High performance web based Automated Dental Identification System

Ali Bahu
West Virginia University

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Recommended Citation
https://researchrepository.wvu.edu/etd/4439

This Thesis is protected by copyright and/or related rights. It has been brought to you by the The Research Repository @ WVU with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you must obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This Thesis has been accepted for inclusion in WVU Graduate Theses, Dissertations, and Problem Reports collection by an authorized administrator of The Research Repository @ WVU. For more information, please contact researchrepository@mail.wvu.edu.
High Performance Web Based
Automated Dental Identification System

By

Ali Bahu

Thesis Submitted to
The College of Engineering and Mineral Resources
West Virginia University
in partial fulfillment of the requirements for the degree of
Master of Science
in
Electrical Engineering

Dr. Yenumula V. Reddy, Ph.D., Chair
Dr. Hany H. Ammar, Ph.D.
Dr. Arun Ross, Ph.D.

Lane Department of Computer Science and Electrical Engineering
Morgantown, West Virginia
2009

Keywords: Dental Identification, ADIS, 3-Tier Architecture, Web Application, Parallel Processing, Grid Based Design, Law Enforcement.

Copyright 2009 Ali Bahu
ABSTRACT

High Performance Web Based Automated Dental Identification System

Ali Bahu

Forensic scientists are using different kinds of systems and techniques for the identification of missing and deceased individuals. Researchers from West Virginia University, Michigan State University, and University of Miami proposed an Automated Dental Identification System (ADIS) which utilizes dental features for the post mortem identification process. A web-based ADIS, namely Web-ADIS, was developed. The initial testing shows that the system can be used efficiently with a small database.

A new and improved system design has been proposed in this research enabling practical use of the ADIS with a large database. We designate this new system as High Performance Web Automated Dental Identification System Grid (HP Web-ADIS Grid). The ADIS tool uses a set of default databases provided by FBI and other law enforcement agencies for identification and matching. However, certain situations like a plane crash, earthquakes, and tsunamis etc. require special reference databases. The proposed system design provides users with the ability to upload their own reference databases for matching purposes. Analysis of the previous work done on ADIS tool revealed the need for performance enhancement. We observed that the Image Comparison component is very slow and time consuming. To improve system performance, a parallel processing grid based system is proposed. The HP Web-ADIS Grid design utilizes multiple machines to process requests resulting in significant reduction in the processing time. The performance testing of HP Web-ADIS was conducted by comparing a system utilizing a single node with that of the proposed grid based configuration. Results obtained show an exponential improvement in the performance of ADIS significantly reducing the overall time for performing record to record comparisons. Forensic scientists would benefit greatly from this improved and highly efficient system enabling faster retrieval of results.
DEDICATION

TO THE BEST
OF THE PLANNERS

&

TO THE OPPOSING WIND
THAT TAKES US HIGHER
ACKNOWLEDGEMENTS

If there is anything good in this research, it is from God almighty and I am responsible for all the mistakes and shortcomings.

I wish to express my appreciation to my advisor Professor Dr. Hany Ammar for his continuous support over many years of my graduate studies and I appreciate his invaluable trust, patience and tolerance towards me and my shortcomings. I am thankful to him for his constant encouragement and his objective criticism. I am obliged to him for having tolerated my constant stream of questions and unscheduled interruptions for advice. His support as a guide has made me a better person in many aspects of life.

I am also very grateful to Dr. Yenumula V. Reddy and Dr. Arun Ross for being valuable members of my examination committee and for being helpful and skillful teachers.

I ought to thank all my WVU research colleagues: Dr. Diaa Eldin Nassar, Satya Chekuri, Dr. Ayman Abaza, Dr. Eyad Haj-Said, and Usman Qurashi. I am very much indebted to all my great friends Sohail Chaudhry, Saleha Javaid, Tayyeb Anwar, and Ahad Ali Syed for their great support and valuable advice.

My family has been a pillar of support throughout my life. My parents and siblings have always offered the moral support necessary to perform well and I am indebted to them for standing by my side in times of need. I am thankful to my father Mr. Qamar Moheuddin, my mother Mrs. Rashida Qamar, my brother Zeb, and my sister Summiya.
SUPPORTING PUBLICATIONS


# TABLE OF CONTENTS

Chapter 1 **INTRODUCTION** .................................................................................................................. 1

1.1 OVERVIEW .......................................................................................................................... 1
  1.1.1 Automated Dental Identification System (ADIS) .................................................... 2
  1.1.2 Web-ADIS using Parallel & Grid Computing ....................................................4

1.2 PROBLEM STATEMENT ..................................................................................................... 9
  1.2.1 A Use Case Scenario for Web-ADIS .............................................................. 11

1.3 RESEARCH OBJECTIVES ................................................................................................. 13

1.4 THESIS ORGANIZATION ................................................................................................. 15

Chapter 2 **LITERATURE REVIEW** ........................................................................................... 17

2.1 TYPES OF FORENSIC IDENTIFICATION SYSTEMS ...................................................... 17

2.2 DENTAL IDENTIFICATION SYSTEMS ........................................................................... 18
  2.2.1 History ............................................................................................................... 18
  2.2.2 Dental Identification Process ............................................................................. 19
  2.2.3 Computer-Assisted Postmortem Identification (CAPMI) .................................. 21
  2.2.4 Win-ID Identification System .......................................................................... 23
  2.2.5 Automated Dental Identification System (ADIS) ............................................. 25

2.3 TECHNOLOGIES INCORPORATED FOR THE SYSTEM DESIGN 26
  2.3.1 Java 2 Platform, Enterprise Edition (J2EE) ......................................................... 26
  2.3.2 JavaServer Pages (JSP) ...................................................................................... 28
  2.3.3 Apache Tomcat .................................................................................................. 30
  2.3.4 Java Database Connectivity (JDBC) ................................................................... 30
  2.3.5 JavaScript .......................................................................................................... 32
  2.3.6 Java Native Interface (JNI) .................................................................................. 32
  2.3.7 JavaBeans(TM) .................................................................................................. 33
  2.3.8 Structured Query Language (SQL) ................................................................. 34
  2.3.9 Oracle 10g ......................................................................................................... 34
  2.3.10 Others .............................................................................................................. 35
Chapter 3 **HP WEB-ADIS: DESIGN & COMPONENTS** .................36

3.1 INTRODUCTION ..................................................................................36

3.2 HP WEB-ADIS APPLICATION SERVER ........................................36

3.3 LOAD BALANCING SERVERS: .........................................................37

3.4 WEB APPLICATION: .................................................................38
   3.4.1 Web-ADIS Authentication: ......................................................40
   3.4.2 Identification Interface: ..........................................................42
   3.4.3 Request History Interface: ......................................................46
   3.4.4 Preprocessing Results Interface: ..........................................48
   3.4.5 Candidate List Interface: .......................................................49
   3.4.6 Match List Interface: ............................................................50
   3.4.7 Configuration Interface: .........................................................51

3.5 HP WEB-ADIS GRID .................................................................52
   3.5.1 Database server: .................................................................54
   3.5.2 Preprocessing/Potential Matching Servers (PP/PMS): ............54
   3.5.3 Image Comparison Servers (ICS): .........................................55

3.6 REQUEST SCHEDULING AND DISPATCHING SERVICE ..........55

3.7 PRE-PROCESSING/POTENTIAL MATCH SERVERS ..................57
   3.7.1 Preprocessing Component: .....................................................57
   3.7.2 Reference Database Pre-processing: ....................................63
   3.7.3 Potential Match Search Component: ....................................64
   3.7.4 Candidate Splitter: ...............................................................65
   3.7.5 HP Web-ADIS Emailer: .........................................................66
   3.7.6 Match List Merging Service ..................................................67

3.8 IMAGE COMPARISON SERVERS ..................................................70
   3.8.1 Image Comparison Component: ..........................................70

3.9 DATABASE SERVER .................................................................73
   3.9.1 User Info Database ..............................................................73
   3.9.2 Image/Features Database .....................................................74

Chapter 4 **SYSTEM TESTING & RESULTS** .................................77

4.1 TESTING SCENARIOS FOR THE WEB APPLICATION ..........77
LIST OF FIGURES AND ILLUSTRATIONS

Figure 1.1: ADIS Block Diagram [3] ................................................................. 4
Figure 1.2: ADIS Grid Architecture Block Diagram ........................................ 7
Figure 1.3: HP Web-ADIS Grid Design ............................................................. 8
Figure 2.1: Dental Ante Mortem (AM) Form ..................................................... 20
Figure 2.2: OMR form for use with the CAPMI system at a disaster scene [5] ...... 22
Figure 2.3: Ante mortem dental record used by WinID [6] .................................. 24
Figure 2.4: Automated Dental Identification System [3] .................................... 25
Figure 2.5: Java 2 Platform, Enterprise Edition Block Diagram [7] .................... 28
Figure 2.6: Java Database Connectivity (JDBC) Block Diagram ....................... 31
Figure 2.7: Java Native Interface (JNI) Block Diagram ................................... 33
Figure 3.1: HP Web-ADIS Application Server with load balancing .................. 37
Figure 3.2: HP Web-ADIS Application Interface ............................................. 38
Figure 3.3: HP Web-ADIS Pass Code Authentication Interface ........................ 39
Figure 3.4: HP Web-ADIS Pass Code Authentication ....................................... 39
Figure 3.5: HP Web-ADIS Create New Account ............................................. 40
Figure 3.6: HP Web-ADIS Authentication ...................................................... 41
Figure 3.7: HP Web-ADIS Home Interface .................................................... 42
Figure 3.8: HP Web-ADIS Identification Interface/ Choose DB ......................... 43
Figure 3.9: HP Web-ADIS Identification Interface/ Choose DB Dropdown Menu .... 43
Figure 3.10: HP Web-ADIS DB upload Interface ............................................ 43
Figure 3.11: HP Web-ADIS Identification ........................................................ 44
Figure 3.12: HP Web-ADIS Identification with reference DB Upload ................. 45
Figure 3.13: HP Web-ADIS Request History Interface ..................................... 46
Figure 3.14: HP Web-ADIS Request History .................................................. 47
Figure 3.15: HP Web-ADIS Preprocessing Details Interface ............................ 48
Figure 3.16: HP Web-ADIS Preprocessing Details Interface with Preview Option .... 49
Figure 3.17: HP Web-ADIS Candidate List Interface ...................................... 50
Figure 3.18: HP Web-ADIS Match List Interface ............................................. 50
Figure 3.19: HP Web-ADIS Configuration Interface ....................................... 51
Figure 3.20: HP Web-ADIS Grid Design.................................................................53
Figure 3.21: HP Web-ADIS Grid Design.................................................................54
Figure 3.22: HP Web-ADIS Request Scheduling & Dispatching Service..............56
Figure 3.23: HP Web-ADIS Preprocessing Block Diagram.....................................58
Figure 3.24: Preprocessing Step1: Cropping Component Block Diagram ..............59
Figure 3.25: Preprocessing Step2: Film Enhancement Component Block Diagram ....59
Figure 3.26: Preprocessing Step3: Film Type Classification Block Diagram..........60
Figure 3.27: Preprocessing Step4: Segmentation Component Block Diagram .........61
Figure 3.28: Preprocessing Step5: Contour Extraction Component Block Diagram ....62
Figure 3.29: Preprocessing Step6: Teeth Labeling Component Block Diagram .......63
Figure 3.30: ADIS Potential Match Search Component Block Diagram [3] .............65
Figure 3.31: HP Web-ADIS Emailer Service’s Email Completion of First Step .........66
Figure 3.32: HP Web-ADIS Emailer Service’s Final Email ..................................67
Figure 3.33: HP Web-ADIS Match List Merging Service......................................68
Figure 3.34: HP Web-ADIS Preprocessing/Potential Match Servers ....................69
Figure 3.35: Image Comparison Component Block Diagram [3] ...........................71
Figure 3.36: HP Web-ADIS Image Comparison Servers.......................................72
Figure 3.37: HP Web-ADIS Image/Feature Database Structure ............................76
Figure 5.1: HP Web-ADIS Overall Subsystem Design ........................................85
Figure 5.2: HP Web-ADIS Web Application Subsystem Design .........................86
Figure 5.3: HP Web-ADIS Preprocessing/Potential Match Subsystem Design ......88
Figure 5.4: HP Web-ADIS Detailed Subsystem Design with Interactions ..............90
Figure 5.5: HP Web-ADIS Deployment Diagram ..................................................91
Figure 6.1: HP Web-ADIS Grid Performance Comparison ....................................94
Figure 6.2: HP Web-ADIS Grid Performance Comparison ....................................94
Chapter 1 INTRODUCTION

1.1 Overview

In the current age of science and technological advancement, humanity has gained and excelled in all aspects of life i.e. from transportation to communications, health care to astronomy, biometrics to robotics and the list goes on and on. The concept of biometrics based system was almost nonexistent about 25 years ago but now it is one of the core computer science areas in research and development. Biometrics systems are not only being used for access control in secure and sensitive environments but also for the forensic identification of missing and deceased individuals. Forensic scientists are using different kinds of identification systems and techniques to identify missing/deceased individuals using the post mortem data and other biometric features. Commonly used techniques include fingerprinting, face recognition, iris recognition, etc. These identification techniques cannot be used in situations where a subject’s body has decayed over time or has undergone a lot of tissue damage. DNA recognition is another technique but is not widely used as it is very expensive and is not well established in the civil sector. Dental records are the most promising and most reliable features for such scenarios where there is significant tissue damage to the body and other features are lost due to decay.

