On Knowledge Grid and Grid Intelligence: A Survey

William K. Cheung and Jiming Liu
Computer Science Department
Hong Kong Baptist University
Kowloon Tong, Hong Kong
{william, jiming}@comp.hkbu.edu.hk

Abstract

The next generation Web Intelligence (WI) aims at enabling users to go beyond the existing on-line information search and knowledge queries functionalities and to gain, from the Web,[1] practical wisdom for problem solving. To support such a Wisdom Web, we envision that a grid-like computing infrastructure with intelligent service agencies is needed, where these agencies can interact, self-organize, learn, and evolve their course of actions, identities and interrelationships for new knowledge creation, as well as scientific and social evolution. In this paper, we first provide an overview of recent development in WI and Semantic/Knowledge Grid. Then, the fundamantal capabilities of the Wisdom Web as well as the conceptual architecture of an intelligent Grid for supporting it are described. Technical challenges for realizing Grid Intelligence are highlighted and recent advancements in related research areas are reviewed.

Keywords: Wisdom Web, Grid Intelligence, Knowledge Grid, Autonomy Oriented Computing

1 Introduction

1.1 Web Intelligence and Wisdom Web

The Web has irrevocably revolutionized the world we live in. This impact is inevitable due to the facts that the Web connectivity rapidly increases and that the online information astronomically explodes. In order not only to live with such a change but also to benefit from the information infrastructure that the Web has empowered, we have witnessed the fast development as well as applications of many Web Intelligence (WI) techniques and technologies [Zhong et al., 2003], which cover:

1. **Internet-level communication, infrastructure, and security protocols.** The Web is regarded as a computer-networked system. WI techniques for this level include, for instance, Web data prefetchng

[1] Here the notion of "Web" should be taken in a broader sense.

systems built upon Web surfing patterns to resolve the issue of Web latency. The intelligence of the Web prefetching comes from adaptive learning based on observations of user surfing behavior.

2. **Interface-level multimedia presentation standards.** The Web is regarded as an interface for human-Internet interaction. WI techniques for this level are used to develop intelligent Web interfaces in which the capabilities of adaptive cross-language processing, personalized multimedia representation, and multimodel data processing are required.

3. **Knowledge-level information processing and management tools.** The Web is regarded as a distributed data/knowledge base. We need to develop semantic markup languages to represent the semantic contents of the Web available in machine-understandable formats for agent based computing, such as searching, aggregation, classification, filtering, managing, mining, and discovery on the Web [Berners-Lee et al., 2001].

4. **Application-level ubiquitous computing and social intelligence environments.** The Web is regarded as a basis for establishing social networks that contain communities for establishing social networks that contain communities of people (or organizations or other social entities) connected by social relationships, such as friendship, coworking or information exchange with common interests. They are Web-supported social networks or virtual communities. The study of WI concerns the important issues central to social network intelligence (social intelligence for short) [Kumar et al., 2002, Huberman et al., 1997]. Furthermore, the multimedia contents on the Web are not only accessible from stationary platforms, but increasingly from mobile platforms [Weiser, 1998]. Ubiquitous Web access and computing from various wireless devices need adaptive personalization for which WI techniques are used to construct models of user interests by inferring from user behaviour and actions [Alesso and Smith, 2002, Billsus et al., 2002].

Figure 1 shows the evolution of WI, starting from the **Web** as a standardized on-line platform, to **Web Based Information Systems** driven by its integration with databases, and then to **Web Based Intelligent Systems** where the role of Artificial Intelligence (AI) is to enable Web based systems with various Web-specific analysis/mining tools for more effective information search and knowledge queries.

We envision that the next paradigm shift in WI should go beyond the role of supporting tools and further incorporate the notion of *wisdom* to be embedded in the Web itself [Liu, 2003]. According to Webster Dictionary, the word *wisdom* means *the quality of being wise; knowledge, and the capacity to make due use of it; knowledge of the best means and the best ends.* We refer to the World Wide Wisdom Web as the Web that can autonomously

- discover the best means and ends,
- mobilize distributed resources,

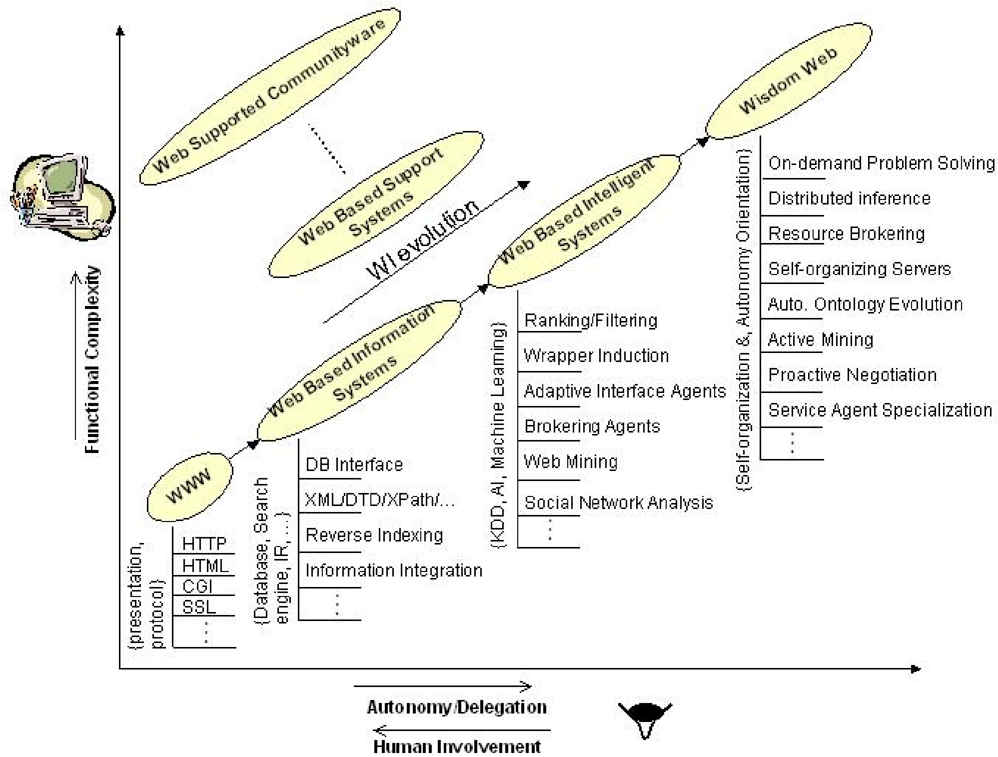

Figure 1: The evolution of Web Intelligence.

- enrich social interaction, and
- enable users to gain practical wisdom of living, working and playing.

For instance, the Wisdom Web should be able to answer queries by autonomously consulting required information, knowledge sources and other computational services. Also, the knowledge discovery and data mining services in the Wisdom Web should be active in, say, advertising/bidding the data that they are good at. A more complete list of the Wisdom Web's capabilities will be described in the subsequent sections.

The challenge of realizing the Wisdom Web urges for serious exploration of the new fundamental role of Artifical Intelligence in it from different perspectives, for instance, the social/behavoral perspective (Social Intelligence) and the collaborative computing perspective (Grid Intelligence[2]). This paper focuses on the latter one and provides a rigorous technical review on how the wisdom notion can be supported by a distributed computing infrastructure (like the Grid to be described in the next section) embedded with intelligent capabilities.

[2]See Section 1.2.1.

1.2 Semantic/Knowledge Grid

Grid computing is a *concept* about a computing infrastructure which can utilize distributed computational resources to solve problems as easy as plugging into the electric Power Grid [Foster et al., 2001]. As most of the existing computing units and environments are not primarily designed for large-scale collaborative computing, researchers working in this emerging area focus on developing (a) a standardized middleware (commonly called *The Grid*) for trust-based distributed resources sharing,[3] and (b) related enabling technologies and tools for application development on top of the Grid middleware [Foster et al., 2001].

