'Genetic Information System Development and Maintenance' Model For Effective Software Maintenance and Reuse

Sanjeev Manchanda* smanchanda@thapar.edu
School of Mathematics and Computer Applications,
Thapar University,Patiala-147004 (INDIA)
**Corresponding author*

Mayank Dave mdave67@yahoo.com
Department of Computer Engg.,
National Instt. Of Technology, Kurukshetra, INDIA

S. B. Singh sbsingh69@yahoo.com
Department of Mathematics,
Punjabi University, Patiala, INDIA

Abstract

The aim of present research work is to develop an information system development process and a model for the development of new generation information systems. New age information systems are those Information systems that are capable of fulfilling the demand of highly dynamic information requirements derived from the competitive environments of the business organizations and support controlling the complexity involved in their maintenance and software reuse. Present research work analyzes the theoretical, financial, technical and practical problems related to the information system development, maintenance and software reuse, to propose an appropriate system development process and a model for the development as well as maintenance of information systems with maximum software reuse. Proposed system development process and model provide inherent support to the business organizations, in getting total control over information system development and maintenance maximum software reuse.

Keywords: Software Maintainability, Software Reuse, Software Configuration Management, System Development Life Cycle and Software Engineering.

1. INTRODUCTION

Global expansion of business organizations has created demand for the development of very complex and globally operating information systems. Present and future generation of information systems need to be capable of fulfilling such demands of these business organizations. Information system development and maintenance processes consume a lot of organizational resources like time, money and effort of their employees as well as system developers. Rising costs of information system development and maintenance, Rapid innovations in information technology, dynamism of users' information requirements, increasing sizes of the databases, global expansion of business organizations, exponentially increasing complexity of information systems, emerging needs of information for the organizations etc. has raised many questions about previously used system development practices, reusability and maintainability of information systems. Even after using advanced technology, investing sufficient funds and placing tremendous effort in system development, organizations face the requirement of changes in existing information systems. Frequently required changes and requirement of extra resources for realizing these changes increase anxiety among management members of organization. The reason of such anxiety is that management personnel of organizations do not understand the requirement of changes and maintenance as an essential part of information systems. For example, when new machinery is purchased, wear and tear, replacement of its certain parts, up gradation of technology and scrapping of machinery after certain period, are inherent requirements of the machinery, so are the information systems, these systems need to be up graded, maintained suitably. Information systems are such parts of business organizations, which take birth with the birth of organization, grow with the growth of organization, live step by step with organization and die with the death of organization (Section 3 includes detailed Discussion about this relationship). So, there is a need to adopt the change(s) as the part of an organization and its information systems. First aim of this paper is to change this viewpoint of management personnel about the requirement of changes in their information systems. Secondly this paper stress upon controlling the system development and maintenance processes by the organization's own employees for better results. Thirdly current paper helps in selection of technology that is highly adaptable and keeps pace with future. Fourthly this paper proposes a system development process and a model that helps implementing the changes frequently and maximizes the software reuse. Lastly paper concentrate upon the issues like cost, reliability, maintainability, quality and reusability with required experimental results. Next section includes literature review of related work, followed by discussion about the relationship between organizations and information systems, proposed information system development process/model for system development and maintenance, efficiency/effectiveness of proposed methodology and model, implementation, result comparisons, conclusion and last but not the least references.

2. LITERATURE REVIEW

In this section, we examine the historical developments related to the System development models/methodologies, Process modeling, Change management, Reliability, Goal orientation in modeling and use of Internet etc. System development models/methodologies, Process modeling, Change management, Reliability etc. are clearly related to the domain of this paper, whereas goal orientation is essential for all the sections of any organization. Goal orientation is substantial for guiding organizational efforts in unique direction and information system should possibly be guided towards the organizational goals. Internet helps in developing globally operating systems. So Internet developments need to be revealed here.

The documented collection of policies, processes and procedures used by a development team or organization to practice software engineering, is known as software development methodology (SDM) or system development life cycle (SDLC). Development of system development methodologies originated from the waterfall model (Royce [47]), in which development was supposed to proceed linearly through the phases of the requirement analysis, design, implementation, testing (validation), integration and maintenance. Researchers criticized Waterfall model for its excessive separation of different phases. In an attempt to overcome the shortcomings of the waterfall model many new software development approaches such as spiral model (Boehm [8]), iterative enhancement (Basili et al. [5]), rapid prototyping (Gomaa [25]), evolutionary prototyping and incremental development (Floyd [22]) had been suggested. In spiral Development process, a desired capability is identified, but the end-state requirements are not known at program initiation. Those requirements are refined through demonstration and risk management, there is continuous user feedback and each increment provides the user the best possible capability. The requirements for future increments depend on feedback from users and technology maturation, whereas the basic idea behind the iterative enhancement was to develop a software system incrementally, allowing the developer to take advantage of what was being learnt during the development of earlier, incremental, deliverable versions of the system. Software development approaches incorporating prototyping have gained respectability as they have proved to be able to dynamically respond to changes in user requirements, reduce the amount of rework required and help control the risk of incomplete requirements. Currently reuse model has achieved tremendous success in information system development (Frakes [24]). The aim of Component-based software development (CBSD) (Aoyama [2]) is to develop new software by widely reusing pre-fabricated software components. Many other researches (Kaushaar et al. [34], Boehm et al. [9], Gordon [27], Alavi [1], Naumann et al. [42], Tate et al. [49], Palvia et al. [43] etc.) contributed in development and growth of these methodologies.

Many researchers have contributed to business process development. A large number of process models were developed (Armenise et al. [3], Bandinelli et al. [4], Bubenko [11], Decker et al. [18], Jarzabek et al. [33], Jacobson et al. [32], Rumbaugh et al. [48] and Marca et al. [40]).

Software configuration (Change) management (SCM) is the discipline (Tichy [50]) that enables us to keep evolving software products under control and thus contributes to satisfy quality and delay constraints. The purpose of SCM is to manage change throughout the software development process (Bersoff et al. [6]). Change is a very natural and intrinsic aspect of software development process. SCM has been the focus of software engineering research and a great amount of research has been carried out on SCM. In the previous research, eight areas of functionality of SCM systems were found: version control, configuration support, team support, change control, build support, process control, status reporting, and audit control (Burrows et al. [12] and Dart [15, 16]). These functional areas mainly cover the management issues of software development. To provide these eight areas of functionality, different SCM systems use different models, such as the checkout/check in model, the composition model, the long transaction model, and the change set model (Feller [21]). In recent years, the focus of SCM research is on software process support in SCM systems (Estublier et al. [20], Leblang [38]), distributed configuration management systems (Hunt et al. [29] and Milewski [41]) and unified version models (Conradi et al. [14]) etc.

Goals are essence of management. Management by objectives (Drucker [19]) is one of the most important motivation factors for the success of any organization. Attempts have been made to incorporate goals into process modeling (Kueng et al. [37]), that suggested an informal approach in which goals provide a basis for process definition. Other model (Khomyakov et al. [35]) based on mathematical systems theory was proposed. This set of concepts extended (Bider et al. [7]) and used for defining a process pattern, allowing the design of generic processes that can be specialized for specific situations. The goals addressed by this approach are operational goals only, termed “functional goals”. Goals and soft-goals are applied for requirements elicitation in combination with scenarios (Rolland [45, 46]) and others contributions to relate goals with process models and its impact on strategic success.

The potential of the Internet to reach a large and growing body of customers, coupled with low communication costs, makes it a very attractive business medium to many organizations. Although there is significant interest in the use of the Internet for business purposes, studies articulating issues that can guide business managers in its use are lacking. Use of Internet and related issues are the hottest research areas. Different studies conducted and contributed (Brandtweiner et al. [10], Cho [13], Gonsalves et al. [26], Hamill [28], Weill [51] and Lee [39] etc.) to the aspects related to the use of Internet for business, marketing, performance and

modeling of systems etc.
Therefore, a great amount of active research has been carried out for developing system development methodologies, business process development, change management, goal orientation and its organizational implementation in modeling and Internet usage. Still we are facing many problems in development and maintenance of information systems. This paper is an effort to identify those problems and finding practical solutions for them.

