

# Healthcare Enterprise Process Development and Integration

Kemafor Anyanwu, Amit Sheth, Jorge Cardoso, John Miller, Krys Kochut  
LSDIS Lab, Department of Computer Science  
University of Georgia  
Athens GA 30602-7404, USA  
Contact: amit@cs.uga.edu, Tel: 706-542-2310, Fax: 706-542-4771

## Abstract

Healthcare enterprises involve complex processes that span diverse groups and organizations. These processes involve clinical and administrative tasks, large volumes of data, and large numbers of patients and personnel. The tasks can be performed either by humans or by automated systems. In the latter case, the tasks are supported by a variety of software applications and information systems which are very often heterogeneous, autonomous, and distributed. The development of systems to manage and automate these processes has increasingly played an important role in improving the efficiency of healthcare enterprises.

In this paper we look at four healthcare and medical applications that involve investigative, clinical, and administrative functions. Based on these applications, we derive the requirements for developing enterprise applications that involve the coordination of a variety of tasks performed by humans, information systems, and legacy applications.

**Keywords:** workflow management systems, healthcare processes, healthcare applications-clinical pathways, immunization tracking, eligibility referral, genome sequencing

## 1 Introduction

The recent push for healthcare reform has caused healthcare organizations to focus on ways to streamline their processes in order to deliver high quality care while at the same time reducing costs. This has precipitated a review and upgrade of clinical and administrative protocols and the increased use of information systems to improve the efficiency of certain processes. Since processes are fundamental building blocks of an organization's success, information technologies that focus on process management and improvement are good candidates for helping healthcare organizations fulfill their corporate vision.

In the past two decades, a special interest has been taken in Workflow Management Systems (WfMSs) as a tool to streamline, automate, and re-engineer business processes. There are many workflow products which adequately support relatively simple processes, such as document management, form processing, and imaging. However, they fall short in meeting the challenges of mission-critical processes, which are often complex,

dynamic, large-scale, and QoS-based (Sheth, Georgakopoulos *et al.* 1996; Cardoso, Sheth *et al.* 2002). These qualities are typical in healthcare processes.

Healthcare processes are very complex, involving both clinical and administrative tasks, large volumes of data, and a large number of patients and personnel. For example, an out-patient clinic visit involves administrative tasks performed by an assistant and clinical tasks performed by a doctor or by a nurse. For an in-patient hospital visit, this scenario involves more activities, and the process entails a duration that lasts at least as long as the duration of patient hospitalization.

Healthcare processes are also very dynamic. As processes are instantiated, changes in healthcare treatments, drugs, and protocols may invalidate running instances, requiring reparative actions (Berry and Myers 1998; Shrivastava and Wheeler 1998). For example, a care pathway for a patient with disease condition ‘A’ may need to be changed as new drugs are discovered.

Large-scale processes often span multiple healthcare organizations and run over long periods of time (Dayal, Hsu *et al.* 1991). This type of process requires highly scalable workflow systems to support large instances (Bonner, Shruf *et al.* 1996). Furthermore, these large-scale processes often need to be integrated with legacy information systems and with distributed, autonomous, and heterogeneous computing environments (Georgakopoulos, Hornick *et al.* 1995); thus, they require support for transactional features and error handling (Worah, Sheth *et al.* 1997).

Another important requirement is the management of Quality of Service (Cardoso 2002; Cardoso, Sheth *et al.* 2002). Healthcare organizations operating in modern markets require Quality of Service (QoS) management. Services with well-defined specifications must be available to patients. An appropriate control of quality leads to the creation of quality care services; these, in turn, fulfill patient satisfaction.

This paper discusses the use of the METEOR workflow system for managing mission-critical healthcare processes. The workflow management and enterprise application integration techniques developed in the METEOR system are intended to reliably support complex, dynamic, large-scale, and QoS-based workflow applications in real-world, multi-enterprise, and heterogeneous computing environments.

An important aspect of the METEOR project is that technology and system development efforts occurred in close collaboration with industry partners. Key healthcare partners have included the Connecticut Healthcare Research and Education Foundation (CHREF), the Medical College of Georgia (MCG), and the Advanced Technology Institute. These collaborations have generated a detailed study of healthcare workflow application requirements, the prototyping of significant healthcare workflow applications with a follow-on trial, and the evaluation of METEOR’s technology.

This paper is structured as follows. In section 2, we discuss the current generation of information systems to support healthcare processes, and we highlight some shortcomings of these systems. Section 3 describes the METEOR system, and section 4 discusses four healthcare workflow applications that use the METEOR system to meet requirements. Section 5 summarizes the benefits of the METEOR approach. Finally, section 6 presents our conclusions.

## 2 Supporting Healthcare Processes with the Current Generation of Workflow Systems

Traditionally, healthcare processes have been managed using limited forms of workflow. Some examples of these are clinical and administrative protocols. However, these “protocols have remained limited in their usefulness in part because developers have rarely incorporated both clinical and administrative activities into one comprehensive care protocol. This lack of integration hinders the delivery of care, as the effectiveness of protocols is often dependent on many administrative tasks being properly executed at the correct time” (Chaiken 1997). Consequently, many healthcare organizations are now turning to workflow management techniques to help improve the efficiency of their work processes.

The trend toward computerizing business processes has led to a large number of commercially available information systems, some of which specifically target the healthcare sector. These systems offer various levels of process support, functionality, and robustness.

At one end of the spectrum, we have customized workflow application systems that support human-oriented and vertical group processes. These processes typically involve a relatively few number of tasks which are executed in a predefined sequence and which require few roles in a single group of an organization. In these types of applications, the process model is embedded in the application, and customers need to configure the application in order to tailor it to their specific process.

Some examples include VMI medical (2002), which offers a pediatric cardiology workflow system; TeleTracking (2002), which enables hospital administrators and staff to effectively manage, coordinate, and deliver quality care to patients; and the Soarian (2002) system, which synchronizes workflows across the entire enterprise and orchestrates patient care by bringing together clinical, financial, therapeutic, and diagnostic information.

