

Visualizing Performance Data for Multicore Systems

High Productivity Tools, IBM
Carole-Jean Wu

Why Multi-Core?

- ▶ Efficiency
 - Power/Performance Ratio
- ▶ Functionality
 - Core Synthesis (Homogeneous/Heterogeneous Cores)



Why Multi-Core?

- ▶ Efficiency
 - Power/Performance Ratio
- ▶ Functionality
 - Core Synthesis (Homogeneous/Heterogeneous Cores)

**Lower Power
Consumption**



**Better
Performance**



Challenges

- ▶ Thread Creation for Optimum Parallelism
- ▶ Efficient Thread Orchestration
- ▶ Software & Hardware Interactions
- ▶ Hardware Utilization



Challenges

- ▶ Thread Creation for Optimum Parallelism
- ▶ Efficient Thread Orchestration
- ▶ Software & Hardware Interactions
- ▶ Hardware Utilization

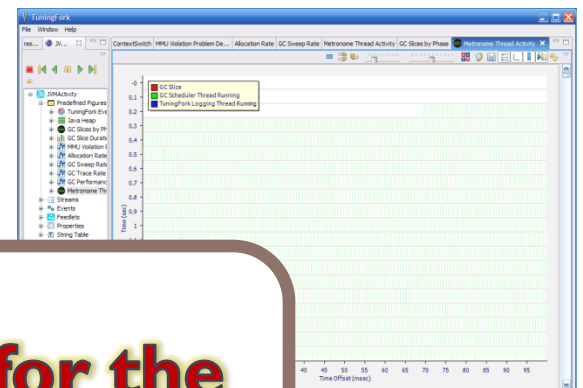
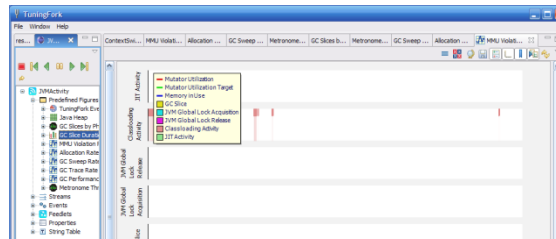
Visualization Tools



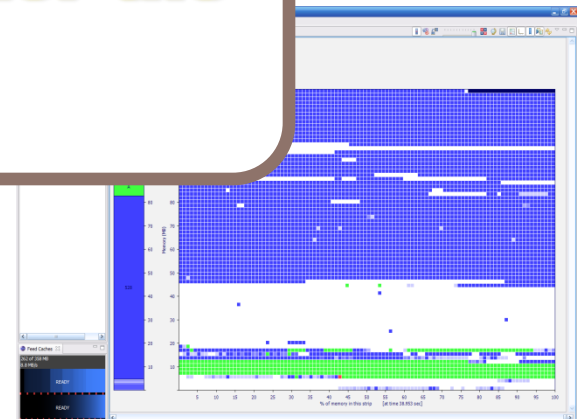
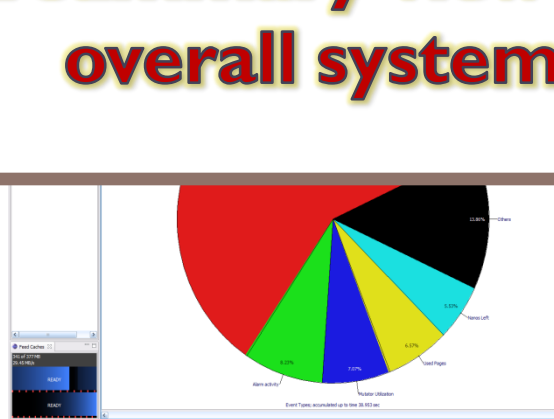
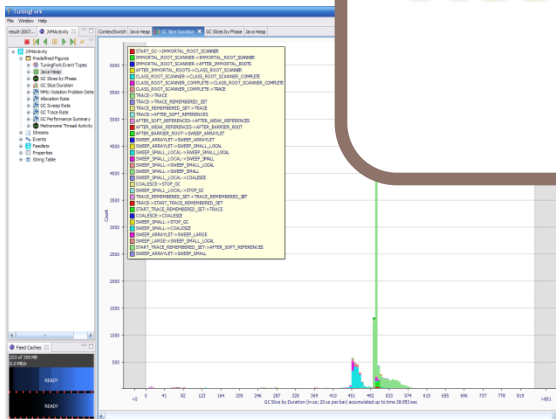
IBM TuningFork

► Real time visualization & performance analysis tool

- Time Series
- Oscilloscope
- Histogram
- Pie Chart
- Heap Memory



Need summary view for the overall system



Timeslice View

- ▶ **Hierarchy for Multi-core Systems**
 - ▶ Processor/Core/Hardware Threads
- ▶ System Activity
 - ▶ Cache Misses (at a time instant)
 - ▶ Blocking/Synchronization
 - ▶ Thermal/Power Consumptions/HotSpots
- ▶ Summary View over time
 - ▶ Average Cache Miss Rate
 - ▶ Overall Processor/Core Utilization



Timeslice View

- ▶ System Hierarchy
 - ▶ Processor/Core/Hardware Threads
- ▶ **Systems Activity**
 - ▶ Cache Misses (at a time instant)
 - ▶ Blocking/Synchronization
 - ▶ Thermal/Power Consumptions/HotSpots
- ▶ Summary View over time
 - ▶ Average Cache Miss Rate
 - ▶ Overall Processor/Core Utilization

