9. Strategic Dimensions of Proposed Multi Tier Architecture

9.1 Two-tier Architecture

Typical client/server systems have fallen into the category of a two-tiered architecture. The user system interface is usually located in the user's desktop environment and the database management services are usually in a server that is a more powerful machine that services many clients. Processing management is being split between the user system interface environment and the database management server environment. The database management server provides stored procedures and triggers.

The client/server architecture is a versatile, message-based and modular infrastructure that is intended to improve usability, flexibility, interoperability and scalability as compared to its earlier centralized, mainframe, time sharing computing. A client is defined as a requester of services and a server is defined as the provider of services. A single computing machine can both act as a server and a client.

Before going deeply into two-tier architecture, here comes a brief about the earlier computing architecture.

9.1.1 Mainframe architecture

With mainframe software architectures, all intelligence has been found within the central host computer. Users interact with the host through a unintelligent terminal that captures keystrokes and sends that information to the host. Mainframe computing architectures are not tied to a hardware platform. Users can connect with the host through any sort of systems including PC or Unix workstations. The main stumbling block with this architecture is that it does not support any graphical user interface or access to multiple databases from geographically dispersed sites.

9.1.2 File sharing architecture

The original PC networks were based on file sharing architectures, where the server downloads files from the shared location to the desktop environment. The requested user job is then run both logic and data in the desktop environment. File sharing architectures work if shared usage and content updating are very low, and the volume of data to be transferred is low. The main problem with this setup is that file sharing gets strained when the number of online users grow significantly. Also network traffic got congested as the full file has to be downloaded to the user machine each time he/she requests for a file.

9.1.3 Client/Server architecture

This approach basically introduced a database server to replace the file server. Using a relational database management system, user queries are answered directly and this primarily reduces the network traffic by supplying relevant query response to the client instead of the total file transfer. It highly improves multi-user updating through a GUI front end to a shared database. In this architecture, remote procedure calls or standard query language statements are being used by clients to communicate with servers.

The two-tier client/server architecture is a good solution for distributed computing when work groups are defined as a dozen to 100 people interacting on a LAN simultaneously. The application exists entirely on the client PC while the database sits out on a server machine. In this type of architecture, the full processing load is lying onto the PC while the more powerful server machine acts as a traffic controller between the application and the database.

This feature leads to very high utilization of the PC resulting in poor performance and there is a sharp increase in network traffic. This is a result of the server maintaining a connection via "keep-alive" messages with each client, even when no work is being done. When the entire application is processed on a PC, the application is forced to make multiple requests to data source for data before presenting anything to the end user. These multiple data requests can affect the precious network performance. Thus there arises a very high inhibition on scalability, availability and performance on the system. Also in a typical two-tier client-server environment, programmers write applications that are closely tied to vendor-specific software.

Another main problem with any two-tiered client/server approach is that of maintenance. If there is a small change affected in an application logic code, all the users accessing that code have to be taken off from the systems. As the days go by, this may lead some sort of confusion about which version of the software is being used and by whom. Thus performance degradation and maintenance issues are the two main bottlenecks prevailing in a two-tier architecture application apart from the issues such as lack of scalability and flexibility and very high network traffic.

9.2 Three-tier Architecture

In order to address the above-mentioned issues in an effective way, software developers community came with an evolutionary but effective and efficient solution referred to as "three-tier architecture". That is, an application can be separated into three logical layers, each with a well-defined set of interfaces. The first tier is called as the presentation layer and it normally consists of a graphical user interface. The middle tier consists of the application logic and the third tier is the data layer. The three logical layers can lie in the same machine also through smart software configuration.

Thus three-tier client/server applications employ an intermediary or middle-tier application server, which operates between client applications and the back-end databases. The middle tier composes business application logic code, which the user calls upon through the front-end graphical user interface to retrieve the desired data. The presentation layer on getting the data from the middle tier formats the data for displaying it to the user. This kind of separations brings a number of unique advantages such as flexibility and performance increment. Also multiple user interfaces can be built and deployed with out bringing any change to the application logic.

There are a variety of ways of implementing this middle tier, such as using Transaction Processing (TP) monitors, message servers, Object Request brokers (ORBs) or application servers. The middle tier can perform queuing, application execution, locating and connecting application objects and database staging. If the middle tier provides queuing, the client can deliver its request to the middle layer and disengage because the middle tier will access the data and return the answer to the client.

The third tier contains the data that is needed for the application.. The data can be from any source of information such an enterprise database like Oracle, MS SQL Server, a set of XML documents, a directory service like an LDAP server, or even any legacy and proprietary systems.

This approach heavily improves those lacking points found in two-tier application. There are two fundamental motivations for using three-tier architecture over a two-tier model:

- Improved Scalability, availability, and performance
- Improved flexibility, and extensibility of business systems

These features are being achieved through managing back-end resources in a more effective and smart manner. There are some good resource management techniques such as pooling and clustering middle-tier servers. Pooling makes three-tier systems more effective by allowing more clients to share scarce resources like database connections, which reduces the

workload on back-end servers. Clustering makes three-tier systems more available and scalable because multiple servers and resources can support fail-over and balance the word loads of a growing client population.

Three-tier systems are more flexible and extensible than their two-tier systems as the business logic and services such as security, persistence, transactions etc reside on the middle-tier and transparent to the client applications. This tends to make these services being applied automatically to client requests and any changes made in the business logic code do not reflect on the clients in any way.

Still it has been found that three-tier methodology lacks some critical features such as reusability of application logic code and scalability. That is, there may arise a situation whereby a collection of application logic code results and they can not be reused and also they do not communicate with one another. Thus there came a need for a viable architecture that mainly facilitates reusability of business logic as reusability phenomena has been found to reduce the cost of software development and the time to market and its quality is assured.

9.2.1 Transaction Processing (TP) monitor technology

The most basic type of three-tier architecture has a middle layer consisting of Transaction Processing (TP) monitor technology. The TP monitor technology is a type of message queuing, transaction scheduling, and prioritization service where the client connects to the TP monitor in the middle layer instead of database server on the back-end. The transaction is accepted by the monitor, which queues it and then takes responsibility for managing it to completion, thus freeing up the client. A typical TP monitor technology provides the ability to update multiple different DBMSs in single transaction connectivity to a variety of data sources including flat files, non-relational DBMSs and even mainframe systems the ability to attach priority to transactions and to give robust security.

For systems with thousands of users, TP monitor technology has been one of the most effective solutions.

The main limitations of this technology are that the implementation code has to be written in a low-level language such as COBOL and there is no visual toolsets to interoperate with the middle tier.

9.2.2 Message Server

This is another viable technology to implement three-tier architecture applications. Messages from clients are being prioritized and processed asynchronously. A typical message consists of a header

that contains priority information, and the address and identification number. The message server connects to the backend RDBMS and other data sources. The main difference between TP monitor technology and message server paradigm is that the message server architecture focuses on intelligent messages, whereas the TP monitor environment has the intelligence in the monitor, and treats transactions as dumb data packets.

9.2.3 Application Server

This helps the application logic code to be deployed in a shared host machine rather than in the user system interface

9.3 N-tier Architecture

Thus, came the notion of n-tier architecture. To turn three-tier architecture into an n-tier system, the middle tier can be allowed to have multiple application objects rather than a single application. Each of these application objects must have a well-defined interface which allows them to contact and communication with one another. An interface is actually brings an idea of contract. That is, each object states through its interface that it will accept certain parameters and return a specific set of results. Application objects use their interfaces to do business processing.

The strategic dimensions of proposed multi tier architecture w.r.t. the defined 11 components in chapter 4 are detailed as under.

9.4 Application Architecture

- Design and Development of applications
- Develop 3-tier or N-tier Applications
- All the applications should be developed using 3-tier or N-tier architecture in order to maximize flexibility and scalability.
- The applications having high usage volumes and/or long life spans will be better served by an N-tier service oriented architecture.
- The logical separation of thee tiers for user interface(s); business rules; and data access code allows for simple, straightforward additions to each of the three-tiers without undue impacts on the others.
- The logical separation of the tiers also allows for changing the platforms where tiers are deployed, resulting in a high degree of scalability. As transaction loads, response times, or throughputs change, a tier can be moved from the platform on which it executes to another, more powerful platform – or be spread over multiple machines – without impacting the other tiers.

