ISE is extending the boundaries of automated job scheduling with the only job scheduler for Windows NT/2000/XP OpenVMS, Tru64 UNIX and Linux. Each version incorporates ISE's unparalleled reputation in the management and automated scheduling of vast quantaties of interdependent tasks. This, along with unmatched support and the experience of more than 25 years in the busines, comprise a scheduling package you won't want to live without.

In today's dynamic environments, where thousands of tasks are processed 24 hours a day, 365 days a year, EnterpriseSCHEDULE is the job scheduler of choice. While most scheduling software runs on only one platform, EnterpriseSCHEDULE is the scheduler that spans the widest spectrum, with OpenVMS, Tru64 UNIX, Windows and Linux versions that fully integrate with each other through the use of a common data format. So regardless of the platform, the controlling central server is able to manage job activity to the high standards ISE customers have grown to expect. In fact, ISE has made each version of the client so similar in functionality, there's virtually no learning curve in migrating from one platform to another.

This unparalleled capability allows consistency to be maintained throughout a company's enterprise reducing or eliminating the learning curve required in mastering an advanced job scheduler on multiple platforms. The software fully integrates across platforms through the use of a common data format. The controlling server is able to manage all job activity to the high standards ISE customers have grown to expect, regardless of platform. This includes a wide range of time based, resource dependent and variable scheduling options that add unmatched versatility and simplicity to your jobs. Agent software brings even more operating environments into the mix, resulting in the most comprehensive scheduling package available.

Features

Four popular platforms, same quality and ease of use

EnterpriseSCHEDULE is a job scheduling solution that initiates, monitors and tracks job activity on OpenVMS, UNIX and Windows NT platforms. The database can be distributed over the

entire network, eliminating any single point of failure. Job
processing can be executed locally on individual nodes or distributed across any combination of these platforms from a central server. Jobs can be scheduled at any interval, on a named day basis or using a calendar with marks for run days.

The software fully integrates across platforms through the use of a common data format. The controlling server is able to manage all job activity to the high standards ISE customers have grown to expect, regardless of platform. Each version of the client was engineered with the user in mind with a consistent user interface that requires virtually no cross training when migrating from one platform to another.

Applications can be developed into highly reliable `fault tolerant' job sets, ideally suited to the emerging trend for `lights out' operations and `hands off' solutions. EnterpriseSCHEDULE eliminates the need for regular operator attendance and effectively makes these applications self correcting. It does so by taking remedial action automatically when a fault occurs and running jobs through to their termination

Interfaces to suit any user's need

The new Windows based client interface that connects seamlessly to any
EnterpriseSCHEDULE server whether it be Windows, Open VMS, UNIX or Linux makes it easier than ever to control, modify and monitor job activity from a central Windows client. And EnterpriseSCHEDULE has a legacy of quality clients that span the spectrum from the versatile MOTIF GUI to the powerful command line (using commands native to the operating system) to the MCL menu system for OpenVMS. The following chart lists where each client can be used:

Available EnterpriseSCHEDULE client interfaces by Platform

	Windows	OpenVMS	UNIX	Linux
Windows Client Interface	✓	✓	✓	✓

MOTIF GUI	✓	✓	✓	✓
Command Line	✓	✓	✓	✓
VT Menu Interface		✓		

Two ways to link job streams using interdependencies

Only EnterpriseSCHEDULE offers the versatility of two types of interdependencies between jobs:

1) an initiate which triggers other jobs and

2) a prerequisite that holds up other jobs until completion.

Identifying dependencies and making connections between tasks and processes, whether within the same cluster or across an entire enterprise-wide network that includes multiple platforms, turns complex scenarios into manageable processing streams. Conditional progression of job streams can be accomplished through programmable resources that determine execution of tasks based on variable status, completion status, resource states and a variety of other factors.

EnterpriseSCHEDULE features automatic restarts, so you can specify that jobs are to be automatically started upon failure and define the number of job restarts to occur before you are notified of a job failure. You can even design jobs to have restart points so that when a job is restarted, it does so at the appropriate point

Advanced conditional job flow

EnterpriseSCHEDULE gives you more control of your job flow with fully programmable resources. Using this powerful feature, you can add crucial decision points that allow you to control execution based on variable evaluation. With the built in conventional programming tools, a job stream can be based on a wide variety of algorithms. Job streams can be simple or complex and can evaluate the availability of system resources. These streams, which are viewable in a live display, are automated and optionally redundant making them both interdependent and user controlled.

Object oriented data set makes for better organization

The EnterpriseSCHEDULE database is organized to keep jobs, calendar definitions, variables and other data in an object oriented fashion that makes accessing and modifying data easier. All objects are organized in easy to access directories (folders in Windows) that classify all objects by type. In the Windows Explorer, all objects are subcategorized in a folder according to type. Jobs can be stored by category, job stream or whatever means desired.

Object oriented categorization also helps prevent data redundancy by alowing for a single instance of calendars, variables etc. that can be accessed by multiple jobs. This object approach means thatwhen a calendar is saved in one place, all deployed jobs using that calendar automatically reflect any changes made to the calendar

A configurable central database

Distributed or centralized support means the EnterpriseSCHEDULE database can be configured across a network of systems either as a centralized network with satellites or as a group of peer-to-peer databases. The entire EnterpriseSCHEDULE database is organized into directories. Any number of directories can be created. Any number of concurrent users can use the system. Access to the job database is controlled from platform-dependent security methods and platform spanning ACLs.

