

Introduction (3)

Process:

- Need a concept, once we have >1 program running
- program that is to be executed by the machine
- Its own data
- sealed off from other processes
- additional administrative data

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 5

Practice I

```
Sep 19 14:20:18 amd64 sshd[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 02:00:01 amd64 /usr/sbin/cron[101031]: (root) CMD (/sbin/evlogmgr -c 'age > *30d*')
Sep 20 02:00:01 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 sshd[6516]: STATS: dropped 0
Sep 20 12:48:43 amd64 syslog-ng[65091]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:44 amd64 sshd[66941]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 21 01:00:01 amd64 /usr/sbin/cron[170555]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 01:00:01 amd64 sshd[110881]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[110881]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 18:43:26 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 22 01:00:01 amd64 /usr/sbin/cron[46741]: (root) CMD (/sbin/evlogmgr -- "severity=DEBUG")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 22 20:23:21 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[24739]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 sshd[231971]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 23 18:04:05 amd64 sshd[65541]: Accepted publickey for esser from ::ffff:192.168.1.5 port 59771 ssh2
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 01:00:01 amd64 sshd[124364]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[132531]: (root) CMD (/sbin/evlogmgr -c 'age > *30d*')
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:15:48 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 13:49:08 amd64 sshd[231971]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_mixer_unsupported: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd[293991]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[14841]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 02:00:01 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[88891]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:47 amd64 sshd[89211]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sshd[19372]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 12:00:01 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd[115861]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:07:17 amd64 sshd[116081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 14:08:33 amd64 sshd[116301]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sshd[129301]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62798
```

Introduction (4)

Process list:

- Information about all processes and their states
 - Each process has a
- Process Control Block (PCB):**
- Identifier (PID)
 - Register values incl. program counter (instruction counter)
 - memory area of the process
 - Lists of open files and sockets
 - Information such as parent PID, last activity, total running time, priority, ...

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 6

Practice: User (1)

```
esser@sony:Skript> emacs test.txt &
[3] 24469
esser@sony:Skript> _
[...]
[3]+ Done                           emacs test.txt
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 8

Practice: User (2)

```
esser@sony:Skript> jobs  
[1]- Running xpdf -remote sk skript-bs.pdf &  
[2]+ Running nedit kap02/index.tex &  
  
esser@sony:Skript> jobs -l  
[1]- 8103 Running xpdf -remote sk skript-bs.pdf &  
[2]+ 20568 Running nedit kap02/index.tex &  
  
esser@sony:Skript> ps w|grep 8103|grep -v grep  
8103 pts/15 S 5:27 xpdf -remote sk skript-bs.pdf
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 9

Practice: User (4)

```
> pstree -p  
init(1)---acpid(2266)  
|--auditd(2727)---{audited}(2728)  
|--cron(3234)  
|--cupsd(2706)  
|--gpg-agent(4031)  
|-hald(2309)-+-hald-addon-acpi(2616)  
| |--hald-addon-stor(2911)  
| |`--hald-addon-stor(2914)  
-kded(4079)  
-kdeinit(4072)-+-artsd(7184)  
| |--kio_file(4402)  
| |`--klauncher(4077)  
| `--konqueror(22430)  
| `--konsole(11064)-+-bash(11065)---ssh(31205)  
| | |--bash(11119)---sux(11444)---bash(11447)  
| | |`--bash(11137)  
| | |`--bash(25637)-+-ssh(4522)  
| | | |`--xmms(7169)-+-{xmms}(7170)  
| | | |`--{xmms}(7171)  
| | |`--bash(15608)  
| `--konsole(4773)-+-bash(4774)---ssh(8037)  
| |`--bash(8040)---ssh(8058)  
| |`--bash(8061)-+-less(15188)  
| |`--nedit(9628)  
`--xpdf(8103)
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 11

Practice: User (3)

