

# **AN INFORMATION ENABLED WORKFLOW ENGINE**

*W. A. Chinthanie Weerakoon and Chandima H. de Silva*  
*hkcweera@yahoo.com, chandima@kln.ac.lk*  
*Department of Statistics and Computer Science,*  
*University of Kelaniya,*  
*Sri Lanka.*

## **Acknowledgement**

The individuals, those who made their selfless contribution and assistance on completing this effectively, are acknowledged. Their patience, munificence and excellent supports were admirable.

## ABSTRACT

Today most of the workflows of document/information processes or transactions are managed by the systems build on IT infrastructure of an establishment. Workflow Management can be done best, by fully or partially automated tools that combine human and machine based activities.

This paper presents the entire life cycle of the development process of such a partially automated tool, the Document/Information enabled Workflow Engine (WFE) and its integration in to the Content Management Solution (CMS) of a reputed software vender. A workflow engine, the generator and the coordinator of document/information enabled workflows, have been designed, implemented and integrated into the company's Content Management Solution, as **an effort to improve application reliability and availability throughout the enterprise**. This approach standardizes the user's software packaging process provides automated document/information enabled workflows, which can be used for rapid deployment and flexible execution of e-documentation processes thereby tying the people, systems, organizational knowledge and goals to congregate into comprehensive, efficient processes, reducing the administrative overhead. In the development process unified modeling approach has been used, which not only provides a methodology for the software development process, but also combines best practices and processes together with the Unified Modeling Language's notations and diagrams for better understanding Object-Oriented Concepts and system development, **in the context of effective management of information**.

# INTRODUCTION

## Background

Development of the personal computer brought about tremendous changes for diverse areas as business, industry, science, education, etc. Due to the wide spread recognition of the power and the utility of computer in virtually in all fields, systems build on IT infrastructure are now available. In the globalized, competitive world the computer become a powerful tool for the most of the productions. Therefore in all parts of the world, nowadays, most of the workflows of information/document processes, transactions etc. are manage by the systems build on IT infrastructure. (In the present, information takes different forms such as text, number, image, audio, and video, so on.) Then, the *Workflow Management* has to act a major role in this multifaceted environment.

*Workflow Management* is a fast evolving technology which is increasingly being exploited by businesses in a variety of industries. Its primary characteristic is the full or partial automation of processes involving combinations of human and machine-based activities, particularly those involving interaction with IT applications and tools. Although, it's most prevalent use within the office environment is staff intensive operations such as insurance, banking, legal and general administration, etc, it is also applicable to some classes of industrial and manufacturing applications.

A *workflow engine* is designed for rapid deployment and flexible execution of the business processes. The workflow engine acts as a workflow coordinator for those business processes, tying people, systems, organizational knowledge and goals together into comprehensive, auditable processes.

Some software vendors have expensive workflow management products available for their own today, which involve workflow management technology and there is a continual introduction of more products into the market. The availability of a wide range of products within the market has allowed individual product vendors to focus on particular functional capabilities and users have adopted particular products to meet specific application needs.

The project described in this paper was carried out at Snaje Lanka Private Limited, a software vendor that had to work in a competitive environment. While working in such an environment, they had to overcome each and every adverse situation. Always, they expect to complete their jobs within projected time frames to the fullest satisfaction of their clients while delivering the solutions. They have done that, but there are several weaknesses;

For the present, in the process of developing document enabled workflows in reaction to their changing business needs; they must do the customization by redesigning and reconstructing with long coding, i.e. the availability of the applications throughout the enterprise is not satisfiable, Further, they have to waste significant time to deploy on developing such packages and Lesser operational flexibility.

## **Goal and Scope**

This provides a common "Reference Model" for workflow engines identifying their characteristics, terminology and components, enabling the individual specifications to be developed within the context of an overall model for workflow systems.

- This workflow Engine is designed to act as a generator and coordinator of document enabled workflow applications, which can be used to rapid deployment and flexible execution of e-documentation processes; tying the people, systems, organizational knowledge and goals together into comprehensive, efficient processes.
- Provide automated document/information enabled workflow to reduce administrative overhead.
- Leverage inherent features of chosen technologies to reduce the amount of custom code required to develop solution.
- Give workflow process owners the ability to modify or create new workflows in reaction to their changing business needs.
- Standardize the user's software packaging process.
- Improve application reliability and availability throughout the enterprise.
- Maximize IT productivity and significantly reduce the time and the cost needed to deploy on developing packages.

## **MATERIALS AND METHODS**

At the initial stage the author had to build a common structure for the WF in an effective manner. To understand the functionalities of a WF, the author has been undergone studying of most possible document/ information WFs, which are used in many processes(ex: News paper publishing process, Issuing reports of patients, Issuing birth/death/marriage certificates, Opening bank accounts), in many organizations(ex: Banks, Register General's office, Ceylon Electricity Board,...).