Another class of applications at this end of the spectrum focuses on supporting information and document management functions. These applications are usually built on the top of data management systems which are designed to capture, store, retrieve, and manage unstructured information objects such as text, spreadsheets, audio clips, images, video, files, and multimedia.

Some examples include CareFlowNet (2002), which provides for the creation, management, and delivery of medical documentation, and SoftMed (2002), which provides a suite of applications for clinical data management, patient information management, and document acquisition and storage.

At the other end of the process support spectrum, we have workflow management systems which are more general purpose systems. These systems provide tools for process definition, workflow enactment, administration, and for the monitoring of workflow processes.

Research prototypes include METEOR (Kochut, Sheth *et al.* 1999), MOBILE (Jablonski 1994), ADEPT (Reichert and Dadam 1998), EXOTICA (Mohan, Alonso *et al.* 1995), and MENTOR (Wodtke, Weissenfels *et al.* 1996). Commercial products include MQSeries Workflow (IBM 2002), Staffware (Staffware 2002), TIBCO InConcert (TIBCO 2002), and COSA Workflow (COSA 2002). General information on workflow

systems can be found at the Workflow and Reengineering International Association (WARIA 2002) and the Workflow Management Coalition (WfMC 2002) Web sites.

The current generation of workflow systems adequately supports administrative and production workflows (McCready 1992), but they are less adequate for some of the more horizontal healthcare processes which have more complex requirements. These types of processes are dynamic and involve different types of tasks; these can be human-oriented, associated with legacy applications, or associated with database transactions. The processes are large-scale, cross-functional, and cross-organizational, where the different participating groups have distributed and heterogeneous computing environments. Workflow infrastructures to support such processes are limited. This is mainly because many systems have a centralized client/server architecture and support only static processes. They also lack support for features such as exception modeling and handling and QoS management.

Another very important requirement that current workflow systems seldom provide is an integration environment. It is clear that the different functional groups of a healthcare organization may require different types of applications to support their processes. For example, integrating Picture Archiving and Communication Systems (PACS) with hospital or radiology information systems will allow radiologists to be presented with collateral patient information. This allows for patient history, clinical information, symptoms, and the previous examination history to be presented to the physician along with images retrieved from the PACS, greatly aiding in the interpretation of images (DeJesus 1998).

Healthcare organizations typically have various information systems, including legacy applications, that are used routinely and need to be integrated. Unfortunately, many workflow systems of the current generation are based on closed, proprietary architectures. This makes supporting interoperability and integration a complicated, if not impossible, task.

The METEOR system was specifically developed to provide a solution to the problems outlined previously. It supplies an infrastructure that supports mission-critical enterprise-wide processes and that integrates heterogeneous, autonomous, and distributed information systems. A general description of the system is given in the next section. For a comprehensive and detailed description, the reader is referred to Miller, Palaniswami *et al.* (1998) and Kochut, Sheth *et al.* (1999).

### **3 The METEOR System**

The METEOR (Managing End to End Operations) system leverages Java, CORBA, and Web technologies to provide support for the development of enterprise applications that require workflow management and integration. It enables the development of complex workflow applications which involve legacy information systems and that have geographically distributed and heterogeneous hardware and software environments, spanning multiple organizations. It also provides support for dynamic workflow processes, error and exception handling, recovery, and QoS management. The METEOR system has been successfully used to prototype and deploy several healthcare applications. Our success is due in part to extensive collaboration with our healthcare industry partners.

The METEOR system includes all of the components needed to design, build, deploy, run, and monitor workflow applications. METEOR provides the four main services shown in Figure 3-1: the Builder, the Enactment, the Repository, and the Manager services.

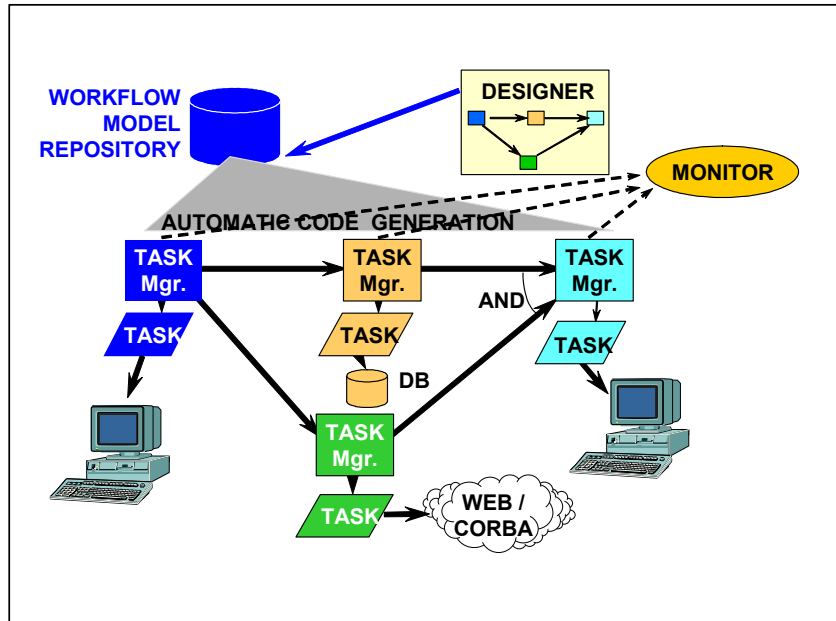


Figure 3-1: METEOR system architecture

### 3.1 The Builder Service

The builder service supports the graphical design of workflows (Lin 1997; Zheng 1997). It includes three main components. The task design component provides interfaces with external task development tools, such as Microsoft’s FrontPage to design the interface of a user task. The network design component is used to set dependencies, data objects, and transition functions among tasks; it is also used to define security domains and roles. The data design component allows the user to specify data objects that are employed in the workflow.

The service outputs an XML-based representation of process definitions which may be formatted to be compliant with the Workflow Process Definition Language (WPD) of the Workflow Management Coalition (WfMC 2002).

