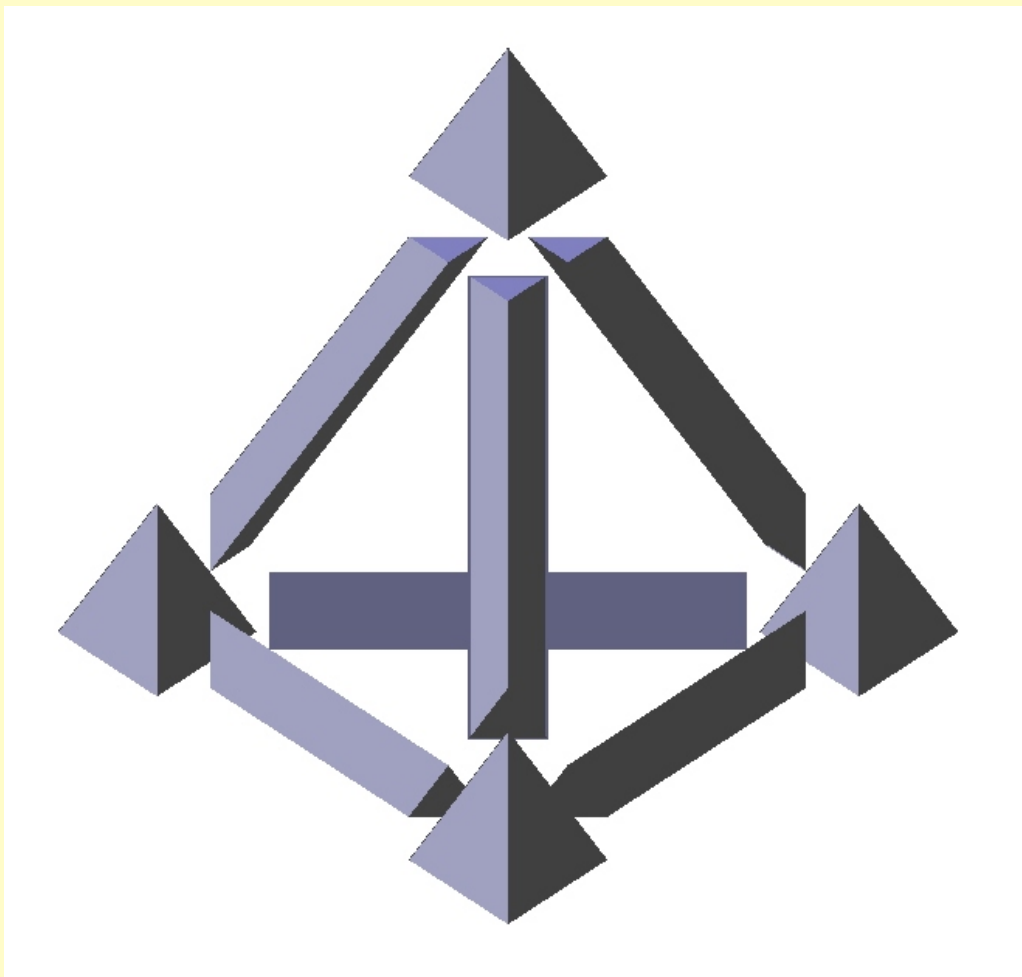


Knowledge Services for Distributed Service Integration



Geodise

[Http://www.geodise.org](http://www.geodise.org)

L Chen¹, Professor SJ Cox², Professor C Goble³, Professor AJ Keane²,
A Roberts³, Professor NR Shadbolt¹, Dr P Smart⁴, F Tao¹

