Open GIS Consortium Inc.

Date: 2003-05-19

Reference number of this OpenGIS© Project Document: **OGC 03-055r1**

Version: 0.7.1

Category: OpenGIS© OGC Interoperability Program Report –Viewpoint Specification

Editor: Louis C. Rose (BAE SYSTEMS)

Critical Infrastructure Collaborative Environment (CICE) Architecture – Engineering Viewpoint

Copyright notice

Warning

This document is not an OGC Standard or Specification. This document presents a discussion of technology issues considered in an Interoperability Initiative of the OGC Interoperability Program. The content of this document is presented to create discussion in the geospatial information industry on this topic; the content of this document is not to be considered an adopted specification of any kind. This document does not represent the official position of the OGC nor of the OGC Technical Committee. It is subject to change without notice and may not be referred to as an OGC Standard or Specification. However, the discussions in this document could very well lead to the definition of an OGC Implementation Specification.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type: OpenGIS© Draft Interoperability Program Report –Viewpoint Specification
Document subtype: OGC Critical Infrastructure Protection Initiative (CIPI)
Document stage: Draft
Document language: English

Contents

i. Preface iii

ii. Document Contributor Contact Points iii

iii. Revision history iv

1. Introduction 1
1.2 Scope 1
1.3 Conformance 2
1.3.1 Viewpoint correspondences 2
1.3.2 Reference points 2
1.4 Normative references 2
1.5 Terms and definitions 3
1.5.1 Critical infrastructure 3
1.6 Policy 3
1.7 Conventions 3
1.7.1 Symbols and abbreviated terms 3
1.7.2 Requirement levels 4

2 CICE Engineering Viewpoint 5
2.1 Service Trading (Publish – Find –Bind) 5
2.2 OpenGIS Service Framework 8
2.3 CICE Multidimensional Architecture 9
2.3.1 CICE Communities and Nodes 9
2.3.2 Horizontal Dimension 10
2.3.3 Vertical dimension 10
2.3.4 Private Sector Dimension 11
2.3.5 OSF and the Multi-dimensional CICE Architecture 13
2.4 Multi-tier architectures 13
2.4.1 Thin and thick clients 13
2.4.2 Multi-tiers for OSF 14
2.5 Bridging Multiple Networks 17
2.5.1 Open Location Services (OLS) 17
2.5.2 Sensor Web Enablement (SWE) 18
2.5.3 Alert Notification System (ANS) 18
2.6 Distribution Transparencies 19
2.6.1 Access Transparency 20
2.6.2 Failure Transparency 20
2.6.3 Location Transparency 20
2.6.4 Migration Transparency 20
2.6.5 Relocation Transparency 21
2.6.6 Replication Transparency 21

2.6.7 Persistence Transparency 21
2.6.8 Transaction Transparency 21
2.7 Information Security Infrastructure 22
2.7.1 Approaches to Information Security 22
2.7.2 Information Security for Interoperable Communities 27

i. Preface

The OpenGIS Consortium (OGC) is an international industry consortium of more than 250 companies, government agencies, and universities participating in a consensus process to develop publicly available geo-processing specifications. This Draft Interoperability Program Report (DIPR) is a product of the OGC Critical Infrastructure Protection Initiative (CIPI), the objective of which is to provide a vendor-neutral interoperable framework that enables the publication, discovery, and use of geospatial information concerned with the protection of critical infrastructure systems in a range of sectors.

The OGC Critical Infrastructure Protection Initiative is part of the OGC's Interoperability Program: a global, collaborative, hands-on engineering and testing program designed to deliver prototype technologies and proven candidate specifications into the OGC's Specification Development Program. In OGC Interoperability Initiatives, international teams of technology providers work together to solve specific geo-processing interoperability problems posed by Initiative sponsors.

ii. Document Contributor Contact Points

All questions regarding this document should be directed to the editor or the contributors:

Rhonda Fetters
SAIC
RHONDA.D.FETTERS@saic.com

Louis C. Rose (editor)
BAE SYSTEMS
louis.rose@baesystems.com

iii. Revision history

Date	Release	Description
2003-02-25	0.0.1	▪ Initial version of the document;
2003-03-11	0.1.0	▪ Update the TOC ▪ Incorporated section 6.3 from Rhonda
2003-03-13	0.2.0	▪ Update the TOC ▪ Incorporated section 6.2 starting with the ORM section input ▪ Added Secure Socket Set description
2003-04-9	0.3.0	▪ Update the TOC ▪ Edited the document for consistency ▪ Added Publish-Find-Bind section
2003-05-1	0.4.0	▪ Reorganized some sections ▪ Update the TOC ▪ Edited the document for consistency ▪ Added OGS Service Framework section ▪ Added the section on transparencies
2003-05-8	0.5.0	▪ Reorganized some sections ▪ Update the TOC ▪ Edited the document for consistency ▪ Added OGS Service Framework section ▪ Added the section on transparencies
2003-05-8	0.6.0	▪ Incorporated comments from reviews
2003-06-6	0.7.0	▪ Incorporated comments from reviews
2003-06-24	0.7.1	▪ Added security diagram to section 2.7.2

.

Critical Infrastructure Collaborative Environment (CICE) Architecture – Engineering Viewpoint

1. Introduction

ISO RM-ODP (ISO/IEC 10746) is the architectural framework adopted by the OGC for specifying its reference architectures. The four main parts of the standard define viewpoints on open distributed processing (ODP) systems. This specification addresses the engineering viewpoint for a system dedicated to the protection of critical infrastructure component. This viewpoint is concerned primarily with the interaction between distinct computational objects[1]: its chief concerns are communication; computing systems; software processes; and the clustering of computational functions at physical nodes of a communications network. The engineering viewpoint also provides terms for assessing the "transparency" of a system of networked components – that is, how well each piece works without detailed knowledge of the computational infrastructure.

1.2 Scope

This Draft Interoperability Program Report (DIPR) specifies the Engineering Viewpoint for the Critical Infrastructure Collaborative Environment (CICE). This open, distributed processing environment crosses organizational boundaries and includes a variety of components deployed within multiple communities. The CICE leverages OGC Web Services to enable:

- the publication of the availability of critical infrastructure services and data;
- the registration and categorization of published service and data providers; and
- the discovery and use of needed critical infrastructure services and data

Critical infrastructure is a very broad term that encompasses many large-scale systems in a range of sectors: energy, telecommunications, transportation, public health services, and more. Safeguarding such systems involves a welter of political, economic, and legal issues that will not be raised here. Rather, the CICE is more about the creation and maintenance of a *common information operating environment* to support operational, planning, and decision-making activities associated with critical infrastructure protection

[1] http://www.cs.tcd.ie/synapses/public/deliverables/part1.pdf

1.3 Conformance

Assessing conformance requires consistency across the various viewpoints (i.e. clear mappings of concepts) and across the models they define. In general, the set of viewpoint specifications should not make mutually contradictory statements. Furthermore, each specification should include correspondence statements that relate it to other viewpoints.

1.3.1 Viewpoint correspondences

The Enterprise, Information, and Computation viewpoints describe a system in terms of its purposes, its content, and its functions. The Engineering viewpoint relates these to specific components linked by a communications network. This viewpoint is concerned primarily with the interaction between distinct computational objects[2]: its chief concerns are communication; computing systems; software processes; and the clustering of computational functions at physical nodes of a communications network. The engineering viewpoint also provides terms for assessing the "transparency" of a system of networked components – that is, how well each piece works without detailed knowledge of the computational infrastructure.

1.3.2 Reference points

A reference point identifies a behaviour or proposition that must be satisfied at a particular interaction point. A reference point may be declared as a conformance test point used to test observed behaviour. Part two of the RM-ODP standard distinguished four categories of reference points: programmatic, perceptual, interworking, and interchange (not all need be used in every viewpoint specification).

1.4 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this Interoperability Program Report. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this document are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies.

IETF/RFC 2119. *Key words for use in RFCs to Indicate Requirement Levels*. March 1997. Available [online]: <http://www.ietf.org/rfc/rfc2119.txt>.

IETF/RFC 2828. *Internet Security Glossary*. May 2000. Available [online]: <http://www.ietf.org/rfc/rfc2828.txt>.

ISO/IEC 10746-2:1996, *Information Technology – Open Distributed Processing –Reference Model: Foundations*.

---

[2] http://www.cs.tcd.ie/synapses/public/deliverables/part1.pdf

ISO/IEC 10746-3:1996, *Information Technology – Open Distributed Processing –Reference Model: Architecture.*

ISO/IEC 9594-8:2001. *Information Technology – Open Systems Interconnection – The Directory: Public-key and attribute certificate frameworks.* [also published as ITU-T Recommendation X.509 (03/00)].

1.5 Terms and definitions

For the purposes of this Interoperability Program Report, the terms and definitions given in ISO 10746-2 and ISO 10746-3 apply. For convenience, some of these terms are repeated below.

1.5.1 Critical infrastructure

Critical infrastructure, as defined by the "US Patriot Act", are described as "systems and assets, whether physical or virtual, so vital to the United States that the incapacity or destruction of such systems and assets would have a debilitating impact on security, national economic security, national public health or safety, or any combination of those matters."

