# axiUm

Implementation Guide

# Contents

# Overview

Welcome to the growing number of institution s in the axiUm family.

Exan Academic would like to welcome you to the growing number of institutions in the axiUm family. During the next few months, we will work together towards the successful implementation of the axiUm software, ensuring the best possible return on your investment. You will be assigned an axiUm Implementation Coordinator to assist you throughout the entire project.

The following 10 Step Plan has been designed to outline the key tasks that are involved in the implementation process.

▼**Implementation Time Line:**

1. Form the Implementation Team
2. Select "Go Live" Date
3. Business Process Analysis
4. Systems Requirements & Installation
5. axiUm Installation
6. Software Configuration
7. Training
8. Data Conversion
9. Testing & Final Checks
10. Go Live

For more information, see Implementation Time Line.

# Implementation Time Line

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|

1. Form the implementation team

2. Select "Go Live" Date

3. Business Process Analysis

4. System Requirements & Installation

5. axiUm Installation

6. Software Configuration

7. Training

8. Data Conversion

9. Testing & Final Checks

10. Go Live
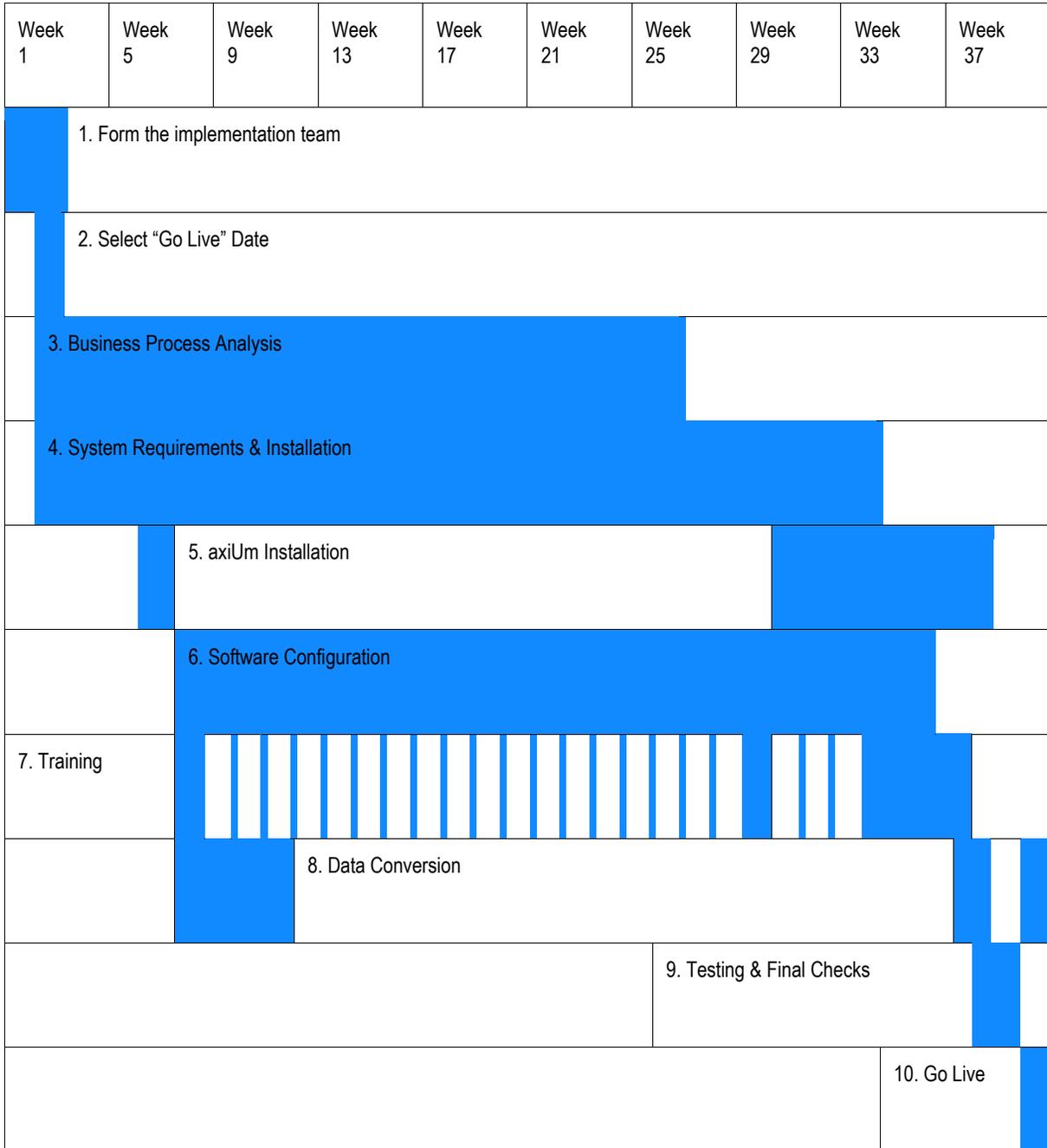
This time line is for sample purposes only. Your project implementation team will ultimately decide on the order and the duration of each step in the project.

# Form the Implementation Team

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|
| | 1. Form the implementation team | | | | | | | | |

The first step to a successful implementation of axiUm is to assign a Project Manager and form the Implementation Team that will manage and guide the project through to its completion.

Project Manager Duties:

- Coordinate activities and assign tasks to Implementation Team and Committees
- Monitor and communicate progress throughout the project with the axiUm Implementation Coordinator
- Address issues and collaborate efforts between the institution and the axiUm Implementation team

> While it is advantageous to have one person designated to the project management, the Implementation Team/Committee is assembled to work with the Project Manager and may be involved in dividing these responsibilities.

The Implementation Team will need to work together to complete the tasks listed in this guide. The team will need to meet regularly to discuss issues, coordinate activities, and update the status of the various aspects of the implementation.

When forming your Implementation Team, these are some factors to consider:

- Assemble a breadth of knowledge including clinical, financial, administrative, and technical representation
- Time available to devote to project
- Decision-making abilities and strength of conviction within the institution

# Select "Go Live" Date

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|---|---|---|---|---|---|---|---|---|---|
| 2. Select "Go Live" Date | | | | | | | | | |

An important step in any successful axiUm implementation is coordinating with your axiUm Implementation Coordinator to agree on a practical date to "Go Live" with the software.

The "Go Live" date is the basis for scheduling milestone events in the implementation process and is what drives the entire implementation process.

Due to the impact of implementing an enterprise-wide solution for your facility, there are many factors to consider with selecting a "Go Live" date. You may also be taking the opportunity to fine-tune (or overhaul) some of your current business processes, which adds complexity to the general roll-out of the system.

In general, we recommend selecting a quiet period in the clinics for your "Go Live", when your operations can absorb a few days of interrupted service to adjust to the new system and processes.

Some other things to consider when selecting your "Go Live" date:

- Logical break in calendar (e.g., end of term or fiscal year end)
- Accreditation status and upcoming assessments
- Holidays (when staff are not available for training)
- Exit date for students in particular (It is not easy to have seniors buy into a system that they will only use for a few weeks).

In planning your implementation, you will also consider the most advantageous method for transitioning to axiUm.

The alternatives include:

**The Comprehensive Approach**

Bringing all clinical areas and features on-line on day one.

**The Progressive Approach**

Bringing all clinical areas on-line at once with a limited set of features. Involves planning beyond "Go Live" date to implement additional modules/features over an extended period.

**The Expansive Approach**

Bringing selected clinical areas on-line with all features on day one. This also involves planning beyond the "Go Live" date to implement clinical areas as they become ready to transition to the new system.

# Business Process Analysis

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|
| 3. Business Process Analysis | | | | | | | | | |

The implementation of axiUm presents you with the opportunity to review your current operations and set a course for the future. This review may lead to adjustments that change a particular aspect of your operation or wide-scale changes that affect the entire enterprise.

For every user/role, processes that are performed in the existing system (or on paper) should be mirrored and processed in axiUm to ensure all vital business processes are accommodated or to uncover potential deficiencies. Through this exercise, users will fine-tune their processes and/or adjust their use of axiUm to achieve the most effective and efficient use of the application.

> To begin this process, please review and respond to the axiUm Business Process Worksheet.

At first, the information collected in the Business Process document will be used to develop a coordinated training plan and assist in the configuration of the new system.

You will also develop standard workflows for common processes that take place regularly in your operation. These workflows will be used to train staff and test the system prior to Go Live.

Initially, the workflows (or scenarios) will outline the current steps involved in a particular process. Over the course of the implementation, each workflow may be adjusted to improve the process if possible, and finally, to map out the steps that will be taken to accomplish the scenario after Go Live. These steps will include actions within axiUm in conjunction with other events that are occurring outside of the scope of the application.

# System Requirements & Installation

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|---|---|---|---|---|---|---|---|---|---|
| | 4. System Requirements and Installation | | | | | | | | |

> Please review sections Technical Requirements and Administrator's Guide to ensure that your technical infrastructure is sufficient and configured to support axiUm in your environment.

With the exception of a few supported peripheral devices, Exan does not make specific hardware recommendations. However, we are available to review and discuss system specifications if you have concerns or questions prior to making a particular hardware purchase.

The time required to complete this step is dependent upon the current state of technology at the institution and the desired level of the axiUm implementation. For instance, if you are planning to move to chair-side, electronic chart operations and you do not have any networking in your clinic, then there is significant work to be done.

Some of the key decisions in this phase include:

- Implementing in a two-tier versus three-tier environment
- Selecting a method of authentication (primarily for faculty approvals)
- Considering student-owned notebooks to be used as clinic workstations

# axiUm Installation

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|---|---|---|---|---|---|---|---|---|---|
| | | 5. axiUm Installation | | | | | | | |

The axiUm technical support group will install or assist you through the system installation procedure.

Prior to software installation, ensure the database server has:

- Oracle database server software installed (preferably with the latest patchset applied).
- A functional starter database created
- An operational instance of the database

Prior to software installation, workstations should be setup with:

- Windows operating system
- Oracle client software, allowing them to connect to the database server
- There should be at least 4-5 stations available for the Initial Training session

> For detailed installation instructions please see Server Installation Guide and the Workstation Installation Guide.

The initial server installation will create a testing/training instance of the axiUm database. This instance is populated with the axiUm "Best Of" starter dataset that is comprised of system setup from various other axiUm implementations. Using this dataset, you will be able to operate axiUm right away without any further configuration.

Typically, the testing/training environment is used exclusively during the early stages of implementation. Adjustments to the system settings/configuration and trial data conversions are performed in this environment. At some point close to "Go Live" date, the configuration from the test/train database will be used to create the Production database. All settings, master files, and user information will be moved into the Production environment and all test data will be cleared to be ready for any final adjustments and the import of converted data.

Client installation may occur at any stage of the implementation, and the total time required is dependent upon the deployment environment (two-tier or three-tier) and whether imaging/ghosting of clients is a possibility.

## Software Configuration

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|
|  |  | 6. Software Configuration |  |  |  |  |  |  |  |

axiUm Version 4.81

To accommodate the needs of any institution, axiUm is highly customizable. Configuration of the system, including the setup of master files, security roles, reports, customized forms, and system options, is vital to the success of the project.

When the software is installed, it is configured with the "Best Of" starter dataset. This allows you to work through the software, making use of all functionality, without first having to invest the initial time on configuration.

Once you are comfortable with the functionality of the application and are prepared to customize the configuration, our support team can assist with removing some or all of the pre-loaded information.

axiUm Configuration includes:

- Maintenance – Master file setup
- PowerAdmin – Security Role Definition
- Station Options – Desktop/Station Settings
- Customized Reports
- Customized Forms, Functions, and Processes

# Training

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|---|---|---|---|---|---|---|---|---|---|
| 7. Training | | | | | | | | | |
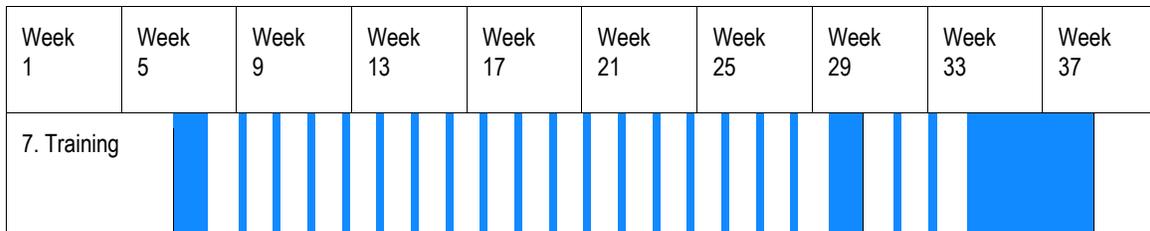
Training of the axiUm system is conducted through a combination of on-site visits and web conference sessions.

Training is an ongoing process that is occurring throughout the implementation period.

The training is divided into four separate phases:

- Initial Overview & Training
- Focus Sessions
- Complete Training –Train the Trainers
- Training of Personnel/End Users

## Initial Overview & Training

This session occurs with an initial site visit shortly after the software installation and is designed to give your Implementation Team a good understanding of the axiUm application and it's features.

Training will be fast-paced and interactive; participants should be prepared to focus on the sessions with limited distraction. Ideally, a room will be available with axiUm-configured workstations and a projection unit. This will allow the participants to learn and practice in a hands-on environment.

## Focus Sessions

To assist in the system configuration and increase the system knowledge of the Implementation Team and subject experts, regular web conference sessions are scheduled.

These sessions will focus on specific aspects of the application and it's configuration. Each subject area (e.g., Patient Registration or Scheduling) will be covered with respect to the features of the module, the configuration required to customize it, and the various options/methods of operating in this area of the application. The sessions should be scheduled weekly on a specific day/time (e.g., every Wednesday at 11am). The topic for each week should be finalized at least a few days prior to the session.

In addition to representatives from the Implementation Team, it is also recommended that appropriate personnel for the subject area are invited to participate.

## Complete Training –Train the Trainers

Once the majority of the system configuration has been completed and the business analysis has rendered the common workflows, this session is scheduled. This will involve a second site visit and the classroom training of key individuals who will, in turn, be responsible for training your end users. Following this session, the participants should be confident in their knowledge of the relevant aspects of the system.

It is recommended that the responsibility of training end users be spread out across multiple subject experts (not necessarily members of the Implementation Team).

## Training of Personnel/End Users

The final training sessions will involve the communication of knowledge to all personnel, including administrative staff members, faculty, and students. These sessions should be conducted using the workflows to condense the application to the scope that applies to the role of the end user being

trained. This training is typically performed over an extended period nearer to the "Go Live" date, but not at the very last minute. Should issues arise from the training with regards to workflows or system setup, there must be enough time available to correct them.

If changes have been made to the responsibilities of particular user groups, then this must also be covered during these training sessions.

# Data Conversion

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|---|---|---|---|---|---|---|---|---|---|
| | | | 8. Data Conversion | | | | | | |

When migrating to axiUm from another software system(s), you will need to decide what to do with your existing data. Specifying what data to convert is one of the most important parts of the implementation process, and it is something that should be considered early on in the project.

> Please review Data Conversion for details of data that can be converted and how it should be presented.

To ensure accurate migration of your present data into axiUm, we will run at least one trial data conversion, ideally months prior to your "Go Live" date. It is extremely important that your team takes the opportunity to review the converted data and verify it's integrity and accuracy. Further, this trial conversion data should be used to test your processes and workflows and to facilitate the training of your staff.