### 3.2 The Enactment Service

There are two enactment services provided by METEOR – ORBWork (Kochut, Sheth *et al.* 1999) and WebWork (Miller, Palaniswami *et al.* 1998). Both services use a fully-distributed open architecture. WebWork is a comparatively light-weight implementation that is well-suited for traditional workflows’ help-desk and data exchange applications. ORBWork is better suited for more demanding, mission-critical enterprise applications

which require high scalability, robustness, exception-handling support, QoS management, and dynamic modifications.

### **3.3 The Repository Service**

The repository service maintains information about workflow definitions and associated workflow applications. The builder service tools communicate with the repository service to retrieve, update, and store workflow definitions, thereby providing support for rapid application development in the builder service. The builder service tools are capable of browsing the contents of the repository and incorporating fragments (either sub-workflows or individual tasks) of existing workflow definitions into the one currently being created.

A detailed description of the first design and implementation of this service is presented in Yong (1998), and a XML based implementation is described in Arpinar, Miller *et al.* (2001).

### **3.4 Management Services**

The tools provided by these services are used for administering and monitoring workflow instances. The administration service is used by the workflow administrator to perform management functions, such as installing and configuring workflow instances, load-balancing, and modifying workflow processes in execution. The monitor provides a tool for querying and viewing the state of workflow instances.

### **3.5 METEOR's Advanced Features**

#### **Automatic Code Generation**

METEOR has a suitable code generator (Miller, Palaniswami *et al.* 1998) that is used to build workflow applications from the workflow specifications generated by the builder service or from those stored in the repository. The code automatically generated from the workflow design stage greatly minimizes the steps required to implement the workflow. This frees the designer from having to worry about details of communication or about data passing among existing tasks.

#### **Fully Distributed System**

The fully distributed architecture of METEOR yields significant benefits in the area of scalability. METEOR's architecture has three major advantages. First, it allows for the support of workflow processes that are geographically distributed. Secondly, it provides support for load-balancing among all the participating host machines. Finally, it eliminates the existence of a single point of failure within the system.

#### **The Use of Standards**

The METEOR system closely follows the specification and interoperability standards set by bodies such as the WfMC (WfMC 2002) and the Object Management Group (OMG 1998). METEOR also supports workflow interoperability standards such as JFLOW

(JFLOW 1998) and SWAP (Swenson 1998), and it utilizes CORBA<sup>1</sup> due to its emergence as an infrastructure of choice for developing distributed object-based applications.

### **Security**

METEOR provides various levels of security, from role-based access control and authentication, to multilevel security (MLS). A MLS workflow system enables globally distributed users and applications to cooperate across classification levels in order to achieve mission-critical goals. Users can program multilevel mission logic to securely coordinate distributed tasks and to monitor the progress of workflows across classification levels (Kang, Froscher *et al.* 1999).

### **Dynamic Changes**

The METEOR system has a layer that permits consistent realization of the dynamic change of instances (Chen 2000). The module guarantees that all consistency constraints which have been ensured prior to a dynamic change are also ensured after the workflow instances have been modified (Reichert and Dadam 1998).

The features designed to handle dynamic changes in workflows are also very useful in supporting scalability, as the load increases. For example, an administrator may decide to move a portion of a running workflow to a new host (or hosts) that have become available for use.

### **Error and Exception Handling**

Error and exception handling, and the recovery framework (Luo 2000)(Worah, Sheth *et al.* 1997) have been defined in a scalable manner. The most advanced component developed was the exception-handling mechanism, which works in the following way. During a workflow execution, if an exception occurs, it is propagated to the case-based reasoning exception handling module; the CBR system is used to derive an acceptable exception handler (Luo, Sheth *et al.* 1998). The system has the ability to adapt itself over time, based on knowledge acquired about past experiences which help solve new problems. As the CBR system collects more and more cases, the global WfMS becomes more and more resistant, thus preventing unwanted states.

### **QoS Management**

The METEOR system allows for the specification of quality of service metrics and requirements (Cardoso 2002; Cardoso, Sheth *et al.* 2002). The implementation of mechanisms to specify workflow quality of service (QoS) is a major advance for METEOR. The system includes a workflow QoS model, estimation algorithms and methods, and monitoring tools. The model allows suppliers to specify the duration, quality, cost, and fidelity of the services and products to be delivered. The available algorithms estimate the quality of service of a workflow, both before instances are started and during instance execution. The estimation of QoS before instantiation allows suppliers to ensure that the workflow processes to be executed will indeed exhibit the

---

<sup>1</sup> A new version of ORBWork that uses RMI instead of CORBA is currently being implemented.

quality of service requested by customers. The analysis of workflow QoS during instance execution allows workflow systems to constantly compute QoS metrics and register any deviations from the initial requirements.

#### 4 Healthcare Applications Prototyped Using METEOR

The healthcare sector has a number of different types of organizations. By healthcare sector we mean both hospital and non-hospital based organizations. Examples of non-hospital based organization are the pharmaceutical companies and laboratories. All these organizations have different requirements. Table 1 gives a summary of the different types of processes, the applications that support them, and their requirements.

	Processes	Example Applications	Requirements
<b>Hospital Based</b>	Clinical	Charting, Scheduling, Discharge Summaries, Reports	Integration with patient data management software; Management of human and automated activities; Exception handling; Ease of use; Support for Dynamic Changes; Security; Role-Based Authorization; QoS management
	Non-Clinical (Administrative and Financial)	Ordering Systems (radiology, pharmacy) Patient Management (billing, accounts receivable, claims filing)	Data Management and Integration; Application Integration; Support for Heterogeneous and Distributed Environments; Security; Support for standards (e.g. EDI and HL7); Exception Handling.
<b>Non-Hospital Based</b>	Laboratory	Laboratory Information Systems	Scalability; Exception Handling; Management of complex data types; Transactional Workflows; Integration with other systems; Support for HAD environments; QoS management
	Pharmaceutical Industry	Clinical Drug Trial Management	Distributed Environment; Scalability; Exception Handling; QoS management

**Table 4-1. Healthcare Processes and Applications**

The rest of this section describes four applications that we have prototyped using the METEOR system. These applications support different types of processes, varying in scale (*i.e.*, number of tasks and roles, and requirements ranging from single server to multiple distributed servers), workflow execution across different workflow system installations, integration of legacy applications, access to databases, and QoS management support.



products. Timing and temporal constraints were specified in the application design, and their logic was programmatically coded.

