Language Service Management with the Language Grid

Yohei Murakami†, Donghui Lin†, Masahiro Tanaka†, Takao Nakaguchi∗, Toru Ishida†‡

†Language Grid Project, National Institute of Information and Communications Technology
3-5 Hikaridai, Seika-Cho, Soraku-Gun, Kyoto, 619-0289, Japan
{yohei, lindh, mtnk}@nict.go.jp

∗NTT Advanced Technology Corporation
12-1 Ekimae-Honmachi, Kawasaki-Ku, Kanagawa, 210-0007, Japan takao.nakaguchi@ntt-at.co.jp

‡Department of Social Informatics, Kyoto Univerisity
Yoshida-Honmachi, Sakyo-Ku, Kyoto, 606-8501, Japan
ishida@i.kyoto-u.ac.jp

Abstract

As the number of language resources accessible on the Internet increases, many efforts have been made for combining language resources and language processing tools to create new services. However, existing language resource coordination frameworks cannot manage issues of intellectual property associated with language resources, which make it difficult for most end-users to get supports for their intercultural collaborations because they always have to deal with the issues by themselves. In this paper, we aim at constructing a new language service management architecture on the Language Grid, which enables language resource providers to control access to their resources in accordance with their own policies. Furthermore, we apply the proposed architecture to the operating Language Grid in order to validate the effectiveness of the architecture. As a result, several service management models utilizing the monitoring and access constraints are occurring to satisfy various requirements from language resource providers. These models can handle paid-for language resources as well as free language resources. Finally, we discuss further challenging issues of combining language resources under each different policies.

1. Introduction

Rapid internationalization continues to expand multicultural society where people with different nationalities coexist. In Japan, the number of foreign residents is over 2,150,000[1], which occupies about two percent of total population. As a result, intercultural and multilingual activities are occurring frequently in daily life, such as questioning foreign patients in hospitals and teaching foreign students in schools, and so on. However, language barriers always make it difficult for such communications.

Although there are many language resources (both data and programs) on the Internet (Choukri, 2004), most intercultural collaboration activities are still lacking multilingual support. To overcome language barriers, we aim at constructing a novel language infrastructure to share and combine language resources on the Internet, and provide multilingual services required in intercultural activity fields.

Many efforts have been put for combining language resources and language processing tools in some previous works, such as UIMA and GATE(Ferrucci and Lally, 2004; Cunningham et al., 2002; Callmeier et al., 2004; Váradi et al., 2008; Boehlke, 2009). However, existing language resource coordination frameworks cannot manage issues of intellectual property associated with language resources, which make it difficult for most end-users to get supports for their intercultural collaborations because they always have to deal with the issues of intellectual property by themselves. To solve this problem, we propose a new language service management architecture on the Language Grid(Ishida, 2006), which enables language resource providers to control access to their resources in accordance with their own policies.

The rest of this paper describes the Language Grid, and then proposes language service management architecture and a management tool to realize various forms for providing language resources by language resource providers. Finally, we explain how Language Grid users provide their language resources by using the proposed language service management architecture.

2. Language Grid

The Language Grid is a multilingual service infrastructure for enabling users to share language resources developed by linguistic specialists and end-user communities, and combine these resources based on their own requirements to support their intercultural collaboration in multicultural society. To this end, language resources are wrapped to form Web services that users can combine by workflows with the WS-BPEL specifications[2] to create customized language services for their activities. These services are also published as Web services so that various intercultural collaboration tools can employ language services.

The feature of the Language Grid is to integrate language services, which is different from EuroWordNet and Global WordNet Grid(Fellbaum and Vossen, 2007) that integrate lexical data based on word meaning. Therefore, the Language Grid enables users to combine machine translation

[1] Annual Statistics of Foreign Residents in Japan in 2007 by the Immigration Bureau, the Ministry of Justice

[2] http://www.oasis-open.org/committees/wsbpel

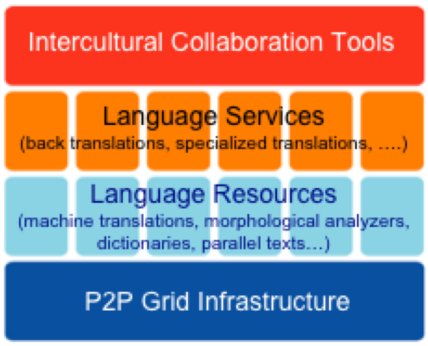

Figure 1: Language Grid Service Layer

services with specialized dictionary services or parallel text services for improving translation quality specific to the users' fields.

