



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 .....	29
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)**

38 Mr. Eric Sweden—National Association for State Chief Information Officers, Vice  
39 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-RM<sup>1</sup>). 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.

---

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

76

## Executive Summary

77

78

79

80

81

82

83

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

84

85



## 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.

## 125 **An Interoperability Strategy**

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

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

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

- 148 **Step One: Agree on common concepts**
- 149 **Step Two: Agree on the relationships and deliverables**
- 150 **Step Three: Assign the work**
- 151 **Step Four: Produce the deliverables**
- 152 **Step Five: Revise the deliverables**

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

## 156 **Consensus on the OASIS Reference Model for SOA**

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

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

## 169 **Creating the Justice Reference Architecture**

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

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

179  
180  
181  
182  
183  
184

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

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

## 188 **What Is Justice Reference Architecture?**

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

194  
195  
196  
197  
198

**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.**

199  
200  
201  
202  
203  
204  
205  
206  
207

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.

208  
209  
210  
211

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

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

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

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

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

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

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

This same information, or portions of it, are necessary to meet the business requirements of related justice, public safety, and homeland security agencies. For example, this information is required to regulate the sale of firearms; complete criminal background checks of employees at schools, child care

252 services, and elder care facilities; identify aliens who have been convicted of  
253 crimes or have entered the country illegally; notify the local community of the  
254 release and location of sexual predators; prevent training in the operation of  
255 aircraft by aliens or other designated individuals who may present a risk to  
256 aviation and national security; do background checks of those transporting  
257 hazardous materials; or create information models to provide information and  
258 predict the spread of disease and its effects, and decide on countermeasures for  
259 potential health epidemics like the avian flu.

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

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

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

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

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

288 Different sources also mean differences in expectations about who can use what  
289 information. Privacy and data quality issues, which are demanding enough when  
290 dealing with a single information system, grow exponentially when dealing with  
291 different disciplines. It is one thing to share the records of a criminal sentencing  
292 hearing held in open court; it is quite another when dealing with health records  
293 or an ongoing criminal investigation. Incomplete or inaccurate data may be an

294 annoyance if the task is to identify leads for subsequent investigations; they are a  
295 different issue entirely if they prohibit one from getting a job, traveling on an  
296 airplane, or lead to incarceration. Working documents in one setting can become  
297 dispositive evidence in another.

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

304 **Requirement 3—JRA must be able to accommodate an infinite range**  
305 **of scales, from small operations with few participants in a rural**  
306 **county to national processes that reach across local, state, federal,**  
307 **and even international boundaries.<sup>2</sup>**

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

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

---

<sup>2</sup> 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.



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

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

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

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

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

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



360 **Requirement 6—JRA must be dynamic and capable of evolving as the**  
361 **information sharing requirements change and the technology is**  
362 **transformed.**

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

373 **Requirement 7—JRA should leverage open industry standards where**  
374 **possible.**

375 The justice environment will benefit from the stabilization of standards as the  
376 basis for an overall approach to interoperability among large and diverse  
377 organizations. The evolution of open industry standards for systems integration  
378 has reached a point where these standards will facilitate interoperability. Many  
379 prominent programming languages, software development environments,  
380 packaged applications, and integration platforms/tools support the standards.  
381 Although some common integration needs are met by competing standards, the  
382 number and significance of competing standards continue to shrink.

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

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

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

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

392 **Requirement 11—JRA should derive service requirements from**  
393 **business process requirements.**

394 **Requirement 12—JRA should preserve data control by the source**  
395 **organization.**

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

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

400 **Requirement 15—JRA should support business agility as the**  
401 **fundamental business requirement.**

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

403 **Requirement 17—JRA should evolve indefinitely in response to**  
404 **changing business requirements.**

