# A consistency maintenance project independent relationship of distributed data copy for Supply Chain Management

Jui-Wen Hung
Department of Information and Management
Ling Tung University
No. 1, Ling Tung Rd., Nantun District, Taichung City 408, Taiwan (R.O.C.)
harng@mail.ltu.edu.tw

*Abstract:* - With global transnational enterprise layout trend, data consistency convergence study is a key of improving competitions and Bullwhip Effect problem in Supply Chain. This paper proposed a specific ODMS dependent replicas correctness maintenance engineering project. A resolution model provides adaptive adjustability update routing policies in hierarchical enterprise database. On the basis of real time updating process framework, several models are built. Different value added update service schema was also analyzed. A comprehensive update of the enterprise data that resident in internal heterogeneity of supply chain knowledge base existed partially dependency relationships with specified ODMS. With this paper on a comprehensive update mechanism of the organization in both dependencies spread copy of the data using flexibility and globally consistence maintenance mechanisms to reach enterprise internal and external data consistency.

*Key-Words:* - Internet Data Base Original Source (IDBOS), Region Data Content Proxy (RDCP), ODMS, portal Agent, global consistency, Supply Chain Management

## 1 Introduction

Supply chain management (SCM) aims to efficiently integrate suppliers, manufacturers, warehouses, and retailers, not merely to ensure that merchandise is produced and distributed in the appropriate quantities, to the right locations, and at the right time, but also to minimize system wide costs while satisfying customer requirements. Many enterprises recognize that rapid information exchange among supply chain members is essential. A novel AI application platform was proposed which consists of active, real time, automation, and global routing technologies. When ODMS object was triggered by update event, all replicas with the specific ODMS dependent relationship could be automatically and real time updated by the cooperation agent [1]. Developing Knowledge Management Systems is a complicated task since it is necessary to take into account how the knowledge is generated, how it can be distributed in order to reuse it and other aspects related to the knowledge flows [8]. For example, active updating pushed by real time, periodic, reservation, and queuing with priority/degradation/preemption etc. The result can expand wide area enterprise data replicas contained ODMS dependency relationships with global consistency [4] maintenance abilities.

Study of this paper is divided into two stages: first stage is to propose a source of knowledge base global consistency project through the AI Agent trigger, web service drivers, the latest version of the knowledge copy are sent to the full requirements of the various enterprise portal-agents from the recent Region Data Content Proxy in order. With a variety of flexibility and customization of pushing the message broadcast mechanisms, including: immediate, timing, appointments, (queue + priority), (queue priority + demotion), etc. Different access permissions were assigned to the portal agent with different levels of knowledge of information access in the nearest regional data content of a proxy server. The second stage is on, when the roots organizations portal (Enterprise Root Portal; ERP) agents to obtain a copy of the latest updating of data, through the portal implementation of the internal ripple-propagation data copies of routing algorithm. This updated copy routing to the next node of the Topography of the portal in Extranet or Intranet area, complete the data message. When ODMS copy routing through an enterprise portal. The portal agent would approach the pipeline process, automatic and adaptive translation data into all kinds of ontology structures format. Topography of enterprise automation reached within the organization, both dependencies distributed copies of the Global Data consistency (global consistency) to update the target. The results of this study, hierarchical enterprise can be very flexible to update

relate information to inside firm and outside supply chain partners.

# 2 Related Researches

### 2.1 (Original Data Message Source, ODMS)

If replicas having partial dependency relationships with specific ODMS object inside transnational enterprises organization can't maintain mutual consistency, the following problems will be happened.

1. The Bullwhip Effect will be occurred in e-business supply chain operation.
2. Replicas consuming application can't make sure the newest version of access copies and its owner. It will also make repetitive replicas updated operations and affect the replicas reuse freshly.
3. It is easy for global transnational enterprise organization to occur horizontal information that lacks harmony.
4. The ODMS can't control its replicas updating flow, transaction, and usage mining results and it can't support different priority and value-added of dependent replicas updating services.
5. The dependent replicas usage mining results can't be sent to ODMS to make adaptive dependency replicas updating route policies and value added updating services.

### 2.2 Supply Chain Management

A Supply chain is a worldwide network of organizations and their associated activities that work together to produce value for the customer [12]. To operate efficiently, supply chain functions must work in a tightly coordinated manner. Jankowska [4] noted that supply chain integration is difficult for two primary reasons:

- Different supply chain facilities may have different, possibly conflicting, objectives.
- Supply chains are dynamic systems that evolve over time.

Croom [3] gathered various relevant definitions of supply chain management. Finally, this investigation concludes by discussing some issues related to supply chain management. These issues are:

- Cost reduction: Enterprises typically strive to reduce production costs, including time, stock, human costs, and so on.
- Information sharing: Enterprises must obtain more information, and thus can design appropriate strategies for enterprise benefit.
- Cross-organization integration: Management in a global society no longer involves striving in isolation. Therefore, enterprises must conduct cross organization integration.

There are three problem must be solved in the above issues. The first is vertical information real time updating and propagation between medical Enterprise and Home-based patients with chronic diseases. The second is horizontal information real time updating and propagation between global transnational enterprise organizations. The third is satisfied with above two points under different kinds of protocols and network topologies. A novel application platform was proposed and the 「replica recursive routing model」 are built within a multistage and hierarchical global transnational enterprise portal.

### 2.3 Agent

Software agents, which now are very widespread, were first used several years ago to filter information, match individuals with similar interests, and automate repetitive behavior [10]. Agents are computer systems with numerous important capabilities. These capabilities are summarized and described below:

- Autonomy [4]: Users do not need to indicate how to work. Agents can finish task independently.
- Adaptive [1]: Agents can judge environmental change and adjust their task actively. Such agents generally have domain knowledge, and thus can easily identify ideal solutions based on the real conditions.
- Anticipatory [1]: Agents should not only predict user information through task or interaction with environment but also should alert users and react appropriately.
- Cooperation [4]: Agents can communicate with other agents by completing specific tasks together.
- Trustworthy [1]: An agent should satisfy the needs of user by using reliable meth ODS to get user's trust.

