Research on Purifying for Ambient Semantic Interoperable Services Oriented to Global Enterprises

Guo Jianfeng
CIMS Center,Department. of Automation
Tsinghua University
Beijing 100084, P.R.China
Guojf@tsinghua.edu.cn

Fan Yushun
CIMS Center,Department. of Automation
Tsinghua University
Beijing 100084, P.R.China
fanyus@tsinghua.edu.cn

***Abstract*—The architecture of the SofTware for Ambient Semantic Interoperable Services (STASIS) was introduced. The functionalities of the STASIS Purifier Tool (SPT) used for filtering the problematic elements in STASIS were proposed. Based on the SPT Plug-in and SPT Server, the structure and framework of SPT are discussed. Finally, the implementation of SPT and the current situation of SPT are presented.**

Keywords-semantic; ontology; interoperable service; STASIS

I. INTRODUCTION

More and more enterprises distributed in all over the world have been involved in the international eEconomy within an eBusiness context. For electronic trading, it has become a key issue for a lot of companies to enhance the interoperability between their information Systems and that of their partners. But usually, it's not enough to enable the information exchange between two systems work well by just converting data-format in these systems. Some information in different systems among diverse enterprises must be mapped semantically to let computers understand them accurately. Therefore, how to design and develop an open platform in internet and its client applications that can be used easily by users and exchange with the central platform effectively will be significant, especially for SMEs [1-4].

SofTware for Ambient Semantic Interoperable Service (STASIS) is a research and development project sponsored under the Europeans Commission's 6th Framework Programme as well as its projects members, which releases in the internet address (www.stasis-project.net). Its objective is for research, development and validation of open, web services based, distributed semantic services for SME empowerment within the automotive, furniture and other sectors. 12 Partners are involved including commercial companies and academics which are spread across Europe and China.

II. STASIS COMPONENT CONCEPTS

STASIS was built in a distributed pattern in Internet. The general structure of STASIS is showed in Figure 1. The STASIS framework consists of a set of tools, services and repositories that together help business to simplify setting up meaningful interactions by allowing them to specify and link their own semantics with that of their partners.

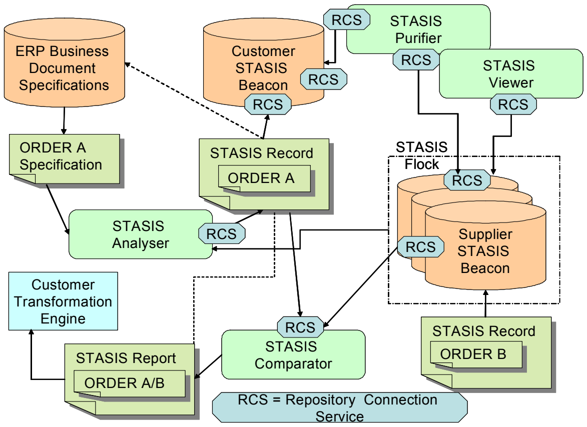


Figure 1. STASIS component concepts

Broadly speaking, the STASIS framework consists of:

- A network (Flock) of distributed repositories (Beacons) that stores the semantic information
- STASIS Analyser Tool (SAT) that allows business to enter their information into the framework
- STASIS Comparator Tool (SCT) to help mapping semantics stored in the framework and to extract cross-references for use between partners
- STASIS Viewer Tool (SVT) for those that just want to have a look
- STASIS Editor Tool (SET) for maintenance of stored data
- STASIS Purifier Tool(SPT) that locates inconsistencies in the stored information
- Shared services, e.g. searching, information retrieval, query distribution and result handling Interfaces for stacking the layers together

III. THE RESPONSIBILITIES OF SPT

Since STASIS is public and open system for any users globally, along with the time going, there would be more and more semantic entities added to it, and all elements in the

Europeans Commission's 6th Framework programme (FP6-2005-IST-5-034980), National 973 Project (2006CB705407), Country's Natural Science Fund of China (60674080), and the 863 Plan of China (2007AA04Z1 50).

system can be mapping each other manually or automatically. Because the distributed mechanism of STASIS, in fact, it's a totally self-organizing system, and there is no central controlling way to manage and maintain if the datum and mapping in STASIS is in good structure or correct. So there must be an individual tool in STASIS to purify all semantic entities for maintenance.