In the recent history, dental records have been used for identification purposes all over the world. For example in 9/11 attack, many individuals were identified using only parts of the jaw bones. Forensic dental experts, by manually matching the dental records, were able to identify about 20% of the 973 victims during the first year after 9/11 [10]. It is important to note that the manual dental identification process of 2,749 WTC (World Trade Center) victims [11] took around 40 months [10] to complete. Among the 200,000 Asian tsunami victims [12], 75% of the deceased individuals were identified using the manual dental identification and only 0.5% of the individuals were identified using DNA [12]. DNA matching is extremely expensive, can be time consuming and is therefore, not a good option for such mass disasters. These examples clearly indicate that dental
identification is the most efficient and effective technique for post mortem identification in mass disaster scenarios like Asian tsunami or WTC attacks. Although dental identification techniques are quite effective and accurate, the manual identification process can be extremely time consuming. Forensic scientists have to search the whole database of dental records in order to come up with a match list. The only bottleneck in manual dental matching is its inefficiency in terms of time required for the entire process. This triggers the need for an automated dental identification system that can cut down the number of dental records for the forensic scientists to review, eliminating the only obstacle in the success of the dental identification process.

1.1.1 Automated Dental Identification System (ADIS)

In 1997, the Criminal Justice Information Services Division (CJIS), a sub-division of Federal Bureau of Investigation (FBI), created a Dental Task Force (DTF). The objective was to improve and enhance the utilization and effectiveness of the National Crime Information Center's (NCIC) Missing and Unidentified Persons (MUP) files. The DTF recommended establishing a Digital Image Repository (DIR) [23] of dental records and an Automated Dental Identification System (ADIS) in order to improve and enhance the utilization and effectiveness of the MUP files. The vision behind the proposed Automated Dental Identification System [24] was to design a tool that can effectively and efficiently reduce the number of candidate records in the match list for a specific subject of interest so that the forensic scientists do not have to traverse the whole DIR in order to come up with a match. To accomplish this extremely difficult and challenging task of designing and developing an automated dental identification system, three teams from West Virginia University, University of Miami, and Michigan State University proposed a system comprising of the following components, also shown in figure 1.1:

- **Record Preprocessing**
  - Record Preprocessing component is subdivided into five major modules:
    - Cropping the dental records into dental films.
    - Grayscale Contrast Enhancement (GCE) of the individual films.
INTRODUCTION

- Classification of the films into Bitewing, Periapical, or Panoramic views.
- Segmentation of each tooth from the films.
- Teeth contour extraction.
- Labeling each tooth according to their location in dental chart labeling.

- Potential Matches Search

Potential Matches Search component comes up with a short candidate list of dental records after matching the subject record with the records in the DIR. The potential match search algorithm uses the high-level dental features i.e. number of teeth, tooth shape, and other related features to come up with a shorter candidate list of records. This algorithm is extremely quick and the main reason behind adding this component is that the image comparison component takes a lot of time to process. It is extremely inefficient to use the image comparison component on the whole DIR. Therefore a short candidate list using the potential match search algorithm is generated resulting in increased performance and efficiency.

- Image Comparison

Image Comparison component uses tooth-to-tooth comparison between subject tooth after alignment and the corresponding candidate tooth for all the visible teeth in the record. This process in repeated for each entry in the candidate list generated by the potential matcher resulting in a short Match List. In short, ADIS uses high-level features for fast retrieval of a candidate list produced by the potential matches search component and then refines the candidate list using low-level image features. This results in a short Match List that can be sent to the forensic scientists in order to come up with an exact match.
1.1.2 Web-ADIS using Parallel & Grid Computing

The vision behind designing and developing the Automated Dental Identification System was to quicken the identification process and to make the system accessible to forensic scientists from their own homes and offices without any hassle. We were successful in designing an automated system for dental matching, but the system lacked many vital features and was unable to fulfill the most important requirements i.e. availability,
INTRODUCTION

accessibility, portability, and speedy processing. In order to fulfill these requirements a web based environment was required that can be accessed using the web browser such that the ADIS components’ functionality is used seamlessly. Also, the successful application of the system required support for multiple users simultaneously. We have designed a web based application accessible through standard browser applications supporting multiple users.

Another major problem noticed during the analysis of previous work done on ADIS was the need for performance enhancement. For example we noticed that the Image Comparison Component running on a single machine can take around four hours to match one subject record to a candidate list of ten records. It takes about 20 minutes for record to record comparison. The image comparison component took around 17 hours to search a database of 50 records. We measured the processing time for running the image comparison component using MATLAB implementation on a 2.66 GHz 1.0 GB RAM Intel Pentium IV PC platform [3]. To address the problem of slow and time consuming matching process, a grid based design has been purposed. The proposed design uses a grid of servers working in parallel to preprocess the requests, prepare candidate lists using potential matcher, and finally generate match lists after performing image comparison on the subject and reference records provided in the candidate list. The utilization of multiple servers, processing multiple requests at the same time proved to be extremely effective and the system was able to achieve the requirements of high performance, efficiency, and parallelism. The new and improved system is designated as High Performance Web Automated Dental Identification System Grid (HP Web-ADIS Grid). The block diagram of the HP Web-ADIS Grid architecture is shown in figure 1.2. The end to end functionality of the system is shown in figure 1.3. The summary of web based grid structure for ADIS, its main components, and their functionality is given below:

- **Web Application Server**
  A system that is responsible for accepting HTTP requests from clients (subject records in our case) using web browsers, and serving them HTTP responses along with data contents. Data contents usually are web pages such as HTML, JSP
documents, and linked objects like images (dental records in our case), preprocessing details, candidate, and match lists.

- **UserInfo Database**
  The request details like subject record type, database to match with, either the user uploaded the reference database zip file or not, request date and time, username of the requester, and other related information entered by users using the web interface are saved on the oracle database. This saved information is then used by the Preprocessing/Potential Match Servers (PP/MS) and the Image Comparison Servers (ICS) to process the request.

- **Image/Feature Database**
  The uploaded subject records are saved on this database. This database system is also used to maintain the default databases provided by FBI and other law enforcement agencies and the databases (reference records) uploaded by users. This database basically contains the records (images) and the corresponding preprocessed files (generated by preprocessing component), containing all the information that is required by the potential match and image comparison components in order to come up with candidate and match lists.

- **Preprocessing/Potential Match Servers (PP/PM Servers)**
  These servers obtain the request information saved on the UserInfo database and use this information to retrieve the uploaded subject record on the Image/Feature database server for record preprocessing (each server can process one request at a particular time). The reference database (if uploaded) is also preprocessed by this server. After the completion of preprocessing, the PP/MS run the potential match component which generates a candidate list. PP/MS then split the candidate list for the Image Comparison servers. The task assignment details for each IC Server are updated in the UserInfo database.
INTRODUCTION

- **Image Comparison Servers (ICS)**
  These servers retrieve the image comparison requests from the UserInfo database, the corresponding subject record, and the candidate list from the Image/Feature database and generate the final match lists for the user to view.

*Figure 1.2: ADIS Grid Architecture Block Diagram*
INTRODUCTION

Figure 1.3: HP Web-ADIS Grid Design
INTRODUCTION

1.2 Problem Statement

This thesis research addresses the development of a new and improved system design for the Automated Dental Identification System tool. After careful analysis and observation of the existing system we identify the following problems:

**Problem 1: Developing a user friendly web interface**
A web accessible user interface is required that seamlessly interacts with different ADIS components for processing the records and displaying the results of each and every step of the matching process. These include Record Preprocessing, Potential Match Search and Image Comparison components. The users should be able to access the web interface using a simple browser application running on any platform. The web interface should also be secure enabling only authorized personnel to access the tool.

**Problem 2: Providing multiple user support**
It is required to make the web application available to multiple users simultaneously. Multiple users should be able to utilize the functionalities provided by the ADIS tool at the same time while allowing each user to submit multiple requests.

**Problem 3: Establishing request and record databases**
A database system should be developed that can maintain user’s request details and can also manage each node’s information and task assignment details. This problem also includes the design and maintenance of an Image/Feature database holding the record images and the preprocessed files containing the dental features’ details extracted using the preprocessing tool.

**Problem 4: Establishing user reference databases**
A web interface that can enable clients to upload their own reference databases is required. The reference databases are zip files containing dental records from a particular event like plane crash in January 2009, tsunami, etc. The system should
be able to preprocess the uploaded reference database in batch. It should also save the record images and the preprocessed files in the Image/Feature database to be used for matching purposes in future.

**Problem 5: Performance Enhancement**
A system design is required that can reduce processing time for each request by utilizing parallel processing and a grid based design. The system shall utilize ‘divide and conquer’ techniques to improve performance by using multiple nodes (individual systems) to process sub parts of the request.

**Problem 6: Developing a result notification system**
The processing time of each request can vary depending on the number of films in the dental records, quality of the dental records, etc. Therefore the need of a notification system is essential. A system is required that can notify the users by sending an email after the completion of each step of the matching process i.e. Record Preprocessing, Potential Match Search, and Image Comparison. The users should be able to view the intermediate results by logging into the web application.

**Problem 7: Developing a configuration interface**
ADIS components were designed by multiple teams from three universities. The teams worked on same components in parallel but used different image processing techniques for generating results. The users should be able to pick and choose different preprocessing, potential matching, and image comparison components using an easy to uses web interface before starting the processing procedure. Users should also be able to choose different realizations, parameters, parameter values, and other related variables for processing using this interface.
1.2.1 A Use Case Scenario for Web-ADIS

Next we discuss a use case scenario for the Web-ADIS system. This use case captures the basic functionality of the system and gives an idea of the fundamental system requirements.

- Login using username and password.
- Submit subject record, record type, and the reference database name to match with i.e. default databases from FBI and other law enforcement agencies. Users can also upload their own reference databases (.zip file containing records) for matching.
- The request information is then saved on the UserInfo database.
- The subject record is uploaded on the Image/Feature database.
- The Preprocessing/Potential matching servers that are always listening for the requests, start processing the requests by different users. One server processes one request at a time.
- After the preprocessing of subject record and the reference database (if applicable) is accomplished, the Preprocessing/Potential matching servers (PP/MS) start the potential matching component to come up with a candidate list.
- After the completion of potential matching routine, candidate list is generated.
- An email is sent to the user to inform him/her about the availability of preprocessing and potential matching results.
- Users can access the preprocessing results and candidate list at this time by logging on to the web application using their username and password information.
- Then the candidate list is divided into sub-candidate lists depending on the number of Image Comparison Servers connected to the system. Each sub-candidate list contains some records from the original candidate list.
- The Image Comparison Servers start processing the request and come up with a match list.
INTRODUCTION

- An email is sent to the user regarding the completion of his/her request.
- Users can access the match list at this time by logging on to the web application using their username and password information.
INTRODUCTION

1.3 Research Objectives

The research objectives to be accomplished by developing the HP Web-ADIS Grid are summarized as below:

- Design and development of a web application that can be accessible to forensic scientists from anywhere in the world through a standard browser. An application where users can access preprocessing results, candidate lists and the match lists for all their requests and can view the request history.

- Design and development of a system that can handle multiple users and can manage multiple requests from each user efficiently.

- Design and development of a parallel processing grid based system that can improve the performance and can reduce the processing time of each ADIS component by utilizing divide and conquer techniques.

- Design and development of a database system to manage the user accounts, their requests, request details, and also the task assignment details for each node in the grid.

- Design, development, and maintenance of a database system for the dental records and the features extracted from the dental records.

