hp®

invent

HP-UX 11.X Technology Review

March 2001

John Fenwick
Hewlett-Packard
19447 Pruneridge Avenue
Cupertino, CA 95014 USA
408-447-4976
FAX 408-447-4364
John_Fenwick@hp.com

Program Outline

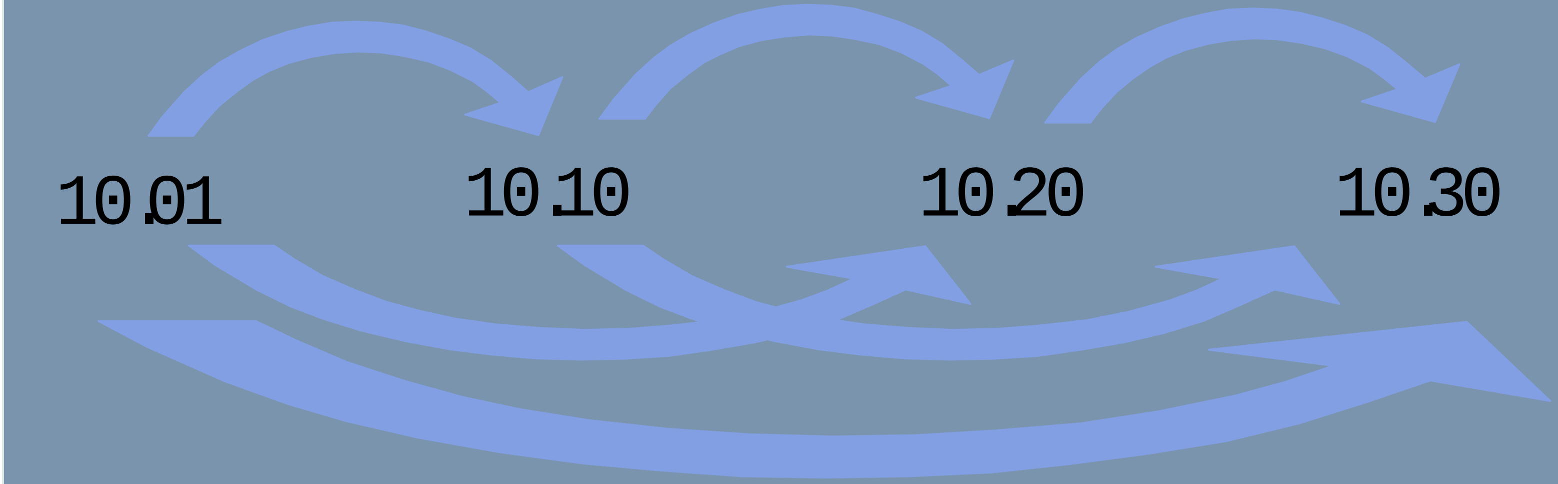
- Introductions and Overview
- "Largeness" Extensions to HP-UX:
  - Filesystems, Files, UIDs, File Descriptors, Process Space
- 64-Bit HP-UX
- Developing for 64-bits
- HP-UX Software Transition Kits
- HP-UX 11 Distributions, Operating Environments
- HP-UX 11.0/11i Technologies:
  - Threads, DLKM, Large Pages, Process Space, System Recovery
  - Partitions, OLAR I/O, Security
- iCOD Instant Capacity on Demand
- Networking enhancements
- HP-UX 11 Compatibility and Futures
  - HP-UX on IA-64

Program Overview

- Technical level of presentation
  - familiarity with System Administration tools
  - familiarity with Software Development concepts
  - familiarity with Operating System fundamentals
- Format of presentation
- References for further study
- Questions

"Largeness" Features in HP-UX: Review of HP-UX 10.X Releases

- Each release is a superset of the previous
- HP-UX 10.01 is the "gateway" to the family
- Upgrade when and if you need



HP-UX 10.10 : CQ1 1996
Main new features

- 128 GB file system
- 3.75 GB RAM (T500, 0.75 GB cards)
- 1.9 GB process data space
- 60K File descriptors/process
- Shared LVM (SLVM) - for OPS
- Spec 1170 (UNIX95)
- CDE - Common Desktop Environment
- 4 byte EUC commands
- DHCP server (including SAM management)

high-end focus

standards focus

SIZE EXTENSIONS

HP-UX 10.20 : CQ3 1996
Main new features

- Performance
  - PA-8000 optimization, Fibre Channel net/storage
  - MP tuning of transports, stacks, sockets, drivers
  - LVM tuning
  - Processor Affinity support
- Large files (local) - 128 GB files
- >60K UIDs - enable 4 billion user IDs
- 64-bit register math (PA8000)
- 2.75 GB Shared Memory via patch
- DHCP client
- Distributed Print Services (Palladium)

high perf

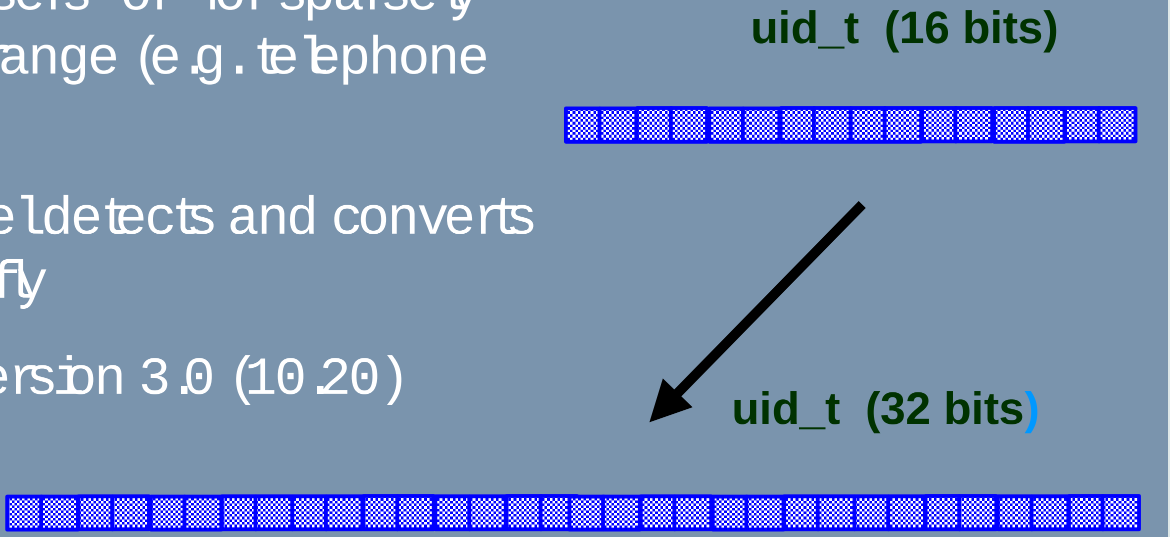
high-end focus

standards focus

SIZE EXTENSIONS

Large UIDs

- Base type of uid increased from 16 to 32 bits
- Changes to kernel, filesystems, libraries, commands, APIs
  - requires recompile to use Large UIDs
- For large number of users -or- for sparsely mapped uids in a large range (e.g. telephone numbers)
- HFS: the HP-UX kernel detects and converts HFS filesystems on the fly
- VxFS: supported in Version 3.0 (10.20)



Using Large UIDs (continued)

```
 < /dev/vg00/lvol4 mounted on /tmp >
% fsadm /dev/vg00/rlvol4
file system          :     /dev/vg00/rlvol4
magic number         :                95014
feature bits         :                    1
file system supports : nolargefiles, longfilenames

% touch lg_uid_file
% chown 99999 lg_uid_file
% ls -l lg_uid_file
-rwxrwxrwx   1 99999   sys   0 Jan 21 15:09 lg_uid_file

% fsadm /dev/vg00/rlvol4
file system          :     /dev/vg00/rlvol4
magic number         :                95014
feature bits         :                    5
file system supports : nolargefiles, largeuids, longfilenames
  <kernel has changed UID structures on the filesystem>
```

HP-UX 10.30 (limited release)
Main new features

- Performance for PA-8000
- 1x1 kernel threads
- 128 GB files (networked)
- NFS Pv3, NIS+
- libc versioning
- Native OpenGL
- Support for new systems & peripherals
- Streams based TCP/IP
- BIND 4.9.3
- Year 2000 clean
- T600

high perf

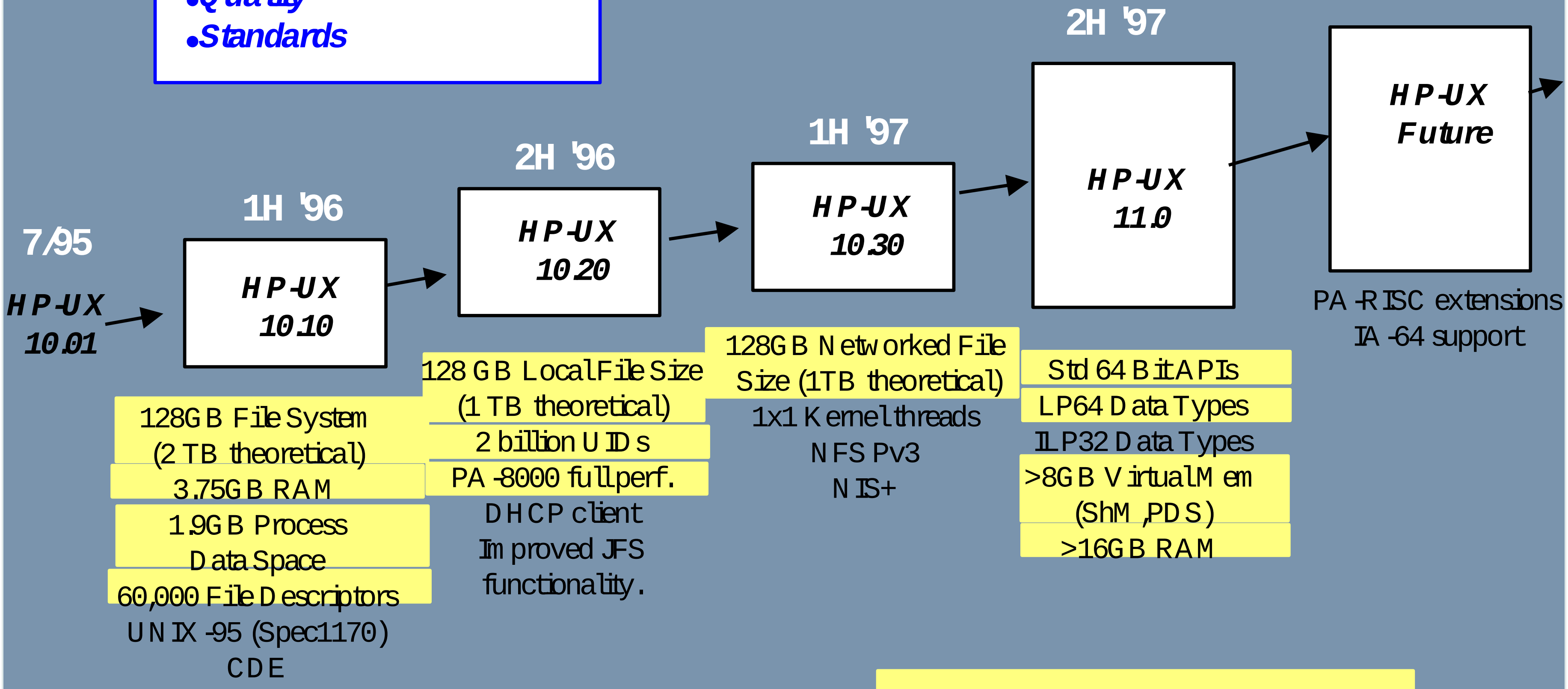
high-end focus

standards focus

SIZE EXTENSIONS

HP-UX Operating System Roadmap

- *Performance*
- *Binary Compatibility*
- *Quality*
- *Standards*



7/95

HP-UX 10.01

1H '96

HP-UX 10.10

128GB File System
(2 TB theoretical)
3.75GB RAM
1.9GB Process
Data Space
60,000 File Descriptors
UNIX-95 (Spec1170)
CDE

2H '96

HP-UX 10.20

128 GB Local File Size
(1 TB theoretical)
2 billion UIDs
PA-8000 full perf.
DHCP client
Improved JFS
functionality.

1H '97

HP-UX 10.30

128GB Networked File
Size (1TB theoretical)
1x1 Kernel threads
NFS Pv3
NIS+

2H '97

HP-UX 11.0

Std 64 Bit APIs
LP64 Data Types
ILP32 Data Types
>8GB Virtual Mem
(ShM, PDS)
>16GB RAM

HP-UX Future

PA-RISC extensions
IA-64 support

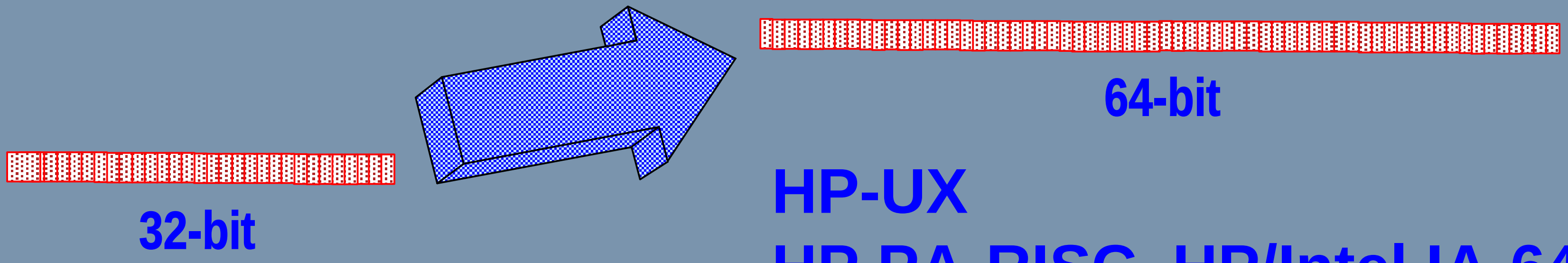
Extend OS Size Capability

Market drivers to 64 bits

- Database vendors, web application servers, and technical software developers have lead the movement to 64-bit computing.

- They have developed databases and applications that handle very large memory and enable access by many more users.

HP's 64-bit strategy: overview



HP-UX
HP PA-RISC, HP/Intel IA-64

HP-UX
HP PA-RISC

Key theme:
Implement evolutionary -- not revolutionary -- product strategies to deliver key new features and protect customers' software investments.

Elements of HP's evolutionary 64-bit strategy

Objective	Why important	Strategy
Lead industry in hardware and software features and performance	Meets customers' increasing functionality and performance demands	Collaborate with partners in developing next-generation UNIX and chip technologies
Provide smooth upgrade path	Ensures investment protection; minimizes upgrade costs in time and money	Continue policy of forward binary compatibility; 32- and 64-bit apps to coexist and communicate
Continue HP's long-standing commitment to standards	Gives partners flexibility in platform selection	Lead standards development and adherence

Benefits of 64-bit computing

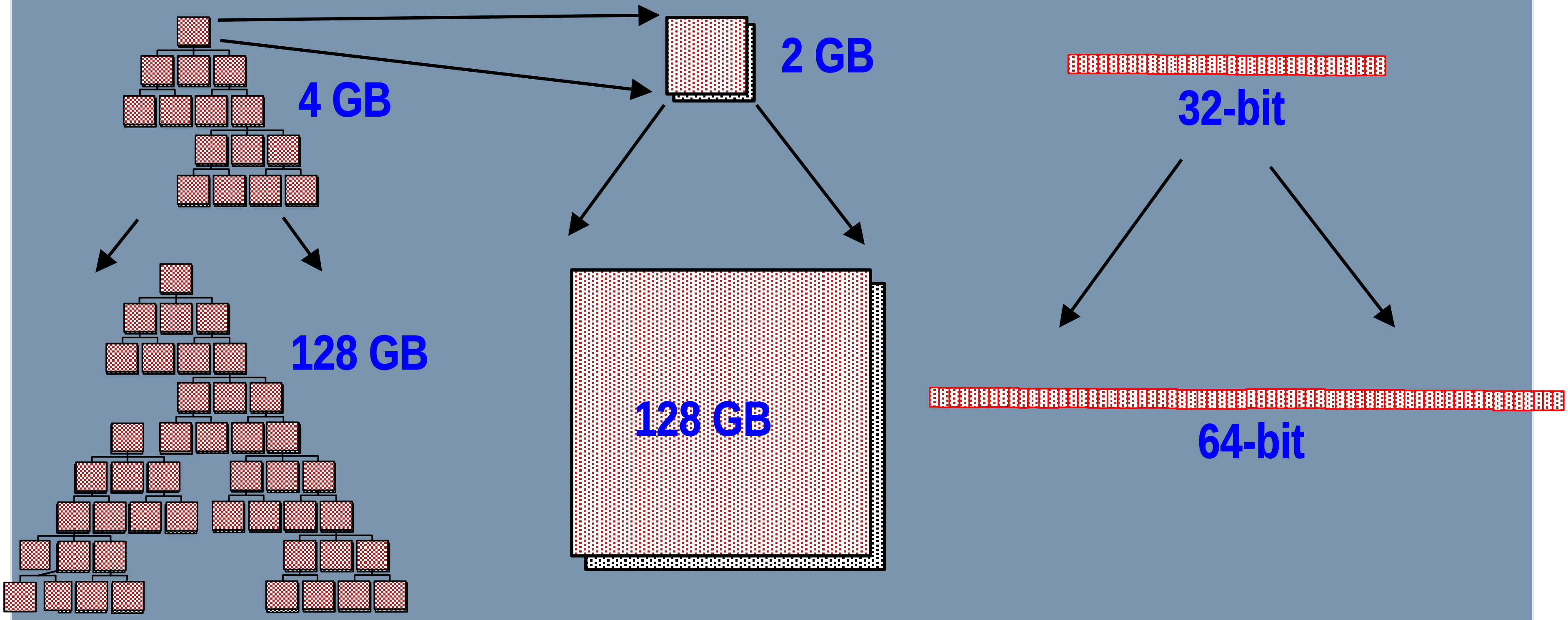
- Scalability
  - Larger applications, larger data spaces, more users
- Potential gains in performance
  - Much larger amounts of data can reside in RAM, resulting in performance gains due to much less (time-consuming) swapping to disk
- These attributes make 64 bits well suited for certain high-end applications
  - very large DB and Decision Support
  - OLTP with tens of thousands of users
  - complex technical simulations
  - web application servers

Smooth upgrade to 64-bit environment

- Investment protection through forward binary compatibility
  - cleanly developed 32-bit applications may run unmodified on 64-bit HP-UX
- No migration; minimal end-user effort
  - No forced recompile
  - No forced recode
  - No data reload

Evolutionary introduction of 64-bit functionality into HP-UX

HP-UX 10.10, 2/96 HP-UX 10.20, 8/96 HP-UX 11.00, 1997



Large file system

Large file size
64-bit register math

Large memory/addressing
64-bit data model

Operating System Data Models: ILP32 and LP64

	ILP32	LP64
Integer	32 bits	32 bits
Long	32 bits	64 bits
Pointer	32 bits	64 bits

- HP-UX 10.0 is ILP32 as are many other UNIXes
- HP-UX 10.[10,20,30] extended the OS capabilities
- HP-UX 11.0 comes in two versions:
  - ILP32 (underlying hardware may require
  - LP64 one or the other)
- can cross-develop between 32- and 64-bit OS versions
- can execute both 32- and 64-bit applications on 64-bit kernel
- can execute only 32-bit applications on 32-bit kernel

HP-UX Operating System: Specifications by Version

Attribute	HP-UX 10.01	HP-UX 10.10	HP-UX 10.20	HP-UX 10.30	HP-UX 11.00/32	HP-UX 11.00/64
Introduced	Jun 95	Feb 96	Aug 96	Aug 97	Nov 97	Nov 97
File system	4 GB	128 GB	128 GB	128 GB	128 GB	128 GB
File size	2 GB	2 GB	128 GB local,2GB network	128 GB local and network	128 GB local and network	128 GB local and network
Physical RAM	2 GB	3.75 GB	3.75 GB	3.75 GB	3.75 GB	4 TB
Shared Mem	1.75 GB	1.75 GB	2.75 GB	2.75 GB	2.75 GB	8 TB
Process data space	0.9 GB	1.9 GB	1.9 GB	1.9 GB	1.9 GB	4 TB
# File Descriptors	2,000	60,000	60,000	60,000	60,000	60,000+
# User IDs	60,000	60,000	2	2 billion	2 billion	2 billion
Threads mode	User	User	User	User and Kernel	User and Kernel	User and Kernel