1.6 Policy

A set of obligation, prohibition, or permission rules that either constrain or enable actions, as related to a purpose. [ISO 10746-2]

1.7 Conventions

1.7.1 Symbols and abbreviated terms

The following symbols and abbreviated terms are used in this document.

ACL	Access Control Lists
ACM	Access Control Modules
ANS	Alert Notification System
API	Application Programming Interface
CA	Certificate Authorities
CI	Critical infrastructure
CICE	Critical infrastructure collaboration environment
CIPI	Critical infrastructure protection initiative
DAC	Discretionary Access Controls
DASC	Distributed Access Control Services
DHS	Department of Homeland Security

DIPR	Draft Interoperability Program Report
DTD	Document Type Description
FGDC	Federal Geographic Data Committee
GML	Geography Markup Language
HTTP	HyperText Transfer Protocol
ISO	International Organization for Standardization
LDAP	Lightweight Directory Access Protocol
ODP	Open Distributed Processing
OGC	OpenGIS Consortium
OLS	Open Location Services
OMG	Object Management Group
ORM	OpenGIS Reference Model
OSF	OpenGIS Service Framework
OWS	OpenGIS Web Services
PKI	Public Key Infrastructure
PFB	Publish – Find – Bind
RM-ODP	Reference Model for Open Distributed Processing
SWE	Sensor Web Enablement
SSL	Secure Sockets Layer
SPM	Security Profile Manager
URL	Universal Resource Location
WFS	Web Feature Server
WMS	Web Map Server
WRS	Web Registry Server
WWW	World Wide Web
XML	Extensible Markup Language

1.7.2 Requirement levels

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

2 CICE Engineering Viewpoint

The OGC CICE architecture is based on the need to enhance the ability of organizations and individuals to use geospatial processing technologies in an open distributed processing environment to address issues associated with assuring the continuity, viability and protection of critical infrastructure.

This open distributed architecture for CICE will be described from four non-overlapping viewpoints. These viewpoints are: Enterprise or business; Information content and system behavior; Computational components and interfaces; and the Engineering viewpoint. This collection of interrelated viewpoints will provide for the development of CICE as an initiative that supports the multiple technology goals of its sponsors within a business/mission based structure. It will also provide a way of ensuring that that the elements of CICE become incorporated into the OGC Reference Model, the FGDC Geospatial Interoperability Reference Model and other appropriate models, so that specifications based reusable elements can be adopted for common use not only within the Critical Infrastructure Community, but also within the entire geospatial information community.

The purpose of the Engineering viewpoint is to describe the distributed nature of the CICE architecture and provide standard definitions, which describe engineering constraints. It focuses on the mechanisms and functions required to support distributed interactions between interoperable communities (nodes) in the system. Since the engineering viewpoint is primarily concerned with the interaction between distinct computational objects, its chief concerns are communications; computing systems; software processes; and the clustering of computational functions at physical nodes of a communications network.

2.1 Service Trading (Publish – Find –Bind)

The core method of communications within the CICE is based on service-oriented architecture that follows a service trading paradigm. Service trading is a fundamental concept that addresses the discovery of available service instances. For CICE, these service instances are those that implement OGC interface specification. The CICE facilitates the offering and the discovery of OGC interfaces which provide services of particular types (e.g., WMS, WFS services). Publishing a capability or offering a service is called “export”. Finding a service request against published offers or discovering services is called “import”. Binding a client to a discovered service is called “service interaction”. This can also be depicted in an equivalent manner as the “Publish – Find – Bind” (PFB) pattern of service interaction. These fundamental roles and interactions are depicted in Figure 1.

This service trading function is elaborated in a separate document (ISO/IEC 13235-1) and refined somewhat in the Object Management Group (OMG) Trading specification, which is technically aligned with the computational view of the ODP trading function. Most importantly, a broker supports dynamic (i.e. run-time) binding between service providers and requesters, since sites and applications are frequently changing in large distributed systems. A broker registers service offers from provider objects and returns service offers, upon request, to requestor objects according to some criteria.

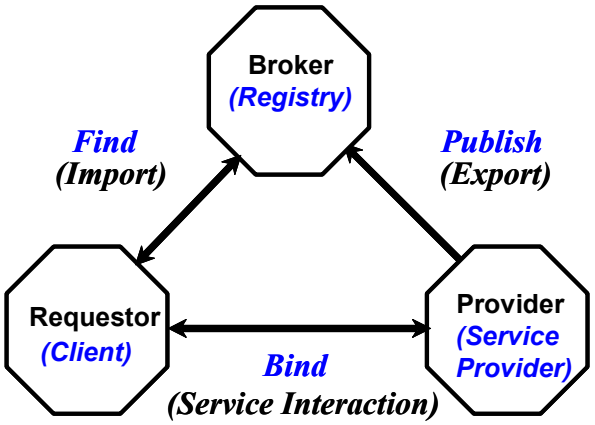

Figure 1 – Service Trading Communication Sturcture

In the CICE architecture, there are three fundamental roles that are defined to actuate the service trading. They are:

Broker - a role which registers service offers from service providers and returns service offers upon request to requestor according to some criteria.

Provider - a role which registers service offers with a broker and provides services to clients.

Requestor - a role which obtains service offers, satisfying some criteria, from the broker and binds to discovered services provided by the provider.

In effect a broker plays the role of "matchmaker" in a service-based architecture, as suggested by the informal sequence diagram in Figure 2.

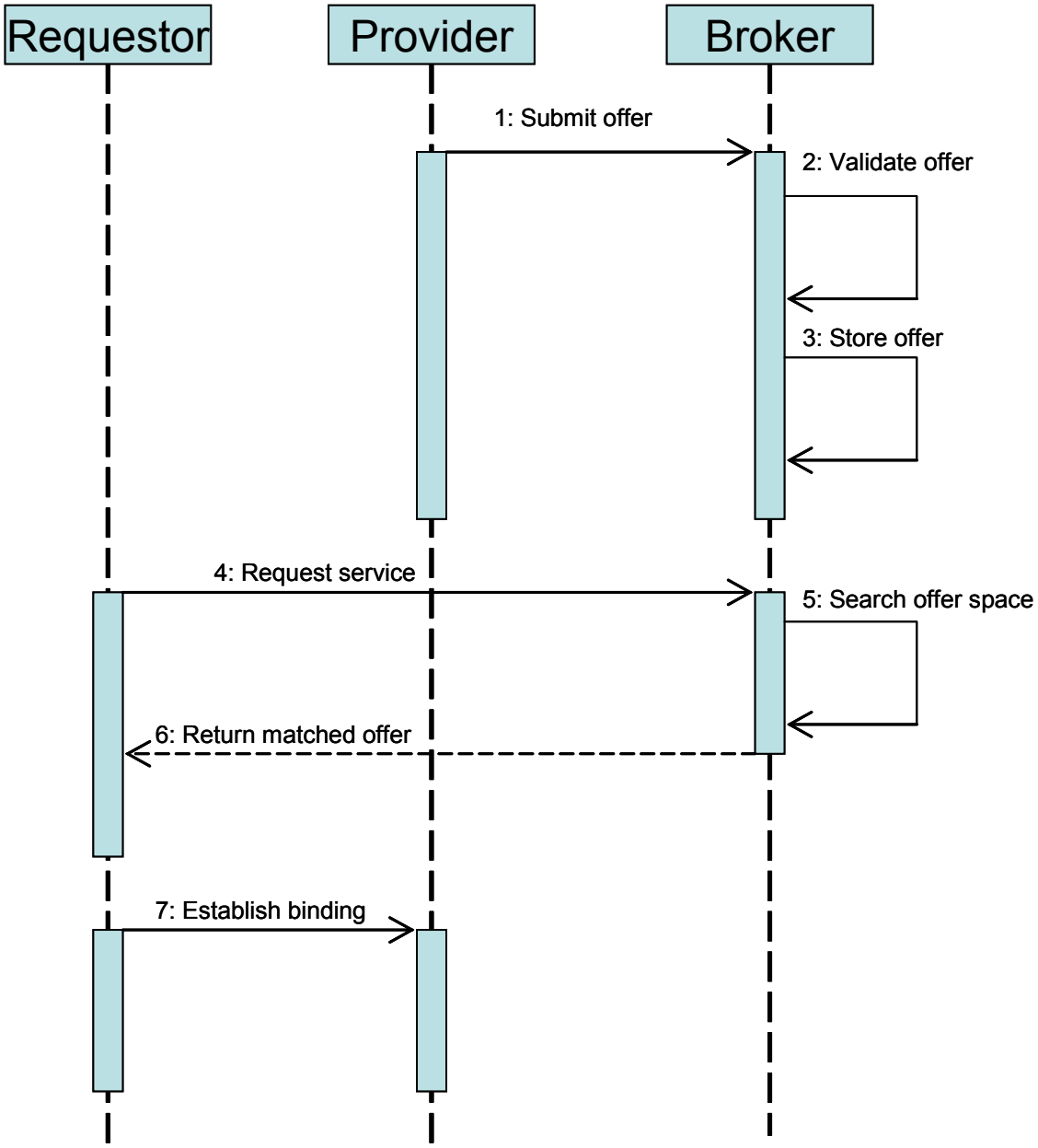

Figure 2 – Trading Interactions

To export (i.e. publish a service offer), an object gives the broker a description of a service, including a description of the interface at which that service instance is available. To import (i.e. find suitable service offers), an object asks the broker for a service having certain characteristics. The broker checks against the descriptions of services and responds to the requestor with the information required to bind with a service instance. Preferences may be applied to the set of offers matched according to service type, some constraint expression, and various policies. Application of the preferences can determine the order used to return matched offers to the requestor.

