INFORMATION TECHNOLOGY JOURNAL

Dynamic Mapping Processing Between Global Ontology and Local Ontologies in Grid Environment

[1]Yonghong Luo, [2]Zhou Yu, [3]Yanhao Zhuang and [3]Zhaopeng Zheng
[1]School of Computer and Communication Engineering, Changsha University of Science and Technology,
960 2nd Section, Wanjiali South RD, Changsha, Hunan, 410114, China
[2]School of Information and Engineering, Henan Institute of Science and Technology, China
[3]School of Civil Engineering and Transportation, South China University of Technology, China

Abstract: In grid computing environment, we can create a global ontology for multiple local ontologies to integrate geographically distributed data sources. However, some data sources are likely to dynamically join and quit grid environment. Thus, it is very important for data semantic access to build dynamic mapping relation between global ontology and local ontologies. In this study, we present mapping algorithms of joining and quitting for data sources in grid system. When a new data source joins grid system, elements mapping between global ontology and its local ontology can be rapidly established through using the method of subtree interception and the corresponding global elements should be added to active elements set. During processing of quitting algorithms, elements mapping between global ontology and the quitted ontology should be canceled and the corresponding global elements need to be move to suspended elements set. In semantic data operation, we regulate only mappings generated from active elements set can be employed for data semantic query. Experimental results show that proposed algorithm effectively fulfils dynamic mappings between global ontology and local ontologies and improves the accuracy of user semantic query in grid environment.

Key words: Grid computing, ontology, dynamic mapping, joining algorithm, quitting algorithm

INTRODUCTION

Grid computing aims to gather geographically distributed resources such as computational power, data, software, equipment to form a virtual organization which can provide users with integrative services of information and application (Foster and Kesselman, 1999). Data resources in one grid system, especially in a virtual organizational context, usually have larger correlation. Related data resources were integrated to provide services, which usually can generate more systemic and more comprehensive data sets and even improve reliability and performance of data services. However, these related data sources may have heterogeneities in data formats and storage structures. In addition, there may be man-made semantic confusions of "synonyms" and "homonyms" among them. Therefore, in order to better achieve access and management of related data sources, we can use ontology technology (Rubiolo *et al.*, 2012) to solve the semantic issues in grid computing environment. Global ontology and local ontologies usually need to be built for application grid to deal with

massive data. This paper assumes that the global ontology and local ontologies are established with LAV scheme (McBrien and Poulovassilis, 2003). Global ontology abstracted and synthesized from data sources can be obtained through integrating domain information included in data sources, as well as describes system global view, so it likes a shared vocabulary which built knowledge model for domain waiting to be integrated and provided common semantic description for data integration (Qasim and Khan, 2009). Local ontology is term relative to global ontology. Each data source has an ontology (called local ontology) that describes its data information and establishes mapping relation with it. Local ontology usually establishes relationship with global ontology using some form of mapping. With the mapping relation, queries on global ontology can be decomposed to subqueries to which data sources are corresponding (Zuofu *et al.*, 2011). Various data sources can freely join and quit grid system as grid computing is inherently distributed and dynamic. How to solve the problem of mapping between global ontology and local ontologies caused by dynamic of data sources is a key

---

Corresponding Author: Yonghong Luo, School of Computer and Communication Engineering,
Changsha University of Science and Technology, 960 2nd Section,
Wanjiali South RD, Changsha, Hunan, 410114, China

link for data integration in grid system. So, this study presents a dynamic mapping strategy between global ontology and local ontologies which uses joining mapping algorithm and quitting mapping algorithm respectively to achieve effective mappings between global elements and local elements.

DESCRIPTION OF ONTOLOGY AND MAPPING

In this study, an ontology is defined as a 6-tuple: $O: = <C, P, H^C, H^P, A, I>$ (Pirro and Talia, 2010), where C is set of concepts, P is set of properties, H^C denotes concept hierarchy which associates each concept with its sub-concepts, H^P denotes property hierarchy that associates each property with its sub-properties, A is set of axioms, I is a set that contains all instances of concepts and properties. So, global ontology is represented as $GO = (GC, GP, H^{GC}, H^{GP}, GA, GI)$; local ontology denotes as $LO_i = (C_i, P_i, H^{ci}, H^{Pi}, A_i, I_i)$, i = 1, 2,.... Mappings between global ontology GO and local ontology LO_i is represented as $M_i = <M_i(c), M_i(p)>$, where $M_i(c)$ is set of concept mappings expressed as $M_i(c) = \{<c_G, c_L>|c \in GC, c \in C\}$, $M_i(p)$ is set of property mappings expressed as $M_i(p) = \{<p_G, p_L>|p_G \in GP, p_L \in P_i\}$.