- Enabling the users to upload their own reference databases in the form of zip file using the web interface. Secondly, batch preprocessing of the uploaded records (in zip file) and archiving them in the Image/Feature database for future.

- Design and development of a mechanism that can notify the users by sending an email after the completion of each step of the matching process (Record
INTRODUCTION

Preprocessing, Potential Match Search and Image Comparison) so that user can view the results of each step by logging into the web application.

• Design and development of a web based tool that can enable the users to utilize different components designed by different teams, and different realization options, parameters, and parameter values according to their needs and preferences.
INTRODUCTION

1.4 Thesis Organization

A brief summary of the topics discussed in chapters are give below:

- **Chapter 1** is titled “Introduction” and gives a general introduction to ADIS, the parallel processing and the grid based design of the web based system. This chapter also sheds light on different problems that will be presented in this thesis.

- **Chapter 2** is titled “Literature Review” and is divided in to two main sections. First section discusses the history of dental identification and the second chapter presents the literature review of the different technologies that were incorporated for the design and development of the parallel processing grid based design for this web based system.

- **Chapter 3** is titled “HP Web-ADIS: Design & Components” and sheds light on all the components that work together for the multi user support and performance enhancement of the system.

- **Chapter 4** is titled “System Testing & Results” and presents all the problems that were encountered and how they were fixed. It also discusses Black box/White box testing, component level and end to end testing, web interface testing, etc. Performance enhancement comparisons and results are also discussed in this chapter.

- **Chapter 5** is titled “Unified Modeling Language (UML) Diagrams” and presents the UML diagrams of the grid based system.

- **Chapter 6** is titled “Performance Enhancement Results” and compares the performance enhancement results of the grid based system to a single node system.
INTRODUCTION

- **Chapter 7** is titled “Conclusions and Future Work” and concludes the thesis and gives some suggested future work as extension to our presented work.

- **Bibliography** is given at the end of this document.
Chapter 2 LITERATURE REVIEW

Section 2.1 and 2.2 of this chapter cover the introduction to forensic identification systems, history of dental identification, and types of dental identification systems that are being currently used. Section 2.3 discusses different technologies that were incorporated in the design and development of the web based application, as well as the techniques and tools used to accomplish parallel processing and grid based design.

2.1 Types of Forensic Identification Systems

A biometric system is a new concept and was almost nonexistent twenty years ago. It is an automated way of identifying individuals by using their physical and behavioral characteristics. Currently, biometric systems are being used widely for access control in secured and sensitive environments like airports, federal buildings, etc. Biometrics based systems are also being used for forensic identification of missing and deceased individuals. Forensic science is a scientific method of gathering and examining evidence at the crime scene or a location of interest. The process of forensic identification applies forensic sciences to identify individuals related to the crime or the incident by using the traces they leave behind at the crime scene. Cases are then solved by using data such as fingerprints, palm prints, tooth bite prints, blood, etc. and other case related evidence. Forensic identification can be classified as Post-Mortem (PM) identification and Ante-Mortem (AM) identification. AM identification techniques include but are not limited to DNA matching, fingerprints, face recognition, ear identification, etc. On the other hand, PM identification is carried out using all the techniques given above in the events of disasters like fires, earthquakes, tsunami, and other severe conditions and circumstances [3]. Fingerprinting, face recognition, and other such techniques cannot be effective in cases where time and severe weather conditions have led to decay in the soft tissues. Dental identification is the best and most effective option under such circumstances. Although, DNA matching seems to be more effective in such conditions but the problem with DNA based identification is that it is very expensive. Also, the matching process is
quite time consuming. Therefore, dental identification appears to be the most effective and efficient mode of Post-Mortem (PM) identification.

2.2 Dental Identification Systems

According to the American Board of Forensic Odontology guidelines [25], forensic scientists use the Post-Mortem (PM) and Ante-Mortem (AM) records for identification process. Following four conclusions can be drawn by utilizing the dental features identifiable in the records [25]:

**Positive identification:** The ante mortem and post mortem records match sufficiently and there are no unexplainable discrepancies.

**Possible identification:** The ante mortem and post mortem data have consistent features. However, identity cannot be established positively because of the quality of either the postmortem remains or the ante mortem evidence.

**Insufficient evidence:** The available information is insufficient to reach a conclusion.

**Exclusion:** The ante mortem and post mortem data are clearly inconsistent.

2.2.1 History

The history of using post mortem dental features for identification purposes goes back to 1776 when body of Dr. Warren, who was killed during the Battle of Breeds and was buried in an unmarked grave, was identified by Paul Revere using dental features. In 1850, a murder committed by Dr. Webster was solved using dental identification. Dr. Webster was proven guilty in US court by using dental features of the victim Dr. Parkman. Dr. Parkman’s dentist used portions of porcelain denture recovered from Dr. Webster’s laboratory furnace to prove him guilty [8]. Establishment of the first forensic dental identification system dates back to 1887 [9] when it was proposed to keep and utilize AM dental charts for forensic identification purposes.
2.2.2 Dental Identification Process

There are many different dental identification processes and techniques for identification and record keeping purposes. For example, Keiser Nielsen proposed a six-digit code for dental description of each tooth [13] for record keeping in a computer based system. The six digit code consisted of two digits for designating individual tooth, two digits for describing the surface or surface combination involved (if any), and two digits for describing treatment types and material used for the treatment [13]. Then these dental description files can be used for AM/PM comparisons manually or by an automated system.

Another Dental Identification process proposed by G. Jones, exploited the fact that the changes in the dental features are unidirectional [14] and that any changes like dental work or decay of the tooth can never be restored to the original form. This identification process utilized 5 surfaces of each tooth as either intact (no decay, no restoration) or changed (filling, crown, missing, and so forth) for identification purposes. By using these rules, a heuristic rule was formulated i.e. two records can never be a match if a PM record indicates an intact surface and an AM record indicates an altered surface [14] or if a tooth is missing in an AM record but is visible in the PM record. These exclusion rules are still used by the most advanced dental identification systems these days. Figure 2.1 shows the commonly used form for dental features.
**Figure 2.1: Dental Ante Mortem (AM) Form**
2.2.3 Computer-Assisted Postmortem Identification (CAPMI)

Computer-Assisted Postmortem Identification (CAPMI) is another forensic dental identification system designed and developed by the U.S. Army Institute of Dental Research [15] in 1983. The main purpose of CAPMI was to automate and improve the performance of dental identification process. CAPMI basically automated the traditional process of chart-by-chart comparison of dental features for identification purposes. CAPMI utilizes the Optical Mark Read (OMR) form [15] shown in figure 2.2 which contains both dental and non dental (height, weight, eye and hair color, age, sex, and blood type) features of the individuals for identification process. The table given below shows a list of dental features along with their assigned codes that are used by CAMPI system.

<table>
<thead>
<tr>
<th>X</th>
<th>AU</th>
<th>Gold</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>Un-erupted and/or impacted</td>
<td>Acrylic or other non-metal</td>
</tr>
<tr>
<td>CAR</td>
<td>Carious</td>
<td>Porcelain or acrylic to metal</td>
</tr>
<tr>
<td>ANOM</td>
<td>Anomaly</td>
<td>Porcelain jacket</td>
</tr>
<tr>
<td>M</td>
<td>Mesial</td>
<td>Stainless steel</td>
</tr>
<tr>
<td>O</td>
<td>Occlusal or Incisal</td>
<td>Temporary</td>
</tr>
<tr>
<td>D</td>
<td>Distal</td>
<td>Three-quarter crown</td>
</tr>
<tr>
<td>F</td>
<td>Facial</td>
<td>Pontic</td>
</tr>
<tr>
<td>L</td>
<td>Lingual</td>
<td>Root canal</td>
</tr>
<tr>
<td>C</td>
<td>Crown</td>
<td>Deciduous</td>
</tr>
<tr>
<td>AM</td>
<td>Amalgam</td>
<td>partial denture</td>
</tr>
</tbody>
</table>
Figure 2.2: OMR form for use with the CAPMI system at a disaster scene [5]
2.2.4 Win-ID Identification System

WinID is a dental computer system that has proven to be very effective in mass disaster situations and in the establishment of databases for missing people. It uses a Microsoft Access database for storing information about restored dental surfaces [6], physical descriptors, pathological, and anthropological findings. The system is an extension of the earlier CAPMI system, and uses codes that are extensions of the CAPMI codes. Two types of codes are used namely primary codes and secondary codes separated by a dash, with most searches using only the primary codes. The table shown below gives the description of both primary and secondary codes [6] along with their small description. Figure 2.3 shows ante mortem dental record used by WinID.

<table>
<thead>
<tr>
<th>Primary Codes</th>
<th>Secondary Codes</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>Annotation</td>
</tr>
<tr>
<td>O</td>
<td>Deciduous</td>
</tr>
<tr>
<td>D</td>
<td>Crown</td>
</tr>
<tr>
<td>F</td>
<td>Resin</td>
</tr>
<tr>
<td>L</td>
<td>Gold</td>
</tr>
<tr>
<td>I</td>
<td>Porcelain</td>
</tr>
<tr>
<td>U</td>
<td>Non-Precious</td>
</tr>
<tr>
<td>V</td>
<td>Pontic</td>
</tr>
<tr>
<td>X</td>
<td>Root Canal</td>
</tr>
<tr>
<td>J</td>
<td>Silver Amalgam</td>
</tr>
<tr>
<td>/</td>
<td>Denture Tooth</td>
</tr>
<tr>
<td>Z</td>
<td>Temporary</td>
</tr>
</tbody>
</table>
Figure 2.3: Ante mortem dental record used by WinID [6]
2.2.5 Automated Dental Identification System (ADIS)

The Automated Dental Identification system is designed by the researchers and developers from West Virginia University, University of Miami and Michigan State University. The main idea behind designing ADIS was to automate and speed up the dental identification process by reducing the number of records to be reviewed by the forensic scientists in order to come up with final match for a specific subject. ADIS can be divided into three main sub modules i.e. Pre-Processing Module, Potential Match Module and the Image Comparison Module [3]. The subject record is pre-processed by the Pre-Processing Module which basically extracts and identifies all the major features from the record. Then the Potential Match Search Module quickly generates a short Candidate List of the potential reference records from the database provided by the law enforcement agencies. This Candidate List is then used by the Image Comparison Module which utilizes tooth to tooth matching techniques and comes up with a Match List containing a small list of possible matches for the submitted subject record. This Match List is then reviewed by the forensic scientists for identifying the possible final match. Figure 2.4 shows the block diagram of ADIS.

Figure 2.4: Automated Dental Identification System [3]
2.3 Technologies Incorporated for the System Design

HP Web-ADIS Grid was designed by utilizing a diverse list of technologies. The web application for HP Web-ADIS Grid is designed using J2EE and JavaServer Pages (JSP). The web application is deployed on the Apache Tomcat web server and container. These technologies were preferred over Microsoft products because of their ability to run on any kind of operating system. This provides more portability and diversity. Another reason behind choosing these technologies over other available options was the Object Oriented architecture of the HP Web-ADIS Grid. Java technologies are used for communication between different servers i.e. Request Scheduling and Dispatching Service, Pre-Processing/Potential Match Servers, Image Comparison Servers, Database Servers, etc. Applications running on these servers are also Java based. Therefore, it is relatively easy and logical to interface JSP with Java by utilizing JavaBeans technologies. JNI or Java Native Interface technology is used for the invocation of the Matlab Engine and the Matlab based components. Oracle 10g based database is being used for the management and maintenance of user accounts, request details, request history, server statuses, task assignment details, etc. A file system based database system inspired by the YouTube architecture is also incorporated for managing and maintaining the subject record images, reference databases uploaded by the users, and the reference databases provided by the law enforcement agencies. A Visual Basic based Emailer service is responsible for sending notifications and information related to the request to users after completion of each step. All these diverse technologies work together to form a fully functional HP Web-ADIS Grid System. Details related to all the technologies discussed above are given in the next sections.

2.3.1 Java 2 Platform, Enterprise Edition (J2EE)

Java 2 Platform, Enterprise Edition (J2EE) offers a cost effective and fast component based approach for enterprise application design, development, assembly, and deployment. The component based solutions are independent of products and APIs of a single vendor. Figure 2.5 shows the block diagram of the Java 2 Platform, Enterprise Edition. Some important features offered by the J2EE platform [16] are:
LITERATURE REVIEW

- Distributed multitier application model
- Reusable components
- Transaction control flexibility
- Unified security model
- Support for web services

Application logic in J2EE is composed of several components as it uses a distributed multitier application model. A J2EE component is a self contained functional unit [16], assembled into a J2EE application, and is capable of communicating with other components. The functionality of a distributed application is implemented as various J2EE components distributed across multiple tiers. The components are placed on different machines based on the tier in the J2EE environment they belong to. Various tiers in a J2EE application [16] are:

- **Client tier**
  Components belonging to this tier run on the client machine. Application clients and applets are examples of such components. A J2EE client can either be a web client or an application client.