Our support team will conduct a thorough validation of your "before and after" record counts and summary totals and carefully review your data to verify that, from our perspective, the conversion is accurate. Please note that any data that is found as non-compliant with our database requirements will be sent back for your review.

The final data conversion will coincide with your "Go Live" date. You will need to factor in the time required to shut down and extract the data from your old system(s) when planning your "Go Live" procedure.

# Testing & Final Checks

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|
|        |        |        |         |         |         | 9. Testing & Final Checks | | | |

It is essential that all of your critical business functions have been tested in axiUm prior to the "Go Live" date.

This is best accomplished by a combination of tests using the details of the Business Process Analysis as a foundation.

This step should confirm the following:

- Data has been accurately converted.
- Hardware and database are installed and are correctly configured
- Training has been completed successfully
- Business processes are sufficiently supported
- All reports and forms required are available and configured in axiUm
- Institution is ready to Go Live with the system

> Through experience, we have developed the Implementation Checklist to facilitate the verification process in preparation for Go Live.

# Go Live

| Week 1 | Week 5 | Week 9 | Week 13 | Week 17 | Week 21 | Week 25 | Week 29 | Week 33 | Week 37 |
|--------|--------|--------|---------|---------|---------|---------|---------|---------|---------|
|        |        |        |         |         |         |         |         | 10. Go Live | |

The project culminates during Go Live with all steps having led towards this ultimate milestone. By this date, you will have tested all facets of the system to ensure a successful and effective transition to axiUm.

The final data conversion takes place and goes smoothly because all issues have been resolved by this time.

Exan Academic support staff will be on site and available to provide additional assistance during the first few days.

To facilitate the "Go Live" period, we recommend:

- Clinics are lightly scheduled for the first day
- axiUm experts are available and easily identified by end users in need of assistance
- Use of two-way radios, PA system, or other means to expedite resolutions
- Setup of a central headquarters for regular assembly of team members to discuss and resolve issues

> Once you have transitioned to axiUm and gone Live with the system, product support is offered by the Exan Academic Technical Support Team (academic@exansoftware.com).

> To ensure continued satisfaction with your axiUm system, we recommend additional follow-up training offered by Exan Academic either by phone, web conference sessions, or on site.

# APPENDIX A: DATA CONVERSION

## Overview

In order to allow the axiUm conversion team to quickly access your data, we request that your data is extracted from your current system and presented in character delimited text/ASCII files.

We will also accept data presented in spreadsheets or have the data imported directly into the axiUm Oracle database (into temporary tables used for the data conversion).

> Each data conversion that we do for axiUm is customized based on the needs of the institution. We do not have standard formats that data must be prepared in before we proceed.

## Data Field Listing/Dictionary

When sending the data files, please include accompanying documentation is that defines the contents of each conversion datafile. For example, a data field listing/dictionary for all data that you are planning to convert.

**For Example**

Text file of patient demographic data has the following (sample) data:

12345~Harris~William~03-MAR-1975~M~A

12346~Stevens~Nancy~19-JUN-1978~F~C

We would like a document that explained the columns like:

| Field | Detail |
| --- | --- |
| Pt. Rec Num | Number - unique identifier for patient in system |
| Pt Last Name | 30 chars |
| Pt First Name | 20 chars |
| Pt Birth Date | DD-MON-YYYY format |

| Pt Gender | M (male), F (female) |
|---|---|
| Pt Status | A (active), C (closed), D (deceased) |

# Conversion Review

Once all the tables and data sets have been submitted, the axiUm data conversion team will review the data provided and make recommendations of where it should be fitted into axiUm's infrastructure.

> It is necessary to discuss what data should be converted into axiUm with your implementation group and your axiUm conversions representative in order to convert all data properly and avoid inconsistencies.

Our experience has been that bringing everything over has not been a very successful approach for most institutions coming over from older systems. The data is often found to be erroneous, incomplete or of little value. In many cases, our technicians must delete the converted data after the institution goes live and identifies many problems with their data. Also, consider that the more data (types) converted will increase the amount of time required to develop, run and test/verify the data conversion.

# What Can be Converted

We have successfully converted the following standard data types in previous implementations.

The conversion of these data types is included in the contracted charge for the data conversion:

- Operator/User information (including students, faculty and staff)
- Fee Schedules
- Patient Demographics (Names, Addresses, Phone Numbers, etc.)
- Patient Recalls
- Patient Medical Alerts
- Patient Chart/Clinical/Account notes
- Patient to Provider/Student assignments
- Patient Opening Balances
- Patient Treatment Plans

In addition to these standard items, we have also worked with and successfully converted the following data types.

Depending upon the complexity involved in converting the data, there may be an additional charge associated with the data conversion:

- Patient Appointments
- Patient Insurance Information (Carrier and Plan Information)
- Patient Employer Information
- Patient Referral Information
- Patient Procedural History (treatments)
- Patient Prescriptions
- Patient Full Financial History (very reluctantly!)
- Payment Plan contracts

# Steps Involved in Data Conversion

▼**To convert data:**

1. Institution decides what data they want to convert, including data sources, data types, and filter criteria if possible.
2. Data conversion files are reviewed by axiUm Conversion group.
3. Correspondence between axiUm staff and institution to clarify conversion details and limitations.
4. Trial conversion run(s) are performed into the training or testing database.
5. Institution to review data for integrity and accuracy. Business processes and workflows should be tested against the data.
6. Based on review, changes to the data conversion are made. If major revisions, then repeat steps 4 & 5.
7. Prior to Go Live, axiUm team to send a list of data to be converted in the final conversion, and when the files are needed by to ensure a successful Go Live.
8. Institution to give a final sign-off on the data to be converted and when all files will be delivered to axiUm team.
9. Final conversion is performed to coincide with Go Live date.

# APPENDIX B: BUSINESS PROCESS

# APPENDIX C: ADMINISTRATOR'S GUIDE

## Introduction

This guide is your primary source of introductory, post-installation, configuration, and technical information for using axiUm. This document does not include any information regarding the functionality and features of the application itself.

It is not the intention of this guide to document all tuning concerns relevant to the Oracle database. There are numerous publications that deal with this subject and their content is in much greater detail than could possibly be presented here.

### Pre-requisites

This guide assumes that the reader is familiar with both Oracle database products and the Microsoft Windows operating systems.

### Intended Audience

This guide is necessary for anyone installing, configuring or administering the axiUm product.

## axiUm Technical Overview

axiUm is an electronic health record and clinic management system created by Exan Academic Software. axiUm is a client/server application, with the database stored on the server and the application's presentation on the Windows-based PC client. The software was developed originally using Borland C++ version 5.01 and is now maintained and enhanced using Microsoft Visual C++ .Net. Some reports and customizable forms that are used within the application were created using Crystal Reports versions 7 through 11.

The client software consists of over 300 program files including .exe, .dll, .bmp files, and others. These files total approximately 100 MB in size. In addition to the axiUm program files, the client must be installed with Oracle Client software.

## Database Dictionary

The axiUm database is comprised of standard Oracle database objects:

- Tables
- Indexes
- Sequences
- Triggers
- Views
- Functions
- Stored Procedures and Packages

Presently, all referential integrity between the axiUm tables is maintained by the application. There are currently no integrity constraints created in the database.

**Tables**

The tables in the axiUm database have been named using eight or fewer characters. The fields in each table are case-sensitive, so to query a particular field the field's name must be enclosed in double quotes. The field representing a table's primary key will almost always be assigned one of the following names: "Id", "Code", "Auto" or "<TableName>" (e.g., "Patient" or "Office"). Occasionally, the primary index will be made up of two or more fields in the table. In either case, the primary key field(s) will usually be the first field(s) defined in the table description.

The following field datatypes are created and used by the axiUm tables:

| Field Datatypes | Definition |
|---|---|
| CHAR(x), where x is between 1 and 255. | The CHAR datatype is used for many alphanumeric fields in the database that are of fixed or short length (less than 30 characters). |
| VARCHAR2(x), where x is between 1 and 4000. | The VARCHAR2 datatype is used for alphanumeric fields of variable or long length (greater than 30 characters). |
| NUMBER(5) | The NUMBER(5) data type is used for short number fields. In many cases the possible values for the field will be limited to 0 and 1 (Boolean field values). |
| NUMBER(10) | The NUMBER(10) data type is used for storing long whole integers. The most common use of this data type is for unique table id's (generated by a sequence). |

| | |
|---|---|
| NUMBER | The NUMBER data type is used for any numeric field that cannot be stored as a NUMBER(5) or NUMBER(10). These fields will usually include decimal precision. The most common usage is for monetary amount fields. |
| BLOB | The BLOB data types are used to store binary field data. The axiUm application has functions to read and write from these fields in the database. The field data cannot be viewed or altered from outside of the application. |

Tables created as part of the axiUm Data Dictionary should not be altered – this will cause errors with the axiUm application. Exan will provide SQL scripts for axiUm version updates. Each table is assigned to one of six data tablespaces based on its growth rate and storage characteristics (see TablespacesTablespaces). As part of your tuning procedure, it may be necessary to move a table out of its default tablespace and into another. While this is fine, you must be aware that version update SQL scripts will assume that a table is in its default location – so the SQL may require modifications. The creation of additional tables within the axiUm schema will not affect the functionality of the application.

**Indexes**

Each table has a unique primary key and may have one or more secondary indexes defined in the database. These indexes may not be altered as they are referenced by the axiUm application. You may create additional indexes to improve performance as needed. The index names will start with the name of the table that they are related to, for example, Patient_Chart. The primary index is assigned the name <TableName>_Primary.

**Sequences**

There are many sequences created and used by axiUm. Most sequences are used to generate unique ids when inserting records into a particular table. These sequences have a standard naming convention: <TableName>_<Field Name of primary key or sequenced field>_SEQ. For example, PATIENT_PATIENT_SEQ and VACATION_ID_SEQ are sequences used by the Patient and Vacation table respectively. Other sequences are used to generate sequential numbers for form printing (e.g., RECEIPT_RECEIPT_SEQ and APPTCARD_APPTCARD_SEQ).

When inserting data into an axiUm table using a non-axiUm process (during data conversion for example), you must be careful to adjust the current value of the sequence (if the table has a corresponding sequence). Otherwise, errors will occur when using axiUm to add a new record to the table.

The sequences are created with default properties with the exception of the starting value that may have been adjusted to allow for imported data. Because of the way that Oracle caches sequence information, it is common that non-sequential values will be attained from a sequence.

You will often see a jump of 20 values when viewing record ID's for a selected table. If this is a concern for a particular table then the sequence may be altered with the "no cache" option. This will ensure that values are sequential with no holes. Oracle warns that this may have a minor performance impact.

**Triggers**

Triggers have been added to the axiUm database to perform auditing on various tables and actions. You may create additional triggers (and tables) to perform auditing on other tables in the database.

**Views**

Views have been introduced into the database for commonly queried sets of data. This allows some of the complexities of the database to be hidden from the user that is building custom reports.

**Functions**

Functions are used to return calculated scores, values, or information that is interpreted and displayed in many areas within the application. For example, a customized function is used to authenticate passwords created by each user, for example, to force password to be at least eight characters and contain at least one number.

**Stored Procedures & Packages**

There are stored procedures and packages compiled in the axiUm database containing PL/SQL procedures and functions that perform batch-processing duties in the application.

# Connection to Database

When axiUm is executed on a client workstation, the application attempts to connect to the appropriate database based on information configured during the client installation. A second connection is optionally made during startup for the reporting portion of the application. This results in two server sessions opened for each client running axiUm on the database. There is an option to prevent the second session from connecting at startup. Using this option, the client will connect and disconnect from the database as needed when printing forms or reports. As this will degrade the performance of printing forms, it is recommended that certain workstations be set to connect at startup and keep two sessions open until exiting from axiUm. The most obvious example is a cashier workstation that prints receipts and statements throughout the day during normal operation.

axiUm connects and communicates with the Oracle database using Oracle Objects for OLE (OO4O). OO4O is a COM-based database connectivity tool that combines seamless and optimized access to

Oracle databases. It is a native driver that generally provides the fastest performance on Windows clients to Oracle databases.

axiUm was developed as a two-tier client-server application with most processing occurring on the client and data residing only on the server. However, axiUm can also be configured to operate in a three-tier architecture through the use of Application Servers (such as MS Terminal Server or Citrix). This type of implementation has proven to be extremely beneficial for setting up remote access and for moving to chairside operations.

## axiUm Transactions

The majority of transactions issued by axiUm during on-line processing are extremely small. They will commonly involve the insert, update, or delete of just a single row. Each transaction is automatically committed when it completes. There are some multi-transaction processes that commit all individual transactions at the end of the process.

There are also some large batch processing events that will generate larger transactions affecting multiple rows in the database. These transactions will account for less than 1% of the database activity during normal operation.

It is important to note the distribution of read-type transactions (SELECT statements) versus write-type transactions (INSERT, UPDATE, and DELETE statements) issued during normal use of the axiUm database. Monitoring of production systems over a distinct period of time revealed that approximately 95% of all statements are "reads" compared to only 5% "writes". This information will be discussed further in Rollback Segments.

## Database Access & PowerAdmin Security

For the axiUm database, all objects are created in the same schema with a single owner. The owner user account is the only Oracle user required by axiUm. Each client workstation running axiUm will connect to the database using the owner's username and password. So that there is no need for any application user to know either value, the username and password are encrypted and stored on each workstation.

axiUm does not require the installation of SQL Plus or any other database-authoring tool on the client. If other applications will be used to access the axiUm database, then additional users should be added to provide access without distribution of the owner's username and password.

Before gaining access to axiUm, each user will be prompted to enter a valid user login ID and password. This user information is stored in the axiUm database and maintained with the PowerAdmin Security Application.

PowerAdmin is a stand-alone application developed by Exan to create and administer axiUm users, user groups and their security rights within the axiUm application. Levels of security must be created and each user is assigned to a security level. The definition of each security level involves defining the access rights to each axiUm module, dialog/window, menu item, report, button, and field in the application.

## Swipe Card Security Access

Magnetic swipe cards may be used to gain access to axiUm, bypassing the requirement to type in a valid username and password at the axiUm login screen. They may also be used in various areas of the application by faculty or administrative staff as a method of authorizing information entered into the system.

The minimum requirement for the encoded information on the magnetic stripe is the username and password of a user in the axiUm database. Both fields accept alphanumeric values with the username 12 characters in length and the password 12. The card may have a single or multiple tracks and the data can be encoded on any of the readable tracks.

You may also encode an effective and/or expiry date on the magnetic stripe. This is useful when creating temporary access cards that will be discarded at the end of the day or week.

## Version Control & Auto-Update Facility

At least two axiUm application updates are released every year. The releases are set to coincide with the summer break (July/August) and the winter break (December). Clients are notified of the enhancements included with a new release in advance of the release date. Further notifications are sent to indicate that the release is available for testing purposes (beta release) and, finally, for deployment in a production environment (full release). Clients are encouraged, but not forced, to stay up-to-date with the current release.

