Architectural Patterns in Open GIS Web Services

Rob Atkinson, Social Change Online, Australia (rob@socialchange.net.au)

Arne J. Berre, SINTEF, Norway (Arne.J.Berre@informatics.sintef.no)

www. isotc211.org

Open GIS Consortium
Spatial connectivity
for a changing world.

www. opengis.org

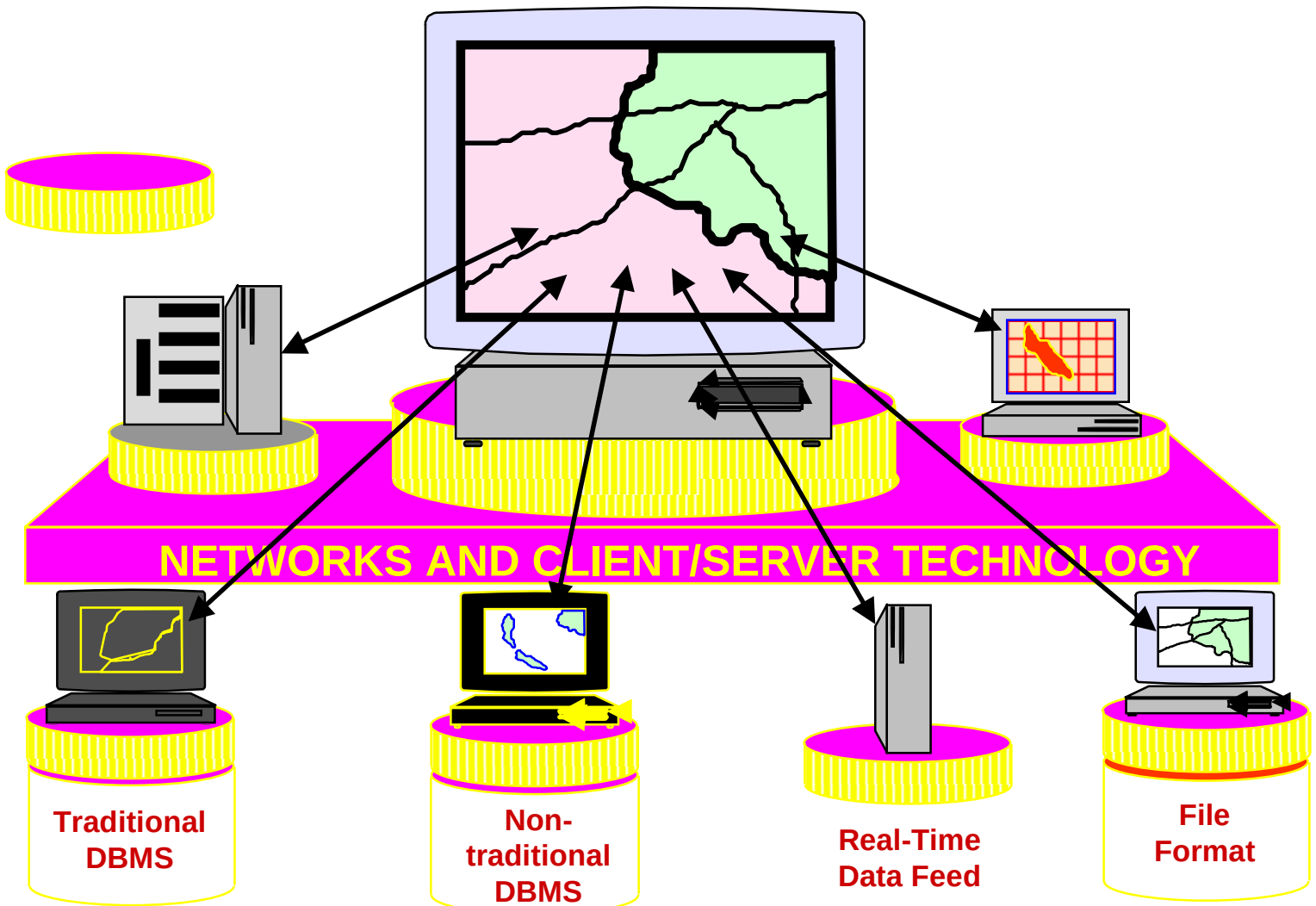
- **Problem: Interoperability of GIS – through standards**
- **Open GIS: OGC and ISO/TC211 standards**
- **Architectural patterns: ISO/OGC Service Architecture**
- **OGC Web Services architecture**
- **Pattern – Service binding template**
- **OGC, ISO/TC211 and OMG MDA**
- **Conclusion**

Architectural Patterns in Open GIS Web Services

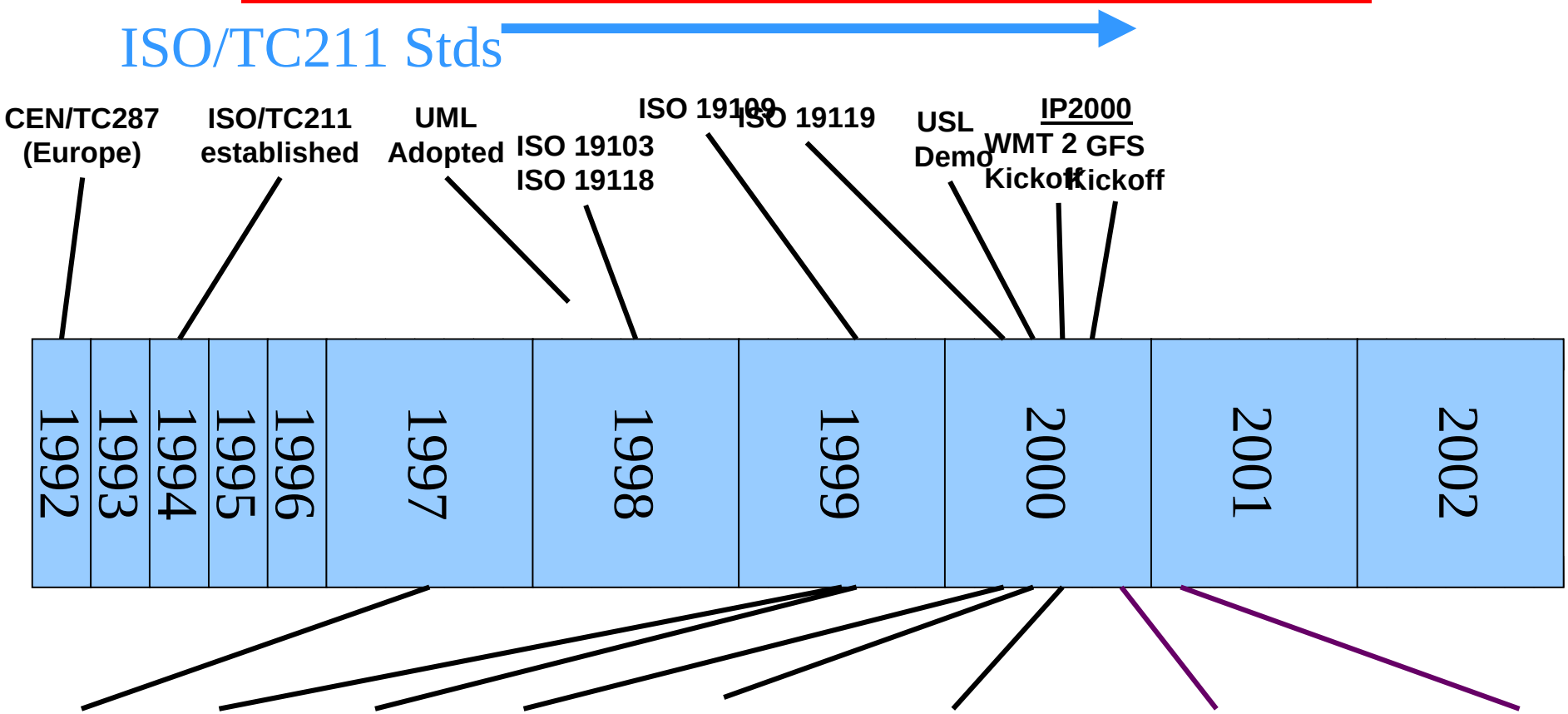
OGC (Open Geodata Consortium) and ISO/TC211 is involved in the creation of standards and recommendations for geospatial information and services.

OGC Web Services provide a vendor-neutral, interoperable framework for web-based discovery, access, integration, analysis, exploitation and visualization of multiple online geodata sources, sensor-derived information, and geoprocessing capabilities. The current UDDI and ebXML registry models provide a basis for business and service descriptions in the OWS registry model. The strong information- and content-oriented focus of GIS web services has led to a further development of architectural patterns for flexible information search and retrieval for general content-oriented web services, based on a dynamic binding template. The relationship to the OMG MDA approach and the ISO 19119 service specification standard of ISO/TC211 will be discussed.

