



MICHIGAN STATE
UNIVERSITY

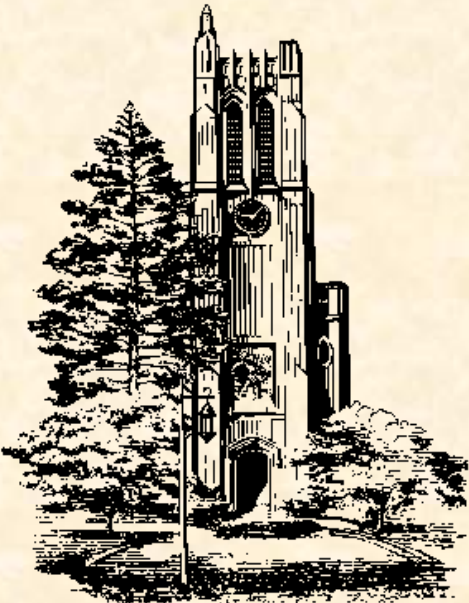
Technical Specification and Schedule: Processor Folding for Linux on Power

Team 3: IBM – Processor Folding on Linux
CSE 498, Collaborative Design

Mark Bernum
Ian De Silva
Josh Detwiler

Department of Computer Science and Engineering
Michigan State University

Fall 2007





Project Overview



- Enabling processor folding on Linux
- IBM System p machines on POWER6 architecture will be used
- Further enhancements for fine tuning processor folding by the user



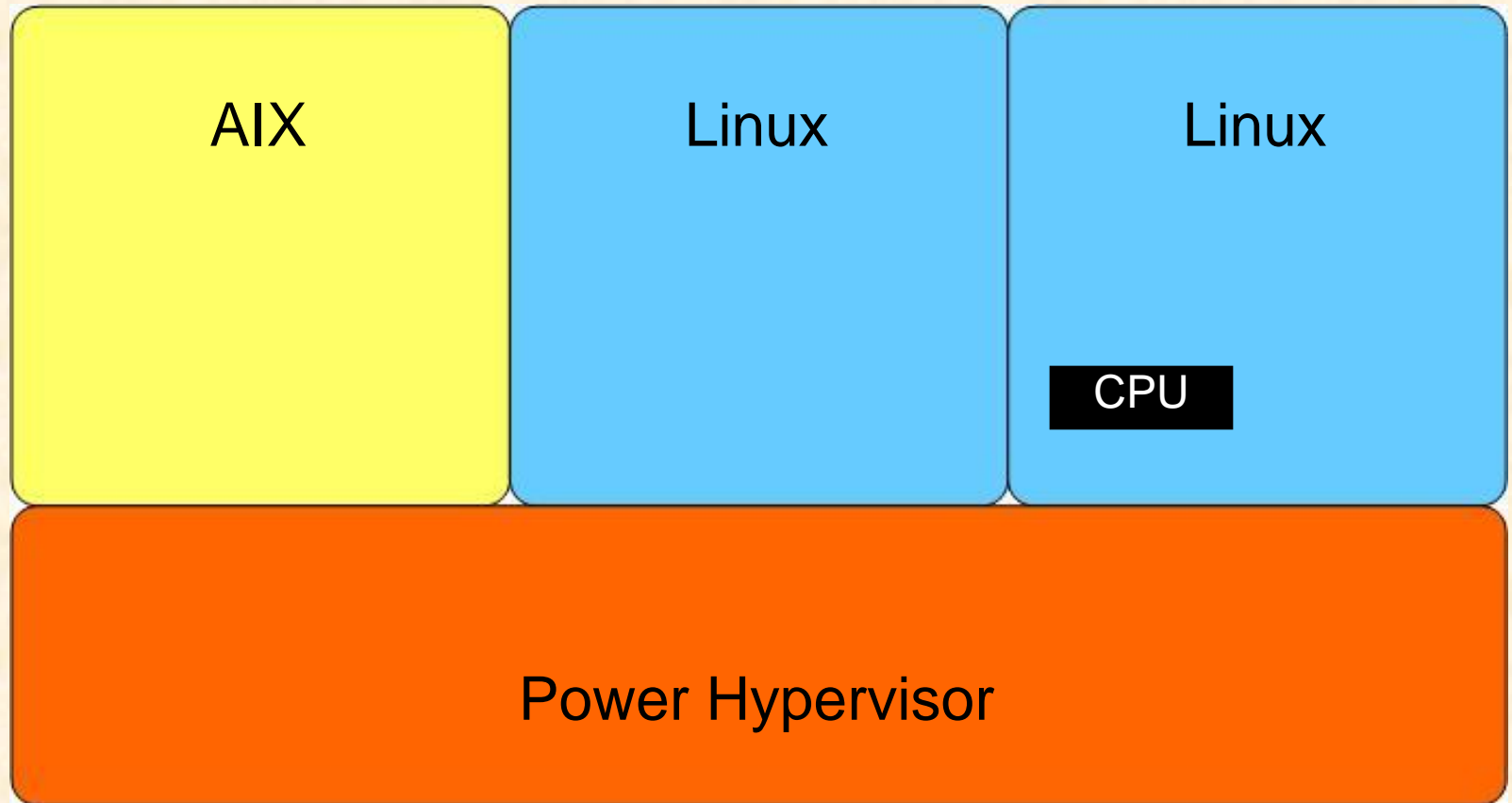
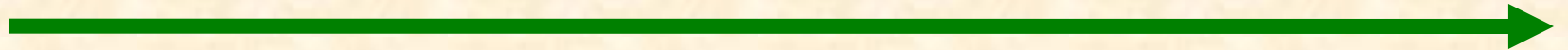
Functional Specifications