- **Web tier**
  Components belonging to this tier run on the J2EE server. Examples include Java Servlet and JavaServer Pages based components.

- **Business tier**
  Components belonging to this tier run on the J2EE server. Examples are Enterprise JavaBeans components.

- **Enterprise information system (EIS) tier**
  Components belonging to this tier run on the EIS server. This tier includes infrastructure systems such as database systems, enterprise resource planning, transaction processing etc.
2.3.2 JavaServer Pages (JSP)

JSP technology proves particularly effective and efficient for web applications working with a wide variety of application and web servers, browsers and tools. JSP makes it easier and faster to build applications having dynamic web pages. Interactive web applications have special requirements and pose challenges as the web page content in such applications is dynamically generated based on a user request or system status. JSP technology addresses these requirements by offering a new approach to web application development using component based application logic. JSP technology speeds up dynamic web page development in a number of ways. For example, content generation is completely separated from presentation. HTML or XML is used for presentation while JSP tags or scriplets are used for dynamic content generation [17]. Since the core logic is encapsulated in tags and beans, JSP pages can be edited without affecting content.
LITERATURE REVIEW

generation. Use of reusable, cross platform components for performing complex processing also speeds up development of web pages with dynamic content. Also, page development is simplified by using tags as much of the dynamic content generation functionality is encapsulated in JSP tags. JSP pages look like HTML or XML pages with special additional elements to be interpreted by the JSP engine. Table below lists the major components of a JavaServer Page [17):

<table>
<thead>
<tr>
<th>#</th>
<th>Syntax Element</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>JSP directive</td>
<td>Instructions are passed to a JSP engine using directives. JSP directives are enclosed in <code>&lt;%@</code> and <code>%&gt;</code> markers. JSP uses several different kinds of directives. For example, a <em>Page directive</em> is used to inform the JSP engine about various page properties. The <em>Include directive</em> is used to include external content into the JSP page. The <em>Taglib directive</em> specifies a library of custom tags that can be invoked by the page.</td>
</tr>
<tr>
<td>2</td>
<td>Fixed template data</td>
<td>This includes any tags which are not recognized by the JSP engine.</td>
</tr>
<tr>
<td>3</td>
<td>JSP tags</td>
<td>XML based tags are used by JSP for implementing most functionality. Some standard tags used by JSP include: <code>jsp:useBean</code>, <code>jsp:setProperty</code>, <code>jsp:getProperty</code>, <code>jsp:include</code>, <code>jsp:forward</code>.</td>
</tr>
<tr>
<td>4</td>
<td>JSP expression</td>
<td>This component of a JSP page is used to print anything enclosed in <code>&lt; %= and %&gt;</code> markers.</td>
</tr>
<tr>
<td>5</td>
<td>JSP scriplet</td>
<td>A java scriplet is any executable java code enclosed in <code>&lt;%</code> and <code>%&gt;</code> markers.</td>
</tr>
<tr>
<td>6</td>
<td>JSP declaration</td>
<td>Declaration in a JSP page is used to initialize variables. These are enclosed in <code>&lt;%! and </code>%&gt;` markers.</td>
</tr>
<tr>
<td>7</td>
<td>JSP comment</td>
<td>Comments in a JSP page are enclosed in <code>&lt;%-- and --%&gt;</code> markers.</td>
</tr>
</tbody>
</table>
2.3.3 Apache Tomcat

Apache Tomcat is a servlet container [26] developed by the Apache Software Foundation. Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run. Apache Tomcat includes tools for configuration and management, but can also be configured by editing XML [26] configuration files. Apache Tomcat is developed in an open and participatory environment and released under the Apache Software License. Apache Tomcat powers numerous large-scale, mission-critical web applications across a diverse range of industries and organizations.

2.3.4 Java Database Connectivity (JDBC)

JDBC API (Application programming interface) provides database independent connectivity between Java and a variety of databases. A set of high level classes provided by JDBC enable ease in database application development while keeping lower level details transparent to the programmer. JDBC allows a programmer to establish connectivity with a database, send SQL commands to the database, and process the results returned [18]. By providing library routines interfacing with the database, the JDBC API enables the integration of SQL calls into a general programming environment. JDBC technology offers a number of benefits including use of already existing databases, ease of application development, and no client side configurations. Figure 2.6 shows the basic block diagram of JDBC.

Table below shows the four categories into which JDBC drivers [18] can be grouped.

<table>
<thead>
<tr>
<th>#</th>
<th>Driver Type</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Type 4</td>
<td>This type of driver allows a direct call from the client machine to the database server. JDBC calls are converted into the DBMS’s network protocol.</td>
</tr>
<tr>
<td>2</td>
<td>Type 3</td>
<td>This type of driver involves the use of a middleware. The JDBC calls are first converted into the middleware protocol by the driver.</td>
</tr>
</tbody>
</table>
The final translation into the database protocol is performed by the middleware server.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Type 1</td>
</tr>
<tr>
<td>4</td>
<td>Type 2</td>
</tr>
</tbody>
</table>

*Figure 2.6: Java Database Connectivity (JDBC) Block Diagram*
2.3.5 JavaScript

JavaScript is an object based scripting language mainly used for client side web development [21]. It can be used to enable more interactive and responsive web programming, perform client side processing of data, control multiple frame navigations etc. JavaScript is a dynamic scripting language. It provides dynamic typing by associating types with values. Also, the properties and their respective values for a JavaScript object can be dynamically modified. JavaScript also provides runtime execution of statements provided as strings. Other dynamic capabilities of JavaScript include object construction at runtime, dynamic creation of scripts, object introspection etc. Rather than using a compile time system of classes JavaScript uses a runtime system based on a few data types including numeric, string, and boolean data [21]. Functions are also supported which can be object properties executing as loosely typed methods.

2.3.6 Java Native Interface (JNI)

JNI is Java’s native programming interface. It enables applications to use native code written in languages such as C and C++. This provides programmers with all the benefits of the Java platform while still making use of legacy code written in other native languages. The native programming interface becomes particularly useful when applications make use of certain platform dependent features that are not supported by Java [20]. Also under certain situations, to benefit from features offered by another language or tool, part of application program might be written in another low level language. The JNI can then be used for calling such functions from Java. JNI allows invocation of native code from the Java environment and vice versa. One way of using the JNI is to write native methods. By defining a method as native, call to the method are made in the same way as any other Java method, but the actual implementation is in another language as part of a native library. Since JNI is a two way interface a Java Virtual Machine (JVM) implementation can also be embedded into native applications using an invocation interface which can execute java components [20]. Figure 2.7 shows the basic block diagram of JNI.
2.3.7 JavaBeans(TM)

JavaBeans is Java’s portable and platform independent component model. Components are self contained software units capable of interoperating based on a set of rules. These components can be dynamically controlled and assembled together to develop applications. A Java bean is a software component written in Java. Java Beans range from simple GUI elements to visual components such as database viewers. Some key concepts [19] about JavaBeans are listed in the table below.

<table>
<thead>
<tr>
<th>#</th>
<th>JavaBean Concept</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introspection</td>
<td>This process is used by builder tools to analyze a bean’s functionality.</td>
</tr>
<tr>
<td>2</td>
<td>Properties</td>
<td>These define various characteristics of a bean modifiable at</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td></td>
<td>design time. A bean’s properties are used both for customization and programmatic use.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Customization</td>
<td>Customization allows modifications in the appearance and behavior of a bean. This can be achieved with property editors or bean customizers.</td>
</tr>
<tr>
<td>4</td>
<td>Events</td>
<td>Events are the communication mechanism among beans.</td>
</tr>
<tr>
<td>5</td>
<td>Persistence</td>
<td>This enables a bean to save and restore its state, such that when it is customized, the changed state is saved and reloaded later with the property changes intact.</td>
</tr>
<tr>
<td>6</td>
<td>Methods</td>
<td>These are same as other Java methods.</td>
</tr>
</tbody>
</table>

### 2.3.8 Structured Query Language (SQL)

SQL is a fundamental component of modern database architecture enabling database creation and manipulation on all major platforms. SQL commands fall into two main categories the Data Definition Language (DDL) commands and the Data Manipulation Language (DML) commands [22]. Used primarily by database administrators, the DDL commands are used for creating and destroying databases and database objects. Examples includes commands for creating databases and tables, commands for altering or dropping tables, commands for selecting a particular database within the database management system etc. The DML commands on the other hand modify information within a database. Examples include commands for inserting data into a table, deleting data from a table or updating data in a table etc. SQL is divided into several language elements including clauses, expressions, predicates, queries, statements, and whitespaces [22].

### 2.3.9 Oracle 10g

Oracle 10g is a commercial object relational database management system. Object relational databases resulted from an integration of relational approach to databases and object oriented concepts. The integration was intended to benefit both from the simplicity of relational databases and strong development features of the object oriented model.
LITERATURE REVIEW

Mostly SQL and PL/SQL are used by back end enterprise applications and web applications for accessing the database [27].

2.3.10 Others

Many other technologies like Visual Basic, Matlab, a third party Load Balancing Service, Microsoft Outlook, etc. are also incorporated in the system design of HP Web-ADIS Grid.
Chapter 3 HP WEB-ADIS: DESIGN & COMPONENTS

This chapter discusses the different components and modules of HP Web-ADIS Grid working together in parallel to fulfill user requests. Section 3.1 gives brief introduction of the system, section 3.2 gives details about the HP Web-ADIS Application Server, section 3.3 explains different aspects of HP Web-ADIS Grid design and its components, section 3.4 discusses Request Scheduling and Dispatching Service, section 3.5 puts light on the Preprocessing/Potential Match Servers, section 3.6 explains different components and functionalities provided by the Image Comparison Servers, and at the end of the chapter we discuss different databases that are maintained and managed to provide the state of the art functionality, flexibility, and accessibility to the end user.

3.1 Introduction

One of the main tasks assigned to the ADIS design and development team was to make the automated dental identification system accessible to the users using a simple web browser application running on any platform. It was also required to improve performance of the automated dental identification system while ensuring swift availability of results to the forensic scientists. This chapter gives deep insight into the system design, techniques, and technologies incorporated in order to make the system accessible using a web browser and to improve its performance.

3.2 HP Web-ADIS Application Server

The HP Web-ADIS Application server hosts the web interface of the application and enables the users to fully utilize the functionality provided by the automated dental identification system. This server provides a user friendly interface to the clients so that
they can effortlessly utilize the functionalities provided by the automated dental identification system from anywhere in the world using a simple web browser application running on any platform. The main application server, running Apache Tomcat servlet container, is deployed with the Web-ADIS application and serves the user’s requests by rendering dynamically generated pages depending on the request type.

3.3 Load Balancing Servers:

The whole web application is deployed on multiple load balancing servers in order to ensure reliability and availability to the end user. A third party load balancing service is used to divide the requests among multiple application servers for load balancing and redundancy. All the load balancing servers are deployed with the same copy of the web application and perform the exact same functionality. Load balancing in HP Web-ADIS is accomplished by using a simple service that sends the incoming requests to different load balancing servers by utilizing a round robin technique. It is important to note that in case of a server failure, the load balancing service automatically removes the failed server from the list of available servers. A simple block diagram of the load balancing technique is shown in figure 3.1.

![Figure 3.1: HP Web-ADIS Application Server with load balancing](image)
3.4 Web Application:

The key functionality of the web application is to securely connect Web-ADIS to the outside world and the clients using a web browser via the http protocol. Users can access the Web-ADIS application by putting the appropriate URL in a web browser. The web browser then redirects the user to the Web-ADIS application’s ‘welcome page’. Figure 3.2 shows the ‘welcome page’ of web application.

![Figure 3.2: HP Web-ADIS Application Interface](image)

Users can then login to the system by entering their username and password. New users are requested to email the ADIS team in order to get the required pass code for account creation. This ensures that only reliable and qualified users can access the web application. The pass code authentication page is shown in figure 3.3. New users after providing the correct pass code are redirected to the ‘create new account’ page where they can create a new account by entering their credentials and other related information. After creating the account, users can login to the system and utilize the functionalities provided by the application. The overall flow of the pass code authentication process and the account creation process is shown in figure 3.4 and figure 3.5 respectively.
Welcome New User,

please e-mail us @ Hannay Ammar to get your access code before you can create a new user-ID on ADIS server and use its services.