- While many of the problems inherent in the existing monolithic and the two-tier applications can be overcome by implementing applications with a three - tier architecture, the goal should always be true, N-tier applications.
- The maximum benefits of an N-tier architecture are realized when many N-tier applications are deployed across the statewide offices, sharing common software services and offering multiple user interfaces.
- With three-tier client/server applications, there is less risk in modifying the code that implement any given business rule.
- Three-tier client/server applications can be made to support multiple user interfaces; character, graphical, web browser, telephones and others

9.4.1 Isolate Customizations to Purchased Software

- Isolate customizations into separate modules from the purchased software itself to improve the ability to upgrade and move to new releases as required over time for purchased line-of-business applications, loosely couple custom application servers. Avoid use of CGI for information publishing, back-end applications or data access.
- Publishing information to the web with HTML or XML via Java servlets reduces overhead and works in conjunction with EJB based components
- The use of ASP or other HTML publishing is acceptable for publishing only (Not business logic) but JSP and servlets are preferred.

9.4.2 Managing Applications

- All applications deployed must be designed, to be managed by SNMP.
- By standardizing on SNMP as the instrumentation protocol, there is an opportunity across the districts to benefit from reusing management instrumentation code.

9.5 Information Architecture

- When accessing relational databases, use the industry standard of ANSI standard SQL
- When using a database access tool that uses SQL calls, do not use any vendor specific extensions.
- Use ODBC from any data access programs rather than vendor-specific database access tools.

- ODBC allows flexibility in programming. A database can be easily modified or relocated. If a change is needed, the change is made to the ODBC configuration file, not to each business intelligence program or tool.
- Implement a server-based ODBC solution rather than a workstation-based ODBC implementation.
- A server based ODBC solution is easier to administer. ODBC database changes and additions are easier to manage, since updates are made to ODBC servers, not every workstation that uses ODBC.
- Use domain name system (DNS) names for databases that are accessible via TCP/IP.
- A DNS server provides the capability for a long or complicated TCP/IP location to be accessed by generic, short alphabetic name. It is basically a lookup service. It maps the generic alphabetic DNS name to its complicated TCP/IP location. The client application programs can be configured to use the generic names when they need to access a database. If the database location changes, the DNS configuration is changed, and no changes are needed to each client configuration.

9.6 Groupware Architecture

9.6.1 Infrastructure - Content Exchange

The following standards have been established for content exchange. These standards ensure seamless processing of documents across the state, and are summarized in a table for easy reference.

For non-editable documents, the standard file format is PDF. Typical application software using this file format includes word processing, imaging systems, and World Wide Web publishing.

Typical Document Input Source	Typical Application Software Used	File Format Standard
Non-editable documents	• Word Processing • Imaging System • World Wide Web Publishing	▪ PDF
Monochrome documents or drw	• Word Processing • Archive & Retrieval • Workflow • Multimedia • Digital Publishing • Pattern Recognition • Geographic Information System	▪ TIFF using CCITT/ITU Group IV compression

Typical Document Input Source	Typical Application Software Used	File Format Standard
Color documents, drawing or photographs	• Multimedia • Word Processing • Digital Publishing • Pattern Recognition • Geographic Information system	▪ GIF ▪ JPEG
Facsimile documents	• Word processing • Archival and Retrieval • Workflow	▪ TIFF using CCITT/ITU Group III compression
Multimedia images	• Multi-media	▪ MPEG

Table 9.1

- PDF is widely deployed for non-editable content exchange
- The reader is available at no cost
- For monochrome documents or drawing, the standard file format is TIFF using CCITT/ITU Group IV compression
- Typical application software uses this file format includes word-processing, archive and retrieving, workflow, multimedia, medical systems, digital publishing, pattern recognition, and geographical information systems.
- For color documents, drawings or photographs, the standard file formats are GIF and JPEG.
- Typical application software this file format includes multimedia, word-processing, medical systems, digital publishing, and geographic information system.
- For facsimile documents, the standard file format is TIFF using CCITT/ITU group III compression.
- Typical application software using this file format includes word processing, archival and retrieval, and workflow.
- For vector or geometric data, the standard file formats are DGN and DWG
- Typical application software using this file format includes CADD and geographic information systems.
- For multimedia images, the standard file format is MPEG-1/2.

- Typical application software using this file format is multimedia.

9.6.2 Communication – Electronic Mail (e-Mail)

- Similar to other groupware products, email protocols and standards are still emerging. For almost every protocol, there are several competing, non-compatible standards. If two email systems confirm to different standards to access their mail servers, errors may occur when messages are sent between the two systems. Approval of new standards is slow, leading to the proliferation of proprietary protocols and protocols extensions. Without consistent standards, a barrier in communication is created between platforms, applications, and components. To overcome these barriers, email gateways have been developed to integrate incompatible email systems.
- Use Simple Mail Transport Protocol (SMTP)
- Simple Mail Transport protocol (SMTP) is the standard transport protocol for sending messages from one MTA to another MTA over the Internet. Using MIME encoding, it enables the transfer of text, video, multimedia, images and audio attachments. It is the predominant transfer protocol utilized by web browser-based email user agents.
- Use Multi-purpose Internet Mail Extensions (MIME)
- Multi-purpose Internet Mail extensions (MIME), a SMTP message structure, is the standard specification for the attachment of audio, video, image, application programs, and ASCII text messages. The content type is stored in the message header as mail extensions. When the message is delivered, the player or application specific to the content type is opened so that the attachment can be viewed in its native format. If the player or application is not included with the browser, then the user must load it. Common image video players are included with most browsers.
- The MIME standard will require standardization of certain protocols in the near future. By its definition, MIME is transformable. Although two applications may be MIME-compliant, each application can use a proprietary or custom set of extensions. The data associated with the proprietary extensions may be lost in transfer. Common protocols cut down on the risk of a loss of data occurring.
- Use Internet Message Access Protocol version 4 (IMAP4).
- Internet Message Access Protocol version 4 (IMAP4) is the standard protocol for access to the mail server. The user has the

option of storing and manipulating messages on the mail server, which is important for job functions that require the user to access email from several different clients. IMAP is also ideal for situations where the user has a low speed link to the mail server. Instead of downloading all messages to the client, IMAP allows the user to select which specific message to download. If a message has several MIME attachments, the user can specify that only the text portion of the message is to be downloaded for viewing. This practice is considerably more efficient in the event that a high-speed link is not readily available.

- Note: Options sometimes exist to configure email servers and clients without IMAP4 settings. Email servers and clients should be implemented using IMAP4.
- Use Lightweight Directory Access Protocol (LDAP)
- Lightweight Directory Access Protocol (LDAP) is the standard directory access protocol. LDAP is based on Directory Access Protocol (DAP), an X.500 standard access protocol. X.500 is a set of CCITT/ITU standards for electronic directory services. LDAP has been proven to be more efficient for MUA to directory services transaction. In addition, LDAP can be utilized to access databases other than the email directories, which will add value to other groupware applications, such as scheduling.
- Select an email server system that allows multiple standards-based email clients.
- When an email server uses IMAP4 standard, any IMAP$-based client can access that server.

9.6.3 Collaboration – Document Management

- As the state progresses with the development of a departmental and statewide document management infrastructure, and universal access to information, there are many obstacles to overcome related to the inter-operability between departmental systems and their interaction with an evolving enterprise level locator service. In the standard area departments planning for EDM systems and services needs to take into account three general sets of guidelines.
- Existing and evolving standards and guidelines being advocated by public institutions and private industry through the Association for Information and Image Management International (AIIM).
- Other standards and guidelines put forth by related parts of the Statewide Technical Architecture, especially as they relate to the

development of document and/or business process-centric applications in n-tier architecture.

