Available online at www.sciencedirect.com
ScienceDirect
European Journal of Operational Research 187 (2008) 1310–1326
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

A Web-based ERP system for business services and supply chain management: Application to real-world process scheduling

C.D. Tarantilis [a,*], C.T. Kiranoudis [b], N.D. Theodorakopoulos [b]

[a] *Department of Management Science and Technology, Management Science Laboratory (MSL), Athens University of Economics and Business, 9th Floor, Room 913, 47A Evelpidon Street and 33 Lefkados Street, 11362 Athens, Greece*
[b] *Department of Process Analysis and Systems Design, National Technical University, 15780 Athens, Greece*

Available online 20 November 2006

Abstract

A Web-based ERP system developed for attacking business problems and managing real-world business processes ranging from simple office automation procedures to complicated supply chain planning is presented. The system's Web-aspect provides significant advantages, as the system is distributed through interoperable, cross-platform and highly pluggable Web-service components. The system involves a powerful workflow engine that manages the entire process event flow within the enterprise increasing efficiency and control at the same time. Business processes, when needed, are controlled by the enterprise quality management system and consequently the ISO directives are accurately followed. A real-world process scheduling system developed for the specific needs of Greek Construction Manufacturing Enterprises is illustrated as a detailed paradigm of the system's capabilities. The problem was formulated to assign project tasks in form of lots to enterprise resources in order that resources idle time and delays in project preparation time were minimized. The problem was solved by a simple and effective heuristic algorithm.

Keywords: ERP; Electronic services; Web-based supply chain management; Construction project management; Business process management

1. Introduction

During the last decade, there has been acknowledged a tremendous change in enterprise-oriented business software where traditional accounting and commercial management products have gradually left their place to integrated solutions that would rigorously deal with every single business aspect of each individual enterprise. These systems, typically called enterprise resource planning (ERP) systems, are usually developed as individual standalone monolithic applications or modular separable tools assembled in a Suite structure for all individual business needs. Most systems in this category have a similar client-server or multi-tier architecture built around a central database server. Typically, in traditional multi-tier standalone ERP applications, critical application components (such as the Application server) are essential for any functionality,

* Corresponding author. Tel.: +30 210 8203677; fax: +30 210 8816705.
E-mail address: tarantil@aueb.gr (C.D. Tarantilis).

0377-2217/$ - see front matter
doi:10.1016/j.ejor.2006.09.015

transaction or access to the database. On the other hand, a new generation of Web-based enterprise information systems is gradually gaining ground, where the system structure is entirely modular, pluggable and separable and no component or module is obligatory for the application's operation. The system presented in this article belongs to the second category, as it is developed on open-source Web-Development platforms and possesses modular and flexible structure.

Web-based techniques are less expensive, more efficient and lately have been the target of most development efforts. Additionally, web-access has been severely facilitated by recent advances in telecommunications and network technology that favor the creation of virtual private network (VPN) structures which unite different enterprise spatial entities (stores, warehouses, offices etc.). VPNs are networks constructed using external network infra-structure (usually the internet) to connect nodes. Web applications are cross-platform in terms of operation system and hardware requirements. Finally, web-based solutions are able to easily interoperate with the whole supply chain entity, consisting what is usually referred as Virtual Enterprise (the enterprise along with its main associates, customers and suppliers) [1].

Besides, ERP systems cover a wide range of functionalities ranging from accounting to commercial operations. In the proposed case, two major issues brought about new insight into business software solution functionalities. The first one is the adoption of quality management concepts in enterprise processes. In order to successfully implement quality management, organizations usually need to abide by the standards of quality management systems (such as ISO). Compliance with total quality standards is very important, since it guarantees that a specific company's products and services are meeting specific quality criteria. To adopt each particular quality standard, several business processes, mostly of bureaucratic nature, have to be carried out in order to coincide with the standard procedures that the enterprise actually follows. As a result, if one wants to control these processes digitally, one has to incorporate them in the main information system of the company, which most of the times is an ERP application. The second issue is the need for the systems to encompass more complex applications, which are vital to the enterprise efforts to exist in ample competitiveness between its market rivals. Critical modules in this direction are the ones that solve Supply Chain Management (SCM) and Logistics problems.

The information system presented in this manuscript possesses two basic parts. A workflow engine that manages the entire business process task flow across the enterprise, incorporating quality management rules and a resource management module, which utilizes a project scheduling management sub-system. The scheduling problem is solved by a suitable heuristic algorithm.

In Section 2, a short list of combined supply chain and workflow management related work is provided. Section 3 discusses the ERP and Web-services integration issue. In Section 4, a detailed description of the general solution framework is presented. Section 5 provides the analysis of the project scheduling algorithm. An illustrating example of one of the system's implementations is reported in Section 6. The conclusions of the text are finally presented in Section 7.

2. Literature review

2.1. Combined business process and supply chain management

Although ample Workflow Management Software systems (WfMSs) exist as stand-alone software applications or even as modules of several types of information systems, either on the commercial software area or in the academic field, the occasions where WfMSs are consolidated with SCM software solutions in the same system or even in many integrated systems, are not proportionately numerous. Nevertheless, there have been some cases in the literature where solutions have been developed combining WfMS and SCM features. For instance, Liu et al. [2] developed an inter-enterprise (Virtual Enterprise) Workflow Supply Chain Management information system, using a light-weight workflow engine and covering ordering, inventory and outsourcing operations of the virtual enterprise. Marquardt and Nagl [3] proposed some concepts on a consolidated platform called CRC IMPROVE, regarding integration of chemical process management software systems with flowsheet and other application tools. However, the problem of job time scheduling in a manufacturing environment and implementing workflow management system solutions at the same time on the same information system, remains in general unexplored.

2.2. Latest developments in ERP systems

ERP systems and their journey along with SCM have been studied in the literature though not extensively. Akkermans et al. [4] produced a research in 23 separate firms about the results and future expectations of ERP systems implementations in a SCM perspective. Among the conclusions of the research was that one major drawback of then contemporary ERP systems, was the lack of Web-based, modular, separable ERP systems and lack of extended (or virtual) enterprise operation capabilities. In this context, one drawback subcategory of the non-modularity of ERP systems is the lack of modular management of the supply chain. Olhager and Selldin [5] presented a survey concerning the implementation of ERP systems in Swedish firms. Moreover, they compared the survey results with similar research conducted in the US. The authors inferred among others that many firms deploying ERPs considered extending system scope mainly to integrate their suppliers, customers or both to the system, to provide additional e-commerce or e-business operations and to increase supply chain functionalities. Ng and Ip [6] shaped an object oriented model based on the traditional N-tier logic for designing Web-based ERP applications. Kelle and Akbulut [7] pointed out the main disadvantages of typical ERP systems in terms of supply chain management and among these is the inability to support complete information flow across the extended enterprise.