3. RELATIONSHIP BETWEEN ORGANIZATION AND INFORMATION SYSTEM

Information systems are cores of business organizations. Flow of Information in an organization is as if blood floats in veins of human body. Information system born with the birth of an organization, it grows as the organization grows and stays alive forever with the organization or up to the death of the organization. Growth of organization directly affects the growth of information systems; changes in organizations create need for changes in information systems as well, e.g. When a firm is originated, its operations are limited up to small geographical areas and a small number of people. Simultaneously, information system is also small and can be handled manually or with little automation by small number of people. Once a company starts expanding its operations from one city to many cities and from one country to many countries, information system is also expanded concurrently to handle the organizational information needs. Even after many developments, software development and maintenance are very tedious tasks. Many problems generated due to information, systems faced by the organizations can be listed as budget of information system, selection of technology, selection of system development companies, duration of system development project, change identification, vision for future changes, flexibility of information system to incorporate future changes, level of changes, maintenance terms of the systems, system improvement policies, control over system development and maintenance, Data security, Fulfillment of right kind of user needs, Connectivity of different users etc. Current paper is an effort to find the solution of many of the problems listed above. First of all, there is a need to change the view about requirement of change(s) in information systems. Business organizations view change requirements of information system as burden for their organization. Organizational members especially need to adopt change requirements of information systems as inherent necessity of their information system. Secondly vision of organizational members should be global and the initial selections of technology need to support global expansion of information systems. Nevertheless to remark that Internet based technologies are the future of information systems. All organizations today need to adopt online technologies that support information exchange worldwide. Thirdly the selection of system development model and processes should support frequent change requirements, software reuse and controlled maintainability of information systems. An information system survives with organization, changes occurring in organization need to change information systems. If information systems reflect the organization, then how information needs of an information system may be understood and implemented in a short span of system development project. There is a fundamental need to develop information systems continuously throughout the life of organization, so that required changes can be implemented effectively and system architecture must support such requirement. In following sections we propose a system development process and a model to develop as well as maintain the information systems continuously throughout the life of an organization.

4. GENETIC INFORMATION SYSTEM DEVELOPMENT AND MAINTENANCE PROCESS

System development and maintainability can be achieved efficiently, if it is under direct control of organization's management. Management of organization can control their own employees far better than the employees of other organizations. Team building of organization's employees and flexible system development process are the ingredients of proposed Genetic information system development and maintenance process as explained below.

4.1 Team Organization for System Development and Maintenance

First of all, there is a need to have a group of personnel (optimum number of people) within organization to analyze, develop, test, implement and maintain the information system for their own organization throughout the life of an organization. This group of people will continuously look forward for the information requirements of the management personnel continuously and build the system or incorporate the changes continuously, without affecting the earlier implemented system at large, but only affected parts will be required to be updated, implemented, documented and only concerned people will be informed about the changes. This group of people will not only look after the information requirements of the management personnel or the public concerned, but also look forward for the technological developments worldwide, so that their organization can be benefited through the involvement of latest technological developments. More people, expertise and organizations may be hired, contracted or outsourced to help the development, whenever it is necessary.

4.2 Development and Maintenance of the system

Secondly there is need to develop the system by iterating the following system development steps throughout the life of an organization continuously:

STEP I: Genetic Creation of Processes and Sub Processes-First of all there is a need to identify information requirements from scratch or from previously implemented major processes and sub processes of the organization and to program them to meet the requirements by formulating a library (L) of processes and sub processes i.e.

$$L = \{P_1, P_2, ..., P_n\} \qquad (1)$$

where $P_k = \{SP_{k1}, SP_{k2}, ..., SP_{kl}\}$, $\forall k = 1 \ to \ n$, $l \geq 1$, $n \ and \ l \ are \ \text{variables}$.

Number of processes and sub processes will be finite and almost different for all the processes, whereas maximum value of l and n for the number of sub processes and processes respectively will be dependent upon the decision of system analyst. Processes and sub processes will work as genes for the genetic development process. The goal of design (from a low-level perspective) is to minimize coupling and maximize cohesion (Kramer [36]), can only be achieved from the division of different information needs in independent categories of processes. A process is defined as a set of sub processes and will fulfill information requirements through execution of proper logic and will provide access to the authenticated database(s). Genetic development means the primary set or sequence of processes that lead to the formation and subsequent development of processes/sub processes. Development of processes and carried out at three different levels as follows:

(i) User Level: Information requirements are identified at this level, so system development process actually begins at this level. First of all, users' information requirements need to be identified. Once users' information requirements are identified, then further level can be sequentially explored. List of identified information requirements needs to be reviewed successively.

(ii) Database level: Logical designs of schema/subschema need to be developed and implemented according to the user requirements and the organizational considerations. Schema / Subschema need to be updated according to the initially defined/emerging/modified information requirements. Detailed design considerations need to be defined so that less frequent changes are required later. Technology used for System development needs to be adaptive for any kinds of changes from logical to physical memory levels.

(iii) Programming Level: Process development is completed by converting user requirements into programming code and connecting databases of significant concern by developing programs and logic to find practical solutions of user requirements/problems. Genetic development effort is constantly required to fulfill the user requirements from the databases through programming.

Library or Set of processes during all iterations of genetic creation of processes will require different processes to undergo fitness screening. Fitness screening of processes/sub processes need to be implemented during every review of processes/sub processes.
Function **f** for fitness of processes/sub processes (based on mutation, selection and reproduction etc.) (Darwin [17] and Forrest [23]) need to be applied on set $L^{(i)}$ to get $L^{(i+1)}$ i.e. Set of Processes after qualifying fitness criteria.

$$\mathbf{f}: L^{(i)} \rightarrow L^{(i+1)} \quad (2)$$

Where $i \geq 0$ represents (i+1)th generation/iteration/review of process/sub process development and $L^{(i+1)}$ will become a set

$$L^{(i+1)} = \{P_1, P_2, ..., P_{n'}\}^{(i+1)} \quad (3)$$

This is having n' number of processes with their sub processes after review in current generation. At the time when new system is developed from scratch library L will be developed by newly developed processes, whereas in next generations this library will undergo development and fitness processing by revising earlier processes or by including new processes. After screening of the processes, n is assigned the value of n' and procedure enters to next step.
STEP II: Implementation of Quality Parameters- Then each process and sub process needs to undergo quality screening based on parameters set for quality assurance and implementation as follows:
Function **q** Quality parameters need to be applied on the set L to get QL i.e. Quality Library.

$$\mathbf{q}: L \rightarrow QL \quad (4)$$

Where QL will become a set $QL = \{QP_1, QP_2, ..., QP_n\}$ of Quality processes and sub processes. Parameters of quality may be based on quality standards, syntactical, logical, environmental and organizational factors etc. Quality parameters are usually dependant upon system analysts' vision about the quality and differ from system to system and place to place.
STEP III: Customization- Now Quality Library needs to be customized according to the needs of individual users as follows.