SIMILARITY AGGREGATING

Elements belonging to different ontologies are usually established mappings based on their similarity. However, a certain kind of similarity is difficult to accurately describe the degree of element similarity. So, it is necessary for us to integrate various similarities between elements. In this paper, we assume that name similarity, profile similarity, structure similarity and semantic similarity between elements (Mao *et al.*, 2010) are represented as $Sim_1(e_i, e_j)$, $Sim_2(e_i, e_j)$, $Sim_3(e_i, e_j)$, $Sim_4(e_i, e_j)$ respectively. Then C1_Sim and C2_Sim, as comprehensive similarities between elements, can be computed using the following expressions:

$$C1\_Sim(ei, ej) = \sum_{n=1}^{4} \alpha_n sig(sim_n(e_i, e_j)) \quad (1)$$

$$C2\_Sim(ei, ej) = \sum_{n=1}^{4} \beta_n sim_n(e_i, e_j) \quad (2)$$

where, α means weight assigned to a certain similarity and satisfying $\Sigma^2 \alpha_i = 1$. The computing formula of α is as follows:

$$\alpha_k = \frac{Sim_k(e_i, e_j)}{\sum_{n=1}^{4} Sim_n(e_i, e_j)} \quad (3)$$

The sig in Eq. 1 is sigmoid function (Kros *et al.*, 2006) and defined as follows:

$$sig(x) = \frac{1}{1 + e^{-5(x-a)}} \quad (4)$$

where, a is the central point of sigmoid function. In Eq. 2, β also means weight assigned to a certain similarity and satisfying $\Sigma\beta_i = 1$ too.

Through calculating the values of C1_Sim and C2_Sim with Eq. 1 and 2, respectively, we can integrate C1_Sim and C2_Sim using F-Measure method (Chen *et al.*, 2004), the integration formula is as follows:

$$F(C1\_Sim(ei, ej), C2\_Sim(ei, ej)) = \frac{2 * C1\_Sim(e_i, e_j) * C2\_Sim(e_i, e_j)}{C1\_Sim(e_i, e_j) + C2\_Sim(e_i, e_j)} \quad (5)$$

DYNAMIC MAPPING

Various data sources are likely to dynamically join and quit application grid system in data integration environment based on grid computing. In order to effectively solve the problem of mapping between global ontology and local ontologies in grid, this paper proposes a dynamic mapping strategy, its steps Fig. 1 are as follows:

Step 1: Data sources need to register in resources registry whether they join or quit grid system, then resources registry pass the information of joining or quitting grid system for data sources to ontology manager