ERP systems were formed as integrated standalone software systems incorporating materials requirements planning, accounting, order entry, distribution and shop floor control functionalities. After that, ERP systems (or extended ERP systems) began accumulating other mainly supply chain related functions, such as demand forecasting, scheduling, warehousing, capacity requirements planning and logistics. In the last years, two very important tendencies are monitored regarding ERP systems, the one relates to the addition of even more functionalities, such as project management, content management, workflows, enterprise portals, customer relationship management, human resource management and knowledge management and the other is linked to the need to disintegrate large-scale ERP applications to autonomous, easily pluggable, reusable and loosely-coupled application components. Accordingly, ERP systems may comprise from parts independent from each other, which will plug and operate like Lego bricks into enterprise systems in contrast to previous closed non-modular architectures. As a result, an enterprise will not have to acquire the whole enterprise software suite, but will be able to choose each module even from different vendors and create a unique, cost-efficient and tailor-made solution. This concept can be realized through Web-Service related methodologies such as SOA (service oriented architecture) and Web-development techniques in general. These directions and evolutions can be traced on the documentation of grand international enterprise software vendors like Oracle, SAP, PeopleSoft, www.salesforce.com or even in smaller outfits like Exact Software Inc or Hyperion. Some vendors have developed distinct products in this philosophy such as the Oracle Fusion Middleware of Oracle.

3. ERP systems and Web services

Enterprise resource planning software systems attempt to integrate all departments and functions across a company onto a single information system that can serve all those different departments' particular needs. Typically, a department with specialized functions and needs may have its own information system, customized to its particular procedures and duties. Nonetheless, the main effort of an ERP implementation is to combine as many functionalities as possible into a single, integrated software program that runs on a single database, in order that the various departments can easily share information and communicate with each other. This approach can have a tremendous payback if companies implement the software properly. Customer order is an illustrating example of a process benefiting from implementation of such systems. Normally, when a customer places an order, it begins a mostly paper-based journey around the company, often being transferred into different software systems along the way. This process may incur delays and errors. Meanwhile, access to critical order information is not directly available to everyone involved when needed. The combination of traditional ERP functions enriched with Business Process Management and SCM functionalities substitutes specialized department operation software systems, reducing costs, complexity and information inconsistency. Most vendors' ERP software is flexible enough to provide modular implementation capabilities.

ERP systems can accelerate business processes execution times, such as that of ordering process mentioned earlier. This is the main concept behind the system proposed and described in this paper. Information flow is realized through a powerful database driven workflow engine, described in detail in the following section.

In the context of ERP, Web Services offer two crucial advantages: ease of integration and cost reductions through the hosted application model. Integration is a major source of expenditure for enterprises due to business software system complexity. Customers and outsourcing vendors may demand access to information provided to internal ERP users—such as order status, inventory levels, and invoice data—even without having an ERP client software. This is where Web Services interfere, enabling seamless data access to the authenticated users at the right time from everywhere without the need of specific software clients. With the availability of Web Services, integration can be achieved with superior reliability, security, manageability, testing and effectiveness. Web Services use object-oriented technology to mix data and programming elements in Web Service methods that can be accessed by different applications. Web Services enable proprietary applications to communicate over the Web. Proprietary ERP applications and Web Services can use integration or other tools—such as SAP's Netweaver, HP's E-Speak toolsets; IBM's Dynamic e-business (infrastructure and software); and Sun ONE (Forte technology and iPlanet's ECXpert)—all of which facilitate data flow and communication across diverse applications. Web-Services combined with ERP provide an integrated, multi-component application software platform ideal for performing multiple business functions.

The enterprise may still require a traditional ERP application for its internal operations. Web Services allow the enterprise to access needed information effectively. As this new technology gains ground, more vendors support Web Service solutions blended with XML technologies. Subsequently, Web Service broker hubs are introduced. A broker hub offers a portal providing a user interface for users so that they can locate, evaluate, subscribe to, cancel, manage, or monitor Web Services. An increasing number of business software vendors are delivering Web Service broker hubs, such as SAP and Oracle for users of mySAP and Oracle E-Business Suite respectively. Intuit and Peachtree offer them with QuickBooks and Peachtree Complete Accounting. Microsoft with Navision is a vendor offering a Web Service broker hub for mid-tier ERP software users.

Fig. 1 shows a common three-tier development structure, which is the most widespread form of N-Tier architectures. According to 3-tier logic, three different layers (tiers) exist, which are independent parts of executable code, presentation, business logic and data layers. Presentation layer is responsible for managing user-interface and sometimes security issues. Business logic layer controls all data transfer between the user-interface and the data tier, retrieving, modifying, storing, deleting and validating data. Data tier is a tier responsible for the database or sometimes is the database itself. Usually, data and business logic tiers comprise the application server of a distributed information system.

The structural differences between a typical N-tier ERP and the proposed Web-based ERP can be understood by comparing Figs. 1 and 2. In contrast to most standalone ERP systems, where all or some of the integral application components, such as the application server, are essential for their operation, in our web-based scaleable approach, each user logins into the system via a web-portal and

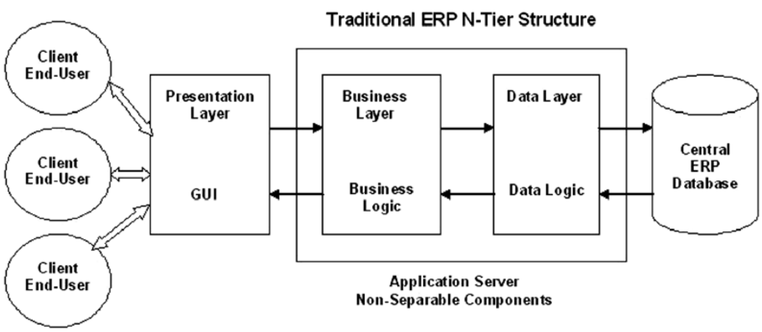

Fig. 1. Architecture of traditional standalone N-Tier ERP software applications.

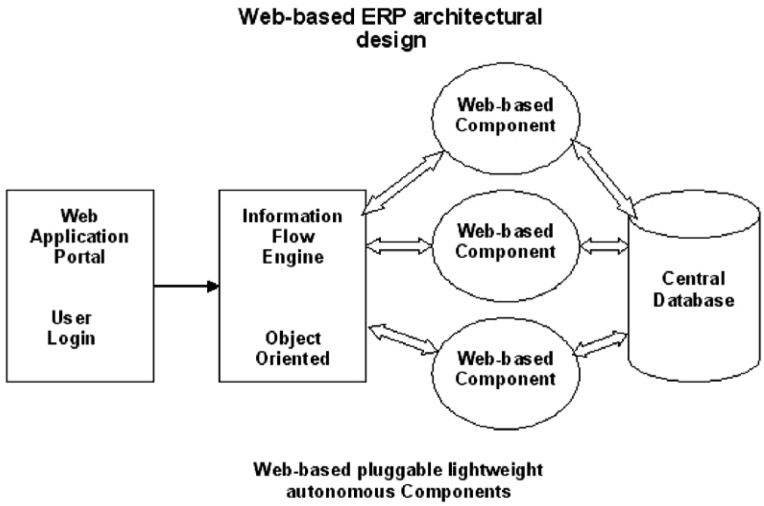

Fig. 2. Architecture of the proposed Web-based ERP with separable software modules.

