Improve Performance of Enterprise Information Sap Portal

Ahmed Mohamed Lamey
Computer Science department
Faculty of Computers and
Information,
Minoufia University, Egypt

Dr.Hamdy M. Mousa
Computer Science department
Faculty of Computers and
Information,
Minoufia University, Egypt

Dr.Arabi keshk
Head of Computer Science
department Faculty of
Computers and Information,
Minoufia University, Egypt

ABSTRACT
Nowadays multiple organizations introduce computer science services throw their web portals in the World Wide Web, to help their third parties like vendor, customer, and also there Employees to gain information from organization data source each one in its specific field without effort ,for this purpose we introduce new algorithm to improve the performance of enterprise information SAP portal. This algorithm is used to add new functionality to the java sap connector (JCO) middleware, this middleware used to build portal for organizations which has implemented SAP ERP Application, this middleware used to connect SAP enterprise resource planning system to any java application which will be very benefit to improve business process management system. This middleware demonstrating poor performance. The loss of performance could be related to multiple components In this paper we will purposed new algorithm which fix two Defect issues of component that demonstrating poor performance which is very important to build an enterprise java portal That connect to SAP system and it is very important to improve performance of business process management life cycle of different business area in the organization.

Keywords
SAP, Java SAP Connector, R/3 architecture, Enterprise information portal, Enterprise resource planning, advanced business application program (ABAP).

1.   INTRODUCTION
An enterprise information portal (EIP) is a framework for integrating information, people and processes across organizational boundaries. It provides a secure unified access point, often in the form of a web-based interface [1], Enterprise information portal has become one of the most important need for all companies because it connect different third party with the enterprise company and also introduce different services to the third parity's of The company like vendor or customer and also employee and employer them self's so building a fast performance Enterprise information portal for different Enterprise, become extremely important to improve business process management cycles, so on this paper we will concentrate on solve performance issues of sap java connector middleware. This middleware is used to build

Enterprise information sap portal, for organizations which has Implemented SAP ERP system Application, next section introduces introduction about ERP System application and java sap connector middleware. Section III introduce The Architecture of the new algorithm , section VI discuss Case study implementation and deployment and section VIII contain Experimental Result ,and at last section contain the conclusion .

2.   SAP   ERP   SYSTEM   AND   SAP JAVA CONNECTOR
Enterprise resource planning (ERP) integrates internal and external management information across an entire organization ,embracing finance/accounting , manufacturing , sales and service, customer relationship management , etc. ERP systems automate this activity with an integrated software application. Its purpose is to facilitate the flow of information between all business functions inside the boundaries of the organization and manage the connections to outside stakeholders [13] . The fundamental advantage of ERP systems is that integrating the Countless processes by which businesses operate saves time and expense. Decisions can be made more quickly and with fewer errors. Data becomes visible across the organization. Tasks that benefit from this integration include Sales forecasting, which allows inventory optimization, Order tracking, from acceptance through fulfillment, Revenue tracking , from invoice through cash receipt, Matching purchase orders (what was ordered), inventory receipts (what arrived), and costing ,SAP is one of ERP systems that can fulfill all the requirement of the organization with the Countless integrating processes as mentioned above, the sap java connector (JCO) is a middleware component that enables the communication between the java programming language and ABAP SAP programing language. SAP JCO supports communication with the SAP Server in both directions: inbound calls (Java calls ABAP) and outbound calls (ABAP calls Java).ABAP stand for advanced business application program it is a high-level programming language Evolved by SAPCompany [9]. SAP JCO can be implemented with Desktop java applications an also with Web server applications. SAP JCO is used as an integrated component in the following applications, SAP Business Connector, for communication with external.