Further investigation has been carried out to learn the functionalities of Workflow Engines, also known as Workflow Management Systems (ex: Werkflow, LegaSuite Workflow Engine, SWFL (Service Workflow Language) Workflow Engine, OpenWFE...) and their integration in to Applications such as Content Management Solution.

## **Fact Finding**

The author has used internet/web search as a fact finding method, also referred related books mentioned in the bibliography. Other than that, with the initial findings and background details the author has identified the importance of accurate facts which will be the basement for this project document. Fact finding process (steps) has been planned and defined to collect proper,

timely and accurate information without missing any key information or data which describes as below.

### ***Step 1: define clear goals and objectives***

In the process of identifying issues and preparing better solution, the following were understood and defined by the author herself, at the initial state of planning for fact finding.

1. *Project Scope* – which guide the author to identify which areas should cover in the process of fact finding
2. *Data Definition* – what are the specific data that will be needed to analyze and take decisions
3. *Rationale for Collecting the Data* – the author has identified, the fact finding should process through the different grounds and personals such as,
  - Senior Managers who define the vision and mission to the organization.
  - Department Managers and officers, those who involved with day to day workflow processes.
  - End Users of the system who are working with the current systems
  - Chief Technical Officer – who define strategic and operational technical plans within organization
  - Technical Staff including System Administrator, Solution Developers and Maintenance Assistants who are directly involved with the day to day system operations in the technical point of view.
4. *Data Maintaining Plan* – The author has planned to maintain all the feedback data in following forms depending on the data collection mechanism such as,
  - Review Records - All the records were captured through system design specifications, user manuals and reports and summarized in to the Microsoft Excel document for the easiness of retrieval in the analysis phase.
  - Interviews - At the time of the interview all the feedbacks and answers to questions were documented in a paper. The author has used variety of pre defined questions.
  - Observation - All the records tracked in the observation process were first captured in to Microsoft word document in symbol form and then summarized in to the Microsoft Excel document.

### ***Step 2: operational definition and methodology***

Once the author has defined the initial goal, objectives and scope it was required to clearly define what data is to be collected and how. In the operational data collection definition following decisions and relevant activities has carried out.

The fact finding process has decide to carryout using following methods in the same order,

- Gather data through the user manuals, previous reports and requirement specifications etc via the internet....

- Observing, the day to day organizational workflows in different type of organizations. (Ex:- Bank)
- Interviews – Having direct discussions with the target personals. Prior to the interviews the author has analyzed which questions to be asked from each individuals and groups to maximum utilization of time. The interviews were the key data collection mechanism in this project since it was very easy and able to collect data directly from the relevant personals.

***Step 3: ensuring repeatability, reproducibility, accuracy and stability***

The author has identified, even though the interviewees were in the different areas in different organizations, generally, they comes out with the same data/information when talking about document enabled workflows.

Initial data gathering sessions and target audiences select carefully assuring the accuracy.

***Step 4: fact finding process***

Once the data collection process has been planned and defined, it has identify that the author had to follow through the process from start to finish ensuring the plan is executed consistently and accurately providing expected outcomes which are critical to analyze and finally identify the solution. The author has continuously revised her project scope and data collection guidelines, goals and objectives.

As planned the author carried out the fact/data gathering in accessing and analyzing relevant records, observing the workflow in variety of organizations, having interviews with the target personnel.

***Step 5: post data collection and verification***

Once it completes the initial fact finding process the author has tracked and documented all the outcomes by avoiding data duplication. The company has arranged brainstorming session with the involvement of all the relevant personals, and then the author was able to collect their further feedbacks and verifications.

**Analysis and Design**

*Workflow* at its simplest is the movement of documents and/or tasks through a work process. More specifically, workflow is the operational aspect of a work procedure:

- how tasks are structured
- who performs them
- what their relative order is
- how they are synchronized
- how information flows to support the tasks (workflow)
- how tasks are being tracked

As the dimension of time is considered in workflow, workflow considers “*throughput*” (The amount of work that a computer can do in a given time period) as a distinct measure. Workflow problems can be modelled and analyzed using graph-based formalisms like *Petri nets* (Considered as the mother of all process definition languages. Petri Nets is a formal and graphical language which is appropriate for modeling systems with concurrency and resource sharing. It is a generalization of automata theory such that the concept of concurrently occurring events can be expressed).

While the concept of workflow is not specific to information technology, support for workflow is an integral part of document management and imaging software. Since 1993, one trade consortium specifically focused on workflow management and the interoperability of workflow management systems has been the Workflow Management Coalition.