2.1. Service Layer

As shown in Figure 1, the Language Grid consists of four service layers.

P2P Grid Infrastructure This infrastructure organizes multiple servers, language grid core nodes and language grid service nodes(Murakami et al., 2006), on the Internet to fulfill end users ' requests. Language Grid users can add their servers to the P2P Grid, and access to usage statistics of their resources.

Language Resources Various language resources will be provided as atomic services with a standardized interface. To increase usability of the language resources for language services, standardization of access entry is quite important(Calzolari et al., 2002). We started working on language resource ontology, which standardizes the interfaces(Hayashi and Ishida, 2006; Hayashi et al., 2008). Language Grid users can easily add new language resources to the Language Grid.

Language Services Various language services for intercultural collaboration can be created by combining existing language resources. We have already implemented Web service workflows including back translations and domain-specialized translations. Language Grid users can easily add new language services to the Language Grid by themselves.

Intercultural Collaboration Tools The interface at the top layer provides Intercultural Collaboration Tools so that the users can utilize the Language Grid, even if they have no programming skills. Collaboration tools are developed using language services explained as above. New tools can be developed by users, and existing tools can be multilingualized.

2.2. Stakeholders

Language Grid user means three types of stakeholders; computation resource providers, language resource providers, and language service users. Computation resource providers register servers which constitute the P2P Grid Infrastructure in the bottom layer, language resource providers register language resources in the second layer, and language service users create language services in the third layer and develop intercultural collaboration tools using the language services.
In addition to these stakeholders, there is Language Grid operator, who manages the registered servers, language resources, language services, and users.

3. Requirements

In order to facilitate supply and usage of language resources on the Language Grid, we should understand the following situations and requirements from stakeholders, especially language resource providers. These requirements caused by complicated intellectual property issues related to language resources.

Language processing tools from for-profit organizations Machine translators are often developed and operated by for-profit companies, and are provided for profit. However, if the application area of the Language Grid does not conflict with an already existing business market, we can collaborate with those companies and receive a substantial discount on prices. One solution is that universities, research institutes or large NGOs voluntarily buy translation services and provide them to the Language Grid without any charge .

Language processing tools from academic organizations Morphological analyzers and other language processing programs are often developed by research institutes or universities. In many cases, researchers can provide their resources without any charge for research purposes. Even if the goal is not for research, if their use can be restricted to non-profit, researchers often agree to provide their resources. Sometimes researchers may request access information to their language resources for checking whether their resources are used properly. For profit use, on the other hand, tools are not always free and contracts cannot be concluded uniformly.

Language data from public organizations Multilingual dictionaries and multilingual parallel texts may or may not be free. Even for non-profit use, if the resources are already being sold, difficult problems exist with regard to the distribution those resources without charge. Since the Language Grid is based on Web services, however, there is a chance to make those resources freely available by setting the upper limit to daily access number. To enhance the security of the language resources, only those who buy the resources may be allowed to access the language resources.

To satisfy the above requirements from language resource providers, as well as combining language resources, the Language Grid must manage language services following their provision policy.

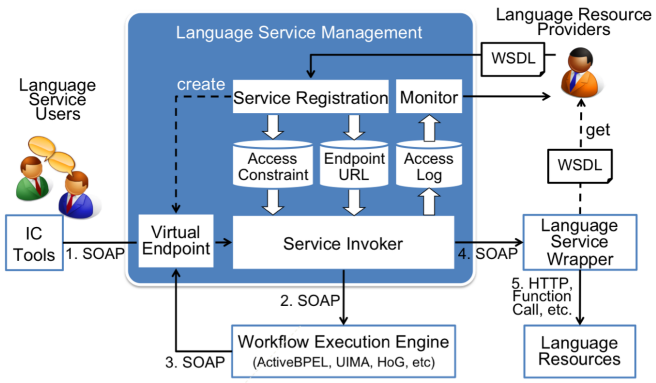

Figure 2: Architecture for Language Service Management

4. Language Service Management

4.1. System Architecture

There have been several frameworks and projects that coordinate language resources and language processing tools, such as UIMA, Heart of Gold, GATE, CLARIN, and D-Spin. Their purpose is to analyze a large amount of text data by linguistic processing pipelines. These pipelines consist of language processing programs with different interfaces, the most of which are provided as open sources by universities and research institutes.

