Paper Title:

A Case Study on the Development Methodology of a Factory Control System and Interface with an Integrated System in the Shipbuilding Industry

Track Title:

Technology and Information Management

Abstract:

As application scale is expanding to the whole organization with large scale distributed systems, we often face the unexpected problems, which are application control in the distributed computing environment, security control, performance degradation, and interfaces with different types of DBMS(Data Base Management Systems). To solve these problems a 3-tier architecture is suggested. Even though 3-tier architecture requires more investment cost compared to a 2-tier and is difficult to implement, it presents a smoother interface between DBMS, flexibility of selecting development software, extensibility, and system development by concurrent engineering. In this study, we present a 3-tier development methodology of a factory control system by concurrent engineering and interface with an upper integration system in the shipbuilding industry.

Authors

First Author: Tae-Hyun Baek, Senior Researcher *

Hyundai Industrial Research Institute, Hyundai Heavy Industries Co., Ltd.
1, Cheonha-Dong, Dong-Ku, Ulsan, 682-792, Korea
Tel: 82-52-230-3202 Fax: 82-52-250-9587 E-mail : baekth@hanmail.net

Second Author: Kuy-Hoon Chung, Ph.D, Chief Researcher

Hyundai Industrial Research Institute, Hyundai Heavy Industries Co., Ltd.
1, Cheonha-Dong, Dong-Ku, Ulsan, 682-792, Korea
Tel: 82-52-230-3202 Fax: 82-52-250-9587 E-mail: khchung@hhi.co.kr

Third Author: Hyung-Shik Kim, Head of Automation Department

Hyundai Industrial Research Institute, Hyundai Heavy Industries Co., Ltd.
1, Cheonha-Dong, Dong-Ku, Ulsan, 682-792, Korea
Tel: 82-52-230-3211 Fax: 82-52-250-9587 E-mail: hskim@hhi.co.kr

Fourth Author: Ju-Chull Park, Ph.D, Professor

Department of Industrial Engineering, University of Ulsan
San 29, Mugeo-Dong, Nam-Ku, Ulsan, 680-749, Korea
Tel: 82-52-259-2177 Fax: 82-52-259-2180 E-mail: jcpark@uou.ulsan.ac.kr

(***** For communication purpose)

Introduction

IT(Information Technology) environment has changed from mainframe in the early stage to open system of client/server environment which presents easier system interconnection, advantageous cost performance ratio and convenience of program portability and accordingly the development of PC, network, RDBMS and GUI Tools. It is called a 1st generation client/server computing model or a 2-tier application model(Umar, 1993). In spite of these advantages, as application scale is expanding to the whole organization with large scale distributed systems, we often face unexpected problems, which are application control in the distributed computing environment, security control, performance degradation, and interfaces with different types of DBMS(Data Base Management Systems).

Under these circumstances new paradigm turned up in the middle of the 90's. The paradigm, called 3-tier or n-tier(multi-tier) architecture offers integration for different types of resources (hardware, DBMS, development tools, etc.). The 3-tier architecture makes new types of software genre as 3-tier-based middleware presenting a smoother interface between DBMS, flexibility of selecting development software, extensibility, and system development by concurrent engineering.

In this study, we present a 3-tier development methodology of a factory control system by concurrent engineering and interface with an upper integration system in the shipbuilding industry.

General Perspective of System

The shipbuilding division in 'H' company has developed an integrated control unit by combining several planning and control systems for enhancement of their work competitiveness for several years. But it needed lots of human resources and development costs and it takes a long time to stabilize integration of the system. Also each department demanded firstly to operate and develop their own, being separated departmental system for efficient planning and control rather than an integrated system. In this case, if it took at least two years for completing an integrated system, it would be better to integrate a division unit after applying and implementing each unit system for the department. In 'H' company case after beginning the 2-tier concept, it was planning to integrate with other systems under 3-tier architecture. Also for system analysis and design they divided into two parts; As-Is and To-Be(Chung, to be published). Figure 1 shows the modules of the developed system; HYPOS(HYundai Paint shop Operating System).