Agents can adapt to environmental changes dynamically, and can model variation of management behavior among different managers [15]. Therefore, this work proposes to construct, using agent technology, and a supply chain communication and negotiation mechanism for reducing purchase cost. A multi-agent approach has been applied to study the framework of supply chain in network economy [9]. Multi-agent architecture is suitable to web services integration [1].

### 2.4 Replicas mutual consistency solutions

Today's global distributed replicas consistent solution have been proposed using data integration middleware techniques like xml-based integration middleware, xml-based data integration platform and xml-powered integration middleware [9]. All these data integration system that enables enterprises to rapidly build web services and applications that can query multiple, disparate data sources and provides a unified result. Web services are an efficient way to

implement Multi-agent Systems (MAS) on the Internet and such systems are particularly adequate for developing and deploying cooperative services [13].

If distributed replicas with partial dependency relationships with specific OPDS object keep mutual inconsistency, the following problems will be happened.

1. The Bullwhip Effect will be occurred in GSCM operation.
2. Enterprise storages can't make sure the newest version copies.
3. Enterprise knowledge base will occur horizontal information lacks harmony.
4. If OPDS site can't control its replicas updating flow and usage mining results, it will not be able to support different priority and value added updating services.
5. The dependent replicas usage mining results can't be sent to OPDS site to make adaptive updating route policies and value added updating services.

# 3 ODMS Model Descriptions

## 3.1 Concept Model for flexibility update copy with ODMS dependencies

In this paper, a novel framework was proposed which consists of active, real time, automation, and global routing updating replica from ODMS domain set. See in figure 1. When ODMS object occurred insert/update events, according to subscribe lists. New copies will push to all global enterprise portal which contained dependency replicas. The enterprise portal server then make one of the choice about discard/ rerouting to lower hierarchy portal/automatic pipeline processing and schema transformation. All wide area ODMS dependent replicas consistency and correctness can be automatic maintenance. Enterprise Intranet application system need not change any source code.

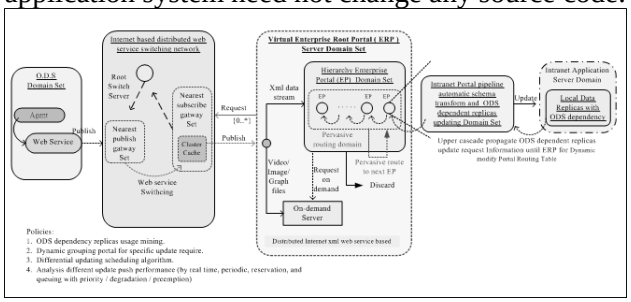



Fig. 1 Concept Model

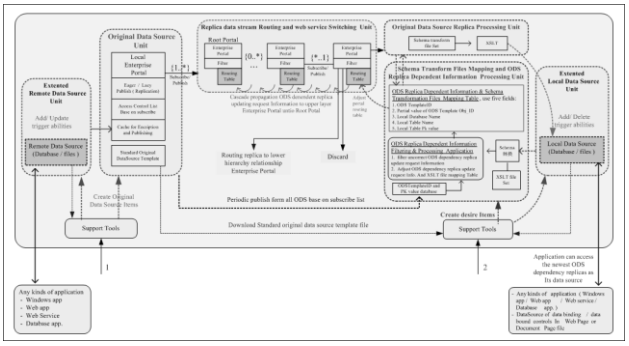



Fig. 2 Infrastructure Model

## 3.2 Intranet and Extranet ODMS Replicas Operation Model

Intranet replicas operation model have three kinds of application. In figure2, replicas with partial specific remote ODMS dependency relationships were denoted by (a). Replicas with ODMS role in extranet enterprise supply chain was denoted by (b). Replicas with partial dependency with a specific self-enterprise intranet ODMS was denoted by (c).

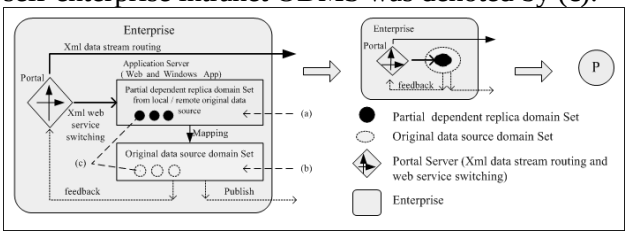



Fig. 3 dependency replicas operation model in Enterprise

The replica consistency problem with local/remote ODMS dependency relationships existed in global transnational enterprise organizations and supply chain operations. In Figure 4, show their dataflow in enterprise intranet and extranet supply chain environment.

Enterprise Portal face ODMS dependency replicas flow have three applications formulate, as follows:

- Apply in enterprise extranet SCM, like route (a):
  (a) . E5, b3, r1, r3, {c7|c8}, {E7|E8}, {b6|b9}, r1, r3
- For redirect updating replica to other hierarchy enterprise, the routing function must be built in., like route (b):
  (b). E5, b3, r1, r2, {c1|c2|c3|c4}, {E1| E2| E3| E4 }, {b1|b2|b4|b7}, r1, r2
- For synchronize trigger multiple EC supply chain and satisfy horizontal replicas consistency, the feedback intranet ODMS updating copy must be built in, like route (c):
  (c). E8, b9, r1, {{r2 → same above (a)} | {r3 → same above (b)}}