- IBM's AIX operating system currently folds the processes of a CPU onto other processors in order to do a dedicated donation to the Power Hypervisor. The Hypervisor can then allow another OS to use that processor. We want to mimic this behaviour on Linux.
- When the utilization of a processor falls below a certain threshold, we want to fold the active tasks onto another processor so we can take the original and do a dedicated donation to another task.
- Existing command line tools will be used to view statistics related to the processor folding. New tools will be created to fine tune the folding process to the user's wishes.

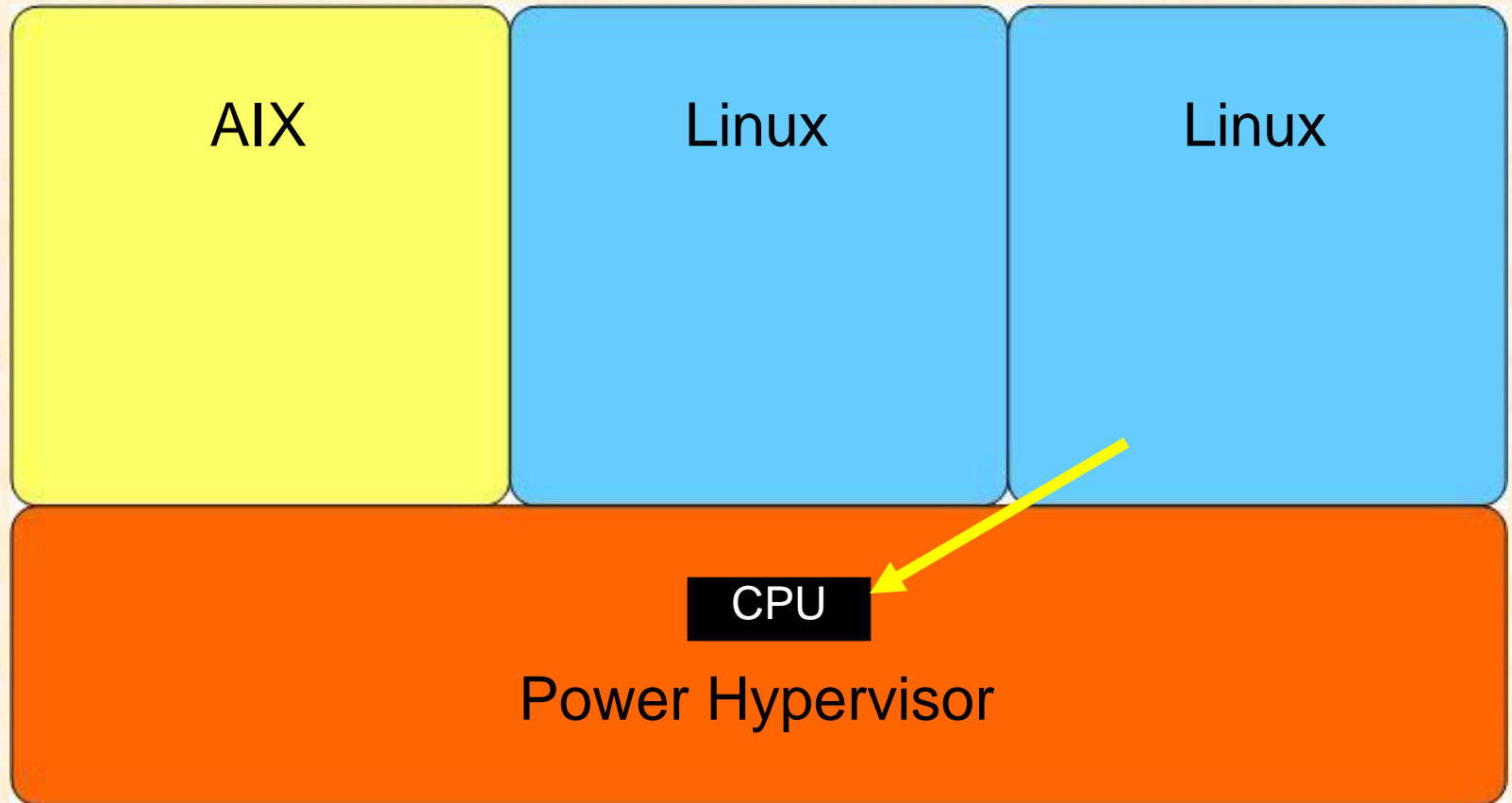
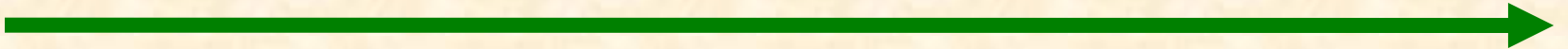


