Argent Job Scheduler and Argent Queue Engine Directory Structure

ARGENT
ENCYCLOPEDIA

Contents

Introduction

This document describes the directory structure of the Argent Job Scheduler and the Argent Queue Engine.

Argent Job Scheduler Home Directory

The default value for the root/home directory of the installed program product is typically C:\ARGENT\SchedulingEngine. Customers can change the directory name during product installation. Throughout the remainder of this document, the term “home directory” will be used to refer to the name of the installed product root directory.

Sub-Directories

The following directories are installed under the home directory:

_KEY

This directory contains the Argent Job Scheduler license file. There is typically only one file in this directory. The license file name is made up of the product name and the issue date of the product license. An example is:

ARGENT_JOB_SCHEDULER_04_AUG_2008_14_19_27.LIC

\DBF\History

This directory contains archived job information.

When the archive format is configured as “Excel comma delimited”, the archived job information is stored in files with a .CSV extension. When “Argent Job Scheduler History File” format is used, the archived job information is stored in files with a .ARC extension. When “ODBC compliant database” format is used, the archived job information is stored in files with a .SQL extension. These files are <u>not</u> SQL queries. The use of the .SQL file extension can be confusing in that regard.

\DBF\History\JobLog

This directory contains job logs for completed jobs. During rollover and archival processing, the job logs for jobs that are eligible to be purged are stored in a sub directory for each date. The job logs from jobs purged on 2 Aug, 2008 are stored in \DBF\History\JobLog\20080802, for example.

Job history information is stored for the length of time configured on the J9P settings display.

\JobDiagram

This directory contains customer-created job diagram files. Files in this directory are named using the name of the job and a .DGR file extension. An example is:

TEST_JOB.DGR

The files in this directory are perpetual and are not purged.

\Logs

This is used to store the product service log files. The service log file is, effectively, a running commentary on the actions a given product takes on behalf of the customer. A service log file can have varying degrees of detail. Four levels of detail are currently supported:

1 – LOG_NONE
This level of detail avoids writing text, if possible. Error messages and other important information are written to the log.

2 – LOG_NORMAL
This level of detail provides some status information, such as the routine starting and stopping of services. This level of detail is adequate in most production environments.

3 – LOG_DETAILED
This level of detail provides some feedback information about completed API calls, completed function calls within the product, and enhanced information about progress of multi-step processing.

4 – LOG_HIGHLY_DETAILED
This level of detail provides the greatest level of information about the product as the code executes. This level of detail should only be invoked on the recommendation of experienced and trained Argent technical professionals.

Changing the level of service file logging level requires changing a Windows registry entry for the product.

For many Argent products, the level of service log file logging can be changed directly from the user interface.In some cases, direct access to the service log file detail value is not supported directly using the user interface.

For the Argent Job Scheduler, the value of the service log file level of detail is controlled by the

HKLM\Software\Argent\SchedulingEngine\DEBUG_LEVEL

registry key. After a typical product installation, this value is set to 2 (LOG_NORMAL). On the advice of experienced and trained Argent technical support professionals, customers may wish to set this value to a higher level.

The service log level of detail can be set to a value between 1 (LOG_NONE) and 4 (LOG_HIGHLY_DETAILED). After making a change, the Argent Queue Engine service should be re-started to be sure the changed value is read in to memory and used for all components of the product.

The current service log file within the \Logs sub-directory will have the name **svc_log.txt**. Each Argent product supports the specification of a maximum service log file size. When the maximum log file size has been reached, the service log file will be renamed, and a new service log file will be created. Typically, the value for the maximum service log file size defaults to 5Mb (5,120Kb).

When the maximum size of a given service log file has been reached, the Argent product will rename the current log file to a name that includes that date and time the service log file reached its maximum size.

As mentioned above, the current service log file for the Argent Job Scheduler is named **svc_log.txt**. When the service log file maximum size is reached, the current service log file is renamed and further service log file messages are stored in **svc_log.txt**.

The Argent Job Scheduler user interface program log file is named **gui_log.txt**. When the service log file maximum size is reached, the current user interface program log file is renamed and further log file messages are stored in **gui_log.txt**.

All Argent products use a standardized approach to writing service log file messages and managing product service log files. This approach calls for opening and closing the service log file for each service log file message that needs to be written.

The result of this approach is that all Argent product service log files can be deleted at any time. The very next service log file message request invoked by the Argent product code will trigger a new file to be created within the \Logs sub-directory, if needed.

\MAPS

This directory contains customer-created Argent SuperMap graphic files.

The files in this directory are perpetual and are not purged.

Argent Queue Engine Home Directory

The default value for the root/home directory of the installed program product is typically C:\ARGENT\QueueEngine. Customers can change the directory name during product installation. Throughout the remainder of this document, the term “home directory” will be used to refer to the name of the installed product root directory.

Sub-Directories

The following directories are installed under the home directory:

\BenchMarks

This directory contains command files that can be used to bench-mark the performance and throughput of the Argent Queue Engine. The command files are organized within sub-directories focused on single- and multiple process environments. The command files are not needed in production environments and the \BenchMarks sub-directory can be safely deleted after product installation.

Customers that want to run bench-mark tests to verify the capabilities of the Argent Queue Engine should retain this directory.

\Dashboard

This directory contains sample files that can be used by developers at customer sites. This directory contains files that customer developers can use as examples of how the Argent Queue Engine information can be used in dashboarding tools.

\JobArchive

This directory contains job archive information. Files in this directory have names similar to “JOBS_4_Aug_2008.ARC”. Each .ARC file in the directory contains job slot information from JobsSlot.bin for each obsolete job detected during routine processing within XS_Main.exe. If the JobsSlot.bin file contains slots for 150 jobs and 30 jobs are eligible for archiving, a single JOBS_d_Mon_CCYY.ARC file is created that contains the job slot information for all of the 30 eligible jobs.