¹ Department of Electronics and Computer Science, University of Southampton

² School of Engineering Sciences, University of Southampton

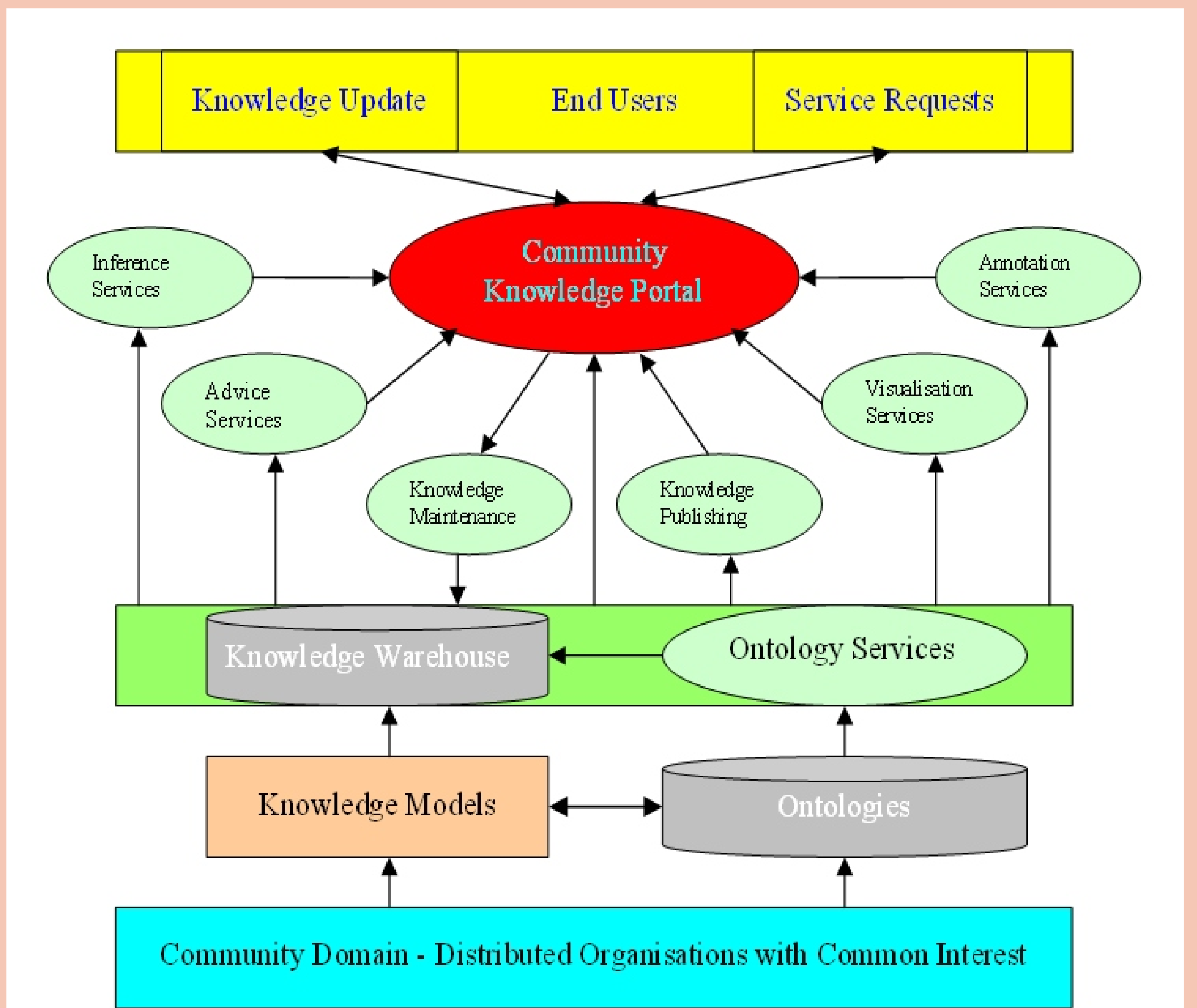
³ Department of Computer Science, University of Manchester

⁴ Epistemics Ltd., Strelley Hall, Nottingham

While computing increasingly addresses collaboration, sharing and interaction with the powerful support of the emerging distributed computing infrastructures such as web services and Grid technologies, there is a growing demand for ontology and knowledge technologies to provide semantic support for service integration, the sharing and coordinated use of knowledge across distributed, heterogeneous, dynamic virtual organisations. We have developed and partially implemented an architecture to provide knowledge services. It has been applied to Grid-enabled design search and optimisation (Geodise).

Distinguishing features of the architecture:

- Tackling the six challenges of knowledge management life cycle in an integrated framework.
- A layered modular structure with each component dealing with a specific aspect of knowledge engineering process.
- Supporting the exploitation of different techniques and tools. Each of them can be updated while others kept intact.
- Using ontologies to generate machine-interpretable semantically-enriched content and also knowledge bases.
- A knowledge portal provides mechanisms for knowledge maintenance and resource control.
- Service-oriented approach - separating domain knowledge from operational knowledge.
- Easy knowledge reuse and sharing through web service technology over the Internet.
- Flexible - adding knowledge any time through portal.
- Extensible - plugging in new functions as knowledge services.
- Robust - adopting new techniques/tools while keeping the system functioning.