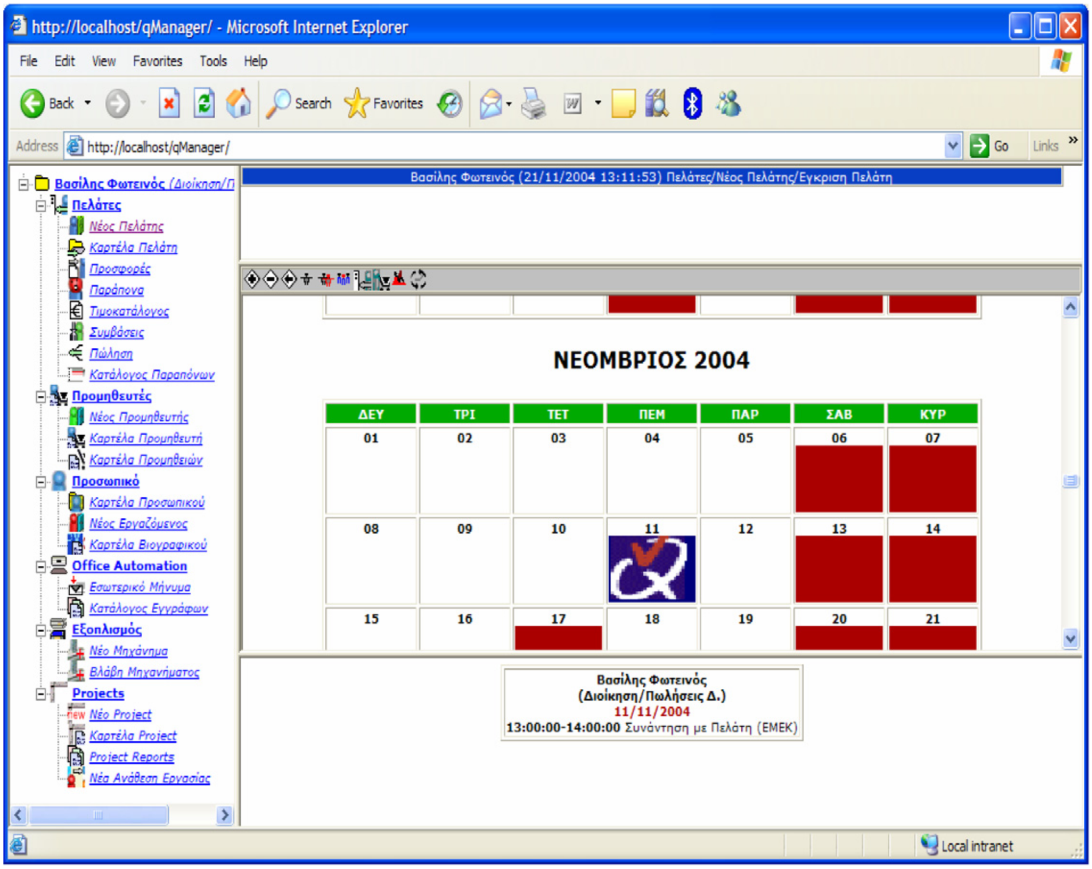

Fig. 3. IE user interface of the tool involving: process/tasks tree frame control, notification event tags notification, calendar frame for personnel programmed tasks with detailed description of calendar events.

then information flow components (including the embedded workflow engine described in the next section), navigate the user through the independent system's functionalities. Each web-service component is autonomous and is easily pluggable even across applications.

In Fig. 3, the main graphical user-interface of the information flow management module of the application is presented. Three areas appear clearly in the figure, on the left side, there is the business process initiation area where each user of the information system is able to initiate the business processes he/she is entitled to. According to the system's philosophy even a simple data form (for example the main customer form) may represent a business process with only one phase. Thus, the accessibility of data as well as the initiation of business processes is controlled by this web-based component. In the other two areas of the user-interface on the right, the upper first presents the events that stem from business process stage activities. These events emerge and lead to other web data forms or non-initiating business process dialogs. The user-interface area below is linked with resource and time management, containing from simple calendars to project/production management Gantt charts.

The system was developed using open source Active Web Server Pages Technology (PHP and JavaScript) and ANSI SQL. One of the system's main requirements is the existence of an RDBMS database application, preferably Oracle, MS SQL Server or MySQL.

4. Web-based business processes and system implementation

In this section, the business process management aspect of the proposed Web-based ERP system is presented. Within this context, many information systems have been developed during the last years that interact with business processes of enterprises. The labels of such systems vary from Workflow Management (WFM), Business Process Management (BPM), to Enterprise Document Management (EDM) and Product Data Management (PDM) systems [8]. Numerous definitions of such systems are provided in the literature [9]. In general, most such systems handle workflows as an abstract entity within the enterprise and do not interact with the actual management of the involved resources. There is also a wide variety of Workflow engines, components that provide the mechanisms to manage workflows dynamically. Although one of the system's core activities is Business Process Management, its functionality goes beyond workflows, to other functionality areas such as Operations or SCM.

Companies need business process management systems in order to automate, enhance and optimize workflows, but also need quality management systems to improve quality in every business activity and to control and standardize business processes. Additionally, successful Quality Management Systems implementation increases performance standards and customer satisfaction. Moreover, quality certifications improve product and brand image and sometimes offer a significant competitive advantage to companies. During a Quality certification procedure, an enterprise should document its business processes in order to certify them according to a Quality Management System's standards (ISO for example). Besides, during that process, a company has to record its organizational structure, job-descriptions and organizational charts. Therefore, such procedures give exposure to business processes and present an excellent opportunity for the enterprise to enhance or re-engineer them. Through the job-description phase, job positions are described in detail and roles of employees in the enterprise are defined. Roles are associated with tasks. All this type of data is the main configuration input of the proposed Web-based ERP.

The whole procedure of combined implementation of the proposed Web-based ERP and the Quality Management System is depicted thoroughly in Fig. 4. In this figure implementation phases are shown from top to the bottom of the figure, requirements (input), customization, deployment, resulting modules (output). The certification according to a quality standard is a corollary of putting into practice the system's Quality Management module and consequently is also shown in the figure near the output modules.

On the other hand, Quality Systems that are not supported with a software solution are very rigid and entail much paperwork. As a result, Quality Systems become slow, difficult to control, in-flexible, sometimes even chaotic and eventually inefficient. Consequently, the need arises for information systems, which help enterprises to build and operate a Quality Management System and at the same time streamline workflows, encounter real-life business problems and manage the actual resources of the enterprise.

The workflow engine is the component of the proposed Web-based ERP information system which handles the control-flow (or process) perspective of the business process management module of the system [10]. That means that the workflow

