

Research on the Application Integration Model for the Agricultural Enterprise of Integrative Production and Marketing

Feng Yang, Xiandi Zhang, Zhongqiang Liu, Zhenzhi Wang, Kaiyi Wang

► **To cite this version:**

Feng Yang, Xiandi Zhang, Zhongqiang Liu, Zhenzhi Wang, Kaiyi Wang. Research on the Application Integration Model for the Agricultural Enterprise of Integrative Production and Marketing. Daoliang Li; Yande Liu; Yingyi Chen. 4th Conference on Computer and Computing Technologies in Agriculture (CCTA), Oct 2010, Nanchang, China. Springer, IFIP Advances in Information and Communication Technology, AICT-346 (Part III), pp.682-695, 2011, Computer and Computing Technologies in Agriculture IV. <10.1007/978-3-642-18354-6_80>. <hal-01563464>

HAL Id: hal-01563464

<https://hal.inria.fr/hal-01563464>

Submitted on 17 Jul 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Research on the Application Integration Model for the Agricultural Enterprise of Integrative Production and Marketing

Feng Yang¹, Xiandi Zhang¹, Zhongqiang Liu¹, Zhenzhi Wang¹, Kaiyi Wang^{1,*}

¹National Engineering Research Center for Information Technology in Agriculture,
Beijing, P. R. China
yangf@nercita.org.cn, wangky@nercita.org.cn(*Corresponding author)

Abstract: To integrate application systems under heterogeneous environments has gradually become a pressing demand for most agricultural enterprise. The vast diversity of implementation technologies increases the complexity of applications integration. With the aim of facilitating the development of integration solutions, we propose the specification of application integration. To achieve this, we analyzed the current situation of information, the environment of application systems, and the specific requirements of integration. We also studied the integration technologies. And then the ESB-SOA model for the agricultural enterprise of integrative production and marketing was proposed on the basis of a number of advanced technologies, such as ESB, SOA, EAI and Web Service. Based on ESB and SOA technologies, the model is able to utilize and integrate effectively various service resources on the geographically distributed computing environments. And through intelligent agent technologies, it supports cooperation and coordination mechanisms for the service activities. To exemplify and validate the profile, a case study is presented in which the proposed profile is used.

Key words: agricultural products; production and marketing integration; management; information; heterogeneous environment

1 Introduction

With the extensive application of information technologies, the information of agricultural enterprises that integrative production, process, circulation and marketing have also entered a new phase. For a long time, different departments in different production processes applied different information systems, resulting in the heterogeneous environments that a variety of hardware devices, operating systems, databases, network protocols coexist. Due to lack of standardization and compatibility, it is difficult to share information resources and there are many “information islands”, applications integration has become an urgent demand for enterprise development [1]. How to use advanced information technologies to share resources in all production processes is a common problem for many enterprises of integrative production and marketing. In addition to study application systems

under the heterogeneous environment in integrated production and marketing agricultural enterprises, an integration model is presented based on ESB (Enterprise Service Bus) and SOA (Service-Oriented Architecture).

2 Enterprise Application Systems Analysis

The integrative production and marketing of agricultural products refers to “on the professional basis, agricultural products production and marketing departments are integrated into a unified whole in the economy and organization, that is, the production, marketing, processing, storage and transportation of agricultural products are united in a joint operation in large enterprises, so that the whole process completely reach organic integration and coordinated development” [2].

However, the lack of planning, distributed disorder, departments division, there are a variety of information systems coexist in the integrative production and marketing enterprise. Among information systems, the development platforms, data storage, network protocols and operating environment are heterogeneous, which resulted in duplicate construction, waste of resources, information block of enterprise systems, and so on. For example, as shown in Fig. 1, there are a variety of information systems that are independent of each other coexist in a vegetables enterprise of integrated production and marketing.

Whether within or between internal and external systems, each system is basically independent of each other. In this way, the following problems are inevitable: (1) The same type of data spread across multiple systems, not a single application can access to all data; (2) There is no unified data model, business personnel can not get a global data view; (3) Different applications can't share data and services, this will result in duplication of development; (4) System can only provide data in accordance with fixed demand, when the user needs a particular type of new data, the system can not automatically adapt [3].