There is an auto-update facility in the application that prompts a client station, running an old version of axiUm, to download the updated axiUm program files from a designated shared directory. For the autoupdate to function properly, the axiUm client must have read-access to the directory where the updated program files are located. The client must also have write-access to the local axiUm program file directory. The location of the auto update folder is stored in the "PracticeVersionSrc" field.

axiUm has built-in version control that prevents axiUm clients from accessing an axiUm database running on a different release version. The version control works based on the comparison of two version numbers each time a client starts axiUm: one number is compiled into each release of the axiUm application, and the other number is stored in the axiUm database. If the numbers match, then axiUm will run. If the database version is newer than the client version, then the client will be prompted to load the update (and the auto-update procedure begins). If the first three digits of the client version are

greater than the database version, then the client is shut down – axiUm will not downgrade the client, nor can it auto-update the database. If the first three digits of the client version are the same as the database version, but the last two digits of the client version are greater, then the client has been manually updated with a patch release and operation can continue. For explanation of the version numbering system, see Version Numbers.

▼**To install an axiUm update:**

1. Copy the updated axiUm program files into the shared auto-update folder.
2. Update the axiUm database using the SQL script created by Exan.
   This script should be reviewed and altered as required by the DBA.
3. Copy the updated PowerAdmin executable file to each administrator workstation.
   PowerAdmin will NOT auto-update. This step must be done manually.

## Version Numbers

Because axiUm is always undergoing development and change, several releases of the application can be in use at any time. The five-digit version number is used to fully identify the release of the software.

An example is shown below.

**4.81.0.3**

Version Number

Maintenance Release Number

Major Release Number

Patch Release Number

| Number Type | Definition |
| --- | --- |
| Version Number | The version number is the most general identifier. In combination with the major release number, they define a general version of the axiUm software. For example, 3.0. |
| Major Release Number | The major release number is used to signify a considerable change in the functionality of the software. The number will be increased with each full release of the application. |
| Maintenance Release Number | The maintenance release number signifies different releases of the general version, starting with 0. The maintenance release number increases when new features or enhancements are added to an existing version. |

| Patch Release Number | The patch release number identifies a specific release version that fixes problems in the maintenance release. A patch release contains fixes for serious bugs that cannot wait until the next major release. The first version of a release will always have a patch release number of 00. With any patch release, there will be no database changes required. As mentioned in Version Control & Auto-Update Facilitypatch releases can be manually installed on specific workstations without fully deploying an update. This allows an institution to update critical systems as needed to resolve specific issues that are affecting limited users. |
|---|---|

To check your current version of axiUm you can open the About axiUm window from the application's menu (under Help). You may also query the Version field from the PRACTICE table in the axiUm database to find the current Database version.

# Database Server Considerations

The following discuss general guidelines for managing axiUm.

## Server Platform

axiUm should be installed with the latest version of the Oracle 11g server and client software. The application is functional on previous versions of Oracle, but for many obvious reasons including improved performance and functionality, the Oracle 11g software should be used.

The axiUm database may be installed on any server platform that complies with Oracle software. There are no axiUm application processes that run on the server. When selecting a platform for the database server, two main factors are usually weighed: performance of the system versus cost (initial and administration) of the system. An Intel server running Windows presents a low-cost solution, but may not be able to match the performance provided by a Unix system. Due to its ease of use, however, Windows is often a logical choice when configuring a new server.

Regardless of the platform selected, there will be documentation printed by Oracle that is pertinent to its installation and use.

## Disk System

Disk I/O problems are probably one of the most common performance issues, either in terms of contention or slow disk writes; disk reads are generally only found to be a problem on some incorrectly configured disk arrays. The goal on any system is to distribute files across disks and controllers so that I/O is evenly balanced and contention is minimized.

The axiUm database consists of six data tablespaces and six index tablespaces (see Tablespace ). It is generally accepted that locating data and index tablespaces on separate disks minimizes disk head movement. A very useful extension of this approach, which is commonly overlooked, is to ensure that these disks are spread across separate controllers wherever possible. This improves the performance of concurrent input/output operations compared with what you would obtain if the disks were simply connected to separate channels on the same controller.

The increasing use of RAID arrays and logical volume managers does not make this advice obsolete. While these provide good automatic load balancing across disk devices, some control is inevitably lost in the process. For example, if a data tablespace and its associated index tablespace are located on the same RAID disk, an indexed access to a table may produce activity on the same physical disk. In general, RAID 0+1 provides the best combination of speed (with striping) and fault tolerance through redundancy (with mirroring). RAID 5 is a popular choice, but due to the complex nature of write operations, especially the parity calculations, there is a high performance penalty on writes to the database. Operations on rollback segments and redo logs are typically write-intensive, with the datafiles being accessed sequentially. For this reason, rollback segments, redo logs and datafiles hosting temporary tablespaces should NOT be located on RAID 5 arrays.

## Creating Multiple Databases

axiUm is an evolving application with major updates released regularly. Due to this and the critical nature of the data, there is normally a second instance created for training and testing purposes. There may also be a need for a development instance, to develop and test add-on reports, customizations or utilities.

The best way to create and maintain the additional instances is to make copies of the production database. To maintain the test system, the process of overwriting it with the production data should be done regularly.

## Tablespaces

The tablespaces used by axiUm data objects (tables and indexes) are created based on the storage characteristics and management requirements of the objects that are stored in each tablespace.

| Tablespace | Row Insertions | Row Deletions |
|------------|----------------|---------------|
| Static     | Low            | Low           |
| Medium     | Medium         | Low           |

| High | High | Low |
| --- | --- | --- |
| Grow | High | Low |
| Dynamo | High | High |
| Blob | Low | Low |

There are commonly six tablespaces created for tables and a corresponding set of six tablespaces for the indexes of those tables. The six tablespaces represent different levels of usage. In addition, there is a 13th tablespace, the HUGE tablespace, that has been created to contain really large data objects (like images, documents or signed forms).

| Tablespace | Details |
| --- | --- |
| Static | Includes objects that remain static in size. Rarely are records inserted or deleted from tables in the STATIC group. The objects are primarily code files. |
| Medium | Includes objects that will tend to grow in size via regular, but limited volume of inserts of records. Rows are rarely deleted from these objects. |
| High and Grow | Objects in either of these groups will experience a high volume of row inserts. They are divided into two groups because of the large size of the objects contained in them. Combined these two groups commonly make up over 80% of the total size of the database. |
| Dynamo | Consists of objects that will have similar amounts of rows inserted and deleted. These objects are the most likely to become fragmented over time. |
| Blob | Includes objects that contain variable-length binary fields. As this data's size is somewhat difficult to estimate, it is separated into its own group. |
| Huge | Includes fast growing objects that contain variable-length binary fields. This data is typically moved onto a separate disk/file system as the data can represent over 80% of the entire system. |

In addition to these 13 tablespaces, the database will typically contain the SYSTEM tablespace, an UNDO tablespace and a Temporary tablespace. Other tablespaces may be created as deemed necessary by the DBA, but it is advised that a temporary work tablespace be created for support and maintenance of the database.

The approximate I/O distribution of the axiUm tablespaces is shown below. Based on these estimated I/O weightings, an initial plan for the database server's disk layout can be derived. These statistics were gathered by monitoring existing axiUm production databases. Depending on how the application is

used, there is a high variance in the statistics (one of the databases monitored showed that PGROW accounted for 50% of the I/O with PMEDIUM at 25% and PHIGH only 2%). For this reason, the actual I/O distribution for each database must be monitored to properly reevaluate the disk layout.

| Tablespace | % of Total Database I/O |
|------------|-------------------------|
| PHIGH | 37 |
| PMEDIUM | 19 |
| PGROW | 16 |
| IHIGH | 10 |
| IGROW | 8 |
| PHUGE | 4 |
| IMEDIUM | 2 |
| PSTATIC | 1 |
| PBLOB | 1 |
| PDYNAMO | 0.5 |
| IDYNAMO | 0.4 |
| ISTATIC | 0.2 |
| IBLOB | 0.1 |

Now that the I/O for each tablespace has been determined, we can discuss the approximate disk space requirements of each tablespace. It is impossible to accurately estimate the size of the axiUm database for a particular institution without first gathering some information.

The main three factors contributing to the database size are:

- Amount of historical data to be converted into the database.
- Volume of data that will be entered into the database.
- Optional modules of axiUm that will be used.

Having said this, a generalization of the disk space requirements can be derived from the data collected during previous installations. It is safe to say that 20 GB of disk space would be adequate to initially store the axiUm database for any production environment. Annual aggregation of between 10 GB (for a non-paperless implementation) to 100 GB (for a paperless operation) would be common. For a basic development or testing environment, a minimum of 10 GB of storage is suggested.

The following table shows the approximate weightings of the tablespaces with regards to storage requirements. This distribution may vary depending on how axiUm is implemented, but in most cases the top 6 tablespaces will make up more than 95% of the database size.

| Tablespace | % of Total Disc Space Required |
|------------|-------------------------------|
| PHUGE | 45 |
| PHIGH | 23 |
| PMEDIUM | 11 |
| PGROW | 7 |
| IHIHG | 6 |
| IGROW | 6 |
| IMEDIUM | 1 |
| PSTATIC | 0.3 |
| PBLOB | 0.3 |
| PDYNAMO | 0.3 |
| IDYNAMO | 0.1 |
| ISTATIC | 0.1 |
| IBLOB | 0.1 |

The temporary tablespace for the axiUm database should be at least 500 MB in size to allow for large sorts to occur without error.

The UNDO tablespace, while normally requiring less than 5 MB (see Rollback Segments), must allow for the occasional large transaction to be performed. A second rollback tablespace may be created to be

used by these large transactions. This allows the primary rollback tablespace to remain at a small size. A minimum of 100 MB of space should be allocated to the rollback segment tablespace(s).

After monitoring your production database, you may decide to further divide the highly used tables into additional tablespaces. You may create new tablespaces and/or modify the tablespace assignments of the axiUm tables and indexes. Note that if you do so, then you may need to modify scripts provided by Exan for subsequent version upgrades. These scripts will assume that an object is in its original tablespace.

## Indexes

There are many places within axiUm in which users can perform ad hoc queries on the database. There may also be unforeseen queries run from outside of axiUm using internally developed reports or processes. The default indexes created for the axiUm database cannot possibly handle all of the possibilities. As a result, there will be queries executed that will perform undesirable full table scans. Although these queries may be relatively few in number, they can consume enough of the available system resources to negatively impact every other user of the database. To improve the performance of such resource-intensive queries, you may be required to create additional indexes in the database.

> Indexes that are initially created by Exan scripts should not be dropped or altered. These indexes are required by the axiUm application, and any change may cause the application to fail. It is recommended to check with axiUm support first before creating any custom indexes.

As records are deleted from a table, the matching entries are deleted from the table's indexes. Unlike tables, Oracle does not reuse the space released within an index. Therefore, indexes on tables from which rows are frequently deleted should have their indexes periodically rebuilt in order to avoid space fragmentation within the index. As discussed in the previous section, the indexes that fall into the category of frequent row deletions are assigned to the IDYNAMO tablespace.

## Rollback Segments

> If using the UNDO tablespace feature of Oracle, then the sizing/allocation of Rollback Segments is not required.

**Size**

As previously discussed (see axiUm), over 99% of all transactions issued during normal processing involve a single statement against a single row in the database. We can assume that since the largest row in an axiUm table is less than 4 KB in size, it follows that the largest of the common transactions generates less than 16 KB of rollback information.

The rollback segments must also be able to handle the other large batch processing transactions that make up less than 1% of all activity. These transactions are generated by axiUm, but do not occur during normal axiUm usage. Processes such as the month-end utility, zip code import or printing a batch of held claim forms will all generate large transactions. The size of these transactions will be dependent upon the amount of data involved. For a realistic volume of data (ex: print 1000 claims in a single batch), the largest transactions will generate approximately 1000 KB of rollback information.

Based on this information, it is suggested to use a rollback segment extent size of 32 KB or 64 KB with an optimal storage setting between 320 KB and 640 KB.

**Quantity**

There are many documented formulas for estimating the number of rollback segments required for a database. The estimates are based on the number of concurrent users divided by some constant value ranging from 4 to as much as 40.

Number of Rollback Segments = CEIL

(Maximum Number of Concurrent Users / x)

where 4 <= x <= 40

In normal production use, the axiUm database activity is dominated by read-type transactions. Based on the average distribution of 98% reads to 2% writes upon the axiUm database, you can conclude that, on average, out of every 100 concurrent users, only two are writing to a rollback segment at any given moment. To allow for peak levels of writing activity, this estimate can be pushed to one in 15 concurrent users writing data to the rollback segments. Therefore, to initially estimate the number of rollback segments to create, use 15 as the value for x in the equation.

Number of Rollback Segments = CEIL (Maximum Number of Concurrent Users / 15)

For example, if you estimate that during peak usage that there will be 100 concurrent users, then you should initially create 7 online rollback segments.

This is only a suggested starting point. With monitoring of the V$WAITSTAT and V$ROLLSTAT tables, you may determine that further adjustments are required. If, for example, there are a lot of transactions waiting for a rollback segment, then the number of segments should be increased.

**Handling Large Transactions**

For database maintenance and batch processing, it may be appropriate to create a second rollback tablespace with a single large rollback segment. This tablespace and segment would be kept offline during normal processing periods, and only brought online as required to handle large transactions.

In this way, online processing needs can perform efficiently using relatively small rollback segments, while large batch processes and database maintenance can proceed without generating errors or having an effect on the regular rollback segments.

# Database Access

Access to the axiUm application is controlled within the application. Users log in to axiUm with their own unique axiUm id and are validated by the application. This means that all users of the AXIUM schema get logged in as AXIUM.

The use of a single database account removes the administration of user accounts from the DBA's tasks. The axiUm application administrator will be responsible for all user authorizations. From an account management perspective, the DBA will only be involved in creating the axiUm owner's account and any accounts created for access to the axiUm database from outside of the axiUm application. For example, to create a query-only access account for use by a reporting tool.

# Suggested Parameters

There are several database initialization parameters you can set (via the init.ora initialization parameter file or through the SPFILE) to improve the performance of axiUm. The major parameters to set are listed in the following sections.

**DB_BLOCK_SIZE**

| | |
|---|---|
| **Default Value** | 4096 (4 KB) |
| **Suggested Value** | 8192 (8 KB) |
| **Reason** | To maximize performance within axiUm, increase the Oracle block size used. Rather than using a 4,096-byte block size, increase it to 8,192 or 16,384. Increasing the database block size will reduce the percentage of each block that is devoted to overhead. Performance will be improved as more data can then be read with each I/O. You cannot easily change the block size of a database after it has been created. |

**DB_CACHE_SIZE**

| Default Value: | 33554432 (33 MB) |
|---|---|
| Suggested Value: | 268435456 (268 MB)<br>(depending on how much memory is available to Oracle on the server). |
| Reason | If you can increase the size of SGA, then use up the additional space with this parameter. 33<br>MB is the default value and is far too small. |

For axiUm, you should be able to attain a 90% hit ratio for the buffer cache during normal operation. If your hit ratio is less than 85%, then try increasing the DB_CACHE_SIZE parameter to improve performance.

**DB_FILE_MULTIBLOCK_READ_COUNT**

The number of blocks read during each physical read is determined by the setting of the DB_FILE_MULTIBLOCK_READ_COUNT parameter. The number of blocks read at a time is limited by the I/O buffer size of your operating system. If your operating system buffer size is 128 KB, and you database block size is 8 KB, then the DB_FILE_MULTIBLOCK_READ_COUNT parameter should be given a value of 16. If you set it to a lower value, the performance of any full table scans will be adversely affected.