According to the nature of the distributed resources it manages, the Grid has been evolving from *Computational Grid* [Foster et al., 2001] (concerning, e.g., job scheduling, system information services, life cycle management) to *Data Grid* [Chervenak et al., 2001] (concerning, e.g., distributed data access, meta-data managememnt, data replication), and recently to *Knowledge Grid* (concerning, e.g., knowledge-based inference, knowledge discovery and data mining) [Cannataro and Talia, 2003, Zhuge, 2004b]. e-Science is by far the most successful Grid application domain [SI-eScience, 2004], which includes scientific simulation experiments using data from remote instruments, massive data visualization, solving complex equations, etc. Recently, Grid computing has also been applied to e-Business [Foster et al., 2002, Kephart and Chess, 2003] where on-demand computing [Gaynor et al., 2004, Huang and Venkatasubramanian, 2003, Ewerth et al., 2004] (e.g., for financial simulation and data analysis) and distributed massive data access within or across enterprises [Foster et al., 2002, Sudra et al., 2000] (e.g., for analyzing customer data collected from POS systems) are required.

To facilitate resource management on the Grid, Semantic/Knowledge Grid related initiatives are numerous [SG-Portal, url, Roure et al., 2003a,b], with the main focus of applying the Semantic Web standards (including RDF, OWL, OWL-S, SWRL) [SW-Portal, url] to annotating Grid resources to support the subsequently required reasoning, for instance, resource matchmaking.

1.2.1 Motivation and Organization of the Paper

In this paper, we argue that wrapping the Grid with semantic annotations is necessary but not sufficient for realizing the Wisdom Web vision. We need a grid-like computing infrastructure that possesses autonomous behaviours at different levels of Grid abstraction (e.g., resource and knowledge levels) with capabilities like self-organization, adaptation, and coordination, forming the "pillow" for the Wisdom Web to be run on it. We refer to this newly emerging research area under WI as *Grid Intelligence*[4].

[3] Currently, the Globus toolkit is the *de facto* standard of the Grid middleware [Globus, url], which has recently been embraced by the Web services technology [SI-Middleware, 2003] for making Grid resources available as Grid services.

[4] The term *Grid Intelligence* was first coined by the authors in the First International Workshop on Knowledge Grid and Grid Intelligence in conjunction with the 2003 IEEE/WIC International Conference on Web Intelligence held in Halifax, Canada in September 2003.

The remaining of this paper is organized as follows. Section 2 provides the fundamental capabilities that the Wisdom Web should possess and the proposed conceptual architecture of an intelligent Grid for supporting it. Section 3 highlights a number of important research challenges, together with reviews of recent advancements in related research areas. Section 4 discusses some recent Knowledge Grid related projects found in the literature and Section 5 concludes the paper.

2 Wisdom Web Needs a Grid with Intelligence

In this section, an operational definition of the Wisdom Web is first provided using an on-line stock recommendation example. Then, the fundamental capabilities of the Wisdom Web are listed, which naturally lead to a conceptual architecture of a grid-like computing infrastructure possessing the capabilities of being "wise".

2.1 An Operational Definition of the Wisdom Web

Assume that a user consults the Wisdom Web for just-in-time stock recommendation via a mobile phone (This type of activities will later on be referred to as *Wisdom Search*). The following outlines the operation of the Wisdom Web in response to the request. Important concepts are highlighted using the italic face, which will further be elaborated in the subsequent section as the fundamental capabilities of the Wisdom Web.

1. Based on the user specified *goal*, the Wisdom Web should be able to identify the current context as "stock market".

2. Then, related on-line computational *services* will be *discovered on-demand* by querying distributed (e.g, peer-to-peer) service repositories. The services found should include those *semantically tagged* as *specialized* in stock data provision, data analysis, information presentation, SMS notification, etc.

3. Afterwards, those services will be *matched* further in terms of their input and output semantics, and chained up (*composed*) as a *service flow* for fulfilling the user specified goal. The composition can be accomplished by searching and modifying from *reusable flows* or it may need to start from scratch if similar flows cannot be identified.

4. *Domain-specific ontology* (e.g., about stock market), *service ontology* (describing services' characteristics) and *problem solving ontology* (describing service flows) are typically required for the aforementioned service discovery and service flow planning. Service ontology is relatively more well-defined and standardized (e.g., OWL-S) that the Wisdom Web can leverage on. For those that are hard to be standardized (e.g., the domain specific ontologies and problem solving ontology), the Wisdom Web needs to discover them, organize them, and maintain a corresponding distributed knowledge base.

5. After service flow planning, each service request (e.g., for stock information provision) within a service flow may be fulfilled by more than one provider. The Wisdom Web can *select* and *schedule* the flow execution according to the past performance of the candidate services (e.g., their response time), the just-in-time status of their committable *resources* (e.g., available bandwidth), as well as other QoS parameters. *Resource ontology* for describing the relationships between different resource concepts is needed to support robust and secure resource matchmaking and sharing.

6. Once the scheduling is finished, the flow can be executed. According to our stock recommendation example, there could be a possibility that the data provided by the chosen stock information provision service cannot be successfully fed into the pre-selected stock data analysis service, the Wisdom Web should be able to detect those unexpected failure, *self-repair* the flow in a just-in-time manner, and continue to *coordinate* all the service switchings along the *flow execution*.

7. Each flow planning and execution experience will be used as *feedback* by the Wisdom Web to update at least the local characteristics of the services, the relationship between the services, and the problem solving ontology for *self-organization* so that optimal utilization and performance can be achieved incrementally and adaptively.

All the aforementioned operational functions will be performed autonomously and seamlessly by the Wisdom Web. The user will receive an SMS message of the stock recommendation as if it is from a stock expert.

2.2 Fundamental Capabilities of the Wisdom Web

As revealed in the stock recommendation example, the Wisdom Web needs to incorporate and standardize at least the following ten fundamental capabilities [Liu, 2003]:

1. **Self-organization:** The Wisdom Web will automatically regulate the functions and cooperations of related Websites and application services available. A Wisdom Web service, later on called *Wisdom Agent*, automatically self-nominates to other services its functional roles as well as corresponding spatial or temporal constraints and operational settings.

2. **Growth/Reproduction:** The population of Wisdom Agents will dynamically change, as new agents are self-reproduced (e.g., spawning more daemons of the same type in some servers) by their parent agents in order to become more specialized, or aged agents are deactivated.

3. **Specialization/Association:** Via **self-organization** and **reproduction**, Wisdom Agents will be specialized in performing some roles in a certain composite service. The association of its role with any service will be measured and update dynamically, for instance, the association may be forgotten if it is not used for some time.

4. **Autocatalysis/Coordination:** As various roles of Wisdom Agents are created through **specialization** and activated by the Wisdom Search requests, their associations with some services and among themselves must be autocatalytically aggregated. In this respect, the autocatalysis of associations is similar to the pheromone laying for positive feedback in an ant colony [Dorigo et al., 1996].

5. **Problem Solving Markup Language (PSML):** PSML is necessary for representing the distributed services and their utilities and for enabling Wisdom Agents to associate their roles and settings as well as to derive their relationships among the services for supporting **AI planning** based problem solving.

6. **Semantics:** The Wisdom Web needs to understand the meanings of words under different contexts and what is right judgement of "best", by understanding the granularities of their corresponding subjects (for instance, profile, locaton and time) and the implications of their ontology definitions.

7. **Metaknowledge:** Besides semantic knowledge extracted and manipulated in the Wisdom Search, it is also essential for the Wisdom Web to incorporate a dynamically created source of metaknowledge (represented using **PSML**) that deals with the relationships between concepts and the spatial or temporal constraint knowledge in planning and executing services. It enables Wisdom Agents to self-resolve their conflict of interest.

8. **Personalization:** The Wisdom Web remembers the recent encounters and relates different episodes together, according to (1) user identity, (2) situation, and (3) attainability of (sub)goals. It may further identify other goals as well as courses of actions for the user as he/she interacts with the Wisdom Web.

9. **Social and Psychological Contexts:** Wisdom Agents running in different geographical regions and for different markets and clients need to determine and conform to user-acceptable social norms and psychological dimensions/factors in their services and communication. Proper use of those social and psychological contexts will make the Wisdom Agents to manifest "smarter" **personalization** and **coordination**.

10. **Planning:** The goal provided by users can be complex in the sense that it may not be solved using one agent but requires decomposing it into further sub-goals to be solved by the coordination of a community of agents, each bearing its own set of constraints. Meanwhile, the capabilities of **metaknowledge**, **personalization**, and **self-coordination** support the planning to be done on-demand. In the Wisdom Web, ontology alone will not be sufficient.

