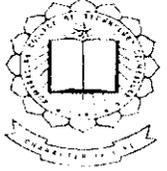


**BUSINESS INTELLIGENCE SYSTEM
USING SERVICE ORIENTED
ARCHITECTURE**



P. 3580

PROJECT REPORT

Submitted by

A.SWATHIKA

Reg. No: 0920108024

*In partial fulfillment for the award of the degree
of*

MASTER OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

COIMBATORE – 641 049

APRIL 2011

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

COIMBATORE – 641 049

Department of Computer Science and Engineering

PROJECT WORK

PHASE II

APRIL 2011

This is to certify that the project entitled

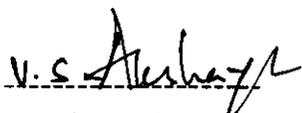
BUSINESS INTELLIGENCE SYSTEM USING SERVICE ORIENTED ARCHITECTURE

is the bonafide record of project work done by

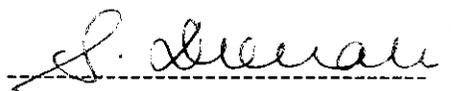
A.SWATHIKA

Register No: 0920108024

of M.E. (Computer Science and Engineering) during the year 2010-2011.


Project Guide

[V.S. AKSHAYA]


Head of the Department

Submitted for the Project Viva-Voce examination held on 25-04-2011


Internal Examiner


External Examiner

DECLARATION

I affirm that the project work titled
 " BUSINESS INTELLIGENCE SYSTEM USING
 SERVICE ORIENTED ARCHITECTURE "

being submitted in partial fulfillment for the award of
 M.E

degree is the original work carried out by me. It has not formed the part
 of any other project work submitted for the award of any degree or
 diploma, either in this or any other University.


 SWATHIKA.A

Register No: 0920108024

I certify that the declaration made above by the candidate is true


 Mrs. V.S.AKSHAYA M.E., (Ph.D.,)

Assistant Professor (SRG)
 Department of Computer Science and Engineering,
 Kumaraguru College of Technology,
 (An Autonomous Institution)
 Coimbatore-641 049.

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Lord Almighty for enabling me to complete this project.

I express my profound gratitude to our Chairman **Padmabhusan Arutselvar Dr.N.Mahalingam, B.Sc., F.I.E.**, for giving this opportunity to pursue this course.

I would like to thank **Dr.S.Ramachandran, Ph.D.**, Principal for providing the necessary facilities to complete my thesis.

I take this opportunity to thank **Mrs.P.Devaki M.E., (Ph.D)**, HOD, Department of Computer Science and Engineering, for his precious suggestions.

I thank all project committee members for their comments and advice during the reviews. Special thanks to **Mr.V.Subramani M.Tech.**, Associate Professor, Department of Computer science and Engineering, for arranging the brain storming project review sessions.

I register my hearty appreciation to the Guide **Mrs.V.S.Akshaya M.E., (Ph.D)**, Assistant Professor(SRG), Department of Information Technology, my thesis advisor. I thank for her support, encouragement and ideas. I thank her for the countless hours she has spent with me, discussing everything from research to academic choices.

I would like to convey my honest thanks to all **Teaching** staff members and **Non Teaching** staffs of the department for their support. I would like to thank all my classmates who gave me a proper light moments and study breaks apart from extending some technical support whenever I needed them most.

I dedicate this project work to my **parents** for no reasons but feeling from bottom of my heart, without their love this work wouldn't possible.

TABLE OF CONTENTS

CONTENTS	PAGE NO
Abstract	vii
Abstract (Tamil)	viii
List of Figures	ix
List of Tables	x
List of abbreviations	x
1. INTRODUCTION	
1.1 Overview of business intelligence	1
1.2 Three kinds of decisions	2
1.2.1 Better business performance	2
1.3 Architecture for business intelligence	3
1.3.1 Informed Decision-Making based on Data	3
1.3.1.1 Source Systems	4
1.3.1.2 Data Staging Area	4
1.3.1.3 Data Warehouse	4
1.3.1.4 User Interface	4
1.4 Architecting principles for Business Intelligence	5
1.4.1 Effective information delivery	5
1.4.2 Technology integration	6
1.5 Need for Business Intelligence	6
1.6 Objectives of BI systems	7
1.7 Business Intelligence applications	9
1.8 Advantages of Business Intelligence	9
1.9 Overview of Service Oriented Architecture	9
1.9.1 SOA Entities	10
1.9.2 Common characteristics of Service- Orientation	12
1.9.3 SOA Features	13
1.9.4 SOA Benefits	13
1.9.5 The Benefits of Moving to SOA	13

2. LITERATURE REVIEW	
2.1 Existing system	14
2.1.1 Consolidated Financial Reporting Module	14
2.1.2 Business Intelligence in Automobile Industry	15
2.1.3 Business Intelligence in Finance Institution	18
2.1.4 Business Intelligence in Reliance Mutual Fund	20
2.1.5 Business Intelligence in Chemical Industry	22
3. SYSTEM ANALYSIS	
3.1 Existing architecture for IT service management	23
3.2 Solution	24
4. SYSTEM SPECIFICATION	
4.1 Hardware Requirements	24
4.2 Software Requirements	24
4.3 Software Description	24
5. PROJECT DESCRIPTION	
5.1 Proposed system	28
5.1.1 Integrated reporting data store	28
5.1.2 SOA enabled BI component services	29
5.1.3 Analytical application solutions	29
5.2 Modules	30
5.2.1 Web service creation	30
5.2.2 Publishing a web service	30
5.2.3 Designing user interface	30
5.2.4 Service invocation	31
5.2.5 Analysis	31
5.3 Benefits of the proposed system	32
6. CONCLUSION AND FUTURE OUTLOOK	33
7. APPENDICES	34
Source code	34
Snap shots	41
8. REFERENCES	57



(Autonomous)
PERUNDURAI ERODE 638 052 TAMILNADU INDIA

School of Computer Technology and Applications

Department of Computer Technology - UG



CERTIFICATE

This is to certify that Mr./Ms. A. SWATHIKA of

KUMARAGURU COLLEGE OF TECHNOLOGY has Participated / Presented

paper entitled BUSINESS INTELLIGENCE SYSTEM USING SERVICE

ORIENTED ARCHITECTURE in the National Conference held from

0th to 11th March 2011 on Emerging Computing and Communication Technologies (NECCT 2011).


Organizing Secretary



HOD



Principal



ABSTRACT

Business intelligence is a management term used to describe applications and technologies which are used to gather, provide access to and analyze data and information about the organization, to help make better business decisions. Business intelligence technologies include traditional data warehousing technologies such as reporting, ad-hoc querying, online analytical processing (OLAP). BI systems combine data gathering, data storage, and knowledge management with analytical tools to present complex internal and competitive information to planners and decision makers. The emergence of the data warehouse as a repository, advances in data cleansing, increased capabilities of hardware and software, and the emergence of the web architecture all combine to create a richer business intelligence environment than was available previously. In this paper, a service-oriented architecture for business intelligence is described that makes possible a seamless integration of technologies into a coherent business intelligence environment, thus enabling simplified data delivery and low-latency analytics. The architecture highlights the importance of service oriented approach and the need to develop BI for its acquisition, integration, cleanup, search, analysis, and delivery. Service-oriented approach is compared with traditional BI architectures illustrating the advantages of the service oriented paradigm by architecting and implementing the framework.

ஆய்வச் சுருக்கம்

வணிக நுண்ணறிவு என்னும் பெயரால் அழைக்கப்படுகின்ற ஒரு அடிப்படைத் தரமான தாவ தாய்நகல் தாவ அணுகக்கூடிய மற்றும் தாவ ஆய்வுத் துறையின் பணிகளைச் செய்கின்ற முக்கிய நுட்பவகளை எடுக்க வழிவகை செய்கிறது. வணிக நுண்ணறிவு நுட்பவியல் என்பது மரபகடங்கு தொழில் நுட்பமான கருத்து ஆய்வு நுட்பவ விலைவாசி (நாடிமரபக பதத்தாய்வு நுட்பவ) துறையிலுள்ள உலகமுகியது. வணிக நுண்ணறிவு கருவிகள் சீக்கலான தகவல்களை எடுதலான தகவல்களை மரற்று செய் திட்டமிடுபவர் மற்றும் நுட்பவிலுடுபவர்களுக்கு வழங்குகிறது. தாவ கிடங்கின் தொன்றுதலான தாவ தாய்வுமரபககம் கணினி மென்பொருள் மற்றும் மென்பொருளின் திறனும் அதிகரித்துள்ளது. துதலுடன் துள்ளப தள கட்டமைப்பு துள்ளந்து உயர்சுலமான வணிக நுண்ணறிவு துள்ளபபி உயர்ந்துள்ளது.

திச்செயல்த் திட்டத்தில் வணிக நுண்ணறிவு சேவை அடிப்படையிலான கட்டமைப்பு துறைய துள்ளபவிலியில்லாத தொழில் நுட்பவியல் துறையமரபப்ட்டின் கருத்து சொல் ஒத்த வணிக நுண்ணறிவு துள்ளப விலைக்கம் படுகிறது. துதல்க் கட்டமைப்பு சேவை அடிப்படையிலான கட்டமைப்பின் முக்கியத்துவத்தை உம் தாவ மட்டில் துள்ளபமரபப்டி தாவ தாய்வுமரபககம் செயல ஆராய்தல் பதப்பாய்வு மற்றும் தாவ துள்ளபமரபககின் வளர்ச்சியின் துள்ளபகளை உம் விளக்குகிறது. சேவை அடிப்படையிலான கட்டமைப்பு மரபு வழிக கட்டமைப்பின் ஒய்யித் தகவல் துள்ளபகளை வரம்புபுத துறைய அமல்படுத்கம்படுகிறது.

LIST OF FIGURES

FIGURE NO	CAPTION	PAGE NO
1.1	Business Intelligence Architecture	3
1.2	Conceptual architecture for Business Intelligence	6
1.3	Role of BI in Decision Making	8
1.4	The “find-bind execute” paradigm	11
3.1	Legacy BI Portal System for IT service management	23
3.2	IT performance management system	24
4.1	Service Oriented Architecture for Business Intelligence	27

LIST OF ABBREVIATIONS

ABBREVIATION	EXPANSION
BI	Business Intelligence
SOA	Service Oriented Architecture
OLAP	Online Analytical Processing
ETL	Extract Transform and Load
MIS	Management Information System
DSS	Decision Support System
EIS	Enterprise Information System
JIT	Just In Time
API	Application Programming Interface
GUI	Graphical User Interface
CRM	Customer Relationship Management
XML	Extensible Markup Language
WS-*	Web Services
SOAP	Simple Object Access Protocol
RDS	Reporting Data Store
HTTP	Hyper Text Transfer Protocol

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF BUSINESS INTELLIGENCE

Business intelligence is a business management term used to describe applications and technologies which are used to gather, provide access to and analyze data and information about an enterprise, in order to help them make better informed business decisions [8].

Business intelligence is the technology and practice of applying information to make decisions. In this way, business intelligence is different than data warehousing, which is more about storing information. It is also more comprehensive than data mining. Information shows its real value when many people can use and share it. This is the goal of business intelligence.

For information to be this usable, it must be trusted, timely, relevant, easy-to-use, and in context. These are all necessary aspects of business intelligence technology. Different business intelligence tools address these factors in different ways. Reporting delivers regular, timely information, with the ability to author reports or queries to get specific details. OLAP analysis, with its multiple dimensions, allows you to compare and contrast information against time and other factors to uncover trends. Score carding presents key performance metrics and whether you've cross pre-determined thresholds. Executive dashboards put information in context, and in an easy-to-understand format.

All of these business intelligence tools should integrate to present a coordinated view of the organization. In this way, people using different tools share the same answers to the same questions, gain more value from information investment, and make better decisions.

For performance management expertise and business intelligence software, Cognos, an IBM company, is the source of vendor-neutral solutions that combine

reporting with enterprise risk management, planning, OLAP analysis, data integration, dashboards, and scorecards. Unrivaled performance management expertise goes into creating business intelligence tools that link data sources and protect investments in SAP, Oracle, and other vendors for superior performance.

1.2 THREE KINDS OF DECISIONS

In effect, there are three kinds of decisions that an organization makes.

- **Strategic decisions** – these are the “big” decisions that companies make (e.g. should we buy a partner, should we enter a new market). The value of these decisions is large – but the quantity is few.
- **Tactical decisions** – this has been where business intelligence traditionally has been implemented. This is the product manager deciding what discount schedule to put in place or a pricing decision for a new product.
- **Operational decisions** – these are the business decisions (often made by people that have never heard of “business intelligence”) that happen a lot on a daily basis – yet, they have smaller business impact when measured by themselves. However, in aggregate, multiple operational decisions add up to a lot of value – and can drive a better business.

