Robot/SCHEDULE Enterprise®

Enterprise Job Scheduling

Robot/SCHEDULE Overview

With more than 15,000 users worldwide, Robot/SCHEDULE® is the most popular job scheduling software for the IBM® System i® (iSeries,® AS/400®). It's easy to see why. Robot/SCHEDULE runs jobs according to a time schedule, in reaction to other events on the system, or after checking on the completion of prerequisite jobs. A Robot/SCHEDULE-operated system runs more smoothly, reliably, and with fewer operational crises than those operated by human operators. And, when you add Robot/SCHEDULE Enterprise, you have a complete enterprise job scheduling solution for your Windows, Linux, and UNIX servers.

What Is Robot/SCHEDULE Enterprise?

You know how Robot/SCHEDULE automates the operation of your System i, allowing it to run unattended, reliably, and at a lower cost. But, your business systems include more than the System i. You also need to manage tasks, events, and processes on Windows, UNIX, and Linux servers. And, you have more important things to do than monitor and manage these machines manually through tasks like launching batch processes, status checking, and file transfers. You need a way to coordinate all your systems for maximum performance and availability.

Robot/SCHEDULE Enterprise is fully integrated with Robot/SCHEDULE to coordinate batch processes and monitor system events across your Windows, UNIX, and Linux systems. You can schedule jobs on the System i using Robot/SCHEDULE for your entire enterprise. Then, the Robot/SCHEDULE Enterprise Server coordinates agent activity for each server, checks server operational status, receives job status information, launches processes, monitors server applications and services, transfers data, and much more. Robot/SCHEDULE Enterprise:

- Extends Robot/SCHEDULE's powerful scheduling features to Windows, UNIX, and Linux servers, including flexible date scheduling, group and reactive jobs, and completion history and reporting.
- Uses event-driven scheduling based on file arrival, directory updates, and services or daemons starting or ending.
- Provides a single user interface to manage your entire enterprise schedule.

End the guessing games that result when jobs have cross-platform dependencies. By adding Robot/SCHEDULE Enterprise, you are adding over a quarter century of job scheduling experience to the Windows, UNIX, and Linux servers on your network.

Schedule And Monitor Jobs Across Your Enterprise

These days, batch job scheduling is an enterprise-wide endeavor. Nightly processes can require integration with non-System i servers, and System i data is often a prerequisite for batch processes to run on a UNIX, Windows, or Linux server. Previously, this was possible only by managing these tasks manually or by using timers. The problem with these approaches is that they are error-prone and not always timely. The ultimate goal is to build an event-driven schedule that takes all environments into account.

Robot/SCHEDULE Enterprise lets you build an event-driven schedule across all your systems for enterprise scheduling, coordinated batch processing, and cross-system monitoring. Robot/SCHEDULE Enterprise works across all your servers to:

- Monitor essential services and daemons
- Schedule batch jobs across your enterprise
- Monitor file arrival and directory creation or modification
- Provide event-driven scheduling across different platforms
- Monitor for return codes from your scripts to determine job success or failure
- Capture output from agent systems for troubleshooting
- Verify availability of the server
- Control the user and working directory on the server

You can use your favorite Robot/SCHEDULE functions—schedule forecasting, job schedule blueprints, job completion history—to schedule and manage jobs and processes on your Windows, UNIX, and Linux servers. And, because it's integrated with Robot/SCHEDULE, Robot/SCHEDULE Enterprise uses the same terminology and familiar, easy-to-use interface.