- There is no limit to the creativity and innovation occurring in the development of solutions to the problems of document management, workflow, and universal access. The framework embodied by these standards is intended to be moving users and developers toward the goals described previously.
- Implement document management systems and components that conform to the Document Management Alliance specifications (DMA 1.0 and ODMA 2.0).
- There are numerous issues related to interoperability among document management applications, services and repositories. Standards are needed to manage the increased life expectancy and complexity of re-usable electronic documents and content.
- The Document Management Alliance (DMA) is a task force of AIIM. DMA confirming products will support open design for user interfaces, workstations, network operating systems and servers. The DMA provides a framework for vendors to deliver products that provide query services (simple transparent access from every desktop to information anywhere on the network), and library services (including check-in and check-out, version control, security, and accountability). The DMA is working with the Open Document Management API (ODMA) group which specifies the common application program interfaces, and high level call interfaces that enables other client applications (such as MS Office) to work seamlessly with DMA compliant document management systems.
- For more information about AIIM standards programs, refer to the Web site: http://www.aiim.org/industry/standards.
- Implement workflow systems that confirm to the interface specifications of the Workflow Management Coalition (WfMC).
- WfMC is another working group of AIIM and is closely aligned with the work of the DMA. As automated workflow systems continue to evolve, the complexities associated with a common approach to process definition, process repositories, object manipulation and transport, and user interfaces are enormous. The workflow Management Coalition (WfMC) has proposed a framework for the establishment of workflow standards. This framework includes five categories of interoperability and communication standards that will

allow multiple workflow products to coexist and inter-operate within a network environment. This framework is contained within a Reference Model for workflow management systems that includes five interface specifications. The model includes the following:

- Process Definition Tools.
- Workflow Enhancement Services.
- Workflow Client Applications.
- Invocation of Native Applications.
- Workflow Package Interoperability.
- At this time, there are many companies designing products that comply with one or more of these interface specifications. HoDs planning production workflow applications that need to route work outside of the production system for processing or decision-making should work carefully with vendors and services to determine functional requirements and WfMC standards compliance.
- For more information on WfMC and the work of the coalition refer the Web site at: http://www.aiim.org/wfmc/index.htm.

- Use Adobe Acrobat Portable Document Format (PDF) for non-editable Electronic Documents.
- All documents in final form and prepared for distribution and publishing with no intention for further modification must be stored and delivered in Adobe-PDF format.
- Ensure hardware/software and image file compatibility using TWAIN, ISIS and TIFF standards.
- For typical business document imaging applications, software that controls the operation of scanner (and some other recognition peripherals) is provided. Not all scanner hardware and scan software are compatible. The industry standards to adhere to are TWAIN and more recently ISIS (Image Scanner Interface Specification). These are API standards that provide low-level integration facilitating the control of the peripherals from many common user applications. For specialized applications (e.g. hand held devices) other standards requirements need to be investigated.
- The scanned images of typical business documents should be committed to storage in Tagged Image File Format (TIFF) version 6.0 using CCITT/ITU Group III or IV compression. Organizations

planning imaging applications should investigate and demonstrate that any product selected is capable of exporting images in a format that they can be reused. Images that cannot be shared are a wasted investment and could result in the loss of critical data.

- Avoid new deployment or migrate away from proprietary image file formats. The current technology direction for image file format is TWAIN. The emerging technology file format is ISIS.
- Select magnetic storage subsystems. Select optical storage subsystems based on smaller standard form factors.
- Typical electronic documents, created with office automation suits, will reside on industry standard magnetic disk that is server or network attached. This will generally be transparent to the users of an EDMS. The images of scanned paper documents might also be stored on standard network attached magnetic disk. Magnetic storage will always provide the most performance in the speed of retrieval, and magnetic disk is increasingly cost competitive with optical disk storage. When selecting any magnetic storage solution, adhere to other parts of the STA that provide the standards for these types of systems.
- Very large document collections (usually image applications) will probably require optical storage subsystems (many are proprietary). Where there is a requirement for the permanent storage of unalterable documents, optical is chosen in the form of Write Once Read Many (WORM) disks. These types of systems generally involve special software that is used to manage the storage and movement of documents from optical to magnetic when documents are requested by users. Optical disks may be mounted in single standalone drive units or they may be loaded into various sizes of "juke boxes". Software handles the retrieval and loading of specific disks in response to user requests. Typical EDMS systems today will use a 5.25 form factor and will be WORM or Compact Disk type formats. Larger disks are available for specialized applications and are generally proprietary.
- Avoid new deployment or migrate away from proprietary or large format optical storage subsystems. The current technology direction is WORM and various types of compact disc in 5.25" format. The emerging technology is magneto Optical and DVD.
- Use extensible Markup Language (XML 1.0) when capturing or authoring document contents that requires further automated

processing by other information systems and Web based clients using standard XML enabled browsers.

- This standard is promulgated by the World Wide Web Consortium (W3C).
- XML is a subset of the Standard Generalized Markup Language (SGML, an ISO standard).
- XML encodes a description of a document's storage layout and logical structure with a document type definition (DTD). It provides a mechanism to combine structured data and unstructured information content.
- XML allows information systems and applications to automatically process XML documents when the systems are combined with an XML processor.
- The specification (DTD) describes the required behavior of XML processors in terms of how they read XML documents, and what information they must provide to the processing applications. For more information about the W3C and XML refer to the Web site at: http://www.w3.org.

9.7 Component ware Architecture

9.7.1 Component Reuse

The standards in this section pertain to component reuse.

No vendor proprietary API calls for infrastructure security services. Use Generic Security Service-API (GSS-API) or Common Data Security Architecture (CDSA) compliant API calls and products.

- Applications requiring security services prior to CDSA products or services being available can use the GSS-API.
- The GSS-API is an Internet Engineering Task Force (IFTF) standard (RFC 2748, released in January 2000, obsoletes RFC 1508 and RFC 2078) and supports a range of security services such as authentication, integrity, and confidentiality.
- It allows for plug-ability of different security mechanisms without changing the application layer.
- It is transport independent, which means it can be used with any underlying network protocol.
- Applications using GSS-API can be retrofit to a CDSA foundation without major modifications; therefore providing an interim step, to CDSA based services.

9.7.2 Component Services

- These standards in this section pertain to component services.
- Custom developed application components must be written in a portable, platform-independent language, such as C, C++, or Java.
- Application components written in a portable language are easier to move from one platform to another because they require fewer modifications to confirm to the new host. This portability allows an application to be more adaptive to changes in platform technology.
- Departmental infrastructure services must be in line with the Common Data Security Architecture version 2.0 (CDSA v2.0) compliant.
- The CDSA version 2.0, architecture is an open Group specification for providing security services in a layered architecture and managed by a Common Security Service Manager (CSSM). CDSA provides a management framework necessary for integrating security implementations. Version 2.0 of the specification is a cross-platform architecture, which includes a testing suite for inter-operability testing.
- A wide range of vendors has announced support for the specification and products for a broad set of platforms can be expected.
- Security protocols such as SSL, S/MIME, IPSec can all be built from the Common Data Security Architecture base.

9.7.3 Object Oriented Components

The standards in this section pertain to object oriented components.

Purchased applications must be CORBA 2.0 or later and IIOP (Internet Inter-ORB protocol) compliant.

- CORBA and IIOP standards are open standards devised for platform independence.

Build or purchase Enterprise solutions on an Enterprise Java Bean and servlet model. Application servers should be compliant EJB 1.1 or better.

- Enterprise solutions benefit from reduced requirements to code underlying services and can focus on business logic.
- Vendors provide solutions based on an EJB model. These solutions can be purchased and used without customization.

Avoid OLE/DCOM and Windows DNA object model for applications with Enterprise or strategic implications.

- OLE/DCOM standards do not scale well and run only on Windows platforms.
- OLE/DCOM applications are not easily portable or integrated into enterprise-wide solutions.

9.8 Data Architecture

9.8.1 Centralized Metadata

Custom systems must comply with CMR element definitions.

- New databases belonging to custom systems must comply with CMR elements definitions. The Metadata Elements Review Team will review the data elements to determine if the elements confirm to existing standards.
- Any new potential data element standards must be proposed and reviewed for approval as a state standard.
- If the data element definition cannot be customized to conform to existing standards, a waiver must be requested.