Connecting the integrated J2EE server with the ABAP environment [3].SAP JCO can also be implemented as a standalone component, for example to establish communication with the SAP system for individual online (web) applications [3]. Sap java connector (JCO) contain different issues due to multiple error sources which listed below in details,Frist too much data that transferred from both sides sap or java, second too many parallel call for multiple functions at the same times, third Poor backend performance or network problems which may corrupt the data while sending in both sides, fourth Bugs or an inadequate programming style [5].So this possible error sources lead to Applications respond slowly. Even hang the loss of the business data which lead to damage of your business cycle and break down business process for all of the organization? This paper proposed new algorithm for how to improve the connection of java application with SAP Enterprise resource planning system (ERP), and make fault tolerance in case of loss the data or in case of application hang up, also the proposed algorithm has been tested with a case study to place a sales order on the SAP ERP System throw java portal application, we will start discuss the new algorithm, and how JCO schema has been modified to fix possible error sources in the next section.
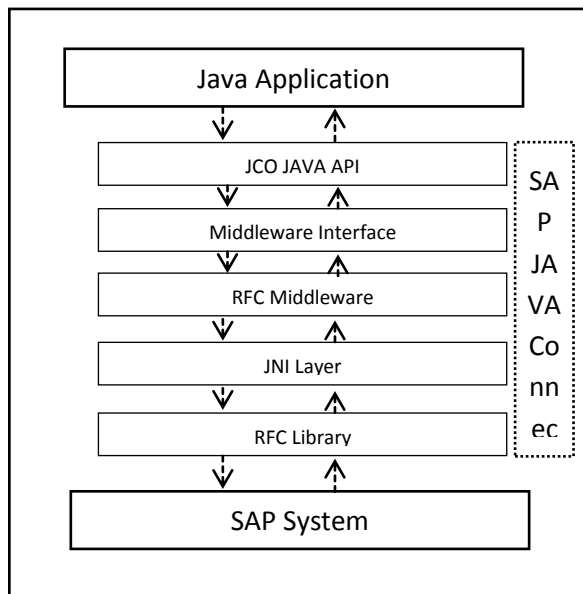

Fig 1:SAP JCO Architecture [5]

Figure 1 shows the technical schema of data conversion in SAP JCO middleware, Starting from a Java application, a Java method is forwarded throw the JCO Java API (Application Programming Interface) and an additional Middleware Interface to RFC Middleware, where it is converted to an RFC (ABAP) call using The JNI (Java Native Interface) layer, and sent to the SAP system. Using the same method in the other direction, an RFC Call is converted to Java and forwarded to the Java application [5].

Now we will show how this can be used to develop a java Portal to improve the business process management cycle. On the SAP side the connector call a remote function which is used to simulate the creation of sales order as an example for case study on the SAP ERP System this function contain import and export parameter to set and get data from the SAP

System and also Receives data from sap system. Next section describes the new algorithm and technique of how it works.

3. THE NEW ARCHITECTURE OF THE ALGORITHM

The new algorithm add new functionality to the JCO middleware it add new component on it, so it can save the data on a data base memory before sending it to the SAP System which make it fault tolerate to problem encourage throw the poor backend of the network and different programming styles, which mentioned Previously The new cashing algorithm provide high reliability and stability for any program use this middleware, ComparingFigure 1 with Figure2. Figure 1 show in the Frist layer the java application with the interface like snapshot of the case study we will describe later in Figure4, from layer 2 to layer 6 contain the SAP java Connector Components and in the last layer contain the Sap System comparing this Figure with Figure number 2 that contain java application and Sap System but it contain additional component that is connected to the sap java connector middleware this new component is the additional part which contain the new added component that will set to the middleware application. Also this new component has different implementation techniques that will be cleared in details in section number 5,Modified algorithm will be added to set in additional component to the JCO Middleware as show in Figure 2. The Figure show how the component of the modifiable algorithm Is set on the middleware when develop a new application the new Component of Cashing algorithm technique will works as the flow chart in Figure 3 show who to send the data in case of either ERP system down or in case of network bad performance or both problems occurs together the system is always worked. The new algorithm work as fault tolerance technique so it simulate the system until it return to its best case so the user hasn't no need to wait for SAP until system response . Which grantee data transfer to sap and also to java application in both bonds? This will be discussed in details in the next section.

3.1 How the new Cashing algorithm techniques work?

Figure 3 describe the flow chart and the sequence on which the algorithm run as the flowing steps.

1- When the project start the algorithm run as aperiodic thread every 30 seconds.
2- It checks if the warehouse of the data contains data.
3- If it contains data it checks for the network if the host is reachable with a good performance send the data.
4- Start again with step 2.

5- End if the algorithm paused.

