



United States
Department of Justice

The Justice Reference Architecture (JRA) Specification

Working Draft V 1.3

**by
The Global Infrastructure/Standards
Working Group**

December 4, 2006



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 the U.S. Department of Justice's Global Justice Information Sharing Initiative. The opinions, findings, and conclusions or recommendations expressed in this document are those of the authors and do not necessarily represent the official position or policies of the U.S. Department of Justice.

Table of Contents

Table of Contents	3
Acknowledgments	5
How to Use This Document	7
Executive Summary	8
Introduction	9
Global's SOA Initiative	9
An Interoperability Strategy	10
Consensus on the OASIS Reference Model for SOA	10
Creating the Justice Reference Architecture	11
What Is the Justice Reference Architecture?	11
Architecture Requirements.....	13
Justice Reference Architecture	19
Graphical Overview	19
Concepts and Relationships.....	21
OASIS Reference Model for Service-Oriented Architecture	21
Core Concepts—Services, Service Consumers, Capabilities, and Real-World Effects	22
Supporting Concepts.....	22
Business Process Models and the Service/Capability Hierarchy.....	25
Interaction, Visibility, Service Models, and Service Interfaces	25
Design and Description of Service Interfaces	28
Capabilities in Detail	30
Service Policies, Service Contracts, and Service Agreements	31
Execution Context.....	31
Illustration Scenarios	33
Elaboration of JRA Concepts	34
Other Considerations	40
Governance	40
Investment Strategies.....	41

System Design.....	41
Infrastructure Specifications.....	41
System Interfaces	41
Glossary	42
References.....	48
Document History	48

Acknowledgements

The Justice Reference Architecture (JRA) was developed through a collaborative effort of the Global Justice Information Sharing Initiative (Global), Office of Justice Programs (OJP), U.S. Department of Justice (DOJ).

Global aids its member organizations and the people they serve through a series of important initiatives. These include the facilitation of Global Working Groups. The Global Infrastructure/Standards Working Group (GISWG) is one of four Global Working Groups covering critical topics such as intelligence, privacy, security, and standards. The GISWG is under the direction of Tom Clarke, Ph.D., National Center for State Courts. The GISWG consists of three committees: Management and Policy, Services Interaction, and Services.

Although this document is the product of Global and its GISWG membership, it was adapted primarily from the technical reference architecture developed by the state of Washington, and sincere appreciation is expressed to Mr. Scott Came, State of Washington and SEARCH, The National Consortium for Justice Information and Statistics, for his guidance and leadership. In addition, parts of the architecture were derived from the Organization for the Advancement of Structured Information Standards (OASIS) Reference Model for Service-Oriented Architecture 1.0 (SOA-RM). Other major contributors include the OASIS Court Filing Technical Committee, OASIS SOA-RM Technical Committee, and the Messaging Focus Group.

Although each member of the GISWG is recognized for their contributions and for volunteering their time to the Justice Reference Architecture, Global would also like to recognize the members of the GISWG Executive Architecture Committee.

Mr. Scott Came—State of Washington and SEARCH, The National Consortium for Justice Information and Statistics, Services Interaction Committee

Dr. Tom Clarke—National Center for State Courts, Chair, GISWG

Mr. Scott Fairholm—National Center for State Courts, Chair, GISWG Services Committee

Mr. Dale Good—CriMNet Program, Chair, GISWG Management and Policy Committee

Mr. Kael Goodman—IJIS Institute, Chair, GISWG Services Interaction Committee

Mr. Ron Hawley—SEARCH, The National Consortium for Justice Information and Statistics, GISWG Management and Policy Committee Chair (2005-2006)

35 Mr. Eric Sweden—National Association for State Chief Information Officers, Vice
36 Chair, GISWG

How to Use This Document

Policymakers, Executives, and Decision Makers

Global is committed to providing Service-Oriented Architecture (SOA) resources, such as this document, to local, state, regional, tribal, and federal justice and public safety organizations. As additional resources become available, these materials will demonstrate the value of the architecture to the stakeholders in a way that is targeted to their particular needs. Other planned resources include strategy, executive summary, case studies from early implementers, management and policy, and other planning briefings, which will be targeted towards managers, chiefs, and executives.

For the purposes of this document, Global has selected a distinguished group of technical and domain representatives from a group of skilled peers who have volunteered to develop this material as a starting point in establishing the Justice Reference Architecture Specification for Service-Oriented Architecture.

Keep in mind that the sections in this document referencing the conceptual diagram, high-level components, and relationships establish definitions that are intended for use by technical architects and project managers who are responsible for identifying all the elements necessary within their jurisdiction to implement SOA. **This document is intended as a formal and complete architectural specification for people with previous knowledge of technical architecture, service-oriented architecture, and supporting industry standards (such as Web services).**

Project Managers, Architects, and Technologists

This report is intended as a resource for a technical audience, including Global Justice XML Data Model (Global JXDM) implementers, architects, developers, system integrators, and other justice and public safety technical practitioners. It provides the background and concepts—a strong foundation—required for the implementation of SOA. Justice Reference Architecture is a new term coined for the justice community, and it is derived from the OASIS Reference Model for Service-Oriented Architecture 1.0 (SOA-RM¹). The reader should refer to the SOA-RM for more detailed information about many of the concepts in this document. JRA is intended to facilitate your SOA implementation by establishing a common language that can be used to exchange data with partner organizations.

¹ <http://docs.oasis-open.org/soa-rm/v1.0/soa-rm.pdf>

71

Executive Summary

72 This document states a set of requirements for justice interoperability and then
73 describes the Justice Reference Architecture (concepts, relationships, and high-level
74 components) Specification that satisfies those requirements. The document then
75 illustrates the architecture through a set of actual scenarios. Finally, the document
76 provides an initial elaboration of some of the concepts and components in the
77 architecture. (This section will be significantly expanded in future versions.)

78

79

Introduction

Global's SOA Initiative

On September 29, 2004, the Global Justice Information Sharing Initiative (Global) Advisory Committee (GAC) unanimously adopted **SERVICE-ORIENTED ARCHITECTURE** (SOA) and the recommendations in the report titled *A Framework for Justice Information Sharing: Service-Oriented Architecture (SOA)*.

Global provides support for SOA by:

- Recognizing SOA as the recommended **FRAMEWORK** for development of justice information sharing systems;
- Promoting the utility of SOA for the justice community; and
- Encouraging the members of the justice community to take these recommended incremental steps in the development of their own systems.

Global's approval was based on the understanding that SOA is an approach that is most likely to result in an infrastructure that will support its vision of how information should be shared among the justice community. If SOA is to be used successfully as the framework for justice information sharing **ARCHITECTURE**, Global must play a proactive leadership role in several areas. The development of the **JUSTICE REFERENCE ARCHITECTURE** was based on the following actions recommended by Global.

- Incorporate SOA into the activities of all of the Global Working Groups. SOA raises issues for security, privacy and information quality, and intelligence that will be given explicit attention and treated as part of a broad initiative.
- Encourage the creation of a mechanism for drawing together the experiences and lessons from the field.
- Reach out to existing national systems to incorporate their efforts into the design of an overall strategy.
- Address the following six issues as priorities—services, standards, interagency agreements, registries, security, and privacy and data quality—because they will be a major part of the agenda for the next set of Global activities.
- Develop a multitiered strategy for the public sector to influence standards. It will include encouraging the creation of a public process (as it did with XML), taking part in industry groups that are developing standards relevant to justice (e.g., OASIS), and developing partnership processes with industry and other public entities.

118 **An Interoperability Strategy**

119 Solving interoperability challenges continues to be a significant problem and a high
120 priority for the justice and public safety community. There are approximately
121 100,000 justice agencies that have the critical need to share information across their
122 various information systems, and this variety creates multiple layers of
123 interoperability problems because hardware, software, networks, and business rules
124 for data exchange are different. The need for information sharing has led to this
125 interoperability strategy and the Justice Reference Architecture (JRA).

126 The strategy for developing JRA involves many steps. This paper details some
127 highly technical and abstract concepts. Understanding these concepts may require
128 significant effort from the reader. Though it may seem strategically questionable to
129 place such a high hurdle at the beginning of a multistep process, doing so actually
130 creates a flexible vocabulary and conceptual framework that will enable the desired
131 interoperability to flourish. Additionally, subsequent steps that will build from this
132 framework will be incrementally more concrete, and will ultimately lead to actual
133 implementation specifications that can be used by practitioners in the field. Global
134 believes that this dynamic interoperability strategy will help to prevent
135 incompatibilities, guide vendors and organizations on how to fit components
136 together, and facilitate communication and interoperability among disparate
137 communities.

138 Global's strategy for JRA, like other work that has preceded it, follows a five-step
139 process:

- 140 **Step One: Agree on common concepts**
- 141 **Step Two: Agree on the relationships and deliverables**
- 142 **Step Three: Assign the work**
- 143 **Step Four: Produce the deliverables**
- 144 **Step Five: Revise the deliverables**

145 As an example, when the Global JXDM project started it had a small set of limited
146 solutions. Through much iteration, Global JXDM has been expanded and refined
147 and addresses a successively larger set of justice domains.

148 **Consensus on the OASIS Reference Model for SOA**

149 One of the justice requirements is to create a common language for talking about
150 architecture across major domains. For instance, it is currently difficult for
151 emergency management personnel to talk to justice personnel about how their
152 respective systems might share data beyond the content standards issue because
153 their ways of communicating about architecture are so different.

