Available online at www.sciencedirect.com
SCIENCE DIRECT®
Technovation 25 (2005) 433–441
technovation
www.elsevier.com/locate/technovation

A prototype multi-agent ERP system: an integrated architecture and a conceptual framework

Bih-Ru Lea[a,*], Mahesh C. Gupta[b], Wen-Bin Yu[c]

[a]Department of Business Administration, School of Management and Information Systems, University of Missouri—Rolla, Rolla, MO 65409, USA
[b]Department of Management, College of Business and Public Administration, University of Louisville, Louisville, KY 40292, USA
[c]Department of Information, Science and Technology, School of Management and Information Systems, University of Missouri—Rolla, Rolla, MO 65409, USA

Abstract

This paper proposes a prototype multi-agent enterprise resource planning (MAERP) system that utilizes the characteristics and capabilities of software agents to achieve enterprise wide integration. A software agent is a self-contained, autonomous software module that performs assigned tasks from the human user and interacts/communicates with other applications and other software agents in different platforms to complete the tasks. Four types of intelligent software agents (coordinating agents, task agents, data collecting agents, and user interface agents) are examined and discussed in the proposed MAERPS architecture. We demonstrate how the proposed prototype MAERP system takes advantage of existing information systems among various functional areas to achieve the system integration of commercially available enterprise resource planning (ERP) systems, while avoiding numerous problems encountered during a typical ERP implementation.

Keywords: Enterprise resource planning (ERP); Software agent; ERP architecture; ERP implementation

1. Introduction

Modern businesses are too complex and dynamic to be managed optimally using traditional information systems or even rigidly structured enterprise resource planning (ERP) systems. Agent-based systems claim to be next generation software capable of adapting dynamically to changing business environment and of solving a wide range of business problems in areas such as supply-chain management (SCM), health care and patient monitoring, process control applications, and knowledge discovery (Papazoglou, 2001). Software agents are sophisticated computer programs that act autonomously on behalf of their users to solve complex problems, and a multi-agent system is a loosely coupled network of software agents that interact to solve problems that are beyond the individual capacities or knowledge of each problem solver.

1.1. Purpose

The purpose of this paper is to explore the use of software agents to achieve the system integration of commercial ERP software packages. A multi-agent-based ERP (MAERP) architecture is proposed to take advantage of the existing information systems and to exploit the capabilities and characteristics of the software agent-based computer systems. The rest of the paper is organized as follows. In the rest of this section, we discuss the evolution in information systems as an enabling technology and establish software agents as an innovation to solve complex business problems. Next, a brief introduction to software agent technology is provided to highlight its properties, various types, and applications in order to establish its viability for developing an ERP type system. Third, the architecture of an MAERP system is proposed. Fourth, a simple example is provided to explain the workings of the proposed MAERP system. We show various phases through which the proposed MAERP system will handle a typical query. Finally, the implications of the MAERP system and future research directions are provided.

1.2. Research background

Since 1950s when transaction processing systems were first introduced, information systems have been successfully implemented in different functional areas, each with its own database and data architecture, to support decision making.

* Corresponding author. Tel.: +1-573-341-4184; fax: +1-573-341-4812.
E-mail address: leabi@umr.edu (B.-R. Lea).

0166-4972/$ - see front matter © 2003 Elsevier Ltd. All rights reserved.
doi:10.1016/S0166-4972(03)00153-6

Although such functional information systems have matured in terms of functionalities over the years of testing, modification, and maintenance, these systems have also caused problems such as data redundancy, information inconsistency and/or inaccuracy, and high system maintenance costs. In addition, such information systems are likely to result in poor decision quality due to the lack of cross-organizational perspective and communication difficulty (Jacobs and Whybark, 2000). Consequently, the whole organization may lose its competitive edges because it does not realize its full potential.

