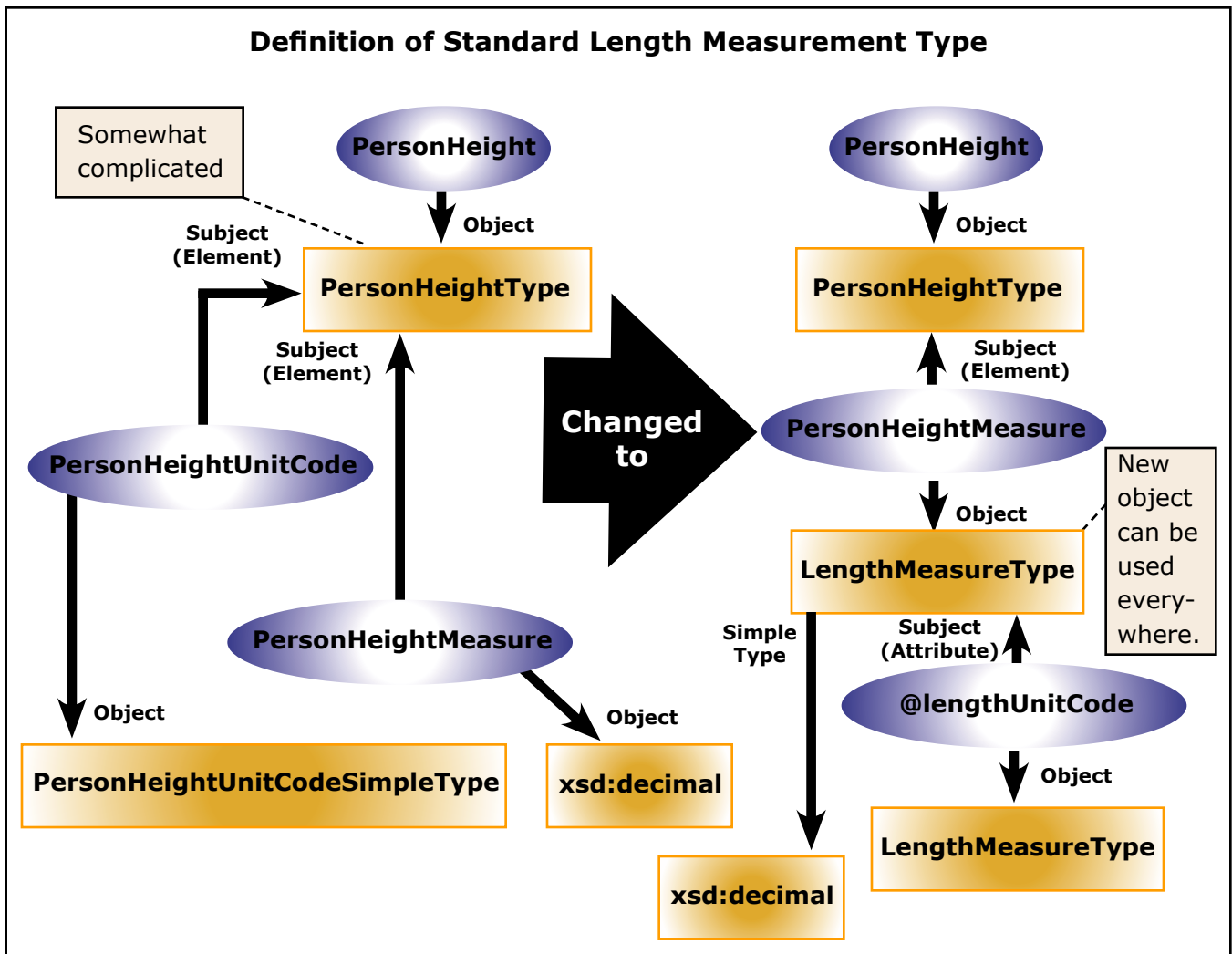


figure 2.6.2

The new schema will create instances such as:

```
<Visitor>
  <PersonName>
    <PersonGivenName>John</PersonGivenName>
    <PersonSurName> Smith </PersonSurName>
  </PersonName>
  <PersonHeight>
    <PersonHeightMeasure @lengthMeasureUnitCode="Inches">75</PersonHeightMeasure>
    <PersonHeightDescriptionText>Witnessed October 19, 1994</PersonHeightDescriptionText>
  </PersonHeight>
</Visitor>
```

What if the HeightDescriptionText information was in a language other than English? It may be useful to encode text strings with a language field, which could provide automatic translation,

or similar technologies. Figure 2.6.3 illustrates that the creation of a new type, TextType, will allow reuse as an object type of other properties.

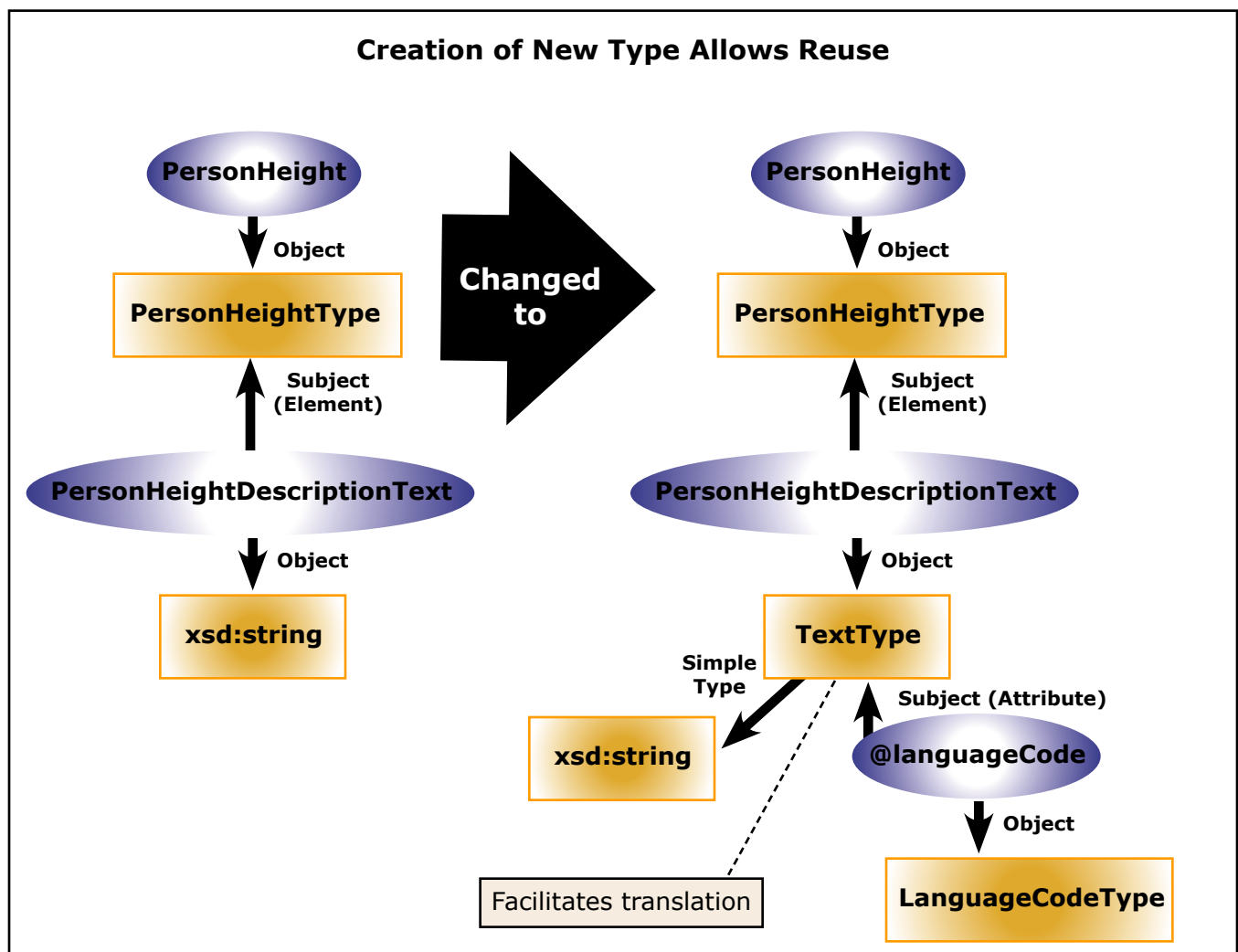


figure 2.6.3

This will allow instances such as:

```
<PersonHeight>  
  <PersonHeightMeasure LengthMeasureUnitCode="Inches"> 75 </PersonHeightMeasure>  
  <PersonHeightDescriptionText @languageCode="eng"> Witnessed October 19, 1994  
    </PersonDescriptionHeightText>  
</PersonHeight>
```

Or

```
<PersonHeight>  
  <PersonHeightMeasure LengthMeasureUnitCode="Inches"> 75 </PersonHeightMeasure>  
  <PersonHeightDescriptionText @languageCode="spa">>  
    Atestiguado De Octubre El 19 De 1994  
  </PersonHeightDescriptionText>  
</PersonHeight>
```

A better way to handle text describing the validity of a data field would be to add an @validOnDate attribute to the PersonHeightType, as illustrated in figure 2.6.4.

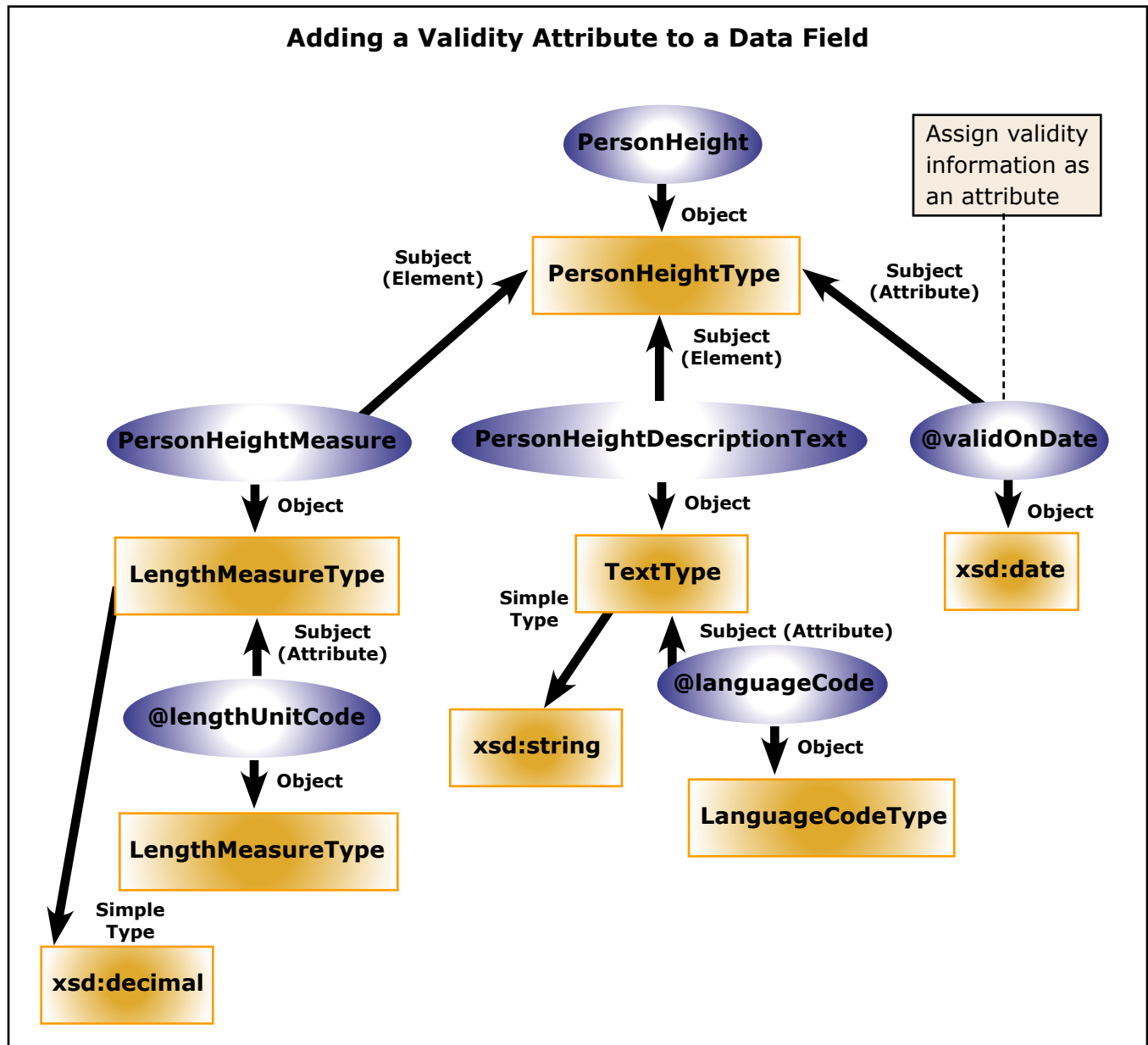


figure 2.6.4

This would provide an instance, such as:

```
<Visitor>
  <PersonName>
    <PersonGivenName>John</PersonGivenName>
    <PersonSurName> Smith </PersonSurName>
  </PersonName>
  <PersonHeight @validOnDate="2003-12-03">
    <PersonHeightMeasure @lengthUnitCode="Inches"> 75
  </PersonHeightMeasure>
    <PersonHeightDescriptionText @languageCode="eng">
      Observed by 12 security cameras
    </PersonHeightDescriptionText>
  </PersonHeight>
</Visitor>
```

If the @validOnDate metadata attribute is useful, users could determine that every data field should have it available. However, adding it to each type would be duplicative and not leverage reuse. It is more efficient to create a base type that includes the @validOnDate attribute and derive the other types from that base type.

Note that PersonGivenName and PersonSurName were of type xsd:string, which is a simple type that cannot carry metadata, so this type is wrapped in a StringSimple Type.

Figure 2.6.5 illustrates the hierarchy.

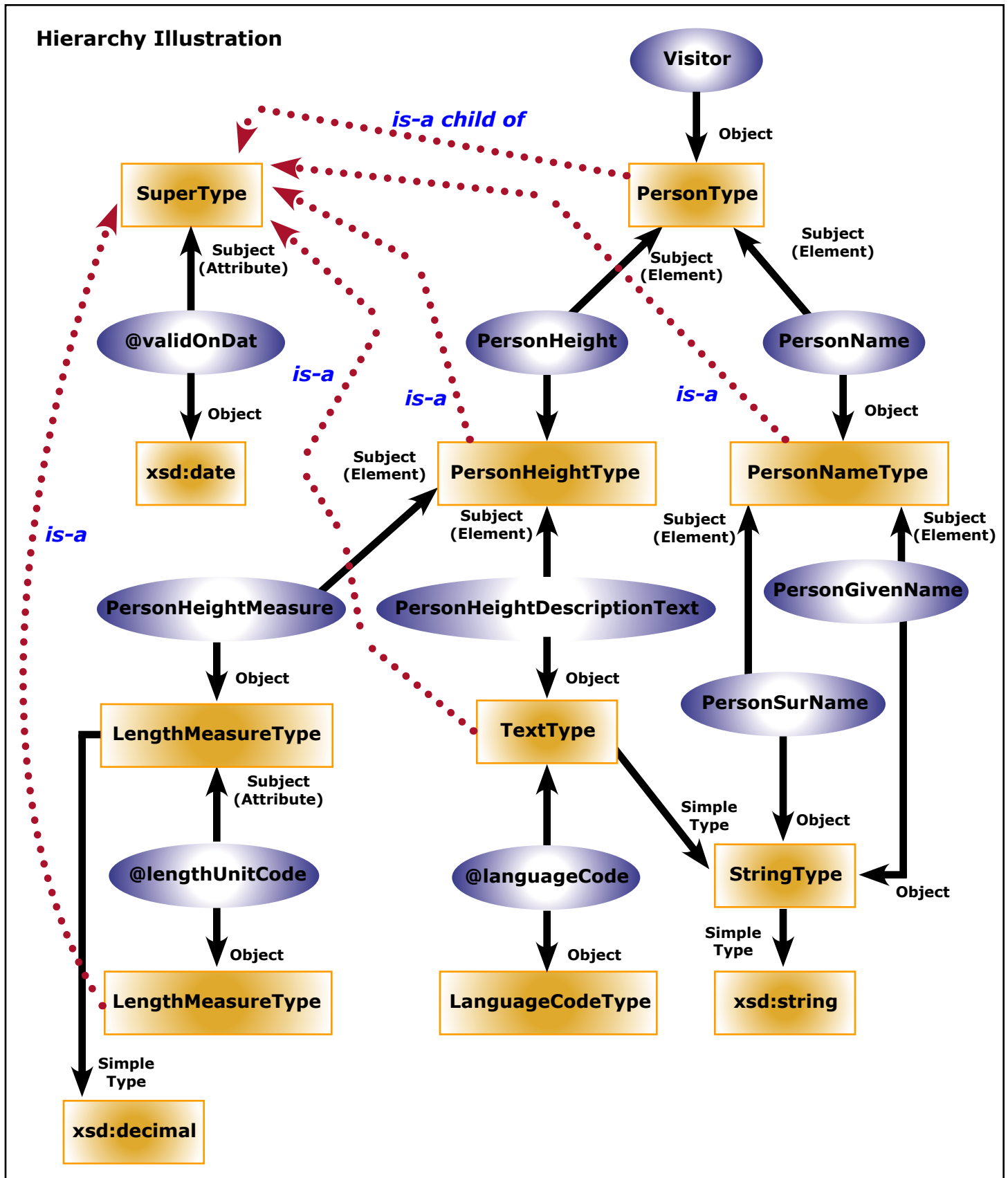


figure 2.6.5

Does this look complicated? The ability to attach metadata to any field is very important to support *semantically rich* information sharing.

- Metadata allows justice systems to assemble data from various sources but still maintain origination information.
- Metadata allows systems to operate from business rules on data access, ownership, and dissemination that are defined by justice stakeholders.
- Metadata can provide information about when the data were generated and by whom or how long the information must be retained by the agency of record.

Metadata is represented in these examples as either an *element* or an *attribute*, both of which XML schema allows for tagging data in XML instances. A more detailed explanation of this topic is provided in module 003, part 5, *XML Schema Elements Versus Attributes*.

How is Metadata Used?

- To index, search, retrieve, navigate (e.g., topical information such as name, title, class, or category).
- To assess data utility through data origin or pedigree (e.g., accuracy, precision, reliability, belief, timeliness).

SuperType Metadata

Figure 2.6.6 illustrates GJXDM SuperType metadata attributes that are inherited by all data elements in the model.

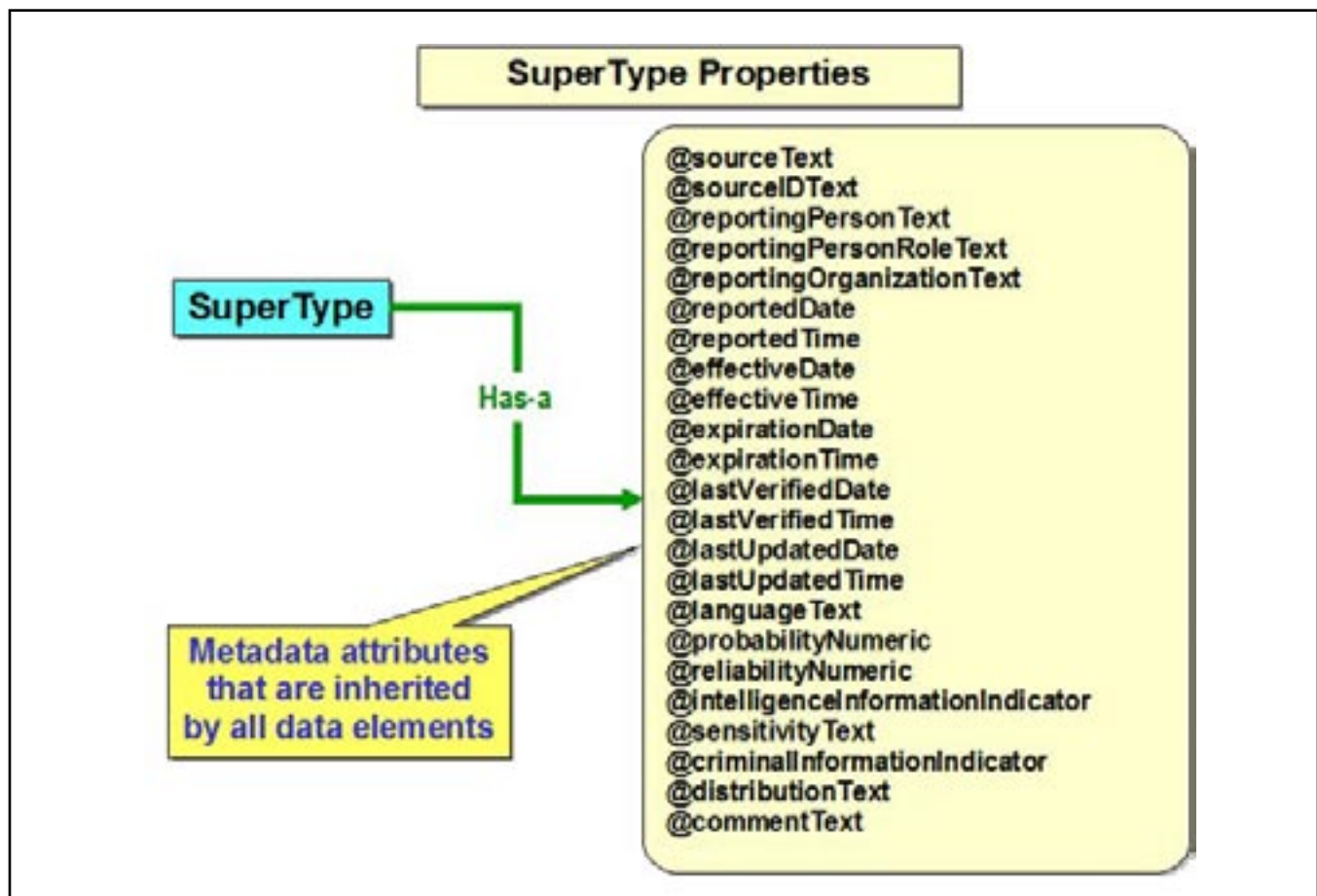


figure 2.6.6

MODULE 003

Global Justice XML Data Model Content

PART 1 Architecture of the Global Justice XML Data Model

The Department of Justice's XML Structure Task Force (XSTF) has defined the structure of the Global Justice XML Data Model (GJXDM) as the representation of data dictionary entities in eXtensible Markup Language (XML) Schema, where:

- Data model types become XML Schema types.
- Properties become XML elements and attributes.

- Data model type inheritance becomes XML Schema type inheritance.
- Duplication of entities is avoided by using references.

Figure 3.1.1 illustrates the overall architecture of GJXDM, which allows for local extensions and instances to be derived from the source standard. This guide identifies each component and provides explicit guidance on how to build XML instances that meet the requirements of jurisdictional justice exchanges.

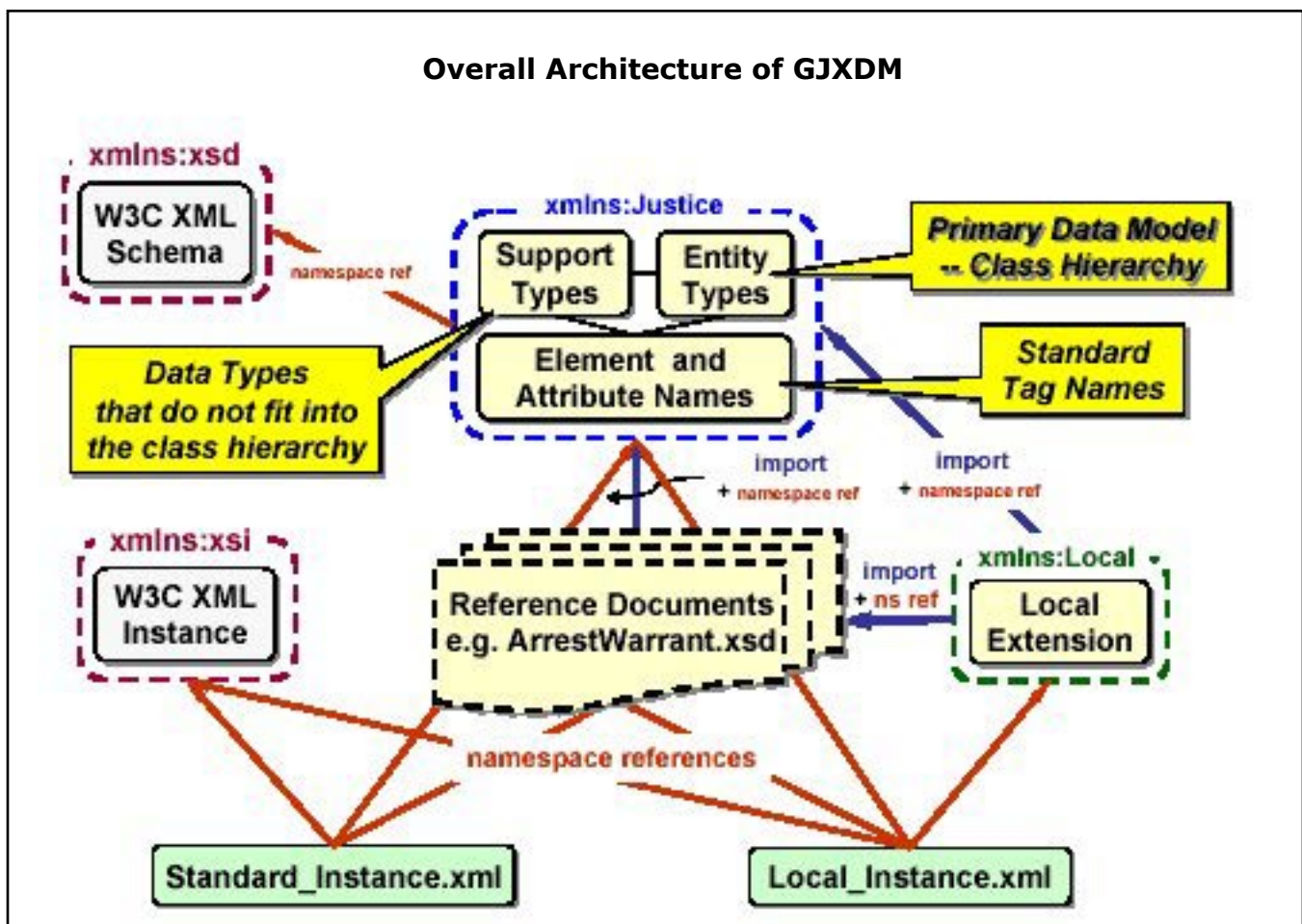


figure 3.1.1

Organization of the Global Justice XML Data Model and Data Dictionary

GJXDM data are voluminous and represent nearly 3,000 components (see figure 3.1.2). These were created by XSTF from an original collection of more than 16,000 elements, which were in turn derived from approximately 34 source specifications developed by a broad variety of justice entities around the nation.

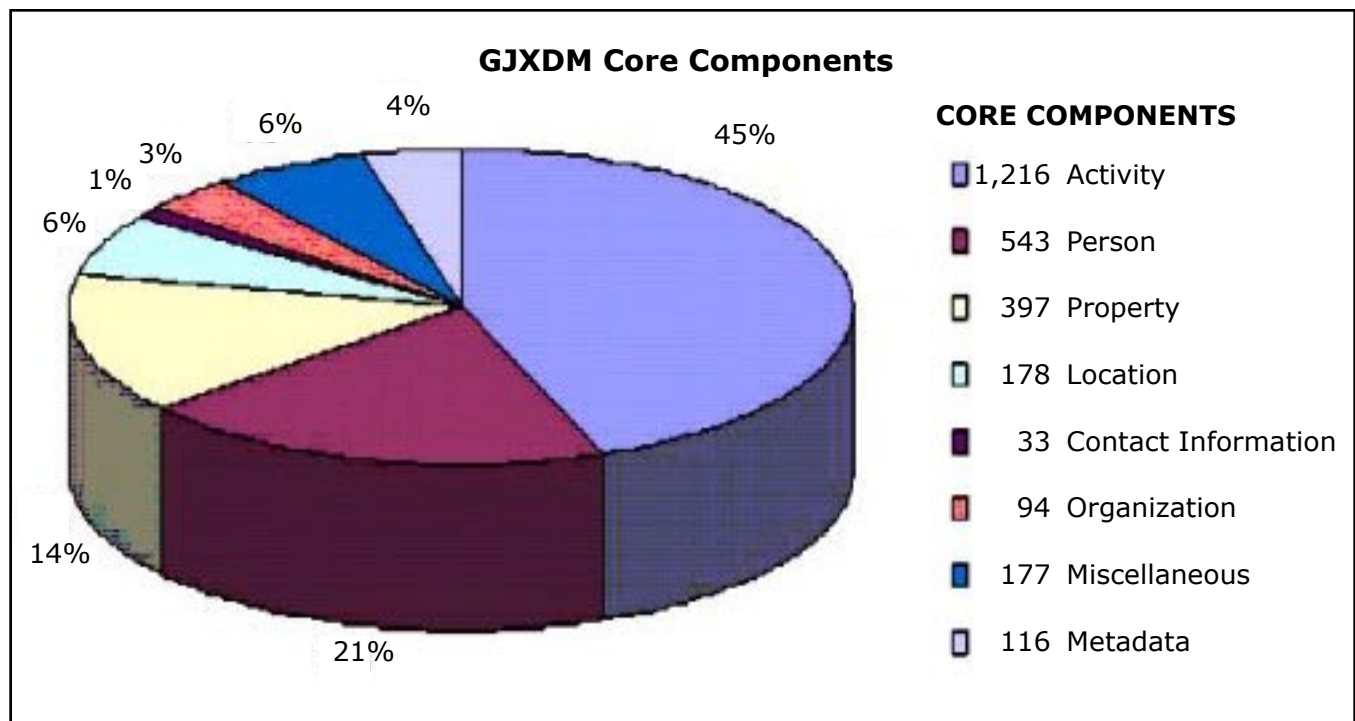


figure 3.1.2

To make them easier to find, the Global Justice XML Data Dictionary (GJXDD) spreadsheet groups data components as:

- **Primary Types**, which include:

- General Types.
- Activity.
- Person.
- Location.
- Contact Information.
- Property.
- Organization.
- Metadata.

- **Secondary Types.**
- **Attributes.**
- **Namespaces.**

[Access the current version of the GJXDD spreadsheet here!](#)

PART 2 Global Justice XML Data Dictionary Core Components

General Types

General Types are common root types that have global applicability. The following are the general types provided in GJXDD:

- **SuperType** is the root object in the type inheritance hierarchy. It incorporates attributes such as, @sensitivityText, @reportingOrganizationText, @reportedDate, @effectiveDate, and so forth. SuperType provides a base object with properties that can be associated to any element in the entire GJXDM.
- **Measure, including:**
 - AgeMeasure.
 - LengthMeasure.
 - WeightMeasure.
 - TimeMeasure.
- **Rate.**
- **Quantity.**
- **Range.**
- **ID.**
- **EnginePowerDisplacement.**
- **BinaryObject.**
- **Numeric.**
- **Amount.**
- **Quantity.**
- **Text.**
- **Percentage.**
- **Status.**
- **PartialContent.**
- **Range.**



Note that the naming convention for attributes differs from that of elements and types. **Elements** and **Types** are described using **UpperCamelCase**. **Attributes** are described using **@lowerCamelCase**.

ActivityType

Activity Objects account for approximately 60 percent of the GJXDD, as illustrated in figure 3.2.1.

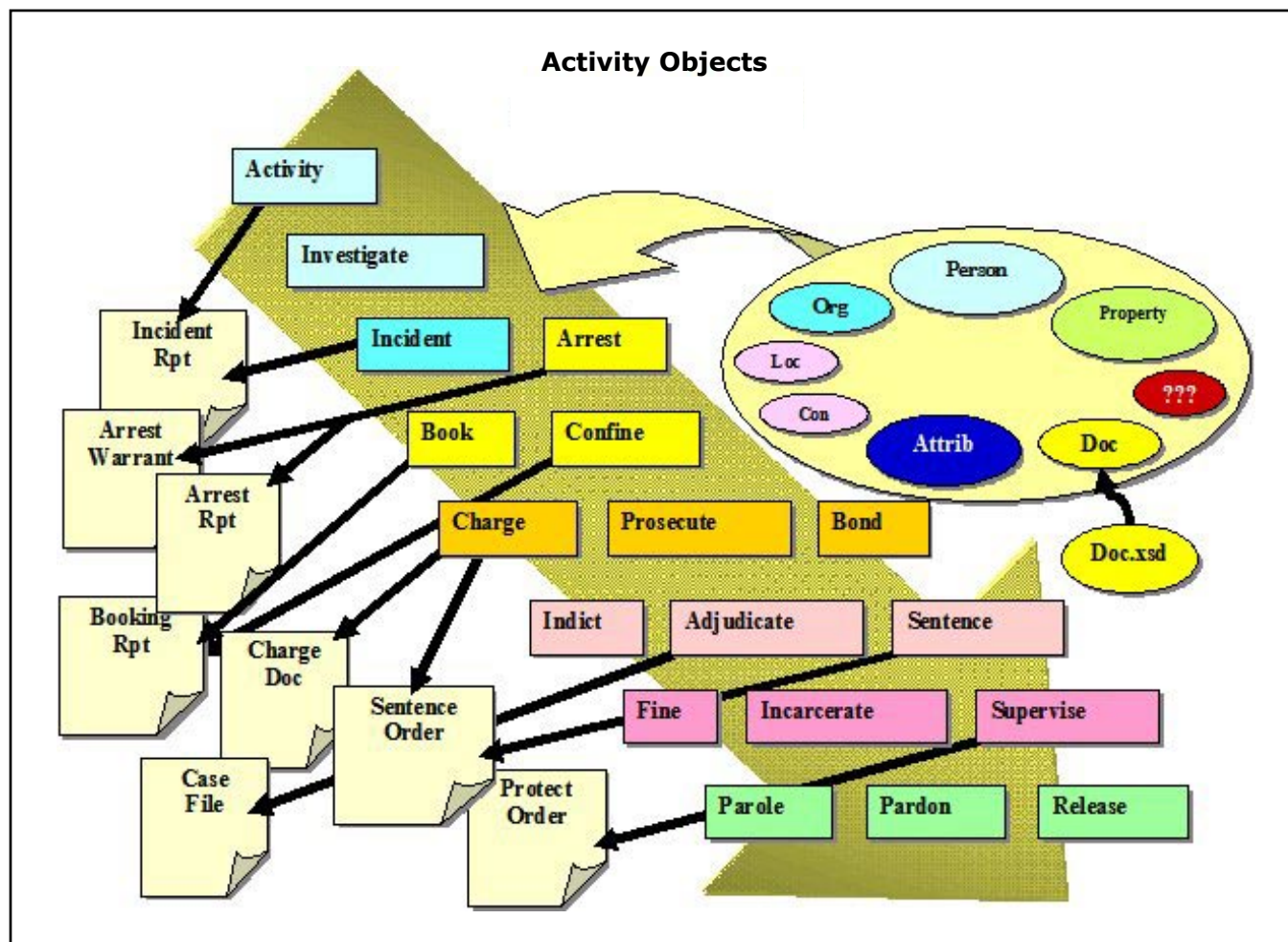


figure 3.2.1

The Activity Objects provide the foundation from which all “justice exchange documents” (currently referred to as “information exchange packages” or IEPs) are derived, such an Arrest Report or Sentencing Order. One or more activities can be associated with each exchange document. The Activity Objects are grouped into **Activity Types** and **Activity Support Types**.