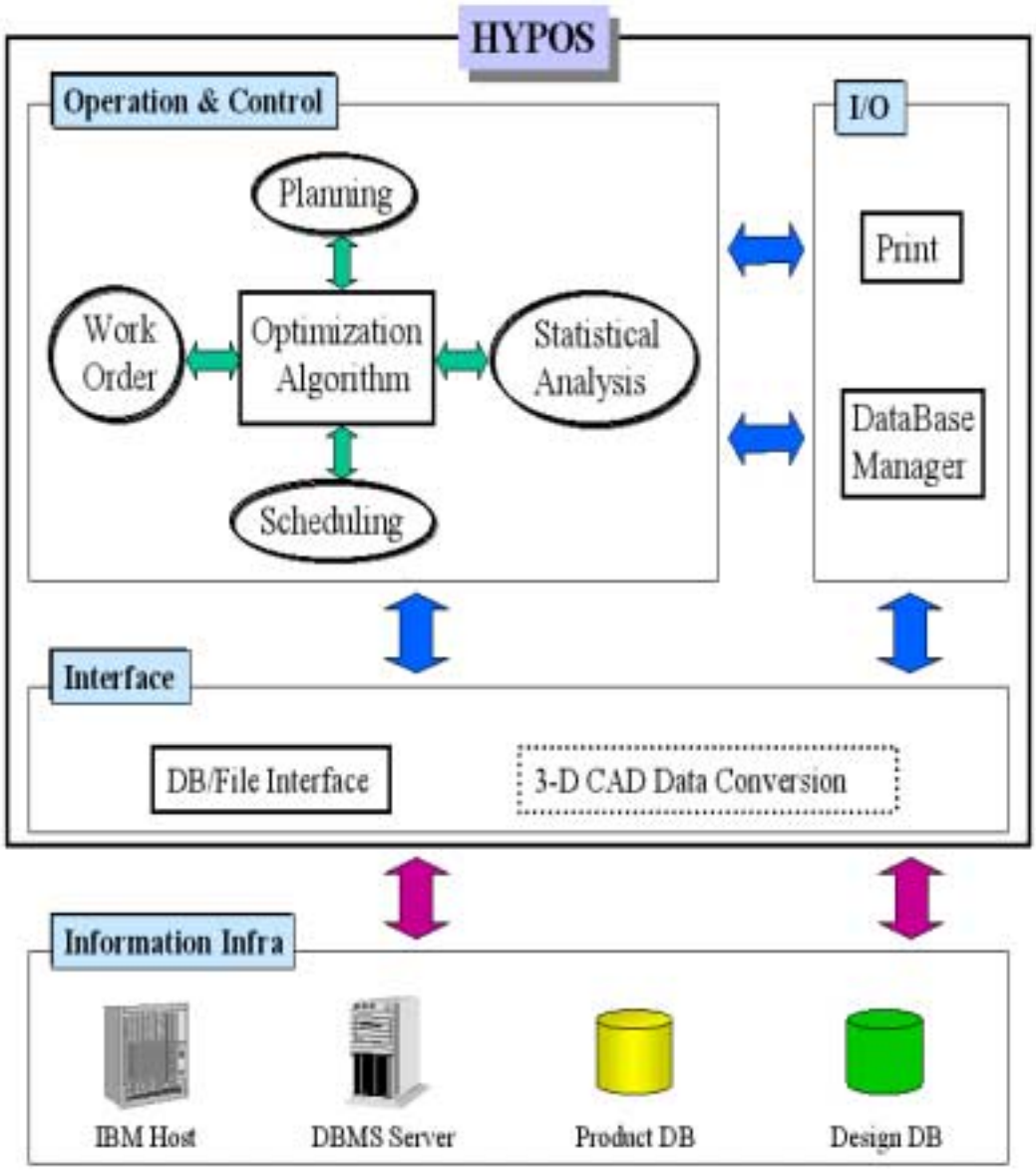

Figure 1. The Modules of HYPOS

Phases of System Design and Implementation

As shown in the Figure 1 we firstly set up relationships and interfaces between HYPOS and other systems, and carried out system design to implement 3-tier architecture as Figure 2.
We also used middleware as Entera^{TM} that offers high performance, scalability, fault tolerance, location transparency and integration security functions by applying standard technology infrastructure based on communication services among applications of distributed systems. Entera^{TM} has the advantages as below(Inprise, 1999).