Different from above frameworks, the purpose of the Language Grid is to translate texts multilingually for supporting intercultural collaboration. Therefore, the Language Grid requires language resources that are created with high costs and associated with complex intellectual property issues, such as machine translators, parallel corpora, and bilingual dictionaries. To manage intellectual properties, language service management architecture is required on the Language Grid as well as the framework of combining language resources, which enables language resource providers to control their services in accordance with their own policies.

Figure 2 shows the system architecture for language service management on the Language Grid. Language resources and language processing tools are wrapped as Web services by language service wrappers. To provide a language resource, the language resource provider obtain WSDL file, description of Web service interfaces, and then register the WSDL file, copyright and license information of the language resource to the Language Grid. By registering WSDL files instead of language resources, language resource providers can provide services while managing their language resources on their own servers.

When *Service Registration component* receives a WSDL file from a language resource provider, it extracts the endpoint URL and its interface information from the WSDL file, creates a virtual endpoint with the same interface on the Language Grid, and publishes it to language service users. The purpose of using virtual endpoints is to prohibit direct access to language resources by hiding their real endpoints from language service users. By this means, the Language Grid can manage the access to the language services. Moreover, language resource providers can easily improve scalability of their services by deploying more language service wrappers and language resources, and relating these endpoints to the created virtual endpoint.

A language service user sends a SOAP request to a virtual endpoint from an intercultural collaboration tool (IC Tool) to invoke a language service. The virtual endpoint forwards the request to *Service Invoker*. *Service Invoker* checks whether the request can satisfy access constraints that are set when the service is registered. If the request is verified, *Service Invoker* obtains original endpoint information, and accesses the language resource. In invoking the service, if several endpoints are related to the virtual endpoint, *Service Invoker* chooses one endpoint whose latency is the lowest of them in order to distribute loads without loads' concentrating on a single endpoint. Responses from language resources are accumulated in *Access Log* database, and are used to validate satisfaction of access constraints and monitor usage of language resources.

In the case of invoking composite services, the request will be sent to *Workflow Execution Engine*. After receiving the request, *Workflow Execution Engine* invokes the atomic services that are defined in a workflow corresponding to the composite service. If the workflow also consists of virtual endpoints, the request will be sent to the corresponding virtual endpoints. Since SOAP communication is used between the language service management block and the workflow execution engine in the language service management architecture, we can apply this architecture to Web service-based language resource coordination frameworks, such as Heart of Gold, UIMA, and D-Spin. We have started to bridge Heart of Gold and the Language Grid(Bramantoro et al., 2008) and will apply the results to combine UIMA and the Language Grid.

4.2. Language Grid Service Manager

Language Grid Service Manager[3] is a Web-based management tool to enable Language Grid users to access various types of management function provided by the Language Grid. It provides language service repository function including registration, deletion, and search of language resources and services, and language service management function including monitoring, and access control of language resources.

Language service repository function registers language services in the Language Grid using information given by the language resource providers, such as WSDL, copyright information, and license information. Furthermore, it can search for language services by service name, service type, supported languages, and access rights. The search result is a list of service profiles, each which shows copyright information and license information given by a language resource provider in registering the corresponding service. In this way, language resource providers can inform users who will use their resources of their copyright information and license information. In the case of a composite service, the service profile provides a list of atomic services composing it in order to prevent it from hiding the constituent atomic services. This list enables users to confirm copyright in-