Step 2: After ontology manager have received changed information of data sources, ontology manager obtains mapping information between global ontology and corresponding local ontology from mapping storage center if it detected that data source will quit grid system and delivers mapping information of quitting data source to quitting mapping module of dynamic mapper which can carry out quitting processing; ontology manager builds the corresponding local ontology for data source if it detected that data source will join grid system, then delivers global ontology and new local ontology to joining mapping module of dynamic mapper which can create mapping relation between global ontology and joining local ontology
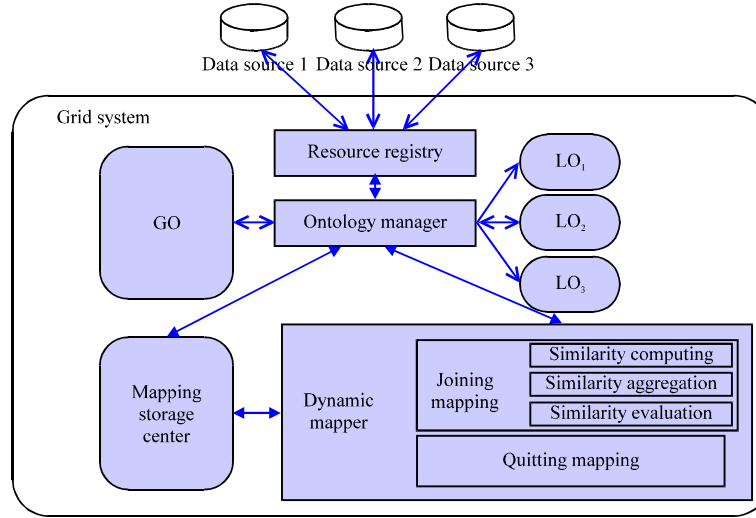
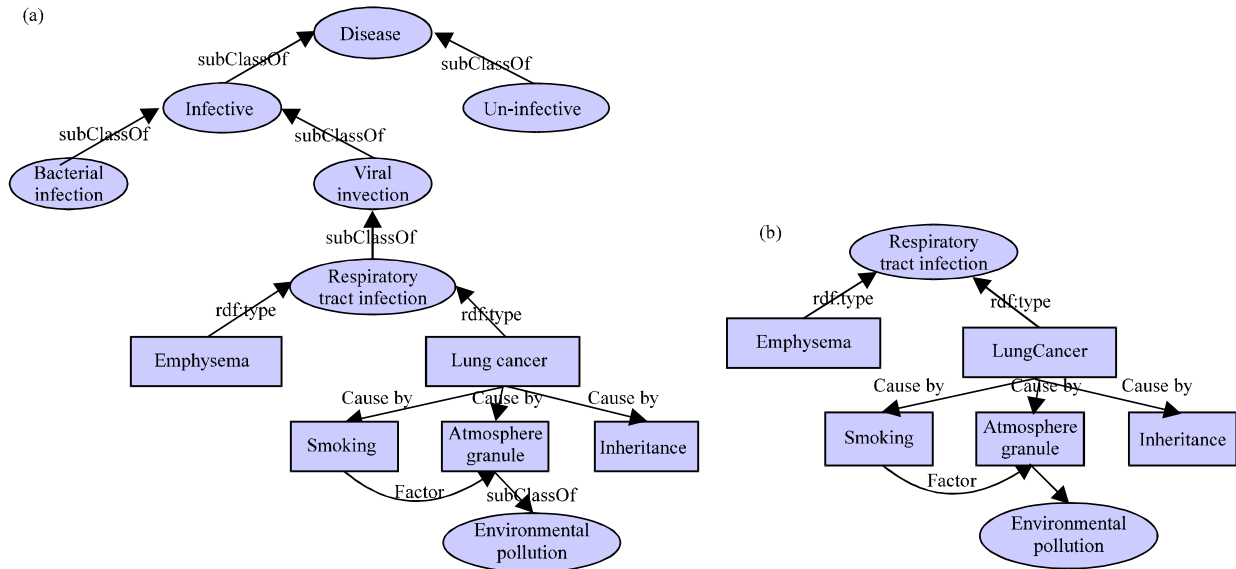Fig. 1: Processing of dynamic mapping

Fig. 2(a-b): Subtree intercepting for joining algorithm, (a) Global ontology tree and (b) Intercepted subtree

Step 3: When all of mapping tasks were completed by dynamic mapper, new mapping relations between global ontology and local ontologies will be stored in mapping storage center

Joining mapping: In this study, global ontology tree is represented as GOT = <GR,GV,GE>, where GR is root node of global ontology tree, GV is set of vertexes belonging to GOT, GE is set of edges belonging to GOT; Local ontology tree denotes as LOT_i = $<LR_i,LV_i,LE_i>$, where i = 1,2,3,....

Definition 1 (Subtree Interception): With the specified node SR$\in$GC as root node, a subtree intercepted from global ontology tree is represented as ST = <SR, SV, SE>, where SR is the root node of ST, SV (SV$\subseteq$GV) is vertex set of ST, SE(SE$\subseteq$GE) is edge set of ST. For example, Fig. 2a is a global ontology tree of Disease, subtree with "Respiratory Tract Infection" as root node intercepted from Disease is shown in Fig. 2b.

Definition 2 (active element set): If elements in global ontology can serve for the current tasks of data

semantic processing in grid, the set of these elements can denote as AES (Active Element Set), namely AES = {AC, AP}, where AC (AC⊆GC) are set of active concepts, AP (AP⊆GP) are set of active properties, SC∩AC = Ø, SP∩AP = Ø.