154 After considerable discussions among the stakeholders, Global adopted the
155 Organization for the Advancement of Structured Information Standards (OASIS)
156 Reference Model for Service-Oriented Architecture 1.0 (SOA-RM). OASIS has
157 approved this standard reference model for describing different architectures using
158 comparable, vendor-neutral language. Global is adopting the OASIS framework for
159 describing its architecture and holding conversations with other domains.

160 **Creating the Justice Reference Architecture**

161 It is important to note that SOA-RM provides a conceptual foundation for not only
162 the justice community, but for any domain to create a **REFERENCE ARCHITECTURE**.
163 JRA builds on the SOA-RM concepts by specifying additional relationships and
164 defining and specifying these adopted concepts.

165 Although there is no perfect solution, and since there is a need to start somewhere,
166 SOA-RM is recommended as the best place to start Global's SOA work efforts.
167 Global began by mapping the SOA components, documenting and leveraging the
168 work that has been already done—like the Global JXDM—and, finally, identifying
169 and filling the gaps.

170 **Justice Reference Architecture is derived from the OASIS**
171 **Reference Model for Service-Oriented Architecture 1.0. The**
172 **OASIS work was developed to provide a conceptual**
173 **foundation for creating a reference architecture. As intended**
174 **by OASIS, JRA builds on or expands from the OASIS model.**
175

176 Specifically, Global is developing a modular architecture that cleanly and
177 appropriately identifies and separates technical and governance layers so that
178 standards can be developed to improve interoperability.

179 **What Is Justice Reference Architecture?**

180 This section defines Justice Reference Architecture (JRA) for Service-Oriented
181 Architecture (SOA) and explains why a reference architecture is useful. Keep in
182 mind that there are potentially many justice reference architectures, but that this JRA
183 focuses entirely on SOA for the justice and public safety community. Out-of-scope
184 components and other considerations are listed on page 40.

185

186

187

188

189

JRA is an abstract framework for understanding significant components and the relationships between them within a Service-Oriented Architecture. It lays out common concepts and definitions as the foundation for the development of consistent SOA implementations within the justice and public safety communities.

190

191

192

193

194

195

196

197

198

JRA is a description of the important concepts in a justice information sharing architecture and the relationships between those concepts. JRA also identifies, at a high level, the kinds of “components” (software systems, hardware infrastructure, policies, practices, intersystem connections, and so on) necessary to bring those concepts to life in a particular context. JRA is generally not specific enough to govern the implementation of any individual software system implementation. Rather, it is a framework for guiding implementations in general, with the aim of standardizing or harmonizing certain key aspects of those implementations to support reusability or interoperability.

199

200

201

202

It is important to note that at this time JRA is not complete. Many sections of this document are still under development, but the document does attempt to identify the necessary concepts, relationships, and components that will require further elaboration and/or implementation.

Architecture Requirements

203

204 This section documents the business requirements to be addressed and satisfied by
205 Justice Reference Architecture. In future revisions, this section will be changed from
206 requirements to guiding principles and goals.

207 As previously described in the Introduction, the justice world has close to 100,000
208 justice agencies, and most of these are very small and have few information
209 technology resources. They use different applications, hardware, and networks that
210 have diverse topologies and interoperability capabilities. Nonetheless, JRA must
211 reflect the influence of the following factors, representing the key characteristics of
212 the justice and public safety environment.

213 **Requirement 1—Justice Reference Architecture must recognize**
214 **innumerable independent agencies and funding bodies from local, state,**
215 **tribal, and federal governments.**

216 For anyone connected to the justice community, this requirement is self-evident.
217 One factor has not changed throughout American history: the business of justice is
218 largely the province of local, state, and tribal government. The independence and
219 number of entities that need to share justice information is almost overwhelming.
220 Certainly, it is beyond the ability of existing conceptual frameworks, computer
221 models, financial resources, or jurisdictional authority to create an integrated network
222 using traditional technology. SOA, however, can be a meaningful bridge. A quote
223 from SOA literature makes this fit clear: “Designing for SOA involves thinking of the
224 parts of a given system as a set of relatively autonomous services, each of which is
225 (potentially) independently managed and implemented, which are linked together
226 with a set of agreements and protocols into a federated structure.” [Sholler]
227 “Autonomous,” “independent,” “agreements,” and “federated” capture the
228 environment for justice information sharing.

229 **Requirement 2—JRA must accommodate information sharing across**
230 **agencies that represent divergent disciplines, branches of government,**
231 **and operating assumptions.**

232 It is difficult, if not impossible, to define precisely the boundaries of the justice
233 community. The obvious list of participants—law enforcement, prosecution, courts,
234 defense counsel, probation, and corrections—is only the beginning. Accurate,
235 timely, and appropriate justice information sharing among the entities is necessary
236 for effective apprehension, prosecution, adjudication, and punishment of an
237 offender. However, these are only some of the objectives.

238 This same information, or portions of it, are necessary to meet the business
239 requirements of related justice, public safety, and homeland security agencies. For
240 example, this information is required to regulate the sale of firearms; complete

241 criminal background checks of employees at schools, child care services, and elder
242 care facilities; identify aliens who have been convicted of crimes or have entered the
243 country illegally; notify the local community of the release and location of sexual
244 predators; prevent training in the operation of aircraft by aliens or other designated
245 individuals who may present a risk to aviation and national security; do background
246 checks of those transporting hazardous materials; or create information models to
247 provide information and predict the spread of disease and its effects, and decide on
248 countermeasures for potential health epidemics like the avian flu.

249 The events of September 11, 2001, resulted in the creation of the
250 U.S. Department of Homeland Security (DHS) with its constituent agencies, such as
251 the U.S. Citizenship and Immigration Services, U.S. Customs and Border Protection,
252 and the U.S. Coast Guard. September 11 also elevated the importance of
253 information sharing between and among public safety agencies such as fire,
254 emergency medical services, and other first-responder organizations.

255 The list would not be complete without the recognition of the numerous entities
256 outside of the justice and public safety communities—such as schools, child care
257 services, transportation, and licensing agencies—that need critical justice-related
258 information to perform daily business activities, such as hiring new personnel,
259 approving gun purchases, or granting professional licenses.

260 Finally, the list of relevant constituencies also includes the public, who expect greater
261 accountability and access to justice information that is considered sensitive or
262 protected by privacy laws in some settings (e.g., state criminal history records in
263 many state repositories and the FBI system), while viewed as public record in others
264 (e.g., criminal history record information in the courts). Increasingly, the public also
265 expects that this access be automated and online.

266 The diversity of justice information consumers carries an attendant consideration:
267 different types of users have different requirements. A judge making a sentencing
268 decision has more time for their task—and a less expedited need for response to
269 inquiry—than an officer on the scene requiring instant access to succinct information.

270 The purposes also vary. For example, it is one thing if the primary objective is to
271 validate the identity or status of an individual (e.g., a law enforcement officer
272 communicating with the Department of Motor Vehicles to check on a driver's
273 license), but another when an exhaustive search for information is required (e.g., a
274 probation officer conducting a pre-sentence investigation of a convicted offender).

275 Different sources also mean differences in expectations about who can use what
276 information. Privacy and data quality issues, which are demanding enough when
277 dealing with a single information system, grow exponentially when dealing with
278 different disciplines. It is one thing to share the records of a criminal sentencing
279 hearing held in open court; it is quite another when dealing with health records or an

280 ongoing criminal investigation. Incomplete or inaccurate data may be an annoyance
281 if the task is to identify leads for subsequent investigations; they are a different issue
282 entirely if they prohibit one from getting a job, traveling on an airplane, or lead to
283 incarceration. Working documents in one setting can become dispositive evidence in
284 another.

285 What this means is that the information system design cannot begin with a clear
286 definition of the boundaries of the organization. Nor can we assume that all of those
287 who participate share a common set of objectives or an understanding of the
288 process. On the contrary, the information system design must assume diversity, even
289 conflicts, in the operating procedures and objectives of the participating
290 organizations.

291 **Requirement 3—JRA must be able to accommodate an infinite range of**
292 **scales, from small operations with few participants in a rural county to**
293 **national processes that reach across local, state, federal, and even**
294 **international boundaries.²**

295 The context for information sharing is not the same everywhere, and the scale will
296 depend upon the objectives and the geographical setting. It is one thing if the
297 objective is to move cases quickly from investigation to arrest through adjudication in
298 a rural county where all of the participants know each other and have ongoing
299 contact on a personal level. It is quite another thing if the objective is to share
300 information about warrants between law enforcement and the judiciary in a large
301 state on a real-time basis. And it is different still if the context moves to a national
302 level, and the objective is to share information among many local, state, tribal, and
303 federal law enforcement and health agencies about a reported health epidemic.

304 The resources required to implement advanced justice information sharing
305 architectures will come from many independent sources, the largest body of which
306 will be local. It is safe to assume that the funds will be spent to meet the immediate
307 needs of the entities within the funding source's jurisdiction and not as a result of
308 priorities that are provided by a state or national plan. An approach to infrastructure
309 design that cannot be adapted to the different scales without losing its internal
310 integrity will quickly be marginalized.

² For clarity, we have changed the original language in the documents to fit the current terminology that is based on the OASIS and JRA work efforts. This current work is based on the requirements from the document titled, A Framework for Justice Information Sharing: Service-Oriented Architecture (SOA), December 9, 2004, which was written by The Global Infrastructure/Standards Working Group.

311 **Requirement 4—JRA must be able to accommodate data sources that**
312 **differ widely in software, hardware, structure, and design.**

313 The history of efforts to develop integrated information systems among local criminal
314 justice agencies around a single hardware and software platform is large and filled
315 with many disappointments. When the focus shifts to the state and national level,
316 the success rate becomes even smaller and is largely populated by single-purpose
317 efforts. The explanation for this phenomenon is relatively simple: technology
318 investment decisions are made by funding sources with their own tax base, budget
319 cycle, and spending priorities. The result is that information system development
320 among local, state, tribal, or federal justice community entities rarely occurs in
321 concert.

