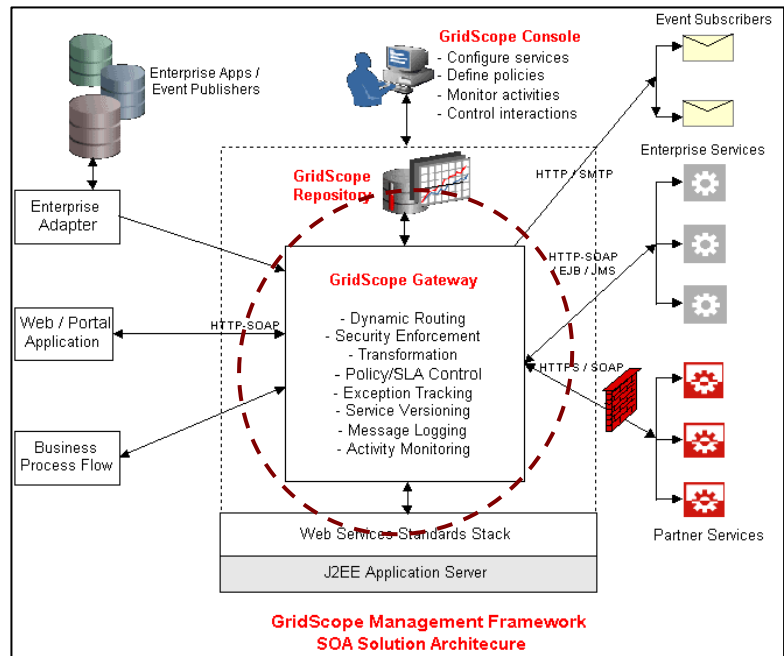


GridScope Service Gateway

The Heart of SOA Management

*A scalable, value-added
“pipeline” for processing services.*



The Gateway Approach

There are many approaches to managing business services. GridScope's design principles seek an approach that is as non-intrusive as possible. To that end, GridScope has adopted a gateway approach to apply value-added management logic to services at run-time. The GridScope Gateway provides a single, well-known access point and serves a proxy between users and service providers.

The Gateway sits on top of the business platform adding minimal overhead to service processing. Unlike an ESB, the GridScope Gateway requires no special API's or software installations on the client or server-side. It effectively isolates the users from any changes in how services are deployed. The Gateway is designed to scale horizontally across multiple servers, offering a high degree of scalability and performance for the overall SOA management layer.

The GridScope Gateway is implemented as a standard-compliant web service, eliminating the need for proprietary software. Connecting to the Gateway is as simple as changing the access point from the endpoint service binding address to the Gateway's address.

Managing vs. Operating

GridScope believes in separating the business logic from the management logic when dealing with business services. The GridScope Gateway does not attempt to address how you operate your business, just how you manage your services.

The Gateway acts as an intermediary to insert value-added management logic at run-time. This frees service and application developers to focus on the business process and business logic in their design-time environment. The Gateway can be thought of as a management execution layer that unifies management logic across all applications.

Dynamic Management for Flexibility

In today's rapidly evolving business environment, the time it takes for a business to reconfigure and provision its processes to respond to new demands can mean the difference between leading the market or being overwhelmed by competitors. A service management solution that requires recoding or service interruptions to update is simply a liability. Dynamically managing services is what gives business the flexibility to respond rapidly to changing business requirements.

GridScope's separation of the management logic and business logic significantly increases the opportunities for dynamic management of service behavior. GridScope uses declarative management logic that is 100% configuration and policy-based. Service and other configuration updates are picked up and stored automatically and applied seamlessly at runtime.

