Assessment Report

California Disposition Reporting Improvement Project (CA-DRIP):

Application of National Guidelines and Standards

November 2012

Prepared by

SEARCH, The National Consortium for Justice Information and Statistics
and
National Center for State Courts

Introduction

Purpose

The purpose of this document is to assess the application of national guidelines and standards by the California Disposition Reporting Improvement Project (CA-DRIP).[1] This assessment has several objectives:

1. Inform CA-DRIP stakeholders of the benefits and advantages of leveraging standards-based integration methods described in the Global Reference Architecture (GRA)[2] and how their current approach aligns with basic concepts, components, and processes defined in GRA.
2. Describe how the current CA-DRIP service development techniques, infrastructure, and supporting documentation align with GRA guidelines.
3. Identify any gaps or inconsistencies between the CA-DRIP architecture and the GRA, and describe the near-term and long-term impacts of these gaps.
4. Provide recommendations and strategies to address any conformance gaps.

GRA conformance is desirable because it encourages stakeholders to implement technology solutions that are agile, interoperable, and vendor-neutral. The use of service oriented architecture (SOA) concepts and open industry standards can also lead to adopting inexpensive tools and technologies, as well as easing the design and development of information exchanges envisioned by CA-DRIP.

By following the GRA/SOA, justice partners can significantly reduce the technology dependencies that impede their ability to respond effectively to policy or business process changes. Adopting GRA/SOA provides stakeholders with the flexibility to make incremental adjustments that limit the “ripple-effect” or impact of changes on partner systems.

A primary means to support this agility is sharing common infrastructure that facilitates information exchange by decoupling a “point-to-point” integration approach. A GRA-conformant shared infrastructure preserves an agency’s autonomy while supporting their information sharing business needs.

The GRA is built upon well-established open industry standards. This allows information sharing partners to use a range of technologies, systems, and products—regardless of the vendor or platform. These standards establish a common set of rules by which information partners share information and eliminate the need for the “my way or your way” discussion among technology staff. They reinforce the notion that individual agencies can preserve their autonomy and procure technology solutions to meet their internal and integration needs.

[1] The CA-DRIP initiative is a collaborative effort among the California Administrative Office of the Courts (AOC), California Department of Justice (DOJ), and Santa Clara County to improve disposition-related business processes through policy and information technology enhancements.

[2] See http://www.it.ojp.gov/GRA

Assessment Process

SEARCH, The National Consortium for Justice Information and Statistics,[3] and the National Center for State Courts (NCSC)[4] developed this assessment based on the following research and resources:

- the project documentation portal[5]
- the California Judicial Branch Justice Partner website[6] for Administrative Office of Courts (AOC) Data Exchange (DX) and E-services information
- conference calls, meetings, and individual stakeholder discussions.

The assessment team reviewed the following two AOC DX disposition data exchange specifications data sheets,[7] which are included in the AOC's broader Data Integration Initiative (which has defined a total of 121 court-related exchange specifications):

1. "Court Sends Initial or Subsequent Disposition Notification (DOJ901)"
2. "Court Receives Disposition Error Report Notification (DOJ802)"

The team also reviewed two other AOC documents as part of this assessment: the *Data Exchange Functional Design Guidelines*[8] and *CCMS Data Exchange Common Technical Requirements*.[9]

With this information, SEARCH and NCSC staff conducted a comparative analysis based on these GRA planning and technical documents:[10]

- *GRA Framework*, v1.9
- *GRA Guidelines for Identifying and Designing Services*, v1.1
- *GRA Service Specification Guidelines*, v1.0.0
- GRA Service Specification Package, v1.0.0 (a package of 12 templates)
- *GRA Execution Context Guidelines*, v1.1
- *GRA Web Services Service Interaction Profile*, v1.3

[3] See http://www.search.org

[4] See http://www.ncsc.org/

[5] AOC administers the Judicial Council of California portal that is available for CA-DRIP stakeholders to store, retrieve, and manage project-related documentation: http://jccprojects.webexone.com

[6] See http://www.courts.ca.gov/partners/integration.htm

[7] These specifications are posted on the Judicial Branch Justice Partner website: http://www.courts.ca.gov/partners/1614.htm; both data sheets were updated April 2012.