322 The reality is that no infrastructure development strategy can assume that all
323 participants will be at the same point in the technology cycle. To paraphrase: new
324 technologies are important, but legacy systems will always be with us.

325 **Requirement 5—JRA must reflect and incorporate the lessons and**
326 **developments of the private sector.**

327 It often surprises the justice community to learn how much of the technology needed
328 to share information is common to the private sector as well. When you think about
329 it, only parts of the data and the transaction definitions are unique to the justice
330 world. The several other technical layers in a transaction that provides a service are
331 driven by open standards defined by private industry and implemented in their tool
332 sets and products. The justice community must learn how to incorporate and
333 leverage private industry.

334 The Global process and the projects sponsored by it must take these powerful trends
335 in the private sector into account. The justice community can have some influence
336 on such decisions, even in the private sector, by more fully participating in the open
337 standards bodies that decide what will be proposed to the market for
338 implementation; continuing collaboration with industry partners such as the IJIS
339 Institute will be necessary to succeed. Often, such participation and collaboration
340 will educate us on how to develop and/or reuse the standards without needing to
341 invent something new and unique for our business problems. And, as Global puts
342 together an agenda for progress, lessons learned are provided from initiatives that
343 have failed as well as succeeded. These discoveries and lessons learned from the
344 private sector will save us money and facilitate the sharing of critical data in ways
345 that increase public safety.

346 **Requirement 6—JRA must be dynamic and capable of evolving as the**
347 **information sharing requirements change and the technology is**
348 **transformed.**

349 The operational requirements of members of the justice community are in constant
350 change. The events of September 11 have elevated intelligence information to a
351 leading priority for law enforcement; the rise of domestic violence cases has
352 expanded the judiciary's need to reach out to the family services community; the
353 increased mobility of the population has complicated probation's efforts to monitor
354 offenders; and the spread of AIDS has put a premium on health management by
355 corrections administrators. An infrastructure design that cannot adapt to an evolving
356 definition of the boundaries and critical components of the justice community will,
357 before long, become irrelevant.

358 **Requirement 7—JRA should leverage open industry standards where**
359 **possible.**

360 The justice environment will benefit from the stabilization of standards as the basis
361 for an overall approach to interoperability among large and diverse organizations.
362 The evolution of open industry standards for systems integration has reached a point
363 where these standards will facilitate interoperability. Many prominent programming
364 languages, software development environments, packaged applications, and
365 integration platforms/tools support the standards. Although some common
366 integration needs are met by competing standards, the number and significance of
367 competing standards continue to shrink.

368 **Requirement 8—JRA must support marketplace diversity.**

369 The marketplace for integration products is highly diverse and is likely to remain so
370 for the foreseeable future. Support for Web services standards, key integration
371 capabilities (such as transformation, content-based routing, and orchestration), and
372 off-the-shelf adapters for applications (such as Enterprise Resource Planning [ERP]
373 packaged applications) exist from a variety of vendors.

374 **Requirement 9—JRA should use a service-oriented design philosophy.**

375 **Requirement 10—JRA should be driven by business need.**

376 **Requirement 11—JRA should derive service requirements from business**
377 **process requirements.**

378 **Requirement 12—JRA should preserve data control by the source**
379 **organization.**

380 **Requirement 13—JRA should minimize dependencies among justice**
381 **business processes and supporting information systems.**

382 **Requirement 14—JRA should treat services as reusable assets to be**
383 **shared beyond the original context as required.**

384 **Requirement 15—JRA should support business agility as the fundamental**
385 **business requirement.**

386 **Requirement 16—JRA should be developed in an iterative way.**

387 **Requirement 17—JRA should evolve indefinitely in response to changing**
388 **business requirements.**

389

390

Justice Reference Architecture

391

Graphical Overview

392

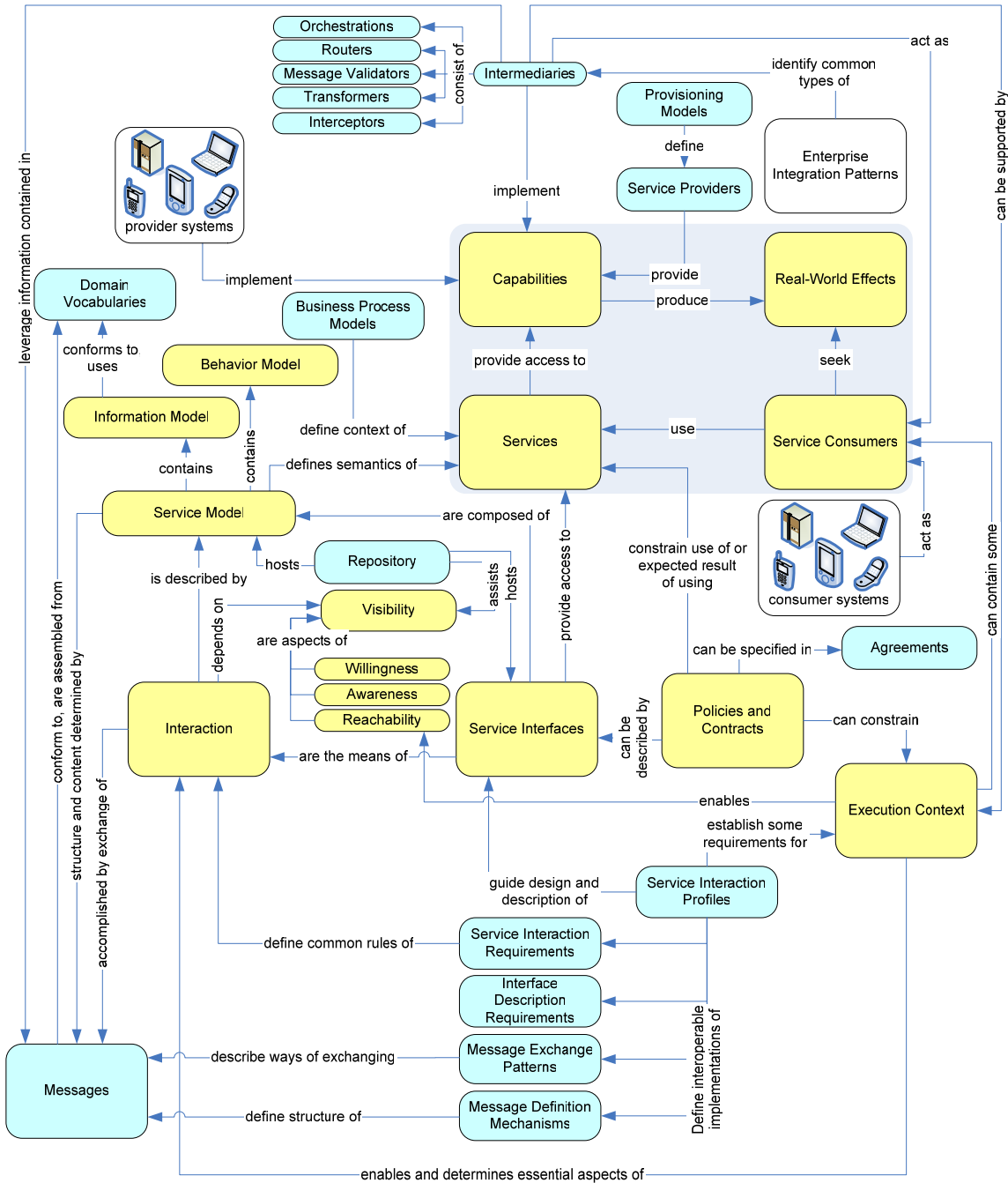
The following diagram depicts the concepts, high-level components, and

393

relationships in the Justice Reference Architecture Specification V 1.3. These

394

elements are described in detail in the following sections.



Justice Reference Architecture
Concept Map
November 3, 2006

Legend	
[Yellow Box]	Concepts from OASIS SOA-RM
[Cyan Box]	Concepts particular to the JRA

395
396
397

398 **Concepts and Relationships**

399 The following sections describe the concepts, components, and relationships
400 depicted in the diagram on the previous page.

401 **OASIS Reference Model for Service-Oriented Architecture for**

402 The Justice Reference Architecture depicted in the diagram above (and defined in
403 this document) adopts and builds on the OASIS SOA-RM.