Function **c** of Customization needs to be applied on QL to give a set CQL.

$$\mathbf{c}: QL \rightarrow CQL \quad (5)$$

Where CQL will become a set $CQL = \{CQP_1, CQP_2, ..., CQP_n\}$ of Customized Quality processes and sub processes. Same module may be customized differently for different users, e.g. finance related data may be more detailed for financial experts, whereas summarized for the marketing analyst. Customization of processes depend upon type of user, user's level in organizational hierarchy, authority etc.
STEP IV: Security- Now security criteria need to be applied on Customized Quality Processes for the secure access of the organizational database(s).
Function **s** of Security criteria need to be applied on CQL to generate a set CSQL.

$$\mathbf{s}: CQL \rightarrow SCQL \quad (6)$$

Where SCQL will become a set $SCQL = \{SCQP_1, SCQP_2, ..., SCQP_n\}$ of Secure Customized Quality processes and sub processes, which may be assigned to the authorized users through their login accounts. Security requirements of different processes may be different, e.g. confidential organizational resources demand more secure access, rather public information resources.
STEP V: Total Quality Management (TQM): It is a philosophy toward continually improving your business and products (Ishikawa [31] and Hyde [30]). Information systems need to be improved continuously. Organizational team of system development personnel needs to search for better options for the organizational information system and then to implement them in the system. These options include the comprehensive search for better technology, better processes, enhancement of earlier processes, search for new information needs, other problems related to earlier implementation.

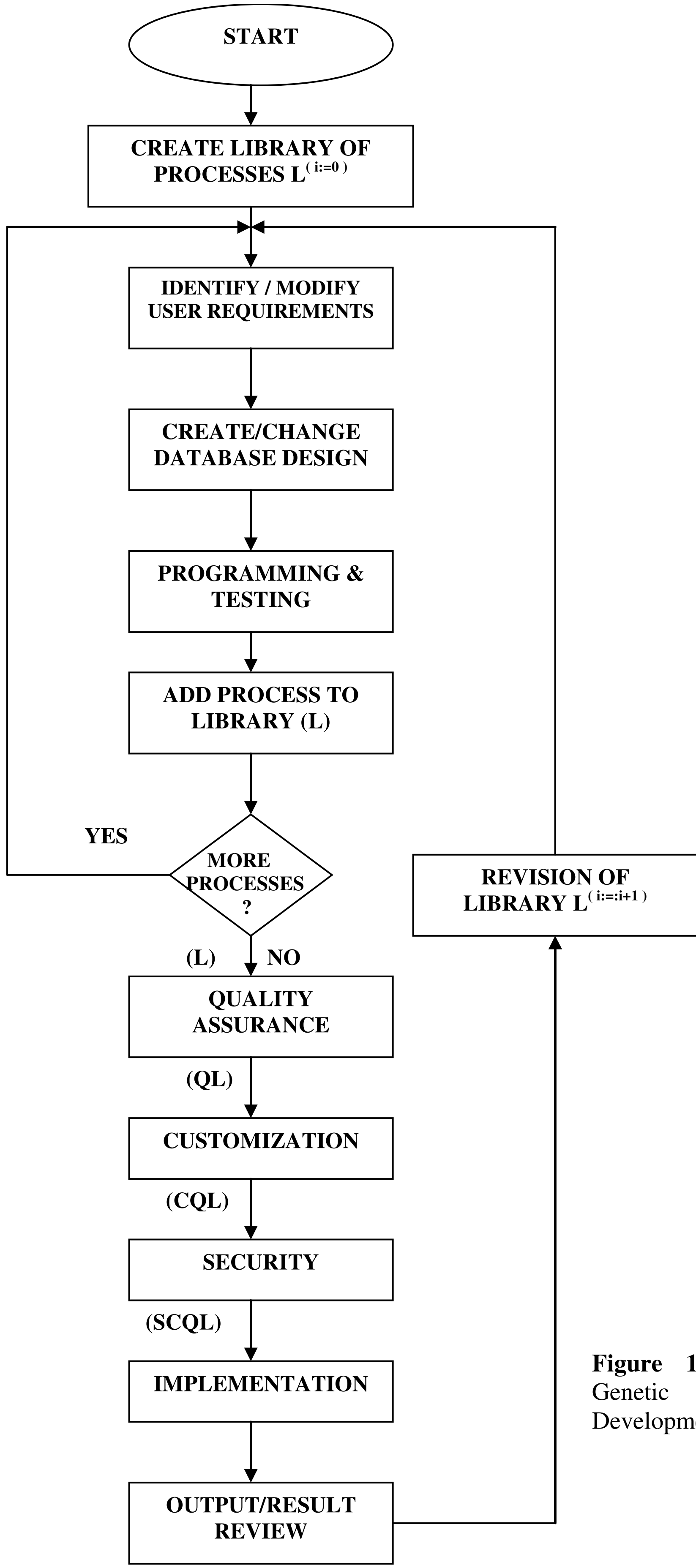

Figure 1: Flow Chart For Genetic Information System Development And Maintenance.

All five steps need to be iterating continuously. Figure 1 illustrates the flowchart that includes the above discussed system development and maintenance procedure with a start and continuous development, but without stopping criteria.
Proposed system development will be a combined and continuous effort of organization personnel (users), system analysts, programmers and Database Administrator (DBA) to develop, implement and maintain information system and business processes. System analyst need to identify organizational structure, personnel, subordinates and their information requirements etc. continuously, DBA need to be involved simultaneously for authorizing required database access to different personnel. Management is involved for identifying their personalized information requirements and its implementation in customized way.

5. PROPOSED MODEL FOR SYSTEM DEVELOPMENT

Different users have different type of information requirements. Type of user, user's level in organizational hierarchy, his/her authority, technical ability of system to fulfil users' information requirements etc. are the factors that play an important role in differentiating different users and their information needs. Categorizing users based on their common information requirements is always a better option, e.g. grouping users on departmental level, common work level etc. Users with common information needs may be grouped together to fulfil their information needs, whereas their authority in organizational hierarchy may help to distinguish their access to the information resources, e.g. marketing department may be grouped together to fulfil their information needs related to marketing aspects. Nevertheless, higher-level managers will have higher as well as summarized access to the databases as compared to the low level executives, who usually have lesser but detailed access to the databases. Processing of information may be distinguished by different type of processing set-ups like one-dimensional data processing, multi-dimensional data processing and knowledge discovery in databases (i.e. data mining or vaguely defined information requirements) etc. Different sets of information requirements demand different kind of processing set-ups and different databases to be connected, e.g. marketing executives may require many databases to be accessed worldwide to get latest information and different type of processing set-ups, so that right kind of information is accessed. On the other hand, financial people require information related to fund flow, sales and expenditures only. Such information requirements, processing set-ups and database connectivity necessitate the development of processes/sub processes for the extraction of required information. Each process is developed to fulfil a homogeneous/related set of users' requirements. Domain of Individual process need to be defined very precisely, so that each process is connected to the databases concerned with its own domain and client/server side logic, security guidelines, quality standards etc. may be settled according to these homogeneous information requirements. Different processes (i.e. heterogeneous set of processes) are combined parallel to each other, with pre-defined accessibility. Implementation of whole system can be explained as a user account login/password authentication based access of a World Wide Web based information system. All management personnel is allocated their personal accounts, which is a page if we talk in terms of World Wide Web. Each user is authenticated through login for its access from the website of the organization. Each account page is a set of windows or drop down menus having some predefined templates for enquiries, access to the database(s) and information transfer (General and specific information) as shown in figure 2.
Programs developed by programmers, are maintained by DBA's library of processes and sub processes meant for the purpose. User may access, query, modify or update database according to the authority provided by DBA. Continually up gradation of library programs is required to fulfil the newly identified information requirements, facilities made available by innovations and new technology. Data warehouse is maintained at back end to collect the information from different sources scattered worldwide.