Referred to above present “information islands” problems for most integrative production and marketing agricultural enterprise are facing, this paper propose a enterprise application integration model based on SOA and ESB. The model is used to complete the task that the system can not complete alone, and to solve the information integration and exchange problems in some cross-system and cross-databases. Thus, more comprehensive services are provided to the various departments of agricultural enterprises.

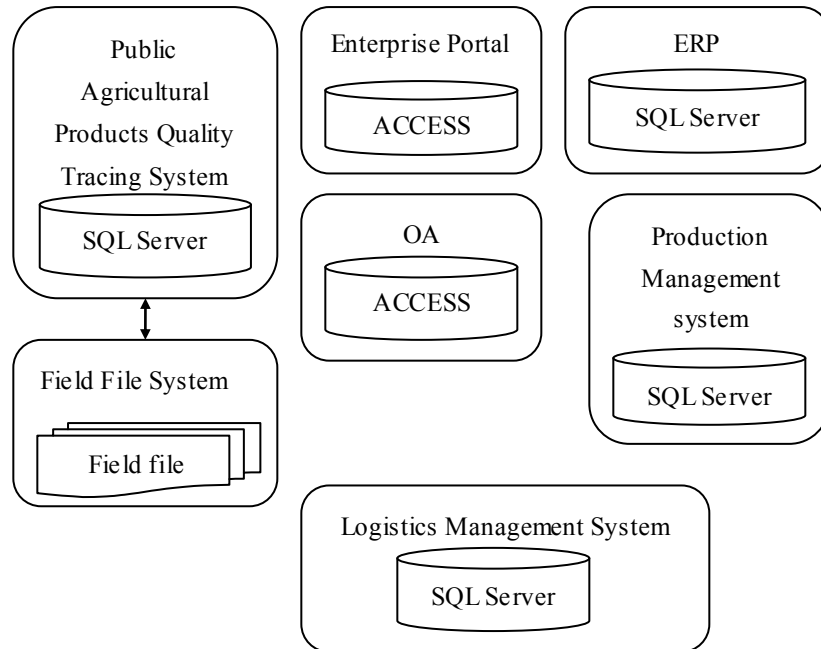


Fig. 1. Information Systems in an enterprise of integrated production and marketing

3 Applications Integration Model Design

Service Oriented Architecture is an architectural paradigm for components of a system and interactions or patterns between them. A component offers a service that waits in a state of readiness. Other components may invoke the service in compliance with a service contract [4].

Suppose all IT resources are refined into service and published, and services can call each other, so that you can call .Net service in the J2EE environment, but this also can be done without SOA. As long as the two IT systems recognized each other's way, it also enables that point to point interconnection even if there is no public or uniform service interface. So we have to admit, if we own service, but the requestor and the service providers are still in need of such the explicit call of point to point between them, then this is not a typical SOA framework. The participation of both services still have to establish the link one to one, this does not meet our scalability, flexibility requirements [5]. As shown in Fig.2.

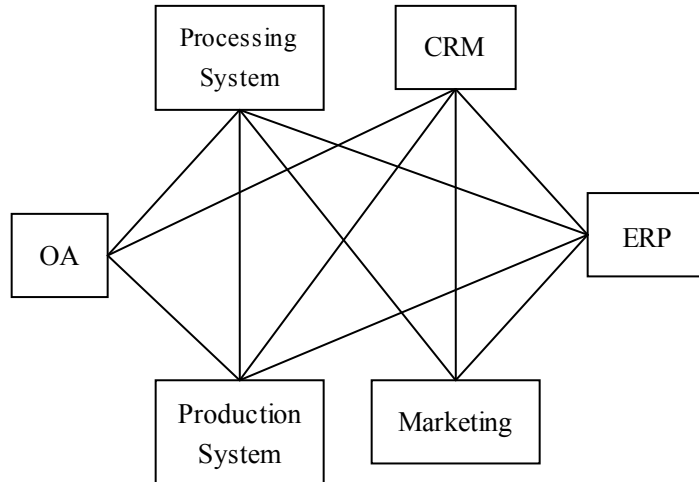


Fig. 2. Point-to-Point service integration

Facing continuing expansion, a tool to manage these services is necessary to simplify the deploying and accessing of services in the system integration process. ESB (Enterprise Services Bus Enterprise Service Bus) is generated in such demand which is popular in the System Integration currently. Whether internal or external users to access various data sources at the time based on point to point manner. Truly reflects the concept of SOA, everything is service, service bus (BUS) in an equal position. And it has good scalability. The concept model of ESB-SOA is shown in Fig.3.

