MedBiquitous
Software Architecture Document

Version 0.3 (Draft)

Revision History

Date	Version	Description	Author
01/11/2003	0.2	Published to Yahoo group for comment.	Simon Johnston
03/11/2003	0.3	Updated after first review with TA Group.	Simon Johnston

Table of Contents

Software Architecture Document

1. Introduction

This document provides a high level overview of the evolving technical architecture for the MedBiquitous Consortium. It outlines the technologies that MedBiquitous members will use for broad collaboration and participation in a distributed network for professional medicine. In facilitating interoperability through standards, MedBiquitous will help its members enhance physician expertise, promote professional collaboration, and raise the level of patient care.

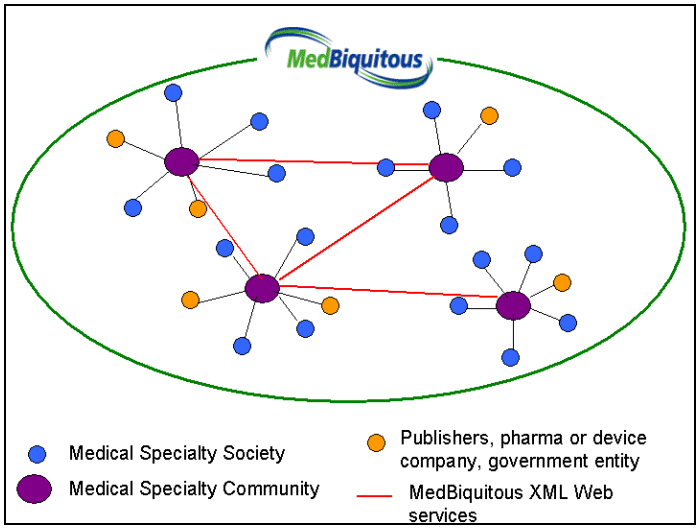
The document provides a high-level description of the goals of the architecture, the use cases support by the system and architectural styles and components that have been selected to best achieve the use cases. This framework then allows for the development of the design criteria and documents that define the technical and domain standards in detail. It is these detailed design documents that will guide the development of the actual MedBiquitous content in terms of messages and services.

By analogy the architecture of a building has to take into account the use of the building, what are the people living/working in it expecting and then has to define the size, shape, structure and so forth. The architecture has a set of guiding principles as well as known criteria and constraints that shape the proposed architecture. The designers then have to develop detailed specifications not only for the selection of materials but the placement of wiring, plumbing, lighting and so forth. Finally the building is finished and populated in accordance with the vision outlined prior to the first pen touching the draftsmen's paper.

1.1 Purpose

This architecture builds on the work of other important standards groups, including the World Wide Web Consortium, Oasis, HL7, and the Advanced Distributed Learning Initiative.

With the ability to integrate data and applications through Web services, many opportunities will arise for greater collaboration among organizations in professional medicine. Some organizations within a specialty may choose to collaborate particularly closely, forming a shared online community. MedBiquitous technologies would facilitate this type of collaboration. MedBiquitous technologies will also facilitate interactions of a single organization with partner organizations or communities. Taken together, the MedBiquitous standards for interoperability will create a common electronic network for professional medicine. The diagram below illustrates this network.



[Note: need to add notions of a single freestanding organization and other freestanding business partners.]

1.2 References

- Community Portal Use Cases, Valerie Smothers, August 2002
- Vision for Technology Development, Valerie Smothers, January 2003.
- Architectural Blueprints – the "4+1" View Model of Software Architecture, Philippe Kruchten, IEEE Software November 1995.
- The Unified Modeling Language Specification v1.4, The Object Management Group, 2003.

1.3 Glossary

ICE The Information and Content Exchange (ICE) Protocol, http://www.w3.org/TR/NOTE-ice

SOAP Simple Object Access Protocol, http://www.w3.org/TR/SOAP/

UDDI Universal Description, Discovery and Integration, http://www.uddi.org

URL Uniform Resource Locator, http://www.w3.org/Addressing/rfc1738.txt

WSDL Web Services Definition Language, http://www.w3.org/TR/wsdl

WSIL Web Services Inspection Language, http://www-106.ibm.com/developerworks/webservices/library/ws-wsilspec.html

WSRP Web Services for Remote Portals, http://www.oasis-open.org/committees/wsrp/

2. Architecture Overview

MedBiquitous is developing an XML framework for professional medicine that includes both XML payload standards and XML Web Services standards. Whenever possible, we embrace existing industry standards that have significant traction. In some cases, we customize existing standards to meet the needs of professional medicine. We are focused on pragmatic technology development that serves the needs of our members.

