Cloud-Computing Economies of Scale

Self Managing Database Systems

James Hamilton, 2009/3/29

VP & Distinguished Engineer, Amazon Web Services

e: James@amazon.com

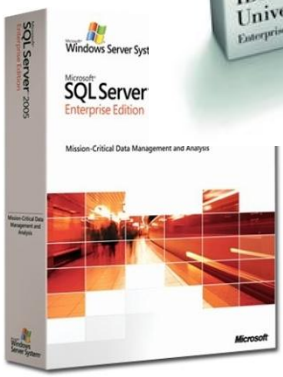
w: mvdirona.com/jrh/work

b: perspectives.mvdirona.com

Agenda

- Services economies of scale
- Services != enterprise IT
- Speeds, feeds, & trends in servers & high-scale services
- Selection S/W & H/W techniques used to scale
- Summary

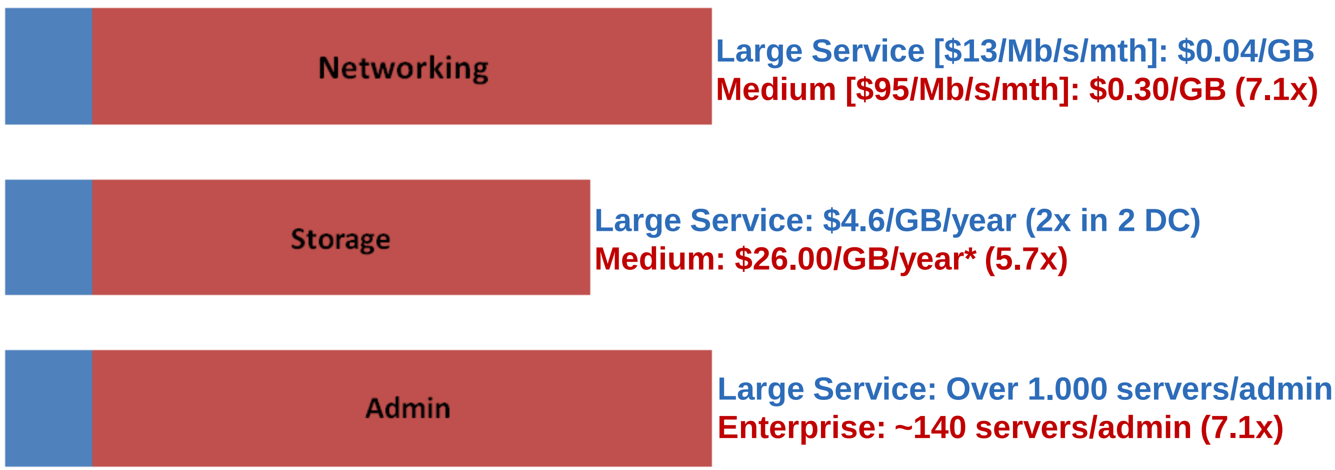
Background & biases



- 15 years in database engine development
  - Lead architect on IBM DB2
  - Architect on SQL Server
- Past 5 years in services
  - Led Exchange Hosted Services Team
  - Architect on the Windows Live Platform
  - Architect on Amazon Web Services
- Talk does not necessarily represent positions of current or past employers

Services Economies of Scale

- Substantial economies of scale possible
- 2006 comparison of very large service with small/mid-sized: (~1000 servers):



- High cost of entry
  - Physical plant expensive: 15MW roughly $200M
- Summary: significant economies of scale but at very high cost of entry
  - Small number of large players likely outcome

Services Different from Enterprises

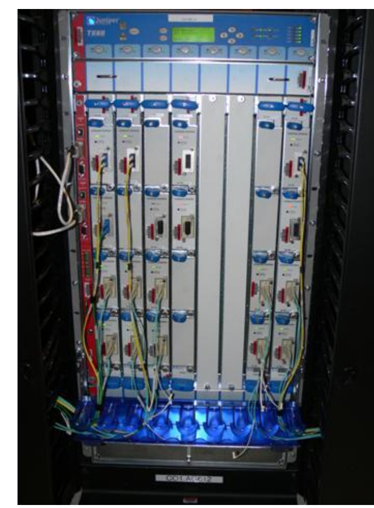
- **Enterprise Approach:**
  - Largest cost is people -- scales roughly with servers (~100:1 common)
  - Enterprise interests center around consolidation & utilization
    - Consolidate workload onto fewer, larger systems
    - Large SANs for storage & large routers for networking

- **Internet-Scale Services Approach:**
  - Largest costs is server & storage H/W
    - Typically followed by cooling, power distribution, power
    - Networking varies from very low to dominant depending upon service
    - People costs under 10% & often under 5% (>1000+:1 server:admin)
  - Services interests center around work-done-per-$ (or joule)