A. *STASIS functional building blocks*

The STASIS functional architecture is showed in Figure 2. Obviously, SPT is a very independent part with the others components. SPT are crawling all elements created by others components and stored in the SRRN (STASIS Registry and Repository Network) continuously to finding the problematic elements, and making the notifications of the corrupted entities to their owners.

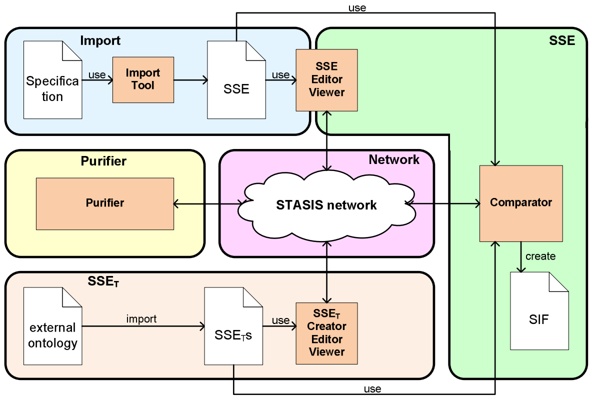


Figure 2. STASIS functional architecture

Before moving on SPT in detailed, some abbreviations in STASIS are introduced as following:

- SSE: STASIS Semantic Entity, an instance of some concept recognized as such in the STASIS environment by means of some semantic extraction or annotation.
- SSET: Terminological STASIS Semantic Entity, a concept recognized as such in the STASIS environment by means of some semantic extraction or annotation
- SNF: STASIS Neutral Format, a representational (hence, in particular, non semantic) encoding of new elements entering the STASIS environment as acquisition of corresponding items of an external (source or target) schema.
- SLS: The STASIS Link Specification, a link used to link two semantic entities in STASIS.
- SIF: The STASIS Interface Format. SIFs are groups of SLS. These are used as output from the comparator to an external mapper.

B. *The functionalities of SPT*

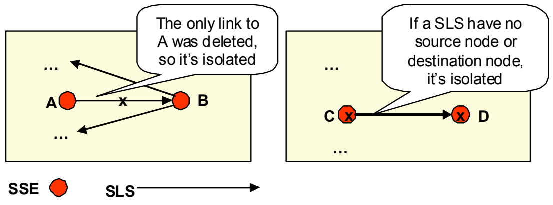
Two main functionalities in SPT are as following:

1) Any user should be able to make subscription for the semantic elements he is interested in because too many datum in the STASIS repositories for users to get and update the elements that are useful to them.

2) Finding the problematic semantic elements in repositories. There are several kinds of problematic elements should and can be found by SPT:

a) The elements are maybe out of date. If some semantic entities have not been used for a relative long time, these elements should be notified and removed.

b) The elements are maybe isolated. When a semantic entity, especially for a SSE, is added to STASIS, it will be mapped to another one based on their semantic similarity, which is implemented by adding a link (SLS) of the two elements manually or automatically. But anyone SSEs or SLSs can be edited by its owner via SET, and for a client user, it's impossible to know if the SSE or SLS he wants to delete is the only source node, destination node, or link for others SLS or SSE. The situations for isolated SSE and isolated SLS are described in Figure 3.



The only link to A was deleted, so it's isolated
...
A
x
B
...
If a SLS have no source node or destination node, it's isolated
...
C
D
...
SSE
SLS

Figure 3. Isolated semantic elements

c) The elements have maybe potential semantic conflicts with others elements. This is a more complex situation. In STASIS, we can discuss this from two ranges, local repository and global repositories. The situations of potential inconsistency can be generally described in Figure 4 and Figure 5.

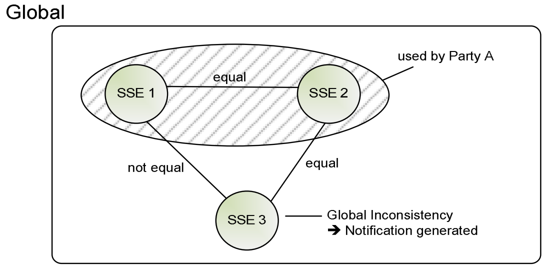
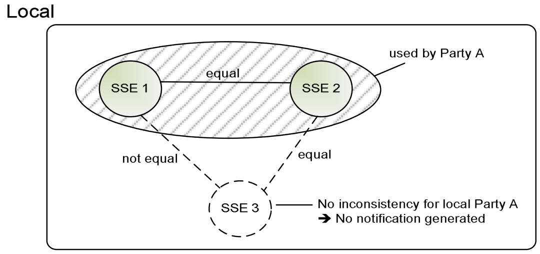


Figure 4. Global inconsistency



Local
used by Party A
SSE 1
equal
SSE 2
not equal
equal
SSE 3
No inconsistency for local Party A
➔ No notification generated

Figure 5. Local inconsistency

Basically, the semantic inconsistency means that some SSEs that are mapped to the same SSE as equal are not equal. From the Figure 4 and Figure 5, obviously, the same situation for an inconsistency will get very different results in local repository and in global repositories. The most important and hardest works to find semantic inconsistency are how to indentify two SSE are not equal and how to optimize the algorithms to finding all this kind of issues in a huge complex net of semantic entities[5].

IV. THE ARCHITECTURE OF SPT

SPT consists of three main parts, configuration part, crawling part, as well notification and subscription part. Configuration part takes charge of setting the parameters for SPT, including running parameters for initializing SPT and filtering criteria for purifying. Crawling part is the most important part within SPT. It takes charge of crawling all kinds of semantic elements by RCS (Repositories Connection Services). Notification and Subscription part is the terminal of SPT. It takes charge of storing the results and notifying the results to uses. Figure 6 show the architecture in detailed.

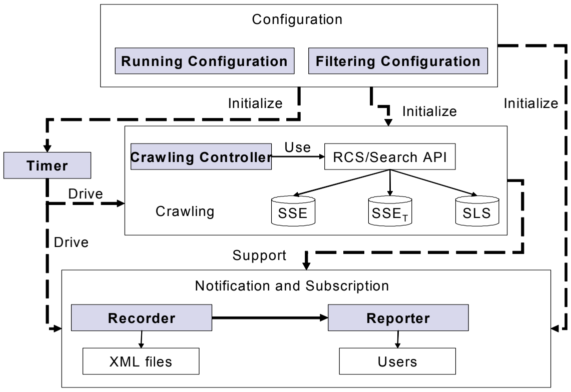


Figure 6. The architecture of SPT

According the two main functionalities of SPT, the SPT is divided into two components, SPT plug-in and SPT Server. SPT plug-in is a part of STASIS workbench that developed with Eclipse to enable the user subscription and to be the interfaces for users to set SPT Server and to get the global error reports and subscription results from SPT Server. The Physical architecture of SPT is showed in Figure 7.

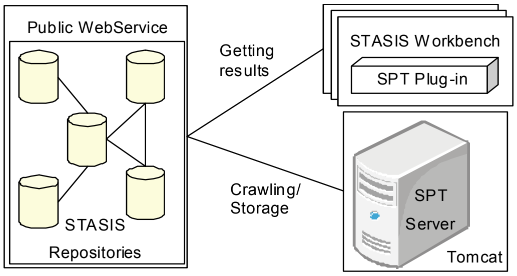


Figure 7. Physical structure of SPT

In a word, SPT Plug-in is an interface for users to interact with SPT, and SPT Server plays a more important role which is facing the SRRN directly and processing all filtering stuff. The framework of SPT Server is showed in Figure 8.

