ABSTRACT

Software components are prime constituents of modern software systems, where high degree of flexibility, quality, usability, efficiency, reliability and portability are expected. By using components, many tens of thousands of man-hours of development time could be saved. Component based development system is based on many aspects such as creating new components, reusing existing components and integration of heterogeneous components.

This work is focused on developing new component for data mining; and developing new components in different platforms and integrating them for specific application. Development of new component relates with creating new component for the process, which will be used frequently and finds very high demand in the business world. Integration of existing components relate with providing interoperability between components developed in different languages and in different platforms, to have a new application system.

This work focuses on two aspects. First one is on creation of an EJB component for finding frequent itemsets, which is used for having decision-making process in any business application. Second one is on the integration of EJB and .NET components through the interoperability tool called Platform Interoperable Solution (PIS).

This work concerns with creation of a new component for mining frequent itemsets and integration of heterogeneous components by providing interoperability between them. Frequent itemsets mining is one of the fundamental problems in data mining and has many applications such as association rule mining, inductive databases, and query expansion. These applications require fast implementations of frequent itemset mining.
The most well known and influential algorithms are Apriori and FP-growth. In Apriori algorithm, multiple scanning of the transactional database is needed to find frequent itemsets, which is too costly. This algorithm performs a breadth-first search in the search space by generating candidate k+1-itemsets from frequent k-item sets. Further it needs n+1 number of scanning, which takes more time to find the frequent itemsets. The frequency of an itemset is computed by counting its occurrence in each transaction. FP-growth is a well-known algorithm that uses the FP-tree data structure to achieve a condensed representation of the database transactions and employs a divide-and-conquer approach to decompose the mining problem into a set of smaller problems. In the first part of this work, an algorithm has been introduced for finding frequent itemsets with less number of scanning and without tree construction.

This work describes an algorithm to find the frequent itemsets, which is a vital process for decision-making. In the proposed algorithm, the number of scanning process has been reduced compared to Apriori algorithm and there is no need of tree construction as in FP-Growth algorithm. This algorithm is simple, easy to implement, easy to understand and involves less computational overhead. This algorithm is compared with Apriori and FP-growth algorithm. From the results, it has been found that this algorithm is simple, fast and easy to understand. Proposed algorithm is made in the form of an EJB component. This EJB component can be accessed from anywhere in an interconnected environment involving different platforms. Through this component, up to 2-itemsets can be found directly from the Support Count Table (SCT) and for identifying more than 2-itemsets, apart from checking for connectedness (fully connected) a logical AND operation has to be performed.
Even though the component had been developed in EJB, .NET client also can access it through the interoperability tool PIS. Thus the EJB component can be accessed by both Java client and .NET client in the process of decision-making for their business application by finding frequent itemsets.

Integration of software components is an emerging concept in the software development process. Interoperability is the key concept for integrating the components developed in different languages or for different platforms. The second part of this work describes an automated tool, Platform Interoperable Solution (PIS), which provides interoperability between heterogeneous components.

The Institution of Electrical and Electronic Engineers (IEEE) provides the generally accepted definition of interoperability as: "The ability of two or more systems or components to exchange information and to use the information that has been exchanged."

The Internet Inter-Orb Protocol (IIOP) and the Simple Object Access Protocol (SOAP) are the existing interoperable standards claim to alleviate the interoperability pains. SOAP is text-based and verbose. Communications using binary protocols such as IIOP will in almost all cases outperform those that use SOAP as the underlying protocol. The IIOP protocol used to access the object is well established and quite efficient compared to the fancier and less powerful Web Services. The development of an application using IIOP encounters some pitfalls, often caused by the complex and non-trivial configuration and use of the various application servers hosting the EJB services. In particular, the service naming and location may heavily differ among the various servers.

IIOP .NET is a .NET remoting channel based on the IIOP protocol, similar to the one used by Java's RMI/IIOP. IIOP is part of the CORBA standard. IIOP .NET acts as an ORB (a CORBA object request broker), which
makes objects defined in .NET application accessible to other remote ORBs, and vice-versa. Java RMI/IIOP implements a subset of the CORBA type system (due to some limitations in Java's type system) and roughly provides the same features as IIOP .NET for the J2EE platform.

In order to overcome the above said pitfalls, an automated design tool Platform Interoperable Solutions (PIS) has been developed using IIOP.NET. PIS eases the task of configuration process. PIS provides the interoperability between the platforms J2EE and .NET. PIS tool is machine independent and location independent. Using PIS, the integration of different components belonging to EJB and .NET can be achieved easily and effectively. In PIS, there is no need for worrying about class path settings, location of components and machine. Through PIS, the interoperability between EJB and .NET can be achieved both on server side and client side. Once the interoperability is achieved then the integration of the EJB and .NET components becomes very simple. PIS had been developed using existing software only.

PIS involves less development cost, maintenance cost and less manual work. It offers the flexibility of configuring components either on client side or on server side. Components of different platforms and different vendors can be integrated for different applications using Platform Interoperable Solution Tool, which provides interoperability. PIS is viewed as an efficient design tool for component based software development. The performance of the tool PIS has been checked for a Banking Application. EJB components had been created for the banking processes of Withdraw and Registration. A .NET component had been created for the banking process of Money Transfer. The complete Banking Application is realized by integrating both EJB and .NET using PIS.
The tool PIS is designed using JAVA, XML and JSP. The integration of heterogeneous components is achieved through the tool PIS, which provides interoperability between heterogeneous platforms. In PIS, the configuration server is developed using java. Thus for using PIS, no additional software is needed. The tool PIS is machine independent and easy to handle. Without having internal working concept of this tool, anyone can use this tool to provide interoperability between heterogeneous platforms. Manual work involved for configuring components from one form to another has been reduced. The time taken for providing interoperability between heterogeneous components is reduced by this tool.

Following studies are made in this work:

- EJB component is created for finding frequent itemsets with a simple algorithm. And the result of this algorithm is compared with Apriori and FP-Growth algorithm.

- An automated tool Platform Interoperable Solution tool has been created for providing interoperability between heterogeneous components like EJB and .NET.

- A Banking Application has been developed by integrating the heterogeneous EJB and .NET components using the tool PIS.

This thesis work demonstrates mining of frequent itemsets in a simple way, involving less time and manual work. The new proposed algorithm is compared with the Apriori and FP-Growth algorithms. And the interoperability between the heterogeneous platforms (J2EE and .NET) is provided with the help of the tool PIS. Using PIS concept a Banking application has been developed through integration of heterogeneous components.