[3] http://langrid.org/operation/service_manager/

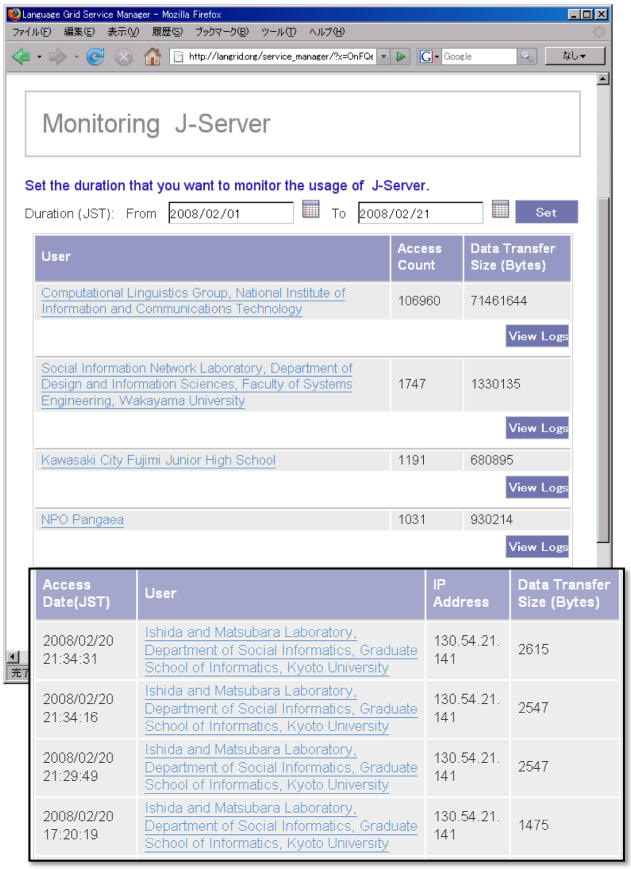

Figure 3: Monitoring and Logging Usage of Language Resources (J-Server)

formation and license information of atomic services even though invoking a composite service.

Monitoring function provides total access count and total data transfer size by each language service user during a specified duration to language resource providers. Moreover, it also shows when each language service user access a language resource from which IP address, and how much data size is sent and received in each request. Figure 3 displays the GUI of Language Grid Service Manager. This enables language resource providers who provide their resources only for the non-profit use to check who uses their resources and whether fraudulent usage occurs or not. Since Language Grid Service Manager also provides user profile information including user's homepage and e-mail address, the language resource providers can understand from the homepage the purpose for which they use the language resources and contact the users by e-mail.

Access control function allows language resource providers to set access rights on each language resource. If a language resource provider finds a language service user who accesses his/her resource excessively, he/she can prohibit the access from the user to the resource. Moreover, language resource providers have two choices in publishing their services: " public mode " that permits every user by default and " members only " mode that prohibits every user by default. Using the " members only " mode, a language resource provider who sells a language resource can

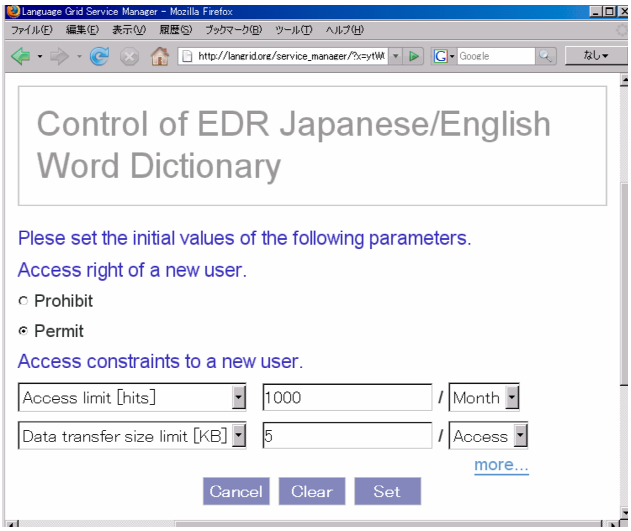

Figure 4: Access Constraint of Language Resources (EDR Japanese/English Word Dictionary)

permit a language service user who purchased the language resource or its license to access the resource.

Access control function provides access constraint settings as well as access right settings (Figure 4). Access constraints include total access count per month, week, and day, and data transfer size (KB) per month, week, day, and request. This function enables a language resource provider who sells a language resource to provide limited service as a trial one to language service users who have not purchased it, and provide various types of service according to the fees.

5. Case Study of Language Service Management

Department of Social Informatics in Kyoto University has operated the Language Grid since December in 2007[4]. Now, 118 organizations from 17 countries participate in the Language Grid. 27 organizations provide language resources, and 67 language resources in 41 languages are registered in the Language Grid. Language resource providers who sell their resources realize various forms of providing their resources by employing the language service management effectively.

National Institute of Information and Communications Technology (called NICT hereafter) distributes a concept dictionary and a bilingual dictionary called EDR as a whole for a fee. That is why NICT has difficulty in allowing language service users to employ EDR freely. Therefore, NICT provides trial service of EDR to every user by setting maximum access counts per month at 1000 counts and maximum data transfer size per request at 15 KB for the concept dictionary, and maximum access counts per month at 1000 counts and maximum data transfer size per request at 5KB for the bilingual dictionary, respectively. These constraints are configured to take about one year to extract all data of EDR. Moreover, NICT has registered the concept

[4]http://langrid.org/operation/

dictionary and the bilingual dictionary of EDR without any restrictions in the "members only" mode. In this way, NICT provides unlimited EDR services to only users who purchase the EDR license.