HP-UX 11.0 Specifications

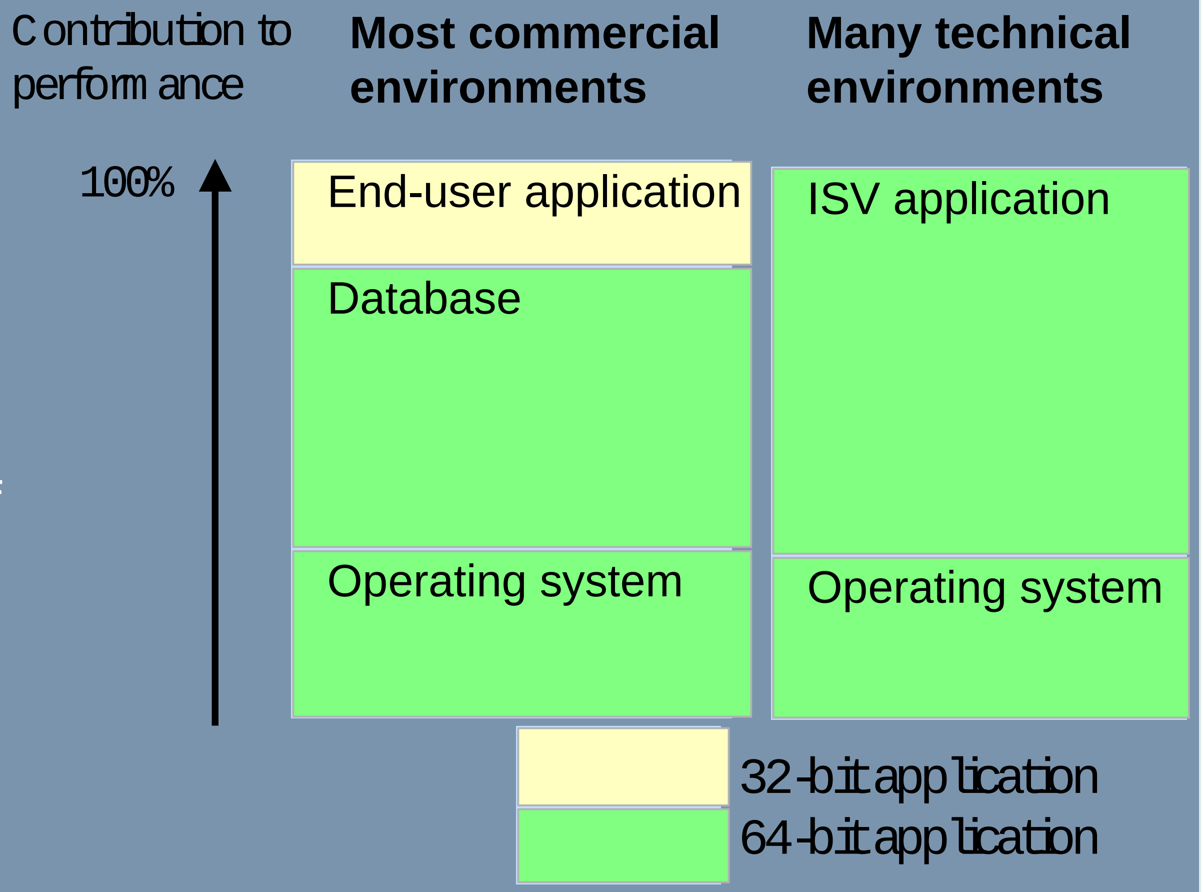
Attribute	32-bit version	64-bit version
CPUs supported	16	32
File system size	128 GB	8 TB
File size (local and networked)	128 GB	8 TB
Physical RAM	3.75 GB	4 TB
Shared memory	2.75 GB	8 TB
Process data space	1.9 GB	4 TB
File descriptors	60,000 plus	60,000 plus
User IDs	2 billion	2 billion
Threads model	User and Kernel	User and Kernel

HP-UX users may gain performance increases without needing to recompile their application

HP-UX will allow an existing 32-bit end-user application to interact with a 64-bit database

Most of the total performance gain will come from recompiles of key DB and ISV applications

Customers need not recompile their applications



Contribution to performance

100%

Most commercial environments

End-user application

Database

Operating system

Many technical environments

ISV application

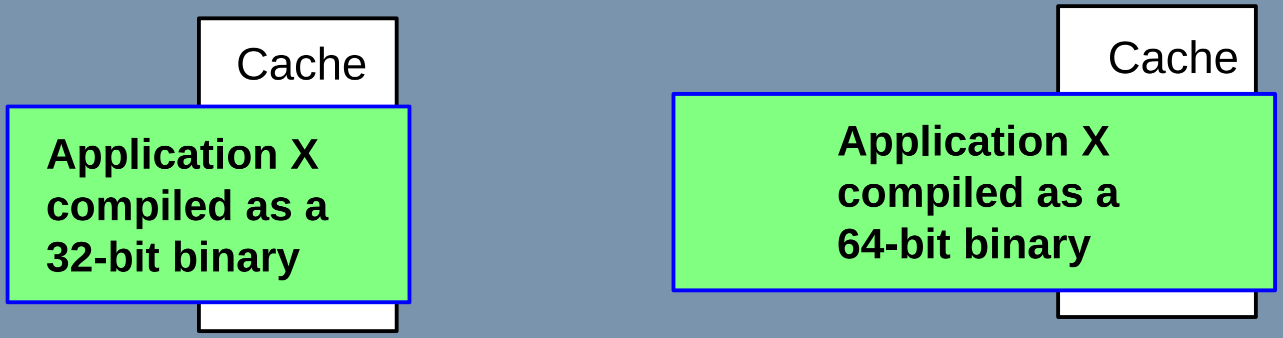
Operating system

🟨 32-bit application

🟩 64-bit application

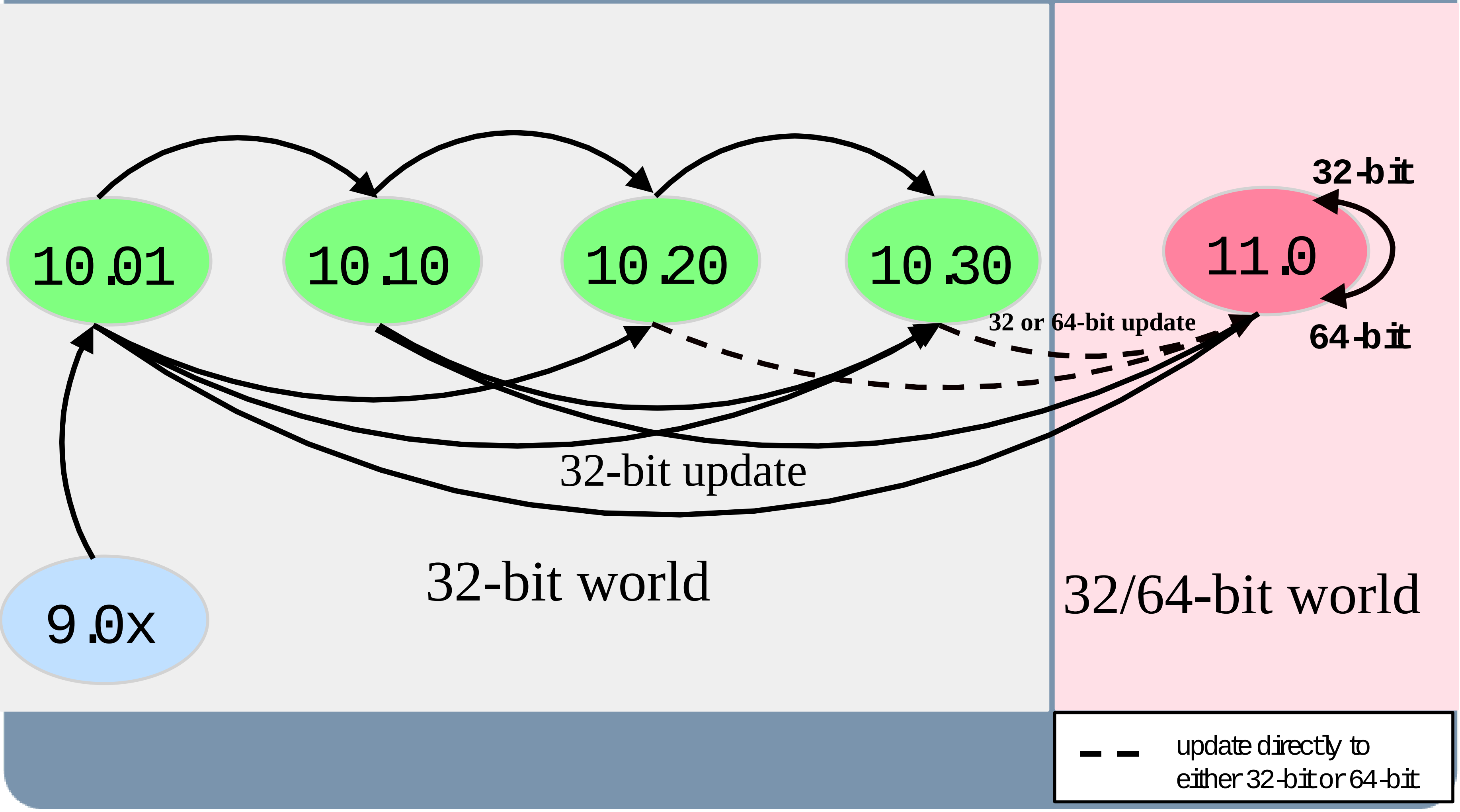
Mixing 32- and 64-bit applications

- 32-bit applications may run faster than recompiled 64-bit versions, due to "cache-fit" effect

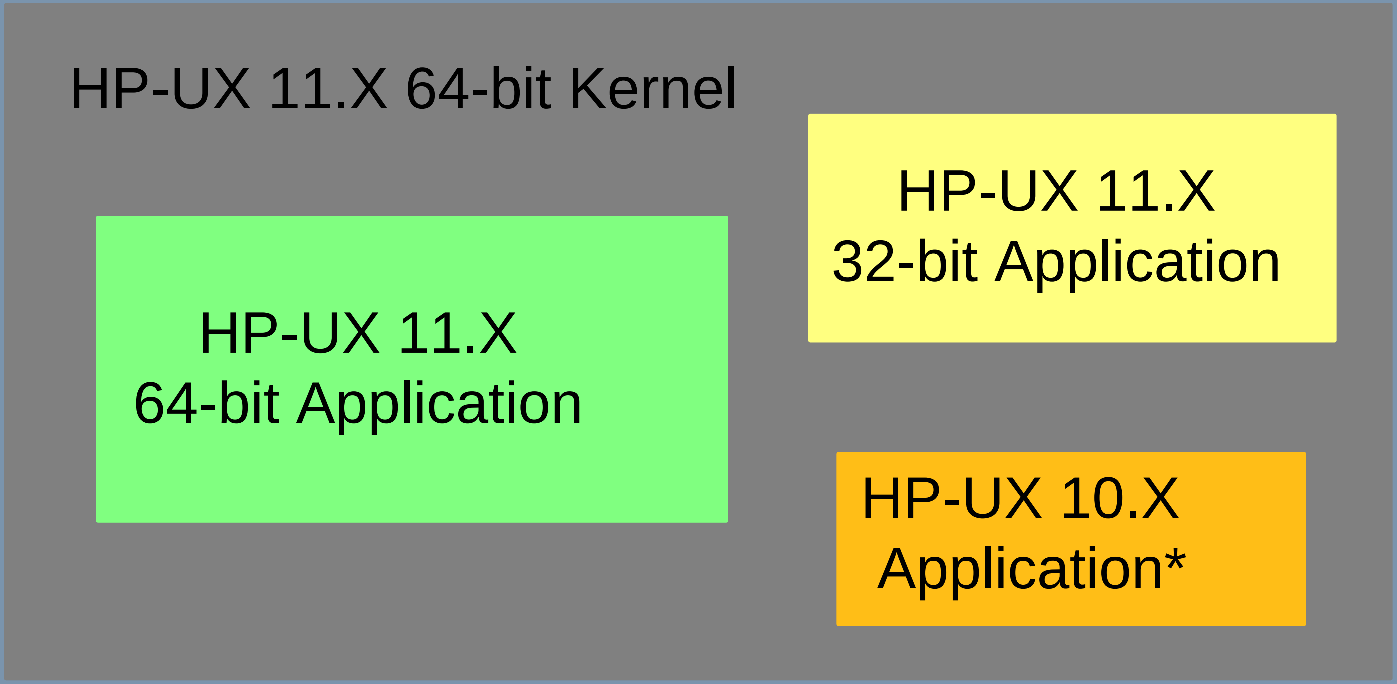

Cache
Application X compiled as a 32-bit binary
Cache
Application X compiled as a 64-bit binary

Best overall performance comes through 32- and 64-bit coexistence, which HP-UX provides

Upgrade Paths to HP-UX 11.0



Applications that run on 64-bit HP-UX 11.X



HP-UX 11.X 64-bit Kernel

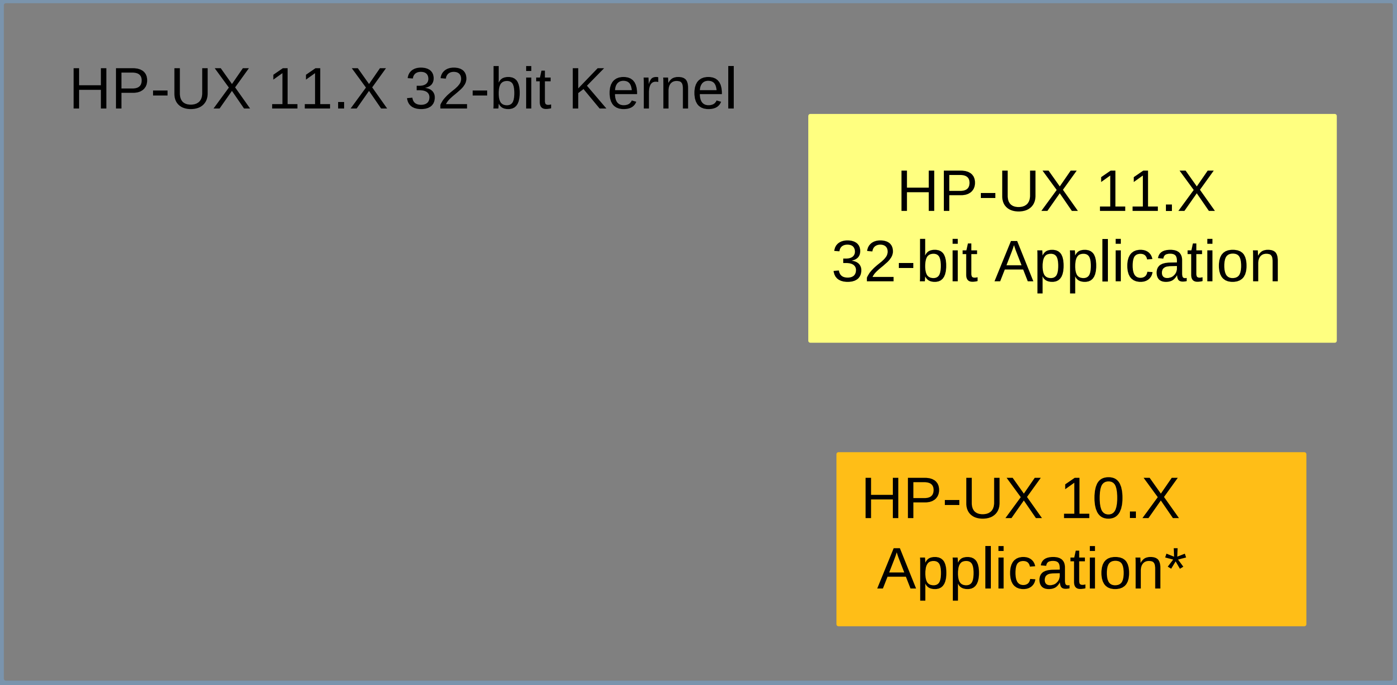
HP-UX 11.X 64-bit Application

HP-UX 11.X 32-bit Application

HP-UX 10.X Application*

✶Well-behaved applications that run on HP-UX 10.X run on HP-UX 11.X (See Compatibility Guidelines).

Applications that run on 32-bit HP-UX 11.X



HP-UX 11.X 32-bit Kernel

HP-UX 11.X 32-bit Application

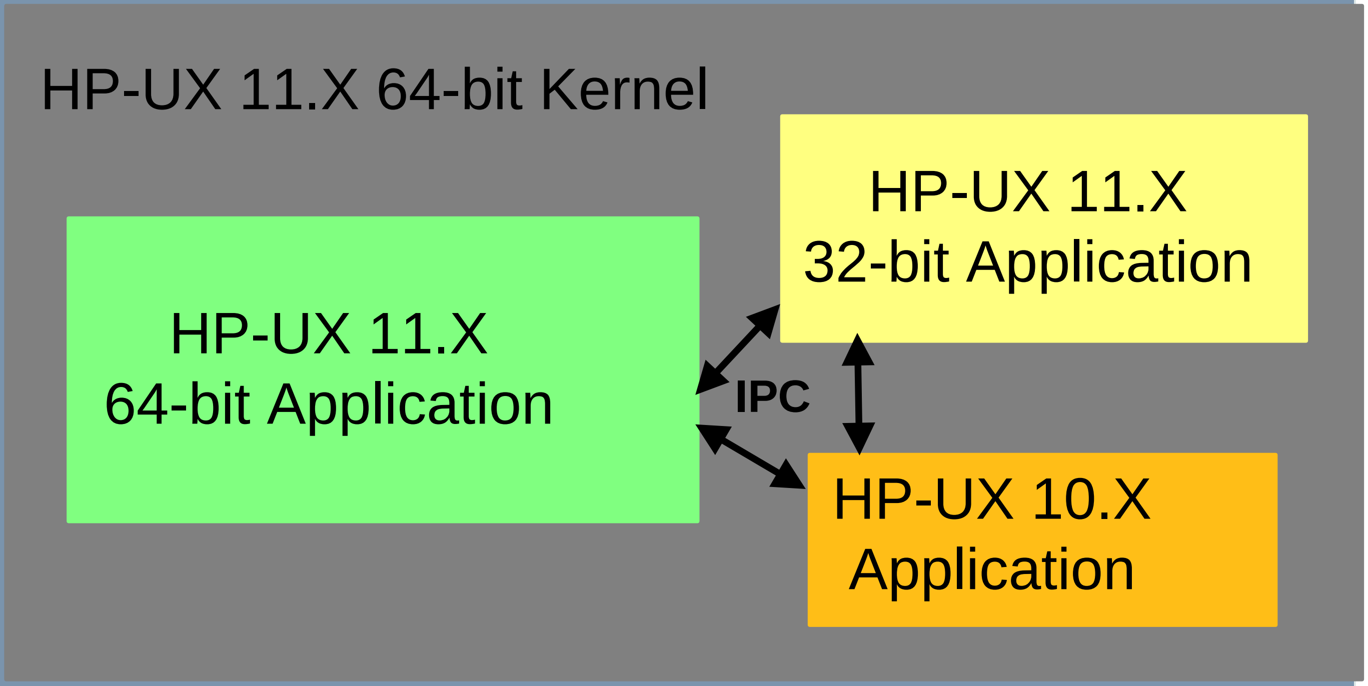
HP-UX 10.X Application*

✶Well-behaved applications that run on HP-UX 10.X run on HP-UX 11.X (See Compatibility Guidelines).

Applications Interoperability on 64-bit HP-UX 11.X

- 32-and 64-bit applications can interoperate on 64-bit HP-UX using standard IPC mechanisms:

– Shared Memory
– Mapped Files
– Sockets
– Signals
– Message Queues
– Pipes
– RPC


HP-UX 11.X 64-bit Kernel

HP-UX 11.X
64-bit Application

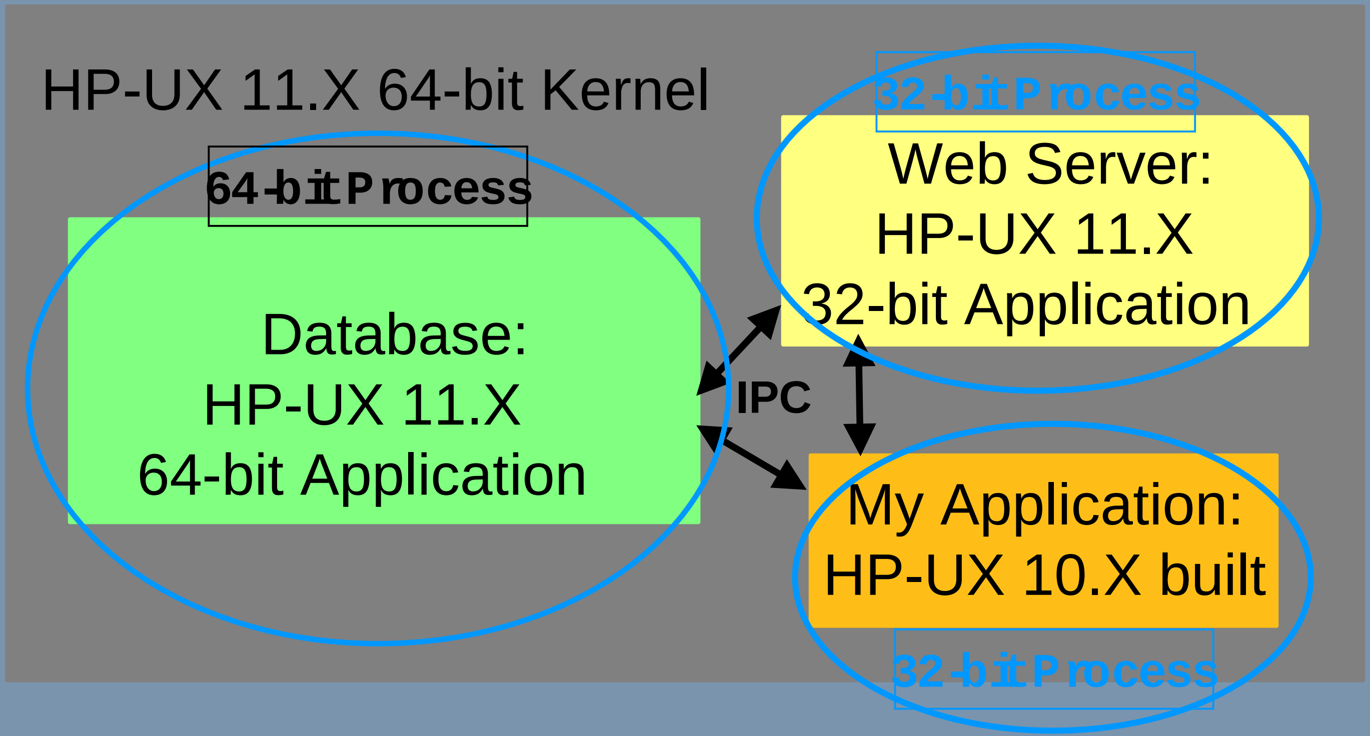
HP-UX 11.X
32-bit Application

IPC

HP-UX 10.X
Application

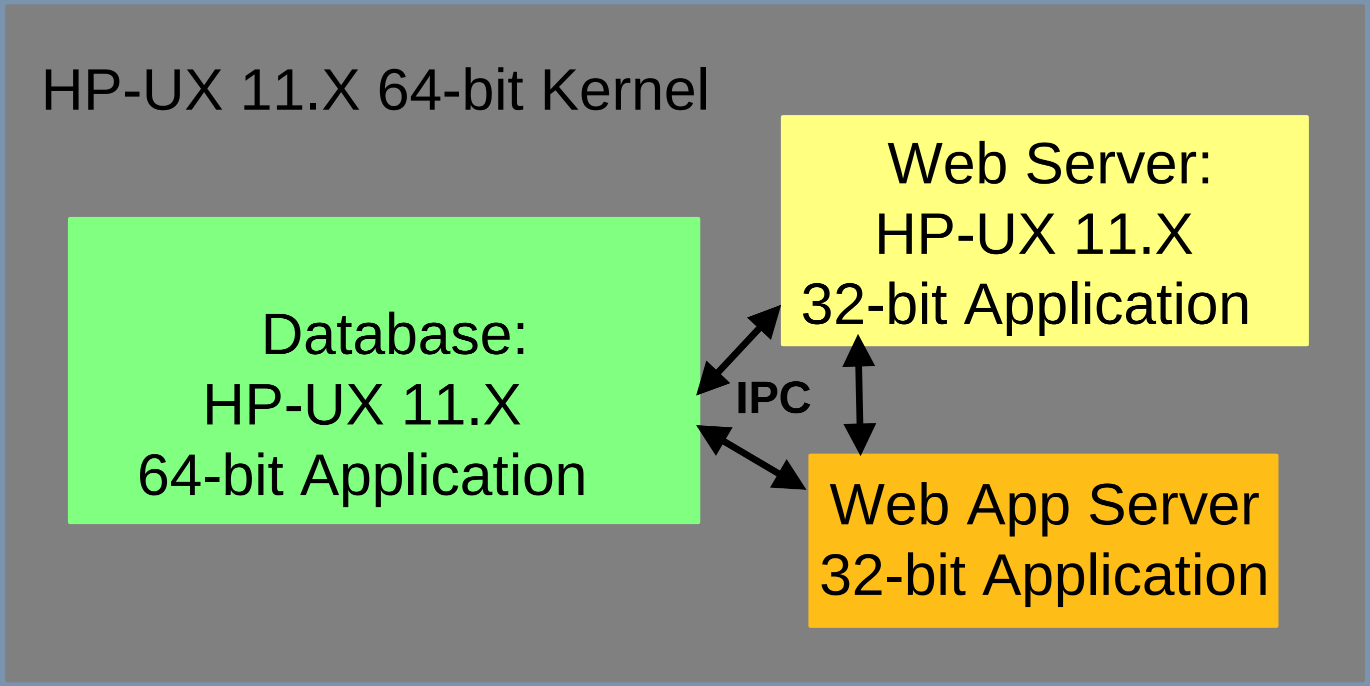
Application Interoperability on 64-bit HP-UX 11.X