2.2 OpenGIS Service Framework

The Publish-Find-Bind service framework is the foundation of the OpenGIS Service Framework (OSF) that is employed in the CICE architecture. In the OSF, the role of broker is implemented with a set of registry services. The role of provider is accomplished by a collection of service types; 1) data services; 2) portrayal services; and 3) processing services. The role of requestor is typically provided by a set of client applications. The OSF is depicted in Figure 3.

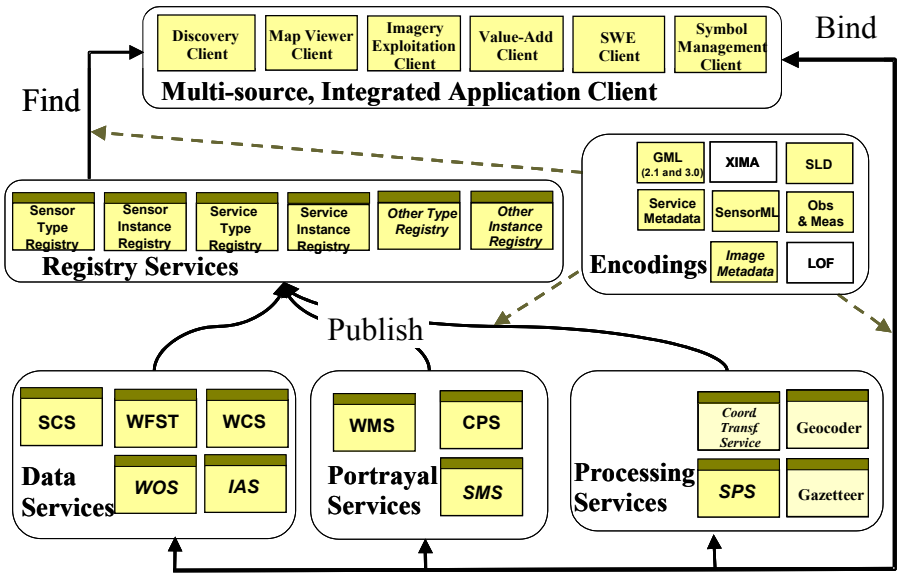

Figure 3 – OpenGIS Service Framework

In addition to providing the clients and services needed to support the requestor, provider and broker roles, the OSF also contains a set of encoding methods that are used as the foundational data structures in the message communications that implement the Publish, Find and Bind operations.

The structure of the OSF shown in Figure 3 might tend to imply that an instance of the OSF is self- contained on a single platform. While this approach could be realized, generally this is rarely the case at all. The OSF, and therefore CICE, is architected to operate on the Internet over the World Wide Web (WWW). Any client (requestor) on the WWW can potentially find registered services (or data) from any OGC registry (broker) on the WWW and bind to any OGC service provider (provider) on the WWW. The distributed operations in the CICE architecture are supported by the instantiations of OGC interface specifications

that are implemented using OGC Web Services. This provides CICE with a distributed operating environment that enables the sharing of geospatial information that fosters an environment of collaboration. The flexibility of applying the OSF in support of the distributed collaborative environment envisioned in the CICE architecture is presented in the subsequent sections of the CICE Engineering Viewpoint.

2.3 CICE Multidimensional Architecture

Analyses of U.S. and Canadian government operations indicate that critical infrastructure protection represents a substantial portion of Federal lines of business. Therefore, the CIPI business environment is firmly grounded in the needs of government to provide information and services to responders and others who need geospatial resources and functionality to detect, prevent, protect, respond and recover from potential threats to critical infrastructure assets. CIPI is an enterprise activity that crosses all sectors of the economy and all levels of government and non-government activity, both nationally and internationally.

The Critical Infrastructure Collaborative Environment (CICE) will involve the collaboration of multiple community nodes with varied system architectures in place that suit the needs of participant organizations. Such organizations will want to leverage their existing infrastructure and may not be willing or able to adapt to a predetermined architecture for both economic and organizational reasons. Therefore, CICE will be a multidimensional architecture that is participant-driven and based on the unique requirements, local data models, organization security policies and technologies that a participant will have in place.

The potential complex horizontal and vertical interactions between government entities, including state and local government, cross-national agencies, the private sector, and general public that CIPI requires will depend on an interoperable network between and across heterogeneous nodes. As such, the CICE Engineering Viewpoint does not define a specific system architecture, but addresses how the horizontal and vertical dimension can negotiate a set of agreements between components. At a minimum, the required agreements will include networking, request/response protocols, service definitions, data security, and information models.

2.3.1 CICE Communities and Nodes

The CICE Architecture can be viewed as consisting of a collection of operating communities, as discussed in the CICE Enterprise Viewpoint. Each CICE community has an operational focus unto itself that most likely consists of a group of departments or agencies that have even more focused operating concerns. For instance, the Department of Homeland Security (DHS) has an overall operational focus to protect the United States homeland from internal and external threats. But DHS, however, includes a diverse set of activities such as border protection, emergency management, interdiction, critical infrastructure protection, and many more. The organizations within DHS that support these activities can be considered operating nodes within a CICE community that must work together to achieve the overall DHS mission. For example, activities relating to coastal security could be available through a Coast Guard node. Every nation's federal level of operations contains many such communities, with each community most likely containing multiple nodes. This scenario is repeated at each level of governmental jurisdiction, such as state or provincial, county and

local levels. In an effort to protect a country's critical infrastructure, the need for sharing geospatial information is essential to providing that protection. To be effective, this collaboration among communities must occur regardless of the jurisdictional level of the collaborating communities. The multiple dimensional aspects of this type of communication are discussed in the following sections.

2.3.2 Horizontal Dimension

The horizontal dimension of the CICE architecture represents the interactions among agencies, organizations, or entities at the same level of critical infrastructure business operations. This horizontal dimension of communication has both an internal and external element to it with respect to each large operating entity.

For example, a country's government operations contain many different agencies that are responsible for monitoring, controlling and, in some cases, operating the country's critical infrastructure. Given that most critical infrastructure is tightly interdependent with one another, the sharing of information in general, and geospatial oriented information in particular among agencies, improves the economy, efficiency, and effectiveness of all agencies involved in the protection of critical infrastructure. Each one of these agencies can potentially be a CICE supported community that will enable the sharing of geospatial information about the various infrastructures that the agencies are controlling or monitoring. This scenario represents the horizontal dimension of an overall CICE architecture that is internal to a particular country's governmental operating environment. In today's world of global communications and shared infrastructure, there is a growing need for international information sharing during times of natural or political crises. This emerging situation requires that the CICE nodes within one country may very well have to communicate with CICE nodes in other countries as well. A presentation of this international CICE communication is shown in Figure 4.

The same scenario plays out at each level of governmental jurisdiction. The states or provinces of each country, the counties of each state, and local governments of each county benefit by sharing geospatial information with respect to protecting their related critical infrastructure. A diagram that is analogous to Figure 4 could be drawn for each level of jurisdiction, but it is believed that the concept is easily applied to each level without additional diagramming.

2.3.3 Vertical dimension

The vertical dimension of the CICE architecture represents the interactions among agencies, organizations, or entities at the different levels of the critical infrastructure business operations.

Critical infrastructure sectors identified in the Enterprise viewpoint include: Communications, Energy and Utilities, Financial Services, Transportation, Law Enforcement, Fire, Government Operations, Public Health and Human Services, Internal Security, Foreign Intelligence and Affairs, and National Defense. Many of these sectors clearly extend beyond a single horizontal dimension and may involve all levels of government, international coordination, and private sector support or involvement.