2.3 Wisdom Web Supported by Grid Intelligence

As revealed by its capabilities and operational definition, the Wisdom Web should be a "live" computing infrastructure which continuously makes use of the retrieval, planning and execution experience to self-organize its logical topology, to create metaknowledge for problem solving, and to specialize Wisdom Agents' roles for later-on task delegation [Liu et al., 2005]. Besides, the Wisdom Web is required to be an on-demand environment, which implies that efficient and adaptive service discovery, negotiation, flow planning and recovery utilizing the latest service characteristics and reusable knowledge have to be supported.

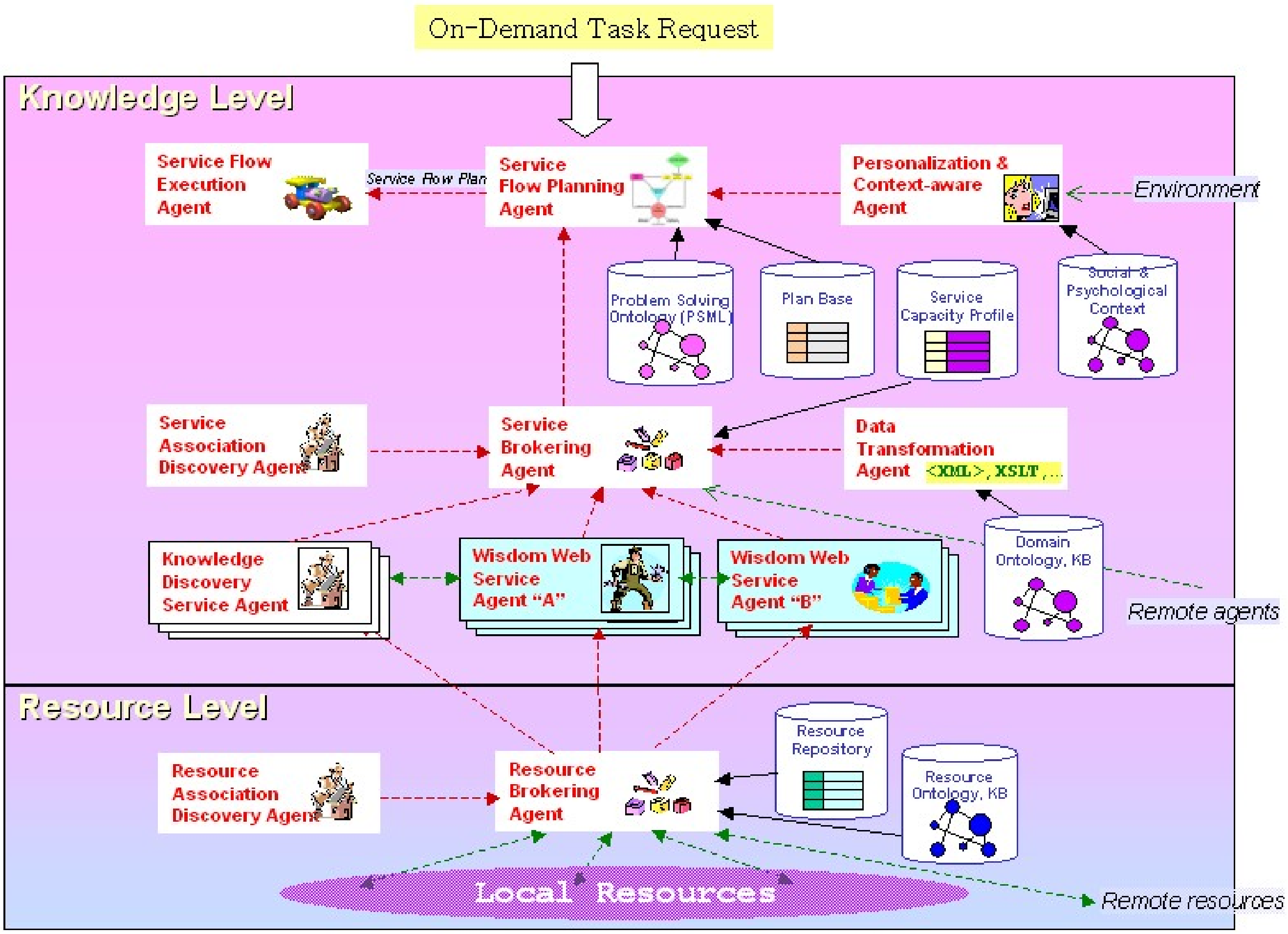

Figure 2: The Conceptual Framework of the Intelligent Grid.

Figure 2 shows the conceptual architecture of the proposed Intelligent Grid for supporting the Wisdom Web. To contrast with some related conceptual Grid framework found in the literature, e.g., [Benatallah et al., 2002, Stork, 2002], we argue that the Grid components should be semantically rich, autonomous and adaptive by themselves and due to this reason, "agents" are later on used for referring to the Grid components.

Resource/Service/Domain/Problem Solving/Sociology&Psychology Ontologies facilitate effective and smart management of different entities in the Grid, including low-level resources (e.g., computing units with 1GB memory space, data files stored in a unix system), services (e.g., a grid service that provides stock analysis results) and service flows (e.g., a set of sequenced grid services for trip planning). Based on those ontologies and additional inference rules (e.g., in SWRL), on-demand service discovery and flow planning can be supported. These semantic descriptions stored in local or remote repositories (e.g., UDDI equivalence in Globus) keep evolving and have to self-organize well to support later-on distributed discovery. Also, social and psychological contexts should play an important role in supporting process personalization (taken care by Personalization and Context-aware Agent) in the Intelligent Grid.

Resource Brokering Agent (RBA) works at the resource level for matching computation and data access requests with resources' capacities and storage contents, respectively, based on their corresponding semantic descriptions. Some of the resource characteristics are dynamic (e.g., transient workload, dynamic data caching and replication, etc.), which implies that RBA requires a protocol to maintain up-to-minute resource characterization.

Service Brokering Agent (SBA) works at the knowledge (or sometimes called service) level for matching service requests with available services (registered in service repositories) according to service capacities as well as input/output syntactic and semantic properties. By comparing with RBA, service descriptions are relatively more stable but involve more diverse domain-specific contexts.

Association Discovery Agents (ADA) is dedicated to monitor the activities of the brokering agent so as to support capacity specialization and role association of the service agents, and at the same time control the service agents' population based on their historial performance and usage.

Data Transformation Agent (DTA) includes different data transformers for transforming the output of a service to fit into the input of another. Syntactic and semantic matching supported by a set of matchmaking rules and an inference engine are typically involved. Besides, it is also responsible for matching heterogeneous ontologies retrieved from distributed knowledge sources.

Knowledge Discovery Service Agent (KDA) works at the knowledge level for providing data mining functionalities to extract knowledge from different data sources and KDA should be able to self-identify their data mining strengths with the help of ADA.

Service Flow Planning Agent (SFPA) starts from the user specified goal and composes a service flow — a directed acyclic graph of services to be requested to achieve the goal. As previously mentioned, planning from scratch, in most of the cases, is time-consuming and may not be feasible. Properly adopting the reusable flows (according to the past experience) implies that it is necessary for SFPA to work closely with (1) ADA for services to search for relevant services, (2) DTA to resolve the input/output format dismatch, (3) KDA to request for on-demand knowledge discovery services needed to compose a plan, and (4) the problem solving ontology component for flow retrieval.

Personalization and Context-aware Agent (PCA) works at the knowledge level to explore user profile and context information and incorporate them as additional hard and soft constraints to support SFPA for making the planning personalized and context-aware.

Service Flow Execution Agent (SFEA) needs to dynamically select *best* service providers for the service requests and coordinates the execution of the composed service flow. Thus, it maintains the execution state and works with SFPA for flow self-repair in case any unexpected service failure occurs during the flow execution. Besides, adaptive QoS characterization of the services is also required for smart services to stand out and poor ones to be gradually excluded in the computational economy.