- Use of 32- or 64-bit Programming Model (ILP32 or LP64) is defined at the Process (Object) boundary



Example: Web Application Server on 64-bit HP-UX 11.X

- 32-and 64-bit applications interoperate on 64-bit HP-UX using standard IPC mechanisms:



Compiler Option/Hardware Architecture Run-time Compatibility

Compiler Option	PA-RISC 1.1 32-bit Systems	PA-RISC 2.0 64-bit Systems	Code Generated
+DA1.1	X	X	PA1.1 32-bit
+DAportable	X	X	PA1.1 32-bit
+DA2.0		X	PA2.0 32-bit
+DD64 or +DA2.0W		X	PA2.0 64-bit

+DD64 is a HP C option for compiling in 64-bit mode.

+DA2.0W is the HP aC++, HP Fortran90, and HP C option for compiling in 64-bit mode.

all options are supported on both 32- and 64-bit systems, can cross-develop for either platform

HP-UX 32-bit and 64-bit Base Data Types

Data Type	ILP32 size(bits)	LP64 size(bits)
char	8	8
short	16	16
int	32	32
long	**32**	**64**
long long	64	64
pointer	**32**	**64**
float	32	32
double	64	64
long double	128	128
enum	32	32

ILP32 to LP64
Porting Concerns

- Fundamental changes:
  - longs and ints are no longer the same size
  - pointers and ints are no longer the same size
  - pointers and longs are 64 bits and are 64-bit aligned
  - Predefined types size_t and ptrdiff_t are 64-bit integral types
- Potential impact:
  - data truncation
  - data type promotion
  - constants
  - enumerated types
  - pointers
  - data alignment and data sharing

Example: code works on 32-bit and will fail on 64-bit

```
int main ()
{
  int *buffer;
  buffer = malloc(sizeof(int));      *buffer = 1234;
  printf("Buffer address: %p\n",     &buffer);
  printf("Buffer contents: %p\n",     buffer);
  printf("Dereferenced value: %d\n", *buffer);
  return 0;
}
```

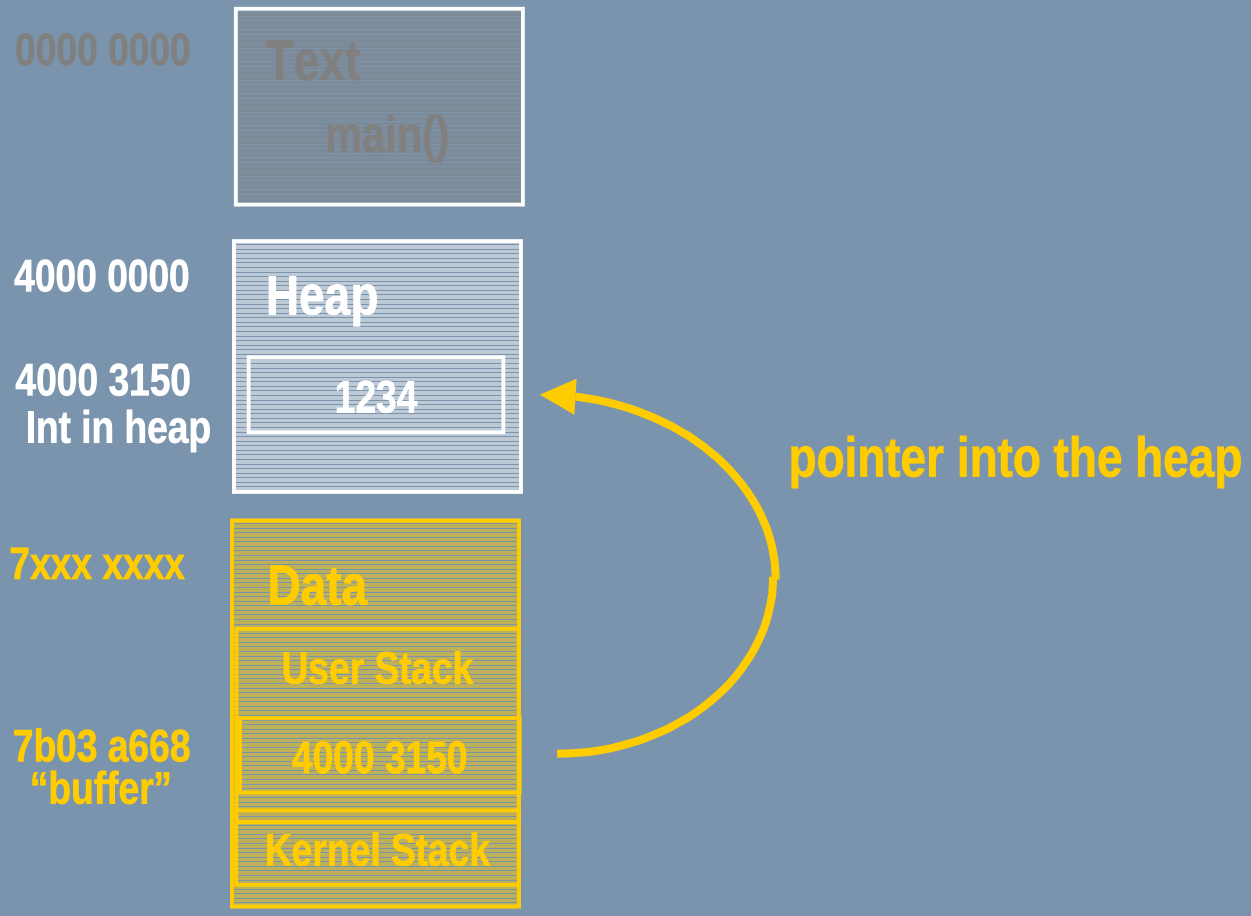
```
$ cc malloc_return.c;       ./a.out
Buffer address: 7b03a668   <-- address in data segment
Buffer contents: 40003150  <-- address in heap
Dereferenced value: 1234   <-- dereference ptr in heap
```

Run on 10.20
ILP32

```
$ cc +DD64 malloc_return.c; ./a.out
$ ./malloc_return
Memory fault(coredump)
```

Run on 11.0
LP64

Example Program - Memory Map



0000 0000

Text

main()

4000 0000

Heap

4000 3150
Int in heap

1234

pointer into the heap

7xxx xxxx

Data

User Stack

7b03 a668
"buffer"

4000 3150

Kernel Stack

Example: code works on 32-bit and will fail on 64-bit

```
int main ()
{
  int *buffer;
  buffer = malloc(sizeof(int));
  *buffer = 1234;
  printf("Buffer address: %p\n",     &buffer);
  printf("Buffer contents: %p\n",     buffer);
  printf("Dereferenced value: %d\n", *buffer);
  return 0;
}
```

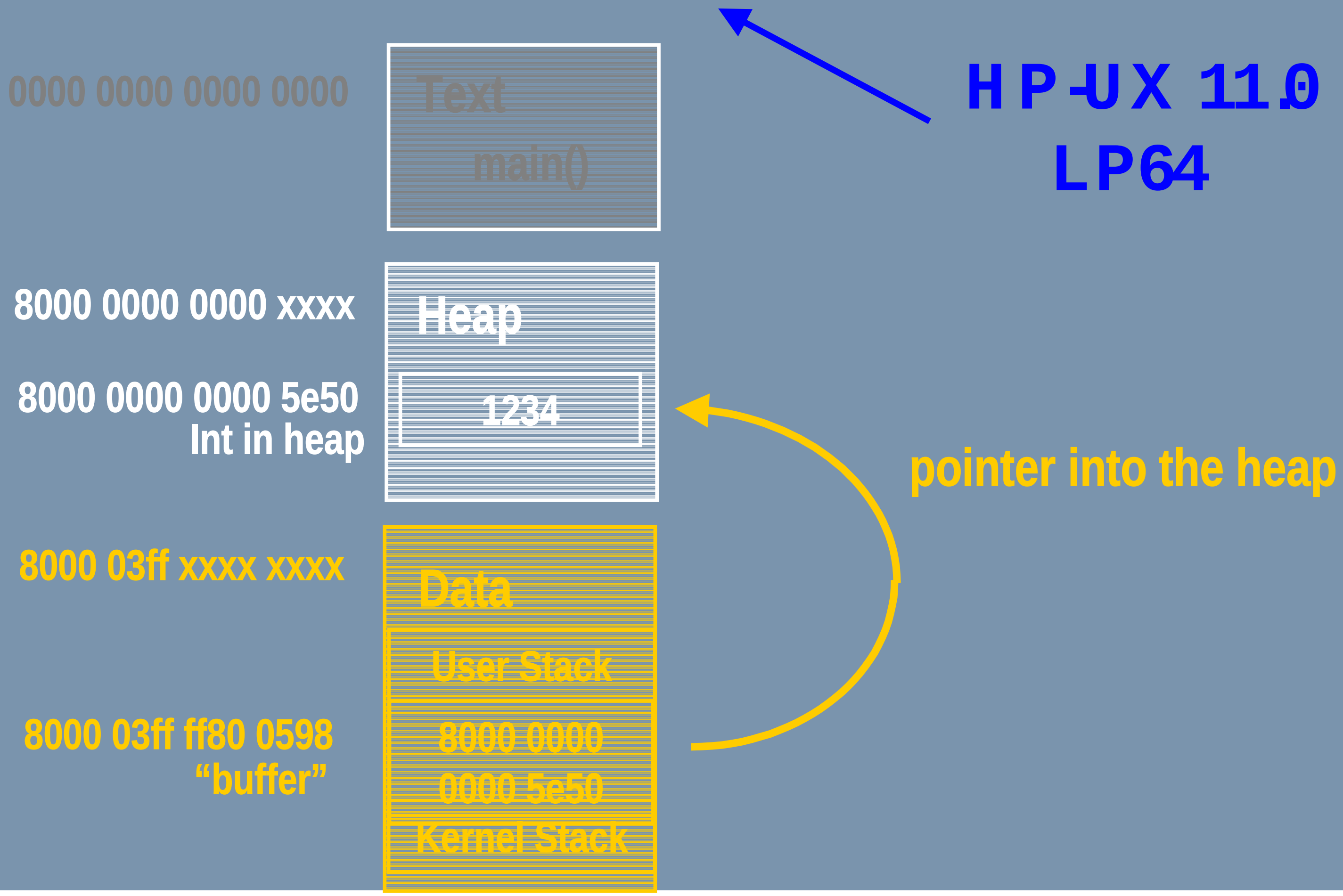
- In C an undefined function return value is integer type 32 bits
- malloc returns a pointer type which in LP64 is now 64-bits
- pointer.64 to integer.32 truncation -> invalid pointer -> core dump

```
#include <stdlib.h>
```

- Fix by including the correct function declaration from <stdlib.h>

Example Program - Memory Map

```
$ cc +DD64 -o malloc_return2 malloc_return2.c
$ ./malloc_return2
Buffer address: 800003ffff800598  <-- data segment
Buffer contents: 8000000000005e50 <-- ptr in heap
Dereferenced value: 1234          <-- dereference ptr.
```



HP-UX 11.0
LP64

0000 0000 0000 0000

Text
main()

8000 0000 0000 xxxx

Heap

8000 0000 0000 5e50
Int in heap

1234

pointer into the heap

8000 03ff xxxx xxxx

Data

User Stack

8000 03ff ff80 0598
“buffer”

8000 0000
0000 5e50

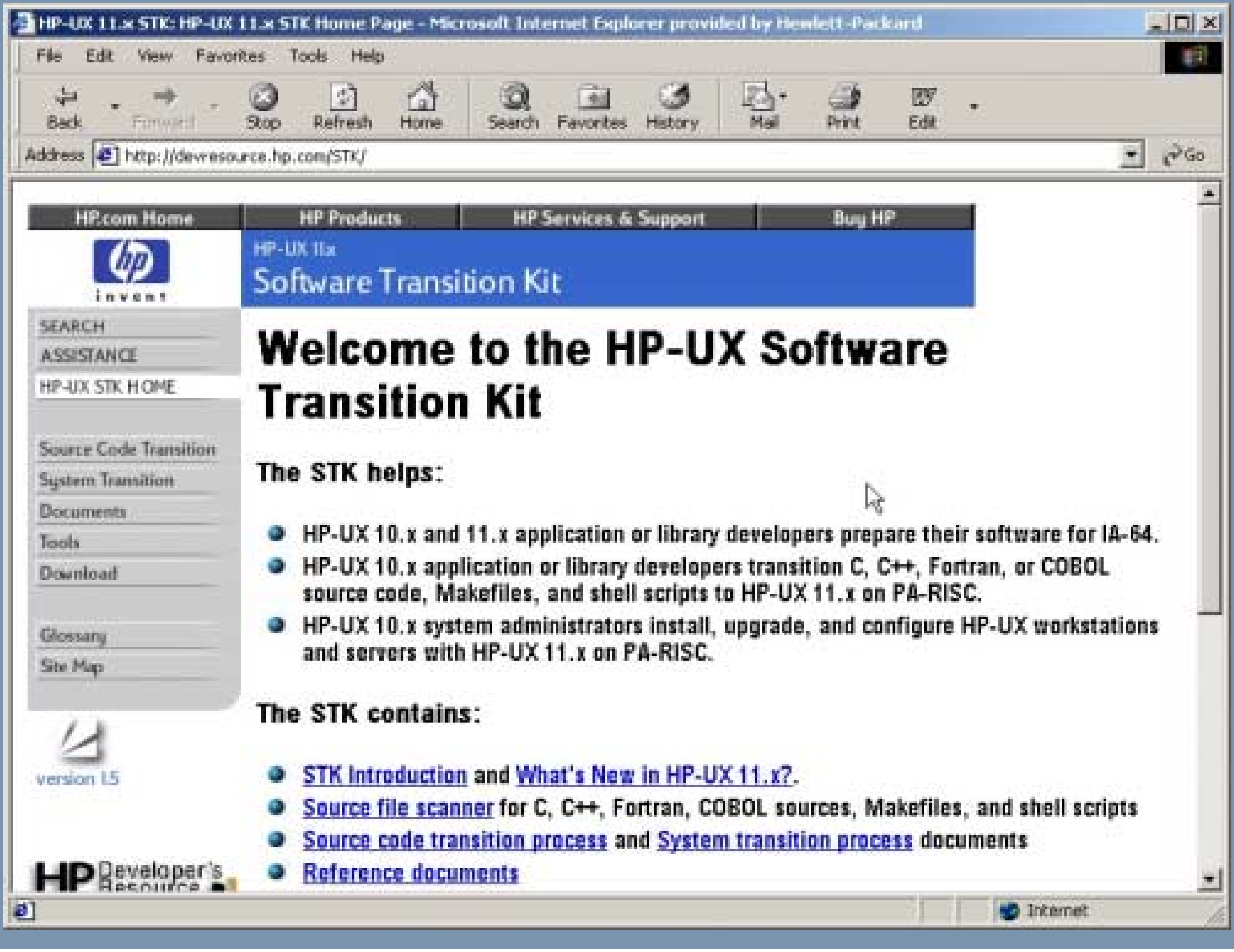
Kernel Stack

HP-UX Transition Tools

- HP-UX 11.X Software Transition Kit (STK). Current: 1.5
  - Transition to 11.X PA-RISC *OR* IA-64
  - tools and documentation to discover changed APIs
  - works on C, C++, FORTRAN, COBOL, scripts, Makefiles
  - http://devresource.hp.com/STK (new)
- HP-UX 11.X Release Notes (/usr/share/doc/*RelNotes)
  - documents system header file changes, system library changes, and lists 64-bit versions of system libraries.
- HP C compiler: both lint and the C compiler provide options to help transition to the 64-bit data model.

HP-UX 11.X STK

Download from: http://devresource.hp.com/STK (new)



Welcome to the HP-UX Software Transition Kit

The STK helps:

- HP-UX 10.x and 11.x application or library developers prepare their software for IA-64.
- HP-UX 10.x application or library developers transition C, C++, Fortran, or COBOL source code, Makefiles, and shell scripts to HP-UX 11.x on PA-RISC.
- HP-UX 10.x system administrators install, upgrade, and configure HP-UX workstations and servers with HP-UX 11.x on PA-RISC.

The STK contains:

- STK Introduction and What's New in HP-UX 11.x?.
- Source file scanner for C, C++, Fortran, COBOL sources, Makefiles, and shell scripts
- Source code transition process and System transition process documents
- Reference documents

HP-UX 11.0 STK
Running API Scanner

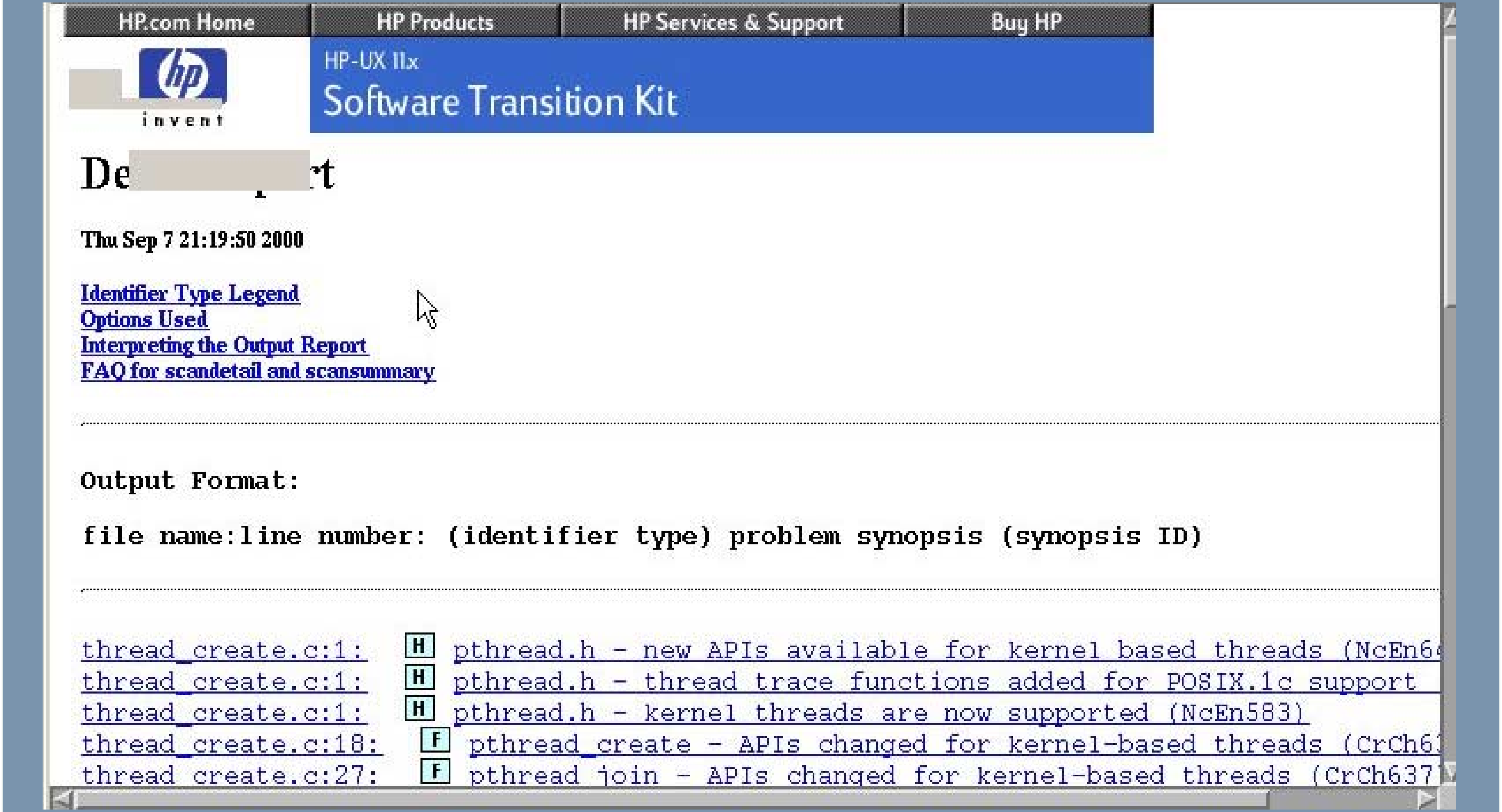
- Download and install STK (SD Depot format)
  - installs into /opt/STK and includes much documentation
  - *Tip*: unpack SD Depot and install just the documentation on PCs
- Run scandetail on source file

```
% scandetail             <testfile>.c     (default) HTML output
% scandetail -o text <testfile>.c     text output
```

- Examine file output:

```
Detail Report
Thu Sep  7 21:26:07 2000
------------------------------------------------------------------------------
Output Format:
file name:line number: (identifier type)problem synopsis (synopsis ID)
     problem description URL
------------------------------------------------------------------------------
thread_create.c:1:  (H)pthread.h - new APIs available for kernel based threads (NcEn640)
     http://devresource.hp.com/STK/impacts/i640.html
thread_create.c:1:  (H)pthread.h - thread trace functions added for POSIX.1c support (NcEn331)
     http://devresource.hp.com/STK/impacts/i331.html
        … (lots more output) …
```

HP-UX 11.0 STK
API Scanner HTML output



De[illegible]rt

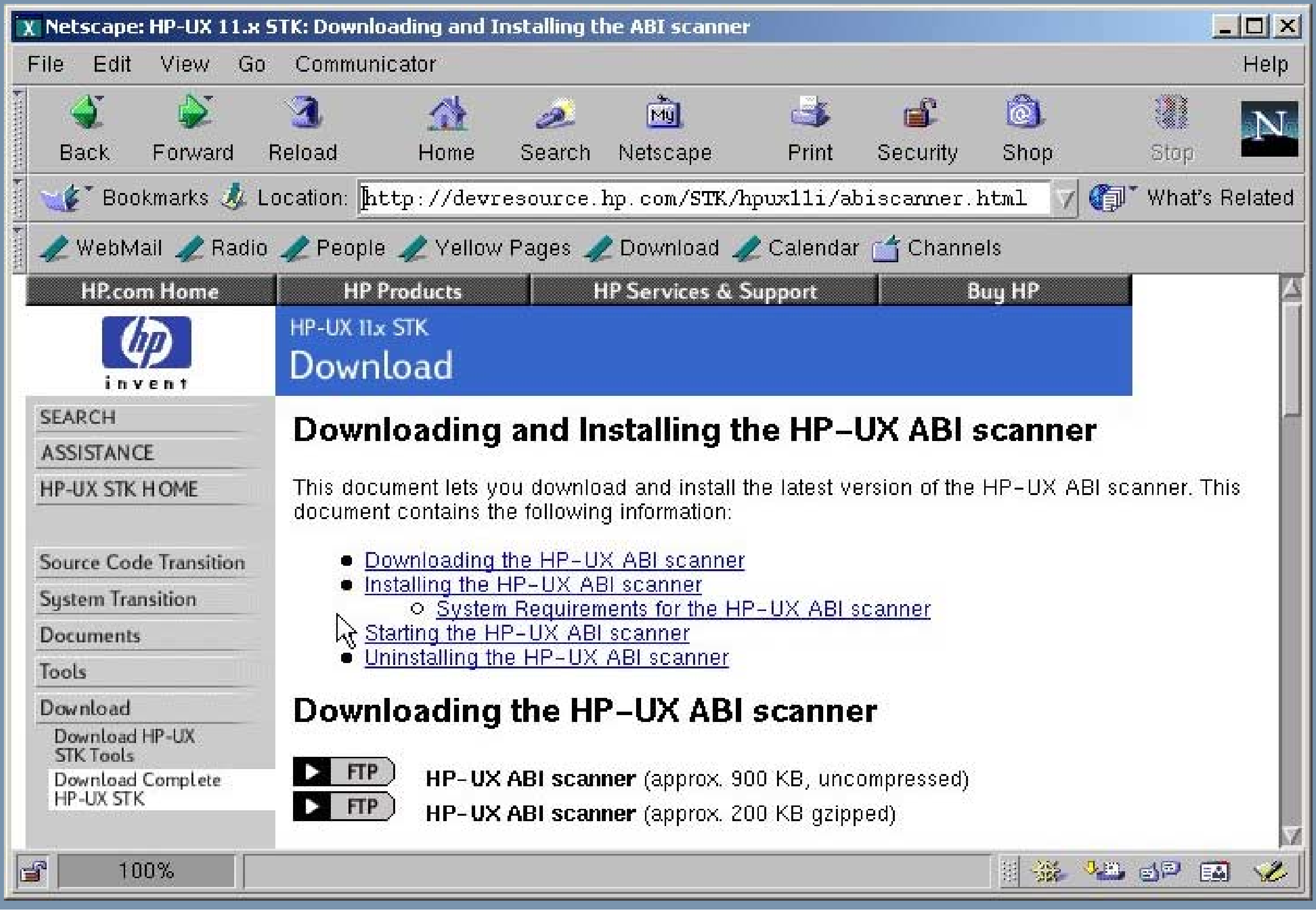
Thu Sep 7 21:19:50 2000

Identifier Type Legend
Options Used
Interpreting the Output Report
FAQ for scandetail and scansummary

```
Output Format:

file name:line number: (identifier type) problem synopsis (synopsis ID)

thread_create.c:1:   H  pthread.h - new APIs available for kernel based threads (NcEn64
thread_create.c:1:   H  pthread.h - thread trace functions added for POSIX.1c support
thread_create.c:1:   H  pthread.h - kernel threads are now supported (NcEn583)
thread_create.c:18:   F  pthread_create - APIs changed for kernel-based threads (CrCh63
thread_create.c:27:   F  pthread_join - APIs changed for kernel-based threads (CrCh637
```

HP-UX 11.0 STK
ABI Binary Scanning (new)

- Download from: http://devresource.hp.com/STK/hpux11i/
- Install ABI scanner on your test system


Netscape: HP-UX 11.x STK: Downloading and Installing the ABI scanner
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks Location: http://devresource.hp.com/STK/hpux11i/abiscanner.html
What's Related
WebMail Radio People Yellow Pages Download Calendar Channels
HP.com Home HP Products HP Services & Support Buy HP
invent
SEARCH
ASSISTANCE
HP-UX STK HOME
Source Code Transition
System Transition
Documents
Tools
Download
Download HP-UX STK Tools
Download Complete HP-UX STK
FTP
FTP
100%

HP-UX 11x STK

Download

Downloading and Installing the HP-UX ABI scanner

This document lets you download and install the latest version of the HP-UX ABI scanner. This document contains the following information:

- Downloading the HP-UX ABI scanner
- Installing the HP-UX ABI scanner
  - System Requirements for the HP-UX ABI scanner
- Starting the HP-UX ABI scanner
- Uninstalling the HP-UX ABI scanner

Downloading the HP-UX ABI scanner

HP-UX ABI scanner (approx. 900 KB, uncompressed)

HP-UX ABI scanner (approx. 200 KB gzipped)

HP-UX 11.0 STK
Running ABI Scanner

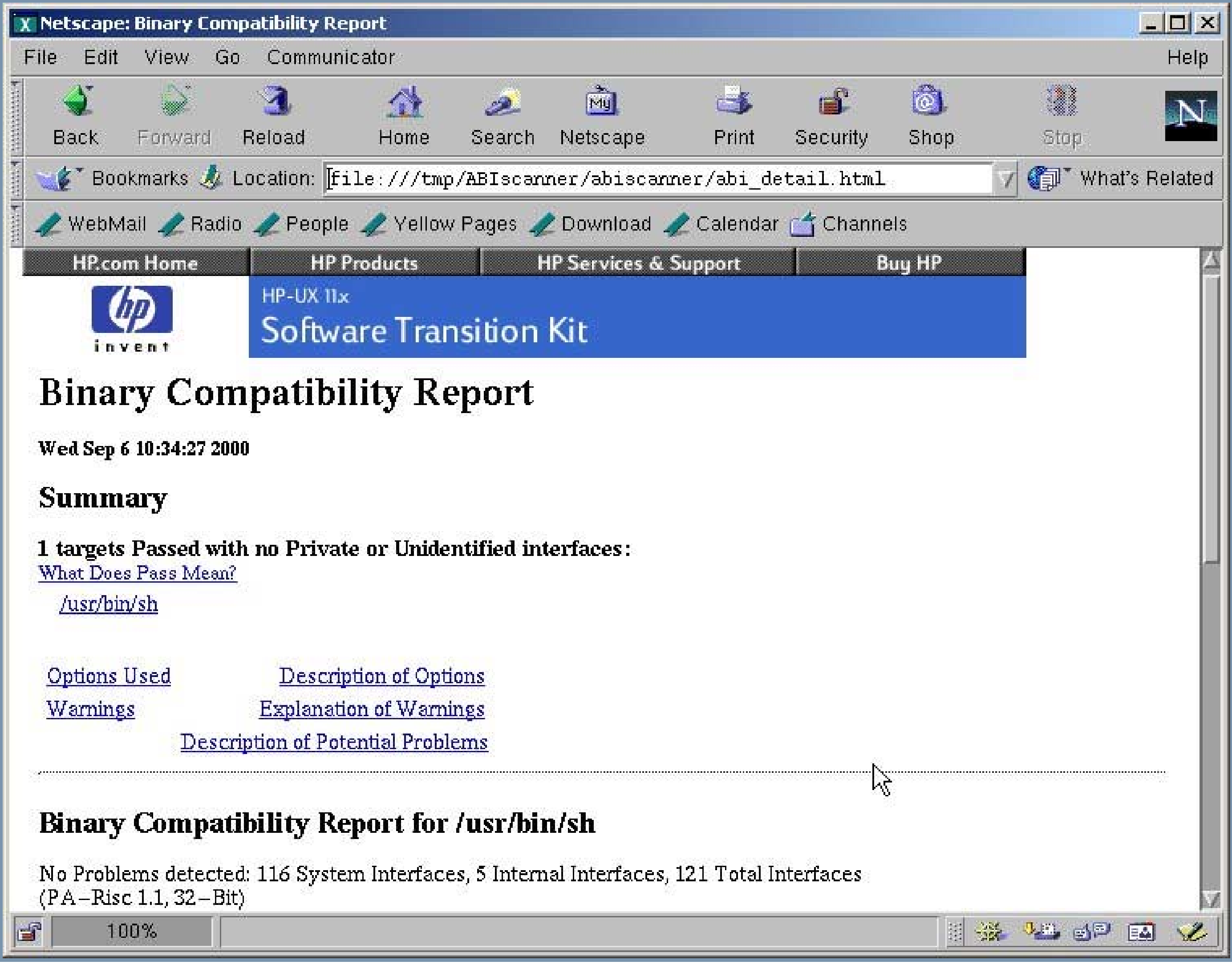
- Run abiscanner on /usr/bin/sh (POSIX shell)

  ```
  % abiscanner /usr/bin/sh
  ```

- Scans for dynamic (shared) linked library interfaces, cannot scan archive (static) linked binaries

- Man page and HTML documentation available

HP-UX 11.0 STK ABIScanner HTML output


Netscape: Binary Compatibility Report
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Bookmarks Location: file:///tmp/ABIscanner/abiscanner/abi_detail.html
What's Related
WebMail Radio People Yellow Pages Download Calendar Channels
HP.com Home HP Products HP Services & Support Buy HP
invent
HP-UX 11x
Software Transition Kit
100%

Binary Compatibility Report

Wed Sep 6 10:34:27 2000

Summary

1 targets Passed with no Private or Unidentified interfaces:

What Does Pass Mean?

/usr/bin/sh

Options Used Description of Options

Warnings Explanation of Warnings

Description of Potential Problems

Binary Compatibility Report for /usr/bin/sh

No Problems detected: 116 System Interfaces, 5 Internal Interfaces, 121 Total Interfaces (PA-Risc 1.1, 32-Bit)

HP-UX 11i

- HP-UX 11i is the HP-UX 11.11 Release
- Additional HP-UX functionality
- Inclusion of capabilities that were released via patches.
- Required for SuperDome platform
- Supported on same platforms as HP-UX 11.0

HP-UX 11i - Supported Systems

Subject to change: see http://www.unixsolutions.hp.com/products/hpux/

Series	Model	32-bit	64-bit
A-Class	A180, A180C	X	
	A400, A500		X
D-Class	Dx10, Dx20, Dx30, Dx50, Dx60	X	
	Dx70, Dx80, Dx90	X	X
E-Class	E/F/G/H/I-Class - All	X	
K-Class	Kx00, Kx10, Kx20	X	
	Kx50, Kx60, Kx70, Kx80	X	X
L-Class	L1000, L2000		X
N-Class	N4000		X
R-Class	R380, R390	X	X
T-Class	T500, T520	X	
	T600	X	X
V-Class	V2200, V2250, V2600		X
SuperDome			X
DomeLite	SD16000		X
700 Series	712, 715/[64,80,100,100XC], 725/100	X	
B-Class	B132L, B132L+, B160L, B180L	X	
	B1000		X
C-Class	C100, C110, C160L	X	
	C160, C180[XP], C200, C240, C360	X	X
	C3000		X
J-Class	J200, J210, J210XC	X	
	J280, J282, J2240	X	X
	J5000, J7000		X
J-Class	J280, J282, J2240	X	X

HP-UX 11.X Feature Set
New and Obsoleted

- For a complete reference and pointers to other documentation, please browse the HP-UX Web Site:

  http://www.enterprisecomputing.hp.com/

  http://www.unixsolutions.hp.com/

- Other relevant web sites:

  http://docs.hp.com (http://www.docs.hp.com)

  http://software.hp.com (http://www.software.hp.com)

- HP-UX Release Notes in distributed in /usr/share/doc/
  - plain text and HTML versions

- For HP-UX 11i most whitepapers are no longer in /usr/share/doc but are available on Web

HP-UX 11.0:
Major New Functionality

- 64-bit Kernel and Objects

  PLUS

- Kernel threads
- Performance optimized page size
- Dynamically loadable kernel modules infrastructure
- New system recovery capability
- Additional new technologies

HP-UX 11i
Major New Functionality

- Support of SuperDome systems
- HP-UX Partitions (on supported hardware)
- OLAR I/O (on supported hardware)
- Instant Capacity on Demand (iCOD)
- Performance improvements and size extensions
- Security enhancements
- Networking enhancements

Binary Compatibility for well-developed applications from 11.0 to 11i

HP-UX 11i
Practical Details

- HP-UX 11i Release now distributed on 2 Installation CDs
  - Only CD #1 is bootable (has LIF partition and Intermediate Loader)
  - OS installation packages are contained across both CDs
- Still have the choice of Install or Upgrade
  - Install: disks/filesystems reinitialized, complete new bits laid down
  - Upgrade: system files laid down, configuration information retained
- swgettools has been discontinued - use update-ux
- Different CD Sets contain different Operating Environments

  (more about this later)
- Installation Screens about the same as HP-UX 11.0
- Sizes for some partitions have increased

HP-UX 11i
Operating Environment Bundles

- The HP-UX 11i release is disatributed in one of the following Operating Environment (OE) software bundles:
  - HP-UX 11i Operating Environment (Base)
  - HP-UX 11i Enterprise OE
  - HP-UX 11i Mission Critical OE
  - HP-UX 11i Technical Computing OE
- Only one Operating Environment can be installed and run
- Operating Environments consist of the core Operating System plus collections of selected applications that make up the OE Bundle
- Contents of OE Bundles are listed in HP-UX 11i Release Notes

  (/usr/share/docs/11iRelNotes.[txt|html])

HP-UX 11i (Base) Operating Environment Bundle

- Apache Web Server
- CIFS/9000 Client and CIFS/9000 Server
- Event Monitoring Service (EMS)
- HP-UX Java 2 (JRE) Platform
- HP-UX Support Tools (Diagnostics)
- instant Capacity On Demand (iCOD)
- Java Plug-In (JPI)
- Netscape Communicator
- Pluggable Authentication Modules Kerberos
- Service Control Manager (SCM)
- Netscape Directory Server (selectable)
- Network Drivers (selectable)
- WebQoSPeak (selectable)

HP-UX 11i
Operating Environments

Package	HP-UX 11i	Enterprise	Mission Critical	Technical Computing
Enterprise Cluster ECM	NO	NO	YES	NO
FirstSpace VRML Viewer	NO	NO	NO	YES
GlancePlus Pak	NO	YES	YES	NO
High Availability Monitors	NO	YES	YES	NO
HP 3D Technology for Java	NO	NO	NO	YES
HP MLIB Math Library	NO	NO	NO	YES
HP MPI Message Parsing Interface	NO	NO	NO	YES
HP Visualize Conference	NO	NO	NO	YES
HP-UX Workload Manager	NO	NO	YES	NO
iCOD	YES	YES	YES	NO
MirrorDisk/UX	NO	YES	YES	NO
MC/ServiceGuard	NO	NO	YES	NO
MC/SG NFS Toolkit	NO	NO	YES	NO
OnLine JFS 3.3	NO	YES	YES	NO
PRM	NO	YES	YES	NO
SCM Service Control Manager	YES	YES	YES	NO

HP-UX 11i
Additional Practical Details

- Default kernel drivers may increase kernel size
    - Gigabit Ethernet, EMS, others
    - may want to remove some of these for smaller systems
- Additional kernel processes started
- Additional User processes
    - samd, others
    - defaults changed for other processes
        - nfsd/biod default to 16 processes

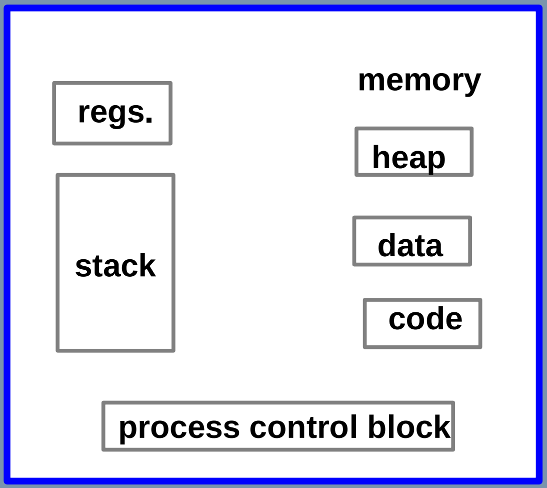
HP-UX Threads in the 10.X and 11.X Releases

- All Thread APIs conform to POSIX standards (Pthread standard)
  - P1003.1c Portable API for Threads Extensions to POSIX 1003.1(a,b)
- Pthreads can be implemented with Mx1, 1x1, or MxN models
- HP-UX 9.X and 10.X offered User Space threads
  - HP-UX User Space (DCE) threads comply with POSIX Draft 4.
  - Threads libraries, include files, and documentation delivered with the core DCE product, based on OSF DCE threads, contained in libcma
- HP-UX Kernel threads delivered starting with 11.0 (10.30)
  - comply with POSIX Draft 10.

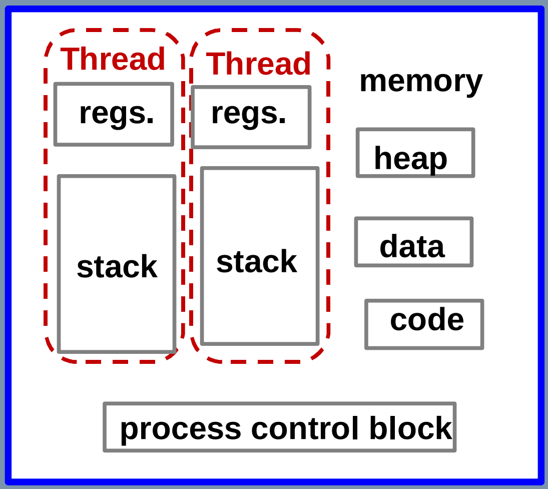
Thread Definition

- A single sequential flow of control that can coexist with other threads in the same process.
- A thread is described by:
  - a unique identifier - Thread ID (TID)
  - shared address space with other threads in process
  - scheduling priority and policy

Traditional Process



Multi-threaded Process



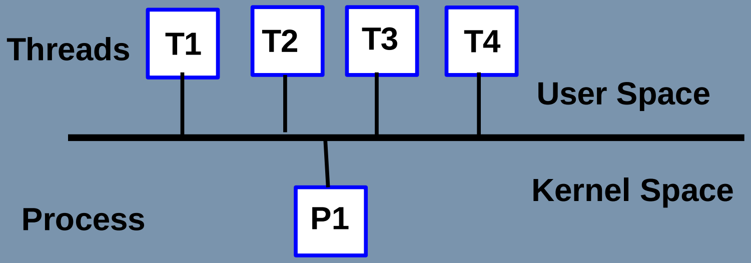
User Threads and Kernel Threads

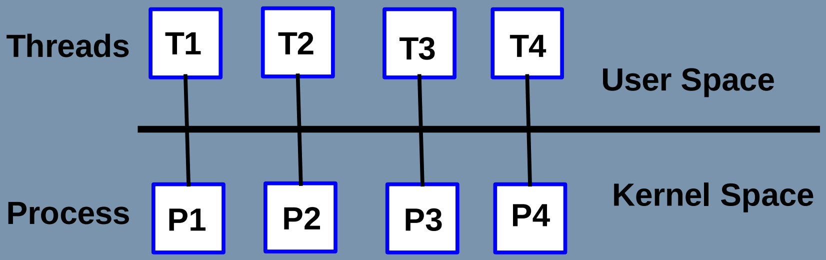
User Threads

- handled in User Space and controlled using the threads APIs provided in the threads libraries
- Mx1: Many to One Model. M Threads exist inside 1 Process that is visible to the Kernel

Kernel Threads

- handled in User and Kernel space and are kernel schedulable entities visible to the operating system
- 1x1: One to One Model. one Thread in one Kernel process. HP-UX implements with LWP (Light Weight Process).