The \JobArchive sub-directory is perpetual and is not purged by the Argent Queue Engine.

\JobDef

This directory is used as a work area during job execution.

The files stored in this directory include transient files with the following general form: <queue_name>.nnn, where nnn is an ascending sequence number. These files are created by the QSHOW command and also by a worker thread controlled by the Job Scheduler user interface. The files are work files used to temporarily store information returned from the XSGetJobsInQueueByFile() API.

When the QSHOW HIGH_PRIORITY /JOBS command is executed, for example, the list of jobs is retrieved using the XSGetJobsInQueueByFile() API and stored in a file named HIGH_PRIORITY.001. The QSHOW command reads the data in the file and display the results. The file is then deleted using the Windows DeleteFile() API.

A worker thread within the Job Scheduler user interface also uses the AQEGetJobsInQueueByFile() API within AQE_API.dll to retrieve job information. The files have the same names as for QSHOW. The files are deleted when the AQEGetJobsInQueueByFile() API completes.

The type of files stored in this directory include .JS1 and .JS2. Both .JS1 and .JS2 files will have the assigned job number formatted into the full name. The full name of a .JS1 file will be J_*nnnnnn*.JS1, where *nnnnnn* is the assigned job number.

The .JS1 files are used as a signaling mechanism between XS_Main.exe and XS_Load.exe. They are transient and are deleted after an operator request has been processed. An operator request to abort a job, for example, will result in a .JS1 file being created. After the abort request has been processed, the .JS1 file is deleted.

Any I/O failure during the creation of a .JS1 file will result in a service log file message similar to the following:

“Failed to write temporary job definition file for job JOB_SAMPLE (J000156)”

The C:\ARGENT\QueueEngine\JobsSlot.bin file is the master list of jobs to be processed. A .JS1 file contains the job slot information for a single job.

.JS2 files are similar to .JS1 files: .JS2 files are used as a signaling mechanism between XS_Load.exe and XS_Main.exe.

A .JS2 file is created after an operator request has been processed and the job status information has been updated.

Summary:
The C:\ARGENT\QueueEngine\JobsSlot.bin is master list of jobs to be processed. A J_*nnnnnn*.JS1 file is an image of single job slot and represents a request. A J_*nnnnnn*.JS2 file is also an image of a single job slot and represents the result of a request.

\JobLog

This directory is used to store job log files. A job log is a text file containing the standard output from each individual job. In addition, each job log file contains information about how the associated job was started and ended.

The name of each job log file typically contains the job name as well as the job number. An example of a job log filename within the \JobLog sub-directory is:

JOB_PROD_PAYROLL_01_J000156.TXT

The contents of the \JobLog directory are purged by AJSPurge.exe during rollover and archival processing based on the job retention interval.

It is important to note that although the Argent Queue Engine supports storing job log files in a directory other than the \JobLog sub-directory, the only directory purged by AJSPurge.exe is \JobLog.

Given the typical product home directory of C:\ARGENT\QueueEngine, the name of the job log directory will be C:\ARGENT\QueueEngine\JobLog. If a customer configures the Argent Queue Engine so that job log files are stored in D:\JobLog_Files, for example, that directory is <u>not</u> purged during archival and rollover processing.

\Logs

This is used to store the product service log files. The service log file is, effectively, a running commentary on the actions a given product takes on behalf of the customer. A service log file can have varying degrees of detail. Four levels of detail are currently supported:

1 – LOG_NONE
This level of detail avoids writing text, if possible. Error messages and other important information are written to the log.

2 – LOG_NORMAL
This level of detail provides some status information, such as the routine starting and stopping of services. This level of detail is adequate in most production environments.

3 – LOG_DETAILED
This level of detail provides some feedback information about completed API calls, completed function calls within the product, and enhanced information about progress of multi-step processing.

4 – LOG_HIGHLY_DETAILED
This level of detail provides the greatest level of information about the product as the code executes. This level of detail should only be invoked on the recommendation of experienced and trained Argent technical professionals.

Changing the level of service file logging level requires changing a Windows registry entry for the product.

For many Argent products, the level of service log file logging can be changed directly from the user interface. In some cases, direct access to the service log file detail value is not supported directly using the user interface.

For the Argent Queue Engine, the value of the service log file level of detail is controlled by the

HKLM\Software\Argent\ArgentQueueEngine\DebugLevel

registry key. After a typical product installation, this value is set to 2 (LOG_NORMAL). On the advice of experienced and trained Argent technical support professionals, customers may wish to set this value to a higher level.

The service log level of detail can be set to a value between 1 (LOG_NONE) and 4 (LOG_HIGHLY_DETAILED). After making a change, the Argent Queue Engine service should be re-started to be sure the changed value is read in to memory and used for all components of the product.

The current service log file within the \Logs sub-directory will have the name **log.txt**. Each Argent product supports the specification of a maximum service log file size. When the maximum log file size has been reached, the service log file will be renamed, and a new service log file will be created. Typically, the value for the maximum service log file size defaults to 5Mb (5,120Kb).

When the maximum size of a given service log file has been reached, the Argent product will rename the current log file to a name that includes that date and time the service log file reached its maximum size.

As mentioned above, the current service log file for the Argent Queue Engine is named **log.txt**. When the service log file maximum size is reached, the current service log file is renamed to **log_dd_Mon_CCYY.TXT** and further service log file messages are stored in **log.txt**.