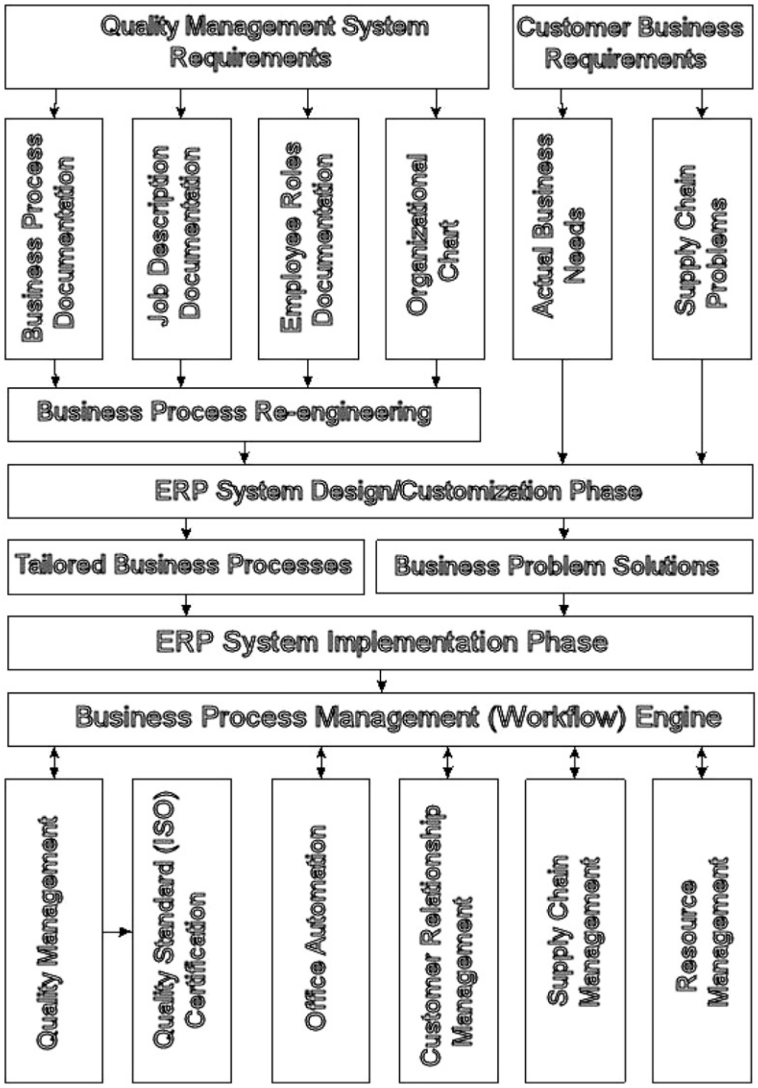

Fig. 4. Structural representation of the system's implementation phases.

engine manipulates the flow of tasks and business events among the groups and users of the system, defining the sequence followed.

The workflow engine has been developed using Object Oriented Programming (OOP) techniques and is comprised of several objects that interact with a relational database and appropriate documents and files, producing sequences of tasks and events, navigating business workflows. PHP scripting language provides most required object-oriented capabilities similar to those of standalone software languages.

The main class components of the workflow engine that formulate a corresponding Application Programmers Interface (API) library are the following:

- *Session* class (*bHierarchical, bByPass* properties, *suggestChild( ), evolve( )*methods)
- *Process* class
- *Task* class
- *Event* class
- *DataClass* class

The *Session* class object contains user information, which also contain group information. Usually, *Tasks*, which in Petri-net terminology are described by places, are assigned to specific groups

[10,11]. *Groups* may represent actual operational departments of a company such as the sales department, or specific job-positions, or even virtual entities in the company. Furthermore, *Tasks* cannot be assigned to groups rather than to specific users/resources of the system. *Events*, which in Petri-net terminology have a similar function with tokens, wait to be enacted in each user's event queue. When a user interacts with an *Event* and causes an action that will proceed the *Process*, *suggestChild*( ) method gets called. This method's role is to navigate the workflow, choosing the correct next *Task* or *Tasks* and finding the appropriate users or groups. Group or user assignment to a *Task* can also be dynamic. For instance, it can be decided by the input of the user of the previous task. Moreover, when the *bHierarchical* property is true, *suggestChild*( ) can seek the next users or groups hierarchically. For example, in a procurement process, every department manager approves the procurement orders of the department members. Then when a procurement order is placed by a user, the system through *suggestChild*( ) method seeks, following the hierarchy through the organizational chart of the company, for a director who is entitled to offer an approval for the procurement order. When the *bByPass* property is enabled, the system can by-pass a *Task* or a *Process* if there is no user available to execute the *Task*. For example, a *Task* in a business *Process* can be by-passed when the resources (users) who have the privilege to perform that *Task* are on vacation. The workflow engine supports AND-splits & joins as well as OR-splits & joins, respectively [10]. Any *Task* can have unlimited number of results and be considered as initial. Then, *evolve*( ) method gets called to create the new *Events*, to send them to the appointed groups or users and to destroy the previous old *Events*. In Petri-net terminology, *evolve*( ) method is used to trigger tokens. An *Event* can be linked to specific data records in tables of the relational database or to specific documents in the system or both. The *Session* object can stop any process when it is demanded.

The *Process* class object is a reusable object component which usually depicts a specific real-life business process. Business processes comprise of a sequence of *Tasks*, which are elementary explicit units of work performed by a user or a group of the system. They may include other sub-processes. The business object acquires the business process data from the relational database.

The *Event* class object represents a link between two consecutive *Tasks* and notifies the resources (groups or users) involved in the execution of a pending *Task*. The *Event* class object provides also information about forms, documents and data related to the task through the *Document* and *DataClass* class objects. The *DataClass* class object, which is used extensively by the system even outside the scope of the Workflow Engine, is the object that operates as the main communication tool with the relational database. A typical workflow mechanism of *process/task/event* creation structure is presented in Fig. 5.

Roles and authorities to execute specific *Tasks* or to initiate *Processes* can derive automatically from the organizational chart of the company or the job-descriptions of the system users. This input data is part of the Quality Management System and needs to be defined to comply with its requirements. The proposed Web-ERP provides the ability to manage any resource of the enterprise (employees, machines, vehicles, materials) either on programmed or on actual schedule (timesheets). In other words, the system tracks not only what was planned, but also what really happens. Moreover, it provides us with a real-time costing ability for every activity of the enterprise.

Apart from the three main functionality areas, the tool operation covers the following modules.

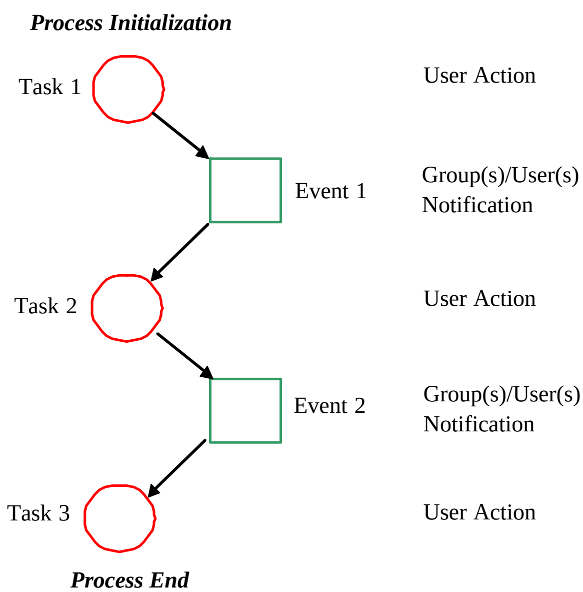

Fig. 5. Workflow mechanism of process/task/event creation structure.

- Office Automation
- Customer Relationship Management (CRM)
- Supply Chain Management (SCM)
- Human Resources Management (HRM)
- Equipment Management (EM)
- Quality Management (QM)

The presented tool has been implemented with great success on over 40 large, small and medium Greek enterprises solving core operational problems on the following sectors: Construction, media, retail, manufacturing, education, software, consulting, financial services, transportation and insurance.

5. Management of the manufacturing process

One the most powerful properties of the proposed Web-ERP tool is the ability to integrate through its workflow and resource planning processes, several supply chain problems. One of the most significant examples is the manufacturing process management tool developed for several Greek manufacturing firms. The infrastructure of the production process in most manufacturing construction companies (such as the ones forming metallic constructions and building hardware) usually entails the simultaneous operation of several major departmental sections that prepare final ordered products. The main characteristic of products prepared by such firms is that although almost none has the same structure, they are made using similar manufacturing procedures. Orders dispatched by the central ordering system, usually suited to customer desire, involve products not known beforehand whose preparation is likely to combine many operations. Moreover, according to the preparation schedule of each product, there are cases where several sections of resources are involved. In addition to the above, the course of manufacturing action is not known before the order is placed [12].