ActivityType (which is a SuperType) contains types and elements common to all justice activities, for example:

- ActivityID
- ActivityTypeText
- ActivityDescriptionText
- ActivityReasonText
- ActivityDate
- ActivityTime
- ActivityEndDate
- ActivityEndTime
- ActivityCompletedIndicator
- ActivityStatus
- ActivityResultText
- ActivityResultDate
- ActivityResultTime
- ActivityCommentText
- ActivityUpdateIndicator

ActivityType Children, derived from ActivityType, include:

- Alert
- Assessment
- Arrest
- Bail
- Bond
- Booking
- Case
 - AppellateCase
- Citation
- Condition
- ConditionGroup

- Conviction
 - DriverConviction
- CourtActivity
- CourtEvent
- CourtOrder
 - ProtectionOrder
 - Warrant
- CustodyTransfer
- DisciplinaryAction
- DriverLicenseWithdrawal
- Force
- Incident
- DrivingIncident
- IncidentResponse
- PassagePoint
- Plea
- Progress
- PropertyDisposition
- PropertySeizure
- Referral
- Release
- ConditionalRelease
- Sanction
- Schedule
- ScheduleDay
- Sentence
- SentenceModification
- ServiceCall
- Submission
- Plea
- Progress
- PropertyDisposition
- PropertySeizure
- Referral
- Release
- ConditionalRelease
- Sanction
- Schedule
- ScheduleDay
- Sentence

- SentenceModification
- ServiceCall
- Submission
- Supervision
- Program
- Term
- VehicleBrand
- VehicleInspection
- VehicleEmissionInspection
- VehicleSafetyInspection
- Visitation

The primary activities listed extend directly from ActivityType. BJA's XSTF has made it a priority to provide clear and nonambiguous definitions for each type and element. Since the prerelease of GJXDD, practitioners and developers from across the nation have provided input, via the [GJXDM feedback site](#), which resulted in the clarification of many of the GJXDD's components, for example, CaseType.

The original definition of CaseType related to a case as defined by a court of law. Following a practitioner comment and review and discussion by XSTF, CaseType was re-defined as follows:

"...An aggregation of information, in any format, that describes a set of related activities and occurrences. The domain or discipline defines the limits of the information included in a specific kind of case. Case is not limited to any particular discipline or domain, and can relate [to] a wide range of areas: a situation requiring investigation or action (as by the police); the object of investigation or consideration; an instance of disease or injury; an instance that directs attention to a situation or exhibits it in action; a set of circumstances or a state of affairs; a situation; a question or problem; a matter; an action or a suit or just grounds for an action; the facts or evidence offered in support of a claim; a set of reasons or

supporting facts; a person being assisted, treated, or studied, as by a physician, lawyer, or social worker."

This clearer definition now allows a broader interpretation of CaseType to include other sorts of cases dealt with in the justice system. A CaseTypeText attribute will allow the user to define the specific context of the case, such as court, prosecutor, local law enforcement, public safety, health, and so forth.

ActivityType Instances include:

- Incident
 - Felony
 - Infraction
 - Misdemeanor
 - Offense
 - Violation
- Plea
 - OfferedPlea
 - FinalPlea
- Program
 - DiversionProgram
 - RehabilitationProgram
 - RiskReductionProgram
- Sentence
 - PreviousSentence
 - AmendedSentence
- Supervision
 - Corrections
 - Detention
 - Incarceration
 - Parole
 - Probation
- Warrant
 - ArrestWarrant
 - SearchWarrant
- CourtOrder
 - Summons

Activity Support Types include:

- Actor
- Affiliate
- CaseDetails
- CaseDisposition
- CaseOfficial
- CaseParticipants
- Caveat
- Charge
- ChargeClassification
- ChargeDisposition
- ChargeReporting
- CourtAppearance
- CriminalTraitDetails
- Disposition
- DriverAuthorization
- DriverHistorySummary
- DriverLicense
- DriverLicensePermit
- DrivingRestriction
- Event
- Evidence
- Exhibit
- Fee
- Image
- IncidentCategory
- Insurance
- JudgePanel
- Jurisdiction
- Lien
- NonstandardCodeDetails
- SentencingGuideline
- SeverityLevel
- Statute
- Target
- Verdict

This list of support types is not complete. There are many others that are key components in making up the definition of an activity. XSTF is currently working to develop GJXDD Context Definitions for all components.

Of special note are Jurisdiction and NonStandardCodeDetails. Both were added in the 3.0.0.1 prerelease. Jurisdiction specifies the geopolitical area in which a person, organization, or object has a specific range of authority. NonStandardCodeDetails was added to accommodate codes that are not otherwise defined. This should only be seen in rare cases.

PersonType

In GJXDM, PersonType is derived from the SuperType. One of the greater challenges XSTF faced was to carefully derive subclasses of Person, which would provide the context necessary to provide information on data exchanged within the justice system.

Following much analysis and review of the prerelease by practitioners and developers, XSTF has determined that only the following eight subclasses of Person are derived from PersonType (see figure 3.2.2).

- **Juror.**
- **Missing Person.**
- **Subject.**
- **RegisteredOffender** (use RegisteredOffenderTypeText or Code to identify specific sort of offender, i.e., SexOffender).
- **Victim.**
- **Witness.**
- **EnforcementOfficial** (refers to any justice or public safety employee, generally related to either law enforcement, corrections, or military).
- **JudicialOfficial** (includes judges, magistrates, defense attorneys, prosecutors).

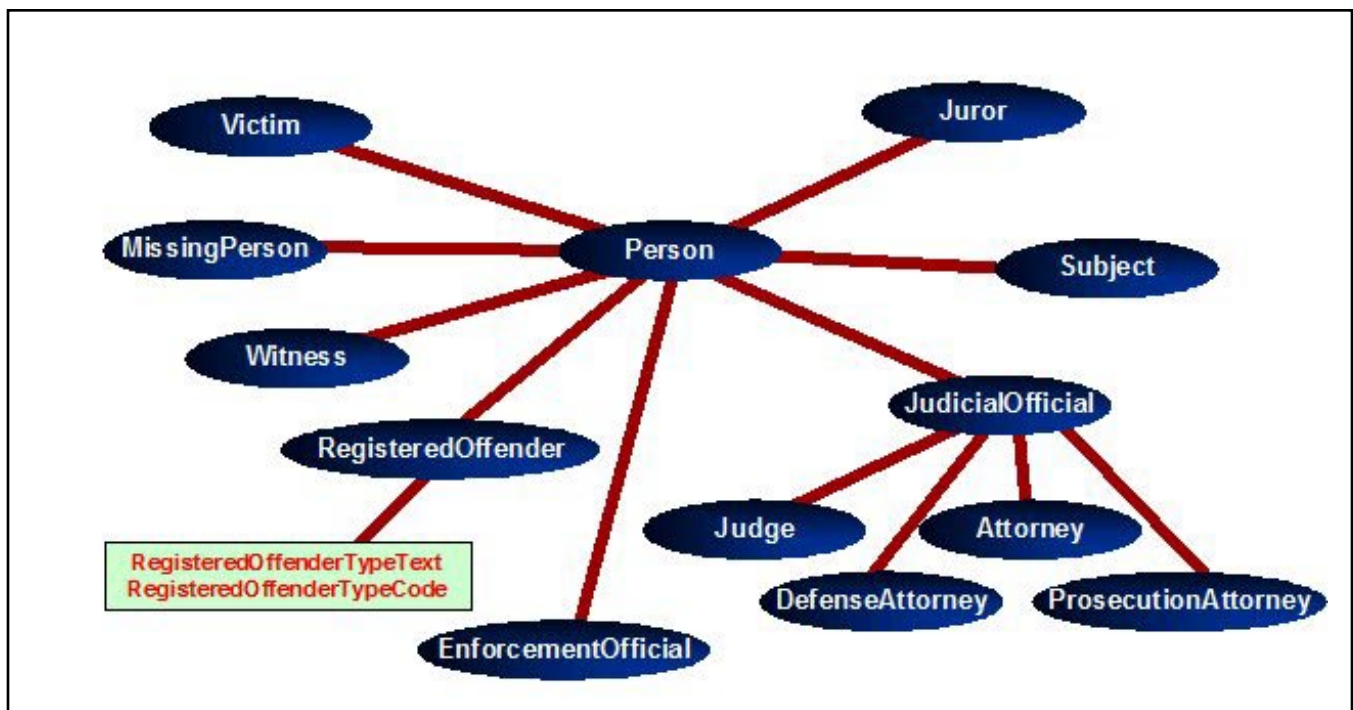


figure 3.2.2

LocationType

BJA's XSTF, in its review of all of the source requirements used by justice agencies within local, state, tribal, and federal jurisdictions, determined that location can refer to either physical location or locations for postal delivery. Part or properties of LocationType include:

- Address
- AddressGrid
- Area
- CrossStreet
- GeographicCoordinate
- Highway
- Locale
- MapLocation
- RelativeLocation
- UTMCoordinate (Universal Transverse Mercator)
 - MGRSCoordinate (Military Grid Reference System)

OrganizationType

In the justice system, an organization can be a party in a legal matter or own property. The following subclasses have been derived from the OrganizationType:

- Court
- CriminalOrganization
- EnforcementUnit
- Facility
- Lessee

PropertyType

In the justice domain, it is important to distinguish property as one of three basic categories: (1) Real Property, (2) Personal Property, and (3) Intellectual Property. XSTF anticipates that there will be additional content added to the Property object in the future.

It is important to note that those elements included in the dictionary under Property that specifically relate to what has happened to it—such as in the evidentiary chain of custody,

e.g., PropertyDisposition—are extended from ActivityType.

- VehicleBasics
 - Vehicle
 - CommercialVehicle
 - Boat
 - AirCraft
- Firearm
- Drug
- Jewelry
- Structure
 - ResidentialStructure
 - CommercialStructure
- Security
- IntellectualProperty
- RealEstate

Final points to keep in mind about the GJXDM components:

- The content will continually change with new requirements.
- The structure will remain stable over time.

ContactInformationType

Finally, XSTF determined from its review of justice agency requirements that Contact Information would best be abstracted as a core object.

- ContactInformation
 - HomeContactInformation
 - WorkContactInformation
 - PrimaryContactInformation
 - EmergencyContactInformation
- TelephoneNumber

XSTF anticipates that as justice agencies begin to share information using the model, and as the universe of exchange requirements expands, GJXDD content will change as new elements are added and current ones revised. XSTF is committed to maintaining the structural integrity of the model that was developed. New releases of GJXDM will occur periodically and always be accessible at <http://it.ojp.gov/>.

PART 3 Metadata in the Global Justice XML Data Model

Metadata

Module 002 introduced the topic of data, or data which more specifically describes the information that is being exchanged. This part will explore this topic in greater detail and provide some specific examples of metadata describing justice information.

When justice agencies exchange information, there may be a requirement to include some *modifiers* regarding the data being exchanged. For example:

- **Probability.** There is a 30 percent probability that the subject, John Doe, is 35 years old.
- **Distribution.** The subject, John Doe, is 35 years old, but this is confidential investigatory information and may not be disseminated to anyone outside the source agency.

- **Source.** Officer David Smith describes the subject, John Doe, as 35 years old.
- **Estimation.** The subject, John Doe, is 35 years old, plus or minus 5 years.

XML provides the ability to describe or modify data with a couple of different mechanisms, either with *inherent* data or with *metadata*. Inherent data is tightly coupled to the object it modifies. For example, a record on a subject may have an “entered-by” property, which is always carried with the subject object.

Metadata is not tightly coupled to the object. Metadata provides comments about the data and may change to fit an evolving understanding of the data. An example of metadata is the probability that a value is accurate or that a relationship between two persons exists. Figure 3.3.1 illustrates the relationship between metadata and the GJXDM objects.

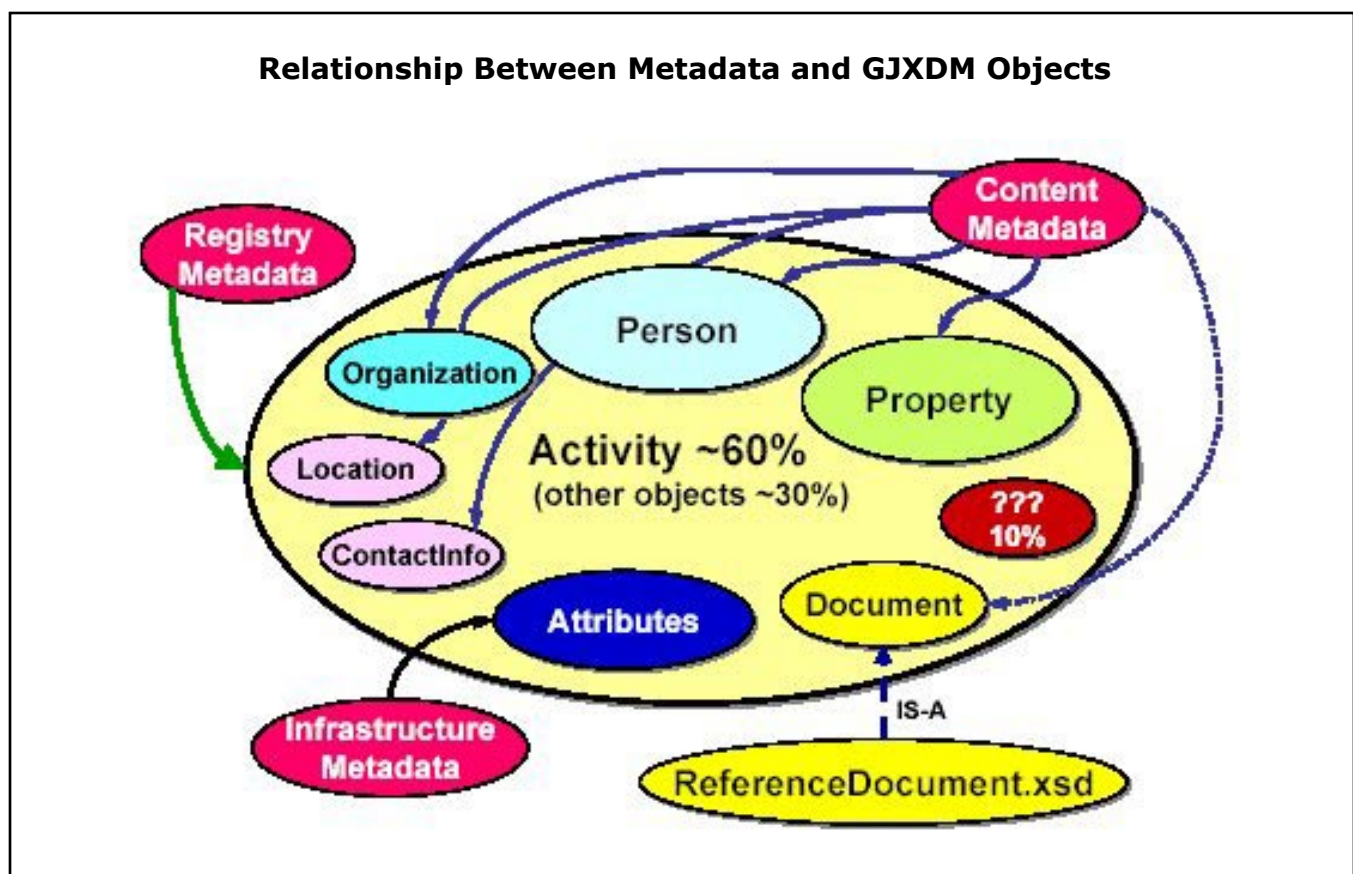


figure 3.3.1

Metadata Examples

Remember that metadata does not represent the *value* of the object; rather, it represents information *about* the value of the object. The following lists 10 metadata examples.

1 – Probability

Probability = the chance that a given value is the *intended* value.

```
<Person>
  <PersonFullName>Bob Jones</PersonFullName>
  <PersonAgeMeasure @confidenceNumeric="30%">45</PersonAgeMeasure>
</Person>
```

2 – Reliability

Reliability = the chance that the given value is the correct value.

```
<Person>
  <PersonFullName>Bob Jones</PersonFullName>
  <PersonAgeMeasure @reliabilityNumeric="30%">45</PersonAgeMeasure>
</Person>
```

3 – Distribution and Sensitivity

Distribution = the dissemination information; in other words, to whom may this data be communicated?

4 – Data Source or Origination

In justice systems integration, agencies often want to be able to track the source of, or party responsible for, a specific instance of data. This information can be carried as part of the data instance. It also can be a persistent part of the value of the object.

5 – Language

An agency may wish to specify the language of a field, specifically a free-text or description field, which could help to enable automatic translation. ISO specifies the language in a document titled ["Technical contents of ISO 639:1988"](#).

6 – Encoding

Encoding refers to the format of the encoded data, such as ASCII or base64. This would be better placed as invariant information inherent to a property.

7 – Precision

Precision is the degree to which the value is known to be valid.

8 – Ranges and Point Estimates

In the course of exchanging justice information, an agency may need to specify approximations and/or estimations of values:

Range represents a low value, paired with a high value.

Point estimate = a target value, paired with a possible delta value (i.e., 45 plus or minus 5).

9 – Partial Values, Patterns

An agency may wish to specify a known part of a value, with a description of how the value may vary. This may be done with regular expressions, for example:

```
<Vehicle>
  <VehicleMake>Toyota</VehicleMake>
  <VehiclePlatePattern>AB.+4.[B8]</VehiclePlatePattern>
</Vehicle>
```

This would specify a license plate with:

- an A, then
- a B, then
- some additional information..., then
- a 4, then
- an unknown letter, and then
- either a B or an 8

Although patterns are not currently part of GJXDM, they can be added through extension.

The following is an example instance:

```
<?xml version="1.0" encoding="UTF-8"?>
<ex:doc
  xmlns:ex="p-ex"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="p-ex pattern-example.xsd">
  <VehiclePlatePattern>ABCL4AB</VehiclePlatePattern>
  <VehiclePlatePattern>ABK4A8</VehiclePlatePattern>
  <VehiclePlatePattern>ABIDriveABigCar41B</VehiclePlatePattern>
</ex:doc>
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="p-ex"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="doc">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="plate" minOccurs="0" maxOccurs="unbounded">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:pattern value="AB.+4.[B8]"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

10 – Dates

Date properties that can be associated with an instance, include, but are not limited to:

- @reportedDate
- @expirationDate
- @lastUpdatedDate
- @lastVerifiedDate
- @measureDate
- ObligationStartDate
- ObligationEndDate
- ObligationDueDate
- ObligationPaidDate
- StatusDate
- BinaryCaptureDate
- IDEffectiveDate
- IDExpirationDate
- AircraftYearDate
- IntellectualPropertyCreationDate
- IntellectualPropertyRegistrationDate
- IntellectualPropertyTerminationDate
- PropertyYearDate
- RegistrationEffectiveDate
- RegistrationExpirationDate
- TitleIssueDate
- PropertyValueDate
- SecurityCollectionStartDate
- SecurityCollectionEndDate
- VehicleModelYearDate
- VehicleInvoiceDate
- VehicleShipDate
- VehicleFirstSoldYearDate
- VehicleBrandDate
- DecalYearDate
- DecalMonthDate
- VehicleTitlePurchaseDate
- BiometricCaptureDate
- BiometricTestDate
- EmploymentStartDate
- EmploymentEndDate

Please see the [GJXDD Spreadsheet](#) for a complete list of elements and attributes that can represent a date.

PART 4 Code Tables and Enumerations in the Global Justice XML Data Model

Code Tables

GJXDM supports the use of both internal and external code tables. External means *external to the justice namespace*, not external to the local work environment.

BJA's XSTF has put forth a great deal of effort collecting and validating codes from recognized authoritative sources relevant to justice entities, such as the FBI and the American Association of Motor Vehicle Administrators.

The code tables listed in GJXDD are *enumerated* types that equate to codes. These are all defined in the GJXDD schema namespaces and downloaded as part of the full GJXDD schema download. These also are referred to as *external enumerations*.

Most codes not only have a code value, but also a text value (this is called a "literal") or a definition.

The following illustration provides an instance of "Scars, Marks and Tattoos" from the National Crime Information Center (NCIC) 2000 Personal Descriptors:

```
<xsd:element name="SMTypeElement" type="SMType" />
  <xsd:simpleType name="SMType">
    <xsd:restriction base="xsd:token">
      <xsd:enumeration value="ART ARM">
        <xsd:annotation>
          <xsd:documentation>Arm, nonspecific, artificial</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="ART BRST">
        <xsd:annotation>
          <xsd:documentation>Breast, nonspecific, artificial</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
      <xsd:enumeration value="ART BRSTS">
        <xsd:annotation>
          <xsd:documentation>Breast implant, left and right</xsd:documentation>
        </xsd:annotation>
      </xsd:enumeration>
    </xsd:restriction>
  </xsd:simpleType>
```

Implementing code tables within GJXDM has required special attention:

- Code tables have been used for decades within the justice community, especially law enforcement, and many justice systems share information using codes, especially vertical exchange between state/local and federal systems.
- Some code tables are extremely large, such as the NCIC maintained by the FBI.
- Some code tables are dynamic and updated frequently, such as tables representing vehicle makes and models.
- Most code tables fall under an administrative domain outside of GJXDM governance, and must be maintained/updated by various authoritative sources.
- Local jurisdictions will more than likely need to extend or supplement locally.
- Code tables in most cases did not exist in a form easily usable (electronically stored) and had to have proxies created for GJXDM.

GJXDM uses separate namespaces for code tables because:

- This facilitates code table ownership and allows updates to be used when ready.
- This allows multiple tables for same conceptual property, such as state abbreviations.
- This makes it easier to extend and add codes.
- This allows the GJXDM namespace to be broken into smaller, more easily consumable parts.
- Separate namespaces allow validation of all codes against the respective source table.
- GJXDM provides an option for a literal representation (TextType).

Below is an example of LocationStateName represented as a code as well as a literal in schema:

```
...
<xsd:element name="LocationStateName" type="TextType" />
<xsd:element name="LocationStateCode.USPostalService"
  type="j-usps:USStateCodeType" />
...
```

Here is an instance of LocationStateName represented with a United States Postal Service (USPS) Code and literal:

```
...
<j:LocationStateName> Alabama </j:LocationStateName>
<j:LocationStateCode.USPostalService> AL </j:LocationStateCode.USPostalService>
...
```

External Code Tables Imports

The following provides some of the external code tables imported by GJXDM. For a complete list, see the [GJXDD spreadsheet](#).

ansi_d20	Motor vehicle admin codes, ANSI D20, Data Dictionary for Traffic Record Systems
can	Province codes for Canada
cap	Alert codes from the Common Alerting Protocol version 0.7 Alert Message Dictionary
census	Employment codes from U.S. Census Bureau
U.S. Department of Defense (DoD)	
dod_exec-12958	Security classification codes from Presidential Exec Order 12958
dod_jcs-pub2.0	Intelligence discipline codes from DoD
dod	Security-level codes from DoD
Federal Information Processing Standards (FIPS)	
fips_10-4	Countries, dependencies, areas of special sovereignty
fips_5-2	Codes for identification of the states, DC and outlying areas of the U.S.
fips_6-4	Counties and equivalent entities of the U.S., and its possessions
International Organization for Standardization (ISO)	
iso_3166	Codes for names of countries and their subdivisions
iso_4217	Codes for currencies and funds
iso_639-2b	Codes for names of languages
iso_639-2t	Codes for names of languages
mn_offense	Statute and offense codes from Minnesota.
ncic	Codes from National Crime Info Center (NCIC) 2000 standard
nibrs	Miscellaneous crime-reporting codes from NIBRS
nonauth	Non-authoritative codes for the direction of a person's pose in an image
ucr	Crime reporting codes from Uniform Crime Reporting
unece	Miscellaneous unit of measure codes, U.N. Economic Commission for Europe Rec 20
usps__	U.S. state and possession abbreviations from the U.S. Postal Service
ut_offender	Plea codes and military discharge codes from Utah Offender Tracking Database

Note that codes for identification of the states are contained in both the USPS code tables and the Federal Information Processing Standards (FIPS) 5.2 (highlighted for emphasis).

Figures 3.4.1 and 3.4.2 on page 66 illustrate examples of each of these code tables, respectively.

Example of FIPS 5-2 Code Table

```

</xsd:enumeration>
-<xsd:enumeration value="WV">
  -<xsd:annotation>
    <xsd:documentation>West Virginia</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
-<xsd:enumeration value="WY">
  - <xsd:annotation>
    <xsd:documentation>Wyoming</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>
- <xsd:simpleTypeName="USStateNumericCodeType">
  - <xsd:annotation>
    <xsd:documentation>A code list that enumerates states.</xsd:documentation>
  </xsd:annotation>
- <xsd:restriction base="xsd:token">
  - <xsd:enumeration value="01">
    - <xsd:annotation>
      <xsd:documentation>Alabama</xsd:documentation>
    </xsd:annotation>
  </xsd:enumeration>
  - <xsd:enumeration value="02">
    - <xsd:annotation>
      <xsd:documentation>Alaska</xsd:documentation>
    </xsd:annotation>

```

figure 3.4.1

Example of USPS Code Table

```

-<xsd:enumeration value="VT">
  -<xsd:annotation>
    <xsd:documentation>VERMONT</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
-<xsd:enumeration value="WA">
  - <xsd:annotation>
    <xsd:documentation>WASHINGTON</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
-<xsd:enumeration value="WI">
  -<xsd:annotation>
    <xsd:documentation>WISCONSIN</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>

```

figure 3.4.2

PART 5 XML Schema Elements Versus Attributes

XML Schema Rules for Rendering Data Dictionary Properties as Elements Versus Attributes (Versus Mixed Content)

XML Schema provides two constructs for tagging data in XML instances—elements and attributes. (It also allows mixed content, which GJXDM does not use at all.) Which construct to employ is not always clear. While data can often be stored as attributes or elements, there are a number of constraints on attributes that generally impact the user's decision:

- XML does not support specification of complex structure in attribute values; attribute values can only have simple content (e.g., a string or Boolean value). Elements may have complex structure with subelements and/or attributes.
- Attributes are not easily extended (for local changes), nor can they participate in type substitution. However, additional subelements and attributes can be added to an element through type subclassing and type substitution.
- Mixed content (mixing simple values and elements together) is hard to work with and harder to specify. This means that properties of a simple data type must be included as attributes (to avoid having to use mixed content).

Federal XML Developer's Guide Rules for Using Elements Versus Attributes

The foregoing constraints have inspired some guidelines for using elements versus attributes. The following are extracted or paraphrased from the draft [Federal XML Developer's Guide](#) (April 2002):

- Attributes should only be used to convey metadata that will not be parsed. In other words, attribute values should be single words or numeric representations, not lists of words or numbers that may require further parsing.
- Attributes, if used, should provide extra metadata required to better understand

the business value of an element.

- Attribute values should be short, preferably numbers or conforming to the XML Name Token convention.
- Attributes with long string values should not be created.
- Attributes should only be used to describe information units that cannot or will not be further extended or subdivided.
- Use attributes to provide metadata that describe the entire contents of an element. If the element has children, any attributes should be generally applicable to all the children.
- One of the key schema design decisions is whether to represent an information element as an XML element or attribute.
 - One issue to keep in mind is that once an information item has been made an attribute, it cannot be extended further.
 - Another issue is that there cannot be multiple uses of it within the same element (i.e., 0 or 1, but no more).
 - Finally, if enumerated, an attribute cannot be extended to add values to its enumeration list, nor can it be used (added or extended) in type substitution.

For these reasons and to promote uniformity, federal guidelines discourage the use of attributes. But if they must be used, keep them simple.

Unfortunately, some of these guidelines can be very subjective. For example, what constitutes metadata and what constitutes data? Often this depends on the user perspective and the application. For a model with very large applicability and a requirement for maximum flexibility, it is probably safer to use elements (that can be more easily extended). Therefore, why use attributes at all? There are times when users may be forced to use attributes to maintain simple content. Moreover, even when simple content is not required, the use of an attribute may prevent unnecessary complexities.

Global Justice XML Data Model Rules for Using Elements Versus Attributes

The Global Justice XML Data Model employs the following rules for depicting elements versus attributes:

- (1) Employ elements whenever possible. Attributes are used as the exception, and only with reasonable justification based on XML limitations or significant avoidance of complexity.

Justification: Elements are much more flexible than attributes. Attributes cannot be complex and cannot occur multiple times. Federal guidelines and best practices suggest the avoidance of attribute use.

- (2) SuperType properties are always attributes.

Justification: The SuperType contains properties that are applicable to all components of GJXDM. Therefore, all fields will include the properties of SuperType. All GJXDM components are derived from SuperType, so that all components inherit all the SuperType properties. At the same time, SuperType has neither complex nor simple content. In fact, it has no content; it's empty. Thus, objects with simple content can still be derived from SuperType, because it cannot contain any subordinate elements. And finally, you can consider SuperType properties generic enough to be metadata (such as @probabilityNumeric, @distributionText, @reportedDate, @expirationDate, etc.) on all GJXDM components.

- (3) DocumentType properties always are elements.

Justification: DocumentType is the root of all reference document types (now referred to as information exchange package documentation (IEPD – see module 003, part 6). DocumentType is derived from SuperType. In GJXDM, IEPDs

derived from DocumentType are the primary basis for information exchange transactions. As such, DocumentType has a set of metadata properties that are common to all documents derived from it. These properties have been provided as elements for the following reasons:

- There is no need for DocumentType to be empty (as there is for SuperType).
- Metadata defined for DocumentType is fairly complex and cannot easily be rendered as attributes.
- The common properties of reference IEPs (documents, transactions) must be able to evolve and be extended by local jurisdictions.
- What may be metadata in a library or relational table sense may be relevant document data to someone else.

- (4) Use attributes for metadata that simply qualify the format or representation of a data value, but are not a required part of that data value, and when such use will avoid complexity.

Justification: Sometimes metadata qualifiers are not an essential part of the data itself. The data can stand alone and still be understood. In other words, the qualifiers can be ignored without harm to understanding. The qualifier simply clarifies or focuses the meaning of the data or its representation. Furthermore, use of attributes for such qualifiers avoids unnecessary content complexity.

Example:

```
<PersonName personNameTypeCode='aka'/>
```

- (5) Properties of simple numeric types are attributes.

Justification: Type Measure, Numeric, Quantity, Rate, and several other types that require qualifiers such as unit of measure or tolerance will be handled with attributes to maintain simple content. These qualifiers are simple and have stable values. The data value and qualifiers cannot be separated without almost total loss of meaning. Furthermore, use of attributes for such qualifiers avoids unnecessary content complexity.

Example:

```
<VehicleSpeedMeasure speedUnit='mph' tolerance='5'>37</VehicleSpeedMeasure>
```

- (6) Mixed content is not supported.

Justification: Mixed content models are confusing and extremely difficult to implement and parse.

Example:

```
<VehicleSpeedMeasure>37<SpeedUnit>mph</SpeedUnit></VehicleSpeedMeasure>
```

PART 6 Global Justice XML Data Model DocumentType

Generally, electronic justice data exchange is accomplished via documents or queries, responses, and other messages. In XML schema terminology, all of these information exchanges are considered to be *XML Documents*.

In GJXDM, when an XML document is referred to, the XSTF often is referring to a standard business exchange container, such as a Rap Sheet, Sentencing Order, or an Incident Report. These sorts of documents are generally considered to be *persistent* in nature. This means that they can be archived, maintained, or reused intact by either the sender or receiver.

Often, the exchange data is represented in document format and can be either displayed as such on a computer screen or printed.

However, XML documents also can be transactional. For example, queries, responses, or messages can be structured as XML documents. A document can be defined as ". . . something which brings together strongly related objects for a well defined business purpose or context. . . " or ". . . that bundle of data that is passed from one agency to another as part of an exchange . . ."⁹ Therefore, the term "XML document" can refer to a message or other form of information, as well as what is traditionally recognized as a document, as illustrated in figure 3.6.1.

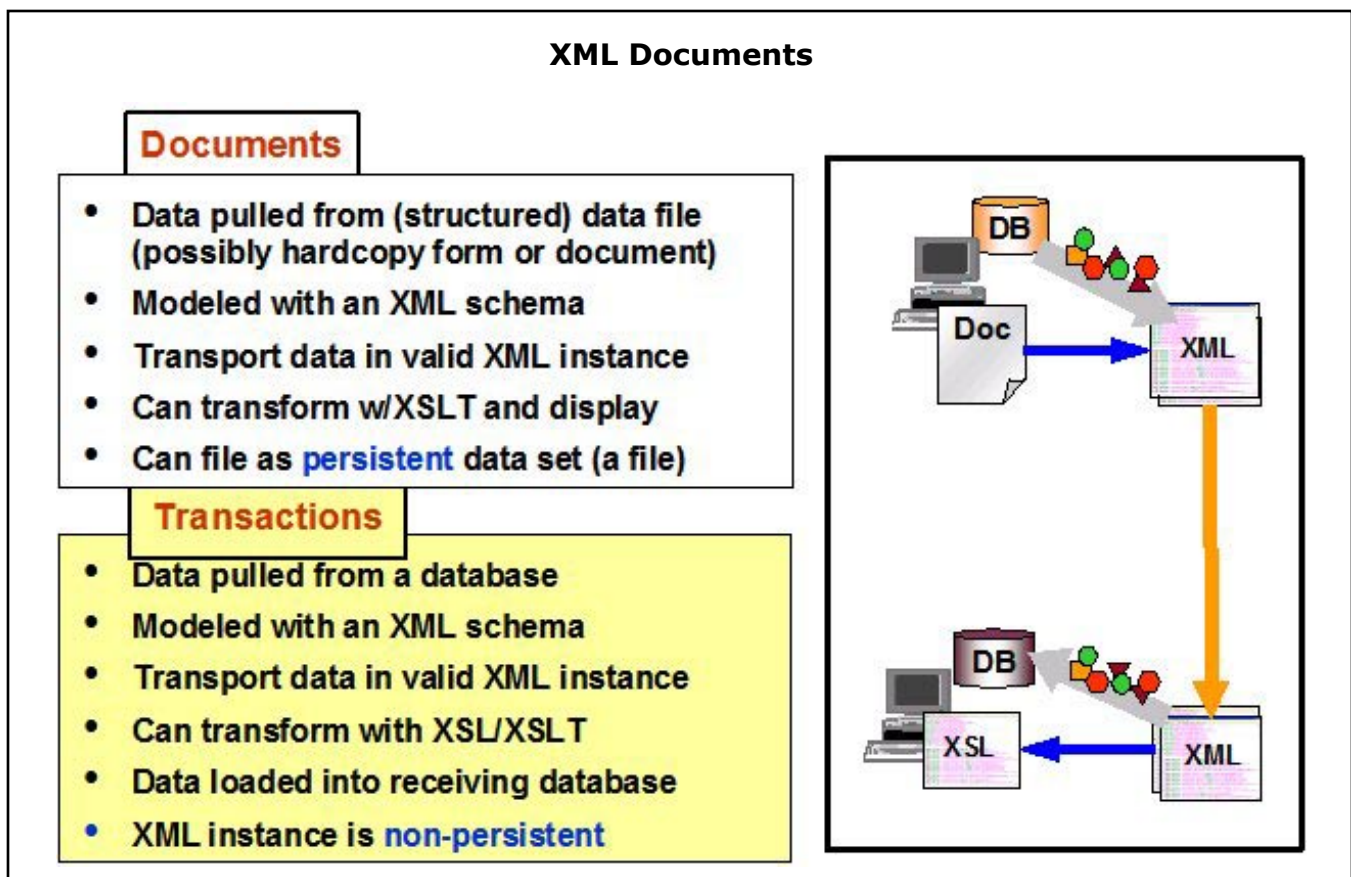


figure 3.6.1

⁹ Gerry Coleman, Wisconsin Crime Information Bureau.

DocumentType Metadata

GJXDM contains a *DocumentType* component from which standard XML document schemas can be derived. The GJXDM *DocumentType* includes commonly used properties, such as control and records management metadata, security and classification metadata, and general document descriptive metadata.

Figure 3.6.2 shows the *DocumentType* metadata properties, including *DocumentControlMetadata*, *DocumentSecurityMetadata*, and *DocumentDescriptiveMetadata*, which are inherited by any exchange document derived from *DocumentType*.

Dublin Core is an open forum engaged in the development of interoperable online metadata standards.

Sources for many of these properties include:

- [Dublin Core Metadata Initiative](#).
- [National Archives and Records Administration Records Management](#).
- [Department of Defense 5015.2-STD Design Criteria Standard for Electronic Records Management](#).

DocumentType Metadata Properties		
DocumentControlMetadata DocumentDispositionInstructionText DocumentCategoryName DocumentCategoryID DocumentPrivacyActIndicator DocumentManagementCyclePeriodText DocumentCopyrightIndicator DocumentCountryCode DocumentSubjectCodeText DocumentVitalIndicator DocumentPermanentRecordIndicator DocumentDispositionAuthorityName DocumentCategoryDescriptionText DocumentSecurityMetadata SecurityLevelCode SecurityForeignReleasabilityText SecurityFGIText SecurityFGICode SecurityDisseminationText SecurityLevelText SecurityNonICMarkingsText SecurityDeclassification SecurityControlText SecurityClassificationUpgrade SecurityClassificationInitial SecurityClassificationDowngrade SecurityClassificationCurrent	DocumentDescriptiveMetadata DocumentFormatText DocumentFormatTypeText DocumentGroupID DocumentID DocumentIntelligenceTypeCode DocumentFileName DocumentKeywordText DocumentFiledTime DocumentFiledDate DocumentFileControlID DocumentExpirationDate DocumentEffectiveDate DocumentDisputedReasonText DocumentDisputedReasonCodeText DocumentDisputedIndicator DocumentFileExtensionText DocumentFileTypeText DocumentRelationText DocumentRightsText DocumentSequenceID DocumentSourceText DocumentContentTableText DocumentStatus DocumentSubjectText DocumentSubmitter DocumentSummaryText DocumentPostDate DocumentTitleText	DocumentLanguageCode DocumentOtherAddresseeName DocumentMediumText DocumentLocation DocumentMediaTypeText DocumentLocationURI DocumentReceivedDate DocumentRelatedResourceText DocumentPublicationDate DocumentLastModifiedDate DocumentSupplementalMarkingText DocumentAuthor DocumentAbstractText DocumentAlternativeTitleText DocumentApprovedIndicator DocumentCommentText DocumentContributor DocumentCutDate DocumentApplicationName DocumentDescriptionText DocumentCoverageText DocumentCreator DocumentCreationDate DocumentSubject DocumentEnteredBy DocumentOtherMetadataField DocumentBinary DocumentSource

figure 3.6.2

BJA's XSTF is well aware of the fact that documents, whether persistent or transactional, represent the true currency of information exchange in justice and public safety. Figure 3.6.3 illustrates an architectural model for Justice Exchange Documents to be developed by local jurisdictions, or provided by various justice domains, such as law enforcement, as models for the justice community.

Within its own target namespace, each justice reference document schema will import the GJXDM namespace (or a subset of it) and

extend DocumentType for its root element.

For example, an Incident Report schema would import the GJXDM namespace and create (1) an IncidentReportType that extends gjxdm: DocumentType, and (2) a complex root element IncidentReport of type IncidentReportType. Consequently, IncidentReportType inherits all of the standard metadata properties of gjxdm: DocumentType. However, each document designates its own target namespace to enable local document customization and prevent name conflicts.

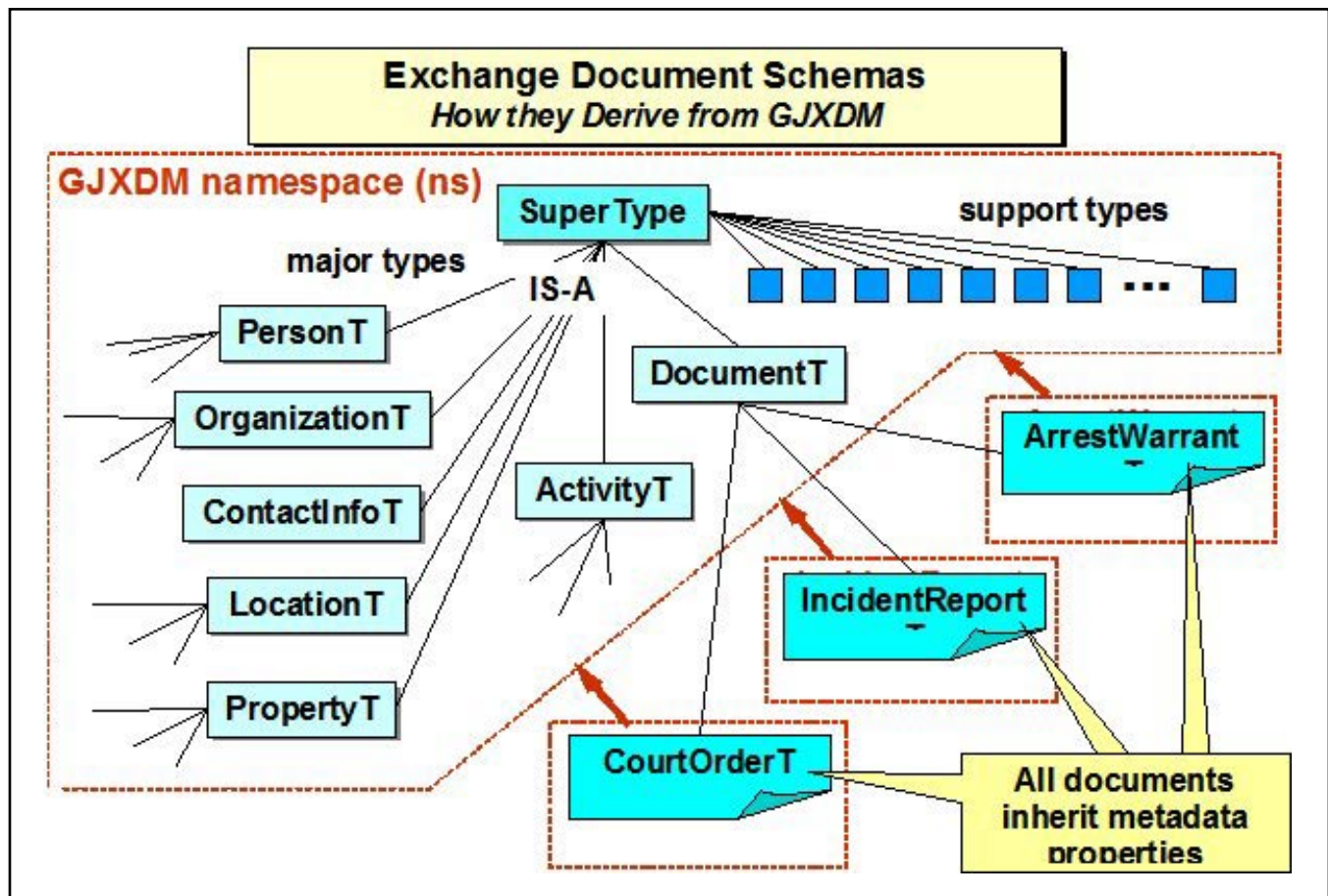


figure 3.6.3

Since the GJXDM Developer Workshop provided at Georgia Tech Research Institute in May 2004, BJA's XSTF has recognized the need to provide GJXDM *Reference Exchange Documents*¹⁰ to serve as models for the justice and public safety community.

GJXDM Reference Exchange Documents will define the standard content of individual information exchanges, such as a Sentencing Order or Arrest Warrant. Practitioners can then leverage and customize these reference documents to fit their circumstances, eliminating months of effort and considerable cost.

Since June 2004, several coordinated efforts, sponsored by BJA, with support from SEARCH, OASIS members, IJIS Institute members, OJP, and appropriate organizations that represent various justice communities of interest, have begun what is hoped will be a continuing collaborative effort to develop Reference Exchange Documents (now referred to as IEPDs) using the GJXDM. Several of these Reference Exchange Document efforts have recently been completed. These IEPDs are provided as examples in module 005.

¹⁰ To ensure consistency with the Federal Enterprise Architecture goals and objectives, XSTF voted recently to change the nomenclature of Exchange Document to Information Exchange Package Documentation, or IEPD. This topic is covered in more detail in module 004.

MODULE 004

Using the Global Justice XML Data Model To Describe Real-life Justice Data

PART 1 Exchange Document Development Process

This module describes a process that can be used to guide the development of a Global Justice XML Data Model (GJXDM) Information Exchange Package (IEP) and associated documentation. This process has evolved from a series of Reference IEP development projects in 2004 and 2005, including Sentence Order, Amber Alert, Incident Report, Arrest/Booking Report, Charging Document, and Field Interview Report Reference IEPs. It incorporates input from SEARCH staff and the XML Advisory Committee of the Integrated Justice Information Systems (IJIS) Institute and has been used to guide exchange development projects at the state and local level.

As with any process, it is important to understand the unique characteristics and needs of each project, and then to tailor process recommendations accordingly. The process described in this section is a guide or template and is intended to be customized as necessary. Even so, it provides a useful starting point in project planning and can help to set high-level expectations regarding milestones, resources, and timelines.

The goals of the process are as follows:

1. The process results in more consistent development of GJXDM-conformant schemas.
2. The process produces ancillary artifacts that address the information needs of a broad range of project stakeholders, including project sponsors, business experts, business and information technology managers, and technologists.
3. The process provides a mechanism for synthesizing the domain/business knowledge of subject-matter experts.

4. The process fosters artifact reuse across projects by improving artifact consistency.
5. The process leverages open industry standards that are familiar to most business analysts, architects, and other technology professionals.
6. The process works with standards-based tools that are readily available in the public domain or at low cost, allowing integration projects to avoid high licensing costs and vendor lock-in.
7. The process seeks to share with the justice community valuable lessons learned and best practices from Reference IEP development projects so that those lessons need not be relearned on future projects.

The remainder of this module will discuss the following five basic steps in the process:

1. Project inception: planning, coordination, and resourcing.
2. Domain modeling.
3. Domain-GJXDM mapping.
4. Schema building.
5. Packaging.

Figure 4.1.1 on page 75 illustrates the exchange document process through schema building and validation.

Exchange Document Process Through Schema Building and Validation

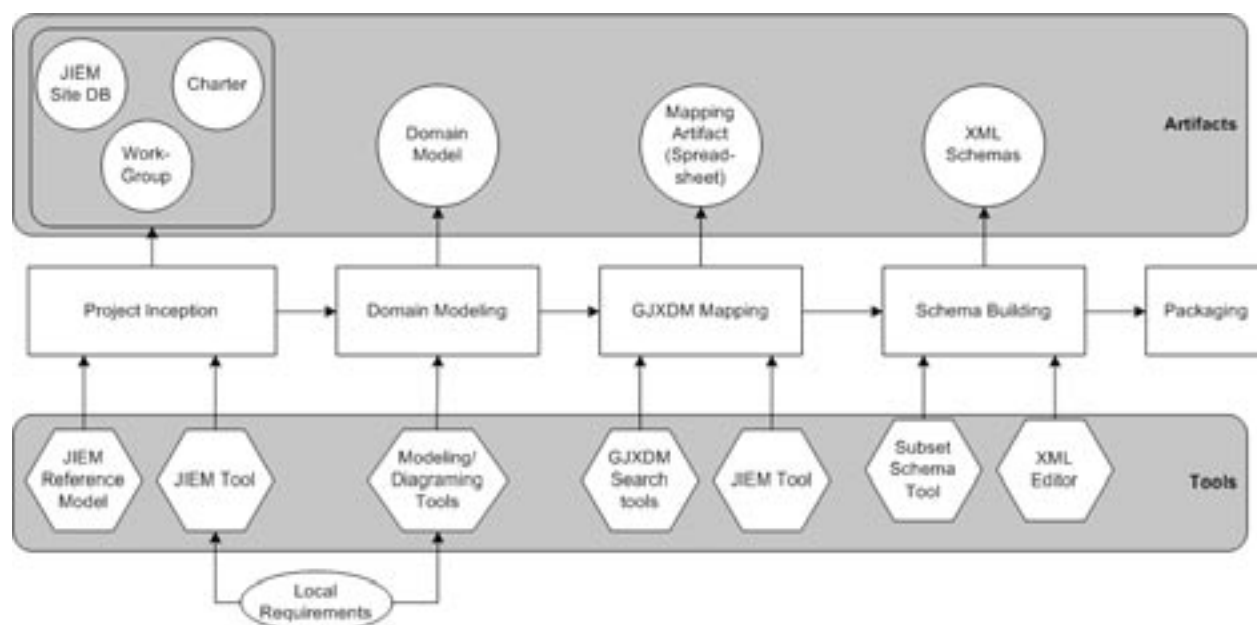


figure 4.1.1

PART 2 Exchange Document Project Inception

The first step in the exchange document development process is project inception. The purpose of this step is to plan the project, establish the process, and provide for human and technology resources.

The **first task** in the project inception phase is to establish a clear vision for the schema development project. The goal of a vision statement is to determine, at a high level, the scope of the project in terms of who the stakeholders are and what business-oriented results they should expect to achieve by the time the project has completed. In addition, if the project has important contextual attributes, those should be noted as well. For example, a project may be a follow-on to a previous project, or it may leave important business objectives out of scope, with the intent to address those objectives on future projects.

A useful starting point for a vision statement is the set of exchanges documented in the [JIEM Reference Model](#), since those exchanges have been identified as the most common across jurisdictions. Using the Reference Model's standard terminology can facilitate communication (since the terminology is well-defined in JIEM) and also improves the likelihood that off-the-shelf artifacts (like reference IEPs) will be available as a starting point for the rest of the steps in the process.

The **second task** in the project inception phase is to define exchange requirements that will be implemented (at least in part) by the schemas to be developed later in the process. This task is an ideal fit for the JIEM Modeling Tool, which allows a modeler to define exchanges between agencies, including constraints and information content. If the vision has established project scope in terms of exchanges in the JIEM Reference Model, the Modeling Tool allows the modeler to import these reference exchanges into the "local" model (called a "site database") as a starting point. The requirements definition

task should be led and facilitated by a certified JIEM modeler, but should include input from business subject-matter experts who represent the interests of the stakeholders identified in the vision statement.

The **third task** in the project inception phase is to establish a process to be followed on the project (such as the process described in this module, with appropriate modifications to address specific needs or risks on particular projects). The process should identify deliverable milestones (e.g., domain model, GJXDM mapping, schemas, sample instances) and target dates on which those milestones are expected to be reached. In setting target dates for each milestone, it is important to set proper expectations with stakeholders and other project participants. In particular, the dates should be viewed as reasonable targets rather than exact predictions, because as the domain model unfolds, hidden complexities may be uncovered that compel either an adjustment to project scope or to milestone dates.

Finally, with scope, requirements, and milestones established, the project inception phase concludes by forming a project workgroup to conduct the rest of the process, and by identifying and procuring any tools or other technology needed for the project.

The project workgroup should consist of the following members:

- **Business Subject-matter Experts** who represent the interests of the stakeholders identified in the project's vision statement. These experts provide crucial business perspective on the information content of the exchange, as well as its context. They should have expertise in the business in general and the information exchange in particular. If existing enterprise software systems are involved in producing or consuming information in the exchange, it is useful if the users of these systems are represented on the workgroup.

- **GJXDM Experts** who have an understanding of structures in GJXDM and how to use available tools to find those structures quickly in an interactive setting. The GJXDM experts should be familiar with identified GJXDM best practices and techniques developed within the national justice community, and should have a firm understanding of XML and XML Schema technologies.
- **A Facilitator** with both business and GJXDM expertise (though perhaps less of each of these than the other workgroup members) who can lead the workgroup through the process. The facilitator's responsibilities also include leading the domain-modeling sessions, so it is important to choose a facilitator and modeling technique together that make this possible.

Detailed tool needs will be identified in later sections within this module; however, the following basic tools will be needed:

- **Tools to support domain modeling**, which will be specific to the modeling technique chosen. However, if reuse of domain-modeling artifacts is an important project goal (and it usually is), the project should select a modeling tool that is based on open standards and is widely available at a relatively low cost.
- **Tools to support mapping** of the domain model to GJXDM. Most workgroups have found that a simple spreadsheet works well.
- **Tools to support creation of schemas.** Again, the tool should support open standards, as well as justice community standards, such as the subset generator want list format.

Finally, once workgroup members have been chosen and milestone dates established, the facilitator should arrange for meeting resources (to support in-person and remote/telephonic meetings, as appropriate) and other communication tools. A project web site has

proved useful for many workgroups; the web site should contain a list of workgroup members and their contact information, a project plan identifying milestones, and a repository for project artifacts.

Figure 4.2.1 on page 78 summarizes the tasks to be completed in the project inception phase.

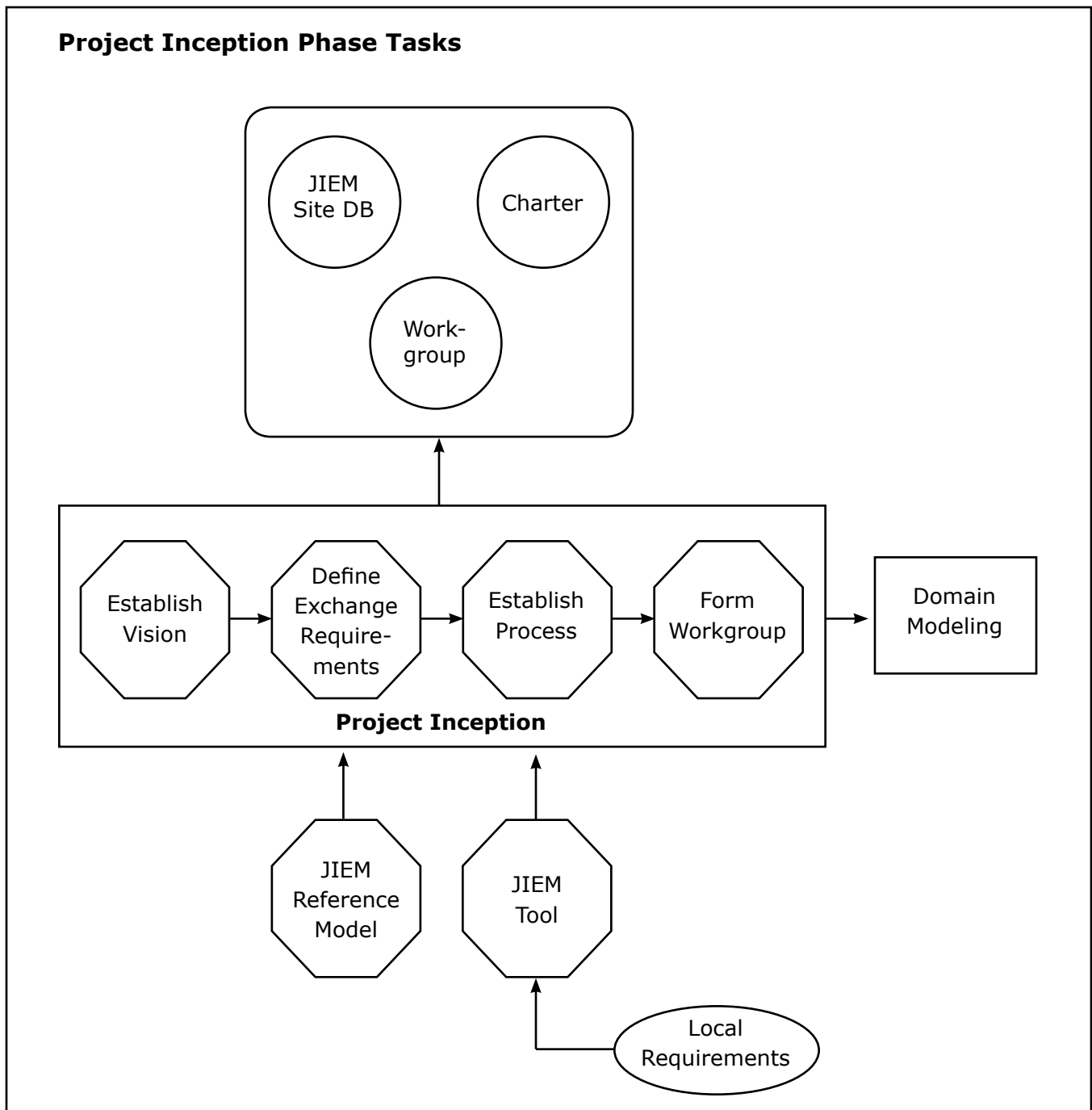


figure 4.2.1

PART 3 Domain Modeling

The second step in the exchange document development process consists of *domain modeling*. Domain modeling is an analysis activity through which business subject-matter experts reach agreement on the content and structure of the exchange document.

The output of the domain-modeling step is, not surprisingly, a domain model. This model can take many forms, as discussed below. However, the form of the model is not as important as its ability to facilitate the building of consensus among the workgroup. That is, the domain model is primarily a communication device—not for communication between the business experts and the schema-building technicians as separate groups, but rather for communication among the workgroup as a whole. The workgroup builds a domain model to represent, in a technology-agnostic way, what the information content of the exchange document needs to be.

It is important that the business subject-matter experts in particular need to be able to build consensus around the model. That is, the model needs to be something that nontechnical participants can agree or disagree with. This factor has important implications for the style and form of the domain model. In particular:

- The model should, in specifying information structures, use names and definitions that have meaning to the workgroup.
- The model should be built in a format and language that is easily understandable by everyone on the workgroup.
- The model should be easily consumable by the workgroup members, ideally without installation of special tools or specialized training.

When modeling the document structure domain, it is important to remember that the ultimate goal of the project is to build GJXDM-conformant schema, and to leverage

the valuable analytical content in GJXDM as much as possible. A significant amount of domain analysis was performed by the XML Structure Task Force in building GJXDM; exchange document development projects are well-advised to look to GJXDM as a source of ideas on how to structure information content in the domain. Leveraging GJXDM structures in the domain model also makes the mapping process (discussed below) easier. However, the workgroup should not feel constrained to use GJXDM concepts in the domain model if doing so would hinder the ability of the group to build consensus on the information content of the exchange document. The conceptual link between domain and GJXDM can be established later, in the mapping phase.

Domain Modeling Process

Prior to the first meeting of the workgroup, the facilitator should prepare a candidate domain model. Experience in Reference **Information Exchange Package** projects has demonstrated that the first workgroup meeting is much more productive if it works from such a “strawman,” as opposed to creating a model from scratch. In building the candidate model, the facilitator should determine if a Reference IEP exists for the document to be developed. If such a Reference IEP is available, and if that IEP contains a domain model, then that domain model is a natural starting point for the workgroup’s model.

The focus of the first workgroup meeting (typically 2–3 days of face-to-face time) should be working through the candidate model, making changes as necessary. The details of how this works will depend largely on the modeling format and notation used. During the session, a workgroup member should be tasked with keeping a list of issues (e.g., business rules that are not possible to document in the model, important concepts that are related but out of scope, topics that require further research outside of the workgroup, and so on).

The goal of the first workgroup meeting is to reach consensus that the domain model

effectively represents the information content of the IEP. It is the facilitator's main job to move the workgroup steadily toward this consensus. A good facilitator will regularly measure the status of the group to make sure that a consensus is forming.

After the first meeting, the facilitator should publish the domain model in such a way that the workgroup members (and perhaps a wider group of reviewers) can consume it as needed. If some minor, unresolved issues remain at the end of the first face-to-face workgroup meeting, these can be addressed remotely if necessary or can be the subject of additional follow-up meetings (if the workgroup is geographically co-located).

Experience has demonstrated that the domain model's comprehensibility is improved with a "high-level" or "summary" diagram that depicts the major document sections graphically and indicates their relationships. This diagram can be formal (e.g., a Unified Modeling Language (UML) Package or Class Diagram) or informal (e.g., a sketch in Visio® or on a PowerPoint® slide.) Such a diagram can orient a reader quickly and indicate where more detailed information on a particular substructure can be found. This technique is especially useful with complex document structures.

Generating a class diagram from schemas may be useful (to some), but this is not domain modeling. Domain modeling is as much about the *communication process* within the workgroup as the creation of an artifact. Also, graphically representing a schema does not change the fact that the structures and names are schema-oriented. XML Schema is not a particularly good format for communicating with nontechnical audiences. Generating a class diagram from the schema may in fact be useful to some technical staff, although many would probably rather just read the schema. In any case, if the generation of a class diagram is determined to be useful, this can be accomplished outside of the IEP artifacts.

Domain Modeling Notation Options

On Reference IEP projects, workgroups have had success building domain models in three formats: a "flat" textual model in the form of a spreadsheet, an informal graphical model, and a more formal graphical model built with the Unified Modeling Language. Other model formats are certainly possible, and this is not meant to be an exhaustive list of the possibilities.

In choosing a notation, facilitators and workgroups should bear the following in mind:

- Choose a notation that the facilitator is familiar with. An IEP development project is usually not the place for someone to learn UML or any other technique.
- Be pragmatic rather than dogmatic about notation formality. Choose something that works for the particular workgroup; there is no one right way to build a domain model.
- Measure the effectiveness of the selected modeling notation early and often, and adjust as necessary. Avoid letting the notation become a barrier to communication or consensus, and remember that the point of the domain model is primarily for communication *within the workgroup*.
- Also, bear in mind the opportunities for reuse of the domain model in other contexts. For example, is the IEP intended to be a statewide baseline, which will be further customized by county or municipal jurisdictions? If so, then closer adherence to open standard notations (like UML) and ubiquitous tools may be warranted.

Each of these notation options will now be examined in detail.

Informal Graphical Modeling

An informal graphical model consists of a diagram that depicts domain entities (things) as symbols, with arrows drawn between entities to indicate relationships. These diagrams are essentially “concept maps” in which the concepts being linked are components or “sections” of an exchange document.

Interpreting the symbols on an informal graphical model is similar to interpretation of classes and relationships on UML Class Diagrams; see the Class Diagram Interpretation guide illustrated in figure 4.3.2.

The advantages of informal graphical modeling are:

- This technique offers a graphical presentation, which can improve

communication of the context of each data element.

- There are generally no new tools to acquire and learn—Microsoft PowerPoint® and Visio® work well.
- This technique is very useful for high-level structural overviews, since it avoids inundating the reader in details.

The disadvantages of informal graphical modeling are:

- This technique is not effective at documenting the fine details of document structure.
- Notation needs to be invented to document important concepts like cardinality and inheritance.

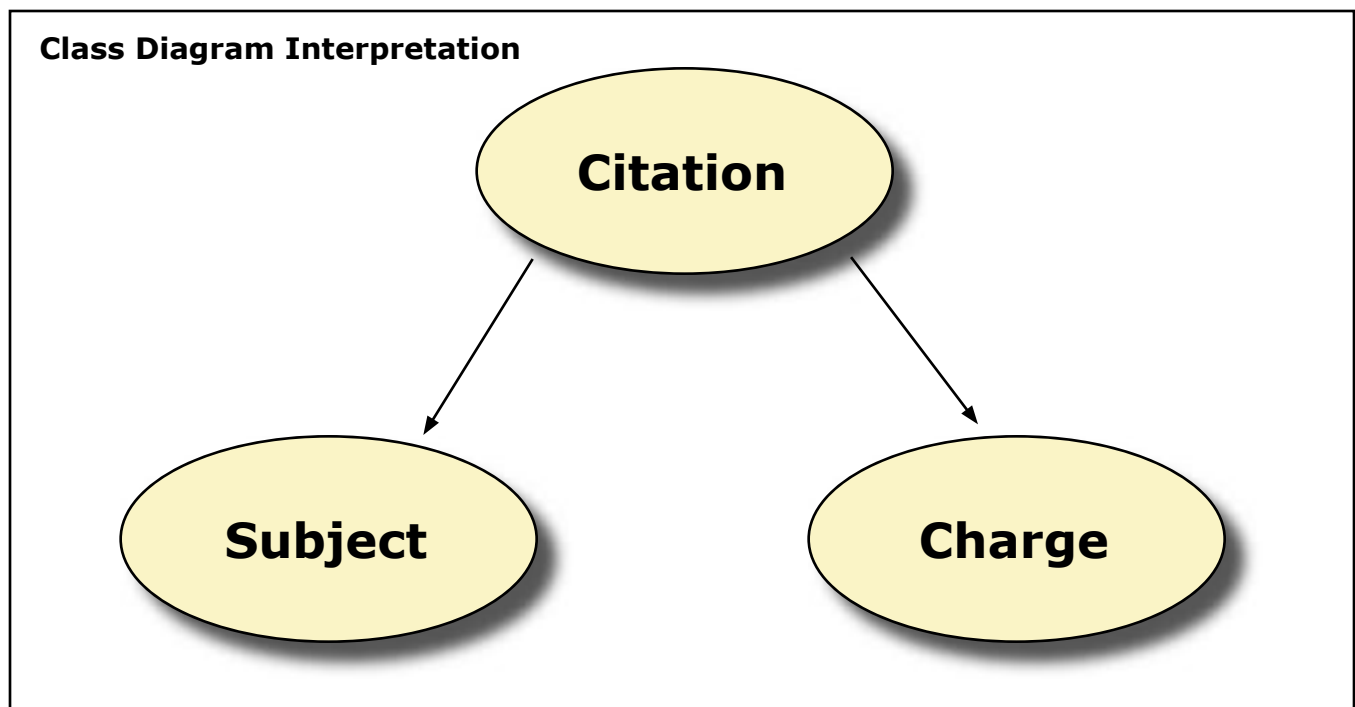


figure 4.3.2

UML Static Structure (Class) Diagrams

UML defines a diagram type, called a Class or Static Structure Diagram, that depicts domain entities and their attributes, as well as the relationships between entities. This type of diagram has built-in facilities for documenting entities at high or low levels of detail and for documenting important concepts like cardinality and inheritance. An example diagram is illustrated in figure 4.3.3.

A guide to UML Static Structure notation appears on the next page.

The advantages of modeling with a UML class diagram are:

- It offers a graphical presentation, which can improve communication of the context of each data element.
- It offers a precise and formal notation for depicting document structure, but at the same time is simple enough to be accessible to a wide range of stakeholders without requiring significant training or explanation.
- It supports object-oriented concepts inherent in GJXDM and XML Schema.
- It is supported by widely-available, low-cost tools (as well as commercial tools that cost more but have more robust features).
- It has widespread adoption in the technology industry and is familiar to most analysts and developers.

The disadvantages of modeling with a UML class diagram are:

- It requires the project to select a UML modeling tool.
- It requires that the facilitator be very familiar with both UML and the selected tool.
- Workgroup participants unfamiliar with UML will require coaching (though usually this is minimal).

Sample Static Structure Diagram

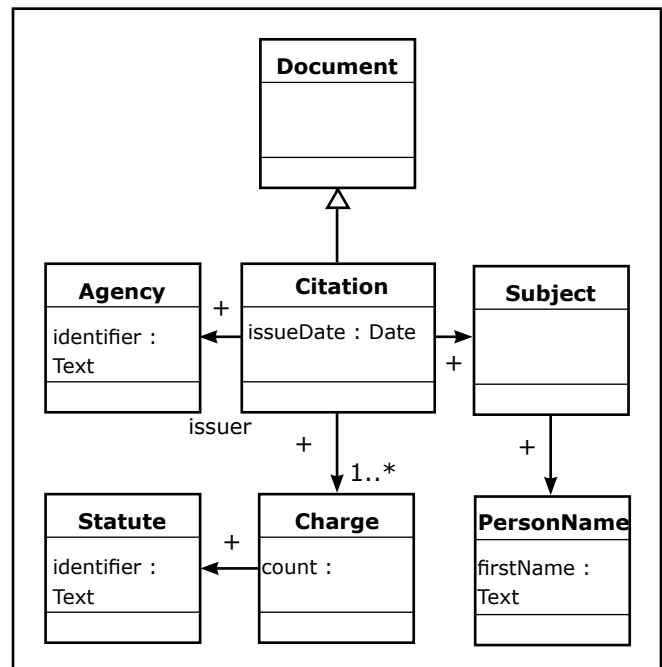


figure 4.3.3

If a workgroup elects to build its domain model using UML, its choice of modeling tool becomes a critical factor in the long-term success of the project. It is recommended to bear the following points in mind when selecting a tool:

- The modeling tool should be easy to use and familiar to the facilitator.
- The modeling tool should support creation of UML-compliant class diagrams.
- The modeling tool should support publishing of diagrams as ordinary image files (e.g., JPG or PNG).
- The modeling tool should support exporting the model's structure in XML Metadata Interchange (XMI) format so that the structure can be exchanged with other modeling tools if necessary.

Guide to Interpretation of UML Class Diagrams

This section provides a quick reference guide to the notation of UML Class Diagrams.

■ *Classes and Properties*

Figure 4.3.4 illustrates a class. The name of the class appears at the top. Below the top line appear “properties,” which are characteristics or attributes of the class.

A class name is always a noun. If one talks about the domain, the classes in the domain model will be the nouns in what is said.

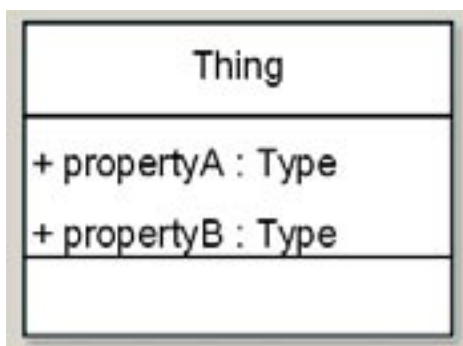


figure 4.3.4

■ *Associations*

To indicate an association relationship between two classes, an arrow is drawn with an open arrowhead, as shown in figure 4.3.5. The direction of the arrow indicates the hierarchy of the relationship; the class where the arrow starts is said to “contain” or “own” the class where the arrow points. At the arrow end, one can indicate how many things are “contained” or “owned” with *cardinality* indicators. This example indicates that “Each Thing1 contains one or more Thing2s.”

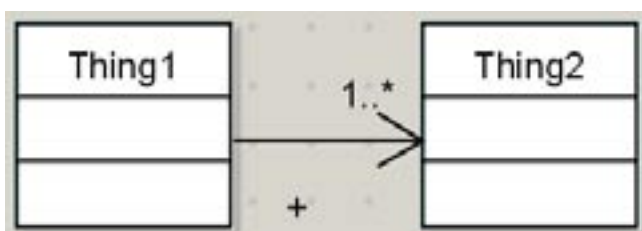


figure 4.3.5

■ *Inheritance*

Often in a model one will encounter entities that have a generalization/specialization relationship to each other. The concept of *inheritance* reflects this relationship. Inheritance is indicated by an arrow with a closed arrowhead, as shown in figure 4.3.6.

The class where the arrow points is the general class; the class where the arrow starts is the specific class.

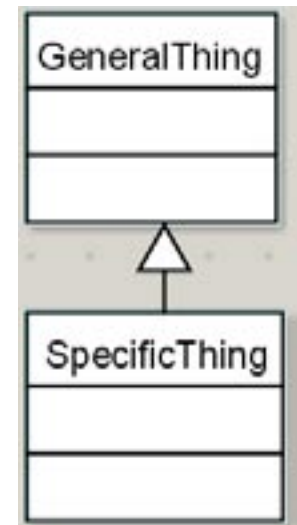


figure 4.3.6

Inheritance is used to share attributes. Everything that is true of the general class is true of the specific class. This is the sense in which the specific class “inherits” the properties of the general class.

This example is read as: “Each SpecificThing is a kind of GeneralThing.”

Figure 4.3.7 illustrates an example of entity inheritance further:

The *general class* entity, **Vehicle**, contains properties such as Vehicle Identification Number, make, and model. These properties are of type string, or, in other words, a combination of alphanumeric characters.

The *specific class* entity, **Truck**, inherits all of the properties of Vehicle, but contains additional specialized properties, such as axle count, which is represented as an integer.

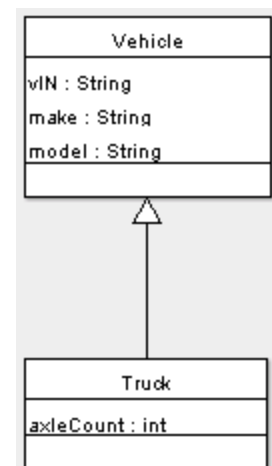


figure 4.3.7

Global Justice XML Data Model Types

An entire type structure from the GJXDM can be used, without having to identify each property contained therein. For example, if one wants to include *all* of the properties contained in the GJXDM VehicleInspection Type, they would be as follows:

- VehicleInspectionTestTypeText.
- VehicleInspectionJurisdictionAuthorityText.
- VehicleInspectionJurisdictionAuthorityCode.
- VehicleInspectionStationID.
- VehicleInspectionInspectorID.
- VehicleInspectionAddress.
- VehicleInspectionSmogCertificateCode.

Some reference IEP projects have evolved a convention of using a UML *stereotype* to indicate the incorporation of an existing GJXDM structure into the domain model. The conventional stereotype label is “GJXDM,” and, like all UML stereotypes, is represented by including the label within guillemots (<<GJXDM>>). An example of this appears in the diagram snippet on the VehicleInspection class (figure 4.3.8).

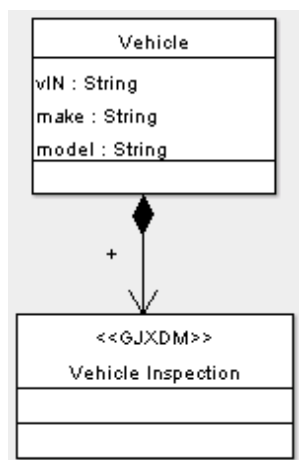


figure 4.3.8

PART 4 Domain-Global Justice XML Data Model Mapping

The third step in the exchange document development process involves associating domain model concepts and structures with types and elements in GJXDM. This association process is called *mapping*.

Each concept or class in the domain model, as well as each individual property or data element, needs to be associated with a particular type or element in the GJXDM schema. If no GJXDM type or element exists for a concept/class and property in the domain model, then a new type or element will need to be created in an *extension namespace* that is specific to the exchange document. The mapping artifact is designed to record these associations and extensions so that they can easily be input into the schema-building process.

(Note that if a spreadsheet is used for the domain model, the mapping artifact will simply

be the addition of a column to identify the association of the business data element to an element in GJXDM. There generally will not be a separate mapping artifact in this case.)

To perform the mapping, it is practically necessary to be able to search quickly and efficiently through GJXDM for types and elements that match the concepts in the domain model. Initially, this capability was provided by the spreadsheet that is included in the GJXDM distribution. While the spreadsheet can still be used, the [Wayfarer tool](#) (available in both online and localized versions) from the National Center for State Courts is a web application specifically designed for searching GJXDM. It is recommended that each workgroup and facilitator try each method to determine which method (or a combination) works best for their situation.

Most reference IEP development projects have recorded the mapping in a simple Excel spreadsheet artifact that looks like figure 4.4.1.

Sample Mapping Spreadsheet

Data Elements (Requirements)		Notes	GJXDM Mapping
Agency	Agency		Citation/CitationAgency/OrganizationName
	County Code	3	"Local code set"
Subject	Agency Code		Citation/CitationAgency/OrganizationLocalID/ID
	Name (Last)		Citation/CitationSubject/PersonName/PersonSurName
	Name (First)		Citation/CitationSubject/PersonName/PersonGivenName
Vehicle	Eye Color		Citation/CitationSubject/PersonPhysicalDetails/PersonEyeColorCode
	Vehicle License Plate Number	1	Citation/CitationSubject/Vehicle/VehicleLicensePlateID/ID
	Vehicle License State	1	Citation/CitationSubject/Vehicle/VehicleLicensePlateID/IDIssuingAuthorityText
Citation Info	License Type	4	"Local code set"
	Citation Number		Citation/ActivityID
	DOT Number	2	"Local field"
	Citation Time		Citation/ActivityTime
	Identification		Citation/CitationIssuingOfficial/EnforcementOfficial/BadgeID/ID
	District		Citation/CitationLocation/LocationLocale/LocaleDistrictName
Mapping Notes			
	1	Add Vehicle to an extension of SubjectType and substitute it into Citation/CitationSub	
	2	Add local components to an extension of CitationType and substitute it into Citation	
	3	Add local code to an extension of JurisdictionType and substitute it into Citation/CitationAgency/OrganizationJurisdiction	
	4	Add local code to an extension of IDType and substitute it into Citation/CitationSubject/Vehicle/VehicleLicensePlateID	
	5	Substitute DrivingIncidentType into Citation/CitationViolation	
	6	Add local codes to an extension of DrivingIncidentType and substitute it into	

figure 4.4.1

The columns of the spreadsheet contain the following information:

Column	Description
Class	The class (entity) from the domain model.
Property or Relationship	The property within a class, or a relationship of one class to another, from the domain model.
Cardinality	The cardinality (one, zero-to-one, zero-to-many, or one-to-many) of the property or relationship within the containing class.
GJXDM Mapping	A description of a type or element in GJXDM to which the property or relationship is to be mapped.
Notes	Any general notes about the association that can assist the schema builder in interpreting or using the mapping information.

The format of the GJXDM Mapping column warrants a bit of explanation. First, most workgroups have used a *path notation* to represent the GJXDM element being associated to the domain model property. (This notation should not be confused with XPath, a World Wide Web Consortium standard mechanism for identifying substructures within an XML document, even though at first glance it resembles XPath.) There are essentially two approaches to defining paths that different workgroups have used.

The first approach attempts to identify the associated GJXDM element all the way from the root of the exchange document structure. For example, a person's driver's license number on a citation might be represented by a path that looks like this:

CitationDocument/Citation/CitationSubject/DriverLicense/DriverAuthorizationID/ID

The advantage of this notation is that it is very precise and indicates exactly where in the resulting XML instance this information can be found.

However, it does have a disadvantage in situations where the same information occurs in multiple places in the domain model. For instance, the domain model for a Citation IEP may allow for inclusion of a witness's driver's license number as well. In the domain model, since both witnesses and subjects are people (actually, person roles), there will be a single Person class, and that Person class would be associated with a DriverLicense class, which would in turn have a property to reflect the driver's license number. The Person class would appear in the mapping spreadsheet on its own, detached from the other classes associated with it. So it is no longer possible to map the Person class properties to a single, full document path.

To handle this situation, paths are written that reflect only the immediate associations of each class. In this case, the following paths are written:

```
CitationDocument/Citation
Citation/CitationSubject
Citation/ActivityWitness
Person/DriverLicense/DriverAuthorizationID/ID
```

In the Notes column, it is helpful to note inheritance hierarchies, if doing so assists in reading the mapping. In this case, it would be helpful to note that CitationSubject and ActivityWitness are both of types that extend PersonType, to indicate that any properties included under Person are automatically included in the subtypes as well. Recognizing the importance of documenting inheritance hierarchies, some workgroups have included a specific column for this purpose.

In addition to path notation, the GJXDM Mapping column also needs to indicate when an association represents an extension. (Extensions are necessary when a domain model concept does not exist in GJXDM.)

Some workgroups have handled this with a color-coding convention, highlighting extension associations with a particular color (traditionally, yellow.) Other workgroups have created a separate column, titled "Extension (Y/N)," that marks the association as an extension with a "Y." Regardless of the notation used, this is an important piece of information for the schema builder, so it is important to have this facility and to keep it consistent within a project.

Finally, it is important to note that, for most exchange documents, there will likely be at least a few associations that cannot be described deterministically. That is, the association may have to be described in simple prose text that the schema builder will need to read to build the schema correctly. One of the consequences of this fact is that, in most cases, it will not be possible to automate fully the creation of schemas from domain model diagrams.

PART 5 Schema Building

The fourth step in the exchange document development process involves creating a set of GJXDM-conformant XML schemas that implement the document structure identified in the previous steps.

The principal input into the schema-building process is the mapping artifact from the previous step. The output is a collection of schemas that contain the identified structures from GJXDM, further constrain them as necessary, include extensions, and identify the top-level document structure. This set of schemas can then be used in a number of ways, as discussed below.

Exchange Document Schema Set

The end-product of the exchange document development process is not a single schema, but a set of linked schemas that serve different purposes.

There are generally four separate schemas in an IEP:

- A **subset schema**, which extracts from the full GJXDM namespace just those types and elements needed for the IEP.
- A **constraint schema**, which adds certain additional constraints or restrictions to the types and elements in the subset.
- An **extension schema**, which defines an IEP-specific namespace to hold types and elements needed for the IEP but that are not in GJXDM.
- A **document schema**, which defines the root element of the IEP.

GJXDM Subset Schema

The GJXDM *subset schema* selects just those types and elements from the full GJXDM schema that are needed for the exchange. A subset schema serves two purposes:

1. It can improve performance when parsing and validating instances, since there is less schema information for the parser or other tool to process.

2. It reduces the amount of information about the IEP's data structure that developers and tools need to work with at design time.

There is a single fundamental rule to which all subset schemas must adhere, namely:

Instances that validate against a subset schema must also validate against the full GJXDM schema.

In practice, this means that conformant subset schemas must have the following characteristics:

- They do not add types or elements beyond what is in GJXDM.
- They do not change the types of elements or the base types of derived types from what is in GJXDM.
- They do not change the name of any type or element in GJXDM.
- They do not change the order of elements that occur within a type in GJXDM.
- They are in the same namespace as the full GJXDM.

A subset schema contains just those types and elements that appear in the mapping artifact, plus any types or elements used by those types, and so on. Subset schemas also can restrict *enumerations* in code list schemas (for example, to restrict them down to just a set of codes used in a jurisdiction), remove *imports* of unused schemas, and remove unused attributes. Subset schemas also can omit documentation structures (i.e., annotation and documentation elements) from the full GJXDM.

It is certainly possible to create conformant subset schemas by hand. However, for exchange documents of significant size, hand-crafting subset schemas that satisfy all the conditions is tedious and error-prone. Consequently, the recommended approach for building subset schemas is to use the online [Subset Schema Generation Tool \(SSGT\)](#).

SSGT presents the schema designer with an interface that permits searching through GJXDM for desired types and elements. When these types or elements are found, the user may mark them for inclusion in the subset. When types or elements are marked for inclusion, SSGT applies the appropriate rules and selects any dependent types and elements as well. This frees the designer from having to manage all of the dependencies.

After marking all of the desired elements and types, the designer can generate the subset. The result is a Zip file containing the core GJXDM schema, plus all of the code list schemas that are referenced in the subset.

SSGT does not require authentication, which makes it easier to use and manage. However,

this means it also cannot persist users' subset schemas online between sessions. To save a partially-completed subset for completion at a later time, the designer can save a *wantlist* out of the tool. The wantlist is an XML document that lists all of the types and elements that have been marked for inclusion. The wantlist can be uploaded into the tool at a later time (even from a different browser or computer, or by a different designer) to "initialize" the tool with previously-selected types and elements. The designer can then make any modifications, additions, or deletions, then generate the subset. (The generated subset package automatically includes a copy of the wantlist as of the time of generation as well.)

Figure 4.5.1 provides a screenshot of SSGT.

SSGT Screenshot

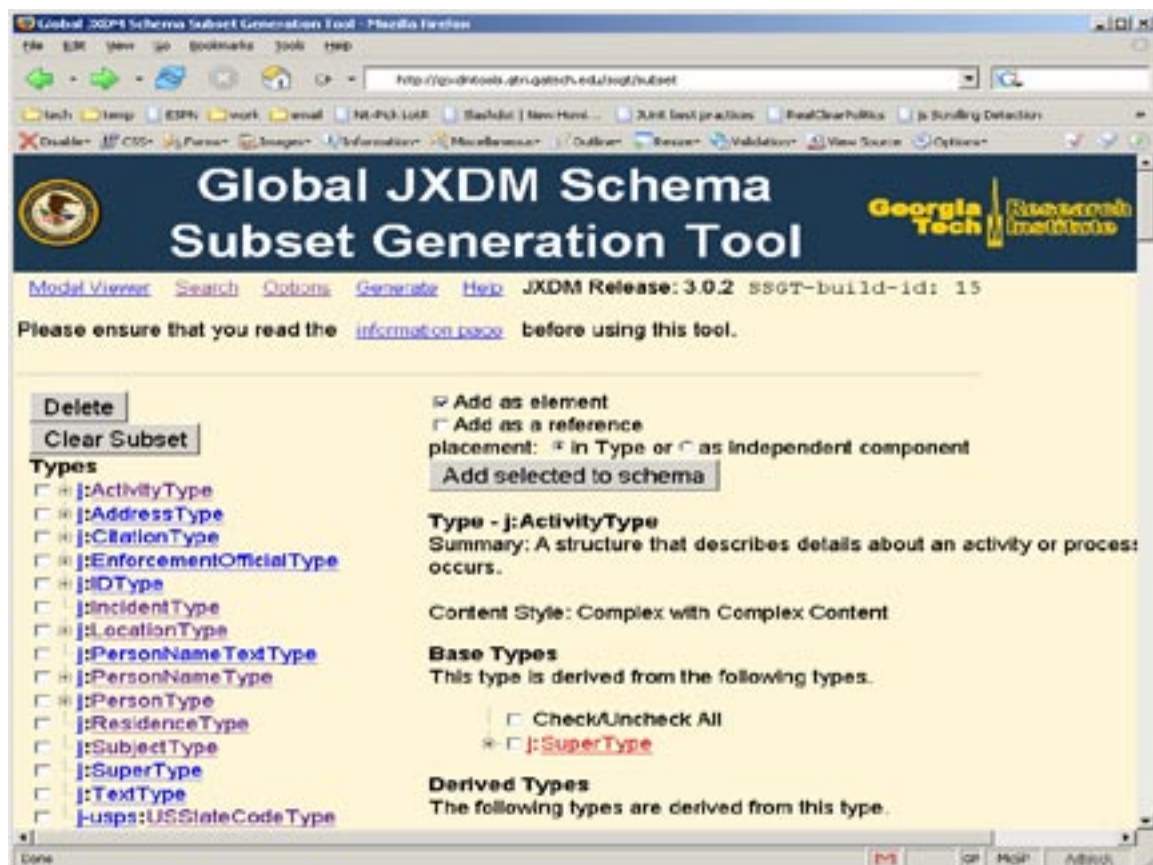


figure 4.5.1

Figures 4.5.2 and 4.5.3 illustrate screenshots of the online process to save a want list using SSGT.

Using SSGT To Save a Want List

The screenshot shows the SSGT web interface. On the left, there is a tree view of XML elements and attributes. The 'Elements' section is expanded, showing a list of elements: `j:ActivityData`, `j:ActivityDescriptionText`, `j:ActivityID`, and `j:ActivityTime`. The 'Attributes' section shows `j:personNameComment:Text`. On the right, there is a 'JXDM Release' dropdown menu set to '3.0.2' with a warning: 'Be careful, changing releases will clear your current want list. Be sure to save it if you would like to keep it.' Below this is a 'Select want list XML file to upload and edit' section with a 'Filename:' label, a text input field, and a 'Browse...' button. At the bottom right, there are two buttons: 'Load Want List' and 'Save Want List'.

figure 4.5.2

Using SSGT To Save a Want List

The screenshot shows an XML document displayed in a web browser. The XML is a 'wantList' document with a release of '3.0.2'. It contains several elements, including 'Citation', 'ActivityType', 'ActivityDate', 'ActivityDescriptionText', 'ActivityID', 'ActivityTime', 'AddressType', 'AddressFullText', 'LocationStateCode', 'CitationType', and 'CitationIssuingOfficial'. The XML is color-coded, with tags in blue and text in red. The browser window shows the address bar with the path 'C:\Documents and Settings\...'. The status bar at the bottom indicates 'Done'.

figure 4.5.3

Constraint Schema

The full GJXDM reference schema provides a common language through which its users can communicate in a manner that is semantically consistent. However, because GJXDM is defined for a large and varying group of users, it is impossible to embed all possible constraints and usages of that language into the reference schema. Therefore, the reference schema is unconstrained, very optional, and over-inclusive. It defines the language, but does not attempt to control exactly how people are going to use it.

As discussed in the previous section, the *schema subset* generated by SSGT allows the user to identify only those types and elements required for the information exchange. However, the types and elements included in the subset still adhere to the GJXDM philosophy of being “optional and over-inclusive.” In particular, the *cardinality* of all the elements is still “zero-to-many,” meaning each element can occur zero, one, or many times within its parent structure. In many cases, the exchange needs to restrict this cardinality further. This kind of cardinality restriction is an example of a business rule that can be implemented in a *constraint schema*.

Constraint schemas are a mechanism to embed constraints and business rules so that they may be validated by an XML Schema validator. Before they are described, it must be noted that the use of a constraint schema is completely optional; there are other ways of checking these business rules, and in some cases constraint schemas may be completely unnecessary. Business rules can be validated outside of XML Schema by embedding them in applications, XML Stylesheets (XSLT), Schematron (an assertion language), or other methods. Alternatively, it may not matter whether the constraints are met or not. Systems can choose to parse out the valid portions of the data they receive and discard the rest. For example, suppose an organization requires the last name of a person to be no more than 30 characters. If it receives an

instance document with a last name of 35 characters, it could choose to simply truncate the last name to its requirement rather than rejecting the instance document as invalid. This illustrates the notion that there are many different ways of dealing with constraints and business rules. XML Schema may not be the most powerful or rigorous method of defining such constraints, but it can be sufficient for validating common kinds of constraints. Furthermore, XML Schema precludes the introduction of new validators or other tools into the information exchange process.

A constraint schema is a simple way to define local business rules. Cardinality constraints, as discussed above, provide the primary constraint applied in constraint schemas. It also is possible to perform further “subsetting” in the constraint schema (e.g., removing elements, types, or enumeration facets) if that is desirable. (However, usually all subsetting is performed in the subset schema.)

It is important to note that the constraint schema does not change the GJXDM namespace. It also does not import the subset schema; rather, it replaces it.

The schema is defined in the same namespace as the GJXDM reference schema and defines the same content, but with the addition of constraints. Constraint schemas often are built by beginning with a copy of a schema subset. From that starting point, the constraint schema is modified to make changes to things like the default GJXDM cardinality and the addition of facets that constrain allowable data values (e.g., maximum name length = 30 characters, minimum age value = 18, license plate number must match pattern `'[A-Z]{3} d{4}'` - three uppercase alpha characters followed by a space and four digits). Choice blocks also can be inserted (e.g., either a person's social security number or both the name and the date of birth must appear in the instance), and types can be constrained differently based on how they are used in the document (e.g., changes can be made to a constraint schema such that only a

person name and badge number can be used with an enforcement official but a full set of person descriptors can be used with a subject).

The constraint schema does not add or change the semantics defined in GJXDM. It is not the place to add local extensions or content. The GJXDM reference schema and/or schema subset still defines the language being used; the constraint schema further defines local business rules about the GJXDM content that can appear in the instances.

The primary rule that must be followed when building constraint schemas is:

Instances that validate against a constraint schema must also validate against the full GJXDM schema.

This means that the only changes that one can make to a constraint schema are those that do not prevent instances from validating against the full GJXDM reference schema or a valid subset. Things that you cannot do in a constraint schema include changing element names, modifying the order or hierarchy in which elements appear, and modifying the definitions or semantics of GJXDM content. For example, changing the name of GJXDM element "PersonGivenName" to "firstName" in a constraint schema is not allowed. Any instance that appears with element "firstName" replacing element "PersonGivenName" because of changes made to the constraint schema would not be a valid GJXDM instance.

To ensure that invalid changes are not made to the constraint schema, even unintentionally, it is important that instances be validated against the full reference schema or schema subset to check for GJXDM language consistency in addition to validating against the constraint schema, which only checks for local business rules. This concept of making two passes to validate, whereby each pass checks for different constraints, is called *multi-pass schema validation*. The only change made during the different validation passes is to the *schemaLocation* attribute: the reference

schema or the schema subset and the constraint schema will have different file names (and possibly different paths). When an instance is validated against both the reference or schema subset and the constraint schema, it is not necessary to check the same thing twice. Anything that already has been checked by the reference or schema subset validation pass can be dropped by the constraint schema. For example, it is not necessary to validate VehicleMakeCode in an instance twice. The reference to the large National Crime Information Center (NCIC) code set can be dropped from the constraint schema.

The following illustration is an example of how a more complicated business rule, constraining a type differently based on how it is used in a document, can be implemented:

Suppose the exchange deals with enforcement officials and subjects. In this IEP, the only piece of information that should be captured for the enforcement official is the name and the badge number; however, a full set of person descriptors should be captured for the subject. In addition, the XML Schema validator must be able to check for this requirement (e.g., it is not sufficient that the person descriptors be made optional and over-inclusive in this case; the schema must not validate if person descriptors appear as part of the enforcement official).

Because GJXDM defines properties and types globally, there is only one definition of PersonType. The enforcement official and the subject both inherit from the same PersonType. This means the schema subset can either include a PersonType that has only a name or can include a PersonType that has a name and a full set of person descriptors. As is, the enforcement official will contain too much information (the name plus person descriptors), or the subject will contain too little (only a name). One could make PersonType only have name, and then extend SubjectType locally to include the additional GJXDM

person descriptors, but there is a better, more reusable way. One can create a constraint schema, which enables GJXDM types to be tweaked as long as the instances being used still validate against the reference schema.

In the constraint schema, the type inheritance from `PersonType` can be dropped. The properties that should be inherited from `PersonType` can be added directly to the enforcement official and subject. This means that one can explicitly add only a name and badge number to the enforcement official and then add the full set of person descriptors to the subject. The properties must be added in the same order and hierarchy in which they appear in the GJXDM `PersonType` so that the instance still conforms to the GJXDM.

It is important to note, again for emphasis, that the constraint schema has the same GJXDM namespace as the full reference schema or the subset. It does not import the subset or reference schema; it is a local copy of the GJXDM that users can modify to add constraints to GJXDM content.

Extension Schema

In many cases, an exchange document will require data structures that do not exist in GJXDM. These structures will be identified in the mapping step, since they will not map to anything in GJXDM. Since subset and constraint schemas cannot add structures to the GJXDM namespace, the new types and elements may be defined in an *extension schema*.

Extension schemas are provided as a mechanism to create reusable local components. If a local component is only expected to be used in a single document, it may be defined in that document's schema. If the local component is expected to be used in multiple documents, it can be defined once in the extension schema and reused (by importing and referencing) in the various document schemas. This often is simpler than defining

and keeping track of the component in each document schema in which it is used.

The extension schema defines an IEP-specific namespace (sometimes called a "local" namespace, but this term has a geographic connotation that is not always relevant). Because the IEP-specific types and elements are not part of GJXDM, there is no equivalent of SSGT for extension schemas. Extension schemas generally must be developed "from scratch" by writing XML Schema constructs.

It is recommended that every type in an extension schema extend some type in GJXDM (even if it only extends `SuperType`). Extending GJXDM types fosters reuse of GJXDM's semantics and also enforces consistency in use of metadata attributes. To make GJXDM namespace types (and elements) available in an extension schema, the extension schema must import the schema that defines the GJXDM namespace (which will be either a constraint schema, subset schema, or the full GJXDM reference schema). If the extension schema uses other namespaces within GJXDM (e.g., a codelist namespace) or outside GJXDM, it must import schemas that define those namespaces as well.

Dynamic Type Substitution and Static Element Replacement

One of the issues to be considered when building extension schemas is how to handle placement of new elements in the schema. The issue arises when the schema designer needs to replace an element in an existing GJXDM structure with a new element (whose type is a derived type of the original element's type).

For example, `CourtOrderType` in GJXDM is likely to be extended often to represent local- or document-specific aspects of a particular kind of court order. For instance, there is currently no native type in GJXDM to represent Sentence Orders. (This may be remedied in the future, but for now serves as an illustration of why `CourtOrderType` will need to be extended.)

From a “case-oriented” point of view, orders can be represented as follows in GJXDM (the type of each element in the containment hierarchy is indicated in parentheses):

```
Case (CaseType)
  CaseCourtEvent (CourtEventType)
    CourtEventActivity (CourtEventActivityType)
      CourtActivityCourtOrder (CourtOrderType)
```

Now, consider what needs to be done to extend CourtOrderType for the requirements of a particular IEP. For example, CourtOrderType currently does not include an element that could be used to contain an open-text narrative that a judge wishes to make part of the order. (Again, this could be remedied in a future version of GJXDM, but the example still serves to illustrate the point.) The way to handle this is to replace the CourtActivityCourtOrder element in the hierarchy above with another element, whose type is a derived type of CourtOrderType, but that includes an element for a narrative.

That is, the structure would be changed to look like this:

```
Case (CaseType)
  CaseCourtEvent (CourtEventType)
    CourtEventActivity (CourtActivityType)
      CourtActivityCourtOrder (xsi:type=ext:CourtOrderWithNarrativeType)
```

That is, CourtOrderWithNarrativeType extends CourtOrderType and includes a new element (e.g., CourtOrderNarrative, of type TextType) to hold the order’s narrative.

There are essentially two options for handling this situation. In both cases, each new element (as well as that element’s type) are defined in the extension schema.

In the first option, called **dynamic type substitution**, the GJXDM structure that contains the new element remains unmodified. When constructing an instance, the structure of a derived type may be substituted for an expected type structure. The original element

remains the same. The only requirements to using type substitution are that the name of the type to substitute must be specified as the value of attribute "xsi:type," an attribute of the original element, and only derived types may be substituted. (For example, SubjectType may be substituted for PersonType, but VehicleType may not be substituted.) Using this option, an XML instance for the court order example above would look like this (assuming the GJXDM namespace is prefixed with "j," the extension namespace is prefixed with "ext," and the Schema Instance namespace is prefixed with "xsi"):

```
<j:Case>
  <j:CaseCourtEvent>
    <j:CourtEventActivity>
      <ext:CourtActivityCourtOrder xsi:type="CourtOrderWithNarrativeType">
        ...
      </ext:CourtActivityCourtOrderWithNarrative>
    </j:CourtEventActivity>
  </j:CaseCourtEvent>
</j:Case>
```

In the second option, called **static element replacement**, the element substitution is done in the schema, rather than the instance. This requires extending types all the way up the containment hierarchy to the root of the schema. (While this does result in a larger number of derived types in the extension schema, it has the advantage of making the element replacements explicit in the schema.) To make the element replacement explicit, it is recommended that the schema designer include documentation in the schema indicating the name of the element being replaced. The "appinfo" namespace included in GJXDM since version 3.0.2 provides an element suited for this purpose.

Using static element replacement, and assuming the same namespace prefixes from the previous page are used, the court order example would look like this:

```
<ext:Case>
  <ext:CaseCourtEvent>
    <ext:CourtEventActivity>
      <ext:CourtActivityCourtOrderWithNarrative>
        ...
      </ext:CourtActivityCourtOrderWithNarrative>
    </ext:CourtEventActivity>
  </ext:CaseCourtEvent>
</ext:Case>
```

Each of the types involved in this structure would appear in the extension schema namespace, but would extend the corresponding types in the GJXDM namespace. (Note: These extensions do not really change the information content of the corresponding GJXDM structures; rather, they make the semantics of the IEP more precise.)

For example, the CourtActivityType type in the extension namespace would look like this:

```
<xsd:complexType name="CourtActivityType">
  <xsd:complexContent>
    <xsd:extension base="j:CourtActivityType">
      <xsd:element ref="ext:CourtActivityCourtOrder" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
```

And the declaration of the CourtActivityCourtOrder element, using the “appinfo” documentation structure (and assuming the GJXDM appinfo namespace is prefixed with “i”), would look like this:

```
<xsd:element name="CourtActivityCourtOrder" type="ext:CourtOrderWithNarrativeType">
  <xsd:annotation>
    <xsd:appinfo>
      <i:info>
        <i:base i:namespace="http://www.it.ojp.gov/jxdm/3.0.2"
          i:name="CourtActivityCourtOrder"/>
      </i:info>
    </xsd:appinfo>
  </xsd:annotation>
</xsd:element>
```

■ **How to Choose Between the Dynamic Type Substitution and Static Element Replacement**

Both dynamic type substitution and static element replacement have advantages and disadvantages.

The main advantage of dynamic type substitution is that it achieves the desired element replacement without requiring definition of as many derived types in the extension schema. Keeping the original element name and path is another big advantage of type substitution. Even if other users don't understand your local CourtOrderNarrative extension, they can still process the original “CourtActivityCourtOrder” very easily because it has the same name and same path as expected.

However, dynamic type substitution also has several disadvantages:

- Complete definition of the containing types is deferred until instances are created.
- The schema designer has to describe where type substitution is required in a nonschema artifact, which makes it difficult to use the schema as the basis for exchange.
- Many common XML tools and messaging infrastructure do not support type substitution in instances.

The advantages and disadvantages of static element replacement are just the reverse of dynamic type substitution. It does result in more derived types in the extension schema. However, the schema can be used as a more complete definition of the IEP structure, making the semantics clear and explicit. Also, static element replacement enjoys broader infrastructure support.

Options for Relating Information: Inclusion, References, Relationships

The GJXDM provides three basic ways of associating two pieces of information: inclusion, referencing, and relationships.

■ Inclusion

The first method is called *inclusion* and means, in the XML sense, that one piece of information is contained within another. Inclusion is one of the fundamental building blocks of GJXDM and is used throughout the data model. It is implemented in GJXDM by an element simply being included in a sequence within a complex type. For example, the `PersonType` structure in GJXDM contains an element called `PersonBirthLocation`, which represents the place where someone was born. Linking the person to that place can be accomplished by inclusion, by including the `PersonBirthLocation` element (which is of `LocationType`) within the `Person` element, like this:

```
<j:Person>
  <j:PersonBirthLocation>
    <j:LocationAddress>
      <j:LocationCityName>Boston</j:LocationCityName>
      <j:LocationStateCode.USPostalService>MA</ j:LocationStateCode.USPostalService>
    </j:LocationAddress>
  </j:PersonBirthLocation>
</j:Person>
```

■ Referencing

The second association method is called *referencing* and means that one piece of information contains a reference or “pointer” to another piece of information described elsewhere in the XML instance. This facility also is provided natively in GJXDM, as every element included within each complex type is paired with a “reference” element that points to a structure at some other point in the instance. These reference elements are of GJXDM’s type *ReferenceType*, which basically just contains an attribute that references an element with a particular unique identifier attribute. Using referencing, the above birth location example would look like this:

```
<j:Location j:id="location1">
  <j:LocationAddress>
    <j:LocationCityName>Boston</j:LocationCityName>
    <j:LocationStateCode.USPostalService>MA</ j:LocationStateCode.USPostalService>
  </j:LocationAddress>
</j:Location>

...

<j:Person>
  <j:PersonBirthLocationReference j:ref="location1"/>
</j:Person>
```


■ Relationship

The third association method is accomplished by using a separate structure, outside of either of the two pieces of information, called a *relationship*, which links the two pieces of information. This facility also is provided by GJXDM's RelationshipType structure, which contains attributes for a "subject" (one side of the association), an "object" (the other side of the association), and a name for the association or relationship. Using a relationship, the above birth location example would look like this:

```
<j:Location j:id="location1">
  <j:LocationAddress>
    <j:LocationCityName>Boston</j:LocationCityName>
    <j:LocationStateCode.USPostalService>MA</ j:LocationStateCode.USPostalService>
  </j:LocationAddress>
</j:Location>

...

<j:Person j:id="person1">
  ...
</j:Person>

...