All Argent products use a standardized approach to writing service log file messages and managing product service log files. This approach calls for opening and closing the service log file for each service log file message that needs to be written.

The result of this approach is that all Argent product service log files can be deleted at any time. The very next service log file message request invoked by the Argent product code will trigger a new **log.txt** file to be created within the \Logs sub-directory, if needed.

\Simulate

This sub-directory contains one relevant file: **simulation.cmd**. This command file is very powerful and **changes the system clock on the system where it is invoked**. The Argent Queue Engine supports the capability to test and anticipate the possible impact on a production environment by date/time changes, specifically, changes that occur when the definition of Daylight Savings Time (or the local, cultural equivalent) takes place.

This is not a production command file and should only be invoked on carefully prepared test-bed systems where changes to the system clock can be managed and controlled.

Production systems where the impact of a date/time change has been adequately tested do not require this support.

On the advice of experienced and trained Argent technical professionals, this sub-directory can be safely deleted.

\Temp

This sub-directory is a work area used by Argent program products. It is created during product installation and contains transient and temporary files.

This file is not used during execution of production customer jobs.

Argent does not recommend deletion of this sub-directory, although files within this directory can be safely deleted.

\Word_Files

This sub-directory contains word processing files that are helpful to customers. The word processing files constitute relevant information for IT professionals and interested parties at customer sites that may need extended operational or technical information about Argent products.

This sub-directory is provided as a benefit to customers' IT professionals and can be safely deleted at any time.

Configurable\Standalone Files

Some file processing performed by the Argent Job Scheduler is configurable and is not tied to a specific directory.

Configuration Information Files

Both the Argent Job Scheduler and the Argent Queue Engine maintain configuration information in three files:

- hosts.dat
- timeouts.dat
- users.dat

The hosts.dat and timeouts.dat files contain information about Queue Engine server connections. The users.dat file contains mapping information used to translate a Job Scheduler user account to a different account on a given Queue Engine server system.

The Job Scheduler GUI Remote/UNIX/iSeries/SAP Server Management dialog (Screen ID J87A and J87B) is used to maintain these files. Customers should only manipulate these files on the advice of trained and experienced Argent technical and support professionals.

Exit Code Files

Typically, a customer job returns an exit code directly to the operating system. In C++, for example, the **return** statement is used to do this. A customer job that calls a C++ **.exe** file will use the **return** statement to tell Job Scheduler its result. The result is an exit code that can be used to configure exit code processing on the J20I display of a job definition.

In some cases, the customer executable that is invoked runs a script. Visual Basic scripts, for example can be invoked using VBScript.exe. Even though the script does not complete successfully, VBScript.exe will complete successfully. To provide a way for the Argent Queue Engine and thus Argent Job Scheduler to retrieve the result of the script, a customer-developed application can format an exit code file that contains the result of the script.

When a job definition is configured for exit code file processing using the J20E display, the Argent Queue Engine server looks in the installed product home directory for a file with the following general name:

<job_name>_nnnnnn.CC

Where *nnnnnn* is the job number assigned by the Queue Engine.

An example would be

COPY_DAILY_FILES_000156.CC

After the job exit code is retrieved from the exit code file, the file is deleted.

License Key Files

A license key received from Argent will have the product name and a date expression in the name, followed by a .LIC extension.

An example of a product license file for the Argent Job Scheduler is:

ARGENT_JOB_SCHEDULER_4_AUG_2008_14_22_33.LIC

Product license files must not be deleted.

Argent recommends storing all license files received from Argent in the _KEY sub-directory.

Internal Tables

The Argent Queue Engine maintains status information in four critically important files within the product home directory:

- JobsSlot.bin
- JobTemplate.bin
- PerfData.bin
- QueueSlot.bin

These files contain product-specific internal information about each job, job template, and queue supported by the Argent Queue Engine. Each of these files are assumed and required to be available to the Argent Queue Engine at all times. These files can not be inspected, opened, read, or manipulated in any way by customer processes, including anti-virus programs.