```
> ps auxw  
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND  
root 1 0.0 0.0 720 92 ? S Jun24 0:01 init [5]  
root 2 0.0 0.0 0 0 ? SN Jun24 1:09 [ksoftirqd/0]  
root 3 0.0 0.0 0 0 ? S< Jun24 0:11 [events/0]  
root 4 0.0 0.0 0 0 ? S< Jun24 0:00 [khelper]  
root 5 0.0 0.0 0 0 ? S< Jun24 0:00 [kthread]  
root 7 0.0 0.0 0 0 ? S< Jun24 0:02 [kblockd/0]  
root 8 0.0 0.0 0 0 ? S< Jun24 0:00 [kacpid]  
root 128 0.0 0.0 0 0 ? S< Jun24 0:00 [aio/0]  
[....]  
esser 5733 0.2 12.2 82420 63428 ? S Jul24 4:05 /usr/lib/opera/opera  
root 2670 0.3 0.0 1368 300 ? Ss 08:24 2:39 zmd /usr/lib/zmd/zmd.exe  
esser 8037 0.0 0.6 6452 3384 pts/13 S+ 11:23 0:05 ssh -X amd64
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 10

Practice: User (5)

- suspend program: Strg-Z
- continue in foreground: fg
- continue in background: bg
- send signal to process: kill
 - suspend (STOP), continue (CONT)
 - terminate (TERM), destroy (KILL)
- detach child from parent: disown

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 12

Practice: User (6)

```
> kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL
 5) SIGTRAP     6) SIGABRT     7) SIGBUS       8) SIGFPE
 9) SIGKILL     10) SIGUSR1    11) SIGSEGV     12) SIGUSR2
13) SIGPIPE     14) SIGALRM    15) SIGTERM     16) SIGSTKFLT
17) SIGCHLD     18) SIGCONT    19) SIGSTOP     20) SIGTSTP
21) SIGTTIN     22) SIGTTOU    23) SIGURG      24) SIGXCPU
25) SIGXFSZ     26) SIGVTALRM  27) SIGPROF     28) SIGWINCH
29) SIGIO       30) SIGPWR      31) SIGSYS      34) SIGRTMIN
35) SIGRTMIN+1   36) SIGRTMIN+2   37) SIGRTMIN+3   38) SIGRTMIN+4
39) SIGRTMIN+5   40) SIGRTMIN+6   41) SIGRTMIN+7   42) SIGRTMIN+8
43) SIGRTMIN+9   44) SIGRTMIN+10  45) SIGRTMIN+11  46) SIGRTMIN+12
47) SIGRTMIN+13  48) SIGRTMIN+14  49) SIGRTMIN+15  50) SIGRTMAX-14
51) SIGRTMAX-13 52) SIGRTMAX-12  53) SIGRTMAX-11  54) SIGRTMAX-10
55) SIGRTMAX-9  56) SIGRTMAX-8  57) SIGRTMAX-7  58) SIGRTMAX-6
59) SIGRTMAX-5  60) SIGRTMAX-4  61) SIGRTMAX-3  62) SIGRTMAX-2
63) SIGRTMAX-1  64) SIGRTMAX
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 13

Practise: Programmer (2)

Execute a different program in the new process:

```
exec ()
```

```
main()
{
    int pid=fork(); /* Create child process */
    if (pid == 0)
    {
        /* child launches external program */
        execl( "/bin/emacs", "/etc/fstab", (char *) 0 );
    }
    else
    {
        printf("You should see an editor window...\n");
    }
}
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 15

Practise: Programmer (1)

Create new process: **fork ()**

```
main()
{
    int pid=fork(); /* Create child process */
    if (pid == 0)
    {
        printf("I'm the child, my PID is %d.\n", getpid() );
    }
    else
    {
        printf("I'm the parent, my child has PID %d.\n", pid);
    }
}
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 14

Practise: Programmer (3)

Waiting for (termination of) child process: **wait ()**

```
#include <unistd.h>           /* sleep()
                                */
main()
{
    int pid=fork();           /* Create child process */
    if (pid == 0)
    {
        sleep(2);            /* go to sleep for 2 seconds */
        printf("I'm the child. My PID is %d\n", getpid() );
    }
    else
    {
        printf("I'm the parent. My child has PID %d\n", pid);
        wait();              /* wait for child */
    }
}
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 16

Practise: Programmer (4)

Truly more than one process:

After fork () two processes in process list

```
> pstree | grep simple
|           |---bash---simplefork---simplefork

> ps w | grep simple
25684 pts/16 S+          0:00 ./simplefork
25685 pts/16 S+          0:00 ./simplefork
```

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 17

Theory

```
Sep 19 14:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:03 amd64 /usr/sbin/cron[30103]: (root) CMD (/sbin/evlogmgr -c 'age > *30d')
Sep 20 12:46:44 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 12:48:43 amd64 sshd[6609]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:54:41 amd64 sshd[6704]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 sshd[6907]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64242
Sep 20 15:43:44 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:13 amd64 sshd[10102]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63375
Sep 20 16:37:13 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 01:00:01 amd64 /usr/sbin/cron[17055]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 /usr/sbin/cron[17878]: (root) CMD (/sbin/evlogmgr -c 'age > *30d')
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:53:39 amd64 sshd[31269]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[46741]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 22 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 /usr/sbin/cron[54991]: (root) CMD (/sbin/evlogmgr -c 'age > *30d')
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 18:04:33 amd64 /usr/sbin/cron[24739]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 /usr/sbin/cron[54991]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 /usr/sbin/cron[25555]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 18:04:33 amd64 /usr/sbin/cron[6515]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 18:04:34 amd64 sshd[6606]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62093
Sep 23 18:04:34 amd64 /usr/sbin/cron[12436]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:03 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd[13351]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainted kernel.
Sep 24 15:42:07 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 /usr/sbin/cron[13351]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:03 amd64 sshd[12931]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[6621]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 /usr/sbin/cron[14841]: (root) CMD (/sbin/evlogmgr -c 'age > *30d')
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd[20211]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 10:59:47 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 11:59:25 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:07:15 amd64 sshd[11554]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:07:15 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 14:08:33 amd64 sshd[11630]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63709
Sep 25 15:25:33 amd64 sshd[12930]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62778
```

Processes (1)

Process details:

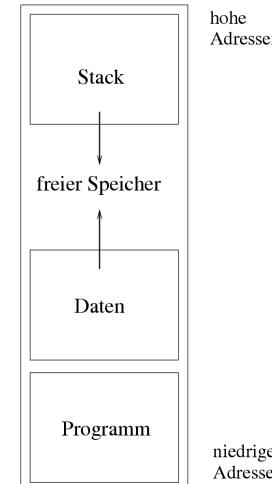
- unique address space
- executable program (code, text)
- data (variables)
- program counter (instruction counter)
- stack and stack pointer
- contents of hardware registers (process context)

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 19

Processes (2)



Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

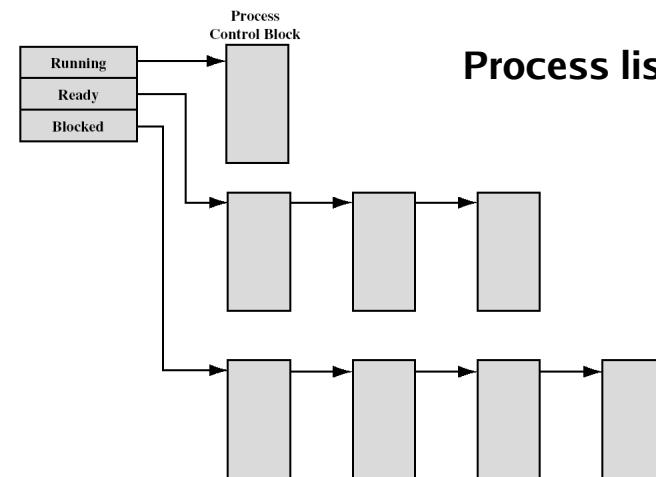
2. Prozesse – Folie 20

Processes (3)

