

BunnyKernel

¯ \ \ / ¯
 \ V /
 / . . \
 = \ T / =
 / ^ \
 { } / \ \ / / \
 \ " " /
jgs (_____ / ^ \ _____)

DePeSlVe

Matúš Dekánek, Tomáš Petrušek, Ľuboš Slovák, Ján Veselý

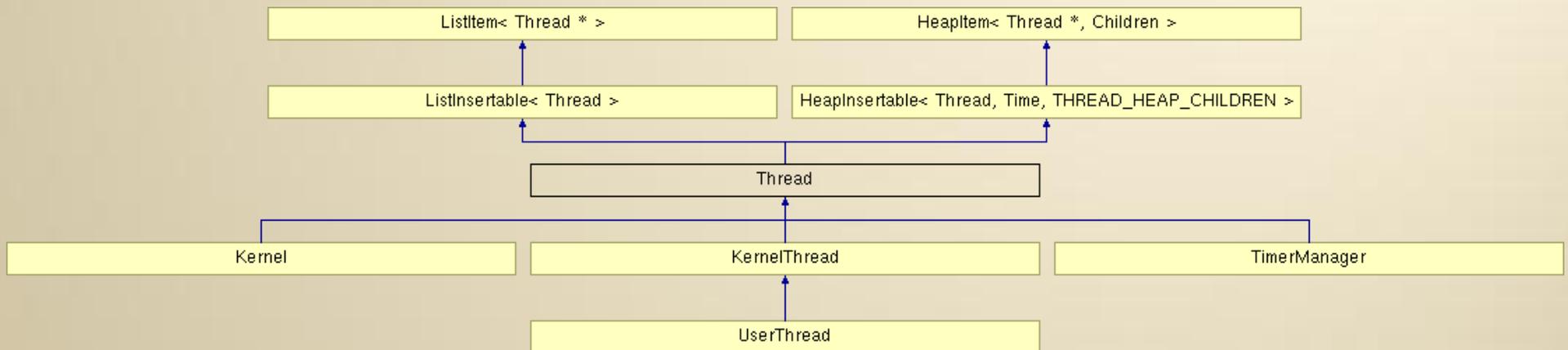
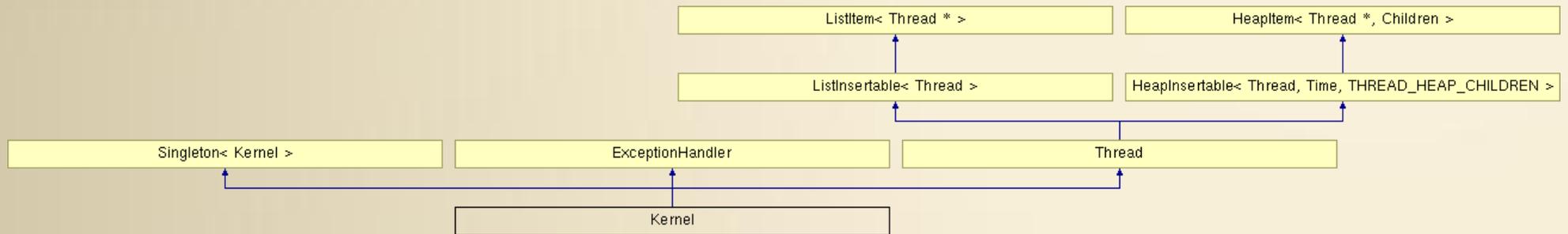
Convenience of work

- Makefile does not need file list
- Using Doxygen for online documentation generation
- Mailing list
- C++
- Separately tested structures

C++

- Advantages:
 - OOP
 - Inheritance
 - Interface classes
 - Templates
- Disadvantages
 - Additional C++ symbols
 - Not all C++ features (Exceptions, Global Object Constructors)
 - API ? (C-style)

Inheritance



Threads and Scheduling

- Scheduler
 - List of threads
 - OOP: Intelligent Scheduler vs. Intelligent Threads
 - Tickless
 - Using Time Slots
 - Emulated hardware
- Timer
 - Heap: 4 – children
 - Rescheduling as Timer Event

Tickless

Tickless base data

	# interrupts	#events	Avg CPU idle residency (uS)
With ticks	2002	59.59	651
Tickless	118	60.60	10161

System activity during idle with and without periodic ticks, with HZ=1000*

Is this the best we can do? Can we further reduce number of events and interrupts?



* Intel® Core™ 2 Duo based system (2 CPU cores)



Kernel as the Init/Idle Thread

- Intelligent thread, may run without scheduler
- Initializing code runs in the thread
 - After initialization it is used as the Idle thread
- Inspired by Linux kernel

Memory Management

- Frame Allocator
 - Multiple frame sizes
 - All levels in one bitmap – speeds up lookups
- Optimized Bitmap
 - Uses 4-byte masks vs. checking every bit

Elements	100 elements	1.000 elements	10.000 elements	100.000 elements	1.000.000 elements
Speed up	2x	5x	13x	30x	32x

Memory Allocator

- Multiple Strategies
 - Same structures for all
 - Runtime strategy change
 - Core class shared by KernelMemoryAllocator and UserMemoryAllocator
- RAM size detection:
 - Detected by moving the mapping of the first page

VMA

- Using Splay Tree structure
 - Utilizing Locality of Reference
- Variable Page Sizes
 - Pros:
 - Fewer refills
 - Smaller TLB utilization
 - Cons:
 - Problems with size borders, address alignment
 - VMA Splitting/Joining

Syscalls and Processes

- Generic Syscall macro
- Arguments in registers vs. Arguments on stack
- Registers: A0 – A3, V0
- Syscall number stored in the instruction call
- Process:
 - Just a special thread
 - Adds few more resources

TarFS

- “no need to implement FS, just linear list of files with separating structures”
- TarFS:
 - Pros:
 - Same block size as msim disk device
 - Easy disk image creation
 - Complete structure supporting, Unix features (symlinks, hardlinks, rights, etc.)
 - Cons
 - Weird format (size in 11 ASCII coded octal digits,...)

Thank you for your attention!

```
Terminal
developer@DePeS1Ve ~/0Sy/bunny $ msim
HELLO WORLD!
| \   / | | \   / | | \   / | | \   / |
 \ | _ | /   \ | _ | /   \ | _ | /   \ | _ | /
 / . . \   / . . \   / . . \   / . . \
 = \ _ T _ / = = \ _ T _ / = = \ _ T _ / = = \ _ T _ / =
 (>o<)   (>o<)   (>o<)   (>o<)
Running on 4 processors
Warning: It's nice to have more processors, but we currently support only one.
Running on MIPS R4000 revision 0.0
Detecting freq...4.2 MHz
Probing memory range...OK
Detected 16 MB of accessible memory
Kernel ends at: 0x80027670.
Stacks(0x1000) end at: 0x80028670.
Mounting root fs...█
```