



hp NonStop servers



a data sheet
from hp

hp general device support extended software

hp GDSX software enables customized device processing and supports multiple network devices

features at a glance

- Integration of special-purpose devices
- User customization
- Built-in management services
- Standard application program interface

standard interface for network devices

HP General Device Support Extended (GDSX) software allows you to add customized device processing capabilities to the HP NonStop Transaction Services/MP (NonStop TS/MP) transaction processing system by serving as an intermediary between NonStop TS/MP and I/O processes. GDSX software is designed primarily to enable a single HP NonStop server application to support many different network devices, preserving your investment in equipment such as point-of-sale (POS) terminals, scanners, and robotics devices. It can also be used as a front-end process to implement or modify data communications protocols, or as a back-end process to concentrate messaging-switching functions. GDSX software can be customized to support nearly any function, from simple data-stream conversion to complex context switching and network resource management.

GDSX software is fully integrated with the Distributed Systems Management (DSM) software products, enabling your operations staff to monitor and control its functions. For compatibility with older application programs, the original Generalized Device Support (GDS) product is included with the GDSX software package.

integration of special-purpose devices

The primary function of GDSX software is to integrate special-purpose devices into networks of NonStop servers (see figure 1).

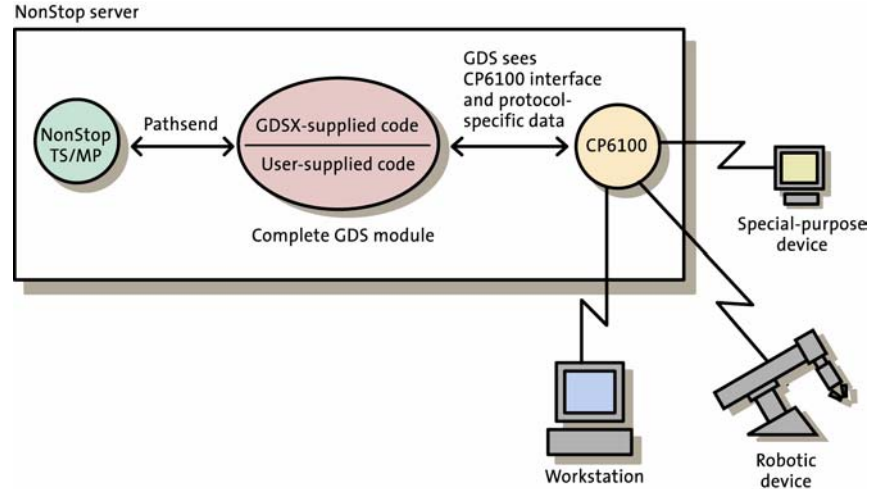


Figure 1. GDSX software allows almost any device to communicate with a NonStop server.

The GDSX process serves as the interface between a NonStop server application, such as NonStop TS/MP software, and an input/output process, such as CP6100. This is useful in situations such as the following:

- The required data communications protocols are not supported by the Pathway/TS Terminal Control Program (TCP).
- Performance is critical. Interpreted Screen COBOL (SCOBOL) may not be efficient enough for programming a data communications protocol. GDSX software can call Pathsend for the most efficient communication with NonStop servers running NonStop TS/MP software.
- Simplicity of programming is critical. Handling a data communications protocol can be awkward with some third-generation languages because they have a limited number of data types and verbs.
- A back-end process must concentrate and serialize requests to and from shared devices, such as devices connected via the Envoy/Anypoint Connection Protocol (ACP).
- A front-end process must concentrate and serialize requests, such as those from automated teller machines (ATMs).

Using GDSX software, programmers can convert the data stream of a special purpose device to make it compatible with a 653x or IBM 3270 terminal data streams. With this capability, an application need only support GDSX software; GDSX modules support the various network devices. As you add terminals or other devices to the system, you can create a GDSX module for each device, with a single application supporting all of them.

user customization

GDSX software consists of two components: GDSX-supplied code (GDSXCODE) and user-supplied code (USCODE; see figure 2).