Commercial off-the-shelf (COTS) systems that support client-controlled data element definitions must comply with the CMR standard data element definitions. Otherwise, the vender must provide a conversion routine to conform to metadata exchange standards for data sharing.

- If an off-the-self system has data formats that can be modified, the data elements should be adapted to conform to the standard data requirements.
- If the data element definition cannot be customized to confirm to existing standards, the vendor must provide conversion routines to conform to the CMR metadata exchange standards.

Use Centralized Metadata Exchange Standards when exchanging data across departments.

- If data needs to be exchanged across office boundaries and the data is physically stored differently, then the data must be exchanged through the exchange standard as specified in the CMR.

9.8.2 Data Access Middleware

The standards in this section pertain to data access middleware.

Use OLE DB or JDBC database access middleware when accessing a database.

- Use OLE DB or JDBC to access a database instead of vendor specific database middleware.
- OLE DB and JDBC allow flexibility in programming. A database can be easily modified or relocated. If a change is needed, the change is made to the OLE DB or JDBC configurations, not to each data access program or tool.
- These technologies are widely supported by the industry and make an application more adaptable to changes in database or other technology requirements.

Implement a server-based OLE DB or JDBC solution as opposed to a workstation-based OLE DB, ODBC, or JDBC implementation.

- A server-based solution is easier to administer. Database changes and addition are easier to manage, since updates are made to database middleware servers, not every workstation that requires access.

Use domain name system (DNS) alias names when accessing databases through OLE DB and JDBC.

- If the database location changes or if the server name changes, the DNS configuration is changed, and no changes are needed to each client configuration.

9.8.3 Data Access Implementation

The standards in this section pertain to Data Access Implementation.

Use the State Service Broker (GUJSB) for inter-department data sharing.

- Service Broker is the standard for inter-department data sharing. Inter-department services deployed using GUJSB can be easily leveraged by other authorized applications.
- The department owning the data is responsible for writing the shared service to access data. This ensures data integrity and proper data interpretation.
- The department requesting the data is responsible for writing the request to retrieve shared data according to the shared service specifications.

Use the industry standard of ANSI standard SQL when accessing relational database.

- When using a database access tool that uses SQL calls, do not use any vendor specific extensions.

9.8.4 Data Security

The standard in this section pertain to Data Security.

Change all default database passwords.

- System administrator accounts have full access to all databases in a database server. Hackers often attempt a login to a system administrator account using a default password. As soon as a database is set up, change all default passwords.

9.9 Application Communication Middleware Architecture

9.9.1 Application Communication Middleware Types

- The standards in this section pertain to application communication middleware types.

There is no Remote Procedure Call (RPC) standard. Uses the GUJarat Service Broker (GUJSB) for inter application communication.

- Even with an RPC that is endorsed by a vendor neutral party, such as The Open Group, there is no standard RPC.
- RPCs are available from different vendors, such as The Open Group's DCE RPC, Sun Microsystems'
- ONC/RPC, and Microsoft's RPC
- Each vendor's version has a different application-programming interface and they do not inter-operate with one another.

There is no Message Oriented Middleware (MOM) standard. Use the state of Gujarat's service broker for inter-application communication.

- At present, all message-oriented middleware is proprietary. Products from different vendors have different application programming interfaces, which do not inter-operate with one another.

There is no distributed transaction processing (TP) monitor standard. Use the state of Gujarat's service broker for inter-application communication.

The applications coordinated by a transaction monitor will run on different platforms with access to different databases and resource managers.

The applications are often developed using different tools and have no knowledge of one another.

Industry standards specify how a TP monitor interfaces to resource managers, other TP monitors and its clients.

- X/Open XA specification defines specifications for two-phase commits that work with distributed databases.
- X/Open TX standard defines transactions.
- X/Open X/ATMI provides a standard transaction management interface.

9.9.2 Application Communication Middleware Brokers

The standards in this section pertain to the application communications middleware brokers.

Use of the service broker is required for inter-application communication.

- The service broker was put in place due to the lack of standards for inter-application communication types such as RPC, MOM, and TP monitors.
- While the lack of standards is not an issue for development of any single application, it poses problems for communication between applications. The broker is proposed as a standard communication paradigm for inter-application communication.

9.10 Integration Architecture

9.10.1 Application Integration

Standards in this section pertain to the application integration.

Clearly Define Application Interfaces

- To integrate applications for which the state has no source code rights, application interfaces must be clearly defined in order to allow reliable communication between applications.
- To facilitate purchase of best-of-breed software while easing application integration issues, the application interfaces must be clearly defined.

The message structure must be documented.

- A message or transaction is the mechanism for extracting data from an application or sending data to an application.
- Programmers' integrated applications need to know record length and type (i.e., whether it is a variable or a fixed length record, and if

it's variable, the delimiting characters used to separate the fields), and know which fields are optional versus required.

- A description of the data for each field is also, necessary.
- Explanations and examples of record formats and field descriptions are helpful and should be included.

The application must be able to transmit and receive messages using a client / server model.

- The client is the process that sends or originates the message. The server is the process that receives the message.
- Clients and servers may communicate using TCP/IP and sockets, or other communication protocols, such as Serial and FTP, as long as they perform the same transmit and receive functionality.
- Packetisation character, which identifies the start and end block strings, and message acknowledgment format, must also be provided.

Purchase line-of-business application software rather than custom developing it whenever possible.

- Purchasing line-of-business application software can permit the department to respond to business needs in a timelier manner than custom developing software.
- Published API's are insufficient because their use requires custom development of departmental applications and it may be impossible to interface two purchased applications. Use of an interface engine provides greater flexibility.

9.10.2 Data Access Integration

Clearly define and publish DTD/Schemas

- This will facilitate their use and re-use.
- Give reference to those DTD/Schemas, which have been developed based on any other globally defined schema, since this will allow incorporation of future changes.
- Wherever the DTD/Schemas apply for intra-state application, and are not expected to be shared in the public domain, adequate care needs to be taken in publishing them.

9.11 Network Architecture

9.11.1 Local Area Network (LAN) Architecture

The following standards have been established to assist various offices of Industries department in the implementation of LAN. The goal is to employ only open systems based on industry-approved standards, but a full complement of open standards does not yet exist for all components of LANs. Therefore, a combination of industry standards, de facto industry standards, mutually agreed upon product standards, and open standards are currently required to support the state's heterogeneous operating environment. All standards will be periodically reviewed.

The standard for LAN cabling is category 5, 6 or 7 Unshielded Twisted Pair (Cat 5 UTP, Cat 6 UTP, or Cat 7 UTP). Unless specific needs exist, such as high EMI or long distances, UTP should be considered for the horizontal runs in cable layouts.

- CAT 5/6/7 UTP can be certified to carry 10/100/1000 MBPS of data.
- It is an industry standard wiring plan and has the support of the IEEE.
- Wiring, cable, connector, and equipment vendors have standardized on this cabling.

The standard for standard link layer access protocol is Ethernet, IEEE 802.3 Carrier Sense Multiple Access / Collision Detection Access Method (CSMA/CD).

- Widely accepted format.
- Reliable, the protocol has been used for years and is very stable.
- Scaleable, faster versions are currently emerging to help manage the increase of data flow.
- 100Base T Gigabit Ethernet has the bandwidth necessary to support the needs of future voice and video requirements.

9.11.2 Wide Area Network (WAN) Architecture

In telecommunications, standards for products and services were created by the originating industry monopoly (i.e., the phone company). Therefore, even though the monopoly has been disbanded, the proven standards that were established have remained. With data communications, however, there have been always been many companies offering individual products and services. Therefore, although interim product standards have emerged as one company's product gained market share, there has been a lack of industry level

standards. Therefore, until industry standards are established, an enterprise must choose to implement product based standards in order to create a manageable solution to the maintenance and management of its data communications infrastructure.

The standard protocol technology is TCP/IP.

- Open protocol.
- Allows Internet access.
- Allows creation of Internet and VPNs.