Processor Folding Illustrated



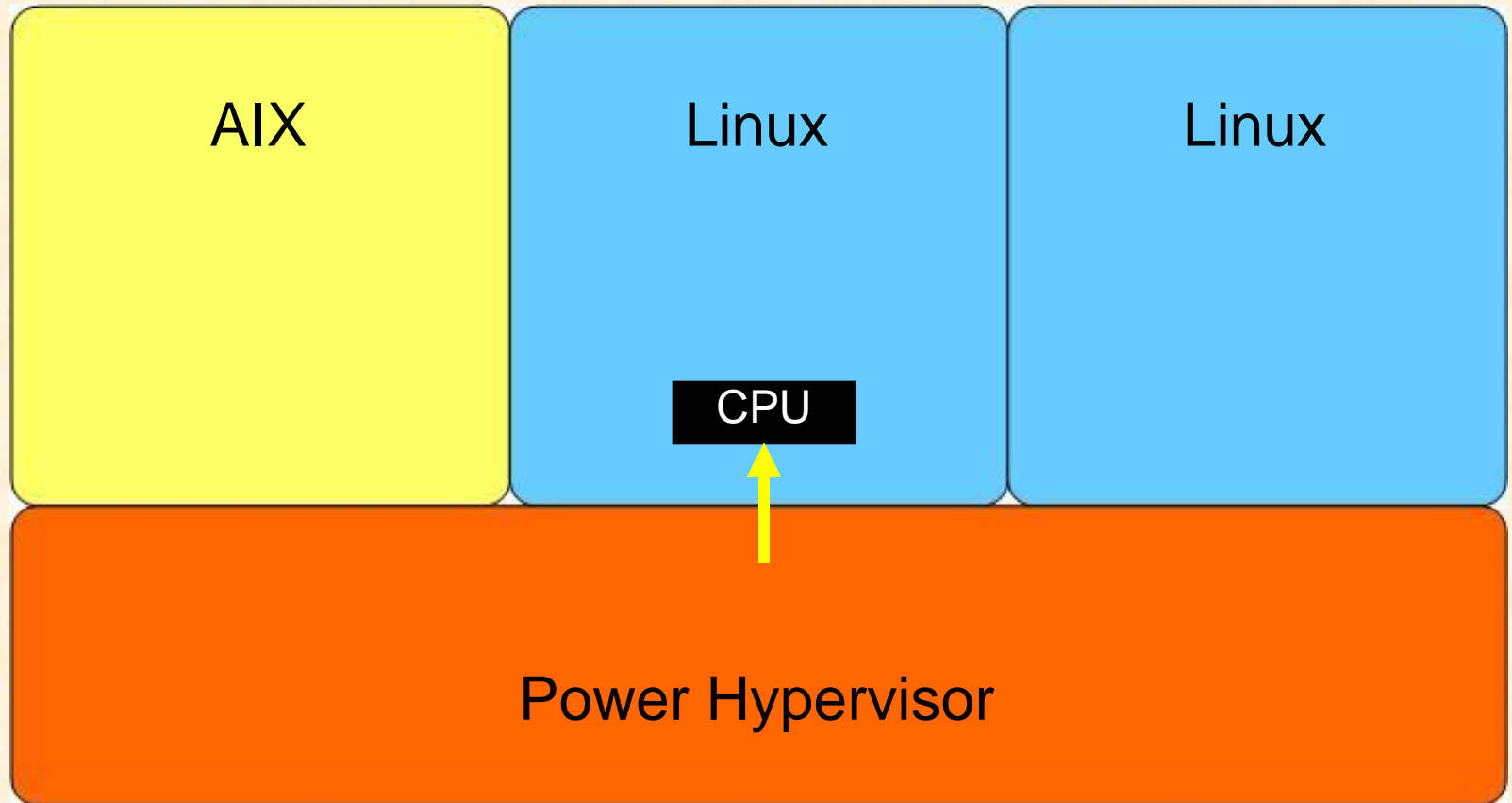
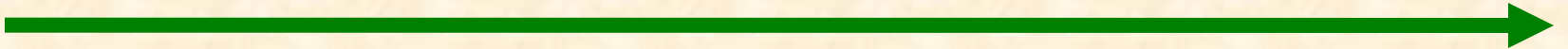


Processor Folding Illustrated





Processor Folding Illustrated



Toolset Requirements



- A command line interface is needed to view information related to the CPU levels. This program can be used to verify visually if the folding is occurring and the associated statistics.
- Command line program to turn processor folding on or off is required as well as a boot menu option.
- Command line program is required to change the threshold at which processor folding will occur. In other words, when the processor utilization falls below that set threshold, the kernel will begin the process to fold the processor.