Problem Goal: Transparent Access to Heterogeneous Geodata and Geoprocessing Services



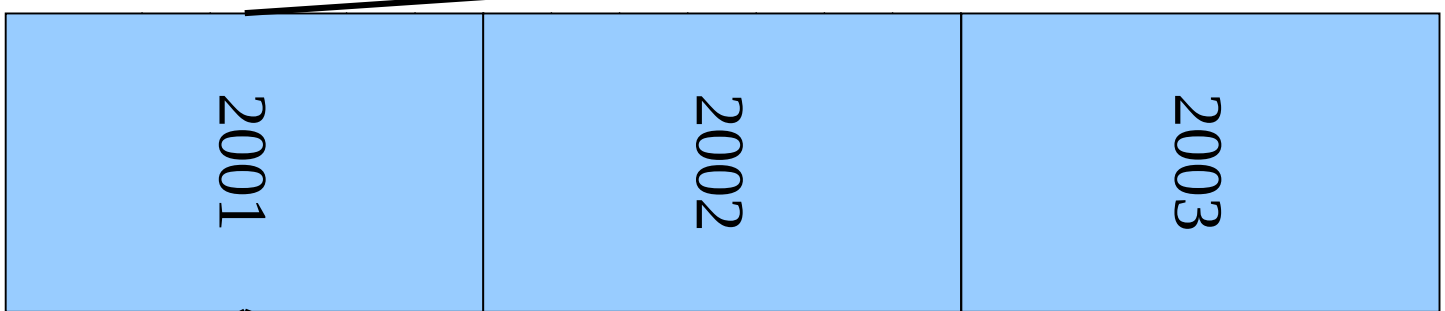
ISO/TC211 & OGC "Roadmap"



2001/2002 Initiatives

IP2001
Decision Support
4D
Info Community Enable
Location Services
Military Pilot Project
WMT-3 — GFS-2

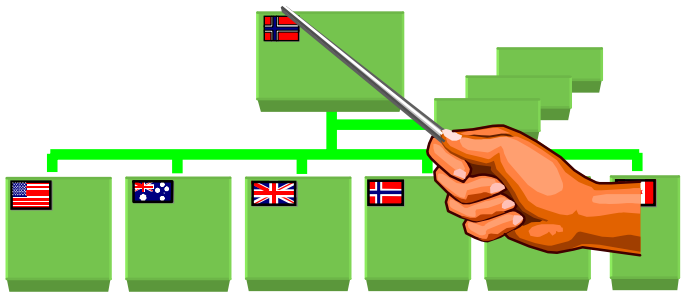
2001 Business
IETF Response
The Open Group
MAGIC Agenda
E-Government
Digital Earth
LIF Agenda
W3C Coordination
SEDRIS Agenda
OpenGIS.net
European Programs



GFS
Gazetteer
WFS, WCS
GeoLink, LOF
Geoparse, Geocode
GML XML Schema
Thesaurus
Type Dictionary

WMS-2
Legend,
Style Sheet &
S.S.Catalog
SLD, Symbol Library
GML Extensions:
IML, Coverage,
Basic Service Model
Filter