**OPTIMIZER_INDEX_CACHING**

| Default Value | 0 |
|---|---|
| Suggested Value | Suggested value: 90 |
| Reason | The default value of 0 is for a data warehouse. For an OLTP system, it should be set around 90. |

**OPTIMIZER_INDEX_COST_ADJ**

| Default Value | 100 |
|---|---|

| | |
|---|---|
| **Suggested Value** | Suggested value: 20 |
| **Reason** | The default value of 100 is for a data warehouse. For OLTP, must be a lot lower |

## PGA_AGGREGATE_TARGET

| | |
|---|---|
| **Default Value** | 16777216 (16 MB) |
| **Suggested Value** | 268435456 (268 MB) |
| **Reason** | This is one of the most important parameters to tuning an Oracle database. 16 MB is the default value and is far too small |

## SGA_MAX_SIZE

| | |
|---|---|
| **Default Value** | 320309648 (320 MB) |
| **Suggested Value** | 454631852 (450 MB)<br><br>(depending on how much memory is available to Oracle on the server). |
| **Reason** | The memory usage on the server is split between SGA and PGA. Your PGA should be set to 100 to 250 MB (PGA_AGGREGATE_TARGET) and the rest of the memory should be allocated to the SGA.<br><br>Oracle is consistently improving the performance of the cost-based optimizer. Although rule-based optimization is still available (as of Oracle 9i), its use is not recommended. For the axiUm database, the cost-based optimizer must be used. Set OPTIMIZER_MODE=CHOOSE or ALL_ROWS. |

Effective use of the cost-based optimizer requires that the axiUm tables and indexes be analyzed regularly. This can be done via a weekly or nightly process that re-analyzes all objects in the database.

There are a few options for gathering these statistics:

- The GATHER_SCHEMA_STATS procedure within the DBMS_STATS package is the recommend procedure to use. If your schema name is AXIUM, then the following command will run the procedure: execute DBMS_STATS.GATHER_SCHEMA_STATS ('AXIUM');
- The ANALYZE_SCHEMA procedure within the DBMS_UTILITY package can be used to analyze all objects in a schema. If the schema name is AXIUM, then the following command will run the procedure: execute DBMS_UTILITY.ANALYZE_SCHEMA ('AXIUM', 'COMPUTE');

These procedures will run for an extended period and should not be executed during normal operating hours.

# APPENDIX D: WORKSTATION SETTINGS

axiUm workstation specific settings are stored in the Oracle database. This allows for centralized administration of all settings and support for Terminal Services deployments. There are a few parameters that are needed prior to connecting to the database and these are stored in the axiUm.ini and axiUmdb.ini files.

Upon the first login on a newly installed workstation, a dialog appears allowing the user to specify what settings to use for this new station:

- axiUm defaults
- copy from a template (see below on how to create a template)
- copy from another station
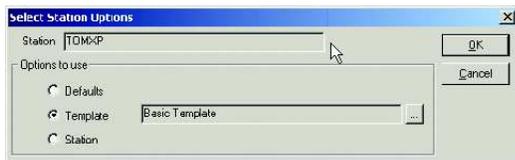


## Maintenance of Workstation Settings

Any axiUm workstation can maintain its own settings or the workstations settings of any other axiUm workstation (or group of workstations) given that the user has a sufficient security level.
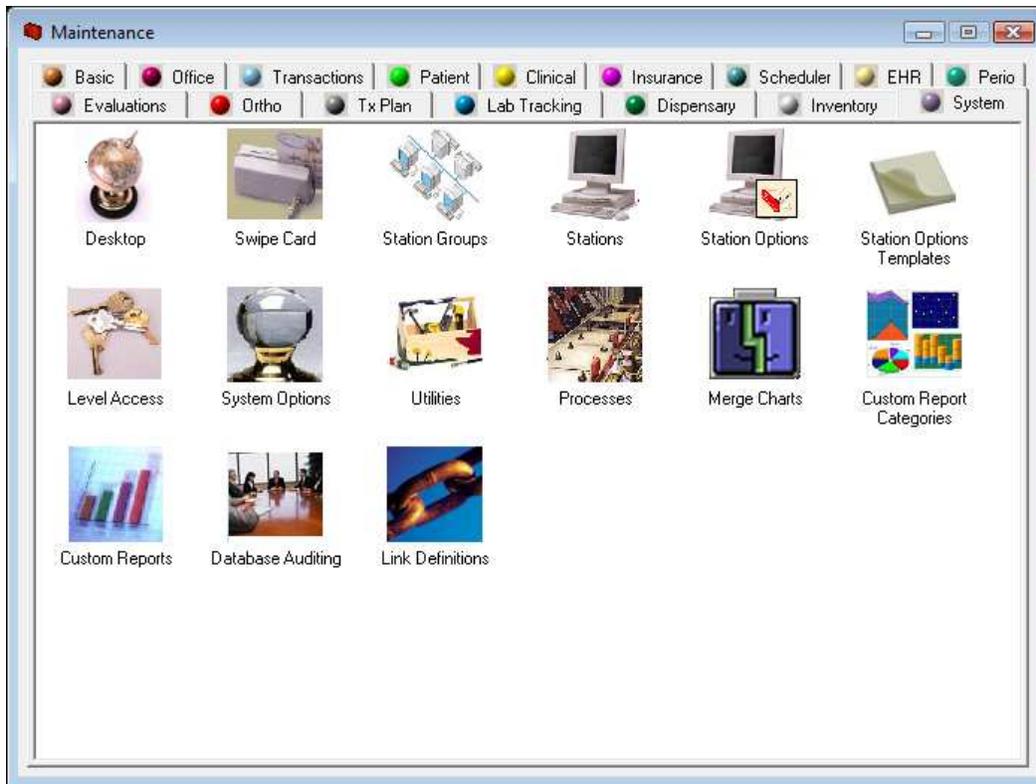
# Maintenance – System

 **Station Codes**

The Station Codes listview shows information about every station that has logged into axiUm. It gives a detailed explanation of each station:

- Internal axiUm station ID assigned it
- Date it was registered
- Windows network name of the terminal/workstation
- Last login date
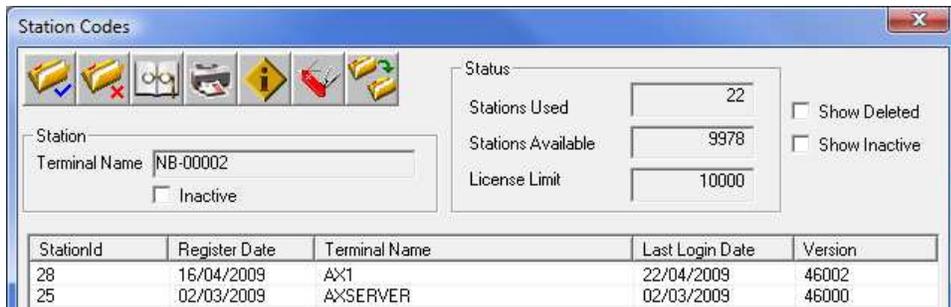
The Status section informs the user of the axiUm License Limit, the number of stations used and the remaining available.



Selecting Show Deleted Stations will cause the list to refresh and display deleted workstations as well as active ones. Deleted station entries can not be restored and once deleted the station will re-register upon its next login.

 **Assign Station to Groups, Station Options, & Copy Options**

A station can be assigned to one or more groups within the Station Codes dialog. Select the "I" icon (Alt-I) and add the group to be assigned to. The utility tool icon (Alt-O) brings up the Station Options dialog (see below) and Copy Options (Alt-C) allows the use of the Copy Station Options dialog (see below).
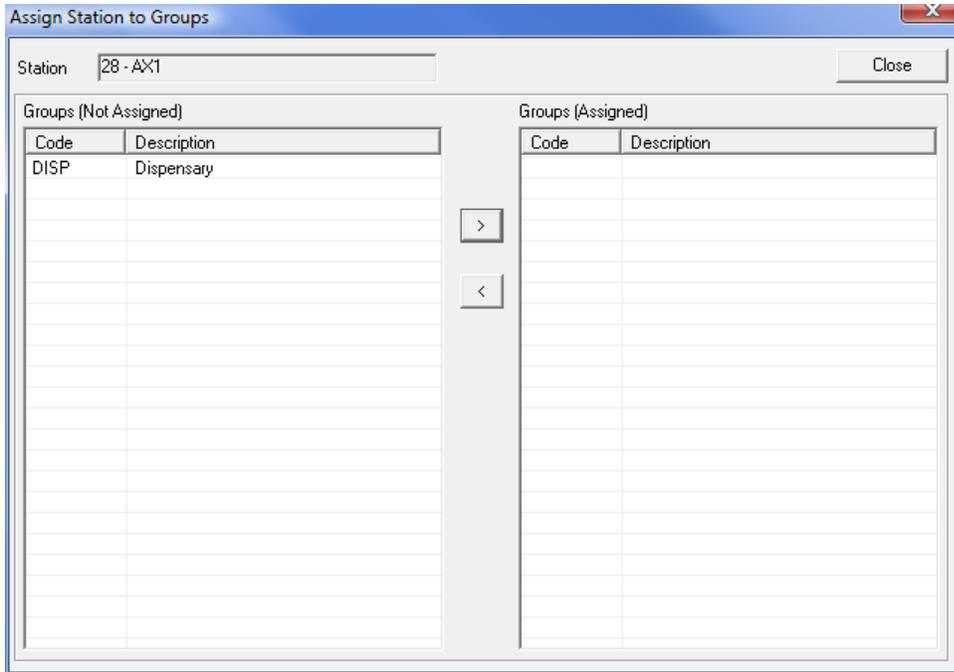
**Station Options**

The Station Options dialog access is via the Utility Tool icon in the Station Codes dialog and the Station Options Dialogs icon in Maintenance. In Station Options, the user can view and modify any station option value. Most of the fields have editing disabled and are for information purposes only. These disabled field values are in the axiUm database and can be altered using SQL if language or other customization is required. The Comments field is editable if the Station Options dialog access is through the Station Options Dialogs icon and store notes up to 4000 characters in length in the INIDEF record.
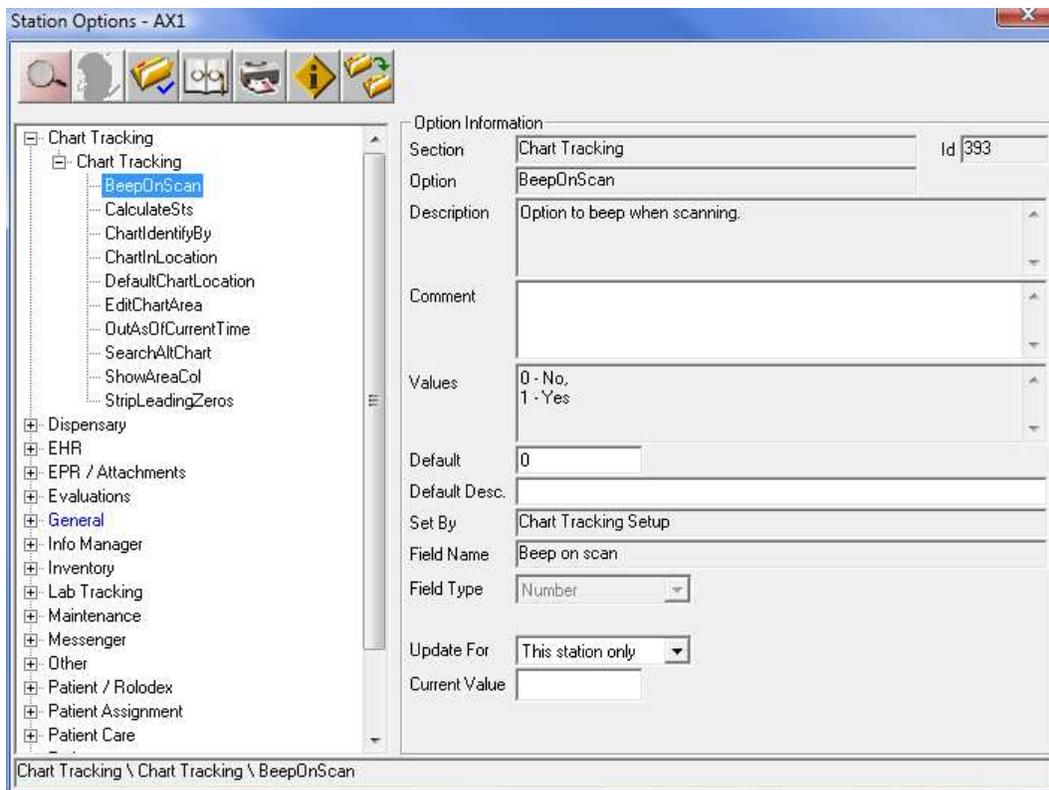
An option changed from the default is color coded blue.

If accessed from the Utility Tool icon in the Station Codes dialog the Update For selection and the Current Value edit field are available to change the value for the current station, all stations or stations in this station's group(s).

Option Information Fields:

| Fields | Definition |
| --- | --- |
| Section | Module to which this option belongs |
| ID | Internal Id code for the option |
| Option | Option name |
| Description | Displays a description of the option. |

| | |
|---|---|
| Comment | Enter a comment about the option. |
| Values | Possible values for use in this option |
| Default | Default value (if there is no default this field is blank) |
| Set By | Where the option is set from |
| Field Name | Field name in the application that sets this option. |
| Field Type | Type of entry (string, number etc.) |
| Default Desc. | Description of the values used in Default field |

Access is through the Station Options Dialogs icon:

| Fields | Definition |
|---|---|
| Update For | Update This Station Only, All Stations, or Stations in Group |
| Current Value | View or Change the current station value. If the value is blank, the default is used |

## Find Station Option

The find station option window (accessed via the icon  Alt-F) allows the user quickly find the option that needs to be entered or modified.
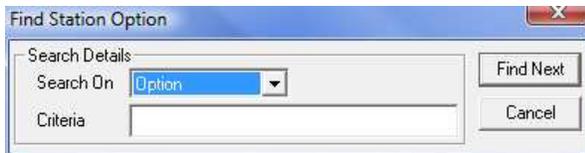


The Search On selection list allows the changing of the field the search will use:

| Fields | Definition |
|---|---|
| Option | Indicates to find the option named in the criteria field. |

| | |
|---|---|
| Set By | Indicates to find the option where the Set By field contains the text entered in the criteria field. |
| Field Name | Indicates to find the option whose prompt in axiUm matches the text in the criteria field. |

Criteria is the text to base the find on. The search will find the text entered in the field selected based on the string appearing anywhere in the field in upper or lower case. For example, the text "chart" would match "Chart", "ChartInLocation" and "AutoChart".

### Clear & Save Data

The Clear and Save icon ( Alt-L) in the Station Options dialog (available only when accessed from the Utility Tool icon ( Alt-O) in the Station Codes dialog) will clear out the customized current value selected and save the change after prompting the user "Are you sure you want to clear this station option?".