405

406

## Justice Reference Architecture

407

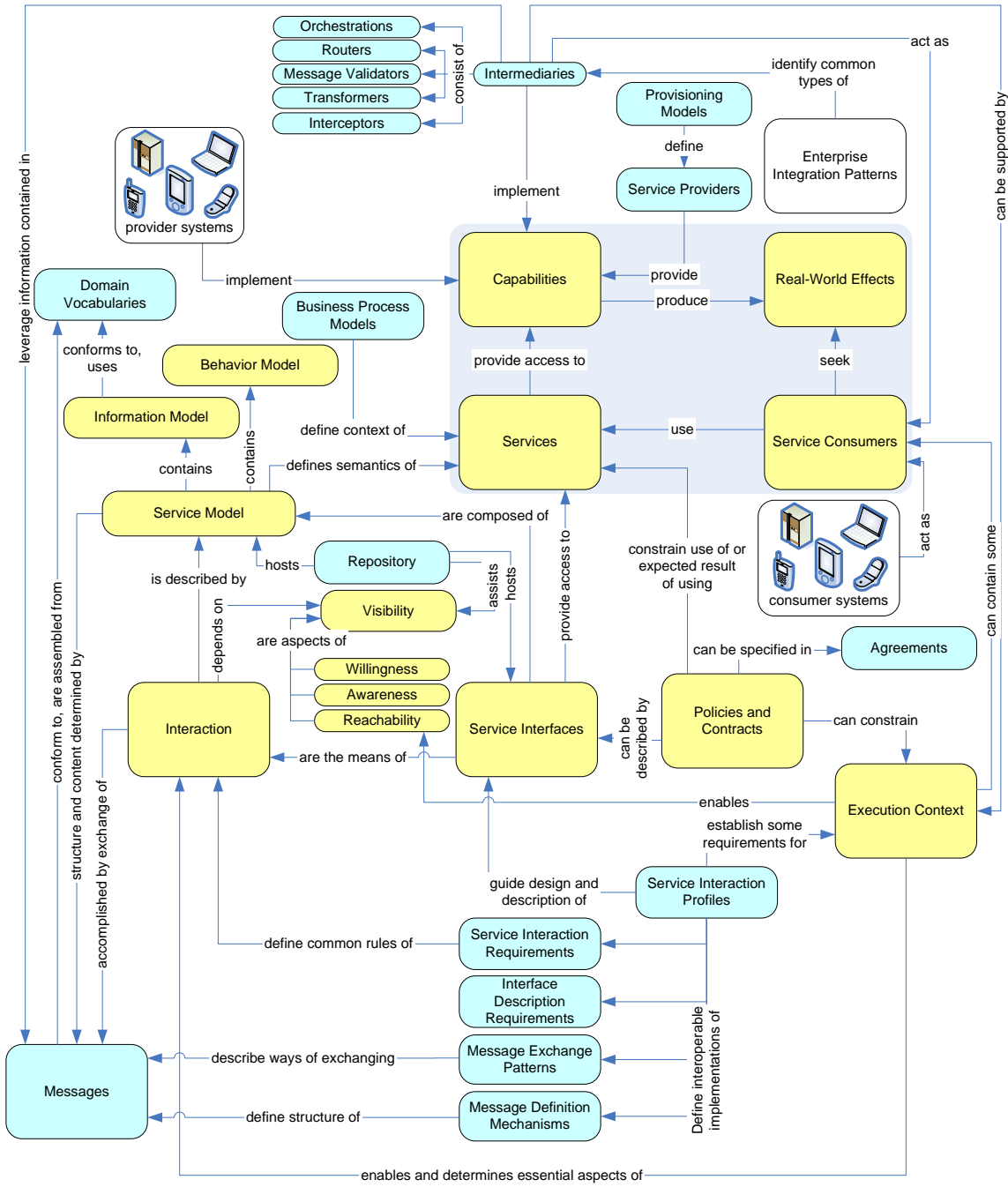
### Graphical Overview

408

The following diagram depicts the concepts, high-level components, and relationships in the Justice Reference Architecture Specification V 1.3. These elements are described in detail in the following sections.

409

410



<b>Justice Reference Architecture Concept Map</b> November 3, 2006	<b>Legend</b> <div style="display: flex; align-items: center; margin-bottom: 5px;"> <div style="width: 20px; height: 15px; background-color: yellow; border: 1px solid black; margin-right: 5px;"></div> <span>Concepts from OASIS SOA-RM</span> </div> <div style="display: flex; align-items: center;"> <div style="width: 20px; height: 15px; background-color: lightblue; border: 1px solid black; margin-right: 5px;"></div> <span>Concepts particular to the JRA</span> </div>
---------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

411

412

## 413 **Concepts and Relationships**

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

### 416 **OASIS Reference Model for Service-Oriented Architecture for**

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

419 The SOA-RM defines its purpose as follows:

420 “A **REFERENCE MODEL** is an abstract framework for understanding  
421 significant relationships among the entities of some environment.  
422 It enables the development of specific reference or concrete  
423 architectures using consistent standards or specifications  
424 supporting that environment. A reference model consists of a  
425 minimal set of unifying concepts, axioms, and relationships within  
426 a particular problem domain and is independent of specific  
427 standards, technologies, implementations, or other concrete  
428 details.” (SOA-RM, p. 4)

429 “The goal of this reference model is to define the essence of  
430 service-oriented architecture and emerge with a vocabulary and a  
431 common understanding of SOA. It provides a normative reference  
432 that remains relevant for SOA as an abstract and powerful model,  
433 irrespective of the various and inevitable technology evolutions  
434 that will impact SOA.” (SOA-RM, p. 4)

435 While the SOA-RM is a powerful model that provides a vendor-neutral, open-  
436 standard definition of service-oriented architecture, its abstract nature means  
437 that further work must be done to create reference architecture. OASIS lays out a  
438 roadmap for the creation of such reference architecture. **Specifically, OASIS**  
439 **recommends that the reference model guide the development of an**  
440 **architecture; that protocols, profiles, specifications, and standards**  
441 **are considered; and that requirements, motivations, and goals are**  
442 **taken into account.** (SOA-RM, p. 5)<sup>3</sup>

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

446 In the JRA diagram, OASIS SOA-RM concepts are shaded yellow with a dashed  
447 line as the border. Concepts and components particular to the conceptual JRA  
448 defined by this document are shaded light blue with a solid border. Relationships

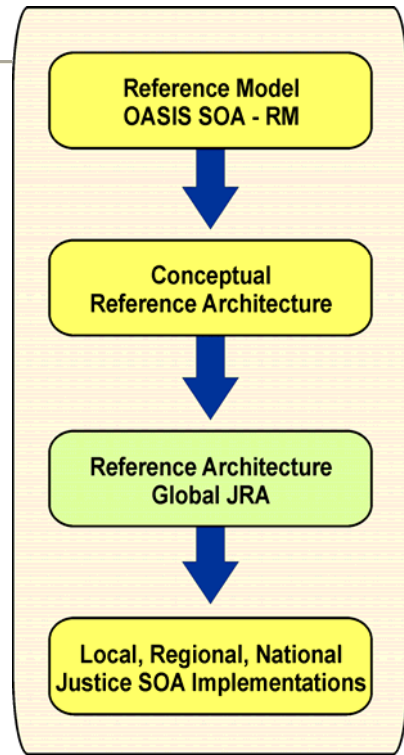
---

<sup>3</sup> 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.