The standard Internet access technology is Domain Name System (DNS) and IP address assignments are provided by the State for those departments participating in the Gujarat State Wide Area Network (GSWAN).

- State must assign IP addresses to allow LANs access to the GSWAN.
- It allows a structured naming conviction and IP address allocation for the state's WAN and domain names.

9.12 Platform Architecture

9.12.1 Server Platform Architecture

Run Distributed application servers on platforms supporting "open" operating systems.

- Open operating systems are available from multiple vendors, such as Windows 2000.
- Open operating systems are in the public domain, but have significant industry support, such as Linux.

Make sure server platforms are POSIX compliant.

Make sure server platforms comply with third party certifications:

UNIX	Microcomputers
Manufacturer is ISO 9002 certified	Manufacturer is 9002 certified
XPG4 Branded UNIX 93	Gartner Group Tier 1 or Tier 2 classified

Table 9.2

Third Party Certification for Server Platform Standards

- Third party certifications foster quality product purchases from manufactures that have demonstrated abilities to deliver and support these products.

9.12.2 Client Platform Architecture

The standards listed below have been established for the client platform architecture.

Two-dimensional (2-d) bar codes should use PDF417 coding standard.

The PDF417 bar code standard is capable of storing data such as product information, maintenance schedule, shipping information or others.

Platforms must comply with third party certifications.

Client platforms must comply with third party certifications as specified in the table below.

UNIX	Microcomputers
Manufacturer is ISO 9002 certified	Manufacturer is ISO 9002 certified
XPG4 Branded UNIX 93	Gartner Group Tier 1 or Tier 2 classified

Table 9.3

Third party certifications for client platforms

This will assure quality of platform hardware and software components.

Avoid proprietary smart cards reader-side APIs.

No standards exist for smart card reader-side APIs for application and platform integration. Use reader-side APIs from established platform vendors, such as PC/SC for the windows environment or use APIs that strictly adhere to the ISO 7816/4 command set.

9.12.3 Storage

Use either SCSI or FC-AL technology for the disk drive interface.

Use RAID with fault-tolerance in the storage subsystems.

SAN based fiber channel technology for large-scale storage deployments running mission-critical applications to be the de facto standard.

9.13 Security and Directory Services Architecture

9.13.1 Identification

The standards in this section apply to security identification.

- ISO 7816/1-4 standards define the electrical resistance, positioning of electrical contacts, communication protocol between card and card reader, and command set recognized by smart cards.
- They correspond roughly to the OSI layered model.
- The command set defined by the ISO 7816-4 standard are included in whole or in part by most smart cards on the market.

ISO 14443A and Mifare Smart Card standards for contact less smart cards.

- ISO 14443A standards for contact less smart card define the characteristics and communication protocols between contact less cards and card reader. These standards are still in development.
- The Mifare architecture is the de facto global interface standard for contact less and is based on ISO 14443A.
- Contact less cards under this standard use RF power and frequency protocols and cover read/write distances up to 10 cms of the reader.

Use PKCS #11 or PC/SC for integration of smart cards and host/reader-side applications.

- PKCS #11 from RSA is a widely accepted standard for integrating smart cards to applications supported by many vendors.
- PC/SC is widely accepted for integration of smart cards on Intel platforms.

Speaker Verification API (SVAPI).

- SVAPI is an API used for incorporating speaker-recognition technology into desktop and network applications.
- A consortium of vendors, technology developers, researchers VARs and end-users developed the SVAPI.
- The SVAPI offers interoperability over distributed environments with related APIs.
- They include SVAPI, the telecom industry's S100, a standard architecture for developing computer-telephony applications, and Java Speech, a standard for speech recognition using Java.

Human Authentication API version 2.0 (HA-API).

- The Human Authentication API(HA-API) is a generic API designed to allow a common set of instructions to integrate biometrics into applications requiring identification.
- It supports the enrollment sampling, processing and verification of biometrics.
- The API supports multiple biometric template types and multiple vendor technologies for each biometric type in one database. This permits an enterprise wide approach to biometric identification while allowing different application-specific biometrics to be used. A single database also facilitates the use of multiple biometrics in a single application.
- The API permits changing the biometric used without requiring application code changes.

Use open standards for smart card masks such as MULTOS.

- Highly secure procedures from manufacturing to card issuer.
- Allows multiple applications on the same card, addition and deletion at any point of time during the file of the card.
- High application level security.
- Manufacturer independent mask supported by several card and chip manufacturer.

9.13.2 Authentication

The standards in this section pertain to security authentication.

Public Key Certificates (X, 509v3)

- Public key authentication must be based on Public Key Certificates.
- Public Key Certificates must be based on the X.509v3 standard.
- Despite the widespread acceptance of this standard, care must be taken when dealing with vendors. Projects should require proof of interoperability with existing or proposed enterprise implementations using X.509v3 certificates. Proprietary extensions to certificates could inhibit interoperability and should be avoided.

9.13.3 Authorization and Access Control

The standards in this section pertain to authorization and access control.

Secure Sockets Layer Version 3 (SSLv3)

- SSLv3 is the most commonly supported protocol for communication between Web Server and browser.
- It authenticates the Web Server and optionally authenticates the user browser.
- Current implementation allow for client authentication support using the services provided by Certificate Authorities.

IP Protocol security extension (IPSec)

- IPSec is an extension to the IP communications protocol, designed to provide end-to-end confidentially for packets traveling over the Internet.
- IPSec works with both the current version of IPv4 and the new IPv6 protocol. IPSec has two modes: sender authentication and integrity but not confidentiality through the use of an Authenticating Header (AH), and sender authentication and integrity with confidentiality through the use of an Encapsulating Payload (ESP).

Cryptography must be based on open standards.

- Cryptographic services identified in this document are based on open, industry accepted, standards.
- The following business requirements and associated cryptographic standards have received wide acceptability and can be found in most products. Only full strength cryptography should be used. For example browsers are often supplied with weakened versions such as 40-bit DES, RC2 and RC4. Only browsers with full strength keys should be used for transactions involving the state. Cryptography with variable length keys should use a minimum key length equivalent to 56-bit DES.

Cryptography Algorithm	**Standards**
Public Key / Private key	RSA (1024 bit keys), ECC (160 bit keys)
Secret key	DES, 3-DES, RC2, RC4, IDEA, CAST (minimum DES equivalent or full length keys)
Message Digest	MD5, SHA-1

Table 9.4

Use S/MIME version 3 for securing e-mail communications.

- S/MIMEv3 provides a consistent way to send and receive secure email including MIME data.
- S/MIME defines a protocol for encryption services and digital signatures.
- Email clients should be evaluated for support of the standard and for interoperability.

Services provided through the Internet (Web-enabled applications, FTP, Mail, News, DNS etc) must be placed on the DMZ or proxied from DMZ.

- Applications services must be protected from unwanted external access and must be located on a DMZ or peroxide from the DMZ.
- All communication from servers on the DMZ to internal applications and services must be controlled.
- Remote or dial-in access to the enterprise must be authenticated at the firewall or through authentication services placed on the DMZ.

9.13.4 Directory Services

Use the statewide directory services infrastructure.

Using the statewide directory services has several benefits:

- The infrastructure is simplified by providing a common interface to view and manage all available resources.
- Directory services are a critical component to statewide initiatives like E-mail and electronic commerce. The current enterprise directory is fault tolerant and highly available from any location that participates. Time, distance, and location do not restrict access to the information contained within the services.
- Coordinated directory services will improve communication between our applications, databases, and network operating systems by providing consistent, reliable information in an efficient and effective manner.

Integrate homogeneous directories into a single tree.

- It is more efficient to link "like" directories into a single tree. Most vendors of directories have implemented either the standards that currently exist or standards that have been proposed. These standards include a mechanism to connect their directories together to built a single tree. This provides the optimum integration of Public Department resources and people without regard to location. A

single tree minimizes infrastructure costs while maximizing the potential for departments to choose how they share resources with other departments including local governments. The singletree approach also allows for improved fault tolerance and better performance especially for departments with geographically dispersed operations. Joining a tree, regardless of manufacturer, must be coordinated with Information Technology Services.