- **Observations:**
  - People costs shift from top to nearly irrelevant.
  - Expect high-scale service techniques to spread to enterprise
  - Focus instead on work done/$ & work done/joule

Limits to Computation

- Processor cycles are cheap & getting cheaper
- What limits the application of infinite cores?
  1. Power: cost rising & will dominate
  2. Communications: getting data to processor
- The most sub-Moore attributes typically require the most innovation
  - Infinite processors require infinite power
  - Getting data to processors in time to use next cycle:
    - Caches, multi-threading, ILP,…
    - All techniques consume power
- Conclusion: power & comm key constraints
  - Impacts DC design, server design, & S/W architecture



Storage B/W lagging Memory & CPU

	CPU	DRAM	LAN	Disk
Annual bandwidth improvement (all milestones)	1.5	1.27	1.39	1.28
Annual latency Improvement (all milestones)	1.17	1.07	1.12	1.11

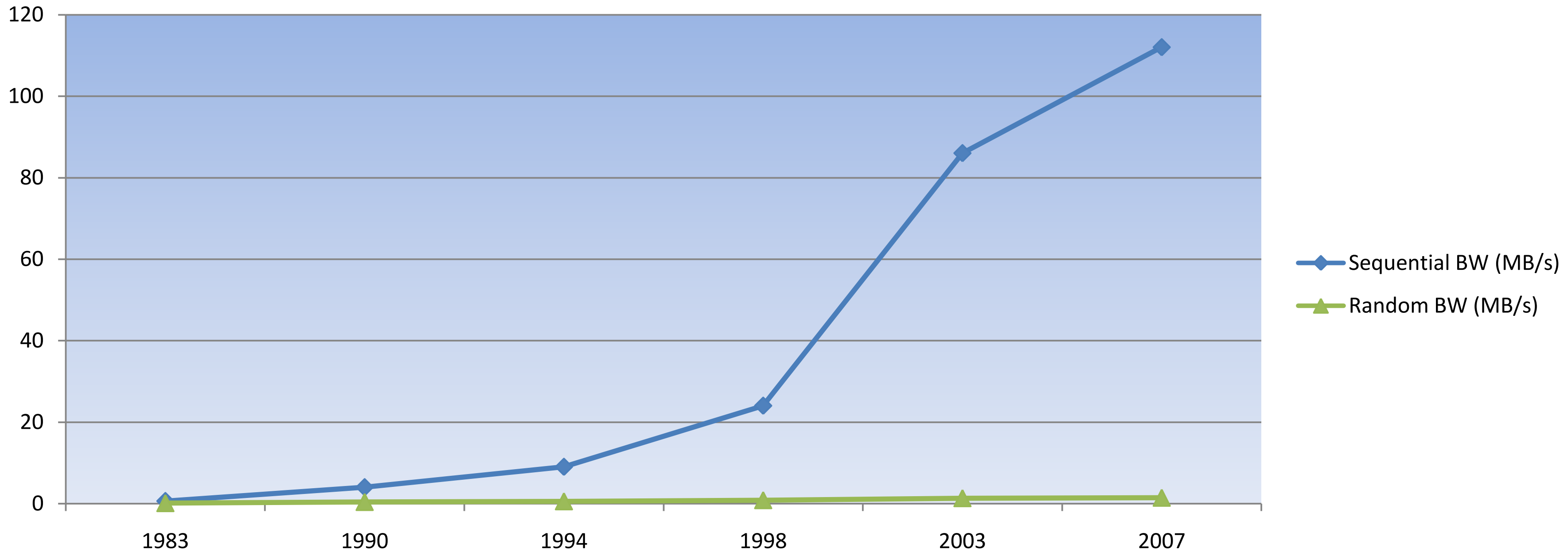
Memory wall

Storage Chasm

- CPU out-pacing all means to feed them
- Hubble's Expanding Universe:
  - Everything is getting further away from everything else [Pat Helland]
- Specifically, disk is getting "further" away from memory sub-system driving larger memories and/or more disks

Table from Dave Patterson: Why Latency Lags Bandwidth and What It Means to Computing