The MedBiquitous Consortium Technical Architecture (TA) is represented using a UML [OMG 2003] model at a high level of abstraction that allows us to visualize, understand and reason about the architecturally significant elements and identify areas of risk that require more detailed elaboration. This document is a way of communicating the UML model in context, to present the information in a structured fashion and to discuss areas of the model.

The technical architecture is decomposed along the following dimensions:

1. Architectural constraints: known technical decisions that are independent of the use cases, i.e. choice of a certain implementation technology to facilitate interoperability.
2. System functionality: Represented by use cases
3. Design layers separating four kinds of concerns:

    a. Domain concerns that focus on the key abstractions representing information common to, and agreed upon by, the community.

    b. Service concerns that focus on interfaces and services are developed that will implement key functionality in a live system.

    c. Portal concerns that focus on the discovery, aggregation and presentation of the community information to users as well as security, membership, personalization and ownership of information.

d. Deployment concerns that focus on the constraints imposed on the architecture by certain deployment considerations.

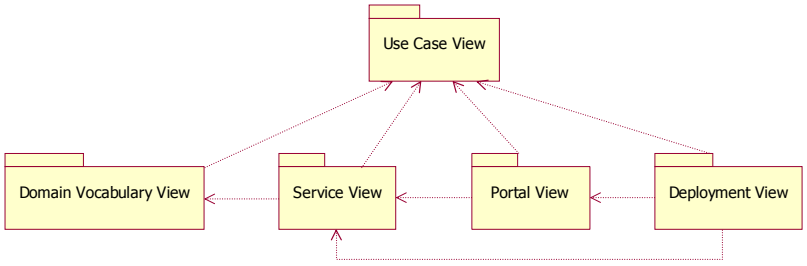
This concerns will be discussed and elaborated further to present an overview, all be it at a high level, of the technical architecture defined by the MedBiquitous Consortium and therefore implemented by member and affiliated organizations.

2.1 Architectural Representation

The architecture of the reference application is represented following the recommendations of the Rational Unified Process (RUP). The UML specification of the system has been divided into the following five views:

- **Use-Case View**: Describes the actors and use cases for the system, this view presents the needs of the user and is elaborated further at the design level to describe discrete flows and constraints in more detail. This domain vocabulary is independent of any processing model or representational syntax (i.e. XML).
- **Domain Vocabulary View**: Describes the key abstractions that make up the domain of discourse. For example the notions of "Member" or "Scientific Journal" are included in the MedBiquitous domain whereas "Engine", "Body", "Number of Wheels" are not (though may be included in a truck manufacturing domain).
- **Service View**: Describes the high-level components that make up the dynamic aspects of the system. A fundamental constraint of this architecture is that it follows a Service-Oriented architectural style as opposed to a distributed object broker style for example.
- **Portal View**: Describes the behavior of the portal infrastructure in aggregating information and presenting it to the user. Key mechanisms, such as information acquisition, query, aggregation, syndication etc. are described here. As far as the user is concerned this is the primary client interface into the MedBiquitous information infrastructure.
- **Deployment View**: Describes potential deployment structures, by including known and anticipated deployment scenarios in the architecture we allow the implementers to make certain assumptions on network performance, system interaction and so forth.

Collectively, the above models form a complete UML specification of the system (the dashed arrows in the diagram below denote a dependency relationship from one model element to another).



Note, this document does not discuss an implementation view, the purpose of the MedBiquitous Consortium is to develop standardized domain schema, message schema and service definitions not to deliver working implementations.

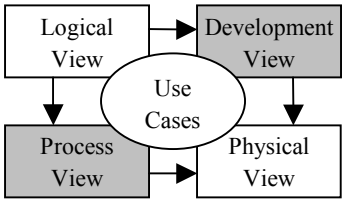
However; it is important that we clearly delineate the definition of the domain concerns from the service concerns.