- It is necessary to tie these single trees from various manufacturers to the authoritative enterprise directory in order to provide authentication services to the authoritative enterprise directory. However, achieving connectivity from one manufacturer's directory to another is complex and difficult. For example, tying one Netscape directory to Novell's NDS can be done but is difficult to implement and maintain. The state currently has dozens of Netscape directories in place. The process would then need to be performed for each of them. However, tying all Netscape directories together into a single tree is fairly straightforward and is tied to the one NDS tree. It is a complicated task but is performed once. Through the Enterprise Directory Services Initiatives, this interoperability between dissimilar directories will be implemented. This will be accomplished through the use of meta-directory technologies.

Use the GUJSB (GUJarat Service Broker (GUJSB) services for directory functions.

- As in-house applications are developed, we must make use of the services that are already available rather than to constantly build new ones. An enterprise directory services infrastructure provides an addressable security service for authentication and authorization as well as repository for digital certificates.
- These services are addressable directly from the enterprise directory or through a service via the GUJarat Service Broker. For more information about GUJarat Broker, refer to the Component ware Architecture.

Use the Centralized Metadata Repository directory schema attributes and object classes.

- A directory is basically a database that has tuned to perform massive reads and frequent writes. Like other databases in our enterprise, directories and their elements must be centralized. For example, where a person object class may have an attribute of "Pager Number". "Pager Number" should be registered in the Centralized Metadata Repository and populated according to that

definition. Therefore, when the directory is queried for that information, the data returned will be as expected. In the past there has been a tendency to populate currently unused directory attribute with data that is not consistent with that attribute. For example, there may be a requirement to enter a pager number in the directory for a user. If there is no attribute for "Pager Number", there may be a tendency to select an attribute that is unused such as "Title". Instead, extend the schema to include a new attribute that precisely defines the data that will be placed there and register it with the Centralized Metadata Repository. Do not store inconsistent information in an unused attribute.

Populate directory objects according to the minimum attribute defined in Distributed Computing Standards and Guidelines.

- Any data source is only as good as he data it contains. If that data is missing, incorrect, or incomplete, the data source cannot be depended upon as an authoritative source for that type of information. A directory is no different. Directories have become much more than an authentication point for network users. In order to supply information on our users, network devices and organizations, directories must be built in as complete and reliable manner as possible.

Use Lightweight Directory Access Protocol version 3 (LDAPv3) for directory access where strong security is not required.

- LDAPv3 is the industry standard lightweight access protocol and does not offer strong authentication or access controls. However, LDAPv3 can provide standards based access to directories for lookups, as a communication mechanism for synchronization tools, public key retrieval, and others. Commercial off-the-self (COTS) applications often require their own directories. Access to the application directory from outside or for the application to communicate with an external directory will require standards based approach. Therefore, when purchasing COTS applications LDAPv3 compatibility is required. LDAPv3 also provides standards based access to the directory for lookups, as a communication mechanism for synchronization tools, public key retrieval, and others.

9.14 System Management Architecture

9.14.1 Operations Management

The following standards have been established to support operational systems management for the enterprise.

Use SNMPv1 (simply called SNMP) protocols.

- The Simple Network Management Protocol (SNMP) is a group of Internet protocols that is the standard for managing TCP/IP based networks.
- It is built into the devices (e.g. concentrators and routers) in the network and in the network operating systems of the servers and workstations.
- The network management system uses SNMP to collect statistics and other information on network devices.
- SNMP is also used to send commands that control the state of network devices.

Use Remote Monitoring (RMON) products.

- RMON products are predicted to become increasingly used in most enterprise networks.
- RMON products provide packet collection, decoding and analysis to the MAC layer of the Operating System Interconnection (OSI) stack using a combination of consoles and hardware and software probes that relied on SNMP MIB data collections.
- In 1992, the Internet Engineering Task Force, IETF, specified the RMON1 standard in RCF1271. The RON1 MIB extends SNMP capability by monitoring sub-network operation and reducing the data collection burden on management consoles and network agents.
- He RMON2 standard was approved by the IETF in January 1997 in RCF2021. RMON2 includes a new MIB to extend network monitoring into the application-monitoring layer.
- RMON functionality is growing to include functions like applications monitoring, report generation and bandwidth allocation.
- All major network device vendors have added RMON MIB collection capability to their products, although the depth of implementation relative to the full RMON specification varies among vendors and products.

Conform to the Desktop Management Interface (DMI) standard.

- The DMI standard was developed by the Desk Top Management Task Force (DMTF), which sets specifications for the management of the desktop environment.

- The DMI is a set of API's that allow different vendor applications to consistently share the desktop.
- It sets the standard for a management platform that enables a common standardized mechanism for systems management of the desktop while permitting vendor differentiation.
- As vendors build desktops with embedded DMI standards, important desktop management information will become available from the newer desktop units.

