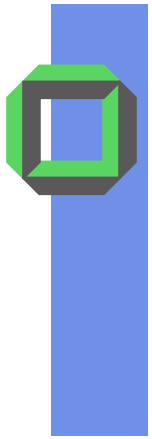
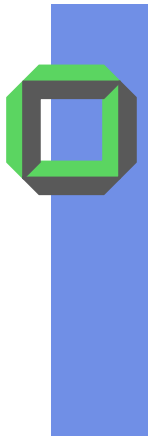


System Calls via Libraries

- For some higher programming languages like C, ADA, FORTRAN etc. the POSIX standard defines processor- and OS-independent library calls
- For C, in the C-standard library /lib/libc there is a library C-function with the same name as its system call counterpart:
  - **`man -S s x`** delivers a description of library function **`x`**
    - Needed includes
    - Allowed parameter and return values
    - Short semantic description
    - Similar or corresponding other library functions
  - **`s`** corresponds to the section of the POSIX manual
    - **`s=2`** means system calls
    - **`s=3`** means library calls

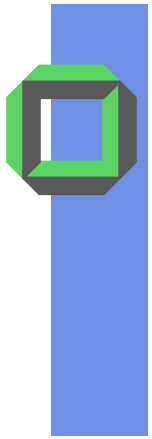
© 2007 Universität Karlsruhe(TH), System Architecture Group



Context of a Process

- HW-context
  - Activity context
    - Contents of registers, i.e. IP, SP, ...
  - AS context
    - Address regions, e.g. heap or stack
    - Opened files
    - Reserved resources
- SW context
  - Process identifier (PID)
  - Process state
  - Accumulated resource usage

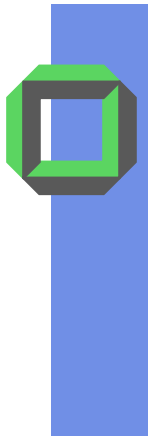
© 2007 Universität Karlsruhe(TH), System Architecture Group



Process Management

- OS needs to know how many and what processes are currently in the system
- Each process needs a data structure PCB helping to identify it
- A PCB contains all information about the process that is needed to control and schedule this process
- The set of all PCBs can be aggregated into a
  - Table
  - List
  - any other fitting data structure

© 2007 Universität Karlsruhe(TH), System Architecture Group



Parallelized Web Server

- Problem analysis:
  - A request for a specific html-page has arrived
  - html-page must be delivered from disk or from cache
  - *What will happen if another request has arrived while the web server is still busy with the previous request?*
- At least two alternatives to design a web server

1. A process (i.e. only one thread)
   - No parallelization possible, bad performance if first request induces one or more blocking system calls
2. A task with multiple worker threads
   - Even in case of blocking system calls, the next ready worker can already work on the new request whilst the first worker waits for the result of the blocking system call

© 2007 Universität Karlsruhe(TH), System Architecture Group