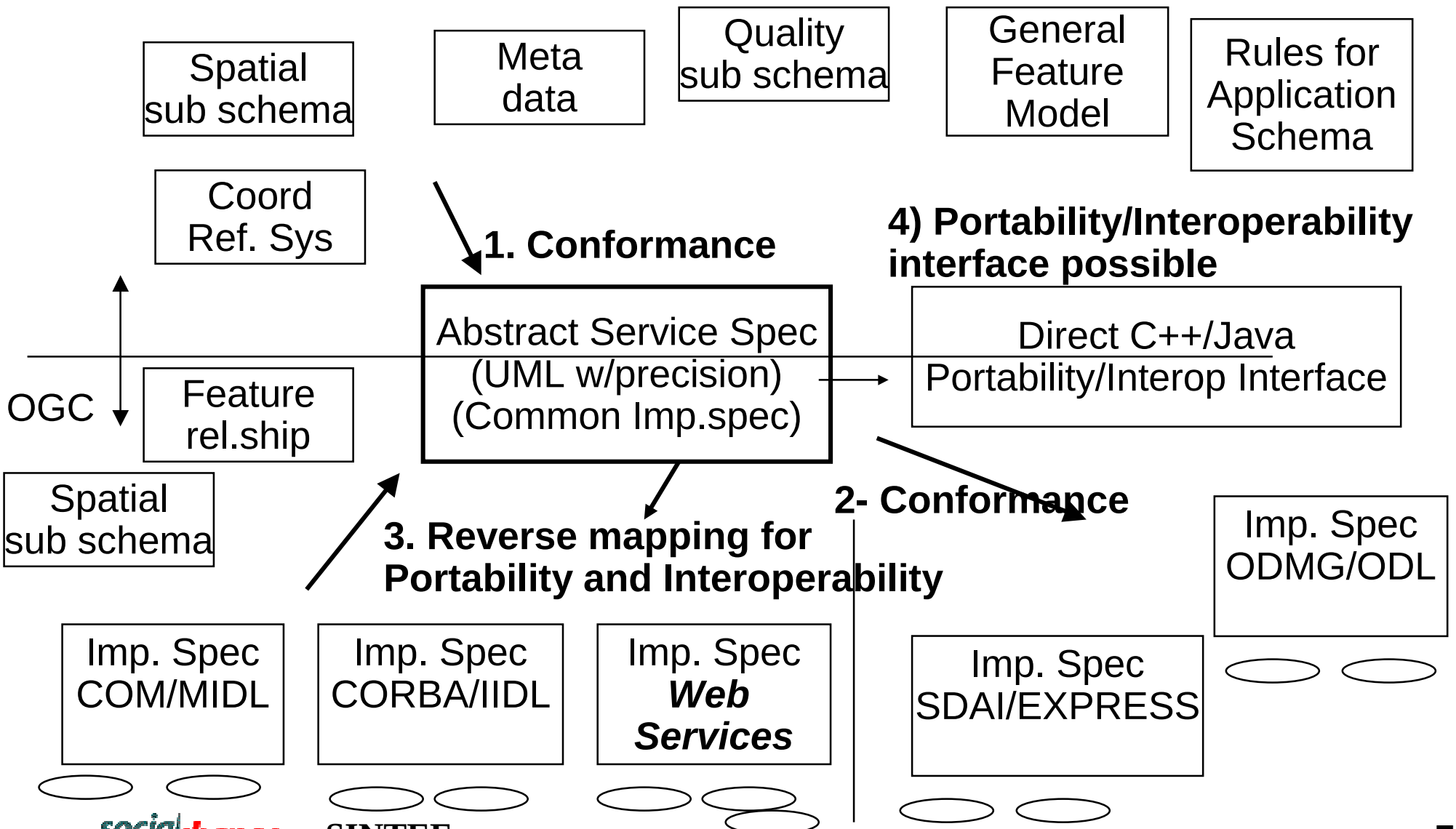
ISO/TC211Overview

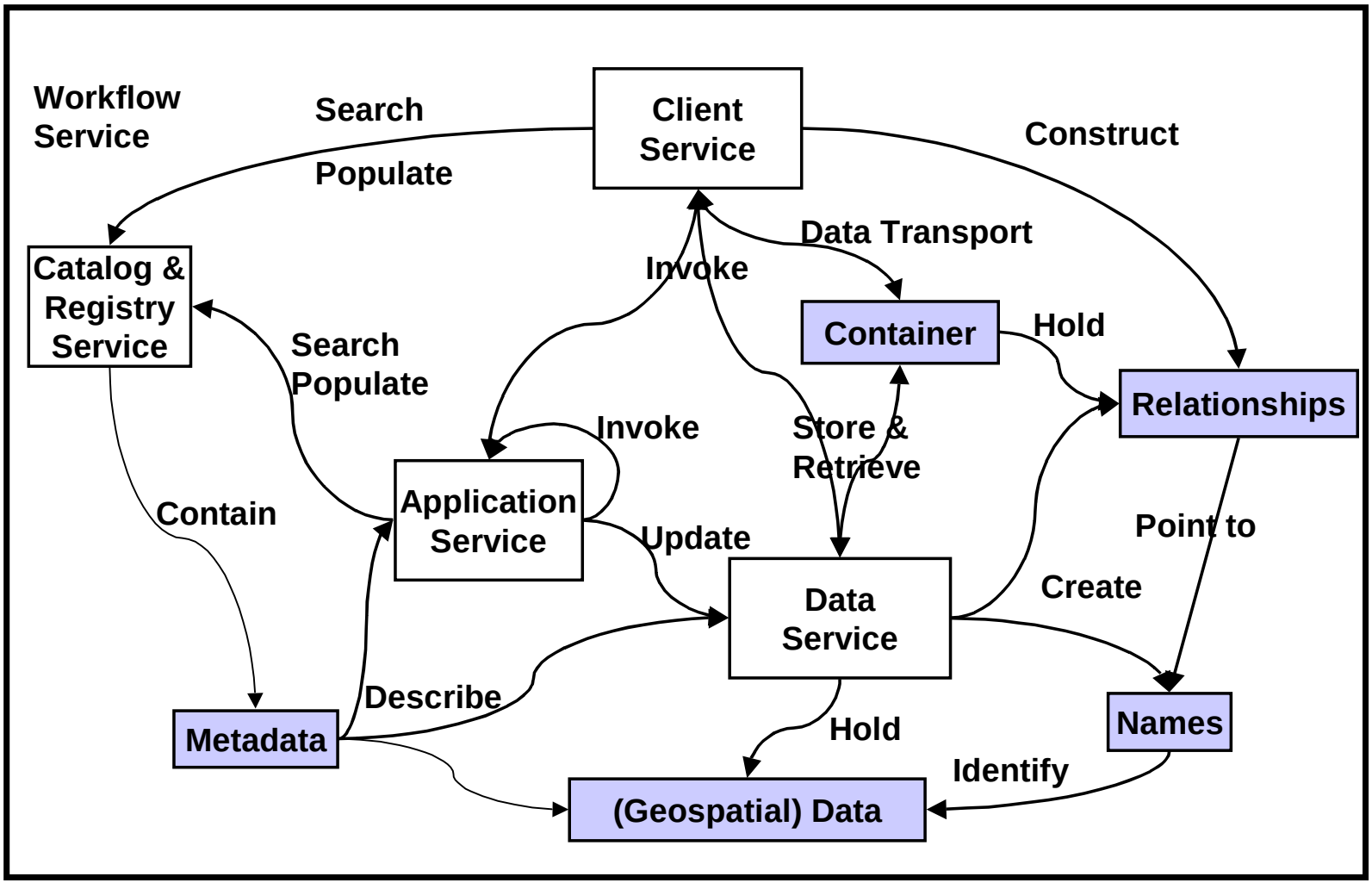


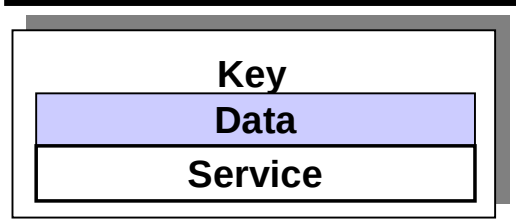
- ISO 19101 - Reference model
- ISO 19102 - Overview
- ISO 19103 - Conceptual schema language
- ISO 19104 - Terminology
- ISO 19105 - Conformance and testing
- ISO 19106 - Profiles
- ISO 19107 - Spatial schema
- ISO 19108 - Temporal schema
- ISO 19109 - Rules for application schema
- ISO 19110 - Feature cataloguing methodology
- ISO 19111 - Spatial referencing by coordinates
- ISO 19112 - Spatial referencing by geographic identifiers
- ISO 19113 - Quality principles
- ISO 19114 - Quality evaluation procedures
- ISO 19115 - Metadata
- ISO 19116 - Positioning services
- ISO 19117 - Portrayal
- ISO 19118 - Encoding
- ISO 19119 - Services
- ISO/TR 19120 - Functional standards + new rev
- ISO/TR 19121 Imagery and gridded data
- ISO/TR 19122 - Qualifications and certification of personnel
- ISO 19123 - Schema for coverage geometry and functions
- ISO 19124 - Imagery and gridded data components
- ISO 19125 - Simple feature access – Part 1-3
- ISO 19126 - Profile - FACC Data Dictionary
- ISO 19127 - Geodetic codes and parameters
- ISO 19128 - Web Map Server Interface
- ISO 19129 - Imagery, gridded and coverage data framework
- ISO 19130 - Sensor and data model for imagery and gridded data
- ISO 19131 - Data product specification
- ISO 19132 - Location based services possible standards
- ISO 19133 - Location based services tracking and navigation
- ISO 19134 - Multimodal location based services for routing and navigation
- ISO 19135 - Procedures for registration of geographic information items