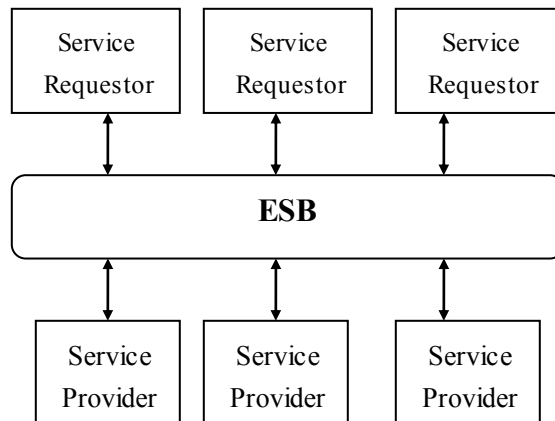


Fig. 3. Enterprise Service Bus Concept model

ESB is composed primarily of by three main parts: the service gateway (Service Gateway), service adapter (Service Adaptor), service registries (Service Register) [6].

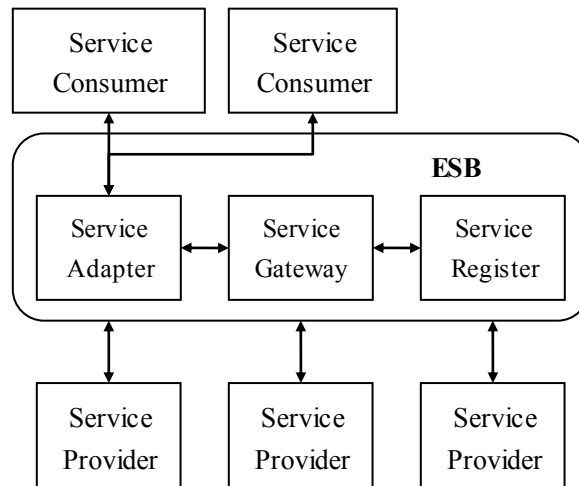


Fig. 4. Enterprise Service Bus architecture

1. Service Gateway is a core component in ESB. Its main role is responsible for such as routing and interactive of services.

2. Service adapter is an important part of ESB, which bears the protocol conversion of message and services to implement different protocol formats of the interaction between services.

3. ESB also needs some service registry to provide help for service routing. However, SOA may still have a separate business services directory, its most basic form may be the services directory in designing, and it's used to achieve the reuse of services in development activities.

3.1 Service Adapter

The function of service adapter is to find, bind, and call web services, and so on. So that the client calls web services in back without knowing the details, and accesses to local or remote web service transparently. Fig.4 describes the implementation of principles of Service Adapter. The Service Adapter consists mainly of three major processing units.

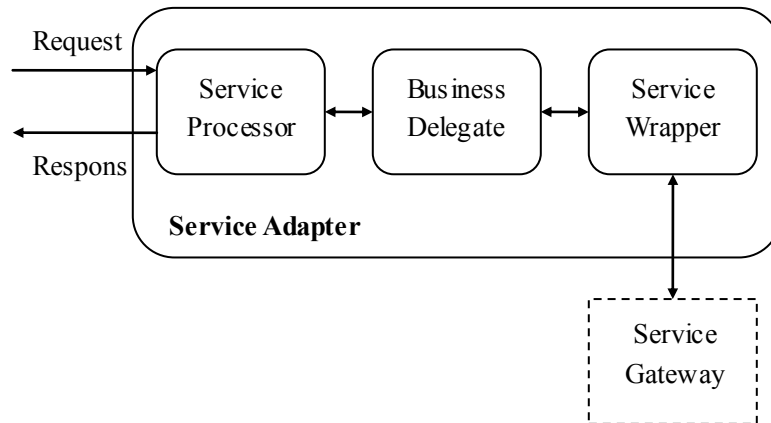


Fig. 5. Service adapter schematic

1) SP (Service Processor): It is used for receiving and parsing SOAP messages passed over the TA (transmission adapter), processing security, such as digital signatures and encryption.