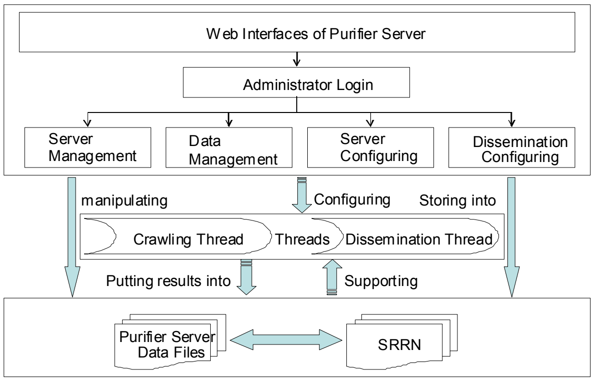


Figure 8. The framework of SPT Server

SPT Server is web application and just open for several administrators. So there must be a user management for its security. The administrators of SPT Server can do several operations for SPT Server, including server management for turning server on/off, date management for handling the reports in server, server configuration for setting the option of server, as well dissemination configuration for sending the reports to relative users. Two main threads are running in SPT Server, one for the crawling and filtering in SRRN, the other one for sending the reports to users. The two threads are running from time to time in a period setting by administrators.

V. THE IMPLEMENTATION OF SPT

A. The implementation of SPT Plug-in

The SPT Plug-in is implemented as Eclipse plug-in for the STASIS RCP workbench. The structure of the SPT Plug-in is showed in Figure 9.

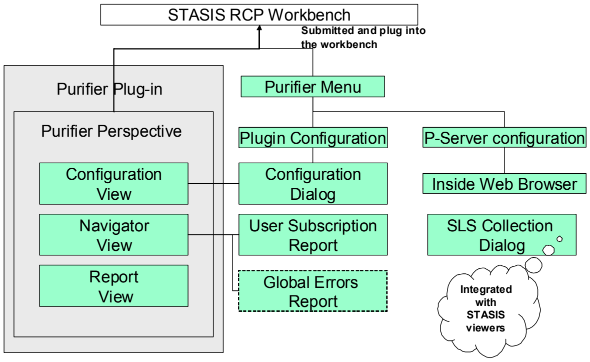
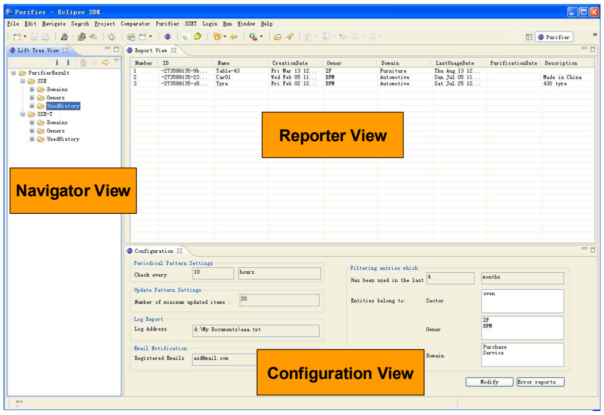


Figure 9. The structure of SPT Plug-in

When SPT Plug-in is integrated into the STASIS workbench, a Purifier perspective and a Purifier menu will be added to the workbench. There are three views within the perspective, including configuration view, navigator view, and report view. The Purifier menu is used to trigger the dialogs for setting configuration of SPT Plug-in and initializing an inside web browser that can connect to SPT Server. An extra dialog in SPT Plug-in are used to find and send SLS errors to SPT Server by users manually since some problematic semantic links can not or hard to be found by SPT automatically. Figure 10 is the screen shot of the Purifier perspective of SPT plug-in. Figure 11 is the screen shot of the Navigator view in Purifier perspective that are used to classify the subscribed results. Figure 12 is the screen shot of a part of Report view in Purifier perspective that is used to show the detailed subscribed results in a table format.



