Department of Veterans Affairs

Automated Surgical Risk Calculator (ASRC)

System Administrator Guide



November 2014

Version 1.1

Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 11/24/2014 | 1.0 | Document Shell | D. Tombs |
| 11/25/2014 | 1.1 | Technical Edit | S. Vetzel |

**Note:** The revision history cycle begins once changes or enhancements are requested after the System Administrator Guide has been baselined.

Table of Contents

[1. System Business and Operational Description 1](#_Toc404670303)

[1.1. Logical System Description 1](#_Toc404670304)

[1.2. Software Description 1](#_Toc404670305)

[1.2.1. Background Processes 1](#_Toc404670306)

[1.2.2. Job Schedules 1](#_Toc404670307)

[1.2.3. Dependent Systems 1](#_Toc404670308)

[2. Routine Operations 2](#_Toc404670309)

[2.1. Administrative Procedures 2](#_Toc404670310)

[2.1.1. System Start-up 2](#_Toc404670311)

[2.1.2. System Shut-down 2](#_Toc404670312)

[2.1.3. Back-up & Restore 2](#_Toc404670313)

[2.1.3.1. Back-Up Procedures 2](#_Toc404670314)

[2.1.3.2. Restore Procedures 3](#_Toc404670315)

[2.1.3.3. Back-Up Testing 3](#_Toc404670316)

[2.2. Security / Identity Management 3](#_Toc404670317)

[2.2.1. Identity Management 4](#_Toc404670318)

[2.2.2. Access control 4](#_Toc404670319)

[2.3. System Monitoring, Reporting & Tools 4](#_Toc404670320)

[2.3.1. Availability Monitoring 4](#_Toc404670321)

[2.3.2. Performance/Capacity Monitoring 4](#_Toc404670322)

[2.3.3. Critical Metrics 4](#_Toc404670323)

[2.4. Routine Updates, Extracts and Purges 5](#_Toc404670324)

[2.5. Scheduled Maintenance 5](#_Toc404670325)

[2.6. Capacity Planning 5](#_Toc404670326)

[2.6.1. Initial Capacity Plan 5](#_Toc404670327)

[3. Exception Handling 6](#_Toc404670328)

[3.1. Routine Errors 6](#_Toc404670329)

[3.1.1. Security Errors 6](#_Toc404670330)

[3.1.2. Time-outs 6](#_Toc404670331)

[3.1.3. Concurrency 6](#_Toc404670332)

[3.2. Significant Errors 7](#_Toc404670333)

[3.2.1. Application Error Logs 7](#_Toc404670334)

[3.2.2. Infrastructure Errors 7](#_Toc404670335)

[3.2.2.1. Database 7](#_Toc404670336)

[3.2.2.2. Application Server 7](#_Toc404670337)

[3.3. Dependent System(s) 7](#_Toc404670338)

[3.4. Troubleshooting 8](#_Toc404670339)

[3.5. System Recovery 8](#_Toc404670340)

[3.5.1. Restart after Non-Scheduled System Interruption 8](#_Toc404670341)

[3.5.2. Restart after Database Restore 8](#_Toc404670342)

[3.5.3. Rollback Procedures 8](#_Toc404670343)

# System Business and Operational Description

This section provides the reader with a high-level description of the system. It should describe what the system does in the context of the Department of Veterans Affairs (VA). For example, describe the system in the context of patient care or how it fits in the administrative process. Details of the system should be placed in one of the subsections. This section should also define who the systems business owners are and provide point of contact information for the organizational elements that own relevant resources.

The subsections are guides and you may add a new section if appropriate.

## Logical System Description

In this section, each of the logical components of the system should be listed and described. Components describe an aggregation of function and not physical devices or locations. In addition to the descriptions of the components, a diagram of the logical system should be included. The diagram should show all the components and the relationships among them, including references to system flow and connectivity. If there are any special or unusual interactions among components then that relationship should be expanded upon.

## Software Description

In this section the software components of the system are enumerated and described. The software components of the system include: the main application and language, operating system, database, Web and application servers, extract transform & load (ETL), messaging, scripts, utilities, reporting tools, and any other software program that is executed as part of the system. The descriptions should include vendor, version, license information, and configuration (especially non-default parameters). The size and configuration of file systems and storage arrays should also be included. Much of this information should be in a CMDB and if so, then how to access the CMDB should be included here as it is above.

### Background Processes

This section should list of all application background processes running on all the components of the system.

### Job Schedules

If batch jobs or any other process is executed at a periodic interval, then intervals should be identified and jobs should be listed in this section.

### Dependent Systems

In this section other systems, either internal or external to the VA, that are used by the application are listed and described. The name, location, function and interface method should be included as part of the description. If the dependent system is required for the normal operation of the application then that should be noted. Where appropriate, links to the CBDB should be included.

# Routine Operations

This section describes at a high-level what is required of an operator/administrator or other non-business user to maintain the system at an operational and accessible state. These functions and tasks should be identified by function or role (such as system administrator, super user, end user, etc.) Descriptions should include additional description details if necessary (such as end user) based on likely level of knowledge. Details should be left to the subsections.