Distinction can be made between "scientific" and "business" workflow paradigms. While the former is mostly concerned with throughput of data through various algorithms, applications and services, the latter concentrates on scheduling task executions, including dependencies which are not necessarily data-driven and may include human agents.

*Scientific workflows* found wide acceptance in the fields of bio-informatics and chem.-informatics in the early 2000s, where they successfully met the need for multiple interconnected tools, handling of multiple data formats and large data quantities. Also, the paradigm of scientific workflows was close to the well-established tradition of Perl scripting in life-science research organizations, so this adoption represented a natural step forward towards a more structured infrastructure setup.

*Business workflows* are more generic, being able to represent any structuring of tasks, and are equally applicable to task scheduling within a software application server and organizing a paper or electronic document trail within an organization. Their origins date back to the 1970s, when they were purely paper-based, and the principles from that period made the transition to modern IT infrastructure systems.

The key driver to gain benefit from the understanding of the workflow process in a business context is that the throughput of the work stream path is modelled in such a way as to evaluate the efficiency of the flow route through internal silos with a view to increasing discrete control of uniquely identified business attributes and rules and reducing potential low efficiency drivers. Evaluation of resources, both physical and human is essential to evaluate hand-off points and potential to create smoother transitions between tasks.

As a way of bridging the gap between the two, significant effort is being put into defining *workflow patterns* (table 1) that can be used to compare and contrast different workflow engines across both of these domains.

In general, workflow techniques are appropriate only for work in which human involvement is limited to key data entry and decision points. For innovative, adaptive, collaborative human work the techniques of Human Interaction Management are required.

Workflow Continuum (Workflow Patterns)					
Collaborative Processes		Semi-Structured Processors		Structured Processors	
Ad-hoc Workflow	Team Room Quick Place	Integrated Team Activity	Chained Activity	Ad-hoc Exceptions	Standard Workflow

**Table 1: Workflow Continuum**

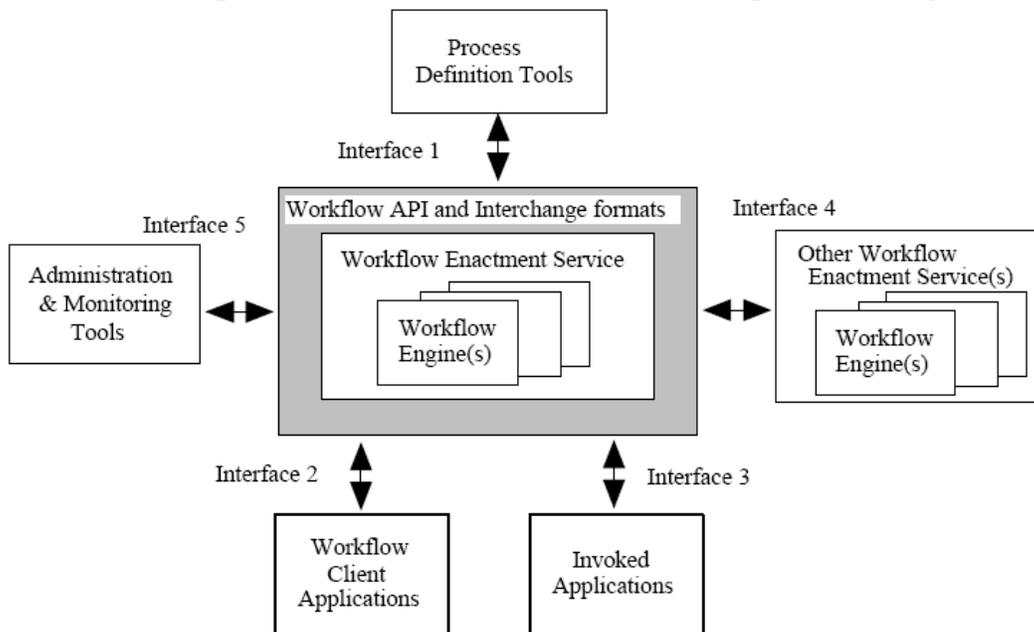
### Workflow Engine

*Definitions:* A Workflow Engine (WFE) is a software component that takes as input a formal description of business processes and maintains the state of processes executions, thereby delegating activities amongst people and applications.

To get started, let's define the basic terminology: a process definition and a process instance.

A *process definition* is a formal description of a business process or procedure. A *process instance* is one execution of a process definition.

When going one level deeper, it is needed to be very careful of the language use. The way to express the steps within a process still hasn't found a common ground yet. That is the major difference between the specs and tools for workflow and business process management.



**Figure 1: Workflow Reference Model - Components and Interfaces**

Another important responsibility of a Workflow Engine is maintaining context information for each process execution.

A *process context variable* is a variable associated with a process instance. E.g. the start-date of a request, the key of a record in the database, a reference to a document in the document management system...