### Copy Station Options

Accessed via the Copy Options icon in Station Codes or Copy Station Options icon in Station Options sub-dialog, this dialog will copy options from other stations or predefined templates.
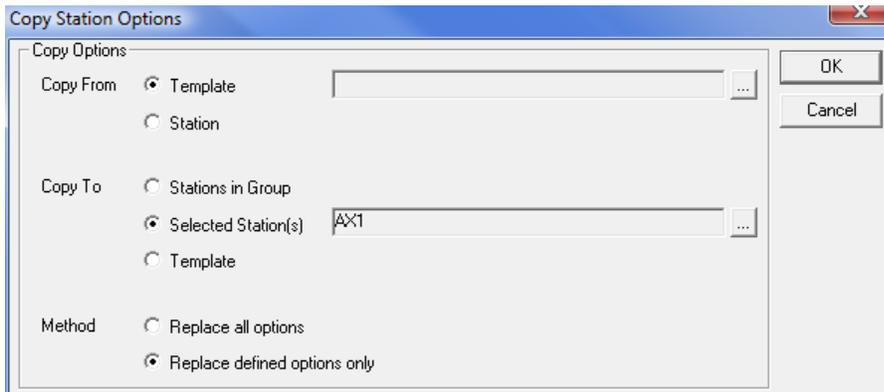


Notice that the sub-dialog version allows the options to be copied anywhere whereas the Station Codes version of the dialog only allows the copying of options to the current station.
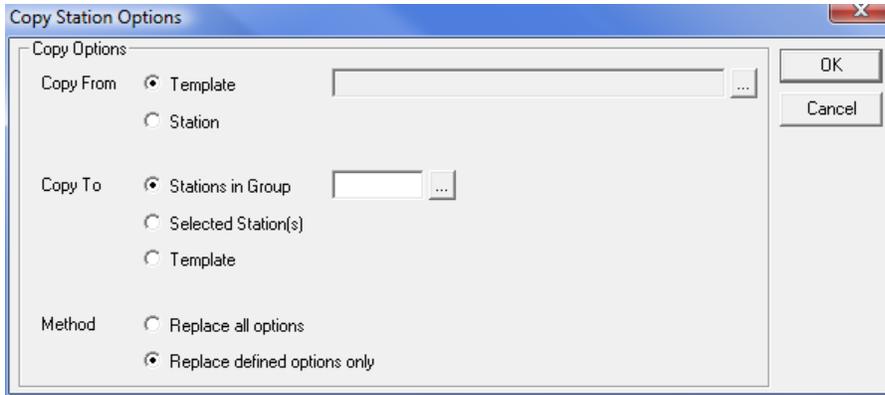
| Fields | Definition |
|---|---|
| Copy From Template | Template will copy station settings from the selected template. |
| Copy From Station | Station copies from the station selected. |
| Copy To Stations in Group | Station in Group: will copy to all station(s) that are members of the selected group. |
| Copy To Selected Station(s) | Selected Station(s): use this to copy to a specific station or multiple selections. |
| Copy To Template | Template: copy to the selected template. |
| Method - Replace all options | Erase any existing settings for the station or station group or template |
| Method - Replace defined options only | Copy only the customized settings from the item selected in the Copy From fields. |

 **Station Groups**

Station Groups allow for the creation of logical groupings for workstations.

Code is a unique string (8 characters) to identify the group of stations. The Inactive checkbox will remove the use of a Station Group without deleting the definition. The Description field (30 characters) explains the purpose of the station group.

 **Station Group Assignment**

The Station Group Assignment dialog is accessed through the  (Alt-I) button on the Station Codes window. Station assignment can be to none, one or several groups. The left list is groups to which the station is not assigned. On the right are groups the station is assigned. Selection is click one or more group listings and click the central arrow buttons to change the station's group memberships.



 **Station Options**

 **Station Option Update**

This dialog is accessed via the  button Alt-I on the Station Options window when an option is selected. It allows the user to view and or modify this station option for one or more stations.
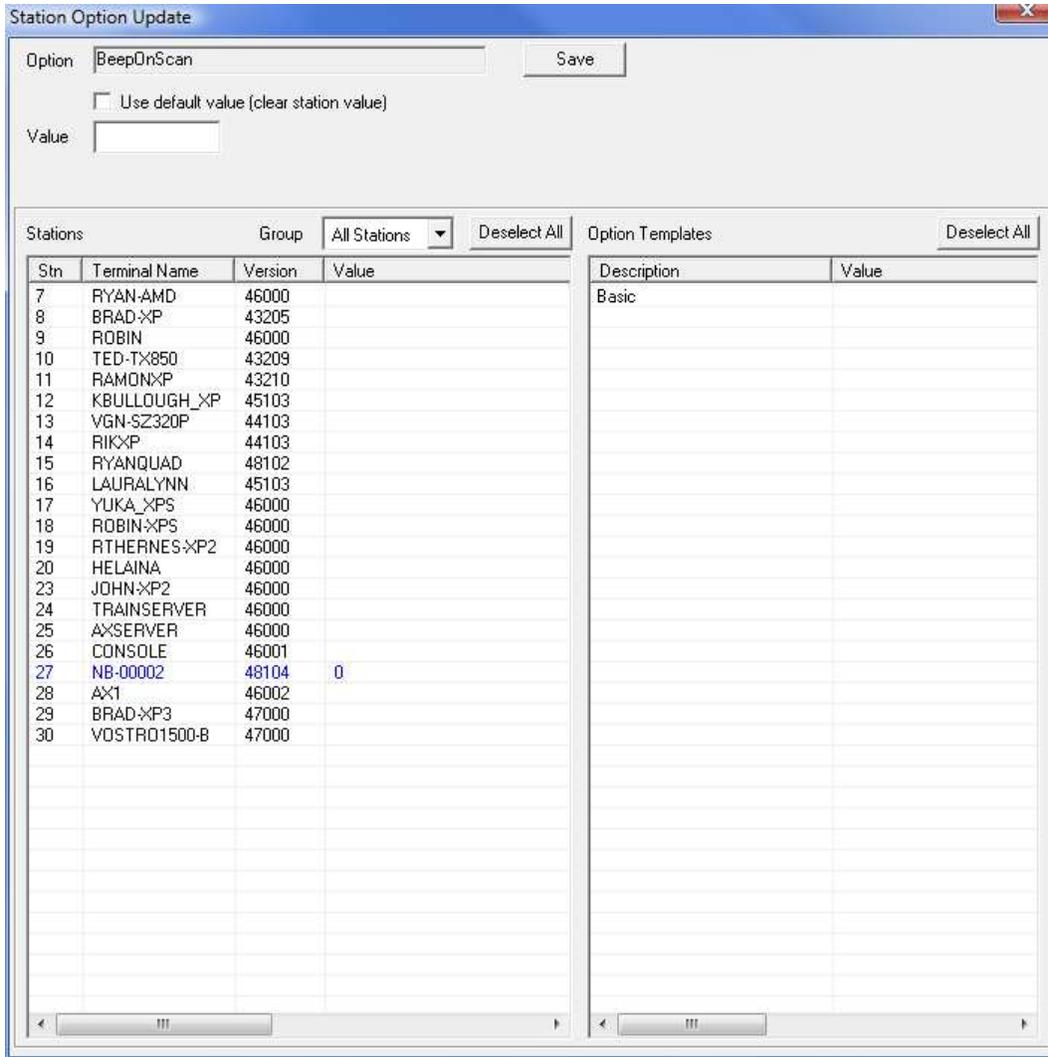
The Value field allows a new value to be set and the Group list sets the group of stations being displayed.

▼To save a change to the option for a selection of stations and/or templates:

1. Select names from the lists.
2. With the names selected, click on the save button at the top of the window.

# Station Options Templates

Station Options Templates allows the user to define a template for use when copying station options to individual stations or station groups.
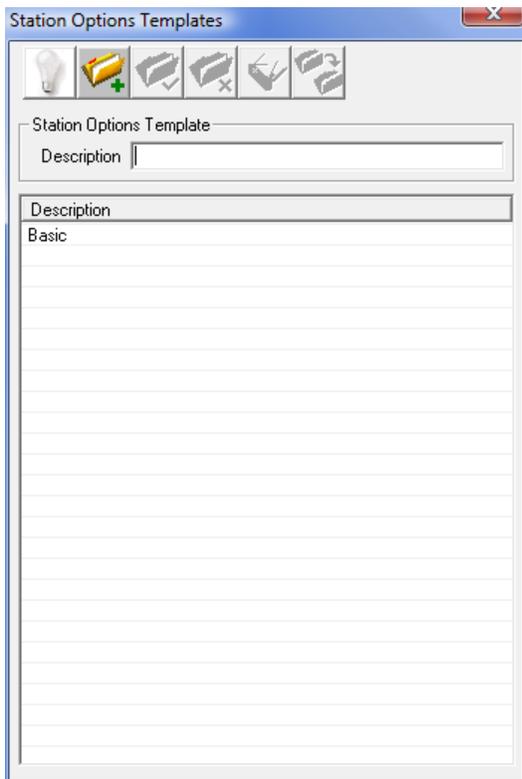


After creating the template description name, the user then clicks on the Station Options  icon to define options, or can click on the Copy Options  icon to copy the options from a station or other existing template.
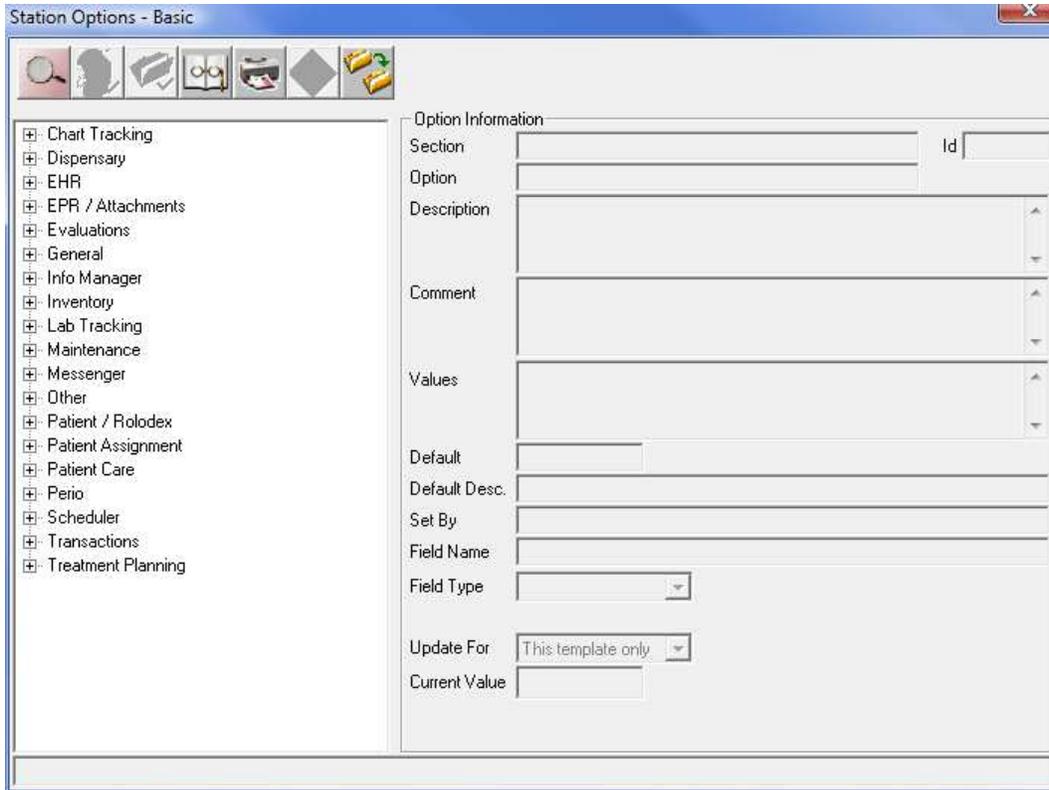
# axiUm 3.2 AXIUM.INI Values

There are still some INI file values used with axiUm. These are all located in the axium.ini and axiumDB.ini files.

| Section | Setting | Values | Default | Description | Notes |
|---------|---------|--------|---------|-------------|-------|
| Database | DBEngine | 1 - BDE<br>3 - OO4O | 3 | This option determines the database interface layer to use for the initial connection. Please note that BDE is only for Oracle 8 and below, and OO4O is for Oracle 9 & up. When the initial connection is valid, axiUm will check the Oracle version. If the DB Engine does not support the version, it will reset (see note) to the correct value and retry the connection. | This is set manually, Automatic reset will occur in some circumstances. When both BDE & OO4O interfaces exist and are setup for use with axiUm, axiUm can determine the correct setting based on the BDE registered databases and Oracle version detected. |

| Database | DbAlias | User Defined | Power | This specifies the database to connect to. For BDE, it is the BDE SQL Links alias for the database; for OO4O, it is the section name of the AXIUMDB.INI where the database connection parameters are stored. | Set manually. |
|---|---|---|---|---|---|
| Database | CacheBlocks | between20-127 | 127 | Specifies the number of blocks held in memory for caching at any one time (OO4O only). | Set manually. |
| Database | FetchLimit | between 20-200 | 200 | Specifies the number of elements of the array into which data is fetched from Oracle (OO4O only). | Set manually. |
| Database | FetchSize | increment of 1024 | 4096 | Specifies the size, in bytes, of the buffer used for retrieved data (OO4O only). | Set manually. |
| Database | PerBlock | greater than or equal to 16 | 16 | Specifies the number of slices that are stored in a single block (OO4O only) | Set manually. |
| Database | SliceSize | greater than or equal to 128 | 256 | Specifies the number of bytes used to store a piece of data in the cache (OO4O only) | Set manually. |
| Debug | SQLTrace | Trace level | 0 | 4 (basic tracing + bind variable substitution)<br><br>8 (basic tracing + database wait events)<br><br>12 (basic tracing + bind variable substitution + database wait events) | Set manually. |
| Debug | Debug | 0 - No<br>1 - Yes | 0 | This is for internal use. It turns logging on and writes support information to txdebug.txt. | Set manually. |
| Debug | Database | 0 - No<br>1 - Yes | 0 | This is for internal use only. Turns debugging on or off related to transaction processing (OO4O only). | Set manually; will write out debug information to the file oo4otx.txt if the flag is on. |
| Debug | Crystal | 0 - No<br>1 - Yes | 0 | This is for internal use only. It displays Crystal SQL in a message box and logs SQL to | Set manually. |

crystal.txt.

| | | | | | |
|---|---|---|---|---|---|
| Debug | InfoMgr | 0 - No<br>1 - Yes | 0 | This is for internal use only. It displays internal SQL in a message box and logs to one of: ediclaim.txt, apexpert.txt, nquery.txt. | Set manually. |
| Office | Citrix | 0 - No<br>1 - Yes | 0 | This determines whether the given station is a Citrix work station. | Set manually. |
| Update | CheckUpdate | 0 - No<br>1 - Yes | 0 | If 1, check update directory to always perform the auto-update during axiUm start-up. | Manually set. |

# APPENDIX E: CREATE USER

```
CREATE USER "AXIUM"

     IDENTIFIED BY "PASSWORD"

     DEFAULT TABLESPACE "TBLSPC_PHIGH"

     TEMPORARY TABLESPACE "TEMP"

     PROFILE DEFAULT

     ACCOUNT UNLOCK;

GRANT "CONNECT" TO "AXIUM";

GRANT ALTER ANY CLUSTER TO "AXIUM";

GRANT ALTER ANY INDEX TO "AXIUM";

GRANT ALTER ANY PROCEDURE TO "AXIUM";

GRANT ALTER ANY SEQUENCE TO "AXIUM";

GRANT ALTER ANY TABLE TO "AXIUM";

GRANT ALTER DATABASE TO "AXIUM";

GRANT ALTER SESSION TO "AXIUM";

GRANT ALTER TABLESPACE TO "AXIUM";

GRANT ANALYZE ANY TO "AXIUM";

GRANT CREATE ANY INDEX TO "AXIUM";

GRANT CREATE ANY PROCEDURE TO "AXIUM";

GRANT CREATE ANY SEQUENCE TO "AXIUM";

GRANT CREATE ANY TABLE TO "AXIUM";

GRANT CREATE ANY VIEW TO "AXIUM";

GRANT CREATE PROCEDURE TO "AXIUM";

GRANT CREATE SEQUENCE TO "AXIUM";

GRANT CREATE TABLE TO "AXIUM";

GRANT CREATE TRIGGER TO "AXIUM";
```

```
        GRANT DELETE ANY TABLE TO "AXIUM";

        GRANT DROP ANY INDEX TO "AXIUM";

        GRANT DROP ANY PROCEDURE TO "AXIUM";

        GRANT DROP ANY SEQUENCE TO "AXIUM";

        GRANT DROP ANY TABLE TO "AXIUM";

        GRANT DROP TABLESPACE TO "AXIUM";

        GRANT EXECUTE ANY TYPE TO "AXIUM";

        GRANT EXECUTE ON DBMS_LOCK TO "AXIUM";

        GRANT INSERT ANY TABLE TO "AXIUM";

        GRANT QUERY REWRITE TO "AXIUM";

        GRANT SELECT ANY SEQUENCE TO "AXIUM";

        GRANT SELECT ANY TABLE TO "AXIUM";

        GRANT UNLIMITED TABLESPACE TO "AXIUM";

        GRANT UPDATE ANY TABLE TO "AXIUM";

        GRANT SELECT ANY DICTIONARY TO "AXIUM";

        ALTER USER "AXIUM" DEFAULT ROLE ALL;
```

# APPENDIX F: DEPLOYING AXIUM WITH CITRIX

The axiUm design allows it to run in a Citrix environment with only minor changes to the setup and a few specific settings in Citrix. The following setup options must be set for the configuration of axiUm as a shared Citrix application.