In construction manufacturing systems such as the one discussed here, there are two basic requirements posed by the production department that runs the operation. The first one involves the time priority of each order. Typically, if a client poses an order at a given time, it is assumed that this order is prepared with higher priority than others dispatched later. This rule is actually followed in the production process of the enterprise to achieve a smooth operation. The second one involves the minimization of order completion times, subject to limitations imposed partly by resources in equipment and personnel and above all by the individual characteristics of the manufacturing process. In most such enterprises, compliance with both criteria depends on the experience of the operating personnel. The proposed system utilizes an efficient module for scheduling manufacturing processes, which schedules all manufacturing operations. In fact, it is supplied with the production profiles of each task, resources available regarding equipment and personnel and customer orders. Subsequently, it advises on the reliable course of action in each section with respect to the following rules:

- each order is scheduled so that it is completed in the least possible time, that is to say order makespan is minimized;
- all tasks in the same order will commence at the same time, unless this is limited by resource availability in each production section;
- time priorities between orders are kept, unless the current set-up situation allows an order to be completed before others of higher priorities without affecting their schedule.

The most characteristic feature of the proposed scheduling system is the fact that ordering information is dispatched by the system to manufacturing sections at specific time intervals. As a result, scheduling decisions instantly affect the scheduling procedure which manages manufacturing resources [13–16].

The complete manufacturing task description according customer's desire and needs involves the preparation of each task in phases that block time in each available resource in a specific way. For each phase, there is a job t assigned to it that involves times t_S^t (start time) and t_F^t (finish time). In the convention used in the algorithm of this manuscript, time within jobs starts at job completion point and moves backwards. Thus, the reference zero point for measuring job preparation time is placed at the task completion time point. The reason for this convention is obvious; job completion is the only valid reference time measuring point regarding job preparation. Obviously, start and finish times for each job can result in various job patterns within each task. Jobs can be overlapping, they can involve the same resource type in parallel or sequentially, etc. Each job can be accomplished by resources of one and only one specific resource type r_i, which belongs to a set of resource types $R = \{r_1, r_2, \ldots, r_N\}$. For each resource type r_i, there is a number of similar resources n_{r_i}, of identical characteristics. As a result, each task phase can be executed in any resource that belongs to the resource type associated with it. Using this notation a task p, is a set of objects (jobs) where each one is characterized by a start and finish time and an associated resource type, $p = \{(t^i_S, t^i_F, r_i),\ i = 1, \ldots, n_P\}$. An order s, is a set of tasks $s = \{(q_{p_i}, p_i),\ i = 1, \ldots, n_s\}$. The algorithm developed, is used to schedule a set of orders S of different dispatch times on the available set of resources R, whereas the number of available resources for each resource type n_{r_i} varies with time, due to equipment malfunctions during operation or the presence of available personnel operating the equipment.

Supposing that an infinite number of resources per resource type was available. The maximum finish time of all tasks in each order would obviously be the order make-span. In addition, all orders would be scheduled to achieve this makes-span. In the real-world system, two sources of delay will appear. If an order involved a large number of tasks, their processing on limited resources would definitely not allow their simultaneous completion. In this case, the second rule mentioned above would be violated, and certain tasks would simply wait for others to be executed, due to the limitation of available resources. This is the first source of delay. In this case, the second rule mentioned above would dictate that all delayed tasks would be executed as close to order dispatch time as possible. In this case, the order make-span is the maximum of finish times plus task delay over all tasks in the order. The second source of delay is caused by the fact that when the order scheduling pattern is determined, fitting the order to the available resources might reach a point where no available time in resources could be found due to the presence of other orders. In this case, older orders have higher priorities and cannot be displaced in time. Therefore, order completion should be delayed until all tasks are assigned to available resources. The proposed system follows two basic principles; time priority of each order and make-span pattern of each order. The implementation of these principles and the evaluation of total order delay in available equipment is carried out in the following two phases for the tasks of each order in the departmental operation schedule. Algorithm formulation is presented in relevant pseudocode functions. The operation of department schedule simply iterates these two phases for each order dispatched by the enterprise ordering system.

```
for each order i dispatched, order by order_dispatch_time
begin
  scheduleOrder(i);
  fitOrder(i);
end
scheduleOrder(i)
begin
  set p as the set of tasks of order i; //copy
  from order tasks set
  while p is not empty
    delay=0;
    set s as the task of maximum finish time
    that belongs to p;
    for all jobs t in s
      set r the associate resource type of job t
      for all machines m in resource type r
        if s does not fit in any machine m
          delay++;
    end
    s.delay = delay;
    remove s from p;
  end
end
fitOrder(i)
begin
  set p as the set of tasks of order i; //copy from
  order tasks set
  delay = 0;
  for all tasks s in order i ordered as searched in
  scheduleOrder(i);
    for all jobs t in s
      set r the associate resource type of job t
      for all machines m in resource type r
        if s delayed by s.delay does not fit in
        any machine m
          delay++;
    end
  end
end
```

The algorithm suggests that each order put forward by the company dispatching system is iteratively treated through two discrete phases. The first one assumes that all resources are available for all order tasks and relevant jobs, inserting task jobs to resources without considering the presence of other orders placed before occupying resources. It greedily selects tasks of decreasing total finish time and puts them in the first available resource adjacent to order dispatch point in order that each one is completed at exactly order completion time.

If this is impossible due to resources already blocked by tasks and jobs selected previously in the order, the task (and its jobs) is shifted (delayed) in time until non-occupied resources are found. The delay between tasks in each order is recorded. The second phase keeps delays constant between tasks of the same order as evaluated in the previous phase and strives to fit each order task job in available resource machines. If even one job cannot find free resource time placement, the entire order is shifted (delayed) until all tasks and jobs are assigned to available resources. Furthermore, the order delay is recorded again. As a result, the two sources of delay, delay between tasks of the same order and delay between orders, are easily tracked and recorded. In all cases, the delay is measured from the exact time point where the order was placed in the system. The most common delay unit is an hour, but the algorithm can function with any other time subdivision.

A typical example of the Gantt chart resource calendar is presented in Fig. 6. When applied to an actual manufacturing process of 3 resource types with 3 resources for each type, the calendar takes the form of Fig. 6. Each horizontal row within a calendar day represents a resource. Furthermore, each horizontal colored bar illustrates a specific job and each different color shows a different order. Grey lines between chart horizontal rows denote the division between resources of different resource types. The figure divides time in working days and working hours.

In Fig. 7, a problem with eight resource types and three available resources for each type is shown. A zoomed frame of a specific day gives greater detail of the assignment of tasks to production lines. The numbers on each bar present the job's order, task and job numbers. The user is able to see the detailed start and finish times for each job and also to form the actual, real-life production schedule.