1.2.1 BETTER BUSINESS PERFORMANCE

The ability to access, use and share data and information in an efficient and relevant way helps improve business performance. Business intelligence capabilities empower employees to:

- Align day-to-day operations with overall company strategy and objectives
- Identify and understand the relationship between business processes and their impact on performance
- Access information relevant to specific user roles and responsibilities
- Analyze data from documents and spreadsheets in an easy way
- Gain contextual insight into business drivers

Monitor the vital business indicators that are needed to move an organization forward such as:

- Current status and trend of essential financial ratios
- Effectiveness and profitability of sales channels
- Crucial operational metrics

In short, business intelligence helps companies gain a comprehensive and integrated view of their business and facilitate better and more effective decision-making.

1.3 ARCHITECTURE FOR BUSINESS INTELLIGENCE

1.3.1 Informed Decision-Making Based on Data

Business Intelligence (BI) enables better decision-making based on facts. It enables the organization to acquire a clear understanding of market behavior and commercial context.

There are common questions that most companies have:

- What are the best selling products?
- Where are the top sales dollars generated?
- Who are the bottom 5 employees based on profitability?

These types of questions can be answered by providing a way for business data to be structured in a way to enable the retrieval of related information.

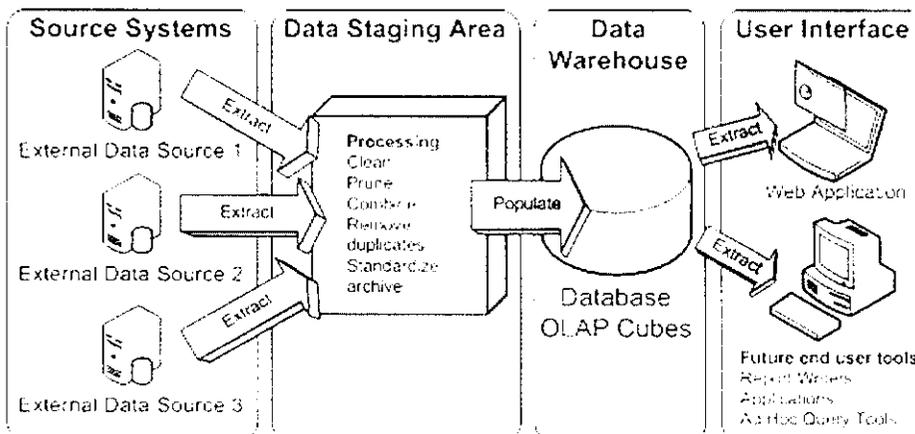


Figure 1.1 Business Intelligence Architecture

1.3.1.1 Source Systems

Most companies have distributed stores of data pertaining to the respective departments. Sales departments typically have metrics related to sales and performance. Inventory warehouses deal with their own set of metrics in relation to physical products or assets. These differing sources of data make up the business intelligence landscape for the company.

1.3.1.2 Data Staging Area

The different source data that comprise the company's business information need to eventually be stored in a data warehouse. There is a process called Extract Transform and Load (ETL) that facilitates means of formatting data from different sources so that they can be loaded into a data warehouse structure.

1.3.1.3 Data Warehouse

The varying forms of data get stored in a Data Warehouse. The Data Warehouse is the structure that facilitates the main work of business intelligence - reporting and analysis. Bits of data are structured in a way to facilitate flexible matrices that ultimately can be configured to display data in multiple dimensions by utilizing the concept of Online Analytical Processing (OLAP). This concept provides a configurable structure to analyze data.

1.3.1.4 User Interface

The business intelligence analysis can materialize in several different ways. Web-based applications can display information dashboards that display information. The data can also be accessed through reports generated by report writers. Custom applications can also display this data as well and any type of Ad-Hoc Query tools.

1.4 ARCHITECTING PRINCIPLES FOR BUSINESS INTELLIGENCE

Liya Wu¹, Gilad Barash¹, Claudio Bartolini describes the architecting principles for business intelligence [8] such as:

- To provide a “single version of the truth” across an entire organization.
- To provide a simplified system implementation, deployment and administration
- To deliver strategic, tactical and operational knowledge and actionable insight.

Because of the focus on information in business intelligence applications, the privileged point of view of the supporting architecture has to be the information view.

From this point of view, the most popular paradigms [3,4,5] are:

- The hub-and-spoke architecture with centralized data warehouse and dependant data marts
- The data-mart bus architecture with linked conformed dimensional data marts
- Independent non-integrated data marts

In today’s heterogeneous environments where many disparate systems and domains hold different parts of the necessary data, the most difficult challenges in achieving the above mentioned objectives are effective information delivery and technology integration.

1.4.1 Effective information delivery

BI systems need to deliver the right information to the right consumers at right time. Since the source information can potentially come from many different and nonintegrated sources, data has to be processed before it is effectively delivered to the end user. There are many data flows existing in any business intelligence system.

Figure 1.2 represents a typical layered view of architecture for business intelligence.

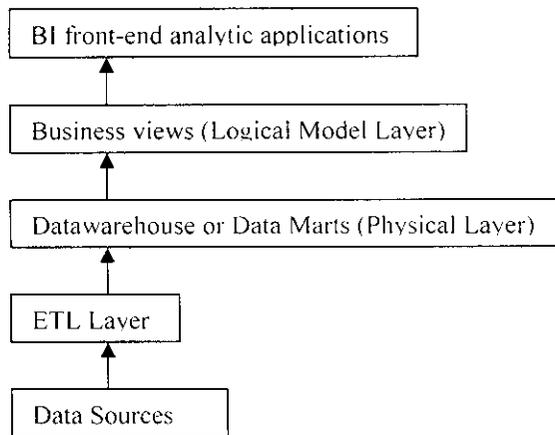


Figure 1.2 Conceptual architecture for Business Intelligence

1.4.2 Technology integration

In most deployed business intelligence environments, multiple “stovepipe” business intelligence systems – each with their tools, processes and data architectures – can be found across multiple business units and divisions of the enterprise. These non-integrated BI systems result in high component redundancy, inconsistent knowledge information, proprietary, non-open standard integration interfaces, and highly maintained point-to-point integration that ultimately increase the cost of development and prevent the achievement of a single version of truth across the organization. For business intelligence to deliver on its promises of real time, zero latency information delivery and closed loop processing, technologies and techniques have emerged or have been introduced. One such evolution is the transformation of traditional BI architectures into service oriented, component-based ones. Moving from a belief that Service Oriented Architecture technology has great potential for delivering enhanced BI, an approach to architecting BI systems using a service oriented approach is presented [8].

1.5 NEED FOR BUSINESS INTELLIGENCE

- **The amount of information is growing rapidly.** Eighty-one percent of respondents say the amount of information available with which to make decisions has "grown significantly" or "increased a lot."

- **Information overload.** Seventy-five percent report feeling overloaded with information, and 40 percent rated their degree of being overwhelmed at 4 on a 5 point scale.
- **The need for timely information is more pressing.** A third of respondents said access to up-to-date information within seconds or hours is critical to their companies.
- **Accessing unstructured information is difficult but increasingly necessary to decision making.** Fifty-five percent of the information dealt with in decision making is unstructured—for example, e-mails, documents, or images—yet two-thirds of respondents use mostly manual methods to search and access such data.
- **To manage the avalanches of information, business intelligence tools are becoming more widespread.** No longer the sole domain of analytical experts in headquarters, single departments or applications, business analytics are used by front-line workers, multiple departments and by users outside the organization.
- **Timely business intelligence has become mission critical to many enterprises.** Sixty-four percent of respondents said that if a business intelligence system was down one day or less, they expected a materially negative impact to business operations. Twenty-one percent said negative results would come from a downtime of one hours or less.

1.6 OBJECTIVES OF BI SYSTEMS

BI systems may be analyzed from different perspectives [6]. Decision makers and organizations should predominantly associate BI with organizational implementation of specific philosophy and methodology that would refer to working with information and knowledge, open communication, knowledge sharing along with the holistic and analytic approach to business processes in organizations. BI systems are assumed to be solutions that are responsible for transcription of data into information and knowledge and they also create some environment for effective decision making, strategic thinking and acting in organizations as shown in Figure 1.3. Value of BI for business is predominantly

expressed in the fact that such systems cast some light on information that may serve as the basis for carrying out fundamental changes in a particular enterprise, i.e. establishing new cooperation, acquiring new customers, creating new markets, offering products to customers

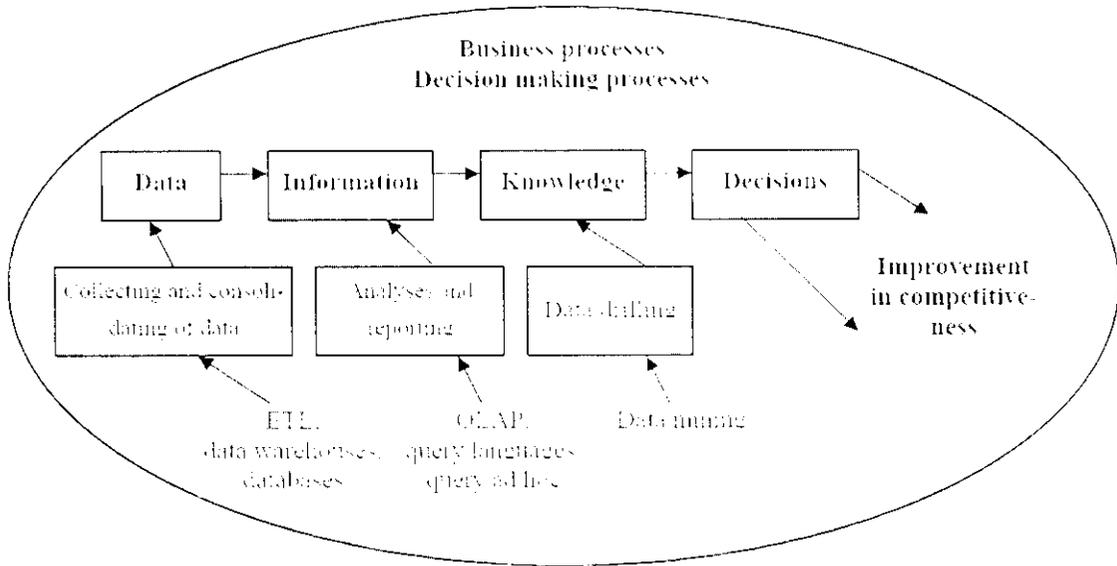


Figure 1.3 Role of BI in Decision Making

BI differs from MIS (i.e. DSS, EIS) in, first of all, their wider thematic range, multivariate analysis, semi-structured data originating from different sources and multidimensional data presentation.

It is assumed that BI may support decision making on all levels of management regardless of the level of their structuralisation. On the strategic level, BI makes it possible to set objectives precisely and to follow realization of such established objectives. BI allows for performing different comparative reports, e.g. on historical results, profitability of particular offers, effectiveness of distribution channels along with carrying out simulations of development or forecasting future results on the basis of some assumptions.

1.7 BUSINESS INTELLIGENCE APPLICATIONS

Business Intelligence is widely used in the following areas:

- Telecom Industry
- Banking
- Automobile
- Finance
- Insurance Companies

1.8 ADVANTAGES OF BUSINESS INTELLIGENCE

- Helps align the organization towards its key objectives
- Enables faster and fact-based decision making
- Combines multiple sources of data for decision making
- Efficient collection and distribution of vital data and statistics
- Accurate sensing of customers' needs and requirements
- JIT (Just in Time) responsiveness to customers' needs and demands
- Ability to adopt and adapt to any market changes or shifts, including economic recessions
- Streamlined business processes which results to a cost-effective business operation
- Accurate business and sales forecasting

1.9 OVERVIEW OF SERVICE ORIENTED ARCHITECTURE

SOA is an application architecture, in which application components or "services" are well defined using common interfaces, utilize a contract to define how services will be invoked, and interact in a loosely coupled manner. In this paradigm, the terms "client" and "server" are purely situational. At one moment, an application could act as a client by calling an external service, while moments later, it may act as a service-provider when called by another application to perform a task.

Using SOA, businesses will be able to generate new services in a flexible and agile way by combining existing logic and exposing it via reusable services. Service Oriented Architecture is an approach that considers all business systems, applications and functions as resources

- A "service oriented architecture" is not one component or technology, it is an approach to designing an enterprise architecture consisting of multiple infrastructure and application components
- It does not mandate building everything from scratch
- It does not mean that it will be more expensive to implement.
- Development costs will be lower over time due to enforcement of a new level of reuse (process, interfaces and components)
- It increases flexibility due to loosely coupling of resources (services), allowing plug and play of underlying technologies and individual components (with multiple vendor and best of breed solutions)
- It supports faster time to market

These resources can be:

- A function that needs to be used by more than one system
- An entire application
- A particular product service (e.g. a scoring service)
- A particular common utility (e.g. assembling a document, printing a document)
- An external web application (e.g. a credit check transaction to the credit bureau)
- A host transaction (e.g. IMS) via web services

1.9.1 SOA Entities

Potts, M 2003, tells about SOA Entities. The "find, bind, and execute" [10] paradigm as shown in Figure 1.4 allows the consumer of a service to ask a third-party registry for the service that matches its criteria. If the registry has such a service, it gives

the consumer a contract and an endpoint address for the service. SOA consists of the following six entities configured together to support the find, bind, and execute paradigm.

