OGC
Open Geospatial Consortium

Old Tool / New Grip: Extending OGC Services with REST & Some GCI Implications

SPG Workshop - ESIP Federation Summer Meeting
Santa Barbara, CA, USA
Josh Lieberman
July 7, 2009

Ideas

- What is a service?
- Politics, religion, and architecture: a short history
- Web, REST, and services
- What are SOAP and HTTP?
- The Resource Web and the Representation Web
- OGC services and markets
- Messaging and reflection
- Refactoring service function and binding
- Connecting the model: ART + CRUD
- Learning to like HATE(OAS)

What is a Service?

- Concepts
  - Distinct, defined functionality
  - Constrained interaction
  - Hidden implementation
  - Separation of concerns
  - Useful and reliable
  - Exchange of value
  - Reusable application component
  - “Headless” and invisible
  - Machine accessible
  - Dynamic

ISO Service Definition

- (ISO 19119:2005) Service: distinct part of the functionality that is provided by an entity through interfaces.
- Interface: named set of operations that characterize the behaviour of an entity.
- Operation: specification of a transformation or query that an object may be called to execute. It has a name and a list of parameters.
- Service Types: Human interaction, model / information management, Workflow / task management, Processing, Communication, system management
- Is this sufficiently broad “and” sufficiently shallow to define geospatial services?

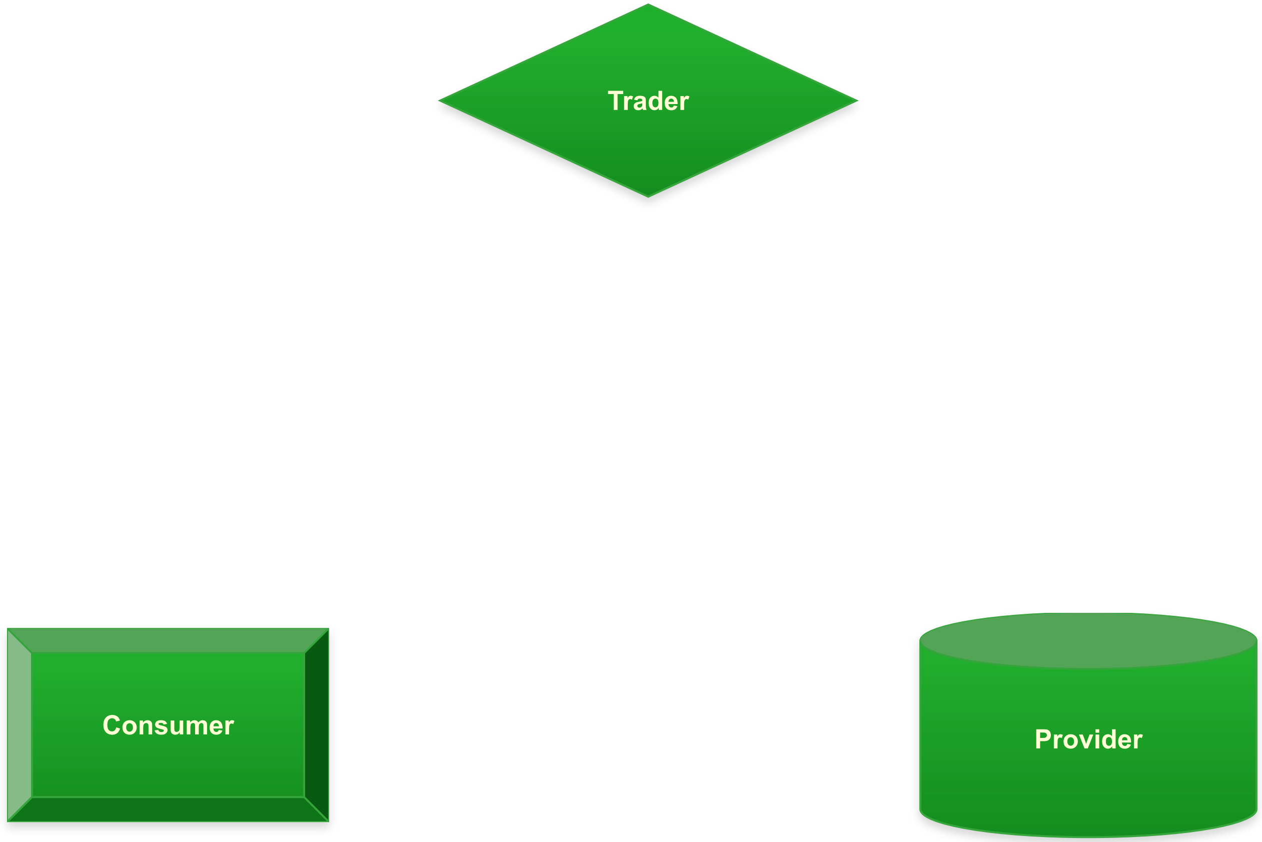
RM-ODP Interfaces

- A signal interface is an interface in which all the interactions are signals
- An operation interface is an interface in which all the interactions are operations
- stream interface is an interface in which all the interactions are flows.

