

Mapping Simulators to Architecture: A Workload Characterization Study

Lee F. Finley

October 28, 2005

What is the Problem?

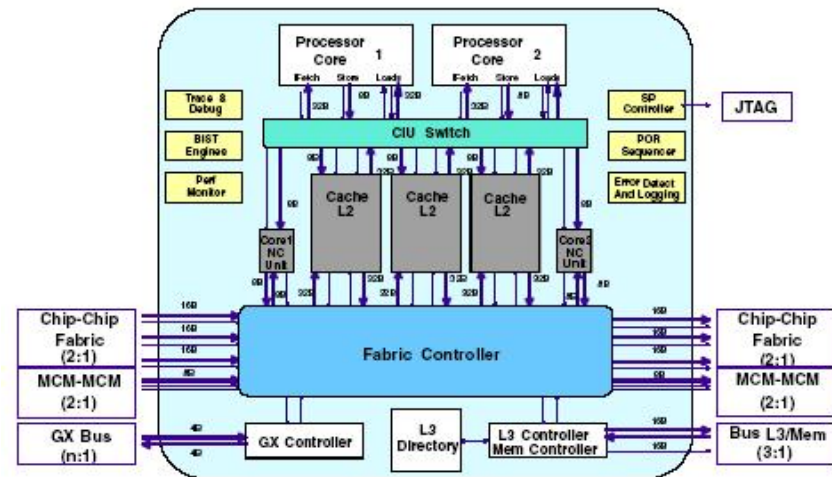
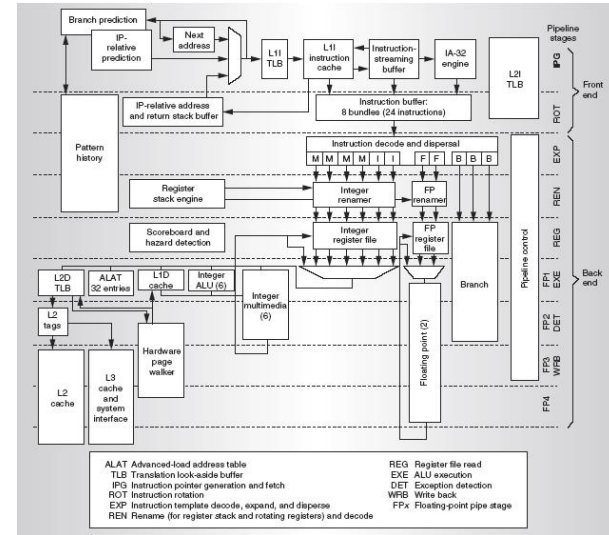
- Who is using simulator?
- Common problem (Execution Time)
- Increasing Details
- Current research/Techniques

What I am doing to HELP?

- Scope (What I am look at?)
 - μ Arch (*Itanium 2, POWER 4*)
 - Simulator (*sim-outorder*)
 - SPEC CPU2000

```

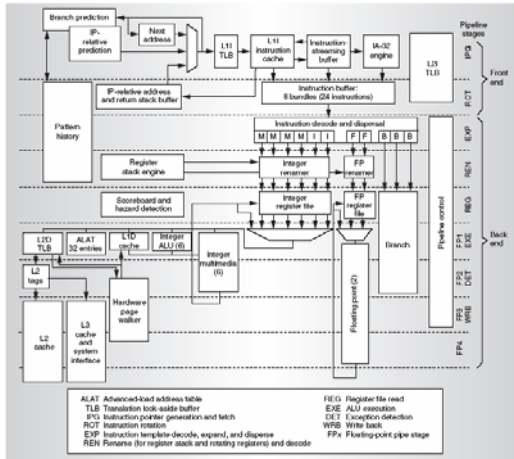
ruu_init();
for(;;){
    ruu_commit();
    ruu_writeback();
    lsq_refresh();
    ruu_issue();
    ruu_dispatch();
    ruu_fetch();
}
    
```