Number	ID	Name	CreationDate	Owner	Domain	LastUsageDate	PurificationDate	Description
1	-273599135-9b...	Table-43	Fri Mar 13 12...	ZF	Furniture	Thu Aug 13 12...		
2	-273599135-23...	Car01	Wed Feb 05 11...	BFM	Automotive	Sun Jul 05 11...		Made in China
3	-273599135-c8...	Tyre	Fri Feb 02 12...	BFM	Automotive	Sat Jul 25 12...		430 tyre

Periodical Pattern Settings
Check every: 10 hours

Update Pattern Settings
Number of mininum updated items : 20

Log Report
Log Address: d:\My Documents\aaa.txt

Email Notification
Registered Emails: asd@mail.com

Filtering entries which:
Has been used in the last: 4 months
Entities belong to: Sector: sven
Owner: ZF BFM
Domain: Purchase Service

Figure 10. Purifier perspective of SPT Plug-in

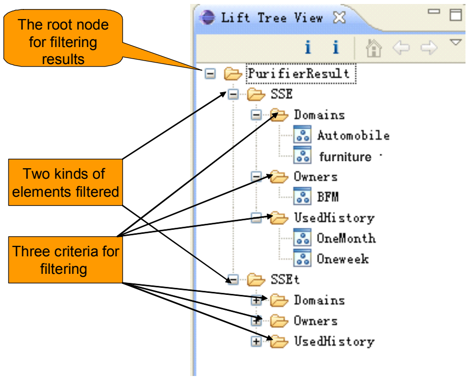
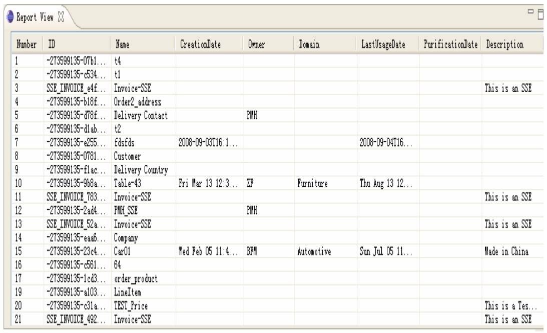


