OpenKnowledge Annual Report

www.openk.org

Our aim is to develop a new form of knowledge sharing that is based not on direct sharing of "true" statements about the world but, instead, is based on sharing descriptions of interactions. By making interaction specifications the currency of knowledge sharing, we gain a context to interpreting knowledge that can be transmitted between peers. The narrower notion of semantic commitment we thus obtain requires peers only to commit to meanings of terms for the purposes and duration of the interactions in which they appear. This lightweight semantics allows networks of interaction to be formed between peers using comparatively simple means of tackling the perennial issues of query routing, service composition and ontology matching.

Summary of Activities

After the first year of the project we have an integrative architecture and an implemented kernel system, supplemented by verification methods and demonstrator applications. This is, to our knowledge, the first single system that shares interaction models in a peer to peer style and uses these to coordinate peers in an opportunistic but reliable way. This is a radical departure from the mainstream in terms of the underlying methods of coordination between peers but it can accommodate mainstream practices. For example, in our interaction modelling language (LCC) we can interpret traditional business process modelling languages (such as BPEL4WS); in our interactions we can conscript existing Web services via standard interfaces (such as WSDL); and we can obtain adaptive behaviours such as ontology matching and mediation using dynamic modification of the interaction model's context.

The project is currently at the end of the first phase of its design cycle. A first version of the kernel system is built and in the coming year it will be extended, tested and applied to a number of applications (primarily in bioinformatics and emergency response although others are emerging).

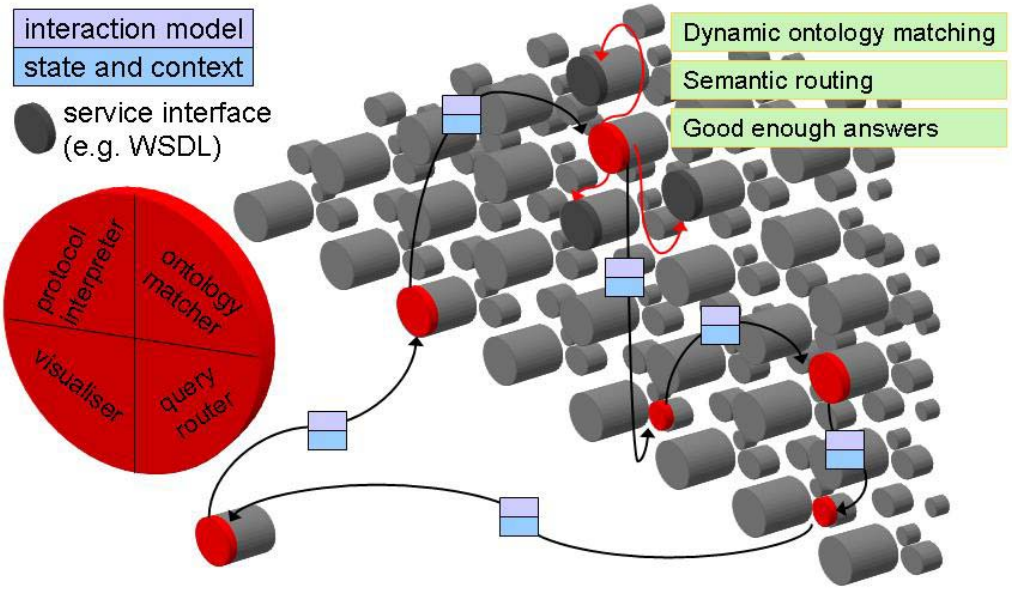
Lightweight Coordination Calculus

A model of interaction is shared between peers and used computationally by each peer to control its own behaviour and communicate to other relevant peers the behaviours expected of them for this interaction. The language we have developed for this purpose is the Lightweight Coordination Calculus (LCC). Interaction models in LCC are declarative, executable specifications that may be understood independently of any particular peer or execution model and run in many different ways. The conditions under which they are run, however, require more run-time support than a traditional executable specification language (see sections below).