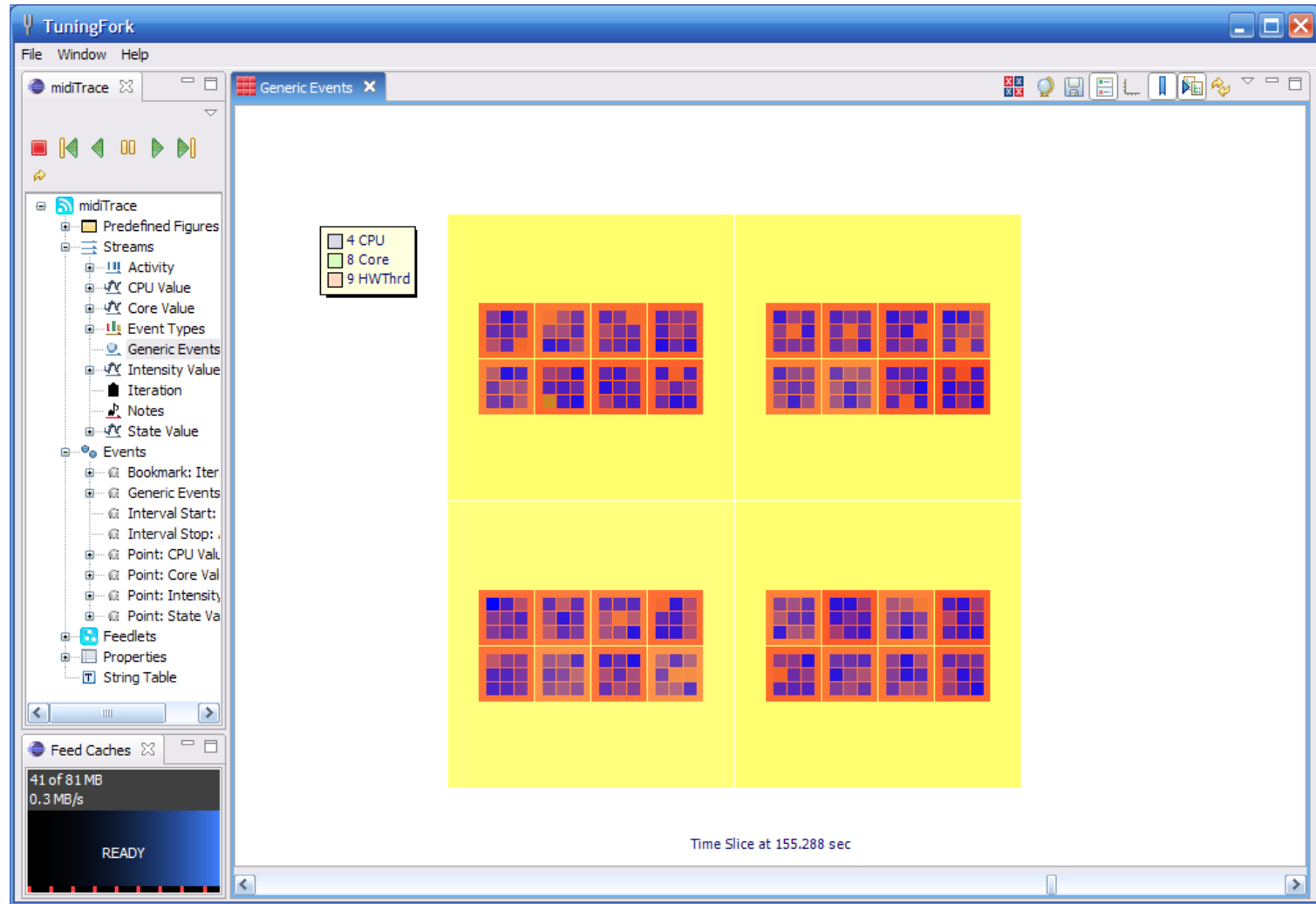


Timeslice View

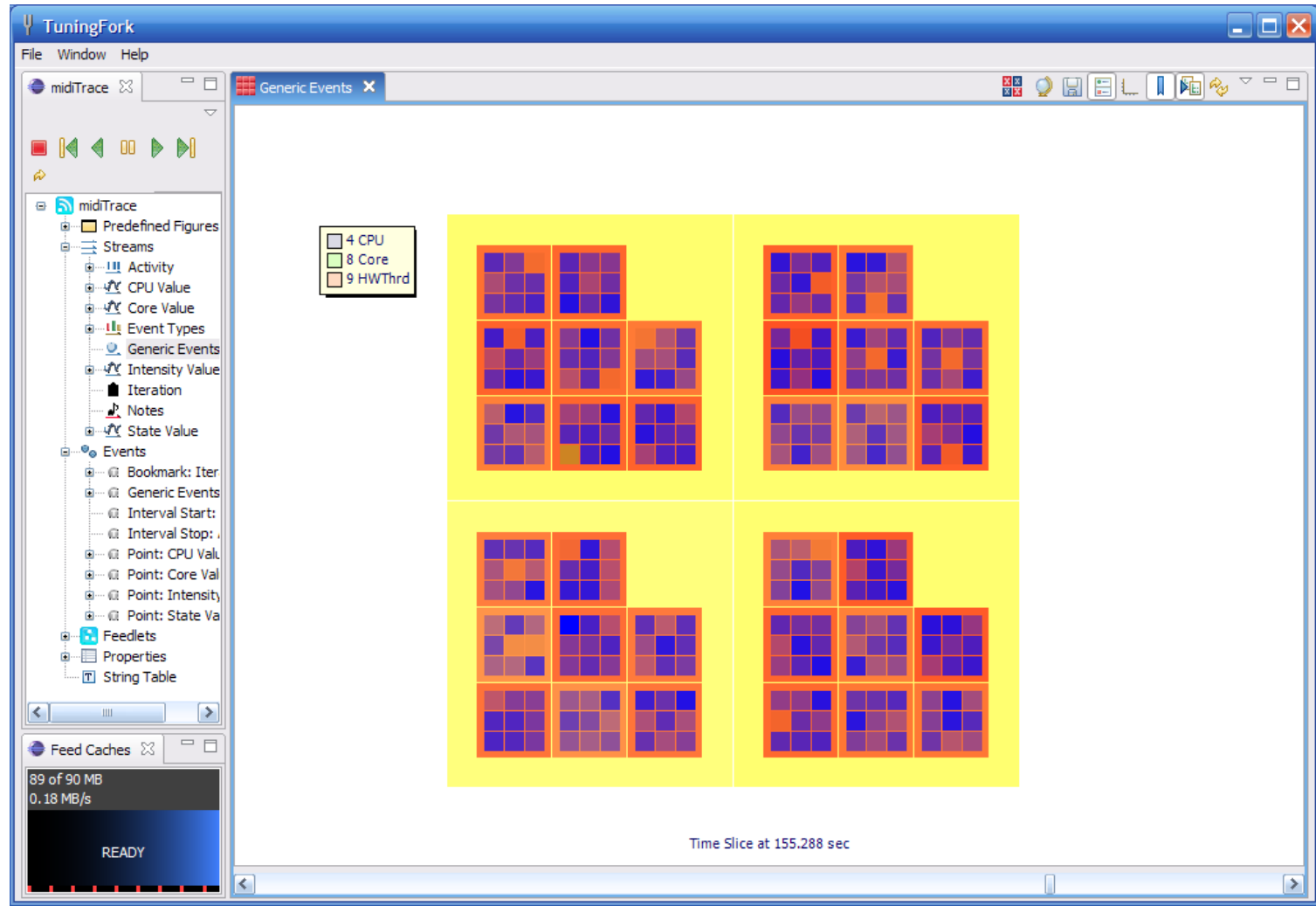
- ▶ System Hierarchy
 - ▶ Processor/Core/Hardware Threads
- ▶ System Activity
 - ▶ Cache Misses (at a time instant)
 - ▶ Blocking/Synchronization
 - ▶ Thermal/Power Consumptions/HotSpots
- ▶ **Summary View over Time**
 - ▶ Average Cache Miss Rate
 - ▶ Overall Processor/Core Utilization



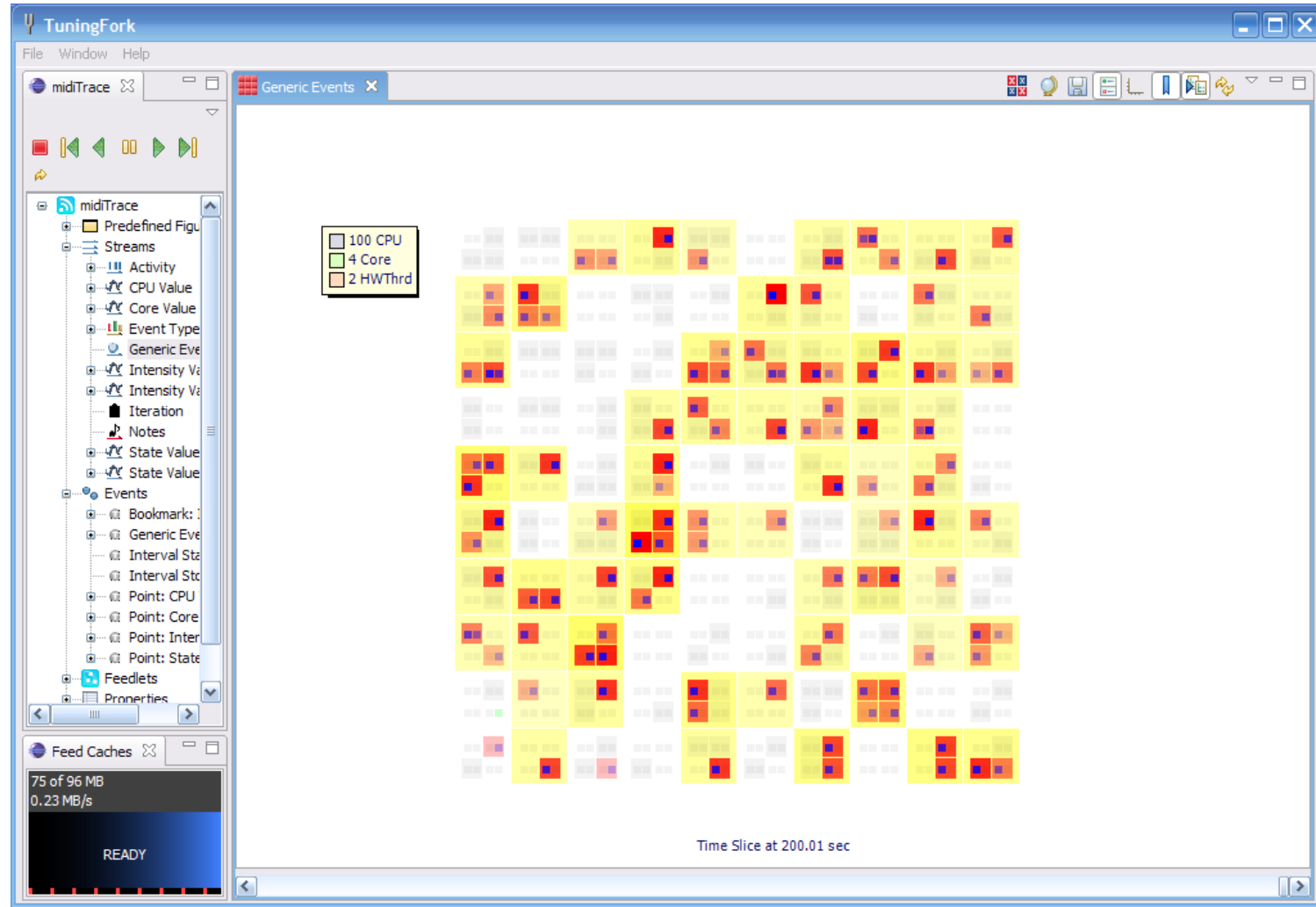
Synthetic Trace (4x8x9)