Typically, the workflow process works as follows; once in a working day production planners approve scheduling plan suggested by the algorithm and after the end of each shift, the system is informed about the actual task completion and the actual task start and finish times. In this way, an

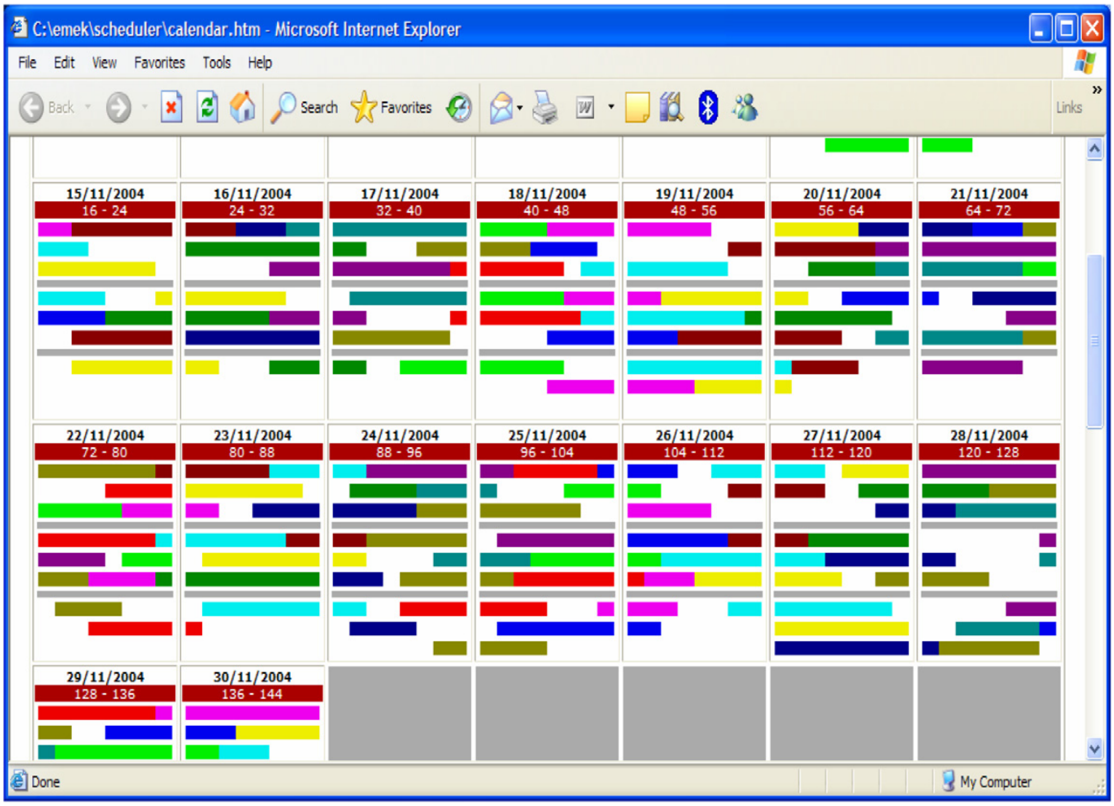

Fig. 6. Typical Gantt Chart Calendar for three resource types and 8 hours shift.

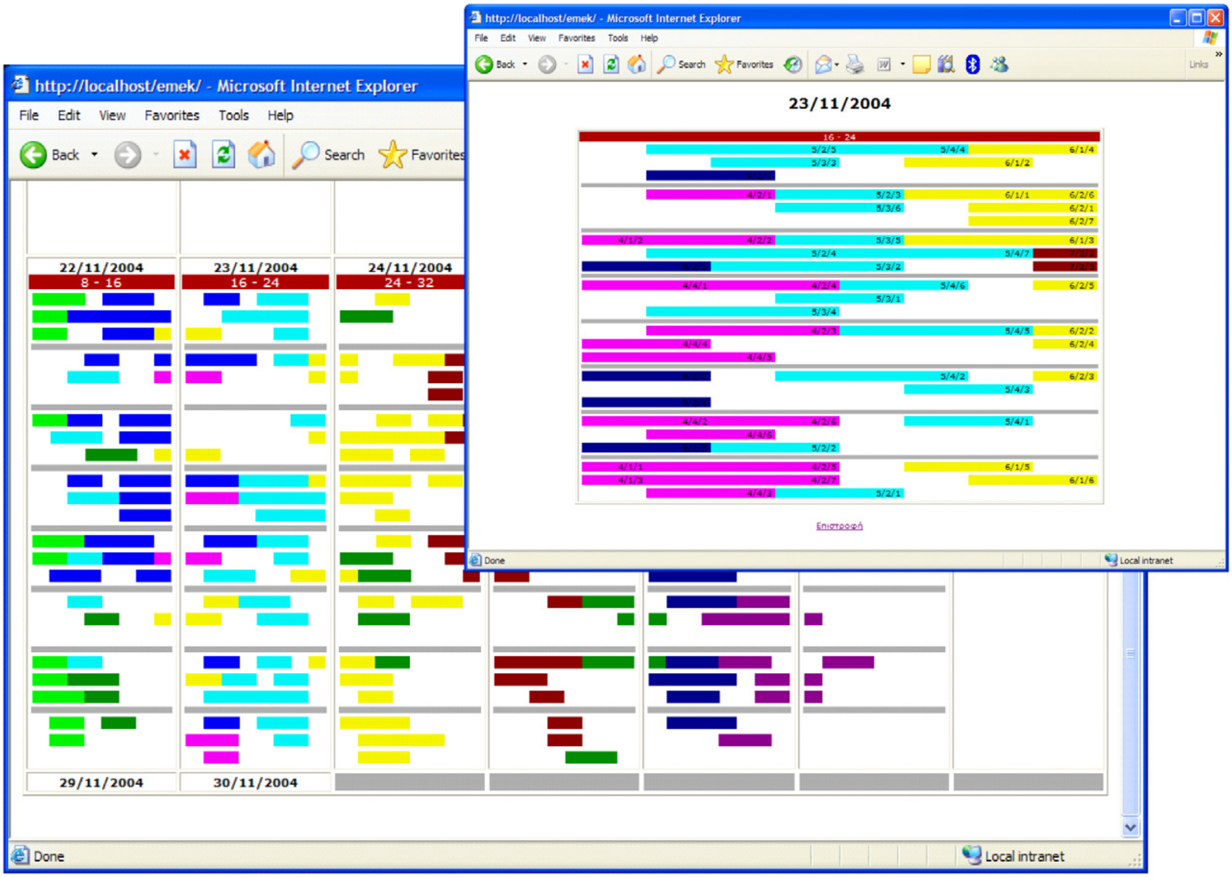

Fig. 7. Actual Gantt chart from an 8-resource type manufacturing process involving a detailed task and job dialog.

on-going process of planning and confirming is put into practice in order to monitor planned and actual tasks. Accordingly, reports on discrepancies between planned and actual process times can be generated.

6. Case study

6.1. Background

A case will be presented where Manufacturing Management integrates with Business Process Management through the information system's utilization. The case study concerns one of the biggest Greek construction companies which focuses on the creation of metallurgical constructions. The enterprise had turnover over 30 million euros in 2004 and played a key role in the development of the infrastructure that was used in the facilities of the Athens 2004 Summer Olympic Games. The enterprise is located in the area of Attika near Athens capital of Greece and operates on great public or private infrastructure works (for example stadiums or large bridges) and at the same time has several industrial customers, such as metal, mining and cement industries, supermarkets, power-plants, shipbuilding enterprises, oil refineries and others. Additionally, the enterprise operates inside and outside of Greece. Furthermore, this company employs approximately 200 persons. Some of them work on construction sites and others in the company's manufacturing facilities.