404 The SOA-RM defines its purpose as follows:

405 “A **REFERENCE MODEL** is an abstract framework for understanding
406 significant relationships among the entities of some environment. It
407 enables the development of specific reference or concrete
408 architectures using consistent standards or specifications supporting
409 that environment. A reference model consists of a minimal set of
410 unifying concepts, axioms, and relationships within a particular
411 problem domain and is independent of specific standards,
412 technologies, implementations, or other concrete details.” (**SOA-**
413 **RM**, p. 4)

414 “The goal of this reference model is to define the essence of service-
415 oriented architecture and emerge with a vocabulary and a common
416 understanding of SOA. It provides a normative reference that remains
417 relevant for SOA as an abstract and powerful model, irrespective of
418 the various and inevitable technology evolutions that will impact
419 SOA.” (**SOA-RM**, p. 4)

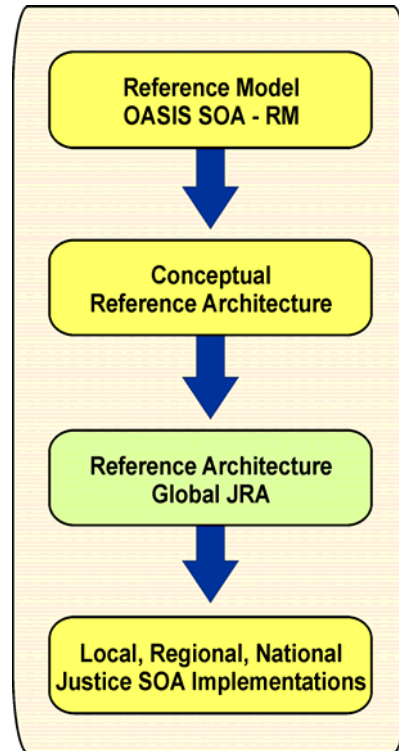
420 While the SOA-RM is a powerful model that provides a vendor-neutral, open-
421 standard definition of service-oriented architecture, its abstract nature means that
422 further work must be done to create reference architecture. OASIS lays out a
423 roadmap for the creation of such reference architecture. **Specifically, OASIS**
424 **recommends that the reference model guide the development of an**
425 **architecture; that protocols, profiles, specifications, and standards are**
426 **considered; and that requirements, motivations, and goals are taken into**
427 **account.** (**SOA-RM**, p. 5)³

428 **JRA does just this. It takes the reference model and adds the protocols,**
429 **profiles, specifications, standards, requirements, motivations, and goals**
430 **appropriate for the justice community.**

³ Note: In the next version of the JRA specification, this paragraph will be clarified to identify where the SOA-RM stops, and where JRA begins.

431 In the JRA diagram, OASIS SOA-RM concepts are
 432 shaded yellow with a dashed line as the border.
 433 Concepts and components particular to the
 434 conceptual JRA defined by this document are shaded
 435 light blue with a solid border. Relationships between
 436 concepts (indicated by arrows) are defined in the
 437 SOA-RM if the arrows connect concepts shaded
 438 yellow. Relationships between cyan-shaded concepts
 439 or between cyan-shaded and yellow-shaded
 440 concepts are particular to JRA.

441 The descriptions of SOA-RM concepts provided in
 442 the following sections are intended to be brief
 443 summaries; consequently, they omit certain details
 444 that appear in the SOA-RM. Concepts listed in bold,
 445 blue caps are listed in the glossary at the end of this
 446 document, and the glossary contains definitions of
 447 the SOA-RM concepts, which are repeated from the
 448 SOA-RM glossary for convenience. The SOA-RM
 449 itself is the primary source for full exposition of
 450 SOA-RM concepts and the relationships between them.



451 **Core Concepts—Services, Service Consumers, Capabilities, and Real-** 452 **World Effects**

453 *These four concepts make up the core of the Global JRA. All other concepts support*
 454 *these concepts. It is strongly advised that these concepts be clearly grasped before*
 455 *reading the section called Supporting Concepts.*

456 JRA begins from the premise that a group of justice partners have **CAPABILITIES** that
 457 they provide to one another. These capabilities “solve or support a solution for the
 458 problems [businesses] face in the course of their business.” (**SOA-RM**, p. 8) That is,
 459 capabilities are the things organizations have to solve problems and therefore add
 460 value, directly or indirectly, to their stakeholders.

461 Note that JRA is generic enough to support virtually any kind of capability.
 462 However, the purpose of JRA is to describe an approach to achieving
 463 interoperability among automated, computer software-based information systems.
 464 Therefore, JRA considers only those business capabilities that are provided by (or
 465 implemented by) information systems. JRA calls these systems **PROVIDER SYSTEMS**
 466 and establishes that provider systems implement capabilities.

467 Each capability produces one or more **REAL-WORLD EFFECTS**, each of which is an
 468 outcome of the business value sought by one of the partners. A real-world effect can
 469 be either the obtaining of information, the changing of something of business

470 relevance to the participating partners, or both. Because JRA establishes that
471 capabilities are implemented by provider systems, real-world effects consist of the
472 functional business requirements of provider systems. That is, real-world effects in
473 JRA are essentially the information made available by provider systems or the
474 outcomes resulting from business processes and workflows automated by provider
475 systems, or both.

476 In a service-oriented architecture, a **SERVICE** is the way in which one partner gains
477 access to a capability offered by another partner. A partner that uses a service to
478 gain access to another partner's capability is called a **SERVICE CONSUMER**. As with
479 capabilities, the architecture is generic enough to support virtually any kind of service
480 consumer. However, since the purpose of JRA is to describe an approach to
481 information systems interoperability, JRA narrows the SOA-RM definition of service
482 consumer to information systems that interact with services directly through an
483 interface that conforms to a service interaction profile (as defined below). JRA calls
484 such systems **CONSUMER SYSTEMS**.

485 One of the most important features of JRA is the separation of consumer systems
486 from provider systems by services in the middle. This is the defining characteristic of
487 a service-oriented architecture and is the key to decoupling systems as called for in
488 many of the Architecture Requirements listed in the section on page 13.

489 The fact that information sharing is one kind of real-world effect allows the
490 architecture to support the traditional view of system integration as "data exchange"
491 or "information sharing." JRA improves this view by encouraging systems to share
492 information in a way that minimizes the dependencies of each system on the
493 implementation of other systems.

494 **Supporting Concepts**

495 A **PROVISIONING MODEL** determines the organizational (perhaps contractual or legal)
496 responsibility for providing a capability, via services, to achieve consumers' desired
497 real-world effect. The entity identified in a provisioning model as responsible for
498 providing a capability is called a **SERVICE PROVIDER**.

499 **SERVICE DESIGN PRINCIPLES**⁴ provide consistent guidance regarding the overall
500 partitioning of capabilities into services and the relationships between services. For
501 instance, service design principles may call for services to represent one concise, self-
502 contained function and may also suggest that services should completely hide the
503 implementation details of the capabilities to which they provide access.

⁴ Principles and guidelines are important components of the conceptual JRA; however, these principles and guidelines are not illustrated on the diagram because they will exist for most of the components.

504 There is a wide variety of ways in which a service can provide access to a capability.
505 In some cases, the provider system that implements the capability may already
506 expose all or some of its functionality as services (through one or more service
507 interfaces, described on page 28). In other cases, the business partner that
508 provisions the capability can purchase an off-the-shelf adapter from the provider
509 system vendor (or a third party) that exposes the system's functionality as a set of
510 services. Finally, the provider system may require reimplementation or custom
511 adaptation to expose functionality as services. This is often expensive and risky, and
512 the desire to avoid this situation should be addressed in the Service Design
513 Guidelines.

514 In general, a given information system can be both a provider system and a
515 consumer system. Similarly, a particular business organization may offer capabilities
516 to its partners and, at the same time, be a consumer of the capabilities offered by
517 others. This has important implications for how the organization should conceive
518 and describe its information systems assets and how it assigns responsibilities for the
519 maintenance and support of those assets. For example, in the past it was common
520 to think of systems as having "client" and "server" components (or "browser" and
521 "server" components), which in turn influenced thinking about systems deployment,
522 networking, security, support, and a range of other issues. These issues deserve
523 reconsideration in an architecture in which a system or system component can be
524 both a "client" (consumer of services) and "server" (provider of services) at the same
525 time. The discussion of service interaction on page 25, and the subsequent
526 elaboration of interaction mechanisms in future iterations of JRA, will reflect the
527 impact of these issues.

528 Note that the concept of a service in JRA does not equate to a "Web service." The
529 term "Web services" is a label for a family of standards and an associated technical
530 approach to communicating between service consumers and services. The
531 architecture supports flexibility in how this communication happens through the
532 notion of service interaction profiles (discussed on page 29). A Web service profile
533 will be developed for the Web services family of standards; however, JRA will
534 include additional profiles that adopt other communication mechanisms, such as
535 MQ, JMS, and ebXML (discussed on page 37).

536 **Business Process Models and the Service/Capability Hierarchy**

537 The previous section described the basic concepts involved in the integration of
538 provider systems and consumer systems. In short, consumer systems seek a real-
539 world effect provided by a capability, and they produce that effect by accessing a
540 service that provides access to the capability. However, these concepts by
541 themselves do not provide the context for the integration of a particular consumer
542 and particular provider. That is, the concepts do not provide a way of describing
543 why a consumer seeks the effect made available by a provider through a service.

544 A **BUSINESS PROCESS MODEL** provides this contextual justification. A business
545 process model is a description (usually formal and often graphical) of a series of
546 activities that culminate in the achievement of some outcome of business value.
547 Some (but not necessarily all) of the steps in this series of activities involve producing
548 a real-world effect provided by a capability, and some of the steps require a
549 consumer to use a service. Each one of these steps, then, provide the contextual
550 justification for service interaction between a particular consumer and particular
551 provider.

552 The execution of the steps described in a business process model can be considered
553 a capability in and of itself. In addition, each of the steps in a business process
554 model can unfold into yet another business process model at a more focused level of
555 detail. In this way, each step in a series of service interactions can itself be a series of
556 service interactions. And, in theory, this recursion of models can go on forever,
557 though in practice it rarely exceeds three or four levels of containment. So, services
558 and capabilities form a hierarchy, where a service provides access to a capability
559 whose real-world effect is to accomplish the coordination of multiple services at a
560 lower level of detail.

561 JRA supports this hierarchy through orchestrations and intermediaries, discussed on
562 page 30.

563 It is important to note that a given service may play a role in multiple business
564 processes. In fact, reuse of services across business processes is an important part of
565 the value of a service-oriented architecture.

566 **Interaction, Visibility, Service Models, and Service Interfaces**

567 Services define what features of a provider system the system owner makes
568 accessible to business partners. Services also provide a logical description of the
569 information exchanged between consumer and provider systems as the consumer
570 accesses the capability.