## Justice Reference Architecture

449 between concepts (indicated by arrows) are  
450 defined in the SOA-RM if the arrows connect  
451 concepts shaded yellow. Relationships between  
452 cyan-shaded concepts or between cyan-shaded and  
453 yellow-shaded concepts are particular to JRA.

454 The descriptions of SOA-RM concepts provided in  
455 the following sections are intended to be brief  
456 summaries; consequently, they omit certain details  
457 that appear in the SOA-RM. Concepts listed in  
458 bold, blue caps are listed in the glossary at the end  
459 of this document, and the glossary contains  
460 definitions of the SOA-RM concepts, which are  
461 repeated from the SOA-RM glossary for  
462 convenience. The SOA-RM itself is the primary  
463 source for full exposition of  
464 SOA-RM concepts and the relationships between  
465 them.



## 466 **Core Concepts—Services, Service Consumers, Capabilities, and Real-World Effects**

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

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

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

483 Each capability produces one or more **REAL-WORLD EFFECTS**, each of which is an  
484 outcome of the business value sought by one of the partners. A real-world effect  
485 can be either the obtaining of information, the changing of something of business  
486 relevance to the participating partners, or both. Because JRA establishes that  
487 capabilities are implemented by provider systems, real-world effects consist of  
488 the functional business requirements of provider systems. That is, real-world  
489 effects in JRA are essentially the information made available by provider systems

490 or the outcomes resulting from business processes and workflows automated by  
491 provider systems, or both.

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

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

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

## 511 **Supporting Concepts**

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

516 **SERVICE DESIGN PRINCIPLES<sup>4</sup>** provide consistent guidance regarding the overall  
517 partitioning of capabilities into services and the relationships between services.  
518 For instance, service design principles may call for services to represent one  
519 concise, self-contained function and may also suggest that services should  
520 completely hide the implementation details of the capabilities to which they  
521 provide access.

522 There is a wide variety of ways in which a service can provide access to a  
523 capability. In some cases, the provider system that implements the capability  
524 may already expose all or some of its functionality as services (through one or  
525 more service interfaces, described on page 27). In other cases, the business  
526 partner that provisions the capability can purchase an off-the-shelf adapter from

---

<sup>4</sup> 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.



527 the provider system vendor (or a third party) that exposes the system's  
528 functionality as a set of services. Finally, the provider system may require  
529 reimplementation or custom adaptation to expose functionality as services. This  
530 is often expensive and risky, and the desire to avoid this situation should be  
531 addressed in the Service Design Guidelines.

532 In general, a given information system can be both a provider system and a  
533 consumer system. Similarly, a particular business organization may offer  
534 capabilities to its partners and, at the same time, be a consumer of the  
535 capabilities offered by others. This has important implications for how the  
536 organization should conceive and describe its information systems assets and  
537 how it assigns responsibilities for the maintenance and support of those assets.  
538 For example, in the past it was common to think of systems as having "client" and  
539 "server" components (or "browser" and "server" components), which in turn  
540 influenced thinking about systems deployment, networking, security, support,  
541 and a range of other issues. These issues deserve reconsideration in an  
542 architecture in which a system or system component can be both a "client"  
543 (consumer of services) and "server" (provider of services) at the same time. The  
544 discussion of service interaction on page 25, and the subsequent elaboration of  
545 interaction mechanisms in future iterations of JRA, will reflect the impact of  
546 these issues.

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

## 555 **Business Process Models and the Service/Capability Hierarchy**

556 The previous section described the basic concepts involved in the integration of  
557 provider systems and consumer systems. In short, consumer systems seek a real-  
558 world effect provided by a capability, and they produce that effect by accessing a  
559 service that provides access to the capability. However, these concepts by  
560 themselves do not provide the context for the integration of a particular  
561 consumer and particular provider. That is, the concepts do not provide a way of  
562 describing why a consumer seeks the effect made available by a provider through  
563 a service.

564 A **BUSINESS PROCESS MODEL** provides this contextual justification. A business  
565 process model is a description (usually formal and often graphical) of a series of  
566 activities that culminate in the achievement of some outcome of business value.  
567 Some (but not necessarily all) of the steps in this series of activities involve  
568 producing a real-world effect provided by a capability, and some of the steps  
569 require a consumer to use a service. Each one of these steps, then, provide the



570 contextual justification for service interaction between a particular consumer and  
571 particular provider.

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

581 JRA supports this hierarchy through orchestrations and intermediaries,  
582 discussed on page 29.

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

## 586 **Interaction, Visibility, Service Models, and Service Interfaces**

587 Services define what features of a provider system the system owner makes  
588 accessible to business partners. Services also provide a logical description of the  
589 information exchanged between consumer and provider systems as the consumer  
590 accesses the capability.

### 591 *Interaction*

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

596 Interaction depends on two things. First, the designers of potential consumers  
597 need to be able to find services and, once found, establish a physical interaction  
598 mechanism with them. These needs are addressed by the concept of **VISIBILITY**.  
599 Second, the designers of potential consumers need a description of the actions  
600 that can be performed on a service, as well as the structure and meaning of  
601 information exchanged during the interaction. These needs are addressed by the  
602 concept of a service's **INFORMATION MODEL** and **BEHAVIOR MODEL**, collectively  
603 called **SERVICE MODELS** in the JRA.

### 604 *Visibility*

605 Visibility, as the name implies, defines how service consumers and the providers  
606 of capabilities "see" each other in a way that enables interaction between them.  
607 JRA identifies three aspects of visibility.