In the late 1970s and early 1980s, the need for enterprise wide integrated systems intensified as global competition became inevitable, and product customization and innovation became important factors to retain customers and subsequently to gain market share (Kalakota and Robinson, 2000). Systems thinking based management philosophies such as total quality management and just-in-time systems were introduced, which necessitated the management of relationships among functional areas and cross-organizational processes. As early as 1969, Blumenthal proposed a framework for an enterprise wide information system. The development of such systems slowly evolved from stand-alone systems (e.g., a standard inventory control system) to material requirement planning/manufacturing resource planning (MRP I and MRP II) systems, and subsequently, to enterprise wide systems to include other functional areas such as sales and marketing, financial accounting, and human resource management. However, attempts to provide a complete enterprise wide software solution were not successful until the mid-1990s due to technical complexity, lack of resource availability, and unclear vision (Kumar and Van Hillegersberg, 2000).

In the mid-1990s, the Gartner Group coined the term "ERP" to refer to next generation systems which differ from earlier ones in the areas of relational database management, graphical user interface, fourth generation languages, client–server architecture, and open system capabilities (Dahlen and Elfsson, 1999; Koch et al., 1999) The integration is normally implemented through the use of a common database among subsystems. All the subsystems "talk" directly to each other, and the data are made available in real time (Jacobs and Whybark, 2000). The information is updated as changes occur, and the new status is available for everyone to use for decision making or for managing their part of business. The decisions made in different functional areas are based on the same current data to prevent non-optimal decisions from obsolete or outdated data. Expected benefits from ERP implementation include lower inventory, fewer personnel, lower operating costs, improved order management, on-time delivery, better product quality, higher productivity, and faster customer responsiveness (Robinson and Wilson, 2001).

Most Fortune 500 companies have already adopted ERP systems, and many midsize companies (less than 1000 employees) are planning ERP implementation. Many organizations have successfully adopted ERP systems, yet many more organizations spent fortunes only to find that business performance has not improved to satisfactory levels within the expected time frame (Robinson and Wilson, 2001; Slater, 1998a). Problems associated with ERP implementation are often classified into technical and organizational aspects. Technical aspects include the technology readiness of an organization, the complexity of commercial ERP software, data loss due to the compatibility of data architectures between the old legacy systems and the new ERP software (Slater, 1998b), and adequacies of newly redesigned business processes (Oliver, 1999; Baatz, 1996). Common organizational factors may include employees' resistance to change, inadequate training, underestimated implementation time and cost, unwillingness to adopt new business processes, and strategic view of technology adoption (Slater, 1998b; Joshi and Lauer, 1999; Mabert et al., 2001).

During the last decade, ERP evolution continued to include inter-organizational processes (i.e., suppliers and customers) and to explore viable alternate approaches (e.g., adopt certain stand-alone or partially integrated functional software modules) by exploiting newly available technologies. For example, National Institute of Standards and Technology (NIST) of the US government supported a *Consortium for Intelligent Integrated Manufacturing Planning-Execution* (CIIMPLEX), and the consortium members reported their findings on an application of a software agent system to enterprise integration for manufacturing planning and execution (Peng et al., 2002). Intelligent business agents claim to be the next generation of model based solutions for business-to-business and E-Commerce applications (Papazoglou, 2001) and have been examined in various research areas, such as SCM (Ito and Saleh, 2000; Yu and Graham, 2002), health care and patient monitoring (Larsson and Hayes-Roth, 1998), process control applications (Van Dyke Parunak, 1998), knowledge discovery (Jensen et al., 1999), and decision support systems (Hess et al., 2000).

2. Software agent: properties, classification and applications

Researchers involved in agent research have used a variety of terms and offered definitions, explicating their use of each term. Although there is no general agreement as to what constitutes an agent, Franklin and Graesser (1996) provide taxonomy of autonomous agents and establishes how they are different from a computer program. In the literature, the term software agent is also referring to as "agent," "autonomous agent," intelligent agent," and "business agent." An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors (Rusell and Norvig, 1995: p. 33). The term "agent" is used to represent software-based computer programs that have two abilities: