Doing MORE with less

David Stanasolovich,
GM, Platform Engineering Capabilities
Information Technology, Intel Corp

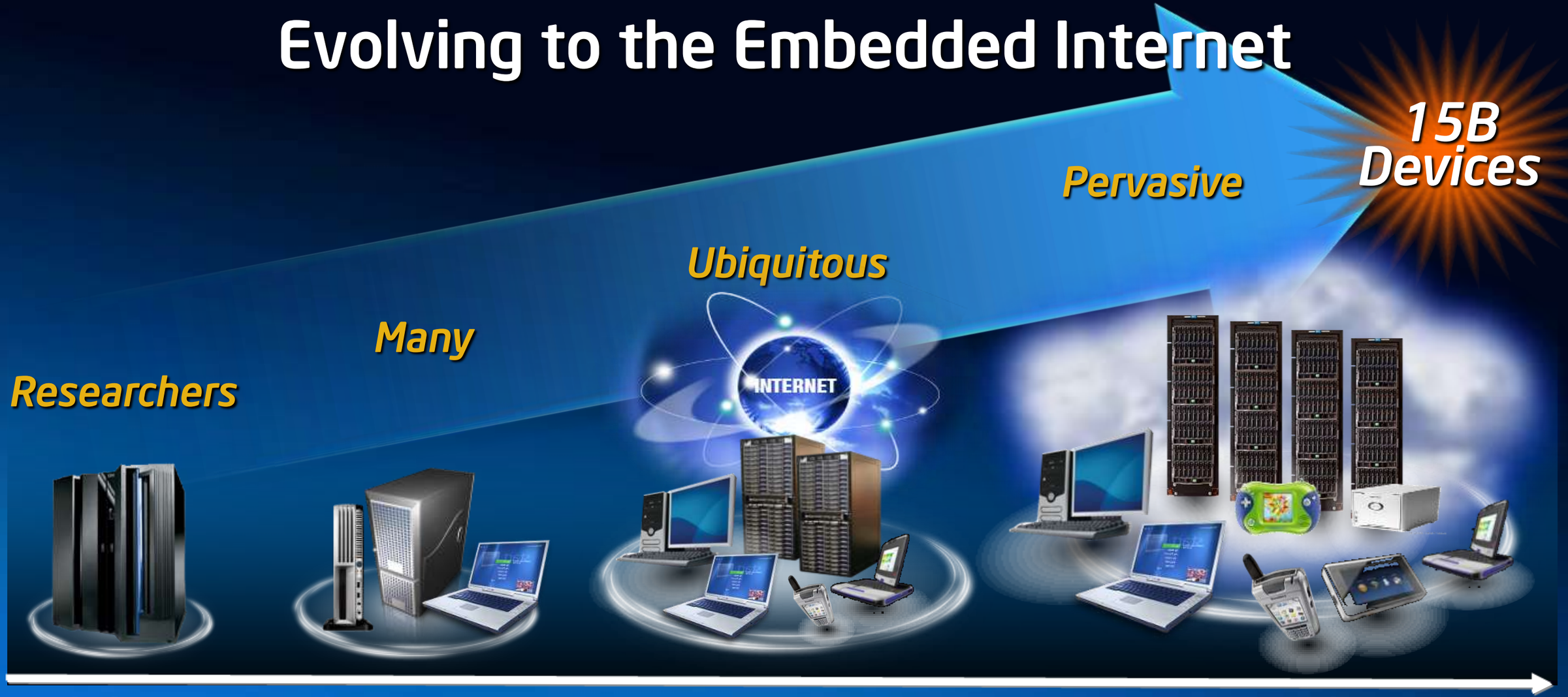
EDPS

intel

Contents

- Evolution of Computing
- Behind the Scenes
- Design Bottlenecks
- Current Issues and Solutions
- Call for Action
  - What we can do together?

Evolving to the Embedded Internet



Requires Dynamic, Efficient and Scalable Infrastructure

Intel Xeon 5500: Optimized for the Cloud

Intelligent Performance

- Up to 125% higher compute performance*

Energy Efficiency

- Automated power states
- ~50% lower idle power*
- Higher operating temperature

Dynamic Virtualization

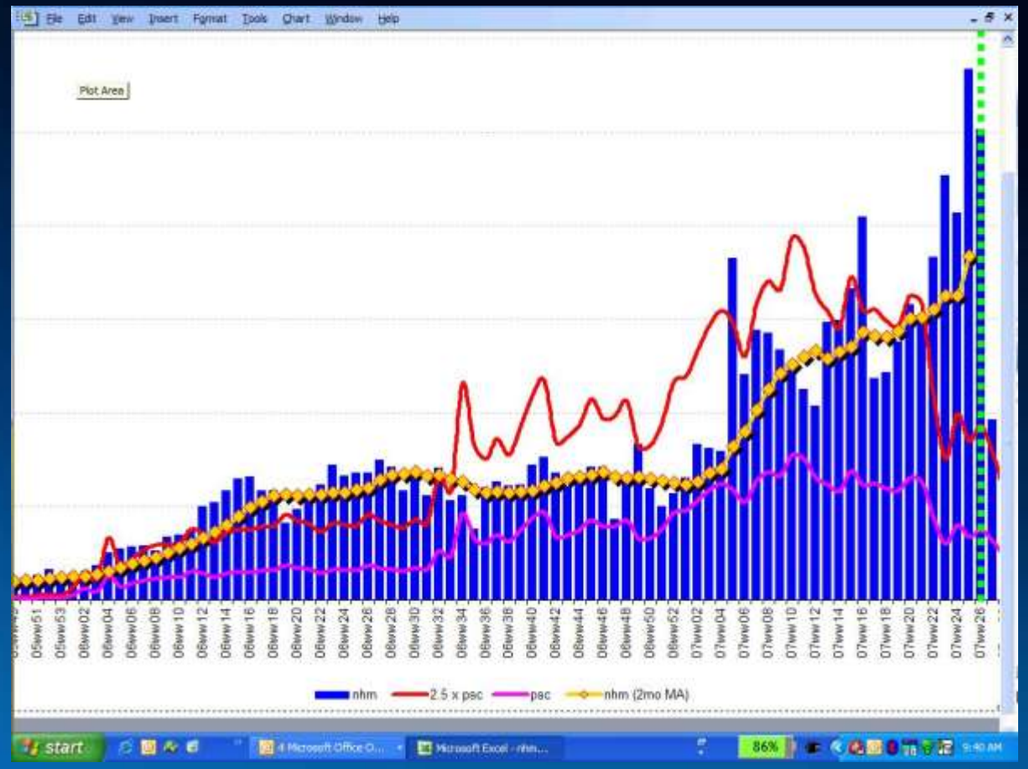
- Up to 2X virtualization performance*
- Enhanced I/O virtualization

EDPS

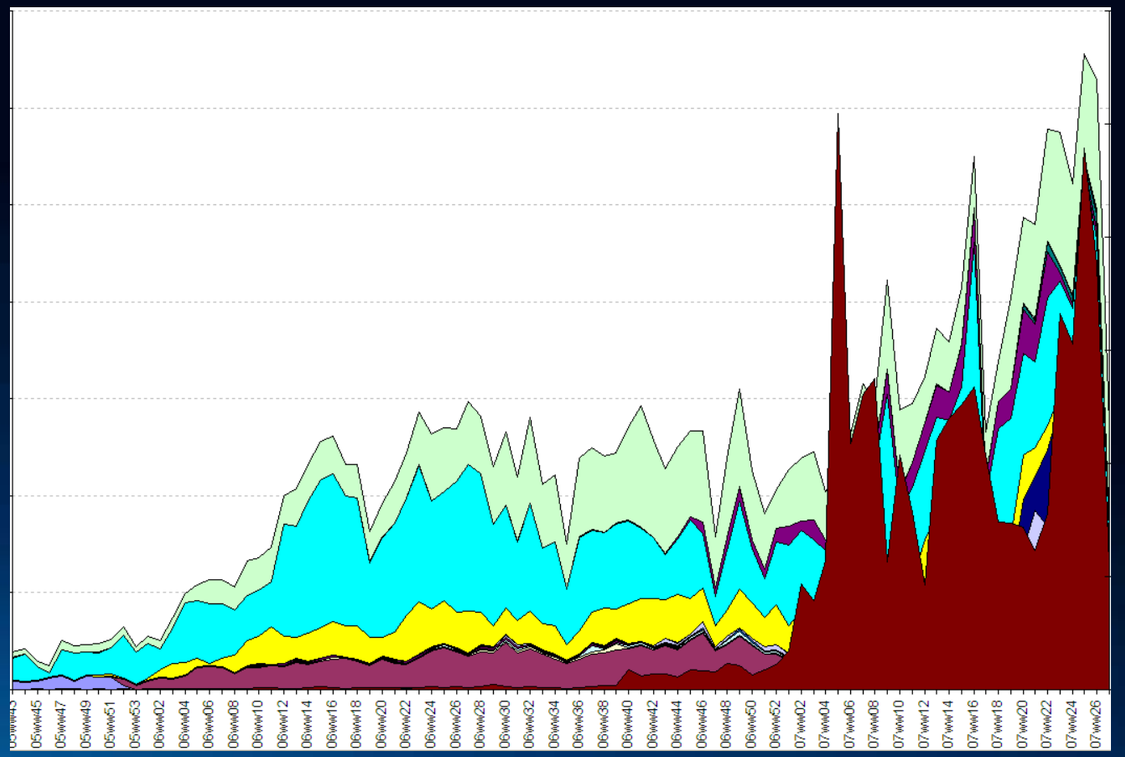
*Up to 125% higher compute performance, 50% idle power, and up to 2X virtualization performance are based on comparison to previous generation 5400 series. Lower cooling cost based on Intel internal analysis (January 2009). For detailed calculations, configurations and assumptions refer to the legal information slide in backup.

intel

What it took to deliver the product?



Total Forecast vs. Actuals



AV simulations took 80% of compute cycles

Only 20% of Compute power is used for Design

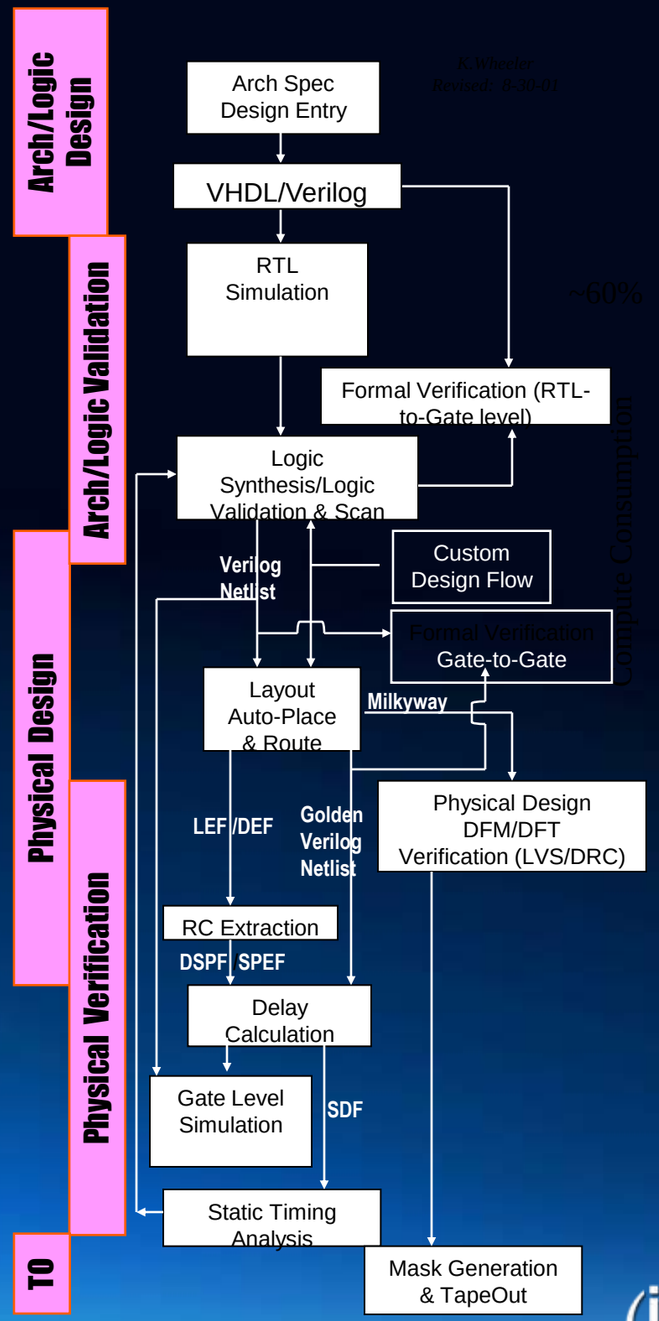
EDPS

intel

Design Flow Bottlenecks

- A typical processor design flow has
  - Dozens of stages with up to 100 tools
  - Generating 1000's of design and log files
  - Consuming Millions of CPU hrs to finish
- Flows and tool versions may differ across projects, resulting in
  - Inconsistent computing environment
  - Harder to share and maintain with updates
- Drive productivity increases through
  - common design methods/tools across projects from RTL to TO.
  - Identifying critical design challenges and gap analysis
  - Address those challenges by the application of design flow process, BKMs and tools.
  - Clearly state reasoning behind any misalignment.

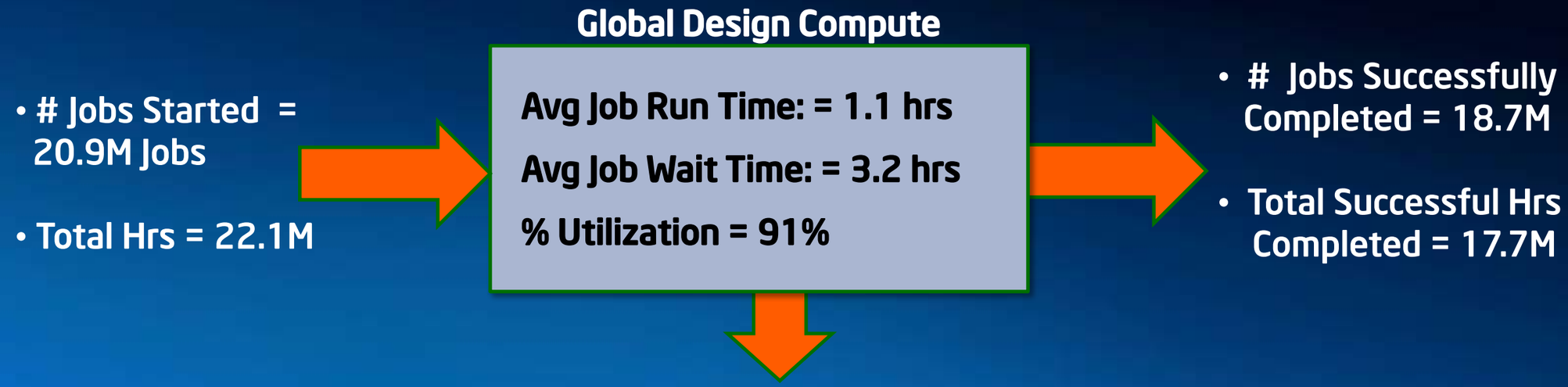
No Easy Wins, hard to get away from Legacy…



EDPS

Design Compute System is an "Information Factory"

Design Global Batch Stats – Average One Week Results



Reduced "Time to Information" Improves Engineering Productivity

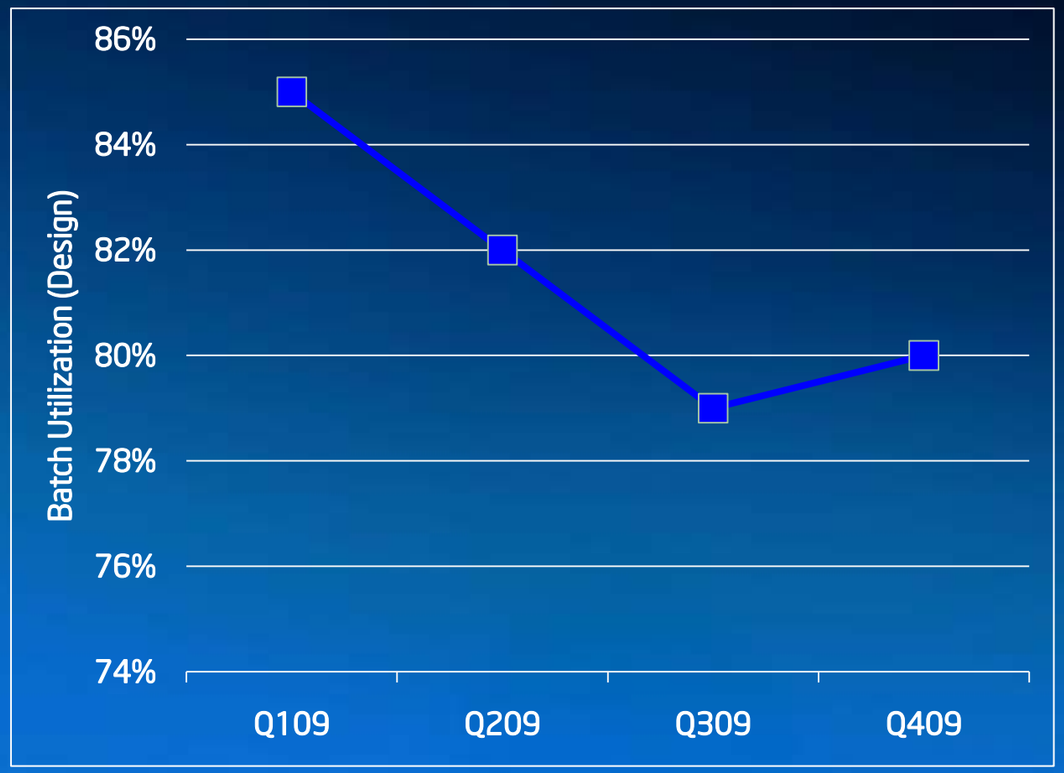
- Reducing both run time and wait time
- Ensure that jobs complete successfully
- Users don't have to worry about what is going on with their job
- Simplified job submission and execution – fewer decisions
- Deliver metrics the user needs automatically

Design Compute Vectors to Balance

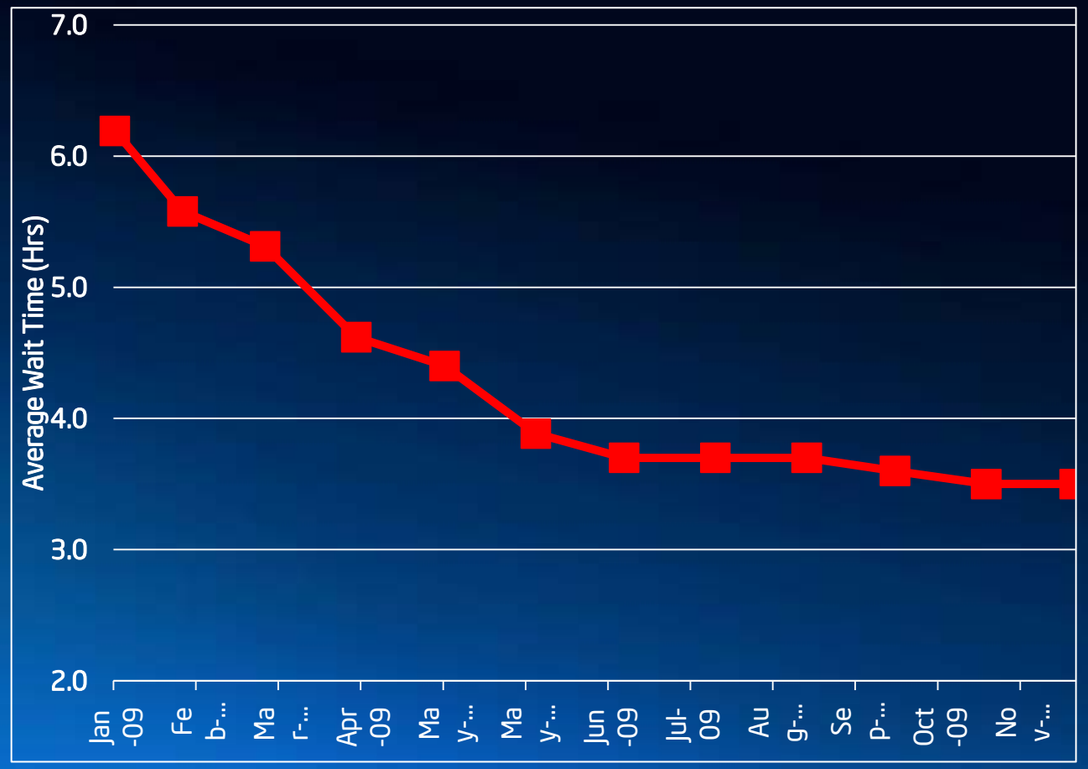
1. Drive Batch and Interactive computing utilization to minimize cost
2. Moderate Batch utilization to reduce wait time queues
3. Continually drive reductions in job fails
4. Innovate to reduce job run times
5. Innovate to eliminate waste and maximize output
6. Utilize data mining and predictive analytics to identify improvements
7. Minimize Design Iterations and time taken for each iteration

Focus Shifts From Utilization to Time to Information

Batch Utilization



Average Compute Job Queue Time

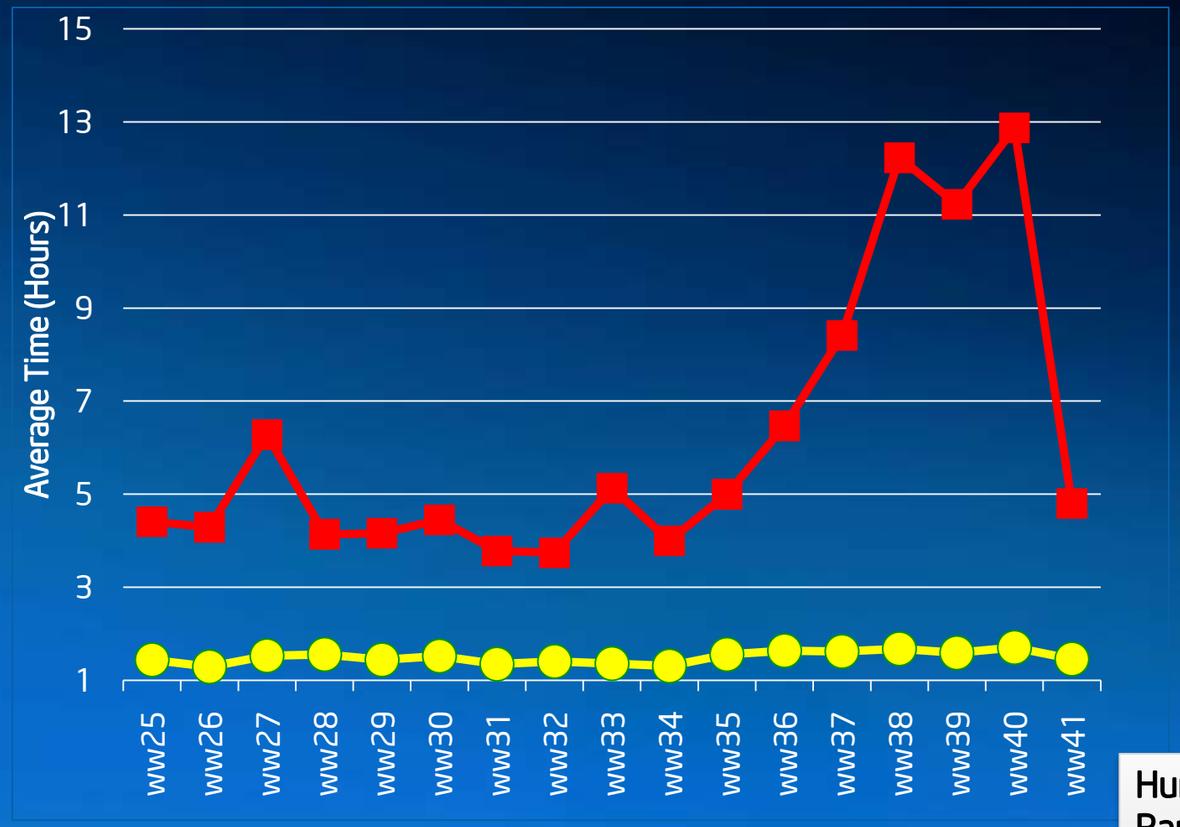


EDPS

intel

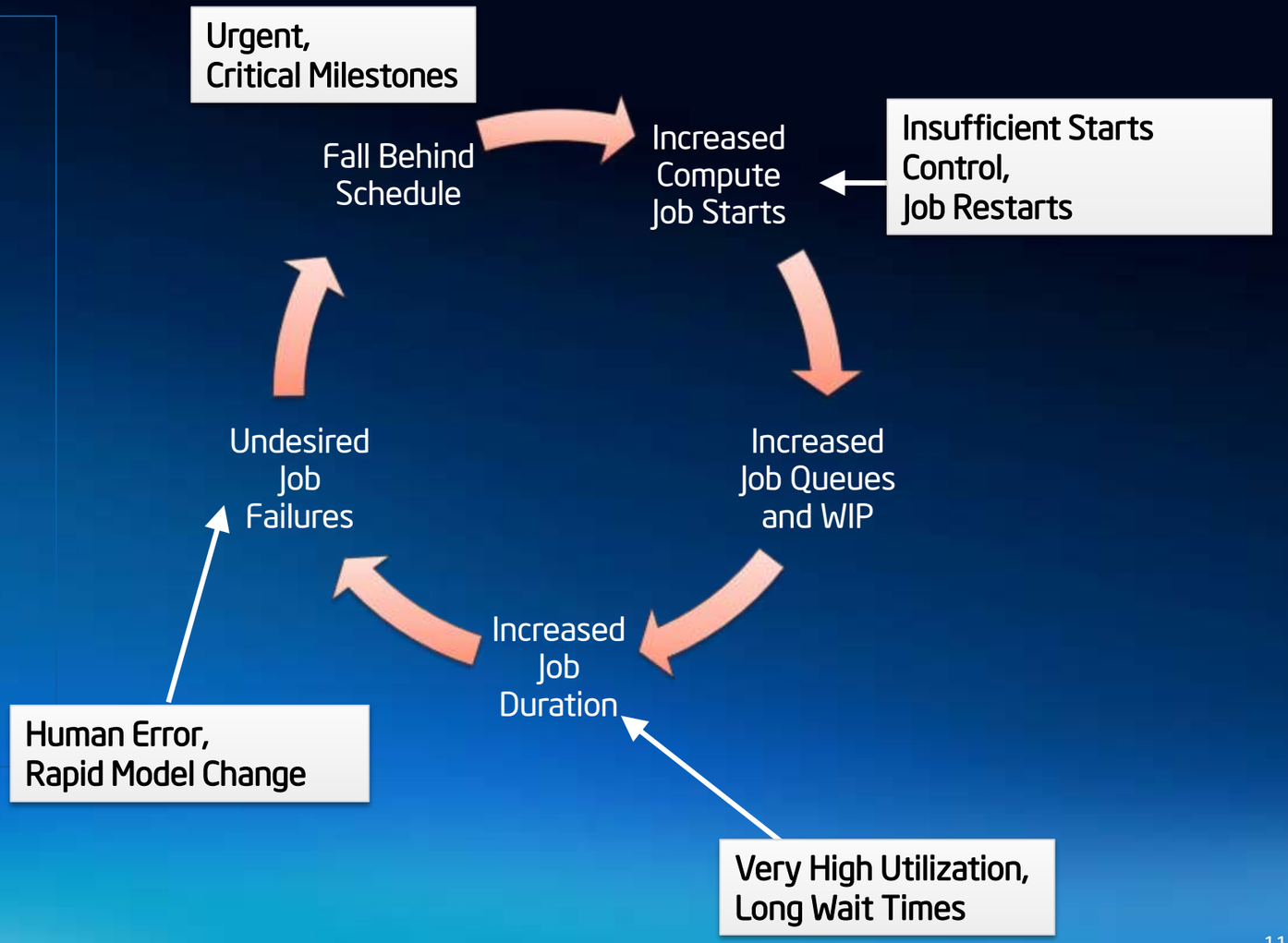
Job Congestion Increases Time to Information

Average Batch Job Wait Time



Caused By

Batch Job "Congestion Loop"



Reducing Compute Job Failures is a Key Improvement

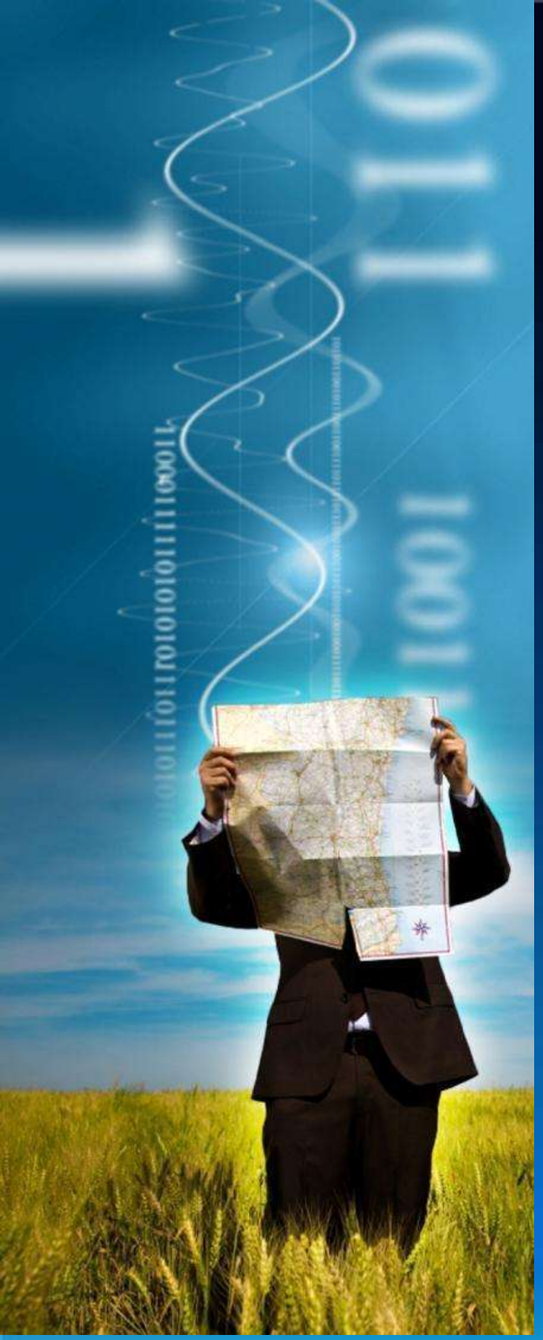
- Myth: “Compute is Free”
  - Not at our compute volumes.
- Batch compute job failures were over 27% in Jan 2009
  - Many causes: Missing files, ran out of disk space, old model, etc
- Method of attack – pareto fails each week, determine root cause, and implement corrective actions
- Result: Reduced to ~ 18%, saving over 35M compute hours

Data Mining, Predictive Analytics Identifies and Drives Improvements

- Beginning to see huge opportunities
  - mining and using data generated from design compute environment
- Data mining, predictive analytics POCs are showing positive results
  - Analyzing “what causes job failures”
  - How to improve job memory requirement accuracy in flight
  - How to forecast job duration
- Many optimization and waste elimination areas are opening

Going Forward…

- We intend to Continue JFR with some help from Data Mining tools
- Drive a common Design and Computing environment
  - Not 100% compliance as silicon products maybe different
    - A custom CPU vs. IP reuse for SoCs etc.
  - Easy to move design jobs from one data-center to another
- Looking to share server pools between interactive and batch jobs
- Apply HPC techniques to speedup individual design stages
- Drive multi-core adoption by EDA Vendors!!



Explore New Computing Models

- Evolve from Client-Server to a truly distributed computing
- Use CPUs of laptops, IA has multi-core processors there too
- Leverage incremental computing
  - Limit the computation to extent of changes made
  - Ensure that output delta corresponds to input changes
- Seek accountability in validation methods for better coverage
- Improve the simulation speed of FC models
- Ability to handle large logic model sizes
  - More features are getting integrated

Call For Action!!

- Recognize the importance of efficient computing environment
- Continue to identify and relieve any productivity bottlenecks
- EDA academia researcher and Industry need to
  - Benchmarks on the latest IA multi-core platforms
  - Evolve efficient Tools and Methods to use the available HW
  - Proliferate training and production tools for efficient HW usage