> **!** When installing axiUm on the terminal server, you will need to put the server in Install Mode.

## Maintenance – System – System Options

Check the Support Citrix Stations box to engage general Citrix support.



## axiUm.ini

The Citrix option under the Office section of the axiUm.ini file is required to be set to 1. This sets up messaging on a central application model so that axiUm messages, alerts and patient check-in will function correctly. The axiUm.ini file is only required on the application server and not on work stations.

# Citrix & Terminal Services Settings

Set the Idle Session limit to a value slightly higher than the axiUm Auto Logout value.

This will ensure that any session will terminate if the application is left in a non-terminated state.
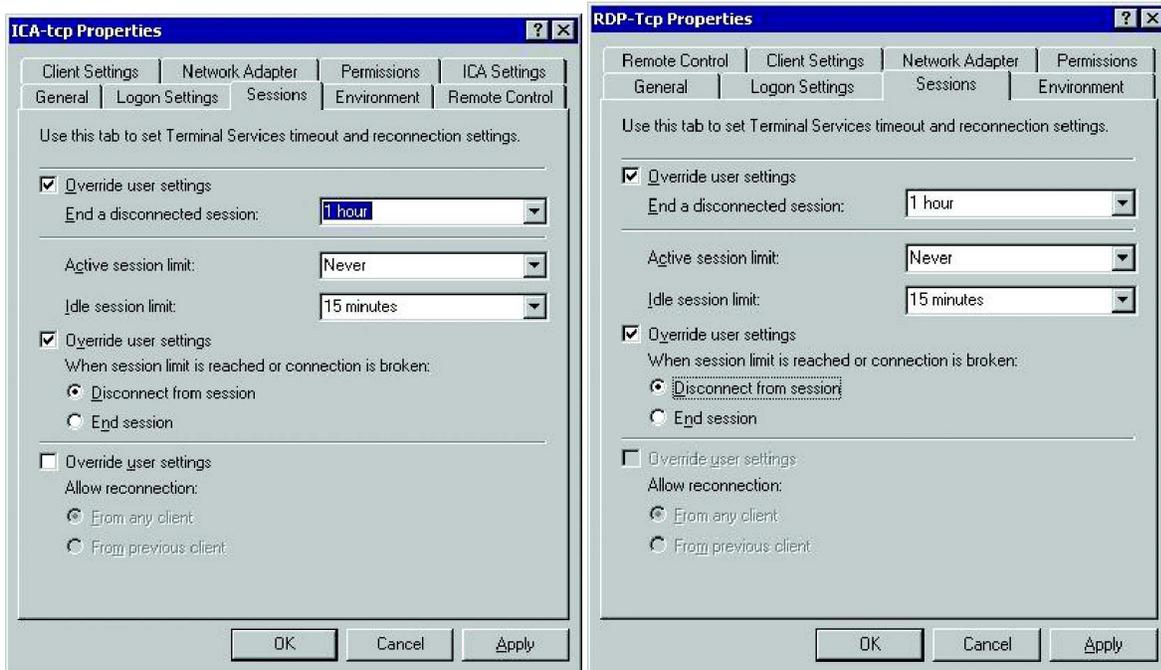


# Data Execution Prevention (DEP)

If DEP is used on the application server be sure to include axiUm on the exception list. Failure to do this can result in axiUm not being able to run in a Citrix configuration. Windows DEP settings are found in: Settings – Control Panel – System – Advanced Tab – Performance Settings – Data Execution Prevention.

# Citrix Application Limits

To run axiUm in Citrix using remote terminals axiUm.exe must be published on the server. When adding axiUm be sure that the "Allow only one instance of application for each user" is set when specifying the Application Limits.
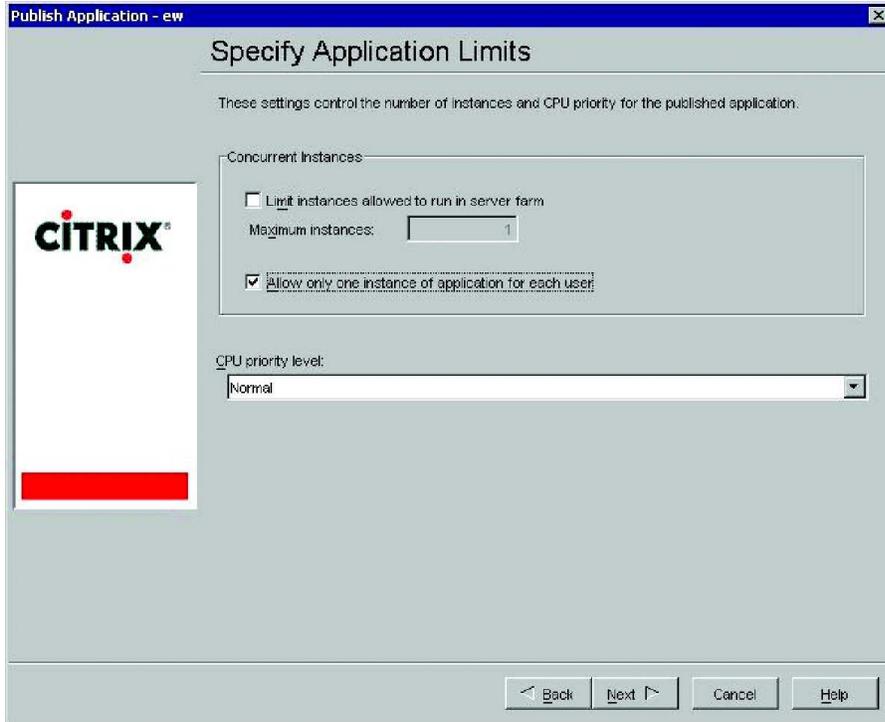
# Enabling axiUm Support

To allow the Exan Support Department to connect and run axiUm through Citrix, publish axiUm with the additional command line parameter "/support" as shown.
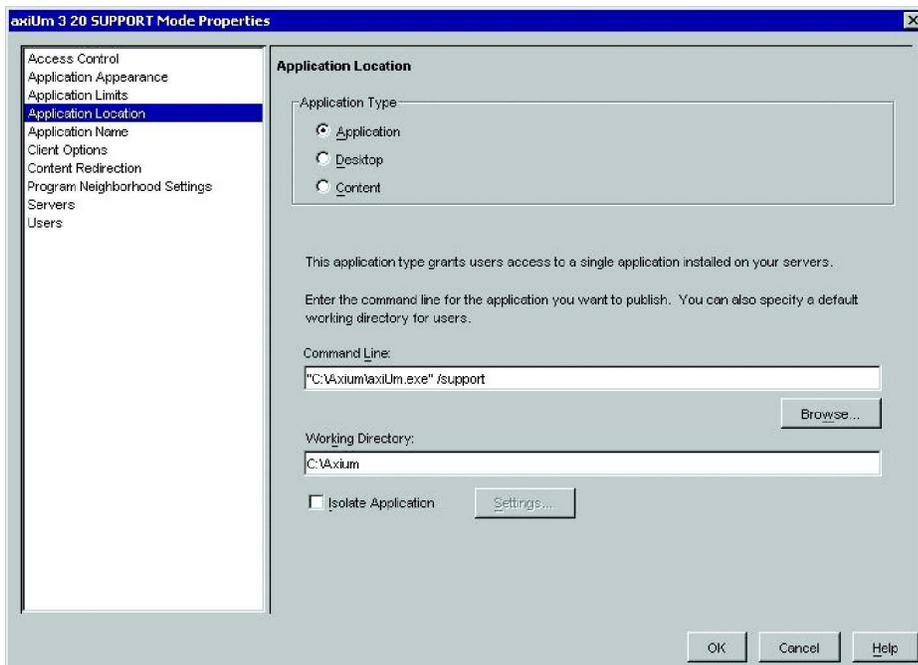
# Windows Printer Settings

If you have problems printing forms from axiUm make sure that the printer has spooling turned off. On the server computer open the printer properties dialog from Start – Settings – Printers and Faxes and select the Advanced tab for the printer being used. Select the option "Print directly to the printer".
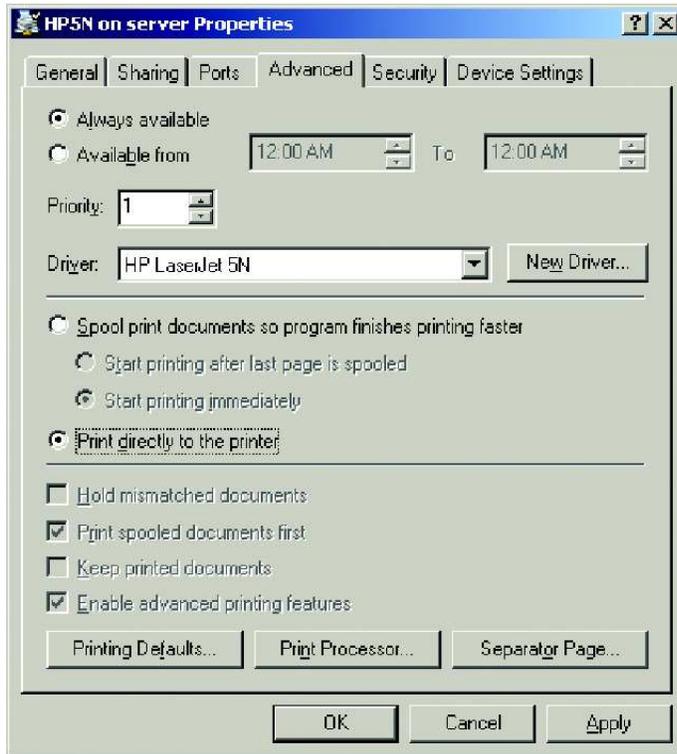
# APPENDIX G: EMAGO TECHNICAL REQUIREMENTS

## Server

Recommended system requirements:

- Pentium 4
- 1GB Ram
- 120GB+ Hard Disk (SATA or SCSI)
- Windows OS (2000 and higher)
- Gigabit network card

## Client

### Emago Hardware requirements

Emago will run on any computer that is able to run Windows 2000, XP, Vista and Windows 7.. The following information gives the minimum specification for smooth performance on a Windows XP system. (The actual specifications can be different based on the infrastructure of the imaging environment, for example, stand alone system, workstations and central server; GP office with only a few users or large institute with hundreds of users).

Minimum system requirements:

- Pentium 3 (800 MHz)
- 128 MB Ram
- 10 GB Hard Disk

Recommended system requirements:

- Pentium 4
- 256 MB Ram
- 20 GB Hard Disk

Many image manipulation procedures are computationally very intensive. System performance is therefore depending on the speed and internal memory of the computer system. A higher clock-

frequency and more internal memory to avoid disk swapping will increase system performance considerably.

The disk size required for digital imaging depends on the kind of image manipulations that are usually performed. Archiving large numbers of digital images requires much disk space. It is possible to combine the main hard disk with other disks that are specially designed for intermediate storage or archiving. The actual storage space that is required needs to be calculated based on the anticipated number of images that has to be archived.

## Monitor Recommendations for Emago

- SVGA or XVGA (1024 x 768 or better)
- High color
- 32 MB or higher
- 17" or higher.

It is recommended to use a monitor that is capable of showing at least 1024x768 pixels, in order to be able to display images side by side. Emago will by default display an image with the magnification that allows showing the whole image on the screen, although it is always possible to zoom in onto the image and use the full resolution.

Some monitors are capable of working in resolutions that are higher than 1024 x 768. Emago will run on these monitors without any problem. It is also possible to run Emago on a dual monitor system as supported by Windows.

The monitor size given above is the minimum size. A larger monitor is recommended to make small details at high resolution better visible.

## Peripheral Devices (Optional)

You may want to use other hardware to scan existing conventional radiographs, to print digital image or to do other image acquisition or export procedures.

This may include:

- System for image acquisition (e.g., Intra-oral sensor system, digital panoramic system, CCD-camera, scanner)
- System for digitizing conventional radiographs (e.g., flat bed scanner with transparency adapter, digital camera)
- CD/DVD writer
- Optional mass-storage device (e.g., optical disk, OM-disk, tape)

- Inkjet printer (preferably 600 DPI), dye sublimation printer or other hard copy device.

Images can be imported from any source: a dental intra-oral sensor system, a system for digital panoramic or cephalometric images, a CCD-video camera, slide scanner, or film scanner in combination. Emago understands most standard image formats by default; the definition of missing image formats can be defined and stored in the on-line image format library.

Images can be printed using a standard Windows compatible printer. The size and the quality of the images are depending on the characteristics of the printer. A laser printer is not recommended. An inkjet printer is usually a good option. The choice of the printing medium (glossy or photographic paper) is important for bright and high contrast images.

# APPENDIX H: IMPLEMENTATION CHECKLIST

> Prior to your "Go Live" date, please use this checklist to review and substantiate all items that apply to your implementation.

## Hardware

Servers:

- Oracle Production database is configured with all setup data and ready for Live usage.
- Oracle Database parameters are set to sufficiently support the operation of the full clinic. Specifically, ensure that the PROCESSES parameter is set to at least two times (2x) the number of axiUm workstations.
- Backup and Disaster recovery procedures are tested and in place.
- axiUm Auto-update folder/shared drive is setup and the PRACTICE."VersionSrc" field is set accordingly.
- For Windows platform servers (either DB or Citrix/Terminal Server), Physical Address Extension (PAE) along with /3GB flag  is setup to enable support of greater than 4 GB of physical memory.(In case of 32 bit server)
- For Citrix/Terminal Server, check axium.ini on every server to enable citrix=1.