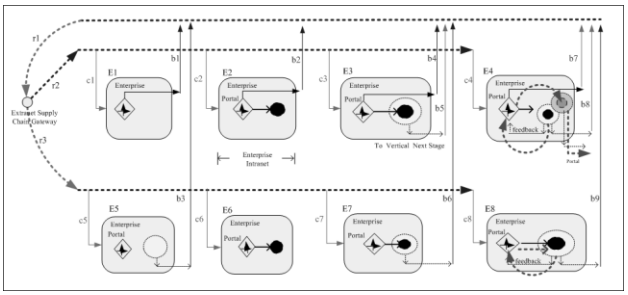


Fig. 4 ODMS replicas In Enterprise Intranet and Extranet Supply Chain

### 3.3 variety of flexibility and customization of the ODMS publish mechanism

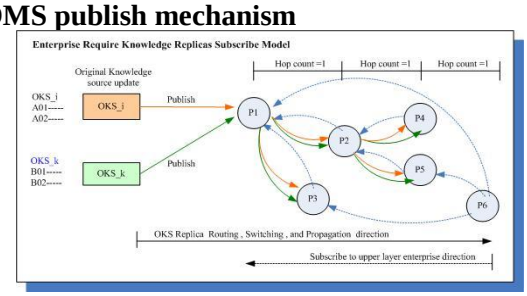



Fig. 5 ODMS subscribe and ripple propagation message chain

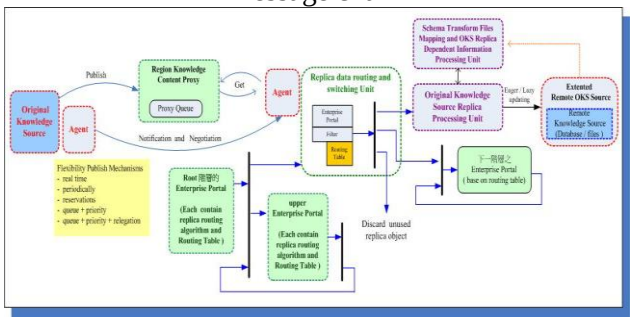



Fig. 6 ODMS subscribe and ripple propagation message chain

### 3.4 portal agent pipeline conversion data ontology messages

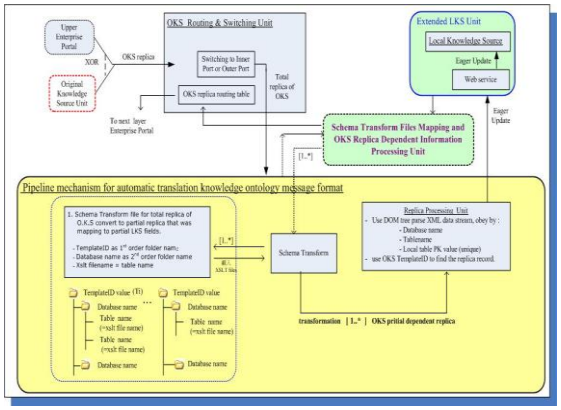



Fig. 7 pipeline conversion data ontology messages

### 3.5 Use case view and Sequential diagram

In figure 8, describe the use case view. The actor's functions of moving describes as follow.

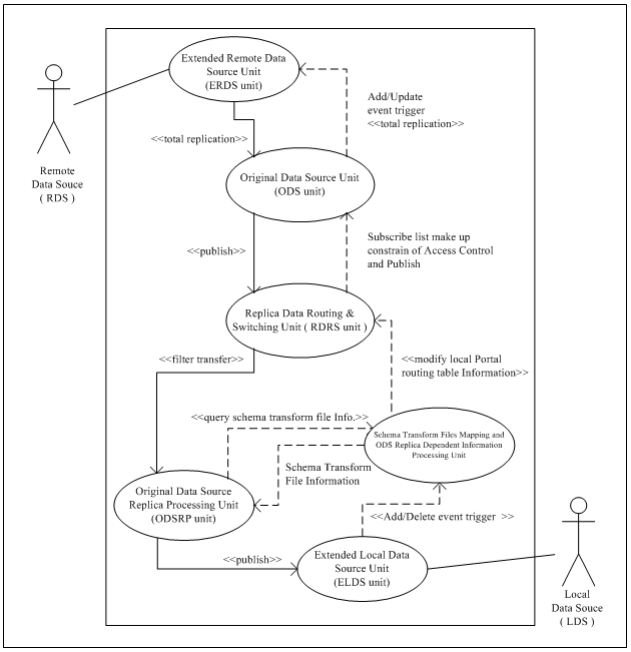



Fig. 8 the use case view

- Remote Knowledge Source (RDS)
  1. Actor's Function
  2. To provide a copy of Data to the Local / Remote client applications all use
  3. Accept the Local / Remote client applications all Data Manipulate (Insert, Delete, Update, Query and other operations)
  4. Accept the Local / Remote Client Database Middleware system of links and query
- Local Knowledge Source (LDS)
  1. With the latest Data (contains a dependency on a copy of the original sources of information) to the Local / Remote client applications all use
  2. Accept the Local / Remote client applications all Data Manipulate (Insert, Delete, Update, Query and other operations)

  Accept the Local / Remote Client Database Middleware system of links and query

Each use case situation detail describe as follow tables.

Table 1. ERDS unit

| Use Case Name | Extended Remote Data Source Unit (ERDS unit) |
|---|---|
| Description | Wrapper from RDS unit, and Improve the unit with insert / update trigger abilities depends on ODS unit needs. When ERDS meet add/ update original data source object event trigger, then total replicate data object to ODS unit cache it |
| Trigger | Remote System application |

| | |
|---|---|
| condition | trigger by Remote Data Source had hanged (Insert / Update Event) |
| Pre-condition | Wrapper from RDS unit, and Improve the unit with insert / update trigger abilities depends on ODS unit needs |
| Message | Data insert / update message |
| Result | Transform into standard original data source schema and total replication |
| Post-condition | Finish insert or update into ODS unit |
| Flow | 1. Create database structure and process mechanism for become RDS of ODS (Extended RDS unit and ODS unit)<br>2. Receive data trigger (Insert/Update Event) from RDS of Remote System application<br>3. Transform data trigged into standard original data source schema<br>4. Replication trigged data into ODS unit |