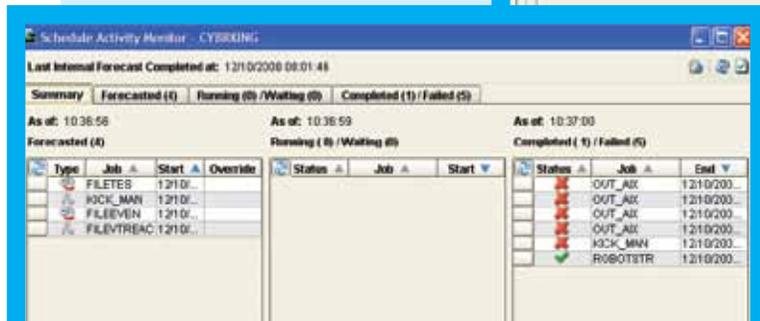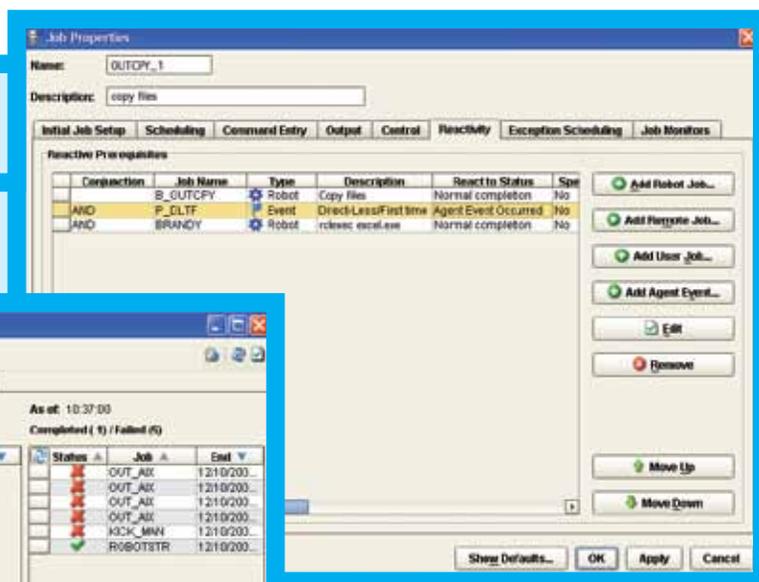# heduling For Your Enterprise

Save Time And Money With Robot/SCHEDULE Enterprise

There is no doubt that automating your operations with Robot/SCHEDULE drastically reduces job scheduling errors, which ultimately saves you time and money. But, the benefits of using Robot/SCHEDULE don't stop there. Robot/SCHEDULE Enterprise is also a great economic solution for your company because it leverages your investment in Robot/SCHEDULE on other platforms. Since it integrates seamlessly with Robot/SCHEDULE, you can use it with other products in the Robot Automated Operations Solution, including

Robot/NETWORK®, Robot/CONSOLE®, and Robot/ALERT.® It also gives your operators more free time since they won't need to spend time manually managing jobs on your servers or learning a new software program—you do it all through the Robot/SCHEDULE Explorer.

Let Robot/SCHEDULE Enterprise make your multi-platform environment—Windows, IBM AIX, Sun Solaris, HP-UX, or Linux—as reliable and easy to use as your System i. You've experienced the power of Robot/SCHEDULE on the System i—now experience it on the rest of your enterprise.

You can define server events as prerequisites to Robot/SCHEDULE jobs for true cross-system reactivity.

The Schedule Activity Monitor (SAM) displays the status of jobs on the servers in your enterprise.
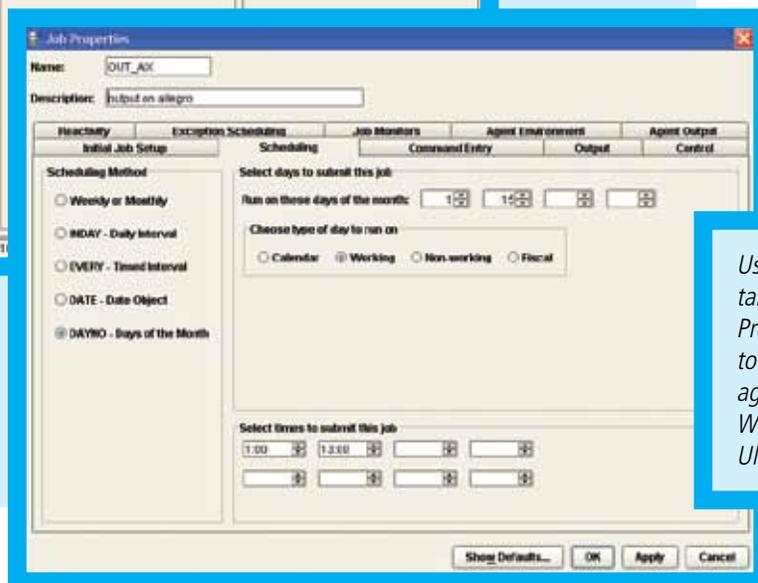
Use the Scheduling tab of the Job Properties window to schedule an agent job on your Windows, Linux, or UNIX server.