Figure 11. Navigator view in Purifier perspective

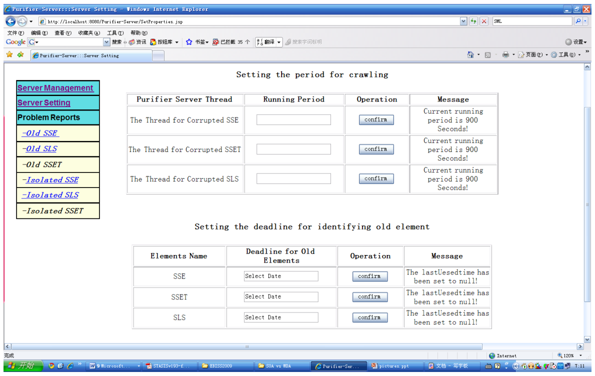


Number	ID	Name	CreationDate	Owner	Domain	LastUsageDate	PurificationDate	Description
1	-273599135-07b1...	t4						
2	-273599135-c534...	t1						
3	SSE_INVOICE_e4f...	Invoice-SSE						This is an SSE
4	-273599135-b18f...	Order2_address						
5	-273599135-d78f...	Delivery Contact		PMH				
6	-273599135-d1ab...	t2						
7	-273599135-e255...	fdsfds	2008-09-03T16:1...			2008-09-04T16...		
8	-273599135-0781...	Customer						
9	-273599135-f1ac...	Delivery Country						
10	-273599135-9b8a...	Table-43	Fri Mar 13 12:3...	ZF	Furniture	Thu Aug 13 12...		
11	SSE_INVOICE_783...	Invoice-SSE						This is an SSE
12	-273599135-2ad4...	PMH_SSE		PMH				
13	SSE_INVOICE_52a...	Invoice-SSE						This is an SSE
14	-273599135-eaa6...	Company						
15	-273599135-23c4...	Car01	Wed Feb 05 11:4...	BFM	Automotive	Sun Jul 05 11...		Made in China
16	-273599135-c561...	64						
17	-273599135-1cd3...	order_product						
19	-273599135-a103...	LineItem						
20	-273599135-c31a...	TEST_Price						This is a Tes...
21	SSE_INVOICE_492...	Invoice-SSE						This is an SSE

Figure 12. Report view in Purifier perspective

B. *The implementation of SPT Server*

SPT Server is implemented as a web application and deployed individually. In fact, it's the middle agent between SRRN and SPT Plug-in, so there need just to be limited simple web pages to setting the option for the threads in SPT Server, for finding the semantic elements which are out of date, for triggering the threads in SPT Server, and for showing error reports of all kinds of semantic elements. Figure 13 is the screen shot for SPT Server setting. Figure 14 is the screen shot for switch the threads in SPT Server. Figure 15 is the screen shot of the isolated SSE report.


Purifier-Server:::Server Setting - Windows Internet Explorer
http://localhost:8080/Purifier-Server/SetProperties.jsp
Server Management
Server Setting
Problem Reports
-Old SSE
-Old SLS
-Old SSET
-Isolated SSE
-Isolated SLS
-Isolated SSET

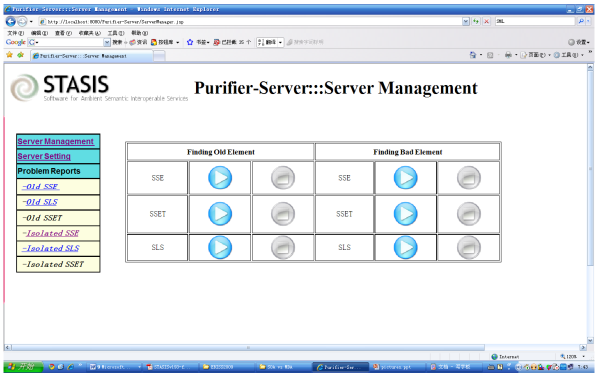
Setting the period for crawling

Purifier Server Thread	Running Period	Operation	Message
The Thread for Corrupted SSE		confirm	Current running period is 900 Seconds!
The Thread for Corrupted SSET		confirm	Current running period is 900 Seconds!
The Thread for Corrupted SLS		confirm	Current running period is 900 Seconds!

Setting the deadline for identifying old element

Elements Name	Deadline for Old Elements	Operation	Message
SSE	Select Date	confirm	The lastUesedtime has been set to null!
SSET	Select Date	confirm	The lastUesedtime has been set to null!
SLS	Select Date	confirm	The lastUesedtime has been set to null!

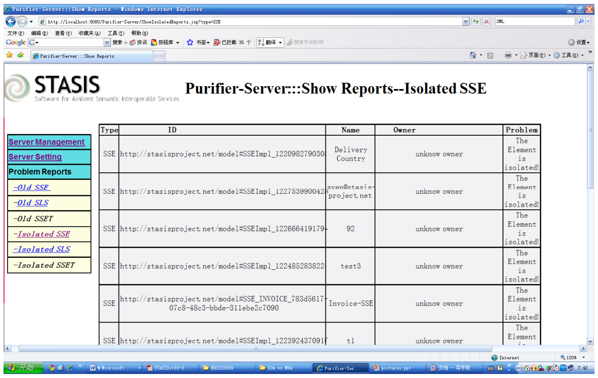
Figure 13. Setting for SPT Server


Purifier-Server:::Server Management - Windows Internet Explorer
http://localhost:8080/Purifier-Server/ServerManager.jsp
STASIS
Software for Ambient Semantic Interoperable Services
Server Management
Server Setting
Problem Reports
-Old SSE
-Old SLS
-Old SSET
-Isolated SSE
-Isolated SLS
-Isolated SSET