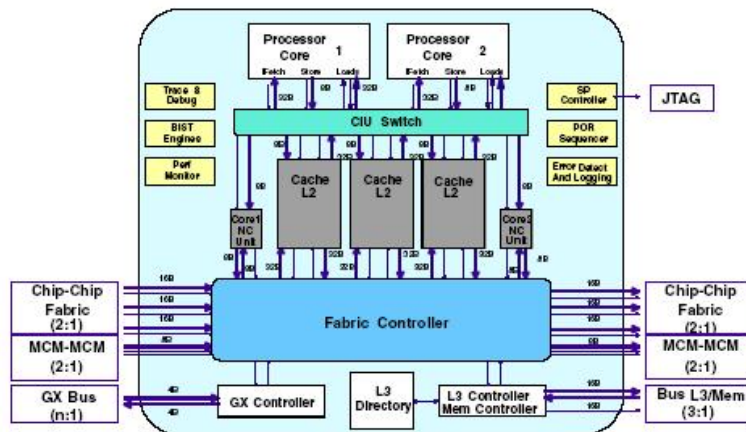
Diagnosing the Problem

- Performance Monitoring Counters
 - Events Captured
 - Branch, Cache
 - Stalls (Define)
 - Data Cache/FPU (L1D)
 - Register Stack Engine (RSE)
 - Branch Misprediction (FLUSH)
 - Front End (FE)
 - Execution Unit (EXE)

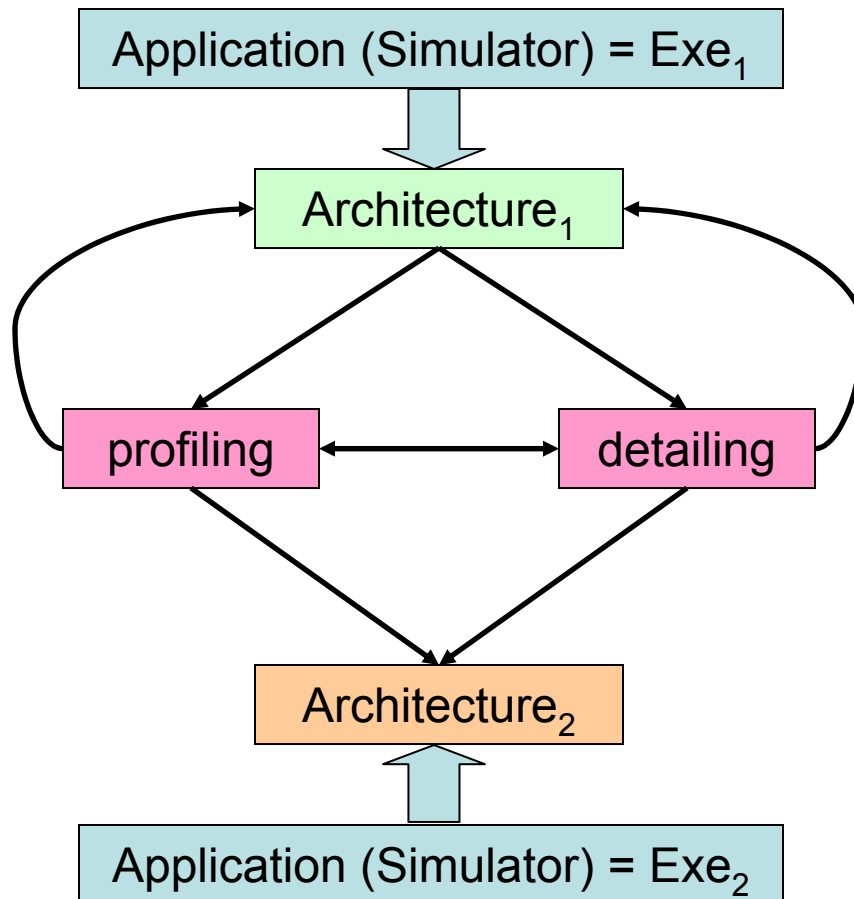
Fixing the Problem



- Architecture Mapping
 - Define
 - Goal (Execution Time)
 - μ Arch Understanding