6.2. System implementation

In this example, this enterprise implemented the Web-based ERP system to streamline many of its critical processes including processes concerning company operations like procurement and project management or processes concerning customers like customer complaints. The business processes involved in the project are presented in detail in Table 1. The system supported both the Greek and the English language, because many of the

Table 1
Typical business processes for the ERP presented

Business process	Pre-ERP conditions	ERP implementation results
Customer complaints	Customer complaints records were paper ISO forms. Management had almost no control	The whole process is automated through the system's workflow engine. Management is notified instantly
Procurement	Procurement requests of various departments were transferred tediously via ISO forms across the enterprise to acquire director and upper management approval. No control, coordination and reporting abilities were available	Every involved employee is notified automatically, all procurement orders are approved by directors and the upper management electronically. Real-time reports available for every procurement request in the enterprise. Process execution times reduced dramatically
Production planning	Production scheduling was manual and based on empirical and intuitive criteria. Production planning team estimated delivery times and schedules, which were inflexible and sometimes not accurate enough	Production scheduling is formed automatically by the algorithm presented in Section 5. The system provided flexible schedules and lower delivery-times. The system offers ability of instant rescheduling with every new order. The input data stem directly from sales department
Quality control	Quality control of final or middle products was kept on ISO forms on hard-copy. No statistical analysis available	Quality control is kept in the system, various methods of data statistical analysis and reporting available
Sales	Promotion activities (calls, meetings, faxes) were arbitrary and there was no data available about their effectiveness. Offers and contracts were on hard-copy form only	Each promotion's activity result is measured. Successful promotion actions are automatically connected with resulting offers and even contracts. Reports are available and decision making is enhanced. Contract, offer data is widely accessible and transparent

employees were not Greek and used Oracle 8i as a relational database management system (RDBMS).

One business process where workflow management co-operated with production planning was the customer offer process. The upper management of the enterprise was severely concerned about the operation of the sales department. Therefore, they expressed a will to quantify and monitor customer relationship data and promotion activities, including customer calls, meetings, offers, offer modifications, after-sales communication and above all contracts. All these types of actions formed a single business process to our system, called the offer process. Moreover, the fact that each contract which involved one or more projects was fully customized and different from anything else produced by the company in the past, incurred the need to transfer contract and project related information directly to the operations department of the enterprise. Furthermore, most customer orders were very sensitive in terms of delivery dates and as a result the enterprise faced a need to forecast reliable project delivery times and to optimize production scheduling.

Consequently, a customized offer process was developed on the system as illustrated in Fig. 8. One of the tasks of this process was the manufacturing management procedure described in the previous section. When the offer process leads to a contract, then the operations department of the enterprise is instantly notified by the system and related project information is added to the input of the scheduling algorithm. In this way, not only were new orders added to the new production schedules, but also production management had the ability to prioritize orders automatically, to

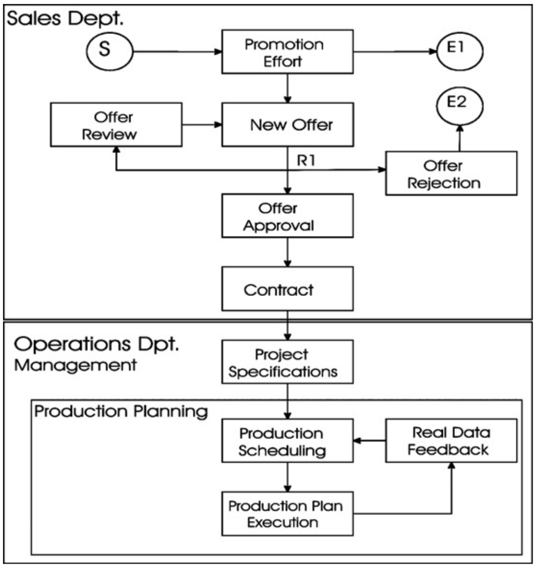

Fig. 8. Business process diagram presenting the case study, where workflow management is integrated with the production scheduling module of the system.

interact with different production planning scenarios and to project accurate assessments of each order's delivery times.

The enterprise has several production lines and each of them is entitled to perform different sets of job-types. Moreover, the organization faced a project scheduling problem, where multiple fully customizable projects needed co-ordination and cost-reduction, both in the company's manufacturing facilities and in the customer's site. Additionally, the firm encountered tight deadlines and sometimes rising costs due to the high use of overtime. Each production line has a definite number of critical resources. Resources that define a production line's capacity are called critical. On some production lines critical resources may be employees and on others machines. Thus, each production line represents a resource type (r_i) and each critical resource, machine or employee, which belongs to the previously mentioned production line forms a member of the set of similar resources (n_{r_i}), as described in Section 4. Production phases (job-types) are divided to lots. A lot is the minimum inseparable amount of work and when it begins processing it should end without interruptions and when it ends the created stock is transferred to the next phase. Lot-sizes for each production line and task are defined by the production planning team. In the algorithm description of Section 5 jobs play the role of lots. The production planning team requested a small modification of the algorithm presented in Section 5. They wanted to be able to prioritize orders regardless the time they entered the order queue in the system. For instance, if a large contractor was willing to pay a very high price, as long as its project is finished in short deadlines, this order could have a greater priority than older ones with more flexible time schedules. Thus, priority grades were introduced into the system and the scheduling algorithm, meaning that among orders of the same priority grade, the older one gets processes first, but an order of a higher priority grade precedes all orders of lower priority grades.

The proposed Web-based ERP contains each piece of productive equipment and employees and their job-descriptions in the database for quality management purposes. Accordingly, tasks, jobs and job-types are defined for each project (order) during the project definition phase as depicted in the Fig. 8. In this figure, S represents process initiation point, $E1$ occurs when promotion efforts do not result to a new offer, $E2$ happens when an offer is rejected by the customer, $R1$ is an OR-split event-condition-action (ECA) point where the following cases arise:

- If an offer needs review, the offer is altered and resent to the customer.
- If an offer is rejected, offer rejection state is reached.
- If an offer is approved the process proceeds to the contract stage.

The user of the system is able to execute the scheduling algorithm. Additionally, the user is able to alter lot-sizes of each task at any time. After each execution, the planning team puts the schedule into practice, then inspects the actual result of the production activities, feeds the actual data input into the system and re-schedules the production plan periodically or whenever seems useful. The planning team of the enterprise decided to follow this procedure on daily basis.

Apart from the business process management and manufacturing management modules of the Web-based ERP, the company utilized other useful modules of the application, such resource management and quality management. The enterprise had already implemented a quality management system (ISO) and had obtained certification of various quality standards including ISO. As a result, the quality management system module of the Web-base ERP was adjusted to the company's already operating quality system.

6.3. Project scheduling

Production line available resources range from one to five depending on project execution phase. The phases and the job type-related critical

Table 2
Project execution phases (job-types)

Index	Phase	Critical resource type
1	Cutting	Equipment
2	Welding (agglutination)	Equipment/personnel
3	Shaping	
4	Puncture/perforation	Personnel
5	Assembly	Personnel
6	Electric welding	Personnel
7	Grinding	Personnel
8	Painting	Personnel
9	Fitting	Personnel
10	Shear nailing	Equipment

Table 3
Problem description, estimated work hours per project per phase

	Phase	Resources	1	2	3	4	5	6	7	8
1	Cutting	3	80	240	40	120	240	80	120	140
2	Welding	3	120	120	40	120	60	240	80	60
3	Shaping	2	40	160	40	120	40	400	40	40
4	Perforation	2	60	200	24	120	24	400	54	40
5	Assembly	3	40	40	48	120	8	400	60	40
6	Electric welding	5	160	120	16	120	240	400	240	260
7	Grinding	3	240	120	24	120	120	400	140	260
8	Painting	2	48	40	24	120	80	240	120	24
9	Fitting	3	40	24	24	120	14	160	40	24
10	Shear nailing	2	24	16	24	120	20	200	40	24

resources are presented in Table 2. At most phases there are up to three work shifts per day. Additionally, minimum and maximum overtime can be chosen in problem formulation. A particular instance of the problem and its subsequent solution will be studied.