Definition 3 (suspended element set): If elements in global ontology can't serve for data semantic processing of current tasks in grid due to data sources quit grid system dynamically, the set of these elements can be represented as SES(Suspended Element Set), namely SES = {SC,SP},where SC(SC⊆GC) is set of suspended concepts, SP(SP⊆GP) is set of suspended properties, SC∩AC = Ø, SP∩AP = Ø.

When a new data source joins grid system and builds its local ontology LO_j, the corresponding ontology tree LOT_j firstly needs to be constructed for LO_j. Then, we should compute similarities between LR_j and each concept in GC and find out the c_i which satisfy the condition of max (sim (Lr_j, c_i|i = 1,2,...)). Next, we intercept subtree ST (ST = <c_i, SV, SE>) from GOT with c_i as root node and build mapping pairs between elements of ST and elements of LOT_j in accordance with procedure of similarity computing and similarity integrating. If some local elements in LOT_j can't form mapping with global elements in ST, we will establish mapping for Rest (remaining elements set of LOT_j) and SES again. The detailed algorithm of joining mapping is as follows:

Input: GO, LO_j, λ;//λ is mapping threshold.
Output: mapping element pairs:

Step 1:	Build ontology trees GOT (GOT = <GR, GV, GE>) and LOT_j(LOT_j = <Lr_j, LV, LE>) for global ontology GO and LO_j, respectively
Step 2:	Max_f = F (C1_Sim (Lr_j, gc_1), C2_Sim (Lr_j, gc_1)), RN = gc_1;//RN denotes root node
Step 3:	For each gc_i in GC do
Step 4:	Calculate F (C1_Sim (LR_j, gc_i),C2_Sim (Lr_j, gc_i))
Step 5:	If max_f<F (C1_Sim (Lr_j,gc_i),C2_Sim (Lr_j, gc_i)) then
Step 6:	Max_f = F (C1_Sim (LR_j, gc_i),C2_Sim (LR_j, gc_i)), RN = gc_i
Step 7:	End if
Step 8:	End for
Step 9:	Subtree ST is intercepted from GOT with RN as root node, namely ST = <RN,SV,SE>
Step 10:	For each element lv_i in LV do
Step 11:	Max_sim = F(C1_Sim(lv_i,sv_1),C2_Sim(lv_i,sv_1))
Step 12:	For each element sv_j in SV do
Step 13:	Calculate F(C1_Sim(lv_i,sv_j),C2_Sim(lv_i,sv_j))
Step 14:	if (F(C1_Sim(lv_i,sv_j),C2_Sim(lv_i,sv_j))>max_sim then
Step 15:	Max_sim = F (C1_Sim(lv_i,sv_j),C2_Sim(lv_i,sv_j)) m1 = i, m2 = j;//m1 and m2 identify subscript of candidate mapping elements
Step 16:	End if
Step 17:	End for
Step 18:	If (max_sim> = λ) then
Step 19:	Establish mapping relation for lv_{m1} and sv_{m2}, removing lv_{m1} and sv_{m2} from LV and SV, respectively;
Step 20:	if (sv_{m2} not in AES) then

Step 21:	Activate sv_{m2} in GOT and add it to AES, removing sv_{m2} from SES
Step 22:	End if
Step 23:	End if
Step 24:	End for
Step 25:	if (LV ≠ Ø) then
Step 26:	Obtain Rest set of LV, let LV = Rest and SV = SES
Step 27:	Return step10
Step 28:	End if
Step 29:	Return

Quitting mapping: With the aforementioned definition, we can divide elements in global ontology into AES and SES two categories. When some data source to which Lo_i is the corresponding local ontology quits grid system, all of mapping element pairs <ge_j, le_j> (j = 1,2,...) between GO and LO_i need to be investigated. If Σle_t ($le_t \in LO_k$, k≠i), the mapping relation <ge_j, le_t> was established, ge_j still belongs to AES, or ge_j will be moved to SES from AES. The detailed algorithm of quitting mapping is as follows:

•	**Input:** M_j (j = 1, 2, ...)//M_j denotes the mapping relation set between global ontology GO and local ontology LO_j
•	**Output:** Updated AES and SES