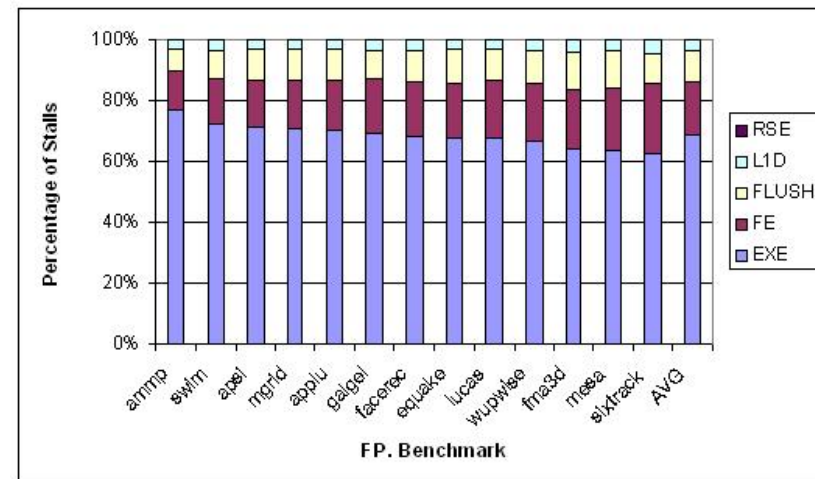
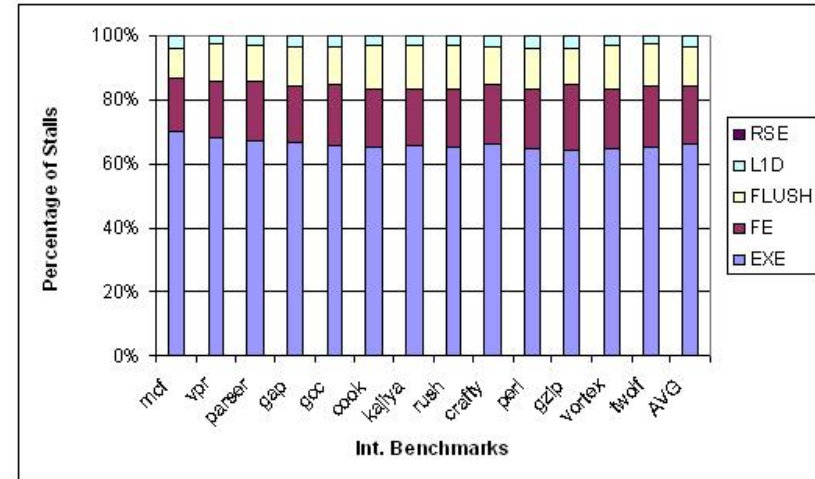
Recap the Method



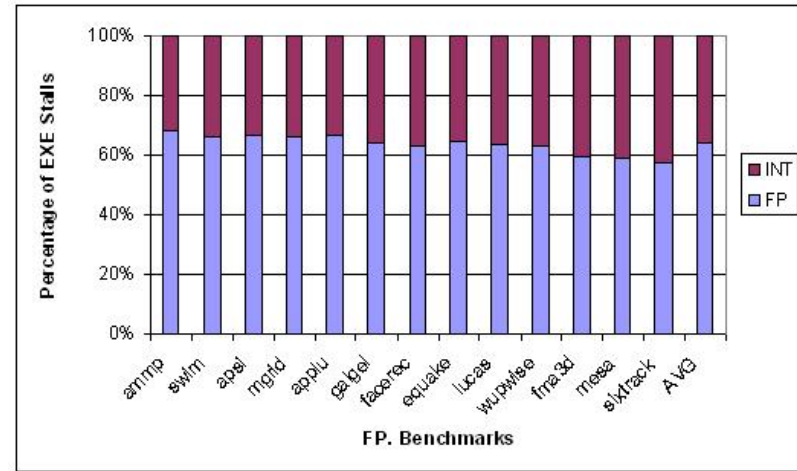
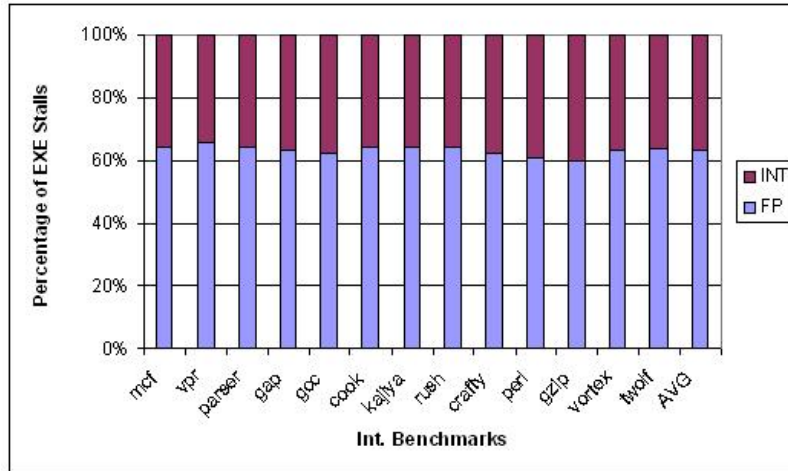
Exe₂ < Exe₁ 😊