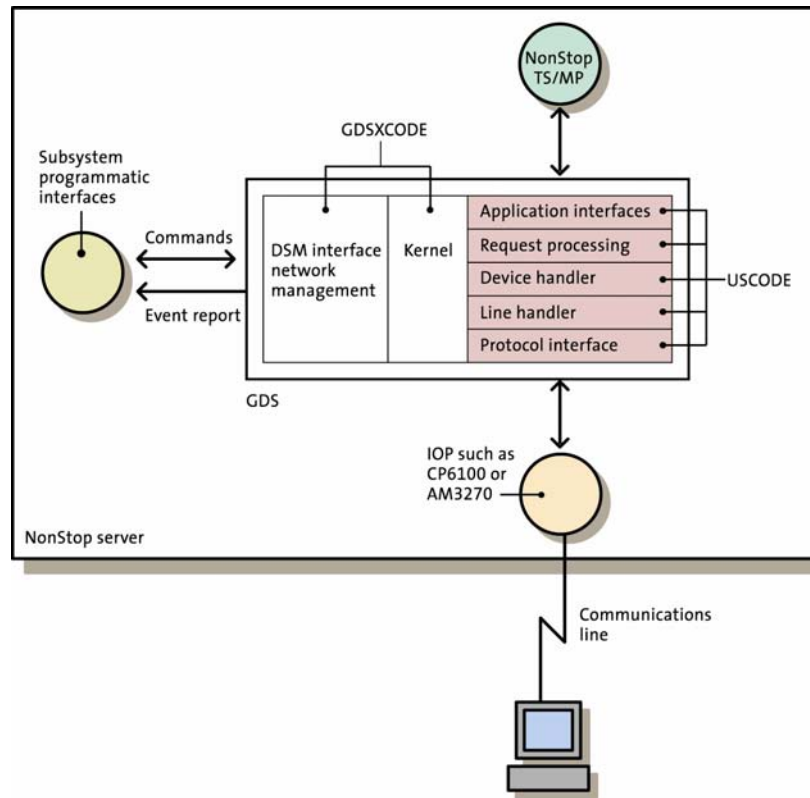


Figure 2. Major components and architecture of GDSX software.

GDSXCODE provides generic service and management routines that help programmers create multi-threaded fault-tolerant processes.

USCODE can be customized to support specific devices that you want to integrate with your NonStop server. It consists of a number of procedures (called user exits) that are called by GDSXCODE to handle data manipulation, protocol conversion, and message routing.

USCODE must be written in Transaction Application Language (TAL). It is compiled separately before being bound with GDSXCODE to produce a functional GDSX process. GDSX software is an accelerated TAL process.

Two of the most important user exits in the USCODE are the device handler (required) and line handler (optional).

device handler for heterogeneous connectivity

With GDSX software, you can write your own device handler to enable an application to communicate with any type of device. With this capability, you can develop an

application without concern for the devices with which it will need to communicate. You can modify the device-handler routines to add functions such as data-stream manipulation and screen formatting.

The device handler is typically used to modify the communication of each thread between a requester or server and another file (often a terminal).

optional line handler for versatile communications

GDSX programmers can develop a line handler to the communications input/output process to manage messages routed to a specified communications line. (A line-handler procedure can also be used to route messages to an external process.) The line handler allows requester threads to use a shared resource—often a communications line—without interfering with each other.

The line handler is line-protocol dependent. It is required only if the access method with which GDSX software interfaces does not provide line-handler functions, for example, interfacing to multidrop lines using CP6100 or to low-level access methods such as X.25 or AM3270 access method.

built-in management services

GDSX software provides a set of management services that includes the DSM family of software products for monitoring and controlling network activities efficiently. GDSX software supports the DSM program interface and Event Management Service (EMS), enabling you to collect, route, and process operations information such as alerts and statistics.

The DSM Subsystem Control Facility (SCF)/Subsystem Programmatic Interface (SPI) offers two types of access: conversational input by means of the SCF or programmatic control by means of a user-written management application that uses the SPI.

The conversational command-line (operator) SCF interface includes generic commands to configure and manage subdevice unit (SU), line, and GDSX objects; control the tracing facility; and display statistics on object status and resource allocation. You can also add commands or modify attributes to extend the functionality of the interface by customizing its user exits.

standard application program interface

The GDSX application program interface (API), called request processing, supports a set of standard system calls for communicating with application programs. The GDSX kernel passes requests from an application to the user-supplied section of the GDSX module.

GDSX request processing fully supports the use of Pathsend, HP NonStop Transaction Management Facility (TMF) software, setmodes, and other NonStop system internals. You can add your own routines to handle special requests.

ordering information

<i>product ID</i>	<i>description</i>
-------------------	--------------------

specifications

server requirements

hardware	Any NonStop server
software	HP NonStop Kernel operating system, any supported software release

utility procedures

The following groups of procedures are available to developers with GDSX software:

- Bit map control
- Break handling
- Debugging facilities
- Fault-tolerance facilities
- Identifier manipulation
- Intertask communication
- I/O control block maintenance
- List manipulation
- Pool management
- Semaphore control
- Task maintenance
- NonStop Kernel procedure calls
- NonStop TMF and Pathsend software

For more information, go to www.hp.com/go/nonstop.

February 2003, first published 1994. All product names mentioned herein may be trademarks of their respective companies. HP shall not be liable for technical or editorial errors or omissions contained herein. The information is subject to change without notice. The warranties for HP products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.



©2003 Hewlett-Packard Development Company, L.P.