## Administrative Procedures

### System Start-up

This section defines how the system is started and brought to an operational state. The procedures should start with the boot of the server or device. If the application or a component is run as a service or background process, it needs to be listed as such, along with a method to verify that it is running. In cases where there are multiple servers and components involved, it is important to describe the correct order in which they start. In this case, a flow chart or an activity diagram of the procedure is the best method of representation.

### System Shut-down

This section defines how the system is shut down and brought to a nonoperational state. The procedures should stop all processes and components. The end state of this procedure should be a state where the start-up procedure can be applied. In cases where there are multiple servers and components involved it is important to describe the correct order in which they are stopped. In this case, a flow chart or an activity diagram of the procedure is the best method of representation.

### Back-up & Restore

In this section, a high-level description of the systems back-up and restore strategy is elaborated. A block diagram should be included that includes all the components that require back-up and the devices or infrastructure that perform the actual back-up and restore.

#### Back-Up Procedures

This section describes the schedule and procedures for performing routine back-up. It is likely that the data center will have standard procedures for the back-up of storage. This section should not redefine or replicate the data centers standards. Rather, what is described here is how to access the standard procedures, and more importantly, how those procedures apply to this particular system. Database and file system names should be listed.

The back-up schedule should be listed along with any variation to the level of the back-up (e.g., full vs. incremental, database vs. full system). After the schedule is presented, follow with the procedures necessary to perform all the different levels described in the schedule. The procedures should list in detail all the steps necessary to perform the back-up, including an estimate of the length of time required.

#### Restore Procedures

This section describes how to restore the system from a back-up. It is likely that the datacenter will have standard procedures for the restoration of storage. This section should not redefine or replicate the data centers standards. Rather, what is described here is how to access the standard procedures and more importantly, how those procedures apply to this particular system. It will contain procedures for all the types of back-up described above. Since restoring a system is not typically a normal operating procedure, this section should have as much detail as possible. The starting state of the system may not be fully known, so it is very important to outline what the starting state is and all assumptions made. The addition of diagrams and flow charts will make this section more accessible to the user and greatly enhance the ability to restore the system in a timely manner.

#### Back-Up Testing

Once back-ups are made, it is important to test periodically to verify that they are accurate and can be used to restore the system. This section describes the procedure to test each of the back-up types described in the back-up section. Like back-ups, testing should be done on a scheduled periodic basis and the testing schedule should be described first. After the schedule is established, test procedures for each type of back-up should be detailed. The detail should include the hardware configuration of the test system, since the tests are not typically done on the production system. Test cases should also be established to ensure the accuracy of the restore. It is not sufficient to simply restore the back-up; basic operational tests must be performed along with specific data quality tests. The system test plan is a good resource when completing this section along with the business stakeholders who can provide important information.

## Security / Identity Management

This section provides a high-level description of the systems security and user management. First, describe the security architecture of the system. This includes the authentication and authorization mechanisms of the system. If there is a separate security plan and/or security administration documents, then links should be included here. A block diagram of the system as it relates to security should be included. It should show all layers of security at the system and network levels. If encryption is used in the system, it should be shown in the diagram and detailed in this section.

### Identity Management

This section defines the procedures for managing users. Describe the procedures for adding new users, giving and modifying rights, and deactivating users. Included in the procedure should be the administrative process for granting access rights and any authorization levels, if more than one exists. Also, the section should describe what level of administrator has the authority for user management.

### Access control

This section describes the systems access control functionality. It is primarily intended to cover security procedures and configurations not covered in the previous section. This may include any password aging and/or strictness controls, user/security group management, key management, and temporary rights.

## System Monitoring, Reporting & Tools

This section describes the high-level approach to system monitoring. It should define what tools are available for use and who is responsible for implementing and maintaining those tools. A diagram of the system and its monitoring tools would be useful in this section.

### Availability Monitoring

This section describes how to determine if the system is fully operational and working correctly. It should be written as a set of procedures on how to determine the overall operational state and the state of the individual components.

### Performance/Capacity Monitoring

This section describes the procedures needed to determine overall system performance and the performance of the individual components. If performance thresholds have been determined for the system, then they should also be included. Additionally, procedures for monitoring the capacity of data stores or network links should be included.

### Critical Metrics

This section provides details on exactly what metrics are critical to validating the normal operation of the system and also any indirect metrics that indicate a problem in their system or related systems. What should be included is a detailed description of the metrics for a particular application and also the upstream and downstream indications of application issues.

## Routine Updates, Extracts and Purges

This section defines the procedures for typical maintenance activities such as updates, on-request or periodic data extracts, database reorganizations, and purges of data. Clearly explain if purging and archiving are permitted for the system, along with the prerequisites that would trigger the event. These procedures need to be as specific as possible, as errors in the process may cause the loss of data. The procedures should also define who is responsible for performing the tasks. Typically, only a database analyst (DBA) will perform operations on the database, whereas an operator or administrator will do software updates or an operation on a non-database file.

## Scheduled Maintenance

This section defines the maintenance schedule for the system. The schedule should define time increments (e.g., yearly, quarterly, monthly) and what should be done at that interval. There should be full procedures for each of the intervals and a time estimate for the duration of the system outage.