[1] The integration layer is not represented explicitly in the J2EE implementation of the Online Auction. This is because we are using the CMP (Container Managed Persistency) for storing system data in a database

This will become clearer in the elaboration below, but at the heart we need to separate the common, domain knowledge from the technical infrastructure and constraints on implementation. Although we have decided that the realization of this architecture will be based on web services there is always the possibility that in the future such technical infrastructures change – the definition of a member organization or a scientific journal does not change at the same pace. We will also see that by separating these concerns in this way we can handle technical details such as transportation, security and privacy concerns in a clean fashion.

2.2 Underlying Architectural Framework

This architecture follows the "4+1" framework [Kruchten 1995] that defines a set of "Views" of the architecture.



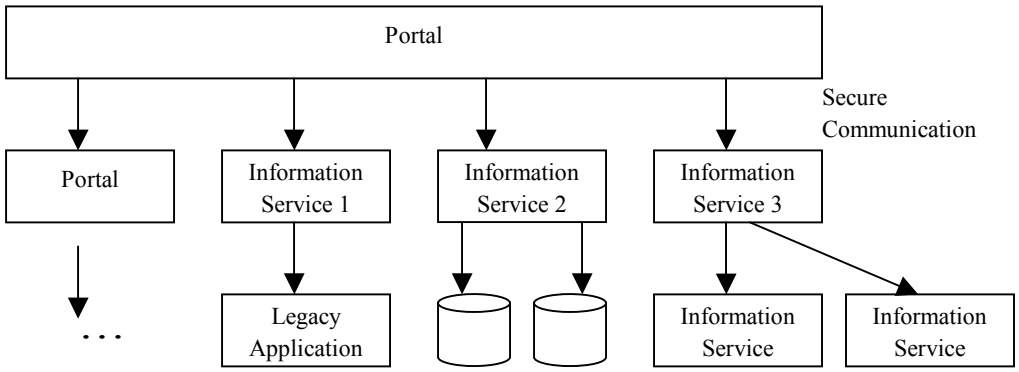
The rest of this document is organized to present the architecture using this framework.

4+1 View	Architecture Concerns
Use Cases	Use Case View
Logical View	Domain Vocabulary and Service Views
Process View	Not Applicable – MedBiquitous does not at this time specify processes or specific scenarios required by providers.
Development View	Not Applicable – MedBiquitous does not mandate any development method, tool or technology.
Physical View	Portal and Deployment Views

3. Architectural Goals and Constraints

Key networking protocol standards such as TCP/IP and HTTP and the markup language HTML have driven the explosive growth of the Web. For the next phase of growth, industries can use XML to enable the sharing of data across multiple systems and XML Web Services to enable rich integration of distributed applications. These technologies will streamline the development process for creating a highly interactive and interoperable suite of software tools for communities of medical professionals.

With this in mind we may envision an end-result as seen from a human user similar to the diagram below, alternate views are possible depending on the role of the user; producer, consumer etc.

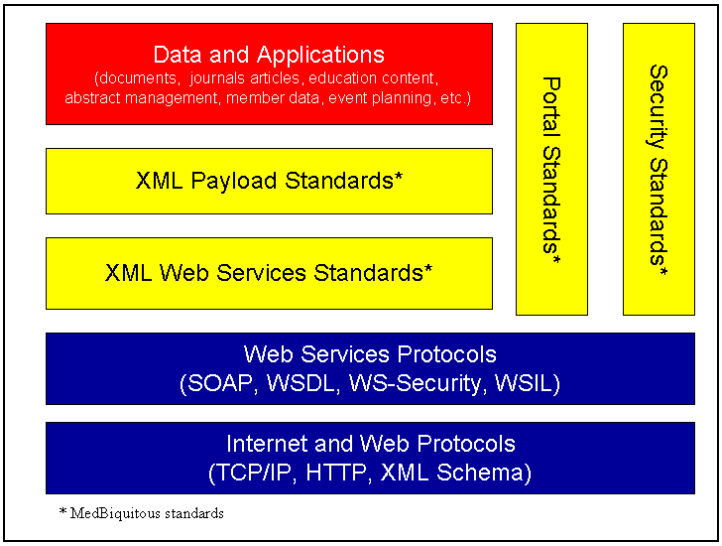


The diagram denotes how the services implemented according to the MedBiquitous standards can be utilized by the Portal infrastructure. We see 3 different implementation styles for the information service in the example above:

- Style #1. This information service interfaces to an existing application and presents that applications information to the community according to MedBiquitous technical standards.
- Style #2. This is a service that bypasses the existing application (due perhaps to an inability to interface through the application itself) and serves information directly from local repositories.
- Style #3. This is an aggregation service, it communicates to two or more downstream information services and aggregates content according to some predefined rules.
- We also see the ability for one portal to access and present information from another portal, although the ICE content syndication standards are more likely than direct screen-scraping technology.