States

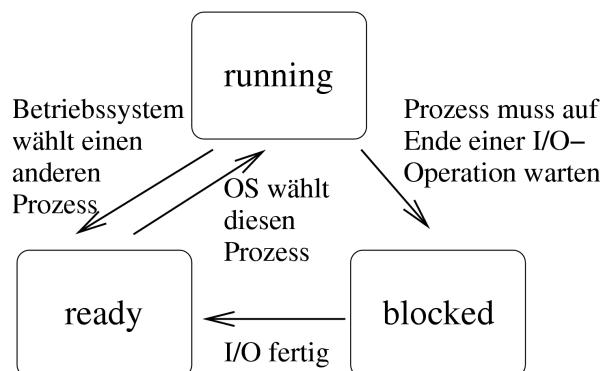
- **running**: active right now
- **ready**: would like to run
- **blocked / waiting**: waits for I/O
- **suspended**: ... by the user
- **sleeping**: waits for signal (IPC)
- **swapped**: data not in main memory (RAM)

Processes (5)



Processes (4)

State transitions



Processes (6)

Hierarchies

- processes create one another
- Creator is called **parent process**, the other one **child process**
- Children are autonomous (i. e.: have their own address space etc.)
- After process termination: return value goes to parent process

Threads (1)

What is a thread?

- Thread of activity in a process
- one of several
- shared access to process data
- but: program counter, stack, stack pointer, hardware registers separate to each thread
- process scheduler handles threads – or doesn't (kernel vs. user level threads)

Threads (3): Examples

Two different threads of activity: Complex calculations with user requests

Without threads:

```
while (1) {  
    calculate_a_bit ();  
    if user_input (x) {  
        process_input (x)  
    }  
}
```

Threads (2)

Why Threads?

- multi processor system: several threads can be active truly simultaneously
- If one thread is blocked by I/O, the others can continue working
- If a program logically consists of parallel tasks, programming with threads is simpler

Threads (4): Examples

Complex calculations with user requests

With threads:

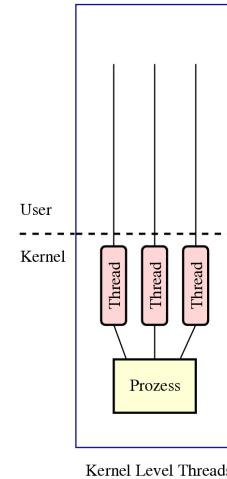
T1:	T2:
<pre>while (1) { calculate_all (); }</pre>	<pre>while(1) { if user_input (x) { process_input (x); } }</pre>

Threads (5): Examples

Server process handling lots of requests

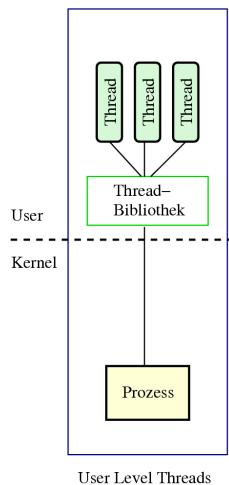
- process opens port
- For each incoming connection: Create new thread which deals with this request
- After termination of connection: destroy thread
- Advantage: No need for process creation (by the operating system!)

Kernel Level Threads



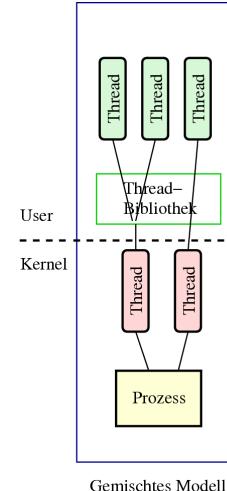
- OS knows threads
- OS handles threads:
 - creation, destruction
 - scheduling
- I/O of one thread won't block the others
- Time consuming: context switch between threads as complex as that between processes