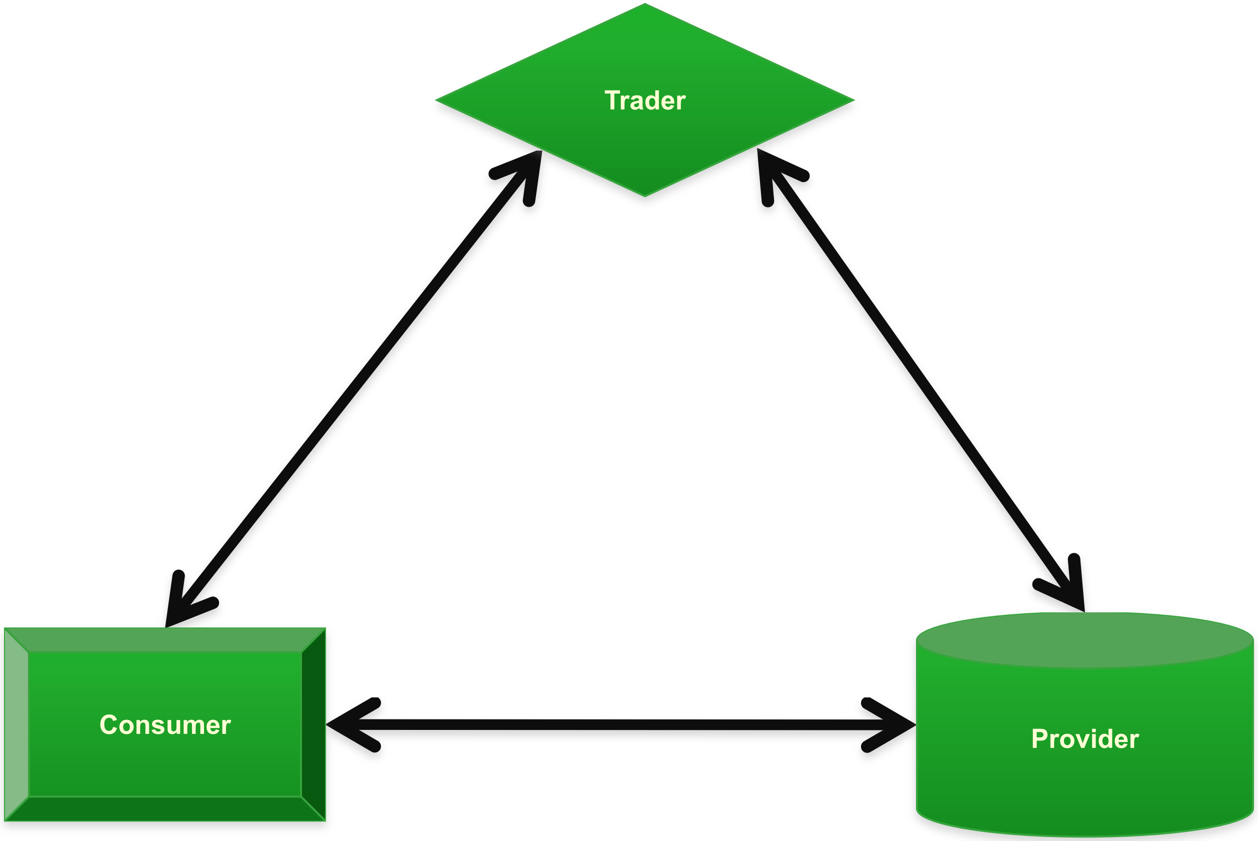
Service ORIENTED Architecture

- (OASIS Reference Architecture for Service Oriented Architecture Version 1.0): a SOA is a medium for exchange of value between independently acting participants; participants (and stakeholders in general) have legitimate claims to ownership of resources that are made available via the SOA; and the behavior and performance of the participants is subject to rules of engagement which are captured in a series of policies and contracts.
- (Wikipedia) SOA: computer systems architectural style for creating and using business processes, packaged as services, throughout their lifecycle.