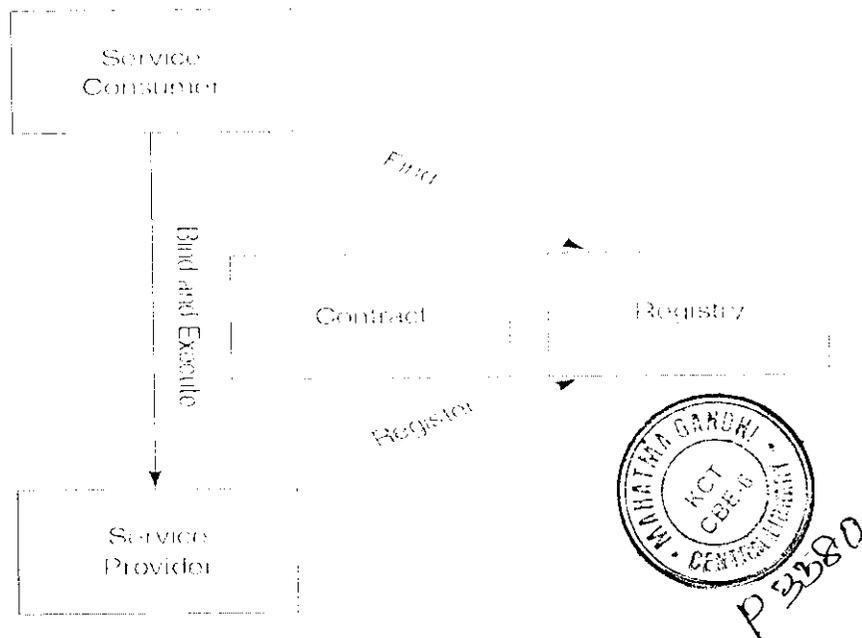


Figure 1.4 The “find-bind execute” paradigm

Service Consumer

The service consumer is an application, service, or some other type of software module that requires a service. It is the entity that initiates the locating of the service in the registry, binding to the service over a transport, and executing the service function. The service consumer executes the service by sending it a request formatted according to the contract.

Service Provider

The service provider is the service, the network-addressable entity that accepts and executes requests from consumers. It can be a mainframe system, a component, or some other type of software system that executes the service request. The service provider publishes its contract in the registry for access by service consumers.

Service Registry

A service registry is a network-based directory that contains available services. It is an entity that accepts and stores contracts from service providers and provides those contracts to interested service consumers.

Service Contract

A contract is a specification of the way a consumer of a service will interact with the provider of the service. It specifies the format of the request and response from the service. A service contract may require a set of preconditions and postconditions. The preconditions and postconditions specify the state that the service must be in to execute a particular function. The contract may also specify quality of service (QoS) levels. QoS levels are specifications for the nonfunctional aspects of the service. For instance, a quality of service attribute is the amount of time it takes to execute a service method.

Service Proxy

The service provider supplies a service proxy to the service consumer [10]. The service consumer executes the request by calling an API function on the proxy. The service proxy, shown in Figure 1.4, finds a contract and a reference to the service provider in the registry. It then formats the request message and executes the request on behalf of the consumer.

The service proxy is a convenience entity for the service consumer. It is not required; the service consumer developer could write the necessary software for accessing the service directly. The service proxy can enhance performance by caching remote references and data. When a proxy caches a remote reference, subsequent service calls will not require additional registry calls. By storing service contracts locally, the consumer reduces the number of network hops required to execute the service.

1.9.2 Common characteristics of service-orientation

The following are the characteristics of service orientation [12]:

- Services are loosely coupled
- Services share a formal contract
- Services abstract underlying logic
- Services are composable
- Services are reusable
- Services are autonomous

- Services are stateless
- Services are discoverable

1.9.3 SOA Features

Competitive requirements for greater business agility, flexibility to respond to change, and reusability to cut costs and increase efficiency are driving businesses to SOA. SOA holds the promise of a standards-based application infrastructure that encourages the development of modular, reusable components that can support end-to-end business processes or composite applications, solve problems created by legacy applications that are hard to integrate.

1.9.4 SOA Benefits

Sefan Linders [12] discusses the benefits of SOA:

- No need to rip and replace existing systems: SOA allows the reuse of all types of existing applications (both packaged and custom built) with a new GUI
- SOA and Web services standards help to ease the pain of integration
- SOA can help businesses respond more quickly and cost-effectively to the changing market conditions: Increased agility and flexibility
- Possibility to source business processes (insourcing / outsourcing)

The greatest benefit of SOA is its infinite flexibility. The greatest drawback of SOA is also its infinite flexibility.

1.9.5 The Benefits of Moving to SOA

- Reuse legacy applications and data in new services, renewing their value
- Access these assets in real time to support business intelligence and customer service initiatives
- Flexibility and interoperability with internal and external services
- Control budget through isolation and incremental changes

CHAPTER 2

LITERATURE REVIEW

2.1. EXISTING SYSTEM

2.1.1 Consolidated Financial Reporting Module

The Challenge:

The existing reporting system was a competitive liability. Anchor wanted a high volume, scalable, visually stunning and flexible reporting system that allowed users across the region to easily choose how, when, and where to view reports and in what format. The major challenge for them was to identify errors and omissions from the data coming from over 30 disconnected locations. the solution helped to identify the same and speed.

Solution:

- MAIA solution connected to the spreadsheet coming from each location containing Trial Balance reports into MS SQL.
- The MAIA Intelligence professional services team did a great job to build new reporting system, databases & data-feed process.
- The solution provided a platform to make consolidated as well as location wise periodical Schedule VI Balance Sheet at a great speed and ease.
- It provided reports to compare, analyze, review and comments on various reports and details of the same.

Benefits of MAIA solution:

- Anchor users like the fact that reports arrive quickly, they look good; have analytical and detailing facility and the whole reporting process is so much easier
- Users can select the report format that best suits their needs. Eventually, this will be the de facto standard
- With this new build and configurable options within the report, Anchor can make changes and turn reports around very quickly

- But more than anything, it's the scope of growing out all these features and functionality that really distinguishes this reporting system
- MAIA solutions helped to speed up the submission of quarterly & periodical consolidated Schedule VI Balance Sheet reports to the management: The solution consolidated financial reporting and analysis thus providing the maximum ROI.

2.1.2 Business Intelligence in Automobile Industry

Problem Overview:

Information is said to be the most important resource of managers and decision makers. Nowadays the difficulty is not in getting the information but rather how to select the relevant information from the abundance of data and information. 1KEY BI gives single source for critical information throughout the organization so can spend less time debating data and more energy making critical decisions [9].

Challenge:

Business users, key personnel, Managers and top management at CEAT (Automobile Industry) wanted information visibility on web. Territory Managers were not able to see entire report in a single sheet. Before going to a dealer, they were doing lot of manual homework. They were not getting correct data while discussion with dealer. Adhering to continuous report requests from business users was becoming a challenge as generation of SAP reports required lot of efforts & time.

Why 1KEY BI?

CEAT selected 1KEY BI because it provided with far more than just trending and reporting. It gave a higher level of business intelligence (BI) capabilities that:

- Integrated easily with the existing production and control systems data from SAP & non SAP data
- Provided deep insight into batch processes

- Facilitated information-sharing among production and supervisory staff so they could better monitor production in their appropriate contexts
- Cost-effective with low TCO
- Dashboard feature

Implementation Plan:

- Develop reports for 1 department at a time so that all kind of reports required can be provided. In the process, the quality and performance is not degraded anywhere.
- SAP data was pulled using SQL Server's SSIS and Microsoft BizTalk and reports created on IKEY.

Users of IKEY BI

- Around 300 users
- 200 Territory Managers
- Top Management
- Business Analysts
- 20 QC / R&D Professionals

Partial List of Reports

Around 100 to 150 real-time reports are being used by CEAT to monitor the business numbers, track productivity, schedule resources based on volumes and manpower rooster from IKEY BI.

- Territory Dashboards
- Treasury Reports
- Accounts Receivable / Payable
- CEO Reports/Dash Boards
- Key Performance Indicators

Sales: New business monitoring across all the products giving snapshots of daily and MTD new business logged in - used by management team to track business across products; this helps them channelize resources if required in certain directions

CRM: Dealers / Distributors Performance based on product sales

Finance: Snapshots of income and expense Snapshots of renewal and revival business

Productivity: IKEY BI allowed CEAT to drill down and make phase changes that could shorten raw material addition times, improve line speeds, reduce the time in quality control, shorten overall cycle and changeover times

Productivity: Real-time reports for departments like Production, Finance, Sales & Distribution and Marketing

Benefits:

IKEY has now become the primary source of information at CEAT.

- Unlock information hidden within data
- Discover trends and proactively address them
- Identify performing and non-performing products, market segments / producers
- Link measurements to strategy and drive desired results
- Free up IT resources by empowering business users
- Improve current service model
- Track key business metrics to improve operational performance
- Single view of SAP & non SAP data for top management
- Easy visibility of information across organization viz. production, finance, sales & distribution and marketing
- More Meaningful Reports & Dash Boards which enable quick business decisions
- Business Users are delighted to have the multi-dimensional reports with graphical interface and visually appealing charts

2.1.3 Business Intelligence in Finance Institution

Challenge

Exim Bank finances, facilitates and promotes India's international trade and investment, and coordinates working of institutions engaged in financing exports and imports. Bank has well-structured business processes [13]. It is officer-oriented and adopts a multidisciplinary approach involving close coordination amongst specialists drawn from several fields such as Bankers, Business School Graduates, Economists, Chartered Accountants, Engineers, Law, Human Resources and Information Technology. Decision making process is consultative across Groups / Levels. Emphasis was on creating and enhancing international competitiveness.

Bank's organizational structure is designed to reduce procedural rigidity while stressing interdependence and co-ordination among various Groups through ease of information flow, and to create an enabling environment so as to facilitate innovation as a part of continuous review of technology and its applications to the Bank. has been proposed to implement Business Intelligence Solution in the Bank.

Technological aspects required in BI solution

- Multi-tier architecture with each tier fully independent – An N-Tier architecture with Thin Client leads to the entire Administration being managed by the experts. Also with this architecture, the tasks are layered wherein different Application and Business Logic can be defined.
- Web Desktop for Browser based access– This helps as the end-user need not be aware of the happenings in the Server side of the product. Every Client will need only the Internet Browser to access the Server.
- Web based administration of the BI software - This leads to monitoring the BI software from any location where Internet Browser is present.
- HTTPS/SSL Support – Secure Client Access to the software.
- Support for Multiple platforms, Database, Web servers is a feature of BI for compatibility with various available architectures and solutions.

- Support for J2EE and XML based Application Server – For future integration with ERP and other Applications of an Organization.
- Support for multiple Web Servers, Application Servers and database servers for Fault Tolerance & Load Balancing.

Methodology:

- It contained a well-defined framework and rich content that guides projects step-by-step along business, data, and technical and metadata paths of development.
- Using this mature and proven methodology reduced risk, minimized surprises, and maximized business value by consistently delivering high quality results.
- A thorough, well-documented and organized set of procedures that moved the project teams successfully through an entire system development lifecycle has been created.
- 1KEY methodology for BI Implementation met a wide variety of needs by incorporating the most desired BI characteristics.

Benefits:

- The bank has large team of executives and a line of senior managers, and their expectations for analysis always needs a deeper dive in the data-set and report design.
- The users were able to get a better experience of the same, and are able to analyze the data as per their expectations.
- User have happily provided the sign-off and are able to save a lot of time on reporting tasks, and able to invest more time and energies on analysis, which is their core strength.

Partial List of Reports:

- Integrated Credit Exposure Individual and group exposure across the EXIM
- Sanction details across various dimensions
- Disbursement details across various dimensions (rep. office wise, group wise, industry wise, country wise, currency wise etc.)

- Repayment, Outstanding, Overdue details across various dimensions
- List of top n loans and advances
- Total credit exposure
- List of new credit sanction/ disbursed / client / product under each scheme for a time frame. (No. of A/C s & Sanctioned amounts)
- Age wise position of proposals overdue for review/ renewal
- Details of bills purchased / discounted/ rediscounted during the year
 - As on from date
 - During the period
 - Renewals
 - Balance overdue as on to date

Regulatory reports:

- Statement of Liabilities & Assets of the FI as on March 31
- Statement showing the profile of Investment as on March 31
- Earning Appraisal

2.1.4 Business Intelligence in Reliance Mutual Fund

Challenge:

Reliance Mutual Fund is doing research at big level for more than 500 companies. There are enormous amount of data being generated; Analyzing that data and developing different reports for different people from the same large data was becoming a challenge. Fund managers at RMF were used to twisting and turning their queries [15]. The IT systems had to handle their requests. They were not getting enough of their analytic tools. Most of RMF's systems would generate static reports from within the application with which they couldn't play around. Though the CRM solution and the transaction and portfolio management solutions had reporting capabilities, they would often fall short when it came to providing detailed reporting.