3.1 Standards

The MedBiquitous consortium has chosen a set of technology standards that support the implementation of a federated information network. These standards are those that underpin the Internet as well as in the emerging area of Web Services. These standards were chosen because of their open and platform neutral manner thus allowing an open and flexible choice for the implementer. The following diagram shows how the MedBiquitous technology standards (indicated by *) build upon one another and upon this existing infrastructure of Internet technologies.



Note that the above diagram is a snapshot of the current state of the technology; detailed design documents in each

area will outline the specific versions of standards to be used in each area.

MedBiquitous standards provide the means to integrate applications and data across the Internet, enabling rich collaboration among many organizations in professional medicine. MedBiquitous development efforts build upon existing Internet protocols, Web protocols, and emerging Web services standards and include the following technologies:

- **XML Payload Standards** for the data and documents produced by medical communities and their business partners.
- **XML Web Services Standards** to enable the delivery and integration of applications for medical communities.
- **Portal Standards** for integrating content and applications from various sources into a cohesive and usable presentation for physicians.
- **Security Standards** and guidelines for exchanging authentication and access data among disparate systems and organizations.

3.2 Syndication

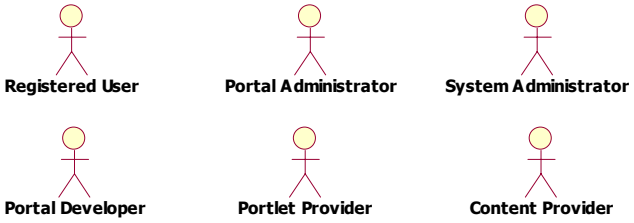
[Discussion of RSS, ICE]

4. Use-Case View

This view presents the users perception of the functionality provided by portal implementations of the MedBiquitous technical standards. These use cases were synthesized from [Smothers 2002] but do not include all descriptive text. Note that this section provides the context for, and scope of the rest of the document. Putting aside overriding architectural constraints outlined above all further development (in terms of design and content documentation) will be in support of one or more of the following use cases.

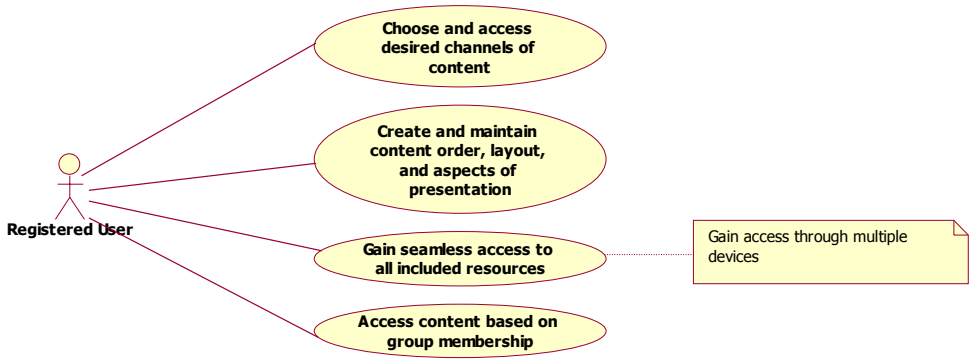
4.1 Actors

The following is the list of known actors that will interact with the deployed system.