[8] See http://www.courts.ca.gov/partners/documents/data_exchange_functional_design_guidelines.pdf

[9] See http://www.courts.ca.gov/partners/documents/dx_common_technical_requirements.pdf

[10] These documents, among others, are available on the Global Standards Package site: http://www.it.ojp.gov/gsp

Assessment Observations and Recommendations

Alignment with Overarching GRA Principles

At its core, the GRA is a collection of requirements and agreements among stakeholders that determine how justice partners will share information. The GRA includes a set of concepts (**principles**) that guide the subsequent business and technical requirements, which are summarized below:

- Independence and Diversity of Information Sharing Partners and Architectures – Integration solutions must recognize each partner is an autonomous entity with unique responsibilities and technology solutions.
- Scalability – Integration solutions should be equally applicable for all jurisdictions, regardless of size.
- Agility – Information sharing solutions should accommodate policy, information flow, and supporting technology changes.
- Reuse and Sharing of Assets – Promote use of existing systems, interfaces, exchanges and infrastructure to support business requirements.
- Alignment with Best Practices and Experiences – Leverage previous experience, best practices, and lessons learned from related initiatives.

The architecture described on the AOC Data Integration website[11] contains many artifacts similar to those suggested by GRA for state integrated justice architectures—specifically the *DX Functional Design Guidelines* and the *DX Common Technical Requirements*. These documents provide stakeholders with a thorough description of the integration approach and Web service functionality for the data exchange specifications. ***Overall, SEARCH and NCSC find that the CA-DRIP initiative and supporting AOC DX architecture aligns with the high-level and overarching principles described in the GRA.***

The CA-DRIP/AOC DX initiative proposes Web Services as the primary means to exchange information—which is an excellent starting point for GRA conformance—and has formalized these Web Services into discrete and well-documented specifications. These specifications also include instructions and guidance to use the Integration Services Backbone (ISB) as the shared infrastructure to facilitate message brokering, translation, and routing among exchange partners.

*Look for the **blue arrow** and blue-shaded text for specific recommended or suggested actions from the assessment team*

[11] http://www.courts.ca.gov/partners/1614.htm

Service Identification and Design

Service developers use the principles outlined in GRA service identification and design guidelines (SIDG) to properly identify, factor, and design services that conform to the GRA. Global has developed a document to guide this activity: *GRA Guidelines for Identifying and Designing Services,* v.1.1.[12] Analysts and developers can follow the process it describes to identify the capabilities that exist in a justice enterprise, document the interactions among these partners, and establish a process to identify and prioritize service development and implementation.

The Service Identification methodology defined in the GRA includes the "decomposition" of business activities to identify the "capabilities" needed to produce the "real world effects." For example, a court supports a capability of "disposing of a charge," with the real-world effect of reporting that charge disposition to other partners (among other things). Focusing on business capabilities is beneficial because, while organizational structures and business processes are situational, the essential capabilities and requirements of businesses tend to remain constant over time. Defining a business capability abstracts the people, processes, and procedures associated with a given business function. The decomposition of the business into capabilities provides the *decoupling* of business activities from these processes and procedures to identify the underlying services to be implemented.

Another essential component of the service identification process is to document the business process and supporting information flows among stakeholders for each line of business. Typically, these processes are modeled in a graphical notation that allows practitioners to understand and identify the service candidates for a given business scenario

Building upon the service identification processes (capability and business models), the GRA provides additional guidance in order to effectively describe and scope service actions. The GRA service design principles "provide consistent guidance regarding the overall partitioning of capabilities into services and the relationships between services."[13] The GRA defines a set of service design principles that the assessment team applied to the DOJ901 and DOJ802 exchange specifications. Based on this review, the assessment team determined that AOC DX specifications are roughly equivalent to GRA service specifications.[14]

SEARCH and NCSC find that the DOJ901 and DOJ802 specifications are generally consistent with the GRA SIDG, although the team suggests a few enhancements to improve the loose coupling and reuse potential.

[12] See http://it.ojp.gov/docdownloader.aspx?ddid=1171

[13] *GRA Guidelines for Identifying and Designing Services*, at pp. 4–5.

[14] The Global Standard Council's "Service Specification Evaluation Worksheet," provided as an appendix to this document, explains each principle in detail and offers guidance on the application of the principles to specific services.

CA-DRIP stakeholders should consider—

1) refactoring the services using business terminology and
2) eliminating unique message types within the payload to better align with the SIDG.