Solution:

- RMF was looking out for a BI tool, which was easy to learn, easy to use and deploy enterprise-wide.
- RMF found IKEY having interface similar to Excel yet powerful enough to deliver the on-the-fly report requirements of the business users.
- RMF chose IKEY as the front-end tool for processing the reports.
- The fact that the deployment was to take only a couple of days went in the application's favor. The entire roll out got completed within a month's time and 1-day each for technical training and functional training for fund managers & research team.

Approach:

- Multiple database access being made available to end-users through IKEY BI. The users were identified by the BI team at RMF based on their requirements and mapping it to IKEY
- Consolidation of information across the database
- User Management through Active Directory, which is supported by IKEY
- Deployed IKEY BI on Citrix server for faster deployment and instant access to reports also through Web
- IKEY was made a standard interface for all reporting & analysis requirements. This saved time & trouble required to build custom reports.

Benefits

- Different types of reports for different people in equity research.
- The users now get different output from the same data.
- From the same report users can now generate different outputs. Everybody can change the output and generate different reports on their own with the same data.

End Users:

- The Fund managers
- Marketing teams

- Operations Team
- Fund Managers, Risk Team, Research, Regional teams

2.1.5 Business Intelligence in Chemical Industry

Challenge

Company of larger size requires actionable information & MIS at various levels in the organization [14]. Top management team requires information which is quick, analyzed already and based on which action can be taken related to customers, employees, production programs, global supply chains. At middle management level, they require information related to tactical decisions & operational matters. At grass root level, field people, sales officers require information related to customer interfaces.

All this information which comes from multiple sources such as SAP, legacy systems, customer contact or care system, lotus note based workflow system, they need to be integrated.

Solution

- Considering the need of immediate information requirement & the Dashboards which are required by the top management for taking quick decisions, various tools are selected.
- The MAIA team evaluated IKEY BI tool very extensively in terms of whether it will suit needs of the top as well as middle management. The conclusion is based upon the capabilities of the product, based upon organizational needs, based upon the speed for Dashboards & key analytics to be deployed in the organization, IKEY is a suitable one.
- IKEY was able to display, 5 dashboards based upon sales, dispatches, safety, production and collection.
- IKEY BI was roll out in two separate areas: 1) IKEY Dashboard for excellence model - where large number of KPIs are required which are monitored 2) IKEY BI for extracting information from SAP & BIW and generate Dashboard which are for the top management.

CHAPTER 3

SYSTEM ANALYSIS

3.1 Existing architecture for IT service management

In this section, the legacy solution and the benefits of re-architecting it into following service oriented principles is described.

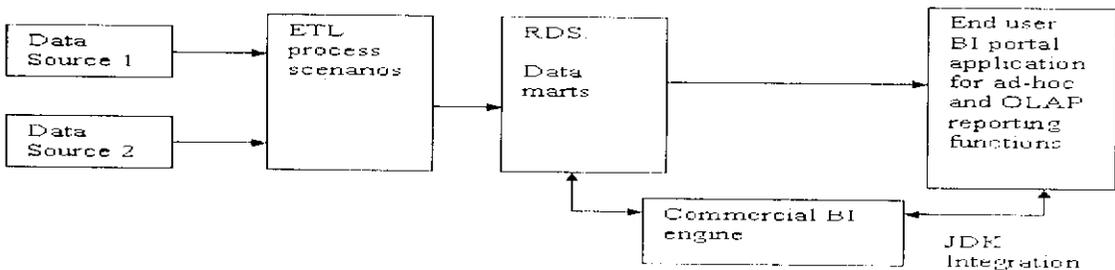


Figure 3.1 Legacy BI Portal System for IT service management

The shortcoming of the legacy systems are:

- Independent non-integrated data marts for separate data sources
- There is no default solution to provide actionable insight across multiple domains from service management and asset management
- Inflexible, non-open, proprietary ETL processes which will be hard to be reused as sharable, loosely-coupled, service oriented pluggable ETL components
- Front-end functions are implemented with tight integration with one specific BI tool. It will be hard to provide agnostic BI portal to be BI tool neutral
- There is no simple closed-loop process workflow to provide the actionable insight back to the source systems. On re-architecting the business intelligence systems using service-oriented architecting principles help overcome these limitations, as was argued in [1,2,16].

3.2 SOLUTION

The first step in re-architecting our legacy system is to break down the legacy components into service-oriented reusable components able to communicate through open standard messaging protocols, based on XML, WS-* and SOAP. The resulting service-oriented architecture of our IT performance management system (SOA-ITPA) is described by figure 4.

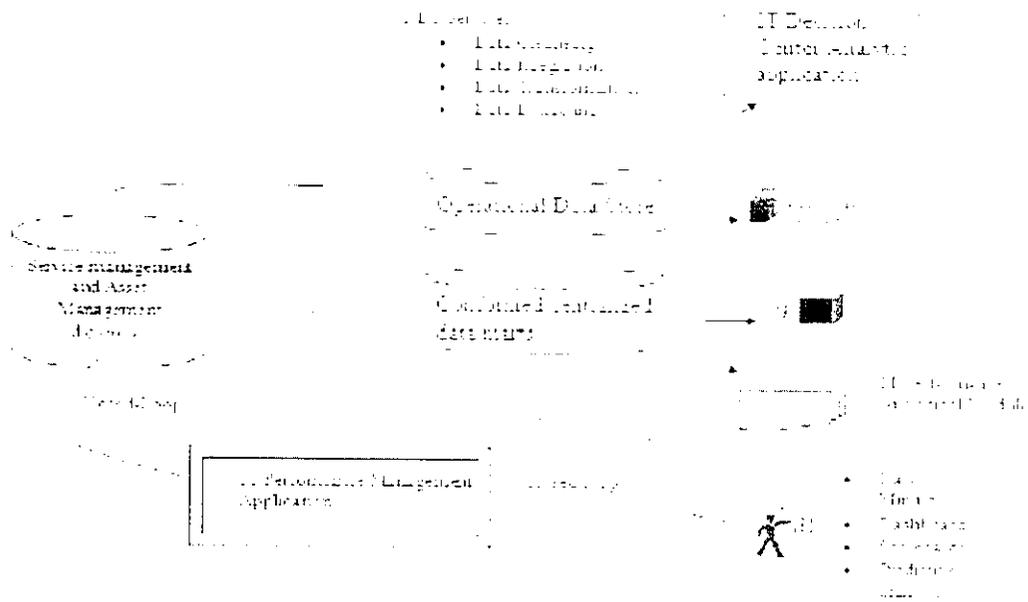


Figure 3.2 IT performance management system

The key benefits our SOA-ITPA architecture are:

- Integrated and consistent “single version of truth” data architecture
- Scalable and flexible ETL processes
- Reusable and extensible services, providing acceptable return on investment
- Actionable insight BI solutions to send BI analytical results to users and help them to understand the information so the appropriate actions can be taken in BI real time environment.

CHAPTER 4

SYSTEM SPECIFICATION

4.1 HARDWARE REQUIREMENTS

Processor	:	Pentium IV
Speed	:	Above 500 MHz
RAM capacity	:	2 GB
Hard disk drive	:	80 GB
Key Board	:	Samsung 108 keys
Mouse	:	Logitech Optical Mouse
Printer	:	DeskJet HP
Motherboard	:	Intel
Cabinet	:	ATX
Monitor	:	17" Samsung

4.2 SOFTWARE REQUIREMENTS

Operating System	:	Windows XP and above
Language	:	Visual Studio 2005
Back End	:	MS Access

4.3 SOFTWARE DESCRIPTION

Microsoft Visual Studio 2005

Product Features:

- Complete development environment with comprehensive application development features, including improved visual designers, code editors, and programming languages for optimum productivity.

- Develop and debug multi-tier server applications from with the unified development environment - even remotely.
- Build tools that extend the Visual Studio Integrated Development environment using the Visual Studio Software Developers Kit.

Microsoft Visual Studio Professional 2005 is one of the most significant developer tools available for building high-performance, multi-tier applications for Windows, the Web, and mobile devices. Designed for developers working alone or in small teams, Visual Studio Pro 2005 features integrated visual database tools for designing databases, tables, and stored procedures. You can also design, debug, and deploy multi-tier applications; and build a wide array of high-performance solutions faster than ever. With Visual Studio Pro 2005's powerful, enterprise-class application platform, you can also easily create and application platform, you can also easily create and deploy client applications, and automatically publish and maintain applications and their dependencies with integrated Click Once support. With more than 50 new controls and hundreds of built-in services for site security, personalization and look and feel, Visual Studio Pro 2005 gives you the tools it takes to build fast, interactive Web applications. Enhanced visual designers and editors speed up development times significantly, while intuitive visual designers help to improve XML editing and XSLT debugging. In short, you can tackle just about any development task, whether its creating dynamic, data-driven applications using an integrated data access, design, and reporting environment, or creating robust applications using the Microsoft .NET Framework 2.0, the .NET Compact Framework 2.0, and native code -- all supported by Microsoft Visual Studio 2005. You can also easily develop for 64-bit systems using a 64-bit version of the .NET Framework.

Visual Studio 2005 Professional Edition includes Microsoft Visual Basic, Microsoft Visual C#, Microsoft Visual C++, and Microsoft Visual J# programming languages; tools for building Windows and Web solutions; SmartPhone and Pocket PC development tools for building Windows CE-based applications; Visual Database tools; advanced debugging tools, including cross-machine debugging; and much more.

Cross Machine Debugging and Better Web Publishing

Visual Studio 2005, combined with ASP.NET 2.0, offers an improved environment for publishing and administering a Web site. With the built-in publishing features, such as integrated File Transfer Protocol (FTP), you can keep files in local project and synchronize them with files located on a remote Web server. Visual Studio automatically logs copied files which help you to identify whether the remote site has been updated with the latest files. And to help protect site and IP, a Web site can be pre-compiled to publish only executable code, no source files, to the production server. The Web site configuration interface makes it easy to set up and maintain applications, and the new Web Site Administration Tool helps you manage remote sites. You can manage cache settings at the server and Web site level to improve the performance of published Web sites. And users can manage their own sites with the new ASP.NET Microsoft Management Console (MMC) plug-in.

The Solution Explorer in Visual Studio 2005 also makes it easy for you to clean up unwanted files. Simply choose Clean Solution from the Build Menu and all the intermediate files and output directories in whole solution will be removed. For Visual C++, you can also use the Project Only submenu of the Build Menu to clean only the project currently selected in Solution Explorer, without cleaning any project dependencies or solution files. Other features include a new checksum feature that matches multiple source file names without any confusion; provider-driven application services that let you extend the powerful features of Visual Studio Team System by adding own ASP .NET application services seamlessly, giving you complete control over the tools and environment.

4.4 DATABASE TECHNOLOGY USED

- **Microsoft Access** - For Business Intelligence system, banking database is used.
- **XML** – It helps for displaying the output

CHAPTER 5

PROJECT DESCRIPTION

The first step in re-architecting legacy system is to break down the legacy components into service-oriented reusable components able to communicate through open standard messaging protocols, based on XML, WS-* and SOAP.

5.1 PROPOSED SYSTEM

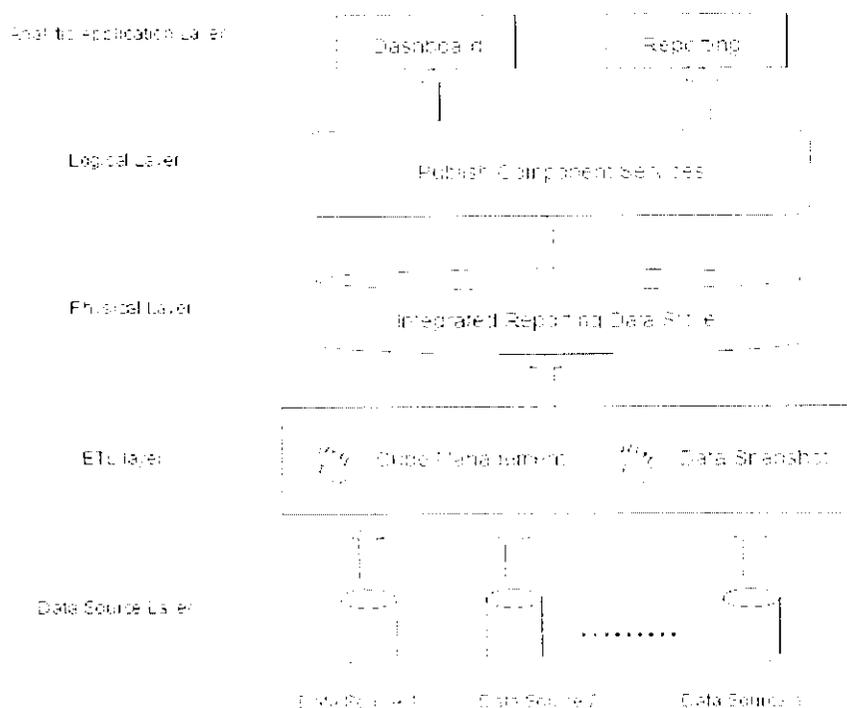


Figure 5.1 Service Oriented Architecture for Business Intelligence

5.1.1 Integrated reporting data store

The essential part of architecture is the integrated centralized integrated reporting data store (RDS). RDS consists of historic, current and predictive data. A number of component services are built to populate data from sources into RDS. To achieve collaboration and closedloop processing, a real-time subscriber service needs to be built

to allow real-time synchronization. Other derived services, (snapshot service, transmit service and extract service) are used to ensure consistency of information providing a deeper insight in the organization performance. The persistent RDS cannot generally store all the (cleansed) data to be shared across the various components, including the ones supporting auditing and tracking activities. Trade-offs needs to be made, depending on how much real-time data is required by the applications, and how complicate are the ETL transformations. However, the principle is to have full, rich, standardized, well-tested and certified ETL services to use for different projects and products.