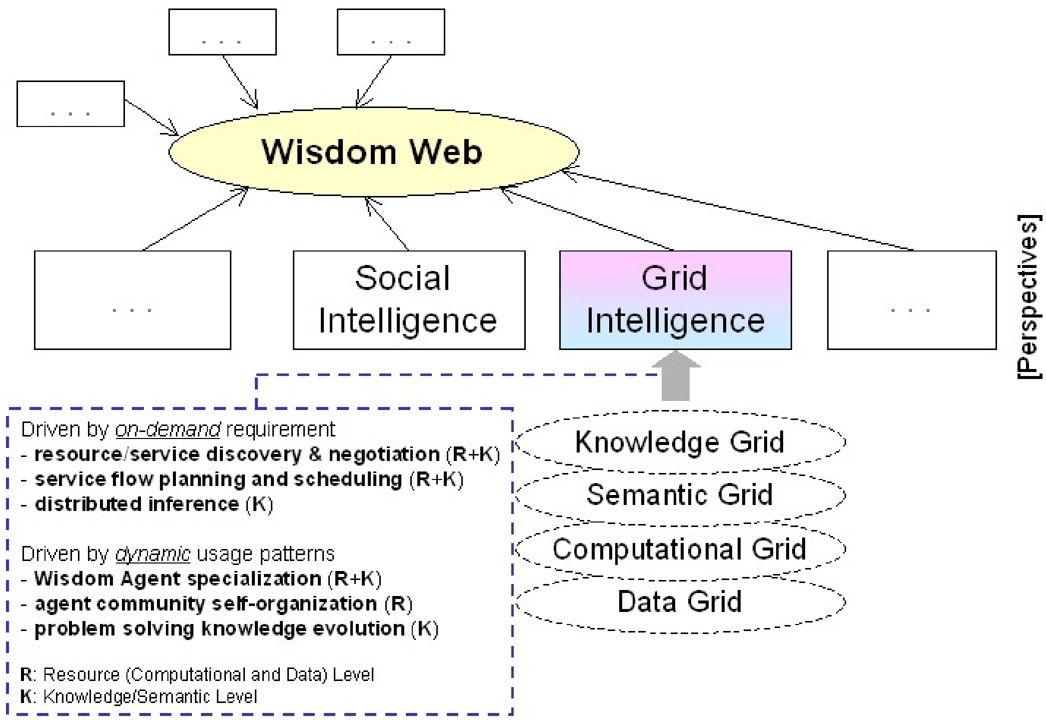

Figure 3: Grid Intelligence as a Collaborative Computing Perspective of the Wisdom Web.

3 Research Issues in Grid Intelligence

In order to implement the conceptual architecture described in the previous section, we need to address a number of important Grid Intelligence related research issues, including

- Driven by the on-demand requirement:

   1. resource/service discovery and negotiation
   2. service flow planning and scheduling

3. distributed inference

- Driven by dynamic usage and retrieval patterns:

1. Wisdom Agent specialization
2. agent community self-organization
3. problem solving knowledge evolution and reuse

3.1 Resource/Service Discovery and Negotiation

Recall that the Grid architecture described in Section 2.3 consists of two abstraction levels. The Wisdom Web discovers (searches and matches) and allows for negotiation with Wisdom Agents of different roles at both the *resource* level (based on resource semantics) and the *knowledge* level (based on service semantics) of the Intelligent Grid for on-demand collaborative problem solving.

3.1.1 Resource Matchmaking

A *resource* can be characterized [Thain et al., 2002] as: (1) a set of static/dynamic attributes (e.g., cpu speed, OS, bandwidth available, data content, etc.) $R = \{R_1, R_2, ...\}$ where R_i is the semantic description of the i^{th} resource attribute, and (2) a set of usage policy related constraints $C_R = \{C_{R_1}, C_{R_2}, ...\}$. A *resource request* can be characterized as: (1) the requestor's profile $P = \{P_1, P_2, ...\}$, (2) a set of requested attributes $Rq = \{Rq_1, Rq_2, ...\}$, and (3) an ordered set of ranking preferences $Rk = \{Rk_1, Rk_2, ...\}$. A **resource ontology** $Onto_R$, which captures the structural organization of the resource concepts, is needed to define the semantics of R, C_R, P, Rq, and Rk.

Let $match_r(Rq, R)$ be a predicate, which is true if a request's description Rq can be *satisfied* by a resource with the attribute R, and $match_c(P, C)$ be a predicate for indicating whether the request's profile P can satisfy the resource constraints C. A resource described as $\{R, C\}$ matches a resource request $\{Rq, P\}$ if $match_r()$ and $match_c()$ are both true, that is

$$match_R(\{Rq, R\}, \{P, C\}) := match_r(Rq, R) \wedge match_c(P, C). \tag{1}$$

This type of attribute-based matching is *symmetric bilateral* [Raman et al., 1998] where a common syntax is assumed for both requests and resources. To support multiple resources and requests, variants of symmetrical multilateral matching have been proposed [Raman et al., 2000]. With more than one resource satisfying a request, Rk can be used for ranking the resources to support service selection.

If the user can specify explicitly their resource requests (using a restricted set of syntax), resource discovery is basically a database search problem. To further extend the Grid's interoperability, asymmetric matching algorithms based on a resource ontology $Onto_R$ can be used [Tangmunarunkit et al., 2003, Huang et al., 2003] so that the common-syntax assumption can be relaxed. Logic-based inference engines can here be used to support the ontology-based matching (see also Section 3.4).

3.1.2 Service Matchmaking

A service is here defined as an on-line software component S_i with a syntactic description $\{I, Sch_I, P, O, Sch_O\}$ (e.g., using WSDL for mark-up [WSDL, url, Foster et al., 2002]), where Sch_I is the schema of its input I, P is a transformation function for processing the input I, and Sch_O is the schema of the output O. A service can also be described based on their semantics as K_S using a service ontology $Onto_S$ (e.g., using OWL-S [OWL-S, url] or BPEL4WS [BPEL4WS, url]). The description includes the semantics of its input K_I, output K_O, capability K_P, process model K_{PM} (if the service is a composite one), and binding B to an exact implementation. Services with the same semantics except their bindings. can be grouped together to define an abstract service profile $K_{Sp} = \{K_I, K_O, K_P, K_{PM}\}$, which becomes open for service binding.

Service matching can be divided into syntactic matching based on $\{I, Sch_I, P, O, Sch_O\}$ and semantic matching based on $\{K_I, K_O, K_P, K_{PM}\}$. The former one has been well studied in the literature [Milo and Zohar, 1998, Rahm and Bernstein, 2001, Giannadakis et al., 2003] and is highly related to the component search and reuse issues in component-based software engineering [Yang and Papazoglou, 2002]. In the following, we only focus on the latter one. Let $match_i(K_I^r, K_I^p)$ denote a predicate for matching input semantic of a service request r and a service provider p. Similarly, $match_o(K_O^r, K_O^p)$, $match_p(K_P^r, K_P^p)$ and $match_{pm}(K_{PM}^r, K_{PM}^p)$ denote the predicates for semantic matching of output, capability and process model, respectively. Thus, a service provider with the profile K_{Sp}^p matching a service request K_{Sp}^r can be expressed as a conjunction of the four predicates. To contrast with resource matchmaking, service matchmaking involves much more diverse vocabularies for defining different types of services. The use of domain ontologies becomes inevitable [Paolucci et al., 2002]. Among the four matching predicates, $match_{pm}()$ is the distinct one as process matching requires sequence modelling and matching techniques [Molina-Jimenez et al., 2003, Wombacher et al., 2004], which are very much different from the matching of the others.

The service matchmaking formulation described above is typically used in the existing literature [Paolucci et al., 2002]. To extend it to fulfill the world wide scope of the Wisdom Web, further considerations like requestors' profiles and preferences as well as providers' constraints at a higher service level (e.g., related to their conflict of interest) should also be considered [Aiello et al., 2002].

3.1.3 Beyond Matchmaking: Search, Negotiate and Select

Before matchmaking can be performed, the Intelligent Grid should support searching for services registered in distributed repositories connected via a peer-to-peer network. The existing peer-to-peer techniques for file sharing should be extended to incorporate also functionalities like knowledge retrieval and heterogeneous ontology matching. Besides, while the matchmaking engine is a core component for service discovery, it is passive. In the Intelligent Grid, the service brokering agent should be *actively* negotiating with different service agents for processes like task bidding [Cheung et al., 2004a, Haque et al., 2005], making agreements and forming virtual organizations (highly applicable to service flow planning and scheduling to be described in the next section) [Norman et al., 2004], and clarifying concepts and establishing understanding between agents [Guha, 2004], with the ultimate goal of forming a real dynamic computational market. The underlying negotiation mechanism can be supported by the latest development of agent communication languages (e.g., KQML [Finin et al., 1994], Q [Ishida, 2002]) and related protocol standards, e.g., service-level agreement [Leff et al., 2003] and services choreography [WSCR, url].