System Components

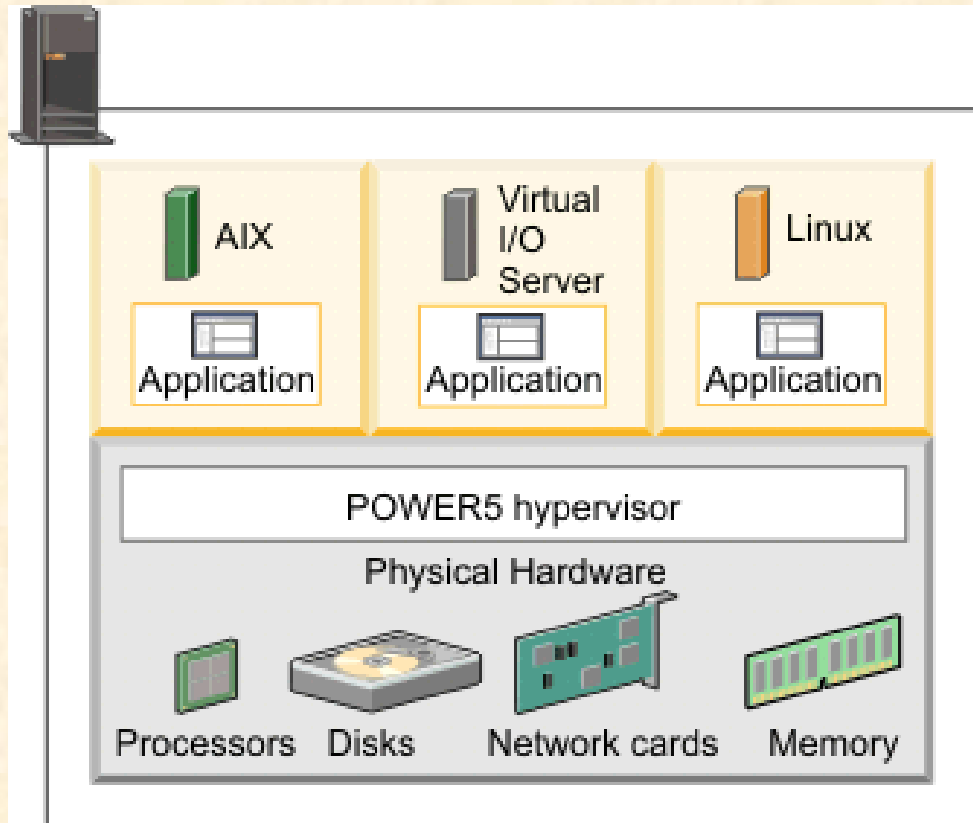


- Hardware Platforms
 - IBM System p machine with multiple logical partitions.
- Software Platforms / Technologies
 - Eclipse 3.2 will be used for editing the C programming language code.
 - Linux Kernel 2.6.23-rc6 will be modified.
 - SVN for version control (we may switch to git).



Architecture Illustrated

- The following shows the architecture of a possible IBM System P machine.




Risks



- Linux Kernel Knowledge
 - No team member assigned to this project has previously worked with the Linux kernel
- Valid Documentation
 - Obtaining quality documentation is a difficult chore. The kernel is under active development and is changing every day.
- Program Interactions with the Kernel
 - Figuring out how to get the statistics for processor utilization and enabling/disabling folding from the command line and boot menus will be tricky.



Project Schedule



1. Design Completion
 - a) Complete design in pseudo code
 - b) Date: 9-17-07
2. Prototype Complete
 - a) Create working user interface
 - b) Date: 10-08-07
3. Code Completion
 - a) Finish implementation of design
 - b) Date: 10-22-07
4. Test Completion
 - a) Finish all test cases
 - b) Date: 11-16-07



Project Schedule

5. Documentation Completion

- a) Complete user docs and man pages
- b) Date: 11-30-07

6. Deliverables due

- a) All portions of project complete
- b) Date: 12-05-07



Questions?



Mark Bernum – bernumma@msu.edu

Ian De Silva – desilva1@msu.edu

Josh Detwiler – detwile3@msu.edu

Special Thanks to:

IBM Corporation
Dr. Wayne Dyksen
Matt Luciw