5.1.2 SOA enabled BI component services

The tightly-coupled legacy ETL system - with dedicated ETL procedures for each component - is not reusable and maintainable. The SOA version has been proposed breaks down the ETL process into generic smaller servicing modules. The design is parameter-driven, and XML metadata are driven around to provide specific functions within ETL service layers. The benefits of this design will be felt more in full when the system grows and becomes more complex, and to meet additional integrated applications' data requirements. The reusability of these services also provides more flexible and scalable ETL process. The implementation of BI analytical modules for dashboard and reporting, performance management guided analysis application or other decision making applications is made easier and quicker by the use of the publisher-subscriber communication paradigm. The RDS contents are published and made accessible to other products and components through the standard web services interfaces and protocols, to make RDS fully functioning in any SOA based organization environments.

5.1.3 Analytical application solutions

In the SOA architecture, the data marts are engineered to meet different applications requirements. They can be in different formats: XML, micro-cubes or Excel files. This also enforces that any data mart implemented as part of the IT performance management system will source its data from the federated, conformed and centralized RDS, tailored to provide the specific IT performance management solutions with data quality standards.

The main purpose of a conformed data mart can be defined as follows:

- To store pre-aggregated, multi-dimensional information
- To control end user access to information
- To provide fast access to information for specific analytical needs or user group
- To represent the end users view and data interface of the data warehouse
- To create the multidimensional/relational view of the data

5.2 MODULES

5.2.1 Web Service creation

- A web service is a software application that works over the internet.
- A web service resides on the web server and is available for use over the internet.
- A request by the client application constitutes construction and sending a SOAP request using HTTP to the web server.
- The web server hosts the class and its methods of a web service, for a client computer to request and use.
- Any client computer can request and use the class and its methods of the web service.

5.2.2 Publishing a Web Service

- After creating a web service, it has to be uploaded it on to a web host. The web host will make available the Web service, to anyone over the internet.
- In the .Net world, web server is the Internet Information Server.
- To work over internet, a web service software application includes a Web API which facilitates other software application to communicate with it.

5.2.3 Designing User Interface

- The web API lets applications like web applications, windows applications, mobile applications communicate with each other using XML, SOAP.

- By adding a web service to an application, the features of the application can be enhanced.
- A web service can be used in windows applications, web applications, web sites and mobile applications.

5.2.4 Service Invocation

- Whenever the client application needs to invoke the Web Service, it will really call the client stub. The client stub will turn this 'local invocation' into a proper SOAP request.
- The SOAP request is sent over a network using the HTTP protocol. The server receives the SOAP requests and hands it to the server stub. The server stub will convert the SOAP request into something the service implementation can understand
- Once the SOAP request has been deserialized, the server stub invokes the service implementation, which then carries out the work it has been asked to do.
- The result of the requested operation is handed to the server stub, which will turn it into a SOAP response.
- The SOAP response is sent over a network using the HTTP protocol. The client stub receives the SOAP response and turns it into something the client application can understand.
- Finally the application receives the result of the Web Service invocation and uses it

5.2.5 Analysis

- The service checks for the database and helps the user in decision making process.

Steps for Analysis:

- i. The service checks for the existing customer
- ii. If the customer is already having loan in one bank and if he applies for another loan in another bank, then the service checks for loan eligibility.
- iii. In the banking database, the analysis is made in such a way that if 80% of the loan is paid, then another loan can be sanctioned to the customer.

- iv. If 80% is not paid, then the customer is not eligible for another loan.
- v. The decision is made based on the eligibility criteria with respect to the total amount paid till date.

5.3 BENEFITS OF THE PROPOSED SYSTEM

- Integrated and consistent “single version of truth” data architecture
- Scalable and flexible ETL processes
- Reusable and extensible services, providing acceptable return on investment
- Actionable insight BI solutions to send BI analytical results to users and help them to understand the information so the appropriate actions can be taken in BI real time environment.

CHAPTER 6

CONCLUSION AND FUTURE OUTLOOK

This project diminishes the drawbacks of the existing business intelligence system and improves simplified data delivery. In particular, with respect to the legacy application, the SOA approach yielded significant gains in simplicity and cost. SOA enables reuse of common business processes for greater efficiency. SOA approach designed for Business Intelligence system helps in decision making process. For simpler BI systems, it could be possible to build viable solutions with the traditional approach. But the SOA approach appears to be the best way to reduce the total development and maintenance cost, and to minimize the risk and impact across an entire enterprise when introducing business intelligence solutions.

In future, security can be added in service invocation module in order to minimize the risk involved.

CHAPTER 7

APPENDIX I

SOURCE CODE

A) Login Form:

```

Partial Class Login
    Inherits System.Web.UI.Page

    Protected Sub Button1_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Dim cusobj As New IndianBank
        Dim sql As String
        sql = "select * from admin where adminname='" & TextBox1.Text & "' and
adminpassword='" & TextBox2.Text & "'"
        If cusobj.Login(sql) = True Then
            Response.Redirect("report.aspx")
            MsgBox("Invalid")
        End If
    End Sub
End Class

```

B) Loan Analysis:

```

Imports System.Web
Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.Data
Imports System.Data.Odbc
<WebService(Namespace:="http://tempuri.org")> _
<WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1)> _
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()> _
Public Class IndianBank
    Inherits System.Web.Services.WebService
    <WebMethod()> _
    Public Function Login(ByVal sql As String) As String
        Dim con As New OdbcConnection("Dsn=bank")
        con.Open()
        Dim f As Boolean = False
        Dim cmd As New OdbcCommand(sql, con)
        Dim dr As OdbcDataReader
        dr = cmd.ExecuteReader()
        While dr.Read
            f = True

```

```

End While
con.Close()
If f = True Then
    Return True
Else
    Return False
End If
End Function
<WebMethod()> _
Public Function AvailableService() As String
    Dim con As New OdbcConnection("Dsn=bank")
    Dim query As String
    Try
        con.Open()
        query = "select * from customer"
        con.Open()
        Dim cmd1 As New OdbcCommand(query, con)
        Dim dr1 As OdbcDataReader()
        con.Close()
        Return True
    Catch ex As Exception
        Return False
    End Try
End Function
<WebMethod()> _
Public Function LoanEligibility(ByVal str As String) As String
    Dim con As New OdbcConnection("Dsn=bank")
    Dim query As String
    Dim flg As New ArrayList
    Dim flg1 As New ArrayList
    Dim i As Integer = 0

    con.Open()
    query = "select * from CustomerLoan where CustomerID='" & str & "'"
    Dim cmd1 As New OdbcCommand(query, con)
    Dim dr1 As OdbcDataReader
    dr1 = cmd1.ExecuteReader()
    While dr1.Read
        flg.Add(dr1(3))
    End While
    query = "select sum(payamt)+sum(InterestAmt) from CustomerLoanPayment where
CustomerID='" & str & "'"
    Dim cmd2 As New OdbcCommand(query, con)
    Dim dr2 As OdbcDataReader
    dr2 = cmd2.ExecuteReader()
    While dr2.Read
        flg1.Add(dr2(0))
    End While
    con.Close()
    Dim str1 As Long
    Dim str2 As Long

```

```
Dim stre As String = "Not Eligible"
```

```
For Each f As String In flg
```

```
    str1 = f
```

```
Next
```

```
For Each fl As String In flg1
```

```
    str2 = fl
```

```
Next
```

```
If str2 >= (str1 * 80) / 100 Then
```

```
    stre = "Eligible"
```

```
End If
```

```
Return stre
```

```
End Function
```

```
End Class
```

C) Service Invocation:

```
Imports System.Data.Odbc
```

```
Imports System.Data
```

```
Imports System
```

```
Imports System.Net
```

```
Imports System.IO
```

```
Imports System.Collections.Generic
```

```
Imports System.ComponentModel
```

```
Imports System.CodeDom
```

```
Imports System.CodeDom.Compiler
```

```
Imports System.Web.Services
```

```
Imports System.Web.Services.Description
```

```
Imports System.Xml.Serialization
```

```
Imports System.Reflection
```

```
Partial Class Report
```

```
    Inherits System.Web.UI.Page
```

```
    Private MethodInfo() As MethodInfo
```

```
    Private service As Type
```

```
    Private MethodName As String = ""
```

```
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles  
Button1.Click
```

```
    DynamicInvocation("http://localhost:3259/webser/Service.asmx?WSDL")
```

```
    DynamicInvocation("http://localhost:3250/webCUB/Service.asmx?WSDL")
```

```
End Sub
```

```
Private Sub DynamicInvocation(ByVal str)
```

```
    Try
```

```

Dim messageTextBox As String
Dim Label1 As TextBox
Dim uri As New Uri(str)
Dim webRequest__1 As WebRequest = WebRequest.Create(uri)
Dim requestStream As System.IO.Stream =
webRequest__1.GetResponse().GetResponseStream()
Dim sd As ServiceDescription = ServiceDescription.Read(requestStream)
Dim sdName As String = sd.Services(0).Name
Response.Write(sdName)
Dim servImport As New ServiceDescriptionImporter()
servImport.AddServiceDescription(sd, [String].Empty, [String].Empty)
servImport.ProtocolName = "Soap"
servImport.CodeGenerationOptions = CodeGenerationOptions.GenerateProperties
Dim [nameSpace] As New CodeNamespace()
Dim codeCompileUnit As New CodeCompileUnit()
codeCompileUnit.Namespaces.Add([nameSpace])
Dim warnings As ServiceDescriptionImportWarnings = servImport.Import([nameSpace],
codeCompileUnit)

If warnings = 0 Then
    Dim stringWriter As New
StringWriter(System.Globalization.CultureInfo.CurrentCulture)
    Dim prov As New Microsoft.CSharp.CSharpCodeProvider()
    prov.GenerateCodeFromNamespace([nameSpace], stringWriter, New
CodeGeneratorOptions())
    Dim assemblyReferences As String() = New String(1) {"System.Web.Services.dll",
"System.Xml.dll"}
    Dim param As New CompilerParameters(assemblyReferences)
    param.GenerateExecutable = False
    param.GenerateInMemory = True
    param.TreatWarningsAsErrors = False
    param.WarningLevel = 4
    Dim results As New CompilerResults(New TempFileCollection())
    results = prov.CompileAssemblyFromDom(param, codeCompileUnit)
    Dim assembly As Assembly = results.CompiledAssembly
    service = assembly.[GetType](sdName)

    MethodInfo = service.GetMethods()

    For Each t As MethodInfo In MethodInfo
        Response.Write(t.Name)
        If t.Name = "LoanEligibility" Then
            Dim obj As [Object] = Activator.CreateInstance(service)
            Dim param1() As ParameterInfo
            param1 = t.GetParameters()

            Dim param2 As Object() = New Object(param1.Length - 1) {}
            param2(0) = TextBox1.Text
            Dim response1 As [Object] = t.Invoke(obj, param2)
            Response.Write("<br>")
            MsgBox("The Customer is : " + response1.ToString())
        End If
    End For
End If

```

```

        End If
        If t.Name = "Discover" Then
            Exit For
        End If
    Next
Else
End If
Catch ex As Exception
End Try
End Sub
Public Function rankservice(ByVal str As String) As String
    Dim con As New OdbcConnection("Dsn=bank")
    Dim query As String
    Dim flg As New ArrayList
    Dim flg1 As New ArrayList
    Dim i As Integer = 0

    con.Open()
    query = "select * from CustomerLoan where CustomerID='" & str & "'"
    Dim cmd1 As New OdbcCommand(query, con)
    Dim dr1 As OdbcDataReader
    dr1 = cmd1.ExecuteReader()
    While dr1.Read
        flg.Add(dr1(3))
    End While

    query = "select sum(payamt)+sum(InterestAmt) from CustomerLoanPayment where
CustomerID='" & str & "'"
    Dim cmd2 As New OdbcCommand(query, con)
    Dim dr2 As OdbcDataReader
    dr2 = cmd2.ExecuteReader()
    While dr2.Read
        flg1.Add(dr2(0))
    End While
    con.Close()
    Dim str1 As Long
    Dim str2 As Long
    Dim stre As String = "Not Eligible"

    For Each f As String In flg
        str1 = f
    Next

    For Each fl As String In flg1
        str2 = fl
    Next

    If str2 >= (str1 * 80) / 100 Then
        stre = "Eligible"
    End If
    Return stre