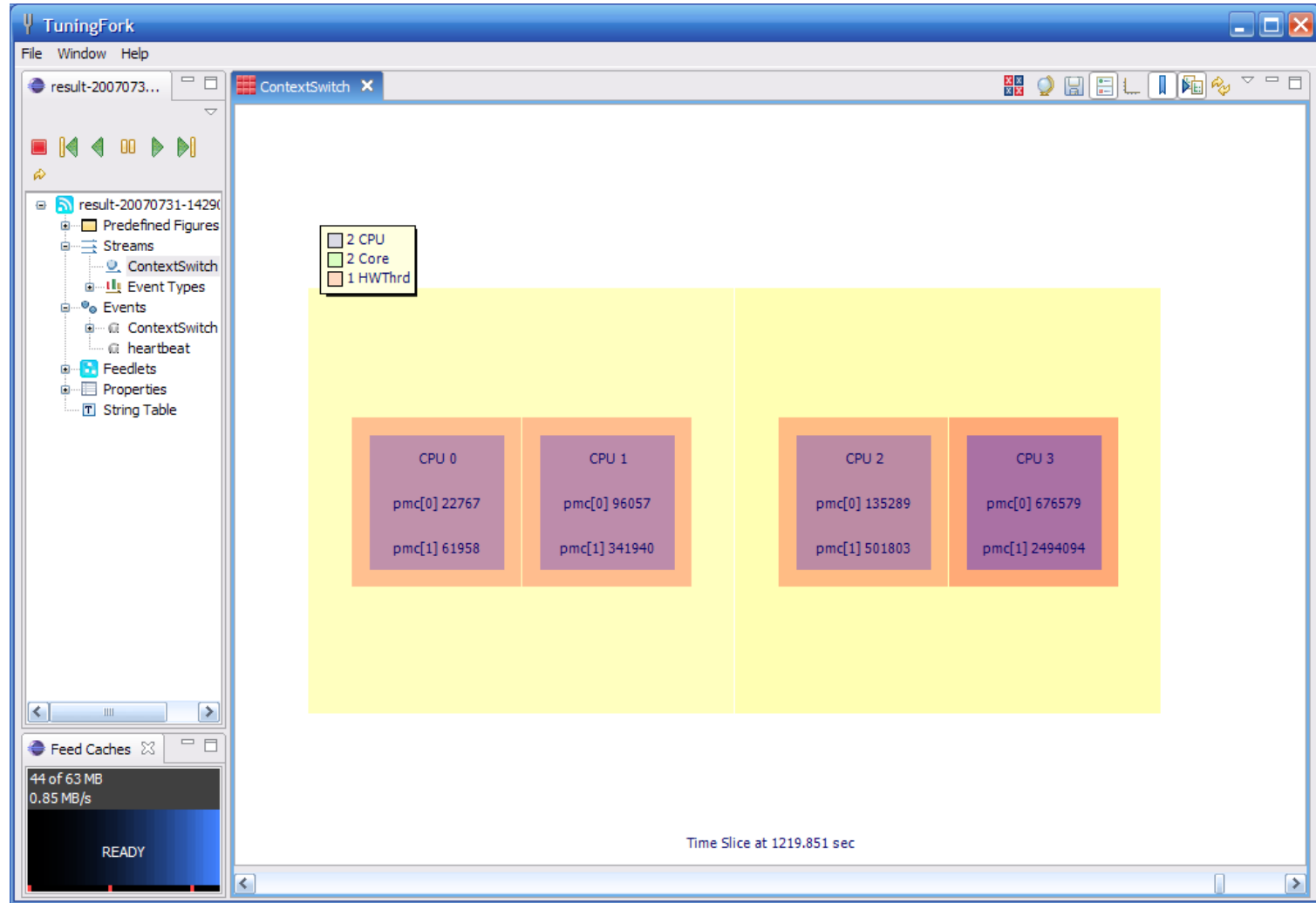
Synthetic Trace (4x8x9)



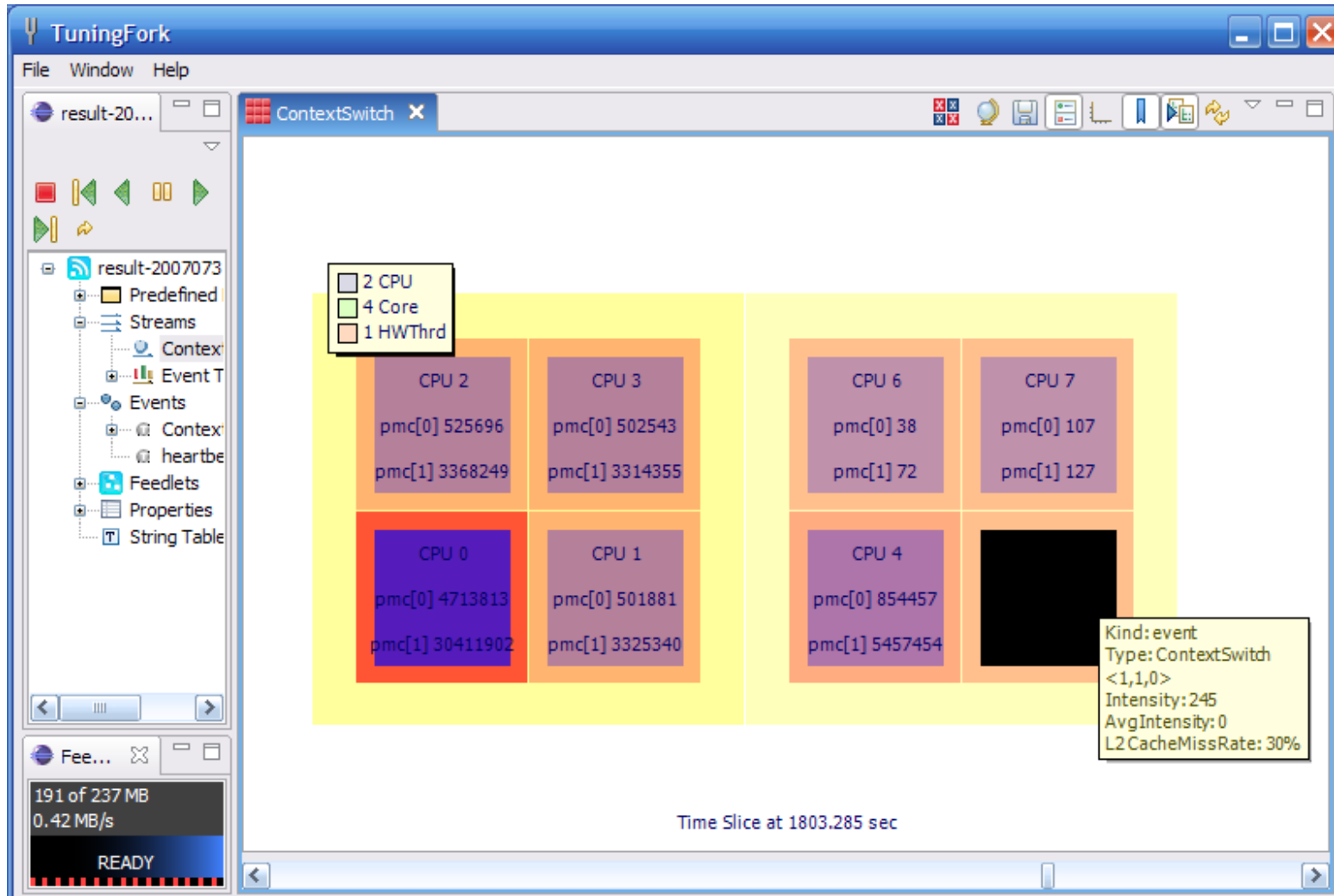
Synthetic Trace (100x4x2) for Large-scale Systems