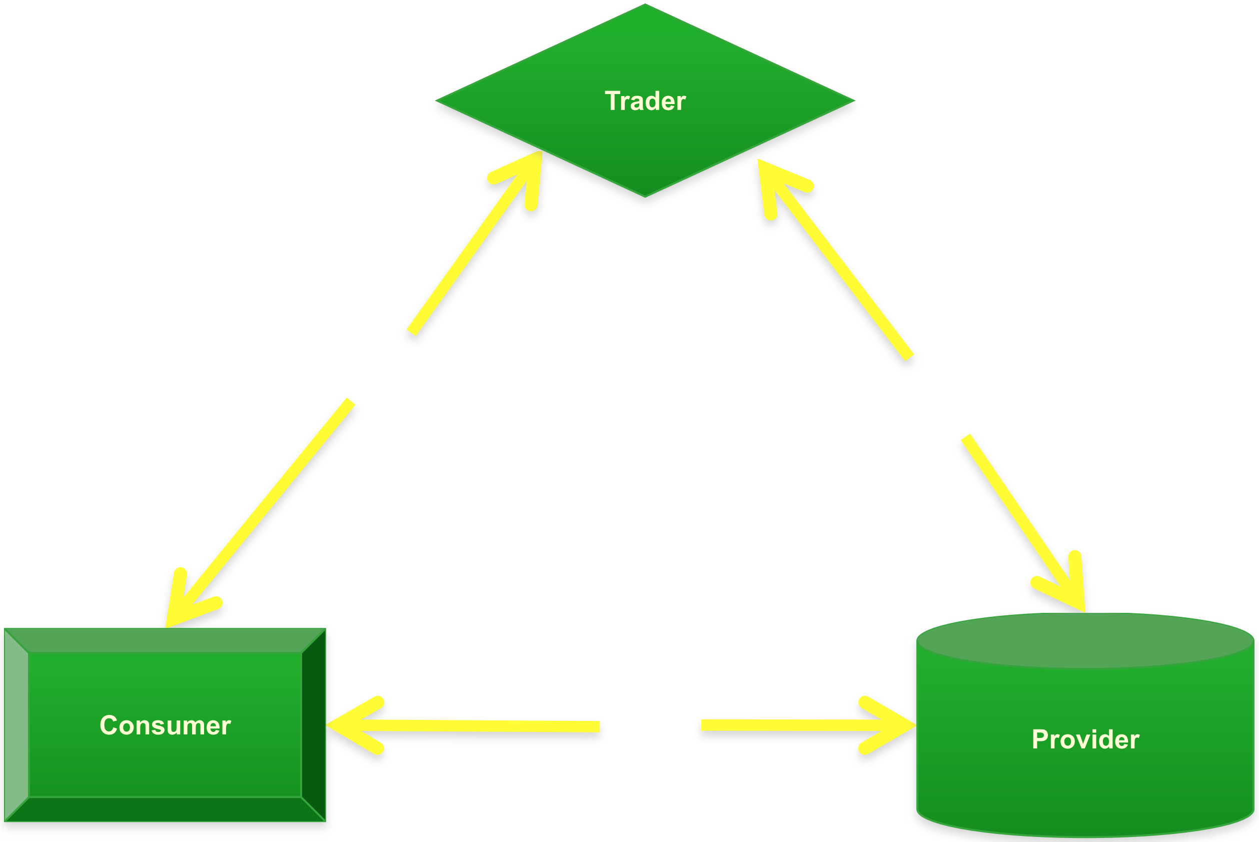
(SEM) SOA Triangle



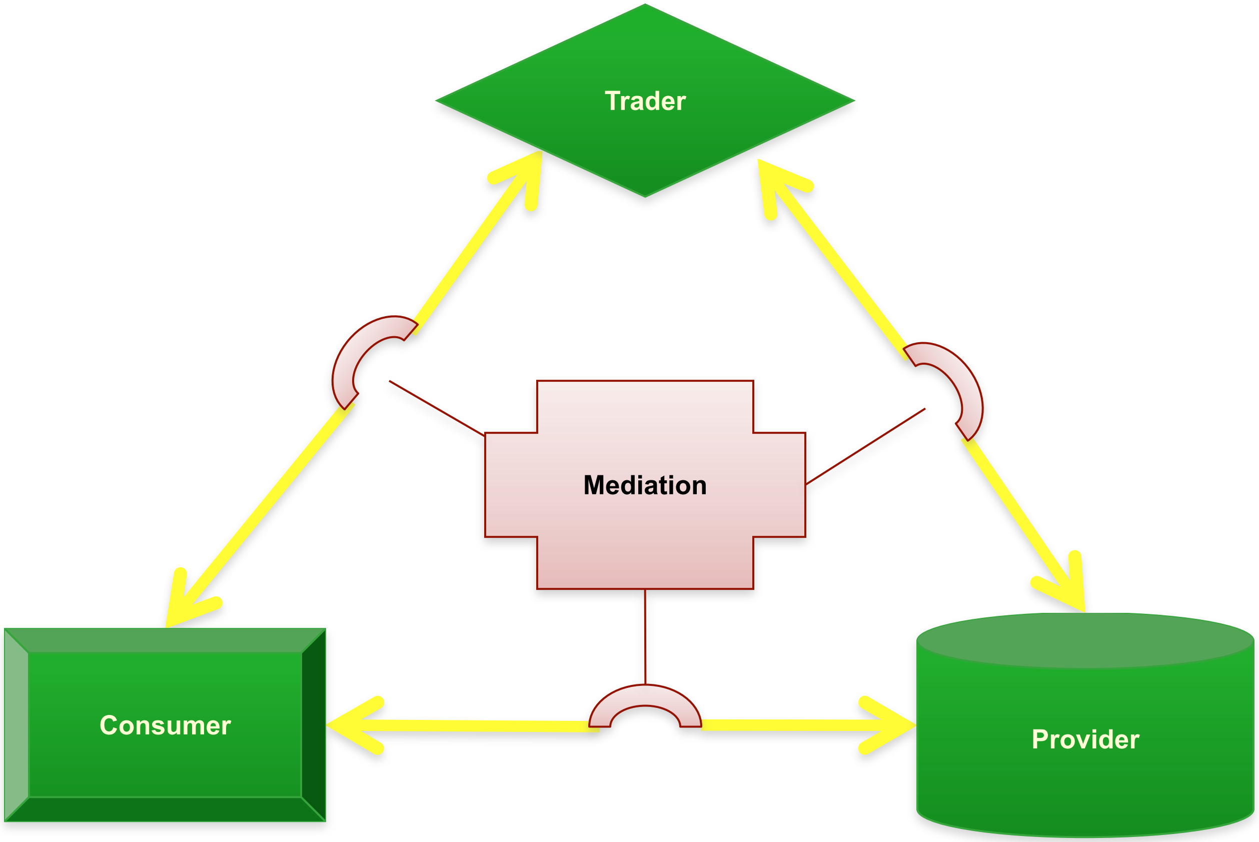
(SEM) SOA Triangle



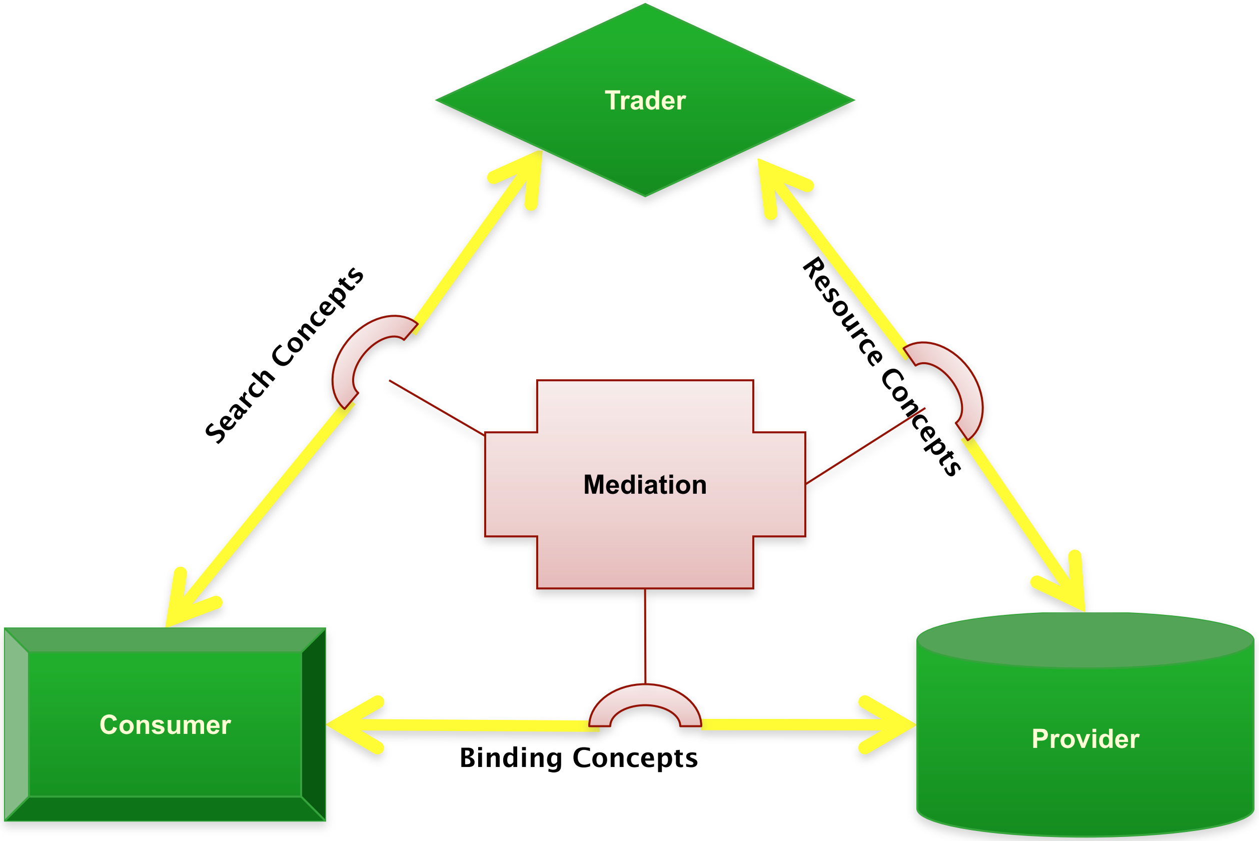
(SEM) SOA Triangle



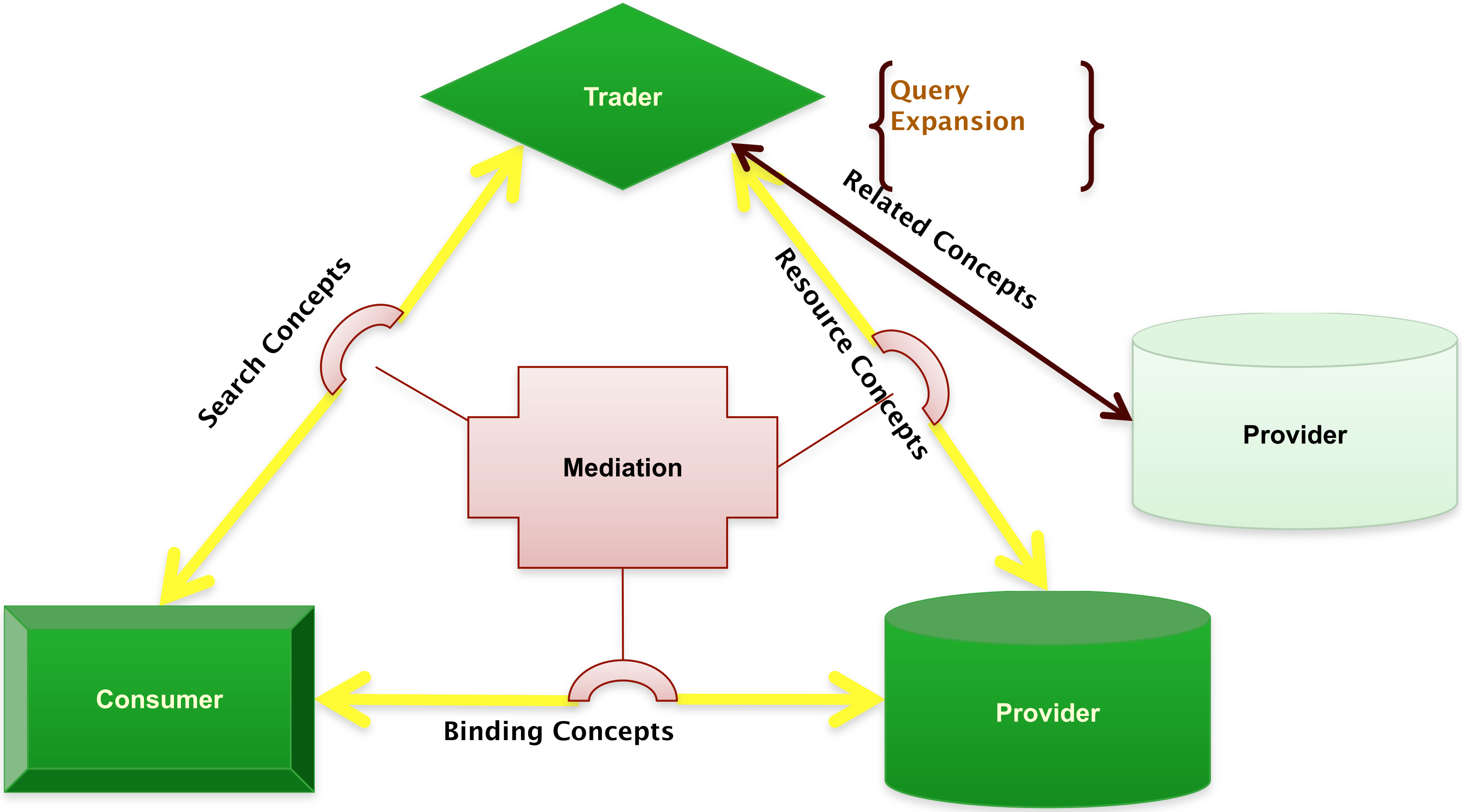
(SEM) SOA Triangle



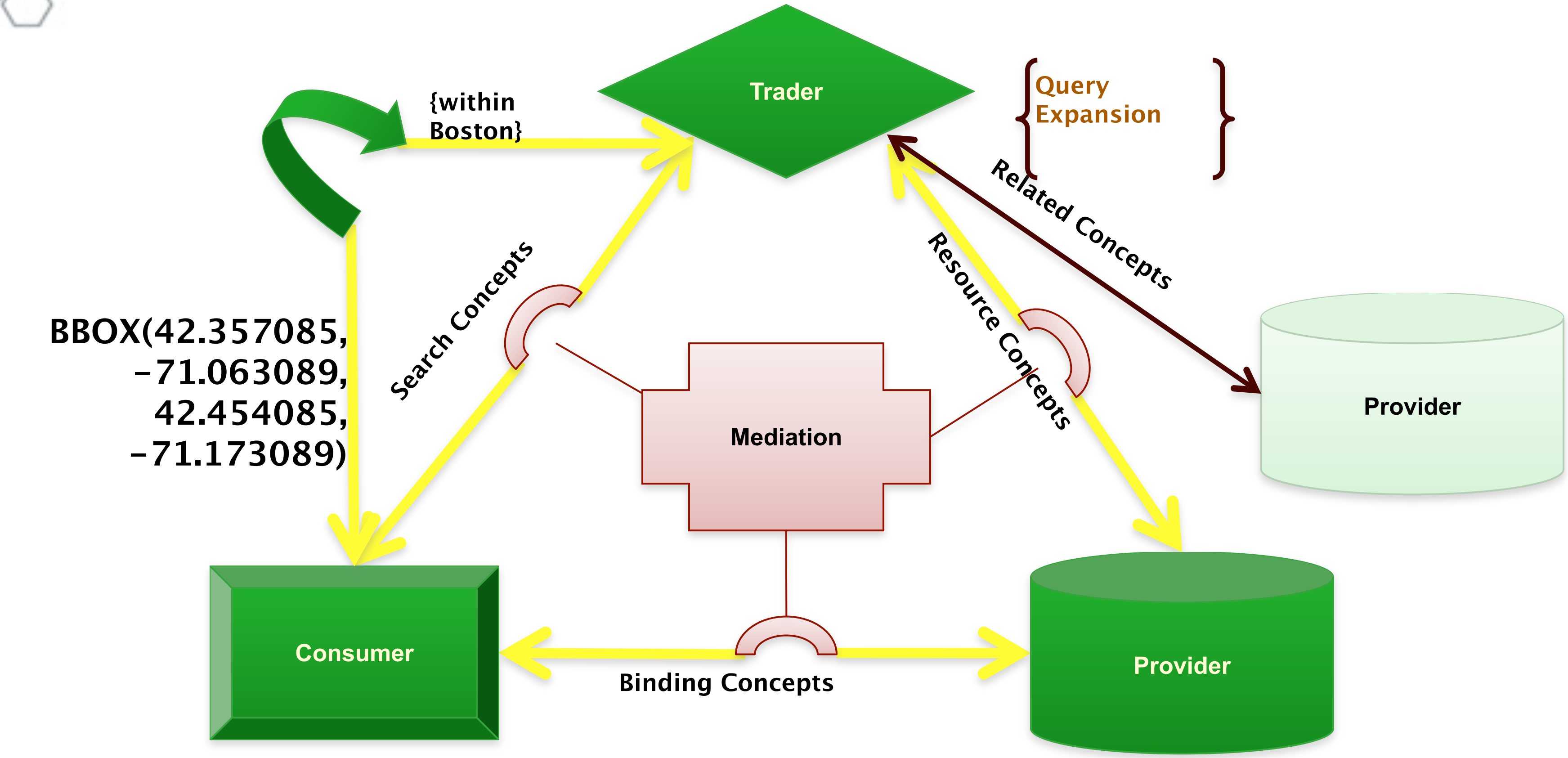
(SEM) SOA Triangle



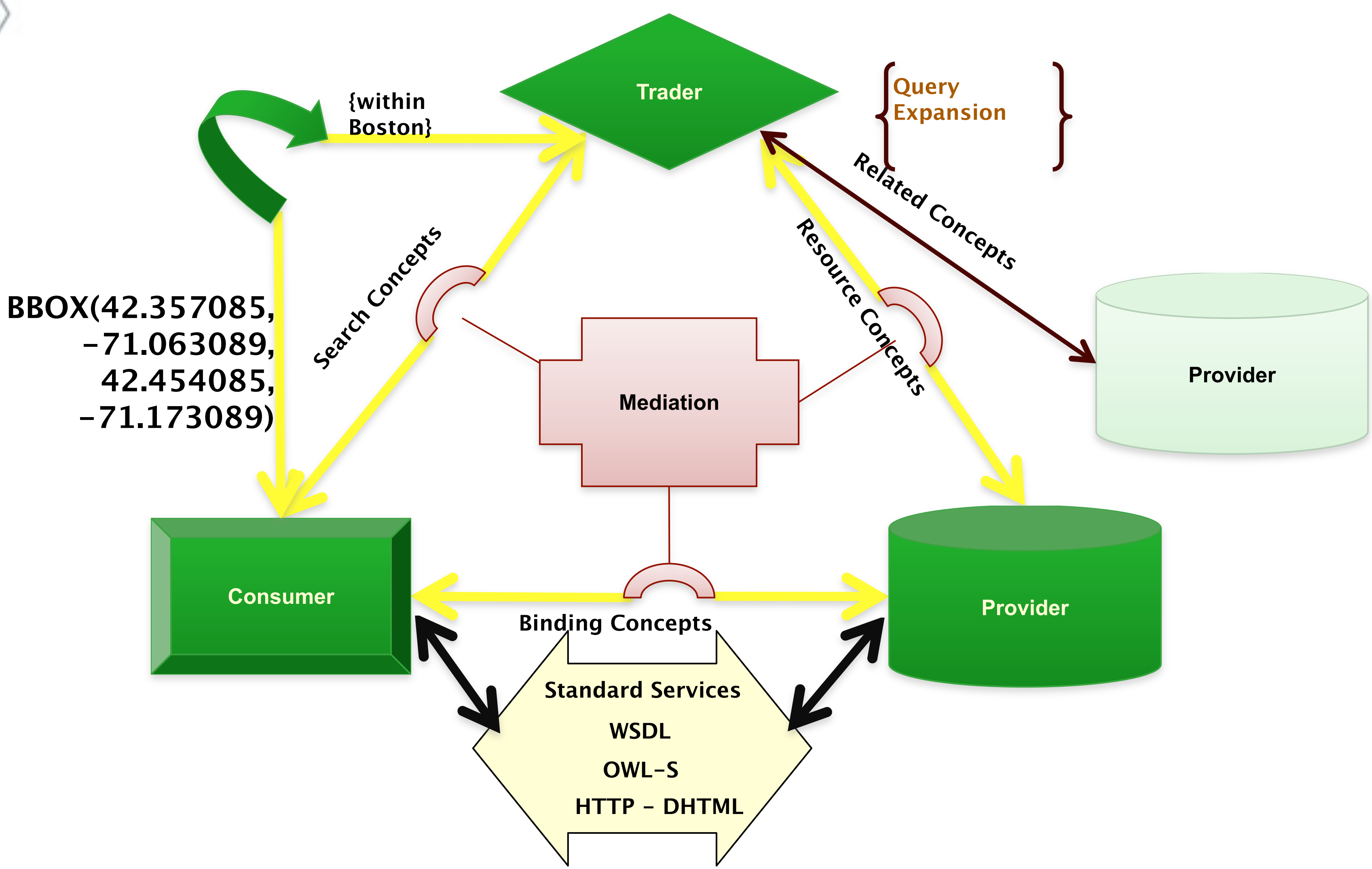
(SEM) SOA Triangle



(SEM) SOA Triangle



(SEM) SOA Triangle



OGC and Services

- Interest in providing geospatial standards to a wide range of constituents
- Web services activity began ~10 years ago with simple URL’s to invoke geospatial functionality, e.g. map images and provide XML-encoded service self-description
- Constituent interest shifted more recently to XML Web Services paradigm using SOAP messaging, WSDL descriptions, UDDI registration over an HTTP transport. SOAP bindings have been developed for most OGC services.
- Most recently, some constituents have shifted to adopting REST principles for Web services which conform more closely to computing practices within the WWW.