ISO

ISO/TC211 and OpenGIS

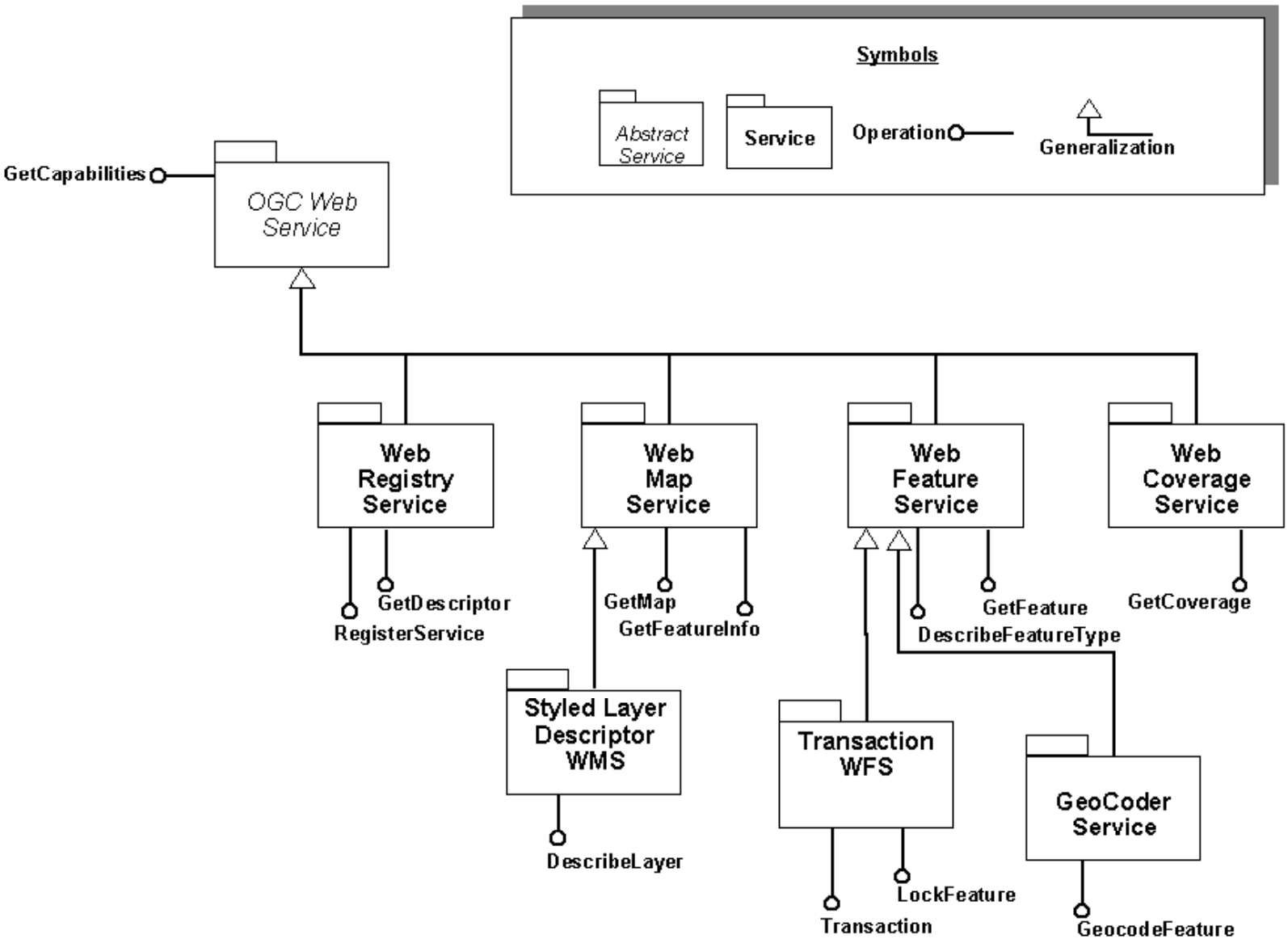


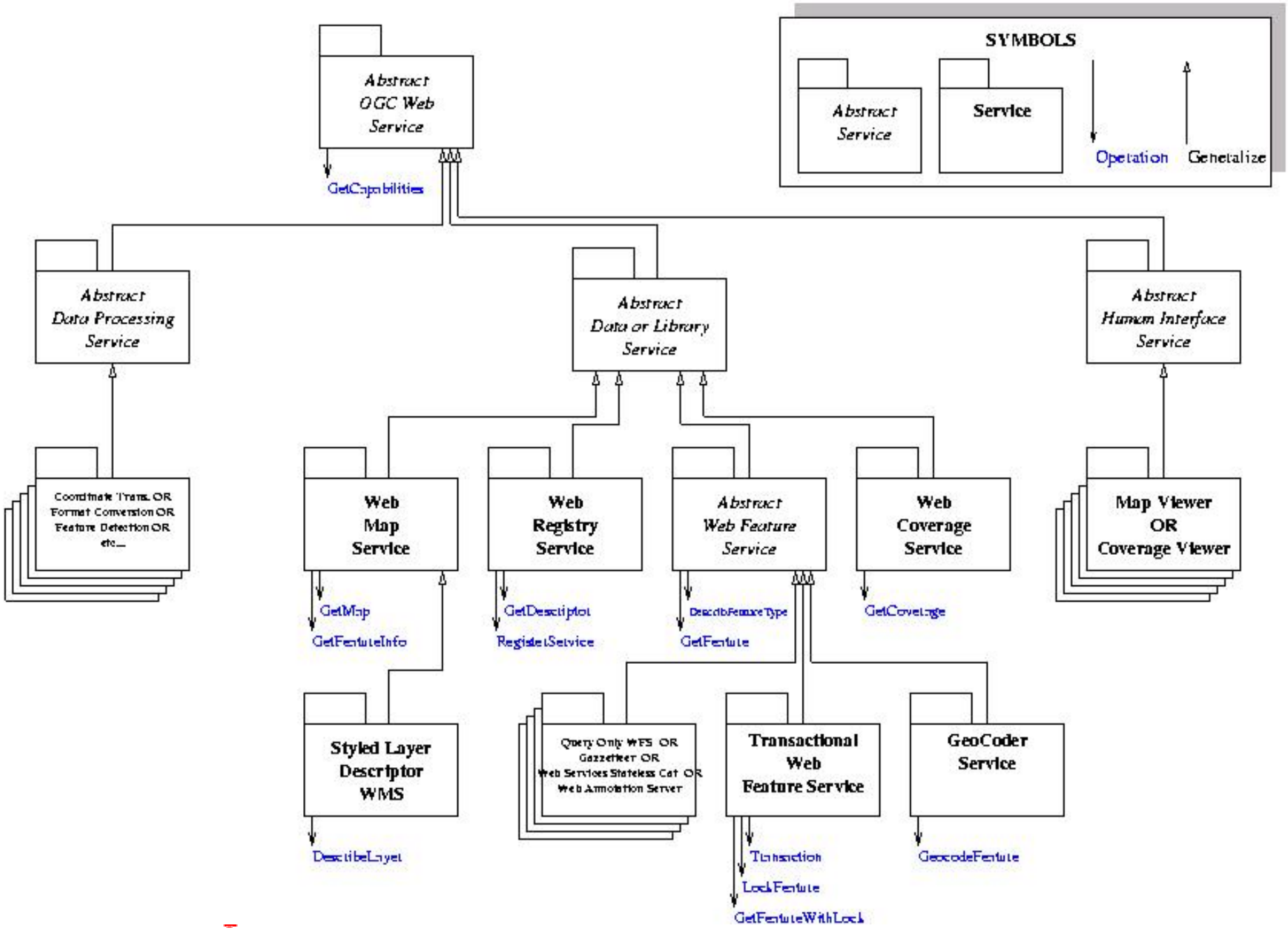


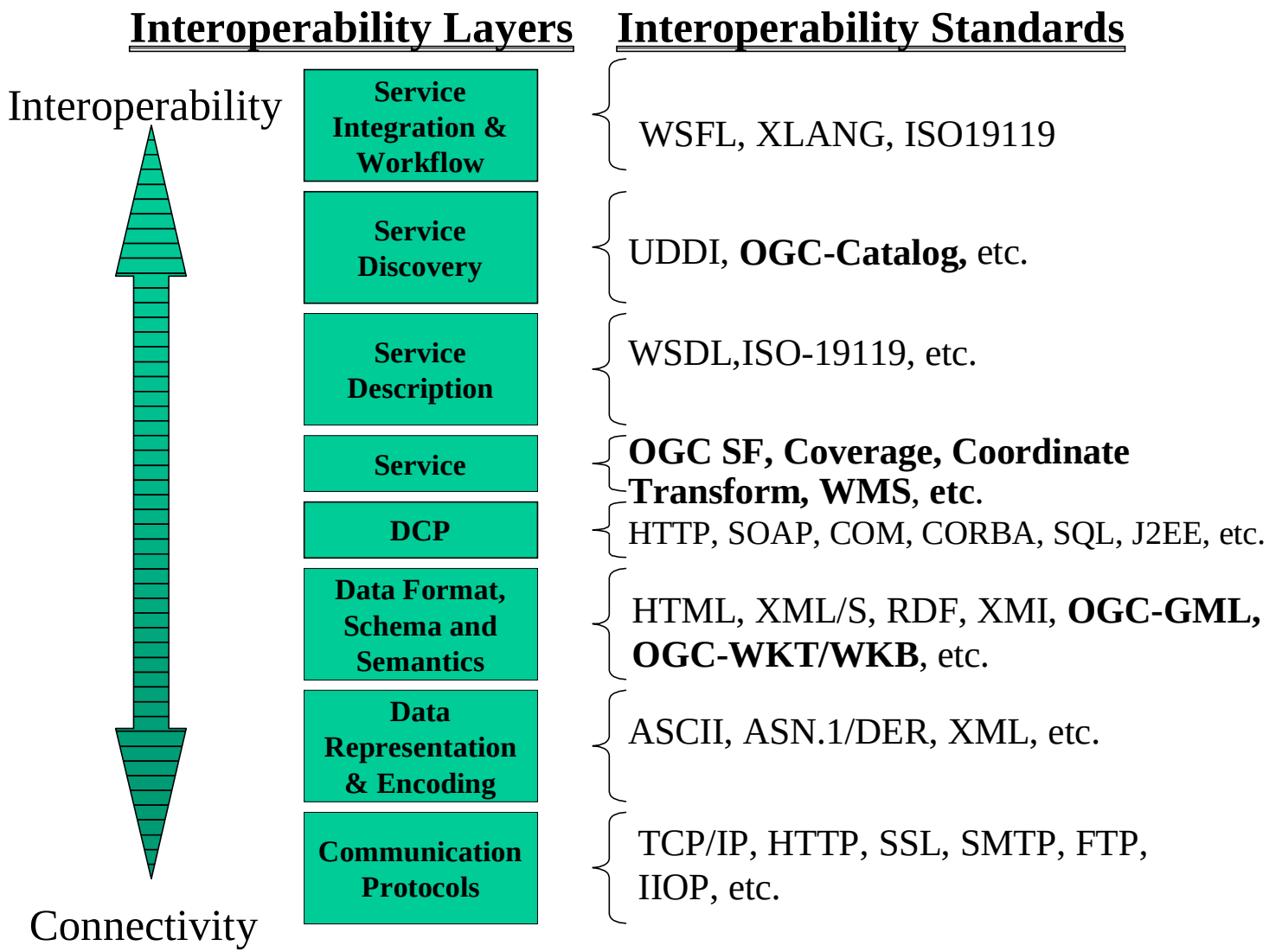


Pattern for Service Architecture

OpenGIS Web Services Architecture




SYMBOLS
Abstract Service
Service
Operation
Generalize
Abstract OGC Web Service
GetCapabilities
Abstract Data Processing Service
Abstract Data or Library Service
Abstract Human Interface Service
Coordinate Trans. OR Format Conversion OR Feature Detection OR etc...
Web Map Service
Web Registry Service
Abstract Web Feature Service
Web Coverage Service
Map Viewer OR Coverage Viewer
GetMap
GetFeatureInfo
GetDescriptor
RegisterService
DescribeFeatureType
GetFeature
GetCoverage
Styled Layer Descriptor WMS
DescribeLayer
Query Only WFS OR Gazzetteer OR Web Services Stateless Cat OR Web Annotation Server
Transactional Web Feature Service
GeoCoder Service
Transaction
LockFeature
GetFeatureWithLock
GeocodeFeature


Interoperability Layers
Interoperability Standards
Interoperability
Connectivity
Service Integration & Workflow
WSFL, XLANG, ISO19119
Service Discovery
UDDI, OGC-Catalog, etc.
Service Description
WSDL,ISO-19119, etc.
Service
OGC SF, Coverage, Coordinate Transform, WMS, etc.
DCP
HTTP, SOAP, COM, CORBA, SQL, J2EE, etc.
Data Format, Schema and Semantics
HTML, XML/S, RDF, XMI, OGC-GML, OGC-WKT/WKB, etc.
Data Representation & Encoding
ASCII, ASN.1/DER, XML, etc.
Communication Protocols
TCP/IP, HTTP, SSL, SMTP, FTP, IIOP, etc.