Both the DOJ901 and DOJ802 specifications target specific integration partners. The naming convention used for these exchanges reflects this limited use: “Courts Send Disposition to DOJ,” or “DOJ Sends Error to Courts.” Following this convention inherently limits other opportunities to reuse these exchanges by other potential partners. For example, correctional agencies may want or need disposition information for intake processing, law enforcement may want dispositions to reduce evidence inventory, or the DOJ may need to send error reports to a law enforcement agency to correct an error on the record.

The CA-DRIP stakeholders should consider revising the data exchange documentation and titles to make these exchanges more abstract and open to other opportunities for reuse.

DOJ901 also uses several message types within the payload content to determine what the service consumer intends to do with the exchange. This approach is intended to distinguish the *type* of interaction, but this approach departs from several of the design principles. By embedding message types within the message content, the precise use of the service is obscured and will require additional implementation details in order to understand how the message type element works (the principle of abstraction). It also means that the service consumer is dependent upon processes provided by the service provider that are beyond the consumer’s control (principles of loose-coupling and autonomy). It also can increase the amount of programming required by service providers and service consumers to support business processes, as opposed to defining these as “actions” at the service level.

The assessment team understands that the CA-DRIP stakeholders are considering including another “message type” in the existing DOJ901 specification to confirm that an arrest record exists in the criminal history repository at the time of charge filing at the court.

Carefully evaluate this change in light of the service design principles.

Does the behavior of this “action” fit with the other service actions (the principle of cohesion)? The similarity of the data (the principle of data affinity) is also a consideration in this design decision. There is seldom one single “right” answer, and stakeholders should balance the action behavior and these other principles to ensure the service is properly factored.

Service Specification Packages

GRA Service Specification Package (SSP): A structured set of documents	
What's contained in an SSP?	• A service description • A service interface description • Service interaction requirements • A service profile • Policies • Contracts • Metadata • Catalog • Change Log
SSP documents…	• Collectively describe the service from both a business *and* technical perspective • Are provided in both human- and machine-readable format, as appropriate • Include associated metadata to be used as part of a service registry/repository
Service Description Document	• Describes the behavior of the service in a human-readable (business) form.
Service Interface Description	• Describes the service interaction in human-readable form, and • Includes machine-readable documents, such as Web Service Description Language (WSDL) and Extensible Markup Language (XML) schema
Service Interaction Requirements	• Express business requirements, such as Authentication, Authorization, Message Integrity, etc. (See footnote 15.)

The previous discussions about the GRA/SOA principles and Service Identification and Design principles provide *guidance* on how best to identify, design, and develop services. The following discussion of three specific SSP elements—**service descriptions**, **service interface descriptions**, and **service interaction profiles**—provides greater detail on how to develop services that *conform to the GRA* and contains normative language that identifies *specific GRA requirements*.

Service Descriptions

A GRA-conformant **Service Description Document** (SDD) provides a business-oriented overview of a service's characteristics. SDDs consist of four sections that define and capture service requirements:

1. Service Overview – This section captures the purpose, scope, capabilities, real-world effects, security classification, and the SSP version.

2. Business Scenarios – This section captures the entities and their roles, along with primary and alternative information flow descriptions and supporting diagrams.

3. Service Interoperability Requirements – This section defines the key business requirements that will shape how the subsequent technical solutions will implement them. These include the service interaction requirements,[15] assumptions, dependencies, execution context,[16] policies/contracts, additional security requirements, privacy requirements, and any other requirements that may not "fit" within the other requirements.

4. Service Model – This section provides links to the information model artifacts (the Information Exchange Package Documentation (IEPD),[17] including National Information Exchange Model (NIEM)[18] schemas, extension schemas, mapping spreadsheets, and sample instances). This section also defines the behavior model which details the specific service actions and business process models.

The assessment team determined that the service descriptions contained in both the DOJ901 and DOJ802 specifications include the majority of the information outlined in the GRA guidelines, but are incomplete when compared to the current version of the GRA Service Specification Guidelines[19] because they do not contain a catalog file or references to service models (i.e., the IEPD, or message payload). Aside from those details, the Service Description Documents conform to the GRA guidelines.

Service Interface Descriptions

The **Service Interface Description Document** (SIDD) contains much of the same information as the Service Description document, as the concepts overlap, but the SIDD elaborates on the technical details of how the business or conceptual requirements will be met. The SIDD is the source of information that defines the specific standards, policies, and interactions that must be supported in a given service. The SIDD transitions from business requirements to technical and implementation requirements.