Step 1:	Let AES = {GC, GP}, SES = Ø
Step 2:	If corresponding local ontology for quitting data source is LO_i, then achieving M_i from mapping storage center
Step 3:	For each mapping element pair <$e_{g,m}$,$e_{i,n}$> in M_i do//$e_{g,m} \in$GO, $e_{i,n} \in LO_i$
Step 4:	For every M_j (j≠i) do
Step 5:	If (not $\Sigma e_{j,k}$, ($e_{j,k} \in LO_j$) and $e_{g,m}$ has mapping relation with $e_{j,k}$) then
Step 6:	Move $e_{g,m}$ to SES from AES and suspend $e_{g,m}$ in GOT
Step 7:	End if
Step 8:	End for
Step 9:	End for
Step 10:	Return

EXPERIMENTAL RESULTS AND ANALYSIS

In order to verify effectiveness of dynamic mapping strategy in this study, we carried out dynamic mapping experiments between global ontology and local ontologies in simulated grid environment GridSim. We mainly evaluated two aspects of these experiments: (1) semantic query accuracy and (2) time and efficiency of dynamic mapping. Experimental data was achieved from six data sources of Edas, openConf, Ekaw, Confious, lasted and Linklings in OAEI Conference. We firstly built global ontology (i.e. conference) and local ontologies for these data sources with Protégé tool (Table 1).

Data source can dynamically quit grid system due to inherent particularity of grid environment. In order to ensure accuracy of semantic query, this paper proposed quitting mapping algorithm for data sources in grid system. In the situation that some data sources quitting grid system, we conducted a total of 6 semantic queries.
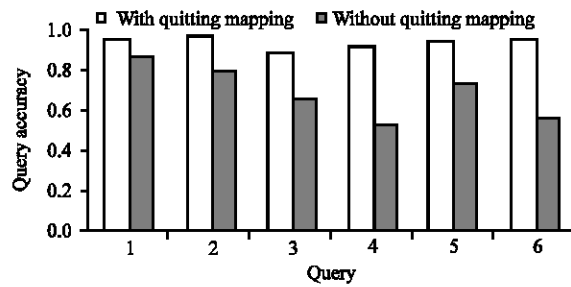
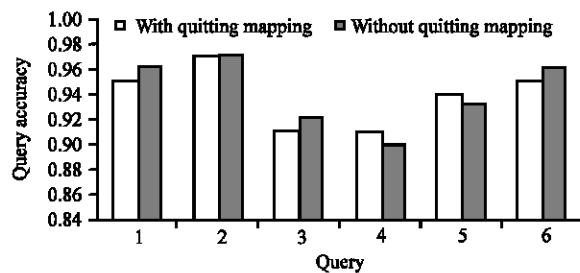Fig. 3: Accuracies comparison of semantic query related to quitting data sources

Fig. 4: Accuracies comparison of semantic query not related to quitting data sources

Table 1: Experimental ontologies description

Ontology Name	No. of concepts	No. of properties
Conference	316	73
Edas	104	20
openConf	62	21
Ekaw	74	18
Confious	57	5
Iasted	140	33
Linklings	37	16

Accuracies of semantic queries related to quitting data sources are shown in Fig. 3. Figure 3 shows that accuracies of semantic queries referred to quitting data sources are greatly affected without processing of quitting mapping while semantic queries with processing of quitting mapping obtain very high accuracy. Accuracies for semantic queries which are not related to quitting data sources are shown in Fig. 4. We can know from Fig. 4 that accuracies of any semantic query are almost the same whether quitting mapping is carried out or not. So, accuracy of any semantic query relies heavily on whether grid system includes the data source referred to the query.

We should build mapping relation between global ontology and its local ontology when a new data source joins grid system in order not to affect data semantic processing. So, this paper proposed joining mapping algorithm for data sources in grid environment. Joining
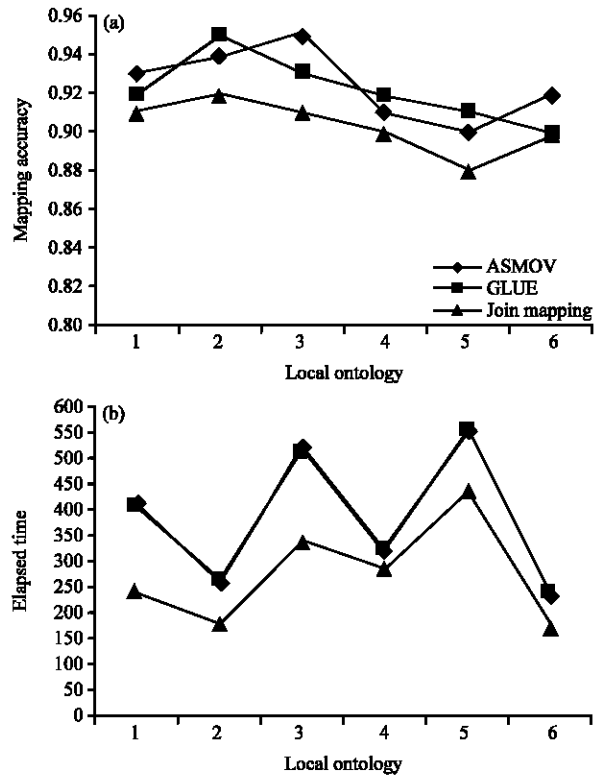