- 608 • A service consumer must have information that makes it aware  
609 of the existence of a service; the possession of this information is  
610 called **AWARENESS**.
- 611 • The service (or capability accessed through the service) must be  
612 willing to interact with the consumer; this is called  
613 **WILLINGNESS**.
- 614 • The consumer and service must be able to communicate with  
615 one another through some kind of communication path or  
616 channel; the existence of such a communication path is called  
617 **REACHABILITY**.

618 In JRA, a **REPOSITORY** will support awareness by hosting service models and  
619 service interfaces. “Hosting” in this context means storing models and interface  
620 descriptions in a central location that is accessible to appropriate stakeholders. A  
621 repository will permit searching for models and interface descriptions based on a  
622 range of identifying criteria. A repository will also map logical service identifiers  
623 with physical addresses. When a consumer wishes to communicate with a service  
624 (identified by a logical identifier), the consumer queries the repository for the  
625 physical address associated with the service’s logical identifier. This decouples  
626 the consumer from the physical location of a service at any point in time, thereby  
627 permitting the physical relocation of the service without impacting the  
628 implementation of the consumer.

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

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

### 637 *Service Models*

638 Service models, consisting of a service’s information and behavior models, define  
639 the semantics of interaction with the service. The behavior model defines the  
640 actions that can be performed on the service; that is, it defines what the service  
641 “does.” The information model describes the information that consumers  
642 exchange with the service in the course of performing those actions.

643 Note that the SOA-RM considers the orchestration and choreography of multiple  
644 services to be “part of the **PROCESS MODEL** of a given architecture.” Yet the SOA-  
645 RM also indicates that a process model (part of the behavior model) applies to a  
646 single service. (**SOA-RM**, p. 15) Because of this lack of clarity in the SOA-RM,  
647 this JRA defines orchestration as a type of capability that leverages other  
648 services; it is described on page 29.

649 In general, service models will be described at conceptual and logical levels of  
650 detail. (Service models have a physical manifestation as well, in the form of the  
651 service interface discussed in the next section.) A conceptual description of a  
652 service model will typically describe, in prose text form, the capability to which  
653 the service provides access, a listing and brief textual description of each action,  
654 and a brief textual description of the information model (e.g., key information  
655 entities, key properties on those entities, and brief definitions). A logical  
656 description of a service model will describe the actions and information  
657 structures in detail but independent of any physical implementation mechanism.  
658 Often this description will be graphical and follow a standard diagramming or  
659 modeling technique, such as Uniform Modeling Language (UML).

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

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

672 The concept of **DOMAIN VOCABULARIES** in JRA includes canonical data models,  
673 data dictionaries, and markup languages that standardize the meaning and  
674 structure of information for a topical or business domain. Domain vocabularies  
675 can improve the interoperability between consumer and provider systems by  
676 providing a neutral, common basis for structuring and assigning semantic  
677 meaning to information exchanged as part of service interaction. Domain  
678 vocabularies can usually be extended to address information needs specific to the  
679 service interaction or to the business partners integrating their systems.

680 **SERVICE MODELING GUIDELINES** govern the style, structure, and description of  
681 service models.

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

### 686 *Service Interface*

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

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

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

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

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

## 705 **Design and Description of Service Interfaces**

706 JRA identifies four architectural elements that guide the design and description  
707 of service interfaces.

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

714 Common service interaction requirements address areas such as security,  
715 reliability, and availability. An initial elaboration of service interaction  
716 requirements appears on page 36.

717 **INTERFACE DESCRIPTION REQUIREMENTS** establish common characteristics of  
718 service interface descriptions. These requirements address areas such as  
719 required interface contents, naming rules, documentation rules, and specification  
720 of a standard structure and format for descriptions.

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

725 **MESSAGE DEFINITION MECHANISMS** are closely related to interface description  
726 requirements, described above. Unlike interface description requirements,  
727 message definition mechanisms establish a standard way of defining the

728 structure and contents of a message. Note that since a message includes the  
729 concept of an “attachment,” the message definition mechanism must identify how  
730 different sections of a message (for example, the main section and any  
731 “attachment” sections) are separated and identified and how attachment sections  
732 are structured and formatted.

### 733 *Service Interaction Profiles*

734 A **SERVICE INTERACTION PROFILE** defines a family of industry standards or other  
735 technologies or techniques that together demonstrate implementation or  
736 satisfaction of:

- 737 • Service interaction requirements.
- 738 • Interface description requirements.
- 739 • Message exchange patterns.
- 740 • Message definition mechanisms.

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

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

### 751 **Capabilities in Detail**

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

### 755 *Orchestrations and Intermediaries*

756 An **ORCHESTRATION**<sup>5</sup> is a capability that coordinates interaction with multiple  
757 services. An orchestration is often implemented using an open industry standard  
758 implementation mechanism (referred to as an **ORCHESTRATION MECHANISM** in  
759 JRA), which allows the implementation to be shared across tools and platforms.  
760 Also, it is often possible to implement orchestrations using a graphical, model-  
761 driven approach, in which the implementer diagrams business processes and  
762 work flows, the steps of which are services that already exist. After the diagram is  
763 complete, the implementer generates a standards-based artifact that is deployed

---

<sup>5</sup> 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.

764 into a software component that exposes the work flow as a service through a  
765 service interface. The promise of this model-driven approach is that less  
766 technical implementers with greater business expertise can be responsible for the  
767 implementation of orchestrated capabilities.