Since the support for advanced features, such as the integration of legacy applications, was not a requirement, this application was developed using the WebWork enactment service of the METEOR system. WebWork allows for the deployment of a simple infrastructure installation with a low cost, and for easy administration.

The application uses three distinct types of tasks: human, transactional, and non-transactional tasks. Human tasks are accessed through web-enabled clients. Transactional tasks control the access to an Oracle database which contains patient information. Non-transactional tasks execute custom-developed applications which perform specific actions inside the workflow process. Examples are the scheduling of ultrasound exams and the calculation of temporal deadlines.

## 4.2 GeneFlow

GeneFlow was developed specifically for the needs of the Fungal Genome Initiative. This is a multi-institution consortium of research groups which is mapping and sequencing the genomes of important fungal organisms.

GeneFlow is a workflow application that handles the needs of data analysis for genome sequencing. Raw “shotgun” DNA sequence data consists of short overlapping DNA sequences. This data comes from automatic sequencing machines. From this raw data, the short overlapping shotgun sequences must be synthesized into larger contiguous sequences of whole chromosomes. These larger sequences are searched for probable genes and other chromosomal features. The results are then electronically published, with the objective of making the annotated genomes available in the public domain.

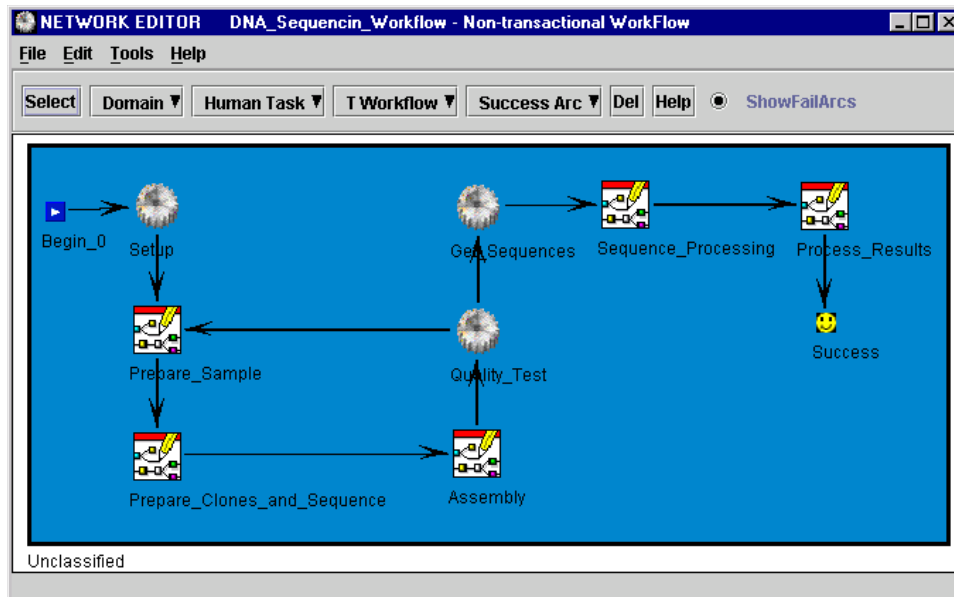


Figure 4-2. Workflow design for GeneFlow

Genomic projects involve highly specialized personnel and researchers, sophisticated equipment, and specialized computations involving large amounts of data. The characteristics of the human and technological resources involved, often geographically distributed, require a sophisticated coordination infrastructure to manage not only laboratory personnel and equipment, but also the flow of data generated.

The quality of service management is an important factor for this application (Cardoso 2002). The laboratory wishes to be able to state a detailed list of requirements for the service to be rendered to its customers. As an example, requirements may include the following constraints:

- The final report has to be delivered in 31 weeks or less, as specified by the customer (e.g., NIH).
- The profit margin has to be 10%. For example, if a customer pays \$1,100 for a sequencing, then the execution of the GeneFlow workflow must have a cost for the laboratory that is less than \$1,000.
- The error rate of the task *Prepare Clones and Sequence* has to be at most  $\epsilon$ , and the data quality of the task *Sequence Processing* has to be at least  $\alpha$ .
- In some situations, the client may require an urgent execution of sequencing. Therefore, the workflow has to exhibit high levels of reliability, since workflow failures would delay the sequencing process.

In this application, METEOR tools are used to wrap genome data analysis applications together in a “genome data assembly line.” Three heterogeneous platforms (SGI, Solaris, and Windows) are used with a single database and a single workflow system. The process requires many human and automated tasks, support for legacy applications integration, and Web-based access to support geographically distributed users. The integration of legacy applications on the SGI, Solaris, and Windows platforms was accomplished by writing Java wrappers for the legacy tasks. These wrappers were then easily integrated with the ORBWork enactment service.

The genetic workflow application presented underlines QoS management requirements. It is necessary to analyze the QoS of workflows during the design phase and also during the execution of instances. At runtime, the system monitors instances and registers any deviations from the initial requirements. When deviations occur, the dynamic change interface can be used to adapt workflow instances, with the goal of restoring their QoS to acceptable metrics.

### 4.3 Eligibility Referral

The Eligibility Referral application was developed for the Connecticut Healthcare Research and Education Foundation (CHREF) to support the process of transferring a patient from one hospital to another. It involves three organizations, two hospitals, and an insurance company.

The design depicted in Figure 4-3 shows a consolidated workflow, including the activities carried out by both the sending and the receiving hospitals.