2) BD (Business Delegate): It is used for controlling the business calling. All business requests are completed through calling for service wrapper.

3) SW (Service Wrapper): It wraps the discovery, binding, calling web services in a unified package, thus to shield the details of various web services.

Service adapter first receives the SOAP message from the transmission adapter (TA), and then parsing SOAP messages for security processing (such as SOAP digital signatures and encryption, etc.), then the SOAP message are passed to the business agent, business agent get corresponding semantic descriptions of each web service WSDL address by looking up the corresponding semantic function, and resolve the semantic description.

3.2 Service Gateway

Service Gateway design is shown in Fig. 6 includes the following components:

1. Service Processor: It operates the message in security and transmits it to the business agent to deal.

2. Business Agent: It is used for controlling the business call, all business requests are completed through it calling for service manager.

3. Service Manager: It wraps the discovery, binding, calling web services in unified package, thus to shield the details of various web services.

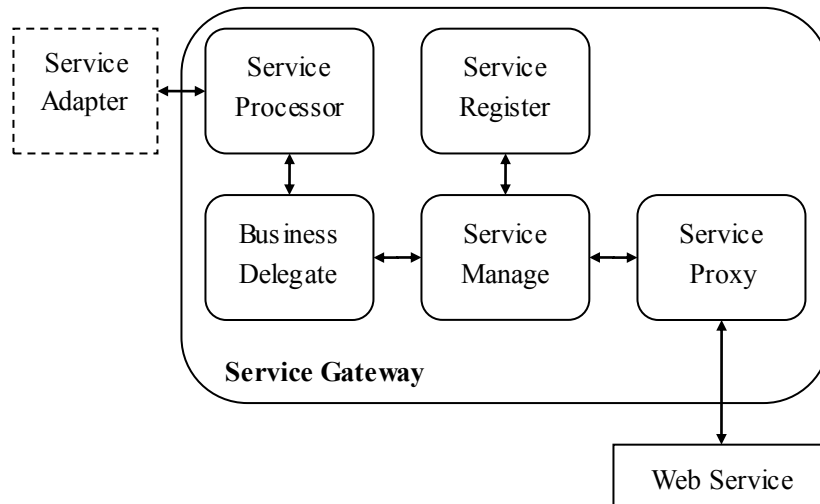


Fig. 6. Service Gateway

The following is a comprehensive description of the Service Gateway:

1. Service consumer sends a request to Enterprise Service Bus (ESB).
2. In ESB, Service Adapter completes protocol conversion. Then, the converted request is send to Service Gateway.
3. The request is parsed in Service Gateway's Service processor and Business Delegate and passed to Service Manage.
4. The request parameters, and important data in the Service Manage have been resolved will be stored in a service-oriented types data.
5. Service Manage gets the binding information and the relative position of service by querying in Service Register. If there is no appropriate information of services, an error message is returned to service consumer.
6. Query results are returned to Service Manage, and then the result as a parameter saved in the Data Window together with the request parameters will be passed to Service Proxy module.
7. Service Proxy module is responsible for calling the appropriate service implementation, and to call the results back to the Data Window and reset the parameters.
8. Service Manage returned the response results to the service processor and Business Delegate.
9. Service Processor will return response back to Service Adaptor, and then returned to service requester by Service Adaptor.

In summary, the dynamic service discovery and routing are achieved through Service Manage, and seamless integration of applications is achieved using Service Proxy.

3.3 Service Registry

Service is registered and discovered through the service registry UDDI (Universal Description Discovery and Integration). UDDI uses XML to describe the service provider and the provided web service, dynamic bind and call service.

In ESB-SOA, the core class of the service registry is "serviceBusPortal", using the "plug-in - Service" configuration file defines the name of the service and location to achieve the service's registration.