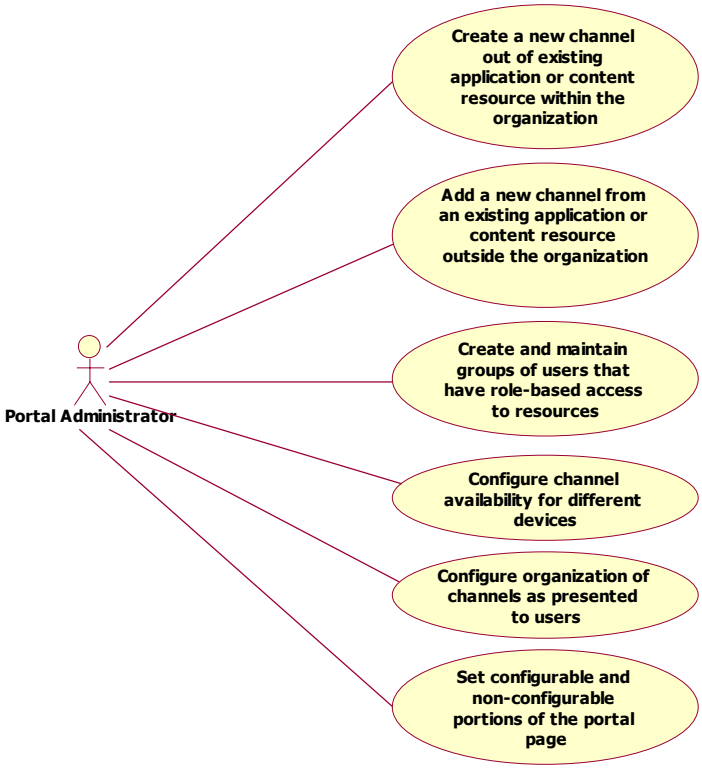


4.2 Use Cases

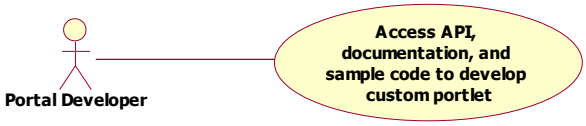
4.2.1 Registered User



4.2.2 Portal Administrator Use Cases



4.2.3 Portal Developer Use Cases



5. Domain Vocabulary View

This section describes the architecturally significant elements of domain vocabulary; it is not intended to be a complete information model. Note also that wherever possible the MedBiquitous consortium will reuse existing

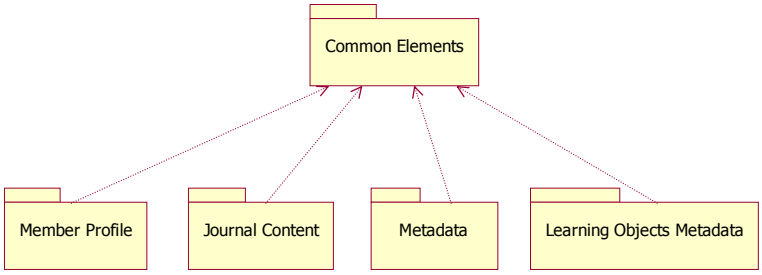
content standards rather than develop their own.

5.1 Overview

The Domain Vocabulary View contains the following top-level packages:

- Member Profile
- Metadata
- Learning Objects Metadata
- Journal Content

The following diagram shows the relationships between the packages in the Domain Vocabulary View, and their dependency on the Common Elements package:



Details of the XML schemas for the domain vocabulary can be found in the MedBiquitous XML Repository at http://www.medbiq.org/repository.

