

# EASY/Exit™

from DTS SOFTWARE

- *Does your site use hard-to-maintain system exits?*
- *Would you like to improve data security with a simpler method for exits?*
- *Would you like to improve DFSMSHsm Recall performance?*
- *Do end-of-month jobs fail due to increased CPU time needs?*

If you answered “yes” to any of these questions, **EASY/Exit®** is for you! EASY/Exit eliminates the dangerous, error-prone task of writing, updating, and maintaining systemwide exits such as IEFUTL (job wait time or CPU time exceeded), IEFACTRT (end-of-job/step statistics), or IEFUJI (job initiation). Instead, EASY/Exit replaces complicated, often unreadable assembler-language exits with easy-to-understand **ACS-like rules** that allow system administrators to **see the logic of these critical routines at a glance**.



## Simple, Obvious, Easy-to-Use

Instead of hundreds of lines of esoteric, poorly documented, and sometimes obsolete code, sites which use EASY/Exit can rely on the simple yet powerful IF-THEN-ELSE statements of the DTS Software rules language to provide critical functionality in a way that is clearly understandable even to the most junior members of the systems programming staff.



## Why Exit Routines?

No operating system, even one as powerful as z/OS and DFSMS, can provide all the flexibility that is required to meet the needs of every installation. Traditionally, sites have relied on exit routines, written and maintained by skilled programmers, to tailor the operating environment and system processing to fill in the gaps and to extend the reach of the operating system. Critical systems such as **DFSMSHsm, TSO, and others** also make use of exits to overcome their limitations.



### The Problem with Exit Routines

Many installations have shied away from the use of exits, even when the **benefits are obvious**, because of the difficulty of writing and maintaining system-level code. Years of assembler-language programming expertise is required, along with great caution. An error introduced into an exit can bring the entire system's processing to a halt. Testing exit routines requires special care, and often frequent IPLs. Once written, exit routines are rarely changed because of the difficulty of revising and maintaining the code.



### Reliability and Functionality with EASY/Exit

With EASY/Exit, installations can safely and reliably use the system's exit routines — **without any assembler code** — to implement systemwide policies and provide

new functions that were never available before. By coding a simple series of IF-THEN-ELSE statements, system administrators can automate such tasks as:

- Adding exit routines to improve security for FTP, TSO Logons/Logoffs, and WTO/WTORs, and RACF
- Control the amount of allowable region, hiperspace, and other resource use
- Specify the automatic actions that particular jobs waiting for devices or volumes should take
- Allow the most important datasets to be recalled first by DFSMSHsm
- Control migration of datasets during DFSMSHsm primary or secondary space management
- Automatically alert the storage administrator when backup of the DFSMSHsm control datasets fails