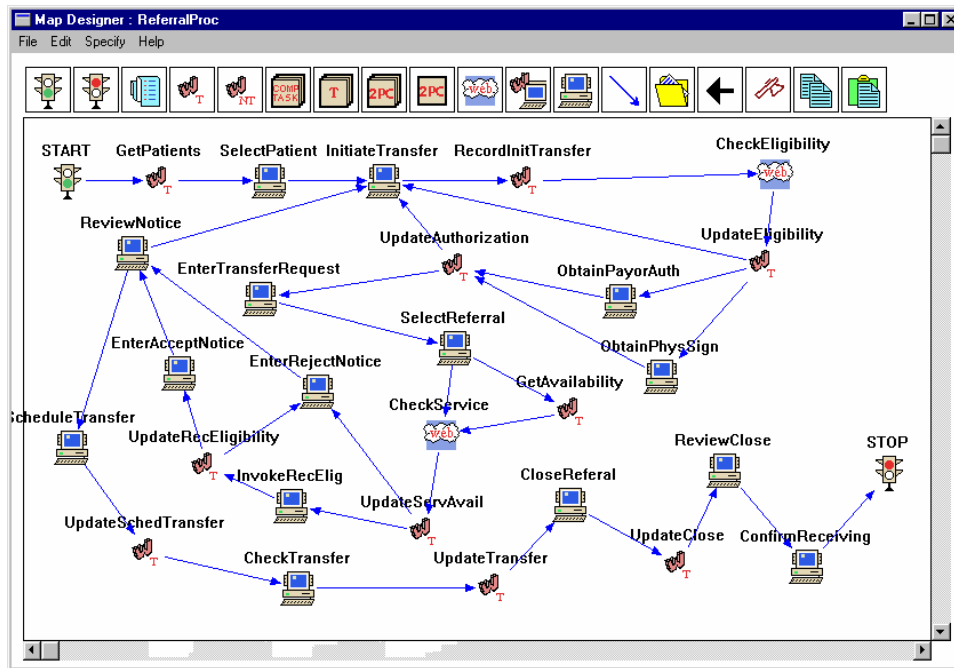


Figure 4-3. Eligibility Referral Workflow

The workflow starts with the sending hospital trying to determine the right placement for a patient that needs to be sent out. Once this is done, the next tasks involve determining the eligibility information, obtaining the necessary payment information, and also getting the physician’s signature for a specific patient. The final step in the sending hospital’s workflow is to receive an acknowledgment from the receiving hospital indicating that it will accept the patient. Once this is done, the sending hospital can update its database, and the receiving hospital will take over from there. The receiving hospital also has its own workflow for processing transferred patients. Workflow instances spans across the two hospitals, interacting with the insurance company through EDI transactions.

The Eligibility Referral application requires an infrastructure that supports distributed and heterogeneous environments. Workflow instances must be managed across multiple workflow system installations. The application accesses multiple databases and web servers. Furthermore, it requires an infrastructure that supports heterogeneous tasks such as human, automated, and transactional tasks with EDI transactions.

In our implementation, we have deployed separate METEOR systems – one for the sending hospital and one for the receiving hospital. A single workflow instance executes tasks across both the hospitals. Each hospital hosts its own web server and a database. The databases are used to find data about patients in order to verify eligibility information.

#### 4.4 State-Wide Immunization Tracking

The Immunization Tracking application has the most advanced requirements of all four examples discussed. The enhancements of healthcare management, with the development

of performance monitoring and reporting frameworks, made childbirth reporting and immunization tracking an important rating criterion. According to the Health Plan Employer Data and Information Set (HEDIS), the childhood immunization rate is one of the most important elements that define the quality of care.

Healthcare resources must be used efficiently to lower costs while improving a high quality of care. To accomplish these goals, healthcare processes need to be automated. Figure 8 shows the scope of the Immunization Tracking application in a schematic form.

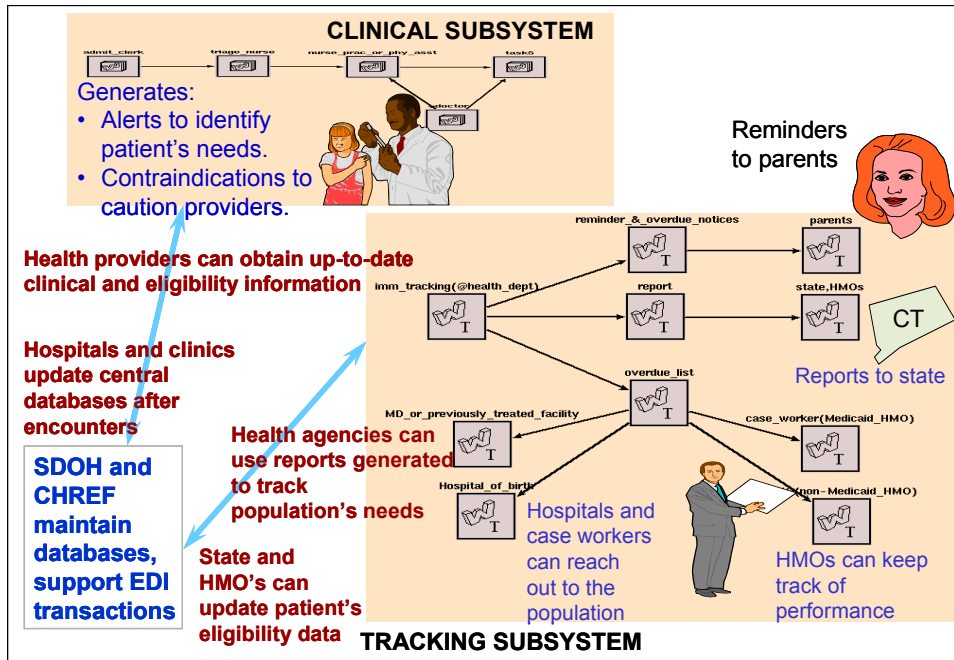


Figure 4-4. Schematic view of the Immunization Tracking application

The process includes on-line interactions for the workflow application between CHREF (the central location), healthcare providers (Hospitals, Clinics, and home healthcare providers), and user organizations (State Department of Health (SDOH), schools, and Department of Social Services (DSS)).