Service registration: users publish their own services to the ESB-SOA system registry. The released service should have service information and service provider information.

Service discovery: the client user sends queries commands and the command is transferred to the class "serviceBusPortal" in center for registration, and then the service will be found according to the location of service defined in "Plug-in-service" configuration file.

Service binding: After found the service, "ServiceBusPortal" load the corresponding service dynamically.

The relationship between them is shown in Fig. 7:

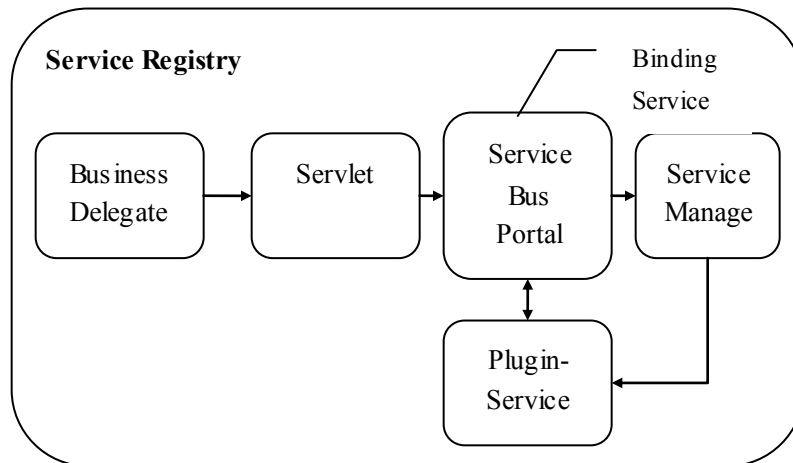


Fig. 7. Service registry

3.4 Log Management

Log service management model is based on Message Service, and uses asynchronous approach. When you need to register log, processing component calls for the API of recording information provided by log service, and then log information will be inserted into the log queue provided by the API. Because it is an asynchronous approach, so insert the log information to the log queue to get the response message will not be blocked. Message Service monitor log queue listens from time to time, when log information is inserted into the log queue, Message Service listener extract log information from the log queue and write to back-log database by Log

Adapter in accordance with the standard format, and then delete the log queue processed log information. When query log, the APIs of log services access the log database through the Log Adapter, and get the specified level of log information. Log management implementation is shown in Fig.9.

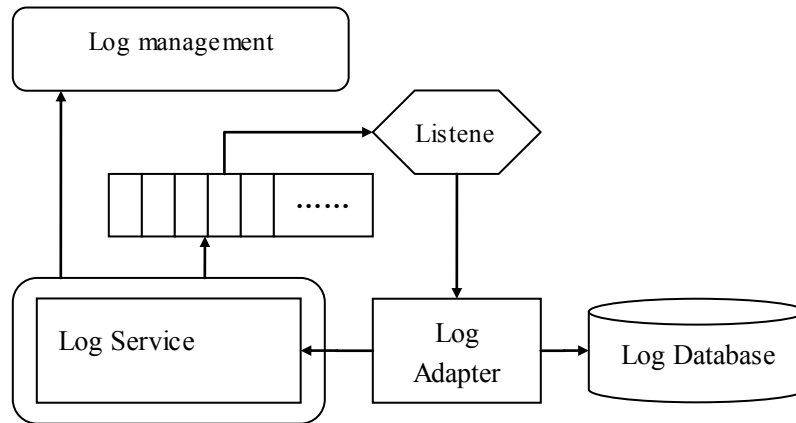


Fig. 8. Log management

4 Model Application

Beijing Tian An Agriculture Development Co. Ltd is a production and business-oriented vegetable agricultural enterprises, after years of information establishment, several information systems have been established, such as ERP, Field Record Management System, Vehicle Transportation Control System, Enterprise portal, OA. However, each of them is independent relatively. And it is hard to share data because of the difference in development language, database. Meanwhile, the current system is mainly used in the enterprise, but the field planting, terminal marketing and sales are not managed in the way with information.