<j:Relationship j:subject="person1" j:object="location1" name="j:PersonBirthLocation"/>
```

Note how the name attribute on the Relationship element is a "qualified name," meaning that it includes a namespace prefix just like XML elements do. This is to support namespaces for relationship names, for the same reasons that namespaces are needed for XML names. Also note that the relationship name is the same as the name of the equivalent subject property.

Following this convention makes the semantics of the relationship clear; however, a schema-aware parser *will not* be capable of validating that this name matches a particular property or any property at all. If an application requires validation of the relationship name, this functionality will have to be implemented outside of the schema. Also, if the relationship does not have an equivalent property in

GJXDM, then the schema designer will need to choose a new name and define that name in the extension schema.

■ ***How to Choose Between the Options***

Inclusion is recommended as the “default” choice and should be used in all cases where the information is truly hierarchical (that is, if there is truly a containment relationship and the contained information is not represented somewhere else). References are useful when the object of an association has unique identity and appears in multiple places. However, schema designers should ensure that the object truly has identity, and should not use references just because the information happens to be the same. References should not be used as a tool to reduce the size of instances, since using a reference conveys semantics about two subjects referencing the same object.

Relationships can be used under similar circumstances as references, but are particularly useful when the association is not already defined in GJXDM. Relationships result in simpler extensions and more flexibility than extending GJXDM types to include new elements. However, using relationships requires the schema designer to choose names for the relationship (in cases where the association is not represented by an existing GJXDM property), and there is no agreed-on standard for what these names should be. Finally, one drawback of using references and relationships is that these mechanisms can potentially add to the complexity and memory consumption of software built to process instances, since that software must maintain a list of objects and their identifiers to re-establish association links as the instance is parsed.

Document Schema

A **document schema** is a schema that contains the root element and the root type for the IEP, plus any local extensions that are not already defined in an extension schema. Since this schema is IEP-specific, it must define an IEP-specific namespace. The root type in this document schema defines the top-level structure of the instance document. In most cases, this root type will be an extension of the GJXDM `DocumentType`, since `DocumentType` is intended to represent “documents.” The document schema will import the extension schema (if it exists) and/or the subset, constraint, or full reference schema, depending on the validation pass.

PART 6 Building Schema-valid Instances and Packaging

Building Schema-valid Instances

In general, it is a good idea to include a sample XML instance in the IEP distribution as a means of illustrating with sample data what an instance should look like. A sample instance also is valuable as a mechanism for testing the integrity of the schema set. (It is not uncommon to discover types and elements missing from the subset schema, or errors in the extension schema, once an instance is first validated using the schema set.)

Beyond providing a sample instance in the distribution, it is useful for the schema designer to keep in mind all the ways in which an IEP schema set might be used. Schemas are useful for validating the structure of instances, but they also are useful in other ways, such as:

- Precise documentation for developers as to what an instance should contain.
- Input into tools, such as XML-object binding tools, that generate programming code for manipulating instances.
- Source of message structures for web services.

The extent to which these schema usage scenarios should be tested is, of course, dependent on which of them, if any, are critical to the project at hand. Nonetheless, keeping in mind how the schemas might be used in the future is a useful exercise in maximizing reuse.

Packaging

The final step in the exchange document development process consists of packaging all of the other artifacts into a unit and writing a short document that describes the artifacts.

It is recommended that the IEP be accompanied by a document (called IEP Documentation) that describes:

- The business purpose of the IEP.
- The list of artifacts that are supplied in the IEP package (domain model artifacts,

mapping artifacts, schemas, sample instances, etc.).

- A list of business rules, usage scenarios, implementation notes, and other information that the workgroup wishes to provide to users of the IEP.
- A summary of proposed changes to GJXDM, including definition modifications.
- A list of who participated in the development of the IEP (i.e., the workgroup members) and what process was used to build the IEP.
- Any testing or conformance assessment performed as part of the IEP development effort.

It is recommended that the IEP be organized in a base directory structure as follows:

- the IEP Documentation document,
- the constraint, extension, and document schemas, and
- at least one sample instance.

In addition, a subdirectory, called “subset,” should contain the contents of the subset schema package produced by SSGT. This directory structure can be assembled into a Zip file, if desired, for easy distribution to stakeholders.

Module 005 provides several example IEPs as models for justice users.

MODULE 005

Global Justice XML Data Model Use Cases for Justice and Public Safety

PART 1 Field Interview Report

Introduction, Purpose, and Scope

Between November 15–17, 2004, the Bureau of Justice Assistance (BJA), with support from SEARCH, the National Law Enforcement and Corrections Technology Center-Southeast (NLECTC-SE), and the Los Angeles County (California) Information Systems Advisory Body (ISAB), sponsored a Law Enforcement Global Justice XML Data Model (GJXDM) Reference Document Workshop. During this workshop, criminal justice practitioners and information technology industry representatives met to create and document a reference Information Exchange Package—IEP—and documentation for a **Field Interview (Investigation) Report**.

A Field Interview Report (FIR) is record of a field officer's stop of an individual or vehicle. The officer uses the FIR to gather information about an individual who is acting suspiciously, may have had the potential of committing a crime, or is being checked while passing through a jurisdiction's boundary. Information is captured about the event to record the individual, vehicle, location, and field officer related to the FIR for the purpose of recording the event for possible future investigations and identification. At the time of the event, *no assumption of guilt of the individual or probable cause has been established by the field officer*, and no other reportable action was taken by the officer. The field officer is only capturing the event for purpose of recording the interview with the individual.

The intent of the Field Interview Report IEP is to provide a *reference model* of consistently structured and reusable pieces of information to be leveraged by the law enforcement community for its internal use and for information exchange with other justice partners.

Participants

Travel to the Field Interview Report IEP workshop and meeting costs were funded through a grant from the U.S. Department of Justice's (DOJ) Office of Community Oriented Policing Services (COPS). Additional meeting space was provided by ISAB. All of the workgroup participants volunteered their time, and several covered their own travel expenses, to participate in the workgroup activities.

The following individuals were part of the Field Interview Report IEP Workgroup:

Winfield Wagner, Crossflo, Facilitator*

Tim Wilson, TriTech, Co-facilitator

Bona Nasution, MTG, Technical

Tom Kooy, Justice Information Sharing Professionals (JISP), Subject-matter Expert (SME)**

Dustin Henson, ARJIS, SME and Technical

Coleman Knight, NLECTC-SE, SME

Karen Cordray, North Charleston (SC) Police Department, SME

*Mr. Wagner was formerly with ARJIS (Automated Regional Justice Information System).

**Mr. Kooy was formerly with CriMNet.

Workshop Methodology

Step 1: Initial Data and Document Collection to Establish IEP Requirements

One month prior to the scheduled workshop, the facilitator began collecting as many source documents, database structures, and forms used by an assortment of criminal justice agencies as possible.

The packet of workshop support information included documents from these California agencies:

- Input screens for field interviews from the Anaheim Police Department.

- Element list from the San Diego Harbor Police Department's database.
- Example FIR used by the San Diego Harbor Police Department.
- Example FIR used by the Los Angeles County Sheriff's Department.

If the members were unable to provide their reports, screens, and database reference documentation before the start of the workshop, they were instructed to bring those items to the workshop for group discussion. During the workshop, the following items were collected and reviewed:

- Field interrogation card used by the Seahawk Task Force (Charleston, South Carolina).

From the collected documentation, the facilitator created domain model documentation to be used in the workshop, which included:

- A Microsoft PowerPoint® diagram illustrating the base objects (complex types) related to the information captured by a FIR that could be used by all agencies.
- A general Microsoft Excel® spreadsheet that identified each GJXDM 3.0.2 simple element, the full GJXDM complex type (object path) for the simple element, and a general description of the simple element required to capture all the information related to the FIR.
- A detailed Microsoft Excel® spreadsheet depicting the complex and simple types used by the FIR, and parent-child object relationships for each.

Step 2: Pre-workshop Distribution of Workshop Documentation

One week prior to the workshop, the initial package of documentation was distributed to the invited members. The facilitator made the request for additional user documentation related to FIRs and support documentation. From that request, the following additional user documentation was added to the package:

- TriTech input screens for field interviews.

The facilitator incorporated the additions and changes into the documentation package, and the final workshop documentation package was distributed to the invited members 3 days prior to the start of the workshop.

Step 3: Workshop Process

The workshop objective was to gather the object and element requirements, represented in a domain model, to support the construction of a reference GJXDM 3.0.2-conformant IEP and documentation representing a FIR.

During the first morning of the workshop, all workgroup members participated in a collaborative 2-hour session on *Level Setting and Overview of the Process*, which provided a background and framework of GJXDM and IEP objectives.

Workgroup members agreed to the following principles for the construction of IEP artifacts:

- Following a principle established by earlier workshops, the base object for the schema would represent a law enforcement reference document represented by *FieldInterviewReport* derived from the GJXDM 3.0.2 *DocumentType*.
- The development of the IEP would use objects and tags derived from or extended appropriately from the GJXDM 3.0.2 model.
- Missing elements or objects (complex types), identified by the workshop members, would be noted and provided to the XML Structure Task Force (XSTF) as potential extensions or modifications to GJXDM.
- The IEP and documentation created would serve as a reference model to support the sharing of information across justice agencies and would not be a representation of any specific agency's system or document/form.

- If a piece of information could be represented in GJXDM 3.0.2 as both a Text Type and Code Type element (i.e., race, sex, hair color, eye color, etc.), the developed schema would include both element-type tag names.
- As much as possible, object relationships would follow the hierarchy of GJXDM 3.0.2 inheritance.
- Development of more specific extension or constraint schemas would not be the responsibility of the IEP workgroup, but that of any individual agency intending to leverage the reference IEP as a baseline for its local implementation.

Using the Microsoft PowerPoint® diagram that illustrated the base objects (complex types) and their hierarchical relationships, the members reviewed, added, modified, and approved the objects (complex types) that describe a FIR. For this workgroup, the term “object” represented a packet of information or GJXDM 3.0.2 objects (complex type elements) that represented the people, places, and/or things described in a FIR.

During breaks in the members’ discussion, the facilitators completed iterations of the following tasks:

- Updated the documentation.
- Constructed a first cut of the schema using the GJXDM sub-setting tool.
- Noted items and ideas that would help future users of the schema in understanding the content of the schema.
- Noted items and ideas that may impact future versions of GJXDM.

Workshop Deliverables

It was the responsibility of the facilitators to create a Zip folder file that provided the necessary documentation to describe and use the IEP schemas, as well as the additional documentation (i.e., domain model(s), GJXDM mapping spreadsheet). The following [artifacts](#) are provided for the Field Interview Report IEP:

Artifact File Name	Description
FIR Object Diagram	PowerPoint® diagram illustrating the objects and their relationships within a FIR
FIR Element List	GJXDM 3.0.2 element list with full path and general description
FIR Domain Model	Visio® diagram of the complete FIR domain (data) model
FIR Want List	GJXDM 3.02 element want list generated by the sub-setting tool provided by Georgia Tech Research Institute (GTRI)
FieldInterview.xsd	Reference schema
jxdm Folder	Completed subset and annotated schema set that represents a “base” GJXDM 3.0.2 FIR Document
FIR Instance	XML instance example of a FIR
FIR Schema User Notes	Items identified by workgroup participants that will help users understand the schema

The [FIR Schema Documentation V1.0 Zip folder](#) also will include a subfolder of all additional workshop and support documentation. The subfolder is titled “FIR reference.”

Post-workshop Process

Following the workshop, the following steps and tasks were completed:

- The facilitator and co-facilitator reviewed the notes and support documentation and finalized the first drafts of the workshop deliverables.
- A draft FIR IEP documentation Zip folder was distributed to the other members of the workgroup for review and comment.
- Comments, corrections, or adjustments identified during review were distributed back to the facilitators.
- The facilitators made appropriate changes or adjustments to the final draft of the IEP documentation.

Field Interview Report Information Exchange Package Proof of Concept

The FIR IEP was successfully implemented in a proof of concept exchange, known as the Southern California Port Security Regional Information Sharing & Analysis (RISA), sponsored by the following agencies:

- San Diego Harbor Police Department.
- Port of Los Angeles.
- Los Angeles Joint Drug Intelligence Group.

In that exchange, FIR information was collected from the following local harbor law enforcement agencies and published in a GJXDM 3.0.2 instance based on the initial FIR IEP:

Agencies/Jurisdiction:

- Los Angeles County Sheriff's Department
 - Source: Consolidated Criminal History Reporting System (CCHRS)
 - Data: Booking and Rap sheet
- Los Angeles Port Police Department
 - Source: Record Management System
 - Data: Field Interview
- San Diego Harbor Police Department
 - Source: Records Management System/Computer-aided Dispatch
 - Data: FIR and Call-for-Service

- Unified Port of San Diego
 - Source: Marine Operations Information System (MOIS)
 - Data: Ship/Cargo Manifest
- San Diego Lindbergh Field
 - Source: Airport Control System
 - Data: Employee Access Control List
- U.S. Coast Guard

The RISA project employs a publish/subscribe GJXDM publication-sharing model. The major goal of the project was to demonstrate that multiple agencies could exchange FIR data and other forms of information between each other and to a single intelligence analysis tool through data exchange.

The implementation of a standardized reference IEP model for FIR, within the data exchange, proved that the RISA model was scalable to support an increasing number of participating agencies with little impact to their systems or day-to-day activity.

RISA publishes data from six distinct originating data sources, representing eight different and unique types of law enforcement and maritime security information (noted above). From concept to implementation, the RISA project was implemented in less than 30 days.

PART 2 Incident Report

Introduction, Purpose, and Scope

Since the release of GJXDM Version 3.0 in January 2004, practitioners and developers from various justice “communities of interest” have identified strategies to develop Reference Exchange Documents, now known as an **Information Exchange Package** Documentation, derived from GJXDM components.

SEARCH recently completed a project to develop GJXDM IEPs for Law Enforcement, including a reference IEP for **Incident Report**. This project was supported by collaboration with the Integrated Justice Technical Committee of OASIS (Organization for the Advancement of Structured Information Standards), NLECTC-SE, the Los Angeles County ISAB, the Maricopa County (Arizona) Integrated Criminal Justice Information System (ICJIS) Project Office, the Law Enforcement Information Technology Standards Council (LEITSC), the Integrated Justice Information systems (IJIS) Institute XML Advisory Committee, JISP, and numerous domain experts from local law enforcement, state law enforcement, transportation, and the FBI.

The business purpose of the Incident Report Reference IEP is as follows:

- To provide a model (and potentially an aspirational standard) for electronic exchange of incident information between law enforcement agencies and local, state, and national partners.
- To provide a baseline model, especially with regard to incident data, for use by other first-responder and incident management organizations.
- To provide a schema that can be used to structure XML instances that law enforcement agencies can use to report to the FBI N-DEx (National Law Enforcement Data Exchange) program; it is likely that such instances would require transformation into a structure specified by a separate N-DEx schema.

The intent of the Incident Report IEP is to provide a *reference model* of consistently structured and reusable pieces of information to be leveraged by the law enforcement community for its internal use and for information exchange with other justice partners.

Participants

Travel to the Incident Report IEP workshops and meeting costs were funded through a grant from the COPS Office. Additional meeting space also was provided by the Los Angeles County ISAB, the Maricopa County ICJIS Office, and the IJIS Institute. Many workgroup participants volunteered their time and covered their own travel expenses to participate in the workgroup activities.

BJA and COPS coordinated the Incident Report IEP Workshops, and SEARCH served as the facilitator. One of the major tasks was the initial selection of participants. Individuals from the criminal justice information management and technology professions were selected for the following reasons:

- Law enforcement practitioners/subject-matter experts (SMEs) with a comprehensive understanding of the day-to-day use and management of the specific document type being developed (Incident Report exchanged horizontally among justice agencies, as well as statistical reporting of incident data to the state, and federal reporting such as Uniform Crime Reporting (UCR), National Incident-Based Reporting System (NIBRS), and N-DEx).
- Technology specialists from the vendor community with practical experience in developing and implementing GJXDM tools and instruments, including schema.
- Participation in either the OASIS technical committees, IJIS Institute, or JISP subcommittees currently working to establish standards for the use of GJXDM.

The following individuals participated in the workgroup to develop an Incident Report IEP:

Participant	Organization	Workgroup Role
Debra Cohen, Ph.D.	COPS Office, U.S. DOJ	Project funding and coordination
Robert Greeves	BJA, U.S. DOJ contractor	Coordination and collaboration
Christopher Traver	BJA, U.S. DOJ contractor	Coordination and collaboration
Jennifer Hicks Zeunick	LEITSC	Coordination, education, and outreach
Scott Came	Justice Integration Solutions	Facilitator
Catherine Plummer	SEARCH	Project Lead, Co-facilitator
John Aerts	Los Angeles County (CA) ISAB	SME
Scott Edson	Los Angeles County (CA) Sheriff's Department	SME
Jeff Harmon	Northrup Grumman*	SME
Jackie Vandercook	Tennessee Bureau of Investigation/NIBRS	SME (NIBRS/UCR)
Paul Herman	Baltimore (MD) Police Department	SME
Scott Shaw	Baltimore (MD) Police Department	SME
Jeffrey Cooper	Baltimore (MD) Police Department	SME
Rick Brown	FBI	SME
Sarah Wilson	FBI	SME (NIBRS/UCR)
Cherie Morgan	FBI contractor	Technical (N-DEx coordination)
Scott Smith	FBI contractor	Technical (N-DEx coordination)
David Kelley	U.S. Department of Transportation/Institute of Electrical and Electronics Engineers	Technical transportation/IEEE 1512 coordination
Eric Lockhart	Marietta (GA) Police Department	SME
Monica North	Albuquerque (NM) Police Department	SME
Dennis Frye	ARJIS	SME
Dustin Henson	ARJIS	Technical
Brenda Ray	Wisconsin Dept. of Administration, Division of Enterprise Technology	SME, Technical

*Mr. Harmon was formerly with the Maine State Police.

Workshop Methodology

In June 2004, a workshop, sponsored by BJA with support from SEARCH and the OASIS Integrated Justice Technical Committee, had been conducted to establish GJXDM base schemas for court reference exchange documents. During that workshop, an initial methodology was established to capture a suite of documentation for developing domain-focused, GJXDM-conforming schemas. The methodology was based on obtaining the data and business requirements for a reference schema based on real-world database structures, documents, and forms used by justice practitioners. Technical participants turned those requirements into subset, extension, constraint, and document schemas based on the GJXDM components.

This methodology, with some modifications to the steps and deliverables, was used as the basis for the Incident Report IEP Workshops, which took place in November 2004 and January 2005. The following is a review of that modified methodology and its steps:

Step 1: Initial Data and Document Collection to Establish the Base Domain Model

One month prior to the scheduled workshop, the facilitators began collecting numerous source documents, database structures, and forms used by an assortment of justice agencies.

Development artifacts reviewed by IEP workgroup teams included:

- Records and case management system input screens.
- Data dictionaries and element lists.
- Example incident reports.
- Supplemental reports collected to augment initial incident data.

If the members were unable to provide their reports, screens, and database reference documentation before the start of the workshop, they were instructed to bring those

items to the workshop for group discussion. During the first workshop, additional reference artifacts were collected and reviewed by team members, including draft FBI N-DEx data element user guides and schemas.

From the collected source material, the facilitators created initial products to be used during the workshop session, including:

- An object-oriented domain model, described with Unified Modeling Language (UML) static structure diagrams illustrating the base objects (complex types) that could be used by all exchange partners.
- Spreadsheet mapping the domain model to GJXDM structures.

Step 2: Workshop Process

During the workshop, the workgroup followed a basic three-step process in developing the Incident Report IEP:

1. Domain modeling.
2. GJXDM mapping.
3. Schema building.

The domain-modeling step was conducted during two face-to-face sessions. The first session was held in Los Angeles County, California on November 15–17, 2004. The second session was held in Ashburn, Virginia on January 25–26, 2005.

During the morning of the first face-to-face session, all workgroup members participated in a collaborative 2-hour session on *Level Setting and Overview of the Process*, which provided a background and framework of the GJXDM and IEP objectives.

Workgroup members agreed to the following principles for the construction of IEP artifacts:

- Following a principle established by earlier workshops, the base object for the schema would represent a law enforcement reference document represented by IncidentReport derived from the GJXDM 3.0.2 *DocumentType*.

- The development of the schema package would use objects and tags derived from or extended appropriately from the GJXDM 3.0.2 model.
- Missing elements or objects (complex types) identified by the workgroup, would be noted and provided to the XML Structure Task Force as potential extensions or modifications to GJXDM.
- The schema package created would be a “reference model” that would enable the sharing of information across justice agencies and not a representation of any specific agency’s system or document/form.
- If a piece of information could be represented in GJXDM 3.0.2 as both a Text Type and Code Type element (i.e. race, sex, hair color, eye color, etc.), the developed schema would include both element type tag names.
- As much as possible, object relationships would follow the hierarchy of GJXDM 3.0.2 inheritance.
- Development of more specific extension or constraint schemas would not be the responsibility of the Incident Report IEP workgroup, but that of any individual agency intending to leverage the reference IEP as a baseline for its local implementation.

The participants then began to review the strawman domain model (see figure 5.2.1, a graphical depiction of the document structure) that had been created by the facilitators from existing paper forms and screen shots. Participants reviewed and revised the model, leveraging GJXDM where appropriate, but without being constrained by GJXDM.

The Incident Report IEP workgroup used [Unified Modeling Language](#) notation simply by drawing the structures on a white board, which were then captured into a UML diagram using [ArgoUML](#) (a freeware tool) at a later time.

Sample strawman domain model



figure 5.2.1

During the discussion, workgroup members navigated GJXDM for hierarchical constructs and to confirm that each element definition met the business requirement of the incident data exchange. Participants used both the [National Center for State Courts’ Wayfarer Tool](#) and the [Global Justice XML Data Dictionary Spreadsheet](#), alternately, displaying the results on an overhead projector to facilitate discussion.

During breaks in the members’ discussion, the facilitators completed iterations of the following tasks:

- Updated the documentation.
- Noted items and ideas that would help future users of the schema in understanding the content of the schema.
- Noted items and ideas that may impact future versions of GJXDM.

In the first face-to-face session, the workgroup explored the current release of the FBI N-DEx schemas, to assess the potential of those schemas to serve as the basis of the Incident Report Reference IEP. The group also settled basic issues of scope, while evolving the domain model.

The Incident Report workgroup required one more face-to-face session to complete its domain model, and met again in Ashburn, Virginia for 2 additional days. In the second face-to-face session, the workgroup segmented and modified a domain model that had been built directly from the structures in the N-DEx schemas. The purpose of the modifications was:

- to incorporate requirements of local incident information exchange.
- to improve consistency with other Reference IEPs.
- to reduce usage of dynamic type substitution and “flat” referencing throughout the structure.

The GJXDM mapping step was completed remotely. The mapping spreadsheet “shell” was prepopulated by extracting information from the domain model segments. The facilitators performed the initial mapping and presented the spreadsheet to the workgroup for review. The completed mapping provided the “want list” to be input into the GTRI [Schema Subset Generation Tool\(SSGT\)](#).

The facilitators then created the schemas; the subset schema was created using the online GTRI SSGT (build 16). The constraint schema was created by transforming the subset schema using a StyleSheet built during previous IEP development projects. (The constraint schema mostly applies more strict cardinality constraints; it also applies choice compositors in a few places.) The extension and document schemas were authored by hand.

The facilitators built a sample instance to illustrate the IEP, as well as to test the integrity of the schemas.

A third face-to-face meeting of a workgroup subset was conducted April 20–21, 2005, at SEARCH headquarters in Sacramento, California. This meeting resolved a small set of remaining technical issues and validated the IEP contents against Incident Reporting schema development efforts in Los Angeles County and Wisconsin.

The artifacts were packaged into a Zip file, and the Incident Reporting GIEPD Overview was written to conclude the project.

Workshop Deliverables

It was the responsibility of the facilitators to create a Zip Folder file that provided the necessary documentation to describe and use the IEP schemas, as well as the domain model and GJXDM mapping spreadsheet. The following artifacts are provided for the [Incident Report IEP](#):

Artifact File Name	Purpose
IncidentReportingGIEPD.doc	The IEP Documentation, in Microsoft Word® format
*.zargo files (9)	UML class diagrams for segments of the domain model, in ArgoUML format
*.png files (9)	UML class diagrams for segments of the domain model, in PNG format
Gjxdm-mapping.xls	Spreadsheet containing mapping of domain model entities to GJXDM
Constraint-schema.xsd	GJXDM-conformant constraint schema
Extension-schema.xsd	GJXDM-conformant extension schema
Document-schema.xsd	GJXDM-conformant document schema
Subset/	Directory containing GJXDM subset package, including a "wantlist" document that can be input into the GTRI subset schema generator
Sample-incident-report.xml	A sample instance valid against the document schema
Constraint-schema-transform.xsl	XSLT StyleSheet that creates the constraint schema from the subset schema (there is a minor amount of hand-editing necessary in the constraint schema)

Additional Business Rules and Workshop Notes

A small set of business rules are associated with the Incident Report Reference IEP. These rules were identified by the Incident Report workgroup in the course of developing the domain model and GJXDM mappings.

It is important to recognize that this set of business rules is not exhaustive. In particular, it does not exhaustively specify the rules that would need to be in place for a particular schema-valid instance to be NIBRS-compliant.

It is recommended that a follow-on project address issues of NIBRS compliance.

- For NIBRS-compliant instances, the PassagePoint structures (incident entry- and exit-points) should only be used for burglaries.
- Usage of PersonAlias (under PersonType) should only go one level “deep” (i.e., PersonAlias elements should not themselves contain PersonAlias elements).
- End dates/times should be later than start dates/times.
- Person relationships should make sense, given the characteristics of the related people (e.g., if Person A is the parent of Person B, and if their ages are known, Person B should be younger than Person A).
- Use @effectiveDate and @expirationDate attributes to place an effective range on an entity itself, not just the information about the entity.
- Incident and Offense are both of IncidentType in GJXDM. Because we only get one chance to define IncidentType, the type must include the union of the properties needed for both Incident and Offense. (This is awkward and should be addressed in a future version of GJXDM.) When building an instance, ensure that elements are being used with the appropriate semantics and within the appropriate containing structure.

In addition, the Incident Report IEP workgroup produced detailed [workshop notes](#) that chronicle the IEP development process and business and technical issues that arose during the process and the disposition thereof. These workshop notes may be of great value to law enforcement information technology staff as they leverage these models for their local implementations.

Development Artifacts

Nonschema artifacts created during the development process are intended to help an implementer better understand the IEP, and could be re-used if the IEP documentation is later modified, extended, or re-purposed. These artifacts include:

- An object-oriented domain model, described with UML static structure diagrams.
- A spreadsheet mapping the domain model to GJXDM structures.
- A “wantlist” document that describes the inputs into the GTRI online subset schema generation tool (this is included with the subset schema package).

In addition, a number of paper and imaged incident report documents were provided to assist the workgroup in developing the domain model. These reports were from the following jurisdictions:

- Los Angeles County Sheriff’s Department.
- New Mexico (Uniform Incident Report).
- West Virginia (Uniform Incident Report).
- FBI N-DEx data element user guides and schemas.

Testing and Conformance

The facilitators tested the integrity of the IEP schemas by parsing a sample instance (included as [sample-incident-report.xml](#)) with the Apache Xerces parser, version 2.6.2. The IEP has not been reviewed for conformance outside of the workgroup.

Feedback

The IEP workgroup will submit definitional changes to some GJXDM elements and types through the regular Bugzilla feedback mechanism required by XSTF. (See appendix 1 for details.)

Post-workshop Process

Following the workshop sessions, the following steps and tasks were completed:

- The facilitators reviewed the notes and support documentation and finalized the first drafts of the workshop deliverables.
- The draft artifacts were distributed to the other participating members of the workshop for their review, via posting on a workgroup Wiki site. (A wiki site is a web page that can be edited by anyone.)
- Comments, corrections, or adjustments identified during the members' review were distributed back to the facilitators.
- The facilitators made the appropriate changes or adjustment, provided by the members' review, to the final draft of the documentation.
- The final workshop draft of the documentation was distributed to BJA and SEARCH for project review, web-based publication, education, and outreach.

Horizontal Analysis

At the conclusion of the four law enforcement-related IEP workshops (Incident Report, Field Interview Report, Booking Report, Charging Document), facilitators from each met for 2 days in Phoenix, Arizona to perform an initial horizontal analysis among the four law enforcement IEPs, as well as the Uniform Rap Sheet IEP developed by the Joint Task Force on Rap Sheet Standardization (see module 005, part 4), for the purpose of:

- Ensuring consistent structures among all law enforcement IEPs.
- Defining core justice data entities to facilitate composition-oriented information sharing using GJXDM.
- Defining high-level aggregate data entities to support the National Information Exchange Model (NIEM).

The horizontal analysis meeting was hosted by the Maricopa County (Arizona) ICJIS project office in its facility on March 3–4, 2005, and attended by the following participants:

Participant	Organization	Workgroup Represented
Scott Came	Justice Integration Solutions	Incident Report
Catherine Plummer	SEARCH	Incident Report
Winfield Wagner	Crossflo	Field Interview Report
Dustin Henson	ARJIS	Field Interview Report
Nancy Rutter	Maricopa County ICJIS	Booking Report
John Ruegg	Los Angeles County ISAB	Charging Document
David Goodwin	Maricopa County ICJIS	Charging Document
Kate Silhol	NLETS–The International Law Enforcement and Public Safety Information Sharing Network	Uniform Rap Sheet

PART 3 Sentencing Order

Introduction, Purpose, and Scope

A GJXDM Reference Exchange Document (now known as Information Exchange Package Documentation or IEPD) workshop was held June 21–24, 2004, in Denver, Colorado, hosted by the Colorado Administrative Office of the Courts (AOC) and sponsored by BJA, with support from SEARCH, OASIS, and the National Center for State Courts (NCSC).

Participants

Workshop participants included representatives of the state AOC agencies in California, Colorado, Missouri, and Washington; Maricopa County (Arizona) ICJIS Project; Los Angeles County (California) ISAB; Pennsylvania Justice Network (JNET) project; and the Office of Justice Programs (OJP), SEARCH, OASIS, and NCSC. The team that met in Denver split up into workgroups to compile several court-related IEPs and documentation, including a **Sentence Order**.

Workshop Methodology

Prior to the workshop, the workshop coordinator collected copies of various forms of sentence orders used around the country, including samples from Maricopa County, Arizona; Cook County, Illinois; California, Missouri, New Mexico, Oregon, Pennsylvania, and Texas; and a U.S. Attorney's office. Colorado provided screen-capture information from its Colorado Integrated CJIS system. Jurisdictional forms were provided to workshop participants on CD-ROM at the beginning of the workshop.

The process that evolved was a bit different than most of the participants had anticipated. On day one, the team worked collaboratively in an attempt to map GJXDM components representing an Order of Protection, using a sample from Pennsylvania. However, the team soon realized that developing an accurate domain data model was a critical first step in the IEP process. The participants from the Pennsylvania JNET project offered to provide an

overview of their domain-modeling process to the other participants.

The Sentence Order workgroup instantly realized the critical importance of subject-matter expertise to support the association of GJXDM components and the use of object inheritance correctly. The use of a domain model, even simply represented with a Microsoft PowerPoint® diagram, proved to be a useful communication strategy for the group to analyze and determine the business requirements of the information exchange, before proceeding with the GJXDM mapping and schema creation.

It was interesting to note that some of the subject-matter experts with no prior exposure to GJXDM, notably the Colorado AOC participants, provided some of the most useful input in the Sentence Order IEP process. Their extensive knowledge of the “business” requirements of sentencing was productively incorporated into the analysis using the domain modeling technique. Colorado's participants (more than likely) would have had difficulty proceeding directly into navigating GJXDM.

On day two, the team split up into workgroups that focused on several court documents. The Sentence Order workgroup began to construct a domain model, representing the various components of the sentence order, then began the process of mapping the objects and their attributes to components of GJXDM. Georgia Tech Research Institute participated as needed via telephone conference and provided its opinions and recommendations.

GJXDM mapping was completed remotely following the workshop. Over the next 6 weeks, the Sentence Order workgroup held several telephone conferences to work through the mapping and the domain model, revising as required to meet clear business requirements. In addition, one of the workgroup facilitators presented the initial domain model and mapping to an NCSC meeting, which resulted in some additional revisions to the initial mapping.

The workgroup experimented with an open-source UML product, ArgoUML, and found it to be very useful in creating a clear domain model, illustrated in figure 5.3.1.

Domain Model Created Using ArgoUML

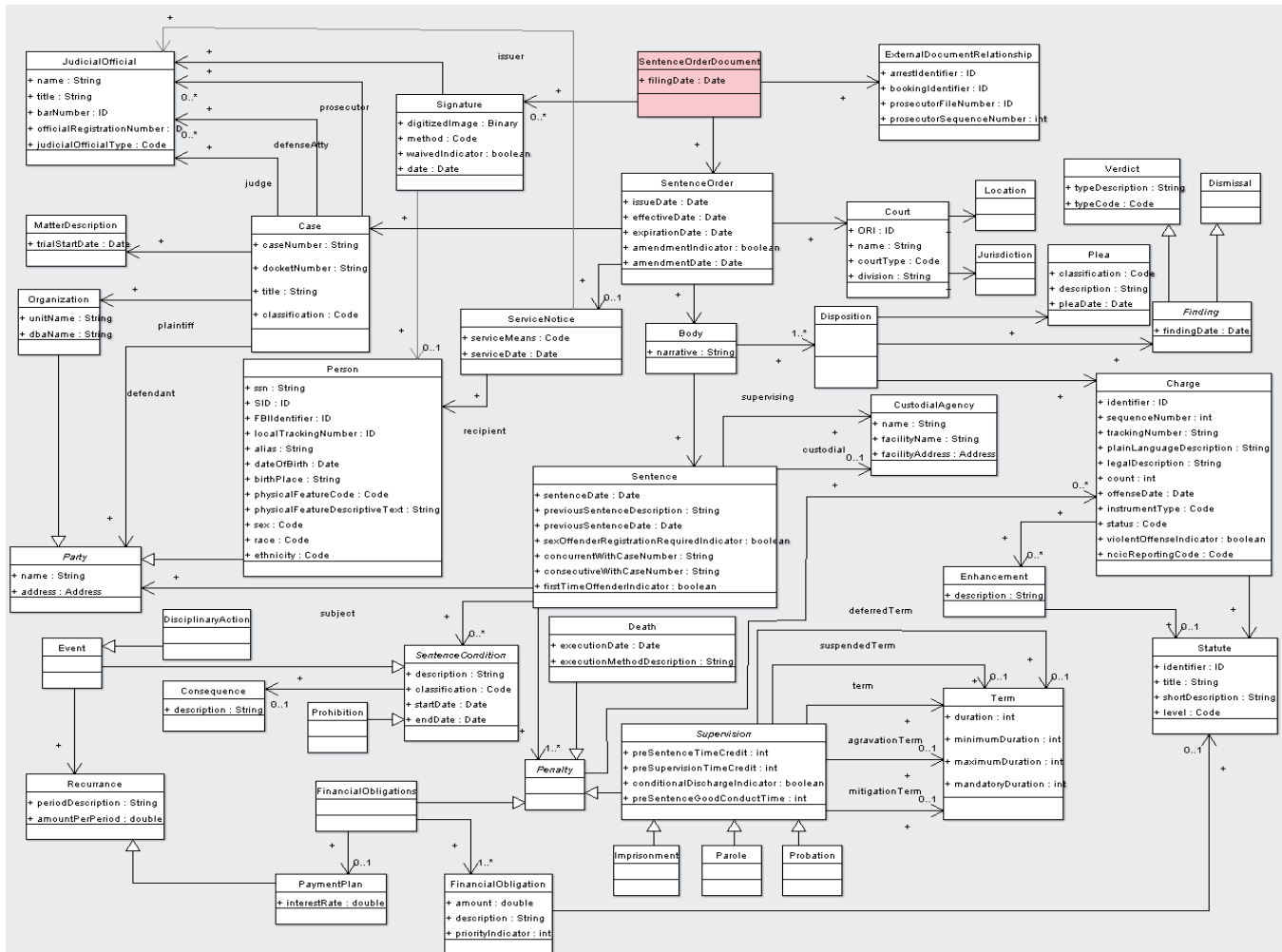


Figure 5.3.1

Workshop Deliverables

The facilitators have created a Zip file providing [Sentence Order IEP schemas and other documentation](#), including the domain model, GJXDM spreadsheet, and detailed workshop notes.

This Sentence Order IEPD is *not* intended to serve as a Reference IEPD, but as a model. A GJXDM IEPD may have the word “reference” in its title if it has been mandated, approved, endorsed, recommended, or acknowledged by a cognizant organization. The Sentence Order IEPD artifacts were provided to NCSC for further vetting, validation, and analysis as a potential Reference IEP.

Workshop Feedback