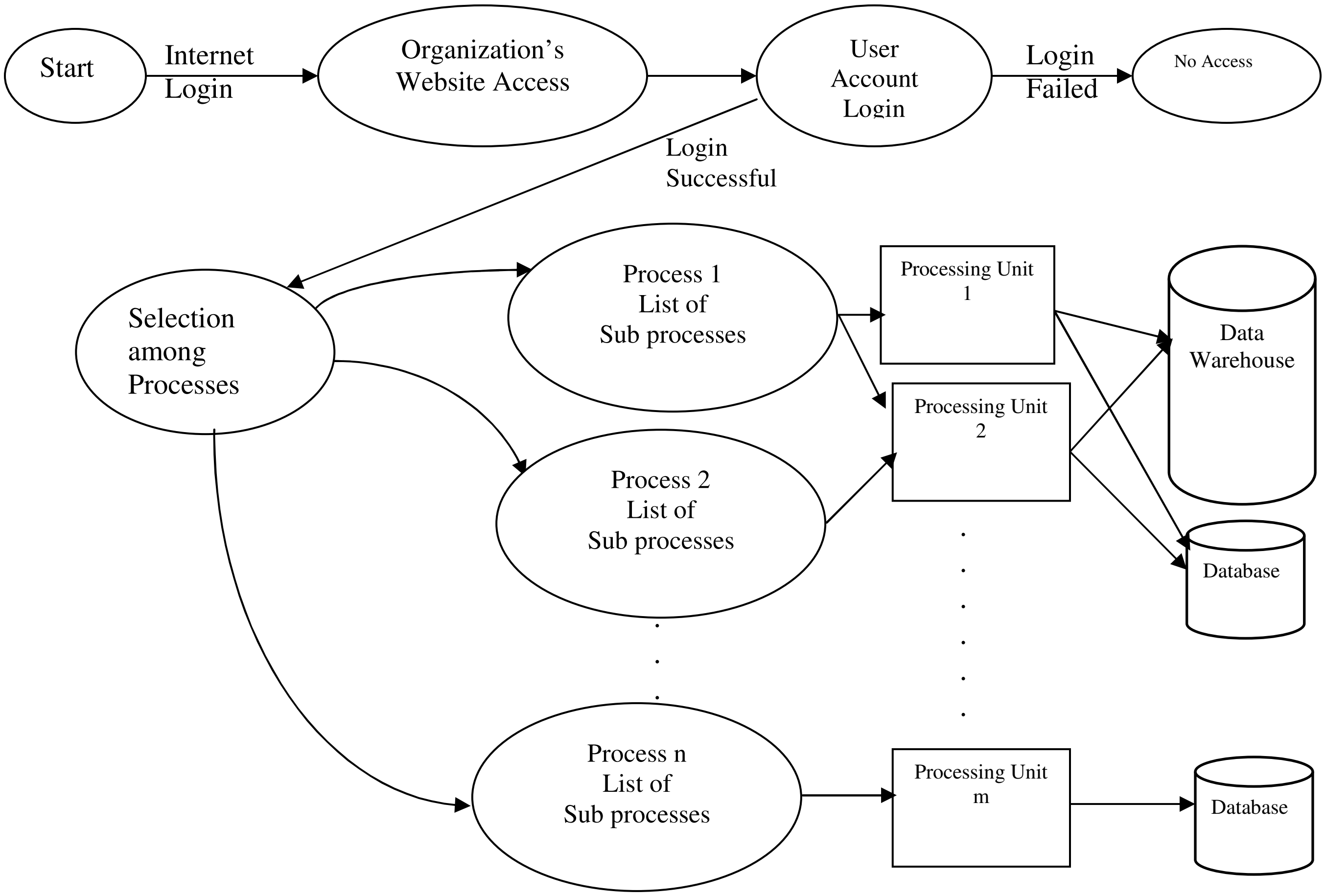

Figure 2: Accessing User account through Internet or Intranet.

User may access its account by connecting from any part of the world through Internet. Each user is provided information from internal and external sources of information of the organization. Key process areas need to be identified and information is collected in central database to furnish the required information, so that one may acquire up-to-date information related to the organization. Different internal/external sources of information are identified and bundled together to furnish the information from external environment of the organization. Key process areas are those critical operations upon which success of any business is dependent. Key Process Areas of any organization may be among Sales, Advertisement, Logistics, Raw Material, Labour, Finance or Production etc., whereas external sources of Information Government Agencies, Competitors, Industry, Suppliers, Customers and many others. These key success factors can be identified and information related to them can be furnished to the user according to their authority. External sources can be identified and their information may be made available directly or after required processing to the user. Organization's Information System Development Team of Analysts, DBAs, Programmers need to work together for identifying emerging information needs of users, considers the matter of inclusion of new technology etc. continuously. Most important aspect of this model is its open behaviour. Processes and sub processes are combined with their logic and access. Individual process is a complete system in all respects having required accessibility, functionality and logic as shown in figure 3. When a user is authorized to access any process or sub process, that process/ sub process is added to user's account, removal of process(s) or sub processes can be done by removing those from the user account.

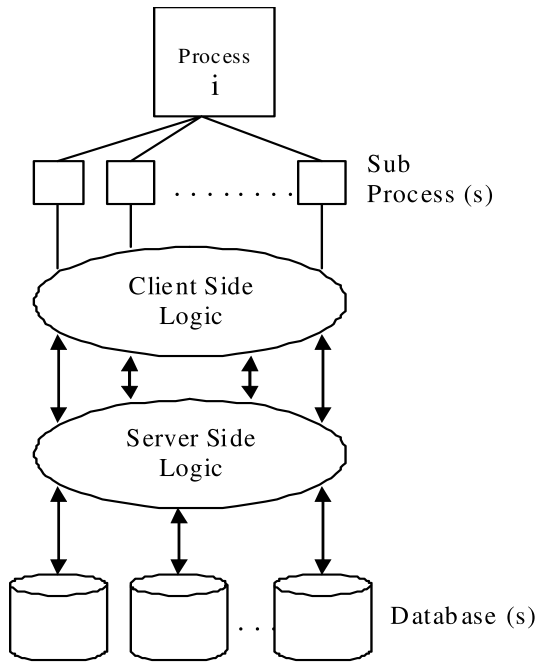

Figure 3: Connectivity of user to database through process/ sub process

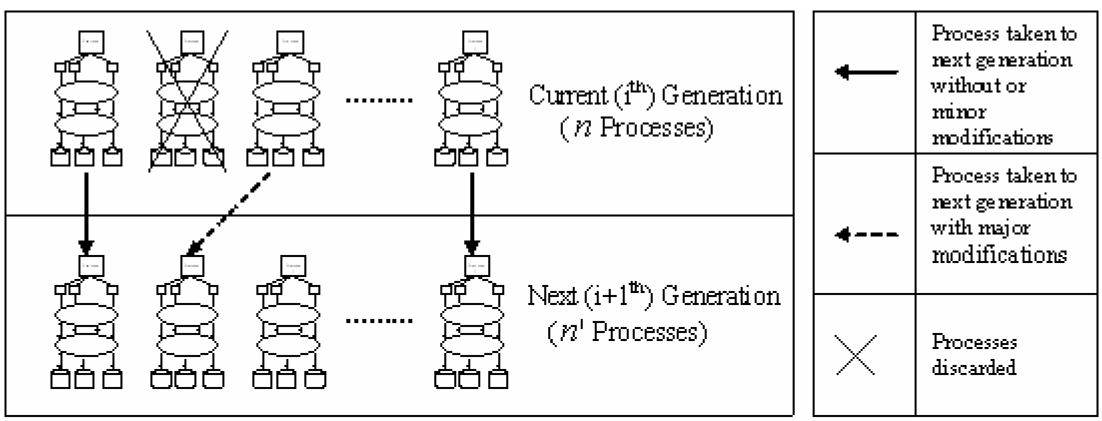

Figure 4: Transition of processes from current generation to next generation.

Different types of database environments may be combined to give access to the user. If data is to be accessed from two or more different type database environments and minimum access time is needed, then these may be combined in organizational data warehouse on regular basis. If real time access is required, then remote database(s) can be accessed with small delays as well. Providing heterogeneous access of database(s) to the user solves problem of adaptation of new technology and processes. Thorough Documentation is needed for the development and updating the set of processes, so that software reuse is maximized. Whenever a change is implemented, affected areas may be rectified and concerned documentation is done. In this way, problems related to maintenance might be controlled. Desired results can only be achieved, if the development and maintenance is carried out continuously. Set of processes is carried forward from current generation to next generation, after implementing the required modifications. Processes may undergo minor modification of its sub-processes, major modification of processes/sub processes, elimination of complete processes (i.e. discarding complete processes) or may require no change in earlier implementation as shown in figure 4.