Table 2. ODS unit

| Use Case Name | Original Data Source Unit (ODS unit) |
|---|---|
| Description | 1. Define standard Original Data Source Template for download.<br>2. Cache data replication from ERDS unit<br>3. Receive replication require subscribe from RDRS unit and ODSRP unit<br>4. data Encryption, Eager and Lazy Replication Mechanism for Publish |
| Trigger condition | Trigger from Extended RDS unit(Insert / Update event) |
| Pre-condition | Create data format and process mechanism for become RDS from ODS (Extended RDS unit and Original Data Source unit) |
| Message | Total replicate data from ERDS unit |
| Result | Publish ODS's replica according from RDRS unit and ODSRP unit that decency require |
| Post-condition | Finish ODS replica publish process |
| Flow | 1. Create data format and process mechanism for become RDS from ODS (Extended RDS unit and Original Data Source unit)<br>2. Receive replica data from ERDS unit and cache it<br>3. Encryption replica cache data<br>4. Publish cache replica data according from RDRS unit and ODSRP unit that decency require |

Table 3. RDRS unit

| Use Case Name | Replica Data Routing and Switching Unit (RDRS unit) |
|---|---|
| Description | 1. Exchange ODS replica route to Enterprise Portal of ODSRP unit depends on Routing Table Information and Routing algorithm<br>2. Manager Enterprise Portal add / exit RDRS unit issues<br>3. Receive lower Enterprise Portal of hierarchy and finish next two objects depends on replica require: (a) update Enterprise Portal data routing table. (b) Trans replica requirement to upper Enterprise Portal of hierarchy by RDRS unit<br>4. In RDRS unit, Each Enterprise Portal can propose subscribe replica require to ODS unit<br>5. Enterprise Portal of RDRS unit has replica filter ability, it can discard / routing to next enterprise portal / transfer into ODSRP unit |
| Trigger condition | 1. Enterprise Portal receive from Extended LDS unit dependency replica require message, update route table, and lead to dependency replica update data flow from upstream to downstream<br>2. Receive publish replica from, lead to dependency replica update data flow from upstream to downstream after RDRS unit Filter |

| | |
|---|---|
| Pre-condition | Each Enterprise Portal in RDRS unit has suitable data structure and process mechanism, and implement process replica route algorithm |
| Message | Pass replica require message from Extended LDS unit |
| Result | 1. Enterprise Portal receive from Extended LDS unit dependency replica require information and update route table, produce replica require update flow from downstream to upstream.<br>2. From upstream Portal filter or publish replica by ODS unit, through RDRS unit Filter and lead to replica route exchange from upstream to downstream |
| Post-condition | Finish dependency replica route data exchange |
| Flow | 1. Receive dependency replica require information from Extended LDS unit, update route table, and drive to replica update information from downstream to upstream<br>2. In RDRS unit, Enterprise Portal propose replica require subscription depend by ODS unit<br>3. Exchange replica of origin data to Enterprise Portal or ODSRP unit depends on Routing Table Information and Routing algorithm (Enterprise Portal of RDRS unit has replica filter ability, it can discard / routing to next enterprise portal / transfer into ODSRP unit) |

Table 4. ODSRP unit

| Use Case Name | Original Data Source Replica Processing Unit (ODSRP unit) |
|---|---|
| Description | 1. Receive origin data dependency replica subscript from Extended Local Data Source unit<br>2. Execute Description / Schema Transformation / Caching dependency replica passed from RDRS unit Enterprise Portal |
| Trigger condition | Receive origin dependency replica routed from RDRS unit |
| Pre-condition | Create suitable data structure and process mechanism for Original Data Source Replica Processing Unit (schema transformer / Caching and Eager or Lazy replication mechanism) |
| Message | Receive origin dependency data replica( Encryption) route from(RDRS unit) |
| Result | Execute Description / Schema Transformation / Caching. / Eager / Lazy Replicate passed from RDRS unit Enterprise Portal, and pass to ELDS unit based on subscribe list |
| Post-condition | Finish Replica Processing |
| Flow | 1. Create suitable data structure and process mechanism for Original Data Source Replica Processing Unit and Extended LDS unit<br>2. Receive origin dependency data replica subscribe from Extended Local Data Source unit<br>3. Execute dependency replica Description / Schema Transformation / Caching passed from RDRS unit Enterprise Portal |

Table 5. STFM & ODSRDIP unit

| Use Case Name | Schema Transform Files Mapping and ODS Replica Dependent Information Processing Unit (STFM&ODSRDIP unit) |
|---|---|
| Description | 1 Cache ODS Template ID and Primary key push by ODS<br>2 Match ODS template for ELDS propose dependency replica information and save some information(database name/table name/local table primary key value/action flag)<br>3 Filter step 1 effective information and step 2 non-correct math information for ODSRP unit query effective transform data format<br>4 Pass correct ODS dependency replica to Portal, and update route table |

| | |
|---|---|
| | 5 Receive ODSRP unit query effective transform format |
| Trigger condition | Receive ELDS add/delete trigger for ODS dependency replica |
| Pre-condition | 1. Create suitable data structure and process mechanism for Schema Transform Files Mapping and ODS Replica Dependent Information Processing Unit.<br>2. Cache all push ODS Template ID and primary key values |
| Message | Receive replica for add/delete trigger from ELDS to ODS |
| Result | Match ELDS propose dependency replica require information and download from enterprise's ODS template. Add some important information (database name/table name/local table primary key value/action flag). Using effective information pushed from ODS and filters non-correct information of match. It can let ODSRP unit query effective data transform format |
| Post-condition | Finish Transform Files Mapping and ODS Replica Dependent Information Processing |
| Flow | 1 Create suitable data structure and process mechanism for Schema Transform Files Mapping and ODS Replica Dependent Information Processing Unit<br>2 Match ELDS propose dependency replica require information and enterprise download template from OSD and Add some information (database name/table name/local table primary key value/action flag), Filter non-correct information in cache pushed from ODS's effective. Let ODSRP unit can query effective trans format<br>3 Pass correct ODS dependency replica require information to Portal, and update route table and serial update active<br>4 Receive ODSRP unit query effective trans format |