Peer to Peer Kernel

A first version of the OpenKnowledge kernel has been built. This is a piece of software that can be downloaded onto a peer and, when run, automatically connects to other available peers to enable sharing of LCC models and data related to their execution. The picture below shows the main elements of the kernel and the way it is used to control interaction between peers. Grey canisters in the picture represent computational

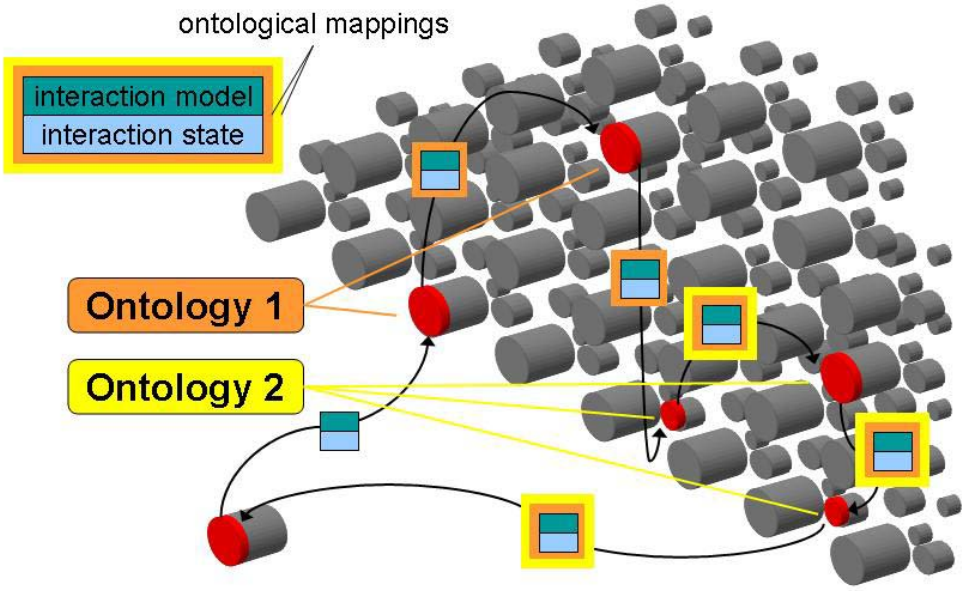
processes on the Internet: potential OpenKnowledge peers. The red circles represent instances of the OpenKnowledge kernel. Some processes will download the kernel and can operate as OpenKnowledge peers. Other processes only offer more traditional interfaces (such as WSDL interfaces for Web services). These more traditional services can be conscripted into interactions via traditional calls from OpenKnowledge peers (in the same way that one Web service can call another). This allows the OpenKnowledge peers system to be "bootstrapped" using existing Web services. OpenKnowledge peers, however can share and run LCC interaction models, with the kernel taking care of the mechanics of this so that individual peers do not need to be aware that a peer to peer infrastructure is being used. To enable this to happen, inside the OpenKnowledge kernel are a number of components: an interpreter for LCC; an ontology matcher (see below); a query router; and mechanisms for visualisation.



The OpenKnowledge kernel provides a rich context for interactions between peers, thanks to the LCC model that is used to facilitate coordination. We shall be using this context to provide facilities necessary for large scale knowledge exchange: dynamic ontology matching (where terminology does not match perfectly between peers); semantic routing (where the issue is to route appropriate interactions between peers); and good enough answers (recognising when the quality of an interaction is sufficient for the task in hand).

Dynamic Ontology Mapping

OpenKnowledge peers do not assume pre-engineered mappings between ontologies (as is traditionally the case in automated knowledge sharing systems. Instead, when there are mismatches between the ontologies of peers we attempt to correct these during an interaction. The picture below describes the intuition involved, where an interaction model is routed through a series of peers that use two different ontologies. As it reaches each peer , the ontology of the peer is mapped where necessary to terms in the interaction model – thus accumulating ontology mappings as context to the specific interaction.