6. EFFECTIVENESS OF PROPOSED MODEL

Organizational employees can understand and identify information requirements of their organization far better than the professionals hired for system development from outside organizations. People from outside agencies may be involved for technical expertise for software development, but Genetic information system development require involving more own organization's employees as system developers. Different problems faced during software development and support provided by proposed model is as follows:

6.1 Cost: Software development is a process of major concern, but a major chunk of total software cost is consumed in maintenance. Genetic development and maintenance cost almost same expenditure, but it helps in developing personalized software, enhancing software reuse and early identification as well as improvement of newly generated information needs on continuous basis, which is profitable in long run. It is due to the reason that the organizational people may understand their organization better than anyone else from outside agency and continuous review enhances the system performance.

6.2 Functionality: Customized or personalized system development helps in achieving higher user satisfaction, as user requirements are understood more closely and system is updated according to these requirements.

6.3 Reliability: When user needs are understood well and are improved whenever changes are required, testing of system in real environment gives maximum fruitful results, if processes are implemented in a well manner and reliability is enhanced.

6.4 Maintainability: Genetic system development and maintenance inherently concentrate on the maintainability of the system.

6.5 Adaptability: System is opened to involve any type of technology and access of databases may be provided directly to the user or it may be connected to the centralized database and suitable changes may be implemented in concerned programs or logic of the system for different users.

6.6 Efficiency: Online/Internet based systems are as good as any offline systems. In case of Online Analytical Processing or Data Mining processing, a small delay in responses may be due to inability of state-of the-art technology to meet desired response time, but still current technology can meet all kinds of information requirements with negligible delays. Open connectivity of improved technology and continuous maintenance of system can help to achieve the desired efficiency.

6.7 Portability and Reuse: Internet connectivity is available with almost all type of technologies with little modifications, so portability is up to the mark for the system. Proposed model sufficiently assures the requirement of proper reuse of software.

6.8 Usability: Users may be trained by simple training programs to use the system and system needs to be developed more intuitively. Different activities of the user pages may be combined in separate windows or drop down menus and details can be accessed through in-depth navigation of the window or connected pages.

6.9 Security: Security of data is always a biggest question, while accessing Internet or online processing. Access to the database need to be secured and multiple levels of security layers may be involved, where authentication is required. Security levels may differ from process to process or database to database as per the requirement.

6.10 Quality: If above discussed parameters are successfully achieved, then quality of information system is assured, not only from the outputs of the system, but also from quality standards defined for the purpose as well.

Proposed model is implemented in almost twenty organizations operating in a variety of business domains. Genetic system development and maintenance model is helpful in finding the solution of many problems related to software development discussed above. However according to the scope of this paper, investigations for testing the model needs to be guided towards reliable maintainability and costs involved. Following case studies concentrate the investigations toward these parameters. Proposed model definitely show the improvement in adaptive and perfective maintenance, so our investigations need to be concentrated upon corrective maintenance/change management at low or no extra cost. So following section will concentrate our investigations in these directions.

7. CASE STUDIES AND HYTPOTHESIS TESTING

7.1 CASE STUDIES

7.1.1 CASE STUDY 1: PEPSI FOODS Ltd. is a renowned soft drink manufacturing, multinational company operating in most of the countries worldwide. Soft drink industry is marketing based business and purely dependent upon eating and drinking habits of the people. Pepsi Foods Ltd. has diversified its business in India not only in soft drinks but also in eatables goods. Information system of the company needs to be efficient enough to meet the dynamic information requirements of the industry. As soft drink industry is marketing operations intensive, its information requirements are highly dynamic. Changing information needs has created a lot of difficulty for the software developers, as maintenance costs rise beyond the estimates and has caused unreliability even after spending sufficient amounts on the system development and maintenance. Earlier software development methodologies and models used, could not meet the requirements.

7.1.2 CASE STUDY 2: Ranbaxy Laboratories Ltd. is a leading pharmaceutical multinational company of India. Rising competition from local and global players has caused the need of an efficient information system to meet the challenge of maintaining leadership and expansion of business globally. Dynamic information needs has forced the company to think over its information technology and information system strategy. Earlier information system development has failed to meet the challenge. Company is looking for any permanent solution for their information needs.

7.1.3 CASE STUDY 3: Usha Power Tec. is a company committed to meet the need for power conditioning in India. Company has record tremendous growth in future and has bright future prospects as well, but rising competition and dynamic information requirements of the information systems has forced the company to think on strengthening its information system. Earlier development is unable to meet the requirement and company is lifelong solutions for its information systems.

7.1.4 CASE STUDY 4: Megaleap Inc. is an online retailing organization. Company deals in a variety of e-Stores to provide a storefront to a variety of manufacturers and retailers. Company operates in a highly dynamic environment. Company has its information system operating, but dynamism of customer choices, retailers offers etc. has emerged the need for incorporating the information dynamism to be incorporated in its information system. So company is looking for long-term solution of its information system development and maintenance at optimized costs.

7.2 Hypothesis Testing

Teams of researchers and software developers were involved in the process of software development for the organizations discussed above. Genetic information system development and maintenance process was tried and the results were tested against data collected from

earlier developments for the same organizations and actual data of the development process and results of case study 1 are presented here, similar results are observed for other organizations discussed above.
Research methodology for proposed research work includes implementation of proposed model in diversified business organizations, Collection of data from actual implementation, Collection of dataset from standard organizations, Comparison of results, Hypothesis testing and concluding with results etc.
For testing parameters like reliable maintenance of information system and total system costs are of major concern. Proposed model is highly adaptive due to adoption of Internet based technologies and team involved on continuous basis for system development work for the perfection of the system. So, Hypothesis testing is performed against the parameters related to reliable corrective maintenance and costs involved. Datasets are collected from the organizations' offices scattered worldwide, whereas Criteria for collection of data was based on Type of Business organization, System development/Maintenance Costs, Size of System, Environment of the system, Scale of operations for system etc. and these factors were also normalized according to the characteristics of the system under study.

7.2.1 Reliability testing through numbers of faults received after initial implementation of information systems.

Reliability of a system depends on the number of faults received and action taken within same week as faults received. First we test the hypothesis related to number of faults received after initial implementation of system. Null and alternate hypothesis for reliability testing in terms of number of faults received after initial implementation of system, are as follow:

H_0 = There is no significant difference between reliability trends of proposed model and other earlier models in terms of number of faults received after new system is implemented.

H_1 = There is a significant difference between reliability trends of proposed and other earlier models.

Significant level is kept 98% for the acceptance of Null hypothesis. On the basis on data collected from earlier implementations and data collected from proposed model's implementation, following graphical comparison is produced. Figure 5 include the difference between the faults received for benchmark system and system under study.