Usually the process definition declares the process context variables. Then, at process instance creation time, the variables are instantiated. All mature Workflow Engine have customizable variable types.

### ***Target Usage***

1. Use a Workflow Engine as *Enterprise Application Integration (EAI)* (a discipline of implementing software requirements that involve multiple dedicated applications) platform. Currently, in most enterprise environments, the IT infrastructure has heterogeneous applications and databases running in an intranet. Typically, these applications have a clear purpose when they are introduced in the organization. Examples are customer management, document management, supply chain, order entry, billing, and resource planning... Let's call them dedicated applications. Each of these dedicated applications contains knowledge about the business processes they have to support. The enterprise fits these automated processes of the software components into their overall non-automated business processes. Once such dedicated applications are installed and running, opportunities arise for new feature requirements that span over multiple applications. Dedicated applications incorporate a number of business processes hard coded in the software. You could say that you buy a set of fixed, automated business processes. A Workflow Engine on the other hand has got no prior knowledge of the domain. A Workflow Engine takes a description of a business process as input and manages the executions of the process instances. That makes it much more flexible than dedicated applications but you have to do the effort to create a formal description of the business process. That is why Workflow Engine and dedicated applications are complementary. A Workflow Engine can be used to manage the overall process. The first way to use a Workflow Engine that was discussed here is to tie together all dedicated applications and create an EAI-platform with it.
2. A Workflow Engine delivers high added value is for the development of workflow software that has a lot of people-related tasks. For this purpose, most Workflow Engines have a very convenient mechanism of creating forms for tasks. This target usage is especially productive in organizations focusing on ISO or CMM certification. Instead of documenting the procedures in a text format, a Workflow Engine allows you to create fully or partially automated support (e.g. by means of a web-application) of the procedures that are modeled as process definitions.
3. Embed a workflow engine inside another application. Remember from the first option that dedicated application incorporate a set of fixed business processes for a specific domain. The

companies that develop the dedicated applications can embed workflow engine software inside of their software. The workflow engine is in this case only used as an application component, hiding it from the application users. The main reason for embedding a workflow engine inside an application is for reuse and maintainability of the application software.

### ***Workflow Level Functionalities***

1. ***Approval:*** This level routes a document or item to a group of people for approval. By default, the approval workflow is associated with the document content type, and thus it is automatically available in document libraries. A version of the approval workflow is also associated by the default with the pages library in a publishing site and it can be used to manage the approval process for the publication of web pages.
2. ***Feedback:*** This level routes a document or item to a group of people for feedback. Reviewers can provide feedback, which is then compiled and sent the person who initiated the workflow, or who works in the next level. By default the feedback workflow is associated with the document content type, and thus it is automatically available in document libraries.
3. ***Decision:*** This level routes a document or item to a group of people for decisions. Decision makers make decisions such as load handling (ex: - A has 10 documents to approve. B has 5 documents to approve. Both have same responsibilities and each of them has an upper limit for the number of documents, i.e. 10. Then, the decision maker will decide to assign the next document to B for approval.)
4. ***Integration:*** (For further Developments) this engaged with the integration of the documents with the other separated systems.

Introducing workflow in an organization delivers benefits on the software development level as well as the business level.

### ***Easy Development***

A Workflow Engine will ease development and even more, the maintenance of standard enterprise software.

*Reduced development risk* - the business analyst will talk the same language as the developer (in terms of states and actions). That implies that the developer will not have to make a translation from user requirements to a software design.

*Centralized implementation* - it's the business processes that change so the biggest advantage of using a workflow system is that the implementation is not a fuzzy combination of software pieces scattered over various systems.

*Rapid application development* - your software is freed from the task of keeping track of the participants in a process, leading to faster development and code that is better maintainable.

## ***Interfaces***

A Workflow Engine takes process definitions as input. For the moment, think of a process definition as a UML activity diagram. Then, the Workflow Engine maintains the state of the executions of these process definitions. For this, it needs to be informed of state-changes. The state changes of the process executions can be logged for monitoring purposes.

1. *Definition* -The definition interface of a Workflow Engine allows process developers to deploy process definitions. Note that the actor 'process developer' is a placeholder for the combination of a business analyst and a software developer.
2. *Execution* -The execution interface allows users and systems to act upon process instances. Process instances are executions of process definitions. The control flow of a process definition is a description of a state machine. The two main methods in the execution interface are starting a process instance and signaling the end of a state to the Workflow Engine.
3. *Application* -The application interface denotes the WFE-initiated-interaction between a Workflow Engine and external systems. When a user or a system manages the execution of a process instance, events are generated (e.g. traversal of a transition). A process definition can specify that a piece of programming logic that has to be executed upon an event. The program logic can communicate with other systems in- or outside of the organization.
4. *Monitoring* -With monitoring interface, managers can gain insights through extracting statistics on the execution logs of the processes. Sometimes, the execution logs are also referred to as the audit trails.