For example, let's consider a scenario in the area of Law Enforcement. When a crime is committed, the first responders are generally the local law enforcement authorities. If

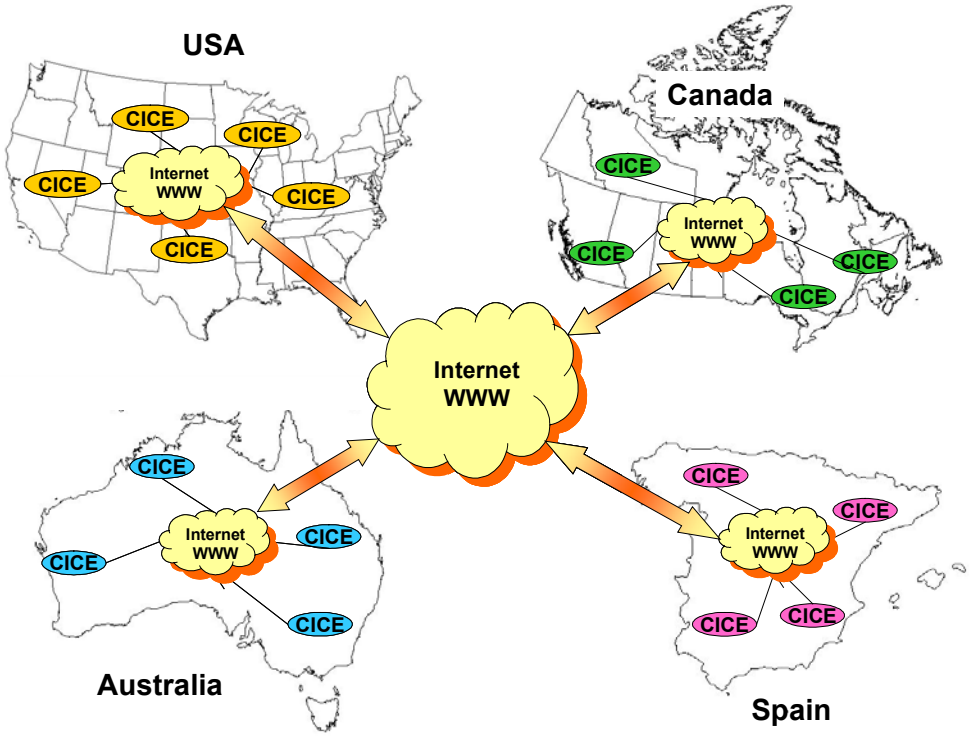

Figure 4 – The CICE Architecture Horizontal Dimension.

the investigation expands outside of the locality's jurisdiction, another jurisdiction might be called into action (the horizontal dimension) along with state law enforcement authorities (the vertical dimension). If the investigation proceeds to another state, this can many times upgrade the crime to a federal crime requiring state and federal law enforcement cooperation (also the vertical dimension). Although this particular example depicts an evolving communications scenario that develops from the lower level to the top, there potentially will be the requirement for any level to engage any level at anytime during a time of crises, whether the crises are natural or man-made. These multilevel geospatial communications scenarios are presented in Figure 5.

2.3.4 Private Sector Dimension

Up to now, the discussion has focused on the interaction requirements within and between each level of government as they work together to protect critical infrastructure. But additionally, there is, however, an explicit requirement for these governments to interoperate

with elements of the private sector. Much of the critical infrastructure in many nations is owned and operated by the private sector. In the United States, in particular, 85%[3] of the critical infrastructure is owned by the private sector. While each organization in the private sector has their own policies and procedures for protecting their infrastructure business, coordinating a protection program that is in a country's best interest has become essential in today's global social-political environment. The need for geospatial interoperability between the government and private sectors is essential to achieving a sound and effective critical infrastructure protection program. From the CICE Architecture Engineering Viewpoint, the private sector can be viewed as another layer in the vertical dimension that will potentially need to interoperate with all levels of the governmental hierarchy shown in Figure 5. To further complicate the picture, some private sector infrastructure companies have global operations in many countries, especially in the communications, financial services and transportation industries.

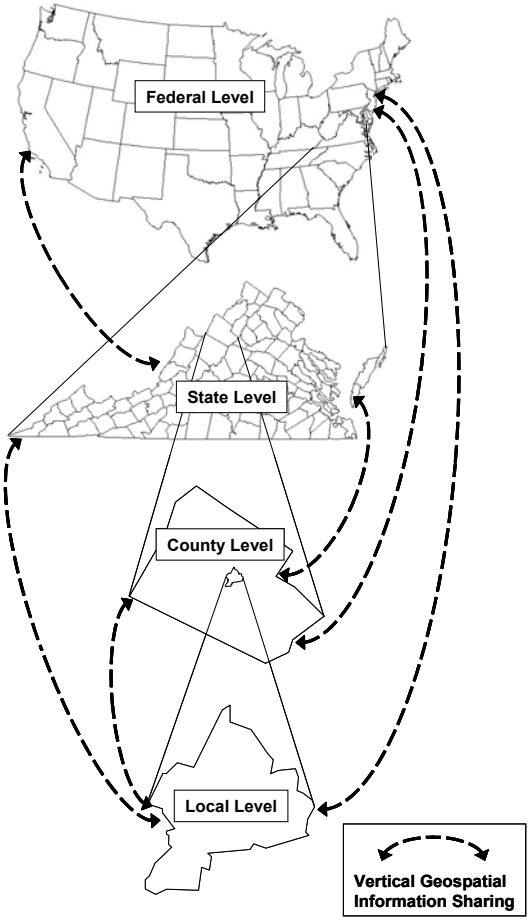

Figure 5 – The CICE Architecture Vertical Dimension.

[3] http://www.dhs.gov/dhspublic/display?theme=52

2.3.5 OSF and the Multi-dimensional CICE Architecture

The relationship between the OSF and the CICE multi-dimensional CICE architecture is in needs of further discussion. The OSF structure shown in Figure 3 identifies the basic Publish-Find-Bind paradigm that is supported by many OGC Web Services. This OSF structure can be applied both as one particular CICE node and/or distributed across an OGC Web Services compliant network.

For example, a particular government agency might have a variety of OGC compliant portrayal, data and processing services providers that publish their services to one or more registries within the organization. Clients that support the organization's business processes can then discover the available services in the registry and bind to these discovered services to provide the necessary end user functionality. But, there is nothing, however, to prevent these same clients from searching registries that are resident on other CICE nodes that identify services that are available from any other CICE node that is on the network. So, in essence, any client can potentially interface with any registry to find any service provider, whether in the horizontal or vertical dimension, as long as the locations of the registries are disseminated and appropriate access controls are followed.

2.4 Multi-tier architectures

The Engineering viewpoint for the CICE describes how the system assigns functions and information to various components, or tiers, along a network within and between interoperable nodes. The horizontal and vertical dimensions of critical infrastructure protection introduce a complex set of network transactions. The assumption that every node between and across these dimensions will have the ability to interface is unrealistic. Therefore, CIPI will be based on multi-tiered architectures where operational nodes exchange information by agreeing on uniform interoperability conventions.

Computational functions, data, and metadata may be found on the server side, in one or more intermediate "middleware" components, or on the client side[4]. Figure 6 shows several categories of services arrayed in a logical 4-tier architecture, and mapped to different physical architectures.

2.4.1 Thin and thick clients

The engineering viewpoint articulates the key distinctions among distributed systems:

- **Thin clients** rely on invoking the services of other components (servers, middleware) for most of the computation they need to function in the system; they also rely on other components to manage most of the data and metadata they need.

[4] http://www.imn.htwk-leipzig.de/~kudrass/Publikationen/OOPSLA99.pdf

- **Thick clients** handle much of the necessary computation and data/metadata management themselves; and rather than invoking the processing services of other components, they obtain their inputs through low-level data-access requests.

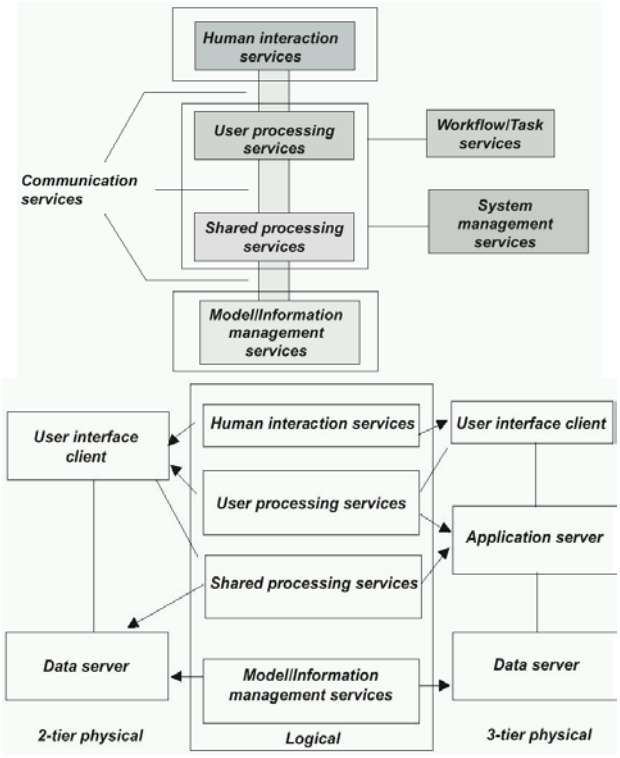

Figure 6- Logical multi-tiered architecture mapped to different physical architectures

A thick client requires less functionality on the part of the server and other components; but a thin client is easier to build or to embed into general-purpose software components. The distinction often has quite tangible implications: thin clients are typically simple software with limited functions and flexibility, and smaller RAM and CPU requirements, often suitable for handheld or mobile devices. Thick clients usually require a significant portion of (at least) a microcomputer's resources, but provide greater flexibility and capacity to decode, transform, render, and interact with retrieved data.