Sometimes the workshop facilitators receive post-workshop feedback, as evidenced from these comments from a participant:

" . . . I just wanted to say thank you for last week. I feel this was a very successful effort and am glad that I was asked to participate. You did a wonderful job in pulling this together so quickly and keeping us on track. I hope that there will be thought given to continuing this effort to build more Court Reference Documents. There is much work to be done and if we can keep up the momentum, we can add a lot of value to the community at large.

Once again, thank you for the excellent leadership!"

Nancy Rutter

Data Administrator

Maricopa County (AZ) ICJIS

PART 4 Uniform Rap Sheet

Introduction, Purpose, and Scope

The Joint Task Force (JTF) on Rap Sheet Standardization has completed a specification for an XML rap sheet that conforms to GJXDM. Version 3.00 of the rap sheet has been available to vendors and justice practitioners since November 2004.

The work of the JTF is the latest product in an effort that began in 1995 with the National Task Force on Increasing the Utility of the Criminal History Record. That task force included representatives from the FBI; FBI Criminal Justice Information Services Advisory Policy Board (CJIS APB); NLETS—The International Law Enforcement and Public Safety Information Sharing Network; NCSC; and SEARCH.

The target implementation for this specification is the interstate delivery of rap sheets in response to inquiry through the FBI's Interstate Identification Index (III). There are many other potential applications for this specification, but the JTF was acutely aware of the problem caused by interstate exchange where every state has created a different format. When implemented by all the states, this specification will create a uniform product for consumption by 30,000 police departments connected to the FBI and NLETS networks.

Participants

Travel to JTF meetings and hosting of teleconference calls were funded by the FBI, and some appointees were able to pay for their own expenses. Where possible, face-to-face meetings were held in conjunction with the FBI's CJIS Division working group, subcommittee, and CJIS APB meetings. The following individuals are or were on the JTF:

CHAIRMAN:

Gerry Coleman

Wisconsin Department of Justice

EX-CHAIRMAN:

John Loverude

Illinois State Police (former)

Advanced Technology Systems, Inc. (current)

CURRENT MEMBERS:

Steve Correll

NLETS—The International Law Enforcement and Public Safety Information Sharing Network

Owen M. Greenspan

SEARCH, The National Consortium for Justice Information and Statistics

Tom Hopper

CJIS Division, Federal Bureau of Investigation

Mike Lesko

Texas Department of Public Safety

Charlie Pruitt

Arkansas Crime Information Center

Glenda Winn

Maine State Police

PARTICIPANTS:

Dr. Paul Anderson

PNL Associates, LLC

William Casey*

Boston Police Department

David Gavin**

Administration Division, Texas Department of Public Safety

Debra M. Jenkins***

U.S. Marshals Service

Jim Martin+

South Carolina Law Enforcement Division

Robert L. Marx+

SEARCH, The National Consortium for Justice Information and Statistics

Frank Minice

NLETS—The International Law Enforcement and Public Safety Information Sharing Network

Bob Slaski

Advanced Technology Systems, Inc.

Tim Sweeney+

National Law Enforcement Telecommunications System

FBI CJIS SUPPORT:

Gary Barron

James Gerst

Andy Herberger

Dean Manson

Lottie Martin

- * Mr. Casey served on the JTF in his capacity as 1st Vice Chair of the FBI CJIS APB
- ** Mr. Gavin served on the JTF in his capacity as Chair of the APB
- *** Ms. Jenkins served as a representative of the FBI CJIS Federal Working Group
- + Indicates past JTF member

Methodology

The first version of the Interstate Rap Sheet Transmission Standard was developed using the EDI (Electronic Data Interchange) standard that had been adopted by ANSI/NIST (American National Standards Institute/National Institute of Standards and Technology) for the Fingerprint Transmission Standard. By 2000, no state had implemented this version, and the JTF believed that XML was a better standard to follow.

Version 2.0 of the rap sheet was formatted using XML. This product predated GJXDM by a couple of years. It was part of the “reconciliation” effort involving JTF, NCSC, the Regional Information Sharing Systems (RISS) program, and the American Association of Motor Vehicle Administrators (AAMVA). It was one of the foundation specifications that initialized the GJXDM effort. The states of Wisconsin, Kentucky, and Maine, and the FBI have programmed to version 2.2x of the rap

sheet standard. Thousands of these are being transmitted over NLETS every day.

As soon as GJXDM version 3.0 was published, the JTF began the work of creating a rap sheet specification conforming to the Global standard. Version 3.00 of the rap sheet was published in late 2004 and is now in the process of being implemented by other states. The FBI also is modifying its application program to upgrade from 2.2x to 3.00. NLETS has produced a StyleSheet that will convert rap sheets in 2.2x to 3.00.

Initial Data Collection

The first data elements were assembled from the Type 2 Record in the ANSI/NIST/FBI Electronic Fingerprint Transmission Standard (EFTS). This document is how criminal subject and arrest information is submitted to the FBI for storage in the Integrated Automated Fingerprint Identification System (criminal history) database. If EFTS is the input document, then the *rap sheet* is the output document—two sides of the same coin.

The rap sheet, however, evolved into something much more extensive. States held data in their repositories that the FBI did not. There was a need to capture and report biometric data like DNA and photo images. The JTF believed that the FBI’s IAFIS and National Crime Information Center systems needed to be reconciled.

The most significant evolution however, was in how data values were to be represented. The EFTS perpetuated the FBI’s reliance on codes. One of the major problems with rap sheets, however, is that users are not always familiar with codes. The JTF decided that all data values needed to be reported in plain English. Consequently, the *hair color* value BLD is not an allowable value. It has to be reported as “Blonde” or “Bald.”

Final Product

Inevitably, as states began to map actual database fields to the XML elements, there were problems. The JTF was responsive, and

made several adjustments to correct the specification according to real-life problems. Today, with thousands of rap sheets transmitted daily, there are actually very few problems or complaints.

One major area of complaint had nothing to do with XML, but rather with the StyleSheet translation into presentation format. Responses to an III inquiry are sent from one state to another using the NLETS network. NLETS applies a StyleSheet to turn the XML rap sheets into a text presentation format. There were early complaints that the amount of whitespace (intended to improve readability) was more of a problem than a help—it used too much paper when sent to a printer.

The JTF had determined early on that a *presentation format* was not officially part of the standard. Therefore, NLETS has merely adjusted the translation in response to user feedback, and the end product seems today to be acceptable.

Specification Components

The [specification](#) is currently published on OJP's Information Technology Initiatives web site.

A [complete history and previous versions](#) are available at the Wisconsin DOJ Web site.

The specification includes these components:

- Narrative history of the project, process description, and identification of task force participants.
- A technical description of the target implementation: the III and interstate rap sheet transmission over NLETS.
- A text data dictionary.
- A spreadsheet of data elements.
- A full set of GJXDM-conformant schema: rap sheet extensions, subset, constraint, and proxy schema.
- An XML rap sheet instance example.
- A sample StyleSheet that will convert instance documents into presentation text.

- An example of a transformed text rap sheet (presentation format).
- Rules for merging rap sheets from multiple states into a single XML document.
- Bibliography.

Importance of a Rap Sheet Standard

Public policy demands background screening of applicants for positions of trust in and outside of government, and volunteers, especially those who work with our most vulnerable populations—children, the elderly, and the disabled. Legislation enacted to strengthen homeland security in the wake of September 11, 2001, expands the types of positions and activities for which background screening, including a criminal history check, is required. This noncriminal justice purpose suitability evaluation enhances public safety by denying jobs and opportunities to those whose criminal history records suggest a potential to do harm or are otherwise unfit.

The criminal history record is central to the effective functioning of the criminal justice system. Research has shown that as many as two-thirds of all persons arrested for criminal offenses have prior criminal records, often including offenses in multiple jurisdictions or states. At every stage of the criminal justice system, the criminal history record supports decisionmaking. It is used by the police in many ways, including as an investigative tool and to determine a suspect's current status as a probationer, parolee, or bailee. The presence or absence of a prior criminal record is arguably the most relevant information to a judge or magistrate making a pretrial decision on whether and under what conditions to release a person on bail. Prosecutors use criminal history records from the moment they become involved in criminal cases until the cases are terminated at the defendants' parole hearings or earlier. Courts customarily receive criminal history information in modified form, such as in bail reports or presentence reports prepared by probation departments, or in presentations by the prosecutor. Among the uses of the

criminal history record by correctional agencies are inmate classification and making decisions about eligibility for good time credits, early release, work furlough, or release on parole.

At the state level, criminal history records are collected, maintained, and disseminated by “state central repositories.” These agencies or bureaus within state government often are housed within the State Police, a cabinet-level agency such as the Department of Public Safety, or the Attorney General’s Office. Typically, state law requires the repository to establish comprehensive criminal history records and establish rules and regulations for their dissemination to criminal justice and noncriminal justice users. All 50 states, Puerto Rico, and the District of Columbia have established central repositories for criminal history records.

At the federal level, the FBI is the criminal history information repository for both federal and foreign offender information and for records of arrests and dispositions forwarded to the FBI from the state records repositories or, to a much lesser extent, from local law enforcement agencies.

A uniform criminal history record format has never been made mandatory. Likewise, no mandatory guidelines regarding the content of criminal history records have ever been promulgated. State and federal repositories have been left to adopt their own record formats and approaches concerning the types of offenses that should be included on criminal history records and the types of information that should be included. Not surprisingly, this has resulted in considerable diversity in the content and formats of the criminal history records presently generated by the state repositories and the FBI, often leading to difficulty in interpreting the information provided. This confusion is frequently heightened when the information user is from a state other than that which provided the information. Similarly, noncriminal justice users often lack the knowledge and experience

to competently interpret the differences in details and layout among the many pieces of information that may surface from a criminal history records check. National adoption of this voluntary Interstate Criminal History Transmission Specification and its associated presentation format resolves many of the difficulties that hamper the exchange and interpretation of criminal history records.

MODULE 006

Global Justice XML Data Model Funding and Conformance

**PART 1 Funding Requirements:
U.S. Department of Justice and U.S.
Department of Homeland Security
Special Conditions**

In FY 2005, the U.S. Department of Justice's (DOJ) Office of Justice Programs (OJP) announced that a [new special condition](#) has been adopted, requiring any OJP grant application with the potential of using XML to use the Global Justice XML Data Model (GJXDM) and to publish the schema to the central OJP repository.

Register Schemas in the Justice Standards Clearinghouse

All recipients of OJP grants for projects implementing XML technology are required to use the GJXDM Specification and Implementation Guidelines. OJP XML grant recipients also must publish all XML schemas resulting from use of the model in the [Justice Standards Clearinghouse](#) (JSC). This requirement is stipulated as a Special Condition to their grant under the title of "Support Public Safety and Justice Information Sharing."

Other organizations not funded by OJP that use GJXDM are encouraged to publish their XML schemas (such as extension, constraint, or proxy schemas) to JSC in order to facilitate potential interoperability of information systems that will enhance the potential for sharing of justice and public safety information.

By supporting the GJXDM standard with real funding conditions, DOJ effectively put "teeth" behind GJXDM. Just before Christmas 2004, the U.S. Department of Homeland Security (DHS) announced that it was mirroring the same special conditions language adopted by OJP. For example, the FY 2005 [Homeland Security](#)

[Grant Program: Urban Areas Security](#)

[Initiative Request For Applications](#) states, as a requirement, under Section C.1.a - Extensible Markup Language (XML) Requirements (pp 14-15), "...To support homeland security, public safety, and justice information sharing, ODP [Office of Domestic Preparedness] requires all grantees to use the Global Justice Data Model specifications and guidelines regarding the use of XML for all HSGP awards. The grantee shall make available without restriction all schemas (extensions, constraint, proxy) generated as a result of this grant, as specified in the guidelines."

GJXDM is intended to be a data reference model for the exchange of information within the homeland security, justice, and public safety communities. GJXDM is a product of DOJ's Global Justice Information Sharing Initiative's Infrastructure/Standards Working Group (GISWG). It was developed by GISWG's XML Structure Task Force (XSTF).

The GJXDM specifications and guidelines include the use of XML to support the exchange of information within the homeland security, public safety, and justice communities.

Most major software vendors fully support the general XML standard, and major database vendors and their database applications provide software development "tools" to assist homeland security technical staff to develop and use XML more efficiently and productively within agency applications. The general XML standard is designed to be independent of vendor, operating system, source application, destination application, storage medium (database), and/or transport protocol.

In addition, the use of XML allows homeland security personnel to share vital information, which no longer entails purchasing new systems or compromising their business practices. XML allows systems already in use and those being developed to communicate with each other and paves the way for future expanded collaboration between agencies.

Further information about the required use of XML and GJXDM specifications and guidelines is available at [OJP's Information Technology \(IT\) Initiatives web site](#).

PART 2 Global Justice XML Data Model Conformance Guidelines

According to the OJP web site, “. . . the goal of *GJXDM conformance* is for the sender and receiver of information to share a common, unambiguous understanding of the meaning of that information. Conformance to GJXDM ensures that a basic core set of information (the GJXDM components) is well understood by the community and carries the same meaning. The result is some level of interoperability that would be unachievable with the proliferation of custom schemas and dictionaries.”

The “informal rules” for GJXDM conformance are provided as follows:

1. Instances must validate against GJXDM reference schema. Schemas conformant to GJXDM must import and reference the [GJXDM Schema namespace](#) or a correct GJXDM Schema Subset (which is the same namespace).
2. If the appropriate component (type, element, or attribute) required for the application exists in the GJXDM, use that component, or, in other words, do not create a duplicate of one that already exists.
3. Be semantically consistent. Use GJXDM components in accordance with their definitions. Do not use a GJXDM element to represent data other than what its definition describes.

Schema Subsets

The Global Justice XML Data Dictionary (GJXDD) has grown to accommodate the needs of a large and varying user base. Though a large dictionary in itself is not a problem, users can experience difficulties when using the large XML schema generated from the full dictionary. In most practical use cases, only a subset of the full GJXDD is required.

Likewise, it is possible to validate with a reduced set (a subset) of the GJXDM components. An online tool that can

automatically generate a correct Schema Subset has been developed to help developers with this process. The [Schema Subset Generation Tool](#) (SSGT) provides developers with the ability to create GJXDM Schema Subsets based on the [Rules for Schema Subsets](#).

Reference Architecture

To understand how to use a GJXDM Schema Subset for validation, it is essential to understand the data model’s *reference architecture*.

In general practice, an XML instance references an XML schema, which in turn references the World Wide Web Consortium (W3C) XML Schema specification (by namespace). The instance is valid if it conforms to its respective schema definition, which then conforms to the W3C XML Schema specification. The GJXDM schema, rendered from GJXDM, introduces a set of types, elements, and attributes as predefined building blocks for use in justice schemas. Within the GJXDM Schema, these components are optional, over-inclusive, and unconstrained. However, for practical use in validating real instances, the correct components must be identified and constrained, as necessary. This process requires validation against two slightly different schemas:

1. The full GJXDM Schema or a correct subset.
2. A constraint schema that reflects the subset, but with applied constraints.

The two schemas represent two distinctly different validation paths. The first validates for conformance to GJXDM, and the second validates for conformance against the user’s required constraints.

In review, instead of:

- an instance,
- the user schema, and
- the W3C schema specification.

The result is:

- an instance,
- the user schema,
- an optional user extension schema,
- the full GJXDM Schema or a GJXDM Schema Subset,
- a corresponding constraint schema, and
- the W3C XML Schema specification.

Component Extension

There are several ways that a local schema might extend GJXDM. A simple set of examples has been prepared that illustrates various [extension methods](#). One of these methods is based on the W3C rules for [extension of XML Schema types](#). W3C Schema rules for type extension allow many possibilities. However, type extension within the GJXDM is intended to maintain a class hierarchy of objects by adhering to a more restrictive set of subclass rules.

To ensure the integrity, consistency, and meaning of the GJXDM class (inheritance) hierarchy, the following rules for type extension must be followed:

1. A derived type may add (by extension) additional fields (elements/attributes) to its base type.
2. A derived type may restrict one or more fields of its base type, but only so that a derived field is a subset of the field of the base type. For example, a derived type may:
 - Restrict an enumeration from a large set of options to a smaller set of options, as long as every option in the derived set appears in the base set.
 - Remove a field of the base type only if the field is optional in the base type.
 - Require a field to appear only if the field is optional or required to appear in the base type.
3. A derived type may not modify a field of its base type such that it violates the

constraints of its base type. For example, a derived type may not:

- Add additional enumerations to a field.
- Remove a field that is required by its base type.
- Modify the type of a field of its base type.

Additional Remarks About Conformance

- *Information exchanges* conform to the GJXDM; systems do not. The way data are labeled or used in one system does not impact GJXDM conformance.
- Conformance relies on how data are packaged as XML for an information exchange.
- Use of some components of GJXDM to exchange information with other justice agencies does not guarantee conformance to GJXDM. Users should be careful to avoid violating conformance Rule 2, listed above. An information exchange either conforms to GJXDM or it does not.
- GJXDM conformance rules are, by design, nonresident. More formally specified rules would be counterproductive to the development of the evolving GJXDM.

PART 3 Global Justice XML Data Model Conformance Guide for the Practitioner

In FY 2005, the DOJ's Office of Justice Programs announced that a [new special grant condition](#) had been adopted, requiring any OJP grant application with the potential of using XML to use GJXDM and to publish the schema to the central OJP repository. Since the initial announcement, the same special condition language has also been adopted by DOJ as a whole and the U.S. DHS. The underlying requirement of the condition is that data exchange among grant recipients must be constructed to *conform* to GJXDM.

According to DOJ, "Conformance to GJXDM ensures that a basic core set of information (the GJXDM components) is well understood by the community and carries the same meaning. The result is some *level of interoperability* that would be unachievable with the proliferation of custom schemas and dictionaries."

Most justice practitioners, including law enforcement officers, prosecutors, judges and court personnel, and IT professionals, clearly understand the nation's need to share information among justice and public safety agencies at all levels of government. Many embrace the notion of interoperability for both voice and data exchange and understand the utility of XML as a universal translator. The reality is, however, that building XML schemas (data exchange instructions) that conform to GJXDM has proven to be a challenge for many of those who have attempted this work.

During 2004, those engaged in the process of building GJXDM exchange content discovered that the important part about creating GJXDM-conformant exchanges *precedes mapping*. Modeling, using domain subject-matter experts, allows the jurisdiction to define how the various objects get associated with correct business context. This enables users to map correctly. Whether using a contractor or using in-house staff, users should be sure that each exchange

has been modeled to reflect their agency's business process. If users start from the point of a reference Information Exchange Package (IEP), this step has already been accomplished so that users can leverage this work and extend it without starting from scratch.

But if users must start from scratch, they should not skip the domain modeling piece! (For more information, see module 004, part 3.)

Since the operational release of GJXDM in 2004, many companies now proclaim that their products are completely GJXDM-conformant. Several even claim that their products can magically transform exchanges into GJXDM, or produce GJXDM-conformant schemas— "plug-n-play" solutions, as it were. The reality is, however, that building GJXDM-conformant schemas is just not that easy. Several members of XSTF have reported reviewing work product this past year that isn't GJXDM-conformant at all, when it was purported to be. How can a practitioner ensure that the deliverables provided are really what they are supposed to be?

Here are a few points to consider when defining project requirements or beginning the procurement or contract negotiation process:

1. **There is no magic GJXDM bullet!**

XSTF members have spent many months working with some very knowledgeable experts from both the private and the public sectors to build reference GJXDM exchange documents, and have not found a way to simply automate this process using any tool or product.

2. Reference IEPs and documentation developed by BJA, with support from SEARCH, the Integrated Justice Information Systems (IJIS) Institute, and other justice partners provide good baseline models for information exchange, and have been developed collaboratively by public-sector subject-matter experts and technical developers. Look for a

complete posting of these on the OJP web site. The initial set of IEPs were developed with open source, nonproprietary tools. The publication of ubiquitous justice IEPs provides, for the first time, tangible models and GJXDM content that can be leveraged by justice agencies, whether large or small, urban or rural, local, county, state, tribal, or federal, to begin on the path of data interoperability throughout the nation.

3. Ask for implementation-specific references.

4. Ask a potential contractor, or define in a request for proposals or information (RFP/RFI), at least some of the following questions:

- What is your experience in criminal justice applications or integrated justice systems?
- Does your company use and provide open standards solutions for data exchange? How? Can your product/application be leveraged by other justice agencies, rather than just provide a proprietary solution for one group of clients?
- Are you a member of—
 - ☐ The IJIS Institute?
 - ☐ The Organization for the Advancement of Structured Information Standards (OASIS)?
 - o On which OASIS Technical Committees do you participate?
 - ☐ Integrated Justice
 - ☐ Court Filing
 - ☐ Biometrics
 - ☐ ebXML
 - ☐ Emergency Management
 - ☐ Other
 - o Describe any work product contributed toward OASIS standards.
- What is your experience using XML for data exchange? Provide details about specific implementations.
- Provide an example of data exchange points and all dimensions thereof.
- What products/implementations have you developed for *justice data exchange/information sharing*?
 - ☐ Using XML?
 - ☐ Using the GJXDM 3.0.x standard?
- Explain how your model supports interface/data requirements beyond the capabilities of the XML definitions. For example, elements may be appropriate in certain exchanges but not others. Do you support the use of the *Justice Information Exchange Model* for capturing and documenting this information?
- Where do you get your information on Justice XML and the GJXDM? How do you keep it updated?
- Has your company provided, or does it plan to provide, any work product that can be leveraged as part of a *Reference Information Exchange Package and Documentation*? Please describe specifically.
- Please provide contacts/references that have knowledge of your work in justice applications and data exchange.

It is the public sector's obligation, and the justice practitioner's standard of care, to ensure that agency exchange content, whether developed in-house or with vendor support, is truly GJXDM-conformant and can be consumed and understood by all exchange partners. This is an important objective that deserves support.

APPENDIX 1

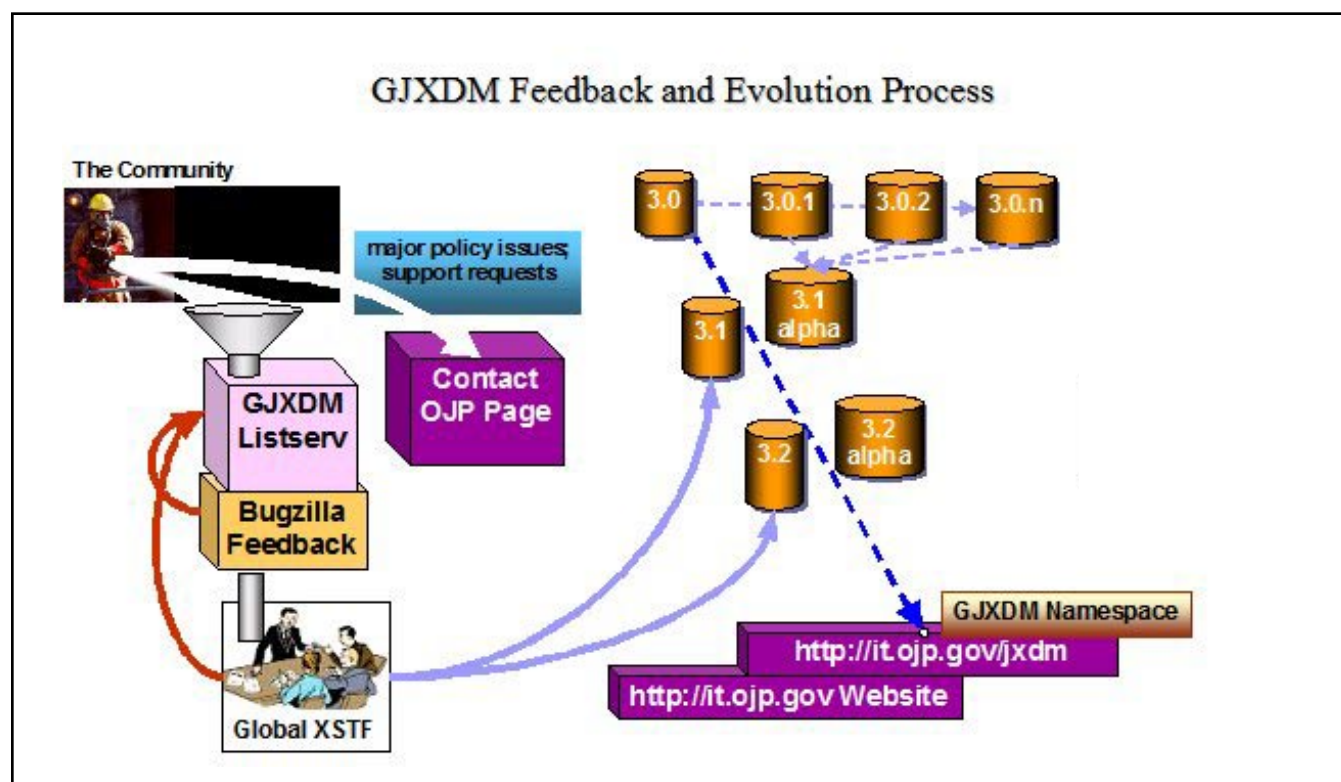
Global Justice XML Data Model Feedback, Release Process, and Documentation

Data Model Feedback, Update, Maintenance, and Evolution

The Global Justice XML Data Model (GJXDM) is sponsored by the Office of Justice Programs (OJP) U.S. Department of Justice (DOJ), with development supported by the Global XML Structure Task Force (XSTF), which works closely with Georgia Tech Research Institute (GTRI). New releases are issued by XSTF, which reviews and evaluates each version of GJXDM.

Since the release of GJXDM in January 2004, XSTF has continued to conduct intensive technical and domain content review, receive

feedback and error reports online from the public, and provide a discussion forum for sharing expertise and support. Subsequent software releases incorporate modifications and improvements to GJXDM based on user feedback and XSTF analysis. The task force generally states that GJXDM will never be “done,” but is a constantly evolving work in progress, particularly as it adds additional content and new domain material, such as juvenile justice components. Figure A.1.1 illustrates the GJXDM feedback and evolution process.



Peer support can be obtained via the [GJXDM Listserv](#), a community of interest for developers, practitioners, and others to promote the exchange of ideas and experiences associated with GJXDM. This listserv exists to support communitywide announcements, comments, requests, questions, discussions, and exchanges that are relevant to GJXDM. Technical questions emanating from the listserv, which have not produced a conclusive answer, may advance to GTRI's Bugzilla Feedback web site for review and analysis by XSTF and GTRI.

The [GJXDM Bugzilla Feedback web site](#) will continue to be the primary means to identify urgent bug fixes and requests for additions, deletions, and modifications. Users can register for Bugzilla access on the web site and will be provided an opportunity to create a login and password. The Bugzilla software, maintained by GTRI, is intuitive and allows the user to easily log a bug report, which may simply be a request for a definition change.

All Bugzilla feedback is reviewed and approved/disapproved by XSTF, which meets at least biweekly via telephone conference or in person to discuss all outstanding bug reports. XSTF authorizes all fixes and changes to GJXDM. All approved additions, deletions, and modifications will be applied to the next release, if timing permits. A cumulative change log is published with each release to maintain an audit trail. Changes will be linked to appropriate bug reports in the feedback history.

Operational Releases

All approved additions, deletions, and modifications are applied to future releases, with a cumulative change log published along with each release. When a reasonable number of updates are approved by BJA's XSTF, a new version is released. This will involve an intermediate decision to determine if this should be a forward-compatible series update or the start of a new series.

XSTF plans to publish new GJXDM releases periodically. There is no specified life cycle for releases. As explained above, the [GJXDM Feedback](#) site provides a mechanism for making suggestions, submitting requests for new or changes to current components, and identifying errors or bugs in the design. The GJXDM release process is illustrated in figure A.1.2 on page 133.

Releases and their technical documentation are accessible by **namespace**.

Operational releases of the GJXDM, from 3.0 forward, will always be available (i.e., will never be removed from their namespaces) on the [OJP web site at the GJXDM namespace](#). This ensures that legacy schemas and instances written to a noncurrent release will *always have the original reference with which to validate*.

To keep abreast of the latest information regarding the timing of new releases, monitor the official [OJP web site](#) and/or register on the [GJXDM Listserv](#).

Data Model Version Numbering

GJXDM versions are numbered by three integers (X, Y, and Z) delimited by dots: Syntax: X.Y.Z

Each integer represents a particular class of change:

- X = Major revisions to the model or representations of the model as rendered in a schema (as XML or other markup).
- Y = Minor changes that do not maintain forward-compatibility.
- Z = Minor changes that maintain forward-compatibility.

A version number should not be confused with decimal numbers. X, Y, and Z are *integers*, not *digits*, and may become greater than 9. For example, "4.89.113" would be a legitimate release number, although unlikely to ever be

used. GJXDM version 3.0 is the first *operational* release. (Note: It is also referred to as version 3.0.0). Since the first operational release on January 15, 2004, two additional releases have been published (3.0.1 and 3.0.2). Version 3.0 instances will be forward-compatible with, or will validate using the version 3.0.2 schema or appropriate 3.0.2 schema subsets.

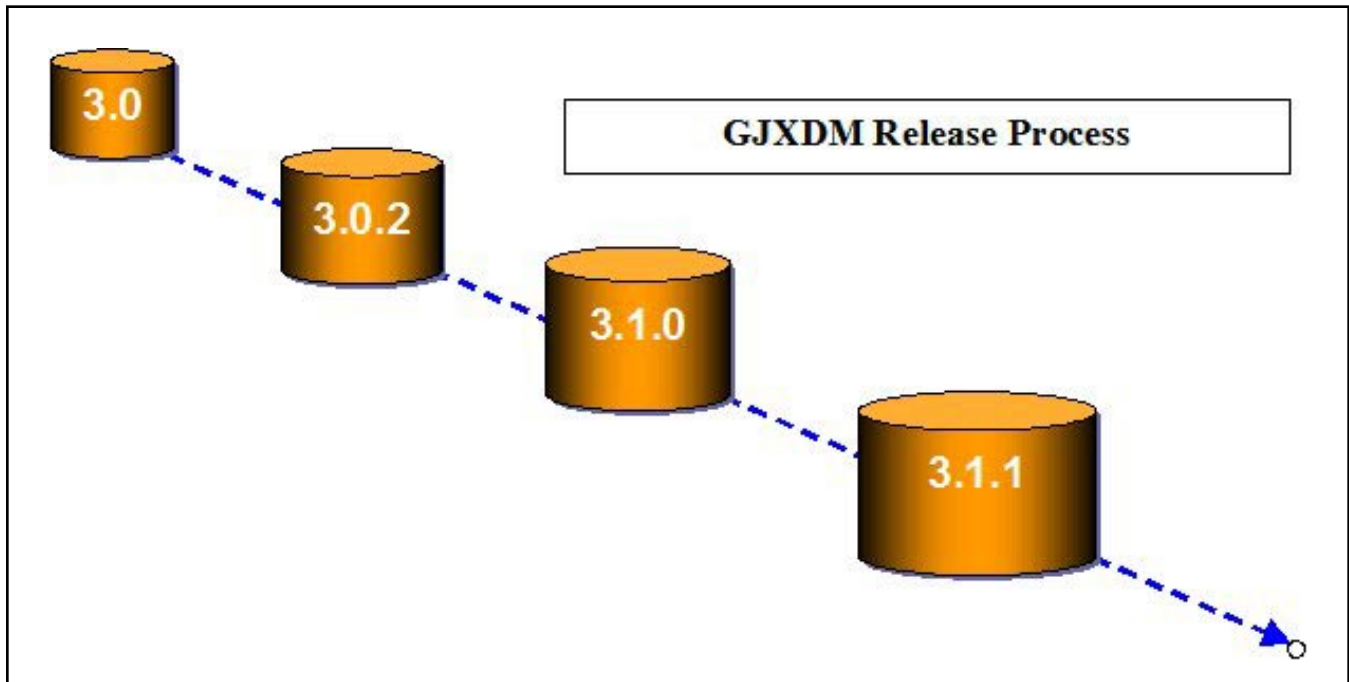


figure A.1.2

Compatibility

The following sequence describes compatibility within the GJXDM 3.0 release series:

3.0(.0) instances validate with 3.0.1, 3.0.2, ..., 3.0.n schemas.

3.0.1 instances validate with 3.0.2, 3.0.3, ..., 3.0.n schemas.

3.0.2 instances validate with 3.0.3, 3.0.4, ..., 3.0.n schemas, and so forth.

Within the 3.0 series, "forward-compatibility is maintained." However, when 3.1.0 (a new series) is released, 3.0.n instances are *not* guaranteed to validate against 3.1.x schemas. However, 3.1.0 instances will validate with 3.1.1, 3.1.2, ..., 3.1.n schemas, etc. (if/when they are released). In other words, within the 3.1 series, forward-compatibility will be maintained.

CAVEATS:

1. To validate a 3.0.0 instance against the 3.0.2 GJXDM schema (when available), the namespace reference in the instance must be changed to 3.0.2 (and in the schema as well if one between the instance and the GJXDM schema is being used).
2. Depending on which GJXDM components are used, it is possible that a 3.0.0 instance validates against the 3.1.0 GJXDM schema (when available). DO NOT assume that every 3.0.0 instance will also validate. No guarantees of validation will exist between 3.0 and 3.1.

Global Justice XML Data Model Version 3.0.2

Some of the enhancements incorporated into version 3.0.2¹¹ include the most current code values for schemas for the NCIC 2000 code tables, definition adjustments and corrections, the addition of more properties and the deprecation of others, and an updated Frequently Asked Questions (FAQs) resource that incorporates content from the GJXDM Version 3.0 Reference Notes along with added FAQs.

In addition, elements can now have “null” values (i.e., no data content between the start and end tag), and the GTRI [Schema Subset Generation Tool](#) is now capable of supporting multiple versions of GJXDM.

An important developmental highlight is that GJXDM version 3.0.2 is *forward-compatible* with all XML instances written for GJXDM version 3.0. Version 3.0 instances validate with all 3.0.2 schemas and schema subsets. Version 3.0 of GJXDM will remain available indefinitely.

Prereleases

GJXDM prereleases used four numbers to designate versions. There were only four prereleases: 3.0.0.0, 3.0.0.1, 3.0.0.2, and 3.0.0.3, which will no longer be updated or supported. However, for transitional reasons, three prereleases—3.0.0.0, 3.0.0.1, and 3.0.0.3—will remain available in their respective namespaces for an undetermined period of time after publication of the initial operational release (which occurred January 2004).

GJXDM prereleases were never meant for production systems. They are equivalent to *alpha* releases of software, and as such, received only minimal review and testing. These prereleases were published only to provide an opportunity for the community to review, test, comment, suggest, adjust, and correct.

Documentation

GTRI maintains current documentation on each GJXDM release, including:

- The [Documentation Spreadsheet](#) of all types and properties.
- A [Change Log](#), which provides a list of all changes since the last release.
- GJXDM Schemas, in both an abbreviated and a fully annotated version.
- GJXDM Schema Archives, which provide a complete set of schemas, rolled up in an archive, including all external code table schemas.

Additional supporting documentation includes:

- [FAQs](#): A list of frequently asked questions regarding GJXDM.
- Reference Notes: Notes on the current status of the data model, including information on past and future versions; data model software tools, including the search tool and subschema generator; and descriptions of other capabilities and support for GJXDM. (Note: Reference Notes have been merged with the FAQ file.)

The most up-to-date documentation will always be available on the official OJP [GJXDM web site](#) (See figure A.1.3 on page 135). Check this site frequently!

¹¹ Current as of March 2005.

GJXDM Web Site

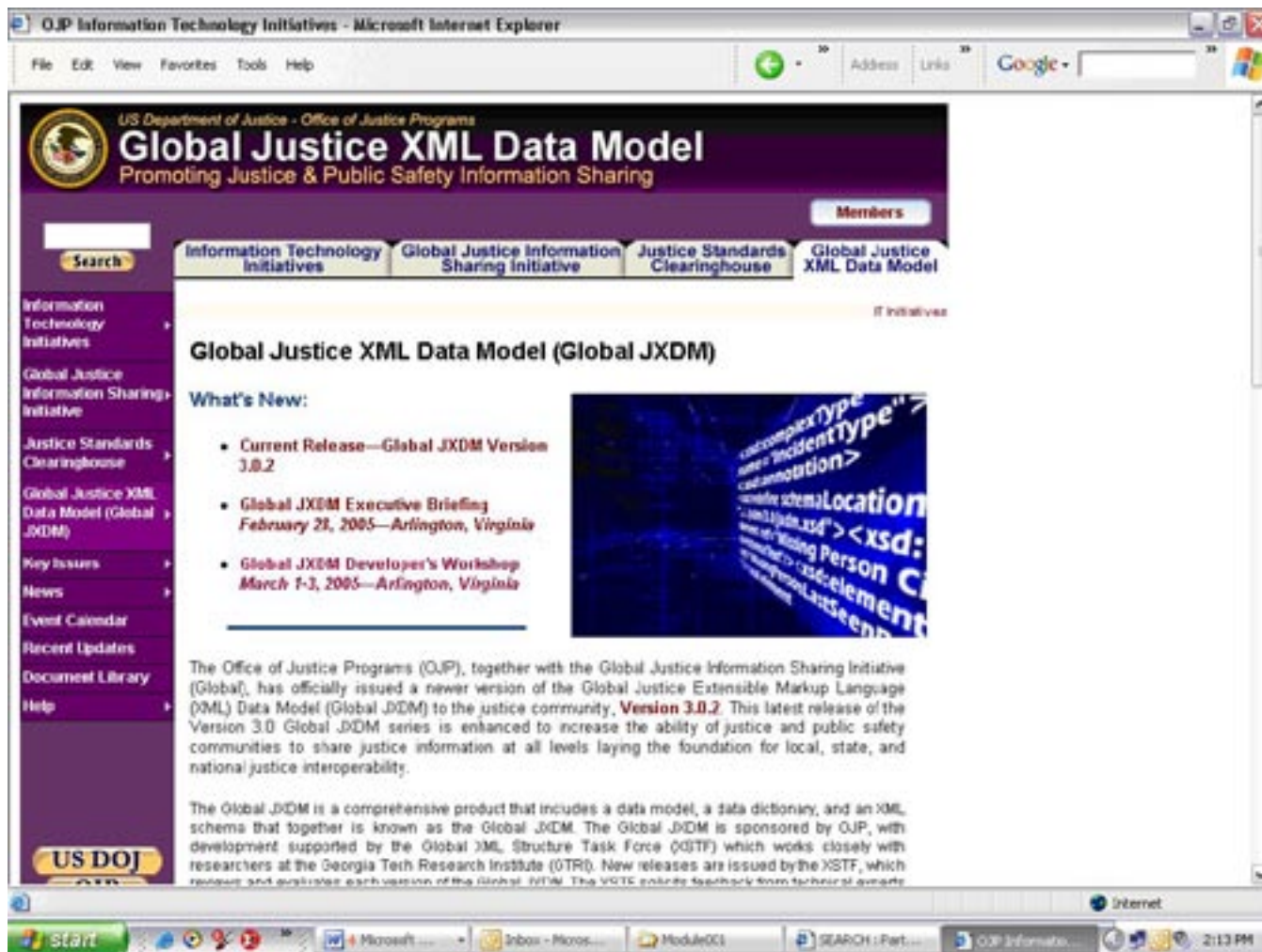


figure A.2.1

The GJXDD spreadsheet provides all of the element names, which are organized hierarchically under the core components (Person, Property, Organization, etc.) with **hyperlinks** to related elements.

The spreadsheet also provides information as to the type of data being represented (date, integer, Boolean, string, etc.) and a precise, context-rich definition of each dictionary component. The definitions were developed and refined over many meetings and conference calls among members of the U.S. Department of Justice's (DOJ) XML Structure Task Force (XSTF) and Georgia Tech Research Institute (GTRI) and represent a commitment to provide reusable components that represent the same meaning for all of us.

Global Justice XML Data Model Wayfarer

Staff with the National Center for State Courts (NCSC) in Williamsburg, Virginia wrote a program called the [Global Justice XML Data Model \(GJXDM\) Wayfarer](#). This tool is available

at no charge on NCSC's web site. The GJXDM Wayfarer tool provides:

- Basic information about elements and types.
- A tree-based overview of the entire data model.
- Searchable functionality, in a number of different ways:
 - Names.
 - Descriptions.
 - Contextual names and descriptions.
- A full list of inherited properties for any element.
- A full list of elements that inherit an element as a property.
- A full list of elements that are "sort of" like a particular type, for example, "person-ish" elements.
- Code tables, including values.

Figure A.2.2 shows a screenshot of the Wayfarer Tool.

GJXDM Wayfarer Tool Web Site

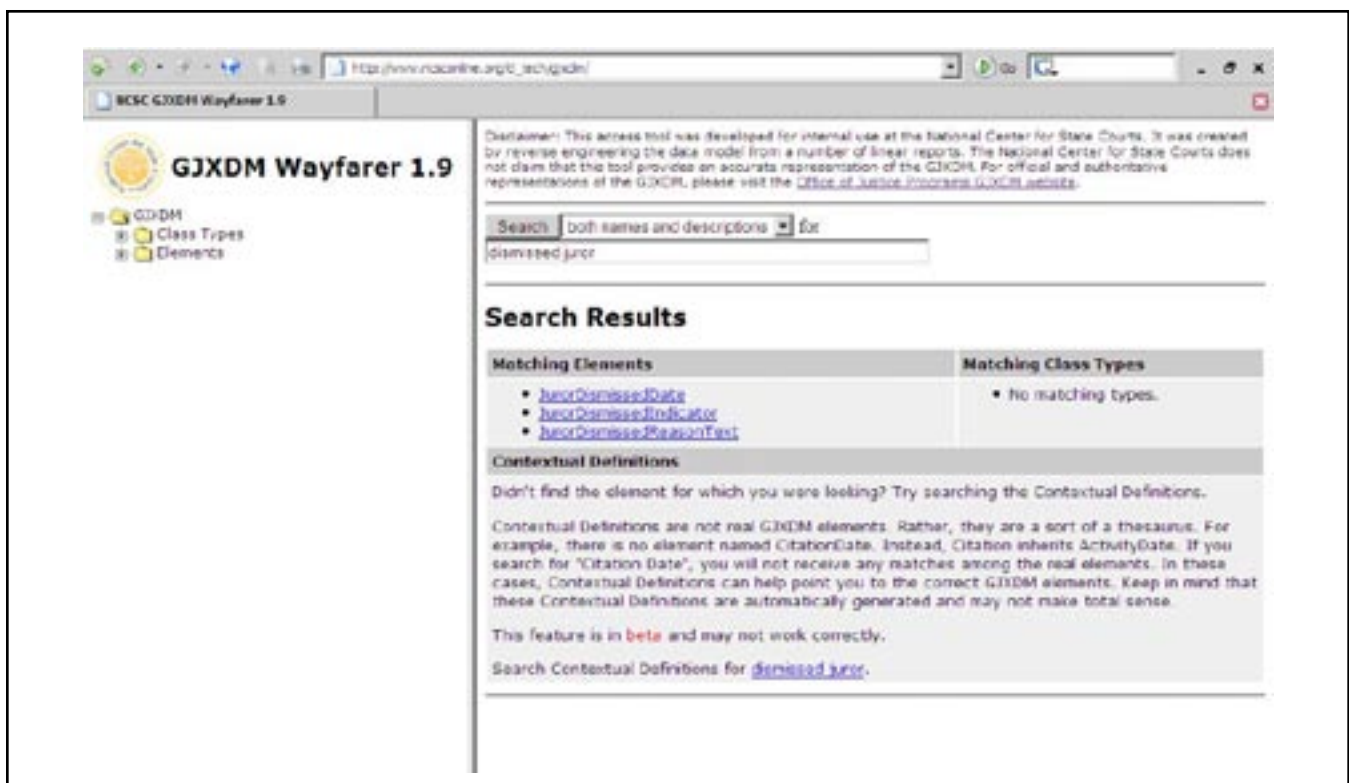


figure A.2.2

There are two versions of Wayfarer available: an online version that requires Internet access and a JavaScript-enabled browser, and a localized version in a Zip file that requires a JavaScript-enabled browser and 30 megabytes of hard disk space.

Wayfarer has been updated to GJXDM 3.0.2, and the localized version is available through the NCSC web site (download [Wayfarerlocalized.zip](#)). Store the Wayfarer Zip file in a separate directory. Then unzip the file in that directory, which will create a gjxdm directory in a local folder. Finally, point the browser at the **index.html** file in a local gjxdm directory.

Please provide questions, comments, or enhancement ideas to Tom Carlson, NCSC, at tcarlson@ncsc.dni.us.

Schema Subset Generation Tool

The Schema Subset Generation Tool (SSGT) enables users to search and navigate GJXDM, find components required for this application, select those components with or without their subcomponents, and generate a correct GJXDM schema subset for validation of XML instances.

SSGT is in spiral development and will change frequently. All users, but particularly first-time users, are encouraged to review the current information about SSGT online on the [SSGT Information Page](#).

Wantlist Specification for the Schema Subset Generation Tool

The Load/Save function in SSGT uses a Wantlist to preserve the details of a schema subset. The Wantlist format is XML. The current Wantlist schema specification is located at [Wantlist Schema](#). (Note: This schema will change once capabilities are added to SSGT, such as global constraints.)

XML Validation Tool

GTRI has assembled a Java application for validating XML instances. It uses the Apache Xerces Java XML Parser library, which has a highly standards-conformant validator. This application may perform more efficient validating than some commercial tools, and it is relatively fast.

The latest full release of this validation tool (version 1.1) was posted on January 30, 2004. To use the tool, download and extract the file at [validateXML-1.1.zip](#) (953k).

This Zip file contains the source files, supporting documentation, and a binary Java Archive (JAR) file for executing the validator. Once the Zip file is extracted, look at the contained index.html for a description of how it works and for relevant links.

XML Validator 1.2 Beta 4

A **beta release of the 1.2 version of the validator**, 1.2b4, is available. Note: The beta version's documentation has not been synchronized with the code. The syntax for this version does *NOT* use "-in" for the input file, as version 1.1 did.

The program seems to work well, but has not been heavily tested. It features a command-line option for assigning schema locations and can process multiple instance files with a single invocation. Users can download and extract the program at [validateXML-1.2b4.zip](#) (1041K).