## **The 4 Layers of a Process Definition of Workflow Engine**

In the next part, the author attempts to answer the question "What is the content of a process definition?" This synthesis is based on principles and concepts that are used in the models of various specs and tools. It reflects the common ground between most of these models. The content of a process definition can be sliced into 4 different layers: state, context, programming logic and user interface.

### ***The State Layer***

The state layer of a business process is all about expressing states and control flow. Control flow in standard programming languages is derived from Von Neumann architectures. There it defines the sequence of instructions that have to be executed. This control flow is determined by the order in which we write the instructions, if-statements, loop-statements, and so on. The control flow in a business process is basically the same. But instead of using an instruction as the basic element, the basic element in a business process is a state.

A state (also known as wait-state) in a process specifies a dependency upon an external actor. The meaning of a state is like "Now system X or person Y has to do something. Wait here until

an external trigger of that actor signals the completion of the task." A state defines a dependency on a result provided by an external party. A typical example of a state is an approval step.

A state in a process definition also specifies which actor the execution depends on. The names given to these actors correspond with the notion of swimlanes in an activity diagram. With this information the Workflow Engine can extract task lists, which is the most common feature amongst Workflow Engines. As said before actors can be people or systems. For the states that require a human actor, the person has to be calculated at runtime by the WFE. The dependency from a Workflow Engine on the organizational data originates from this calculation.

The *control flow* of a process definition is a set of states combined with the relations between the states. The logic between the states specifies which execution paths run concurrent and which are exclusive. Concurrent paths of execution are modeled with forks and joins, while exclusive paths of execution are modeled with decisions and merges. Note that in most models, every state has an implicit merge in front of it.

Often UML activity diagrams are used to model business processes. While it is an intuitive and widespread notation, a major problem poses in the diagram interpretation. In UML activity diagrams, no distinction is made between a state and an action (see 'Why the term 'activity' should be banned) they are both modeled as activities. The lack of this distinction (referred to as the lack of a state concept) is the most heard criticism by the academic community on UML activity diagrams. A second problem with UML activity diagrams was introduced in UML version 2.0. When multiple transitions arrive in an activity, previous versions specified a default merge interpretation. The 2.0 version changed this to a default join, which requires synchronization. In author's opinion, the graphical part of UML activity diagrams can still be used to model the state-layer of business processes on the condition that the semantics of these 2 constructs are changed as follows:

1. In the diagram presentation of a business process, only model the state layer (states and control flow) and leave the actions out of the diagram. This means that every rectangle in the diagram should be a state instead of an activity
2. If multiple transitions arrive in a state, define the default to be a default merge without synchronization

During process execution, a token is the pointer used by a Workflow Engine to keep track of the current state of a process. This corresponds to the notion of program counter in Von Neumann architectures. When a token arrives in a state, the token is assigned to the external party for which the Workflow Engine is waiting for. The external party can be a user, a group, or a computer system. Let's define an actor as a user or a system that acts upon the execution of a process. The assignment of an actor to a token is the only occasion where the Workflow Engine needs to access the organizational data. This assignment allows the Workflow Engine to extract task lists.

### ***The Context Layer***

A process context variable -or variable for short- is a variable associated with a process instance. Process variables allow a process developer to store data during the lifetime of the process instance. Some Workflow Engine's have a fixed set of types, while in other systems; it is possible to define your own custom types.

Note that variables can also be used to store references. A variable could reference e.g. a record in a database or a file on a network drive. When to use references typically depends on whether other applications use the referenced data.

Another interesting aspect related to process variables is how a Workflow Engine turns data into information. Workflow is all about orchestrating the tasks and data between various heterogeneous systems across an organization. For the human tasks of a business process, the Workflow Engine collects the relevant data-items from the other involved systems like e.g. databases, document management systems. In each human step of the business process, only the relevant data-items are shown that are collected and calculated from these heterogeneous systems. By presenting only relevant data-items, data that resides on the disparate systems is transformed and presented as information.