2.4.2 Multi-tiers for OSF

The Open Location Services (OLS) and OpenGIS Web Services (OWS) initiatives investigated an extension to the OSF that can be applied to a CICE node that investigated the use of server side client applications. Server-side client applications are defined as "the main server-side components of client applications". In the OSF configuration, which was presented in section 2.2, classes of application clients were identified that all operated at the same level in the architecture. The OWS Services Framework investigated the use of

application servers to provide this server side processing for application clients. This extension to the OSF describes how these components run on the server side of the network, drawing on "user application logic" (business logic) to invoke Registry, Processing, Portrayal, and Data services, and to interact with client-side components through a Web/Portal Server. These components generalize the "viewer client generators" of Web mapping to support thin (small, simple) clients running on mobile devices such as cell phones. Server-side client applications fit into a larger architecture of services, depicted below.[5]

OpenGIS Services are accessible from Application Services operating on user terminals (e.g. desktop, notebook, handset, etc.) or servers that have network connectivity and that utilize OpenGIS service interfaces and encoding specifications (Figure 7). Users may use Application Services to access Registry, Portrayal, Processing and Data Services, depending upon the requirements and designed implementation of the application. Application Services commonly, but not necessarily, provide user-oriented displays of geospatial content and support user interaction at the user terminal. Application Services may be realized as marked-up text (e.g., HTML or XML) transferred across a network from a server, software modules (e.g., Java classes or ActiveX components) transferred across a network and executed on a local system, or as executable code resident on a local system. OpenGIS Application Services may also support privacy and access controls based on authenticated user identity, however such controls will typically be provided by an authentication server or some other access control mechanism. Figure 7 illustrates the distinction between client-side and server-side Application Services.

Client-side Application Services should:

- Provide the means to find geospatial-based services and data resources through search and discovery mechanisms of Registry Services;
- Provide access to geospatial data (e.g, geographic features and images) and other geospatial-based Application Services and Data Services;
- Provide drill-down access to features corresponding to geospatial data (e.g, jurisdiction references, telephone numbers of responding agencies and civilian populations) to be used with specific applications such as alert and notification;
- Integrate with a range of deployment platforms from Web browser-based to desktop to wireless handsets;
- Portray geospatial information in graphic, image, and/or text form;
- Support user interaction via keyboard, cursor or other human-machine interfaces

[5] "Application services" is used in the engineering view as a grouping of services on the users terminals.

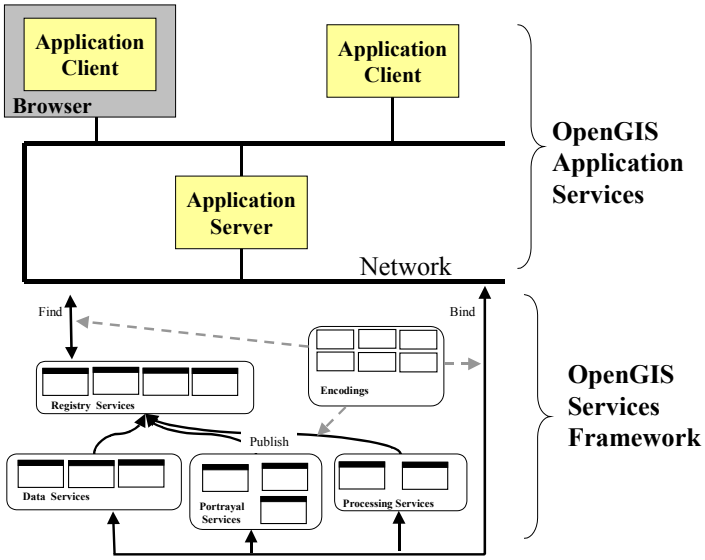

Figure 7 – Application Services and the OWS Services Framework

Application Services should be able to execute not only on the user's desktop (or handset), but also on a server on the network. Examples of server-side Application Services include compute-intensive (and/or I/O-intensive), server-based applications like those required for Image Processing or Route Determination.

Server-side Application Services:

- Implement user application logic (business logic) that utilizes supporting OpenGIS Framework Services such as Registry, Processing, Portrayal, and Data Services.
- Interact with client-side Application Services through an appropriate network protocol depending on the platform being used.
- May be deployed as components of Web Portals and web-accessible business services that add-value to underlying OpenGIS Framework Services.

The above discussion of client-side and server-side Application Services notwithstanding, the OGC Services Framework does not distinguish the myriad options for deploying applications on a network. Instead, any user-facing software component that performs a service that satisfies user-requirements, whether it executes on the client or on the server side of a network connection, is simply an Application Service. The Application Services described below categorize applications by logical function and not physical deployment. Implementations of OpenGIS Application Services are, through standardized interfaces and encodings, freely able to mix and match the capabilities of OpenGIS Services Framework into physical implementations to meet market or application-specific requirements.

2.5 Bridging Multiple Networks

The CICE architecture must also provide a way to describe systems whose components bridge more than one communications network. Each one of the CICE nodes is, in essence, its own self-contained communications network, regardless of where the CICE fits in the horizontal or vertical dimensions. There will be a need for most of these CICE nodes to reach out to communication networks that are not part of the OGC Service framework in order allow the GIS information processed by a CICE node to interact with other established communications networks.

Three examples of how different OGC initiatives have addressed this issue of bridging to multiple networks are discussed in the following sections. All of these methods might potentially be incorporated into CICE nodes as the need arises.

2.5.1 Open Location Services (OLS)

The location services market demands technology that subscribes to the principal of simplicity so that these services will be widely adopted throughout the wireless-IP realm. The Open Location Services (OLS) initiative introduced **gateway services,** which link location application services (accessed via the Internet or the Web) with mobile wireless-IP platforms, in support of small form factor mobile terminals:

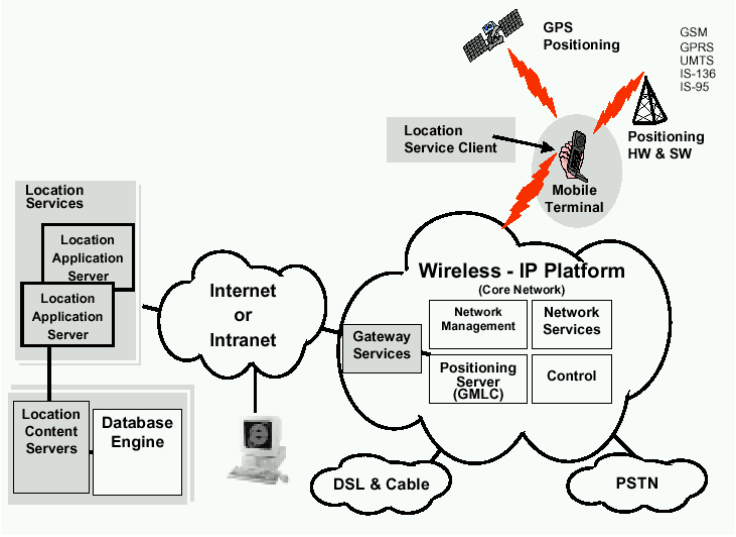

Figure 8 - OpenLS system concept

The Gateway Service is employed to obtain the position of the subscriber's mobile terminal from the network. A Location Service Client sends the request to determine a position to the Gateway. The Gateway calculates the position of the subscriber's mobile terminal and forwards to the Location Service Client, which may store it for as long as needed.

2.5.2 Sensor Web Enablement (SWE)

The OpenGIS Web Services (OWS) initiative defines a Sensor Web Enablement (SWE) thread to link environmental sensors to the World Wide Web. A Sensor Collection Service (SCS) server gathers readings from in-situ environmental sensors via a private network (cellular, microwave, etc.), and provides summaries or interpretations of those readings to SCS clients over the Web, as depicted below.

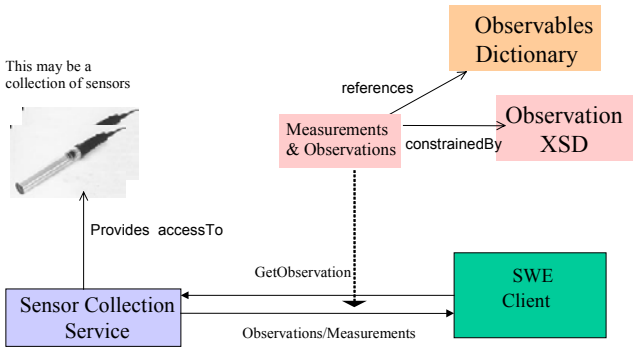

Figure 9 - Sensor Collection Service concept

The SCS is one of many interrelated components in the Sensor Web Environment. The SCS is a piece of software that accepts requests for information about sensors and provides access to and responds with information about a sensor, set of sensors or sensor proxy or sensor station. The SCS provides two basic types of information: 1) the specific capabilities of the sensors belonging to the SCS and 2) the observations or measurements supported by the SCS

2.5.3 Alert Notification System (ANS)

The first phase of the first Critical Infrastructure Protection Initiative (CIPI 1.1) investigated the need for a standard method for handling the geospatial aspects of an Alert Notification System (ANS). An ANS is used for the dissemination of alerts (directly or indirectly) to affected populations, captive population managers, first responders, policy makers and more. An ANS provides different recipients/subscribers with different views into the overall 'situation' based upon their 'profile' which could include a range of information including their interests, jurisdiction (and role), their geographic proximity, and/or their security clearance. Also, an ANS Alert could be 'portrayed' differently for different dissemination channels (e.g., land-line phone, cell phone, 3G mobile computing platform, desktop browser, television, etc.), depending on the technical infrastructure available to different recipients/subscribers.