Simulator Stalls

- What are they?
- Execution Units

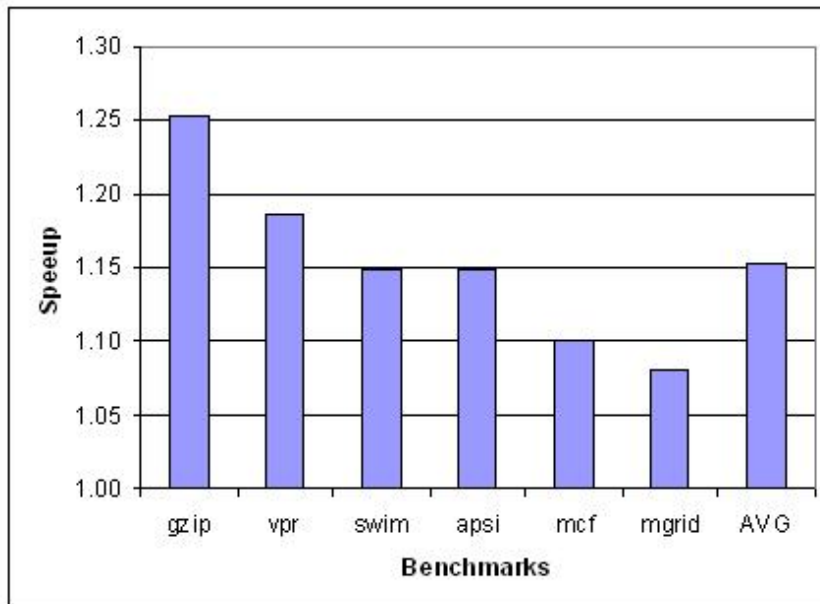


Execution Unit Stalls



- INT 37%
- FP 63%
 - 43% exe. total stalls

Mapping the Simulator



- Moddi3 & FP exe. Stalls
- FP Execution Unit
- Execution Time—**SPEEDUP**
- μ Arch. Differences

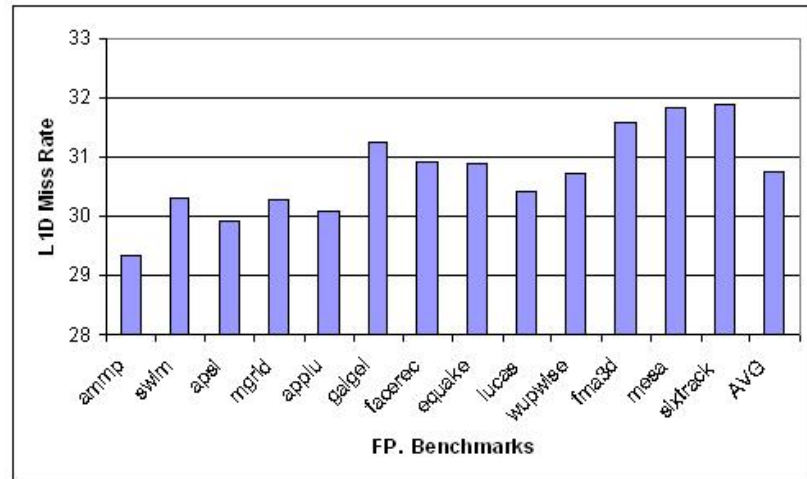
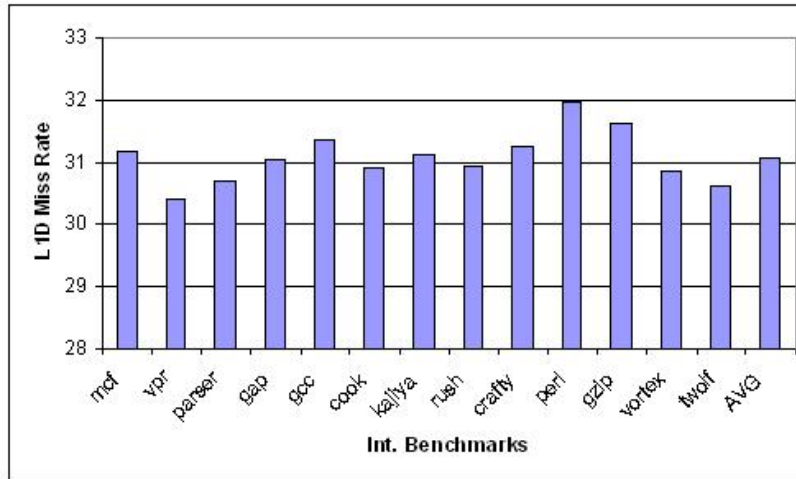
Applicability/ Future Work