```

End Function
End Class

D) Service for Checking Loan Eligibility:

```
Imports System.Web
Imports System.Web.Services
Imports System.Web.Services.Protocols
Imports System.Data
Imports System.Data.Odbc
<WebService(Namespace:="http://tempuri.org/")> _
<WebServiceBinding(ConformsTo:=WsiProfiles.BasicProfile1_1)> _
<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()> _
Public Class IndianBank
    Inherits System.Web.Services.WebService
    <WebMethod()> _
    Public Function Login(ByVal sql As String) As String
        Dim con As New OdbcConnection("Dsn=bank")
        con.Open()
        Dim f As Boolean = False
        Dim cmd As New OdbcCommand(sql, con)
        Dim dr As OdbcDataReader
        dr = cmd.ExecuteReader()
        While dr.Read
            f = True
        End While
        con.Close()
        If f = True Then
            Return True
        Else
            Return False
        End If
    End Function

    <WebMethod()> _
    Public Function AvailableService() As String
        Dim con As New OdbcConnection("Dsn=bank")
        Dim query As String
        Try
            con.Open()
            query = "select * from customer"
            con.Open()
            Dim cmd1 As New OdbcCommand(query, con)
            Dim dr1 As OdbcDataReader()
            con.Close()
            Return True
        Catch ex As Exception
            Return False
        End Try
    End Function
    <WebMethod()> _
```

```

Public Function LoanEligibility(ByVal str As String) As String
    Dim con As New OdbcConnection("Dsn=bank")
    Dim query As String
    Dim flg As New ArrayList
    Dim flg1 As New ArrayList
    Dim i As Integer = 0

    con.Open()
    query = "select * from CustomerLoan where CustomerID=" & str & """"
    Dim cmd1 As New OdbcCommand(query, con)
    Dim dr1 As OdbcDataReader
    dr1 = cmd1.ExecuteReader()
    While dr1.Read
        flg.Add(dr1(3))
    End While

    query = "select sum(payamt) + sum(InterestAmt) from CustomerLoanPayment where
CustomerID=" & str & """"
    Dim cmd2 As New OdbcCommand(query, con)
    Dim dr2 As OdbcDataReader
    dr2 = cmd2.ExecuteReader()
    While dr2.Read
        flg1.Add(dr2(0))
    End While
    con.Close()
    Dim str1 As Long
    Dim str2 As Long
    Dim stre As String

    For Each f As String In flg
        str1 = f
    Next

    For Each f1 As String In flg1
        str2 = f1
    Next

    If str2 >= (str1 * 80 / 100) Then
        stre = "Elgible"
    Else
        stre = "Not Eligible"
    End If

    Return stre
End Function

End Class

```

APPENDIX II

SNAPSHOTS

Directory listing of Services:

Directory Listing -- /webCUB/ - Windows Internet Explorer

http://localhost:49601/webCUB

File Edit View Favorites Tools Help

Directory Listing -- /webCUB/ Page Tools

Directory Listing -- /webCUB/

Sunday, January 01, 2006 08:59 AM	edms
Wednesday, March 08, 2011 08:31 AM	edms
Monday, January 01, 2006 11:41 PM	edms
Wednesday, March 08, 2011 08:31 AM	edms
Sunday, January 01, 2006 10:14 AM	ELB
Sunday, January 01, 2006 10:11 AM	ELB

Version Information: ASP.NET Development Server 3.0.101

Internet Explorer Protected Mode: On

100%

List of available services:

z:\CUBBank Web Service - Windows/Internet Explorer

http://localhost:45901/ser/00E/ser_kurmbk

File Edit View Favorites Tools Help

z:\CUBBank Web Service Page View

CUBBank

The following operations are supported. For information, see [WS-HTTPBinding](#).

-
-
-

This web service is using <http://tempuri.org/> as its default namespace.

Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace to identify its usage. Although the namespace is not used in the Web client, the client must be aware of XML Web services that are under development. For published XML Web services, you can use a permanent namespace.

Your XML Web service should be identified in a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces follow the GUID-based method to identify resources on the Web, XML Web services do not use GUIDs.

For XML Web services implementing SOAP 1.1, the default namespace is `http://schemas.xmlsoap.org/soap/envelope/`. The `xmlns` attribute is an attribute applied to the `soap:Envelope` element in the XML Web service. For SOAP 1.2, the default namespace is `http://schemas.xmlsoap.org/soap/envelope/12/`.

```

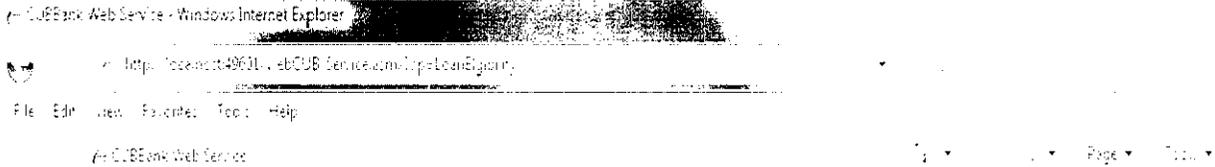
<?xml version="1.0" encoding="utf-8" ?>
<WebServices xmlns="http://tempuri.org/WebServices/"/>
  <public class NsWebService
  </implementation>
</WebServices>

<?xml version="1.0" encoding="utf-8" ?>
<WebServices xmlns="http://tempuri.org/WebServices/" ?>
  <public class NsWebService
  </implementation>
</WebServices>

```

Done Internet Explorer 11.0.9600.17529

Checking Loan Eligibility:



CUBBank

Click to return complete list of operations

LoanEligibility

Test

To test the service using the HTTP POST, click on the button below.

Execute a test

URL:

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders should not be replaced with actual values.

```
POST /webService.svc/LoanEligibility HTTP/1.1
```

```
Host: localhost
```

```
Content-Type: text/xml; charset=utf-8
```

```
Content-Length: Length
```

```
SOAPAction: "http://tempuri.org/LoanEligibility"
```

```
<?xml version="1.1" encoding="utf-8" ?>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://tempuri.org/LoanEligibility" ?>
```

```
<Body>
```

```
<LoanEligibility ?>
```

```
</LoanEligibility ?>
```

```
</Body>
```

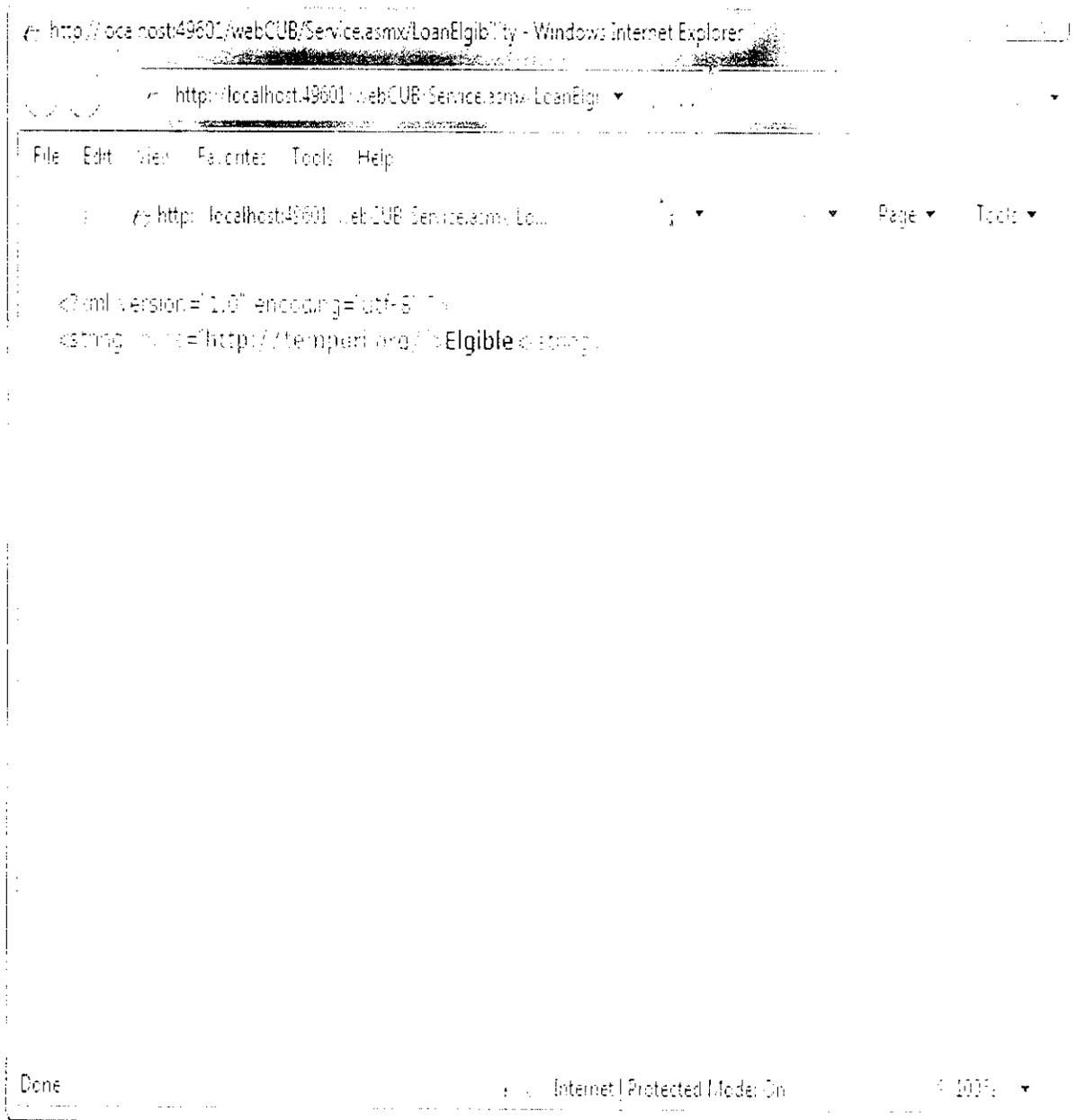
```
</soap:Envelope ?>
```

```
HTTP/1.1 200 OK
```

```
Content-Type: text/xml; charset=utf-8
```

```
Content-Length: Length
```

```
<?xml version="1.1" encoding="utf-8" ?>
```

XML output display:

The screenshot shows a Windows Internet Explorer browser window. The address bar displays the URL `http://localhost:49601/webCUB/Service.asmx/LoanEligibility`. The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The main content area displays the following XML output:

```
<?xml version='1.0' encoding='utf-8' ?>  
<string name='http://tempuri.org/Eligible'>
```

The status bar at the bottom of the browser shows "Done" on the left, "Internet | Protected Mode: On" in the center, and "9:00 AM" on the right.

User Interface showing loan details:

Untitled Page - Windows Internet Explorer

http://to.albhoct49825.webdlns.homeva.jp

File Edit View Favorites Tools Help

Untitled Page

WELCOME TO LOAN ANALYSIS WIZARD

[Home](#)
 [About](#)
 [Login](#)
 [Check Status](#)

Eligibility:

Individuals / traders/ small businessmen with a proven track record and having active and satisfactory dealings with the bank in the past for at least one year.

Purpose of the Loan:

The loan may be granted for meeting business requirements or any other genuine financial requirements of the applicant.

Permissible amount of the Loan:

1. The facility will be granted as a loan.
2. The maximum amount of loan will be Rs.20.00 Lakhs or 40% of the realisable value of the immovable property whichever is less

Margin:

60%

Processing Fees:

Processing Fees at 1% of the loan amount to be charged. And no other hidden charges as charged by other banks

Done

Internet (Protected) Home On

12/10

User Interface showing details about the bank:

Internet Explorer - Untitled Page - Windows Internet Explorer

http://localhost:49325/WebIn/About.asp

File Edit View Favorites Tools Help

← Untitled Page

WELCOME TO LOAN ANALYSIS WIZARD

[Home](#) [About](#) [Loan](#) [Check Status](#)

The bank, 'The Kumbakonam Bank Limited' as it was then called was incorporated as a limited company on 31st October, 1904. The first Memorandum of Association was Kumbakonam including Sarvashri R. Santhanam Iyer, S. Krishna Iyer, V. Krishnaswami Iyengar and T.S. Raghavachariar. Shri T.S. Raghavachariar was the First Agent of Santhanam Iyer who became the Secretary of the bank under the amended Articles of Association which created the office of a Secretary to be in charge of the Bank's Management till his death in 1926. He was succeeded by Shri. S. Mahalinga Iyer as Secretary who subsequently became the First full-time Managing Director of the bank and held the position of Secretary from 1926 to 1929 and that of Managing Director from 1929 to 1963.