3.2 Service Flow Planning

In the Intelligent Grid, explicit service requests should not be initiated by the user, but be derived based on the user specified high-level goal. A **user goal** is here referred to as a natural semantic description of a user specified problem $\mathcal{G}_{user}$, from which a precise goal $\mathcal{G}$, with explicit semantic description of *expected output or effect*, has to be deduced. Transforming $\mathcal{G}_{user}$ to $\mathcal{G}$ falls into the fields of human computer interaction and natural language processing, and, due to the focus of this paper, will not be further discussed. Given a precise goal $\mathcal{G}$, a service flow can be derived to fulfill it via planning. A service flow can formally be expressed as a directed acyclic graph $F(N, E)$, where the set of nodes N corresponds to service profiles and the set of directed edges E corresponds to consecutive execution orders of service profile pairs. The *input* (*output*) of a service flow is thus defined as the union of the all the inputs of the service profiles associated to the nodes with no *incoming* (*outgoing*) edges. Such a flow describes how a goal $\mathcal{G}$ can be achieved by coordinating the executions of the involved services according to $F(N, E)$. On-demand service flow planning is the process to discover and assign each node in F a service profile Sp_i such that:

1. $\mathcal{G}$ can be fulfilled by the output of F,
2. the input of F is either null or can be provided by the user,
3. the ordered service profile pair (Sp_i, Sp_j) corresponding to each element in E returns true values for $match_{io}(K_O^i, K_I^j)$.

The formulation is similar to that of AI planning, where the service profile Sp can be considered as an abstract action with K_I being its precondition, K_O its effect, K_P its operation, and K_{PM} its detailed composition for implementing the operation [Blythe et al., 2003a,b, Gil et al., 2004]. Planning related reasoning formalisms, such as situation calculus [Narayanan and McIlraith, 2002, Berardi et al., 2003, 2004] and hierarchical task networks (SHOP2) [Wu et al., 2003a], have been adopted. However, in the Wisdom Web, the wide distribution of the heterogeneous grid services, the on-demand requirement, and the new expectation for generic problem solving impose new research challenges.

1. First, the flow planning process (SFPA) has to be tightly integrated with the the service discovery process (SDA), which means that the operators possibly needed for the planning are to be discovered from remote sources. In the Intelligent Grid, the remote sources are also agents of different roles which *actively* cooperate with the SDA and thus SFPA for the planning [Brenner and DesJardins, 2002].

2. Also, the on-demand requirement implies that a goal requiring a complex flow cannot be planned from scratch, hinting the need of distributed retrieval and reuse of past flows/subflows. In the literature, the knowledge-based approach [Chen et al., 2003, Sirin et al., 2003] has been adopted to make related planning problems more tractable. For example, a data mining ontology including concepts like data cleansing, data pre-processing, data mining, etc., and their relationships has been proposed for supporting goal-driven on-demand data mining processing [Cannataro and Comito, 2003] (see Section 3.7).

3. Besides, depending on the application domains, consideration of constraints, profiles and preferences during the planning could also be incorporated [McIlraith and Son, 2002].

3.3 Service Flow Scheduling and Recovery

Given a service flow F with each of its nodes associated with a service profile as described in the previous section, the remaining steps are to 1) select particular service implementations for the nodes and 2) execute the flow.

Service scheduling (also sometimes called service selection) is defined as the assignment of particular service implementations to all the nodes in F so that the execution can be carried out according to some scheduling strategies for performance optimization. Instead of using conventional job scheduling approaches which are mostly deterministic and may not work best in the dynamic Grid environment, more robust and adaptive strategies should be adopted in the Intelligent Grid. For instance, one can assume each node with a local selection strategy designed to achieve an overall objective, e.g., to balance out the load through the Grid and at the same time taking care of the dynamic characteristics of the individual nodes adaptively [Wang and Liu, 2003, Cheung et al., 2004b,a]. Or one can also consider the selection of all the services within one flow as a whole so as to provide more robust and fault-tolerance service selection to meet some QoS promise for each flow execution [Pistore et al., 2004, Sample et al., 2002, Mao et al., 2001]. Also, the undeterministic conditions of individual services can also be modeled using probabilistic models, e.g., Markov Decision Process [Doshi et al., 2004], for computing the optimal policy of each state of a service flow, where the state of a flow changes when an action (service) is successfully taken (invoked) or not.

Service flow execution requires the coordination of a set of distributed grid services where the execution state and data flow management is unavoidable. There exist different approaches for the coordination infrastructure, namely, centralized [Casati and Shan, 2001], decentralized [Fauvet et al., 2001], and hybrid (e.g., centralized control but decentralized data flow) [Liu et al., 2002]. The Intelligent Grid should be equipped with some execution ontology for capturing concepts for supporting distributed service execution and orchestration processes. Examples of the concepts include service alternatives, process migration, execution status, management policies.

As the performance of distributed services, e.g., availability, is highly undeterministic, the two steps — service flow scheduling and service flow execution may need to interleave with each other, especially when a flow execution cannot be proceeded at a certain point that the SFEA needs to work with the SFPA for flow repair [Gil et al., 2004].

3.4 Distributed Inference

One practical Wisdom Web issue that has to be addressed is how to integrate distributed and centralized information sources/structures. The notion of Problem Solver Markup Languages (PSML) has been proposed in [Liu, 2003, Su et al., 2005] for declaring ways for 1) discovering and collecting globally distributed contents and knowledge from Web-supported, semantic social networks, and 2) integrating with locally operational knowledge/databases in an enterprise or community for locally centralized, adaptable Web intelligence services. Distributed inference engines that can perform automatic reasoning on the Web by incorporating contents and metaknowledge, autonomously collected and transformed from the semantic Web, with the locally operational knowledge/databases are thus needed. A feasible way to work to extend some Semantic Web enabled prolog-like inference engine [FaCT, url, RACER, url] by including dynamic contents and metaknowledge collection and transformation agents. For example, Ohsuga *et al.* previously used KAUS, a knowledge-based system that involves Multi-Layer Logic and databases, for representing local information sources and for performing inferences [Ohsuga, 1990]. Inference Web [McGuinness and Silva, 2004] is another recent project which is a semantic web enabled inference engine for question-and-answer applications with explanation. It addresses partially the *autonomous* data collection by explaining the data derivation details.

3.5 Wisdom Agent Specialization

To support just-in-time orchestration of the highly dynamic computational resources, another important research issue of Grid Intelligence is to accurately *characterize* the dynamic behavior and quality of the Wisdom Agents (e.g., currently commitable computing power, bandwidth, reputation, etc.), the evolving role of the agents (e.g., specialization based on their dynamic cache contents, nature of jobs they work best, etc.) [Tsui et al., 2003], as well as the the spatial and temporal contraints among the agents (e.g., the dynamic trust between resource providers and requesters at different time and in different geographical regions, or triggered by the evolving roles of the agents). The adaptative characterization and specialization capabilities of the agents can facilitate the Intelligent Grid to achieve in an autonomous manner best overall performance, including high efficiency, utilization and reliability [Cao et al., 2003] .