- OGC’s role continues to be providing standards to a wide range of constituents. Goal of supporting different service interaction paradigms with consistent functionality.

OGC and Services

- The scope of the Services subgroup includes both guidance for an updated model for OGC services and a best practice for specifying OGC services which simplifies and enhances interoperability across a range of implementation types.
- It is the mission of the Services Group to address both the general, abstract and the specific, concrete aspects of a clearer and more flexible update to the OGC services model and specification practice.

REST Principles and Practice

- (Fielding) Style: “An architectural style is a coordinated set of architectural constraints that restricts the roles/features of architectural elements and the allowed relationships among those elements within any architecture that conforms to that style.”
- “The Representational State Transfer (REST) style is an abstraction of the architectural elements within a distributed hypermedia system.”
- “REST is defined by four interface constraints: identification of resources; manipulation of resources through representations; self-descriptive messages; and, hypermedia as the engine of application state.”
- “The central feature that distinguishes the REST architectural style from other network-based styles is its emphasis on a uniform interface between components”
- “Unlike the distributed object style [31], where all data is encapsulated within and hidden by the processing components, the nature and state of an architecture's data elements is a key aspect of REST.”
- “REST components communicate by transferring a representation of a resource in a format matching one of an evolving set of standard data types, selected dynamically based on the capabilities or desires of the recipient and the nature of the resource.”
- “…REST relies instead on the author choosing a resource identifier that best fits the nature of the concept being identified.”