User Level Threads



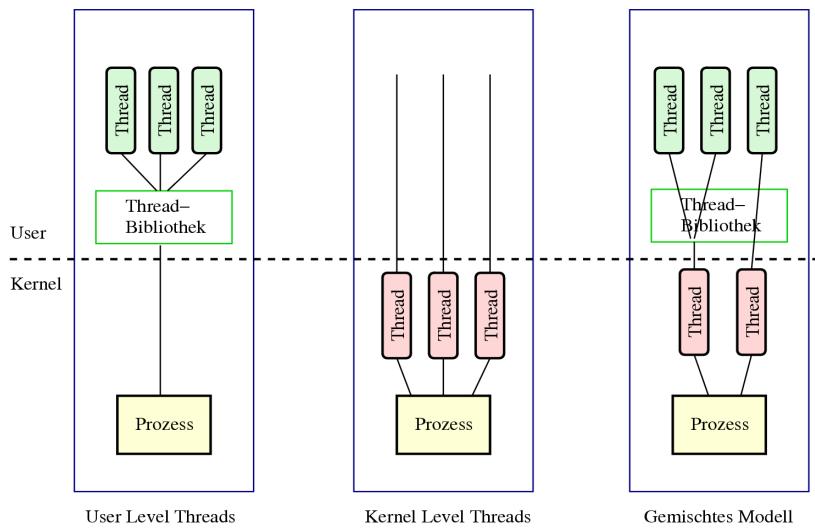
- OS has no thread concept, only deals with processes
- program uses thread library which is responsible for:
 - creation, destruction
 - scheduling
- If a thread waits for I/O, the whole process waits
- otherwise pretty efficient

Mixed Threads



- Combine both kinds of threads
- KL threads + UL threads
- thread library distributes UL threads onto KL threads
- e.g. I/O parts in one KL thread
- Best of both worlds:
 - I/O blocks only one KL thread
 - switching between UL threads is efficient
- SMP: use several CPUs

Thread types, overview



Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 33

Practice II: Threads

```

Sep 19 14:20:18 amd64 sshd[20494]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61557
Sep 19 19:27:41 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 /usr/sbin/cron[29278]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 20 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 sshd[101031]: (root) CMD (/sbin/evlogmgr -c "age > 30d")
Sep 20 02:00:01 amd64 sshd[6516]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62004
Sep 20 12:46:44 amd64 sshd[6516]: STATS: dropped 0
Sep 20 12:48:43 amd64 sshd[165091]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62105
Sep 20 12:48:44 amd64 sshd[66941]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62514
Sep 20 15:27:35 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 evlogmgr[7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 evlogmgr[7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd[10140]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63546
Sep 20 21:11:00:01 amd64 /usr/sbin/cron[170555]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 21 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 01:00:01 amd64 /usr/sbin/cron[124836]: (root) CMD (/sbin/evlogmgr -c "age > 30d")
Sep 21 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd[110881]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63397
Sep 21 17:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 18:43:26 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 sshd[112081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64391
Sep 21 19:43:26 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 /usr/sbin/cron[46741]: (root) CMD (/sbin/evlogmgr -- "severity=DEBUG")
Sep 22 01:00:01 amd64 /usr/sbin/cron[124836]: (root) CMD (/sbin/evlogmgr -c "age > 30d")
Sep 22 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[124739]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 23 01:00:01 amd64 /usr/sbin/cron[124739]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[124739]: STATS: dropped 0
Sep 23 01:00:01 amd64 /usr/sbin/cron[124739]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd[65541]: Accepted publickey for esser from ::ffff:192.168.1.5 port 59771 ssh2
Sep 23 18:04:05 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 01:00:01 amd64 /usr/sbin/cron[124836]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 24 01:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 /usr/sbin/cron[132531]: (root) CMD (/sbin/evlogmgr -c "age > 30d")
Sep 24 02:00:01 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 13:15:48 amd64 sshd[123971]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64456
Sep 24 13:49:08 amd64 sshd[231971]: Accepted rsa for esser from ::ffff:87.234.201.207 port 61330
Sep 24 13:49:08 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd[193991]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62566
Sep 24 20:25:31 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 01:00:01 amd64 /usr/sbin/cron[14841]: (root) CMD (/sbin/evlogmgr -c "severity=DEBUG")
Sep 25 01:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 /usr/sbin/cron[14841]: (root) CMD (/sbin/evlogmgr -c "age > 30d")
Sep 25 02:00:02 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd[8889]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64183
Sep 25 10:59:25 amd64 sshd[8889]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd[89211]: Accepted rsa for esser from ::ffff:87.234.201.207 port 64253
Sep 25 11:30:02 amd64 sshd[13972]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62029
Sep 25 11:59:25 amd64 syslog-ng[7653]: STATS: dropped 0
Sep 25 11:59:25 amd64 sshd[115841]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62822
Sep 25 14:05:37 amd64 sshd[115861]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62951
Sep 25 14:06:10 amd64 sshd[116081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 63392
Sep 25 14:07:17 amd64 sshd[116081]: Accepted rsa for esser from ::ffff:87.234.201.207 port 62776