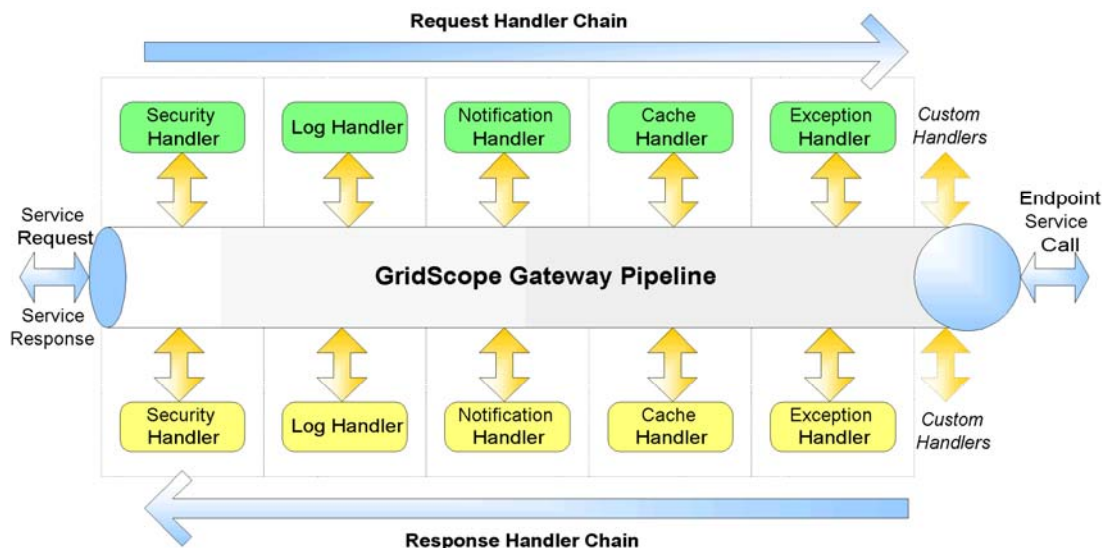
Automating Service Processing

The Gateway applies management logic at runtime by forming a virtual pipeline of compartmented management logic, called handlers, dedicated to various management functions (such as caching, transformation, and monitoring). The pipeline is designed as a series of request handlers that sequentially handle the request message on one end and a series of response handlers that sequentially handle the response message on the other end. At the center of the pipeline is the service access logic.

The handlers get the "intelligence" to dynamically execute management logic from key provisioning and control information stored in the Repository. The control information can be stored in the form of policies, SLAs, configuration profiles, etc. This control information can also be cached by the Repository to dramatically improve the processing time, reducing it to near zero.

The benefit of the GridScope Gateway is to provide a means of dynamic, run-time control of business services. This drives a secondary benefit of increasing the options and flexibility for managing business services, ultimately resulting in a more agile business enterprise.





Gateway Features

Performance Caching

Caching is performed at various levels including data, components, services, and configurations depending on where optimization is needed to achieve high performance. Caching values are customizable and can be set and adjusted dynamically to respond to changing business requirements.

Exception Management

Eliminates manual searching of multiple logs by providing centralized visibility to all types of exceptions: system, application, and business-level. Also provides

- Real-time detection of exceptions as they occur
- User-defined targeted alerting
- Rule-based routing to exception handlers
- Searchable summaries of exception events

Transformation

Transform source data into a target format based on specified transformation rules using the XSLT standard to transform the data formats.

Security and Access Control

Use web services security standards including WS-Security and SAML to provide the following security capabilities:

- Federated Authentication: intercepts service requests and adds the appropriate username and credentials. Authenticates service requests before dispatching the request for execution.
- Authorization: validates each service request to ensure the sender has the appropriate privileges.
- XML Encryption: encrypts XML content at the element level for both request and response messages.

- Replay Attack Prevention: prevents hostile replay attacks.

Logging

Done for requests and responses and can be done for each individual processing component and at various severity levels.

Monitoring

Track, in real-time, various business processes being routed and executed to provide visibility into those processes.

Service Level Agreement (SLA)-based management

Use SLAs to advertise and create service level agreements with customers. This helps them to provide the appropriate amount of resources and prioritize the servicing of customer requests. For example, a customer can agree to a WSLA that guarantees that 50 orders will be processed per second. However, anything over 50 orders will not have a guaranteed response time.

The SLA-based management system uses two key technologies

- Web Service Level Agreements (WSLA), a system that provides SLA definitions and compliance monitoring
- Web Services Middleware Management (WSMM), a system that provides run-time control mechanisms to support guaranteed performance.

Versioning

Version Web Services and proactively assist in the migration of the client applications to use new versions of the Web Service as soon as they become available. Versioning includes the ability to register different versions of the same service, providing necessary API and libraries for all versions to the client, and help in the migrating clients to the new versions. Versioning enables the transformation of requests to older versions

of the service to the newer version using XLST-based transformation. If no version number is explicitly specified, the default version of the service is invoked.

Custom Business Logic Processing

Insert custom business logic during service processing. For instance, specific business logic can be performed either on the service request prior to submitting the request for execution, or on the service response prior to sending the response back to the client.