- Generality
- Validation
- Portability

Thank You!

Backup

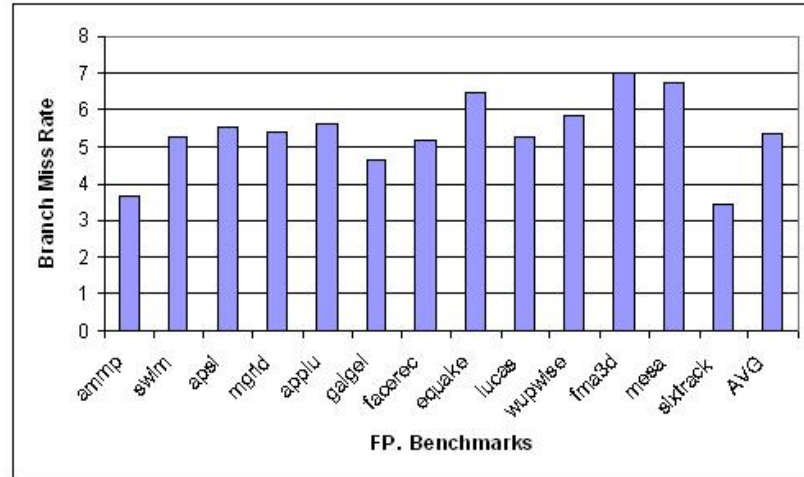
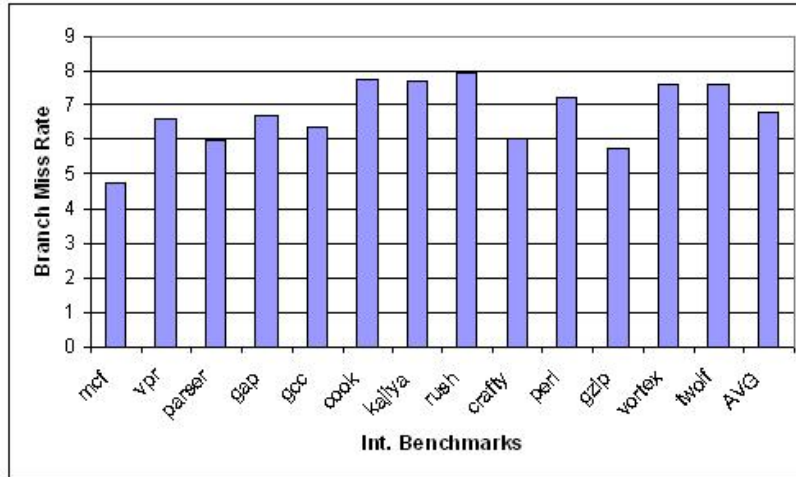
Benchmark Independence



- Define
- Why a concern?
- Determining
 - Avg
 - Span

- L1D
 - Avg
 - INT 31.1%
 - FP 30.7%
 - Span
 - INT 1.6%
 - FP 2.5%

Cont.



- Branch
 - Avg
 - INT 6.8%
 - FP 5.4%
 - Span
 - 3%