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

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

776 **INTERCEPTORS** are capabilities that receive a message and use the message  
777 content to trigger a secondary action; generally, the interceptors pass the  
778 message unaltered to the next step in a process. Most interceptors capture  
779 information from the message for reporting or analytical purposes.<sup>6</sup>

780 Routers, transformers, and interceptors are collectively called **INTERMEDIARIES**.  
781 An intermediary is any capability that receives messages from a consumer and  
782 subsequently, as a service consumer itself, interacts with another service. The  
783 term “intermediary” indicates that these capabilities sit between other services  
784 and “mediate” the interaction by managing, controlling, brokering, or facilitating  
785 the transmission of messages between them.

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

791 Support for router, transformer, and orchestration capabilities is a common  
792 feature in many integration platforms, and therefore support for these  
793 capabilities is a consideration in choice of execution context (discussed on page  
794 32).

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

---

<sup>6</sup> 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.



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

### 803 **Service Policies, Service Contracts, and Service Agreements**

804 **SERVICE POLICIES** and **SERVICE CONTRACTS** express rules that govern the  
805 interaction between a service consumer and a service. A policy is an assertion by  
806 either a consumer or service provider of that participant's requirements for  
807 willingness to interact. A policy also has an enforcement aspect and must be  
808 stated in such a way as to permit enforcement. A **SERVICE CONTRACT** is an  
809 agreement by the parties involved, and there is a process associated with the  
810 agreement action. Whereas a policy is an assertion by one participant in the  
811 interaction, a contract is an agreement between the participants that expresses  
812 some expectation or requirement of the interaction. And whereas policy  
813 enforcement is generally the responsibility of the participant who asserts the  
814 policy, contract enforcement may involve resolution of disputes that arise  
815 between the parties.

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

### 819 **Execution Context**

820 **EXECUTION CONTEXT** is "the set of infrastructure elements, process entities,  
821 policy assertions, and agreements that are identified as part of an instantiated  
822 service interaction." (**SOA-RM**, p. 24)

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

826 JRA considers execution context to be primarily the supporting infrastructure  
827 elements that permit service consumers and services to interact. These  
828 infrastructure elements consist of:

- 829 • Data networks used by service consumers and services to  
830 exchange information.
- 831 • Integration infrastructure (hardware and software) that makes  
832 service interfaces available and handles higher-level message  
833 routing, transformation, and orchestration.
- 834 • Common capabilities that support service interaction; examples  
835 include access control services, policy decision services, public  
836 key infrastructure (PKI), and metering services.

837 Execution context can implement (or support the implementation of) some  
838 service interaction requirements, such as reliability and availability. Service

839 interaction profiles, contracts, and policies can constrain the behavior of  
840 execution context elements by requiring particular technologies or techniques or  
841 establishing service level policies, for example.

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



845

## **Illustration Scenarios**

846

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

847

## Elaboration of JRA Concepts

848

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

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

### Services and Related Deliverables

857

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

#### Services

860

861 The SEARCH Justice Information Exchange Model (JIEM) Reference Model 1.1  
862 will be used as the starting point to define services in JRA. The list of key  
863 Information Exchange Package Documentation (IEPD) that have already been  
864 developed will be used to further narrow the initial list of services to define. (See  
865 [http://it.ojp.gov/iepd/.](http://it.ojp.gov/iepd/))

#### Service Design Principles

866

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

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

872

873

874

875

876

877

878

879

880

881

882

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

---

## 883 **Future Service Deliverables**

- 884 • *Identification of Service Definitions*
- 885 • *Service Specification Guidelines*

## 886 **Business Process Models**

887 Business Process Models are explained starting on page 24.

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

## 893 **Interaction, Service Models, and Related Concepts**

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

## 896 **Domain Vocabularies**

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

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

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

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

## 904 **Registries/Repositories**

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

## 908 **Future Interaction and Service Model Deliverables**

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

- 912 • *Registries/Repositories Principles*
- 913 • *Registries/Repositories Requirements*
- 914 • *Registries/Repositories Guidelines*

- 915 • *Service Description*
- 916 • *Service Modeling Guidelines*

## 917 **Design and Description of Service Interfaces**

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

### 921 **Service Interaction Requirements**

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

- 926 • **Service Consumer Authentication:** Information provided  
927 with messages transmitted from service consumer to service  
928 that verifies the identity of the consumer.
- 929 • **Service Consumer Authorization:** Information provided  
930 with messages transmitted from service consumer to service  
931 that documents the consumer’s authorization to perform certain  
932 actions on and/or access certain information via the service.
- 933 • **Identity and Attribute Assertion Transmission:**  
934 Information provided with messages transmitted from service  
935 consumer to service that asserts the validity of information  
936 about a human or machine, including its identity.
- 937 • **Service Authentication:** The ability of a service to provide a  
938 consumer with information that demonstrates the service’s  
939 identity to the consumer’s satisfaction.
- 940 • **Message Nonrepudiation:** Information provided in a  
941 message to allow the recipient to prove that a particular  
942 authorized sender in fact sent the message.
- 943 • **Message Integrity:** Information provided in a message to  
944 allow the recipient to verify that the message has not changed  
945 since it left the control of the sender.
- 946 • **Message Confidentiality:** Information provided in a  
947 message to prevent anyone except an authorized recipient from  
948 reading the message or parts of the message.
- 949 • **Message Addressing:** Information provided in a message  
950 that indicates where a message originated, the ultimate  
951 destination of the message (beyond physical end point), a  
952 specific recipient to whom the message should be delivered (this  
953 includes sophisticated metadata designed specifically to support  
954 routing), and a specific address or entity to which reply  
955 messages (if any) should be sent.