At the most basic level, an ANS Alert would include 'header metadata' and a 'spatial data package'. The header metadata would authoritatively inform a recipient/subscriber of the possible need to take action in the face of an event. The spatial data package would provide the recipient/subscriber with situational awareness necessary for him/her to properly respond in a spatially informed manner.

An example of an ANS architecture is presented in Figure 10:

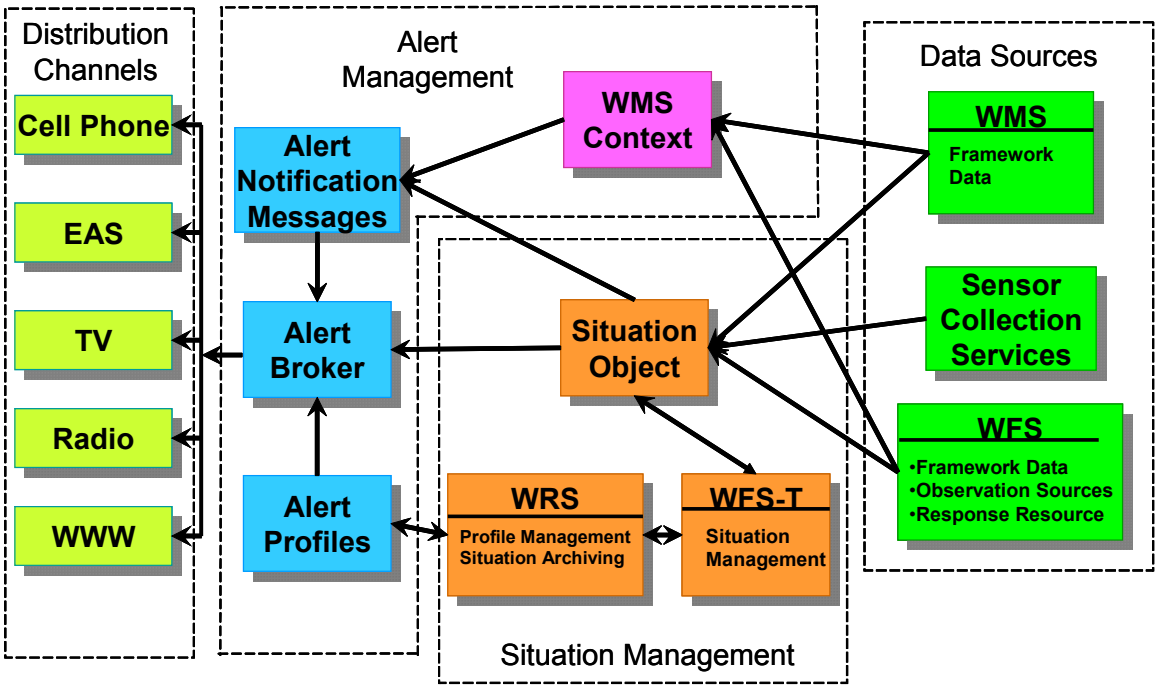

Figure 10 – Alert Notification System Architecture

The ANS Broker manages the notification process. The Broker may have awareness of multiple ANS services capable of providing notification using a variety of means. For a given Situation, the Broker may choose to use one or more ANS services to provide notification, based on location, profile data, and alert notification media. The Broker has awareness of the state of notification, and can stop or modify the notification based on the Situation Object.

2.6 Distribution Transparencies

The CICE Architecture is structured to specifically address the eight distribution transparencies that are discussed in the ORM which were carried forward from the RM-ODP engineering viewpoint guidelines. It is essential that a distributed system hides complexities associated with system distribution from applications, otherwise the system will be rendered unusable. The CICE architecture addresses these transparency issues through the use of the Publish-Find-Bind service trading paradigm (See section 2.1) and the implementation of the OGC interface specifications using the HTTP protocol, XML and XML schemas. The subsequent sections describe how of the CICE architecture's use of the OSF as the foundation addresses each of the eight transparencies called for in the RM-ODP.

2.6.1 Access Transparency

Access transparency is concerned with hiding differences in data representation and making the invocations of access mechanisms transparent to the user. In CICE, this is accomplished with mainly with the design of the OGC interface specifications and their implementations. Each OGC compliant server (e.g., WFS, WMS, and WRS) is required to provide a GetCapabilities service along with a set of services that are focused on the needs of the particular server. The GetCapabilities service is designed to provide service-level metadata that identifies a server's information content and acceptable request parameters. The server capabilities are returned in an XML formatted document that complies with an associated XML schema. Using this method, the service-level metadata describes the capabilities of the server in a consist manner regardless of each servers' underlying data representation. In addition, the use of the HTTP GET and POST operations as the underlying routine access mechanisms for all OGC interface specifications completely hide this access from the invoking software.

2.6.2 Failure Transparency

Failure transparency is concerned with hiding a service's failure recovery mechanism from the invoking software. Each OGC interface specification requires each service to produce a Service Exception Report in XML format that is valid according to the Service Exception Document Type Description (DTD). With this exception processing handled over HTTP, the particular error processing of the service is fully transparent and the invoking software (or user) can process the Service Exception Report in what ever manner best fits the circumstances at the time of the failure.

2.6.3 Location Transparency

Location transparency is concerned with hiding the physical location of any data or service from the invoking platform. In the CICE architecture, the physical location of both data and services is handled with data and service registries that are defined in the Web Registry Server (WRS) specification. The WRS represents the Trader role that was presented in section 2.1. A registry stores the metadata about the location of data and services that have been published by the service providers. It is each service provider that is aware of the location of its data and services; and with this information published to a registry, a requesting client can search the registry's metadata that describes the type of data or service of each registry entry. When the client finds the data or service of interest, it need only retrieve the location from the registry metadata in order to retrieve that data or invoke the service. Since all OGC compliant data and service providers operate in the HTTP environment, the location of the data or service is provided with a Universal Resource Location (URL) that can be accessed by the client using a standard method. The result is that the location of data or service is completely transparent to the requesting client.

2.6.4 Migration Transparency

Migration transparency is concerned with hiding the relocation of a data set or service from a requesting component (that can be a client, another server or middleware). In the CICE

Architecture, migration transparency is handled with an inherent capability from the approach to the location transparency. The HTTP environment, along with the use of the WRS registries, allow data and server providers to migrate to different physical locations in a manner that is transparent by simply updating a registry entry with a new location of the data or service provider.

2.6.5 Relocation Transparency

Relocation transparency in concerned with the hiding of the dynamic relocation of an interface from other interfaces bound to it. This type of dynamic relocation of currently bound interfaces remains a difficult research and engineering problem that is not currently addressed in the CICE Architecture. A non-dynamic relocation of an interface is effectively a migration of a service that is handles by the Migration Transparency.

2.6.6 Replication Transparency

Replication transparency is concerned with concealing the behaviour associated with the replication of objects. In the CICE Architecture, replication transparency is handled in several ways. First, any of the data or service providers can be fully replicated in the OSF and any client can choose which copy to access. Also, these replicated servers can be registered in one or more of the registries on the network. Again, with HTTP being the consistent protocol for all OGC interface implantations, these replicated servers are transparent to any invoking process whether it be a client or another server.

2.6.7 Persistence Transparency

Persistence transparency in concerned with concealing the deactivation and reactivation of objects in the RM-ODP. In CICE, the concept of object deactivation and reactivation is not particularly germane when viewed in the object-oriented framework that is the focus of the RM-ODP. In the CICE architecture, geospatial information persistence is maintained by the individual OGC compliant servers. Persistence transparency in CICE is achieved through the use of a geospatially oriented adaptation of XML Schema representation called the Geography Markup Language (GML). GML provides a variety of objects for describing geography; including features, coordinate reference systems, geometry, topology, time, units of measure and generalized values. GML is the encoding mechanism that is used by the OGC interface specifications to provide a consistent representation of geospatial information that specifically makes the underlying persistence of data and services in the service providers transparent to the requesting clients.

2.6.8 Transaction Transparency

Transaction transparency in concerned with hiding problems of coordination between the activities of groups of data and service providers. The use of the Internet platform and HTTP make the transaction transparency inherent in the architecture of CICE. All transactions in the OSF are passed over HTTP using XML based data formats across all of the OGC compliant servers and are consistent and simultaneous requests can be issued across the

network. In addition, some implementations of Web Map Servers have the ability to cascade (access) other WMSs on the network and have the results rendered as if the result came from one WMS. This capability makes these cascaded transactions, used to create this combined map image, completely transparent to the end user.