KODENSHA Co., Ltd. (called KODENSHA hereafter) allows us to provide translation service based on J-Server, a machine translator, to third party, if the application area of the Language Grid does not conflict with an already existing business market. Now, Kyoto University and NICT purchased J-Server software to provide translation service to other language service users. If the application area of the Language Grid conflicts with the existing market, NICT and Kyoto University prohibit the conflicting users from accessing J-Server. Furthermore, KODENSHA has registered J-Server ASP service operated by KODENSHA in the "members only" mode. In this way, KODENSHA provides latest J-Server service to only users who purchase the J-Server ASP license. KODENSHA has registered Japanese and Simplified Chinese, Japanese and Traditional Chinese, Japanese and Korean, and Japanese and English translation separately to increase service variations. This enables users to purchase language pairs that they need to use.

Kyoto University provides to language service users various language services based on machine translation software, translation ASP service, and text-to-speech engine that Kyoto University purchased from some companies; KODENSHA Co., Ltd., Cross Language Inc., Translution, and HOYA CORPORATION. Since Kyoto University concluded an agreement that establishes the provision of the language resource for only non-profit use with each language resource developer, Kyoto University has to monitor whether the language resource is abused or not. In fact, by monitoring the access to the language resources, Kyoto University detected that a user accessed J-Server of KODENSHA excessively from a specific IP address. Kyoto University obtained contact address from the user profile and contacted the user in order to confirm whether the user employed it for non-profit use.

Gengo-Shigen-Kyokai (called GSK hereafter) is a non-profit organization promoting the distribution of language resources and language processing tools on behalf of the language resource providers. Therefore, GSK considers managing language resources of Language Grid users who are willing to sell their resources. The plan is to deploy the resources on GSK's server and register them to the Language Grid in "members only" mode. GSK will be able to reduce Language Grid users' burden of providing language resources for a fee, undertaking sales and operation of language resources.

6. Discussion

Service management is one of the four significant issues in services computing domain, along with service foundation, service composition, and service design and development(Papazoglou et al., 2007). Especially, in an open environment where several stakeholders coexist like the Language Grid, it is necessary to take into consideration policies of service providers in composing services. Below we address the issues expected to be raised in language service composition by the proposed architecture.

The first issue is how language service users know the availability of composite language services consisting of atomic language services with different policies. The language service users should check whether they can satisfy every policy of the constituent atomic language services. To allow the users to check them, the Language Grid Service Manager provides a list of the service types constituting a composite service in the service profile of the composite service. The users choose an atomic language service, which corresponds to each service type and whose policy is satisfied.

Another issue is how completion of composite language services is assured in the open environment. In the open environment, the execution of composite language services may fail due to the runtime environment. One possible failure is caused by restrictions set out by language resource providers. Assume that several end users who use the same user ID invoke an composite language service including an atomic language services with an access limit. In this case, the number of invocation exceeds the limit while executing the composite language service even though each end user can satisfy the limit before executing it. Since language service providers can easily change their policy in the open environment, this types of problems often occurs. To solve this problem, the user has to add an exception handler to switch to another similar atomic language service in the composite language service. However, if the user does not have the right to modify the composite language service, we need another solution independent of the composite language service. Another possibility is that we extend the language service management architecture to supervise the execution of composite language services(Tanaka et al., 2009). This approach can be applied to various types of composite language service.

So far, we focus on several functions which language resource providers need for language service management. However, language service users also need a function for language service management. The function is managing QoS. For example, language service users want to know which combinations of language resources are best to satisfy users' needs. To estimate QoS of composite language services, the language service management architecture must hold the QoS of each atomic language service including quality of contents as well as latency and costs. The aggregation of these QoS values is also a hot topic in services computing domain(Zeng et al., 2004).

7. Conclusion

To share and coordinate language resources on the Internet, we need not only technologies but also institutional design considering incentives among stakeholders(Ishida et al., 2008). In this paper, we have proposed language service management architecture to realize non-profit operation of the Language Grid. The essence of this architecture is as follows.

- This architecture allows language resource providers to set out various restriction according to their policy; restrictions on users who may be licensed to use their language resources, and on the number of times

that their language resources may be accessed and the amount of data that may be transfered from the language resources.

- This architecture provides user interface called Language Grid Service Manager, which enables language resource providers to easily access language service management function; monitoring how the provided language resources are used and setting out their provision policy.

- This architecture contributes to improvements of accessibility and usability of language resources on other language resource coordination frameworks because the architecture is independent of workflow execution engines.