REST Pillars

1. Resources (anything of specific interest to be referenced on a server) are identified by URI’s.

2. Interaction limited to uniform interface (e.g. HTTP) for CRUD exchange of resource representations.

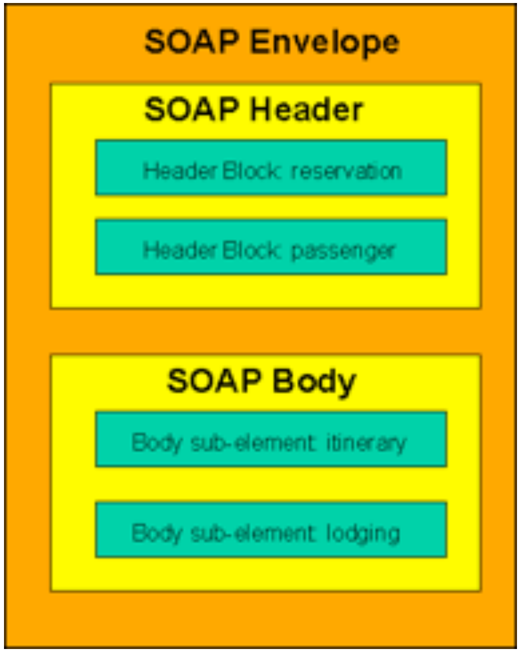
3. Servers control their own namespaces (e.g. URL patterns).

4. HATEOAS - hypermedia (e.g. HTML) as the sole means for an application to form and invoke URI’s.

SOAP and REST

- SOAP (Simple Object Access Protocol) is an XML messaging format.
- Generally used with WSDL as a remote processing invocation, using HTTP POST as a transport mechanism (RPC).

- REST (Representational State Transfer) is a set of architectural principles which attempts to describe the most scalable and loosely coupled WWW structures.
- Often adopted only in part (e.g. structured URL's) and does not explicitly deal with Web Services (Hypermedia)



SOAP Envelope

SOAP Header

Header Block: reservation

Header Block: passenger

SOAP Body

Body sub-element: itinerary

Body sub-element: lodging

Eye of the Invoker

• Procedure Call	• Message	• Resource Locator
• Communication protocol	• POST	• Transaction operation
• Invoke engine	• GET	• Fetch information resource
• Resource Typing & Syntax	• RESTful	• Hypermedia & Linking
• Invocation	• Process	• Job resource

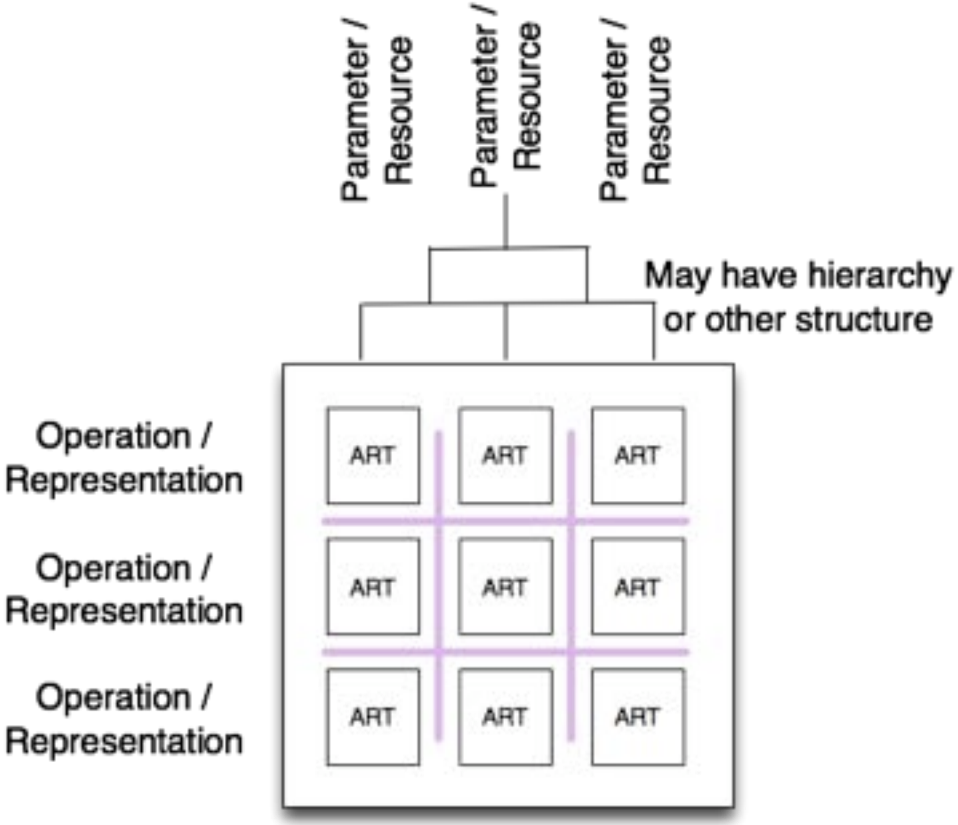
Helping the World to Communicate Geographically

Conversation and Reflection

- Basic unit of service interaction is the message (e.g. request-response conversation).
- Each message is a representation of a distributed computing resource. Even a URL is a compact representation.
- Basic service functionality consists of transforming one representation to another, consistent with the resource it represents.
- Service functionality is built up of collections of messages.
- Metadata representations provide everything a client needs to know about the remote computing resource (reflection).
- Gray area between a representation subset (e.g. CGI params) of a described resource and a query or process resource with its own description (e.g. a “job”).

Crosswalking RPC and REST

- □ Close parallel between RPC “operation” and REST “representation.
- □ Close parallel between RPC “parameter” and REST “
- □ Organization of Resources / behaviors and messages
- □ Question of Media Typing
- □ Unresolved exclusivity of “HATEOAS” for altering application state
- • Possibility of “REST Overlay” on typed procedure
- □ What is the difference between a website and a service?



Parameter / Resource

Parameter / Resource

Parameter / Resource

May have hierarchy or other structure

Operation / Representation

Operation / Representation

Operation / Representation

ART ART ART

ART ART ART

ART ART ART

ART - Atomic Representation Transformer

+ CRUD - Create / Update / Delete

Paths Forward

- Service functionality independent of interface syntax and model. E.g. Message or signal definition
- Structured (local or global) URL as representation
- XML request as a resource
- Extended DCP metadata as URL template
- REST hypermedia “overlay”
- Do all services look like Atom Pub?
- Is there anything special about Spatial here?

Discussion

- Methodology for defining service functionality independently of “binding”. Possibility (John Herring) to use CRUD operations for this purpose, but they may be too low-level. Also a role for formal semantic definition of higher service functionality.
- Can service guidance can be incorporated solely into a core OWS service specification, or will accompanying policy documents will be needed?
- Updates to the TC211 service model might be split between an ISO 19119 model and a new part of ISO 19115 for service metadata (Nicolas Lesage).