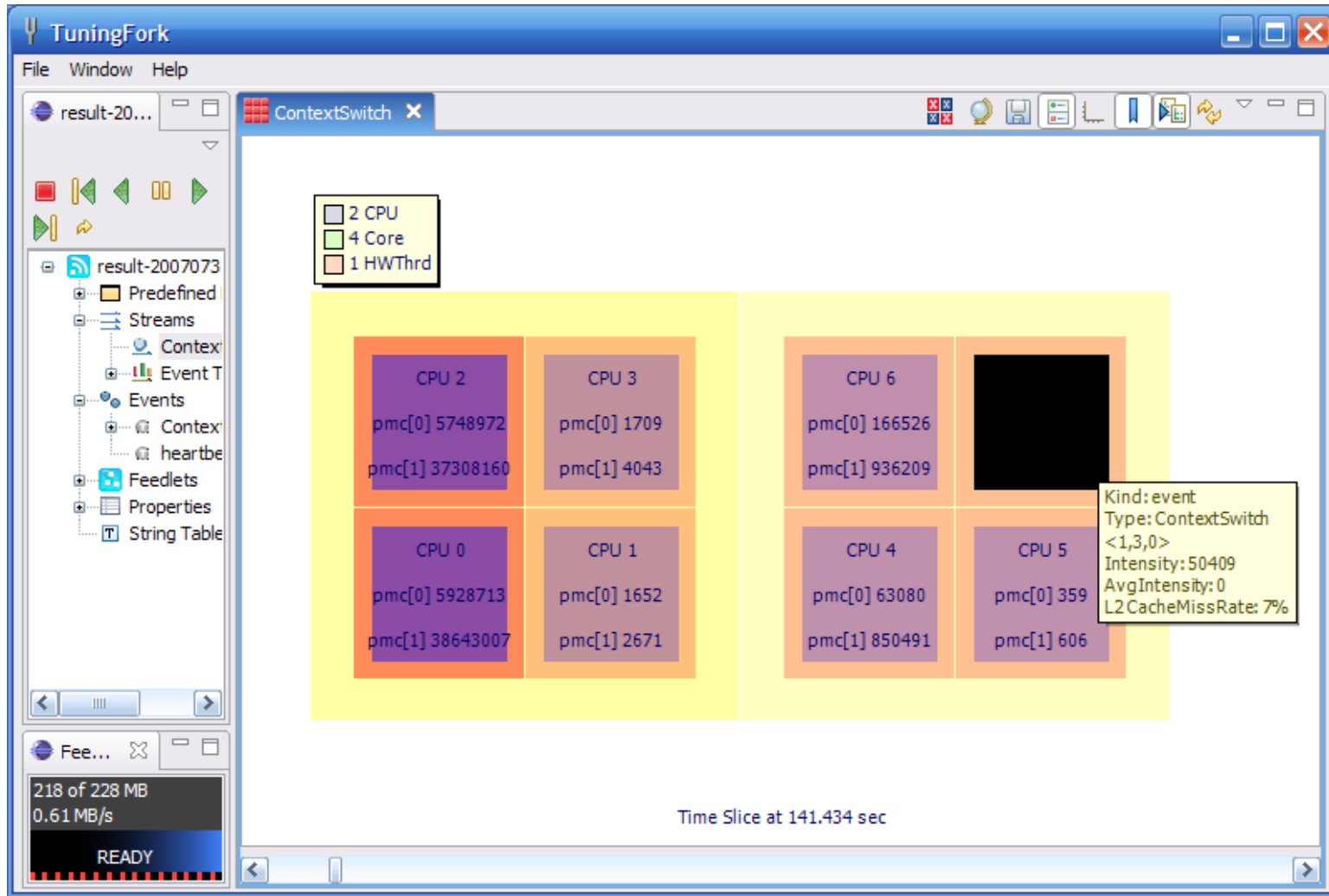
JBB trace on 2-Xeon(LV DualCore)