Workstations:

- Standard users (Windows O/S) have access to all required functionality (including axiUm and Oracle files/folders).
- For Citrix/Terminal Server,check axium.ini on every server to enable citrix=1
- All workstations are installed and verified to properly connect to the Production database. It is recommended to make changes to Test/Train environments to differentiate them from Production (e.g., modify Practice Names, Splash Screen, Desktop bitmap and/or Exan Logo bitmap).
- Emblem/Seal image has been inserted into the Patient Picture area (edit ExanLogo.bmp).

Peripherals:

- Printers functioning and accessible from appropriate workstations.
- Signature pads tested and working (with axiUm).
- Authentication devices working (and tested with end-users).

- Scanner (flatbed or feeder-style) is setup and configured to scan documents into axiUm.
- For tracking of Paper Charts:
  - Barcode scanners have been tested to scan in/out charts,
  - Label template has been created to generate new paper chart labels,
  - Label printers have been tested to confirm available of paper chart labels

Other:

- Remote connection information to the database has been provided and tested by axiUm support. Typically, access is through VPN, IP Tunnel, Remote Desktop or other secure method.
- Access to other system (that is being replaced by axiUm) can be disabled to prevent data entry in the wrong place.

# End Users

- Have been trained in their areas of axiUm.
- Are aware of any changes in their role/responsibilities.
- Have had the opportunity to use axiUm in a test/train environment.
- Have access to materials (aka Cheat Sheets) to support common procedures.
- axiUm and Operating System accounts have been created for all users.
- Know their various usernames/passwords and when and where to use them.
- Authentication tokens (Swipe cards, smart cards, proximity device, etc) have been created and distributed.
- Are confident that their business processes are sufficiently supported in the system.

# axiUm

- Substantiate all setup using the Workflows; log in as end user and follow process to identify problems.
- Maintenance – codes completely configured (see Maintenance Checklist).
- PowerAdmin security levels created and tested.
- Desktop/WorkStation Settings configured including Station Options and Templates.
- Overnight processes reviewed, specified and scheduled to run at specified time.
- If using Active Directory for access to axiUm, then confirm that it is working correctly for all users (check User and Workstation setup).
- Perform final verification of all converted data using a combination of reports, account checks and scenario tests.
- Confirm that all essential custom reports have been created and are available within Info Manager, Personal Planner, etc.

- All generic user accounts (in axiUm) have been disabled or their passwords have been altered.
- All modules and features that will not be used have been disabled through PowerAdmin.

Paper forms tested, including:

- Patient Statements, including test of Expressbill, payment plan statement and setup of monthly batch report.
- Patient Receipt.
- Patient Estimate.
- Others.

Insurance Claims:

- All paper claim forms tested using Live data (from trial conversion).
- EDI claims submission setup complete and test batch prepared and sent.

Televox (for appointment confirmation):

- Create Info Manager report and send test batch to Televox.
- Review sample confirmation file sent back from Televox.

Digital Imaging:

- Verify all digital imaging applications are accessible from icons within axiUm (and that the bridge/interface is working).
- Verify images are available in the EHR Imaging tab.

# APPENDIX I: SERVER INSTALLATION GUIDE

## New axiUm Server Installation

### Pre-requisites

- Information regarding VPN, Remote Desktop or IP Tunnel to connect through firewall to gain access to the database server is required to permit the axiUm technical support team to assist with the installation.
- Password for the SYS oracle user - when the Oracle database is installed/created, you will need to set this password. This password will be required to create the axiUm schema.
- User Name/Password for the axiUm schema owner - The axiUm schema owner is the only user required for the axiUm database. We recommend using AXIUM as the username, but select a secure password for the user.
- Directory created/available on the database server or SAN for the axiUm datafiles. You will require between 20-100 GB of disk space available for the system.

### Oracle Software Installation & Database Creation

> These steps are to be carried out by a technical representative at the institution with assistance from the axiUm technical support team.

1. Install Oracle Database Server software

   Version 10g or higher (preferably with the latest patchset applied).

2. Create starter Oracle database and start instance

   Character set for the database CANNOT be a UNICODE type (e.g., UTF-8). An ASCII set like WE8MSWIN1252 or US7ASCII must be selected when creating the database.

   For suggested parameter settings see Suggested Parameters.

3. Create 13 Tablespaces (and Datafiles) using script file (tablespace.sql).

   For instructions on creating tablespaces and datafiles, see Tablespace.

Edit the directories in the script (or make changes as necessary if using Oracle managed files).

4.  Create axiUm schema owner Oracle user with UserName "xxxx", Password "yyyy" . (createuser.sql).

    For instructions on creating usernames, see Create User.

    Edit the username and/or password for the user in the script.

    Edit the temporary tablespace name in the script file if necessary.

    You must be logged in/connected as the Oracle SYS user to create the user.

## axiUm Schema Object Creation

> These steps are to be carried out by the axiUm technical support team based on the selection of either Option 1 or Option 2. To facilitate the implementation, we highly recommend Option 1 for all new installations.

**Option 1 Using "Best of" Starter Dataset**

Import the "Best of" starter dataset using the Oracle IMP utility:

imp xxxx/yyyy@new_db file=bestof.dmp fromuser=bestof touser=xxxx

**Option 2: Creating Empty Database with no Starter Data**

> Connect to db as xxxx/yyyy, so that all objects are created in the axiUm schema.

1.  Run Sequence creation script file (sequence.sql).
2.  Run Table/Index creation script files (blob.sql, dynamo.sql, grow.sql, high.sql, medium.sql, static.sql, dllortho.sql, dlldispense.sql, dlllabtrack.sql, dlltxplan.sql, dllperio.sql, dllinventory.sql).
3.  Run Trigger creation script file (trigger.sql). Script initiates individual trigger creation scripts.
4.  Run Package creation script file (package.sql). Script initiates individual package creation scripts.
5.  Run Function creation script file (function.sql). Script initiates function creation scripts.
6.  Run View creation script file (view.sql). Script initiates view creation scripts.
7.  Create Workstation setting definitions (create_inidef.sql).
8.  Load ADA procedure codes – use procedur.ctl and procedur.txt

# Copying an axiUm Database

> The following steps can be used to copy an existing database/schema. This may be used to infrequently refresh a Training environment with data from your production instance. It would also be used regularly (nightly) to create a Reporting database to be used for running large reports on (day old) Production data.

For these procedures, the following parameters have been used:

|  | Original Database | New Database |
| --- | --- | --- |
| Username | axium_old | axium_new |
| Password | Pass | PASS |
| Tnsnames alias | axold | axnew |

# Create New Database

> This procedure would be used to create a database on a different server or as a new database instance on the same server. If you are opting to create an additional axiUm schema within an existing database, then you may omit steps 1 and 2.

1. Create a new Oracle database. Use scripts or DB Configuration Assistant.
2. Create 13 Tablespaces (and Datafiles) using script file (tblspace.sql).

   For instructions see Tablespace.

   Edit the directories in the script (or make changes as necessary if using Oracle managed files).

3. Create axiUm schema owner Oracle user with UserName "axium_new", Password "PASS" 3. (createuser.sql).

   For instructions see Create User.

   Edit the username and/or password for the user in the script.

   Edit the temporary tablespace name in the script file if necessary.

You must be logged in/connected as the Oracle SYS user to create the user.

4. Export the axiUm user from original db:

   exp axium_old/pass@axold file=axold.dmp owner=axium_old

5. Import the original axiUm user schema and data into the new database:

   imp axium_new/pass@axnew file=axold.dmp fromuser=axium_old touser=axium_new ignore=y

# Re-create the New Database

> This procedure is to be used to copy the data to a database that was created previously.

1. Drop the axium_new user and all objects

   SQL> drop user axium_new cascade;

   You must be logged in/connected to the axnew database as the Oracle SYS user to drop the user.

2. Re-Create axiUm schema owner Oracle user with UserName "axium_new", Password "PASS" (createuser.sql).

   For instructions see Create User.

   Edit the username and/or password for the user in the script.

   Edit the temporary tablespace name in the script file if necessary.

   You must be logged in/connected as the Oracle SYS user to create the user.

3. Export the axiUm user from original db:

   exp axium_old/pass@axold file=axold.dmp owner=axium_old

4. Import the original axiUm user schema and data into the new database:

   imp axium_new/pass@axnew file=axold.dmp fromuser=axium_old touser=axium_new ignore=y

# APPENDIX J: TABLESPACE

```
-- TABLES TABLE SPACE (6000 MB)


--14 MB allocated

CREATE TABLESPACE TBLSPC_PBLOB

     DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\pblob1.ora'

     SIZE 100M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--24 MB allocated

CREATE TABLESPACE TBLSPC_PSTATIC

          DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\pstatic1.ora'

          SIZE 100M AUTOEXTEND ON NEXT 100M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--1420 MB allocated

CREATE TABLESPACE TBLSPC_PHIGH

          DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\phigh1.ora'

          SIZE 2000M AUTOEXTEND ON NEXT 1000M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--756 MB allocated

CREATE TABLESPACE TBLSPC_PGROW

          DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\pgrow1.ora'
```

```
                 SIZE 1600M AUTOEXTEND ON NEXT 1000M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--515 MB allocated

CREATE TABLESPACE TBLSPC_PMEDIUM

                 DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\pmedium1.ora'

                 SIZE 1000M AUTOEXTEND ON NEXT 500M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--20 MB allocated

CREATE TABLESPACE TBLSPC_PDYNAMO

                 DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\pdynamo1.ora'

                 SIZE 200M AUTOEXTEND ON NEXT 100M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


--262 MB allocated

CREATE TABLESPACE TBLSPC_PHUGE

                 DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\phuge1.ora'

                 SIZE 1000M AUTOEXTEND ON NEXT 1000M

     EXTENT MANAGEMENT LOCAL

     SEGMENT SPACE MANAGEMENT AUTO;


-- INDEXES TABLE SPACE (1500 MB)


--1 MB allocated

CREATE TABLESPACE TBLSPC_IBLOB
```

```
                    DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\iblob1.ora'

                    SIZE 20M AUTOEXTEND ON NEXT 100M

         EXTENT MANAGEMENT LOCAL

         SEGMENT SPACE MANAGEMENT AUTO;


--19 MB allocated

CREATE TABLESPACE TBLSPC_ISTATIC

                    DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\istatic1.ora'

                    SIZE 40M AUTOEXTEND ON NEXT 100M

         EXTENT MANAGEMENT LOCAL

         SEGMENT SPACE MANAGEMENT AUTO;


--230 MB allocated

CREATE TABLESPACE TBLSPC_IHIGH

                    DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\ihigh1.ora'

                    SIZE 600M AUTOEXTEND ON NEXT 500M

         EXTENT MANAGEMENT LOCAL

         SEGMENT SPACE MANAGEMENT AUTO;


--163 MB allocated

CREATE TABLESPACE TBLSPC_IGROW

                    DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\igrow1.ora'

                    SIZE 600M AUTOEXTEND ON NEXT 500M

         EXTENT MANAGEMENT LOCAL

         SEGMENT SPACE MANAGEMENT AUTO;


--45 MB allocated

CREATE TABLESPACE TBLSPC_IMEDIUM

                    DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\imedium1.ora'
```

```
              SIZE 200M AUTOEXTEND ON NEXT 500M

        EXTENT MANAGEMENT LOCAL

        SEGMENT SPACE MANAGEMENT AUTO;


--7 MB allocated

CREATE TABLESPACE TBLSPC_IDYNAMO

              DATAFILE 'D:\ORADATA\AXIUM\DATAFILE\idynamo1.ora'

              SIZE 40M AUTOEXTEND ON NEXT 100M

        EXTENT MANAGEMENT LOCAL

        SEGMENT SPACE MANAGEMENT AUTO;
```

# APPENDIX K: TECHNICAL REQUIREMENTS

## Introduction to axiUm Systems Architecture

axiUm was developed as a client-server application with the application residing on a Windows PC-based client communicating with an Oracle database server. However, through the use of MS Terminal Services and Citrix, axiUm has been successfully deployed in a three-tier architecture with all client processing done on the middle tier or application server.

There are benefits (costs, risks, maintenance, etc) associated with each architecture that must be deliberated by the institution before making the final decision. The optimum deployment architecture is usually comprised of a mixture of thick and thin clients, strategically deployed throughout the institution.

When installed in a standard client/server environment, the database (and all program data) resides on the Oracle server, while all the application program files are installed on Windows PC's.

There are no axiUm software components installed on the database server; this allows the flexibility of supporting any Oracle server platform.

When installed in a thin client environment, the database still resides solely on the Oracle server, but now all axiUm software components are installed on one or more application servers. The thin client systems are installed simply with the software to connect to the application server (e.g., Citrix client or MS Terminal Services).

axiUm has the ability to be installed as a standard client /server architecture or a three-tier server/ application server/thin client architecture.

## Hardware Component Overview

This list is a general overview of the types of components needed for axiUm to be implemented in it's entirety.

Specific hardware requirements for servers/workstations/application servers can be found on the following page.

You must have:

- Oracle Database server and storage array.

- Backup system for database server. In a chartless environment, the value of your electronic data is priceless. You need to minimize (or prevent altogether) the possibility of down-time and/or data loss in the case of a system failure.
- Workstations (PCs). Including keyboard and pointing device (mouse, light pen, touch screen).
- Networking Equipment. To connect PC's to server: Switches/Routers.
- Digital Imaging Hardware.
- Printers. Laser, Dot Matrix, Impact, Label are all supported, but not required.
- Bar code readers. Used for tracking of patient charts and instrument kits.
- Magnetic swipe card readers. Used for user logon/authentication and for approval/authorization functions. We recommend using the Magtek swipe readers. These can be ordered through www.magtek.com. You will need to determine the number of tracks on the current magnetic cards and order the appropriate card reader for the number of tracks. There will also be some set up in axiUm required.
- Magnetic swipe card writer. If cannot use track/data from existing ID card.
- Signature pads. Needed if acquiring patient/student/faculty signatures on-line. Alternatively, this could be achieved using light pens. Signature Capture Devices are needed at each terminal where a Consent, Treatment Plan, or Form may need to be signed. When set up in a Citrix environment you will need to have signature pads with a serial port connection (not USB), otherwise USB is good. We recommend using Topaz systems (available through www.computimeusa.com or www.topazsystems.com). The most cost effective model is the Sig Lite, Topaz's low-cost pressure-sensitive electronic signature pad. SigLite features high-quality capture techniques with a low-cost touch pad and stylus.
- Scanner. For scanning paper documents into the electronic patient record.
- Digital Camera. To capture patient images.

# Specific Hardware requirements

## Oracle Database Server Requirements

- Oracle Database 10g or higher. Contact Oracle for a list of supported hardware platforms.

Minimum specifications:

- Dual processor
- 8 GB RAM
- 250 GB or Higher
- Dual 100 MBps NIC's or single Gigabyte Network Interface Card (NIC)

Recommended specifications:

- Quad Processor
- 16 GB RAM
- 500 GB or Higher
- Dual 100 MBps NIC's or single Gigabyte Network Interface Card (NIC)

## Application Server

To support up to 40 thin clients:

- Dual processor
- 8 Gigabytes RAM
- 40 Gigabytes of disk space
- Dual 100 MBps NIC's or single Gigabyte NIC

## Thin Client System

The minimum requirements are dependent on the terminal services option employed, for example, Citrix or MS Remote Desktop.