Next OGC Steps

- Guidelines Discussion Papers
- WMTS and other specification updates
- ISO 19119 and 19115 updates
- Core OWS Service follow on from OWS Common

GEOSS and Service Infrastrucure

- GEOSS - Global Earth Observing System of Systems
- Support of Societal Benefit Areas
- Common Infrastructure for registration, discovery, sharing of earth observation data and computing resources
- Architecture Implementation Pilot (Phases 1 and 2) for building GCI, operating capabilities (contributed data, services, applications, exemplars of GEOSS usage.
- AIP also serves to flesh out the “system of systems” concept.
- Range of participants in GEOSS, from commercial software providers, to off-the-street browsers.

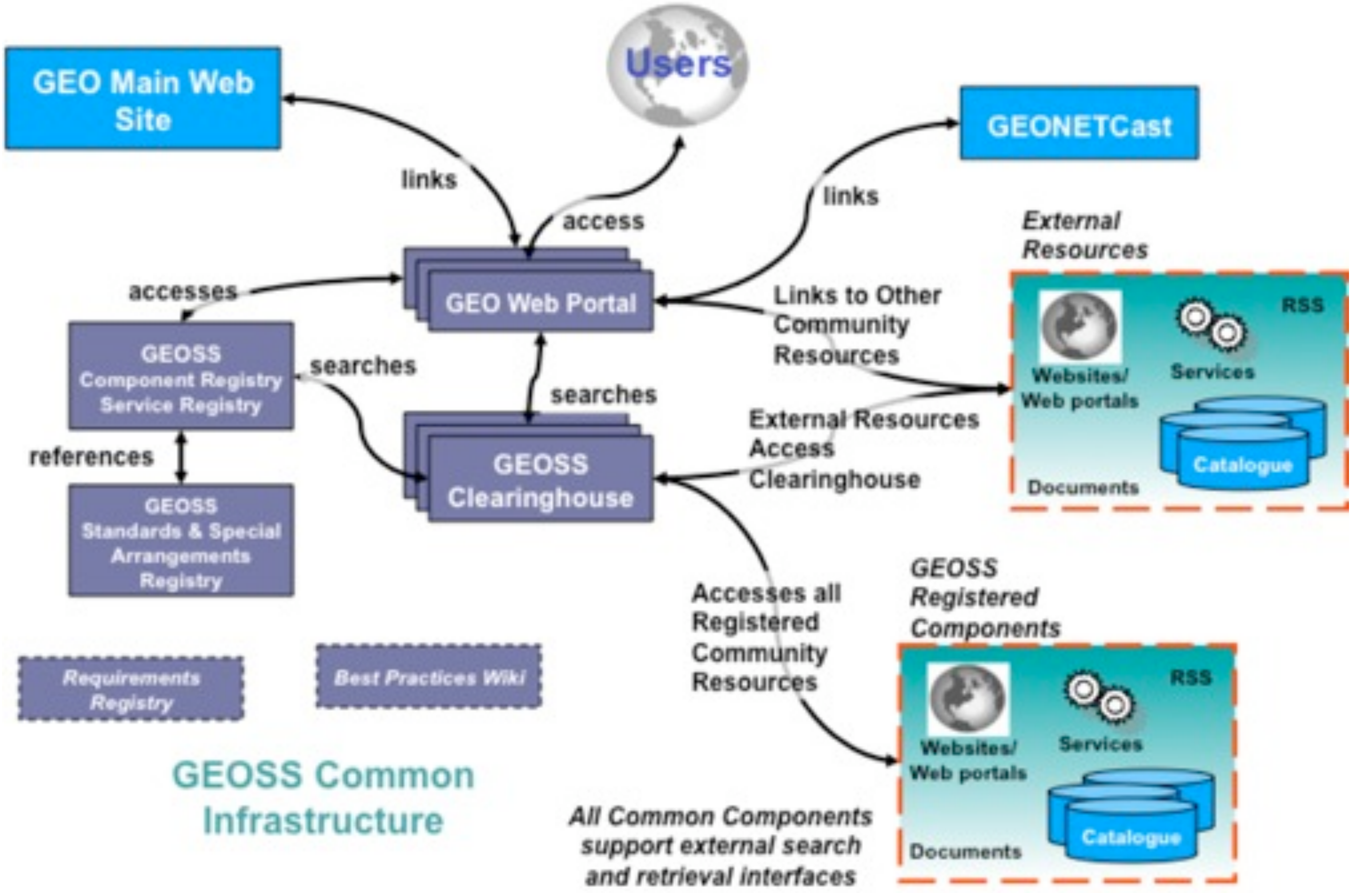
Data Sharing

The GEOSS 10 Year Implementation Plan defines the GEOSS Data Sharing Principles:

> "There will be full and open exchange of data, metadata, and products shared within GEOSS, while recognizing relevant international instruments and national policies and legislation. All shared data, metadata, and products will be made available with minimum time delay and at minimum cost. All shared data, metadata, and products for use in education and research will be encouraged to be made available free of charge or at no more than the cost of reproduction."

GEOSS Strategic Guidance Document, GEO Task Team AR-06-02, 14 December 2006

GEOSS Core Component Interactions


GEO Main Web Site
Users
GEONETCast
links
access
links
External Resources
accesses
GEO Web Portal
Links to Other Community Resources
Websites/ Web portals
Services
RSS
Catalogue
Documents
GEOSS Component Registry Service Registry
searches
searches
External Resources Access Clearinghouse
references
GEOSS Clearinghouse
GEOSS Standards & Special Arrangements Registry
Accesses all Registered Community Resources
GEOSS Registered Components
Requirements Registry
Best Practices Wiki
Websites/ Web portals
Services
RSS
Catalogue
Documents
GEOSS Common Infrastructure
All Common Components support external search and retrieval interfaces