The bank in the beginning preferred the role of a regional bank and slowly but steadily built for itself a place in the Delta District Thanjavur. The first Branch of the bank was opened at Nagapattinam in 1930. Thereafter, branches were opened at Sannanallur, Ayyampet, Tirukattupalli, Tiruvarur, Manapparai, Mayuram and Porayar within a span of twenty years. The bank came under the Reserve Bank of India Act, 1934, on 22nd March 1945.

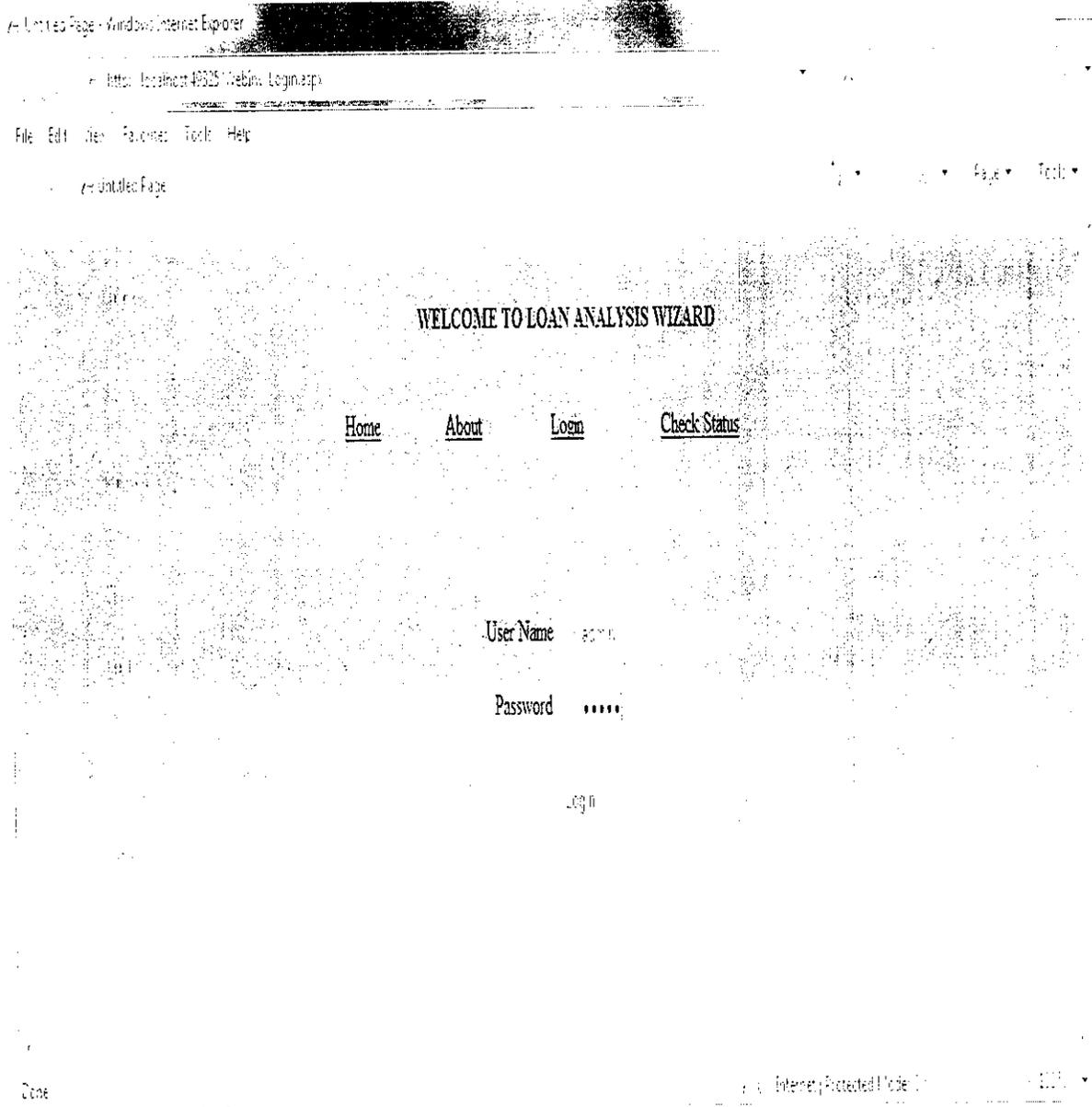
The Bank celebrated its Golden Jubilee on 14th November, 1954 at Kumbakonam under the Presidentship of Shri. C.R. Srinivasan, Editor, 'Swadesmitran' & Director, Kumbakonam.

In 1957, the bank took over the assets and liabilities of the Common Wealth Bank Limited and in the process annexed to it the five Branches of Common Wealth Bank at Jayankondacholapuram and Ariyalur.

In 1963, Shri. R. A. Venkataramani Iyer took charge as the Chairman of the Bank which position he held upto 1969.

Internet - Protected Mode On

Login Screen:



Checking Status:

As Untitled Page - Windows Internet Explorer

http://localhost:49325/Website/Report.aspx

File Edit View Favorites Tools Help

z Untitled Page Page Tools

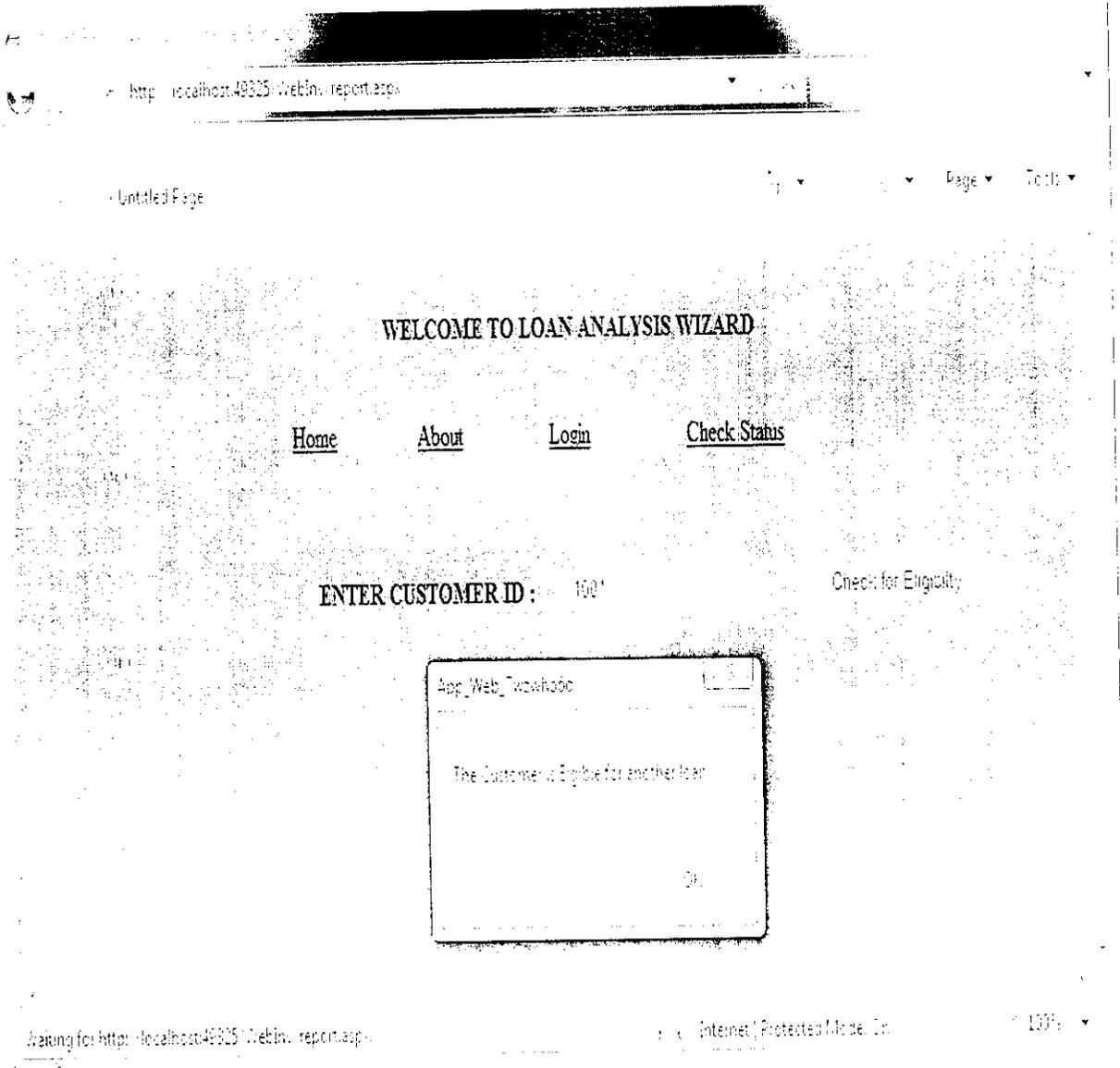
WELCOME TO LOAN ANALYSIS WIZARD

[Home](#) [About](#) [Login](#) [Check Status](#)

ENTER CUSTOMER ID : 1001 [Check for Eligibility](#)

Internet | Protected Mode | On 100%

Output Display:



Customer Table:

Customer : Table						
CustomerID	CustomerName	Address	Salary	dob	zip	
1001	Kane	Comcast	50000	4-8-1982	641001	
1002	Calandra	Comcast	60000	11-9-1985	641049	
1003	Saewald	Comcast	25721	6-3-1990	640003	
1004	Acel	Comcast	40000	4-9-1981	641007	
1005	Yasmin	Comcast	65322	12-5-1988	640001	
1006	Aolene	Comcast	21500	5-6-1986	641002	
1007	Tallon	Comcast	74801	12-6-1985	641004	
1008	Jenice	Comcast	36869	5-2-1938	641047	
1009	Sabrina	Comcast	30906	7-11-1938	641048	
1010	Samuel	Comcast	23887	1-1-1983	641049	
1011	Randy	Comcast	45278	6-2-1983	640005	
1012	Bailey	Comcast	93278	1-8-1934	641109	
1013	Oldrina	Comcast	71344	3-8-1961	641009	
1014	Vanessa	Comcast	86546	4-11-1960	640001	
1015	Rangford	Comcast	56485	4-4-1981	641002	
1016	Olivia	Comcast	89549	11-12-1982	641007	
1017	Barth	Comcast	56000	7-8-1983	640008	
1018	Rafaal	Comcast	64651	6-11-1982	641057	
1019	Hyles	Comcast	24256	12-12-1990	641056	
1020	Mysti	Comcast	36541	4-4-1989	640035	
1021	Thearl	Comcast	56565	4-6-1988	641078	
1022	Rachel	Comcast	65847	5-12-1986	641039	
1023	Oswin	Comcast	78425	3-6-1986	641158	
1024	Nicholas	Comcast	36546	4-6-1986	641151	
1025	Acot	Comcast	96658	8-8-1984	641051	
1026	Payden	Comcast	35412	11-6-1970	640019	
1027	Balder	Comcast	10203	11-3-1971	641119	
1028	Safford	Comcast	36540	1-8-1959	641128	
1029	Radford	Comcast	36689	2-5-1960	641145	
1030	Laxford	Comcast	25621	3-6-1965	641045	
1031	Bayhard	Comcast	66375	4-6-1964	640007	
1032	Hylke	Comcast	63786	5-11-1989	641007	
1033	Taotha	Comcast	65787	3-9-1972	640056	
1034	Lawrence	Comcast	64638	8-9-1970	641054	
1035	Tarynn	Comcast	38654	12-9-1974	640059	

Record: 1 of 350

Customer Loan Payment Table:

CustomerLoanPayment : Table					
	LoanID	CustomerID	PaymentDate	Payamt	InterestAmt
▶	L236547	1001	1-10-2007	8000	8000
	L236547	1001	1-11-2007	15000	8000
	L236547	1001	1-12-2007	15000	8000
	L236547	1001	1-13-2007	15000	8000
	L236547	1001	1-3-2008	30000	8000
	L236547	1001	1-4-2008	30000	8000
	L236548	1002	2-10-2007	13000	12000
	L236549	1003	2-10-2007	100000	4000
	L236550	1004	7-9-2007	48000	8000
	L236551	1005	8-10-2007	65322	4000
	L236553	1007	10-1-2007	20000	40000
	L236554	1008	11-10-2007	10000	20000
	L236555	1009	11-10-2007	4000	4000
	L236556	1010	12-10-2007	4000	4000
	L236557	1011	12-10-2007	8000	8000
	L236558	1012	12-10-2007	20000	36000
	L236559	1013	1-10-2008	20000	32000
	L236560	1014	2-10-2008	25000	32000
	L236562	1016	3-8-2008	20000	20000
	L236563	1017	3-11-2008	16000	16000
	L236564	1018	3-11-2008	15000	24000
	L236565	1019	4-8-2008	10000	10000
	L236566	1020	4-8-2008	10000	12000
	L236567	1021	4-8-2008	10000	12000
	L236568	1022	5-8-2008	20000	24000
	L236570	1024	5-8-2008	7000	10000
	L236571	1025	5-8-2008	25000	32000
	L236572	1026	6-8-2008	8000	12000
	L236574	1028	6-8-2008	7000	18000