Table 6. ELDS unit

| Use Case Name | Extended Local Data Source Unit (ELDS unit) |
|---|---|
| Description | 1. Wrapper from LDS unit, and Improve the unit with add/delete trigger abilities。<br>2. Receive message from LDS unit<br>3. When ELDS meet add/ delete information from origin data replica event, then announce ELDS unit update route table from Enterprise Portal of RDRS unit<br>4. Subscribe origin data dependency replica require for ODSRP unit |
| Trigger condition | Trigger From Local System application manipulate Local Data Source (Insert/Delete) event. When ELDS meet add/ delete event for origin data replica, then announce ELDS unit update route table for Enterprise Portal of RDRS unit |
| Pre-condition | Wrapper from LDS unit, and Improve the unit with add/delete trigger abilities。 |
| Message | Receive new replica data from ODSRP unit publish |
| Result | Update latest dependency replica to LDS, and provide system application newest replica data |
| Post-condition | Finish update latest dependency replica to LDS |
| Flow | 1. Create suitable data structure and process machismo for Extended LDS unit<br>2. Receive from Local System application manipulate to Local Data Source (Insert / Delete Event)<br>3. When ELDS meet add/ delete origin data source replica event, then announce ELDS unit update route table for Enterprise Portal of RDRS unit<br>4. Receive from ODSRP unit publish latest replica data, update to LDS, and provide system application newest replica data |

In figure 9, describe the UML sequential diagram about the concept model system object.

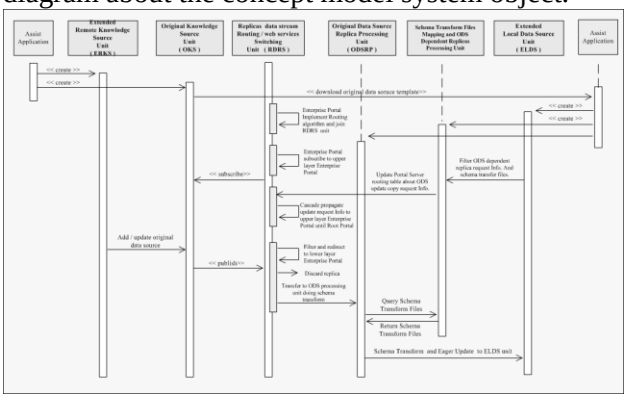


Fig. 9 System object sequential diagram

# 4. ODS updating stream routing algorithm

Routing algorithm shown in Figure8 was built into each enterprise portal for assisting the xml data stream route path choosing.

## 4.1 Basic principle

1. ODS dependency replica update request information always maintained by enterprise portal
2. Information Included UUID value, PK value , Hop count, web service client etc. need to log in portal routing table from lower hierarchy enterprise request issues.
3. Portal's replica update routing information can come from intranet and extranet enterprise.
4. When a ODS update replica publish to Portal, it will discard, send to processing unit, or reroute to lower portal depend on routing table check results.
5. For each route request from lower hierarchy portal will add a HC value to routing table's Hop Count value field.

## 4.2 Routing algorithms

Apply the rules to each route request. The ASF is sent from lower layer enterprise portal by web service. In distributed and asynchronous algorithm, each node will transfer its distance vector copy to all adjacencies at random time.

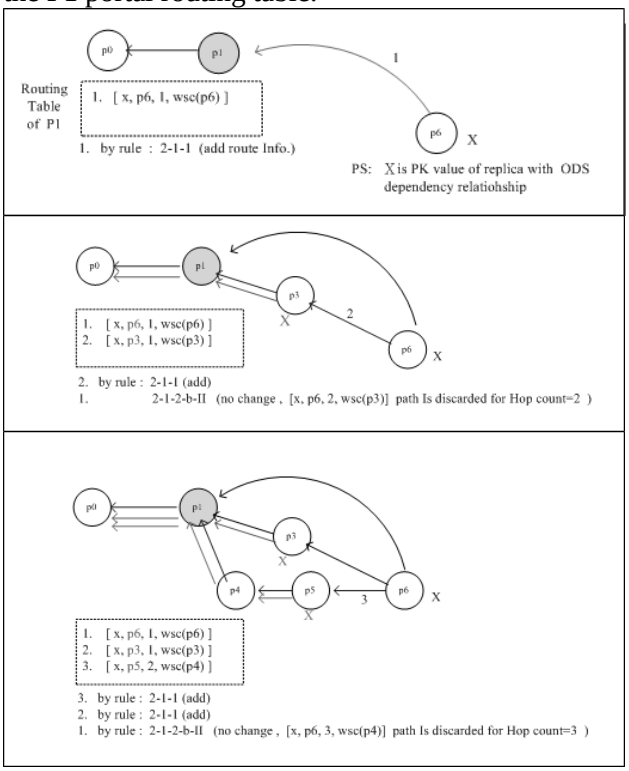
```
If (ASF==add) {
 // (rule.2-1) : ASF == add
  If (route request info.(UUID、PK、DPTD) sent from lower portal does not exist){
     //(rule.2-1-1)
     add this information into RT and upper cascade propagate this add event registry}
  Else If (route request info (WSC_port) identical){
        //(rule. 2-1-2-a)
        replace with the new route request info. and cascade propagate to upper portal when
        TTL trigger.}
  Else If (route request info (WSC_port) not identical){
       //(rule. 2-1-2-b)
       If (new HC value < RT's HC){
        //(rule. 2-1-2-b-I)
        replace with the new route request info. and cascade call back to lower hierarchy portal
        to disable the route with the same DPID when its WSC_port <> null }
       Else {
        //(rule. 2-1-2-b-II)
        Local portal do nothing then cascade call back to lower hierarchy portal to disable the
        route with the same DPID when its WSC_port <> null }

 }
Else {
// (rule.2-2) : ASF == delete
  If (route request info does not exist in RT){
   //(rule. 2-2-1)
   Local portal do nothing }
  Else {
   //(rule. 2-2-2)
    Find and delete the record in RT, and upper cascade propagate delete event registry }
 }
Recursive()
```