571 *Interaction*

572 JRA refers to a consumer's accessing the features of a capability through a service as
573 **INTERACTION**, defined as "the performing [of] actions against a service." (**SOA-RM**,
574 p. 15) Service interaction generally involves the exchange of information between
575 the consumer and the service.

576 Interaction depends on two things. First, the designers of potential consumers need
577 to be able to find services and, once found, establish a physical interaction
578 mechanism with them. These needs are addressed by the concept of **VISIBILITY**.
579 Second, the designers of potential consumers need a description of the actions that
580 can be performed on a service, as well as the structure and meaning of information
581 exchanged during the interaction. These needs are addressed by the concept of a

582 service's **INFORMATION MODEL** and **BEHAVIOR MODEL**, collectively called **SERVICE**
583 **MODELS** in the JRA.

584 *Visibility*

585 Visibility, as the name implies, defines how service consumers and the providers of
586 capabilities “see” each other in a way that enables interaction between them. JRA
587 identifies three aspects of visibility.

- 588 • A service consumer must have information that makes it aware of
589 the existence of a service; the possession of this information is
590 called **AWARENESS**.
- 591 • The service (or capability accessed through the service) must be
592 willing to interact with the consumer; this is called **WILLINGNESS**.
- 593 • The consumer and service must be able to communicate with one
594 another through some kind of communication path or channel; the
595 existence of such a communication path is called **REACHABILITY**.

596 In JRA, a **REPOSITORY** will support awareness by hosting service models and service
597 interfaces. “Hosting” in this context means storing models and interface descriptions
598 in a central location that is accessible to appropriate stakeholders. A repository will
599 permit searching for models and interface descriptions based on a range of
600 identifying criteria. A repository will also map logical service identifiers with physical
601 addresses. When a consumer wishes to communicate with a service (identified by a
602 logical identifier), the consumer queries the repository for the physical address
603 associated with the service’s logical identifier. This decouples the consumer from the
604 physical location of a service at any point in time, thereby permitting the physical
605 relocation of the service without impacting the implementation of the consumer.

606 The concept of willingness is related to authorization and access control policies, in
607 that a common reason for lack of willingness to interact is that the consumer is not
608 authorized to conduct the requested interaction. Willingness often manifests in
609 service descriptions, as well as policies, contracts, and agreements (discussed on
610 page 31). A **SERVICE MODEL** is defined as the information needed in order to use,
611 or consider using, a service.

612 The concept of reachability is closely related to the concept of execution context
613 (discussed on page 31).

614 *Service Models*

615 Service models, consisting of a service’s information and behavior models, define the
616 semantics of interaction with the service. The behavior model defines the actions
617 that can be performed on the service; that is, it defines what the service “does.” The

618 information model describes the information that consumers exchange with the
619 service in the course of performing those actions.

620 Note that the SOA-RM considers the orchestration and choreography of multiple
621 services to be “part of the **PROCESS MODEL** of a given architecture.” Yet the SOA-
622 RM also indicates that a process model (part of the behavior model) applies to a
623 single service. (**SOA-RM**, p. 15) Because of this lack of clarity in the SOA-RM, this
624 JRA defines orchestration as a type of capability that leverages other services; it is
625 described on page 30.

626 In general, service models will be described at conceptual and logical levels of detail.
627 (Service models have a physical manifestation as well, in the form of the service
628 interface discussed in the next section.) A conceptual description of a service model
629 will typically describe, in prose text form, the capability to which the service provides
630 access, a listing and brief textual description of each action, and a brief textual
631 description of the information model (e.g., key information entities, key properties on
632 those entities, and brief definitions). A logical description of a service model will
633 describe the actions and information structures in detail but independent of any
634 physical implementation mechanism. Often this description will be graphical and
635 follow a standard diagramming or modeling technique, such as Uniform Modeling
636 Language (UML).

637 A **MESSAGE** is defined as the entire “package” of information sent between service
638 consumer and service (or vice versa), even if there is a logical partitioning of the
639 message into segments or sections. For instance, if an interface expresses actions as
640 operations or functions that take arguments, and a particular operation has two
641 arguments, both arguments would be considered part of the same message, even
642 though they may be logically separated within the message structure. A message
643 also includes the concept of an “attachment,” in which there are several additional
644 sections (attachments) that relate to a distinct, “primary” section.

645 In JRA, the exchange of messages is the only way in which consumers and services
646 can communicate. This establishes a linkage between the Federal Enterprise
647 Architecture Data Reference Model (FEA DRM) and JRA: a message in JRA equates
648 to an Information Exchange Package (IEP) in the DRM.

649 The concept of **DOMAIN VOCABULARIES** in JRA includes canonical data models,
650 data dictionaries, and markup languages that standardize the meaning and structure
651 of information for a topical or business domain. Domain vocabularies can improve
652 the interoperability between consumer and provider systems by providing a neutral,
653 common basis for structuring and assigning semantic meaning to information
654 exchanged as part of service interaction. Domain vocabularies can usually be
655 extended to address information needs specific to the service interaction or to the
656 business partners integrating their systems.

657 **SERVICE MODELING GUIDELINES** govern the style, structure, and description of
658 service models.

659 As previously stated, a repository should contain service model description artifacts
660 for each level of detail. The availability of service model descriptions to consumer
661 system designers, implementers, and purchasers is a key factor in establishing
662 visibility and the reuse of services.

663 *Service Interface*

664 Service models describe the actions available from a service and the information
665 exchanged between a consumer and the service during the performance of those
666 actions. In this way, the service models describe the “what” of interaction.

667 A **SERVICE INTERFACE** “is the means for interacting with a service. It includes the
668 specific protocols, commands, and information exchange by which actions are
669 initiated [on the service].” (**SOA-RM**, p. 22) A service interface is what a system
670 designer or implementer (programmer) uses to design or build executable software
671 that interacts with the service. That is, the service interface represents the “how” of
672 interaction.

673 JRA considers the service interface to be the physical manifestation of the service
674 models. Best practices call for a service interface to be described in an open-
675 standard, referenceable format (that is, a format whose contents are capable of
676 automated processing by a computer).

677 Note that at least some policies and contracts can be described in a service’s
678 interface.

679 The format, structure, and allowable contents of a service interface are established by
680 **INTERFACE DESCRIPTION REQUIREMENTS**, described in the following section.

681 **Design and Description of Service Interfaces**

682 JRA identifies four architectural elements that guide the design and description of
683 service interfaces.

684 **SERVICE INTERACTION REQUIREMENTS** define common rules of service interaction.
685 Typically, these requirements are not directly related to the capability used by the
686 service consumer, nor are they related to the real-world effect resulting from use of
687 that capability. Rather, the requirements enforce (or support the enforcement of)
688 policies or contracts or otherwise protect the interests of particular business partners
689 or the business organization overall.

690 Common service interaction requirements address areas such as security, reliability,
691 and availability. An initial elaboration of service interaction requirements appears on
692 page 36.

693 **INTERFACE DESCRIPTION REQUIREMENTS** establish common characteristics of
694 service interface descriptions. These requirements address areas such as required
695 interface contents, naming rules, documentation rules, and specification of a
696 standard structure and format for descriptions.

697 **MESSAGE EXCHANGE PATTERNS** identify common sequences of message
698 transmission between service consumers and services. They provide a label to a
699 series of message transmissions that have some logical interrelationship. An initial
700 elaboration of message exchange patterns appears on page 37.

701 **MESSAGE DEFINITION MECHANISMS** are closely related to interface description
702 requirements, described above. Unlike interface description requirements, message
703 definition mechanisms establish a standard way of defining the structure and
704 contents of a message. Note that since a message includes the concept of an
705 “attachment,” the message definition mechanism must identify how different sections
706 of a message (for example, the main section and any “attachment” sections) are
707 separated and identified and how attachment sections are structured and formatted.

708 *Service Interaction Profiles*

709 A **SERVICE INTERACTION PROFILE** defines a family of industry standards or other
710 technologies or techniques that together demonstrate implementation or satisfaction
711 of:

- 712 • Service interaction requirements.
- 713 • Interface description requirements.
- 714 • Message exchange patterns.
- 715 • Message definition mechanisms.

716 Service interaction profiles are included in JRA to promote interoperability without
717 forcing the organization to agree on a single way of enabling service interaction.
718 Each service interface will support a single profile; a service will have multiple
719 interfaces if it supports multiple profiles. By supporting a profile, an interface
720 establishes the mode of interoperation it allows from service consumers; any
721 consumer that also supports that profile can “reach” the service.

722 JRA explicitly recognizes that a service interaction profile may be further constrained
723 by an implementer to require specific techniques, technologies, or mechanisms, as
724 long as the additional constraints remain consistent with the original profile.

725 Capabilities in Detail

726 JRA identifies several types of capabilities to assist decision makers in understanding
727 where certain capabilities should be deployed in the organization and what
728 relationships they may have to other capabilities and services.

729 *Orchestrations and Intermediaries*

730 An **ORCHESTRATION**⁵ is a capability that coordinates interaction with multiple
731 services. An orchestration is often implemented using an open industry standard
732 implementation mechanism (referred to as an **ORCHESTRATION MECHANISM** in
733 JRA), which allows the implementation to be shared across tools and platforms.
734 Also, it is often possible to implement orchestrations using a graphical, model-driven
735 approach, in which the implementer diagrams business processes and work flows,
736 the steps of which are services that already exist. After the diagram is complete, the
737 implementer generates a standards-based artifact that is deployed into a software
738 component that exposes the work flow as a service through a service interface. The
739 promise of this model-driven approach is that less technical implementers with
740 greater business expertise can be responsible for the implementation of orchestrated
741 capabilities.