The system can be best explained in terms of the clinical and tracking subsystems shown in Figure 4-4. The clinical subsystem has been designed to provide the following features:

- Roles for Admit Clerk, Triage Nurse, Nurse Practitioner, and Doctor;
- Worklist for streamlining hospital and clinic operations;
- Automatic generation of Medical Alerts (e.g. delinquent immunizations) and Insurance Eligibility Verification by the Admit Clerk; and
- Generation of contraindications for patients visiting a hospital or clinic to caution medical personnel regarding procedures that may be performed on the patient.

The tracking subsystem involves reminding parents and guardians about shots that are due or overdue and informing field workers about children who have not been receiving their immunizations.

Additionally, health agencies can use the available data to generate reports (for submissions to authorities such as the DSS and State Government) and to determine the health needs of the state.

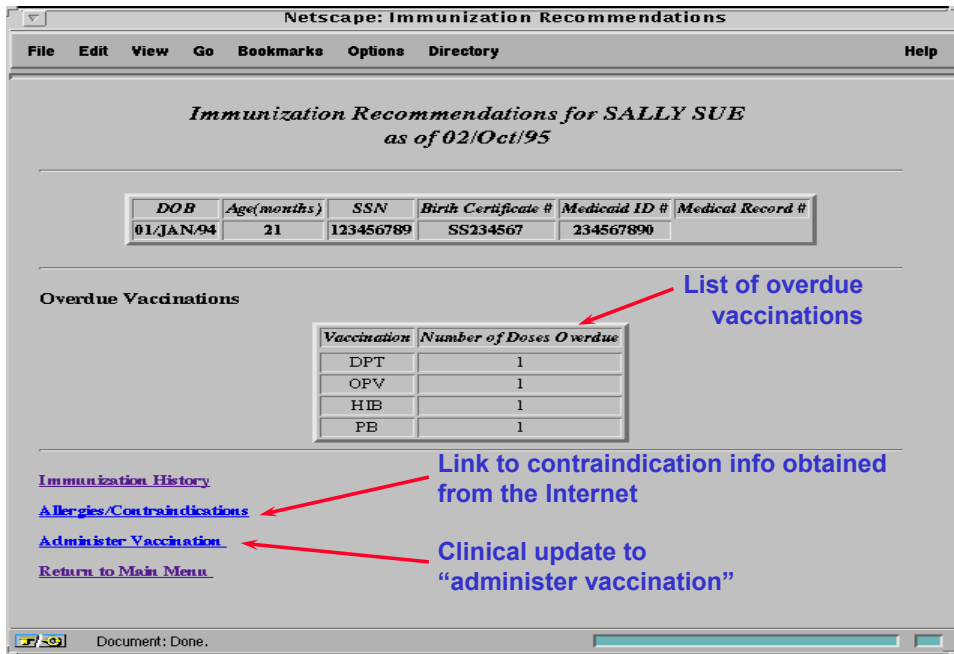


Figure 4-5. Provider’s Interface for Immunization Recommendation

The development of the application was completed with constraints due to end-user and system requirements. Some of the important requirements for this application, as determined by CHREF, include:

- Support for a distributed client/server-based architecture in a heterogeneous computing environment. This distribution should be transparent to users;
- Support for inter- and intra- enterprise wide coordination of tasks;
- Provision of a standard user-friendly interface for all users of the system (see Figure 4-5);
- Support for a variety of tasks: transactional, non-transactional, human, and application;
- Capability to use existing DBMS infrastructure across organizations;
- Low cost of the system acquisition and deployment for the providers and user organizations;
- Ease of use, modification (re-design), scalability, extensibility, and fast design-to-implementation;
- Use of standards, including EDI, for interactions among autonomous organizations, where possible;

- Security authorization for users and secure communication (required, as patient data is typically confidential).

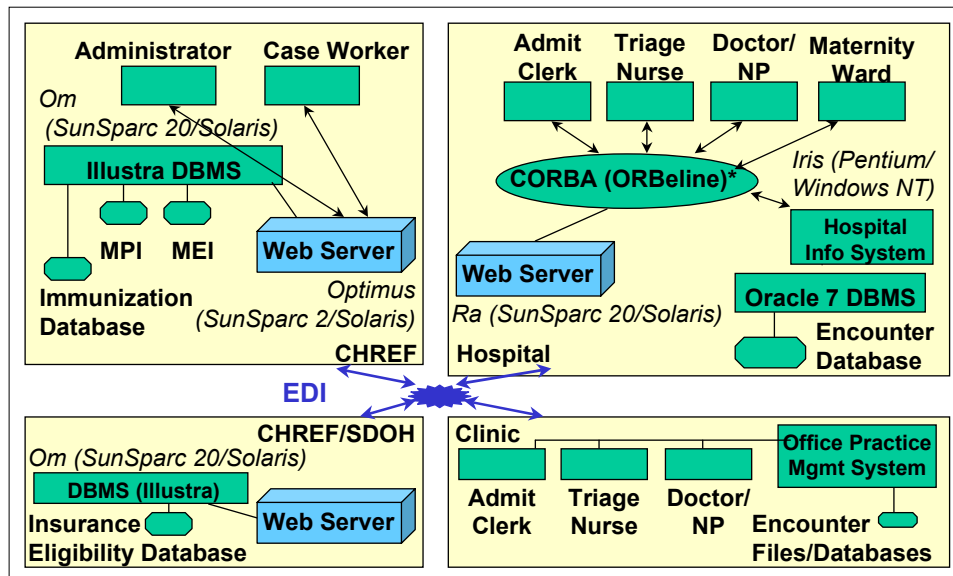


Figure 4-6. Implementation test-bed for the Immunization Tracking application

Based on these requirements, we have created a system test-bed on which the application was implemented (Figure 4-6). The system included a heterogeneous and distributed infrastructure, with multiple Web servers, CORBA servers, and multiple databases. The workflow involves 13 tasks, including tasks for the admit clerk, triage nurse, and for the eligibility check. Autonomous software systems and heterogeneous computer platforms were running in Georgia and in Connecticut; these included Solaris 2.4, Windows/NT, as well as Windows95, and five databases on two DBMS (Illustra and Oracle).