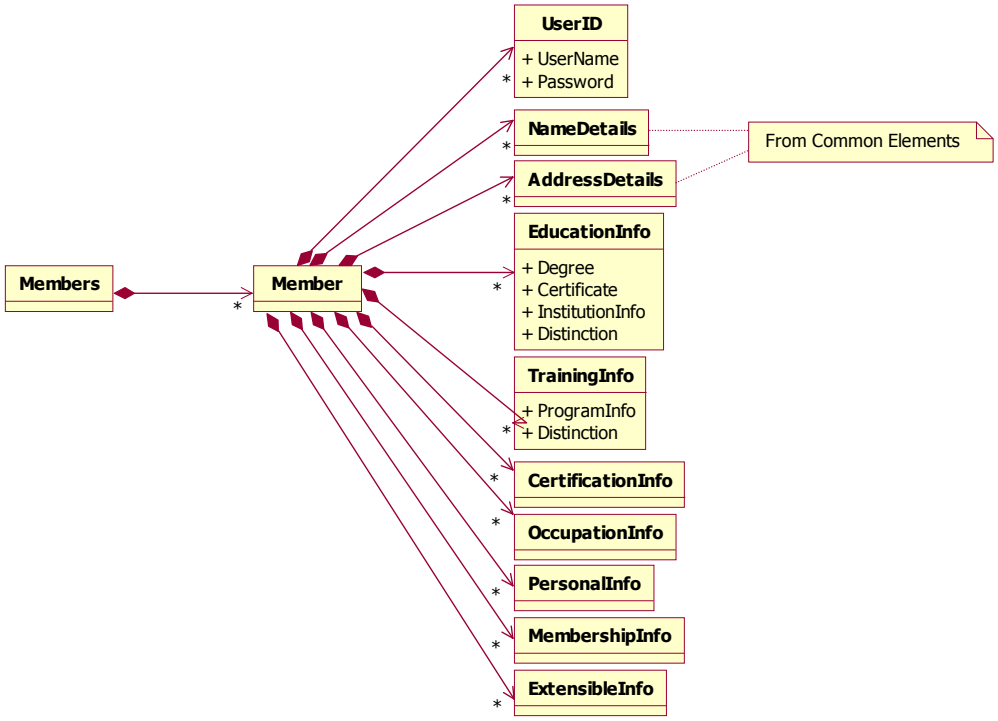
5.2 Common Elements

TBD

5.3 Member Profile

Provides a consistent way of defining society membership information for exchange with other entities.

Used By: Medical communities, societies, and their business partners.



5.4 Metadata

Provides a standard for describing general content to facilitate search, retrieval and content syndication.

Used By: Medical communities, societies, and search services

5.5 Learning Objects Metadata

Provides a standard for describing modules of medical education to facilitate search, retrieval, and delivery of education to other systems.

Used By: Medical communities, societies, CME providers pharmaceutical and device companies, content developers

5.6 Journal Content

Provides a set of standards for journal articles, tables, and other journal content to facilitate the exchange of this information.

Used By: Publishers, content aggregators, libraries

6. Service View

A critical activity for MedBiquitous is to create Web Services Description Language (WSDL) standards for many of

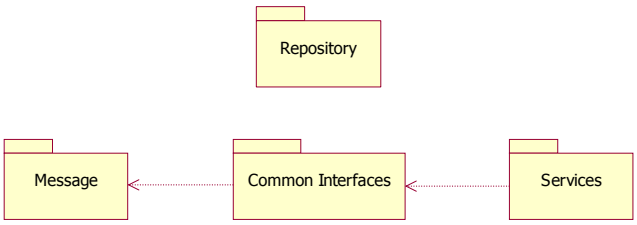
[2] Such components have also been called assembly components and application server components.

[3] Component services are the operations specified in the component's interface.

[4] These enterprise component service realizations describe the part of the use-case realization performed by the enterprise component.

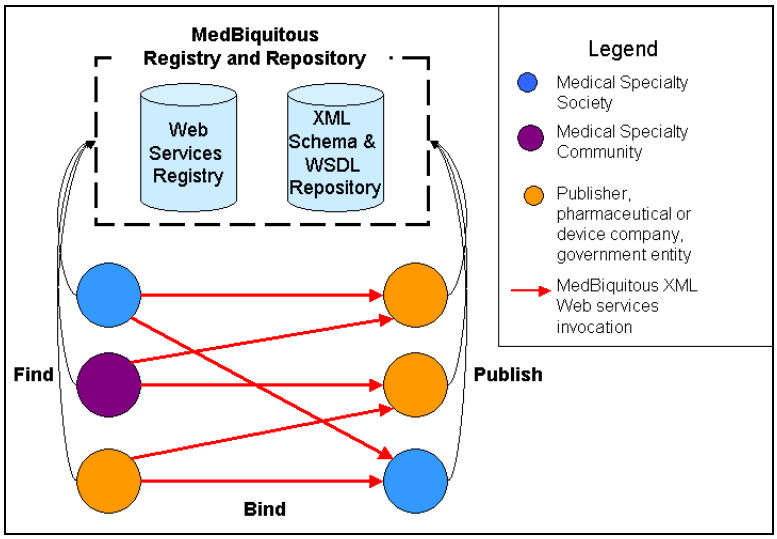
[5] We use the term "mechanism" here to describe a composition of a set of implementation strategies for selected patterns.

the important Web services that are needed by medical professionals. This section will outline the known service requirements, focusing on specifying only the services and interfaces required, not on the details of operations or message content.