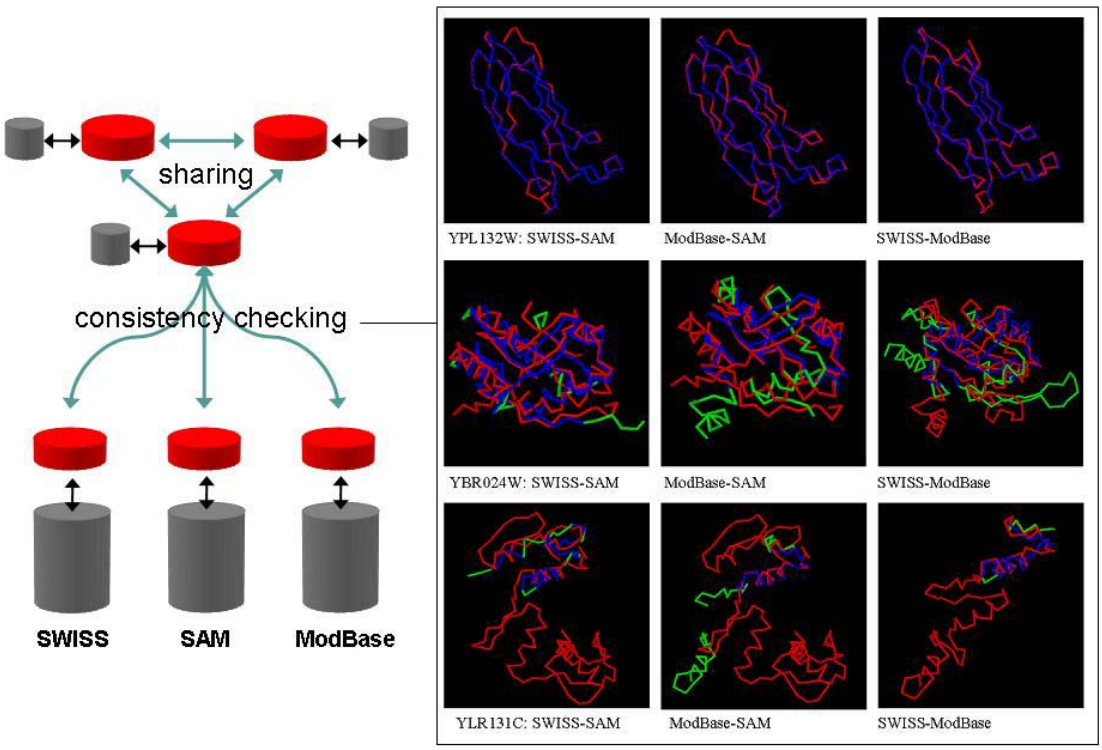


Although this might at first sight appear more difficult then traditional ontology matching, several properties of interaction models make it simpler. Most importantly, the mappings used for a specific interaction are not required to hold in general – only for that one particular interaction – so we do not need to discover generic mappings between contexts. Secondly, peers can maintain a rich database of associations between previous (successful and unsuccessful) interactions and ontology mappings, so accuracy may be improved via (shared) experience.

