

System Programming

Prof. Dr. Antônio Augusto Fröhlich

`guto@lisha.ufsc.br`

`http://www.lisha.ufsc.br/~guto`

Sep 2006

Motivation

“We now have JAVA! Why would we still care about system programming?”

- Someone has to care for your JVM and JRE!
- Someone has to care for **compilers'** code generators!
- Someone has to care for **operating systems!**
- Several have to care for **embedded systems!**

“Who are these poor people?”

- We, **Computer Scientists!**

Programming: System X Application



■ Application

- User interface
- High-level prog. lang.
 - High-level of abstraction
 - High productivity
 - Complex runtime
 - Automatic memory management
 - Active objects
 - Component repositories
- World of **JAVA**, PHP, Python

■ System

- **Hardware** abstraction
- Low-level prog. lang.
 - Resource-constrained environments
 - Little runtime overhead
 - Small runtime library
 - Direct access to hardware
- World of **C**, C++

System Programming and Compilers



- High-level programming language constructs get translated into machine language
 - **Code generators** (back-end)
 - Architecture-specific translation
 - **Optimizations**
 - Memory access (e.g. alignment, cache, TLBs)
 - Flow control (e.g. pre-fetch, reordering)
 - CISC operations

System Programming and Operating Systems



- Programming languages abstract **hardware entities** that must be managed by the OS
 - Process context exchange
 - Memory management
 - I/O
- **Runtime libraries** for HLPL
 - Some complex operations are not directly translated into machine code by the compiler
 - A RTL function is called instead

System Programming and Embedded Systems



“Hardware and software which forms a component of some larger system and which is expected to function without human intervention.”

(Foldoc)

- You don't see the computer!
 - Typical embedded system
 - Microcontroller with software in PROM
 - Starts running some specific-purpose application program as soon as it is turned on and will not stop until it is turned off (if ever)

What Exactly are we Talking About?

Where are the processors?

(Tennenhouse, CACM 43(5):44)