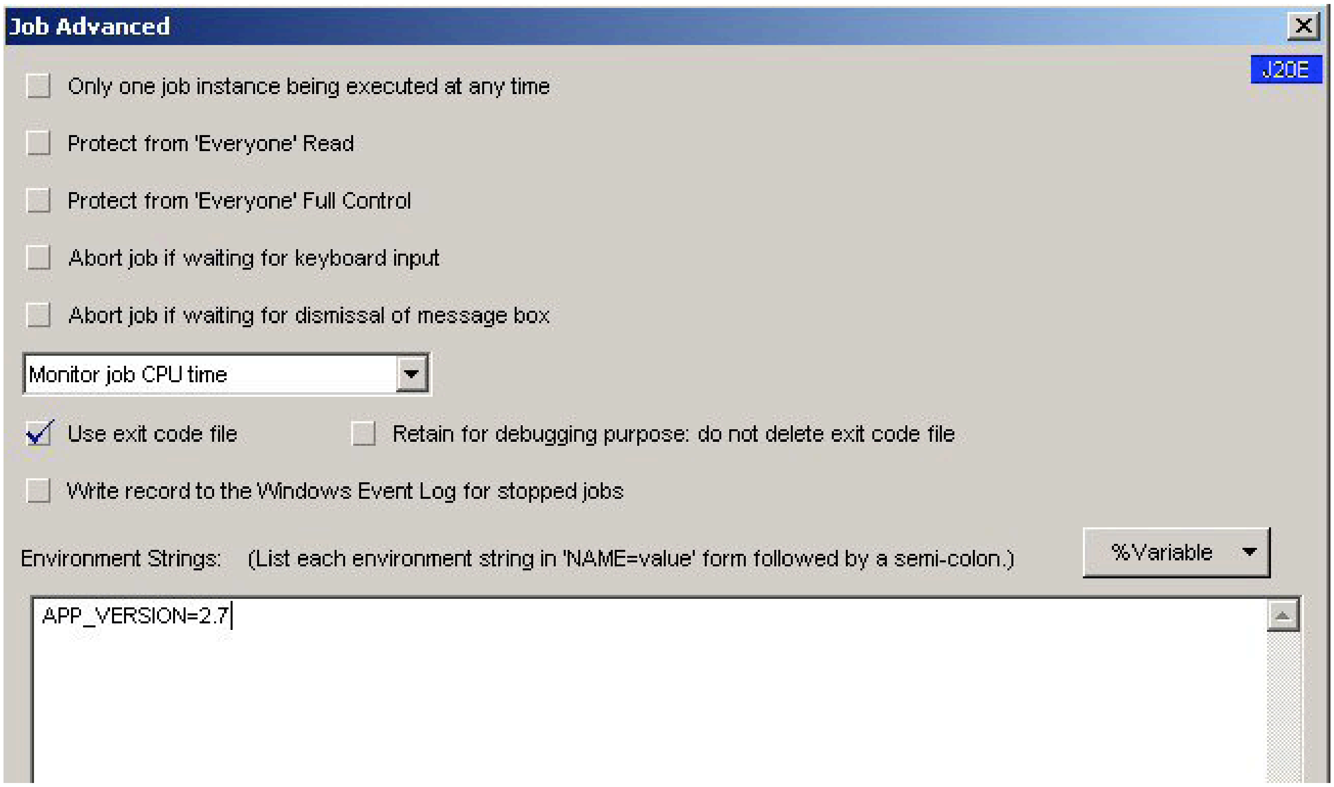
Environment Strings, Substitution Variables, and Global Text String Macros

The Argent Job Scheduler supports three important constructs that can be combined to provide great flexibility to customer-developed applications.

Environment Strings

Environment string support provides a way for a customer to set the value of an environment variable that will be accessible to customer jobs during execution. Environment strings are defined for a job using the Advanced Settings display (Screen ID J20E).

A customer application, for example, could use an environment string to indicate an application version. This is shown below:



Job Advanced

J20E

□ Only one job instance being executed at any time

□ Protect from 'Everyone' Read

□ Protect from 'Everyone' Full Control

□ Abort job if waiting for keyboard input

□ Abort job if waiting for dismissal of message box

Monitor job CPU time

☑ Use exit code file □ Retain for debugging purpose: do not delete exit code file

□ Write record to the Windows Event Log for stopped jobs

Environment Strings: (List each environment string in 'NAME=value' form followed by a semi-colon.)

%Variable

```
APP_VERSION=2.7
```

Environment strings can be read by a customer application using batch command file syntax ("%VARIABLE_NAME%") or by calling Windows APIs.

Substitution Variables

The Argent Job Scheduler supports several expressions known as substitution variables. They are:

%ServerNode%	- Name of node where job executes
%JOB_FILE%	- Job command file name
%JOB_NAME%	- Job name
%JOB_NUMBER%	- Job number assigned by Queue Engine
%LOG_FILE%	- Log file name
%QUEUE%	- Queue name assigned by Queue Engine
%EXIT_CODE_FILE%	- Exit Code file name
%JOB_STATUS%	- Job status

Substitution variables can be used in job parameters, environment strings, and alert text.

Please note that the job number and queue name are assigned by the Argent Queue Engine after the job has started executing and are not available until the job has started.

A job that has %JOB_NUMBER% defined in an alert text that is sent when the job fails to start will not have the expected results since, by definition, the job number is not available until the job has started executing.

A job that has a list of queues or a queue macro expressed within the job definition should not use the %QUEUE% substitution variable until the job has started because the value of the queue name is not resolved until the job is ready to start.

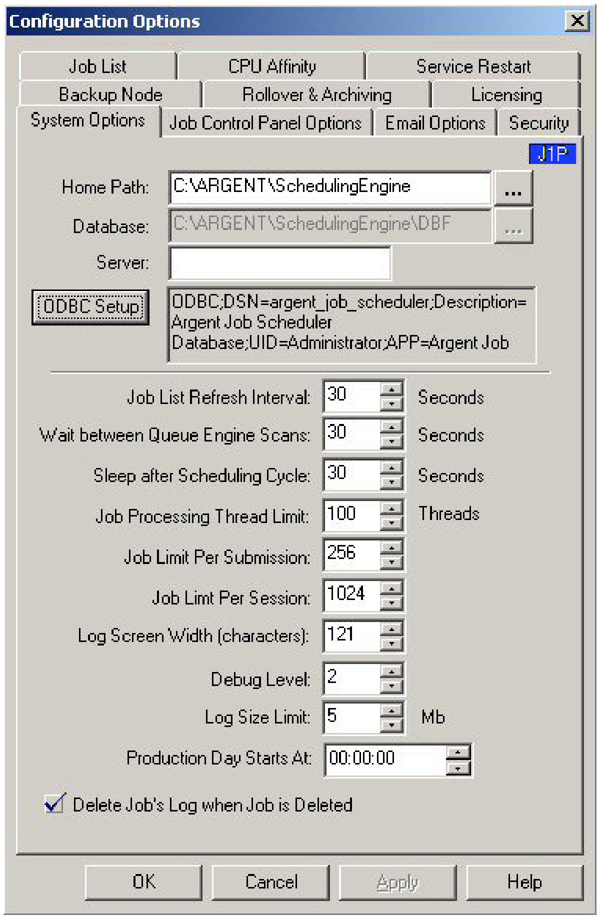
Global Text String Macros

Global text string macros can be defined using the J48 display. Global text string macros can be used in job parameters, environment strings, and alert text. A global text string can be defined, for example, that expresses the current date. The global text string macro can then be used as a job parameter to pass the current date to the job.