2.7 Information Security Infrastructure

The information security infrastructure for CICE is a heterogeneous collection of a variety of implementations of information security communities. These communities arise from the multiple organizations at different levels of jurisdiction (government) that are an inherent part of the CICE architecture. The organizations at each of these levels have their own laws or policies to which they adhere. These laws and policies which yield differences in the information security approaches of the organizations. This heterogeneity exists not only between different levels of jurisdiction, but also within the organizations at the same level of justification. For example, at the Federal level, there exist organizations that are required to share their information by law, such as USGS, while others have varying degrees of strength in their information security approach such as Census, Department of Defense and the Department of Homeland Security. At the state and local level, the approaches to information security are also widely varied dependent on the differing laws and policies within each state or locality.

As a further complicating factor, all levels of justification that deal with critical infrastructure protection will most likely be required to interact with the private sector. As an example of this, in the United States 85% of the critical infrastructure in the United States is owned by private industry, so any governmental plan for critical infrastructure protection must interact significantly with the private sector. In the private sector, however, companies have their own approaches to information security that additional include concerns about intellectual property rights and protecting information that helps the companies maintain a competitive advantage.

The reminder of this section addresses the complexities of information security in the multidimensional network of organizations in the CICE by first discussing approaches to information security. This is followed by a discussion of information security communities and how those communities can be structured to address information security interoperability from an engineering viewpoint.

2.7.1 Approaches to Information Security

The following sections describe a variety of approaches to information security that are applicable to the architecture and operations of CICE nodes. A community of CICE nodes will generally have an information security policy that dictates one (or possibly more) of these approaches as their information security posture.

2.7.1.1 Open Access

Clearly the most straightforward approach to information security is to provide open or public access to what ever information a CICE node would deem appropriate. This type of open access to geospatial information would be analogous to a public website. Do note, though, that open access does not mean there is no information security. As with any public website, any CICE node that provides public access must ensure that the access is provided to only that information (or those services) that the owner of the node desires to be openly accessible. All of the appropriate discretionary access controls (described in Section 2.7.1.3) are required, along with the appropriate website navigational controls, to ensure that the open access does not violate the integrity of the information on the open site.

2.7.1.2 User Login Authentication

The first level of information security access control uses an identification and authentication that requires users to employ a login authentication. This form is what is most generally used to control access to information across the Web. This typically consists of a user being assigned a user identification code (username) by a system administrator and an initial password. An alternative to this is a user may be allowed to provide the user identification code, if the security policy of the CICE allows. In this case, the user specified identification code must be verified as being a unique identifier for the CICE node. In either case, the user must provide a user specified password that is associated with the user identification code. The user identification codes and the associated passwords are stored in a database on the CICE node. The security policies that are being enforced on that system determine the method with which the user ID/password database will be protected from unauthorized access.

Before a user can gain access to OGC Web Services on a CICE node that have chosen to restrict the access using login authentication, the user will be prompted to supply a user identification code and the associated password. This information is validated against the database that contains that valid IDs and passwords. If the supplied user identification code and password are valid, access is then provided to the services for the remainder of the online session. Once the user has gained access via the user login authentication, the CICE node that is being accessed may provide further local restrictions on both services and data using the discretionary Access Controls that are described in the next section.

2.7.1.3 Discretionary Access Control

Discretionary Access Controls, while hidden behind the service interface, do require some public infrastructure. Access Control Modules (ACM) are the conceptual software modules responsible for enforcing the DAC security policy. The central idea of DAC is that the owner of an object, who is usually its creator, has discretionary authority over who else can access that object. DAC, in other words, involves owner-based administration of access rights. It should be noted that individuals that have system level privileges can override the owner-based provided access controls. These access controls generally grant, or deny, read, write, execute and delete privileges to classes of users, typically at the directory and file levels. In addition, a DAC implementation may include Access Control Lists (ACL) where the

privileges are granted or denied using lists of individual users, again at the directory and file level.

A DAC security policy requires a vocabulary of user credentials (privileges and roles), a vocabulary of resource security attributes, and the rules using those vocabularies to execute the security policy. The CAC Information Security Working Group has the responsibility to develop these vocabularies. Resource security attributes can be implemented within a component so a public infrastructure is not needed. User Credentials, however, come from outside of the component interface. A trusted means of providing this information to the ACM is needed. For purposes of CICE, those credentials will reside on the node hosting the protected service. Future iterations of the CICE architecture may stand up a separate Security Profile Service to handle user credentials.

2.7.1.4 Distributed Access Control Services

Distributed Access Control Services allow a *jurisdiction* to share web services with users within the same or with other *jurisdictions* in a controlled way. A *jurisdiction* is an administrative entity that has the ability to:

1. Authenticate its users;
2. Provide web services; or
3. Both authenticate and provide web services.

An organization may correspond to a single DACS *jurisdiction*; alternatively, each department, lab, or workstation within an organization may be a separate DACS *jurisdiction*. Each *jurisdiction* runs a DACS server, associated with a web server, that is the *jurisdiction's* initial point of contact for users and other *jurisdictions* that request access to web services that are under the auspices of its DACS.

DACS provides infrastructure and support for *Single Sign-On* and access control to jurisdictional web services. Having been successfully authenticated by one *jurisdiction*, a user may then access web services at any jurisdiction within an *information community*, subject to restrictions on those resources established by their administrators. The information flow of the DACS environment is shown in Figure 11.

Interacting with existing authentication systems, DACS provides a uniform way for users to identify and authenticate themselves. DACS aims to support virtually any authentication method that can be implemented as a web service. By indirectly invoking existing authentication services, DACS avoids the significant administrative task of creating and maintaining a user account for the same user at multiple jurisdictions. Users can be authenticated in exactly the same way as they already are being authenticated by their jurisdiction, while introducing custom authentication procedures used only in conjunction with DACS. For users, a *Single Sign-On* eases the burden of having to remember multiple account names and passwords.

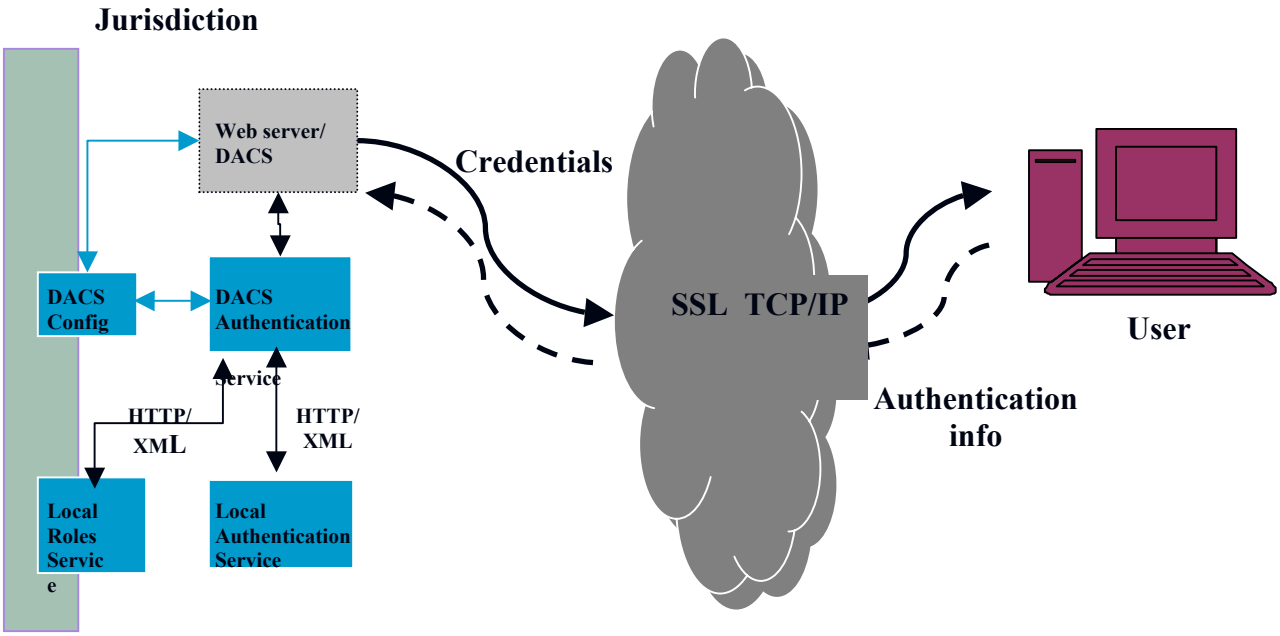

Figure 11 - Distributed Access Control Services Information Flow

2.7.1.5 Public Key Infrastructure

A Public Key Infrastructure (PKI) is a security framework in which PKI-enabled applications can be deployed and trusted. A PKI-enabled application is one that includes code to interface PKI products that implement encryption and digital signatures processing, and to access and interface PKI infrastructure elements such as certificate authorities that issue and verify the authenticity of a digital certificate.