9.15 Generality Model

9.15.1 IIMS Context

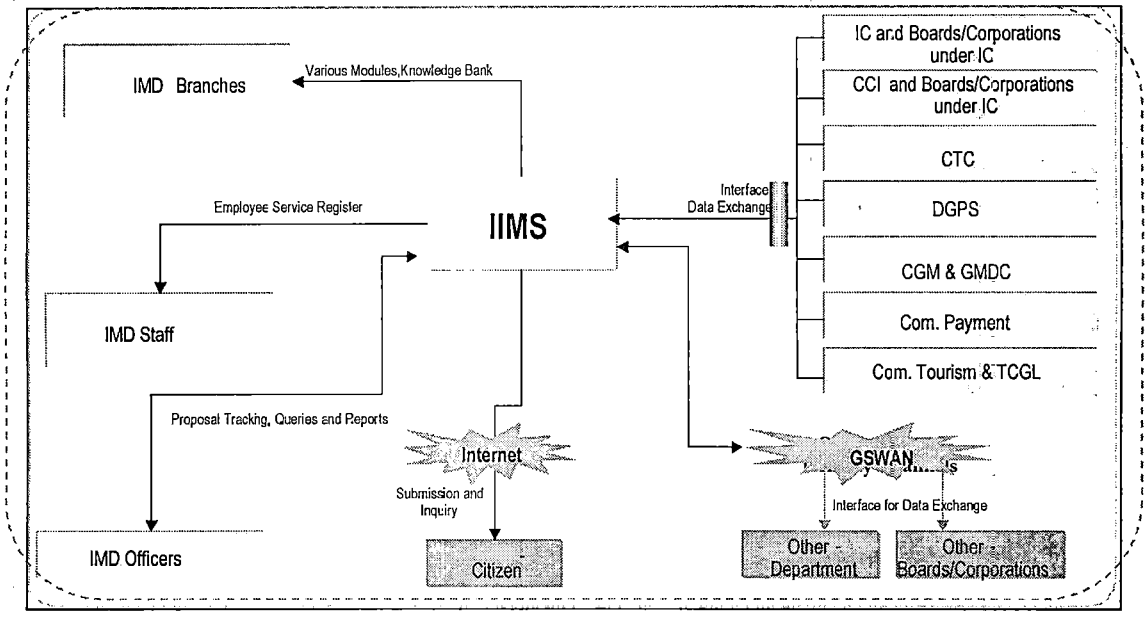

Figure 9.1

9.15.2 IIMS Functional Architecture

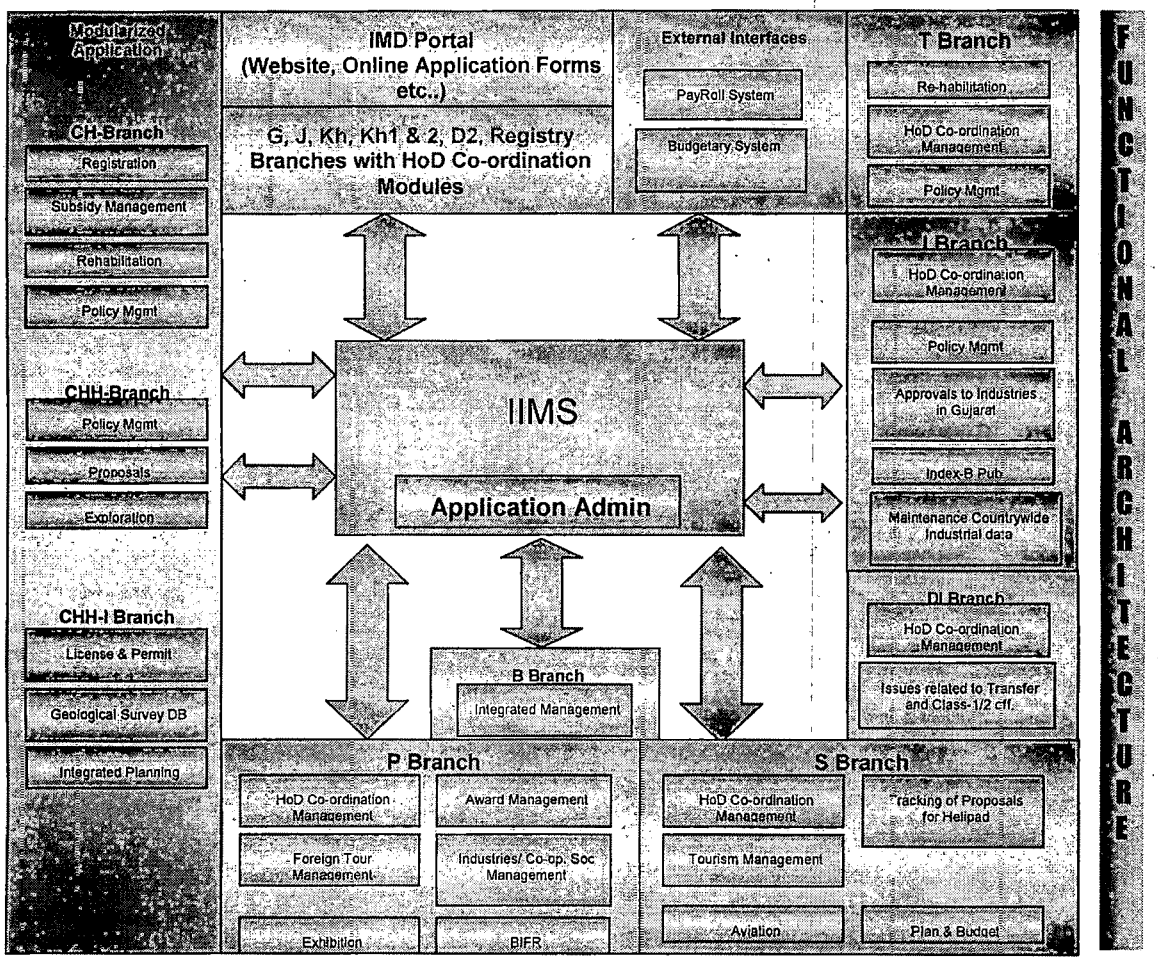

Figure 9.2

9.15.3 IIMS System Architecture (Overview)

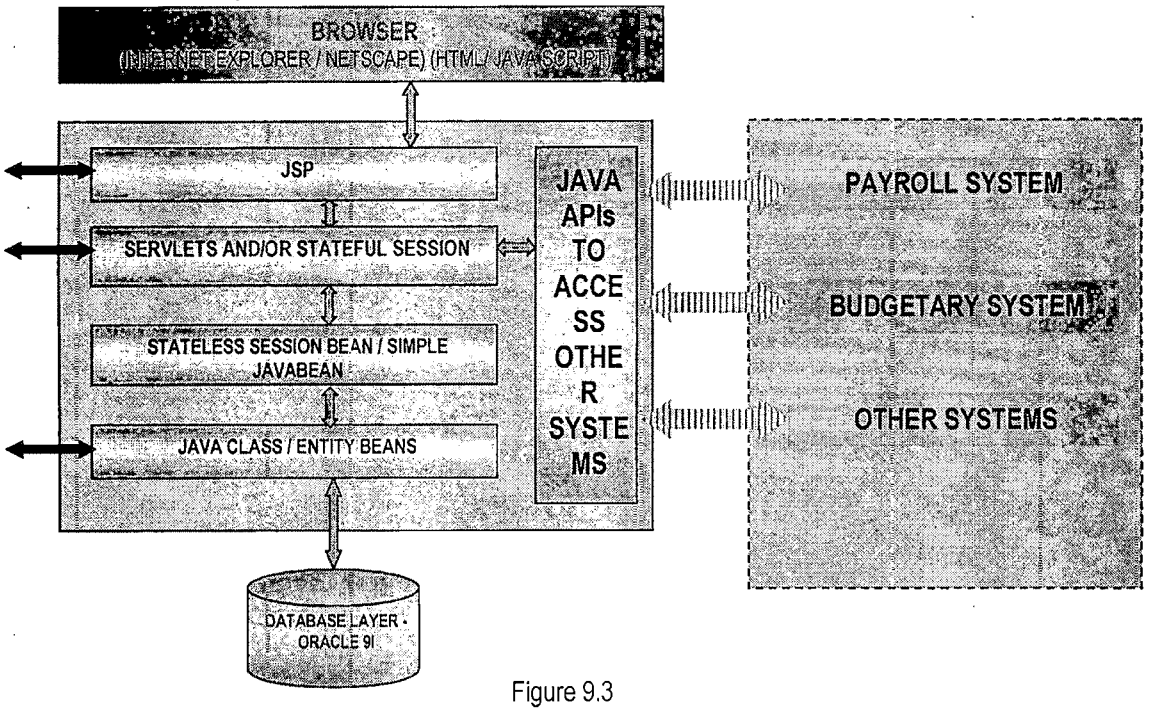

Figure 9.3

9.15.4 IIMS Software Components

Parameters	Platforms	
	Component	Solution
System Architecture	n-tier	3 tier web based
	J2EE Complaint	Yes
Front end tools		Internet Explorer
Back end tools	Application Server	Oracle 9iAS - Enterprise Edition
	Database	Oracle 9iDB – Enterprise Edition
	Backup & Recovery	Included
	Reports	Included
Programming Languages		Java
		SQL, PL/SQL
Commercial Software	Transliteration	CDAC n-trans, Indica, Trimax
		(Evaluation after requirement study)
Business Intelligence		Custom built.

Table 9.5

9.15.5 IIMS Security Architecture

- Application Security
  - Authentication - certificate based and host based authentication
  - Audit, Encryption, Access control
- Database Security
  - Server-Enforced, Row level Access Control
  - Comprehensive Auditing, Audit trial
- Network - Data Transfer Security
  - Encryption of Critical Data in Transmission
  - Selective database encryption feature
- External Security
  - Firewalls, Anti-virus Kits
- Other Security
  - Backup and Recovery Strategy, Disaster Recovery

9.15.6 IIMS Reengineering of Systems & Procedures

- Multi disciplinary team for the study
  - Consultant holding expertise in IT, Government, Networking and Internet Technology Specialists.
- Interaction with major stakeholders of the IMD
- Holistic and consensual approach