3.6 Agent Community Self-Organization

The agents in the Intelligent Grid possess a peer-to-peer (P2P) logical structure, forming a social network. In particular, let $\{A_i\}$ denote a set of autonomous agents, each being one of the resources providers (or one of the distributed resource repositories), and storing a table $neighhors(A_i)$ pointing to its peers. A resource request $\{P, Rq, Rk\}$, found unable to be served by A_i, can be forwarded to a particular subset of $neighbors(A_i)$ chosen according to a search strategy where a number of related works have been proposed in the peer-to-peer literature for addressing the search efficiency issue. In the Intelligent Grid, $neighbors(A_i, t)$ should also be adaptively evolved based on the past history of *P2P search*, *resource usage*, *knowledge usage*, resulting in an evolving grid topology to support efficient (performance-based [Maj et al., 2003, Yolum and Singh, 2003]) and reliable (reputation-based [Maximilien, 2002]) just-in-time resource/knowledge sharing. There is also related work on derivation of characteristics based on the local interaction of the distributed resources to assist the formation of agents' coliations [Iamnitchi et al., 2004]. Further embracing intelligence into the service coordination can start with modeling the interaction of services as that in multi-agent systems [Buhler and Vidal, 2004, Singh and Huhns, 1999, Liu et al., 2004]. The social behaviour consideration in multi-agent systems (e.g., cooperation, competition, proactiveness) has been recognized as a new dimension for enriching the autonomy of the coordination infrastructures.

3.7 Problem Solving Knowledge Evolution and Reuse

Generic problem solving using distributed knowledge components has been considered in the literature [Tate, 1998, V. Richard Benjamins and Fensel, 1999, Martinez and Lesperance, 2004], where one of the fundamental challenge lies in how to represent the metaknowledge of actions taken before and their relationships as problem solving experience (ontology) marked up using PSML. For instance, the distributed evolving problem solving knowledge structures can be supported by techniques like problem solving episodes [Veloso, 1994, Stuber et al., 2003] which summarize past experience in solving problems/sub-problems of different types, by different users, and from heterogeneous sources. In addition, efficient indexing, updating, and retrieval are also important research issues to meet the on-demand requirement [Schank, 1980, Ginsberg, 1993, Kim, 2004]. Furthermore, how the conceptual problem solving knowledge should take into account of the execution performance experience and peer evaluation to further self-organize the underlying structure is another direction for widening the metaknowledge capability of the Wisdom Web.

4 Related Semantic/Knowledge Grid Projects

- The Knowledge Grid Project in Italy [Cannataro and Talia, 2003] aims to provide an environment for designing, deploying and executing distributed data mining applications on the Grid. It consists of two hierarchical levels, namely the *Core K-grid layer* and the *High level K-grid layer*. The former one offers the basic services for managing metadata about resource and coordinating the application execution on top of Grid. The latter one includes services for composing, validating and executing distributed knowledge discovery tasks, as well as services for managing the discovered knowledge. VEGA is a prototype developed with a GUI for accessing the Knowledge Grid.

- In UK, there are several projects related to Knowledge Grid. The objective of the *Discovery Net Project* [Sairafi et al., 2003] is to design an architecture to support knowledge discovery and data mining processes in a Grid-enabled distributed computing environment. The proposed architecture consists of three major modules, namely, *Resource Discovery Server*, *Knowledge Server* and *Meta-information Server*. An XML-based language called Discovery Process Markup Language (DPML) was proposed for modeling processes where data are pipelined through computation services and data services for knowledge discovery. Users can compose processes via a GUI and the applicability of the project was illustrated under a bioinformatics context (e-Science). The myGrid Project in UK [Wroe et al., 2003] is yet another project on composing distributed knowledge services in Grid for problem solving. It adopts heavily the Semantic Web technology (DAML+OIL) for *matchmaking* bioinformatics related requests and services/data (e-Science). The proposed architecture consists of several key components, namely *Ontology server*, *FaCT reasoner*, *Matcher* and *Ranker* to make the matchmaking possible.

- The Knowledge Grid Project (VEGA-KG) in China [VEGA-KD, url, Zhuge, 2004a,b] introduces the concept of Knowledge Space which tries to specify a knowledge resource based on its location (e.g., an URI), knowledge level (e.g., concept, axiom, rule or method) and knowledge category (e.g., knowledge, information). It treats the Knowledge Grid somewhat as a distributed knowledge base and proposes a SQL-like language called KGOL for accessing knowledge resources and developing Knowledge Grid related applications. Knowledge Base Grid (KB-Grid) [Wu et al., 2003b] is another Knowledge Grid project in China, which suggests a paradigm for organizing, discovering, utilizing and managing Web knowledge base resources, and has been applied to support knowledge services of Traditional Chinese Medicine.

In the literature, there also exist different software tools that implement the idea of the Knowledge Grid and Grid Intelligence to different extents. Examples include Artemis [Tuchinda et al., 2004], Proteus [Cannataro and Talia, 2003], ARION [Houstis et al., 2003], Osiris+O'Grape [Weber et al., 2003], etc. We believe that more are going to come to further enrich the KGGI related development.

5 Conclusion

In this paper, we first argued that the notion of "wisdom" should be the paradigm shift in the next generation Web Intelligence, and then described the fundamental capabilities of the Wisdom Web and a corresponding Intelligent Grid for supporting it. We believe that the Wisdom Web vision requires not only a semantically annotated Semantic/Knowledge Grid, but also a Grid *with* its architecture, characteristics and associcated metaknowledge capable of self-organizing, evolving and re-configuring on-demand to support generic problem solving. We hope that the Grid Intelligence research issues highlighted in this paper and the detailed review of recent advancements in related research areas form a roadmap for future research towards an autonomy oriented grid architecture for running the Wisdom Web.

Acknowledgement

This work has been partially supported by RGC Central Allocation Group Research Grant (HKBU 2/03/C). The authors would like to thank Ning Zhong for his encouragement and support.

References

M. Aiello, M. Papazoglou, J. Yang, M. Carman, M. Pistore, L. Serafini, and P.Traverso. A request language for web-services based on planning and constraint satisfaction. In *Proceeding of the VLDB Workshop on Technologies for E-Services*, 2002.

H.P. Alesso and C.F. Smith. *The Intelligence Wireless Web*. Addison-Wesley, 2002.

B. Benatallah, M. Dumas, M.C. Fauvet, and F.A. Rabhi. Towards patterns of web services composition. In *Patterns and Skeletons for Parallel and Distributed Computing*. Springer Verlag, 2002.

D. Berardi, D. Calvanese, D.G. Giuseppe, M. Lenzerini, and M. Mecella. Synthesis of composite e-services based on automated reasoning. In *Proceedings of the ICAPS 2004 Workshop on Planning and Scheduling for Web and Grid Services (P4WGS 2004)*, 2004.

D. Berardi, D. Calvanese, D.G. Giuseppe, M.Lenzerini, and M.Mecella. e-service composition by description logics based reasoning. In *Proceedings of the 2003 Description Logic Workshop (DL 2003)*, pages 75–84. CEUR Electronic Workshop Proceedings, http://ceur-ws.org/Vol-81/, 2003.

T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284:34–43, 2001.

D. Billsus, C.A. Brunk, C. Evans, B. Gladish, and M. Pazzani. Adaptive interfaces for ubiquitous web access. *Communications of the ACM*, 45:34–38, 2002.

J. Blythe, E. Deelman, and Y. Gil. Planning for workflow construction and maintenance on the Grid. In *Proceedings of the ICAPS'03 Workshop on Planning for Web Services*, Trento, Italy, June 2003a.

J. Blythe, E. Deelman, Y. Gil, C. Kesselman, A. Agarwal, G. Mehta, and K. Vahi. The role of planning in grid computing. In *Proceedings of the 13th International Conference on Automated Planning and Scheduling*, Trento, Italy, June 2003b.

BPEL4WS, url. http://www-106.ibm.com/developerworks/library/ws-bpel/.

M. Brenner and M. DesJardins, editors. *Proceedings of the AAAI-02 Workshop on Planning with and for Multi-Agent Systems*, 2002.

P. Buhler and J.M. Vidal. Integrating agent services into BPEL4WS defined workflows. In *Proceedings of the Fourth International Workshop on Web-Oriented Software Technologies*, 2004.

M. Cannataro and C. Comito. A data mining ontology for grid programming. In *Proceedings of the First Workshop on Semantics in Peer-to-Peer and Grid Computing*, Budapest, Hungary, May 2003.

M. Cannataro and D. Talia. Knowledge grid: An architecture for distributed knowledge discovery. *Communications of the ACM*, January 2003.

J. Cao, D.P. Spooner, S.A. Jarvis, S. Saini, and G.R. Nudd. Agent-based grid load balancing using performance-driven task scheduling. In *Proceedings of the 17th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2003)*, Nice, France, April 2003.