Notes on Critical Configuration Settings

This section highlights some Job Scheduler configuration settings that can be critically important in production environments.

System Options



Home Path: C:\ARGENT\SchedulingEngine

Database: C:\ARGENT\SchedulingEngine\DBF

Server:

ODBC Setup: ODBC;DSN=argent_job_scheduler;Description=Argent Job Scheduler Database;UID=Administrator;APP=Argent Job

Job List Refresh Interval: 30 Seconds

Wait between Queue Engine Scans: 30 Seconds

Sleep after Scheduling Cycle: 30 Seconds

Job Processing Thread Limit: 100 Threads

Job Limit Per Submission: 256

Job Limt Per Session: 1024

Log Screen Width (characters): 121

Debug Level: 2

Log Size Limit: 5 Mb

Production Day Starts At: 00:00:00

☑ Delete Job's Log when Job is Deleted

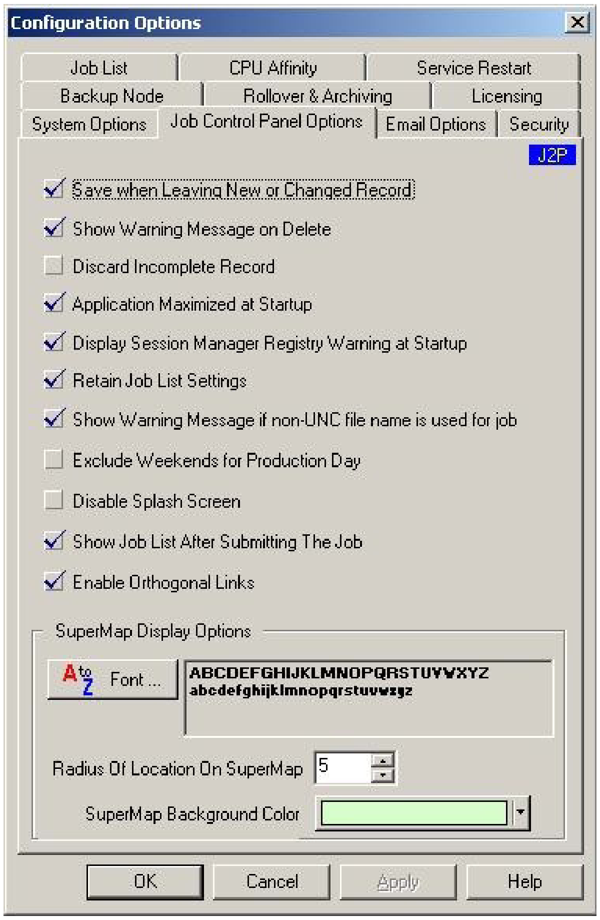
Home Path, Database, ODBC Setup

The values for these three prompts can only be changed during upgrade and installation processing. Changing the value of the Home Path, for example, can disrupt production environments by making it impossible for the Argent Job Scheduler to find installed components of the product.

Changing the ODBC definition, for example, does not result in migration of database tables from an older database to a newer database. Changing the ODBC can only be successfully performed as part of a comprehensive database migration strategy assisted by trained and experienced Argent Support and technical professionals.

Changing these values directly using the J1P display is not supported. These values can only be changed on the advice of trained and experienced Argent support and technical professionals.

Job Control Panel Options



☑ Save when Leaving New or Changed Record

☑ Show Warning Message on Delete

☐ Discard Incomplete Record

☑ Application Maximized at Startup

☑ Display Session Manager Registry Warning at Startup

☑ Retain Job List Settings

☑ Show Warning Message if non-UNC file name is used for job

☐ Exclude Weekends for Production Day

☐ Disable Splash Screen

☑ Show Job List After Submitting The Job

☑ Enable Orthogonal Links

SuperMap Display Options

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz

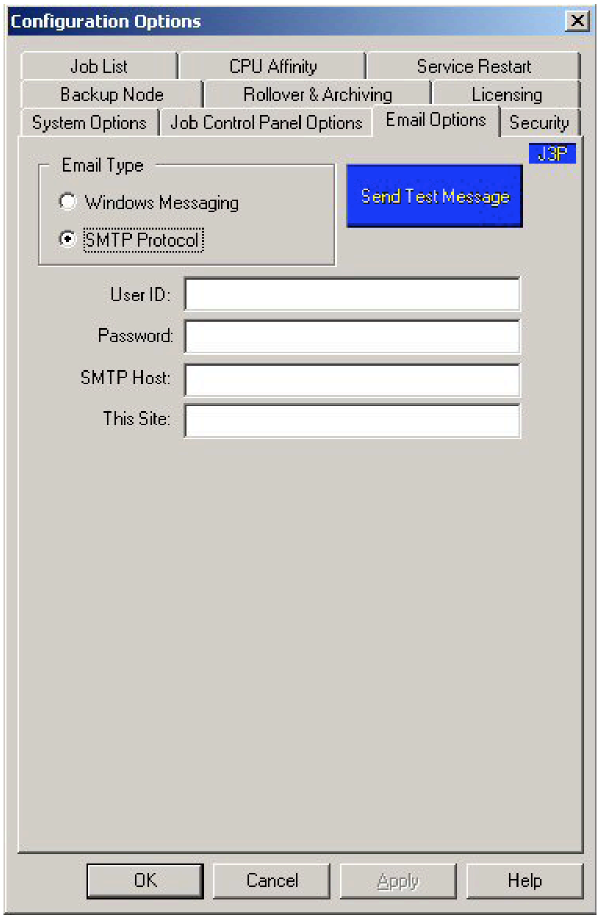
Radius Of Location On SuperMap: 5

SuperMap Background Color: 🟩

Show Warning Message on Delete

Checking this option instructs Job Scheduler to present a confirmation dialog when Job Scheduler definitions are deleted. When this option is un-checked, this important confirmation dialog is suppressed. Argent strongly recommends leaving this option checked.