```

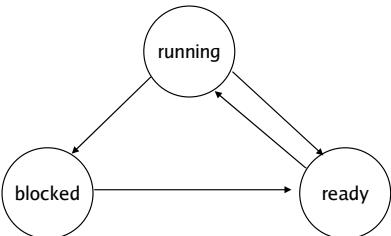
Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 35

Thread states

- Process states suspended, sleeping, swapped etc. not used for threads
- Only three thread states



Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 34

Threads in C programs (1)

Linux: pthread library (POSIX Threads)

	thread	process
Create	<code>pthread_create()</code>	<code>fork()</code>
Await termination	<code>pthread_join()</code>	<code>wait()</code>

- include library headers
`#include <pthread.h>`
- compile:
`gcc -lpthread -o prog prog.c`

Hans-Georg Eßer, FH München

Betriebssysteme I, WS 2006/07

2. Prozesse – Folie 36

Threads in C programs (2)

- Process: `fork()` creates identical copy / clone of calling process.
- Thread: `pthread_create()` gets function to be executed in new thread as argument.
- Thread: `pthread_join()` waits for a *specific* thread.

Threads in C programs (3)

1. define thread function

```
void *thread_function(void *arg) {  
    ...  
    return ...;  
}
```

2. create thread:

```
pthread_t thread;  
  
if ( pthread_create( &thread, NULL,  
                    thread_function, NULL ) ) {  
    printf("Error while creating thread.\n");  
    abort();  
}
```

Threads in C programs (4)

```
#include <pthread.h>  
#include <stdlib.h>  
#include <unistd.h>  
  
void *thread_function1(void *arg) {  
    int i;  
    for ( i=0; i<10; i++ ) {  
        printf("Thread 1 says Hi!\n");  
        sleep(1);  
    }  
    return NULL;  
}  
  
void *thread_function2(void *arg) {  
    int i;  
    for ( i=0; i<10; i++ ) {  
        printf("Thread 2 says Hallo!\n");  
        sleep(1);  
    }  
    return NULL;  
}  
  
int main(void) {  
    pthread_t mythread1;  
    pthread_t mythread2;  
  
    if ( pthread_create( &mythread1, NULL,  
                        thread_function1, NULL ) ) {  
        printf("Error creating thread.");  
        abort();  
    }  
  
    sleep(5);  
  
    if ( pthread_create( &mythread2, NULL,  
                        thread_function2, NULL ) ) {  
        printf("Error creating thread.");  
        abort();  
    }  
  
    sleep(5);  
  
    printf("still here...\n");  
  
    if ( pthread_join ( mythread1, NULL ) ) {  
        printf("Error while joining.");  
        abort();  
    }  
  
    printf("Thread 1 is gone.\n");  
  
    if ( pthread_join ( mythread2, NULL ) ) {  
        printf("Error while joining .");  
        abort();  
    }  
  
    printf("Thread 2 is gone.\n");  
    exit(0);  
}
```

Threads in C programs (5)

No „parent“ or „child threads“

- POSIX threads have no concept of relatedness as processes do (parent / child process)
- Waiting for a thread requires thread variable: `pthread_join (thread, ...)`

Threads in C programs (6)

Process with several threads:

- only one entry in process list
- status: „l“, multi-threaded
- Command `ps -eLf` gives thread information
 - NLWP: number of light weight processes
 - LWP: thread ID

```
> ps auxw | grep thread
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
esser     12022  0.0  0.0  17976   436 pts/15    S1+  22:58   0:00 ./thread

> ps -eLf | grep thread
UID      PID  PPID  LWP  C  NLWP STIME TTY      TIME CMD
esser    12166  4031  12166  0    3 23:01 pts/15  00:00:00 ./thread1
esser    12166  4031  12167  0    3 23:01 pts/15  00:00:00 ./thread1
esser    12166  4031  12177  0    3 23:01 pts/15  00:00:00 ./thread1
```