GTRI would appreciate receiving feedback, error reports, and recommendations regarding the validator. Please use the mechanisms provided in appendix 1 to provide feedback and comments.

Scripts to Strip Documentation

To **strip xsd:annotations** from the GJXDM release version 3.0.1 distribution, users can download the following package for use in DOS or Unix: [schema-tools-3.0.1-1.0.zip](#) (4k).

Global Justice XML Data Model Class Hierarchy

The entire [GJXDM Class Hierarchy for version 3.0](#) is a downloadable spreadsheet. This reference is a simple visual representation of all complex type derivations and simple types in tree format. (It is not a Unified Modeling Language (UML) diagram.)

Subsequent releases will have the class hierarchy in a tab of the GJXDM spreadsheet for that version.

SEARCH Justice Information Exchange Model© Modeling Tool

The [SEARCH JIEM project](#), funded by the Bureau of Justice Assistance (BJA), DOJ, was designed to facilitate the planning and implementation of justice information sharing throughout the nation.

Among other things, BJA, with support from SEARCH, has developed a conceptual framework to present the flow of justice information among criminal justice agencies; define the key events that trigger the need to share information; identify the agencies involved in the exchange; and describe the nature of the information exchange.

Project staff have developed a web-based modeling tool and a methodology to capture detailed information regarding the processes, events, agencies, information, and conditions associated with justice information sharing. In January 2004, BJA, with support from SEARCH, released version 3.0 of the JIEM Modeling Tool, which provides enhanced functionality and outputs.

Version 3.0 incorporates the [JIEM Reference Model](#) version 1.0.1, which provides jurisdictions with 663 reference exchanges developed from 5 years of research conducted in nearly two dozen jurisdictions. In addition, the JIEM Modeling Tool interfaces with GJXDD version 3.0. The Tool has been used by more

than 70 jurisdictions around the nation and in Canada to help them define their requirements for information exchange.

Unified Modeling Language Tools

Today, there are a substantial number of both commercial and open-source tools for enabling model-driven development. Microsoft's **Visio**® (available through the Microsoft Office suite) has recently been used to support domain modeling by several members of GJXDM Information Exchange Package teams.

One of the most popular open source tools for UML design is [ArgoUML](#). This is a full graphical user interface desktop application that allows creation of virtually the entire set of UML diagrams, with the ability to save those diagrams in the XML Metadata Interchange (XMI) format, which many modeling tools use as their source for code generation. Users can go to the ArgoUML web site for all file releases and resources.

XML Resources

XML Websites

- [World Wide Web Consortium \(W3C\)](#)
- [XML.com](#)
- [XML Cover Pages](#)
- [Organization for Advancement of Structured Information Standards \(OASIS\)](#)

Introductions to XML

The following list of introductory and tutorial articles on XML is extracted from the complete chronological listing of articles collected for the XML Cover Pages.

- ["Extending Your Markup: An XML Tutorial."](#) By André Bergholz.
- ["Media-Independent Publishing: Four Myths About XML. \[The W3C's Working Group Chair Dispels Some Myths About XML.\]"](#) By Jon Bosak.

-
- ["XML, Java, and the future of the Web."](#) By Jon Bosak.
 - ["XML - Questions & Answers."](#) By Jon Bosak.
 - ["Microsoft's Vision for XML."](#) By Adam Bosworth.
 - [\[Tutorial Introduction\]: "Declaring Elements and Attributes in an XML DTD."](#) By Ronald Bourret.
 - ["Beyond HTML: XML and Automated Web Processing."](#) By Tim Bray
 - ["SGML and Meta-information: From SGML DTDs to XML-DATA."](#) By François Chahuneau.
 - ["The Evolution of Web Documents: The Ascent of XML."](#) By Dan Connolly, Rohit Khare, and Adam Rifkin.
 - ["The Extensible Markup Language \(XML\)."](#) ETHOS [European Telematics Horizontal Observatory] Technology Briefings Series 1.
 - ["XML: Document and Information Management."](#) By Todd Freter.
 - ["Introduction to XML."](#) By Lars Marius Garshol.
 - ["Capturing the State of Distributed Systems with XML."](#) By Rohit Khare and Adam Rifkin.
 - ["X Marks the Spot. Extensible Markup Language Opens the Door to a Motherlode of Automated Web Applications."](#) By Rohit Khare and Adam Rifkin.
 - ["Keeping Tabs Online. Doing Business on the Net is Hard Because the Underlying Software is So Dumb. XML Will Fix That."](#) By Michael Krantz.
 - [Experts' Revolution. XML: A Professional Alternative to HTML."](#) By Ingo Macherius.
 - ["XML: Enabling Next-Generation Web Applications."](#) By Microsoft Staff.
 - ["Markup and Core Concepts."](#) By Erik T. Ray.
 - ["SGML and XML Concepts."](#) From the Society of American Archivists and the Library of Congress. Chapter 6 from the EAD Application Guidelines for Version 1.0. EAD Overview, useful generally as an SGML/XML tutorial.
 - ["A Gentle Introduction to XML."](#) From the [Text Encoding Initiative \(TEI\)](#) Guidelines [XML version] Chapter 2.
 - ["A Technical Introduction to XML."](#) By Norm Walsh.
 - [XML Basics Quick Start.](#) From [ZVON.org: The Guide to the XML Galaxy](#)

General Comments

For general comments or questions, please send e-mail correspondence to: it@ojp.gov.

APPENDIX 3

Glossary of Justice and Information Technology Terms

800 Megahertz (MHz). 800 MHz refers to public safety radio systems using channels located in or near the 800 MHz band. Approximately 300 channels located in the 800 MHz spectrum band have been assigned for use by state and local public safety entities. The disadvantage is that this higher frequency has less range and so a greater infrastructure is needed to cover the same range as lower frequencies.

Access. To interact with a system entity in order to manipulate, use, gain knowledge of, and/or obtain a representation of some or all of a system entity's resources.

Access control. Protection of resources against unauthorized access; a process by which use of resources is regulated according to a security policy and is permitted by only authorized system entities according to that policy.

Access control information. Any information used for access control purposes, including contextual information. Contextual information might include source Internet Protocol (IP) address, encryption strength, the type of operation being requested, time of day, etc. Portions of access control information may be specific to the request itself, some may be associated with the connection via which the request is transmitted, and others (for example, time of day) may be "environmental."

Access rights. A description of the type of authorized interactions a person or system can have with a resource. Examples include read, write, execute, add, modify, and delete.

AFIS (Automated Fingerprint Identification System). AFIS is a database of digitized offender fingerprint files. A user can enter a fingerprint and a computer will generate a list of possible matches within minutes. The matches are then examined and verified by a fingerprint expert.

Architecture. Architecture refers to the design of a system. It may refer to either hardware or software, or a combination of both. The software architecture of a program or computing system is the structure or structures of the system. This structure includes software components, the externally visible properties of those components, the relationships among them, and the constraints on their use.

Artifact. A piece of digital information. An artifact may be any size and may be composed of other artifacts. Examples of artifacts: a message, a URI, an XML document, a Portable Network Graphics (PNG) image.

Asynchronous. An interaction is said to be asynchronous when the associated messages are chronologically and procedurally decoupled. For example, in a request-response interaction, the client agent can process the response at some indeterminate point in the future when its existence is discovered. Mechanisms to do this include polling, notification by receipt of another message, etc.

Attribute. A characteristic of an object or entity. An object's attributes are said to describe the object. Objects' attributes are often specified in terms of their physical traits, such as size, shape, weight, and color, etc., for real-world objects. Objects in cyberspace might have attributes describing size, type of encoding, network address, etc.

Authentication. Authentication is the process of verifying that a potential partner in a conversation (or data exchange) is capable of representing a person or organization.

Authorization. The process of determining, by evaluating applicable access control information, whether a subject is allowed to have the specified types of access to a particular resource. Usually, authorization is in the context of authentication. Once a subject is

authenticated, it may be authorized to perform different types of access.

AVL (Automatic Vehicle Locator). AVL uses Global Positioning System technology to locate the position of patrol cars on a digital map. This information assists the dispatcher in knowing which calls should be assigned to which officers.

Binding. An association between an interface, a transmission protocol, and a data format. A binding specifies the protocol and data format to be used in transmitting messages defined by the associated interface. See also *SOAP binding*.

BIOS (Basic Input/Output System). BIOS controls the startup of the machines or computers and other functions, such as the keyboard, display, and disk drive. BIOS is stored on read-only memory and is not erased when the computer is turned off. BIOS on newer machines is stored on flash read-only memory, allowing it to be erased and rewritten to update BIOS.

CAD (Computer Aided Dispatch). A computer system that assists 911 operators and dispatch personnel in handling and prioritizing calls. Enhanced 911 will send the location of the call to the CAD system, which will automatically display the address of the 911 caller on a screen in front of the operator. Complaint information is then entered into the computer and is easily retrievable. The system may be linked to mobile data terminals (MDTs) in patrol cars, thereby allowing dispatchers and officers to communicate without using voice. The system may also be interfaced with NCIC, AVL, or a number of other programs.

Cardinality. The number of instances of an entity in relation to another entity, e.g., one-to-one, one-to-many, many-to-many

CDPD (Cellular Digit Packet Data). A data transmission technology that uses unused cellular phone channels to transmit data in packets.

Class. A description of a set of objects that share the same attributes, operations, methods, relationships, and semantics

Client/Server architecture. A network model that a computer or process server uses to provide services to the workstations (clients) connected to that computer (server). This architecture allows the client to share resources such as files, printers, and processing power with other clients.

Community of interest. A group of professionals informally bound to one another through exposure to a common class of problems, and common pursuit of solutions, and thereby embodying a store of knowledge. The justice and public safety domain is considered a community of interest.

Compliant. Hardware and software capable of satisfying a particular requirement, such as manipulation of four-digit dates, is deemed "compliant."

Component. A component is a software object, meant to interact with other components, encapsulating certain functionality or a set of functionalities. A component has a clearly defined interface and conforms to a prescribed behavior common to all components within an architecture.

Computer crime mapping. Computer crime mapping allows a department to display calls for service on a computerized pin map that aids in crime analysis efforts.

Conceptual data model (CDM). A data model that defines the real-world entities, and the relationships between these entities, in a business context. A CDM is typically constructed as an Entity Relationship Diagram (ERD), e.g., Unified Modeling Language (UML) class diagram.

Confidentiality. Assuring information will be kept secret, with access limited to appropriate persons.

Connection. A transport layer virtual circuit established between two programs for the purpose of communication.

Conversion. Conversion is the translation of valid values into another format on a permanent basis; for example, translating two-digit years to four-digit year values.

Core data type. Basic business data items that describe common concepts used in general business activities.

Data. Facts represented in a readable language (such as numbers, characters, images, or other methods of recording) on a durable medium. Data on its own carries no meaning. Empirical data are facts originating in or based on observations or experiences. A database is a store of data concerning a particular domain. Data in a database may be less structured or have weaker semantics (built-in meaning) than knowledge in a knowledge base. Compare data with *Information*.

Data architecture. A component of the design architecture, the data architecture consists of among others, data entities, which have attributes and relationships with other data entities. These entities are related to the business functions.

Data class. A set of data objects that share a common structure and a common behavior. The terms "class" and "type" are usually (but not always) interchangeable; a class is a slightly different concept than a type, in that it emphasizes the classifications of structure and behavior.

Data dictionary. A file that defines the basic organization of a database. It will contain a list of all files in the database, the number of records in each file, and the names and types of each field.

Data element. A basic unit of data having a meaning and distinct units and values. A uniquely named and defined component of data

definition; a data "cell" into which data items (actual values) can be placed; the lowest level of physical representation of data.

Data element [Federal Enterprise Architecture (FEA) Data Reference Model]. Physical description of the data used within an Information Exchange Package. A representation of a data object, a data property, and a data representation.

Data mart. A collection of data that is organized to support a specific application. The data is sometimes optimized for this application.

Data model. A graphical and/or lexical representation of data, specifying their properties, structure, and inter-relationships

Data object. A basic definition of the data element. Anything that exists in storage and on which operations can be performed, such as files, programs, or arrays. A collection of data elements that are aggregated for or by a specific application.

Data property. Description of the data element in context of the data object.

Data Reference Model (DRM). One of the five models in the FEA Reference Model framework, to aid in describing the types of interaction and exchanges that occur between the federal government and its various customers, constituencies, and business partners.

Data representation. Describes how data is described within the property and object layers.

Data standards. Data standards are agreed-upon terms for defining and sharing data.

Data type. A specification of the permissible content for a class of objects, where the content can be comprised of one or more literal values, i.e., positive integer, or any complex data structure, i.e., hierarchy of child elements

within an XML core component.

Data warehouse. An implementation of an informational database used to store sharable data sourced from an operational database-of-record. It is typically a subject database that allows users to tap into a company's vast store of operational data to track and respond to business trends and facilitate forecasting and planning efforts.

Database. A data structure that stores metadata, i.e., data about data. In general, it is an organized collection of information.

Digital signature. A value computed with a cryptographic algorithm and appended to a data object in such a way that any recipient of the data can use the signature to verify the data's origin and integrity.

Discovery. The act of locating a machine-processable description of a web service-related resource that may have been previously unknown and that meets certain functional criteria. It involves matching a set of functional and other criteria with a set of resource descriptions. The goal is to find an appropriate Web service-related resource.

Discovery service. A service that enables agents to retrieve web services-related resource description.

Document. Any data that can be represented in a digital form.

Document Type Definition (DTD). World Wide Web Consortium (W3C) Recommendation. A document-centric XML schema language.

Dublin Core Metadata Initiative (DCMI). Dublin Core is an open forum engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models. DCMI's activities include consensus-driven working groups, global conferences and workshops, standards liaison, and educational efforts to

promote widespread acceptance of metadata standards and practices.

Electronic Data Interchange (EDI). The automated exchange of any predefined and structured data for business among information systems of two or more organizations.

Element [XML]. The fundamental building block of an XML document. XML elements can contain other elements and/or text data. XML elements are composed of a start tag, content, and end tag.

Encryption. A process that translates plain text into a code. The reader of an encrypted file must have a key to decrypt the file. This involves cryptographic transformation of data (called "plaintext") into a form (called "ciphertext") that conceals the data's original meaning to prevent it from being known or used. If the transformation is reversible, the corresponding reversal process is called "decryption," which is a transformation that restores encrypted data to its original state.

End point. An association between a binding and a network address, specified by a URI, that may be used to communicate with an instance of a service. An end point indicates a specific location for accessing a service using a specific protocol and data format.

Entity [XML]. An information-sharing unit. All agencies are entities; so are courts and legislative bodies. Private organizations that share governmental information are also entities, as are private persons.

Firewall. A system designed to prevent unauthorized access to or from a private network. Firewalls are often used to prevent Internet users from accessing private networks connected to the Internet.

Functional specifications. Formal descriptions of a software system used as a blueprint for implementation. Specifications should state the purposes of the program,

provide implementation details, and describe the specific functions of the software from the user's perspective.

Gap analysis. The difference between projected outcomes and desired outcomes.

Gateway. An agent that terminates a message on an inbound interface with the intent of presenting it through an outbound interface as a new message. Unlike a proxy, a gateway receives messages as if it were the final receiver for the message. Due to possible mismatches between the inbound and outbound interfaces, a message may be modified and may have some or all of its meaning lost during the conversion process. For example, an HTTP PUT has no equivalent in SMTP.

GPS (Global Positioning System). A satellite navigation system operated by the U.S. Department of Defense. It provides coded satellite signals that can be processed by a GPS receiver, enabling the receiver to compute position, velocity, and time.

GUI (Graphical User Interface). GUI (often pronounced "goosey") is a program interface that uses a computer's graphic systems to make a program more user-friendly. GUI may include standard formats for representing text and graphics, making it easier to share data between programs running on the same GUI.

Hardware. Objects used to store and run software, such as a computer, monitor, keyboard, disk, and printer.

HTML (Hypertext Markup Language). A language that allows one to tag a document, primarily with markup used for presentation, for example, font size, typeface, headings, paragraphs, tables, etc.

IAFIS (Integrated Automated Fingerprint Identification System). A new (July 1999) national online fingerprint and criminal history database run by the FBI. Justice agencies that submit urgent electronic requests for

identification will receive a response within two hours.

Identifier. An identifier is an unambiguous name for a resource.

III (Interstate Identification Index).

Designed and run by the FBI, III is part of IAFIS and contains criminal history records for almost 30 million offenders and can be queried using a name, birth date, and other information.

Information. Contextual meaning associated with, or derived from, data.

Information Exchange Package (IEP). An IEP represents a set of data that is transmitted for a specific business purpose. It is the actual XML instance that delivers the payload or information. (The word "package" as used here refers to a package of the actual data, not a package of artifacts documenting the structure and content of the data.) An IEP can be prefixed with "GJXDM" to indicate or highlight that the IEP is conformant to the Global Justice XML Data Model, as in "GJXDM Information Exchange Package." The fact that an IEP is GJXDM-conformant may be readily apparent from the context, so it is not absolutely necessary to use the word "GJXDM" even if the IEP is GJXDM-conformant. (See also *Reference*.)

Information Exchange Package

documentation. A collection of artifacts that describe the structure and content of an IEP. It does not specify other interface layers (such as Web services). It can optionally be prefixed with "GJXDM" to indicate or highlight that a resulting IEP is GJXDM-conformant. This term replaces "Exchange Document." (See also *Reference*.)

Instance [XML]. Representation of the values of all the XML items.

Integrity. Assuring information will not be accidentally or maliciously altered or destroyed.

Interface. A program or device that connects programs and/or devices.

Internet. A decentralized global network connecting millions of computers.

Interoperable [Data]. Interoperable means to be functionally equivalent or interchangeable components of the system or process in which they are used.

Intranet. A secure private network that uses TCP/IP protocols.

LAN (Local Area Network). A computer network that connects workstations and personal computers and allows them to access data and devices anywhere on the LAN. A LAN is usually contained within one building.

Laptop. A computer that has capabilities beyond that of the mobile data computer. It may contain report writing and accident reconstruction programs.

LAWN (Local Area Wireless Network). A LAN that uses high-frequency radio waves rather than wires to communicate between nodes.

Legacy system. Older software and hardware systems still in use and generally proprietary.

Lexicon. Provides a glossary and cross-reference for words that may have multiple meanings. The purpose is to create common definitions to allow for clearer understanding.

Live scan. A machine that replaces ink-and-roll fingerprints. Fingers are rolled across a platen, scanned into a computer, and converted to a digital form of storage. Fingerprint cards are then printed out on a laser printer. The machine will immediately reject low-quality prints.

Logical data model. A model of the logical representation of objects about which the enterprise records information, in either automated or nonautomated form. It would

be represented as a fully attributed, keyed, normalized entity relationship model reflecting the intent of the semantic model.

Loose coupling. Coupling is the dependency between interacting systems. This dependency can be decomposed into *real dependency* and *artificial dependency*:

- Real dependency is the set of features or services that a system consumes from other systems. The real dependency always exists and cannot be reduced.
- Artificial dependency is the set of factors that a system has to comply with in order to consume the features or services provided by other systems. Typical artificial dependency factors are language dependency, platform dependency, application programming interface (API) dependency, etc. Artificial dependency always exists, but it or its cost can be reduced.

Loose coupling describes the configuration in which artificial dependency has been reduced to the minimum.

MDC (Mobile Data Computer). A microcomputer used by public safety agencies to access databases for information on persons and property. The MDC uses wireless communication and allows an officer to exchange information with the dispatcher and other officers without using voice channels.

Message. The basic unit of communication between a requester and a provider. In the context of a web service, the message contains the data to be communicated to or from a web service as a single logical transmission. See also *SOAP message*.

Message correlation. The association of a message with a context. Message correlation ensures that the requester can match the reply with the request, especially when multiple replies may be possible.

Message Exchange Pattern (MEP). A template, devoid of application semantics, that describes a generic pattern for the exchange of messages between exchange partners. It describes the relationships (e.g., temporal, causal, sequential, etc.) of multiple messages exchanged in conformance with the pattern, as well as the normal and abnormal termination of any message exchange conforming to the pattern.

Message receiver. An exchange partner that receives a message.

Message reliability. The degree of certainty that a message will be delivered and that sender and receiver will both have the same understanding of the delivery status.

Message sender. The exchange partner that transmits a message.

Message transport. A mechanism that may be used by exchange partners to deliver messages.

Metadata. Represents information about the data and could include value constraints, naming rule, etc.

Metadata registry. An information system for registering metadata

Namespace. Namespaces are the solution to naming conflicts in XML. Using XML namespaces can help alleviate issues that arise where XML elements and attributes use identical names. XML namespaces help to identify and resolve conflicts between elements that have the same name but mean different things. A namespace is a domain that contains a set of XML element names.

NCIC or NCIC 2000 (National Crime Information Center). NCIC is a computer system maintained by the FBI that can be queried by local agencies via state computer systems known as "control terminal agencies." NCIC contains 17 files with over 10 million

records, plus 24 million criminal history records contained within the Interstate Identification Index (one of the 17 files). Files include the III, the Missing Persons File, the Unidentified Persons File, the U.S. Secret Service Protective File, and the Violent Gang/Terrorist File.

Network. A network is created when two or more computers are joined by some type of transmission media allowing them to communicate directly, or to share storage devices and peripherals. Transmission media can include cable lines, telephone lines, or satellite systems.

NIBRS (National Incident-Based Reporting System). An incident-based crime reporting system, run by the FBI, through which data is collected on each single crime occurrence. NIBRS data is designed to be generated as a byproduct of local, state, and federal automated records systems. NIBRS collects data on each single incident and arrest within 22 offense categories made up of 46 specific crimes called Group A offenses. Specific facts are collected for each of the offenses coming to the attention of public safety agencies. In addition to Group A offenses, there are 11 group B offense categories that only report arrest data. NIBRS is expected to eventually replace UCR.

NLETS. NLETS—the International Justice and Public Safety Information Sharing Network, formerly known as the National Law Enforcement Telecommunications System, is a high-speed communications network and message switch that connects almost every public safety agency in the country. It allows local agencies to make inquiries into state databases to access criminal history records, vehicle registration records, driver's license files, etc. NLETS also interfaces with NCIC and other national files and allows states to exchange information with each other.

Node. A node can be a computer or some other device such as a printer. Every node has a unique network address.

Nonrepudiation. Method by which the sender of data is provided with proof of delivery and the recipient is assured of the sender's identity, so that neither can later deny having processed the data.

OASIS (Organization for the Advancement of Structured Information). A not-for-profit consortium that advances electronic business by promoting open, collaborative development of interoperability specifications.

Object. Anything perceivable or conceivable; a real-world entity.

Object-oriented programming (OOP). OOP combines data structures and functions (computer directions) to create "objects," making it easier to maintain and modify software.

OMG (Object Management Group). The industry group dedicated to promoting object-oriented technology and its standardization.

Ontology. An explicit formal specification of how to represent the objects, concepts, and other entities that are assumed to exist in some area of interest and the relationships that hold among them. In computer science, an ontology is the attempt to formulate an exhaustive and rigorous conceptual schema within a given domain, a typically hierarchical data structure containing all the relevant entities and their relationships and rules (theorems, regulations) within that domain.

Open architecture. Open architecture systems are designed to allow system components to be easily connected to devices and programs made by other manufacturers.

Operating system. The basic program used by a computer to run other programs. An operating system recognizes input from the keyboard, sends output to the display screen, and keeps track of files and directories on the disk and controlling peripheral devices such as disk drives and printers. It provides a platform

for other software applications.

OWL (Web Ontology Language). OWL is intended to be used when the information contained in documents needs to be processed by applications, as opposed to situations where the content only needs to be presented to humans. OWL can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms. This representation of terms and their interrelationships is called an ontology. OWL has more facilities for expressing meaning and semantics than XML, RDF, and RDF-S, and thus OWL goes beyond these languages in its ability to represent machine-interpretable content on the Web.

Permission. A policy that prescribes the allowed actions of an agent and/or resource.

Person or organization. A person or organization may be the owner or agents that provide or request Web services.

Physical data model. A physical representation of the objects of the enterprise. The representation style of this model would depend on the technology chosen for implementation. If relational technology is chosen, this would be a model of the table structure required to support the logical data model in a relational-style model. In an object-oriented notation, this would be a class-hierarchy/association-style model.

Platform. The underlying hardware or software for a system. The term is often used as a synonym for operating system

Policy. A constraint on the behavior of agents, persons, or organizations.

Primitive types. Primitive types, as distinct from composite types, are datatypes provided by a programming language as basic building blocks. Typical primitive types include:

- Character.

- Integer (with a variety of precisions).
- Floating-point number with binary representation usually conforming to the Institute of Electrical and Electronics Engineers (IEEE) standards for floating point representation.
- Fixed-point with a variety of precisions and a programmer-selected scale.
- Boolean, having the values “true” and “false.”
- String, a sequence of characters.
- Reference (also called a “pointer” or “handle”), a small value referring to another object, possibly a much larger one.

Privacy policy. A set of rules and practices that specify or regulate how a person or organization collects, processes (uses), and discloses another party’s personal data as a result of an interaction.

Property. A characteristic common to all members of an object class.

Proprietary. The term “proprietary” generally refers to a system whose manufacturer will not divulge specifications that would allow other companies to duplicate the product. It is also known as a closed architecture.

Protocol. A set of formal rules describing how to transmit data, especially across a network. Low-level protocols define the electrical and physical standards to be observed, bit- and byte-ordering, and the transmission and error detection and correction of the bit stream. High-level protocols deal with the data formatting, including the syntax of messages, the terminal-to-computer dialogue, character sets, sequencing of messages, etc.

Proxy. An agent that relays a message between a requester agent and a provider agent, appearing to the Web service to be the requester.

Quality of service. An obligation accepted

and advertised by a provider entity to service consumers.

RMS (Records Management System). An RMS stores computerized records of crime incident reports and other data. It may automatically compile information for UCR or NIBRS reporting. Can perform greater functions when integrated with other systems such as CAD and GPS.

Reference. Information Exchange Package Documentation may have the word “Reference” in its title if it has been mandated, approved, endorsed, recommended, or acknowledged by a cognizant organization, e.g., “GJXDM Information Exchange Package Documentation for a Reference Incident Report.” Reference IEP Documentation often may be used as the basis for IEP Documentation meeting the specific business needs of an information-sharing enterprise. This term replaces “Reference Exchange Document” and “Reference Document.”

Reference architecture. The generalized architecture of several end systems that share one or more common domains. The reference architecture defines the infrastructure common to the end systems and the interfaces of components that will be included in the end systems. The reference architecture is then instantiated to create a software architecture of a specific system. The definition of the reference architecture facilitates deriving and extending new software architectures for classes of systems. A reference architecture, therefore, plays a dual role with regard to specific target software architectures. First, it generalizes and extracts common functions and configurations. Second, it provides a base for instantiating target systems that use that common base more reliably and cost effectively.

Regression test. A regression test is performed before production to identify and prevent errors and verify that unchanged software will continue to function as designed.

Relational Database Management System.

A type of database management system that stores data in related tables. New types of data can more easily be added, and the user can view the data in multiple ways.

Repository. An information system used to store and access architectural information, relationships among the information elements, and work products.

Resource Description Framework (RDF).

A Semantic web standard that provides a framework for asset management, enterprise integration, and the sharing and reuse of data on the Web.

Scaleable. A term that describes how well a system can be adapted and expanded to meet increased demands.

Schema. Specification of the characteristics and relationships of a class of objects.

Schema [XML]. A specification to define the structure of XML documents and to specify datatypes for attribute values and element content. In addition to the DTD, there are several XML schema languages, including XML Schema (W3C), Schematron, and RELAX NG.

Scope creep. The slow and continuous expansion of the scope of a project, such as data type or routine, resulting in a broad, unfocused, and unmanageable scope and usually leading to cost-overruns, missed deadlines, and loss of original goals.

Security architecture. A plan and set of principles for an administrative domain and its security domains that describe the security services that a system is required to provide to meet the needs of its users, the system elements required to implement the services, and the performance levels required in the elements to deal with the threat environment. A complete security architecture for a system addresses administrative security, communication security, computer security,

emanations security, personnel security, and physical security, and prescribes security policies for each. A complete security architecture needs to deal with both intentional, intelligent threats and accidental threats. A security architecture should explicitly evolve over time as an integral part of the evolution of its administrative domain.

Security policy. A set of rules and practices that specify or regulate how a system or organization provides security services to protect resources. Security policies are components of security architectures. Significant portions of security policies are implemented via security services, using security policy expressions.

Security service. A processing or communication service that is provided by a system to give a specific kind of protection to resources, where said resources may reside with said system or reside with other systems, for example, an authentication service or a public key infrastructure (PKI)-based document attribution and authentication service. A security service is a superset of AAA (authentication, authorization, accounting) services. Security services typically implement portions of security policies and are implemented via security mechanisms.

Semantic model. A model of the actual enterprise objects (i.e., things, assets) that is significant to the enterprise. Typically, the semantic model would be represented as an entity/relationship model and would be at a level of definition expressing concepts (i.e., terms and facts) used in the significant business objectives/strategies implemented later as business rules.

Semantic web. The Semantic web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. It is based on the RDF, which integrates a variety of applications using XML for syntax and URIs for naming.

Server. The program in the client/server architecture that answers client's requests. The term "server" is also used to designate the computer that makes resources available to the workstations (clients) on the network.

Service. An abstract resource that represents a capability of performing tasks that form a coherent functionality from the point of view of data providers and requesters.

Service description. A set of documents that describe the interface to and semantics of a service.

Service interface. The abstract boundary that a service exposes. It defines the types of messages and the message exchange patterns that are involved in interacting with the service, together with any conditions implied by those messages.

Service semantics. The semantics of a service is the behavior expected when interacting with the service. The semantics expresses a contract (not necessarily a legal contract) between the provider entity and the requester entity. It expresses the effect of invoking the service. Service semantics may be formally described in a machine-readable form; identified but not formally defined; or informally defined via an agreement between the provider and the requester.

Service-oriented Architecture (SOA). An architectural style whose goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer. Both service provider and service consumer are roles played by software agents/brokers on behalf of their owners. The communication can involve either simple data exchange or it could involve two or more services coordinating some activity. Some means of connecting services to each other is needed.

SOAP (Simple Object Access Protocol). The formal set of conventions governing the format and processing rules of a SOAP message.

SOAP application. A software entity that produces, consumes, or otherwise acts upon SOAP messages in a manner conforming to the SOAP processing model.

SOAP binding. The formal set of rules for carrying a SOAP message within or on top of another protocol (underlying protocol) for the purpose of exchange. Examples of SOAP bindings include carrying a SOAP message within an HTTP entity-body, or over a TCP stream.