Binding Patterns for OpenGIS Web Services
A Special Case?

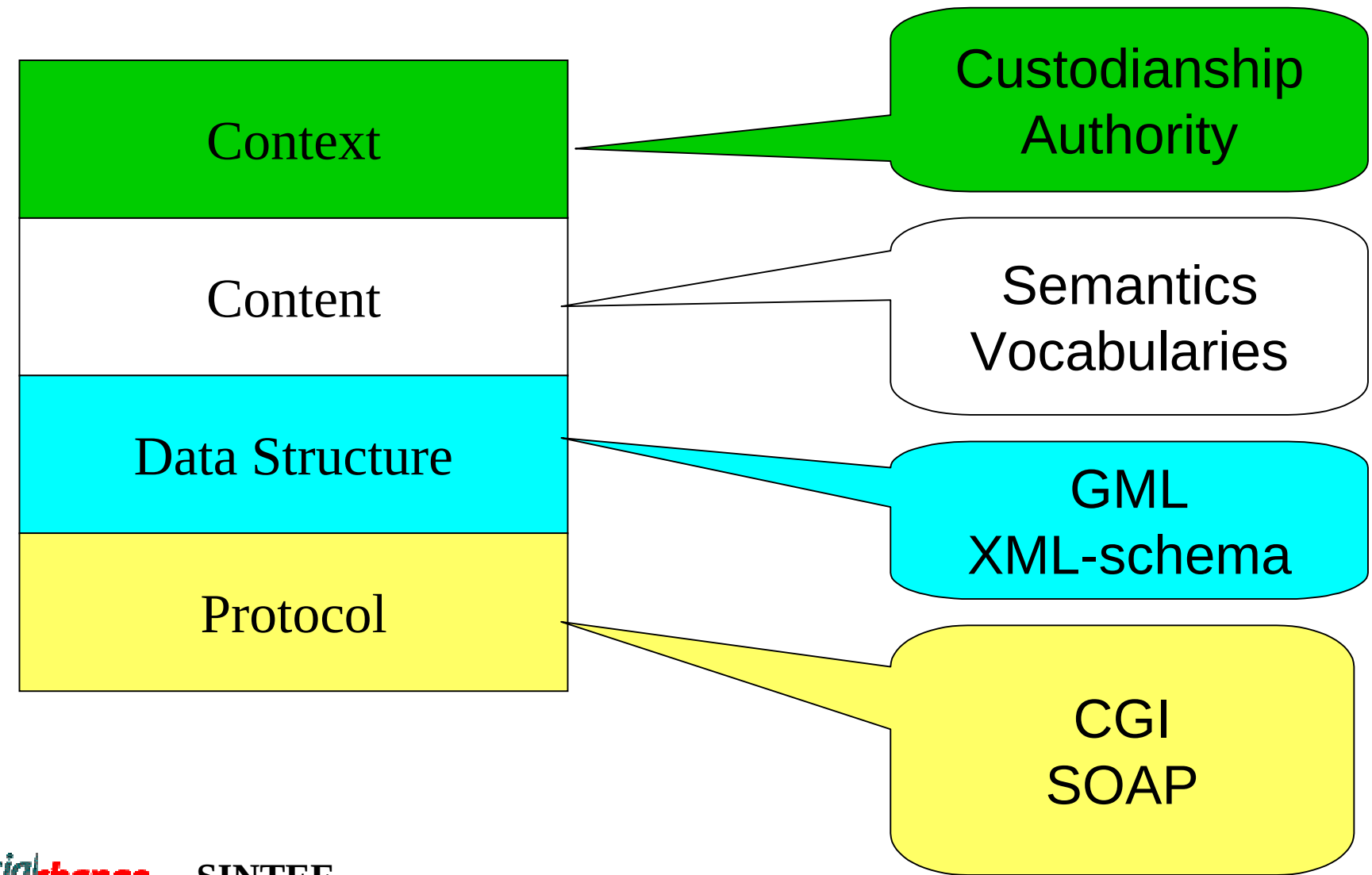
- **Much content has a geographic facet.**
- ***Most* structured data is geographically organized.**
- **Geographic data can be terabytes and time dependent – exploitation is often an matter of choosing the right bit.**
- **A natural fit for demand access services.**
- **Content centric….**

- **BindingTemplate: A generalised content-aware recipe for invoking services**

Content Centricity

- Semantics of service primarily a function of the content – not the access protocol
- Well known interfaces to exploit application-centric content
- Unlike “stock ticker” – application-centric interfaces to well known content
- OpenGIS defines standards for interoperability

Interoperability…



Motivation

- Need a consistent way to describe geographical content for service instances.
- Use Cases need to support persistent early binding – “e.g. show me what the traffic is like today, (using service X I discovered yesterday).”
- Need a consistent way to “chain” services.
- WSDL “binding” section inadequate – “types” only provide structural information.

Synopsis

- Operations invoked by parameterized messages
- Valid parameter values are constrained by service instances
- Service invocation requires all mandatory “unbound” parameters to be bound
- Service chaining requires bindings to be described throughout the chain

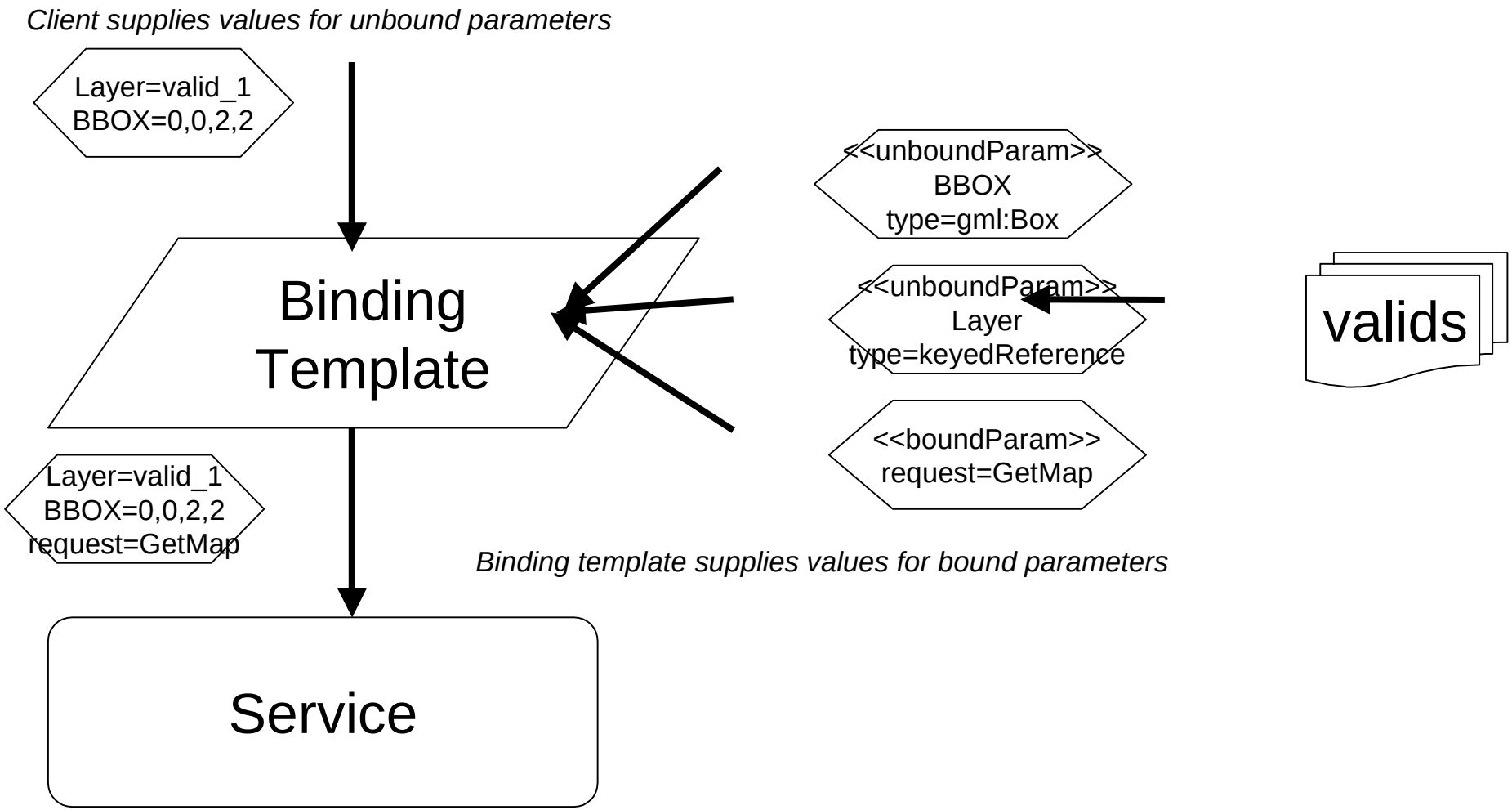
Binding Templates

- A recipe for invoking services
- Generalisation of a common pattern appearing in OpenGIS Use Cases
- Reference objects from registries that define semantics of service
- binding templates may reference each other, creating the service chain to be invoked and allowing mix of early and late binding