According to the actual needs of Tian An Com., we built enterprise application integration platform based on the above proposed ESB-SOA enterprise application integration model, and developed a mobile sales management system, sales management system, mobile production management system, production management systems, and integrated systems by applications integration platform. The application integration logic structure of Tian An Com. is shown in Fig.9.

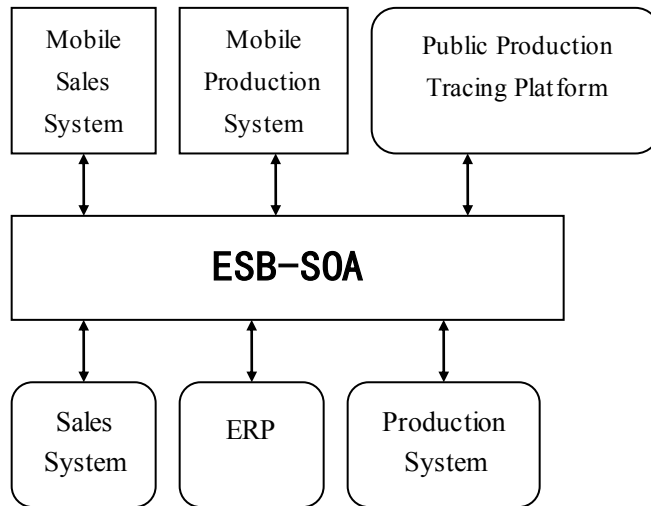


Fig. 9. Application Integration Logic Structure of Tian an Com

4.1 Production Application Integration

The production management of Tian An Com. is mainly related to the record of the field plant. On the one hand, the field record is sent to the enterprise's production management system, and then submit to the public quality tracing platform. Through the ESB-SOA platform, production management system registers records management services to Service Registry, mobile production systems sends a request to the ESB-SOA, the ESB-SOA finds production management services upon request information, and then the service calls for APIs to submit field file information to the service of production management system in the server. On the other hand, public quality tracing platform registers services to the ESB-SOA Service Registry, enterprise production management system sends a request to the ESB-SOA to submit a field record to the public quality tracing platform. Field records management flowchart is shown in Fig.10, system implementation is shown in Fig.11 and Fig.12.

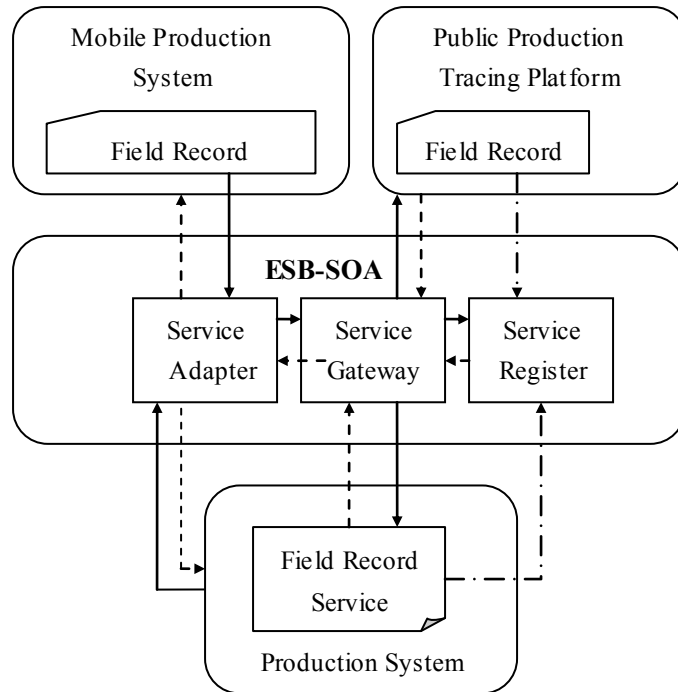


Fig. 10. Field Records Management Flowchart



Fig. 11. Mobile Production System



Fig. 12. Production System

4.2 Sales Application Integration

According to the sales information need of Tian An Com., a mobile sales system and sales management systems in server are implemented. At the same time, sales management system and ERP registered to Service Registry in ESB-SOA platform. Thus, mobile client sends request to the ESB-SOA to call for services to submit the order, and then the sales management system on the server sends request to the ESB-SOA to call for ERP service to submit the order. The flowchart of sales management is shown in Fig.13, systems implementation is shown in Fig.14 and Fig.15.