The CA-DRIP specifications the assessment team reviewed did not include a specific document that met the requirements of a GRA SIDD, but much of the same type of information is contained in the Common Technical Requirements document.

[15] Service interaction requirements express business requirements, such as Authentication, Authorization, Message Integrity, Confidentiality, Addressing, Reliability, and Transaction Support. Business stakeholders can use these to indicate if the specific requirement applies to the service. The example SDD on the Global Standard Package site includes this list of standard requirement offerings: http://www.it.ojp.gov/gsp

[16] The Execution Context section defines the infrastructure necessary to ensure the service is visible and available to its consumers.

[17] See https://www.niem.gov/training/Documents/Mod08_NIEM_PI_IEPD_Concepts.pdf

[18] See http://www.niem.gov

[19] The current version of the Service Specification Guidelines was published in May 2012 and unavailable when the DOJ901 and DOJ802 SDDs were established. For the current Guidelines version, see http://it.ojp.gov/docdownloader.aspx?ddid=1215

The CA-DRIP stakeholders should consider establishing a SIDD for each data exchange specification.

Service Interaction Profiles

A GRA **Service Interaction Profile** (SIP) is an approach to meeting basic interaction requirements (defined in the SIDD) between service consumers and providers. This approach uses a cohesive group of technologies, standards, or techniques in meeting those basic interaction requirements. Two of the three GRA profiles use Web Services (WS) technologies:

- The **Web Services SIP** implements a basic WS interface. It includes the basic WS industry standards, such as such as SOAP (Simple Object Access Protocol), WS-Security, and WS-ReliableMessaging.
- The **Reliable-Secure Web Services SIP** implements a more robust and secure WS interface. It builds upon the basic Web Services SIP to add more "advanced" WS industry standards such as WS-SecureConversation.

SEARCH and NCSC encourage the CA-DRIP stakeholders to adopt the GRA Web Services SIP for their Web Services-based exchanges.

The AOC DX approach employs Web Services technologies, but no documentation demonstrated that this approach conforms to either the Web Services or Reliable Secure Web Services SIPs. This observation is based on a review of the California AOC Court Case Management System (CCMS) Common Services Header Schema, described in the Common Technical Requirements document.[20] The Common Services Header Schema is intended to address many of the GRA interaction requirements, although it does this without using the GRA-approved mechanisms. For example, the ISB uses a combination of elements contained in the Common Service Header Schema such as Source, Target, and Interface Name to perform the functional equivalent of the Service Authentication/Authorization requirements, but it does this outside of the GRA WS-SIP specification. Similarly, the combination of Document Type, Correlation ID, ISB Transaction ID, and Distribution ID will provide functionality comparable to the WS-Addressing standard. Also, unlike the GRA design methodology, there is no clear linkage that shows how the individual components in the Common Technical Requirements support a specific business needs.

This is not to say that the approach adopted by the AOC will not work, but it simply indicates that this approach may not conform to the GRA WS-SIP requirement of using WS-Addressing. During a recent meeting, AOC staff confirmed that the Common Technical Requirements do not use the WS-Addressing standard, and plan to incorporate this and other WS-* standards in the near future.

[20] See http://www.courts.ca.gov/partners/documents/dx_common_technical_requirements.pdf, at pp. 11–13.

This is a very positive step, and NCSC and SEARCH encourage the CA-DRIP stakeholders to continue pursuing full adoption of the Web Services stack.

The current AOC DX/DRIP approach may require integration partner programmers to perform additional development efforts to implement the exchange based on AOC's unique requirements rather than an industry standards-based approach as recommended by the GRA.

The use of industry standards would significantly reduce the development process by using available tools and reference materials. Standards-based solutions also provide confidence that the solutions have a certain degree of durability and applicability in other integration scenarios (with other partners, who presumably will choose standards-based solutions).

NIEM Conformance

Finally, GRA conformance requires the use of NIEM for the message payload.

While the AOC/DRIP specifications use some NIEM element and structure (type) names, the data exchanges do not fully conform to NIEM. The two primary NIEM conformance issues relate to incorrect use of the NIEM namespace and the reinvention of NIEM data concepts.

1. Misnaming NIEM namespaces – the exchanges have renamed the NIEM schema target namespaces from standard namespace names to implementation-specific names. This violates the NIEM's Naming and Design Rules (NDR).
2. Some of the exchanges define data concepts that are part of the NIEM standard data model. For example, DOJ802 defines a complex type called "OrganizationContactInformationAssociationType" in the extension schema, but this concept exists in the NIEM core.