Disk Random BW vs Sequential BW

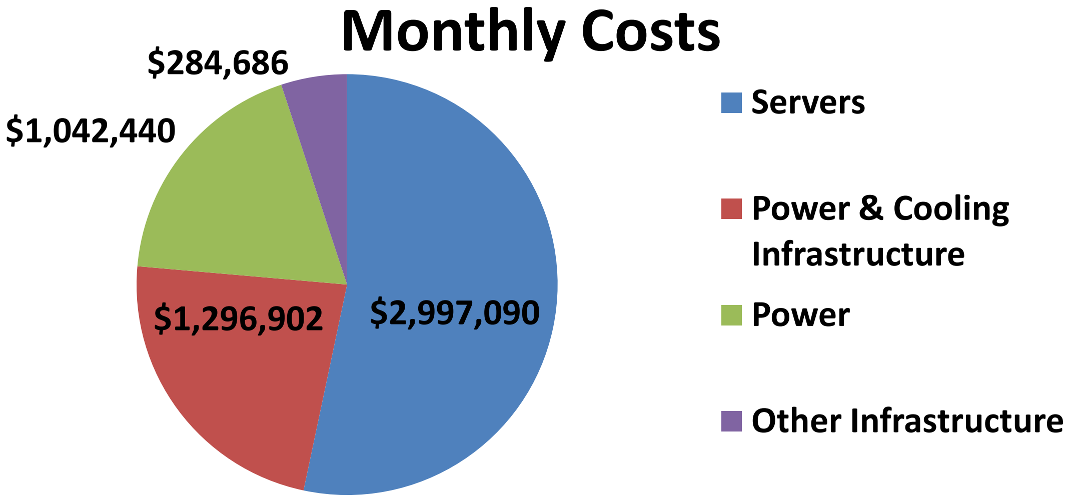


Source: Dave Patterson with James Hamilton updates

- Disk sequential BW lagging DRAM and CPU
- Disk random access BW growth roughly 10% of sequential BW growth
- **Conclusion**: Storage Chasm widening requiring larger memories & more disks

Power & Related Costs Dominate

- **Assumptions:**
  - Facility: ~$200M for 15MW facility (15-year amort.)
  - Servers: ~$2k/each, roughly 50,000 (3-year amort.)
  - Average server power draw at 30% utilization: 80%
  - Commercial Power: ~$0.07/kWhr



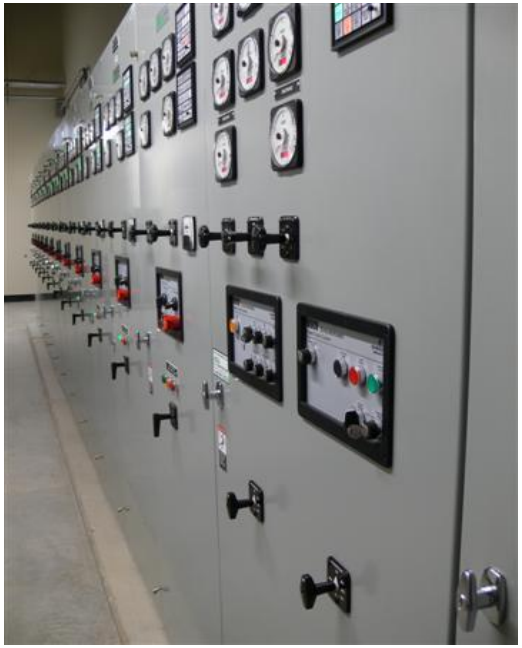
3yr server & 15 yr infrastructure amortization

- **Observations:**
  - $2.3M/month from charges functionally related to power
  - Power related costs trending flat or up while server costs trending down

Details at: http://perspectives.mvdirona.com/2008/11/28/CostOfPowerInLargeScaleDataCenters.aspx

Fully Burdened Cost of Power

- **Infrastructure cost/watt:**
  - 15 year amortization & 5% money cost
  - =PMT(5%,15,2MM,0)/(15MW) => $1.28/W/yr
- **Cost per watt using $0.07 Kw*hr:**
  - =-0.07*1.7/1000*0.8*24*365=> $0.83/W/yr (@80% power utilization)



- **Annually fully burdened cost of power:**
  - $1.28 + $0.83 => $2.11/W/yr

Details at: http://perspectives.mvdirona.com/2008/12/06/AnnualFullyBurdenedCostOfPower.aspx