Email Options



Email Type

○ Windows Messaging

● SMTP Protocol

User ID:

Password:

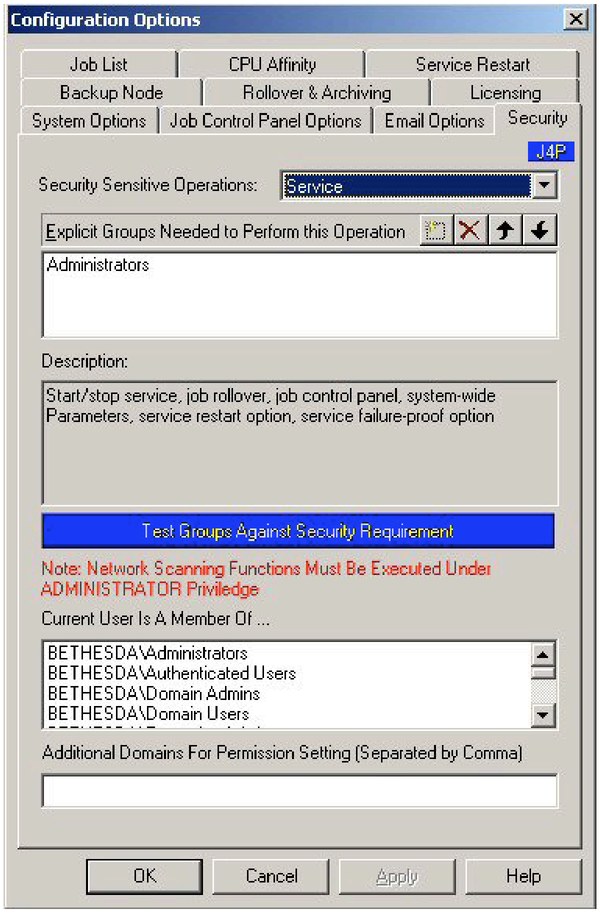
SMTP Host:

This Site:

Email options should only be changed during an outage period when the Argent Job Scheduler and the Argent Queue Engine are both quiesced, and all Job Scheduler and Queue Engine services have been ended.

Changing e-mail configuration options while the Job Scheduler or Queue Engine is active is not supported. Changing this option while the Job Scheduler or Queue Engine is active could result in important alert messages failing to reach intended recipients.

Security



Security Sensitive Operations: Service

Explicit Groups Needed to Perform this Operation

Administrators

Description:

Start/stop service, job rollover, job control panel, system-wide Parameters, service restart option, service failure-proof option

Note: Network Scanning Functions Must Be Executed Under ADMINISTRATOR Priviledge

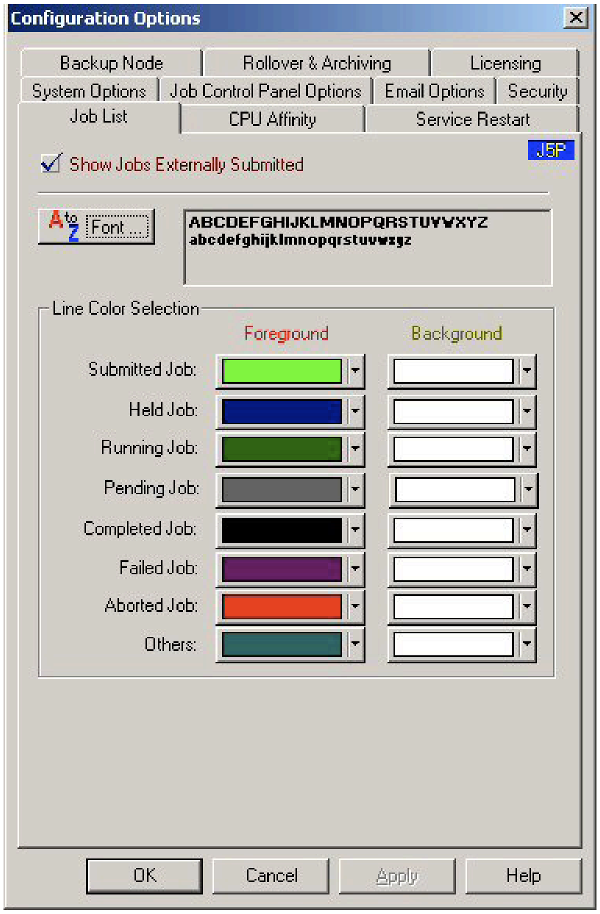
Current User Is A Member Of ...

BETHESDA\Administrators
BETHESDA\Authenticated Users
BETHESDA\Domain Admins
BETHESDA\Domain Users

Additional Domains For Permission Setting (Separated by Comma)

Customers should review usage of the Job Scheduler Control Panel (GUI) and grant access only to users authorized to make changes. Starting or stopping the Job Scheduler Scheduling Engine service, for example, should only made available to authorized operators and administrators.