Define And Manage Jobs Across Your Network—From One Console

Robot/SCHEDULE Enterprise is your central console for job scheduling across your enterprise. You use Robot/SCHEDULE's easy-to-use graphical user interface, the Robot/SCHEDULE Explorer, to manage jobs and monitor events on your Windows, UNIX, or Linux servers. See agent jobs using job blueprints, schedule forecasts, the job schedule list, the job completion history, and the Schedule Activity Monitor—the same tools you use to view all your existing Robot/SCHEDULE jobs.

The Explorer lets you manage your job schedule and access all the tools for working with your jobs. From the Explorer, you can launch or display other Robot/SCHEDULE tools—the Job Monitor Events Log, the Job Creation Wizard, and the Job Schedule List Viewer—directly from the toolbar. You don't have to physically move to another system to set up jobs on your Windows, UNIX, or Linux servers. You also can use the Explorer's filtering options to view only agent jobs, or use the Tree View to see agent jobs sorted by agent system.

Built-in File Transfers

Built-in file transfers allow you to transfer files across your enterprise easily and securely, using FTP and FTPS. The easy-to-use function builder sets up the file and directory information, and Robot/SCHEDULE Enterprise adds the appropriate FTP command to the job. This helps satisfy PCI, SOX, HIPAA, and other regulations where your auditors require data to be secure.

Share Agent Scheduling Objects To Save Time

Agent jobs can use three scheduling objects: commands (scripts) that need to run, an environment to run them in, and return codes to let the operator know how the job completed. Defining these objects on thousands of jobs would be very time-consuming, especially when many of the jobs use identical settings.

Robot/SCHEDULE Enterprise allows you to create agent command sets, agent environments, and agent return codes that can be shared by many different jobs on any of your servers. For example, if you need to run the same script to delete temporary files on your Windows servers, you can use the same command set to make sure that each job is done the same way, without the overhead of setting up each Windows server individually. Or, if all batch processes on your UNIX server need the same working directory, user, and environment variables, you can create one agent environment to handle this. By using shared scheduling objects, you save time when you need to update your jobs. Instead of updating the settings for every job, you can update the scheduling object once and be done.

Automate Your Custom Scripts

Robot/SCHEDULE Enterprise makes it easy to run your custom CRON jobs or Windows BAT files as part of your enterprise business processes. By doing this, your scripts run as part of an event-driven schedule across all your platforms, so you don't have to worry about relying on timed jobs in your schedule. You can even tie in event notification so that all processes are monitored, which allows you to keep up on your SLA's.

You can use Robot/SCHEDULE Enterprise commands to set a reserved command variable, which are dynamic variables used throughout the Robot products. For example, you could use a reserved command variable in your script to automatically insert the current date and time into a bank transaction file.