Records: ◀ ▶ ▶▶ # of 90

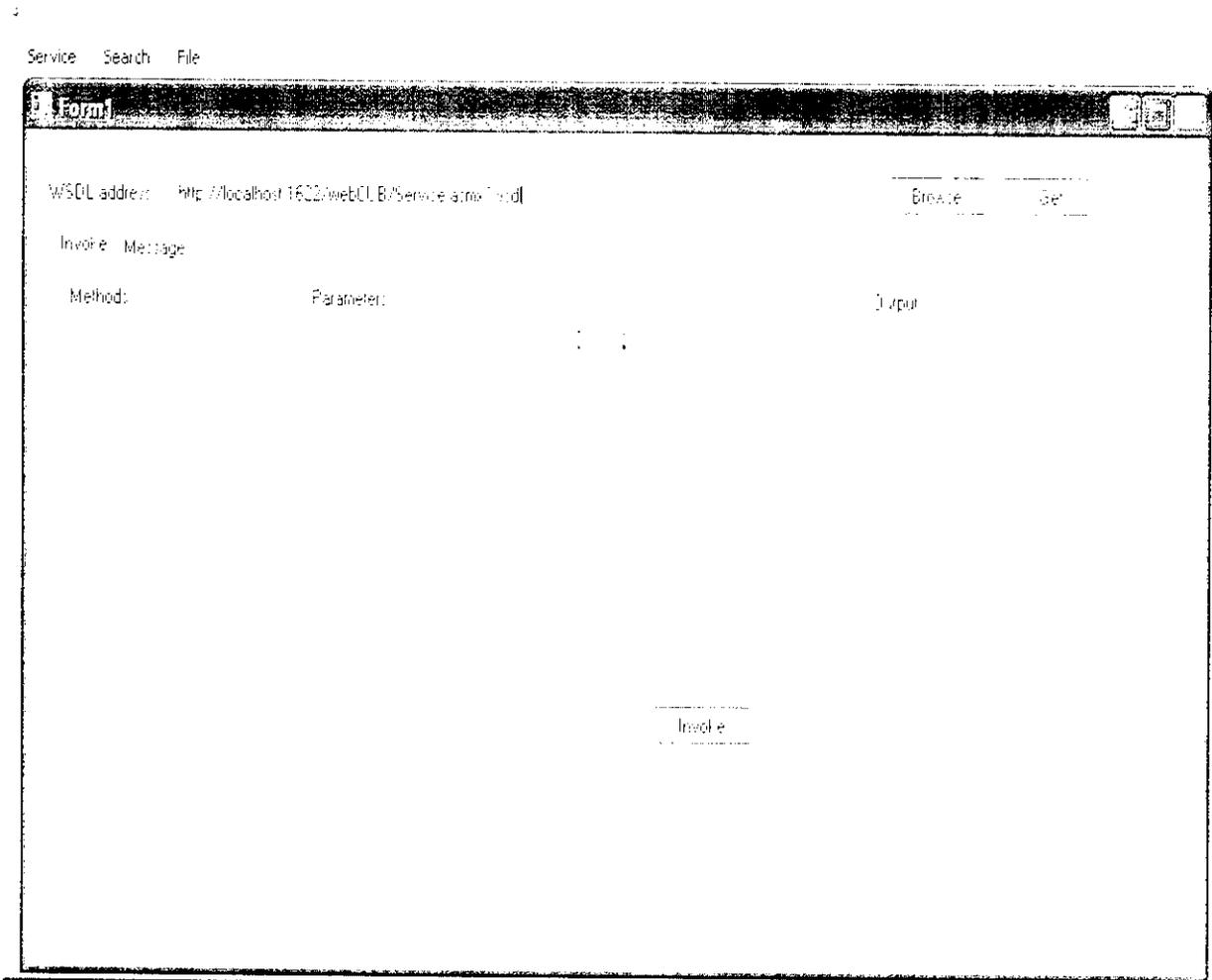
Customer Loan Table:

CustomerLoan : Table

CustomerID	LoanID	LoanName	TotalAmt	MonthlyInterestAmt	LoanSanctionDate
1001	L236547	Personal	200000	8000	1-10-2007
1002	L236548	Housing	300000	12000	2-10-2007
1003	L236549	Car	100000	4000	3-10-2007
1004	L236550	Housing	210000	8000	7-9-2007
1005	L236551	Agriculture	100000	4000	8-10-2007
1006	L236552				
1007	L236553	Business	1000000	40000	10-1-2007
1008	L236554	Housing	500000	20000	11-10-2007
1009	L236555	Personal	100000	4000	11-10-2007
1010	L236556	Agriculture	130000	4000	12-10-2007
1011	L236557	Car	200000	8000	12-10-2007
1012	L236558	Business	900000	36000	12-10-2007
1013	L236559	Housing	800000	32000	1-10-2008
1014	L236560	Business	800000	32000	2-10-2008
1015	L236561				
1016	L236562	Housing	500000	20000	3-8-2008
1017	L236563	Car	400000	16000	3-11-2008
1018	L236564	Business	800000	24000	3-11-2008
1019	L236565	Agriculture	250000	10000	4-3-2008
1020	L236566	Education	300000	12000	4-3-2008
1021	L236567	Personal	300000	12000	4-3-2008
1022	L236568	Business	600000	24000	5-8-2008
1023	L236569				
1024	L236570	Agriculture	250000	10000	5-8-2008
1025	L236571	Housing	300000	12000	5-8-2008
1026	L236572	Education	300000	12000	6-8-2008
1027	L236573				
1028	L236574	Car	250000	10000	6-8-2008
1029	L236575	Agriculture	200000	8000	7-8-2008
1030	L236576	Personal	100000	4000	8-8-2008
1031	L236577	Education	350000	14000	8-8-2008
1032	L236578	Business	350000	14000	8-8-2008
1033	L236579	Housing	700000	28000	9-8-2008

Records: 14 of 100

Using the Services URL in Windows Application



Listing available services:

Service Search File

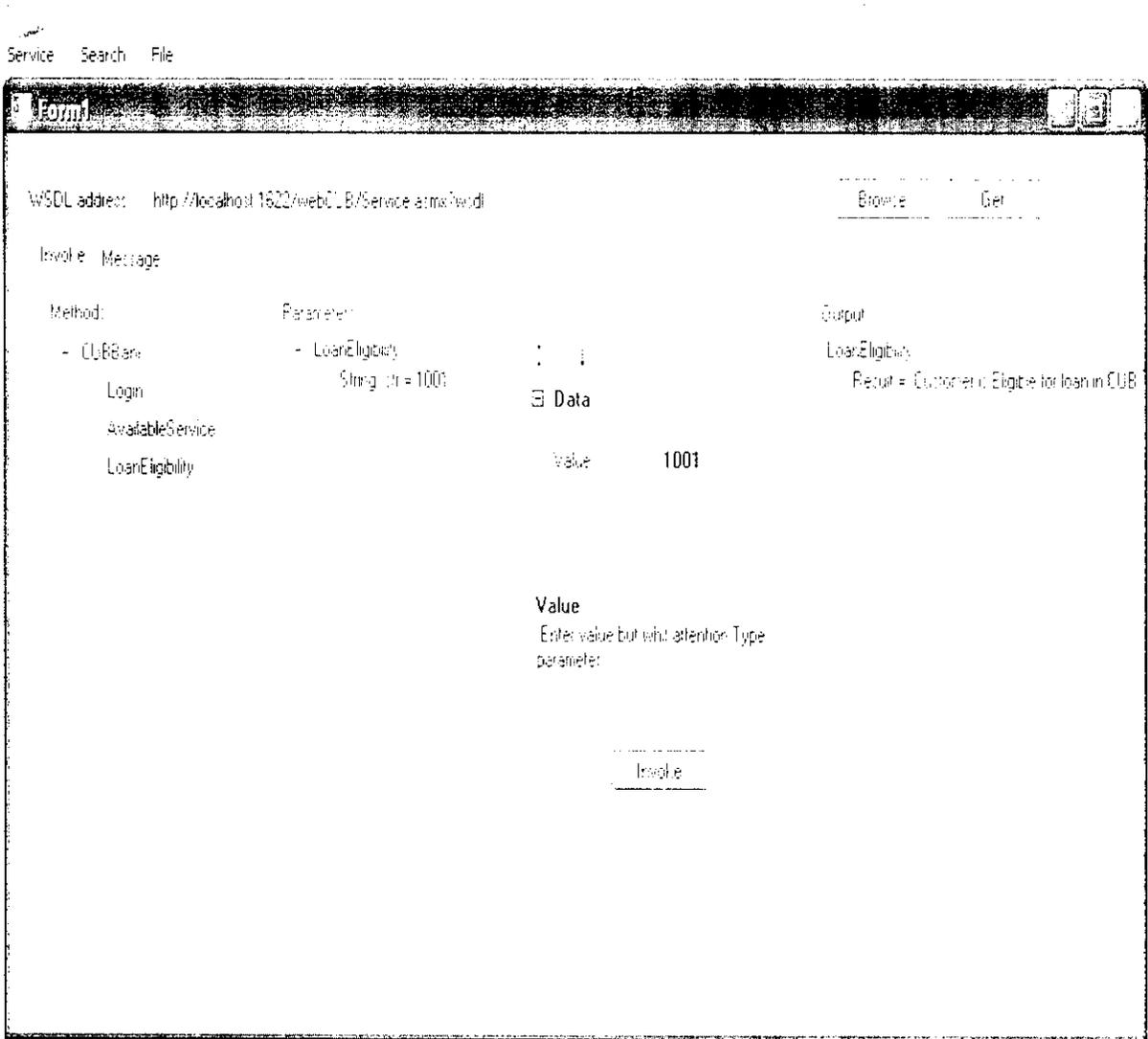
The screenshot shows a web service client window titled "Form1". At the top, there is a menu bar with "Service", "Search", and "File". Below the menu bar, the WSDL address is displayed as "http://localhost:1622/webCLB/Service.asmx?wsdl". To the right of the address are "Browse" and "Get" buttons.

Below the address bar, there is a section labeled "Invoke Message". Underneath, there is a table with three columns: "Method", "Parameter", and "Output".

Method	Parameter	Output
- CUBBank	- LoanEligibility	: ...
Loan	String str	Data
AvailableService		
LoanEligibility		1001

Below the table, there is a section labeled "Value" with the text "Enter value but with attention Type parameter". At the bottom of the window, there is an "Invoke" button.

Executing Loan Eligibility service:



8. REFERENCES

- [1] L. Agosta (2006), 'Data Strategy Advisor: Data Warehousing Raises the Bar on SOA'.
- [2] L. Agosta (2006), 'Data Strategy Advisor: Advance from Traditional to Dynamic Data Warehousing'.
- [3] T. Ariyachandra, H. J. Watson (2005) 'Key Factor in Selecting a Data Warehouse Architecture,' Business Intelligence Journal, Vol. 10, No. 2, PP.19-26.
- [4] T. Ariyachandra, H. J. Watson(2006), 'Which Data Warehouse Architecture is Most Successful,' Business Intelligence Journal, Vol. 11, No. 1, PP. 4-6.
- [5] C. Bussler, 'B2B integration: Concept and architecture'
- [6] Celina M. Olszak and Ewa Ziemia (2007). 'Approach to Building and Implementing Business Intelligence Systems', Interdisciplinary Journal of Information, Knowledge, and Management Volume 2.
- [7] Gamma, E., Helm, R., Johnson, R., and Vlissides (1994) 'Design Patterns: Elements of Reusable Object-Oriented Software'.
- [8] Liya Wu¹, Gilad Barash¹, Claudio Bartolini (2007) 'A Service-oriented Architecture for Business Intelligence', IEEE International Conference on Service-Oriented Computing and Applications
- [9] Niranjana Bhalivade (2011) , '1KEY Business Intelligence Customer Case Study', Comprehensive Business Intelligence Application Addresses Data Analysis & MIS Reporting Needs of CEAT Limited, MARCH, www.maia-intelligence.com

- [10] Potts, M. (2003), 'Find Bind and Execute: Requirements for Web Service Lookup and Discovery'. www.talkingblocks.com/resources.htm#.
- [11] J.R. Putman (2001), 'Architecting with RM-ODP', Prentice-Hall.
- [12] Sefan Linders (2008). 'Opportunities and limitations of using SOA concepts and technologies for building BI applications: a Delphi Study'.
- [13] Sunita Sindwani , IKEY BI, information request catalyst at Export Import Bank of India, www.eximbankindia.com
- [14] Vikas Gadre, Customer Case Study TATA Chemicals adopts IKEY Agile BI Suite
- [15] Vinay Nigudkar, 'Overview Comprehensive Business Intelligence Application Addresses Data Analysis & Reporting Needs of Reliance Mutual Fund', www.maia-intelligence.com
- [16] H. J. Watson (2005), 'Real Time: The Next Generation of Decision- Support Data Management', Business Intelligence Journal, Vol.10, No. 3, PP 4-6.