We have already applied the proposed architecture to the Language Grid operated by Kyoto University. So far, 67 language resources and about 80 language services are registered in the Language Grid. 6 language resources of them are language processing tools from for-profit organizations, such as machine translators and a text-to-speech engine, 9 language resources are language processing tools from academic, such as morphological analyzers and dependency parsers, and the remaining are language data from public or non-profit organizations, such as parallel corpus and bilingual dictionaries. This statistics shows users can create a new composite services by combining a few general-purpose language resources developed by linguistic professionals with various domain-specific language resources developed by communities for their activities.

Acknowledgements

This work was supported by Strategic Information and Communications R&D Promotion Programme (SCOPE) of the Ministry of Internal Affairs and Communications of Japan. future approaches.

8. References

V. Boehlke. 2009. A Prototype Infrastructure for D-Spin-Services Based on a Flexible Multilayer Architecture. In *Proc. of Text Mining and Services Conference (TMS'09)*.

A. Bramantoro, M. Tanaka, Y. Murakami, U. Schäfer, and T. Ishida. 2008. A Hybrid Integrated Architecture for Language Service Composition. In *Proc. of the Sixth International Conference on Web Services (ICWS'08)*, pages 345–352.

U. Callmeier, A. Eisele, U. Schäfer, and M. Siegel. 2004. The Deep Thought Core Architecture Framework. In *Proc. of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pages 1205–1208.

N. Calzolari, A. Zampolli, and A. Lenci. 2002. Towards a Standard for a Multilingual Lexical Entry: The EAGLES/ISLE initiative. In *Proc. of the CICLing*, pages 264–279.

K. Choukri. 2004. European Language Resources Association History and Recent Developments. In *SCALLA Working Conference KC 14/20*.

H. Cunningham, D. Maynard, K. Bontcheva, and V. Tablan. 2002. GATE: An Architecture for Development of Robust HLT Applications. In *Proc. of the Fortieth Annual Meeting on Association for Computational Linguistics (ACL'02)*, pages 168–175.

C. Fellbaum and P. Vossen. 2007. Connecting the Universal to the Specific: Towards the Global Grid. In *Intercultural Collaboration*, number 4568 in Lecture Notes in Computer Science, pages 1–16. Springer-Verlag.

D. Ferrucci and A. Lally. 2004. UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *Journal of Natural Language Engineering*, 10:327–348.

Y. Hayashi and T. Ishida. 2006. A Dictionary Model for Unifying Machine Readable Dictionaries and Computational Concept Lexicons. In *Proc. of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 1–6.

Y. Hayashi, T. Declerck, P. Buitelaar, and M. Monachini. 2008. Ontologies for a Global Language Ifrastructure. In *Proc. of the First International Conference on Global Interoperability for Language Resources (ICGL'08)*, pages 105–112.

T. Ishida, A. Nadamoto, Y. Murakami, R. Inaba, T. Shigenobu, S. Matsubara, H. Hattori, Y. Kubota, T. Nakaguchi, and E. Tsunokawa. 2008. A Non-Profit Operation Model for the Language Grid. In *Proc. of the First International Conference on Global Interoperability for Language Resources (ICGL'08)*, pages 114–12.

T. Ishida. 2006. Language Grid: An Infrastructure for Intercultural Collaboration. In *Proc. of the IEEE/IPSJ Symposium on Applications and the Internet (SAINT'06)*, pages 96–100.

Y. Murakami, T. Ishida, and T. Nakaguchi. 2006. Infrastructure for Language Service Composition. In *Proc. of the Second International Conference on Semantics, Knowledge, Grid (SKG-06)*.

M.P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann. 2007. Service-Oriented Computer: State of the Art and Research Challenges. *IEEE Computer*, 40(11):38–45.

M. Tanaka, T. Ishida, Y. Murakami, and S. Morimoto. 2009. Service Supervision: Coordinating Web Services in Open Environment. In *Proc. of the Eighth International Conference on Web Services (ICWS-09)*, pages 238–245.

T. Váradi, P. Wittenburg, S. Krauwer, M. Wynne, and K. Koskenniemi. 2008. CLARIN: Common Language Resources and Technology Infrastructure. In *Proc. of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 1244–1248.

L. Zeng, B. Benatallah, A.H.H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang. 2004. QoS-Aware Middleware for Web Services Composition. *IEEE Transactions on Software Engineering*, 30(5):311–327.