Figure 3.3: HP Web-ADIS Pass Code Authentication Interface

Create New Account (Passcode Authentication)

Figure 3.4: HP Web-ADIS Pass Code Authentication
3.4.1 Web-ADIS Authentication:

As illustrated in figure 3.5, user information is checked for validity before the account creation process. The information is checked by using JavaScript on client side before sending it to the server. The information is also checked at the server side to ensure correctness. After the validity check, the account creation process starts which can be divided into two main steps. In the first step appropriate databases, directories, and structures are created for the new user on the Image/Feature database. This database is then used to save user’s uploaded records, uploaded databases, preprocessed records, candidate lists, and the match lists for each request. After the successful creation of the account on the Image/Feature database, user’s account information is saved on the UserInfo database. This database contains login ids, passwords, addresses, etc. for each user. User is then redirected to the welcome screen of the application. Both the UserInfo and Image/Feature databases will be discussed later in this chapter.

The users with a valid username and password can simply login using the login page. The authentication process of Web-ADIS is highly secure and ensures security by utilizing an elaborate authentication process. The authentication process checks the login information against the UserInfo database to ensure validity. User is then redirected to the ‘reenter
info page’ or the ‘Web-ADIS home page’ depending on the response generated by the validation module.

![Authentication Diagram](image)

*Figure 3.6: HP Web-ADIS Authentication*

After successfully login into the system, the user is redirected to the ‘home page’ of the application. Figure 3.7 shows the application home page. The home page provides four main options to the user i.e.

- User can click on the introduction and tutorial video to learn about the functionality of the application and how to use it.
- User can click on the identification option to start the identification process.
- User can review the request history and a can access all the requests and their results using this interface.
- User can configure the application by picking and choosing different tools designed by different teams and their parameters values. The settings are then saved on the configuration database and applied when the preprocessing, potential matching, and image comparison tools are invoked for processing.
3.4.2 Identification Interface:

The identification option provides the functionality of uploading the subject record for matching purposes. The user, on this interface, is asked to upload an image file, record type i.e. AM or PM (anti mortem or post mortem record), and the database name from the dropdown menu to match with. The User can match the subject with all the databases or can select one of the databases provided by FBI and other law enforcement agencies. As shown in figure 3.8, this interface also asks the user if he/she wants to create and upload a new database on the server. If the user wants to upload a new database on the server, he/she is taken to the ‘upload database page’, shown in figure 3.10. The user can then upload a zip file onto the server. The user is also asked to name the new database and to specify the type of the database they are uploading i.e. AM or PM records. After the successful submission, the zip file is saved on the Image/Feature database at a specific location reserved for each user’s uploaded zip files. The user can later choose his/her uploaded databases for identification purposes.
One of the prime functionalities of the identification interface is that it retrieves all the databases uploaded by the user in the past, enabling him/her to use one of those databases for matching purposes. This option is illustrated in figure 3.9. The databases uploaded by the user and the related details are retrieved from the Image/Feature database.
After gathering all the required information, the request details i.e. subject record type, database to match with, reference database uploaded or not, request date, request time, etc. are saved in the UserInfo database. The uploaded record is saved in the Image/Feature database at an appropriate location reserved for each user’s uploaded records. Figure 3.11 shows the basic identification process and its flow. Figure 3.12 shows the process flow for identification when the user opts to upload a reference zip file.

Figure 3.11: HP Web-ADIS Identification
Figure 3.12: HP Web-ADIS Identification with reference DB Upload
3.4.3 Request History Interface:

The request history interface, as shown in figure 3.13, enables the user to preview old and processed requests and check on the status of the new and pending requests. This interface retrieves request details for the specific user from the UserInfo database. It also retrieves subject image, preprocessing results, candidate lists, and match lists from the Image/Feature database for the completed requests. A pending sign is displayed for the requests that are not complete and are still being processed by the system. Figure 3.14 depicts the overall process flow for this interface.

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Subject Image</th>
<th>Subject Category</th>
<th>Reference Uploaded</th>
<th>Preprocessing Results</th>
<th>Candidate Lists</th>
<th>Match Lists</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009-02-09</td>
<td>15:12:40</td>
<td>AM 082.jpg</td>
<td>AM</td>
<td>NO</td>
<td>Preprocessing Details</td>
<td>Candidate List</td>
<td>Match List</td>
</tr>
<tr>
<td>2009-02-10</td>
<td>16:40:25</td>
<td>AM 011.BMP</td>
<td>AM</td>
<td>NO</td>
<td>Preprocessing Details</td>
<td>Candidate List</td>
<td>Match List</td>
</tr>
<tr>
<td>2009-02-10</td>
<td>14:26:22</td>
<td>AM 008.bmp</td>
<td>AM</td>
<td>NO</td>
<td>Preprocessing Details</td>
<td>Candidate List</td>
<td>Match List</td>
</tr>
<tr>
<td>2009-02-10</td>
<td>14:48:17</td>
<td>AM 009.BMP</td>
<td>AM</td>
<td>NO</td>
<td>Preprocessing Details</td>
<td>Candidate List</td>
<td>Match List</td>
</tr>
</tbody>
</table>

*Figure 3.13: HP Web-ADIS Request History Interface*
Figure 3.14: HP Web-ADIS Request History
3.4.4 Preprocessing Results Interface:

The preprocessing results interface allows the user to view the results of the preprocessing component for the uploaded subject record. This interface is used to view the results of each sub module of the preprocessing component. The preprocessing component can be divided into the following sub parts i.e. record cropping, film enhancement, film segmentation, film type classification, contour extraction, and teeth labeling. The preprocessing results interface is shown is figure 3.15 and 3.16. The preprocessing component and its modules will be discussed in detail later in this chapter.

Figure 3.15: HP Web-ADIS Preprocessing Details Interface
3.4.5 Candidate List Interface:

Candidate list interface, as shown in figure 3.17, is used to view the results of potential match search component. This interface displays the candidate list containing the potential matches for the specified subject record, the image preview of the potential matches, the distance or the closeness of the potential match record to the subject record, and the database name to which the potential match belongs to.
3.4.6 Match List Interface:

Figure 3.18 shows the Match List interface which is used to view the results of image comparison component. This interface displays the match list containing the possible matches for the specified subject record, the image preview of the possible matches, the distance or the closeness of each match to the subject record, and the database name to which the possible match belongs to. The Match List is generated by the Image Comparison Component using the Candidate List and only contains the closest matches to the subject record.
3.4.7 Configuration Interface:

The Web-ADIS design and development team comprised of research engineers from three different universities (West Virginia University, University of Miami and Michigan State University), working together in parallel and sometimes designing the same components with different identification and pattern recognition techniques. Secondly, the different Matlab based tools that are used by the system allow the user to select different realization options and other related parameters for identification process. Therefore, an interface was needed that could allow the user to select different tools (preprocessing, potential matching, and image comparison) designed by different teams and can allow him/her to set the realization techniques, parameter values, and other related options. For example, by using this interface the user can select the preprocessing and cropping module designed by WVU, record enhancement module by UM, and teeth labeling module developed at MSU. The user can save the configurations on to the server. Any new processing requests by that user will then utilize the new saved configuration settings. Figure 3.19 shows the configuration interface of HP Web-ADIS application.

![Configuration Interface](image)

*Figure 3.19: HP Web-ADIS Configuration Interface*
3.5 HP Web-ADIS Grid

The older version of Web-ADIS was able to process one request at a time. The reason behind this was that the requests were being directly sent to the Matlab based tools such as preprocessing, potential matcher, and image comparison components. The problem with the Matlab based tools is that only one instance of the component can be invoked at one time. Hence in the older version of Web-ADIS, all requests were being queued until the completion of the first request and some of them were being lost with the expiration of JSP sessions. The requests were being lost as the corresponding information was not saved anywhere other than the JSP session objects.

Another problem with the older design was that it was single server based. Therefore, the system took a lot of time to fulfill a user’s request as the image comparison process is extremely time consuming. In order to improve system performance and to make it available to multiple users with multiple request facility, a new design was essential.

The new and improved design called the HP Web-ADIS Grid was inspired by the YouTube architecture where multiple ordinary machines work together to fulfill user requests. In the YouTube architecture, data and paths are saved on a relational database while files and videos are saved on a file system based database. HP Web-ADIS Grid applies the same idea and saves the request details, subject image paths and locations, paths to the feature databases, task assignments, servers, and their details in an Oracle based database. The subject records, images, reference databases, candidate lists, and match lists are saved on the Image/Feature database. The Image/Feature database is file system based, conforms to certain set of rules, and follows a specific structure.

The HP Web-ADIS Grid is linked to two types of databases. First one is the UserInfo database containing information related to user accounts, user requests, and status of each node connected to the grid. Second one is the Image/Feature database which is used to keep the default reference databases (records and preprocessed files) provided by the law enforcement agencies. It also maintains the reference databases uploaded by the user. Subject records and zip files uploaded by the users while submitting a new processing request are also saved on the Image/Feature database.
When users access the web application for submitting their requests, all required information is saved on UserInfo and Image/Feature databases. This information is later used by the HP Web-ADIS Grid for processing. The grid comprises of three main servers i.e. Preprocessing/Potential Matching Servers (PP/PMS), Image Comparison Servers (ICS) and Database Server. Figure 3.20 depicts the block diagram of HP Web-ADIS Grid. The end to end functionality is shown in figure 3.21.
3.5.1 Database server:

The database server can be further divided into UserInfo database and Image/Features database. UserInfo database is an Oracle based database and holds information related to user accounts, request history and details, servers details and statuses, servers processing assignment details, etc. The Image/feature database on the other hand contains the reference databases provided by the law enforcement agencies, reference databases uploaded by the users, the uploaded subject records for matching and so on. The reference databases contain the record images, the preprocessed files, and the enhanced films extracted from the uploaded subject record.

3.5.2 Preprocessing/Potential Matching Servers (PP/PMS):

PP/PM servers are running preprocessing and potential matching components written in Matlab. Each PP/PM server checks the UserInfo database for new assignments. If there is an assignment for a server, it will start processing right away. After completion of subject preprocessing, potential matching, and/or batch preprocessing of uploaded reference database PP/PM servers divide the candidate list generated by the potential matching tool
into sub candidate lists depending on the number of available Image Comparison Servers (ICS). The information related to the sub candidate lists is then uploaded on the UserInfo database and the Image Comparison Servers process the remaining part of the request. After completion of the request these servers check the UserInfo database again for any new assignments. If there are no new assignments, they go into a sleep mode until a new assignment is assigned.

3.5.3 Image Comparison Servers (ICS):

The IC servers check theUserInfo database for assignments. If there is an assignment they start processing the request by using the subject image, subject preprocessing file, candidate list, and the reference record database. After the processing is complete, each ICS saves the partial match list on the Image/Feature database. All these partial match lists are then merged to generate the final match list. The IC servers go in sleep mode when there are no task assignments for them to process.

3.6 Request Scheduling and Dispatching Service

After a request is submitted by the user through the web interface, the request data is saved into the UserInfo database. Immediately after the insertion of the data in the table, an Oracle based trigger is fired that checks for the number of available Preprocessing/Potential Matching Servers and recent assignment history. The current or new task is assigned to the server that was not assigned any task recently. Assignment details for the PP/PM servers are also updated in the UserInfo database. This modified information is then read by the PP/PM servers and they start processing the assigned tasks accordingly. Figure 3.22 explains the whole functionality of the request scheduling and dispatching service.
Figure 3.22: HP Web-ADIS Request Scheduling & Dispatching Service
3.7 Pre-processing/Potential Match Servers

Preprocessing/Potential Match Servers or PP/PM servers are one of the main components of the HP Web-ADIS Grid design. These servers perform very diverse activities and tasks. The PP/PM servers continuously look for any new assigned tasks by checking the UserInfo database periodically. If there are no task assignments for that specific PP/PM server, it goes into the sleep mode. If there is a new assignment, the server starts processing accordingly. The first step that is taken by the PP/PM servers is to get all the request related details from the UserInfo database. This information includes the location where the subject image was uploaded on Image/Feature database, subject image type (AM or PM), reference database name to match against, request timestamp, username of the requester, either the Zip file was uploaded or not, if yes then the path where it was saved, and other request related data. It is important to note that the request timestamp is used to uniquely identify each request and no two requests can have the same timestamp value. Request timestamp basically holds the year, month, day, hour, minute, second, millisecond, and microsecond value with the time zone displacement. After gathering all the information, the PP/PM server retrieves the subject record from the Image/Feature database by using the information it received from the UserInfo database. It then starts preprocessing the uploaded subject record and saves the preprocessing results on Image/Feature database at a location already specified in the UserInfo database. The details about the preprocessing component are given below:

3.7.1 Preprocessing Component:

The preprocessing component is a Matlab based tool. It is composed of six major sub modules that work together to come up with the preprocessed text files and the dental images containing all the information related to the dental features. The figure 3.23 briefly depicts the basic block diagram of the preprocessing component and its sub modules. The six major sub modules of the preprocessing component are: Record Cropping Component, Film Enhancement Component, Film Classification Component,
Segmentation Component, Contour Extraction Component, and Teeth Labeling Component. Details about these modules are given below:

3.7.1.1 Record Cropping:

Images uploaded by the users are a collection of radiographic films called a dental record. One dental record can have multiple films in it. The first step in the preprocessing stage is to crop each film into separate image files and crop out all the irrelevant data often written on the dental records by the dentists [3]. The figure 3.24 shows the cropping process.
3.7.1.2 Film Enhancement:

Film quality is extremely important for good results. Better quality films and images are easier to preprocess than those that are not very clear and have a lot of noise [3]. Poor quality records are also hard to segment and label [1]. The enhancement module basically improves the teeth visibility by reducing noise against the dark areas of the image. The results of film enhancement module are shown in figure 3.25.