Job List



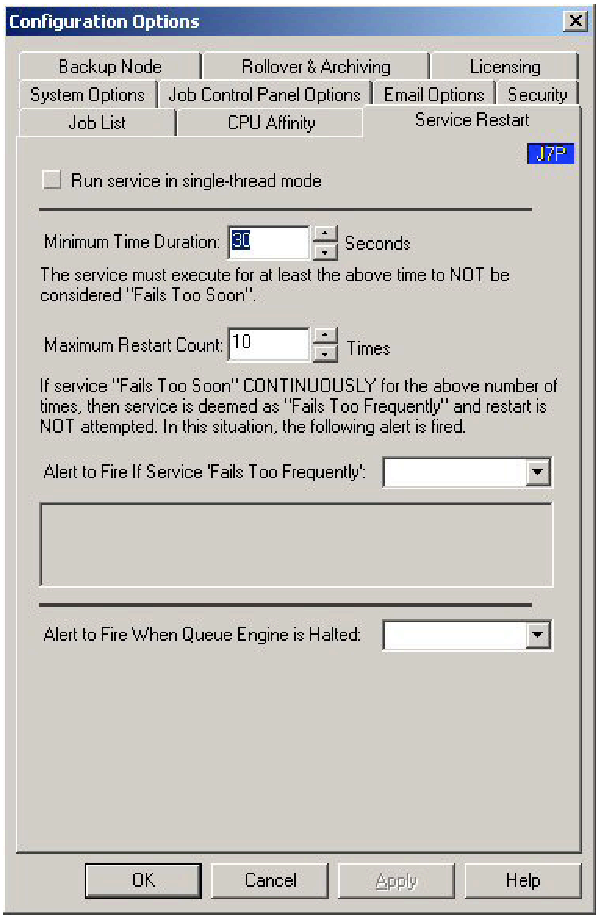
☑ Show Jobs Externally Submitted

Line Color Selection

	Foreground	Background
Submitted Job:	🟩	⬜
Held Job:	🟦	⬜
Running Job:	🟩	⬜
Pending Job:	⬛	⬜
Completed Job:	⬛	⬜
Failed Job:	🟪	⬜
Aborted Job:	🟥	⬜
Others:	🟦	⬜

Prior to the Argent Job Scheduler version 7.0A-0801-J, the “Show Jobs Externally Submitted” option exhibited errant behavior that could result in large numbers and volumes of files accumulating in the C:\ARGENT\QueueEngine\JobDef directory. While this issue has been addressed, Customers using a version of the Argent Job Scheduler prior to version 7.0A-0801-J must leave this option unchecked.

Service Restart



☐ Run service in single-thread mode

Minimum Time Duration: 30 Seconds

The service must execute for at least the above time to NOT be considered "Fails Too Soon".

Maximum Restart Count: 10 Times

If service "Fails Too Soon" CONTINUOUSLY for the above number of times, then service is deemed as "Fails Too Frequently" and restart is NOT attempted. In this situation, the following alert is fired.

Alert to Fire If Service 'Fails Too Frequently':

Alert to Fire When Queue Engine is Halted:

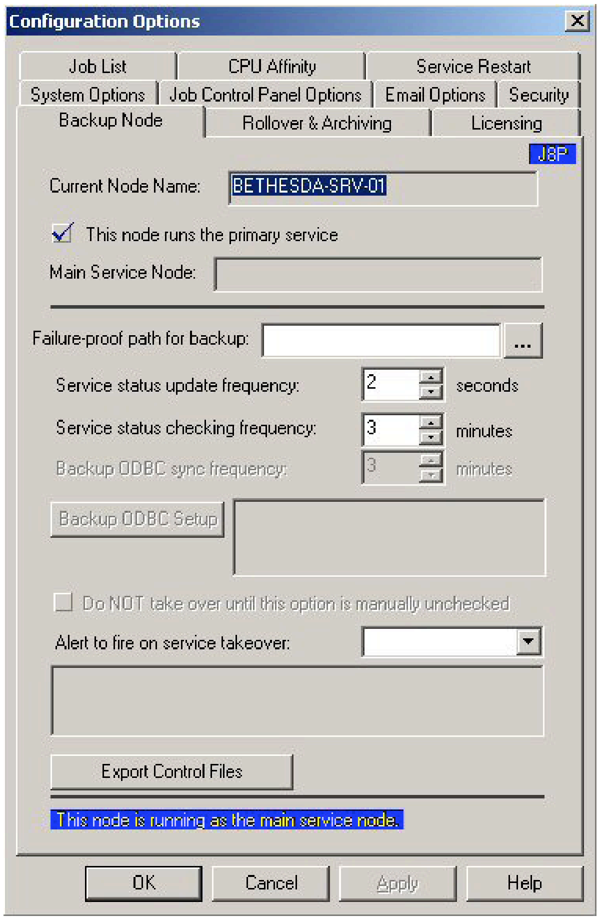
The “Run service in single-thread mode” option must never be checked. This option is obsolete and is no longer supported by the product. Checking this option could result in disruptions for production Job Scheduler environments.

Historically, Microsoft has delivered extremely poor and/or marginal support for controlling service programs. This configuration display originally attempted to provide configurability, management, and control in scenarios where Microsoft’s Windows Service Control Manager (SCM) behaved errantly or not at all.

Customers experiencing misbehavior such as *“The Argent Job Scheduler continues to run smoothly, but I can not stop or re-start the Windows service”* must contact Argent Support for assistance.

Customer must not change the values on this display except on the advice of trained and experienced Argent Support and technical professionals.