User Threads vs. Kernel Threads

Mx1 User Threads

- all threads mapped into one process
- not visible to kernel
- fast creation/control
- no kernel overhead
- if one thread blocks, all threads are blocked

1x1 Kernel Threads

- each user thread mapped to kernel thread
- each thread scheduled independently, so if one blocks others can still run
- some overhead due to kernel calls

Threads - programming with Concurrency and Parallelism

- Concurrency - multiple threads may be in progress at the same time

- Parallelism - multiple threads are executing at the same time
  - On a MP system, Kernel Threads may execute in parallel on different CPUs
  - potentially a HUGE performance win if multiple processors active

- Demonstrate multi-Processor scaling and concurrency issues with a multi-threaded: 'Hello, World!", then will become a form of compute-bound application

Programming with Threads program pseudo-code

```
/*
 *  Create and execute 2 threads (pseudo-code)
 */

main()
{
  pthread_create(thread1);          /* create first  thread   */
  pthread_create(thread2);          /* create second thread   */

  pthread_join(thread1);            /* wait for first  thread */
  pthread_join(thread2);            /* wait for second thread */
}

thread1()                           /* Thread 1 will run here */
{
  printf("Hello, World!\n");
}

thread2()                           /* Thread 2 will run here */
{
  printf("Goodbye, World!\n");
}
```

```
% cc  +DA1.1  -lpthread  -o 2threads  2threads.c
```

Compile and Run

```
% ./2threads
Hello, World!
Goodbye, World!
```

Programming with Threads - order of execution

- run the program

```
% ./2threads
Hello, World!
Goodbye, World!
```

... OR MAYBE ...

```
% ./2threads
Goodbye, World!
Hello, World!
```

- Is the order of execution guaranteed?

It is not guaranteed unless one uses the thread scheduling and synchronization APIs

Programming with Threads - compute-bound example

- Each thread will now execute a basic compute-bound application
- Threads will execute concurrently as the kernel schedules each thread
- Threads may execute in parallel on a multi-processor system
- Default thread scheduling policy on HP-UX is to schedule threads across different processors

```
void busywork(int maxcount)
{

  int count1, count2;
  for (count1=0; count1<maxcount; count1++)
    for (count2=0; count2<maxcount; count2++);
  pthread_exit( (void *)  NULL);
}
```

busywork: loop for (count*count) times

Programming with Threads - compute-bound execution

- Compile and run as before
- execute the program on 1-way C-110:

```
% timex 2busythreads
real        17.10
user        16.79
sys          0.03
```

- execute the program on 2-way D-270:

```
% timex 2busythreads
real         5.06
user        10.05
sys          0.01
```

Real (wallclock) time

Compute time 2X real time!

Programming with Threads - compute-bound example

- Realtime (wallclock time) is halved on the 2-way system
  - Each thread executes in parallel on a different processor

- This program MP-scales perfectly to a 2-way system, but would not scale any better on a system with more than 2 processors.

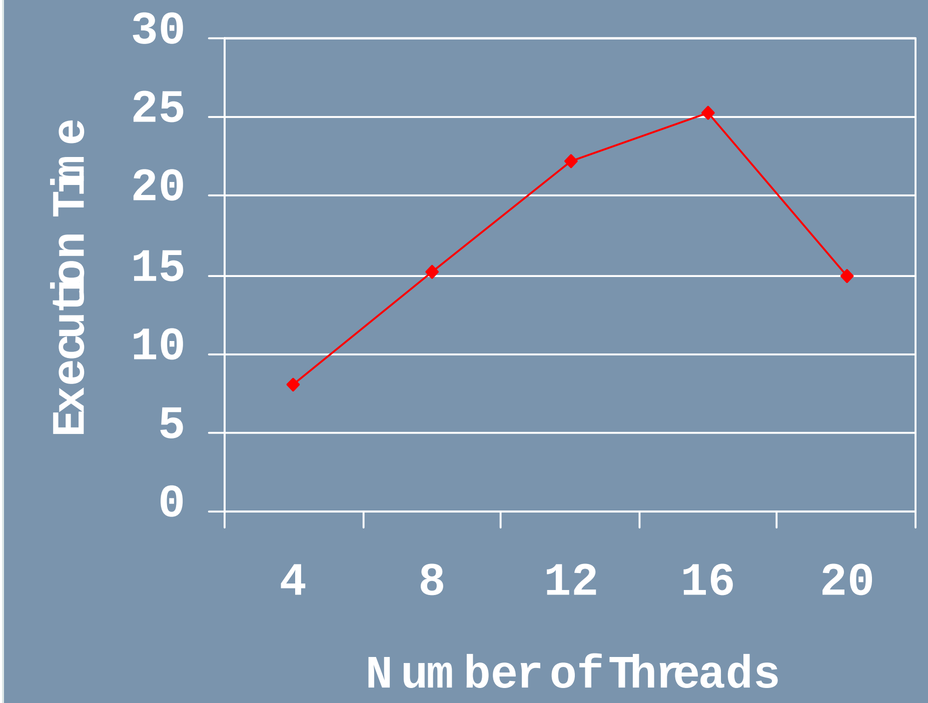
- This simple program example is not concerned with synchronization, global resources and contention, asynchronous events - many of the interesting elements of real-world programming. Spinlock contention may be a serious problem for global shared resources in threaded applications.

Threads - Performance Improvements

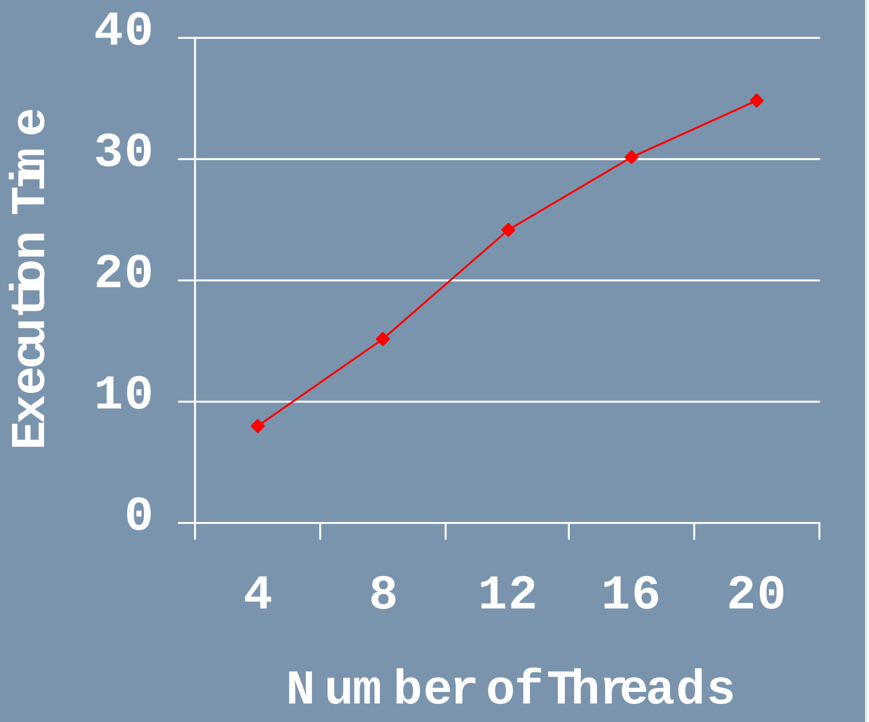
- Significant performance improvements have been made to the run-time libraries
  - install the latest libc patches (delivered with HP-UX 11i)
- Spinlock contention for malloc(3c) improved with new libc
- Use new environment variables to tune malloc for multi-threaded programs:
  - **_M_ARENA_OPTS** tune number of malloc arenas
  - **_M_SBA_OPTS** tune malloc small block allocator
- See programming examples in the HP-UX 11i Release Notes

Performance Improvements for multi-threaded web application

Untuned malloc: thread spinlock thrashing



Tuned malloc: no spinlock contention



Extending Thread Capability

- MxN Threads will be supported in a future release of HP-UX 11i

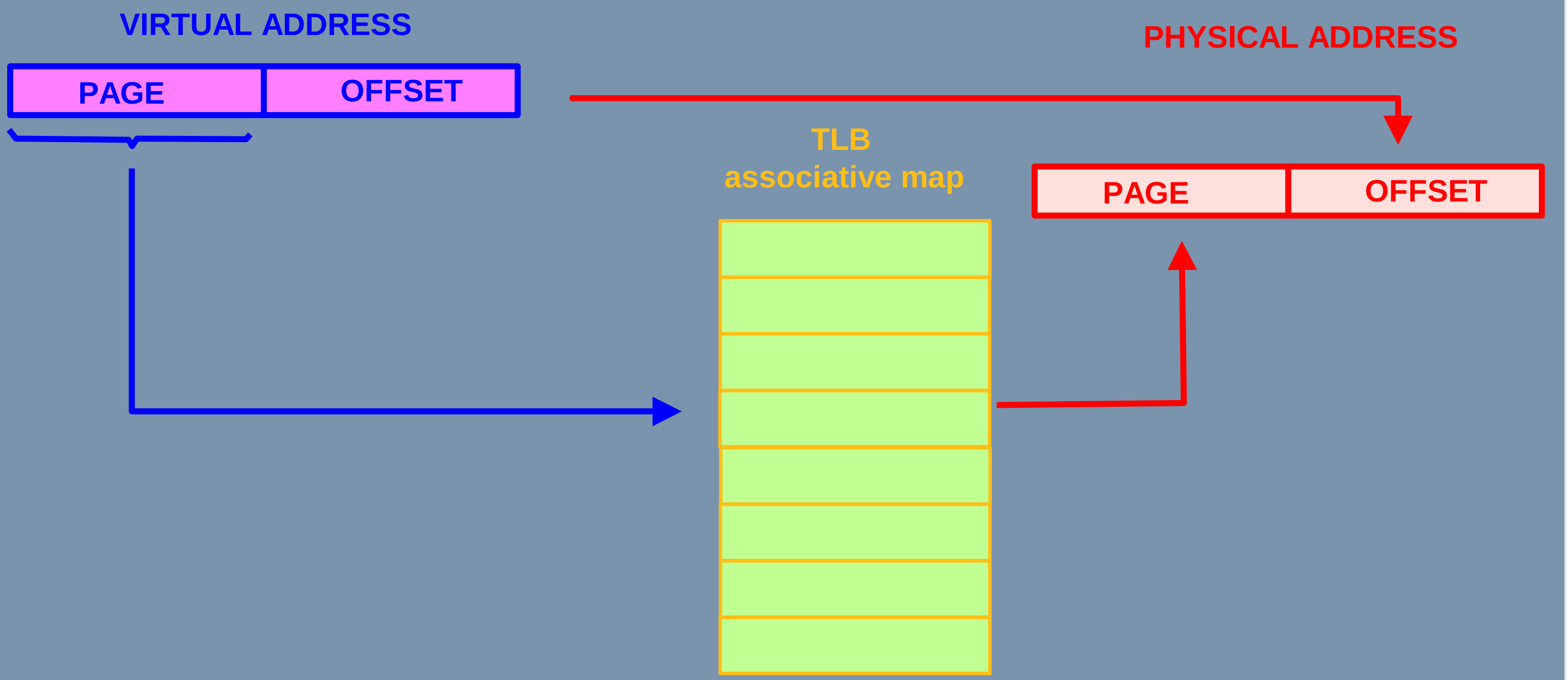
Large Virtual Memory Page Size

- Virtual Memory Page Size was fixed at 4 KBytes for all HP-UX releases prior to 10.20
- HP-UX 10.20 introduced Large Pages - VM page size for an executable process could be explicitly set using the chatr command to:

  4K
  16K
  64K
  256K
  1M
  4M
  16M
  64M
  256M
  L (use largest available size)
  D (use kernel default size)

- Page size could be set for Text, Data, Stack, Memory Mapped Files, and other memory objects.
- Only for PA-8000 processors

Virtual Memory Page Lookups and Translation Lookaside Buffer (TLB)



TLB is a hardware lookup (extremely fast) of virtual page addresses. If page entry is not in TLB, a software search (slower) of all virtual page addresses must be done by the kernel to get the physical page address.

Performance Optimized Page Size

- HP-UX 11.0 introduces variable page size (aka. Large Pages or Performance Optimized Page Size).
- The SysAdmin can request a page size using the chatr() command or the kernel can specify a page size.
- The kernel tries to honor the request but may use a smaller page size if there is competition for memory.
- Kernel tunable parameters to control operation:
  - vps_pagesize default page size used by kernel
  - vps_ceiling maximum page size used by kernel
  - vps_chatr_ceiling maximum size a user can set

(probably don't want or need to change these tunables)

Advantages to Performance Optimized Page Size

- Larger virtual address ranges can be mapped using fewer TLB entries, so there will be fewer TLB misses

- POPS will offer performance advantages for applications that:
  - are experiencing significant TLB misses
  - have large Reference Sets (e.g. large Data Segments or Text Segments)

Performance Optimized Page Size Tools and Documentation

- chatr command (see chatr(1) man page)

chatr +pi <size>	text page size
chatr +pd <size>	data page size

size = 4K, 16K, 64K, 256K, 1M, 4M, 16M, 64M, 256M, L (largest), D (kernel default)

- kernel tunable parameters:

vps_pagesize	default page size kernel will use
vps_ceiling	max page size kernel will select
vps_chatr_ceiling	max page size a user can select

- /usr/contrib/bin/vps_stats
  - report page statistics

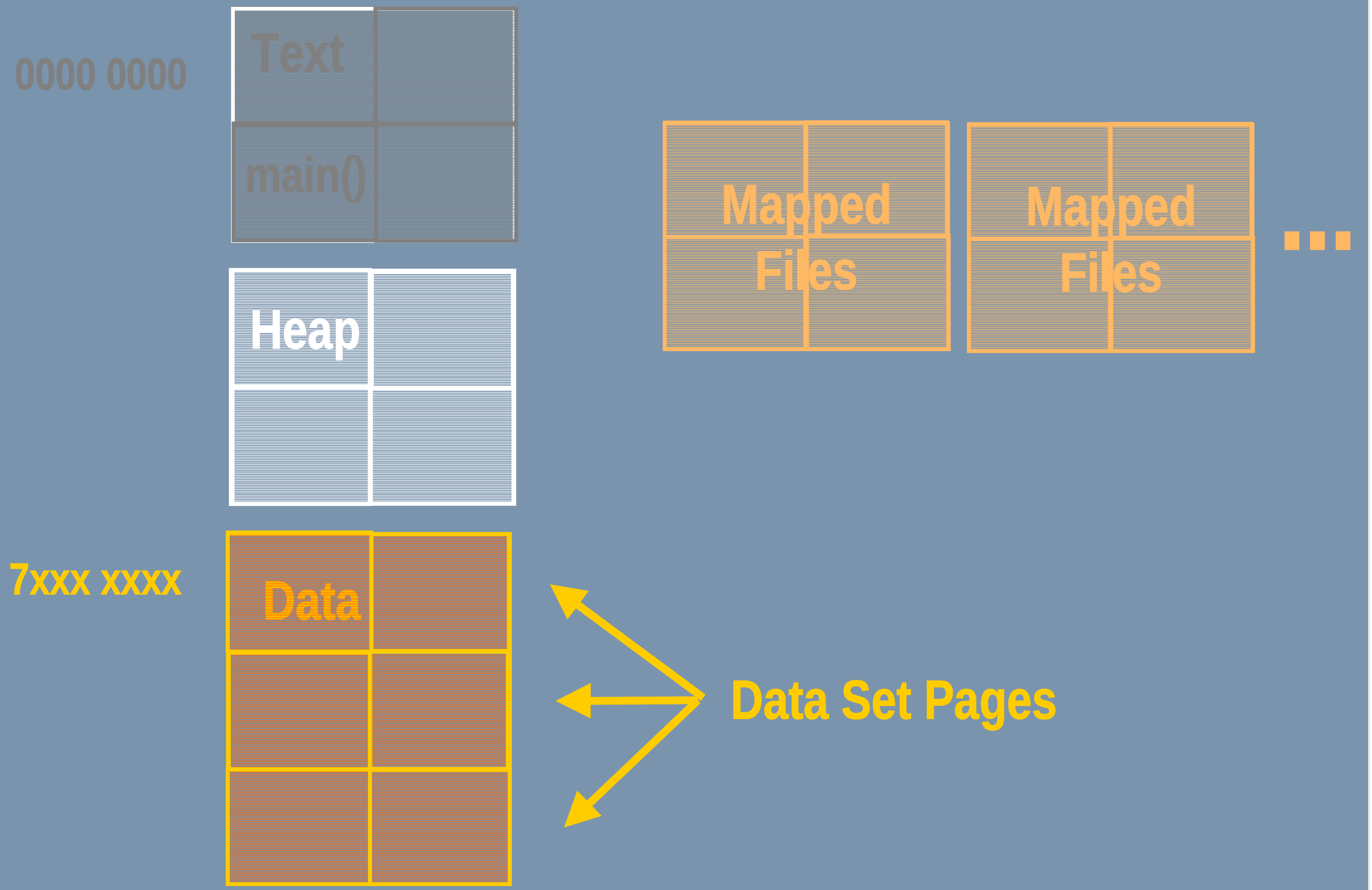
Performance Optimized Page Size Example Programs

```
/*
 * Allocate/touch large data space - good locality
 */

#define SIZE 256
int big_array [SIZE] [SIZE] [SIZE];

main()
{
  int i,j,k;       /* array indices */

  for ( i=0; i<SIZE; i++ )
    for ( j=0; j<SIZE; j++ )
      for ( k=0; k<SIZE; k++ )
        big_array[i][j][k] = k;
}
```

65 MB Data Segment
16 Million 4-byte ints

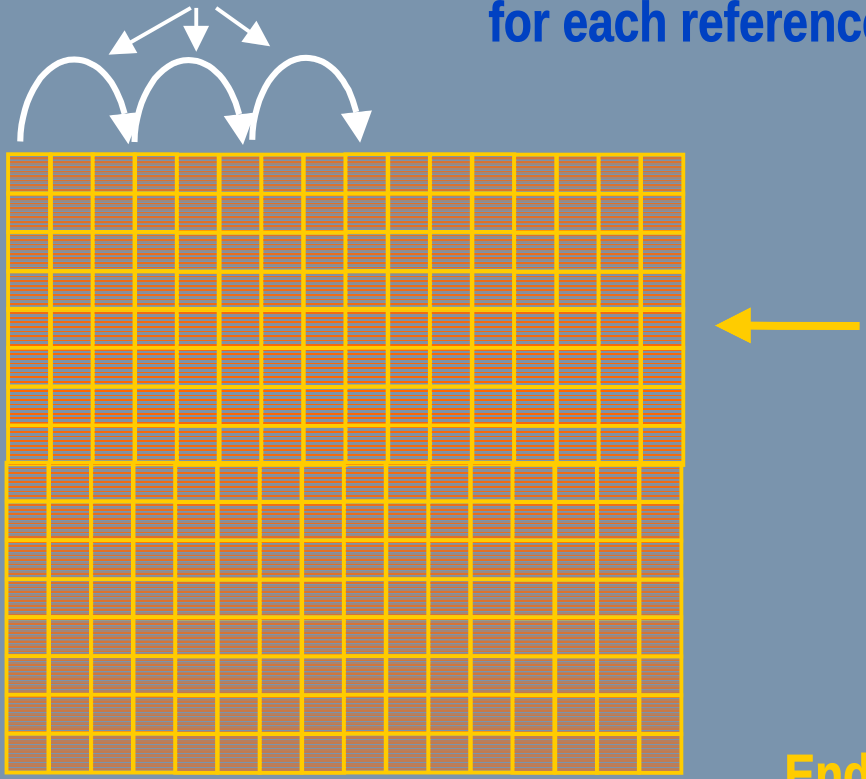
Touch each element
16 Million iterations

Memory Map - Virtual Memory Pages



Example Program - Good Locality

GOOD Data Locality: 4 byte address jumps for each reference



Start Address
big_array(0,0,0)

65 MB Data Segment
16 Million 4-byte ints
(65 MB/pagesize) Pages

End Address
big_array(256,256,256)

Example Program - Poor Locality

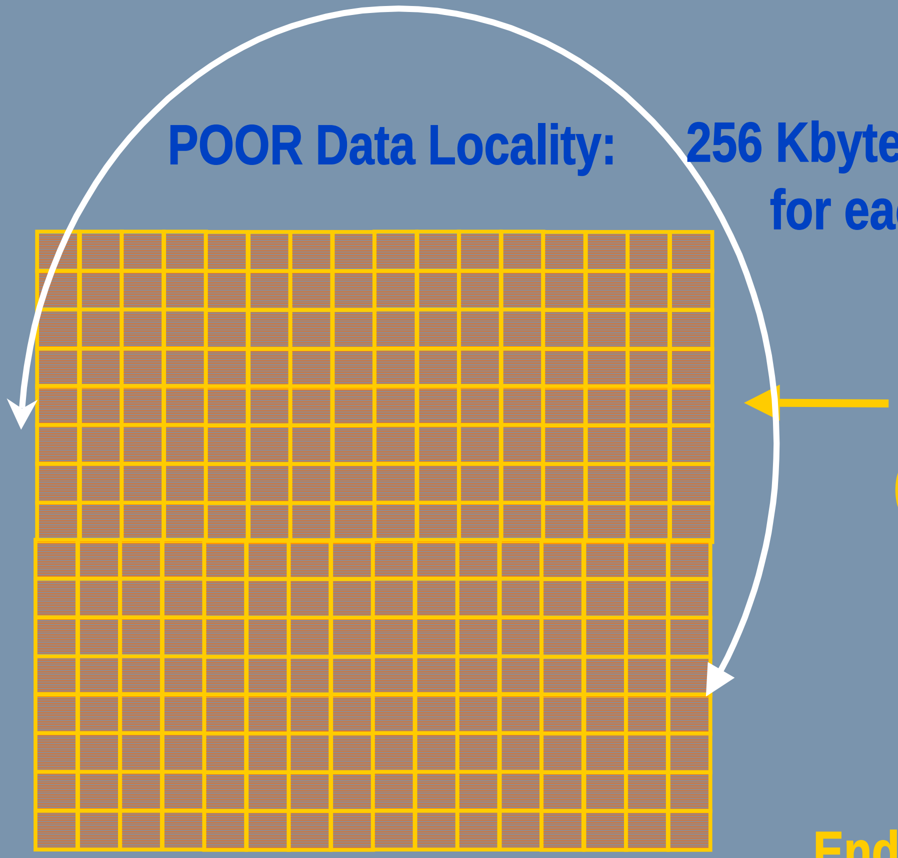
```
/*
 * Allocate/touch large data space - poor locality
 */

#define SIZE 256
int big_array [SIZE] [SIZE] [SIZE];

main()
{
  int i,j,k;      /* array indices */

  for ( i=0; i<SIZE; i++ )
    for ( j=0; j<SIZE; j++ )
      for ( k=0; k<SIZE; k++ )
        big_array[k][j][i] = k;
}
```

65 MB Data Segment
16 Million 4-byte ints

Poor Locality -
256K address jumps
for each reference

All we've done is reverse these indices!