Identify Server Issues With Robot/SCHEDULE Enterprise Tools

Robot/SCHEDULE Enterprise allows you to easily troubleshoot your servers from a central location. Agent output options allow you to coordinate error log information from your servers. You can send error logs directly to a spooled file or store them as a text file on the IFS and view them at your convenience. You never even have to leave your desk—the results appear in the
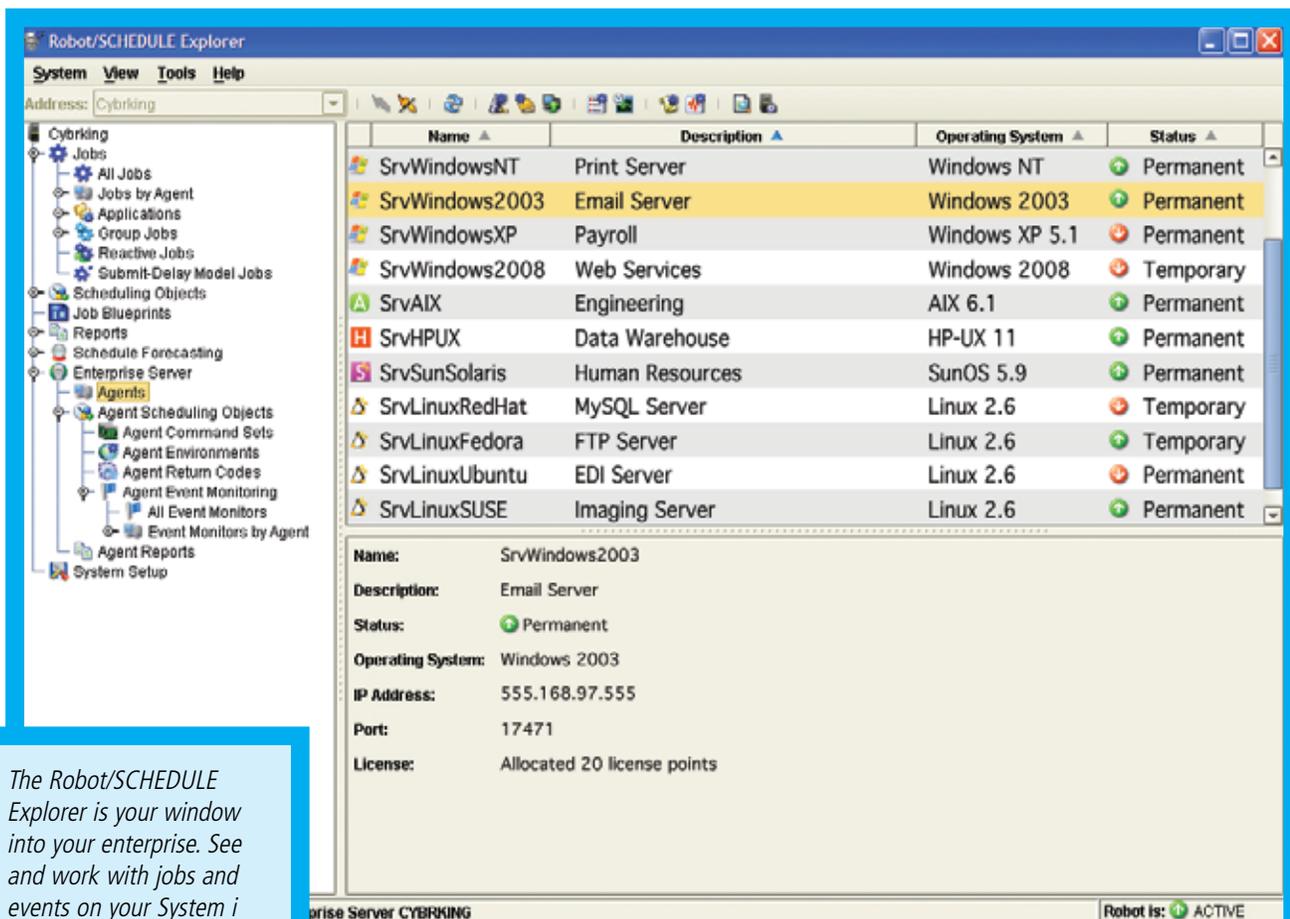
job completion history. You can even start a Telnet session to the server through Robot/SCHEDULE Enterprise for quick troubleshooting access.

Since Robot/SCHEDULE Enterprise is fully integrated with Robot/SCHEDULE, you can use Robot/SCHEDULE's troubleshooting and reporting features, including the Good Morning Report, audits, Job Schedule List, Schedule Activity Monitor, job completion history, and schedule forecasting. And, since you are managing your servers from a graphical user interface, you can easily share troubleshooting information with your operations team by copying text from Robot/SCHEDULE Explorer windows and pasting it into an e-mail message or a report. You can satisfy your auditors, management, and operations team.

Job Blueprints

Robot/SCHEDULE's job schedule blueprint provides a graphical view of all, or part, of your job schedule and identifies the relationships between jobs, including prerequisite and reactive jobs, and group members. This becomes even more important as you monitor jobs and events from multiple servers in your enterprise.