3.7.1.3 Film Classification:

The record image can contain many different types of films representing different sections of the jaw. The image can be taken from different angles and views. The kind of
films found in a record can be divided into three main categories i.e. Bitewing, Periapical and Panoramic [3]. Bitewing class of films shows both upper and lower jaws in a single film, Periapical class shows either the upper or the lower jaw of the subject, and the Panoramic class shows all the teeth in both jaws in a single film. This component basically identifies the type of each film in the record. Figure 3.26 shows the different kinds of films found in the record image.

![Figure 3.26: Preprocessing Step3: Film Type Classification Block Diagram](image)

3.7.1.4 Segmentation:

The segmentation module identifies and divides each film into constituent regions and extracts the features of interest from those regions. These features are then used by the pattern recognition and image matching part of the identification process performed by the Image Comparison Component. Each segmented tooth represents a Region of Interest (ROI) that contains distinctive features used in the subsequent steps of identification. The segmented tooth is represented by a rectangular area in the film that bounds one tooth [2]. The figure 3.27 shows the block diagram of the segmentation component.
3.7.1.5 Contour Extraction:

Contour Extraction module extracts the contours and borders of each tooth shown in the film. As discussed earlier, the Image Comparison Component utilizes tooth level matching and that is why the contour extraction module can efficiently and effectively identify each tooth and the areas of interest, generating better results. Contour extraction module is also used to differentiate the tooth from the background and the neighboring tooth [3]. The figure 3.28 shows the block diagram of the Contour Extraction Component.

Figure 3.27: Preprocessing Step4: Segmentation Component Block Diagram
3.7.1.6 Teeth Labeling:

Teeth labeling module annotates each tooth. The tooth annotations can then be used by the Image Comparison Component for tooth to tooth matching process [3]. Teeth labeling is done for each film in the record. Figure 3.29 shows the block diagram of the Teeth Labeling Component.
3.7.2 Reference Database Pre-processing:

After completion of subject’s preprocessing, the PP/PM server checks for the reference database uploaded. If a zip file was uploaded by the user, then the PP/PM server gathers all the details related to the new uploaded reference database, its type (AM or PM), its name, path of the zip file where it was saved, and the destination location where the zip file will be extracted. Using all these details, the zip file is extracted at a special location specified for that specific user in the Image/Feature database and given a name as directed by the user.

After the completion of database naming, zip file extraction, and database creation process, a PP/PM server invokes the batch preprocessing component. Batch preprocessing component is used to preprocess all the new records uploaded by the user in the zip file. Batch preprocessing is exactly the same process like the subject record preprocessing but is done in batch. All the reference records uploaded by the user in the new database are processed simultaneously.

Potential Matching Component is invoked after the successful completion of preprocessing tasks. The PP/PM server prepares the subject record files and the reference databases to match with, before invoking this component.
3.7.3 Potential Match Search Component:

Potential Match Search uses the subject record and the reference database to generate a list of potential matches for that specific subject. This list of potential matched is called the Candidate list. The main idea behind the implementation of the Potential Match Search was to reduce the database size for the time consuming Image Comparison Component. The Potential Match Search uses two techniques to effectively reduce the database size for the Image Comparison Component [3].

The first technique uses color features called Eigen images for processing. The Eigen images technique is an appearance based technique with extremely low computational cost. This step also consists of view normalization which is used to align the segmented teeth vertically [3]. View normalization is introduced to normalize the data (each tooth in the film) before the matching process. It is important to note that the Potential Match Search Component is a Principal Component Analysis (PCA) based system. PCA is used to project each tooth in a lower dimensional space making the tooth-to-tooth comparison efficient with extremely low computational cost. The second technique uses the teeth contours descriptors, the shapes of each tooth, and edge orientation histograms as features to describe the tooth contour [3]. This information is then used for the matching process. Figure 3.30 depicts the block diagram of the Potential Match Search Component.
After the successful generation of the Candidate List by the Potential Match Search Component, the Candidate List is saved on the Image/Feature database at a location specified in the UserInfo database.

### 3.7.4 Candidate Splitter:

After getting the final Candidate List, the PP/PM server initiates the Splitter Service. The main idea behind the Splitter Service is that it divides the Candidate List into Sub-Candidate Lists depending on the available Image Comparison Servers. The Splitter Service creates new sub-candidate list files for each IC Server and assigns it a part of the original Candidate List for processing. After the sub-candidate list creation, the UserInfo database is updated with the new assignment details and the details related to the sub-
Candidate Lists. The IC Servers can access this information and start processing the Image Comparison request accordingly.

### 3.7.5 HP Web-ADIS Emailer:

HP Web-ADIS Emailer Service is a Visual Basic based service which is invoked by the PP/PM Servers twice for each request. The email service checks the status of each request. Whenever it sees a completed request, it gets the email address of that specific requester from the UserInfo database and sends an email immediately. Emails are sent to the requester twice during the request processing. The first email is sent after the completion of the first step i.e. preprocessing of the uploaded subject record, and/or preprocessing of the uploaded reference database, and the availability of the Candidate List. The second email is sent to the user when the final Match List is available.

This Visual Basic based Emailer uses the email address, requester’s username, first name, last name, request time, request date, subject record’s name, and subject record’s type. Based on this information, a formatted email is sent to the user. Figures 3.31 and 3.32 show the emails sent by the Emailer Service.

![Emailer Service's Email](image)

*Figure 3.31: HP Web-ADIS Emailer Service’s Email Completion of First Step*
3.7.6 Match List Merging Service

The Match List Merging Service checks for the sub-Match Lists generated by the IC Servers using the sub-Candidate Lists. The Merging Service checks the status of each request by checking the results generated by each IC Server. If all sub-Match Lists for a specific request are available, it merges all the Sub-Match Lists and generates the final Match List containing the results from all the IC Servers. After the merging process, the UserInfo database is updated and request status is changed to Finished. An email is then sent to the requester by the HP Web-ADIS Emailer Service. Figure 3.33 illustrates the process flow for the Match List Merging Service.

Figure 3.32: HP Web-ADIS Emailer Service’s Final Email
Figure 3.33: HP Web-ADIS Match List Merging Service
The entire process flow for the PP/PM Servers and their constituent modules and services are shown in figure 3.34.

![Preprocessing/Potential Match Servers][1]

**Figure 3.34: HP Web-ADIS Preprocessing/Potential Match Servers**
3.8 Image Comparison Servers

The main idea behind the implementation of the grid based design was to improve the performance of the system and to ensure speedy accessibility to the results. The main bottleneck in the older design was the time taken by the Image Comparison Component to process the request. It took about 5 to 6 hours to process a one to one comparison. After looking at these issues with performance, a grid based design was proposed. The HP Web-ADIS Grid divides the Image Comparison task among multiple IC Servers reducing the processing time exponentially. Each IC Server periodically checks the UserInfo database for any task assignments and goes into sleep mode if there are none. When a task is assigned to the IC Server, it gets the request details i.e. subject record location, subject record type, sub-Candidate List location, and the destination at which the sub-Match List will be saved from the UserInfo database. After gathering all the required information, the Image Comparison Component utilizes the required files and information from the Image/Feature database in order to come up with a sub-Match List. After the generation of the sub-Match List, it is saved at the location already provided in the UserInfo database. The IC Server then updates its availability status to “Available” and the request status to “Finished”. It then checks for any new task assignments and if there is one, it starts the process all over again.

3.8.1 Image Comparison Component:

The Image Comparison Component is a Matlab based tool and this section discusses the details about its functionality and the techniques it utilizes for matching purposes. In region of interest (ROI) pair selection step the corresponding segmented teeth pair or the regions of interest (ROI) are selected for both the subject and the candidate records. It is accomplished with the help of the results from the teeth-labeling module in record preprocessing step. After the completion of ROI step, the teeth alignment kicks in. Teeth alignment is used for the pair-wise region of interest (ROI) alignment. It is important to note that we always start with a hypothesis that the two objects (ROI) are matched. Appropriate transformations to restore major geometric discrepancies between them are
applied accordingly. In the low-level feature extraction step, a set of nonlinear filters are used to map an ROI in the corresponding feature spaces [3]. After the completion of the low-level feature extraction step, the micro-decision step is initiated. This step uses Bayesian Classification Layer that calculates the Posterior Probability of match between a pair of ROIs using the differences between spatially corresponding features of the ROI-pair [3]. The dental record usually contains multiple films showing the same tooth with different angles and views. Therefore, the process of determining the match status of a subject/reference tooth-pair is referred as micro-decision making. It is based on comparison of multiple views of this tooth-pair. Finally the multiple micro-decisions are combined into a macro-decision which determines the match status of the subject/candidate record-pair. It also determines whether to put the candidate record on the match list or not. If the determination process passes a candidate record then that candidate record is written in the final match list along with a ranking score. The block diagram of the Image Comparison Component is shown in figure 3.35.

Figure 3.35: Image Comparison Component Block Diagram [3]
The entire process flow for the Image Comparison Servers and their constituent modules are shown in figure 3.36.

**Figure 3.36: HP Web-ADIS Image Comparison Servers**
3.9 Database Server

The database server is the heart of HP Web-ADIS Grid design. The database server is basically used to hold user’s account information, request details and history, servers connected to grid and their status, task assignments to the servers and request status, subject records, reference records, the preprocessed files generated by the Preprocessing Component, Candidate Lists, and the Match Lists. The database server can be divided into two main parts i.e. UserInfo database and the Image/Feature database. Both these databases are discussed below in detail.

3.9.1 User Info Database

User Info database is an Oracle based database system to manage and maintain the data related to the users and their requests. This database is accessed by the web application, the PP/PM Servers, and the IC Servers. This database can be divided into three main parts:

3.9.1.1 User Details Database:

This database contains the user’s account information and the HP Web-ADIS Grid uses this information for the authentication process. Whenever a new user creates an account, all the information related to his account i.e. first name, last name, username, password, gender, department name, street address, city, state, and email address are saved into this database. It is important to note that the Web-ADIS Emailer Service also uses information like username, first name, last name, and email address from this database before sending an email regarding the request status.

3.9.1.2 Request Details Database:

The information provided by the user for identification purposes i.e. subject record, subject record type, reference database to match against, reference database uploaded or not, along with the details related to the date and time of the request are saved on this database. As users can submit multiple requests, each request is uniquely identified by
the Oracle time stamping functionality which uniquely time stamps each request. It is important to note that this database only contains the paths and locations of the uploaded subject record and/or the uploaded reference database. The physical subject record image files and the reference database zip files are stored on the Image/Feature database which is discussed in the next section.

This database is used by the PP/PM Servers to collect the new request details before starting the processing. The Emailer Service also periodically checks this database for status of the requests and sends email for the completed ones. Request Details database is also used by the ‘request history interface’ for displaying request history and statuses.