Example Program - Poor Locality



POOR Data Locality:

256 Kbyte address jumps for each reference

Start Address big_array(0,0,0)

65 MB Data Segment
16 Million 4-byte ints
(65 MB/pagesize) Pages

End Address big_array(N,N,N)

Test Program
Default Compile and Run

- default compile will set page size to the kernel default "D"
  - default kernel tune is for 16K page size
- Program Memory Map will consist of:
  - Text Segment: very small
  - Data Segment: large

    4096 entries of 16 Kbyte pages = 65 Mbytes
- Use vps_stats to report Memory Map Pages

Execution Times
Default (16 KB) pages

- Execution time: measure with "timex"

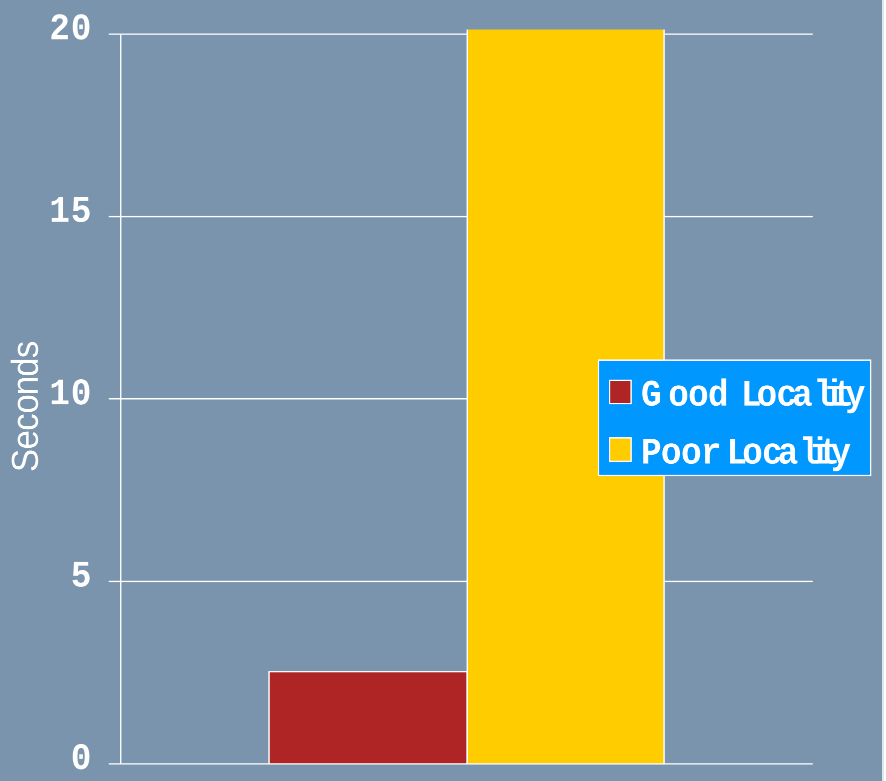
- TLB misses: measure with "cyclemeter" (a contributed tool) or PerfUX or GlanceUX.

- Good Locality program:

Runtime	2.52 sec.
Data TLB misses	51,535

- Poor Locality program:

Runtime	20.49 sec.
Data TLB misses	17,062,436



Execution Times
chatr to 256 KByte pages

- Change Data Page size:

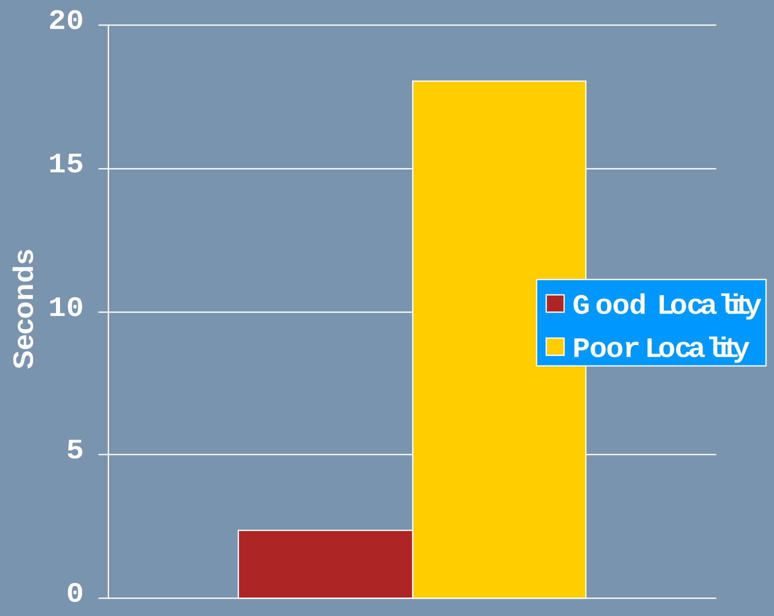
  “chatr +pd 256K big_array”

- Good Locality program:

Runtime	2.35 sec.
Data TLB misses	34,562

- Poor Locality program:

Runtime	18.06 sec.
Data TLB misses	16,906,509

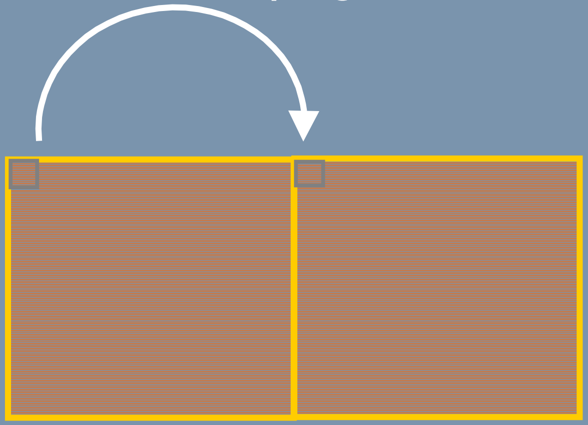


Change to 256 Kbyte Pages - why little performance change?

- Data Page Size is 256 Kbytes
- "Poor Locality" C-program references 4-byte integers from address (A,B,C) to (A+65536,B,C), ...
- CurrentAddress + (4 bytes * 65536) =

  CurrentAddress + 256K

  this is ***just over*** the next page boundary



Execution Times chatr to 4 MByte pages

- Change Data Page size:

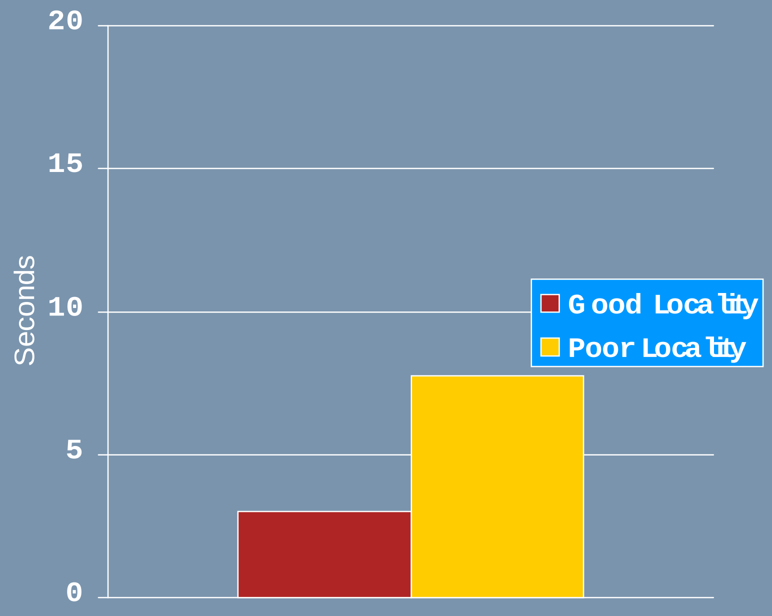
  "chatr +pd 4M big_array"

- Good Locality program:

Runtime	2.99 sec.
Data TLB misses`	33672

- Poor Locality program:

Runtime	7.75 sec.
Data TLB misses 30,227	



Variable Pages - Other Considerations

- Other factors affecting performance:
  - other ongoing system activity
  - memory page contention/thrashing/locking issues
  - cache sizes and cache fits

- Best performance is a combination of:
  - ✔ fast and efficient hardware (PA-8000)
  - ✔ correct system and kernel tunes
  - ✔ well-written software

Process Address Space Extensions

- HP-UX 11i allows one to change a 32-bit application's address spaces from Shared Memory to Private Data Space
  - 32-bit HPUX address spaces nominally 1GB each

Q1	Private Text	(and Data if EXEC_MAGIC)
Q2	Private Data	
Q3	shared objects	-> can change to Private Data
Q4	global shared objects	-> can change to Private Data

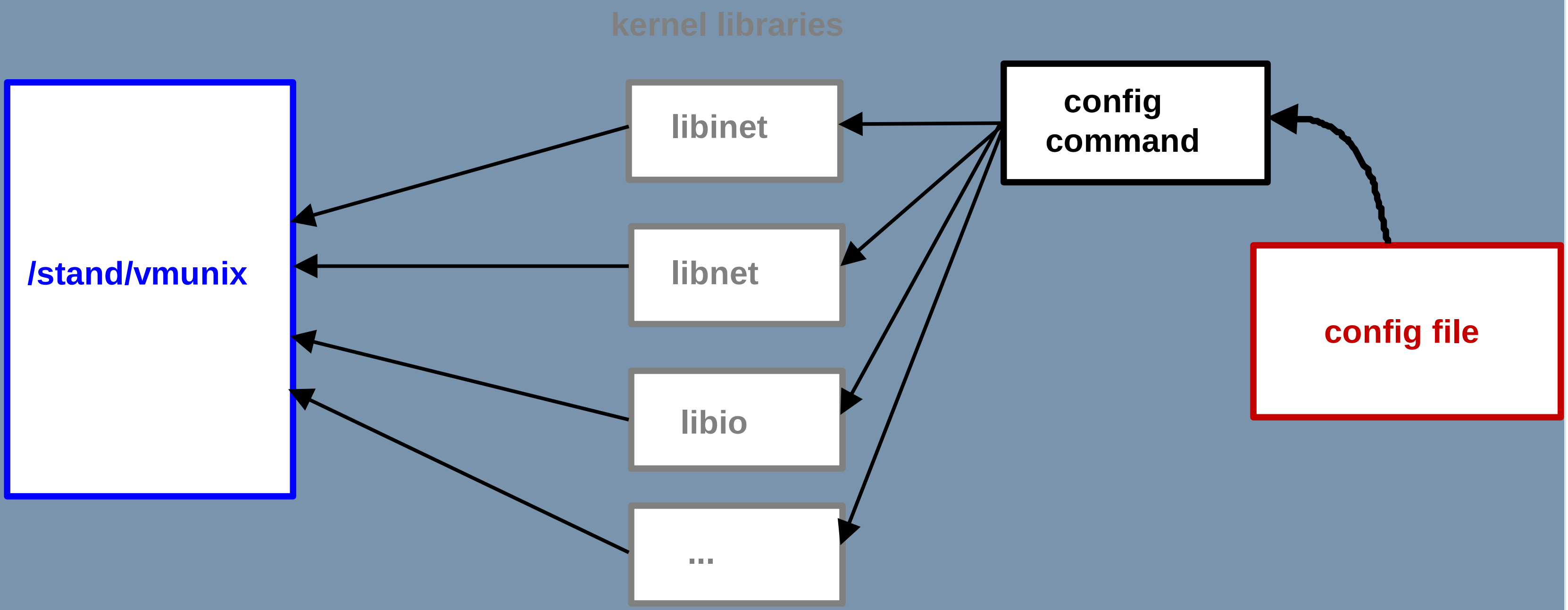
  - Change on a pre-program basis using the chatr(1) command:

```
chatr +q[3|4]p [enable|disable]
```

- Memory Windows: a new mechanism to explicitly create and share additional memory segments among processes
- Documentation: see Release Notes, Whitepapers, man pages

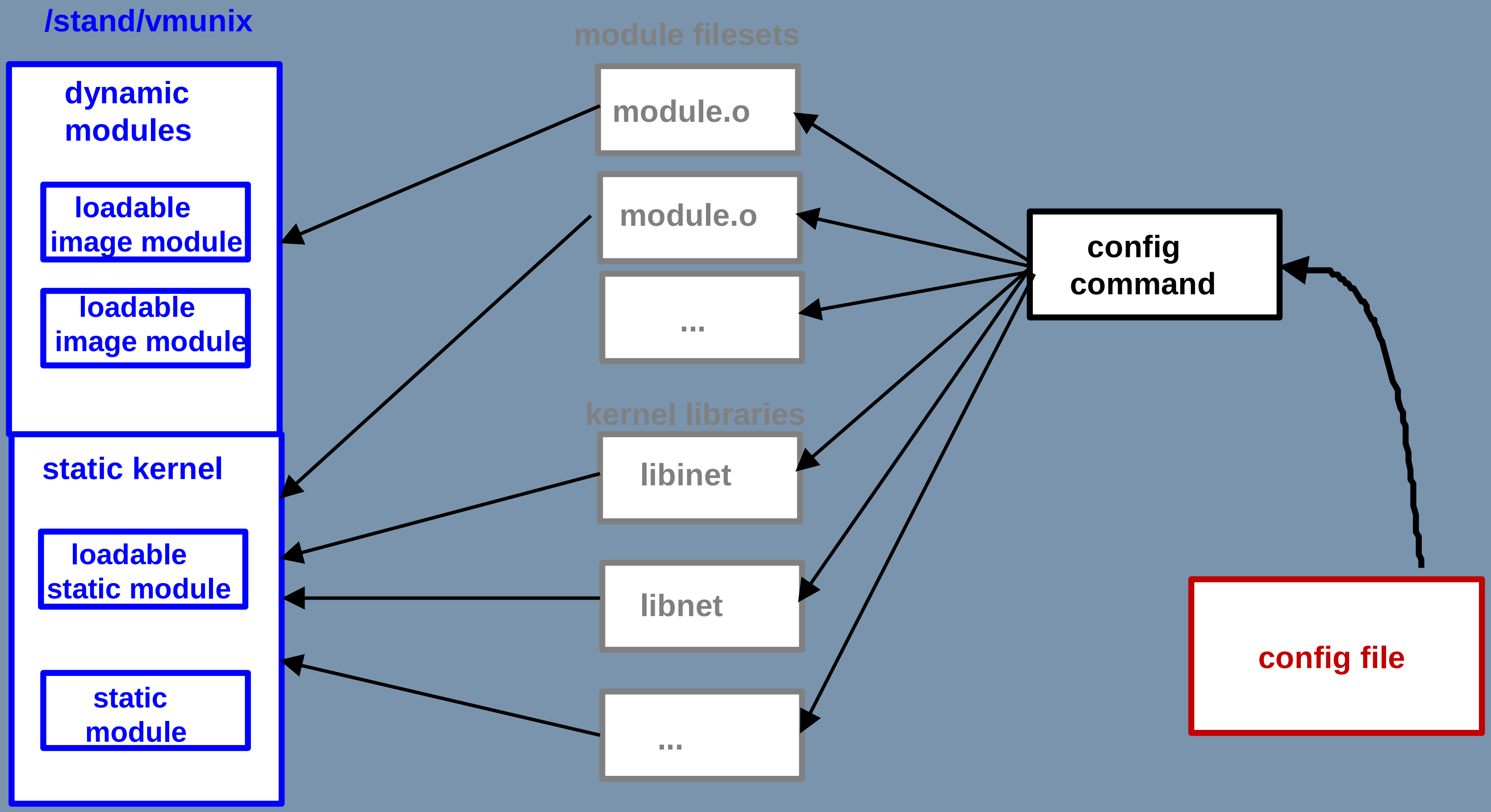
Static Kernel Configuration

- Prior to 11.0 Release, HP-UX kernels have been statically configured and built
- config command to build kernel



Dynamic Kernel Configuration

Dynamically Loadable Kernel Modules allow one to load, unload, and configure kernel modules without rebooting



Configuring Kernels

- Static kernel configuration remains exactly the same
  - config command and kernel config files
- To build a new kernel module:
  - config -M <module>
- To update the kernel with the new module:
  - config -u /stand/system -or-
  - kmupdate (new command)

New Commands for configuring Kernels

kmsystem	**set control flags in system files**
kmtune	**interface to set tunable parameters**
kmupdate	**update system with new kernel or loadable modules**
kminstall	**install/remove/update a module in a system**
kmmodreg	**register/unregister a module with the system**
kmadmin	**general administrative interface for DLKM**

Dynamic Kernel Tunables

- Certain Kernel Tunable Parameters can be immediately changed in HP-UX 11i and take effect without requiring kernel regen or reboot

- Interface through SAM : Kernel Configuration : Configurable Parameters
  - display field in SAM indicates type "Static" or "Dynamic"
  - Change to dynamic tunable parameters will take effect immediately

- Dynamic Tunable Parameters (current):

```
maxuprc    maxfiles_lim   maxtsiz maxtsiz_64bit
msgmax     msgmnb         shmmax  shmseg
core_addshmem_read        core_addshmem_write
```

System Recovery

- Use to recover Root Filesystem with all your installation and customization
- Root recovery might ordinarily require these steps:
  - cold install
  - configure
  - reinstall patches
  - reinstall applications
  - reinstall user data and files
- Other uses for System Recovery:
  - Modify Root Filesystem size
  - Modify primary swap size
  - Convert Root Filesystem from HFS to VxFS
  - Clone a system
- Supported on 11.0 and 10.X
- Installed with Ignite-UX, from DART releases or HP Web

System Recovery

make_recovery	make a system recovery tape ("make_recovery -A" - entire core VG)
save_config	create a configuration file that details the current system hardware and software configuration
check_recovery	compare current system configuration to last configuration file
print_manifest	print hardware configuration (CPUs, LVM and disks, I/O), OS configuration, installed products)

System Recovery - print_manifest utility

```
System Information
    Your Hewlett-Packard 9000 computer has software installed and configured as follows.

-------------------------------------------------------------
NOTE: You should retain this information for future reference.
-------------------------------------------------------------

System Hardware
    Model:              9000/777/C110
    Main Memory:        128 MB
    Processors:         1
    OS mode:            32 bit
    HW capability:      32 bit
    LAN hardware ID:    0x0060B001BA5B
    Software ID:        2011905808
    Keyboard Language:  PS2_DIN_US_English

    Storage devices                  HW Path       Interface
    SEAGATE ST15150W 4095 Mb         8/12.6.0      GSC built-in Fast/Wide SCSI Interface
    SEAGATE ST32430N 2048 Mb         8/16/5.4.0    Built-in SCSI
    HP C1533A                        8/16/5.3.0    Built-in SCSI

    I/O Interfaces
    Class          H/W Path       Driver         Description
    ext_bus        8/12           c720           GSC built-in Fast/Wide SCSI Interface
    ext_bus        8/16/0         CentIf         Built-in Parallel Interface
    audio          8/16/1         audio          Built-in Audio
    tty            8/16/4         asio0          Built-in RS-232C
    ext_bus        8/16/5         c720           Built-in SCSI
    lan            8/16/6         lan2           Built-in LAN
    ps2            8/16/7         ps2            Built-in Keyboard/Mouse
    pc             8/16/10        fdc            Built-in Floppy Drive
    hil            8/20/1         hil            Built-in HIL
    tty            8/20/2         asio0          Built-in RS-232C
    graphics       10/16          graph3         Graphics


Installed Software
    Your system was installed with HP-UX version B.11.00.

    Your system has the following software products installed and
    configured on the system disk drive(s).

    Product           Revision          Description
    B3782EA           B.10.20           HP-UX Media Kit (Reference Only. See Description)
    B3884EA_AGN       B.10.20           HP-UX 32-User License
    B3899BA           B.11.01.01        HP C/ANSI C Developer's Bundle for HP-UX 11.00 (S700)
    B3911DB           B.11.01.01        HP aC++ Compiler   (S700)
    B3919EA_AGS       B.11.00           HP-UX Unlimited-User License
    B4580AA           B.11.00.01        HP-UX 11.00 Software Transition Kit
    B5455CA           C.01.16.00        HP-UX Development Kit for Java*
    B5724AA_APZ       A.1.45            HP-UX Installation Utilities (Ignite-UX - S700 - 10.20)
    DCEProg           B.10.20           DCE Programming and Archive Libraries
    HPUXEngCR700      B.10.20           English HP-UX CDE Runtime Environment
    J2559C            D.06.15           Hewlett-Packard JetAdmin for Unix Utility
    UXCoreMedia-J     B.11.00           HP-UX Japanese Media Kit (Reference Only. See Description)
    XSWGR1100         B.11.00.39        HP-UX Extension Pack, June 1998
```