Fig 10. Routing algorithm

Table 7. Routing table fields' definition

| Routing Table (RT) fields | Description |
|---|---|
| action state flag (ASF) | denote when add/delete OKMS dependent replicas in enterprise LKS |
| OKMS template ID (UUID) | OKMS template identify defined by OKMS site |
| OKMS replica Primary key value(PK) | instance of specific OKMS template |
| local connect / remote connect (LC / RC) | dependency replica update request issued by local enterprise or lower level enterprise portal |
| Destination Portal ID (DPID) | original enterprise Identify that issues the update request |
| Hop Count value (HC) | the distance between the enterprise contained the newest routing information and request enterprise |
| WSC_port | call next stop portal web service |

The following, show as Figure 11, is a demonstration for using above algorithm to adjust the P1 portal routing table.



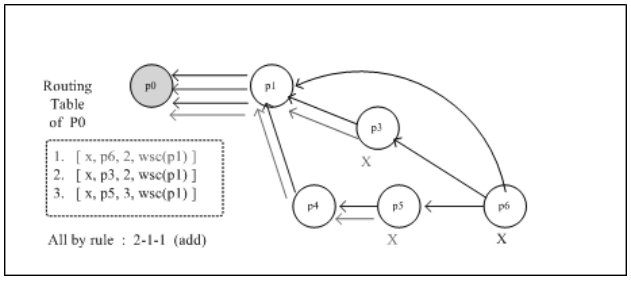



Fig. 11 One example of presented algorithm

Figure 12, 13, 14 is a demonstration for enterprise portal add/change/delete its position issues.

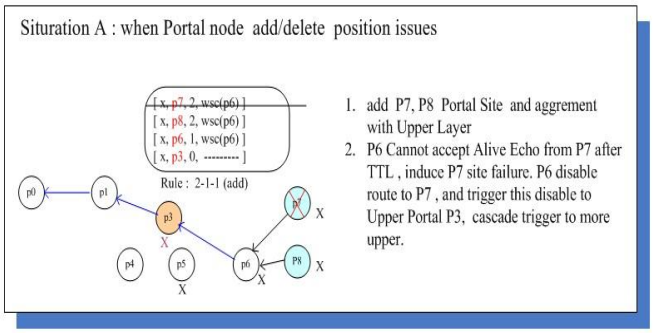



Fig. 12 Demonstration when portal node add/delete its position issues

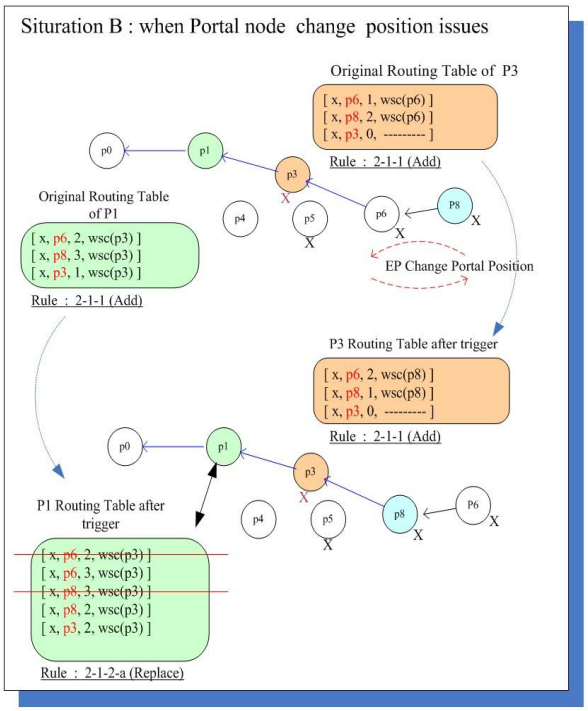



Fig. 13 Demonstration when portal node changes its position issues

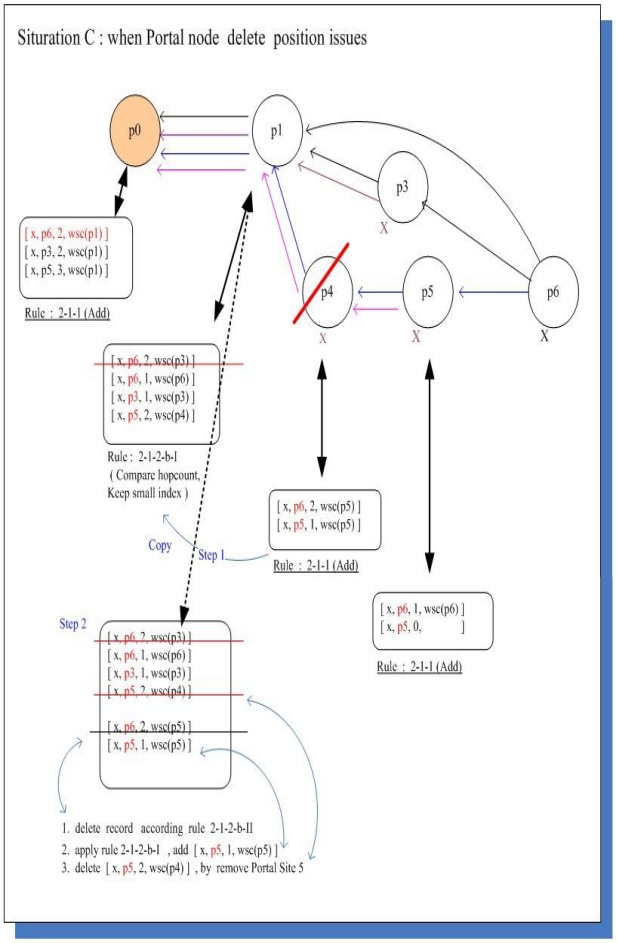



Fig. 14 Demonstration when portal node deletes its position issues

# 5. Estimate of routing algorithms for distributed global consistency

In this section we estimate three kinds of distributed consistency schemas applied in hierarchical and distributed storage environment like schools. 1. P2P push based eager replica updating. 2. Proxy pull based lazy replica updating. 3. ODS push based and active web service propagate routing algorithm for global consistency.