F. Casati and M. Shan. Dynamic and adaptive composition of e-services. *Information Systems*, 6(3), 2001.

L. Chen, N.R. Shadbolt, C. Goble, F. Tao, S.J. Cox, C. Puleston, and P.R. Smart. Towards a knowledge-based approach to semantic service composition. In *Proceedings of the 2nd International Semantic Web Conference, Lecture Notes in Computer Science, LNCS 2870*, pages 319–334, Florida, USA, 2003.

A. Chervenak, I. Foster, C. Kesselman, C. Salisbury, and S. Tuecke. The data grid: Towards an architecture for the distributed management and analysis of large scientific datasets. *Journal of Network and Computer Applications*, 23:187–200, 2001.

W.K. Cheung, J. Liu, K.H. Tsang, and R.K. Wong. Dynamic resource selection for service composition in the grid. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, Beijing, China, 2004a.

W.K. Cheung, J. Liu, K.H. Tsang, and R.K. Wong. Towards autonomous service composition in a grid environment. In *Proceedings of the 2004 IEEE International Conference on Web Services*, San Diego, 2004b.

M. Dorigo, V. Maniezzo, and A. Colorni. The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on System, Man, and Cybernetics - Part B*, 26(1):1–13, 1996.

P. Doshi, R. Goodwin, R. Akkiraju, and K. Verma. Dynamic workflow composition using markov decision processes. In *Proceeding of the 2004 IEEE International Conference on Web Services*, pages 576–582, July 2004.

R. Ewerth, T. Friese, M. Grube, and B. Freisleben. Grid services for distributed video cut detection. In *Proceedings of the 6th IEEE International Symposium on Multimedia Software Engineering*, Miami, USA, 2004.

FaCT, url. http://www.cs.man.ac.uk/~horrocks/FaCT.

M. Fauvet, M. Dumas, B. Benatallah, and H. Paik. Peer-to-peer traced execution of composite services. In *Proceedings of the VLDB Workshop on Technologies for E-Services*, 2001.

T. Finin, Y. Labrou, and J. Mayfield. KQML as an agent communication language. In *Proceedings of the International Conference on Information and Knowledge Management*, pages 456–463, New York, 1994. ACM Press.

I. Foster, C. Kesselman, J.M. Nick, and S. Tuecke. Grid services for distributed system integration. *IEEE Computer*, June 2002.

I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid: Enabling scalable virtual organizations. *International Journal of High Performance Computing Applications*, 15:200–222, 2001.

M. Gaynor, M. Welsh, S. Moulton, A. Rowan, E. LaCombe, and J. Wynne. Integrating wireless sensor networks with the grid. *IEEE Internet Computing - Special Issue on the Wireless Grid*, July/August 2004.

N. Giannadakis, A. Rowe, M. Ghanem, and Y. Guo. InfoGrid: Providing information integration for knowledge discovery. *Information Sciences*, 155:199–226, 2003.

Y. Gil, E. Deelman, J. Blythe, C. Kesselman, and H. Tangmunarunkit. Artificial intelligence and grids: Workflow planning and beyond. *IEEE Intelligent Systems - Special Issue on e-Science*, January/February 2004.

A. Ginsberg. A unified approach to automatic indexing and information retrieval. *IEEE Expert: Intelligent Systems and Their Applications*, 8(5):46–56, 1993.

Globus, url. The Globus Alliance, http://www.globus.org.

R. Guha. Semantic negotiation: Co-identifying objects across data sources. In *Proceeding of the 2004 AAAI Spring Symposium Series: Semantic Web Services*, March 2004.

N. Haque, N. Jennings, and L. Moreau. Resource allocation in communication networks using market-based agents. *International Journal on Knowledge Based Systems*, 18, 2005.

C. Houstis, S. Lalis, V. Christophides, D. Plexousakis, E. Vavalis, M. Pitikakis, K. Kritikos, and A. Smardas. A data, computational and knowledge grid: The case of the ARION system. In *Proceedings of the 5th International Conference on Enterprise Information Systems*, Angers, France, April 2003.

L. Huang, Z. Wu, and Y. Pan. A scalable and effective architecture for grid services' discovery. In *Proceedings of the First Workshop on Semantics in Peer-to-Peer and Grid Computing*, Budapest, Hungary, May 2003.

Y. Huang and N. Venkatasubramanian. Supporting mobile multimedia services with intermittently available grid resources. In *Proceedings of the International Conference on High Performance Computing*, 2003.

B.A. Huberman, P.L.T. Pirolli, J.E. Pitkow, and R.M. Lukose. Strong regularities in world wide web surfing. *Science*, 280:96–97, 1997.

A. Iamnitchi, M. Ripeanu, and I. Foster. Small-world file-sharing communities. In *Proceedings of the IEEE Infocom 2004*, Hong Kong, 2004.

T. Ishida. Q: A scenario description language for interactive agents. *IEEE Computer*, pages 42–47, November 2002.

J.O. Kephart and D.M. Chess. The vision of autonomic computing. *IEEE Computer*, 36(1):41–50, 2003.

J. Kim. Memory-based meta-level reasoning for interactive knowledge capture. Internal Report of Echo Project, Information Sciences Institute, 2004.

R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. The web and social network. *IEEE Computer - Special Issue on Web Intelligence*, 35(11):32–36, 2002.

A. Leff, J.T. Rayfield, and D.M. Dias. Service-level agreements and commercial grids. *IEEE Internet Computing*, 7(4), July/August 2003.

D. Liu, K. Law, and G. Wiederhold. Data-flow distribution in FICAS service composition infrastructure. In *Proceedings of the 15th International Conference on Parallel and Distributed Computing Systems*, Louisville, KY, 2002.

J. Liu. Web Intelligence (WI): What makes Wisdom Web? In *Proceedings of the 18th International Joint Conference on Artificial Intelligence (IJCAI-03)*, pages 1596–1601, Acapulco, Mexico, Aug. 9-15 2003. Morgan Kaufmann Publishers (Invited Talk).

J. Liu, X. Jin, and K. Tsui. *Autonomy Oriented Computing: From Problem Solving to Complex Systems Modeling*. Springer, 2005.

J. Liu, S. Zhang, and J. Yang. Characterizing web usage regularities with information foraging agents. *IEEE Transactions on Knowledge and Data Engineering*, 16(5):566–584, May 2004.

A. Maj, J. Jurowicz, J. Kolak, and K. Cetnarowicz. A multi-agent system for dynamic network reconfiguration. *Multi-agent Systems and Applications III, LNAI 2691*, 2003.

Z.M. Mao, E.A. Brewer, and R.H. Katz. Fault-tolerant, scalable, wide-area internet service composition. Technical Report UCB/CSD-1-1129, CS Division, EECS Department, U.C. Berkeley, 2001.

E. Martinez and Y. Lesperance. Web service composition as a planning task: Experiments using knowledge-based planning. In *Proceedings of the ICAPS'04 Workshop on Planning and Scheduling for Web and Grid Services*, Whistler, Bristish Columbia, Canada, 2004.

E.M. Maximilien. Conceptual model of web service reputation. *ACM SIGMOD Record*, 31(4):36–41, December 2002.

D.L. McGuinness and P.P.D. Silva. Explaining answers from the semantic web: The inference web approach. *Journal of Web Semantics*, 1(4):397–413, 2004.

S. McIlraith and T. Son. Adapting golog for composition of semantic web services. In *Proceedings of the 8th International Conference on Principles of Knowledge Representation and Reasoning*, 2002.

T. Milo and S. Zohar. Using schema matching to simplify heterogeneous data translation. In *Proceedings of the 24th International Conference on Very Large Data Bases*, pages 122–133, 1998.

C. Molina-Jimenez, S. Shrivastava, E. Solaiman, and J. Warne. Contract representation for run-time monitoring and enforcement. In *Proceedings of the 2003 IEEE International Conference on Electronic Commerce*, 2003.

S. Narayanan and S. McIlraith. Simulation, verification and automated composition of web services. In *Proceedings of the 11th World Wide Web Conference*, Honolulu, Hawaii, USA, May 2002.

T. J. Norman, A. Preece, S. Chalmers, N. R. Jennings, M. Luck, V. D. Dang, T. D. Nguyen, V. Deora, J. Shao, A. Gray, and N. Fiddian. Agent-based formation of virtual organisations. *International Journal of Knowledge Based Systems*, 17(2-4):103–111, 2004.