EnterpriseSCHEDULE Cluster technology leveraged

	WIndows	OpenVMS	UNIX	Linux
Shared database	✓	✓	✓	✓
Lock resource manager	✓	✓	✓	✓

Server to server auto failover	✓	✓*
Job submission at the logical cluster level	✓	✓*

* In development for Tru64 UNIX

Fully monitor job activity

With EnterpriseSCHEDULE, comprehensive job status is always just a click away. Live displays monitor job progress and deliver notices via broadcast messages or e-mail when abnormalities occur. Users can watch as jobs cycle through their scheduled runs and the system automatically performs crucial tasks.

The new Windows client interface features even more monitoring capability, with a colorful display that let’s you monitor job runs by job or by event. Current status of jobs as well as system resource usage is also available.

Versatile and easy to manipulate time based controls

Three distinct rescheduling methods are available, based on:

1) start time and an interval.

2) name and time of the day to run the job.

3) a marked calendar (e.g. those days marked with an `X') and the time of day.

Calendar based job scheduling enables you to create calendars that recognize holidays and weekends when jobs are not to run, and mark specific days on which jobs are to run. Additionally you can create fiscal calendars for jobs concerning financial matters (such as periodic financial reports).

The ISE Advantage

Reliability

With thousands of critical tasks on the line, system reliability is integral to data center operations. Companies can be assured of 24 x 365 resilience with the hot-swap capabilities; multi-site, disaster-tolerant clusters; and dynamic server resource reallocation of OpenVMS on AlphaServer systems. EnterpriseSCHEDULE also works continuously without any resets, so cluster rollover capabilities are automatically available in the system.

Scalability

In todays dynamic business climate, companies need an affordable system now that can accommodate theirfuture growth requirements. EnterpriseSCHEDULE is highly scalable, accommodating unpredictable and massive growth. EnterpriseSCHEDULE utilizes today's dynamic processor allocation features via automatic scheduling to meet the job load. In the meantime, the quantity of automated jobs can scale right along with a system as it expands.

Diversity

EnterpriseSCHEDULE spans operating platforms, including the Windows NT, OpenVMS, UNIX and Linux operating systems. Jobs and processes can be created on and these diverse systems and the data they generate and control can be completely integrated throughout the enterprise.

Server to server Design

EnterpriseSCHEDULE features a cooperative architecture through agents and homogeneous servers talking to each to each other. This distributes processing and is extremely fault tolerant (no single point of failure as in Master/Agent architecture).

Embedded database

The EnterpriseSCHEDULE database is embedded, independent and complete There is need fo third party database support necessary

Cost savings

Because the Enterprise SCHEDULE system is diverse across a full spectrum of platforms, it saves time and money in purchase time, reduction in maintenence costs and the need for additional servers

Highlights at a glance

Job Specifications

- **Job command list can be any operating system command or script (Windows NT batch or executable, Unix Shell scripts, or DCL command file)**
- **Jobs can be passed multiple parameters**
- **Resource variables add conditional programmability to job processing**
- **All job, calendar, variable etc. data stored in central or distributed database proprietary to EnterpriseSCHEDULE**
- **Prerequisites hold up subsequent jobs**
- **Initiates signal subsequent jobs to execute based on exit status**
- **Prejob actions check conditions before job executes**
- **Unlimited numbers of resource requirements**
- **Post job actions execute after job runs**
- **Two level folders similar to Windows folder system.**
- **Jobs are stored in folders. The look and feel resemble the native file system, but the SCHEDULE contents actually reside in a database. This database supports 2 levels of folders.**

Job Scheduling Options

- **Advanced calendars allow selective scheduling based on a marked calendar.**
- **Named day scheduling allows the processing of jobs on particular days of the week at a certain time.**
- **Interval scheduling allows jobs to be processed at a selected interval.**
- **Customizable calendar settings (ie. based on fiscal calendar etc.)**
- **Job processes can run concurrently, in a stream or based on resource status.**
- **Jobs can be scheduled with Hold status**
- **Job runs can be based process previous job events, resource status, scheduled events, manual submission and variable status.**
- **Job runs can be based on the creation or modification of a file**

Fault tolerance

- **High Availability fault tolerant server logic includes multiple level exception handling at code level**
- **Handles system errors and unforeseen errors**

- **Server rollover at cluster level**
- **Definable server activity classes**

Monitoring of job status

- **Job progress is monitored through a series of states representing important events in a jobs execution.**
- **Current status of job available throughout job run and fillowing complation**
- **Notification of events can be sent via pager, email or pop up window.**
- **Job Alerts (in Windows Interface) sends pop up messages indicating important events**
- **History data reports provide valuable data about past job activity**

Job Group Capabilities

- **Up to 10 independent Schedule server groups can be maintained simultaneously.**
- **For example, groups 0 – 9, databases and communication are separate.**
- **Example usage – group 0 production jobs group 1 development jobs**
- **Installation produces group 0 automatically. Additional groups can be produced**

Windows NT is a trademark of Microsoft Corp.
Tru64 UNIX, VMS and OpenVMS are trademarks of Compaq