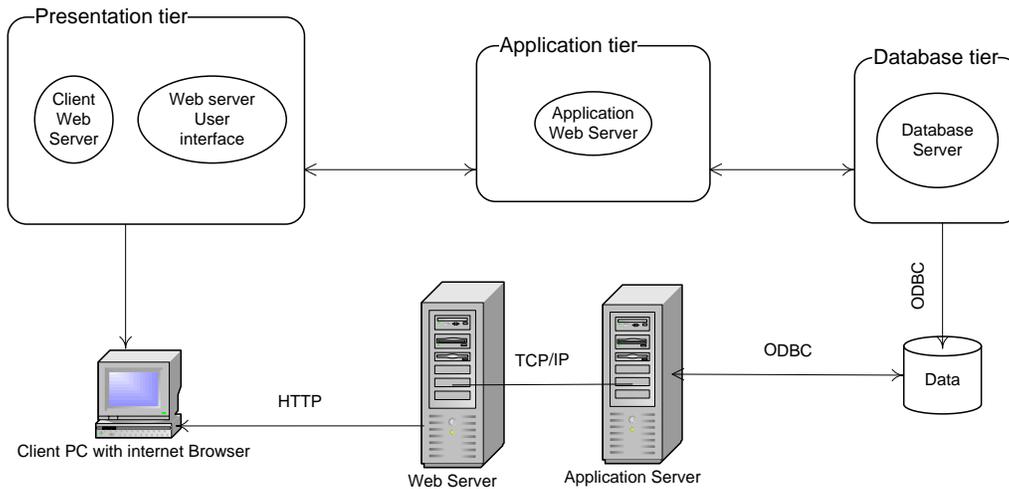
### ***The Programming Logic Layer***

Recall that an action is a piece of programming logic to be executed by the Workflow Engine upon a specified event that occurs during process execution. With programming logic we mean a piece of software in binary or in source format, in any programming or scripting language. The programming logic layer combines all these pieces of software with the information that specifies on which events these pieces of software need to be executed. Examples of programming logic for integration are sending an email, sending a message on a message broker, fetching data from an ERP-system and updating a database.

### ***The User Interface Layer***

When an actor triggers the end of a state, usually that is the event where data is fed into the process variables. E.g. in the holiday example, when a boss signals to the WFE that the state 'evaluation' is done, the boss provides the value 'approved' or 'disapproved' into the process. Some Workflow Engines allow specifying the each state what data can be fed into the process and how it should be stored into the process variables. From that information a user interface form can be generated to signal that requests information from a user. To see an example of a generic web application that allows actors to submit generated forms based upon a process definition.

## General Discussion about the System Architecture and Design



**Figure 2: System Architecture**

Client/Server architecture is used here and it refers to a process involves at least two independent entities, one a client and the other a server. The client makes a request of the server, and the server services the request. A request can take the form of an SQL query submitted to an SQL database engine. The database engine in turn processes the request and returns a result set. This exemplifies the client/server relationship. For faster, more cost-effective application performance these pieces can be separated and application logic can be distributed between them. A major benefit of a client /server application is reduced network traffic and, in most cases, an incredible faster execution time.

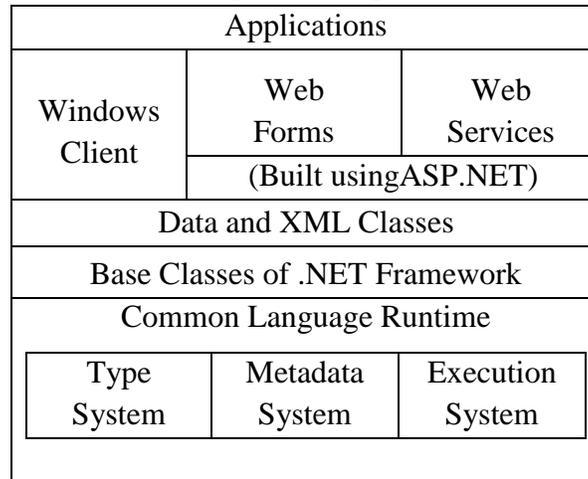
Although both the main frame and fileserver-based systems continue to provide service to business, they fail to provide a truly scalable framework for building competitive business solution. The major factor is that logic must be executed on the main frame in centralized architecture, or on the client in a file server-based architecture.

Object modelling methodologies such as UML (Unified Modelling Language) are becoming increasingly popular in software design and engineering. These methodologies go beyond database design to specify detailed design of software modules and their interactions using various types of diagrams. Static design of the system was done by using the class diagrams. In class diagrams, *operations* on objects are specified, in addition to specifying the database schema structure.

The dynamic aspects of the system involve the interaction of instances in all the views of the system architecture, including instances of classes, interfaces, components and nodes. Here the author models the flow of control by time ordering, emphasizing the passing of messages as they unfold over time, which is a particularly useful way to visualize dynamic behavior in the context of a use case scenario. The sequence diagram does a better job of visualizing these simple iteration and branching.

## System Construction

.NET Framework 2.0 was used in developing Workflow Engine. This provides runtime environment that manages the code at execution time, irrespective of the language and repository of built in classes and interfaces that correspond to a specific functionality.