2.7.1.5.1 Registration Authority

Authentication and Identification within the CICE can be accomplished through Public Key cryptology. Public Key cryptology uses a private key (known only to the user) and a public key known to everyone. Anything encrypted with a private key could only have been encrypted by the owner of that private key. Likewise, anything encrypted with the public key can only be decrypted with the private key (i.e. the owner). Public Key Infrastructures use these public/private key pairs to perform Identification and Authentication. Who then assures that the owner of a private key is who they claim to be?

Registration Authorities are responsible for assuring that the owner of a private key is who they claim to be. In most cases, the Registration Authority is the body that issues the key pairs. The degree of assurance a private key has depends on how rigorous the Registration Authority is in verifying the identity of a key requestor.

2.7.1.5.2 Certificate Authority

Certificate Authorities (CA) are responsible for providing public keys for individuals and assuring the association between that individual and the key. CAs will usually be set up based on an individual's organization. FEMA, for example, can be expected to set up a CA. The local security policy determines which CAs to trust. A user certificate coming from an "untrusted" CA can be rejected.

2.7.1.5.3 Security Profile Manager

Due to the large population of potential users of data hosted on a CICE node, security policies must be based on the user's role and credentials, not on their ID. Yet the role and credentials data must be available to the service in a trusted fashion. A Security Profile Manager (SPM) is the authoritative source for this information. Each emergency responder will have a "home" SPM where all of the information about their clearances, roles and authorizations will reside. Access to these directories will be through an Lightweight Directory Access Protocol (LDAP) interface. This interface will be protected with authentication and encryption. Access to the credentials data for a single individual will require a valid certificate for that individual. These directories will be maintained by the same organizations that investigate and assign the credentials to that individual.

2.7.1.6 Secure Socket Layer

Digital certificates encrypt data using Secure Sockets Layer (SSL) technology is the industry-standard method for protecting web communications. The SSL security protocol provides data encryption, server authentication, message integrity, and optional client authentication for a TCP/IP connection. Because SSL is built into all major browsers and web servers, simply installing a digital certificate turns on their SSL capabilities.

SSL provide three capabilities:

- SSL authenticates that the server you've connected to is the one it purports to be.
- SSL creates a secure communication channel by encrypting all communication between the user and the server.
- SSL conducts a cryptographic word count to ensure data integrity between the server and the user. The word count or checksum provides a count of the number of bytes in a document and ensures the exact number of bytes is transmitted and received. With SSL, even this checksum is encrypted so it cannot be modified. If a message is not received in its entirety, it is rejected and another copy of the message is sent automatically.

For the CICE architecture, the certification exchanges between client and server can be handled by the SSL protocol. SSL is the most widely used protocol for implementing cryptography on the Web. As shown in Figure 12, SSL provides a secure enhancement to the standard TCP/IP sockets protocol used for Internet communications. SSL is added between

the transport layer and the application layer in the TCP/IP protocol stack. HTTP is the application most commonly used with SSL.

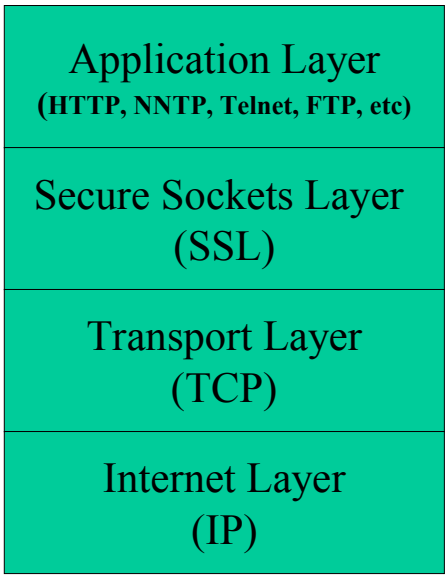

Figure 12 – TCP/IP Protocol Stack with SSL

2.7.2 Information Security for Interoperable Communities

The Enterprise viewpoint defines two types of critical infrastructure communities: Domain and Federation Communities. A Domain Community is one, which is made up of organizations and individuals that have a common interest around a set of functional activities, a local geography, or responsibilities for similar operations. A Federation Community is a larger grouping of Domain Communities and contains two or more members from Domain Communities. They could also consist of groupings of different sectors into multi-sector Federations or they could be multinational Federations or regional groupings of countries. Relating these communities to the discussion in section 2.3, a Domain Community would equal to a federal government agency, an organization with a focused interest using a defined set of functional activities. A Federation Community would equate to a group of agencies that are required to interact with one another in order to accomplish their missions. Figure 13 shows an overall view of the CICE security architecture that provides both protected and open access to services and data.

Each one of these Domain Communities will develop a security policy that is appropriate for their organization and it will be implemented and enforced using one or more of the

approaches presented in section 2.7.1. It should be made clear that these variant approaches to information security do not necessarily interoperate with one another.

This presents the CICE architecture with an engineering challenge. How can different information security communities share information among them if they are enforcing different security policies using different implementation approaches? For the CICE nodes that desire (or are required) to interact and share data and services, there is a requirement that they coordinate their information security policies to enable the appropriate access controls. Most of the information security approaches that where discussed in section 2.7.1 are pretty generally applied to applications on the Internet. The DACS, however, is being worked in a series of OGC initiatives and is particularly focused on distributed access control mechanisms that support the OpenGIS Web Services that are the foundation of the CICE architecture.

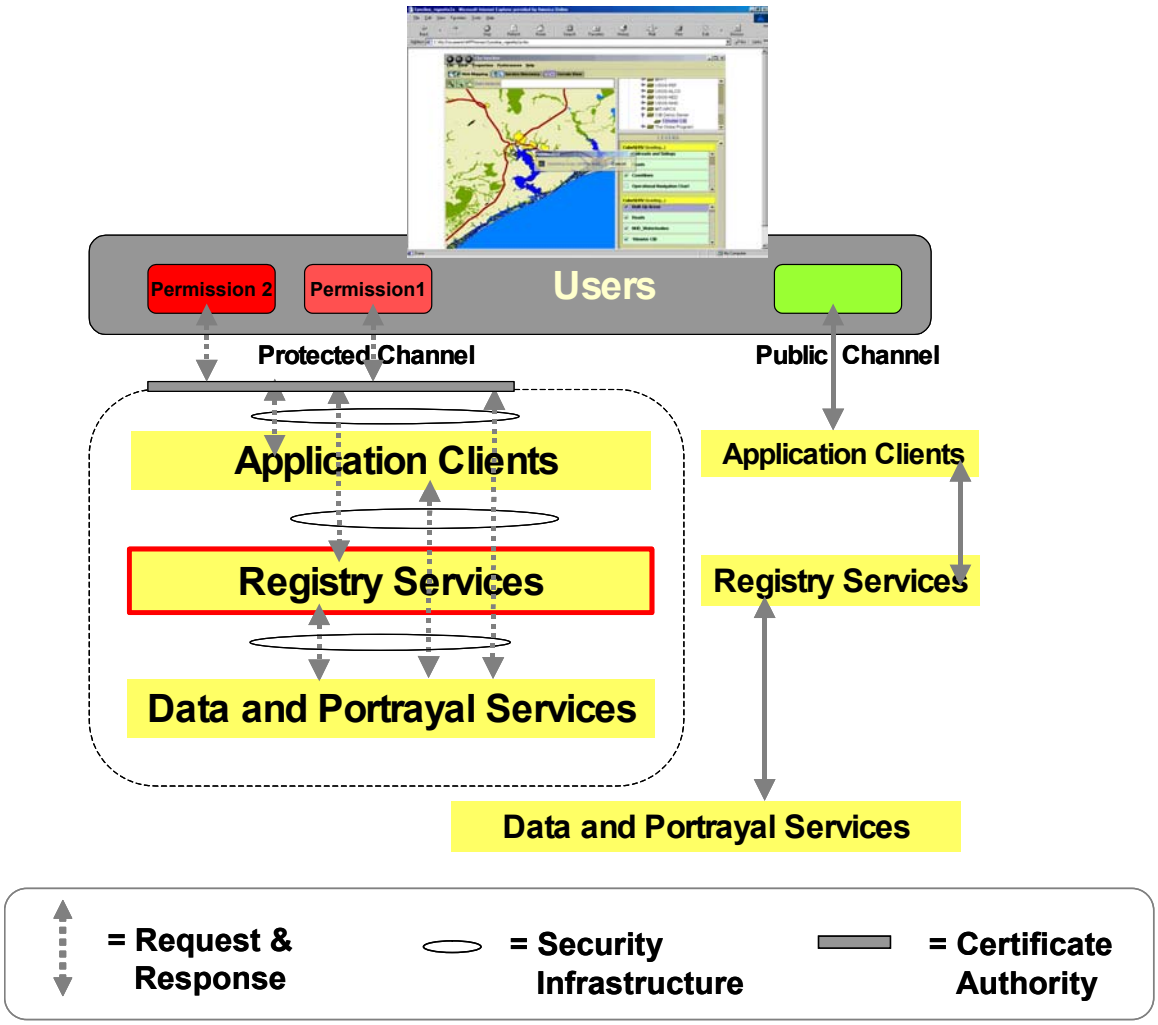

Figure 13 – CICE Security Architecture