



hp services

education

course description

Guardian API programming (U4152S)

course overview

This 5-day course is for C, TAL and pTAL developers. You will learn how to use the Guardian API to obtain operating system services—interprocess communication, process creation, Enscribe file access, and more. Hands-on lab exercises give you many opportunities to practice the techniques learned in the classroom discussions.

audience

- Systems programmers
- Application programmers
- Programmer analysts

benefits to you

- Coding for interprocess communication
- Developing applications that access Enscribe files
- How to handle terminal input and output
- Different types of system messages
- Creating and managing new processes
- Available timing facilities
- How `nowait` I/O can be used to improve application performance

prerequisites

- Concepts and Facilities course
- TAL/pTAL Programming course or C Programming in the Guardian Environment course

next steps

NonStop Kernel Architecture

to order

You can order this course online at <http://education.hp.com>. At the site, select a country, then choose "registration" or "Book a course" and fill out the online registration form.

why hp education?

- Experienced and best-in-the-field HP instructors
- Comprehensive student materials
- State-of-the-art classroom facilities
- Hands-on practice
- Focus on job-specific skills
- More than 120 locations worldwide
- Customized on-site delivery
- Online instructor-led and self-paced training at <http://itresourcecenter.hp.com>

guardian API programming (U4152S)

module	key topics
introduction to guardian API	<ul style="list-style-type: none">• Major functions of the NonStop Kernel operating system• Identifying the required header or declaration file for any Guardian procedure• The NonStop Kernel's message-based architecture• Lab Exercise: Determine the appropriate procedure to use for a given function• Lab Exercise: Identify the parameter types used by a given procedure• Lab Exercise: Determine whether a given procedure returns a value or sets the condition code
file system concepts	<ul style="list-style-type: none">• Concept of a file and purpose of an I/O process and different types of file names• Coding the procedure calls that are common to all file accesses• Using the DEFINE and ASSIGN functions to equate physical and logical file names• Coding the procedure calls to perform file-name conversion• Lab Exercise: Study the different methods of supplying a file name at execution time• Lab Exercise: Gain experience with the Inspect debugger
interprocess communication	<ul style="list-style-type: none">• Requester-server application model• Coding the procedures for interprocess communication• Differences between system and application messages and TACL program startup protocol• Lab Exercise: Code a program to use the INITIALIZER or get_startup_msg procedure to handle the TACL program startup protocol• Lab Exercise: Code the program to handle system open and close messages
Enscribe database access	<ul style="list-style-type: none">• Enscribe file types and coding the procedures to create Enscribe files• Coding the procedures to access and manage access of any Enscribe file and appropriate error checking• Lab Exercise: Code programs to perform Enscribe file accessing including updating• Lab Exercise: Apply record locks
terminal input and output	<ul style="list-style-type: none">• Determining the home terminal of a process and describe its use• Procedures used to write to and read from a terminal• Errors that can occur during I/O• Numeric conversions that might need to be performed on input data and use of the FIXSTRING procedure• Lab Exercise: Code a program to determine the process home terminal and perform I/O operations to it• Lab Exercise: Use interprocess communication to pass data to a server process
nowait I/O processing	<ul style="list-style-type: none">• Coding the procedures for nowait I/O operations• One function of the SETMODE procedure and Internal and external request management• Lab Exercise: Modify a given program to perform nowait I/O on selected files
process management	<ul style="list-style-type: none">• Calling the procedures used to create a new process• Calling the PROCESS_GETINFOLIST_ procedure to obtain information about processes• HIGHPIN and HIGHREQUESTERS object file flags and Lab Exercise: Code a procedure to start executing a new process
NonStop TS/MP	<ul style="list-style-type: none">• Major components of the NonStop Transaction Services/MP (NonStop TS/MP) environment• Capabilities of a Pathsend requester and procedures used by Pathsend requesters• Design considerations of NonStop TS/MP servers• Lab Exercise: Modify a program to use the Pathsend procedures
data integrity and application availability	<ul style="list-style-type: none">• Components of the NonStop TM/MP subsystem• Functions performed by NonStop TM/MP and techniques used to ensure application availability

for more information

For more information on HP Education Services, contact any of our worldwide offices or visit our worldwide web site on the internet at <http://education.hp.com>

Technical information in this document is subject to change without notice.

Microsoft®, Windows®, MS Windows®, and Windows NT® are U.S. registered trademarks of Microsoft Corporation. UNIX® is a registered trademark of the Open Group.

©Copyright Hewlett-Packard Company 2000. All Rights Reserved. Reproduction, adaptation, or translation without prior written permission is prohibited except as allowed under the copyright laws.

9/02 U4152S