Fig. 5(a-b): Comparison of elapsed time and mapping accuracy for ASMOV, GLUE and Join mapping method

mapping experiments based on 6 data sources were conducted with GLUE (Park and Kim, 2007), ASMOV (Jean-Mary *et al.*, 2009) and joining mapping method presented in this paper. Experimental results on elapsed time and mapping accuracy is shown in Fig. 5. Figure 5 indicates that mapping accuracy of join mapping method is slightly less than GLUE and ASMOV, but its elapsed time is far lower than GLUE and ASMOV. However, task execution time is an important factor for evaluating performance in grid system, so the proposed method is effective.

CONCLUSION AND FUTURE WORK

Various data resources are likely to dynamically join and leave grid system in data integration environment based on grid computing. This paper proposed a dynamical mapping strategy so as to timely establish mapping relation for ontologies in grid system. The strategy solves dynamic mappings between global ontology and local ontologies with joining and quitting algorithms. The experimental results show that algorithms presented in this paper are helpful to data semantic

processing, while query accuracy and mapping accuracy obtained through these algorithms remain to be improved. In the future, we will study expansion optimization of user semantic query based on work achieved in this study.

ACKNOWLEDGMENT

REFERENCES

Chen, T.Y., F.C. Kuo and R. Merkel, 2004. On the statistical properties of the f-measure. Proceedings of the 4th International Conference on Quality Software, September 8-9, 2004, Braunschweig, Germany, pp: 146-153.

Foster, I. and C. Kesselman, 1999. The Grid: Blueprint for a Future Computing Infrastructure. Morgan Kaufmann Publishers, San Francisco, CA, USA.

Jean-Mary, Y.R., E.P. Shironoshita and M.R. Kabuka, 2009. Ontology matching with semantic verification. Web Semant., 7: 235-251.

Kros, J.F., M. Lin and M.L. Brown, 2006. Effects of the neural network s-sigmoid function on KDD in the presence of imprecise data. Comput. Oper. Res., 33: 3136-3149.

Mao, M., Y. Peng and M. Spring, 2010. An adaptive ontology mapping approach with neural network based constraint satisfaction. Web Semant., 8: 14-25.

McBrien, P. and A. Poulovassilis, 2003. Data integration by bi-directional schema transformation rules. Proceedings of the 19th International Conference on Data Engineering, March 5-8, 2003, Bangalore, India, pp: 227-238.

Park, S. and W. Kim, 2007. Ontology mapping between heterogeneous product taxonomies in an electronic commerce environment. Int. J. Electron. Commerce, 12: 69-87.

Pirro, G. and D. Talia, 2010. UFOme: An ontology mapping system with strategy prediction capabilities. Data Knowledge Eng., 69: 444-471.

Qasim, I. and S. Khan, 2009. Semantic mapping between global and source ontology using wordnet. Proceedings of the 2nd International Conference on Computer, Control and Communication, February 17-18, 2009, Karachi, Pakistan, pp: 1-5.

Rubiolo, M., M.L. Caliusco, G. Stegmayer, M. Coronel and M.G. Fabrizi, 2012. Knowledge discovery through ontology matching: An approach based on an Artificial Neural Network model. Inf. Sci., 194: 107-119.

Zuofu, Z., X. Chao amd G. Hongmei, 2011. Local ontology based semantic query decomposition algorithm. Proceedings of the International Conference on Internet Technology and Applications, August 16-18, 2011, Wuhan, China, pp: 1-4.