3.1.1 Algorithm proceduralin step number three works as flow.

Send packet request with 32 bytes to the destination for aperiodic 30 seconds as show in the sample data.

Pinging Destination IP [0.0.0.0] with 32 bytes of data: as show and calculate the average replay time in Milliseconds, as show.Assume the number of replays within the 30 seconds as show below.
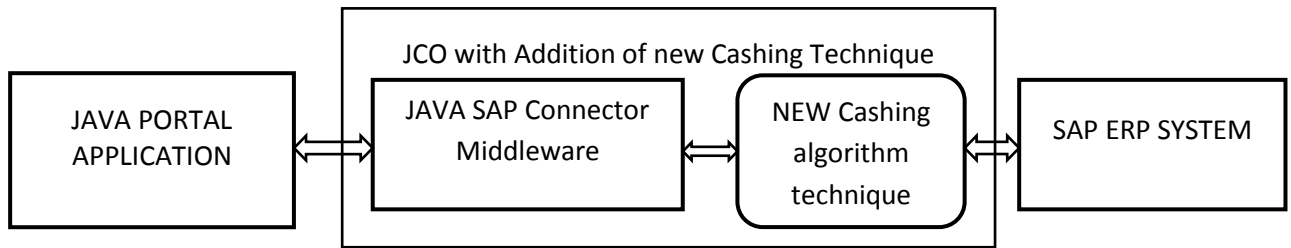
Fig 2: SAP JCO Architecture with the new cashing algorithm 1.1

Reply from Destination IP [0.0.0.0]: bytes=32 time=328ms TTL=42

Reply from Destination IP [0.0.0.0]: times=851ms TTL=46

Reply from Destination IP [0.0.0.0]: time=1115ms TTL=46

Request timed out.

Reply from Destination IP [0.0.0.0]: time=1019ms TTL=46

Reply from Destination IP [0.0.0.0]: time=1128ms TTL=46.

So average time:

6 (replays) – 1 (for time out replay) = 5 replays.

And the total average time for the 5 replays is:

$$x = \frac{328 + 851 + 851 + 1115 + 1019 + 1128}{5}$$
$$= 1058.4 \text{ Milliseconds.}$$

Know check if this vale is less than or equal to the trouble shouting value of best network performance.

The trouble shouting value will be also measured in the best network performance within the 30 second as above it normally will be = 1 Milliseconds if source and the destination on the same local network .

If the average time matches the trouble shouting value then sends the data throw the middle ware to the sap application as we will discuss in the next section with the real business case study. The trouble shouting value measure is different from one environment to another based on the network implementation, so after creates troupe shouting value it will be measure by get the average. Run the ping technique five times and get different values for the troubleshooting value get the average for all the troubleshooting value this value can be taken as the value of the best case of the network and also can be fixed to this environment .
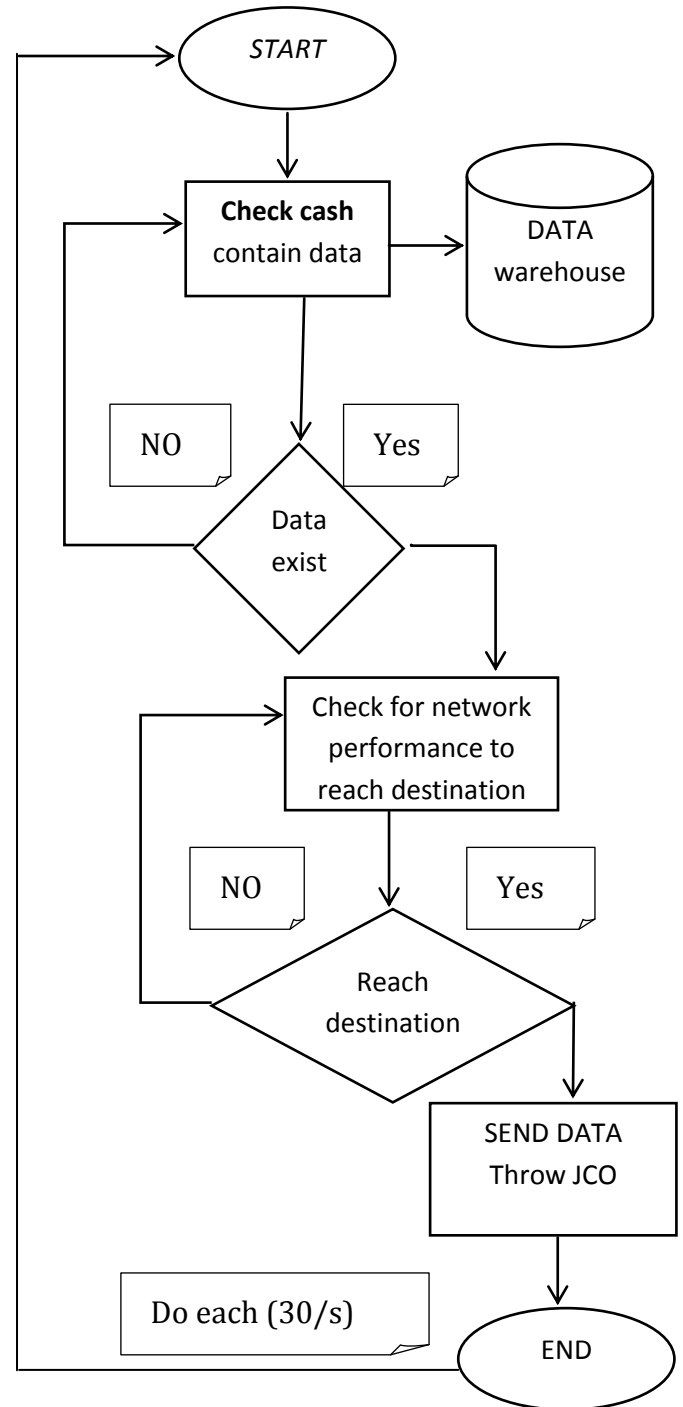