- 956
- 957
- 958
- 959
- 960
- 961
- 962
- 963
- 964
- 965
- 966
- 967
- 968
- **Reliability:** Information provided with messages to permit message senders to receive notification of the success or failure of message transmissions, and to permit messages sent with specific sequence-related rules either to arrive as intended, or fail as a group.
  - **Transaction Support:** Information provided with messages to permit a sequence of messages to be treated as an atomic transaction by the recipient.
  - **Service Metadata Availability:** The ability of a service to capture and make available (via query) metadata about the service. Metadata is information that describes or categorizes the service and often assists consumers in interacting with the service in some way.

### 969 **Service Interaction Profiles**

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

### 975 **Message Exchange Patterns**

976 JRA will identify the following message exchange patterns:

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

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

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

## 988 **Future Service Interaction Deliverables**

- 989 • *Service Interaction Profile Guidelines*
- 990 • *Interface Description Requirements*
- 991 • *Message Definition Mechanisms*

## 992 **Capabilities in Detail and Related Components**

993 To cross reference the concepts and related deliverables in this section, please  
994 review page 29. The JRA Work Plan includes the following deliverables.

### 995 **Provisioning Models**

996 Although not part of the normative JRA, best practices for **PROVISIONING**  
997 **MODELS** provide guidance on how best to implement key facilitation services like  
998 message validation, orchestration, routing, and transformation using  
999 intermediaries or other means. The GISWG plans on documenting Provisioning  
1000 Model Guidelines and Principles.

### 1001 **Enterprise Integration Patterns**

1002 Although not part of the normative JRA, the existing best practices can be  
1003 combined with the provisioning models to indicate preferred approaches to the  
1004 implementation of key services within a community. The GISWG will adopt  
1005 existing best practices by reference. **(Patterns)**

### 1006 **Future Deliverables**

- 1007 • *Orchestration Guidelines*
- 1008 • *Orchestration Principles*
- 1009 • *Orchestration Mechanisms*

## 1010 **Policies, Contracts, and Agreements**

### 1011 **Model Policies and Contracts**

1012 It is possible for every JRA service provider to establish a unique set of policies  
1013 and business requirements for each service. This approach would create almost  
1014 insurmountable barriers to the widespread consumption of services for cost  
1015 reasons alone. The definition of model policies and contracts will provide  
1016 reusable policies across common services and sets of related services, based on  
1017 national policies on security, privacy, and other policy requirements. Given the  
1018 current local and state variations in policy based on statute and court rule, these  
1019 model policies must necessarily be aspirational initially. The GISWG will develop  
1020 and recommend potential model policies and contracts.

1021 **Model Agreements**

1022 These model agreements (termed memorandum of understanding [MOUs], etc.),  
1023 together with model contracts, lay out standard provisions for consuming JRA  
1024 services. The GISWG will develop and recommend potential model agreements.

1025 **Execution Context**

1026 Version 1.4 of the JRA specification will reference an initial elaboration of the  
1027 Execution Context concept.

1028

## Other Considerations

1029

1030

1031

1032

1033

1034

1035

1036

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

1037

## Governance

1038

1039

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

1040

1041

1042

1043

1044

1045

1046

1047

1048

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

1049

1050

1051

1052

1053

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

1054

1055

1056

1057

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

1058

The governance document will include at least the following deliverables:

1059

1060

1061

1062

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



## 1063 **Investment Strategies**

1064 JRA does not offer guidance on how to fund the implementation of execution  
1065 context (in particular, infrastructure to support service interaction via messages)  
1066 or the design and implementation of individual services. Identifying high-value  
1067 and modest-risk services for initial implementation is an important success factor  
1068 in the establishment of an SOA, borne out by experiences in many domains. The  
1069 reader is advised to consult material from other sources on technology  
1070 investment strategies and the provisioning of shared and consolidated services.  
1071 In particular, the National Association of State Chief Information Officers  
1072 (NASCIO) has published useful guidance in this area, including the NASCIO  
1073 Enterprise Architecture Toolkit.

## 1074 **System Design**

1075 JRA does not include actual applications nor does it propose a design of any  
1076 information system. The requirements addressed by JRA focus on the  
1077 interoperability of systems, not the systems themselves. JRA does identify the  
1078 need for a set of design guidelines that should impact information system design  
1079 choices. But these guidelines will address only the integration aspects of systems  
1080 in support of business processes that involve the sharing of justice information.

1081 JRA is a set of reference standards and guidelines that such systems may  
1082 implement to improve interoperability and information sharing. Global will  
1083 reach out to existing local, state, tribal, and national systems to incorporate the  
1084 JRA specification into their technical strategies.

## 1085 **Infrastructure Specifications**

1086 Though the concept of execution context defined above includes the physical  
1087 infrastructure necessary to support service interaction, JRA does not identify  
1088 specific networks nor does it propose a detailed design for an infrastructure to  
1089 support systems integration. Requirements for integration infrastructure could  
1090 be derived from further elaboration and specification of some of the concepts and  
1091 components documented in JRA.

## 1092 **System Interfaces**

1093 JRA does not identify specific interfaces between systems. It is intended to  
1094 provide a framework or road map for the definition of these interfaces. Specific  
1095 services based on the JRA set of policies, guidelines, and standards may be  
1096 implemented by any system that finds it useful as a guide. The governance  
1097 document may separately recommend that certain systems implement certain  
1098 services or that certain types of organizations consume services from particular  
1099 systems, but that is beyond the scope of JRA.