742 **ROUTERS** are capabilities that receive a message, examine it, and transmit it to one
743 or more destinations based on the contents. In general, routers can be designed to
744 operate on any of the information contained within the message; they may use
745 information about the origin of the message, routing directive information contained
746 within the message or the main content of the message itself.

747 **TRANSFORMERS** are capabilities that receive a message and transform it into another
748 format before transmitting it on to another destination.

749 **INTERCEPTORS** are capabilities that receive a message and use the message content
750 to trigger a secondary action; generally, the interceptors pass the message unaltered
751 to the next step in a process. Most interceptors capture information from the
752 message for reporting or analytical purposes.⁶

753 Routers, transformers, and interceptors are collectively called **INTERMEDIARIES**. An
754 intermediary is any capability that receives messages from a consumer and
755 subsequently, as a service consumer itself, interacts with another service. The term
756 “intermediary” indicates that these capabilities sit between other services and

⁵ In version 1.4 of the JRA, we will change the name of the orchestration concept to something more generic that encompasses orchestration, choreography, and collaboration.

⁶ The concept of interceptor defined here is similar to, but separate and distinct from, the notion of an interceptor as defined in the SOAP protocol [reference needed to SOAP standard]. The definition of this concept in JRA is not intended to imply any implementation technique or technology.

757 “mediate” the interaction by managing, controlling, brokering, or facilitating the
758 transmission of messages between them.

759 Routers and transformers are useful mechanisms for decoupling the senders and
760 recipients of messages. They tend to centralize and share certain kinds of logic so
761 that the logic can be maintained independently of the provider and consumer
762 capabilities at the edges; sharing also improves the likelihood of reuse, since it is
763 easier to reuse functionality if it encapsulates a single task.

764 Support for router, transformer, and orchestration capabilities is a common feature in
765 many integration platforms, and therefore support for these capabilities is a
766 consideration in choice of execution context (discussed on page 32).

767 Routing, transformation, and orchestration capabilities are well understood and well
768 documented in the integration architecture literature. The most common flavors of
769 these capabilities have been collected into pattern form as **ENTERPRISE**
770 **INTEGRATION PATTERNS**. (**Patterns web site**) JRA incorporates these patterns by
771 reference.

772 Orchestrations and intermediaries are a key component in implementing business
773 process models and also lead to the formation of service/capability hierarchies.

774 **Service Policies, Service Contracts, and Service Agreements**

775 **SERVICE POLICIES** and **SERVICE CONTRACTS** express rules that govern the
776 interaction between a service consumer and a service. A policy is an assertion by
777 either a consumer or service provider of that participant’s requirements for
778 willingness to interact. A policy also has an enforcement aspect and must be stated
779 in such a way as to permit enforcement. A **SERVICE CONTRACT** is an agreement by
780 the parties involved, and there is a process associated with the agreement action.
781 Whereas a policy is an assertion by one participant in the interaction, a contract is an
782 agreement between the participants that expresses some expectation or requirement
783 of the interaction. And whereas policy enforcement is generally the responsibility of
784 the participant who asserts the policy, contract enforcement may involve resolution
785 of disputes that arise between the parties.

786 A **SERVICE AGREEMENT** is a document that establishes policies and contractual
787 elements for a given interaction or set of interactions (that is, for one or more
788 services).

789 **Execution Context**

790 **EXECUTION CONTEXT** is “the set of infrastructure elements, process entities, policy
791 assertions, and agreements that are identified as part of an instantiated service
792 interaction.” (**SOA-RM**, p. 24)

793 Execution context is the primary enabler of the reachability aspect of visibility.
794 Execution context includes the set of infrastructure elements that provide a physical
795 communication path between service consumers and services.

796 JRA considers execution context to be primarily the supporting infrastructure
797 elements that permit service consumers and services to interact. These infrastructure
798 elements consist of:

- 799 • Data networks used by service consumers and services to exchange
800 information.
- 801 • Integration infrastructure (hardware and software) that makes
802 service interfaces available and handles higher-level message
803 routing, transformation, and orchestration.
- 804 • Common capabilities that support service interaction; examples
805 include access control services, policy decision services, public key
806 infrastructure (PKI), and metering services.

807 Execution context can implement (or support the implementation of) some service
808 interaction requirements, such as reliability and availability. Service interaction
809 profiles, contracts, and policies can constrain the behavior of execution context
810 elements by requiring particular technologies or techniques or establishing service
811 level policies, for example.

812 Finally, execution context can support intermediary capabilities (as defined above)
813 directly in the integration infrastructure, such as routers, transformers, interceptors,
814 and orchestrations.

815

Illustration Scenarios

816 In version 1.4 of the JRA, this section will include scenarios that illustrate the
817 concepts in the architecture.

Elaboration of JRA Concepts

818

819 The purpose of this section is to establish a direction and initial “straw model” for the
820 components to be defined in detail within JRA. Note that many of these
821 components are currently deliverables within the JRA Work Plan for the 2006 time
822 frame. The GISWG will develop these concepts in incremental steps over time as
823 noted in the Plan. The components that are future deliverables and the other
824 concepts that are more mature are also listed below.

825 In version 1.4 of JRA, this section will change to be a list of pointers to additional
826 documents that fully elaborate and define some of the concepts in JRA.

827 Services and Related Deliverables

828 The JRA deliverables related to services are documented in this section. To cross
829 reference the definitions of corresponding concepts in this section, see page 22.

830 Services

831 The SEARCH Justice Information Exchange Model (JIEM) Reference Model 1.1 will
832 be used as the starting point to define services in JRA. The list of key Information
833 Exchange Package Documentation (IEPD) that have already been developed will be
834 used to further narrow the initial list of services to define. (See
835 <http://it.ojp.gov/iepd/>.)

836 Service Design Principles

837 Note: In version 1.4 of JRA, this list of principles will be removed and replaced (in a
838 separate document) with the service design principles developed by the Services
839 Committee.

840 The following initial list of service design principles is summarized from the text by
841 Thomas Erl. (**Erl**)

- 842 • Services should be designed for reuse.
- 843 • Services should be designed so that they may participate in a
844 composition with other services to form a higher-level service.
- 845 • Services should share only a formal contract with their consumers.
846 Consumers are dependent only on the service’s interface, not the
847 implementation details of the capability to which the service
848 provides access.
- 849 • Services are stateless, meaning that during an interaction with a
850 service, a service consumer supplies all information necessary to
851 conduct the interaction and makes no assumptions about
852 information retained from prior interactions.

853 **Future Service Deliverables**

- 854
 - ***Identification of Service Definitions***
 - 855 - ***Service Specification Guidelines***

856 **Business Process Models**

857 Business Process Models are explained starting on page 24.

858 Although not part of the normative JRA, these business process models may be
859 drawn from normative guidance within specific communities for specific services,
860 such as fusion centers or the exchange of classified intelligence data. They are also
861 useful as guides to more complex orchestrated services that support core business
862 processes within the justice community.

863 **Interaction, Service Models, and Related Concepts**

864 To cross reference the concepts and related deliverables in this section, please see
865 page 25.

866 **Domain Vocabularies**

867 The domain vocabulary for JRA is the Global Justice XML Data Model (Global
868 JXDM) Version 3.0.3. Information about the data model can be accessed at:

869 <http://it.ojp.gov/jxdm>

870 An expanded data model drawing on parts of the National Information Exchange
871 Model (NIEM) may be incorporated. Information on its status may be obtained at:

872 <http://www.niem.gov>

873 **Registries/Repositories**

874 Several SOA registries are now under pilot development in the justice community
875 and could potentially be used to host JRA. Further research is being compiled, and
876 the documentation listed below is currently under development.

877 **Future Interaction and Service Model Deliverables**

878 The GISWG is currently evaluating various approaches to best elaborate the
879 following components. These components will be completed as part of the JRA
880 Work Plan, and will be documented once the deliverables have been solidified.

- 881
 - ***Registries/Repositories Principles***
 - 882 - ***Registries/Repositories Requirements***

- 883 • **Registries/Repositories Guidelines**
- 884 • **Service Description**
- 885 • **Service Modeling Guidelines**

886 **Design and Description of Service Interfaces**

887 As a cross reference, the concepts and related deliverables in this section correspond
888 to the concepts that are explained in the section starting on page 28. The JRA Work
889 Plan includes the following deliverables.

890 **Service Interaction Requirements**

891 The following is an initial list of candidate service interaction requirements. Note that
892 when these requirements refer to “Service Consumer,” this is not a human being, but
893 an information system that interacts with a service. This is consistent with the JRA
894 usage of the term, as defined on page 22.

- 895 • **Service Consumer Authentication:** Information provided with
896 messages transmitted from service consumer to service that verifies
897 the identity of the consumer.
- 898 • **Service Consumer Authorization:** Information provided with
899 messages transmitted from service consumer to service that
900 documents the consumer’s authorization to perform certain actions
901 on and/or access certain information via the service.
- 902 • **Identity and Attribute Assertion Transmission:** Information
903 provided with messages transmitted from service consumer to
904 service that asserts the validity of information about a human or
905 machine, including its identity.
- 906 • **Service Authentication:** The ability of a service to provide a
907 consumer with information that demonstrates the service’s identity
908 to the consumer’s satisfaction.
- 909 • **Message Nonrepudiation:** Information provided in a message
910 to allow the recipient to prove that a particular authorized sender in
911 fact sent the message.
- 912 • **Message Integrity:** Information provided in a message to allow
913 the recipient to verify that the message has not changed since it left
914 the control of the sender.
- 915 • **Message Confidentiality:** Information provided in a message to
916 prevent anyone except an authorized recipient from reading the
917 message or parts of the message.
- 918 • **Message Addressing:** Information provided in a message that
919 indicates where a message originated, the ultimate destination of
920 the message (beyond physical end point), a specific recipient to
921 whom the message should be delivered (this includes sophisticated

922 metadata designed specifically to support routing), and a specific
923 address or entity to which reply messages (if any) should be sent.