Instead of re-defining this concept, DOJ802 should reference it from the NIEM core.

Summary

The SEARCH and NCSC assessment team determined that the CA-DRIP and California AOC architecture meets the majority of the GRA requirements and guidelines and can be fully conformant by adopting the recommendations outlined in this assessment. The CA-DRIP stakeholders have indicated they fully support the adoption of national standards and recognize the long-term value and return on the investment of implementing standards-based integration solutions.

Appendix A

Global Standards Council (GSC)

GRA Service Specification Evaluation Worksheet (Version 1.0)

The GSC will be provided with the *Service Description Document* (SDD) along with this worksheet to evaluate each Service Specification.

To be completed by requestor:

Service Name	
Service Version #	
Requestor	
Requestor Email	
Date	

To be completed by GSC:

Assigned to	
Date Completed	
General Comments	
Additional Questions	
GSC Recommendation	

General Questions

Item	Rating	Comments
Potential for Implementation and Adoption Is the service likely to be used in a large number of jurisdictions?	☐ Yes ☐ No ☐ Not Sure	
Factorization Is the service properly factored within the context of business processes and information flows associated with the service interactions? This is largely a general summary of application of the design principles, detailed below. However, this is a place to document general design issues or problems with the surrounding business process.	☐ Yes ☐ No ☐ Not Sure	

GRA Design Principles

(Instructions: Review the service's adherence to the following GRA principles using a score of 1-Poor, 2-Fair, 3-Good, 4-Very Good)

Item	Rating	Comments
Reusablility To achieve reusability, logic is divided into services with the intention of promoting **reuse**. Service orientation encourages reuse in all services. Applying design standards that make each service potentially reusable increases the chances of being able to accommodate future requirements with less [service]. Issues to watch for that indicate reusability problems: • Has the name or scope of the service been artificially narrowly scoped? Could the service potentially provide benefit to other business functions or add value to other exchanges? • Does anything in the description unnecessarily constrain the context in which the service operates? • Does the service description indicate a focus on an immediate project issue or objective, rather than a broader enterprise business problem domain? • Does the service description unnecessarily specify specific groups, systems, agencies, etc., that can use or benefit from the service? • Could other business processes or consumers use the information or functionality of the service in the future, aside from the immediate project stakeholders or objectives? • Does the service have the exact same (or mostly the same) information model as another service? Reusability issues often occur when an SSP development effort is chartered with *information* in mind, rather than a business *process* or business *function*. It can also sometimes arise when an SSP is simply intended to "wrap" an existing IEPD. There is nothing wrong with wrapping an IEPD per se, but it is necessary to consider if the exchange represented by the IEPD should be refactored to be more broadly usable.		

Item	Rating	Comments
Loosely Coupled Loosely coupled services maintain a relationship that minimizes dependencies and requires only that they maintain an awareness of each other. **Loose coupling** is a condition wherein a service acquires awareness and knowledge of another service while still remaining independent of that service. One of the fundamental requirements of the GRA, the requirement for agility, is directly supported by establishing a loosely coupled relationship between services. Loose coupling and reusability are tightly linked. One usually implies the other. Issues to watch for that indicate tight coupling: • If the service has a dependency on another service for the project or scenario at hand, will that dependency exist for all projects and scenarios? If not, then consider encapsulating the dependency in a separate service so that they can be reused independently. • Is the service unnecessarily dependent (via language in the description) on being provisioned by a specific agency or at a specific location? • Are the dependencies between services stated as minimally as they can be? If service A has two actions and half of its consumers use one action and the other half use the other, then the actions are tightly coupled (and should be defined in separate services). • Is the scope of the service as narrowly defined as possible? (Services should do one thing and do it well.) • All of the actions on a service should typically be implemented together, or not at all. • Does the service's description contain any details or assumptions about the implementation or nature of consumers? • Does the name of each action on the service effectively convey what that action does? Avoid generic action names like "performExchange" or "input," because then consumers will need to rely on (that is, become coupled to) information outside of the action definition.		