Fig3: New component Flow chart 1.2

4. CASE STUDY IMPLEMENTATION AND DIFFERENT DEPLOYMENT METHODS.

Fig 4: Snap shot of portal application to create sales order on SAP system using new algorithm.   1.3

In the Figure 4 shows screen shot of the case study java portal which I create using the proposed algorithm to send data to sap throw remote function call (RFC), it works as flow , first the java application sends the data to middleware the middleware pass data to the data cash of the new algorithm component, the new algorithm start and run its periodic procedural as described in the flow chart of Figure 3, Once the algorithm run it will not stop until all the application stop it will work as tracker for the data in the database that is created based on the application requirement and business need once it found the data , it make its checkup for the network performance to reach destination   as described earlier in step 3 of the cashing algorithm.

In Figure 4 , the application consist of two parts first part is the java application side and the second side is the remote function call called BAPI in the SAP side which consist of customize and Standard RFC's ,

 Frist the application call BAPI function that is created using the advanced business application program one of the fourth generation sap system code to check if the user is registered on the sap database throw the portal login screen after that it allow access to the user to multiple functions , the main function is to enable the customer to add the list of the materials and the quantities of these material he want to buy and other customer data like name and other then the customer submit the create button after the customer fill data required as mentioned above it send the data to the database , then the new algorithm work as Figure 3 shows , then ABAP function module is used to create the order on the SAP side. This function includes the import and export parameters and the source code in advanced business application program (ABAP) Language to simulate the creation of the sales order, the SAP system will be called using the Remote function calls using the SAP Java Connector (JCO).

BAPIs are the standard SAP interfaces RFC's functions. They play an important role in the technical integration and in the exchange of business data between SAP components, and also

Exchange of business data Between SAP and non-SAP components. BAPIs enable you to integrate these components and are therefore an important part of developing integration scenarios where multiple Components are connected to each other, either on a local network or on the Internet. BAPIs allow integration at the Business level, not the Technical level. This provides for greater stability of the Linkage and independence from the underlying communication Technology [10].

The case study connect to the SAP system to create sales order on the enterprise resource planning SAP system, it also enable the user to do multiple functions that is related to sales order creation. Also the case study enables the customers of the company to set their sales orders on it and after that the portal application is responsible for sending the data to the sap system using the new cashing algorithm building inside it with the previous concept as described earlier. Next section describes different methods of how we can deploy this application with the algorithm.