- Implementing open distributed systems under different types of hardware circumstances
- Accessing application without regard to location of hardware and database

- Keeping the independence of protocol on network
- Open structure easily integrated and interfaced with internal and external systems
- Ease of system standardization by consistent interface between client and server
- Ease of system maintenance

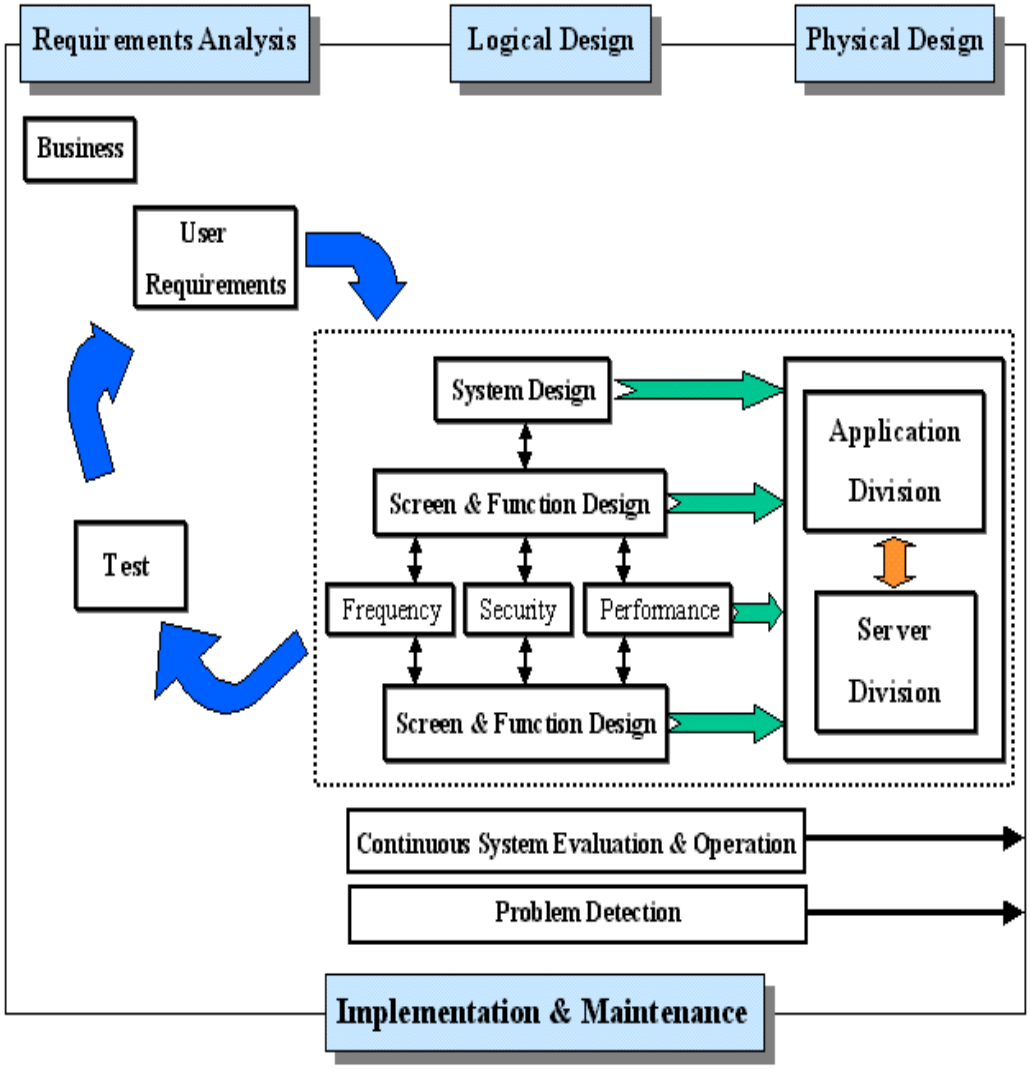

Figure 2. Application Design for 3-Tier Architecture Implementation

Through such phases we completed system development configuration considering interfaces with related systems as Figure 3.