SOAP body. A collection of zero or more *element information items* targeted at an ultimate SOAP receiver in the SOAP message path.

SOAP envelope. The outermost *element information item* of a SOAP message.

SOAP feature. An extension of the SOAP messaging framework typically associated with the exchange of messages between communicating SOAP nodes. Examples of features include "reliability," "security," "correlation," "routing," and the concept of message exchange patterns.

SOAP header. A collection of zero or more SOAP header blocks, each of which might be targeted at any SOAP receiver within the SOAP message path.

SOAP header block. An *element information item* used to delimit data that logically constitutes a single computational unit within the SOAP header. The type of a SOAP header block is identified by the fully qualified name of the header block *element information item*.

SOAP intermediary. A SOAP intermediary is both a SOAP receiver and a SOAP sender and is targetable from within a SOAP message. It

processes the SOAP header blocks targeted at it and acts to forward a SOAP message toward an ultimate SOAP receiver.

SOAP message. The basic unit of communication between SOAP nodes.

SOAP message exchange pattern (MEP). A template for the exchange of SOAP messages between SOAP nodes enabled by one or more underlying SOAP protocol bindings. A SOAP MEP is an example of a SOAP feature.

SOAP message path. The set of SOAP nodes through which a single SOAP message passes. This includes the initial SOAP sender, zero or more SOAP intermediaries, and an ultimate SOAP receiver.

SOAP node. The embodiment of the processing logic necessary to transmit, receive, process, and/or relay a SOAP message, according to the set of conventions defined by this recommendation. A SOAP node is responsible for enforcing the rules that govern the exchange of SOAP messages. It accesses the services provided by the underlying protocols through one or more SOAP bindings.

SOAP receiver. A SOAP node that accepts a SOAP message.

SOAP role. A SOAP node's expected function in processing a message. A SOAP node can act in multiple roles.

SOAP sender. A SOAP node that transmits a SOAP message.

Software. A set of computer instructions or data stored in an electronic format.

Spectrum. The array of channels, like the channels on a television, available for communications transmissions. Commonly referred to as a spectrum, these channels are a finite natural resource; they cannot be created, purchased, or discovered.

SQL (Structured Query Language). A language used specifically by a relational database to query, modify, and manage information.

State. A set of attributes representing the properties of a component at some point in time.

Synchronous. An interaction is said to be synchronous when the participating agents must be available to receive and process the associated messages from the time the interaction is initiated until all messages are actually received or some failure condition is determined. The exact meaning of "available to receive the message" depends on the characteristics of the participating agents (including the transfer protocol it uses); it may, but does not necessarily, imply tight time synchronization.

Systems software. Systems software consists of the operating system and all utilities that enable the computer to function.

Taxonomy. A hierarchical classification or categorization of a set of things

TCP/IP (Transmission Control Protocol/Internet Protocol). TCP/IP is standard transmission protocol used to connect hosts on the Internet.

Transaction. Transaction is a feature of the architecture that supports the coordination of results or operations in a multiple-step interaction. The fundamental characteristic of a transaction is the ability to join multiple actions into the same unit of work, such that the actions either succeed or fail as a unit.

Type. A description of a class of objects that share the same operations, abstract attributes and relationships, and semantics.

UCR (Uniform Crime Reports). UCR is a city, county, and state public safety program operated by the FBI that provides a nationwide

view of crime based on the submission of statistics by public safety agencies throughout the country. The following offenses are recorded: murder and nonnegligent manslaughter; forcible rape; robbery; aggravated assault; burglary; larceny theft; motor vehicle theft; arson; and hate crimes.

Unified Resource Identifier (URI). A URI points to an external file that defines the namespace. The URI may either be a **URL** (Universal Resource Locator) that points to a web server or a **URN** (Universal Resource Name) that names a resource, but which does not specify a de-referenceable network object.

Validation. The evaluation of a system during development or at the time of completion to determine if it satisfies all the requirements.

WAN (Wide Area Network). A WAN consists of two or more LANs connected via telephone lines or radio waves.

Web browser. A software application used to locate and display web pages. It may be able to display graphics, sound, and video in addition to text.

Web service. A web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other web-related standards.

Web Services Description Language (WSDL). An XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are

combined into abstract endpoints (services). WSDL is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate.

World Wide Web. A system of Internet servers that support HTML-formatted documents.

World Wide Web Consortium (W3C). The World Wide Web Consortium is the international standards body for interoperable technologies (specifications, guidelines, software, and tools) that support the development of the World Wide Web to its full potential, *including XML*. W3C is a collaborative forum for information, commerce, communication, and collective understanding.

XML (eXtensible Markup Language). XML is a structured language for describing information being sent electronically by one entity to another. XML Schema defines the rules and constraints for the characteristics of the data, such as structure, relationships, allowable values, and data types.

XML Namespace. A simple method for qualifying element and attribute names used in XML documents by associating them with namespaces identified by URI references

XML Schema Language (XSD). W3C Recommendation. Specification for defining the structure, content, and semantics of XML documents. Defines a richer set of datatypes than the DTD. XML schemas support namespaces.

APPENDIX 4

Global Justice XML Data Model Partners

American Association of Motor Vehicle Administrators (AAMVA)

www.aamva.org

AAMVA is a nonprofit organization striving to develop model programs in motor vehicle administration, police traffic services, and highway safety. AAMVA serves as an information clearinghouse for these same disciplines and acts as an international spokesperson for these interests.

Automated Regional Justice Information System (ARJIS)

www.arjis.org/

ARJIS is a complex criminal justice enterprise network used by local, state, and federal agencies in the San Diego, California region. ARJIS is chartered with supporting a regional web-based enterprise network that uses technical and operational standards to build interfaces to all criminal justice systems in the region. The ARJIS secure Intranet, ARJISNet, contains data on the region's crime cases, arrests, citations, field interviews, traffic accidents, fraudulent documents, photographs, gang information, and stolen property. ARJIS also is used for tactical analysis, investigations, statistical information, and crime analysis.

Bureau of Justice Assistance (BJA), U.S. Department of Justice (DOJ)

www.ojp.usdoj.gov/BJA/

The Bureau of Justice Assistance (BJA) is a component of the Office of Justice Programs (OJP), U.S. DOJ. Its mission is to provide leadership and services in grant administration and criminal justice policy development to support local, state, and tribal justice strategies to achieve safer communities. BJA's overall goals are to (1) reduce and prevent crime, violence, and drug abuse and (2) improve the functioning of the criminal justice system. To achieve these goals, BJA programs emphasize enhanced coordination and cooperation of federal, state, and local efforts. BJA has three primary components: Policy, Programs, and

Planning. The Policy Office provides national leadership in criminal justice policy, training, and technical assistance to further the administration of justice. It also acts as a liaison to national organizations that partner with BJA to set policy and help disseminate information on best and promising practices. The Programs Office coordinates and administers all state and local grant programs and acts as BJA's direct line of communication to states, territories, and tribal governments by providing assistance and coordinating resources. The Planning Office coordinates the planning, communications, and budget formulation and execution; provides overall BJA-wide coordination; and supports streamlining efforts.

Conference of State Court Administrators (COSCA)

<http://cosca.ncsc.dni.us/>

COSCA is dedicated to the improvement of state court systems. Its membership consists of the state court administrator or equivalent official in each of the 50 states, the District of Columbia, Puerto Rico, American Samoa, Guam, Northern Mariana Islands, and the Virgin Islands. Its mission is to provide a national forum to assist state court administrators in the development of a more just, effective, and efficient system of justice, by:

- Identifying and studying issues and, when appropriate, developing policies, principles, and standards relating to the administration of judicial systems.
- Providing an effective network for the exchange of information, ideas, and methods to improve state courts.
- Facilitating cooperation, consultation, and exchange of information by and among organizations directly concerned with court administration.
- Assisting in the formulation and implementation of national issues that affect state courts.
- Establishing and maintaining an

organization that is open, inclusive, participatory, dynamic, and responsive.

- Offering educational opportunities.

Corrections Technology Association (CTA)

www.correctionstech.org/

CTA is a network of professionals brought together to discuss, collaborate, and promote the technology issues affecting corrections and probation and parole. They represent a collective interest before all levels of government and the public. Members meet with [corporate sponsors](#) in order to maintain an awareness of the products and services available that are related to technology in corrections. In addition, CTA members receive and share information, ideas, and experiences with technology executives across the United States. CTA's mission is to provide an association of and a forum for technology executives serving the corrections field to address challenges, promote information sharing, and advise on emerging technologies affecting corrections.

Criminal Justice Information Services (CJIS) Division, Federal Bureau of Investigation (FBI)

www.fbi.gov/hq/cjisd/cjis.htm

The CJIS Division was established in February 1992 to serve as the focal point and central repository for criminal justice information services in the FBI. It is the largest division within the FBI. Programs that were initially consolidated under the CJIS Division include the National Crime Information Center (NCIC), Uniform Crime Reporting (UCR), and Fingerprint Identification. The CJIS Division is also responsible for these initiatives: the Integrated Automated Fingerprint Identification System (IAFIS), NCIC 2000, and the National Incident-Based Reporting System (NIBRS).

CriMNet

www.crimnet.state.mn.us

CriMNet is an enterprise architecture that puts in place a statewide framework of people, processes, data, standards, and technology focused on providing accurate and

comprehensive data to the criminal justice community in the State of Minnesota. It provides the means to put "the right data in the hands of the right people at the right time and in the right place." The CriMNet integration effort is not one single project but incorporates many projects that are being developed by criminal justice organizations throughout Minnesota. The integration architecture is driven by local operational needs and uses standards that will support the exchange of data across existing and developing systems.

Georgia Tech Research Institute (GTRI)

www.gtri.gatech.edu

GTRI is the nonprofit applied research arm of the Georgia Institute of Technology in Atlanta, Georgia. With more than 1,000 employees, GTRI supports approximately \$100 million in research yearly for more than 200 clients in industry and government. GTRI researchers have played a pivotal role in the engineering support and technical guidance of the Global Justice XML Data Model (GJXDM).

GJXDM Training and Technical Assistance Committee (GTTAC)

<http://www.it.ojp.gov/gttac>

BJA's GTTAC is a consortium of organizations engaged in technical assistance and training related to technology in the justice field, specifically GJXDM. GTTAC, established in January 2004 as an outreach effort for GJXDM, supports the development of data model training and technical assistance. It is related to, but external from, DOJ's Global Justice Information Sharing Initiative, and is an operating entity on behalf of GJXDM to assist the justice community's need to better understand and implement the data model. GTTAC's mission is to coordinate the work of national service providers in providing training and technical assistance on issues related to the implementation of GJXDM. Major projects include building GJXDM Information Exchange Package descriptions, creating a national virtual help desk centered on the GJXDM, and coordinating regional large-scale GJXDM training events.

Global Justice Information Sharing Initiative (Global)

www.it.ojp.gov/global

DOJ's Global is a "group of groups" that represent more than 30 independent organizations spanning the spectrum of law enforcement, judicial, correctional, and related bodies. Member organizations participate in Global out of shared responsibility and shared belief that together they can bring about positive changes in interorganizational communication and data sharing. The Global Advisory Committee (GAC) serves as an advisory committee for DOJ. Global aids its member organizations and the people it serves through a series of important initiatives. These include the facilitation of the Global Working Groups, development of technology standards, creation of white papers on data sharing issues, and the dissemination of information via the Global web site. The work of GAC has implications of the highest importance and helps to make GAC the foremost voice for justice information sharing.

Global Infrastructure/Standards Working Group (GISWG)

www.it.ojp.gov/topic.jsp?topic_id=57

Successful data exchange is greatly facilitated by the development and adoption of standards that enable transparent integration of disparate systems. DOJ's GISWG is implementing a coordination process to identify information-sharing standards within the justice community. This effort includes publishing, cataloging, and sharing these standards to promote collaborative efforts and offer blueprints to those beginning the information exchange planning process.

Global XML Structure Task Force (GXSTF)

www.it.ojp.gov/xstf

BJA's GXSTF is a component of GISWG. GXSTF addresses the requirements, design, structure, and implementation for the Global Justice XML Data Model. Its vision is to significantly advance justice information sharing by providing a common language and vocabulary that reduces cost and technical barriers. More specifically,

GXSTF has developed a consistent, extendable, maintainable XML schema reference specification for data elements and types that represent the data requirements of the general justice and public safety communities. GXSTF is heavily involved in the GJXDM release process and approves all fixes, additions, deletions, and modifications to each implementation. GXSTF also is responsible for GJXDM guidance, review, and issue resolution.

Institute for Telecommunication Sciences (ITS)

<http://www.its.bldrdoc.gov/>

The National Telecommunications and Information Administration (NTIA), an agency of the U.S. Department of Commerce, is the Executive Branch's principal voice on domestic and international telecommunications and information technology issues. NTIA works to spur innovation, encourage competition, help create jobs, and provide consumers with more choices and better quality telecommunications products and services at lower prices. ITS is the chief research and engineering arm of NTIA. ITS supports NTIA telecommunications objectives such as promotion of advanced telecommunications and information infrastructure development in the United States, enhancement of domestic competitiveness, improvement of foreign trade opportunities for U.S. telecommunications firms, and facilitation of more efficient and effective use of the radio spectrum. ITS also serves as a principal federal resource for solving the telecommunications concerns of other federal agencies, local and state governments, private corporations and associations, and international organizations.

Integrated Criminal Justice Information System (ICJIS)

www.Maricopa.gov/ICJIS

ICJIS is a project sponsored by Maricopa County, Arizona that provides integration services for sharing information between its internal stakeholders (Sheriff's Office, County Attorney Office, Clerk of the Superior Court, Superior Court, and indigent representation

agencies) and to provide these stakeholders information-sharing capabilities with external agencies that include municipal, state, and federal agencies. ICJIS stakeholders share information between 25 criminal justice systems, in addition to all state and federal information sources.

Integrated Justice Information Systems (IJIS) Institute

<http://iwg.ijis.org/>

The mission of the IJIS Institute is to contribute to the implementation of integrated justice information systems throughout the country by applying the knowledge and experience of the information technology industry. As information technology (IT) professionals responsible for the achievement of solution systems, the IJIS Institute believes that experience and perspective will improve the quality and reduce the time to market for solutions. The IJIS Institute supports the initiative of the Office of Justice Programs to involve industry in its information-sharing programs and believes that the program will benefit from its unique and collective experience.

International Association of Chiefs of Police (IACP)

www.iacp.org/

IACP is the world's oldest and largest nonprofit membership organization of police executives, with over 20,000 members in over 89 different countries. IACP's leadership consists of the operating chief executives of international, federal, state and local agencies of all sizes. Founded in 1893, the association's goals are to advance the science and art of police services; to develop and disseminate improved administrative, technical, and operational practices and promote their use in police work; to foster police cooperation and the exchange of information and experience among police administrators throughout the world; to bring about recruitment and training in the police profession of qualified persons; and to encourage adherence of all police officers to high professional standards of performance and conduct. IACP has accomplished its goals

through launching historically acclaimed programs, conducting ground-breaking research, and providing exemplary programs and services to its global membership.

International Organization for Standardization (ISO)

www.iso.org

ISO is a network of national standards institutes from 145 countries working in partnership with international organizations, governments, industry, business, and consumer representatives.

Joint Task Force on Rap Sheet Standardization

www.doj.state.wi.us/les/XML/jtf.htm

The Joint Task Force is an endeavor by the FBI and NLETS–The International Justice and Public Safety Information Sharing Network to bring about a national standard for the exchange of criminal history rap sheets. Sponsored by the FBI, members include staff of the FBI, NLETS, and states that operate criminal history repositories. In 1995, the National Task Force on Increasing the Utility of the Criminal History Record recommended expanded data content, a presentation format (page layout) for the expanded content, and the creation of a transmission format for the interstate sharing of criminal history information. The National Task Force included representatives from the FBI CJIS Advisory Policy Board (APB); NLETS; the National Center for State Courts; and SEARCH, The National Consortium for Justice Information and Statistics. Its members were a diverse array of justice practitioners drawn from the judiciary; prosecution; court administration; local, state, and federal law enforcement; juvenile justice pretrial services; and state criminal records repositories.

Justice Information Exchange Model (JIEM)

www.search.org/programs/technology/jiem.asp

JIEM® consists of a framework that defines universal dimensions of information exchange, a research and planning methodology for modeling the operational dynamics of this

information exchange, and a web-based software application—the JIEM Modeling Tool—that enables data collection, analysis, and reporting by users and researchers.

Justice Information Sharing Professionals (JISP)

www.jisp.us

JISP is a national network of local and state justice and public safety integrators responsible for the facilitation, collaboration, and advocacy of information sharing. JISP was created to focus on the need to enhance communication among practitioners. JISP coordinates a member-only Internet mail list at <http://groups.yahoo.com/group/JISP/>.

Law Enforcement Information Technology Standards Council (LEITSC) Technology Center

www.leitsc.org/

LEITSC is funded through the Office of Justice Programs, DOJ, to address the issue of IT standards specific to the law enforcement community. The mission of LEITSC is to foster the growth of strategic planning and implementation of integrated justice systems by promoting the merits of IT standards, providing advice to the nation's law enforcement community on technical aspects of IT standards, sharing practical solutions, and representing the voice of law enforcement in the expansion of justice and public safety IT standards. LEITSC partners include IACP, the Police Executive Research Forum, the National Sheriffs' Association, and the National Organization of Black Law Enforcement Executives.

LegalXML Court Filing Standard Initiative

http://it.ojp.gov/topic.jsp?topic_id=65

Pursuant to discussions at an August 1999 planning meeting, the COSCA/NACM Joint Technology Committee (JTC) formed an e-filing standards subcommittee to pursue an Internet electronic filing specification for the courts. To that end, in December 1999, the JTC voted to partner with LegalXML, a nonprofit organization that facilitates development of XML standards

for application within the legal community. This coalition produced the LegalXML Court Filing Standard.

LegalXML Integrated Justice Initiative

www.oasis-open.org/committees/tc_home.php?wg_abbrev=legalxml-intjustice

The purpose of the Integrated Justice Technical Committee is to support XML specifications for exchanging data among justice system branches and agencies. While its principal focus will be on data pertaining to criminal cases, its scope will include certain data exchanges in civil cases, such as civil protection order, child support enforcement, and dependency and neglect cases. The Committee will also serve as a vehicle for vetting the Global Justice XML Data Dictionary being developed under the auspices of the Infrastructure/Standards Working Group of the Global Advisory Committee, DOJ.

Los Angeles County Information Systems Advisory Body (ISAB)

<http://bos.co.la.ca.us/Categories/ExecOffice/CriminalJusticeCoordComm.htm>

ISAB is a member of the Countywide Criminal Justice Coordination Committee, and is responsible for overseeing the development and coordination of criminal justice information systems in Los Angeles County, California. ISAB has overall responsibility for the planning, development, implementation, and management of countywide criminal justice data and telecommunication system projects and providing related technical assistance to ISAB members, which include the Sheriff, District Attorney, Public Defender, Probation Department, Coroner, Superior and Municipal Courts, Chief Administrative Office, Internal Services Department, Los Angeles County Police Chiefs Association, Los Angeles Police Department, and the Alternate Public Defender.

Los Angeles County Sheriff's Department (LASD)

<http://www.lasd.org/>

LASD is the largest sheriff's department in the world. It serves the entire county of Los

Angeles, California, encompassing 4,061 square miles with a population of about 10 million. LASD manages the largest county jail system in the world, which averages about 19,000 inmates daily. In addition, the department is responsible for police services in the unincorporated areas and 41 incorporated cities for which police and traffic services have been contracted.

Mapping Alaska's Justice InterChanges (MAJIC)

www.dps.state.ak.us/cjiab/documents/default.asp?passParent=100&passCategory=MAJIC+Project

The MAJIC program, managed by the state's Criminal Justice Information Advisory Board, brings together the systems in many organizations statewide, including the Department of Public Safety, the Alaska Court System, the Public Defender Agency, and the National Law Enforcement and Corrections Technology Center-Northwest.

National Association for Court Management (NACM)

www.nacmnet.org/

NACM is an organization of court management professionals dedicated to the improvement of courts and the development of court managers. NACM has been a leader in advancing the profession; is committed to improving the administration of justice and promoting the interdependence of court managers and judges; is dedicated to providing members the opportunity for the finest continuing education available; and is committed to improving the public's access to trial courts and to educating the public on the role of the courts. NACM provides court management professionals the opportunity to increase their proficiency while working with colleagues to improve the administration of justice.

National Association of State Chief Information Officers (NASCIO)

<https://www.nascio.org>

NASCIO represents state chief information officers and information resource executives

and managers from the 50 states, 6 U.S. territories, and the District of Columbia. State members are senior officials from any of the 3 branches of state government who have executive-level and statewide responsibility for information resource management. Representatives from federal, municipal, and international governments and state officials who are involved in information resource management but do not have chief responsibility for that function participate in the organization as associate members. Private-sector firms and nonprofit organizations may join as corporate members. NASCIO's mission is to foster government excellence through quality business practices, information management, and technology policy. Its vision is government in which the public trust is fully served through the efficient and effective use of technology.

National Center for State Courts (NCSC)

www.ncsconline.org

NCSC provides up-to-date information and hands-on assistance that help court administrators to better serve the public. Through original research, consulting services, publications, and national educational programs, NCSC offers solutions that enhance court operations with the latest technology, collects and interprets the latest data on court operations nationwide, and provides information on proven "best practices" for improving court operations in areas such as civil case management.

National Crime Information Center (NCIC)

www.fas.org/irp/agency/doj/fbi/is/ncic.htm

NCIC is a computerized index of criminal justice information (i.e., criminal record history information, fugitives, stolen properties, and missing persons). It is available to local, state, and federal law enforcement and other criminal justice agencies and is operational 24 hours per day, 365 days per year. The purpose for maintaining the NCIC system is to provide a computerized database for ready access by a criminal justice agency making an inquiry and to provide prompt disclosure of information in the system from other criminal justice agencies.

This information assists authorized agencies in criminal justice and related law enforcement objectives, such as apprehending fugitives, locating missing persons, and locating and returning stolen property, as well as protecting law enforcement officers.

National Governors Association (NGA)

www.nga.org/

NGA is a public policy organization that acts as the collective voice of the nation's governors. It provides governors and their senior staff members with services that range from representing states on Capitol Hill and before the Administration on key federal issues to developing policy reports on innovative state programs and hosting networking seminars for state government executive branch officials. The NGA Center for Best Practices focuses on state innovations and best practices on issues that range from education and health to technology, welfare reform, and the environment. NGA also provides management and technical assistance to both new and incumbent governors.

National Incident-Based Reporting System (NIBRS)

www.ojp.usdoj.gov/bjs/nibrs.htm

The FBI's Uniform Crime Reporting (UCR) Program, which began in 1929, collects information about crimes reported to the police. In 1982, DOJ's Bureau of Justice Statistics and the FBI sponsored a study of the UCR Program, with the objective of revising it to meet law enforcement needs into the 21st century. A 5-year redesign effort to provide more comprehensive and detailed crime statistics resulted in the NIBRS program, which collects data on each reported crime incident. The UCR program is currently being expanded to NIBRS.

National Institute of Justice (NIJ)

www.ojp.usdoj.gov/nij

NIJ is the research, development, and evaluation agency of DOJ. NIJ provides objective, independent, nonpartisan, evidence-

based knowledge and tools to meet the challenges of crime and justice, particularly at the state and local levels. NIJ's principal authorities are derived from the Omnibus Crime Control and Safe Streets Act of 1968, as amended (see 42 USC §§ 3721-3723). The NIJ director establishes the agency's objectives, guided by the priorities of the Office of Justice Programs, DOJ, and the needs of the field. NIJ actively solicits the views of criminal justice and other professionals and researchers to inform its search for the knowledge and tools to guide policy and practice.

National Institute for Standards and Technology (NIST)

www.nist.gov/

Founded in 1901, NIST is a nonregulatory federal agency within the U.S. Commerce Department's Technology Administration. Its mission is to develop and promote measurement, standards, and technology to enhance productivity, facilitate trade, and improve the quality of life. NIST carries out its mission in 4 cooperative programs:

- The NIST laboratories conduct research that advances the nation's technology infrastructure and are needed by U.S. industry to continually improve products and services.
- The Baldrige National Quality Program promotes performance excellence among U.S. manufacturers, service companies, educational institutions, and health care providers, among other activities.
- The Manufacturing Extension Partnership, a nationwide network of local centers offers technical and business assistance to smaller manufacturers.
- The Advanced Technology Program accelerates the development of innovative technologies for broad national benefit by co-funding research and development partnerships with the private sector.

National Telecommunications and Information Administration (NTIA)

www.ntia.doc.gov

NTIA, an agency of the U.S. Department of Commerce, is the Executive Branch's principal voice on domestic and international telecommunications and IT issues. NTIA works to spur innovation, encourage competition, help create jobs, and provide consumers with more choices and better quality telecommunications products and services at lower prices.

NLETS—The International Justice and Public Safety Information Sharing Network

www.nlets.org

The mission of NLETS—The International Justice and Public Safety Information Sharing Network is to provide, within a secure environment, an international criminal justice telecommunication capability that will benefit, to the highest degree, the safety, security, and preservation of human life and the protection of property. NLETS will assist those national and international governmental agencies and other organizations with similar missions that enforce or aid in enforcing local, state, federal, or international laws or ordinances.

Office of Community Oriented Policing Services (COPS), U.S. DOJ

www.cops.usdoj.gov/

The COPS Office was created in 1994 to advance the practice of community policing as an effective strategy in communities' efforts to improve public safety. The COPS Office provides grants to tribal, state, and local law enforcement agencies to hire and train community policing professionals, acquire and deploy cutting-edge crime-fighting technologies, and develop and test innovative policing strategies. COPS-funded training helps advance community policing at all levels of law enforcement—from line officers to law enforcement executives—as well as others in the criminal justice field. Because community policing is by definition inclusive, COPS training also reaches state and local government leaders and the citizens they serve. This broad range of programs helps COPS offer agencies support in

virtually every aspect of law enforcement, and it is making America safer, one neighborhood at a time. COPS has invested \$11.3 billion to add community policing officers to the nation's streets and schools, enhance crime-fighting technology, support crime prevention initiatives, and provide training and technical assistance to advance community policing.

Office of Justice Programs (OJP), U.S. DOJ

www.ojp.gov/

Since 1984 OJP has provided federal leadership in developing the nation's capacity to prevent and control crime, improve the criminal and juvenile justice systems, increase knowledge about crime and related issues, and assist crime victims. OJP carries out its mission through the following bureaus and offices: the Bureau of Justice Assistance, the Bureau of Justice Statistics, the National Institute of Justice, the Office of Juvenile Justice and Delinquency Prevention, and the Office for Victims of Crime. Through the programs developed and funded by its bureaus and offices, OJP works to form partnerships among federal, state, and local government officials to control drug abuse and trafficking; reduce and prevent crime; rehabilitate neighborhoods; improve the administration of justice in America; meet the needs of crime victims; and address problems such as gang violence, prison crowding, juvenile crime, and white-collar crime. Although some research and technical assistance is provided directly by OJP's bureaus and offices, most of the work is accomplished through federal financial assistance to scholars, practitioners, experts, and state and local governments and agencies. Many of the program bureaus and offices award formula grants to state agencies, which, in turn, subgrant funds to units of state and local government. Formula grant programs in such areas as drug control and system improvement, juvenile justice, victims compensation, and victims assistance, are administered by state agencies designated by each state's governor.

Office of the Chief Information Officer, U.S. Department of Homeland Security

<http://www.dhs.gov/dhspublic/>

Pennsylvania Justice Network (JNET)

www.pajnet.state.pa.us

JNET is a collaboration of municipal, county, state, bordering states, and federal justice agencies that develop and provide a secure, online integrated justice system that allows participating agencies to access driver and offender records and other justice information. Local and state police officers, JNET's largest users, have immediate access to critical criminal justice information that helps them to perform their jobs more effectively.

Regional Information Sharing Systems (RISS)

www.iir.com/riss

BJA's RISS® Program is composed of six regional centers that share intelligence and coordinate efforts against criminal networks that operate in many locations across jurisdictional lines. Typical targets of RISS activities are drug trafficking, terrorism, violent crime, cybercrime, gang activity, and organized criminal activities. Each of the centers, however, selects its own target crimes and the range of services provided to member agencies.

Science and Technology Directorate, U.S. Department of Homeland Security (DHS)

www.dhs.gov/dhspublic/display?theme=53

The Science and Technology Directorate is the primary research and development arm of DHS. It provides federal, state, and local officials with the technology and capabilities to protect the homeland. Its strategic objectives are:

- Develop and deploy state-of-the art, high-performance, low-operating-cost systems to prevent, detect, and mitigate the consequences of chemical, biological, radiological, nuclear, and explosive attacks.
- Develop equipment, protocols, and training procedures for response to and recovery from chemical, biological, radiological, nuclear, and explosive attacks.
- Enhance the technical capabilities of the department's operational elements and

other federal, state, local, and tribal agencies to fulfill their homeland security related missions.

- Develop methods and capabilities to test and assess threats and vulnerabilities, and prevent technology surprise and anticipate emerging threats.
- Develop technical standards and establish certified laboratories to evaluate homeland security and emergency responder technologies, and evaluate technologies for SAFETY Act certification.
- Support U.S. leadership in science and technology.

SEARCH, The National Consortium for Justice Information and Statistics

www.search.org

SEARCH, The National Consortium for Justice Information and Statistics, is a nonprofit membership organization created by and for the states, dedicated to improving the quality of justice and public safety through the use, management, and exchange of information; application of new technologies; and responsible law and policy, while safeguarding security and privacy. SEARCH provides services—including technical assistance, training, information-modeling tools, systems and technology procurement and implementation, conferences, workshops and symposia, research, publications, surveys, policy analysis, and liaison—in the areas of integration and information-sharing, high-technology crime, and criminal history policies. Constituents served are local, tribal, county, regional, and state agencies and organizations, including law enforcement and public safety; first responders; prosecution and defense; adjudication; detention, corrections and probation; and other disciplines, such as transportation, drivers' licensing, vehicle registration, public health, and social services.

U.S. Department of Transportation (DOT)

<http://www.dot.gov/>

The mission of the U.S. DOT is to "Serve the United States by ensuring a fast, safe, efficient, accessible, and convenient transportation

system that meets our vital national interests and enhances the quality of life of the American people, today and into the future.” The Office of the Secretary of DOT oversees the formulation of national transportation policy, promotes intermodal transportation, negotiates and implements international transportation agreements, assures the fitness of U.S. airlines and enforces airline consumer protection regulations, issues regulations to prevent alcohol and illegal drug misuse in transportation systems, and prepares transportation legislation. Administrations and bureaus operating under DOT include the Federal Aviation Administration, Federal Highway Administration, Federal Motor Carrier Safety Administration, the Federal Railroad Administration, the Federal Transit Administration, The Maritime Administration, The National Highway Traffic Safety Administration, The Pipeline and Hazardous Materials Safety Administration, The Research and Innovative Technology Administration, The Saint Lawrence Seaway Development Corporation, and The Surface Transportation Board.

Washington Justice Information Network (JIN)

www.jin.wa.gov

The mission of JIN is to improve public safety by providing criminal justice practitioners with complete, timely and accurate information and to improve operating efficiency by facilitating the integration of disparate systems throughout the state. Its governance board includes key state and local members of the justice community. Using the principles of services-oriented architecture, JIN is designing and deploying a model for information sharing in Washington State’s justice community.