The workflow application system supports the maintenance of several central databases:

- The Master Patient Index (MPI) records the personal information and medical history of patients;
- The Master Encounter Index (MEI) records brief information pertaining to each encounter of each patient, at any hospital or clinic in the state;
- An Immunization Database (IMM) records information regarding each immunization performed for a person in the MPI;
- Eligibility databases (ELG) provide an insurance company's patient eligibility data; and
- The Detailed Encounter databases (ENC) provide detailed encounter information, as well as relevant data for each patient served by a particular provider.

The application design and implementation calls for CHREF to manage the MPI, MEI, and the IMM centralized databases for the state of Connecticut. The ELG database, containing data provided by insurance companies, is used in this application for EDI-

based eligibility verification, using the ANSI X12 standard. In particular, Web-based access is used to submit eligibility inquiries using ANSI 270, and responses are received using ANSI 271 transactions. Each participating provider organization (each hospital or clinic) has its own ENC database. When a structured database is used at CHREF, we promote the use of ODBC compliant DBMS. The DBMSs (Illustra and Oracle) used in this application are ODBC compliant.

## **5 Benefits of the METEOR Approach**

METEOR offers unique solutions to realize the promise of recent advances in distributed computing infrastructure, middleware, and Web technologies. By quickly integrating applications and information systems to support complex and dynamic business process management, METEOR provides technical advantages that distinguish it from other systems. These distinctions are outlined in Table 5-1 for the applications discussed in Section 4.

Example Application	Capabilities	Benefits
All	Graphical building of complex applications	Ability to visualize all application components; reduced needs for expert developers; rapid deployment.
All except Clinical Pathways	Support for heterogeneous, distributed computing environment; open-systems architecture and use of standards	Seamless deployment over networked heterogeneous (Solaris and NT) server platforms; ease of integration of legacy/existing applications; appeal to customers preferring non-proprietary and multi-vendor solutions.
All	Automatic code generation	Significantly reduced coding and corresponding savings in development cost; reduced need for expert developers; rapid deployment.
All	Integration of human and automated activities	Natural modeling of complex business activities/processes.
All except Clinical Pathways & GeneFlow	Fully distributed scheduling	High scalability and performance, minimal single point of failure
GeneFlow	QoS management	Specification, analysis, and monitoring of quality of service metrics.
None	Dynamic changes	Rapidly adapt to changes in business processes.
All	Traditional security	Support for roles and security on open internetworking.
Eligibility Referral & IZT	Database middleware support	Simplified access to heterogeneous variety relational databases on servers and mainframes.
Eligibility Referral & IZT	Workflow interoperability standards	Integration with other vendor's products, interoperability in multi-vendor and inter-enterprise applications such as e-commerce.
None	Transaction support, exception-handling and automatic recovery, survivability	7x24 operation and support for mission-critical processes.
All	Different levels of security (roles, authorization, network)	Flexible support for a broad range of security policies.
None	Component repository	XML-based reusable application components for rapid development of new applications.

**Table 5-1. Benefits of the METEOR approach**

## 6 Conclusion

Based on the deployment of real-world workflows, we have drawn a set of requirements for workflow systems supporting healthcare applications. Today's healthcare processes require capabilities for mission-critical workflow support and enterprise integration. Indispensable features include the seamless deployment over networked and heterogeneous server platforms; rapid deployment of applications; ease of integration of legacy/existing applications; high scalability and performance; specification, analysis, and monitoring of quality of service metrics; and adaptation to changes.

While comprehensive solutions encompassing these features are lacking in many of the current generation of workflow systems, the METEOR system distinguishes itself by focusing on supporting these high-end requirements.

The METEOR approach enables a rapid design-to-development via automatic code generation. The workflow model and enactment system support a variety of indispensable activities – user and application (automated) tasks – to be used in real-world organizational processes. The workflow engines support heterogeneous and distributed computing environments. This allows workflow process and data to be distributed within and across enterprises. Reliability is an inherent part of the WfMS infrastructure; it includes support for error handling and recovery by exploiting transaction management features. A well-defined hierarchical error-model is used for capturing and defining logical errors, and a recovery framework provides support for the detection and recovery of workflow system components in the event of failure. The system also supports a dynamic change interface, QoS management, and a case-based reasoning subsystem to effectively handle exceptions.

## 7 Acknowledgements

The METEOR team consists of Kemafor Anyanwu, Jorge Cardoso, Prof. Amit Sheth (PI), Prof. Krys Kochut (co-PI), and Prof. John Miller (co-PI). Key past contributors include: Ketan Bhukhanwala, Zhongqiao Li, Zonghwei Luo, Kshitij Shah, Souvik Das, David Lin, Arun Murugan, Devanand Palaniswami, Richard Wang, Devashish Worah, and Ke Zheng.

This research was partially done under a cooperative agreement between the National Institute of Standards and Technology Advanced Technology Program (under the HIIT contract, number 70NANB5H1011) and the Healthcare Open System and Trials, Inc. Consortium. Additional partial support and donations were provided by Iona, Informix, Hewlett-Packard Labs, and Boeing.