6.1 Repository

Web services enable applications on the Web to connect to other applications on the web in an automated fashion. These connected applications can share information and work together as if they were parts of a single application. MedBiquitous will maintain a registry of Web services and a repository of XML schema and Web service Description Language (WSDL) files to enable developers to implement Web services. With the information contained in the registry, the application in need of a service can bind to the application providing the service and issue a command. The Service Provider then performs the desired task for the Service Requester. The following diagram illustrates this process. MedBiquitous Communities, publishers, pharmaceutical and device companies, government entities and software vendors, all map as a Service Provider in the Web services domain.



The MedBiquitous Repository will make available four types of information:

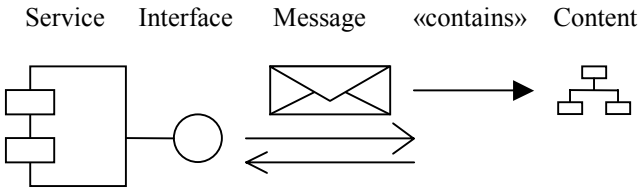
- **Information XML Schemas.** These define the MedBiquitous-standardized XML documents that represent the domain elements identified by the vocabulary experts.
- **Message XML Schemas**. These wrap elements from the Information Schemas to form the payload of Web services SOAP messages.
- **Web Service Interface Specifications.** This is a searchable collection of MedBiquitous-standardized WSDL documents that abstractly define the service interface, including the types, message, and port types.
- **Web Services Implementation Descriptions.** This collection of WSDL documents defines specific implementations of MedBiquitous Web services by consortium members. It refers to a specific interface specification and contains the information, such as the URL of the service, needed to invoke the service.

The MedBiquitous Registry will initially be provided via Web Services Inspection Language (WSIL) documents

that can be easily searched for services that meet the developer's needs. As usage of Web services evolves within online medical communities, this registry can evolve into a more sophisticated implementation based on a private UDDI Registry.

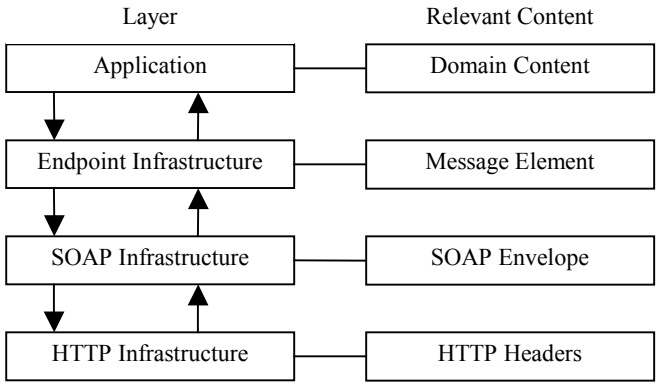
6.2 Message Structure

As we mentioned in section 3 above it is important to separate out the structures that define a message passed in or out of a service action and the domain content that is used as the content of the message. The following diagram demonstrates this separation.



This allows us to separate information about the "instance" of a message from the content in the message, so for example the same content might be sent to two different end points but have different associated message-level information included. This separation also allows the definition of properties that are to be consumed by the MedBiquitous aware end-point but not necessarily by the end-user of consumer of the content.

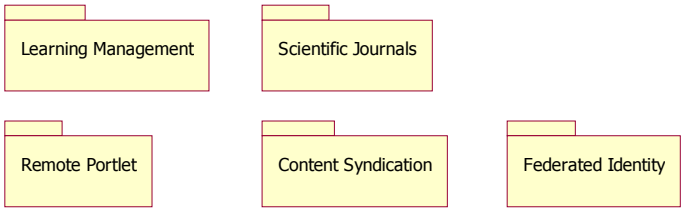
This separation also allows us to adhere to a common layered approach to the architecture that clearly defines the content relevant to each layer in a simple model as shown below.



6.3 Common Interfaces

TBD

6.4 Services



6.4.1 Learning Management

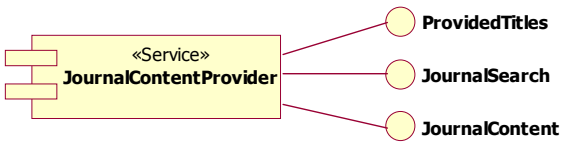
Provide and receive modules of medical education . Send and receive CME completion information.