Backup Node



Current Node Name: BETHESDA-SRV-01

☑ This node runs the primary service

Main Service Node:

Failure-proof path for backup:

Service status update frequency: 2 seconds

Service status checking frequency: 3 minutes

Backup ODBC sync frequency: 3 minutes

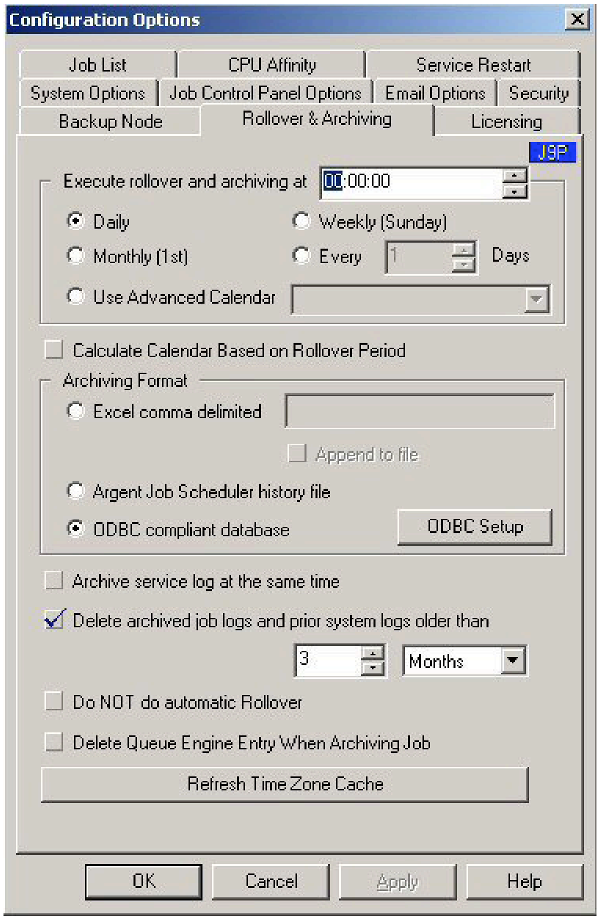
□ Do NOT take over until this option is manually unchecked

Alert to fire on service takeover:

This node is running as the main service node.

Changes to this display are only supported during setup and installation. Changes to the values on this display can only be made at the direction of trained and experienced Argent Support and technical professionals.

Rollover and Archiving



Execute rollover and archiving at 00:00:00

● Daily ○ Weekly (Sunday)

○ Monthly (1st) ○ Every 1 Days

○ Use Advanced Calendar

□ Calculate Calendar Based on Rollover Period

Archiving Format

○ Excel comma delimited

□ Append to file

○ Argent Job Scheduler history file

● ODBC compliant database

□ Archive service log at the same time

☑ Delete archived job logs and prior system logs older than 3 Months

□ Do NOT do automatic Rollover

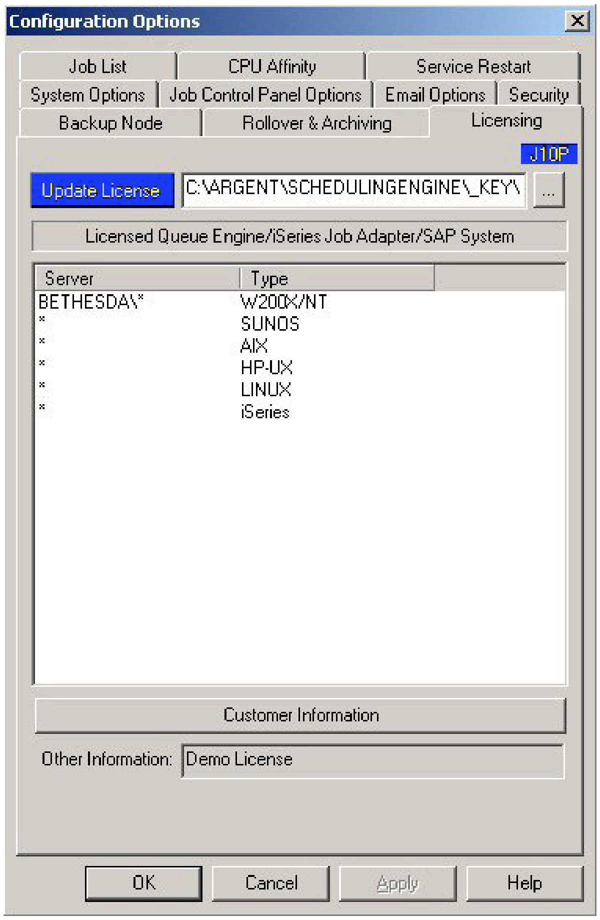
□ Delete Queue Engine Entry When Archiving Job

Argent strongly recommends that customer leave rollover processing values at their configured defaults. Argent does not recommend changing the time or the periodicity of rollover processing.

Rollover processing should be executed daily at midnight.

Other values can cause confusion about the definition of a “production day” and can cause non-expert Job Scheduler users to become concerned.

Licensing



C:\ARGENT\SCHEDULINGENGINE_KEY\

Licensed Queue Engine/iSeries Job Adapter/SAP System

Server	Type
BETHESDA*	W200X/NT
*	SUNOS
*	AIX
*	HP-UX
*	LINUX
*	iSeries

Other Information: Demo License

This configuration display is provided as a way for customer to update and renew their license agreements with Argent. Changes to this display should only be made on the advice of trained and experienced Argent Support and technical professionals.

Note: ArgSoft Intellectual Property Holdings Limited has created this White Paper for informational purposes only. ArgSoft Intellectual Property Holdings Limited makes no warranties, express or implied, in this document. The information contained in this document is subject to change without notice. ArgSoft Intellectual Property Holdings Limited shall not be liable for any technical or editorial errors, or omissions contained in this document, nor for incidental, indirect or consequential damages resulting from the furnishing, performance, or use of the material contained in this document, or the document itself. All views expressed are opinions of ArgSoft Intellectual Property Holdings Limited. All trademarks are the property of their respective owners.