Transparent Process Migration
in the Beowulf System

Jeffrey Moyer (phro@wpi.edu)

May 4, 1999

– Typeset by FoilTEX –

Overview

- Background Information
- Process Migration
- Distributing Load
- Implementation
- Applications
- Future Work

Background Information

- What is Beowulf?
- Okay, so where did it come from?
- Who uses this stuff, anyway?
- Terms: Node, Computer, Machine, Workstation, System

Process Migration

- Definition: The checkpointing of a process at some point either before or during its execution, and subsequent (re)starting of that process on a different node.

- Checkpointing
  - Process State
    * CPU Registers
    * Memory Contents
    * Open Files
    * Shared Library Info.
    * System V IPC Structures (pipes/shm/sem)
    * Open network sockets

Distributing the Load

Process migration makes little sense unless the process gains system resources in its new environment.

- Load balancing vs. Load sharing
- Load sharing algorithm
Transfer	Location
Selection	Information

Information Policy

- Responsible for determining
  - What information do we collect?
  - Where do we get this information from?
  - When do we collect information?
- Example: Broadcast state information.

Transfer Policy

Responsible for determining when to migrate a process.

- Threshold Value
- Sender Initiated
- Receiver Initiated

Selection Policy

Responsible for determining which process to migrate.

- Preemptive vs. Non-preemptive Migration

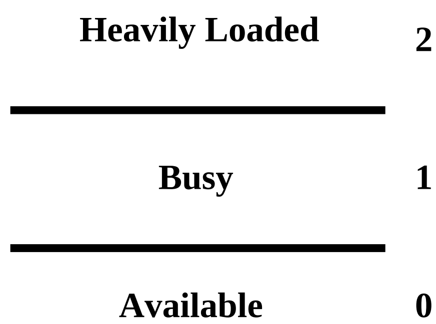
Location Policy

Responsible for determining to which host the task is transferred.

- Based on data provided by Information Policy

Multi-Leader Load Sharing Algorithm

Wills-Finkel, “A Scalable Approach to Load Sharing in the Presence of Multicast”



- Multicast Groups
  - ◇ Leader
  - ◇ Available

- Threshold Values (012,022,...)

Implementation

- Information Policy:
  - System load average
  - 5 second intervals
  - Each node keeps this information

- Transfer Policy:
  - Sender Initiated

Implementation - 2

- Selection Policy:
  - Migratable Flag
  - Preemptive & Non-Preemptive
  - K.I.S.S.

- Location Policy:
  - Left up to the Leader

Migration Diagram

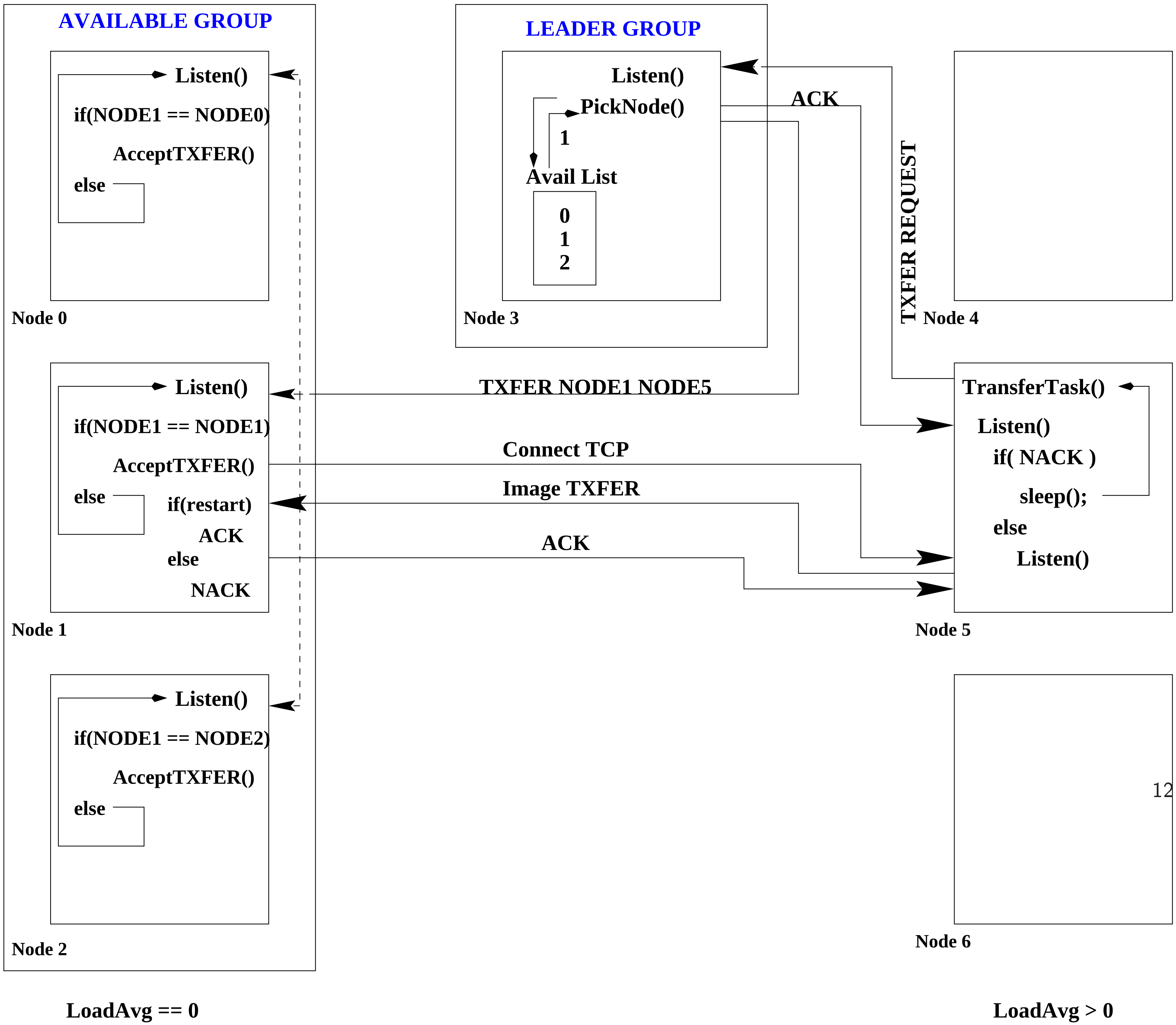


Figure 1: Flow of Execution for Migration Request.

Applications

- Out-of-box Utilization
- Fork-parallel Applications
- Build-server
- High Availability

Future Work

- Checkpoint Network Sockets
- High Availability
- Redirect Application Output
- Memory Ushering

Acknowledgements

- Jon 'maddog' Hall
- Eduardo Pinheiro
- Dr. David Finkel