### 3.9.1.3 Task Assignment Database:

This database is used to hold information related to the tasks assigned to each Image Comparison Server. This database contains information related to the subject image location, subject image’s preprocessed files and data, location of the partial candidate list, and destination location of the partial Match List. It also contains request status information i.e. whether it is waiting to be processed or has been processed. It is important to note that this database is updated by the PP/PM Servers after the completion of Preprocessing and Potential Matching process. The task assignment for each IC Server is saved on this database by the PP/PM Servers. It is then read by the IC Servers for Image Comparison step. Request for each IC Server can be uniquely identified by the timestamp value for that particular request.

### 3.9.2 Image/Features Database

The Image/Feature database is the second main database of the HP Web-ADIS Grid design. This database is a file system based structure designed to keep uploaded subject records, reference databases, and databases uploaded by different users along with their corresponding preprocessing details. Candidate Lists and the Match Lists are also saved in this database. This database follows a special design and specific rules.

This database can be divided into two main parts. First is the ADIS default databases containing all the databases provided by FBI and other law enforcement agencies. The
second part of this database consists of a special hierarchical structure for each user. It is designed to save uploaded subject records and their preprocessing details at a specific location. Also, the uploaded zip files and reference databases are saved at a predefined location. This database is used by both the PP/PM and IC Servers while processing user’s requests. The results are also saved in this database, which are then accessed by the web application for displaying and presentation purposes. Figure 3.37 depicts the structure of this database system and its design.

It is important to note that the special hierarchical structure is created for each user on this database at the time of account creation. Users are assigned a space on this database with uploading privileges at that time. This enables users to upload their own databases and subject records for identification purposes. Image/Feature database design is highly secure and all the uploaded information (subject records and uploaded reference databases) and the request activity can only be accessed by the user who uploaded it.
Figure 3.37: HP Web-ADIS Image/Feature Database Structure
Chapter 4 SYSTEM TESTING & RESULTS

The HP Web-ADIS Grid system went through extensive testing before it was made available to the end users. The system went through all kinds of testing techniques including Black Box, White Box, Unit, Incremental Integration, Functional, End-to-end, Load, Performance, Usability, Security, Compatibility, and Beta Testing. This chapter will discuss the different kinds of testing scenarios that were conducted to ensure system’s reliability and accessibility.

4.1 Testing Scenarios for the Web Application

The web Application of HP Web-ADIS Grid gives a face to the whole system. It is basically the interface that users can interact with in order to use the functionalities provided by the system. The main idea behind making the ADIS accessible to the users online was to make it easy for them to use the system functionalities from their anywhere without any difficulty. Therefore, there was a need of an easy to use web interface which makes system usage possible without any training. Another requirement was to make the system compatible to all kinds of browser applications and operating systems ensuring portability and accessibility.

4.1.1 Interface and Usability Testing

The web interface was tested for ease of use and high accessibility. This interface was designed with the view that the web application will be used by forensic scientists and individuals with limited computer literacy. The web interface was introduced to diverse range of users and was approved for its ease of use.

4.1.2 Unit and Incremental Integration Testing

The entire web application was tested to ensure that it meets all requirements set forth for the web interface. All functionalities provided by the system like new account creation
interface, login interface, identification interface, reference database uploading interface, request history interface, subject pre-processing interface, candidate list interface, and the match list interface were tested on the functional and unit level. Each module of the web application was also tested in Incremental Integration mode to ensure that it seamlessly integrates and works perfectly with other components of the web application.

4.1.3 Load and Performance Testing

As required by all web applications, HP Web-ADIS ensures that multiple users can access the web application simultaneously without any delays or problems. The web application is designed keeping in mind the fact that it will be used by multiple clients submitting requests simultaneously. The system was tested in load and performance testing mode with multiple users trying to access the application at the same time. The users were simulated by writing a simple program in Java, which used different usernames and passwords for accessing the web application. The web application was able to handle the load efficiently and passed the load and performance testing without any troubles.

4.1.4 Security and Compatibility Testing

The security of the web application was one of the main concerns during the testing process. The entire web application was tested for any security flaws or glitches. Login interface was tested to ensure that only the users with correct credential are able to access the protected area. Furthermore, the authentication process was tested by checking the JSP session variables for each logged in user. The Create New Account interface was tested to ensure that only users with the correct pass code can create accounts. HP Web-ADIS web application successfully passed all the security test cases and scenarios. Compatibility of the web application is another important factor. The HP Web-ADIS web interface was tested on diverse operating systems like Unix, Linux, and Windows based operating systems. The web application was also tested in a diverse range of browsers to make sure that it works perfectly in all major browsers currently available. HP Web-ADIS web application successfully passed all the compatibility test cases and scenarios.
SYSTEM TESTING & RESULTS

4.1.5 Beta and End-to-End Testing

The end-to-end testing of the web application was conducted by five different users, continuously using the system for many days and submitting multiple requests. All the scenarios and test cases set forth for end-to-end testing were passed and all user requests were processed successfully. Users were able to create new accounts, submit new requests, upload new databases, and review their results without any issues.

4.2 Testing Scenarios for the PP/PM Servers

The PP/PM Servers are one of the main components of the HP Web-ADIS Grid design. As explained earlier, PP/PM Servers are responsible for pre-processing uploaded subject records, pre-processing uploaded reference databases, and generating Candidate Lists by utilizing the Potential Match Search Component. The PP/PM Servers were tested with a diverse range of test cases to ensure stability and reliability of the system.

4.2.1 Unit and Incremental Integration Testing

As explained previously, the PP/PM Servers can be divided into five major parts on the basis of functionality i.e. Subject Pre-processing, Zip File Extraction, Reference database Pre-processing, Potential Match Search, Candidate List Splitting process for IC Servers, and Email utility for notifying the users about the request status. Each of these five sub modules were tested individually in the Unit testing mode. These modules were also tested in Incremental Integration testing mode to ensure that each component integrates and works perfectly with the other components of the PP/PM Server. Each module of PP/PM Server was tested by submitting repeated requests using a SQL script. The script continuously submitted same set of requests over and over again with minor modifications to cover all possible scenarios. The PP/PM Server successfully passed all the testing scenarios.
4.2.2 Load and Performance Testing

The load and performance testing of PP/PM Server was done by submitting multiple requests in a loop using a SQL script. PP/PM Server utilizes Matlab based component for Pre-processing and Potential Matching steps, which requires a lot of processing time and system memory. Therefore, special attention was given to the computational time and memory requirements during the testing process to ensure system stability in all possible scenarios. We are proud to say that the system ran continuously for 2 weeks and processed all the requests without any problems. The simulated users were able to get email notifications and were able to view the results throughout the testing period of 2 weeks.

4.2.3 Data Security Testing

As PP/PM Servers are hidden from the users, their security was not a major issue. Data security was tested during this process to make sure that data and requests of one user are secure and are not visible to others. It was also tested that request details are processed and saved at appropriate locations. The PP/PM Server successfully passed all the data security testing scenarios.

4.2.4 End-to-End Testing

The end-to-end testing of PP/PM Server was done by submitting new requests in the UserInfo database in an automated mode. A SQL script was used to repeatedly submit new requests to the server. The PP/PM server was successful in processing the requests defined in the SQL script by Pre-processing subject and Reference records, Zip file Extraction, and sub-Candidate Lists generation for IC Servers. The PP/PM Server performed perfectly in all the test scenarios and passed all the requirements set forth for the end-to-end testing.
4.3 Testing Scenarios for the IC Servers

IC Servers are one of the most important building blocks of the HP Web-ADIS Grid design. The IC Servers are responsible for Image Comparison Process and generate partial Match Lists which then are merged to form a final Match List. Image Comparison is the most time consuming component and requires a lot of memory and processing power. Therefore, utmost care and attention was given while testing IC Servers. The IC Servers were tested with a diverse range of test cases to ensure stability and reliability of the system.

4.3.1 Unit and Incremental Integration Testing

Unit and Incremental Integration testing of IC Servers was conducted exactly like the PP/PM Servers.

4.3.2 Compatibility Testing

The compatibility testing of Image Comparison Component was conducted in order to ensure its compatibility with the grid structure. This testing was done by designing a grid of Image Comparison Servers in Visual Basic. Visual basic was used for prototyping and compatibility testing because it is relatively easy to program and implement. The Image Comparison Component was found compatible with the grid design after this testing scenario.

4.3.3 Load and Performance Testing

Load and Performance testing of IC Servers was conducted exactly like the PP/PM Servers.
4.3.4 End-to-End Testing

The end-to-end testing of IC Servers was performed by submitting new requests in the UserInfo database in an automated mode using a SQL script. The IC Servers successfully performed Image Comparison and generated sub-Match Lists for all requests.

4.4 Testing Scenarios for the Database Servers

Both the Oracle based UserInfo database and File System based Image/Features database were tested for data integrity, safety, reliability, and availability. UserInfo database was tested in multiple test scenarios to ensure that the data is being read and written by the different servers and the web application properly. The security of the databases was also tested to ensure that they are available to appropriate users and applications only. Both the databases were tested to ensure that all the updates are being saved properly and accurately. The Image/Features database was tested for data security as only authenticated users should be allowed to access these databases. The Database Servers successfully passed all the data security testing scenarios without any troubles and issues.

4.5 End-To-End Grid Testing Scenarios

End-to-End system testing was conducted in two modes. First scenario was the real world mode in which the servers were connected to the Matlab components. In the real world mode of testing, the entire system was tested from request submission to results generation. One PP/PM Server and four IC Servers were used in this mode of testing. Second mode of testing was named simulation mode. In this mode the Matlab components were not connected to the servers. In simulation mode the end-to-end system testing was conducted by running five instances of PP/PM Servers on a single machine and fifteen instances of IC servers running on five machines. The Matlab based tools were replace by simulation code written in java. This was done as only one instance of Matlab can be invoked at a particular time on one machine. We are proud to say that end-to-end system testing was extremely successful and the HP Web-ADIS Grid performed
SYSTEM TESTING & RESULTS

perfectly in all the test scenarios and passed all the requirements set forth for the end-to-end Grid testing.
Chapter 5 UNIFIED MODELING LANGUAGE (UML) DIAGRAMS

This chapter contains the UML diagrams for the HP Web-ADIS Grid. UML stands for Unified Modeling Language and it is the standard language for visualizing, specifying, and documenting the artifacts of a software intensive system. HP Web-ADIS Grid is composed of several collaborating subsystems. These subsystems interact with each other to deliver the overall system functionality. A UML subsystem design diagram is used to identify various system components and to highlight the interaction among them. These subsystems or components are deployed on different servers and client machines. A UML deployment diagram identifies the deployment location of each subsystem.

5.1 Subsystem Design

The HP Web-ADIS Grid is composed of the following subsystems

1. User Interface Subsystem
2. Web Application Subsystem
3. DBStore Subsystem
4. Preprocessing/Potential Match Subsystem
5. Image Comparison Subsystem

The overall subsystem design diagram for HP Web-ADIS Grid has been shown in figure 5.1. The figure also depicts the interactions among various subsystems through their interfaces. Some of these subsystems are further composed of subsystems or submodules. Detailed description of each subsystem and its interfaces has been given.
5.1.1 User Interface Subsystem

The User Interface subsystem handles all interactions with the client. All user requests are processed by this subsystem before they are forwarded to the HP Web-ADIS Grid. This subsystem has two main interface methods Notify and Update. Whenever a user accesses the ADIS tool through the web based user interface, all the information related to the request is collected and passed onto the web server using the Notify interface method. Similarly, when other system components are done with processing and there is a need to update the user interface, the Update interface method for this subsystem is accessed.

5.1.2 Web Application Subsystem

The Web Application subsystem encapsulates functionality of the ADIS web application. This subsystem is further composed of three main subsystems or sub-modules. The detailed subsystem design diagram for the Web Application is shown in figure 5.2.
5.1.2.1 User Authentication Subsystem

As discussed earlier, the authentication process of Web-ADIS is highly secure and ensures security by utilizing an elaborate authentication process. The authentication process checks the login information against the UserInfo database to ensure validity. User information is also checked for validity before the account creation process. User Authentication module of the Web Application subsystem includes all functionality needed for user authentication. This module or subsystem has two interface methods Notify and Inform. Notify is used by the User Interface subsystem to start the authentication process once a request is submitted by the user. The User Authentication module uses its Inform interface method to update other modules of the web application about the status of the login or account creation request. The DBStore subsystem is also accessed during this process through its interface methods.