HARDWARE CONFIGURATION

INSTALLED SOFTWARE

System Recovery - print_manifest (continued)

```
LVM File System Configuration
    This system is configured with Logical Volume Manager (LVM) file systems.
    Refer to the File System layout section for information on the LVM layout.

Disk layout
    LVM disk                   Device file        HW Addr      size   vol. grp
    SEAGATE ST15150W           /dev/dsk/c0t6d0    8/12.6.0     4095   /dev/vg00
    non-LVM disk               Device file        HW Addr      size   swap
    SEAGATE ST32430N           /dev/dsk/c1t4d0    8/16/5.4.0   2003   0

File System layout
    LVM Device file            mount point    size  fs type
    /dev/vg00:
    /dev/vg00/lvol3            /              84    hfs
    /dev/vg00/lvol2            swap           256
    /dev/vg00/lvol1            /stand         48    hfs
    /dev/vg00/lvol7            /usr           700   hfs
    /dev/vg00/lvol5            /opt           1000  hfs
    /dev/vg00/lvol8            /var           160   hfs
    /dev/vg00/lvol6            /tmp           32    hfs
    /dev/vg00/lvol4            /home          1800  hfs
    /dev/vg00                  unallocated    12
    Device file                  mount point       size      fs type
    /dev/dsk/c1t4d0              /mnt/c1t4d0      2003       hfs
```

DISK AND LVM LAYOUT

```
Swap configuration
    type       size  priority  device/location
    dev        256      1      /dev/vg00/lvol2

Kernel Configuration
    The following drivers or parameters are configured into your system's
    kernel.  After installing HP-UX, use the sam(1m) command to configure
    the following items into the kernel:
         STRMSGSZ                65535
         default_disk_ir             1
         maxdsiz              0X10000000
         maxssiz              0X10000000
         maxtsiz              0X10000000
         nstrpty                    60
```

KERNEL CONFIGURATION

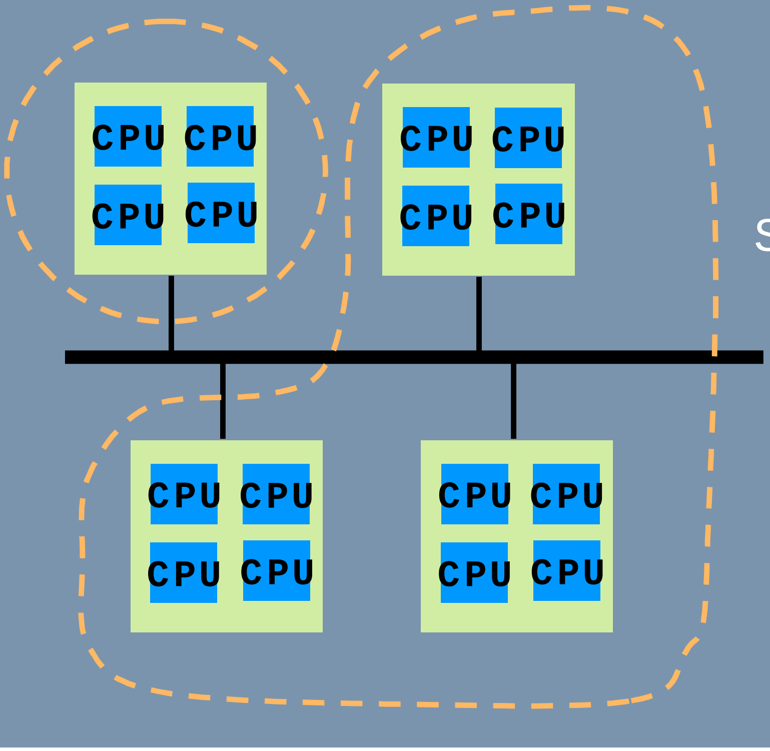
```
System Information
    The following parameters were set on the configured target:
         hostname:           hp46t250
         IP address:         15.14.120.250
         subnet mask:        255.255.248.0
         gateway IP address: 15.14.120.250
         time zone:          PST8PDT
         DNS domain name:    cup.hp.com
         DNS IP address:     15.13.185.120
         DNS IP address:     15.13.192.134
```

NETWORK CONFIGURATION

HP-UX Partitions

- Supported starting with SuperDome systems and HP-UX 11i
- Create multiple partitions (instances) of HP-UX within same system
  - static partitions - must reboot to effect configuration change

System #1: hp-jwf1
1 cell/4 CPUs

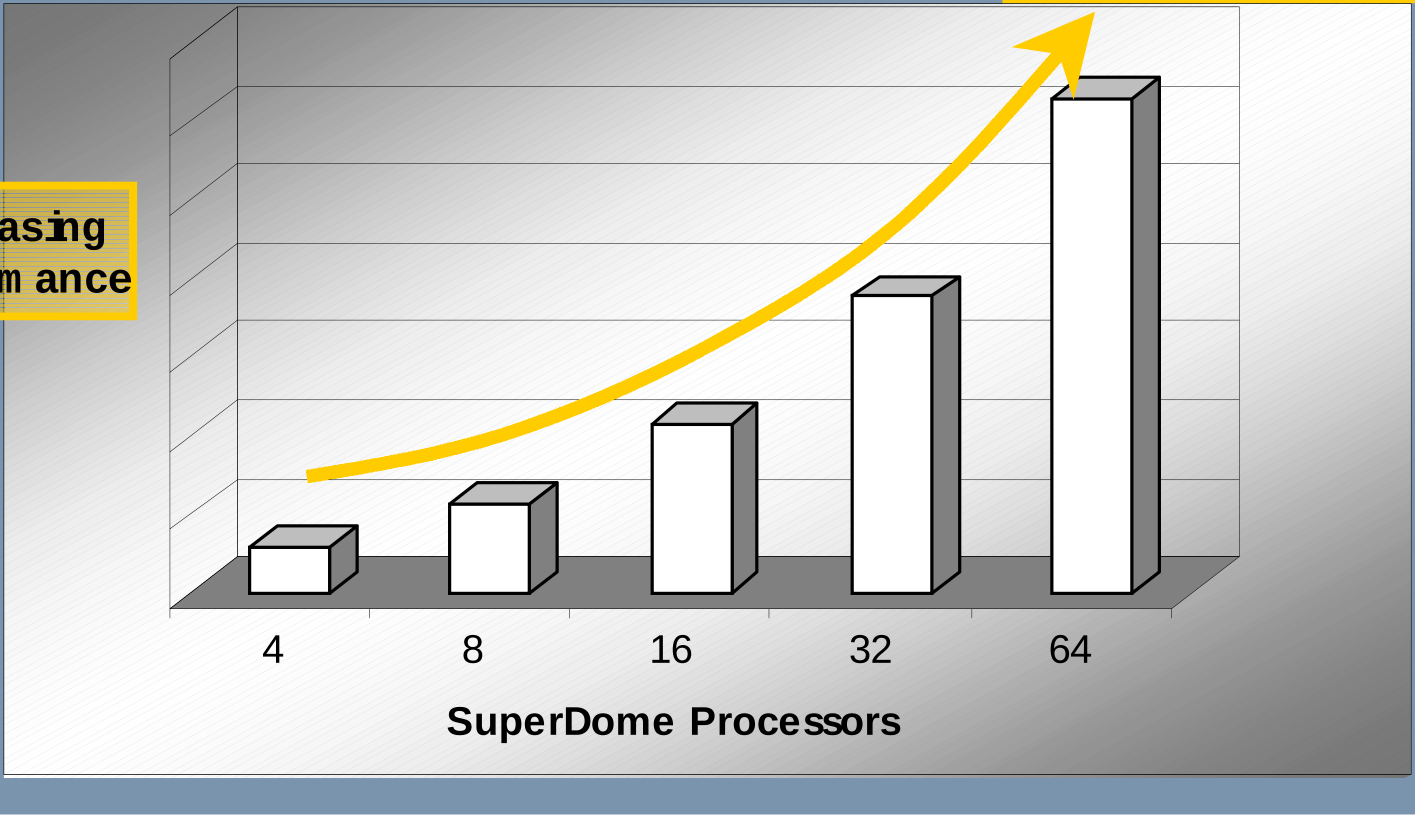


System #1: hp-jwf2
3 cells/12 CPUs

HP-UX Partitions: scale the number of CPUs in a system

- SuperDome: maximum of 64 CPUs
  - ➨ 4 CPUs per cell
  - ➨ 8 cells per system
  - ➨ 2 SuperDome systems configured together

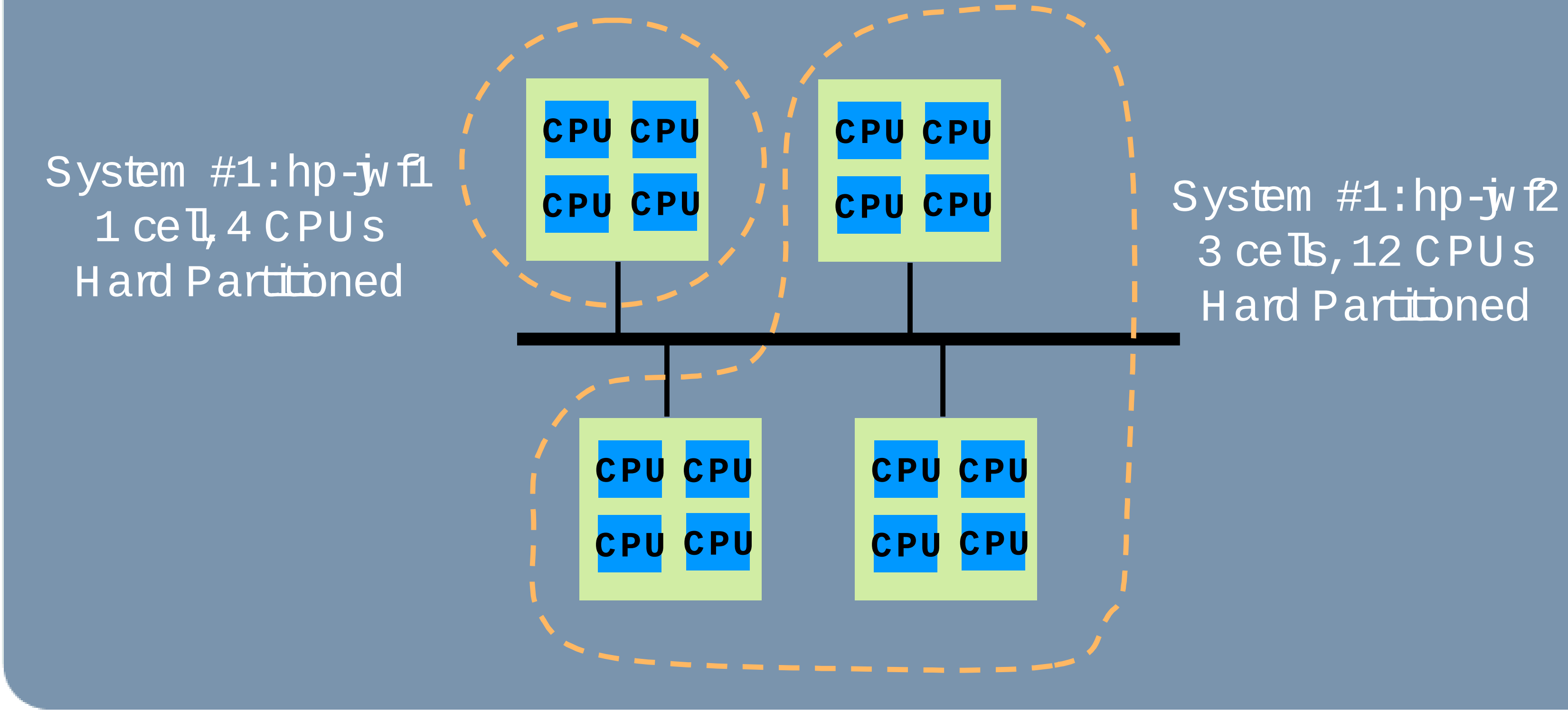
HP-UX 11i and SuperDome Scale to the Enterprise!



Increasing Performance

Hard Partitions and Virtual Partitions

- Hard Partitions: hardware isolation
- Partitioning takes place at Cell boundaries
- Must be supported by underlying hardware - SuperDome
- Hardware isolation and protection in event of underlying hardware fault



System #1: hp-jwf1
1 cell, 4 CPUs
Hard Partitioned

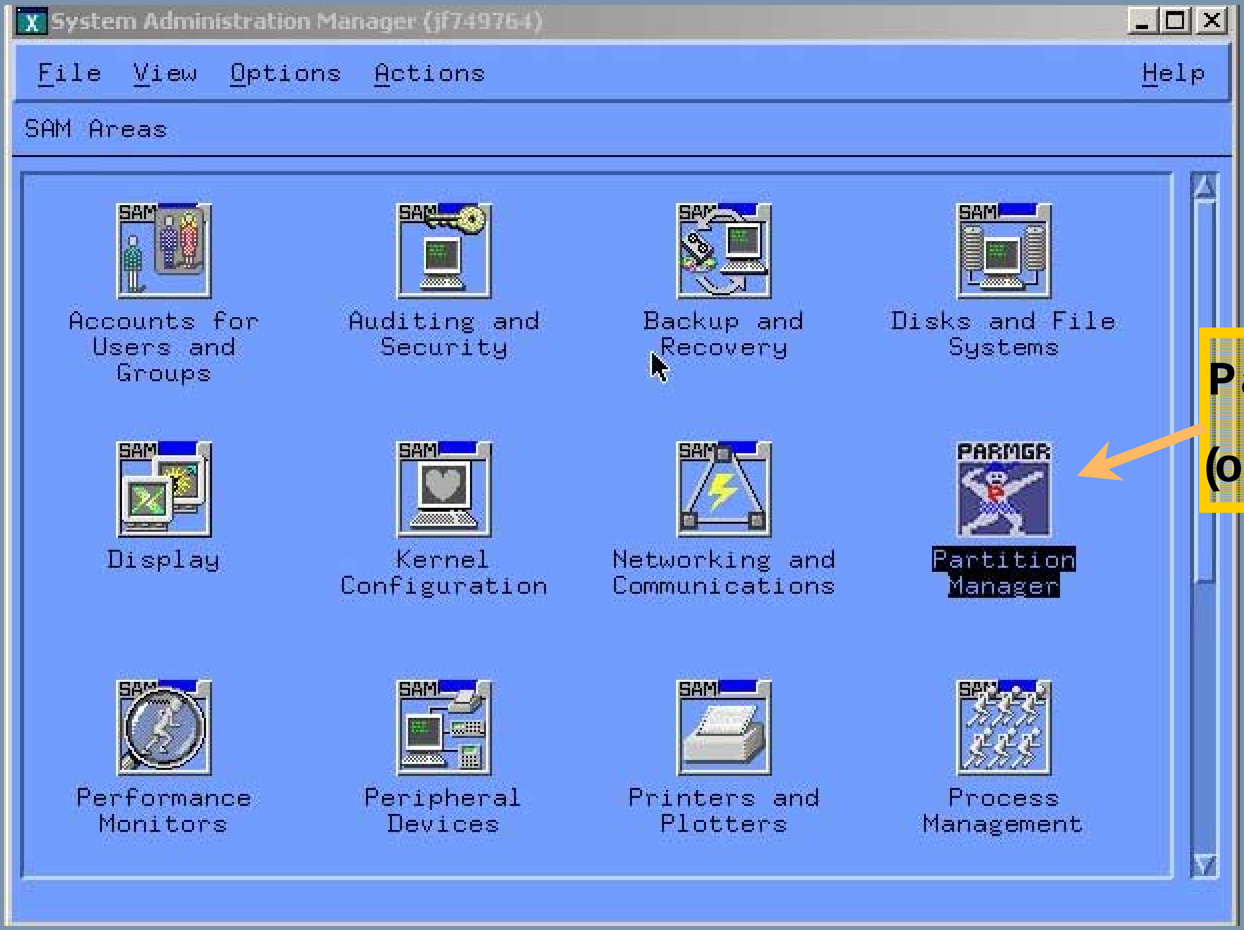
System #1: hp-jwf2
3 cells, 12 CPUs
Hard Partitioned

HP-UX Partitions

- Hard Partitions supported with HP-UX 11i first release
- Administration of Partitions:
  - Partition Manager under SAM (use the GUI)
  - command line options:

parmgr	invokes the Partition Manager GUI
parstatus	display partitions and resouces
parmodify	modify existing partition
parcreate	create a new partition
parremove	remove existing partition
parunlock	unlock configuration data
fruled	turn LEDs on/off
frupower	turn power on/off for cells, cabinets, I/O chassis

Partition Manager: configuration under SAM



Partition Manager

(or invoke parmgr)

HP-UX Partitions

- Changes to other commands:

shutdown and reboot	-"R" option to reconfigure partitions
setboot	root cell number added to bootpath

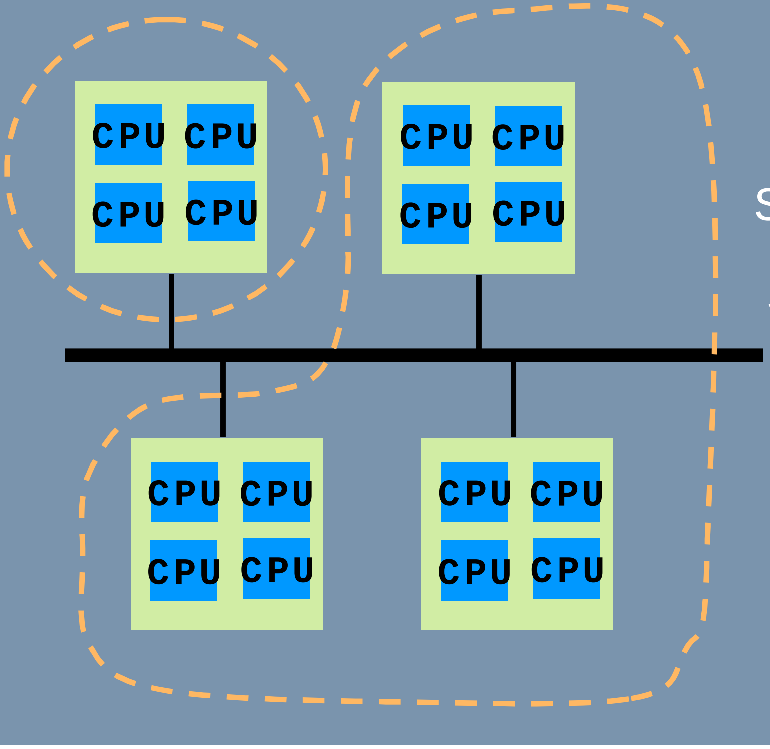
- All partition commands operate on systems with appropriate hardware!

  default operation otherwise

Hard Partitions and Virtual Partitions

- Virtual Partitions: software (kernel) partitions
- Partitioning takes place at CPU boundaries
- To be supported on SD, N, L-class
- Software isolation in event of kernel panic

System #1: hp-jwf1
1 cell, 4 CPUs
Virtual Partitioned

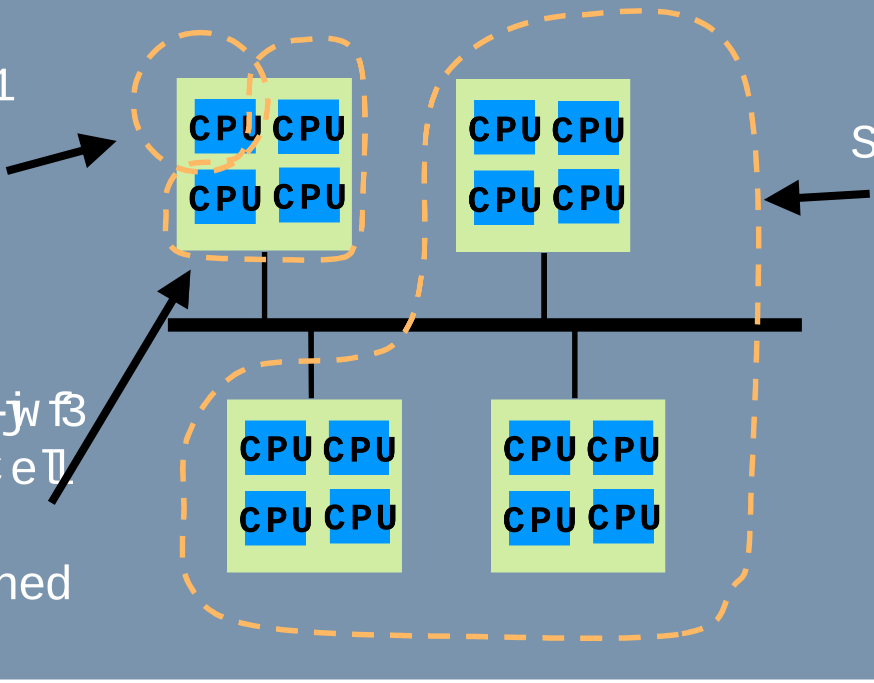


System #1: hp-jwf2
3 cells, 12 CPUs
Virtual Partitioned

Hard Partitions and Virtual Partitions

- Will be able to combine Hard Partitions and Virtual Partitions within the same system
  - Hard partitions on hardware Cell boundaries (on supported systems)
    - – hardware isolation and protection
  - Virtual partitions on CPU boundaries (on supported systems)

System #1: hp-jwf1
1 CPU in 1 Cell
Hard and
Virtual Partitioned



System #1: hp-jwf2
3 cells, 12 CPUs
Hard Partitioned

System #3: hp-jwf3
3 CPUs in 1 Cell
Hard and
Virtual Partitioned

OLAR for I/O

- Supported starting with HP-UX 11i and SD, N, and L-class systems
- Administration through SAM menus
- Supported operations:
  - addition of new I/O cards
  - replacement of I/O cards

HP-UX Security Enhancements

UNIX “Stack Smashing” attacks:

- Many commercial UNIXes, based on BSD and SysV code base, had a number of security holes subject to “Stack Smashing” attacks. In this attack a malicious intruder sends an improperly formatted or overly long message to one of a number of Internet daemons running with root privileges on the system.