Mechanisms

- Syntax for persisting an invocation recipe
- Derived from Registry Entry (ebRIM model)
- Can be passed as a complex parameter to a service further down the chain
- (e.g. tell a Web Map Server how to query a Web Feature Server and how to portray the results)

Invoking a service



Early vs. Late Binding

Most invocations can be either early of late bound to particular messages

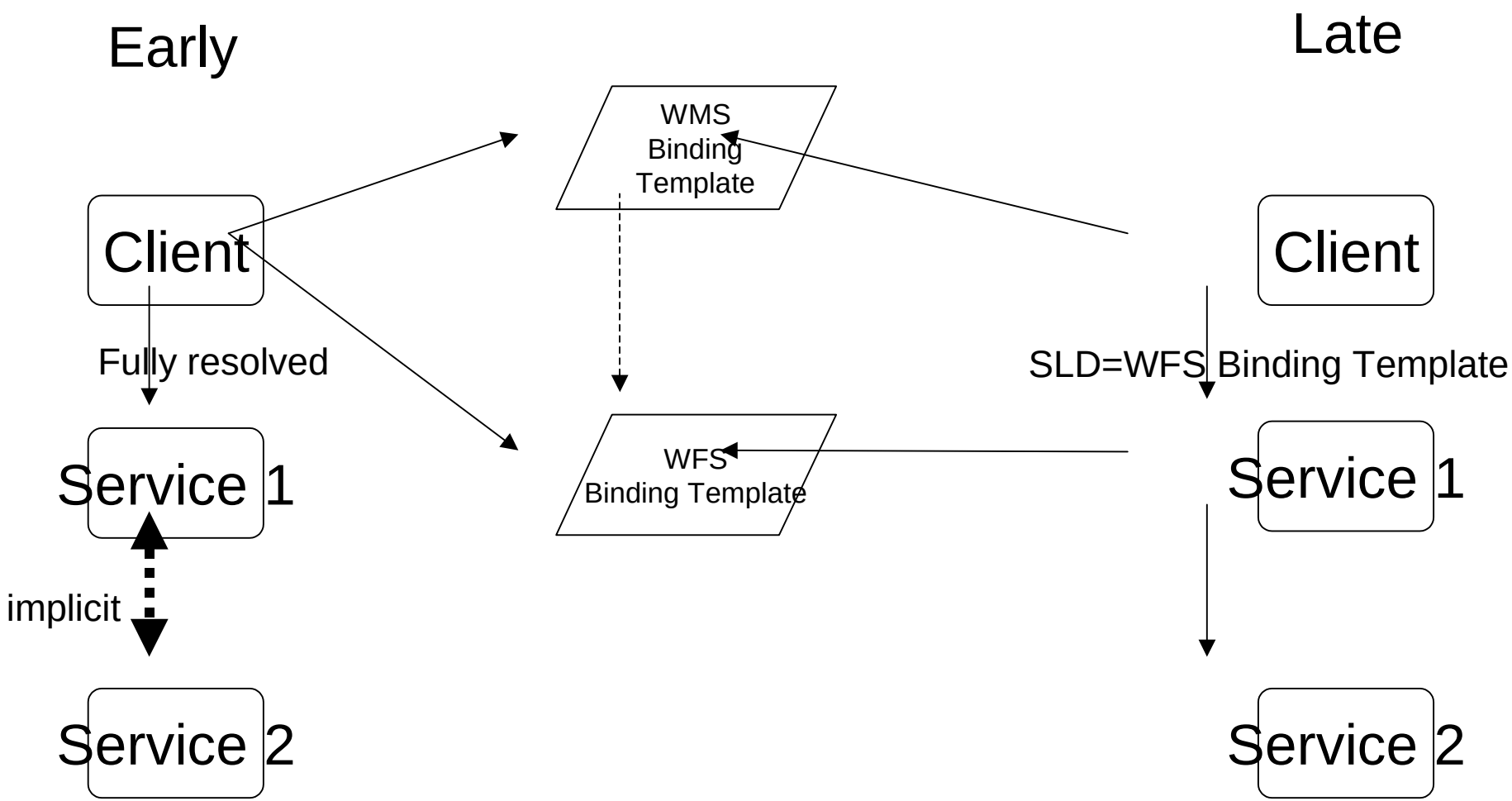
Early binding

- E.g. get the most recent weather report