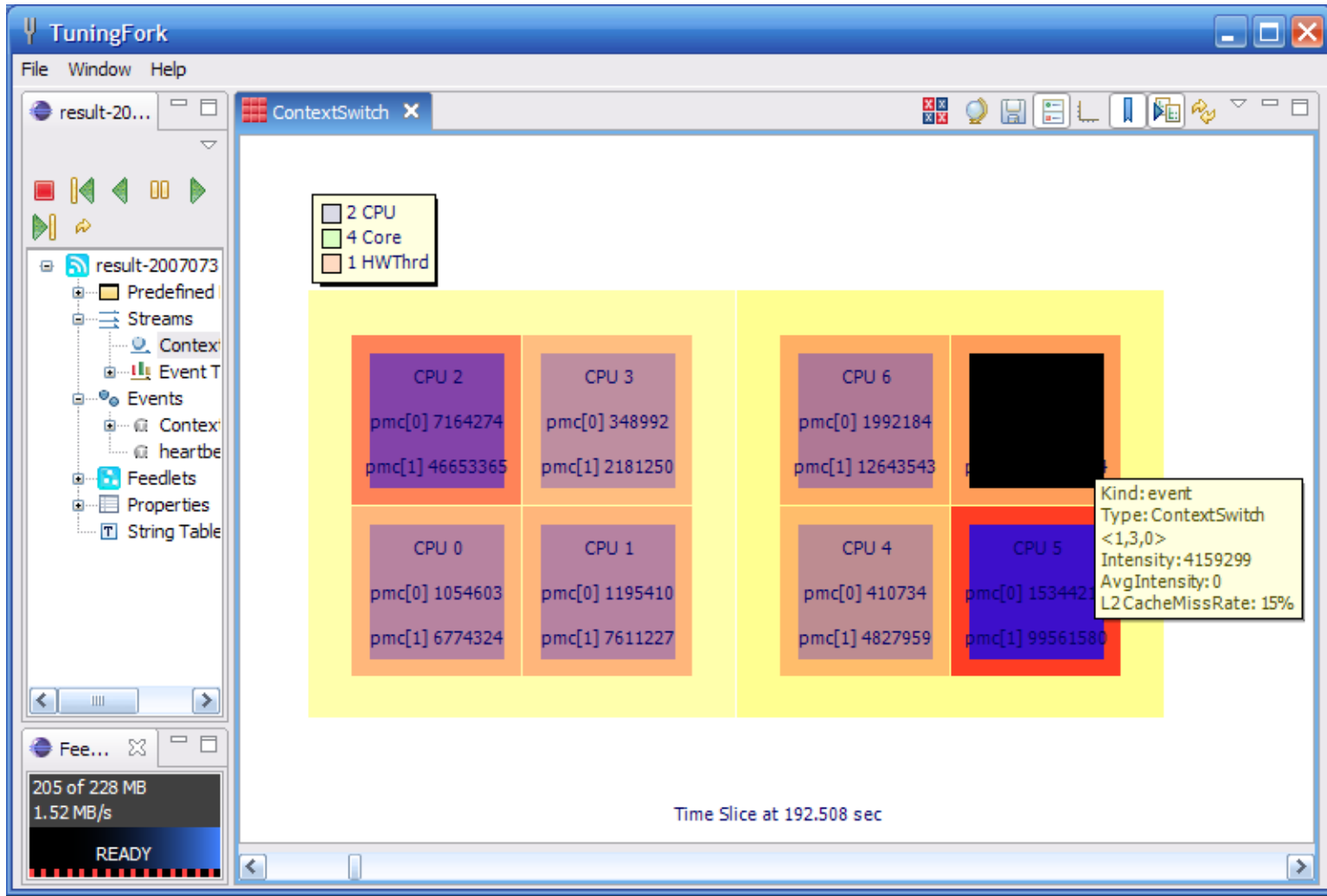
JBB trace on 2-Xeon(5343 QuadCore)



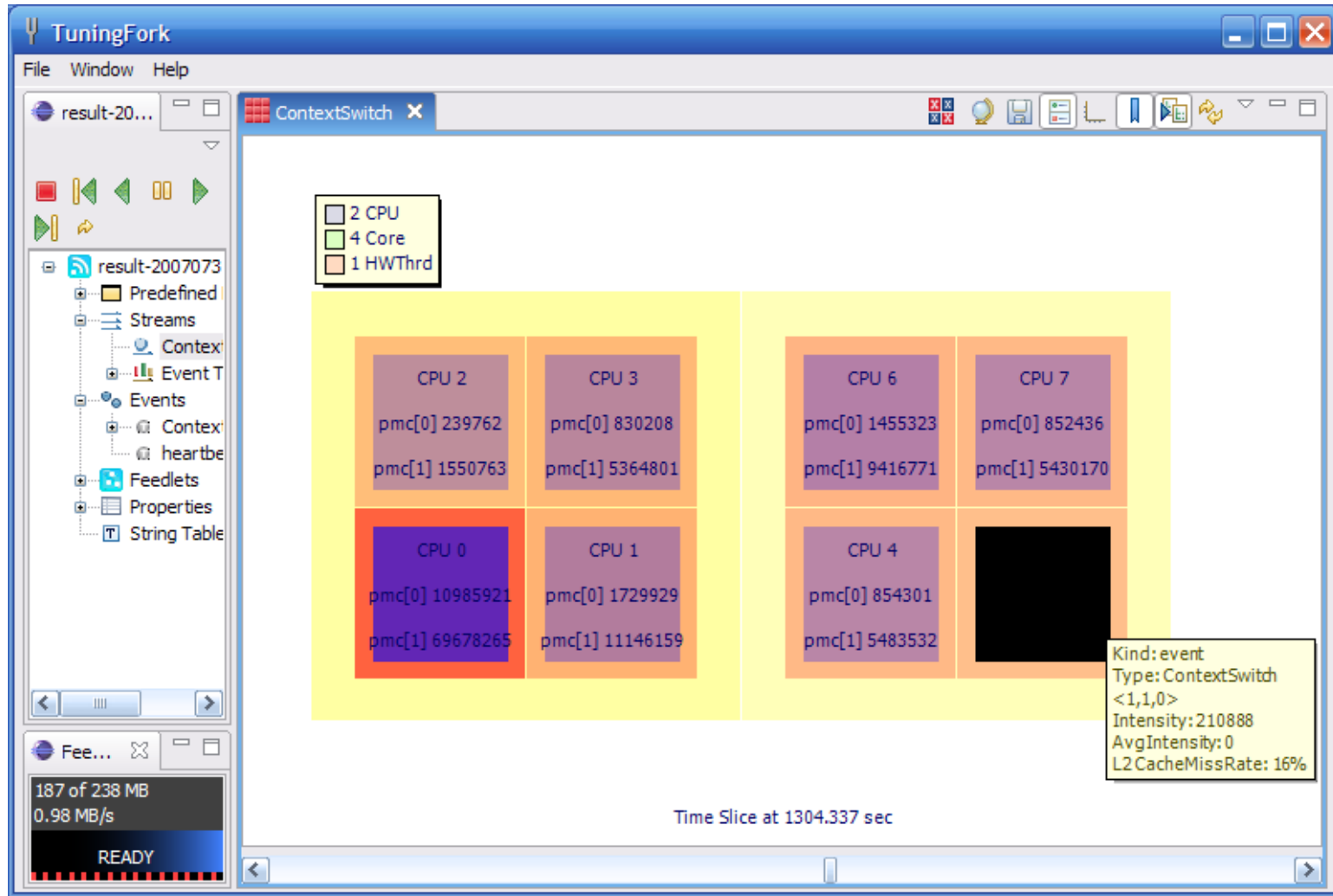
TimesliceView over Time



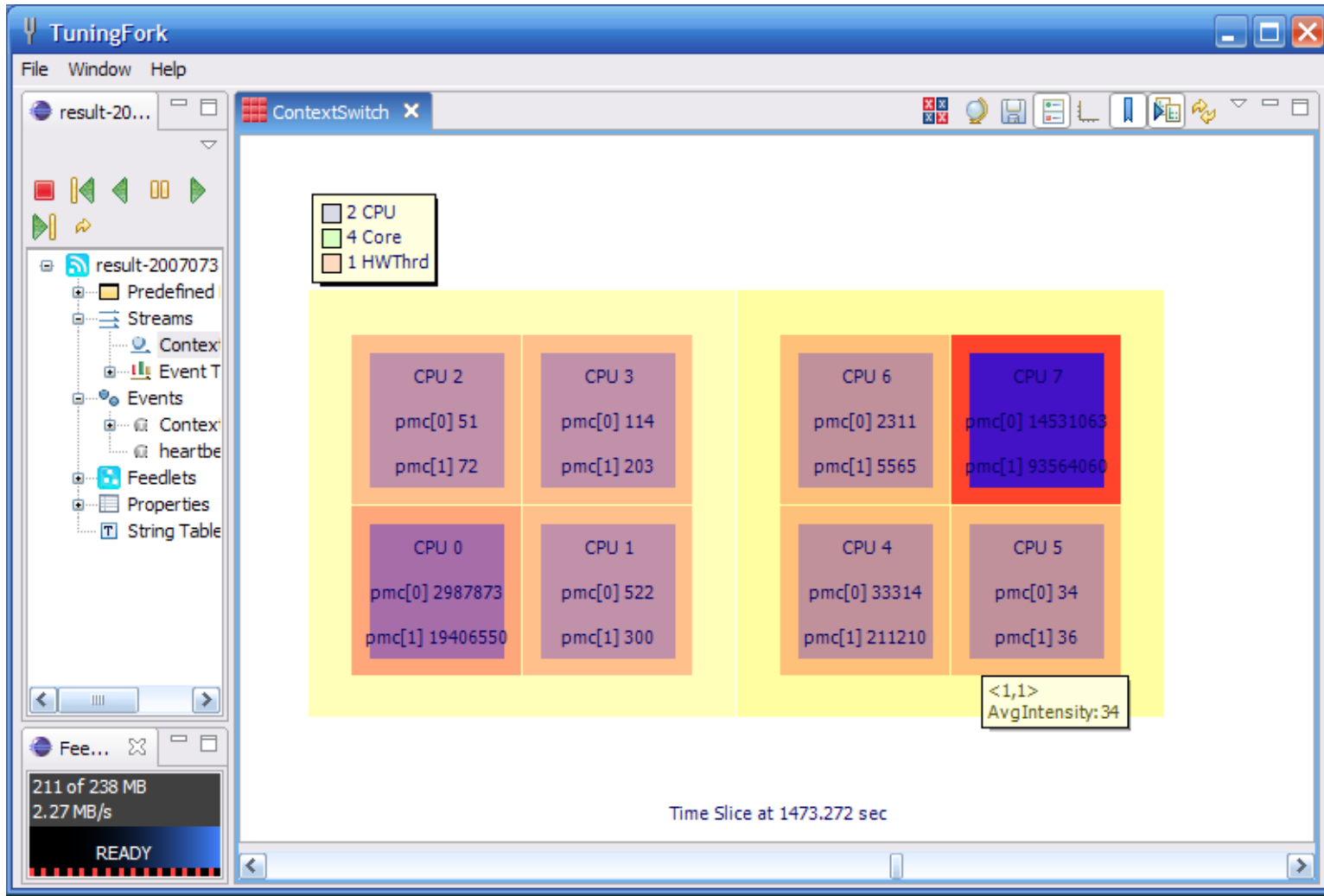
TimesliceView over Time



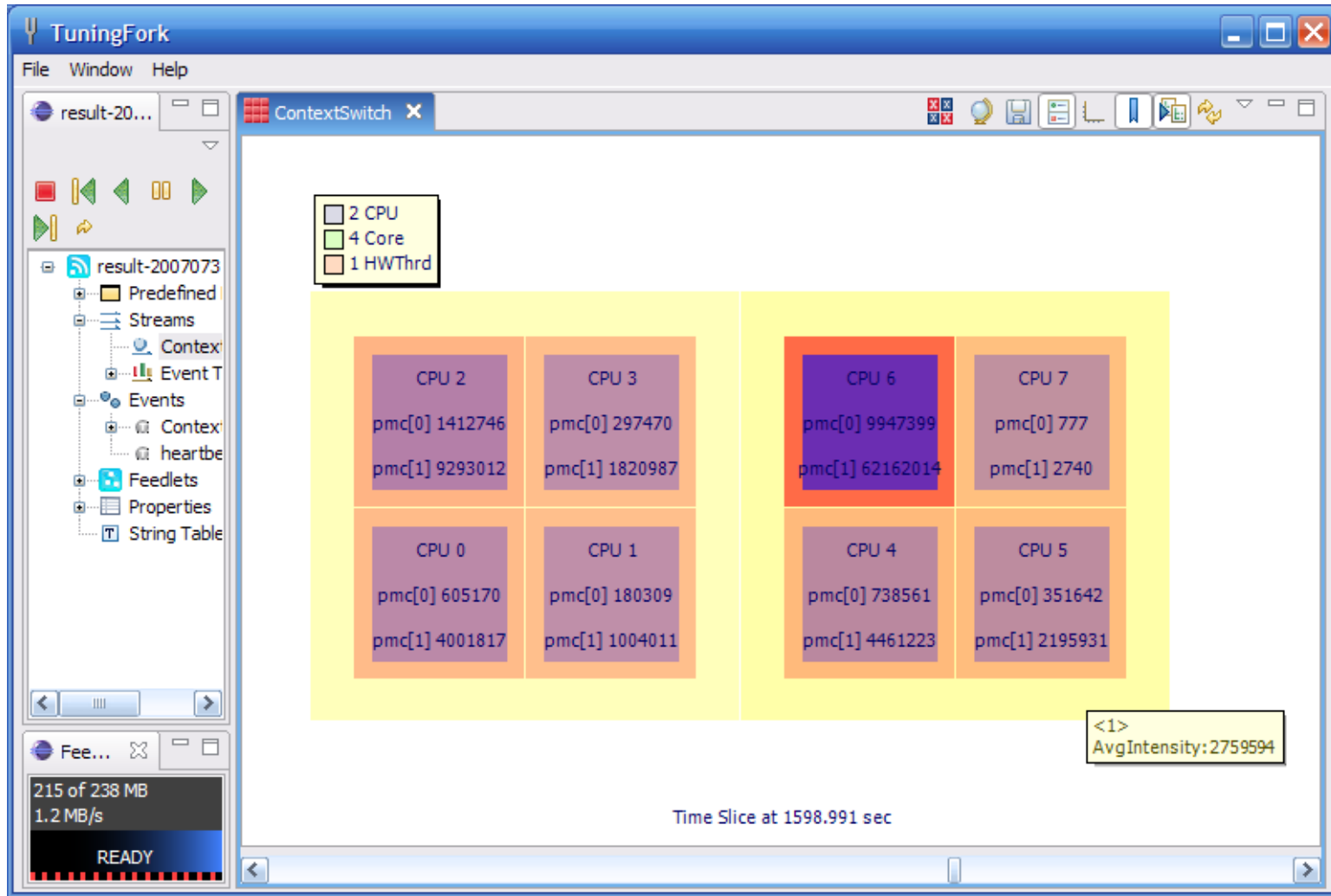
TimesliceView over Time



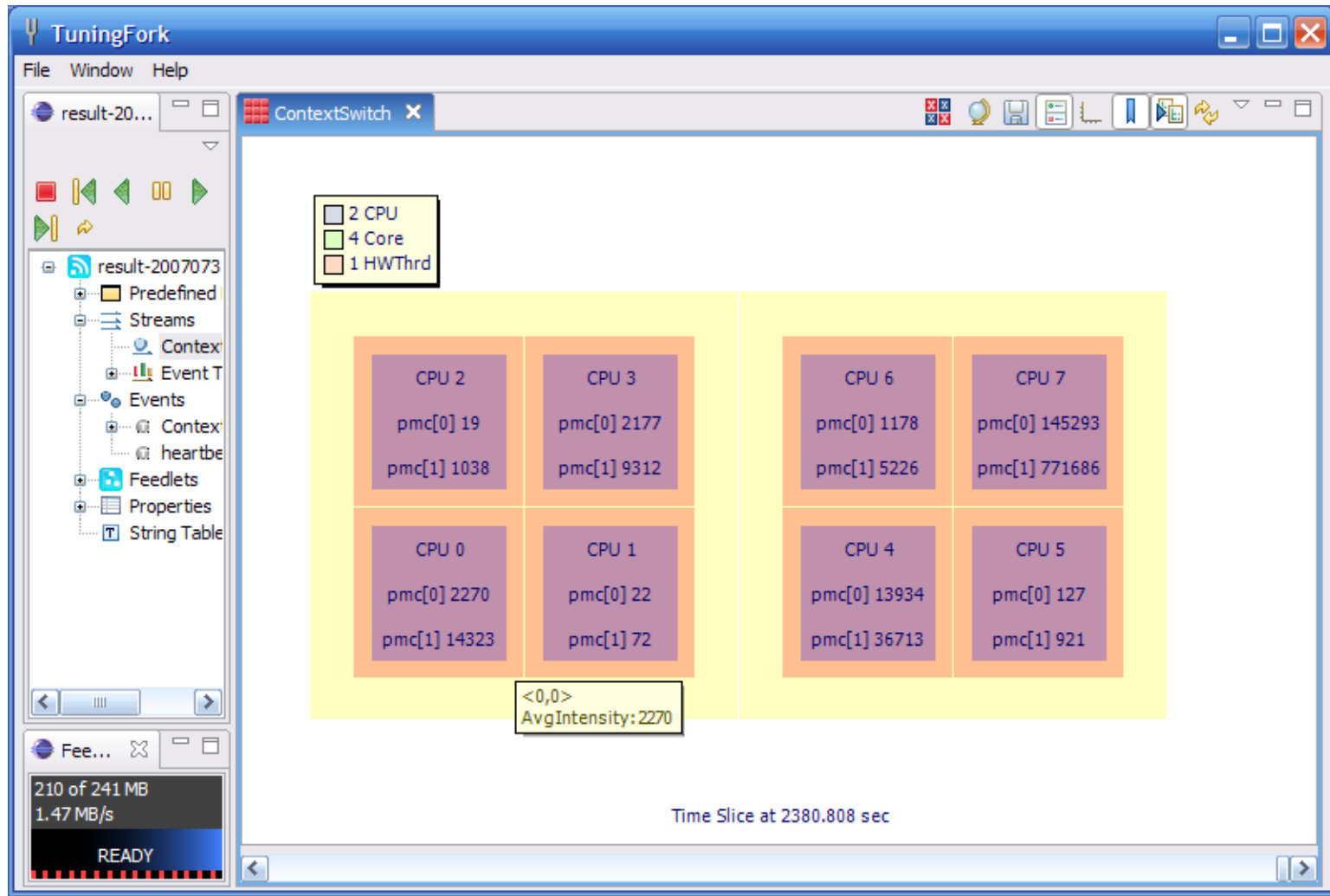
TimesliceView over Time



TimesliceView over Time



TimesliceView over Time



Conclusion

▶ Timeslice View

- ▶ Provides real-time visualization of instantaneous and overall system behavior
 - ▶ Cache Accesses/Cache Misses
 - ▶ Synchronization
 - ▶ Thermal Hotspots
- ▶ Helps architects to visualize large-scale multi-core traces, characterize application behaviors, and achieves better overall performance tuning.



Special thanks 😊 to

- ▶ Qi Ming Tang

- ▶ for providing multi-core traces

- ▶ Peter Sweeney

- ▶ for guidance on TuningFork trace generation

- ▶ David Bacon and Evelyn Duesterwald

- ▶ for technical guidance on TuningFork Timeslice View