5.1.2.2 Request History Subsystem

The functionality needed to maintain history of all requests submitted by a user is included in this module. When accessed through its Update interface method by the authentication module, this module accesses the UserInfo database to update the request history for the particular user.
5.1.2.3 Identification Subsystem

The Identification module or subsystem is the part of the web application where the steps just before request submission to the ADIS tool are performed. The subsystem is informed about identification requests through its Update interface method. Another Update interface method is used by the Preprocessing/Potential Match subsystem on completion of request for updating the web application with the final results. The identification module then updates the user interface. This module has a generalization-specialization relation with two sub-modules. The two specializations of this module are the Default Database subsystem and the Reference Database subsystem. The Default Database subsystem deals with processing request which require access to only default ADIS databases. The Reference Database subsystem on the other hand, process requests when user uploads a reference database as a .zip file. Both these subsystems have an Inform interface method. In case of Default Database subsystem this method is used to access the Image/Feature database and also the Preprocessing/Potential Match subsystem. In case of Reference Database subsystem, the Inform interface method is used for accessing a module of Preprocessing/Potential Match subsystem specialized for the extraction of reference databases.

5.1.3 DBStore Subsystem

The DBStore Subsystem encapsulates all interactions with the databases in the HP Web-ADIS Grid. This subsystem has two specializations, a UserInfo DB subsystem to handle interactions with the UserInfo database, and an Image/Feature DB subsystem for the Image/Feature database. Both the UserInfo DB subsystem and the Image/Feature DB subsystem have Get Data interface method for retrieving data from the corresponding database and Set Data method for storing or updating data in the databases.

5.1.4 Preprocessing/Potential Match Subsystem

The Preprocessing/Potential Match subsystem encapsulates all functionality related to preprocessing an identification request and finding a potential match. This subsystem is composed of several other subsystems or sub-modules. The detailed subsystem design
Unified Modeling Language (UML) Diagrams

diagram for the Preprocessing/Potential Match subsystem, with each of its subsystem and their corresponding interfaces, is shown in figure 5.3.

5.1.4.1 Record Preprocessing Subsystem

This subsystem further has two specializations. The Reference DB Record Preprocessing subsystem deals with the preprocessing of records in the uploaded reference database after extraction. The Subject Record Preprocessing subsystem deals with preprocessing of subject records from the default databases. Both these subsystems are accessed through their Preprocess interface methods.

5.1.4.2 Potential Match Search Subsystem

This subsystem includes the functionality for finding potential matches against an identification request. The subsystem is accessed through its interface method Perform Search. The same method on completion of the search process accesses Email Notification and Candidate List Splitter subsystems.

Figure 5.3: HP Web-ADIS Preprocessing/Potential Match Subsystem Design
Unified Modeling Language (UML) Diagrams

5.1.4.3 Email Notification Subsystem

All functionality related to sending email notifications during the identification process is contained in this subsystem. The subsystem is accessed through its interface method Notify.

5.1.4.4 Candidate List Splitter Subsystem

This subsystem performs the functionality of splitting the Candidate List generated as a result of matching process. The splitting is performed to divide the task among several available image comparison servers. Notify interface method is used to access this subsystem. The same method notifies the image comparison subsystem about the candidate lists to start the image comparison process.

5.1.4.5 Match List Merging Subsystem

This subsystem is used to perform the final merging of the partial match lists generated by different image comparison servers in the grid. Merge interface method is used to access the subsystem which also updates the web application on completion of the process.

5.1.4.6 Reference DB Extraction Subsystem

This subsystem deals with the special requests where a reference database is uploaded for matching process. Since the reference database is uploaded as a .zip file, it has to be extracted. Extract interface method is used for accessing the subsystem. This subsystem accesses the Image/Feature database for storing and updating the newly extracted records.

5.1.5 Image Comparison Subsystem

This subsystem encapsulates the functionality of the image comparison servers and is accessed through the interface method Notify.

Figure 5.4 shows the detailed subsystem design diagram highlighting all subsystems in detail and how these subsystems interact through their interfaces to perform the overall functionality of the HP Web-ADIS Grid.
5.2 Deployment Diagram

Deployment diagram for the HP Web-ADIS Grid and interactions among different subsystems have been shown in figure 5.5. Client machine will have the User Interface subsystem or module. The Web Application subsystem will be deployed on the web server, the Preprocessing/Potential Match subsystem will be deployed on the Preprocessing/Potential Match server, and the Image Comparison subsystem will be deployed on the Image Comparison servers. The Database server will maintain the DBStore subsystem.
Figure 5.5: HP Web-ADIS Deployment Diagram
Chapter 6 PERFORMANCE ENHANCEMENT RESULTS

The performance testing of HP Web-ADIS was conducted by comparing a system utilizing a single node with that of a grid based system comprising of four nodes. Each node is a pc with Pentium IV CPU (3.00 GHz) and 2 GB of RAM. Results from each system are compared in the table below.

A total of 309 individual record to record comparisons of dental images were performed. Single node based design took 175 hours to perform the 309 comparisons whereas HP Web-ADIS Grid took only 43 hours for processing the same request. On average, single node based design took 8.5 hours to process each request. HP Web-ADIS Grid, on the other hand, took about 2 hours on average. It is important to note that for single node configuration 1-to-1 comparison of a subject and a reference record took 32 minutes for processing. HP Web-ADIS Grid design with 4 nodes working in parallel took around 8 minutes for the same comparison. Figure 6.1 depicts the performance enhancement of HP Web-ADIS Grid design.

Therefore based on these results we conclude that the grid based design causes an exponential improvement in the performance of ADIS. Further performance improvements can be realized by adding more nodes to the HP Web-ADIS Grid.
### HP Web-ADIS Grid Performance Testing Results

<table>
<thead>
<tr>
<th>#</th>
<th>Candidate List Size (Records)</th>
<th>1 Node Approximate Time Values in Hours (+/- 30 Mins)</th>
<th>4 Nodes Approximate Time Values in Hours (+/- 30 Mins)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>5</td>
<td>1.3</td>
</tr>
<tr>
<td>2</td>
<td>41</td>
<td>26</td>
<td>6.5</td>
</tr>
<tr>
<td>3</td>
<td>24</td>
<td>12.5</td>
<td>3.2</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4.5</td>
<td>1.1</td>
</tr>
<tr>
<td>5</td>
<td>23</td>
<td>9</td>
<td>2.5</td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>3</td>
<td>0.75</td>
</tr>
<tr>
<td>7</td>
<td>10</td>
<td>5.5</td>
<td>1.5</td>
</tr>
<tr>
<td>8</td>
<td>28</td>
<td>15</td>
<td>3.8</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>4</td>
<td>0.8</td>
</tr>
<tr>
<td>10</td>
<td>12</td>
<td>7</td>
<td>1.8</td>
</tr>
<tr>
<td>11</td>
<td>15</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>12</td>
<td>7</td>
<td>3.5</td>
<td>0.75</td>
</tr>
<tr>
<td>13</td>
<td>16</td>
<td>9</td>
<td>2.3</td>
</tr>
<tr>
<td>14</td>
<td>12</td>
<td>6.5</td>
<td>1.7</td>
</tr>
<tr>
<td>15</td>
<td>4</td>
<td>1.5</td>
<td>0.4</td>
</tr>
<tr>
<td>16</td>
<td>10</td>
<td>2.5</td>
<td>0.5</td>
</tr>
<tr>
<td>17</td>
<td>12</td>
<td>6</td>
<td>1.2</td>
</tr>
<tr>
<td>18</td>
<td>11</td>
<td>14</td>
<td>3</td>
</tr>
<tr>
<td>19</td>
<td>15</td>
<td>12</td>
<td>2.9</td>
</tr>
<tr>
<td>20</td>
<td>40</td>
<td>19</td>
<td>5</td>
</tr>
<tr>
<td>Sum (Hours)</td>
<td>309</td>
<td>174.5</td>
<td>43.0</td>
</tr>
<tr>
<td>Sum (Days)</td>
<td>309</td>
<td>7.3</td>
<td>1.8</td>
</tr>
<tr>
<td>Time Per Comparison</td>
<td>1-to-1</td>
<td>32 Minutes</td>
<td>8.1 Minutes</td>
</tr>
</tbody>
</table>
Performance Enhancement Results

Figure 6.1: HP Web-ADIS Grid Performance Comparison

Figure 6.2: HP Web-ADIS Grid Performance Comparison
CONCLUSION AND FUTURE WORK

Chapter 7 CONCLUSION AND FUTURE WORK

This chapter summarizes all the requirements of the HP Web-ADIS Grid system, the way they were solved, and the research outcomes. This chapter also sheds light on possible future enhancements and improvements that can be done to this work.

7.1 Conclusion

Use of dental records is one of the most efficient, cost effective, and accurate ways of identification for deceased or missing individuals in mass disasters. Dental identification is particularly useful in situations where a subject’s body has undergone tissue damage due to the severe weather conditions or other environmental factors. Manual dental identification is a powerful post mortem identification technique but the time required for the process limits its application. This triggers the need for an automated dental identification system. In this thesis dissertation we have proposed a High Performance Web Automated Dental Identification System Grid (HP Web-ADIS Grid). We have addressed several key problems pertaining to automating the dental identification process, developing a web interface for the system, availability of the tool to the users, and performance enhancement. The key goals accomplished are:

- The main bottleneck in the performance of ADIS was the time it was taking during the different stages of processing. To improve system performance, a parallel processing grid based system is proposed. The HP Web-ADIS Grid design utilizes multiple machines to process requests resulting in significant reduction in the processing time. The performance testing of HP Web-ADIS was conducted by comparing a system utilizing a single node with that of the proposed grid based configuration. Results obtained show an exponential improvement in the performance of ADIS significantly reducing the overall time for performing record to record comparisons.
CONCLUSION AND FUTURE WORK

- Quick and easy availability of ADIS was one of the basic requirements for the success of this research. To fulfill this requirement, a web based interface has been developed. The web interface can only be used by users with special access privileges ensuring secure and authorized access to the tool.
- The new and improved web interface allows multiple users to create their accounts and submit multiple requests for processing simultaneously.
- A result notification module has been implemented which updates the users on completion of each intermediate step during the identification process. An email is sent informing the availability of results and processing details.
- The new system enables users to upload their own reference databases for matching purposes.
- The new system supports a configuration interface to allow users to choose different preprocessing, potential matching, and image comparison components. Users can also choose different realizations, parameters, parameter values, and other related variables for processing using this interface.

Forensic scientists would benefit greatly from the proposed grid based Web-ADIS system. The easily accessible web interface and highly efficient grid design would enable faster access to results speeding up the identification process for missing and deceased individuals.

7.2 Future Work

This research solved some of the major problems related to the performance enhancement, automation, and availability of the previous version of Automated Dental Identification System (ADIS). However, there are many interesting problems that have to be addressed for further improvements. Following is the list of some of those problems:

- Non dental features like age, sex, height, weight, etc. are not considered by the current HP Web-ADIS Grid system during the identification process. A lot of
records can be discarded from the reference database even before the Potential Match Search resulting in a more accurate and efficient identification system.

- Currently, HP Web-ADIS Grid does not support dynamic addition of new nodes to the list of available PP/PM and IC Servers. However, the system was designed to support this functionality. Only minor modifications will be required in the current design to make dynamic addition of nodes possible.

- The current Grid design for HP Web-ADIS is LAN (Local Area Network) based and all nodes should be part of the LAN to be part of the Grid. The current design of the Grid can be converted into a web based Grid. The web based Grid can then utilize new nodes from anywhere in the world for processing. The current system was designed by keeping this option in mind and can be modified into a web based Grid.

- Configuration preferences set by users through the configuration interface are saved on the servers but are not being used by the Matlab based components. Modifications to the Matlab based components will be needed to make this option work.

- The current web application, the web pages, and the data that is being sent to and from the user is not encrypted. The web application can be made more secure by using encryption techniques and SSL Digital Certificates.

- Current HP Web-ADIS Grid system utilizes a third party load balancing service which is not smart and divides the incoming requests among different servers in a round robin fashion. Smarter and more intelligent load balancing techniques can be utilized to improve the performance and availability of the web application.

- Each of the Matlab components i.e. Preprocessing, Potential Match, and Image Comparison are prone to errors and these errors are not reported by the Matlab
based tools. There should be a way to validate the results of each Matlab based component before using them for the next step. Furthermore, users should be notified of any exceptions generated by the Matlab components.
I: BIBLIOGRAPHY


I: BIBLIOGRAPHY


[22] Structured Query Language (SQL),
http://databases.about.com/od/sql/a/sqlfundamentals.htm?terms=SQL


[24] Integrated Automated Finger Identification System,,
http://www.fbi.gov/hq/cjisd/iafis.htm


[27] M. Cyran, P. Lane, Oracle Database Concepts, 10g Release 1 (10.1), 2003, Oracle Corporation.