You can submit a blueprint for all agent events or individual jobs that belong to a particular agent server. A blueprint for an event maps out the jobs that react to the event. You also can submit a blueprint for a specific event. The job blueprint shows other jobs and agent events related to the job, such as a file arriving or a process starting or ending.

The Robot/SCHEDULE Explorer is your window into your enterprise. See and work with jobs and events on your System i and your enterprise servers.

Create Event-Driven Schedules With Event Monitors

Many of the scheduling needs on Windows, UNIX, and Linux servers are based on file arrival information. Typically, a file arrives from an external source, such as a retail point-of-sale (POS) machine, and a job needs to be launched to handle the file. Robot/SCHEDULE Enterprise Event Monitors make this very easy by allowing you to monitor for events such as changes to a file or directory, a file that's not growing, when a process starts or ends, or your own user-defined events. Then, you just define the monitored event as a prerequisite for a Robot/SCHEDULE reactive job. Event-driven scheduling takes away the guessing game for enterprise scheduling. And, Robot/SCHEDULE Enterprise ensures that your jobs run when they're supposed to—even when the triggering event is on another server.

Jobs can be triggered by an application, a Windows service, or a daemon starting. For example, you might check every minute to make sure a critical daemon is running on your AIX server. If the daemon ends, you can launch a script to restart it and notify your team. Application developers also can trigger events from their programs through API interfaces to Robot/SCHEDULE Enterprise. Robot/SCHEDULE Enterprise maintains a complete history of monitored events, saving time if you need to diagnose a processing problem quickly.

A Typical Scenario

Most enterprise schedules rely on timed events or manual procedures to process information. Consider a chain of retail stores whose end-of-day process starts at 10:00 p.m. when the managers transmit sales and inventory data from their Linux POS terminals to a series of Windows servers. At 10:30, each server forwards the data to an AIX server, which runs a job to consolidate the data. At 11:30, an FTP process sends the consolidated data to a System i, which posts the day's transactions. Following the 2:00 a.m. backup, a reporting job scheduled at 3:30 sends the updated inventory files back to the Windows servers for distribution to the individual stores in time for the next business day.

Sounds easy; but what happens if all the events don't complete as planned—a manager forgets to send a file; one of the Windows servers locks up; the consolidation job runs before all the Windows servers have sent their transaction files; or the operator misses the tape mount message during the backup? A schedule based on time exposes you to missed deadlines and information that's out of date.

The flow chart on the next page shows how Robot/SCHEDULE Enterprise manages the same process.

The job schedule blueprint shows an event-driven schedule that starts when the POS file arrives on an agent system.
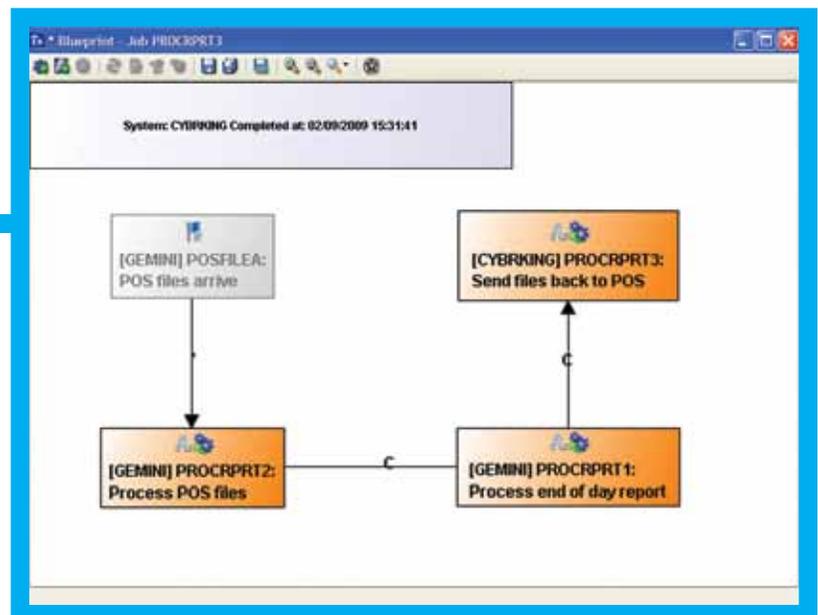
s To Events On Agent Systems

1 A scheduled job at each store sends the day's sales and inventory data from the Linux POS terminal to one of the Windows servers.