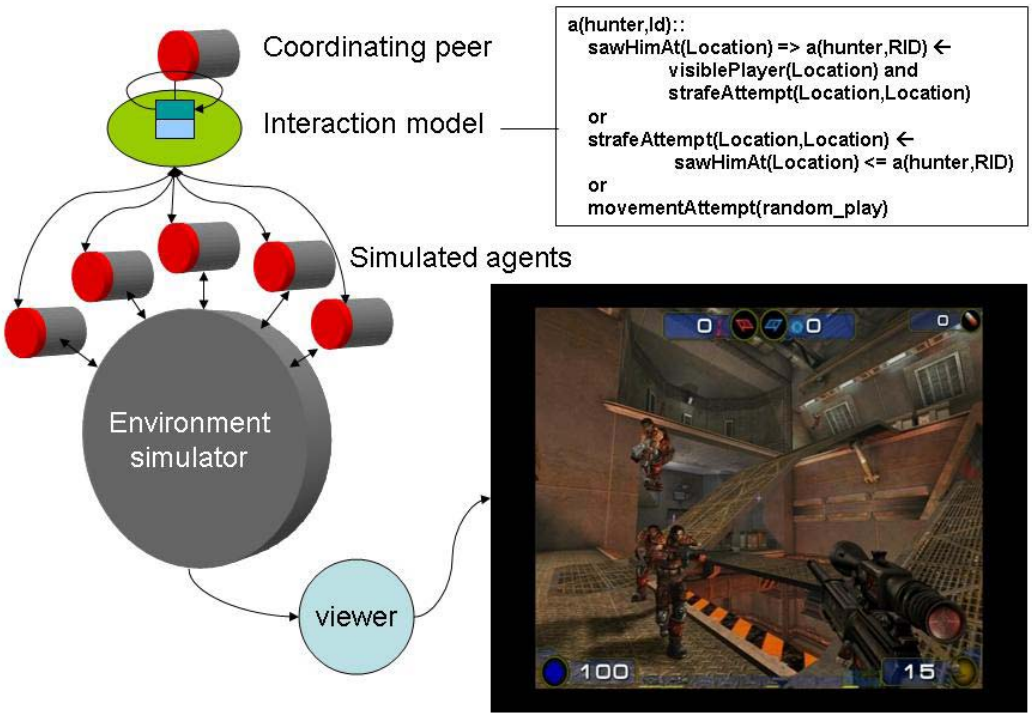
Application Testbeds

We are developing applications in numerous domains, focusing on two contrasting areas: bioinformatics and emergency collaboration. These two foci are chosen because of their contrasting natures. Bioinformatics often requires complex but stable interactions within peer groups that value high quality repeatable results. Emergency collaboration takes place in more chaotic situations where the environment has a strong influence on the nature and viability of interaction.

An example bioinformatics application is in proteomics, where a common part of the analysis of protein function is to search for similar proteins from centrally curated databases. A problem is that established databases are not necessarily consistent with each other in terms of the protein structures they predict. The picture below shows, on the right, some comparisons obtained from three commonly used sources of Yeast protein data. On the left is the peer architecture used to “lift” data from the data bases into the peer to peer system, where it can then be curated by individual experimenters (acting as local quality filters) and shared.



The emergency response community does not have the investment in Web services that one finds in the bioinformatics community, so our applications here concentrate on engineering more traditionally curated data for peer to peer coordination and on simulation of chaotic environments. The picture below shows the architecture of a simulation system that uses the Unreal Tournament game engine to provide a controlled but chaotic environment in which to study the effectiveness of LCC models in coordinating agents in real time. The picture below shows the overall architecture of the simulation framework, with a view taken from one of the experiments.



User Involvement, Promotion and Awareness

The project has published its early technical results in the major international conferences in its area. For example, our work on an interaction modelling language is published in Autonomous Agents and Multi-Agent Systems 2007; our work on model checking for interaction verification is published in the International Joint Conference on Artificial Intelligence 2007; and we have presented papers at several workshops of the International Semantic Web Conference. We have also begun to publish peer reviewed results in our application communities – a paper on our protein structure comparison results will appear in the Grid Computation and Computational Biology workshop of the International Conference on Molecular Biology. For details of these and other papers see the project Web site (www.openk.org).

We are in the process of developing links to other groups who may use the technology we are developing in order to further their own research. Our aim in doing this is to expand beyond the testbeds directly associated with the project itself, providing a novel technology to those who want to use it in new applications. Our first promising new direction is in medicine where we are investigating the possibility that OpenKnowledge technology might support medical knowledge sharing and protocol deployment in the UK and in China.

Future Work

In the coming year we have the following foci for research.

- Peer to peer interaction model routing: further analysing and adopting the available methods for sharing information about peers and interaction models.
- Dynamic ontology mapping: we possess a repertoire of methods for mapping ontologies in interaction contexts – the next step is to select and integrate these.
- Good enough answers: we shall develop methods for estimating the quality of answers obtained during interactions, using peer groups as sources of shared experience.
- Trust metrics: we shall develop methods by which one might assess the trust placed by a peer in an interaction.
- User-centred applications: we shall explore how the OpenKnowledge approach can be integrated with established, user-centred applications such as Magpie and Aqualog.
- Testbeds: we shall extend our current work in bioinformatics and in emergency coordination, and will explore “outreach” efforts in areas such as medicine.
- Testing and verification: we shall extend our tools for simulation and verification to allow us to explore more properties of interactions – particularly those related to peer group formation/stability and scaling.

Each research focus above feeds back to the system kernel, which will become generally available, both as a downloadable system (so you can become an OpenKnowledge peer) and as open source software (so you can adapt and improve on the interaction management methods we have devised).