Knowledge acquisition and modelling lay the foundation for the knowledge service architecture, which produces ontologies and the types of template and structure where knowledge can be held. It has been carried out by the CommonKADS knowledge engineering methodology and the PC PACK knowledge management integrated tools.

Concept Ladder

- optimization method
 - stochastic optimization method
 - evolutionary programming method
 - genetic algorithm method (GA)
 - population based incremental learning (PBIL)
 - simulated annealing method
 - design of experiments method
 - random exploration with charge
 - relative random search method (ADRANS)
 - heuristic optimization method
 - exhaustive testing
 - linear optimization method

Diagram Tool

- optimization method
 - single-variable minimization method
 - stochastic optimization method
 - heuristic optimization method
 - linear optimization method
 - exhaustive testing
 - slope-based optimization method

Protocol Editor

5.2. Choice of Optimizer

Here, as has already been noted, the initial input data for the design do not allow all the criteria used as constraints to be met (i.e., it is not a feasible design). This restricts the choice of optimizer since some do not work when given an initially infeasible design. However, limits have also already been chosen for the design variables, and the optimizers will therefore seek to remain within these limits, reducing the scope of the task somewhat (some of the methods within the OPTIVAR library treat these range constraints as advisory by default and may, therefore, explore beyond their limits; this may be over-ridden if desired). To give a representative cross-section of the methods available the following techniques have been applied:

- (1) the method of successive linear approximation -APPROX;
- (2) the genetic algorithm -GA, with a line pass, external penalty function -OPTIM1 and
- (3) the Hooke and Jeeves direct search -SEEK, with the Blotto-McCormick combined external and internal penalty function -OPTIM2

Knowledge representation and publishing is to represent knowledge in a well-structured, well-indexed form and make them ready for sharing and use. Geodise domain knowledge has been represented in CommonKADS task models, concept hierarchy and a design workflow and published in the form of knowledge web and XML format.

Design Workflow

Geodise Domain Models

Knowledge Web in XML

Knowledge Web

Knowledge Publication

Ontologies serve as the conceptual backbone for knowledge sharing and management in the above integrated architecture for knowledge services. We have developed ontologies for design search and optimisation using both Protégé and OilEd ontology tools.

Protégé Ontology Concept Instances

OilEd Ontology in DAML

Protégé Ontology Concept Hierarchy

Protégé Ontology in HTML Format

- The aim of knowledge portal is:
- To make knowledge available and accessible
 - To provide tools for knowledge reuse and exchange
 - To provide security infrastructure
 - To manage knowledge resource
 - To support online forum and maintain mailing lists
 - To disseminate the latest advance of the domain

Current functions of knowledge portal include:

- Security mechanism
- Knowledge publication
- Service registration
- Knowledge retrieval and update
- Service application information
- Resource management via version control

Knowledge Portal Browser-based Interface

Knowledge Warehouse:

- Design Workflow
- Domain Knowledge
- Domain Ontology
- Optimisation Knowledge Base
- User Profile Ontology

Knowledge Services:

- Annotation Service
- Ontology Service
- Reasoning & Inference Engine

Security Control

Existing Community User

UserID:

Password:

Login as an anonymous user will limit your access to some knowledge resources.

OS Tester - Microsoft Internet Explorer

Ontology Service CGI Interface

SOAP-based Ontology Service WSDL

Ontology Services

Ontology Service Presentation in Portal

Annotation Process

Metadata Added after Annotation

Ontology service provides a Java API giving full access to any DAML+OIL ontology that is available over the Internet. The API is exposed through both a SOAP-based web service and a CGI interface for common ontological operations, such as subsumption checking, navigating concept hierarchy and retrieving concept and attributes.

Annotation is necessary to add semantic content to document or website, thus facilitating information sharing, reuse and automatic machine processing. We have used OntoMat- Annotizer annotation tool in Geodise to annotate the workflow for particular design problems and then save them in a knowledge base. The semantically enriched archive can then be queried, indexed and reused later to guide future designs.