Item	Rating	Comments
Abstraction The principle of abstraction allows services to act as black boxes, hiding their details from the outside world. The scope of logic represented by a service significantly influences the design of its actions and its position within a process. The scope of logic a service represents is influenced by the principle of **service abstraction**. Issues to watch for that indicate abstraction problems: • Does the name of the service, or the description, contain the name of the system(s) that will support the implementation of the service? • Does the name of the service, or the description, reveal how the implementation *works*, rather than what the service *does*? • If the organization providing the service wishes to change how it does so, will the service description need to change in order to remain accurate/current? • Is the service focused on information rather than functionality? (Sometimes the distinction is subtle.) Watch for words in the service name like “XYZ Exchange Service” or “ABC Information Service.” Information exchange is a means to an end, not an end in itself. Generally speaking, it is useful to combine the *noun* of the information being exchanged, with a *verb* that encapsulates the real-world effect. For example, if the subject of an exchange is “warrants,” it would be more appropriate to identify a “Warrant Issuance” or “Warrant Reporting” or “Request for Service of Warrant” service, than a “Warrant Exchange” service.		

Item	Rating	Comments
Composability Under the principle of **composability**, collections of services can be coordinated and assembled to form composite services. A service can represent any range of logic from various types of sources, including other services. The main reason to implement this principle is to ensure that services are designed so that they can participate as effective members of other service compositions, when required. Composability is simply another form of reuse; therefore, actions need to be designed in a standardized manner and with an appropriate level of granularity to maximize collaboration opportunities. If services are reusable, autonomous, loosely coupled, and cohesive, they will generally be composable. When considering this principle, though, watch specifically for language that constrains the use of the service to particular contexts.		

Item	Rating	Comments
Autonomy **Autonomy** requires that the range of logic exposed by a service exist within an explicit boundary. As a result, services have control over the logic they encapsulate. This principle allows a service to execute self-governance of all its processing. It also eliminates dependencies on other services, which frees a service from ties that could inhibit its deployment and evolution. Autonomy problems can also force consumers/ implementers to know undocumented details about the service's behavior, or to understand how to choose among duplicate or redundant services. Issues to watch out for that indicate autonomy problems: • Does the service use underlying systems or information that are also used or managed by other services? (For reference SSPs, consider the "typical" implementation.) • Are there multiple services in the enterprise that provide the same functionality (real-world effect)? • Does the service duplicate another service's behavior and scope, with the only difference being the underlying implementation, system, or agency? • Does the service have the exact same (or mostly the same) information model as another service? If two services operate on the same data, this may impair the ability of each of them to manage that information autonomously. Duplicating behavior or operating on the same data raises the possibility that the service does not fully control or encapsulate its logic. Typically the remedy for this situation is to factor out the common behavior or data into a component service that is invoked by multiple services that encapsulate any process variation.		

Item	Rating	Comments
Cohesiveness This principle dictates that services expose functions that belong together because of their purpose. **Cohesiveness** applies to the functions a service performs and the information it manipulates and communicates. To achieve cohesiveness, a service should perform only those functions that are related to each other and be responsible for information that is semantically connected. For instance, a service that submits fingerprint information for identification and at the same time submits driver license information for driver history verification would not be cohesive. A successful approach to achieve cohesiveness is analyzing the functions and the messages a service is responsible for and making sure they are related and interdependent. Issues to watch out for that indicate cohesiveness problems: • Does the information model of the service contain many sections of unrelated or loosely related information? • Would a change to the action or information model of the service impact all consumers equally, or some consumers more than others? • Does anything in the information model of the service have little business relationship to what the service does? • Do the actions in the service's action model all act on the majority of the information model? Or are some actions related to one part of the information model, and other actions related to separate parts of the information model? • Would an implementation of the service generally implement all the actions together, or would some actions be unavailable or unimplemented depending on circumstances? If you look at a service (either the information or action model) and think, "This service has everything but the kitchen sink in it!", then there is likely a cohesiveness problem. Be careful of SSP development efforts that begin with the idea of economizing on SSP development resources by including many information exchanges in one service.		

Item	Rating	Comments
Statelessness Services should minimize the amount of **state information** they manage and the duration for which they retain it. "State information" is data specific to a current activity. While a service is processing a message, for example, it is temporarily "stateful." If a service is responsible for retaining state for longer periods of time, its ability to remain available to other requestors will be impeded. Statelessness is a preferred condition for services and one that promotes reusability and scalability. Issues to watch out for that indicate statelessness problems: • Does the service description contain language requiring that consumers invoke its actions in a particular order? • Does the service description contain language assuming that prior interactions have occurred?		