S. Ohsuga. Framework of knowledge based systems - multiple meta-level architecture for representing problems and problem solving processes. *Knowledge Based Systems*, 3(4):204–214, 1990.

OWL-S, url. http://www.daml.org/services/owl-s/1.1.

M. Paolucci, T. Kawmura, T. Payne, and K. Sycara. Semantic matching of web services capabilities. In *Proceedings of the 1st International Semantic Web Conference*, 2002.

M. Pistore, P. Bertoli, and P. Traverso. Planning and monitoring web service composition. In *Proceedings of the ICAPS04 Workshop on Planning and Scheduling for Web and Grid Services*, Whistler, British Columbia, Canada, June 2004.

RACER, url. http://www.sts.tu-harburg.de/∼r.f.moeller/racer.

E. Rahm and P.A. Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10 (4), 2001.

R. Raman, M. Livny, and M. Solomon. Matchmaking: Distributed resource management for high throughput computing. In *Proceedings of the Ninth IEEE International Symposium on High Performance Distributed Computing*, pages 290–291, Pittsburgh, Pennsylvania, 2000.

R. Raman, M. Livny, and M.H. Solomon. Matchmaking: Distributed resource management for high throughput computing. In *Proceedings of the 7th IEEE International Symposium on High Performance Distributed Computing*, 1998.

D.D. Roure, M.A. Baker, N.R. Jennings, and N.R. Shadbolt. The evolution of the grid. In Anthony J.G. Hey Fran Berman and Geoffrey Fox, editors, *Grid Computing: Making The Global Infrastructure a Reality*, pages 65–100. John Wiley & Sons, 2003a.

D.D. Roure, N.R. Jennings, and N.R. Shadbolt. The semantic grid: A future e-science infrastructure. In Anthony J.G. Hey Fran Berman and Geoffrey Fox, editors, *Grid Computing: Making The Global Infrastructure a Reality*, pages 437–470. John Wiley & Sons, 2003b.

S. Al Sairafi, F.S. Emmanouil, M. Giannadakis, Y. Guo, D. Kalaitzopolous, M. Osmond, A. Rowe, J. Syed, and P. Wendel. The design of Discovery Net: Towards open grid services for knowledge discovery. *International Journal of High Performance Computing Applications*, 17(3), 2003.

N. Sample, P. Keyani, and G. Wiederhold. Scheduling under uncertainty: Planning for the ubiquitous grid. In *Coordination Models and Languages*, pages 300–316, 2002.

R.C. Schank. Conceptual information retrieval. In *Proceedings of the 3rd Annual ACM Conference on Research and Development in Information Retrieval*, pages 94–116, 1980.

SG-Portal, url. Semantic Grid Community Portal, http://www.semanticgrid.org.

SI-eScience. *IEEE Intelligence Systems - Special Issue on e-Science*, January/February 2004.

SI-Middleware. *IEEE Internet Computing - Special Issue on Middleware for Web Services*, January/February 2003.

M.P. Singh and M.N. Huhns. Multiagent systems for workflow. *International Journal of Intelligent Systems in Accounting, Finance, and Management*, 8:105–117, 1999.

E. Sirin, J. Hendler, and B. Parsia. Semi-automatic composition of web services using semantic descriptions. In *Proceedings of Web Services: Modeling, Architecture and Infrastructure Workshop in ICEIS*, April 2003.

H.G. Stork. Webs, grids and knowledge spaces - programmes, projects and prospects. *Journal of Universal Computer Science*, 8(9), 2002.

A. Stuber, S. Hassas, and A. Mille. Combining MAS and experience reuse for assisting collective task achievement. In *Proceedings of the 5th International Conference on Case-Based Reasoning Workshop on From*

Structured Cases to Unstructed Problem Solving Episodes for Experience-Based Assistance, Trondheim, Norway, June 2003.

Y. Su, L. Zheng, N. Zhong, C. Liu, and J. Liu. Distributed reasoning based on Problem Solver Markup Language (PSML). In *Proceedings of the 2005 International Conference on e-Commerce, e-Technology and e-Service*, Hong Kong, March 2005.

R. Sudra, S.J.E. Taylor, and T. Janahan. Distributed supply chain simulation in grids. In *Proceedings of the 2000 Winter Simulation Conference*, 2000.

SW-Portal, url. The Semantic Web Communicty Portal, http://www.semanticweb.org.

H. Tangmunarunkit, S. Decker, and C. Kesselman. Ontology-based resource matching in the Grid - the Grid meets the. In *Proceedings of the 2nd International Semantic Web Conference*, Sanibel-Captiva Islands, Florida, USA, October 2003.

A. Tate. Roots of SPAR - shared planning and activity representation. *The Knowledge Engineering Review, Special Issue on "Putting Ontologies to Use"*, 13(1):121–128, 1998.

D. Thain, T. Tannenbaum, and M. Livny. Condor and the grid. In *Grid Computing: Making the Global Infrastructure a Reality*. John Wiley & Sons Inc., 2002.

K. Tsui, J. Liu, and M. Kaiser. Self-organized load balancing in proxy servers: Algorithms and performance. *Journal of Intelligent Information Systems*, 20(1):31–50, 2003.

R. Tuchinda, S. Thakkar, Y. Gil, and E. Deelman. Artemis: Integrating scientic data on the grid. In *Proceedings of the 16th Annual Conference on Innovative Applications of Artificial Intelligence*, San Jose, CA, July 2004.

J. Wielemaker V. Richard Benjamins, B.J. Wielinga and D. Fensel. Towards brokering problem-solving knowledge on the internet. In *Lecture Notes In Computer Science, Proceedings of the 11th European Workshop on Knowledge Acquisition, Modeling and Management*, pages 33–48, 1999.

VEGA-KD, url. http://kg.ict.ac.cn.

M.M. Veloso. Flexible strategy learning: Analogical replay of problem solving episodes. In *Proceedings of AAAI-94*, pages 595–600, 1994.

Y. Wang and J. Liu. Dynamics of agent-based load balancing on grids. In *Proceedings of the 2nd International Conference on Autonomous Agents and Multi-Agent Systems*, pages 804–811, July 2003.

R. Weber, C. Schuler, P. Neukomm, H. Schuldt, and H.J. Schek. Web service composition with ÓGRAPE and OSIRIS. In *Proceedings of the 29th International Conference on Very Large Data Bases*, Berlin, Germany, September 2003.

M. Weiser. The future of ubiquitous computing on campus. *Communications of the ACM*, 41(1):41–42, 1998.

A. Wombacher, P. Fankhauser, B. Mahleko, and E. Neuhold. Matchmaking for business processes based on choreographies. *The International Journal of Web Services Research*, 1(4):14–32, Oct-Dec 2004.

C. Wroe, R. Stevens, C. Goble, A. Roberts, and M. Greenwood. A suite of DAML+OIL ontologies to describe bioinformatics web services and data. *International Journal of Cooperative Information Systems*, 12(2):197–224, 2003.

WSCR, url. Web Services Choreography Requirements, http://www.w3.org/TR/ws-chor-reqs.

WSDL, url. http://www.w3.org/TR/wsdl.

D. Wu, E. Sirin, J. Hendler, D. Nau, and B. Parsia. Automatic web services composition using SHOP2. In *Proceedings of the 12th International World Wide Web Conference*, Budapest, HUNGARY, May 2003a.

Z. Wu, H. Chen, and J. Xu. Knowledge base grid: A generic grid architecture for semantic web. *Journal of Computer and Technology*, 18(4):462–473, 2003b.

J. Yang and M. Papazoglou. Web components: A substrate for web service reuse and composition. In *Advanced Information Systems Engineering*, Toronto, Canada, May 2002.

P. Yolum and M.P. Singh. Dynamic communities in referral networks. *Web Intelligence and Agent System: An International Journal*, pages 105–116, 2003.

N. Zhong, J. Liu, and Y.Y. Yao. *Web Intelligence*. Springer, 2003.

H. Zhuge. China's e-science knowledge grid environment. *IEEE Intelligent Systems*, 19(1):13–17, January/February 2004a.

H. Zhuge. *The Knowledge Grid*. World Scientific Co., 2004b.