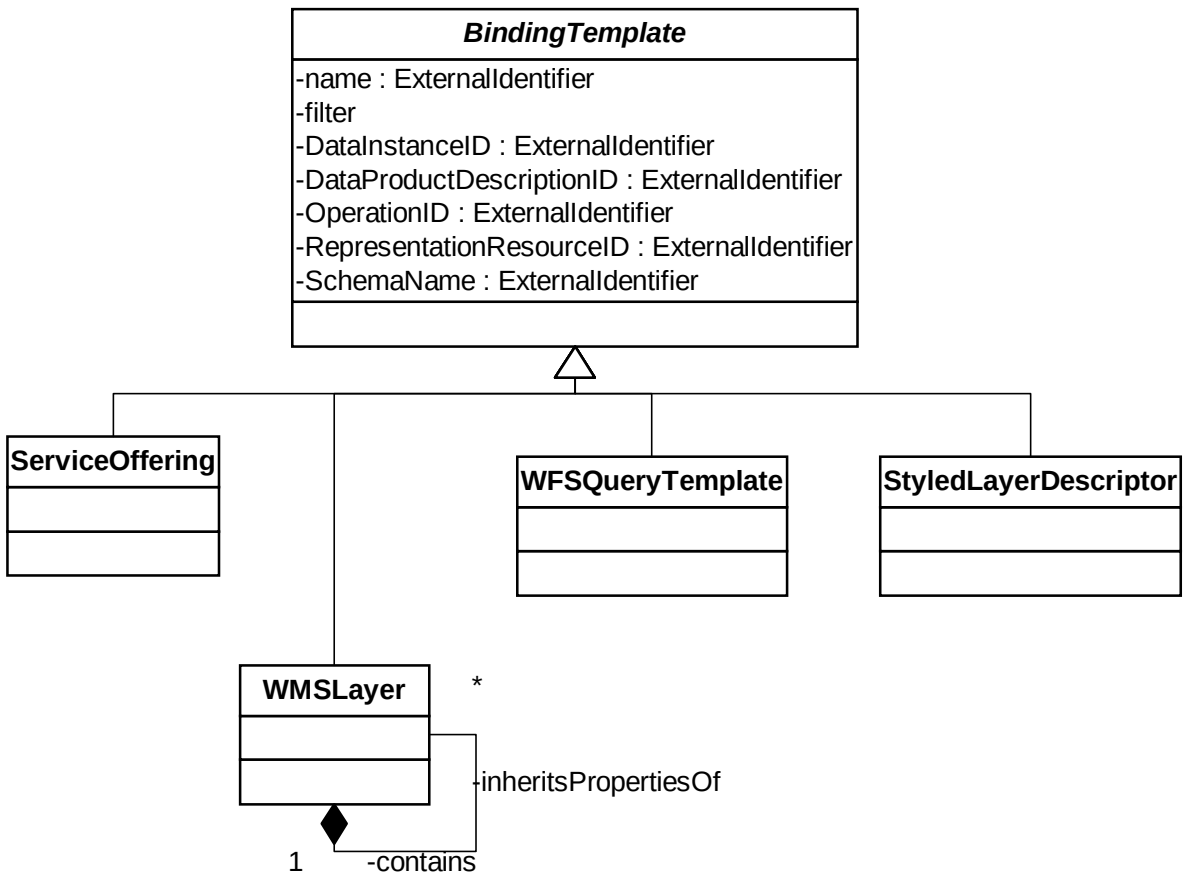
Late binding

- zoom to the area of flooding

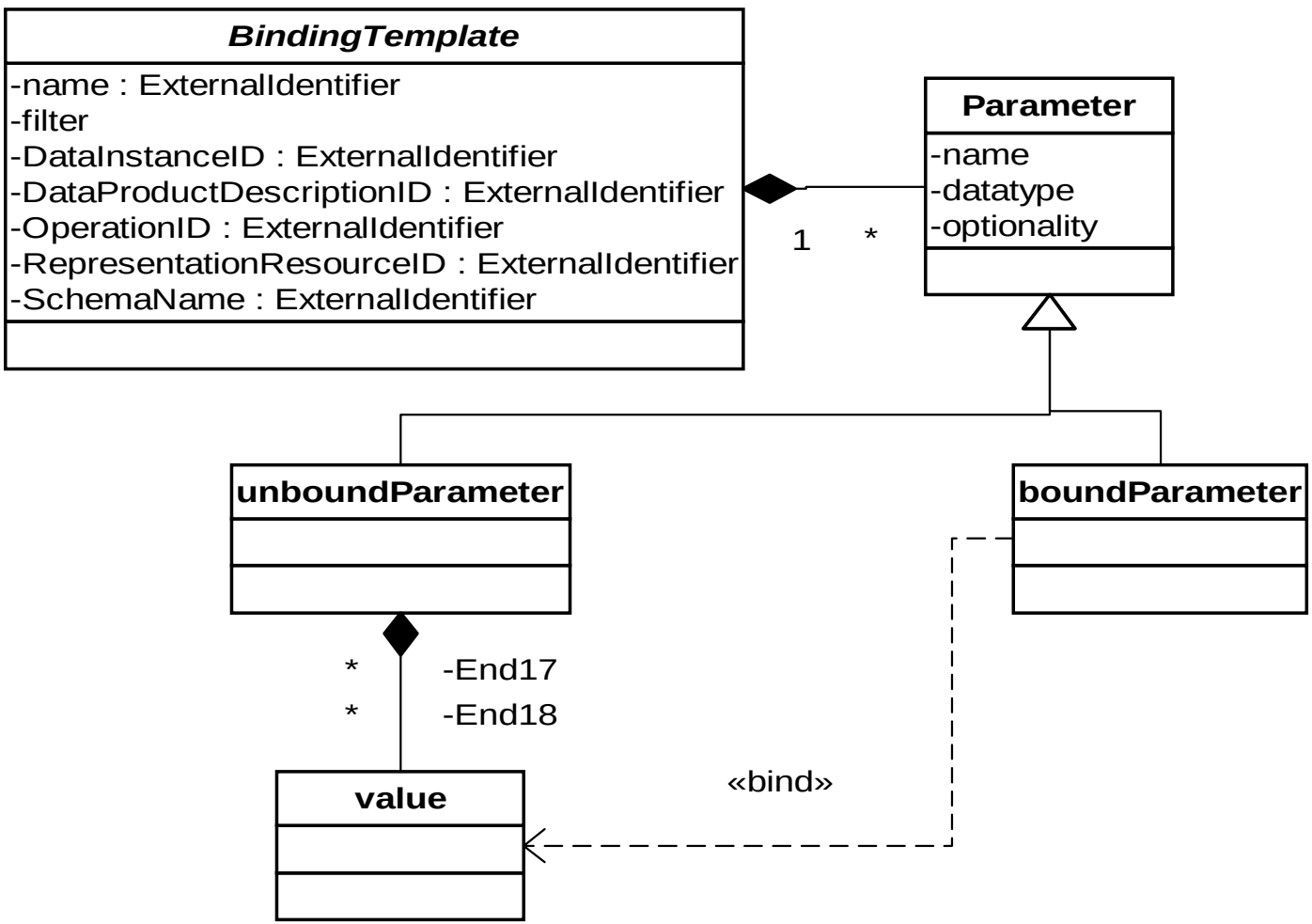
Web Map Server Example



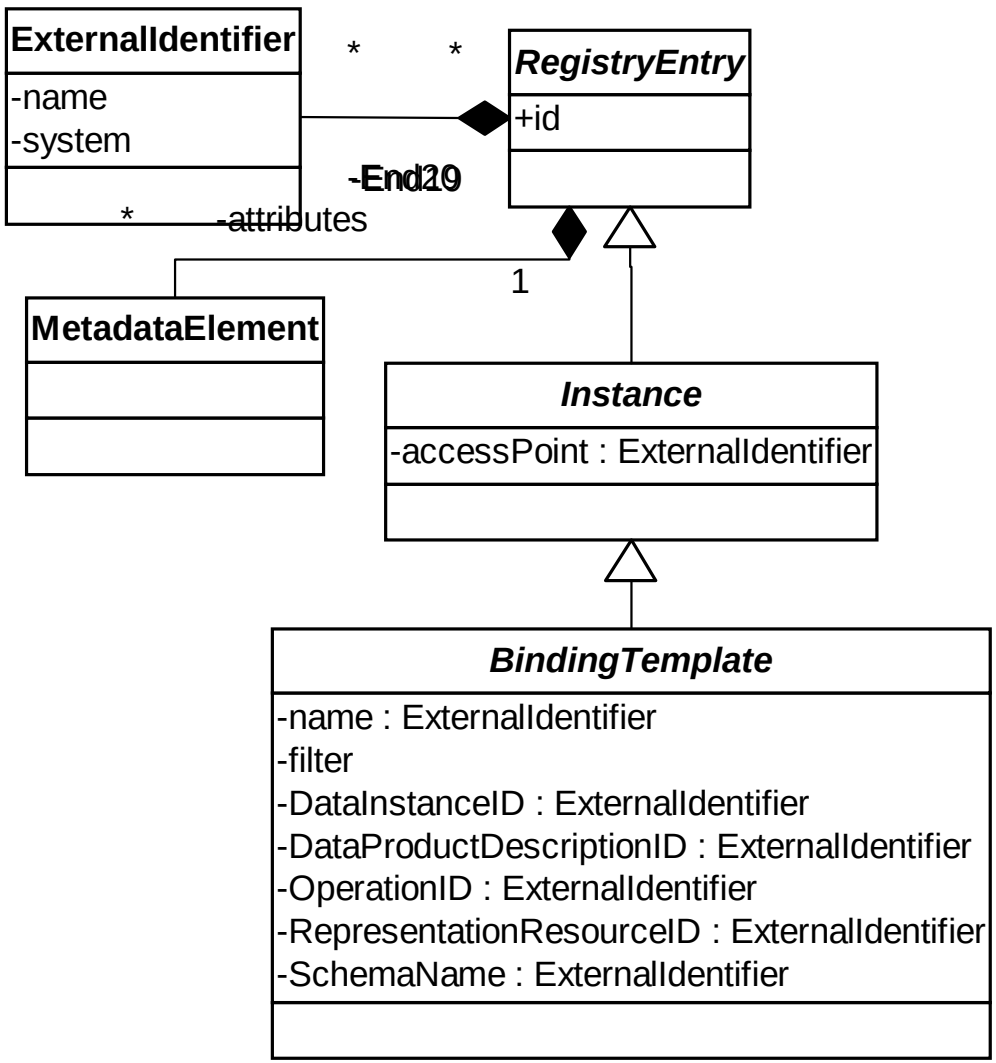
Instances of BindingTemplates



Parameter Model



BindingTemplate



Note:

- Can be retrieved from a registry
- Has name, attributes to support persistence
- Contents (attributes TBD by further modelling)
- TBD - modelling relationships to other BindingTemplates