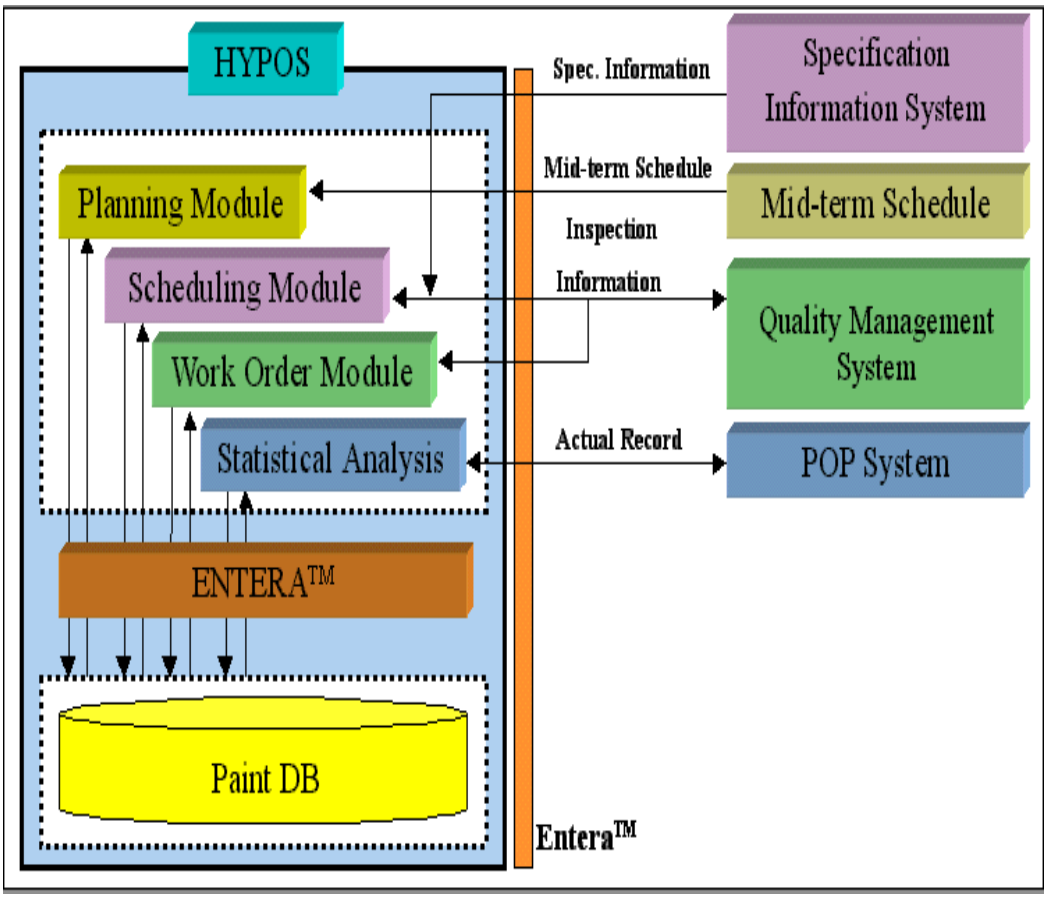

Figure 3. HYPOS Configuration Considering Interfaces with Related Systems

Conclusion

When developing HYPOS concurrent engineering was applied by carrying out system design and application development side by side due to a restriction in the work period, which is a six month. It required a great deal of time and effort to complete the system within a short-term. Also to succeed on a large-scale project based on 3-tier architecture, each component should be individually tested even though other related components are yet to be developed. It helps to concentrate the GUI developer on user interfaces, business developer on business logic and database developer on DB.

References

Chung, K. H., K. K. Cho, T. H. Baek and J. C. Park, "An Operational Scheduling System for Paint Shop in the Shipbuilding Industry", *International Journal of Industrial Engineering*, to be published.

Inprise International, Inc., *Entera/Fx User's Guide*, 1999.

Popkin Software and Systems, *SystemArchitect User Guide*, 1998.

Rajkumar, T. M. and D. L. Dawley, "Designing and Managing Client/Server DBMSs," *Information System Management*, Vol. 24, No. 6., pp.11-19, 1994.

Salemi, Joe, *Guide to Client/Server Databases*, Ziff-Davis, 1995.

Umar, Amjad, *Distributed Computing & Client/Server Systems*, Prentice Hall, 1993.