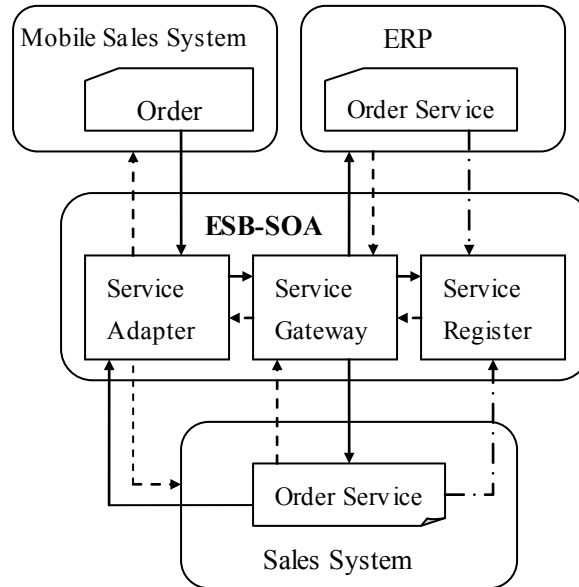


Fig. 13. The flowchart of sales management



Fig. 14. Mobile Sales System



Fig. 15. Sales System

5 Concluding Remark

Because of many production process and loose management, the agricultural enterprise of integrative production and marketing has special requirements for information. Based on the analysis of systems application status in the agricultural enterprise of integrative production and marketing integration, an ESB-SOA model of application integration is proposed based on SOA and ESB technologies. And on this basis, an application integration platform was implemented in Tian An Com. according to the ESB-SOA model. Thus, different departments in Tian An Com. could share data resources through the platform and interoperability among systems become possible. Moreover, information management is extended to plant record management in field and order management in supermarket. More important is that ESB-SOA has good scalability, and it is easy to plug and play for new application.

Acknowledgements. This research is funded by the Support of Science and Technology Project of the State, Programmed award No. 2008BADA0B02.

References

1. Huang Guoqing, Zhang Yong. The Choice Model of EAI Suited for Integration the Information Systems of Supply Chain Management, *Computer Engineering and Applications*, 2005(23): 221 – 229.
2. Li Xiaoxiang, Guo dong. Integrative Production and Marketion of Angricultral Products, *Commercial Economy*, 2000(2-1):145 – 147.
3. Zhang Jinming, Chen Chuanbo. Study on the Service-oriented Enterprise Application Integration Architecture and its Application, *Huazhong University of Science & Technology*, 2005.
4. Zhou Xiaoyan, Chen Xiuquan. The Research on Application of Enterprise Service Bus in SOA Implementation, *Dalian Maritime University*, 2009.
5. Lü Hongwei. Enterprise Application Integration Solution Based on Service-Oriented Architecture and Enterprise Service Bus, *Computer Applications and Software*, 2010, 27(4): 215 – 216.
6. Cai Liping. Research and implementation of an adapter for Web Services with ESB-SOA architecture, *Journal of Southwest University for Nationalities*, 2008, 34(5): 1047 – 1050.
7. Qin Ding, He Zhiming. The Design and Implementation of Service Mediation Framework Based on ESB, *Journal of Jiangxi University of Science and Technology*, 2009, 30(3): 49 – 52.
8. Du wanya, Lu peng, Zhang hongyan. The Design and ImPlementation of ESB based SOA Framework, *Beijing Jiaotong University*, 2008.
9. Cai Zhaoquan. Implementation of Heterogeneous System Integration based on ESB, *Computer Applications*, 2008, 28(2):538 – 540.
10. Chen Zhiwang, Wu Kaijun. Research and Design of Double Service Bus Base on SOA, *Control & Automation*, 2010, 26(3-3): 43 – 45.
11. Liu Zhaoyun, Sun Shimin, Wang Jiyong. The Trend of Agricultural Supply Chain Management in China, *Commercial Research*, 2009(3): 162 – 164.