- 924 • **Reliability:** Information provided with messages to permit
925 message senders to receive notification of the success or failure of
926 message transmissions, and to permit messages sent with specific
927 sequence-related rules either to arrive as intended, or fail as a
928 group.
- 929 • **Transaction Support:** Information provided with messages to
930 permit a sequence of messages to be treated as an atomic
931 transaction by the recipient.
- 932 • **Service Metadata Availability:** The ability of a service to
933 capture and make available (via query) metadata about the
934 service. Metadata is information that describes or categorizes the
935 service and often assists consumers in interacting with the service in
936 some way.

937 **Service Interaction Profiles**

938 Several service interaction profiles have already been prioritized for development:
939 Web services, MQ, JMS, ebXML, fixed wireless, and mobile wireless. A draft of the
940 Web services service interaction profile is available as part of the OASIS Legal XML
941 Electronic Court Filing 3.0 committee draft specification.

942 **Message Exchange Patterns**

943 JRA will identify the following message exchange patterns:

944 The **FIRE-AND-FORGET** pattern calls for the sender of a message (which could be the
945 service consumer or service) to send the message and not expect a reply message
946 back from the recipient. This pattern is useful for one-way transmission of
947 information, such as notification that an event has occurred.

948 The **REQUEST-REPLY** pattern calls for the sender of a message to send the message
949 and expect a reply back from the recipient.

950 These two patterns are considered “primitive” patterns, in that they are the
951 fundamental building blocks of more complex information exchange scenarios. For
952 instance, the complex **PUBLISH-SUBSCRIBE** pattern involves an initial request-reply
953 exchange in which the subscriber subscribes to a service, followed by the service
954 using the fire-and-forget pattern to notify subscribers of an event.

955 **Future Service Interaction Deliverables**

- 956 • ***Service Interaction Profile Guidelines***
- 957 • ***Interface Description Requirements***
- 958 • ***Message Definition Mechanisms***

959 **Capabilities in Detail and Related Components**

960 To cross reference the concepts and related deliverables in this section, please review
961 page 30. The JRA Work Plan includes the following deliverables.

962 **Provisioning Models**

963 Although not part of the normative JRA, best practices for **PROVISIONING MODELS**
964 provide guidance on how best to implement key facilitation services like message
965 validation, orchestration, routing, and transformation using intermediaries or other
966 means. The GISWG plans on documenting Provisioning Model Guidelines and
967 Principles.

968 **Enterprise Integration Patterns**

969 Although not part of the normative JRA, the existing best practices can be combined
970 with the provisioning models to indicate preferred approaches to the implementation
971 of key services within a community. The GISWG will adopt existing best practices by
972 reference. **(Patterns)**

973 **Future Deliverables**

- 974 • ***Orchestration Guidelines***
- 975 • ***Orchestration Principles***
- 976 • ***Orchestration Mechanisms***

977 **Policies, Contracts, and Agreements**

978 **Model Policies and Contracts**

979 It is possible for every JRA service provider to establish a unique set of policies and
980 business requirements for each service. This approach would create almost
981 insurmountable barriers to the widespread consumption of services for cost reasons
982 alone. The definition of model policies and contracts will provide reusable policies
983 across common services and sets of related services, based on national policies on
984 security, privacy, and other policy requirements. Given the current local and state
985 variations in policy based on statute and court rule, these model policies must
986 necessarily be aspirational initially. The GISWG will develop and recommend
987 potential model policies and contracts.

988 **Model Agreements**

989 These model agreements (termed memorandum of understanding [MOUs], etc.),
990 together with model contracts, lay out standard provisions for consuming JRA
991 services. The GISWG will develop and recommend potential model agreements.

992 **Execution Context**

993 Version 1.4 of the JRA specification will reference an initial elaboration of the
994 Execution Context concept.

Other Considerations

995

996 This document does not identify everything necessary for a successful approach to
997 interoperability among various justice information systems. Other essential factors
998 that need to be addressed but that are not addressed in this document are
999 governance, detailed systems designs, infrastructure specification, or specification of
1000 interfaces between justice systems. These other factors will likely relate to concepts
1001 and components in JRA, so as companion documents that address these other
1002 factors are developed, they should reference JRA when appropriate.

1003 Governance

1004 The issue of interoperability among justice and public safety information systems
1005 raises a set of governance and decision-making questions, such as:

- 1006 • Under what circumstances and through what process is a shared
1007 interface to an information system allowed to change?
- 1008 • Through what process does the organization assess the compliance
1009 of system interfaces with architectural standards?
- 1010 • Through what process does the organization adopt new
1011 architectural standards or change existing ones?
- 1012 • How does the organization reach agreement on the meaning of
1013 information exchanged between interoperable systems?
- 1014 • How do partners enforce agreements and resolve disputes?

1015 Governance business processes and standards will be delivered as a companion
1016 document to JRA. Governance areas of particular concern include registries,
1017 intermediaries, orchestration, and the execution context. JRA currently does not
1018 include these and other aspects of political governance that underpin or support the
1019 technical architecture.

1020 Technical governance is another area that remains to be specified. Issues like
1021 change control and version management go beyond political decisions to practical
1022 administration when operational systems implement a set of technical standards.

1023 The governance document will include at least the following deliverables:

- 1024 • Model policies, agreements, and contracts
- 1025 • IEPD and Service Interaction Profile (SIP) governance processes
- 1026 • Principles for registries, orchestration, services, and provisioning
1027 models

1028 **Investment Strategies**

1029 JRA does not offer guidance on how to fund the implementation of execution
1030 context (in particular, infrastructure to support service interaction via messages) or
1031 the design and implementation of individual services. Identifying high-value and
1032 modest-risk services for initial implementation is an important success factor in the
1033 establishment of an SOA, borne out by experiences in many domains. The reader is
1034 advised to consult material from other sources on technology investment strategies
1035 and the provisioning of shared and consolidated services. In particular, the National
1036 Association of State Chief Information Officers (NASCIO) has published useful
1037 guidance in this area, including the NASCIO Enterprise Architecture Toolkit.

1038 **System Design**

1039 JRA does not include actual applications nor does it propose a design of any
1040 information system. The requirements addressed by JRA focus on the
1041 interoperability of systems, not the systems themselves. JRA does identify the need
1042 for a set of design guidelines that should impact information system design choices.
1043 But these guidelines will address only the integration aspects of systems in support of
1044 business processes that involve the sharing of justice information.

1045 JRA is a set of reference standards and guidelines that such systems may implement
1046 to improve interoperability and information sharing. Global will reach out to existing
1047 local, state, tribal, and national systems to incorporate the JRA specification into their
1048 technical strategies.

1049 **Infrastructure Specifications**

1050 Though the concept of execution context defined above includes the physical
1051 infrastructure necessary to support service interaction, JRA does not identify specific
1052 networks nor does it propose a detailed design for an infrastructure to support
1053 systems integration. Requirements for integration infrastructure could be derived
1054 from further elaboration and specification of some of the concepts and components
1055 documented in JRA.

1056 **System Interfaces**

1057 JRA does not identify specific interfaces between systems. It is intended to provide a
1058 framework or road map for the definition of these interfaces. Specific services based
1059 on the JRA set of policies, guidelines, and standards may be implemented by any
1060 system that finds it useful as a guide. The governance document may separately
1061 recommend that certain systems implement certain services or that certain types of
1062 organizations consume services from particular systems, but that is beyond the scope
1063 of JRA.

Glossary

1064

1065 **Architecture**

1066 A set of artifacts (that is: principles, guidelines, policies, models, standards,
1067 and processes) and the relationships between these artifacts that guide the
1068 selection creation and implementation of solutions aligned with business
1069 goals.

1070 **Awareness**

1071 A state whereby one party has knowledge of the existence of the other party.
1072 Awareness does not imply willingness or reachability.

1073 **Behavior Model**

1074 The characterization of, and responses to, temporal dependencies between
1075 the actions on a service.

1076 **Business Process Models**

1077 A description (usually formal and often graphical) of a series of activities that
1078 culminate in the achievement of some outcome of business value. Some (but
1079 not necessarily all) of the steps in this series of activities involve producing a
1080 real-world effect provided by a capability, and some of the steps require a
1081 consumer to use a service. Each one of these steps, then, provides the
1082 contextual justification for service interaction between a particular consumer
1083 and particular provider.

1084 **Capabilities**

1085 Real-world effect(s) that service provider(s) are able to provide to a service
1086 consumer.

1087 **Consumer Systems**

1088 The information system that gains access to another partner's capability
1089 offered by means of a service.

1090 **Domain Vocabularies**

1091 Includes canonical data models, data dictionaries, and markup languages that
1092 standardize the meaning and structure of information for a domain. Domain
1093 vocabularies can improve the interoperability between consumer and
1094 provider systems by providing a neutral, common basis for structuring and
1095 assigning semantic meaning to information exchanged as part of service
1096 interaction. Domain vocabularies can usually be extended to address
1097 information needs specific to the service interaction or to the business
1098 partners integrating their systems.

1099 **Enterprise Integration Patterns**

1100 Enterprise integration has to deal with connecting multiple applications
1101 running on multiple platforms in different locations. Enterprise Integration
1102 Patterns help integration architects and developers design and implement
1103 integration solutions more rapidly and reliably. Most of the patterns assume
1104 a basic familiarity with messaging architectures. However, the patterns are
1105 not tied to a specific implementation.