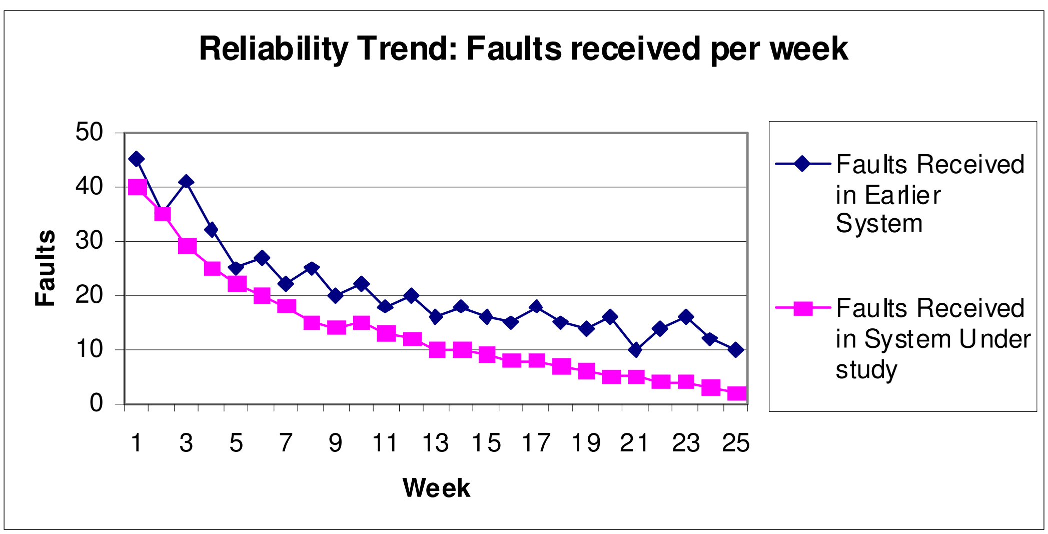

Figure 5: Reliability Trend: Faults received per week after initial implementation

Continuous decline in number of faults is recorded from proposed methodology as compared to the earlier system. Above data are applied to test the hypothesis, chi-square tests are performed for testing goodness of fit and results show the rejection of null hypothesis implying the acceptance of alternative hypothesis. i.e. there is a significant difference between faults received from current system and earlier implementations.

7.2.2 Reliable corrective maintenance testing through action taken within same week against the faults received after initial implementation of information system.

Reported faults need to be corrected as soon as possible. Faults received and action taken to correct them within same weeks as they received is the criteria used for testing reliable corrective maintenance. Null and alternate hypothesis for reliability testing are as follow:

H_0 = There is no significant difference between reliability trends of proposed model and other earlier models in terms of action taken against faults received after new system implemented.

H_1 = There is a significant difference between reliability trends of proposed and other earlier models.

Significant level is kept 98% for the acceptance of Null hypothesis. Figure 6 include the comparison of action taken against the reported faults within same week for proposed model.

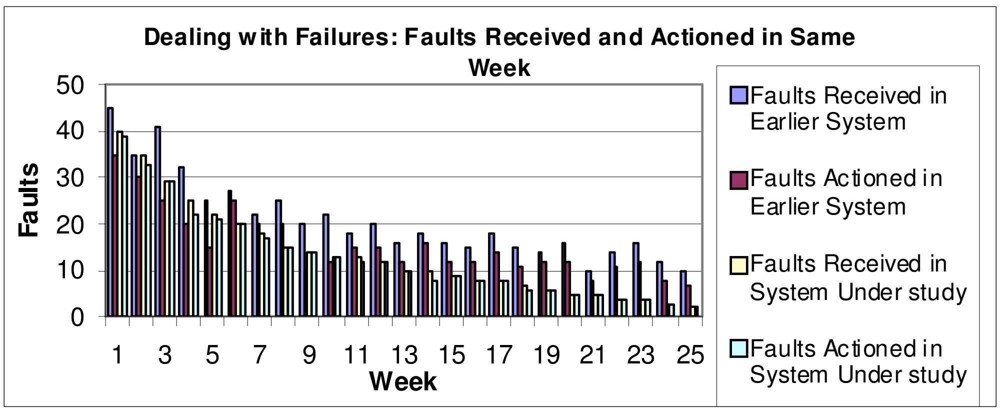

Figure 6: Dealing with failures: Faults Received and Action taken in Same Week

Figure 7 includes the comparison of percentage of faults corrected within the same week as fault(s) reported for earlier system development and for proposed model implementation.

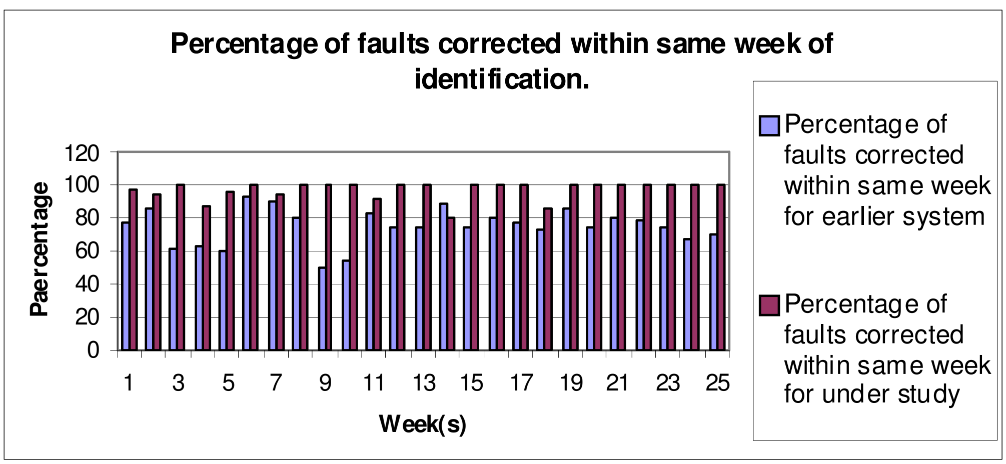

Figure 7: Percentage of faults corrected within same week of identification.

Above data is applied to test the hypothesis, chi-square test for testing goodness of fit and results indicate the rejection of null hypothesis implying the acceptance of alternative hypothesis and indicating a significant improvement in taking action against faults received.

7.2.3 Testing the difference in cumulative costs for the systems based on other models and proposed model.

Genetic information system development and maintenance involves a team of company employees to look after the system continuously contributing to higher system development costs, but maintenance costs for such system development will be lowest possible or will hide behind development cost. So, cumulative costs (including system development and maintenance costs including effort involved) for the system will be very similar to the system developed and maintained through other models. Null and alternate hypothesis for testing the differences between cumulative development and maintenance costs for earlier models and proposed model are defined as follow:

H_0 = There is no significant difference between cumulative development and maintenance costs of proposed model and other earlier models.

H_1 = There is a significant difference between cumulative development and maintenance costs of proposed and other earlier models.

Significant level is kept 95% for the acceptance of Null hypothesis. On the basis collection of standard and actual datasets following graphical comparisons are produced. Figures 8 and 9 include the comparison of the systems based on costs involved.