## Standard PC Requirements for Client/Server Environment

Client Workstation:

- PC-platform (not Macintosh or UNIX-based)
- Minimum operating system of Windows 2000 Professional.
- Working Ethernet Network Interface Card (NIC),
- 512MB of RAM, and a min. of a 1GHz processor.
- Display adapter and monitor capable of supporting a minimum screen area of 1024 by 768 pixels.
- Sound card and microphone/headset if using voice activation (for periodontal charting, note dictation, etc).

Recommended specifications:

- Pentium 4 processor
- 1 GB Megabytes RAM
- Windows XP Professional operating system

# Software Requirements

- Oracle SQL Net client (v.10g  or higher recommended).

Other applications that are utilized within axiUm and are recommended for standard client workstations are:

- Microsoft Word
- Adobe Acrobat Reader
- Digital Imaging software
- Report writing application. Recommended for technical staff to create/enhance custom reports. Crystal Reports is considered the standard report writer by our user's group and efforts are made to share custom reports within the user's group.

# APPENDIX L: WORKSTATION INSTALLATION GUIDE

## Pre-requisites

Required:

- Oracle client software installation CD (you will require this to install any clients; it is not provided by Exan or included with the axiUm client installation).
- A supported Microsoft Windows Operating System (2000, XP, Vista, Windows 7).

Optional:

- Adobe Acrobat Reader
- Microsoft Word
- Digital Imaging Software
- Signature Pad Software
- Smart Card/Biometric Software

Connection information for all Oracle databases created:

> This information would have been set by the Server Installation procedure.

- Oracle (axiUm) User Name
- Oracle Password for (axiUm) Use
- Oracle Host (IP Address or Network Name of Server)
- Oracle Port (e.g., 1521 or 1526)
- Oracle SID
- TNSNAMES.ORA Alias

## Oracle Client Installation

> Oracle 11g was used in this example.

1. Run the setup.exe program in the Client folder of the Oracle CD. To start the installation, select the Next button in the Welcome dialog.
2. Change the PATH field to C:\oracle\ora11g or similar local drive directory and click on the Next button in the Specify File Locations dialog.
3. Select Custom installation and next in the Installation Type dialog.
4. In Available Product Components, select:
- Oracle Windows Interface 11.x.x.x (all workstations)
- Oracle Network Utilities 11.x.x.x (administrators only, optional)
- SQL*Plus 11.x.x.x (administrators only, optional)

then continue with the Next button.

5. Use the default setting for Oracle Services and click Next.
6. Review the Summary dialog and, if correct, select Install.
7. During installation the Progress dialog shows completion progress.
8. After the installation of the files a Configuration Assistants dialog is presented and then the Configuration Assistant Welcome dialog.
9. In the Configuration Assistant Welcome dialog select the Cancel button.
10. In the final End of Installation dialog, click Exit and end the installation.


# Oracle Client Configuration

The TNSNAMES.ORA file that is found in the ORACLE_HOME\Network\Admin\ folder creates an Oracle TNS definition that allows access to the axiUm database.

This first example defines PRODUCTION_DB as follows:

- HOST - axium.aschool.edu
- PORT – 1521
- SID - axium

```
PRODUCTION_DB =

(DESCRIPTION =

(ADDRESS_LIST =

(ADDRESS    =    (PROTOCOL    =    TCP)(HOST    =
axium.aschool.edu)(PORT=521)))

(CONNECT_DATA = (SID = axium))

)
```

If a Training database is in use, add its connection info as well.

In this example of the definition TRAIN_DB, the host is in the form of an IP Address and the SID is replaced with a SERVICE_NAME entry. Your DBA will be able to provide the required settings.

The SID identifies an instance used to find default configuration files on that host. A service name is a collection of instances and many instances can register as the same service. This is useful for load balancing with Oracle Real Application Clusters (RAC).

- HOST – 123.203.253.123
- PORT – 1526
- Service Name - axtrain

```
TRAIN_DB =

(DESCRIPTION =

(ADDRESS_LIST =

(ADDRESS  =  (PROTOCOL  =  TCP)(HOST  =  123.203.253.123)(PORT  =
1526)))

(CONNECT_DATA = (SERVICE_NAME = axtrain))

)
```

> To prevent errors in the application, there are specific privileges that are required to Oracle client files. These privileges are listed in Oracle Dependency and must be available to any Windows user that is operating axiUm.

# axiUm Installation

- Log in as a user with local administrator rights.
- Run SETUP.EXE from the axiUm installation CD or folder. If this is an administrator workstation, select PowerAdmin in the list to install it as well as axiUm.

The only other option allows you to specify the destination folder for the axiUm program files.

## Running axiUm for the First Time

You can run axiUm from the Desktop or from the Start Menu. When first run axiUm, you will be prompted for the user name, password, and name of the database definition. To prevent having to

enter this information every time, it is saved in an encrypted format in the axiUm.ini file in the axiUm program files folder.

## PowerAdmin

The PowerAdmin default password is poweradmpass. This can be changed in the PowerAdmin program. This password, unlike all others in axiUm, is workstation specific and is stored locally in an encrypted format. To prevent having to enter this information every time, it is saved in an encrypted format in the poweradmin.ini file in the axiUm program files folder.

## New Station Registration Key

To have control over what stations axiUm can be installed on, a "New Station Registration Key' may be specified. In this case, when axiUm is run for the very first time, the user is prompted to enter this registration key, and axiUm will not start until the key has been entered correctly. This will only occur the first time axiUm is run.

## Additional Features of the axiUm Setup

The axiUm installation file allows the installation of alternative, additional or customized files in the axiUm installation directory. This is useful when an axiUm setup requires unique bitmaps for example. To do this, create a sub-folder called extra below the directory where setup.exe resides and when setup.exe is run everything stored in that folder will be copied to the installation directory at the end of the setup.

> ✏️ To prevent errors in the application, there are specific privileges that are required to axiUm client files. All Windows users will require read/write access privileges to the GEOM*.DBA files that reside in the axiUm program files folder.

Additionally, users will need access to a temp folder on the workstation.

# axiUm Configuration

To configure the axiUm client to connect to a database, there are only 3 files to consider:

## TNSNAMES.ORA

(found in the ORACLE_HOME\Network\Admin\ directory on the workstation)

```
axumprod =

(DESCRIPTION =

(ADDRESS_LIST =

(ADDRESS = (PROTOCOL = TCP)(Host = myserver.school.edu)(Port =
1521)))

(CONNECT_DATA = (SID = AXUM))

)
```

## AXIUM.INI

[Database]

DbAlias=UserdefineddbName //suggest to use the same as TNSNames.Ora entry name

> ✏️ The following parameters (also in the [Database] section) can be adjusted to tune the Oracle
> client:

```
SliceSize=256

PerBlock=16

CacheBlocks=127

FetchLimit=200

FetchSize=4096
```

## POWERADM.INI

```
[database]

DBAlias=axium11

DBEngine=3

SliceSize=256
```

PerBlock=16

CacheBlocks=127

FetchLimit=200

FetchSize=4096

# APPENDIX M: ORACLE DEPENDENCY

## Oracle 9 Client

```
-- Need Read and Execute Privileges to the following files

$ORACLE_HOME\bin\OCI.DLL

$ORACLE_HOME\bin\OIP9.DLL

$ORACLE_HOME\bin\ORAANSI.DLL

$ORACLE_HOME\bin\ORACLIENT9.DLL

$ORACLE_HOME\bin\ORACLM32.DLL

$ORACLE_HOME\bin\ORACOMMON9.DLL

$ORACLE_HOME\bin\ORACORE9.DLL

$ORACLE_HOME\bin\ORAGENERIC9.DLL

$ORACLE_HOME\bin\ORALDAPCLNT9.DLL

$ORACLE_HOME\bin\ORAMTS.DLL

$ORACLE_HOME\bin\ORAN9.DLL

$ORACLE_HOME\bin\ORANCDS9.DLL

$ORACLE_HOME\bin\ORANCRYPT9.DLL

$ORACLE_HOME\bin\ORANHOST9.DLL

$ORACLE_HOME\bin\ORANL9.DLL

$ORACLE_HOME\bin\ORANLDAP9.DLL

$ORACLE_HOME\bin\ORANLS9.DLL

$ORACLE_HOME\bin\ORANMS.DLL

$ORACLE_HOME\bin\ORANMSP.DLL

$ORACLE_HOME\bin\ORANNTS9.DLL

$ORACLE_HOME\bin\ORANNZSBB9.DLL

$ORACLE_HOME\bin\ORANONAME9.DLL
```

```
$ORACLE_HOME\bin\ORANRO9.DLL

$ORACLE_HOME\bin\ORANTCP9.DLL

$ORACLE_HOME\bin\ORANTNS9.DLL

$ORACLE_HOME\bin\ORAPLS9.DLL

$ORACLE_HOME\bin\ORASLAX9.DLL

$ORACLE_HOME\bin\ORASNLS9.DLL

$ORACLE_HOME\bin\ORASQL9.DLL

$ORACLE_HOME\bin\ORATRACE9.DLL

$ORACLE_HOME\bin\ORAUNLS9.DLL

$ORACLE_HOME\bin\ORAVSN9.DLL

$ORACLE_HOME\bin\ORAWTC9.DLL

$ORACLE_HOME\bin\ORAXML9.DLL

$ORACLE_HOME\bin\ORAXSD9.DLL


-- Need Read Access to the following files

$ORACLE_HOME\bin\oip9.tlb

$ORACLE_HOME\bin\oracle.key

$ORACLE_HOME\bin\OraOLEDB.tlb


-- Need Read access to the following folders

$ORACLE_HOME\network\admin

$ORACLE_HOME\network\mesg

$ORACLE_HOME\ocommon

$ORACLE_HOME\oo4o\mesg

$ORACLE_HOME\oracore

$ORACLE_HOME\plsql\mesg

$ORACLE_HOME\rdbms\mesg
```

# Oracle 10 Client

```
-- Need Read and Execute Privileges to the following files

$ORACLE_HOME\bin\OCI.DLL

$ORACLE_HOME\bin\OIP10.DLL

$ORACLE_HOME\bin\ORAANSI10.DLL

$ORACLE_HOME\bin\ORACLIENT10.DLL

$ORACLE_HOME\bin\ORACOMMON10.DLL

$ORACLE_HOME\bin\ORACORE10.DLL

$ORACLE_HOME\bin\ORAGENERIC10.DLL

$ORACLE_HOME\bin\ORAHASGEN10.DLL

$ORACLE_HOME\bin\ORALDAPCLNT10.DLL

$ORACLE_HOME\bin\ORAN10.DLL

$ORACLE_HOME\bin\ORANCDS10.DLL

$ORACLE_HOME\bin\ORANCRYPT10.DLL

$ORACLE_HOME\bin\ORANHOST10.DLL

$ORACLE_HOME\bin\ORANL10.DLL

$ORACLE_HOME\bin\ORANLDAP10.DLL

$ORACLE_HOME\bin\ORANLS10.DLL

$ORACLE_HOME\bin\ORANNZSBB10.DLL

$ORACLE_HOME\bin\ORANRO10.DLL

$ORACLE_HOME\bin\ORANTCP10.DLL

$ORACLE_HOME\bin\ORANTNS10.DLL

$ORACLE_HOME\bin\ORAOCR10.DLL

$ORACLE_HOME\bin\ORAOCRB10.DLL

$ORACLE_HOME\bin\ORAPLP10.DLL

$ORACLE_HOME\bin\ORAPLS10.DLL

$ORACLE_HOME\bin\ORASLAX10.DLL
```

```
$ORACLE_HOME\bin\ORASNLS10.DLL

$ORACLE_HOME\bin\ORASQL10.DLL

$ORACLE_HOME\bin\ORAUNLS10.DLL

$ORACLE_HOME\bin\ORAUTS.DLL

$ORACLE_HOME\bin\ORAVSN10.DLL

$ORACLE_HOME\bin\ORAXML10.DLL


-- Need Read Access to the following files

$ORACLE_HOME\bin\oip10.tlb

$ORACLE_HOME\bin\oracle.key


-- Need Read access to the following folders

$ORACLE_HOME\network\admin

$ORACLE_HOME\network\mesg

$ORACLE_HOME\nls\data

$ORACLE_HOME\oo4o\mesg

$ORACLE_HOME\oracore

$ORACLE_HOME\plsql\mesg

$ORACLE_HOME\rdbms\mesg
```

## Oracle 11 Client

```
-- Need Read and Execute Privileges to the following files

$ORACLE_HOME\bin\OCI.DLL

$ORACLE_HOME\bin\ORACLIENT11.DLL

$ORACLE_HOME\bin\ORACLM32.DLL

$ORACLE_HOME\bin\ORACOMMON11.DLL

$ORACLE_HOME\bin\ORACORE11.DLL

$ORACLE_HOME\bin\ORAGENERIC11.DLL
```

```
$ORACLE_HOME\bin\ORALDAPCLNT11.DLL

$ORACLE_HOME\bin\ORAN11.DLL

$ORACLE_HOME\bin\ORANCDS11.DLL

$ORACLE_HOME\bin\ORANCRYPT11.DLL

$ORACLE_HOME\bin\ORANHOST11.DLL

$ORACLE_HOME\bin\ORANL11.DLL

$ORACLE_HOME\bin\ORANLDAP11.DLL

$ORACLE_HOME\bin\ORANLS11.DLL

$ORACLE_HOME\bin\ORANMS.DLL

$ORACLE_HOME\bin\ORANMSP.DLL

$ORACLE_HOME\bin\ORANNZSBB11.DLL

$ORACLE_HOME\bin\ORANONAME11.DLL

$ORACLE_HOME\bin\ORANRO11.DLL

$ORACLE_HOME\bin\ORANTCP11.DLL

$ORACLE_HOME\bin\ORANTNS11.DLL

$ORACLE_HOME\bin\ORAPLS11.DLL

$ORACLE_HOME\bin\ORASLAX11.DLL

$ORACLE_HOME\bin\ORASNLS11.DLL

$ORACLE_HOME\bin\ORASQL11.DLL

$ORACLE_HOME\bin\ORAUNLS11.DLL

$ORACLE_HOME\bin\ORAVSN11.DLL

$ORACLE_HOME\bin\ORAXML11.DLL

$ORACLE_HOME\bin\ORAHASGEN11.DLL

$ORACLE_HOME\bin\ORAOCRB11.DLL

$ORACLE_HOME\bin\ORAPLP11.DLL

$ORACLE_HOME\bin\ORAPLS11.DLL

$ORACLE_HOME\bin\ORAUTS.DLL
```

```
-- Need Read Access to the following files

$ORACLE_HOME\bin\oracle.key


-- Need Read access to the following folders

$ORACLE_HOME\network\admin

$ORACLE_HOME\network\mesg

$ORACLE_HOME\oracore

$ORACLE_HOME\plsql\mesg

$ORACLE_HOME\rdbms\mesg

$ORACLE_HOME\nls\data
```

Note: Read and execute access is required for executables like DLLs while only read access would be needed for  most text files like message files and sql scripts. Possible batch or cmd files would need execute permissions again.