Used By: Medical communities, societies, specialty boards, licensing boards, healthcare IT systems

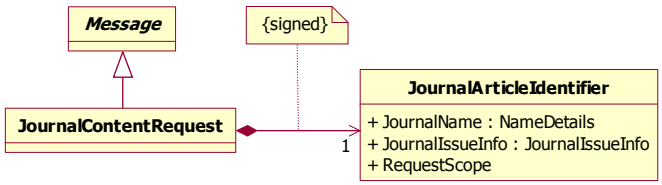
6.4.2 Scientific Journals

Provide and receive online delivery of journal for integration into existing websites. Search and retrieve journal data. Send and receive subscriber information.

Used By: Journal publisher, content aggregator, medical organizations



The diagram above demonstrates how to represent in the UML a service that exposes a set of interfaces. By separating the notion of a service which is an actual software entity that exists at some identified end point from the interfaces which are a specification of some set of behavior we gain the flexibility to reuse common interfaces across a number of services. In this way an interface can be defined as a stand-alone entity, versioned and managed independently from any service that implements it (in UML terms the service realizes the interface).



The diagram above demonstrates how the abstract "Message" structure is used, that the "JournalContentRequest" message is a specialization that includes an element from the Journal Content Domain.

As an example, given SOAP 1.2 and a simple UML→ XML serialization we could expect to see the following request message in XML. Also note that the constraint {signed} in the model results in the generation of an XML Digital Signature.

```
<env:Envelope xmlns:env="http://www.w3.org/2002/12/soap-envelope">
  <env:Header>
    ...
  </env:Header>
  <env:Body>
    <m:JournalContentRequest xmlns:m="http://medbiq.org/schema/jrnlcnt">
      <m:JournalArticleIdentifier>
        <m:JournalName>. . .</m:JournalName>
        <m:JournalIssueInfo>. . .</m:JournalIssueInfo>
        <m:RequestScope>. . .</m:RequestScope>
      </m:JournalArticleIdentifier>
      <sig:Signature xmlns:sig="http://www.w3.org/2000/02/xmldsig#">
      </sig:Signature>
    </m:JournalContentRequest>
  </env:Body>
</env:Envelope>
```

6.4.3 Content Syndication

Request, send, and receive syndicated content.

Used By: Medical communities, societies, content providers, government health organizations (alerting)

6.4.4 Remote Portlet

Integrate applications from one organization's system into the website portal for another organization.

Used By: Medical communities, societies, pharmaceutical and device companies, service providers

6.4.5 Federated Identity

Link user accounts to enable single sign on across multiple web resources

Used By: Medical communities, societies, publishers, content aggregators, CME providers, content providers

7. Portal View

The medical communities that serve as information hubs for physicians need to weave together services and information from a variety of internal and external sources to serve the needs of their members. Individual physicians will want to further customize their own information views for personal relevance. For these reasons, a key part of the MedBiquitous technical architecture is portal software, which enables the creation of a common entry point for delivering aggregated and integrated information and resources.

In the MedBiquitous network, these portals will enable integration of internal and external applications, data, and content. To achieve economies of scale, organizations may choose to structure interactions with outside business partners or other communities as Web services that plug into a portal framework.

Portals typically include a suite of core applications that provide valuable services for members and organizations. There is great modularity and loose coupling of applications to the portal when used with a Web services design. For non-core applications, the Web services design is optimal, taking over for what now is done in an ASP model. The portal engine makes use of Web services proxy clients to dynamically fetch and assemble data and content from remote service providers. Web service endpoints can also be exposed externally for other portals, thus a portal can be a service provider as well as a service requester.

7.1 Portlets

Portlets are channel-like Web components designed to be aggregated in the context of a composite page. Typically, to create a personal portal page (like My Yahoo), a single page request to the portal engine invokes multiple portlets, each of which produces a fragment of markup for the final page.

Recently, the major portal software vendors have begun working within OASIS to develop a standard for portlets. Called Web Services for Remote Portals (WSRP), this standard will define a pluggable, user-facing, interactive Web services with a common, well-defined interface and protocol for processing user interactions and providing presentation fragments suitable for aggregation by portals. WSRP-compliant portlets will be able to run on all WSRP-compliant portals without requiring any service specific adapters; a single, generic adapter on the portal side will be sufficient to integrate any WSRP service. WSRP will standardize the presentation layer of these Web services. The WSRP interfaces are defined in the WSDL and include a specification for metadata for self-description and publishing in WSRP registries. If the OASIS WSRP standard is successful, it could be en enormously valuable technology for MedBiquitous members.

8. Deployment View

TBD