**Figure 3: .NET Framework.**

ASP.NET 2.0 was used as the language and it supports with some brilliant new features that enable faster development of web applications with lesser code. A web application or more specifically, an ASP.NET application is simply a collection of files (ASP.NET source file, images, Configuration files and so on) that rely on a web server to execute any intended functionality. Without a web server enables us to serve ASP.NET pages over the internet and let anyone with web access internet with the dynamic content generated by executing the ASP.NET files on the server. Therein lays the largest difference between desktop-based applications and web applications. Thankfully, ASP.NET tries to minimize these differences, which enables us to migrate fairly easily to web applications if we are a traditional desktop developer.

ADO.NET is the main data access system and protocol that ASP.NET 2.0 uses. It uses a disconnected data architecture, where the data we work with is just a copy of the data in the actual database.

ASP.NET AJAX, an entirely new approach to Web development and is a free framework for building a new generation of richer, more interactive, highly personalized cross-browser web applications. In most applications now-a-days, whenever a small section of a web page needs to be changed, it is mandatory to reload the entire web page. This requires a complete round trip to sever that is, of course, quite inefficient and sometimes results in long waits for end users. To overcome this inefficiency, an entire new approach to web development, called AJAX, has been introduced.

Microsoft SQL Server is a RDBMS. An RDBMS includes databases, the database engine, and the applications that are necessary to manage the data and the components of the RDBMS. The RDBMS organizes data into related rows and columns within the database. Here, as the backend

software Microsoft SQL server 2005 was used. SQL Server 2005 includes rich support for Internet language formats, performance and availability features to ensure uptime, and advanced management and tuning functionality to automate routine tasks and lower the total cost of ownership. Means of providing more flexibility *Microsoft SQL Server 2000* was also used. To Design unlimited reports to use in workflow engine and workflows, Crystal Reports designer was used. Crystal Reports for Visual Studio .NET and for Visual Studio 2005 are fully integrated in the Visual Studio .NET and Visual Studio 2005 family of products.

### ***Coding standards***

To sustain the maintainability, coding should be unambiguous and preserving standards is very important.

- At the beginning of an each module, a module heading was included.
- At the beginning of an each procedure or function, Function or sub heading was included.
- Variables were defined in a standardized way.
- Controllers were given meaningful ids.
- Comments
- Means of of providing standardized reports for workflow details, log details, etc... ‘Crystal reports’ was used.

## **CONCLUSION**

### **Characteristics of the System**

System offers following main functions;

1. Create new Workflows.
2. Edit existing Workflows.
3. Remove existing unwanted Workflows.
4. Retrieve Workflow details.
5. Allocate Workflows.
6. Store & Retrieve Log details.
7. Work in workflows.
8. Send and receive e-mails related to the workflow matters (ex: to inform users about their tasks).
9. Access user accounts.

## **Workflow Engine: Features and Specifications**

### ***File Routing and Approval***

- Supports Ad-hoc, Administrative and Production workflow routes
- Automatically routes documents and document folders for editing, review and approval on custom, pre-designed workflow "routes"
- Email notification works with corporate email systems.
- Source file(s) can be edited in place or duplicated to a working copy
- Working copies are automatically copied to predefined or user-specified working directories
- Files can be routed forward or backward, through any number of route "steps"
- Multiple people can be assigned to each route "step", for individual or group approval
- Rejected documents can automatically be returned to the originating sender or to other specified individuals

### ***Flexible Workflow Design***

- Unlimited number of custom workflow designs
- Custom workflow designs for different users or departments
- Custom icons represent each workflow "route step"
- User security tokens can be assigned to each stage of the route designs

### ***Audit Trails and Reporting***

- Workflow transaction history tracks all file movements and database updates for rollback if needed
- Managers can reassign routed documents
- Workflow history database is ODBC-compliant for reporting with Crystal Reports.

### **Benefits**

- This project presents a fascinating and exciting challenge and offers opportunities for important decision making and creativity for the author herself.
- This is a spectacular opportunity to recognize a work space, to apply the knowledge the author's learnt so far.

Further, via this project, the author has acquired;

- 1) Ability to employ good judgments.
- 2) Improvement of the ability to deal with problems in an effective manner.

- 3) The Knowledge of ASP.NET 2.0, AJAX 1.0, SQL Server 2005, SQL Server 2000, ADO.NET 2.0, Workflows, Workflow Management Solutions/ workflow Engines.
  - 4) Fulfill the partial requirement of a project in BSc (Special) degree in Statistics and Computer Science, University of Kelaniya.
  - 5) Learnt the value of a small group of dedicated people (team) in a work place.
  - 6) Improved patience.
- For users,, main advantages are;
    - 1) **Reduced time** spans for development and implementation.
    - 2) **Better service for their clients / Improved customer service** - consistency in the processes leads to greater predictability in levels of response to customers
    - 3) **Better process control** - improved management of business processes achieved through standardizing working methods and the availability of audit trails
    - 4) **Increased flexibility** and **quick response** to changing business needs.
    - 5) **Improved efficiency** - automation of many business processes results in the elimination of many unnecessary steps
    - 6) **A significant support to stay at the forefront of the competition, Business process improvement** - a focus on business processes leads to their streamlining and simplification.