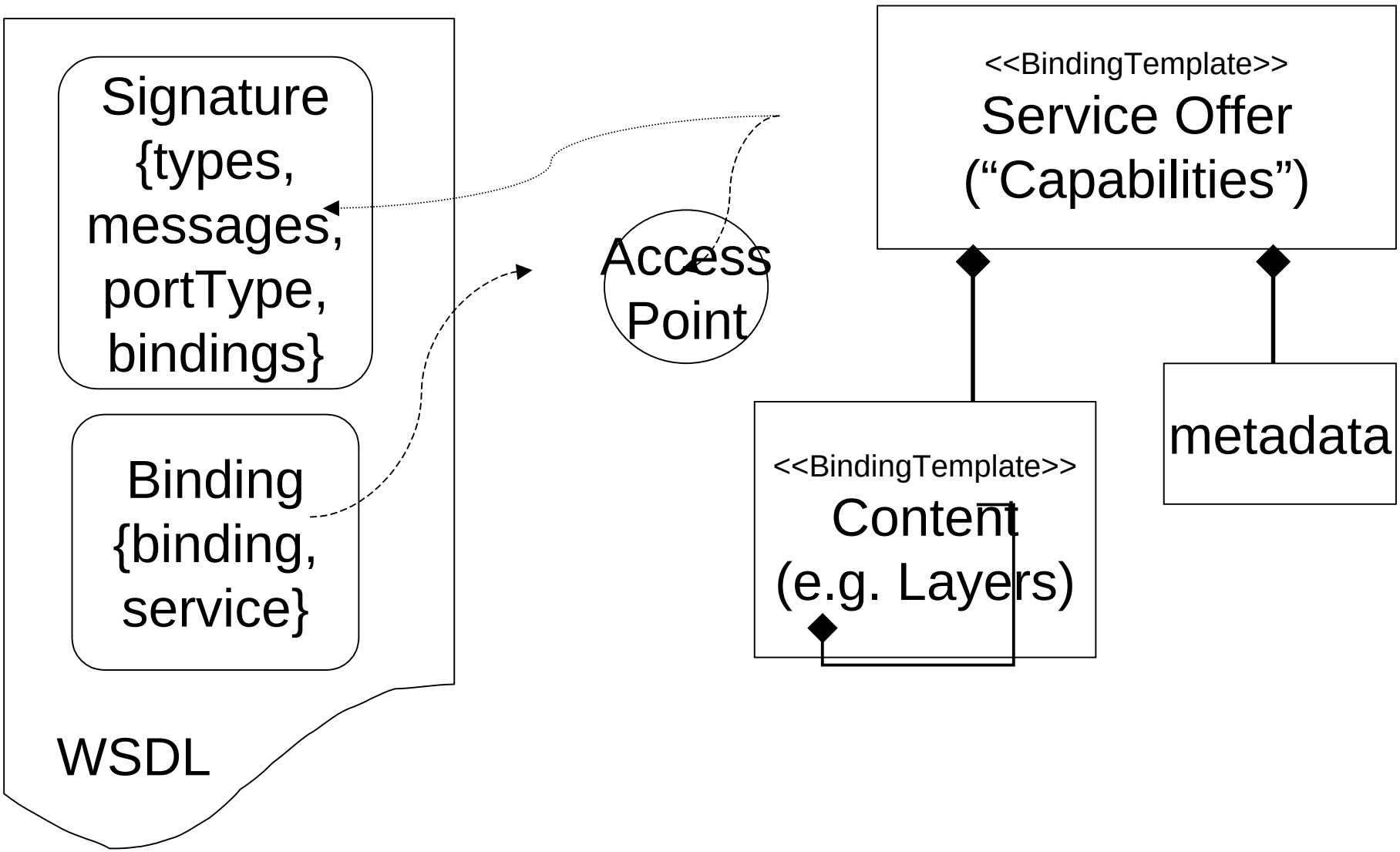
Benefits

- Common pattern across all interface specifications
- Common rules for resolving bindings and invoking services established
- Ability to deploy content access services in a meaningful way (well known types and content domain mechanisms available)
- Ability to store service invocation patterns in registries

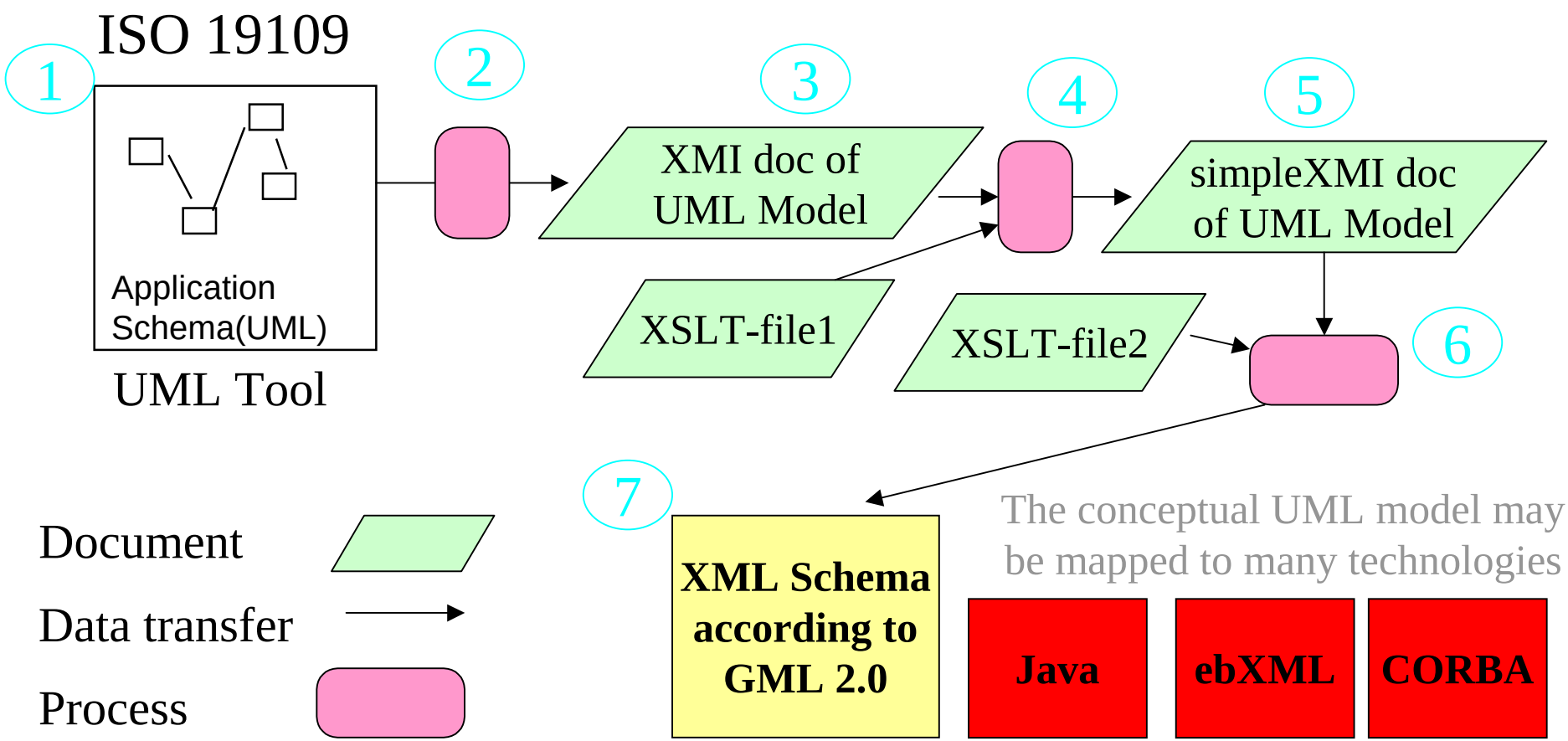
Why not just WSDL?

- WSDL provides some of what we need
- Granularity does not match early/late binding behaviour
- Data Types not expressive enough
- Enumerations not powerful enough for real world content (29000 species of flora in NSW Australia!)
- OpenGIS interface specifications mainly provide semantic definitions of parameters – these are not per-service!

WSDL ++


Signature
{types,
messages,
portType,
bindings}
Binding
{binding,
service}
WSDL
Access
Point
<<BindingTemplate>>
Service Offer
("Capabilities")
<<BindingTemplate>>
Content
(e.g. Layers)
metadata

GeNorway project - XSLT-based code generation from UML to GML for data exchange



\GML2.0 spec. city application schema example

```
<complexType name="RiverType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
          <sequence>
             <element ref="gml:centerLineOf"/>
          </sequence>
      </extension>
</complexContent></complexType>
<complexType name="RoadType">
   <complexContent>
      <extension base="gml:AbstractFeatureType">
         <sequence>
             <element name="linearGeometry" type="gml:LineStringPropertyType"/>
             <element name="classification" type="string"/>
             <element name="number" type="string"/>
         </sequence>
      </extension>
</complexContent> </complexType>
```

simpleXMI - reducing the complexity of XMI

```
<class name="Road" superClass="CityFeature" abstract="false">
        <attribute name="classification" type="CharacterString"/>
        <attribute name="number" type="CharacterString"/>
        <attribute name="linearGeometry" type="GM_Curve"/>
</class>
<class name="River" superClass="CityFeature" abstract="false">
        <attribute name="centerLineOf" type="GM_Curve"/>
</class>
<class name="CityModel" abstract="false">
        <attribute name="dateCreated" type="Date"/>
        <relationship name="cityMember" otherClass="CityFeature"
                cardinality="0..*" collectionType="set"
                aggregationType="composite"/>
</class>
```

Example and ISO 19118 encoding

```
<complexType name="Road">
                <sequence>
                        <element name="classification"
type="CharacterString"/>
                        <element name="number" type="CharacterString"/>
                        <element name="linearGeometry"
type="GM_Curve"/>
                </sequence>
                <attributeGroup ref="IM_ObjectIdentification"/>
        </complexType>
```

Conclusion

- **The OGC (and ISO 19119) Service architecture patterns and the service binding template could be reused by other ”information-intensive” domains**

To support ISO/TC211 and OGC and other similar domain standardisation Organisations, OMG should as soon as possible produce the following:

- **Well defined rules and tools for creating platform-independent service and information models**
- **Well defined rules and tools for mappings to platform-specific models for the most important platforms: Web services/XML, CORBA, J2EE/EJB, SQL, ...**
- **A version of XMI (2.0?) that can produce human readable (HUTN) XML – similar to the GML and ISO 19118 schema specifications currently being made**
- **A tool-independent model diagram interchange format that can be used to support interoperability between multiple UML tools (the diagram interchange RFP!)**