1106 **Execution Context**

1107 The set of technical and business elements that form a path between those
1108 with needs and those with capabilities and that permit service providers and
1109 consumers to interact.

1110 **Framework**

1111 A set of assumptions, concepts, values, and practices that constitutes a way of
1112 viewing the current environment.

1113 **Information Model**

1114 The characterization of the information that is associated with the use of a
1115 service. The scope of the information model includes the format of
1116 information that is exchanged, the structural relationships within the
1117 exchanged information, and the definition of terms used.

1118 **Interaction**

1119 The activity involved in making use of a capability offered, usually across an
1120 ownership boundary, in order to achieve a particular desired real-world
1121 effect.

1122 **Interface Description Requirements**

1123 Establishes common characteristics of service interface descriptions. These
1124 requirements address areas such as required interface contents, naming rules,
1125 documentation rules, and specification of a standard structure and format for
1126 descriptions.

1127 **Interceptors**

1128 Interceptors are capabilities that receive a message and use the message
1129 content to trigger a secondary action; generally, the interceptors pass the
1130 message unaltered to the next step in a process.

1131 **Intermediaries**

1132 Routers and transformers are collectively called intermediaries. This term
1133 indicates that routers and transformers generally sit between other services
1134 and “mediate” the interaction by managing the transmission of messages
1135 between them or by reformatting messages in transit.

1136 Justice Reference Architecture

1137 JRA is an abstract framework for understanding significant components and
1138 relationships between them within a service-oriented environment. It lays out
1139 common concepts and definitions as the foundation for the development of
1140 consistent service-oriented architecture (SOA) implementations within the
1141 justice and public safety communities. The term refers to the modular
1142 architecture that cleanly and appropriately identifies and separates technical
1143 and governance layers so that standards can be developed to improve
1144 interoperability. JRA is being developed by Global; it leverages the work of
1145 others, such as the state of Washington, and builds upon the work of OASIS.

1146 Messages

1147 The entire “package” of information sent between service consumer and
1148 service (or vice versa), even if there is a logical partitioning of the message
1149 into segments or sections.

1150 Message Definition Mechanisms

1151 Establishes a standard way of defining the structure and contents of a
1152 message; for example, Global JXDM- or NIEM-conformant schema sets.
1153 Note that since a message includes the concept of an “attachment,” the
1154 message definition mechanism must identify how different sections of a
1155 message (for example, the main section and any “attachment” sections) are
1156 separated and identified and how attachment sections are structured and
1157 formatted.

1158 Message Exchange Patterns

1159 Identifies common sequences of message transmission between service
1160 consumers and services. They provide a label to a series of message
1161 transmissions that have some logical interrelationship.

1162 Message Validators

1163 An intermediary that examines a message to ensure that the contents adhere
1164 to established business rules.

1165 Orchestrations

1166 A capability that coordinates interaction with multiple services. An
1167 orchestration is often implemented using an open industry standard
1168 implementation mechanism (referred to as an orchestration mechanism in
1169 JRA), which allows the implementation to be shared across tools and
1170 platforms.

1171 Process Model

1172 The characterization of the temporal relationships between and temporal
1173 properties of actions and events associated with interacting with the service.

1174 Provider Systems

1175 The information system that offers the use of capabilities by means of a
1176 service.

1177 Provisioning Models

1178 The responsibility/models for making a service available to customers in a
1179 manner consistent with formal (or occasionally informal) customer
1180 expectations.

1181 Reachability

1182 The ability of a service consumer and service provider to interact.
1183 Reachability is an aspect of visibility.

1184 Real-World Effects

1185 The actual result(s) of using a service, rather than merely the capability
1186 offered by a service provider.

1187 Reference Architecture

1188 A reference architecture is an architectural design pattern that indicates how
1189 an abstract set of mechanisms and relationships realizes a predetermined set
1190 of requirements.

1191 Reference Model

1192 A reference model is an abstract framework for understanding significant
1193 relationships among the entities of some environment that enables the
1194 development of specific reference or concrete architectures using consistent
1195 standards or specifications supporting that environment.

1196 A reference model consists of a minimal set of unifying concepts, axioms, and
1197 relationships within a particular problem domain, and is independent of
1198 specific standards, technologies, implementations, or other concrete details.

1199 Repository

1200 Stores models and interface descriptions in a central location that is accessible
1201 to appropriate stakeholders. A repository will permit searching for models
1202 and interface descriptions based on a range of identifying criteria. A
1203 repository will also map logical service identifiers with physical addresses.

1204 Routers

1205 A capability that receives a message, examines it, and transmits it to one or
1206 more destinations based on the contents. In general, routers can be designed
1207 to operate on any of the information contained within the message; they may
1208 use information about the origin of the message, routing directive information
1209 contained within the message or the main content of the message itself.

1210 Services

1211 The means by which the needs of a consumer are brought together with the
1212 capabilities of a provider.

1213 Service Agreements

1214 A document that establishes policies and contractual elements for a given
1215 interaction or set of interactions (that is, for one or more services).

1216 Service Consumers

1217 An entity that seeks to satisfy a particular need through the use of capabilities
1218 offered by means of a service.

1219 Service Contracts

1220 An agreement by two or more parties regarding the conditions of use of a
1221 service.

1222 Service Design Principles

1223 The documentation to provide consistent guidance regarding the overall
1224 partitioning of capabilities into services and the relationships between
1225 services.

1226 Service Interaction Profiles

1227 Defines a family of industry standards or other technologies or techniques that
1228 together demonstrate implementation or satisfaction of:

- 1229 ○ Service interaction requirements.
- 1230 ○ Interface description requirements.
- 1231 ○ Message exchange patterns.
- 1232 ○ Message definition mechanisms.

1233 Service interaction profiles are included in JRA to promote interoperability
1234 without forcing the organization to agree on a single way of enabling service
1235 interaction. Each service interface will support a single profile; a service will
1236 have multiple interfaces if it supports multiple profiles.

1237 Service Interaction Requirements

1238 Define common rules of service interaction. Typically, these requirements are
1239 nonfunctional in nature, in that they are not directly related to the capability
1240 used by the service consumer, nor are they related to the real-world effect
1241 resulting from use of that capability. Rather, the requirements enforce (or
1242 support the enforcement of) policies or contracts or otherwise protect the
1243 interests of particular business partners or the business organization overall.

1244 Service Interfaces

1245 The means by which the underlying capabilities of a service are accessed.

1246 Service Model

1247 Interaction depends on two things. First, the designers of potential consumers
1248 need to be able to find services and, once found, establish a physical
1249 interaction mechanism with them. Second, the designers of potential
1250 consumers need a description of the actions that can be performed on a
1251 service, as well as the structure and meaning of information exchanged during
1252 the interaction. These needs are addressed by the concept of a service's
1253 information model and behavioral model, collectively called service models in
1254 JRA.

1255 Service Modeling Guidelines

1256 Documents guidelines for services provided and consumed among partners.
1257 It provides guidance as well as compliance information regarding the
1258 modeling and description of services to promote consistency.

1259 Service-Oriented Architecture (SOA)

1260 Service-Oriented Architecture is a paradigm for organizing and utilizing
1261 distributed capabilities that may be under the control of different ownership
1262 domains. It provides a uniform means to offer, discover, interact with, and
1263 use capabilities to produce desired effects consistent with measurable
1264 preconditions and expectations.

1265 Service Policies

1266 A statement of obligations, constraints, or other conditions of use,
1267 deployment, or description of an owned entity as defined by any participant.

1268 Service Providers

1269 An entity (person or organization) that offers the use of capabilities by means
1270 of a service.

1271 Transformers

1272 A capability that receives a message and transforms it into another format
1273 before transmitting it on to another destination.

1274 Visibility

1275 The capacity for those with needs and those with capabilities to be able to
1276 interact with each other.

1277 Willingness

1278 A predisposition of service providers and consumers to interact.

1279

References

1280

1281

Erl

1282

1283

1284

Erl, Thomas. *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice-Hall, 2005

GISWG

1285

1286

1287

GISWG. A Framework for Justice Information Sharing: *Service-Oriented Architecture*. Global, December 9, 2004.

Patterns

1288

1289

1290

1291

Hohpe, Gregor, and Woolf, Bobby. *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. Addison Wesley, 2004. <http://www.eaipatterns.com>

Sholler

1292

1293

Sholler, Daniel. *Demystifying Service-Oriented Architecture*, META Practice, June 9, 2004.

SOA-RM

1294

1295

1296

1297

Reference Model for Service-Oriented Architecture 1.0, Committee Specification 1. OASIS, July 19, 2006. <http://www.oasis-open.org/>

1298

1299

1300

1301

1302

Document History

1303

Date	Version	Editor	Change
March 25, 2006	1.0	Scott Came	Initial Draft
March 28, 2006	1.0	Tish Cunningham Kim Geer	Editorial changes and IIR QC
May 1, 2006	1.1	Monique La Bare	Integrate comments from EAC, glossary, introduction, acknowledgements, insert scenario, editing page numbers

Date	Version	Editor	Change
June 1, 2006	1.1	Tom Clarke	Elaboration of concepts and principles.
June 28, 2006	1.1		Reordered elaboration of concepts, added warrant scenario
November 2, 2006	1.?	Scott Came	Consistency edits Edits resulting from October GISWG meetings Reflect comments of Iveta Topalova and Martin Smith
December 6, 2006	1.3	Kim Geer Dolores Parker	Formatting Editorial changes and IIR quality control

1304

1305