---

## Glossary

1100

### 1101 **Architecture**

1102 A set of artifacts (that is: principles, guidelines, policies, models,  
1103 standards, and processes) and the relationships between these artifacts  
1104 that guide the selection creation and implementation of solutions aligned  
1105 with business goals.

### 1106 **Awareness**

1107 A state whereby one party has knowledge of the existence of the other  
1108 party. Awareness does not imply willingness or reachability.

### 1109 **Behavior Model**

1110 The characterization of, and responses to, temporal dependencies between  
1111 the actions on a service.

### 1112 **Business Process Models**

1113 A description (usually formal and often graphical) of a series of activities  
1114 that culminate in the achievement of some outcome of business value.  
1115 Some (but not necessarily all) of the steps in this series of activities involve  
1116 producing a real-world effect provided by a capability, and some of the  
1117 steps require a consumer to use a service. Each one of these steps, then,  
1118 provides the contextual justification for service interaction between a  
1119 particular consumer and particular provider.

### 1120 **Capabilities**

1121 Real-world effect(s) that service provider(s) are able to provide to a service  
1122 consumer.

### 1123 **Consumer Systems**

1124 The information system that gains access to another partner's capability  
1125 offered by means of a service.

### 1126 **Domain Vocabularies**

1127 Includes canonical data models, data dictionaries, and markup languages  
1128 that standardize the meaning and structure of information for a domain.  
1129 Domain vocabularies can improve the interoperability between consumer  
1130 and provider systems by providing a neutral, common basis for structuring  
1131 and assigning semantic meaning to information exchanged as part of  
1132 service interaction. Domain vocabularies can usually be extended to  
1133 address information needs specific to the service interaction or to the  
1134 business partners integrating their systems.

---

## 1135 **Enterprise Integration Patterns**

1136 Enterprise integration has to deal with connecting multiple applications  
1137 running on multiple platforms in different locations. Enterprise  
1138 Integration Patterns help integration architects and developers design and  
1139 implement integration solutions more rapidly and reliably. Most of the  
1140 patterns assume a basic familiarity with messaging architectures.  
1141 However, the patterns are not tied to a specific implementation.

## 1142 **Execution Context**

1143 The set of technical and business elements that form a path between those  
1144 with needs and those with capabilities and that permit service providers  
1145 and consumers to interact.

## 1146 **Framework**

1147 A set of assumptions, concepts, values, and practices that constitutes a way  
1148 of viewing the current environment.

## 1149 **Information Model**

1150 The characterization of the information that is associated with the use of a  
1151 service. The scope of the information model includes the format of  
1152 information that is exchanged, the structural relationships within the  
1153 exchanged information, and the definition of terms used.

## 1154 **Interaction**

1155 The activity involved in making use of a capability offered, usually across  
1156 an ownership boundary, in order to achieve a particular desired real-world  
1157 effect.

## 1158 **Interface Description Requirements**

1159 Establishes common characteristics of service interface descriptions.  
1160 These requirements address areas such as required interface contents,  
1161 naming rules, documentation rules, and specification of a standard  
1162 structure and format for descriptions.

## 1163 **Interceptors**

1164 Interceptors are capabilities that receive a message and use the message  
1165 content to trigger a secondary action; generally, the interceptors pass the  
1166 message unaltered to the next step in a process.

## 1167 **Intermediaries**

1168 Routers and transformers are collectively called intermediaries. This term  
1169 indicates that routers and transformers generally sit between other  
1170 services and “mediate” the interaction by managing the transmission of  
1171 messages between them or by reformatting messages in transit.

## 1172 **Justice Reference Architecture**

1173 JRA is an abstract framework for understanding significant components  
1174 and relationships between them within a service-oriented environment. It  
1175 lays out common concepts and definitions as the foundation for the  
1176 development of consistent service-oriented architecture (SOA)  
1177 implementations within the justice and public safety communities. The  
1178 term refers to the modular architecture that cleanly and appropriately  
1179 identifies and separates technical and governance layers so that standards  
1180 can be developed to improve interoperability. JRA is being developed by  
1181 Global; it leverages the work of others, such as the state of Washington,  
1182 and builds upon the work of OASIS.

### 1183 **Messages**

1184 The entire “package” of information sent between service consumer and  
1185 service (or vice versa), even if there is a logical partitioning of the message  
1186 into segments or sections.

### 1187 **Message Definition Mechanisms**

1188 Establishes a standard way of defining the structure and contents of a  
1189 message; for example, Global JXDM- or NIEM-conformant schema sets.  
1190 Note that since a message includes the concept of an “attachment,” the  
1191 message definition mechanism must identify how different sections of a  
1192 message (for example, the main section and any “attachment” sections)  
1193 are separated and identified and how attachment sections are structured  
1194 and formatted.

### 1195 **Message Exchange Patterns**

1196 Identifies common sequences of message transmission between service  
1197 consumers and services. They provide a label to a series of message  
1198 transmissions that have some logical interrelationship.

### 1199 **Message Validators**

1200 An intermediary that examines a message to ensure that the contents  
1201 adhere to established business rules.

### 1202 **Orchestrations**

1203 A capability that coordinates interaction with multiple services. An  
1204 orchestration is often implemented using an open industry standard  
1205 implementation mechanism (referred to as an orchestration mechanism in  
1206 JRA), which allows the implementation to be shared across tools and  
1207 platforms.

### 1208 **Process Model**

1209 The characterization of the temporal relationships between and temporal  
1210 properties of actions and events associated with interacting with the  
1211 service.