## 8 Bibliography

- Arpinar, I. B., J. A. Miller and A. P. Sheth (2001). An Efficient Data Extraction and Storage Utility for XML Documents. Proceedings of 39th Annual ACM Southeast Conference, Athens, GA, pp. 293-295.
- Berry, P. M. and K. L. Myers (1998). Adaptive Process Management: An AI Perspective. ACM Conference on Computer Supported Cooperative Work, Seattle, Washington.
- Bonner, A., A. Shruf and S. Rozen (1996). Database Requirements for Workflow Management in a High-Throughput Genome Laboratory. Proceedings of the NSF Workshop on

- Workflow and Process Automation in Information Systems: State-of-the-Art and Future Directions, Athens, GA. pp. 119-125.
- Cardoso, J. (2002). Quality of Service and Semantic Composition of Workflows. Ph.D. Dissertation. Department of Computer Science, University of Georgia, Athens, GA.
- Cardoso, J., A. Sheth and J. Miller (2002). Workflow Quality of Service. International Conference on Enterprise Integration and Modeling Technology and International Enterprise Modeling Conference (ICEIMT/IEMC'02), Valencia, Spain, Kluwer Publishers.
- CareFlowNet (2002). CareFlowNet Home Page, <http://www.careflow.com>.
- Chaiken, B. (1997). Workflow in Healthcare, <http://www.araxsys.com/araxsys/resources/workflow.pdf>.
- COSA (2002). COSA Workflow. <http://www.ley.de/cosa/index.htm>.
- Dayal, U., M. Hsu and R. Ladin (1991). A Transactional Model for Long-running Activities. Proceedings of the 17th International Conference on Very Large Databases. pp. 113-122.
- DeJesus, X. (1998). "Integrating PACS Power." Healthcare Informatics **15**(20): 97.
- Georgakopoulos, D., M. Hornick and A. Sheth (1995). "An Overview of Workflow Management: From Process Modeling to Infrastructure for Automation." Distributed and Parallel Databases, An International Journal **3**(2): 119-153.
- IBM (2002). MQSeries Workflow. <http://www-3.ibm.com/software/ts/mqseries/workflow/>.
- Jablonski, S. (1994). MOBILE: A Modular Workflow Model and Architecture. Proceedings of the 4th International Working Conference on Dynamic Modelling and Information Systems, Noordwijkerhout, Netherlands.
- JFLOW (1998). "OMG BODTF RFP #2 Submission, Workflow Management Facility, Revised Submission, <ftp://ftp.omg.org/pub/docs/bom/98-06-07.pdf>," 4 Jul 1998.
- Kang, M. H., J. N. Froscher, A. P. Sheth, K. J. Kochut and J. A. Miller (1999). A Multilevel Secure Workflow Management System. Proceedings of the 11th Conference on Advanced Information Systems Engineering, Heidelberg, Germany, Springer-Verlag. pp. 271-285.
- Kochut, K. J., A. P. Sheth and J. A. Miller (1999). "ORBWork: A CORBA-Based Fully Distributed, Scalable and Dynamic Workflow Enactment Service for METEOR," Large Scale Distributed Information Systems Lab, Department of Computer Science, University of Georgia, Athens, GA.
- Lin, C. (1997). A Portable Graphic Workflow Designer. M.Sc. Thesis. Department of Computer Science, University of Georgia, Athens, GA.
- McCready, S. (1992). There is more than one kind of workflow software. Computerworld. **November 2**: 86-90.
- Miller, J. A., D. Palaniswami, A. P. Sheth, K. J. Kochut and H. Singh (1998). "WebWork: METEOR2's Web-based Workflow Management System." Journal of Intelligence Information Management Systems: Integrating Artificial Intelligence and Database Technologies (JIIS) **10**(2): 185-215.
- Mohan, C., G. Alonso, R. Guenthoer and M. Kamath (1995). "Exotica: A Research Perspective of Workflow Management Systems." Data Engineering Bulletin **18**(1): 19-26.
- OMG (1998). BODTF RFP #2 Submission, Workflow Management Facility, Revised Submission, <ftp://ftp.omg.org/pub/docs/bom/98-06-07.pdf>.
- Reichert, M. and P. Dadam (1998). "ADEPTflex - Supporting Dynamic Changes of Workflows Without Losing Control." Journal of Intelligent Information Systems - Special Issue on Workflow Management **10**(2): 93-129.
- Sheth, A., D. Georgakopoulos, S. Joosten, M. Rusinkiewicz, W. Scacchi, J. Wileden and A. Wolf (1996). "Report from the NSF Workshop on Workflow and Process Automation in

- Information Systems," Department of Computer Science, University of Georgia, Athens, GA, Technical Report UGA-CS-TR-96-003, October 1996.
- Shrivastava, S. K. and S. M. Wheater (1998). Architectural Support for Dynamic Reconfiguration of Distributed Workflow Applications. IEEE Proceedings Software Engineering. pp. 155-162.
- Soarian (2002). Soarian Home Page, <http://www.smed.com/solutions/products/soarian/index.php>.
- SoftMed (2002). SoftMed Home Page, <http://www.softmed.com/>.
- Staffware (2002). STAFFWARE. <http://www.staffware.com/>.
- Swenson, K. (1998). SWAP - Simple Workflow Access Protocol. Workshop on Internet Scale Event Notification, Irvine, CA.
- TeleTracking (2002). TeleTracking Home Page, <http://www.teletracking.com/>.
- TIBCO (2002). TIBCO InConcert. [http://www.tibco.com/products/in\\_concert/](http://www.tibco.com/products/in_concert/).
- VMImedical (2002). VMImedical Home Page, <http://www.vmimedical.com/>.
- WARIA (2002). Workflow and Reengineering International Association, <http://www.waria.com/>.
- WfMC (2002). Workflow Management Coalition, <http://www.wfmc.org/>.
- Wodtke, D., J. Weissenfels, G. Weikum and A. K. Dittrich (1996). The MENTOR Project: Steps Towards Enterprise-Wide Workflow Management. Proceedings of the International Conference on Data Engineering, New Orleans.
- Worah, D., A. Sheth, K. Kochut and J. Miller (1997). "An Error Handling Framework for the ORBWork Workflow Enactment Service of METEOR," LSDIS Lab. Department of Computer Science, University of Georgia, Athens, GA, Technical Report, June 1997.
- Yong, J. (1998). The Respository system of METEOR workflow management system. M.Sc. Thesis. Department of Computer Science, University of Georgia, Athens, GA.
- Zheng, K. (1997). Designing Workflow Processes in METEOR2 Workflow Management System. M.Sc. Thesis. LSDIS Lab, Computer Science Department, University of Georgia, Athens, GA.