## 5.1 Simulation goal

All administrative information on the hard disk containing the ODS copies of partial dependency set. Through a set of comprehensive, proactive and real-time update mechanism can achieve consistency of the global convergence goals. So that all the schools' decentralized system, will no longer have access to inconsistent or outdated information.

## 5.2 The scope of simulation

The scope of simulation is based on Ling Tung university campus administrative group, access the information on disk which was part of dependency with Personnel part ODS copy. At present, for improving the application performance by the latest copy, accuracy and timeliness toward 2 directions. Including:

1. Replicas of time and space to improve regional exchange of information on the frequency

problem.

2. Accuracy and real-time upgrade distributed dependency copy.

The following will discuss 3 kinds' global consistency models for evaluating and comparing the effectiveness of its operations.

*(a) P2P push based eager replica updating.*

The uses of P2P subscribe/publish eager push mode send updated copy of the personnel changes to subscribed agency. The advantage is immediately, one-way dissemination to all registered administrative units. The shortcomings is the Personnel must to set up individual Peer to Peer's relations subscribe lists. Personnel must also be effective at any time to verify the transmission list of groups available. How to ensure to meet the goal of global convergence consistency is the most difficult. The operation is shown in Figure 15.

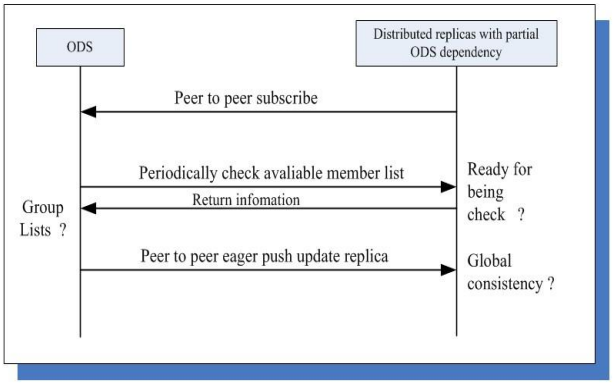



Fig. 15 P2P push based eager replica updating

*(b) Proxy pull based lazy replica updating*

Personnel sent the staff-to-date update copy to school proxy. If all administrative units of application required the ODS dependency copy must through pull and data integration mechanism way. The advantage of lazy changes is to reduce non-essential message conversion and the time to send a message. The disadvantage that proxy server is a single point of failure risks. The operation is shown in Figure 16.

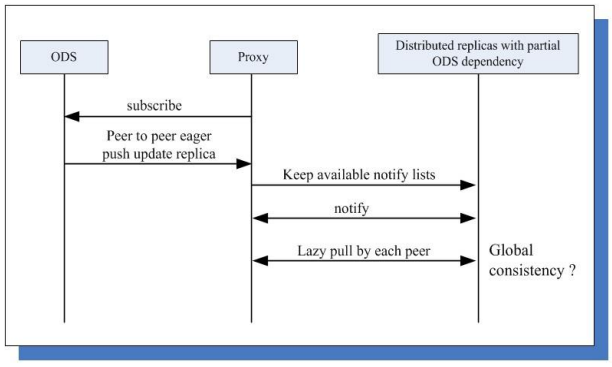



Fig. 16 Proxy pull based lazy replica updating

*(c) ODS push based and active web service propagate routing algorithm*

The goal of routing algorithm proposed is using Subscribe/Publish and active XML web service propagates routing Mechanism for global consistency. In a comprehensive, active and hierarchical automation ripple propagation way to reach global consistency of dependency information copy in school. All administrative units' application is able to integrate information at the lowest cost. Advantage: 1. No longer based on individual applications repeat investment in different database middleware system. 2. There is no need to periodically pull to obtain a copy of dependency-to-date information. For the shortcomings: 1. Root portal and business units to be implemented with replicas routing algorithm. 2. Enterprise portals have to implement ODS update replica schema transformation. The operation is shown in Figure 17.

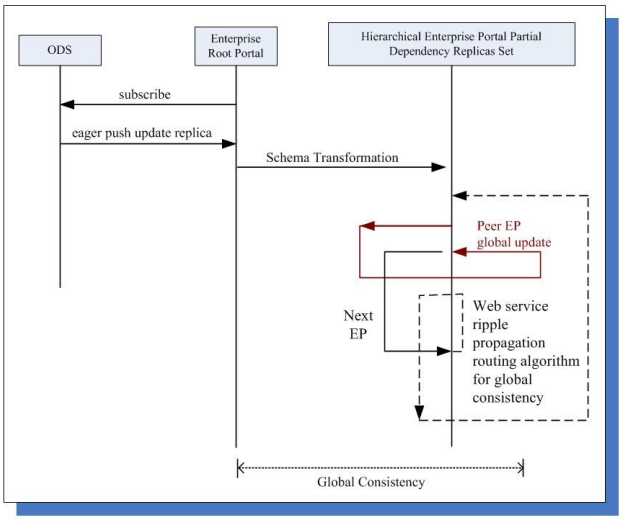



Fig. 17 ODS push based and active web service propagate routing algorithm

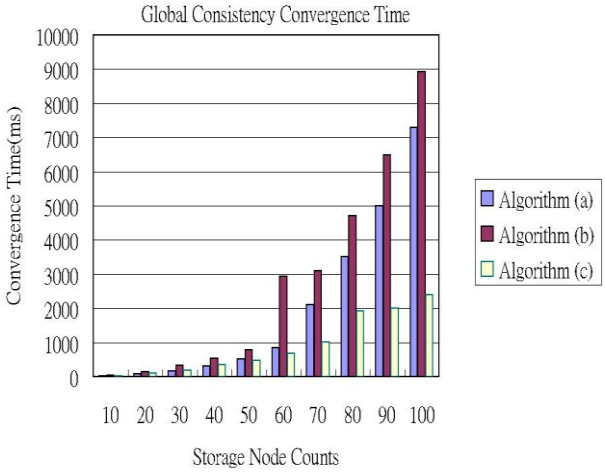



Fig. 18 Comparison (a) (b) (c) of different algorithms on global consistency convergence time

In Figure 18, assume in measure the global consistency convergence time is ignored the following metric including processing delay,

queuing delay, transmission delay. There are considered the storage replica updating delay and network propagation delay. In practically, then transmission delay is usually between several μs. and ms.