## Capacity Planning

This section describes the process and procedures for performing capacity planning reviews. Typically capacity reviews are done once or twice a year and use data collected via system monitoring utilities. In this section, a schedule for capacity planning reviews should be defined, who should perform them, and how the results of the review are presented. The section should also define who is responsible for making adjustments in the system’s capacity.

### Initial Capacity Plan

This section should contain an initial capacity plan that forecasts for the first 3-month period and a 12-month period of production.

# Exception Handling

This section gives a high-level overview of how system problems are handled. It should describe the general expectations of how administrators and other operations personnel should respond to and handle system problems. Try to define a boundary between the type of issues that are appropriate for operators and administrators to resolve and issues that need escalation up the support structure. Reference data center standards where appropriate.

The subsections below are intended to give operators and administrators the information they need to know to detect and resolve system and application problems. The subsections are typical of any system and should be considered the minimum set.

## Routine Errors

Like most systems, <System Name> may generate a small set of errors that may be considered routine in the sense that they have minimal impact on the user and do not compromise the operational state of the system. Most of the errors are transient in nature and only require the user to retry an operation. The following subsections describe these errors, their causes, and what, if any, response an operator needs to take.

While the occasional occurrence of these errors may be routine, getting a large number of an individual error over a short period of time is an indication of a more serious problem. In that case the error needs to be treated as an exceptional condition.

In most systems there is a small subset of errors that routinely occur. These errors are generally minor and in most instances can be ignored. The three following subsections are three general categories of errors that typically generate these kinds of errors. Your system may have additional categories, so add sections as needed.

### Security Errors

This section lists all security type errors that a user or operator may encounter. Each individual error should be listed, with a description of what it is, when it may occur, and what is the appropriate response to the error.

### Time-outs

This section lists all time-out type errors that a user or operator may encounter. Each individual error should be listed, with a description of what it is, when it may occur, and what is the appropriate response to the error.

### Concurrency

This section lists all concurrency type errors that a user or operator may encounter. Each individual error should be listed, with a description of what it is, when it may occur, and what is the appropriate response to the error.

## Significant Errors

Significant errors can be defined as errors or conditions that affect the system stability, availability, performance, or otherwise make the system unavailable to its user base. The following subsections contain information to aid administrators, operators, and other support personnel in the resolution of errors, conditions, or other issues.

### Application Error Logs

This section describes the applications error logging functionality, lists the locations of where they are stored and what, if any, special tools are needed to view the log entries. For each log, the maximum size, growth rate, rotation and retention policy should be described. Also, if the system sends error or alarm messages to external systems, that should also be elaborated.

### Infrastructure Errors

VA IT systems rely on various infrastructure components. These components will have been defined in the Logical and Physical Descriptions section of this document. Most, if not all, of these infrastructure components generate their own set of errors. Each component has its own subsection and describes how errors are reported. The subsections are a typical list of components and are meant to be modified for each individual system.

The subsections are not meant to replicate existing documentation on the infrastructure component. If documentation is available online then a link to the documentation is appropriate. Each subsection should contain implementation- specific details such as database names, server names, paths to log files, etc.

#### Database

This section describes the system/application specific implementation of the database configuration as it relates to errors, error reporting, and other pertinent information on causes and remedy of database errors. If links to component documentation is available then it should be provided here.

#### Application Server

This section describes the system- or application-specific implementation of the application server configuration as it relates to errors, error reporting, and other pertinent information on causes and remedy of application server errors. If links to component documentation is available, then it should be provided here.

## Dependent System(s)

Many systems are dependent on other VA systems for normal operation. This section describes the errors, and error reporting as it relates to these systems, and what remedies are available to administrators for the resolution of these errors.

## Troubleshooting

This section should contain any helpful information on troubleshooting the system that has been learned as part of the development and testing processes, or from the operation of similar systems.

## System Recovery

The following subsections define the process and procedures necessary to restore the system to a fully operational state after a service interruption. Each of the subsections starts at a specific system state and ends up with a fully operational system.

The subsections defined below are typical, but not comprehensive. It may be necessary to add additional subsections to fully cover the range of failure and restart possibilities. The sections generally define how to recover from the crash of XYZ, by bringing the system to a known state and then restarting components of the system until it is fully operational. Where possible, use references to the Start-up and Shut-down procedures in Section 2.

### Restart after Non-Scheduled System Interruption

This section is intended to describe the restart of the system after the crash of the main application. If the system is not too complicated, it is possible to cover the failure of other components as alternate flows to the main processes. However, for a complex system it will be necessary to cover each component individually as a separate section.

### Restart after Database Restore

This section describes how to restart the system after restoring from a database backup.

### Rollback Procedures

Describe the rollback procedures that pertain to this system. A rollback is an operation that returns the database to some previous state. Rollbacks are important for database integrity, because they mean that the database can be restored to a clean copy, even after erroneous operations are performed. They are crucial for recovering from database server crashes. By rolling back any transaction which was active at the time of the crash, the database is restored to a consistent state.

The rollback feature is usually implemented with a transaction log.