Purifier-Server:::Server Management

Finding Old Element			Finding Bad Element		
SSE	▶	■	SSE	▶	■
SSET	▶	■	SSET	▶	■
SLS	▶	■	SLS	▶	■

Figure 14. The triggers for SPT Server


Purifier-Server:::Show Reports - Windows Internet Explorer
http://localhost:8080/Purifier-Server/ShowIsolatedReports.jsp?type=SSE
STASIS
Software for Ambient Semantic Interoperable Services
Server Management
Server Setting
Problem Reports
-Old SSE
-Old SLS
-Old SSET
-Isolated SSE
-Isolated SLS
-Isolated SSET

Purifier-Server:::Show Reports--Isolated SSE

Type	ID	Name	Owner	Problem
SSE	http://stasisproject.net/model#SSEImpl_122098279030	Delivery Country	unknow owner	The Element is isolated!
SSE	http://stasisproject.net/model#SSEImpl_122753990042	sven@stasis-project.net	unknow owner	The Element is isolated!
SSE	http://stasisproject.net/model#SSEImpl_122666419179-	92	unknow owner	The Element is isolated!
SSE	http://stasisproject.net/model#SSEImpl_122485283822	test3	unknow owner	The Element is isolated!
SSE	http://stasisproject.net/model#SSE_INVOICE_783d5617-07c8-48c3-bbde-311ebe2c7090	Invoice-SSE	unknow owner	The Element is isolated!
SSE	http://stasisproject.net/model#SSEImpl_122392437091	t1	unknow owner	The Element

Figure 15. Isolated SSE report

In SPT Server, there is a Web Service that is always running in background. SPT Plug-in can access the error reposts in SPT Server by the Web Service to get the reports about problematic semantic entities that are only belonging to the active user in SPT Plug-in.

VI. Conclusion

STASIS will play a significant role for global enterprises, especially for SMEs. Because of the self-organizing mechanism of STASIS, there must be an extra tool in STASIS to find and notify problematic semantic entities from huge elements added by the distributed users. So far, the SPT have been able to find the elements that are out of date and the elements that are isolated well. But there are still some researches about finding semantic inconsistency that should be done further.

Acknowledgment

The research work is obtained financial support from the Europeans Commission's 6th Framework programme (FP6-2005-IST-5-034980), the National 973 Project (2006CB705407), the Country's Natural Science Fund of China (60674080), and the 863 Plan of China (2007AA04Z150). The interested readers are able to access http://www.stasis-project.net for information.

References

[1] Pease A, Niles I, Li J. The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications[R]. Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web. 2002, 28.

[2] Abels S, Campbell S, Hahn A. Interoperability and Standards In: The Single European Electronic Market in the perspective of the i2010 strategy[R]. Workshop Booklet SEEM i2010 - Information Space for SMEs, Brussels. 2005.

[3] Abels S, Campbell S, Hahn A. Accessing and Managing Heterogeneous Information in the Single Electronic European Market[C]. Proceedings of the e-Challenges 2005 Conference, Ljubljana, Slovenia. 2005.

[4] Vlachakis J, Abels S, Campbell S, et al. SEEM-enabling technologies. Proceedings of the e-Challenges 2004 Conference, Vienna,Austria. 2004.

[5] Fan Y, Huang C, Wang Y, et al. Architecture and operational mechanisms of networked manufacturing integrated platform[J]. International Journal of Production Research,Vol. 43, No. 12, 15 June 2005, 2615–2629.