# 6. Conclusions

The paper proposed a specific ODMS dependent replicas wide area correctness maintenance engineering project - 「A resolution model provide adaptive adjustable update routing policies, update reserve priority ,and value-added updating services based on usage mining results of replicas with specific ODMS dependency relationships」. It's mainly contribution as follows.

1. Well define internet knowledge base original source architecture that Agent cooperation with web service publishes the latest data changes copy of message.
2. Different value added update service schema was also analyzed. For example, active updating pushed by real time, periodic, reservation, and queuing with priority / degradation / preemption etc.
3. Proposed ODMS update copy cascade routing algorithm suitable for enterprise with hierarchy topologies.
4. Proposed a pipeline process in enterprise portal for translation knowledge update copy into adaptive ontology message format.
5. The ODS dependency replicas usage mining rate is a function of dependency replica operation response time and turnaround time. The updating route policy scheduling rate is a function of usage mining and reservation priority. The production rate of activity represented as a time variable. The cost and limitation for web service routing the interval time between the portal activities are also considered.

With these models, optimization of the correct copy inputs and switching time of each activity in a recursive routing are calculated to obtain the minimum total cost of the project. The result can expand wide area enterprise replicas contained ODMS dependency relationships with real time wide area correctness and consistency maintenance abilities.

# References


[1]. M. Barbuceaunu, R. Teigen, and M.S. Fox," Agent Based Design and Simulation of Supply Chain Systems," *WET-ICE '97, IEEE Computer Society Press*, pp. 36-41, 1997.

[2]. M. Brahimi, M. Boufaïda and L. Seinturier, "Integrating Web Services within Cooperative Multi Agent Architecture," AICT/ICIW, pp. 197, IEEE Computer Society, 2006.

[3]. S.Croom, , P. Romano, and Giannakis M.," Supply Chain Management: An Analytical Framework for Critical Literature Review," *European Journal of Purchasing and Supply Management*, Vol. 6., pp. 67-83, 2000.

[4]. M.R. Genesereth, and S.P. Ketchpel," Software Agents," *Communication of the ACM*, Vol. 37., pp. 48-53, 1994.

[5]. C.S. Hsieh, J.W. Hung, "*Integration Agent Negotiation and Data Global Consistency forms Automatic and None Bullwhip Effect Suuply Chain*," WSEAS Transactions on Information Science and Applicatiions, Issue. 6, Vol. 6, pp. 1037-1050, 2009.

[6]. A.M. Jankowska, K.Kurbel and D. Schreber, "An architecture for agent-based mobile Supply Chain Event Management," Int. J. of Mobile Communications", Vol. 5, pp. 243-258. 2007.

[7]. Y.B. Kim and S.W. Lee, "Performance Evaluation of Mobile Agents for Knowledge-Based Web Information Services KES-AMSTA," Lecture Notes in Computer Science, Vol. 4496, pp. 209-218, Springer, 2007.

[8]. O.B. Kwon, "Multi-agent system approach to context-aware coordinated web services under general market mechanism," Decision Support Systems, Vol. 41(2), pp. 380-399, 2006.

[9]. L. Lu, G. Wang, "A study on multi-agent supply chain framework based on network economy", Computers & Industrial Engineering, Vol. 54, pp. 288-300, 2008.

[10]. P. Maes," Agents that Reduce Work and Information Overload," *Communication of the ACM*, Vol. 37., pp. 31-40, 1994.

[11]. M.P. Martinez, R.J. Peris, B. Kemme, G. Alonso. "Consistent Database Replication at the Middleware Level", ACM Transactions on Computer Systems (TOCS). In Press. 2003

[12]. M.O. Shafiq, A. Ali, H.F. Ahmad and H. Suguri,"AgentWeb Gateway - a middleware for dynamic integration of Multi Agent System and Web Services Framework", 14th IEEE International WoRDShops on Infrastructure for Collaborative Enterprise, pp. 267-270. 2005.

[13]. N. Singer, J.M. Pecatte and S. Trouilhet, "Combining Web Services and Multi-Agent Technology to Increase Web Cooperative Capacities," International Conference on Internet and Web Applications and Services (ICIW'07), pp. 13-19, 2007.

[14]. K.Sycara, K. Decker, A.Pannu, M. Willliamson, and D. Zeng," Distributed Intelligent Agents," *IEEE Expert*, 1996.

[15]. A. Vizcaíno, J.P. Soto and J.P. Rodríguez and M. Piattini, "A Multi-agent Model to Develop Knowledge Management Systems," HICSS, pp. 203, IEEE Computer Society, 2007.
[16]. K. Walsh, A. Vahdat, and J. Yang. Enabling Wide-Area Replication of Database Services with Continuous Consistency.
[17]. M. Wang, J. Liu, H. Wang, W.K. Cheung, X. Xie, "On-damand e-supply chain integration: A multi-agent constraint-based approach", Expert Systems with Applications, Vol. 34, pp. 2683-2692, 2008.
[18]. S.K. Yung, and C.C. Yang," Intelligent Multi-Agents for Supply Chain Management," *IEEE SMC'99 Conference Proceedings*, pp. 528-533, 1999.