5. Deployment methods.

There are two ways to implement the application with the new cashing algorithm, Frist is to deploy it as standalone application on server, second is to deploy it at the same server with the JCO middleware, the decision of the deployment technique is set based on the amount of the data flow between the two application, in both cases the algorithm will work fine and will do its Functionality as required mentioned in the Proposal algorithm, also most companies which require deploying all its owned applications in one server. However, this case is applicable with the proposed algorithm. This requires companies to maintain both a J2EE server to run J2EE/Java applications inside SAP web application server, so also we can merge the two applications on one server and deployed the java application on the SAPnetweaver web application server [12].

6. EXPERIMENT RESULTS.

Table 1. Implementation results

	Criteria	After	Before
a	send data SAP system / per seconds	5 / seconds	60 / second
b	Guarantee of Send data on system down	yes	no
c	Guarantee of Send data on network down	yes	no

This result table describes the difference before and after applies the architecture algorithm into Middleware component. And is measured when simulating create sales order on the SAP system.

a. Frist criteria the time of sending the data from the java system to sap system has been decreased from 5 seconds to 60 seconds This has been done using many different concepts of reference [2] which will be cleared on chain of future papers.

b. The Second criteria are sending data from the java system to sap system in case of sap system down has been granted after Appling the purposed algorithm.

c. Also third criteria have grantee during the running of the algorithm because its technicality is to check for the network performance before sending the data.

7. CONCLUSIONS

In this paper we purposed new algorithm to improve building enterprise information sap portal Specially building portal application which connect to the SAP ERP system using java programming language by add new component into the java sap connector middleware and also we discussed the need of enterprise information portal and its important need to improve the business process management life cycle throw the case study E-commerce customer portal site to Place his sales order on SAP ERP System , and approved by case study and results .how to fix Possible Error Source raised from the JAVA SAP Connector (JCO) specially sending data in case of poor backend of the network performance to reach SAP destination, To build an Stable enterprise information portal with the standard design principals To be beneficial for different business process management life cycle of different business area.

8. REFERENCES.

[1] Boye, Janus (2005-01-18). "Portal Software: Passing Fad or Real Value?"See CMS Watch.

[2] Thomas G. Schuessler, "www.ARAsoft.de" Developing Applications with the SAP Java Connector"(JCO), 2001–2002.

[3] SAP AG, "SAP Java Connector" SAP Library, 2004 SPS23. Retrieved June10, 2011 from the World Wide Web:
http://help.sap.com/saphelp_nw04/helpdata/en/6f/1bd5c6 a85b11d6b28500508b5d5211/frameset.htm

[4] Bidgoli Hossein, (2004). The Internet Encyclopedias, Volume 1, John Wiley & Sons, Inc. p. 707.

[5] SAP AG, "JCO Performance, JCo Exceptions" SAP Library, 2004. Retrieved June10, 2011 from the World Wide Web:
http://help.sap.com/saphelp_nw04/helpdata/en/7b/ca6641 ffb68147e10000000a1550b0/frameset.htm

[6] Ilja Daniel Werner," SAP Java Connector - Operation and Development", SEP 30, 2007.

[7] SAP AG, "SAP System", 2011.

[8] Alfred Barzewski, Carsten, Wolf Hengevoss and Karl Kessler, "Java Programming with SAP NetWeaver ", Aug 4, 2008

[9] Ulrich Gellert, Ana Daniela Cristea, "WebDynpro ABAP for Practitioners", Sep 10, 2010.

[10] Gareth M. deBruyn, Robert Lyfareff and Anil Thakur, "Java and BAPI Technology for SAP (Prima Tech's SAPBook Series)", Jan 15, 2000.

[11] Krunal J. Rana," MIGRATION OF J2EE/JAVA APPLICATIONS TO SAP NETWEAVER, master thesis.JAVA DEVELOPMENT INFRASTRUCTURE (SAP NWDI)", California State University, Chico, Spring 2009.

[12] Karl kessler, peter Tillert, panayot Dobrikov,"java programming with sap web application server ", Galileo press Gmph, 2005.

[13] Simha R. Magal, Jeffrey Word, "Integrated Business Processes with ERP Systems", March 26, 2011