9.15.7 IIMS Identify 'as-is' & 'to-be'

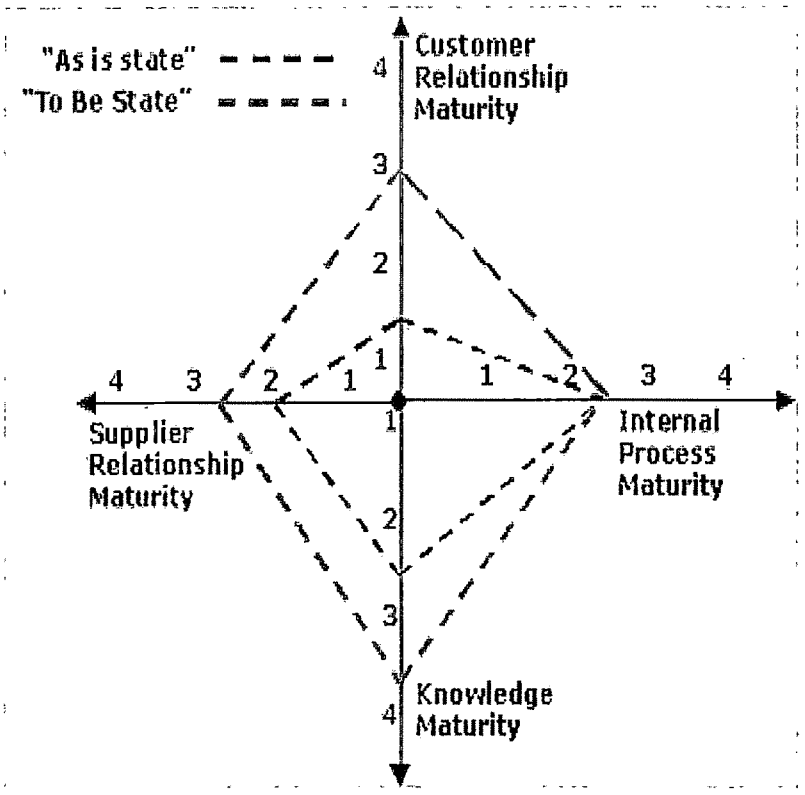

Figure 9.4

9.15.8 IIMS Reengineering of Systems & Procedures

- Familiarization → Data collection → Data aggregation→ System Perspective + Business Perspective

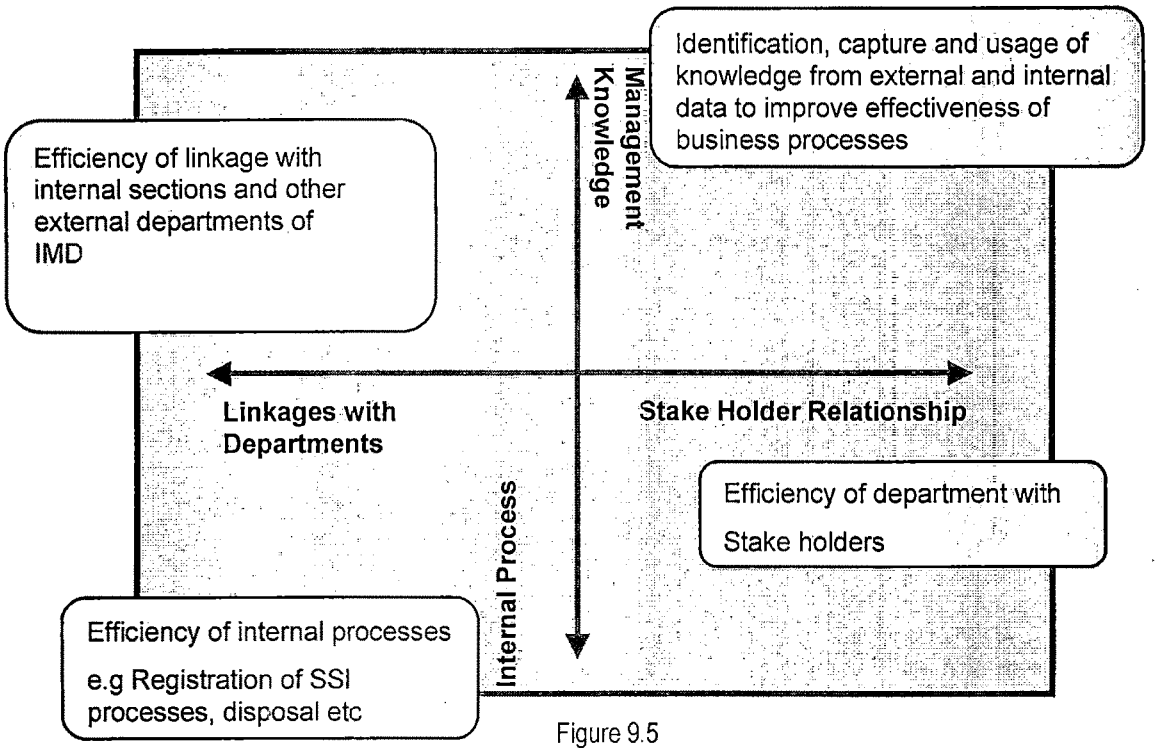

Figure 9.5

9.15.9 IIMS Customized S/W Development

- Customised software development; Activity to be performed phase wise:
  - System Study / Requirements analysis / Reengineering phase
  - Prototype screens for entry/query/reports
  - Design Phase (High level & Low level design)
  - H/W and Network capacity sizing
  - Development of various Modules
  - Unit, System and Integration Testing
  - Acceptance Testing of Integrated Software
  - User Training for Integrated Software
  - Pilot implementation
  - Roll Out of the Integrated Software
  - Post Implementation support and maintenance

9.15.10 IIMS Development and Maintenance Phases

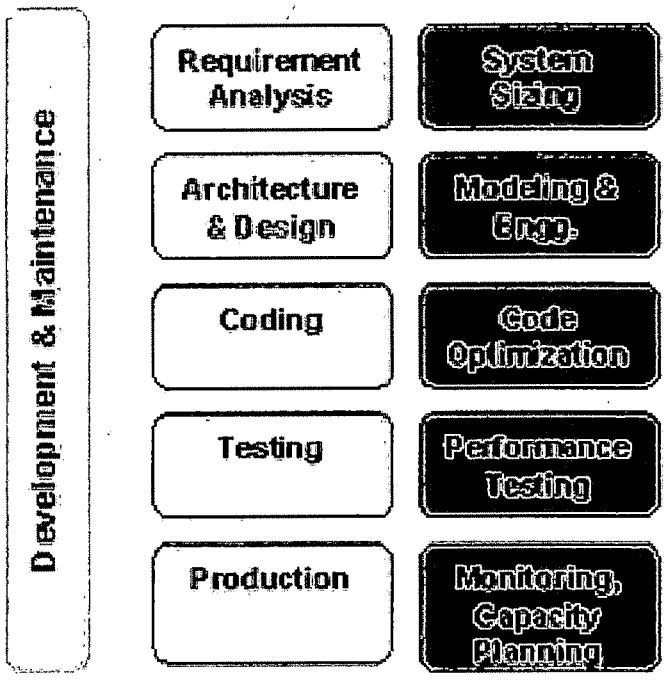

Figure 9.6

9.15.11 IIMS Quality Assurance and Maturity

Issues & Resolution

➢ Compartmentalized information	• Single Interface
➢ Multiple User Interfaces	• Single Sign-on Password
➢ Multiple repositories	• Single Directory service providing User Authentication
➢ Complex Information Exchange	• Enterprise Integrated Solution
➢ Repetitive activities	• Workflow, Automated Task/Reports
➢ Difficult to track files	• Inward and Outward Tracking
➢ Delay in information collection/communication	• SLA could also be defined, Push Communications
➢ Historical Data	• Data entry screens and Interfaces

Table 9.6

Salient Features of IIMS

- Workflow automation solution
- Monitor information flow and provide **load balancing** in case of large volume of requests.
- Architecture based on open, inter-operable standards.

- The architecture will be scaleable and capable of delivering high-performance in varied conditions.

- The architecture and design features to enable **adoption of multiple delivery channels**
  - Departments, Internet, Intranet, kiosks etc.Exhaustive and Reliable Data Storage

Gujarati Interface

- The proposed solution includes Screens/Reports in English as well as Gujarati
- User customizable option for selecting the language
- TransliterationUsing Dictionary of 25,000 Words
  - Would help in generating reports in both English as well as Gujarati with the Fonts used by IMD.
  - Standard phrases transliteration/ Screens in Gujarati and English wherever required

Project Management

- Planning
  - Project Plan
- Organization
  - Steering Committee
  - Consultant Project Leader
  - Consultant Technical Team
  - Consultant Functional Team
  - IMD Project Manager
  - IMD Technical team
- Periodic Meetings
- Monthly Progress Reports

With n-tier architecture, one can have multiple applications using a common set of business objects across an organization. This promotes the standardization of business practices by creating a single set of business functions for the entire organization to access. If a particular business rule changes, then changes have to be made to only the business object and if need, to its interface also