---

**1212 Provider Systems**

1213 The information system that offers the use of capabilities by means of a  
1214 service.

**1215 Provisioning Models**

1216 The responsibility/models for making a service available to customers in a  
1217 manner consistent with formal (or occasionally informal) customer  
1218 expectations.

**1219 Reachability**

1220 The ability of a service consumer and service provider to interact.  
1221 Reachability is an aspect of visibility.

**1222 Real-World Effects**

1223 The actual result(s) of using a service, rather than merely the capability  
1224 offered by a service provider.

**1225 Reference Architecture**

1226 A reference architecture is an architectural design pattern that indicates  
1227 how an abstract set of mechanisms and relationships realizes a  
1228 predetermined set of requirements.

**1229 Reference Model**

1230 A reference model is an abstract framework for understanding significant  
1231 relationships among the entities of some environment that enables the  
1232 development of specific reference or concrete architectures using  
1233 consistent standards or specifications supporting that environment.

1234 A reference model consists of a minimal set of unifying concepts, axioms,  
1235 and relationships within a particular problem domain, and is independent  
1236 of specific standards, technologies, implementations, or other concrete  
1237 details.

**1238 Repository**

1239 Stores models and interface descriptions in a central location that is  
1240 accessible to appropriate stakeholders. A repository will permit searching  
1241 for models and interface descriptions based on a range of identifying  
1242 criteria. A repository will also map logical service identifiers with physical  
1243 addresses.

**1244 Routers**

1245 A capability that receives a message, examines it, and transmits it to one or  
1246 more destinations based on the contents. In general, routers can be  
1247 designed to operate on any of the information contained within the  
1248 message; they may use information about the origin of the message,  
1249 routing directive information contained within the message or the main  
1250 content of the message itself.

---

**1251 Services**

1252 The means by which the needs of a consumer are brought together with  
1253 the capabilities of a provider.

**1254 Service Agreements**

1255 A document that establishes policies and contractual elements for a given  
1256 interaction or set of interactions (that is, for one or more services).

**1257 Service Consumers**

1258 An entity that seeks to satisfy a particular need through the use of  
1259 capabilities offered by means of a service.

**1260 Service Contracts**

1261 An agreement by two or more parties regarding the conditions of use of a  
1262 service.

**1263 Service Design Principles**

1264 The documentation to provide consistent guidance regarding the overall  
1265 partitioning of capabilities into services and the relationships between  
1266 services.

**1267 Service Interaction Profiles**

1268 Defines a family of industry standards or other technologies or techniques  
1269 that together demonstrate implementation or satisfaction of:

- 1270 ○ Service interaction requirements.
- 1271 ○ Interface description requirements.
- 1272 ○ Message exchange patterns.
- 1273 ○ Message definition mechanisms.

1274 Service interaction profiles are included in JRA to promote  
1275 interoperability without forcing the organization to agree on a single way  
1276 of enabling service interaction. Each service interface will support a single  
1277 profile; a service will have multiple interfaces if it supports multiple  
1278 profiles.

**1279 Service Interaction Requirements**

1280 Define common rules of service interaction. Typically, these requirements  
1281 are nonfunctional in nature, in that they are not directly related to the  
1282 capability used by the service consumer, nor are they related to the real-  
1283 world effect resulting from use of that capability. Rather, the  
1284 requirements enforce (or support the enforcement of) policies or contracts  
1285 or otherwise protect the interests of particular business partners or the  
1286 business organization overall.

**1287 Service Interfaces**

1288 The means by which the underlying capabilities of a service are accessed.

---

**1289 Service Model**

1290 Interaction depends on two things. First, the designers of potential  
1291 consumers need to be able to find services and, once found, establish a  
1292 physical interaction mechanism with them. Second, the designers of  
1293 potential consumers need a description of the actions that can be  
1294 performed on a service, as well as the structure and meaning of  
1295 information exchanged during the interaction. These needs are addressed  
1296 by the concept of a service's information model and behavioral model,  
1297 collectively called service models in JRA.

**1298 Service Modeling Guidelines**

1299 Documents guidelines for services provided and consumed among  
1300 partners. It provides guidance as well as compliance information  
1301 regarding the modeling and description of services to promote  
1302 consistency.

**1303 Service-Oriented Architecture (SOA)**

1304 Service-Oriented Architecture is a paradigm for organizing and utilizing  
1305 distributed capabilities that may be under the control of different  
1306 ownership domains. It provides a uniform means to offer, discover,  
1307 interact with, and use capabilities to produce desired effects consistent  
1308 with measurable preconditions and expectations.

**1309 Service Policies**

1310 A statement of obligations, constraints, or other conditions of use,  
1311 deployment, or description of an owned entity as defined by any  
1312 participant.

**1313 Service Providers**

1314 An entity (person or organization) that offers the use of capabilities by  
1315 means of a service.

**1316 Transformers**

1317 A capability that receives a message and transforms it into another format  
1318 before transmitting it on to another destination.

**1319 Visibility**

1320 The capacity for those with needs and those with capabilities to be able to  
1321 interact with each other.

**1322 Willingness**

1323 A predisposition of service providers and consumers to interact.

1324

## References

1325

1326

1327

**Erl**

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

1328

1329

1330

**GISWG**

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

1331

1332

1333

**Patterns**

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

1334

1335

1336

1337

**Sholler**

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

1338

1339

1340

**SOA-RM**

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

1341

1342

1343

1344

1345

1346

1347

1348

## Document History

1349

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



---

<b>Date</b>	<b>Version</b>	<b>Editor</b>	<b>Change</b>
			numbers
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

1350

1351