## FUTURE ENHANCEMENTS

Here are some suggestions for future enhancements;

- Workflow Engine can be modified to use, even when a workflow is needed to be imported.
- Normally, information/document is stored in information/document libraries, and then each of the information/document has unique index number. If this index number has changed then the information/document should be automatically add to a workflow. Then the Reindex option can be added as a starting condition.
- If the users want to export a information/document, then the information/document should be automatically added to a workflow. Then this Export option can be added as a starting condition.
- It would be more flexible to have the level 'Integration' to use in integrating workflows with other applications.
- Further, for the present these workflows are compatible with some of the other softwares such as Microsoft Word, Excel, Power point and Ms Paint. But, some other softwares for video, audio files are not compatible with the developed system. Therefore the system must enhance to overcome this issue.

- While handling information/document, some of the image processing techniques should be coupled with the system (at least) to a certain degree.

## BIBLIOGRAPHY

### Books

- [1]. Ali Baharami, '*Object Oriented Systems development*', (1999), Irwin McGraw-Hill
- [2]. Applewood, c. HTML complete 2<sup>nd</sup> edition India
- [3]. Dreamtech Software Team, '*ASP.NET 2.0 Black Book*', (2006), Dreamtech Press
- [4]. Elmasri, Navathe, Somayajulu, Guptha, '*Fundamentals of Database Systems*', (2006), Pearson Education
- [5]. Grady Booch, James Rumbaugh, Ivar Jacobson, '*The Unified Modeling Language User Guide*' (*The Ultimate tutorial to the UML from the original designers*), (2006), Pearson Education
- [6]. Deitel H.M., Deitel P.J., Nieto T.R., '*Visual Basic .NET How to Program, 2/E*'
- [7]. Kenneth E. Kendall and Julie E. Kendall, '*System Analysis and Design*', (1998), Prentice Hall
- [8]. Perry Edwards, '*System Analysis and Design*', (1993), McGraw-Hill
- [9]. Steven A. Smith, Rob Howard, '*ASP.NET Developer's Cookbook*', The ASP Alliance.

### World Wide Web Home Pages

- [10]. '*Add or change a workflow for a list, library, or content type*  
<http://office.microsoft.com/en-us/sharepointserver/>
- [11]. '*Content management system*' <http://www.wikiepedia.org/>
- [12]. 2 Tony Marston, '*An activity based Workflow Engine for PHP*', <http://www.radicore.org/>
- [13]. [Damon Armstrong](#), '*.NET Application Architecture: the Data Access Layer*' 11 July 2006 [11/12/2007]
- [14]. David Braun, Jeff Sivils, Alex Shapiro, Jerry Versteegh '*Unified Modeling Language (UML) Tutorial*' < [http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML\\_tutorial/](http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/) >

- [15]. Logan Vadivelu, Radha K. Varadharajan, *'Design and implement a composite workflow solution using IBM middleware, Part 1: Using business process management and a services-based approach to create a highly responsive IT architecture'* [Online] < <http://ibm.com/> > [10/31/2007]
- [16]. Logan Vadivelu, Radha K. Varadharajan, *'Design and implement a composite workflow solution using IBM middleware, Part 3: Using portal technology to build a flexible, adaptable UI'* [Online] < <http://ibm.com/> > [10/31/2007]
- [17]. Logan Vadivelu, Radha K. Varadharajan, *'Design and implement a composite workflow solution using IBM middleware, Part 2: Model, manage and monitor your business process'* [Online] < <http://ibm.com/> > [10/31/2007]
- [18]. Tom Baeyens *'State of Workflow'* May 2004 [10/31/2007]  
<<http://www.TheServerSide.com/>>
- [19]. < <http://Webopedia.com/> > [10/31/2007]
- [20]. *'Building a Generic Database Tool Using .NET and Three Tier Architecture'*  
<<http://dotnetjunkies.com/>> [11/12/2007]
- [21]. Bank of Ceylon, *'ISBS training Guide Branch Teller System'* (User Manual)
- [22]. Bean Software, ASP.Net Tutorial [Online] [11/12/2007]
- i. *Cascading Style Sheets*
  - ii. *Menus in ASP.Net*
- [23]. *Cascading style sheets* [online] <<http://www.w3.org/Style/CSS/>> [11/12/2007]
- [24]. Hemant Kumar Meena, Indradeep Saha, Koushik Kumar Mondal, T.V. Prabhaka *'An Approach to Workflow Modeling and Analysis'* [10/31/2007] [Pdf]
- [25]. MSDN Library