LINUX POS
LINUX POS
LINUX POS
LINUX POS
LINUX POS
LINUX POS
LINUX POS
LINUX POS

7 At the end of the process, the administrator receives a report from Robot/SCHEDULE that shows the completion times and statuses for the night. A hidden benefit is that the process finishes earlier because it is all event-driven. If any issues arise during the night, Robot/SCHEDULE makes sure someone is notified.

2 When all prerequisite events are met on each Windows server, a Robot/SCHEDULE reactive job sends the data to the AIX server. An independent process for each Windows server runs to verify that the data transfers complete by a specified time. On-call staff is notified automatically if a process is running late.

WINDOWS
WINDOWS
WINDOWS
WINDOWS

6 When each Windows server receives its files, it launches processes to send the new files to each store POS terminal.

3 On the AIX server, separate file event monitors check for the file transfers from each Windows server. Once all files arrive, a reactive job consolidates the data and sends a completion status to the System i.

4 A reactive job on the System i runs an FTP GET operation to retrieve the data from the AIX server and post the day's transactions. Completion of the job triggers the nightly backup on the System i.

SYSTEM i

AIX SERVER

5 When the backup completes, several reactive jobs run the final reports. The report jobs are prerequisites for the jobs that send the updated inventory files back to the Windows servers.

Join The 21,000 Customers Using Our Automated Operations Solution

Build A Solution Set To Automate Your Enterprise Schedule

When you buy any Robot product from Help/Systems, you're also buying the opportunity to integrate other Robot products to create an automation solution set that solves your System i operations problems. You'll save money, simplify tasks, eliminate bottlenecks, improve processing, reduce errors, and satisfy your auditors. The benefits and possibilities are unlimited. Here's how you can make Robot/SCHEDULE Enterprise a part of your automation solution set.

Robot/SCHEDULE Enterprise, Robot/SCHEDULE, Robot/NETWORK, And Robot/ALERT

Managing multiple System i systems in a network can be an all-consuming job. When you add Windows, UNIX, and Linux servers to your enterprise, you need to coordinate jobs on each system, and often jobs or events on a server are prerequisites to jobs on the System i. By combining Robot/SCHEDULE Enterprise with Robot/NETWORK, the network management software, an event on your Linux server can trigger a Robot/SCHEDULE job on a Robot/NETWORK Node, which in turn sends a completion status to the Host system.

Including Robot/NETWORK in your solution set helps you manage the Robot products and job schedules across multiple System i systems or partitions. The Robot/SCHEDULE Master on the Host system provides centralized distribution of job scheduling instructions, while the Robot/NETWORK Status Center lets you monitor the status of batch jobs on all your System i systems from a central location. If an important job fails, you see the system in trouble. Adding Robot/ALERT to your solution set allows each Robot product to send an e-mail, text, or pager message to notify you if a job fails or requires attention.

This complete integration allows your Windows, UNIX, Linux, and System i servers to work together to automate your enterprise schedule.

Award-Winning Robot Automated Operations Solution

Whether you have a single System i, or a network of them, Help/Systems is committed to providing you with products that automate their operation. The products of the Robot Automated Operations Solution look and act the same. All of the Help/Systems products talk to each other through the Solution's common component interface. This integration makes all the products powerful, yet easy to learn and use. Our product expertise has won us many awards from numerous publications. And our customers have awarded us 60,000 times with product purchases.

Commitment To Excellence

Help/Systems became America's first ISO 9001-certified software company in 1992. Since then, we have maintained our certification under the updated 9001:2000 standard. This international quality standard covers software design, development, marketing, product support, and training. Help/Systems continues to demonstrate that it has an excellent software quality assurance system in place, full management commitment to quality, and a well-trained and motivated staff. This certification applies to all company procedures for ensuring customer satisfaction—from those done by the receptionist to the duties of the CEO.

help/systems

World's Leader in Power Systems Software Solutions

952/933-0609
info@helpsystems.com
www.helpsystems.com