GCI and External Resources

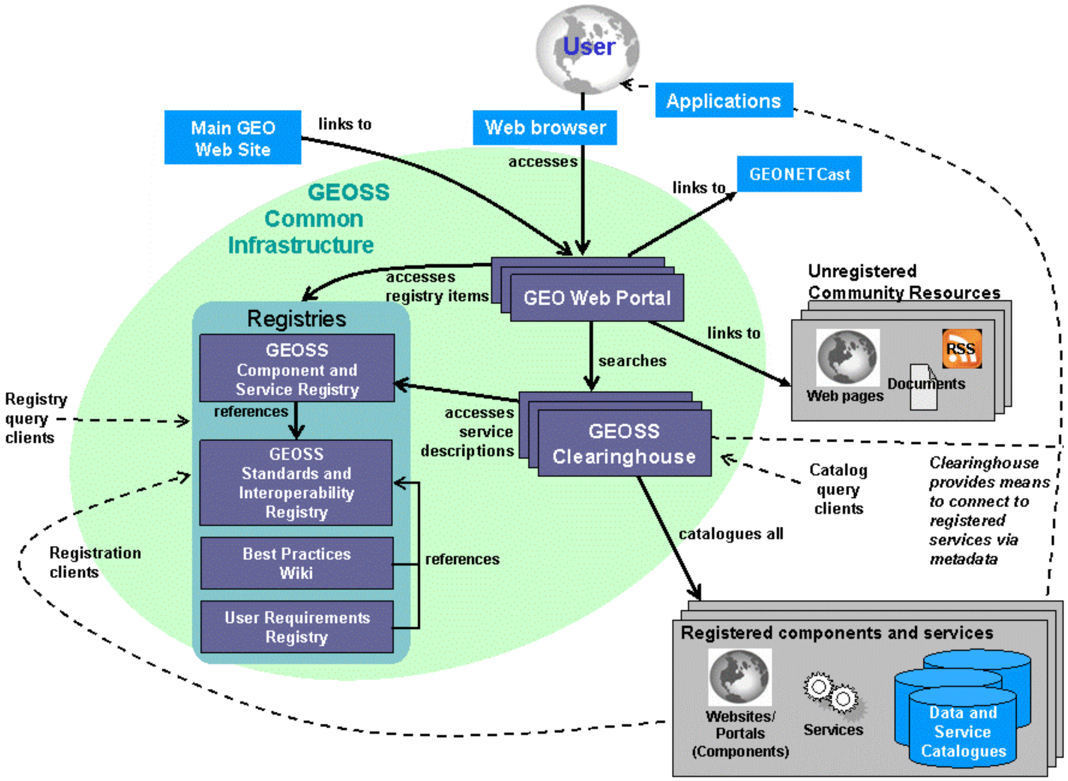
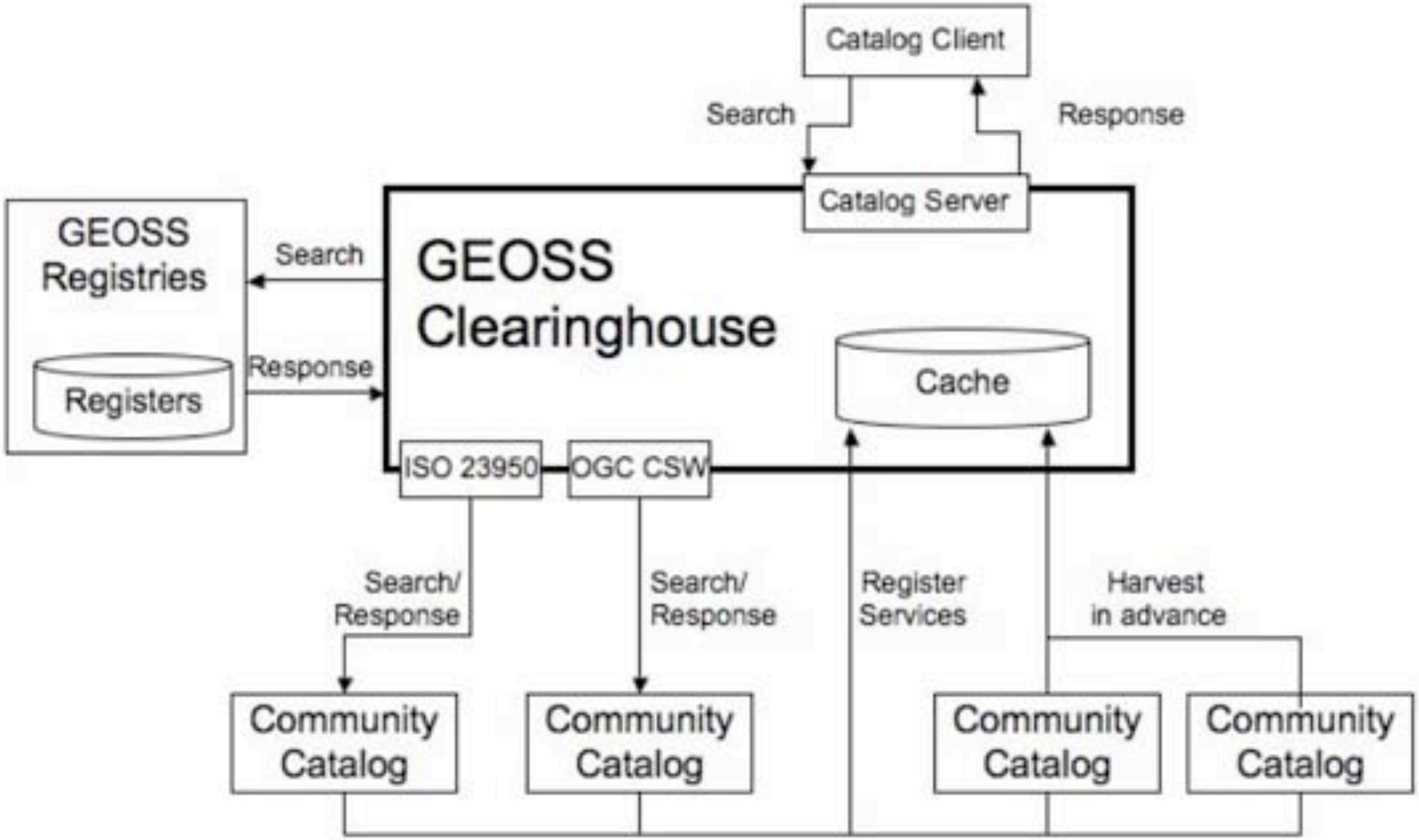

User
Applications
Main GEO Web Site
links to
Web browser
accesses
GEONETCast
links to
GEOSS Common Infrastructure
accesses registry items
GEO Web Portal
Unregistered Community Resources
Web pages
RSS
Documents
links to
searches
Registries
GEOSS Component and Service Registry
references
GEOSS Standards and Interoperability Registry
Best Practices Wiki
User Requirements Registry
Registry query clients
accesses service descriptions
GEOSS Clearinghouse
Catalog query clients
Clearinghouse provides means to connect to registered services via metadata
Registration clients
references
catalogues all
Registered components and services
Websites/ Portals (Components)
Services
Data and Service Catalogues

Figure 2 – An object-interaction diagram depicting the major relationships between the GCI component services and selected external resources. (Modified from GEO Secretariat "The GEOSS Common Infrastructure Establishment Process Document" March 2008)

GEOSS Clearinghouse “Catalog of Catalogs”

Web Portal, Client, DS Client, Desktop application



End-to-end support of GEOSS Users

- Differentiation of infrastructure components is useful for technical users, but not for casual or non-technical users who see only the portal
- Questions to be asked at the portal by users flow through to the interfaces and metadata to be provided by contributors
- Should portal discovery continue seamlessly into drill-down, explore, and exploit?
- How can cross-community data sharing really be promoted and supported - “how do you know what you don’t know?”