- The daemons attempt to copy the improperly formatted message onto a stack by using an unbounded copy routine. The data copy overwrites the stacked return address and forces return to specially inserted code that typically grants root privileges to the intruder.

- This type of attack is easily automated and often used by “script kiddie” attackers.

HP-UX Security Enhancements

- Security Security Announcements
  - CERT Computer Emergency Response Team at CMU
  - http://www.cert.org
  - list of CERT Advisories CA -<Year> -<Number>
- Example for "Stack Smashing" attack on syslogd:
  - CA-1995-13
  - Series of HP-UX patches were issued for all supported releases and contained in all subsequent releases.

HP-UX Security: Stronger protection against "Execute Protected Stack" attacks

- PA-RISC hardware and HP-UX Kernel can set hardware protection on memory region boundaries which can prevent code execution from a memory region ("deny execute permissions"). Set this to deny execute permissions from Program Stack memory regions.

- The hardware detects and then the kernel traps such attempted access and prevents execution access.

- No performance degradation, no recompilation required.

- Minor compatibility change.

Execute Protected Stacks

- Enable in HP-UX kernel with new (HP-UX 11i) configurable parameter “executable_stack” in the system file:

```
    executable_stack  0  # enable executable stack protections
```

- Enable protection on a program-by-program basis using chatr

```
    chatr +es  <target_program>
```

- Introduced with HP-UX 11i, but turned off by default to maintain maximum compatibility

- Will become the default on HP-UX at a future release

iCOD Instant Capacity on Demand

- Instant activation of pre-installed CPUs on L, N, and V-Servers
- HP-UX 11.0 and HP-UX 11i support
- iCOD Phase 1: CPUs activated at Boot Time
- iCOD Phase 2: CPUs activated dynamically
- iCOD auditing process sends e-mail notification to HP
- iCOD reference: http://www.unixsolutions.hp.com, iCOD link

iCOD Phase 2
Dynamic processor activation

- Administrative commands:

icod_modify	-m	change number of activated processors
		processors immediately put in use by HP-UX
	-c	change contact information
icod_notify		send or stop contact email notices
icod_stat		display processor allocation status

Using iCOD

```
% icod_stat             # run icod_stat on 8-way N-Class
Version: 2.0

Hostname:        hp46t45
DNS domain name: cup.hp.com
IP address:      15.14.120.45
NIS domain name: .
System model:    9000/800/N4000-44
Serial number:   USM39353SB
Software ID:     641339333

Contact name:    XXXXXXXXXXXXXXXXXX
Contact e-mail:  XXXXXXXXXXXXXXXXXX
Contact phone:   XXXXXXXXXXXXXXXXXX

Total processors:                  8
Allocated processors:              6

iCOD deallocated processors:       0
Other deallocated processors:      2
Firmware deconfigured processors:  0
-------------------------------------
Total number of unused processors: 2

Current number of HP-owned (iCOD) processors: 2
  … Change History Follows ...
```

HP-UX 11.X Networking

- ndd - Network stack tuning and configuration
- IP aliasing and Logical IP addresses
- NFS PV3
- NIS+
- CIFS - File Sharing HP-UX to Windows

HP-UX 11.X Networking
ndd

- Distributed in 11.00 Release
- dynamic tuning and configuration of networking stack
  - timeouts, queue sizes, connection pools, etc.
- Configuration file applied at UserSpace startup
  - /etc/rc.config.d/nddconf
- ndd -h
  - gives entire set of tunables (very long)
- See man pages, Release Notes for documentation for this tool
- Networked applications may have tuning information for parameters accessed by ndd.

HP-UX 11.0 Networking
IP aliasing

- 10.X Release supported ifalias command -
  - assign multiple IP addresses to same interface
- 11.0 Release: set additional IP addresses using ifconfig command
  - ifconfig lan0 inet 15.75.183.227 netmask 255.255.255.240
  - ifconfig lan0:1 inet 15.75.183.231 netmask 255.255.255.240
  - netstat, other commands use “:N” notation, default is “:0”

NFS Protocol Version 3 Advantages

- Added Support for Large Files
- Improved Performance
- Enhanced File Access control
- New APIs

- Delivered in 10.30 and 11.0 Releases
- Workstation ACE2 Release for 10.20 OS

NFS 3 Support for Large Files

	NFS Version 2	NFS Version 3
File Size and Offsets	32 bits	64 bits
Maximum File Size Supported	2 GBytes	128 GBytes

NFS PV3 Performance Improvements

- Function calls now return attributes to reduce subsequent getattr() function calls

- Read/write blocks can be larger than the previous 8 Kbyte limit

- Performance of asynchronous write operations much improved: write request from client returns immediately, commit request (NEW) from client causes update to disk on server

- Weak cache consistency: if client modification time matches server modification time the client cache is assumed to be valid

- Remains fully interoperational with NFS V2

NIS+

- Improved performance and security over NIS

- Continues to support NIS and local file access

- Capability to add future services online without rebooting or reconfiguring

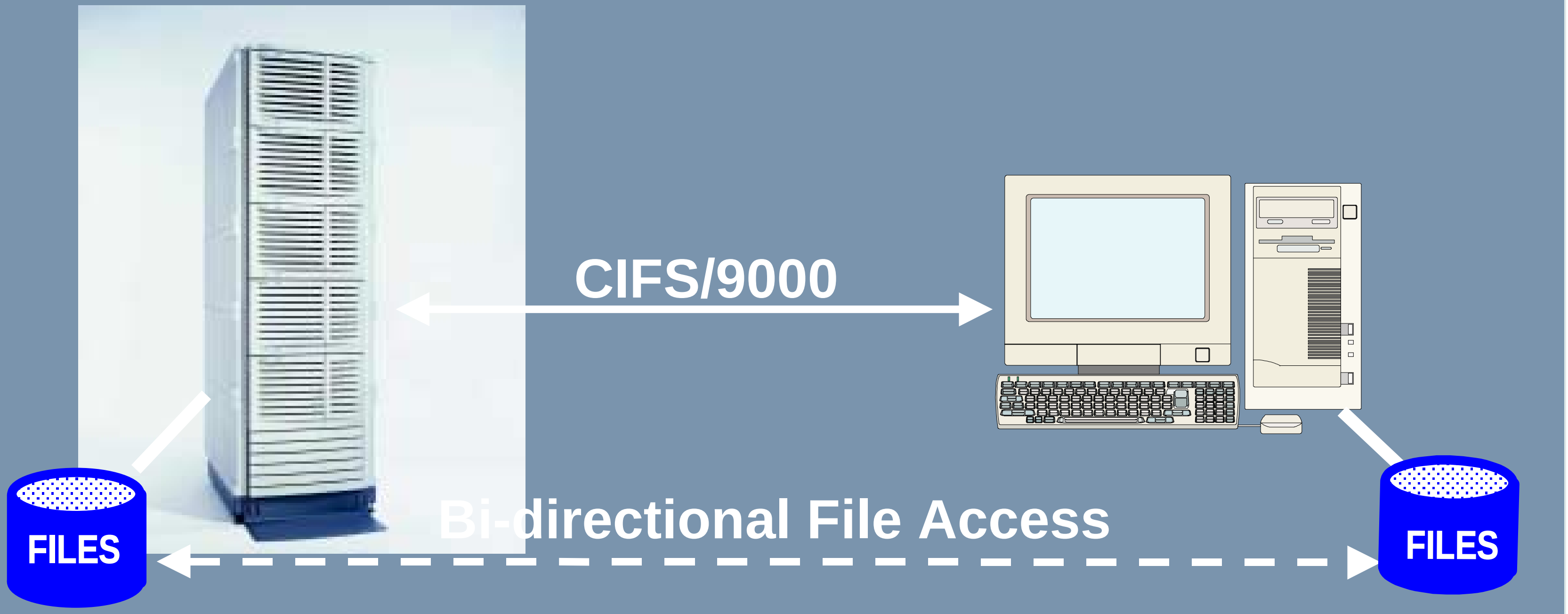
- Delivered in the 10.30 and 11.0 Releases

NIS+ vs. NIS
Comparison of Features

	NIS	NIS+
Map updates (Master to Slaves)	Entire map updated	Incremental changes updated
Update propagation	Manual	Automatic
Data Access Restrictions	None	Access controlled on per-entry basis
Authentication	None	Secure RPC
Administration	Must be done on singl Master Server	May be made within hierarchical namespace
Namespace	Flat	Hierarchical
Contacting Servers	UDP broadcast	coldstart config file and directory cache

CIFS/9000
UNIX-Windows File Sharing

- File-sharing both Server- and Client-side HP-UX to/from Windows
- Uses Windows SMB network protocol



CIFS/9000 Server Component

CIFS/9000
Server
FILES

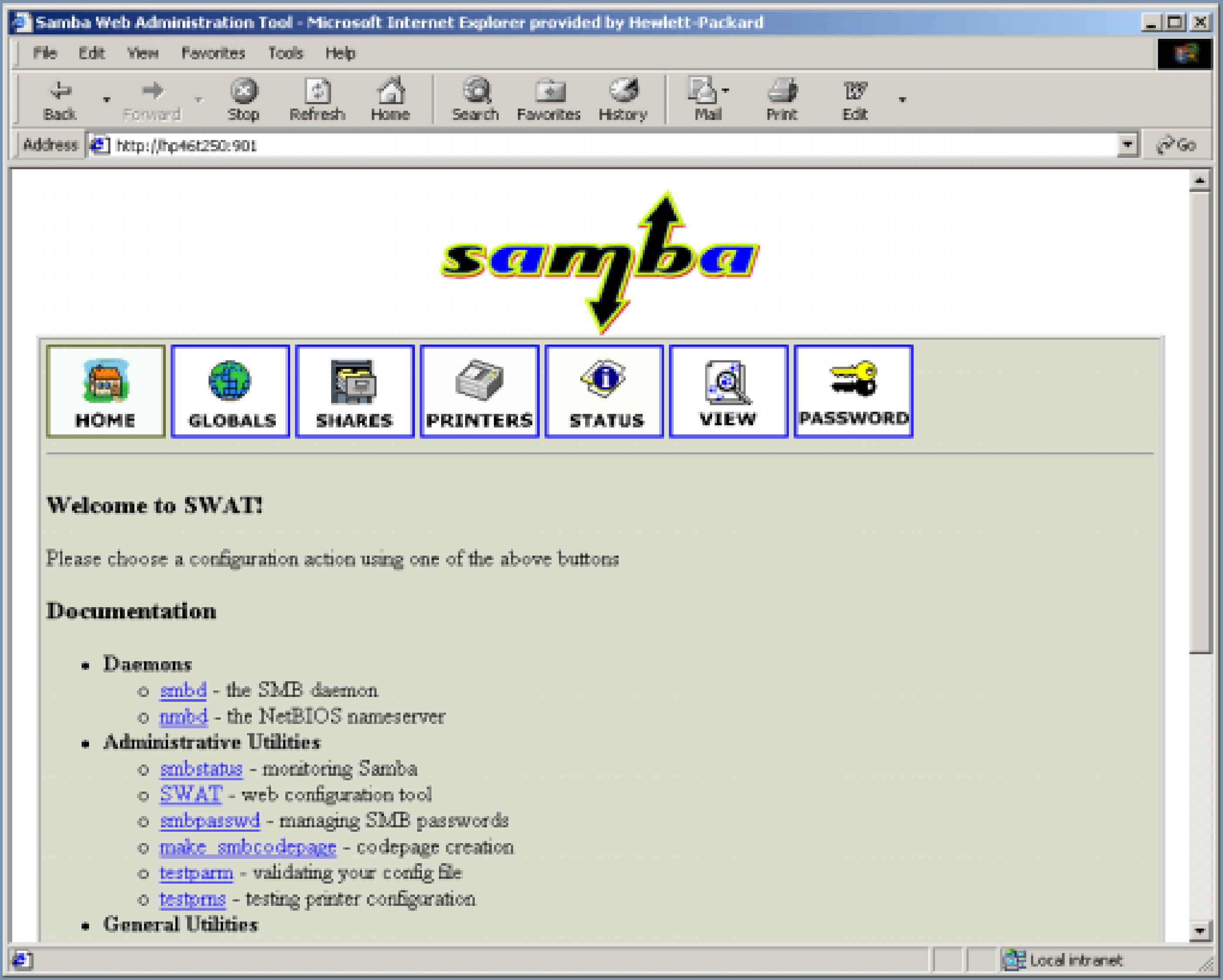
- Based on Open Source Samba
- smbd/nmbd daemons run on Server
- No kernel modifications required

CIFS/9000 Server Component

CIFS/9000
Server
FILES

- Install Server product
- run samba_setup
  - creates smb.conf
- smbd/nmbd daemons started in run-level 2

Swat-Samba Web



Samba Web Administration Tool - Microsoft Internet Explorer provided by Hewlett-Packard
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit
Address http://hp46t250:901
Go
samba
HOME
GLOBALS
SHARES
PRINTERS
STATUS
VIEW
PASSWORD
Local intranet

Welcome to SWAT!

Please choose a configuration action using one of the above buttons

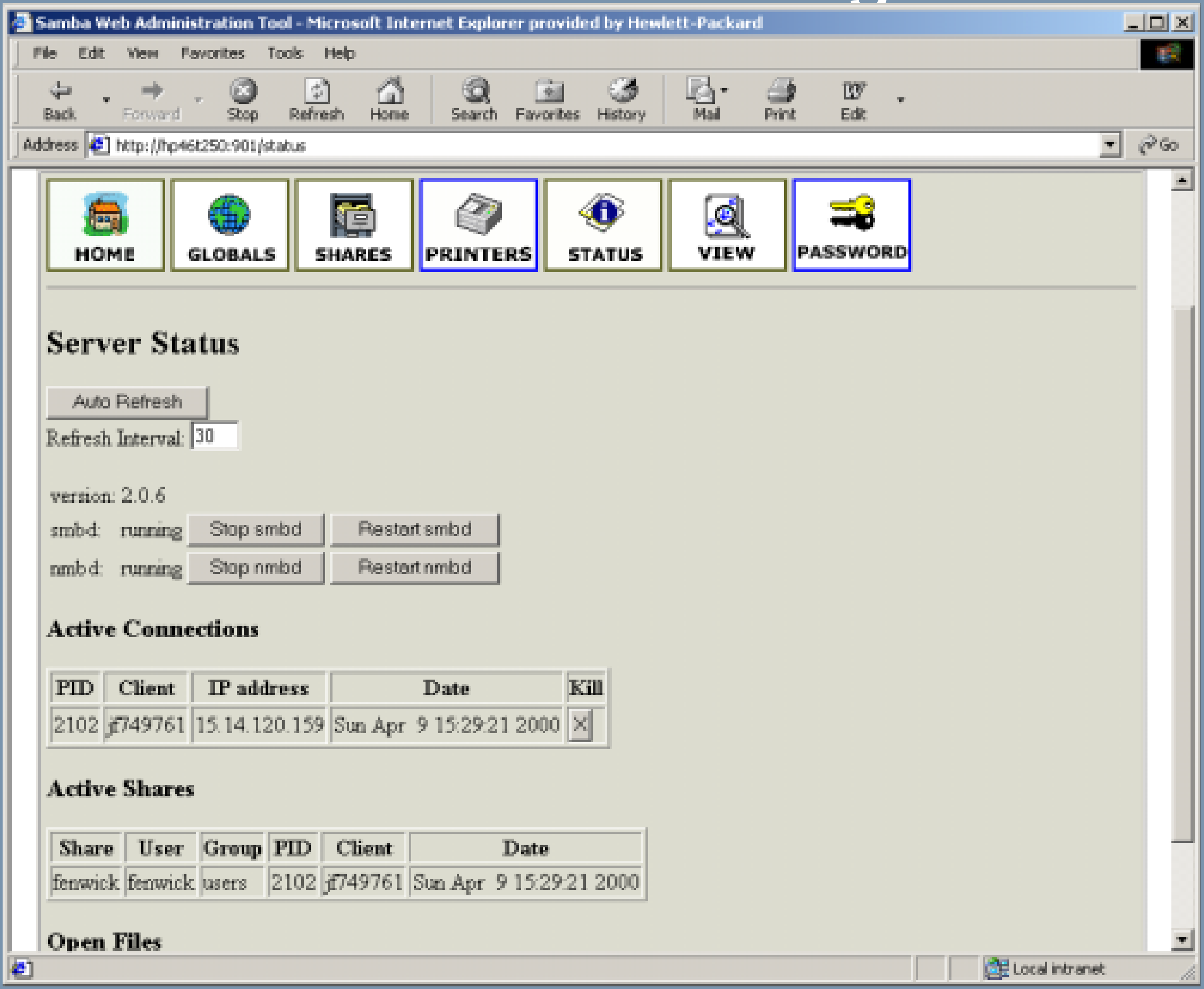
Documentation

- Daemons
  - smbd - the SMB daemon
  - nmbd - the NetBIOS nameserver
- Administrative Utilities
  - smbstatus - monitoring Samba
  - SWAT - web configuration tool
  - smbpasswd - managing SMB passwords
  - make_smbcodepage - codepage creation
  - testparm - validating your config file
  - testprns - testing printer configuration
- General Utilities

Samba Web Administration Tool

- Included in HP-UX Samba Server distribution
- GUI management of Samba configuration
  - connect to Server:901 port
- Requires root privileges for most operations
- Access to Samba utilities and documentation

Samba: Status using swat



http://hp46t250:901/status

Server Status

Auto Refresh

Refresh Interval: 30

version: 2.0.6

smbd: running [Stop smbd] [Restart smbd]

nmbd: running [Stop nmbd] [Restart nmbd]

Active Connections

PID	Client	IP address	Date	Kill
2102	jf749761	15.14.120.159	Sun Apr 9 15:29:21 2000	☒

Active Shares

Share	User	Group	PID	Client	Date
fenwick	fenwick	users	2102	jf749761	Sun Apr 9 15:29:21 2000

Open Files

CIFS/9000
Client Component

- Install CIFS Client product
- Kernel modifications at vnode layer
  - kernel modifications required
- Start CIFS Client daemon on Server:

```
/opt/cifsclient/bin/cifsclient start
```

(not entered in automatic run-level startup)

CIFS/9000
Client Component

- Login to Windows system

```
% cifslogin jf749761 jfenwick
```

- Mount remote Windows filesystem:

```
% cifsmount //jf749761/C /mnt/dos-c
```

- List files on PC from HP-UX:

```
% ls /mnt/dos-c/

AUTOEXEC.BAT   IO.SYS         Omnibook       XHD3D35
ntldr
BOOTLOG.PRV    MAESTRO.COM    PQMAGIC        bin
temp
BOOTLOG.TXT    MSDOS.---      Program Files  bios
CONFIG.AGO     MSDOS.SYS      RECYCLED       boot.ini
CONFIG.SYS     MSDOS.^^^      SCANDISK.LOG   bootsect.dos
DATA          My Documents   SETUPLOG.OLD   bootsect.lnx
```

Additional Networking Enhancements

- New FTP June 1998 (Patch PHNE_14479)
    - improved logging, security, on the fly compression
- Kerberos v5 1.0
    - Provides encryption and authentication; simplified installation
- DNS 4.9.6
    - Eases load balancing through round robin
- Sendmail 8.8.6 May 1999 EAP
    - Prevents system overload with anti-spamming feature
- BIND 4.9
    - Improves response times and enhances security
- Gateway Daemon 3.5.1 (gated)

HP-UX and Futures

- HP-UX and IA-64
  - IA-64 = the new computer architecture being introduced by HP and Intel
- HP-UX and AOII
  - AOII = Always On Intranet Infrastructure
  - A strategic direction for HP that encompasses many of the technologies presented here

HP-UX and IA-64

- Upcoming release of HP-UX 11i Operating System will run on IA-64

- Cleanly developed HP-UX 11 applications will run on IA-64

- Applications may run on IA-64 as native IA-64 code or translated PA-RISC code

HP-UX on IA-64

- HP-UX will support execution either as:
  - Native IA-64 compiled Objects
  - Dynamic runtime-translated PA-RISC object
- See IA-64 Documentation for programming guidelines

Preparing for IA-64

- Download IA-64 STK Software Transition Kit to test software for IA-64 operation

  http://devresource.hp.com/devresource/Topics/IA64/IA64.html

- Additional documentation and references available there hp®

invent