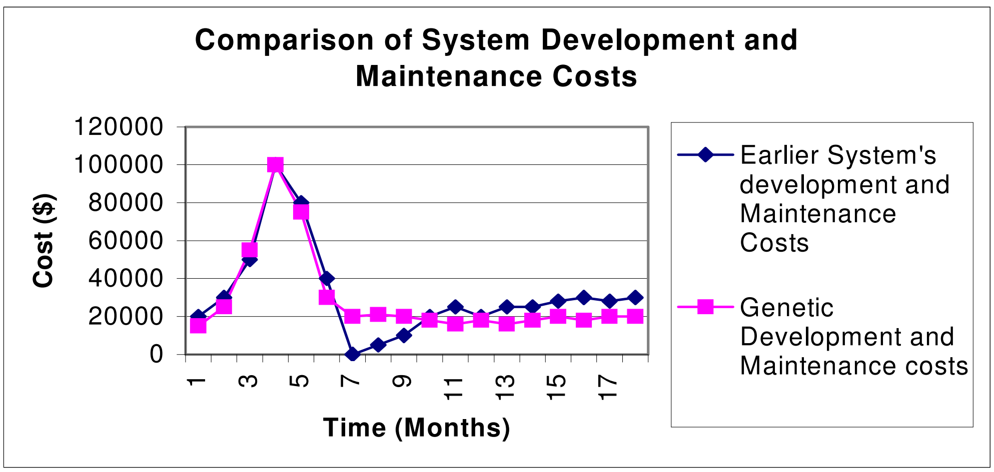

Figure 8: Comparison of System Development and Maintenance Costs

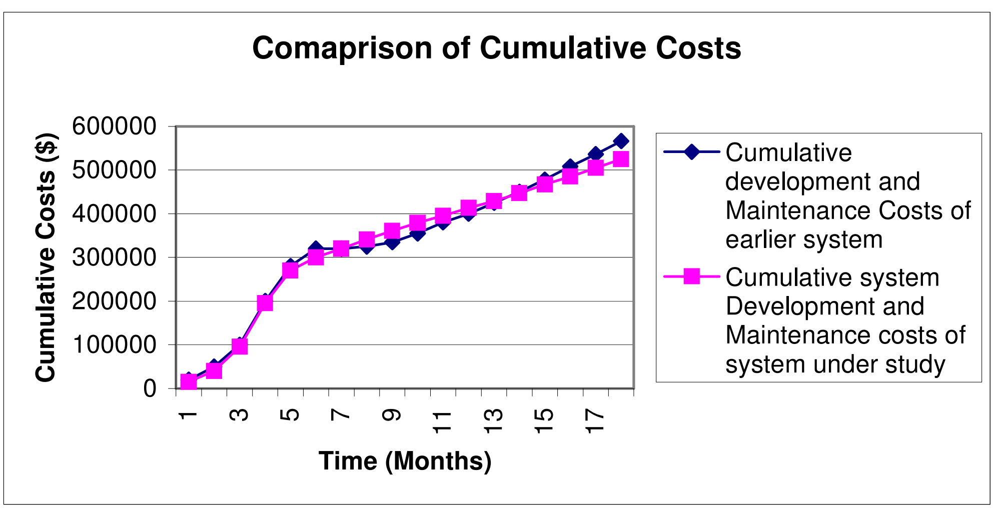

Figure 9: Comparison of Cumulative Costs

From comparison, we find that with time, maintenance costs for other benchmark system increases very rapidly, whereas it remains under control for proposed model. Cumulative costs for system development and maintenance are almost same for a newly developed system, followed by decline in costs for other models and then followed by tremendous increase in total costs of the system development and maintenance for benchmark system, whereas for proposed model cumulative costs are lesser in long run. Above data is applied to test the hypothesis chi-square test for goodness of fit and results indicate the acceptance of null hypothesis implying the rejection of alternative hypothesis, which implies that there is no significant difference between the total costs of the earlier models and proposed model.

7.2.4 Comparative analysis to check the effect of proposed model over software reuse

Effect of proposed model on software's reuse is studied and compared, based on the Lines of Code (LOC) reused during different revisions of software. The de-facto standard for measuring the reuse level (Poulin [44]), is the percentage of software (LOC) reused as compared to the Total size of software (LOC), i.e.

$$\frac{\text{Reused Software}}{\text{Total Software}} X\,100 \qquad (7)$$

Software reuse is compared for system under study with earlier used system in terms of percentage of lines of codes reused during each revision. Figure 10 displays the percentage of software reused during each revision.

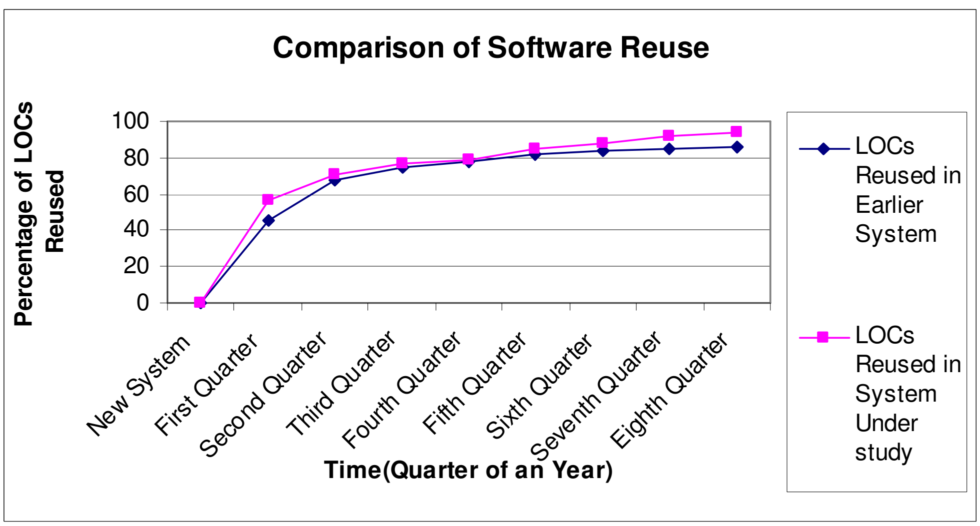

Figure 10: Comparison of Software Reused in Percentage of Lines of Code reused

Analysis of above comparative study shows improvement in software reuse for the system developed through proposed model.

7.2.5 Generalized comparison of proposed model with other models

Comparative study of different models based on general observations and expert opinion about different models, is summarized in Table 1, it contains the Comparison of models based on costs, reliability, maintainability, adaptability, portability and reusability.

Model(s) / System Factors	Water Fall Model	Spiral Model	Iterative Model	Proto-typing model	Component Based Development	Genetic development and maintenance
Development cost	Moderate	Moderate	High	High	High	High
Maintenance cost	High	High	Moderate	Moderate	Moderate	Low
Reliability	Low	Moderate	High	Moderate	Moderate	High
Maintainability	Low	Moderate	Moderate	Moderate	Moderate	High
Adaptability	Low	Moderate	Moderate	High	High	High
Portability	Low	Moderate	Moderate	High	High	High
Reusability	Low	Moderate	Moderate	Moderate	High	High

Table 1: Comparison of different methodologies with cost, reliability, maintainability and adaptability of information systems

Above these advantages, many other aspects like user satisfaction, goal orientation, human and organizational factors are also observed to be better for proposed model as compared to any other implementation.

8. CONCLUSION AND FUTURE DIRECTIONS

From the paper, we conclude that Organization and Information Systems progress step by step together, so the development process of information systems cannot be restricted to a limited span of information system development project. Increasing complexities of business processes and information systems demand better change management, Maximum Software Reuse and continuous effort for system development and maintenance. Positive results from the implementation of proposed model indicate the future success of Genetic information system development and maintenance model for new age information systems and for the success of the organizations for improving their business processes. From results, we conclude that proposed model helps in software maintainability and software reuse significantly without extra expenditure. Authors are working on the implementation of the proposed process and model in other diversified business organizations and for its acceptance worldwide. We hope that problems related to software development and maintenance will be minimized, by using Genetic system development and maintenance model and organizations will be able to flourish their operations without any worry about the maintenance and cost related problems of the information systems.

9. REFERENCES

1. Alavi M. *"An Assessment of the Prototyping Approach to Systems".* Communications of the ACM, 27, 6, 556-564, 1985

2. Aoyama M., *"Component-Based Software Engineering: Can it Change the Way of Software Development?".* Proceedings Volume II of the 1998 International Conference on Software Engineering April, 1998

3. Armenise P., Bandinelli S., Ghezzi C. and Morzenti A. *"A survey and assessment of software process representation formalisms"*. International Journal of Software Engineering and Knowledge Engineering, Vol. 3 No. 3, pp. 410-26, 1993

4. Bandinelli S., Fugetta A. and Grigoli S. *"Process modelling in the large with SLANG".* Proceedings of the 2nd International Conference on Software Process, Berlin, pp. 75-93, 1993

5. Basili V.R. and Turner A.J. *"Iterative Enhancement A Practical Technique for Software Development"* IEEE Transactions on SoftwareEngineering 1, 4, 390-396, 1975

6. Bersoff E. *"Elements of software configuration management"*. IEEE Trans. Software Engg., 1984

7. Bider I., Johannesson P. and Perjons E. *"Goal-Oriented Patterns for Business Processes".* position paper for Workshop on Goal-Oriented Business Process Modeling (GBPM'02), 2002