The problem is composed of eight concurrent projects, each of them containing all 10 project execution phases. Detailed problem data is presented in Table 3. A common lot-size is chosen for all phases of 8 hours and all phases in all projects are considered successive. That means that a project execution phase can initiate only after a processed lot has been received by the previous one. Each project has unique deadlines and for each phase a different number of critical resources is available. Production lines work in three 8-hour shifts and no overtime is into consideration.

Before the advent of the system, due to the large complexity of the problem, no precise estimates were achievable. Nonetheless, there was an elementary project planning process, according to which work hour estimations were added per project execution phase. The sum was divided to the number of available critical resources for the specific phase. The phase with the maximum ratio was defined as the production bottleneck and this phase defined the resulting plan. In the case of the problem described in Table 3, bottleneck phase is the grinding operation. Exact project execution phase initiation times could not be calculated, but a safety time space was considered between production phases from 1 to 5 business work days. In our case the old planning method produced a schedule of a total duration of 800 work hours. On the contrary, the proposed system produced a detailed planning solution described in Table 4 in 600 work hours. For a typical plan number of projects in process ranges from 4 to 55 with up to 10 project execution phases.

Table 4
Problem of Table 3 system project planning results in work hours

Project	1	2	3	4	5	6	7	8
Execution time	240	240	120	208	320	600	384	504

With this kind of input the scheduling algorithm can produce results is less than a second in a typical client PC. In general, in several real-life examples, total project execution times were reduced from 0% to 40% and detailed work hour cost reports were available. A comparison was made between the results of 22 past project schedules and the ones that would have been produced by the system bearing in mind problem scope (total number of projects). The results are presented in Table 5, showing the average project completion time improvement for each category.

6.4. ERP implementation benefits

Through the use of the system the enterprise obtained the following results:

- Enhanced company operation through streamlining, improving and controlling business processes of major importance such as procurement, cus-

Table 5
Relationship between completion time improvement and problem scope

Total number of projects	1–10	11–20	21–30	31–40	41–50	>50
Schedules in this category	3	8	5	3	1	2
Completion time improvement (%)	9.2	10.4	22.3	12.3	8.5	29.5

tomer offers, customer complaints, equipment maintenance, marketing campaigns and others.

- Significant cost-reductions and time-savings in all the above mentioned business processes.
- Ability to manage service related personnel and related costs through the use of the resource management module of the system. In the past, the company could allocate only the productive resources' (workers and construction site related costs) cost to each company activity. Now by taking advantage of the resource management (timesheets) module of the system, the enterprise is able to manage the cost of the service personnel (engineering, R&D departments, etc.) involved.
- Upgraded use of the company's already operating quality management system, which was not supported by an information system. The use of the proposed ERP system enabled the enterprise to avoid much paperwork, to reduce personnel's occupation times with quality management issues and to provide report insight to the management.
- Flexible and efficient production planning by implementing the manufacturing management (scheduling) module of the system. Project delivery times and idle times were reduced significantly, productivity was raised, more precise delivery time assessment incurred stock level minimization and customer satisfaction improved.
- Facilitated communication and data, transfer of critical information for the whole enterprise. Now employees have instant access to real-time data, documents and reports that concern their duties. View of information flow is fully customized according to each user position.
- Finally, the company exploited the abilities to control sales and promotion activities through the system, received quantitative data about the results of each promotion technique and managed to increase sales department efficiency.

7. Conclusions

In this paper, a novel Web-based ERP system developed for attacking business problems and managing real-world supply chain problems was presented. It involved a powerful workflow business process mechanism and a complete enterprise resource management engine. The system implementation includes a bridge between business process management and production scheduling. Moreover, information system capabilities are presented through an illustrating case study on one of the system's implementations, revealing a hands-on view to the system implementation benefits.

This work may present an opportunity to integrate genuine and innovative business problem solutions within the context of the latest trend in ERP systems as well as in enterprise software in general, which is the utilization of Web-based techniques. For instance, an extension of this work may be the employment of intelligent vehicle routing problem (VRP) solution methodologies to an integrated Web-based ERP system specialized for the needs of transportation management.

References

[1] G. Dimitrios, S. Hans, C. Andrzej, Managing process and service fusion in virtual enterprises, Information Systems 24 (6) (1999) 429–456.

[2] J. Liu, S. Zhang, J. Hu, A case study of an inter-enterprise workflow-supported supply chain management system, Information & Management 42 (2005) 441–454.

[3] W. Marquardt, M. Nagl, Workflow and information centred support of design processes—the IMPROVE perspective, Computers and Chemical Engineering 29 (1) (2004) 65–82.

[4] H.A. Akkermans, P. Bogerd, E. Yucesan, L.N. van Wassenhove, The impact of ERP on supply chain management: Exploratory findings from a European Delphi study, European Journal of Operational Research 146 (2) (2003) 284–301.

[5] J. Olhager, E. Selldin, Enterprise resource planning survey of Swedish manufacturing firms, European Journal of Operational Research 146 (2) (2003) 365–373.

[6] Johny K.C. Ng, W.H. Ip, Web-ERP: The new generation of enterprise resources planning, Journal of Materials Processing Technology 138 (2003) 590–593.

[7] P. Kelle, A. Akbulut, The role of ERP tools in supply chain information sharing, cooperation, and cost optimization, International Journal of Production Economics 94 (2005) 41–52.

[8] I. Choi, C. Park, C. Lee, Task Net: Transactional workflow model based on colored Petri Net, European Journal of Operational Research 136 (2002) 383–402.

[9] M. Weske, W.M.P. van der Aalst, H.M.W. Verbeek, Advances in business process management, Data and Knowledge Engineering 50 (2004) 1–8.

[10] W. van der Aalst, K. van Hee, Workflow Management, Models, Methods, and Systems, The MIT Press, 2002.

[11] E. Gudes, A. Tubman, AutoWF – a secure Web workflow system using autonomous objects, Data & Knowledge Engineering 43 (2002) 1–27.

[12] S. Bischof, E.W. Mayr, On-line scheduling of parallel jobs with runtime restrictions, Theoretical Computer Science 268 (2001) 67–90.

[13] E. Naroska, U. Schwiegelshohn, On an on-line scheduling problem for parallel jobs, Information Processing Letters 81 (2002) 297–304.
[14] C.D. Tarantilis, C.T. Kiranoudis, A list-based threshold accepting method fro job shop scheduling problems, International Journal of Production Economics 77 (2002) 159–171.
[15] D. Ye, G. Zhang, On-line scheduling with extendable working time on a small number of machines, Information Processing Letters 85 (2003) 171–177.
[16] N. Megowa, A.S. Schulzb, On-line scheduling to minimize average completion time revisited, Operations Research Letters 32 (2004) 485–490.