8. Boehm B. W., Gray T. E. and Seewaldt T. *"Prototyping Versus Specifying: A Multiproject Experiment"*. IEEE Transactions on Software Engineering, SE-10, 4, 290 -402, 1984

9. Boehm Barry W. *"A spiral model of software development and enhancement"*. IEEE Computer, 21(5):61-72, 1988

10. Brandtweiner R and Scharl A. *"An institutional approach to modeling the structure and functionality of brokered electronic markets"*. International Journal of Electronic Commerce 3(3), 71-88, 1999

11. Bubenko J., Rolland C., Loucopoulos P. and De Antonellis V. *"Facilitating 'fuzzy to formal' requirements modelling".* Proceedings of the IEEE 1st Conference on Requirements Engineering, ICRE'94, Colorado Springs, CO and Taipei, pp. 154-8, 1994

12. Burrows C., George G. and Dart S. *"Configuration Management".* Ovum Ltd., 1996

13. Cho N. J. *"Internet business in Korea: the state-of-art"* Management and Computer 188-220, 1999

14. Conradi R. and Westfechtel B. *"Towards a Uniform Version Model for Software Configuration Management"*. Proceedings of the Seventh International Workshop on SoRware Configuration Management, New York., 1997

15. Dart S. *"Spectrum of Functionality in Configuration Management Systems".* Technical Report CMU/SEI-90-TR-11, Software Engineering Institute, Pittsburgh, Pennsylvania, December, 1990

16. Dart S. *"Concepts in Configuration Management Systems".* Proceedings of the Third International Workshop on Software Configuration Management, pages 1-18. ACM SIGSOFT, 1991

17. Darwin Charles. *"The Origin Of Species".* 6th Edition, John Murray, London, P. 278, 1902

18. Decker S., Daniel M., Erdmann M. and Studer R. *"An enterprise reference scheme for integrating model based knowledge engineering and enterprise modeling".* Proceedings of the 10th European Workshop on Knowledge Acquisition, Modeling and Management, EKAW'97, Springer-Verlag, Heidelberg, Lecture Notes in Artificial Intelligence, LNAI 1319, 1997

19. Drucker Peter F. *"The Practice of Management. Contributors".* Publisher: Harper & Row., New York, 1954

20. Estublier J., Dami S. and Amiour M. *"High Level Process Modeling for SCM Systems"*. Proceedings of the Seventh International Workshop on Software Configuration Management, New York., 1997

21. Feller P. *"Configuration Management Models in Commercial Environments"*. Technical Report CMU/SEI-91-TR-7, Software Engineering Institute, Pittsburgh, Pennsylvania, March, 1991

22. Floyd C. *"A Systematic Look at Prototyping, in: Budde, R., Kuhlenkamp, K., Mathiassen, L. and Zullighoven, H. (Eds.) Approaches to Prototyping"*. Springer-Verlag: Heidelberg, 1-17, 1984

23. Forrest Stephanie. *"Genetic Algorithms"*. ACM Computing Surveys, Vol. 28, No. 1, 1996

24. Frakes William and Terry Carol. *"Software Reuse: Metrics And Models"*. ACM Computing Surveys, Vol. 28, No. 2, 1996

25. Gomaa H. *"The Impact of Rapid Prototyping on Specifying User Requirements"*. ACM SIGSOFT Software Engineering Notes 8, 2, 17-28, 1983

26. Gonsalves G.C., Lederer A.L., Mahaney R.C. And Newkirk H. E. *"A customer resource life cycle interpretation of the impact of the World Wide Web on competitiveness: expectations and achievements"*. International Journal of Electronic Commerce 4(1), 103-120, 1999

27. Gordon V.S. and Bieman J.M. *"Rapid Prototyping: Lessons Learned"*. IEEE Software, 12, 1, 85-95, 1994

28. Hamill J. *"The Internet and international marketing"*. International Marketing Review 14(5), 300-323, 1997

29. Hunt J., Lamers F., Renter J. and Tichy W.F. *"Distributed Configuration Management via Java and the World Wide Web"*. In Proceedings of the Seventh International Workshop on Software Configuration Management, New York., 1997

30. Hyde A. *"The Proverbs of Total Quality Management: Recharting the Path to Quality Improvement in the Public Sector"*. Public Productivity and Management Review, 16(1), 25-37, 1992

31. Ishikawa K. *"What is Total Quality Control?"*. The Japanese way. Englewood Cliffs, New Jersey, Prentice- Hall, 1985

32. Jacobson I., Booch G. and Rumbaugh J. *"Unified Software Development Process"*. Object Technology Series, Addison-Wesley, New York, 1999

33. Jarzabek S. and Ling T.W. *"Model-based support for business reengineering"*. Information and Software Technology, Vol. 38 No. 5, pp. 355-74, 1996

34. Kaushaar J.M. and Shirland L. E. *"A Prototyping Method for Applications Development End Users and Information Systems Specialists"*. MIS Quarterly, 9, 4, 189 -197, 1985

35. Khomyakov M. and Bider I. *"Achieving workflow flexibility through taming the chaos"*. paper presented at OOIS 2000 – 6th International Conference on Object Oriented Information Systems, Springer-Verlag, Berlin, pp. 85-92, 2000

36. Kramer Stefan And Kaindl Hermann. *"Coupling And Cohesion Metrics For Knowledge-Based Systems Using Frames And Rules"*. ACM Transactions On Software Engineering And Methodology, Vol. 13, No. 3, Pages 332–358, 2004

37. Kueng P. and Kawalek P. *"Goal-based business process models: creation and evaluation"*. Business Process Management Journal, Vol. 3 No. 1, pp. 17-38, 1997

38. Leblang D.B. *"Managing the Software Development Process with ClearGuide"*. Proceedings of the Seventh International Workshop on Software Configuration Management, New York., 1997

39. Lee S. *"Business use of internet-based information systems: the case of Korea".* European Journal of Information Systems, Vol. 12, Issue 3, Sep, 2003

40. Marca D.A. and McGowan C.L. *"Business Process and Enterprise Modeling".* Eclectic Solutions, Inc., San Diego, CA, 1993

41. Milewski B. *"Distributed Source Control System".* In Proceedings of the Seventh International Workshop on Software Configuration Management, New York., 1997

42. Nauman J.D. and Jenkins M. *"Prototyping: The New Paradigm for Systems Development".* MIS Quarterly, 6, 3, 29-44, 1982

43. Palvia P. and Nosek J. T. *"An Empirical Evaluation of System Development Methodologies".* Information Resources Management Journal, 3,23-32, 1990

44. Poulin J. S. "*Measuring Software Reuse: principles, practices, and economic models"*. Reading, MA, Addison-Wesley, 1997

45. Rolland C. *"L'e-lyee: L'escritoire and Lyeeall".* Information and Software Technology, Vol. 44, pp. 185-94, 2002

46. Rolland C. *"Reasoning with goals to engineer requirements"*. ICEIS'03– Fifth International Conference on Enterprise Information Systems, Angers, France, pp. 11-19, 2003

47. Royce W. W. *"Managing the Development of Large Software Systems: Concepts and Techniques".* Proceedings WESCON, August, 1970

48. Rumbaugh J., Blaha M., Premerlani W., Eddy F. and Lorensen W. *"Object Oriented Modeling and Design"*. Prentice-Hall, New York, NY, 1991

49. Tate G. and Verner J. *"Case Study of Risk Management, Incremental Development and Evolutionary Prototyping".* Information and Software Technology 42, 4, 207-214, 1990

50. TICHY W. F. *"Tools for software configuration management"*. Proceedings of the International Workshop on Software Version and Configuration Control (Grassau, Germany), J. F. H. Winkler, Ed., Teubner Verlag, 1–20, 1988

51. Weill P. *"The relationship between investment in information technology and firm performance: a study of the valve-manufacturing sector"*. Information Systems Research 3(4), 307-333, 1992