Agenda

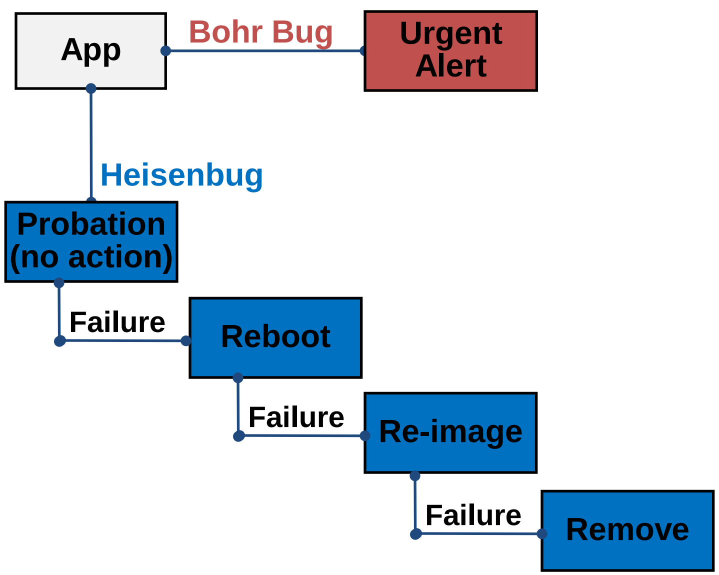
- Services economies of scale
- Services != enterprise IT
- Speeds, feeds, & trends in servers & high-scale services
- Selection S/W & H/W techniques used to scale
- Summary

Partitioned & Redundant

- Scalability & availability only achieved through partitioning & redundancy
  - Over 4 nines only through redundancy
    - Best hardware never good enough
    - Highly reliable S/W evolves VERY slowly
  - RDBMS hard to scale & manage
    - "Hand" partitioned clusters the norm (complex and high touch)
- Lower quality hardware in large numbers more reliable in aggregate than high-quality hardware
- Repeating a trend seen before in disk
  - Expect the same trend to again play out in networking
- Reliable service built on unreliable S/W & H/W

ROC Service Design Pattern

- Recover-Oriented Computing (ROC)
  - Assume software & hardware will fail frequently & unpredictably
  - Only affordable admin model at high scale
- Heavily instrument applications to detect failures



- **Take machine out of rotation and power down**
- **Set LCD/LED to "needs service"**

Bohr bug: Repeatable functional software issue (functional bugs); should be rare in production

Heisenbug: Software issue that only occurs in unusual cross-request timing issues or the pattern of long sequences of independent operations; some found only in production

Relaxed Consistency Models

- Full ACID semantics unaffordable in real dist. systems
  - Consistency, availability, or partition-tolerance
    - Pick any two*
  - Financial transactions often used as examples of needing ACID yet two-phase commit seldom used
- Relax consistency model exploiting knowledge of application semantics
  - Caches & temporal inconsistency
- Hairball problem in social networks
  - Redundant application maintained partitioned views
  - Caching (e.g. memcached)

***** CAP Conjecture, Eric Brewer***

Some Data "Pulled" to Core And Some to Edge

- User data pulled to the edge (close to user)
  - Highly interactive web applications
  - Social & political restrictions on data movement
    - e.g. Patriot Act concerns & jurisdictional restrictions
  - Application & data availability
  - Techniques:
    - Content Distribution Networks
    - Geo-partitioned and/or geo-redundant applications
- Aggregated data pulled to network core
  - Data mining & analysis workloads run central
    - e.g. MapReduce workloads

High-Scale Data Analysis

- Yield management first used by airlines
  - Airplane more expensive than computation
- Falling computing cost allows optimization & yield-management of more resources
- Heavily used in retail:
  - Shelf-space optimization, supply-chain mgmt, …
- Financial community has widely implemented automated trading & data analysis compute farms of 1,000s of nodes
- Analysis systems dominate transactional systems
  - Transactional workload growth tend to be related to business growth
  - Analysis workload growth bounded only by decline cost of computing

Memory to Disk Chasm

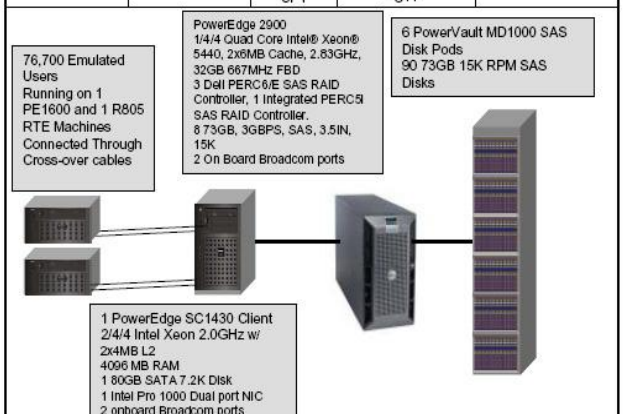
- Disk I/O rates grow slowly while CPU data consumption grows near Moore
  - Random read 1TB disk: 15 to 150 days*
- Sequentialize workloads
  - Essentially the storage version of cache conscious algorithms
    - e.g. map/reduce
  - Disks arrays can produce acceptable aggregate sequential bandwidth
- Redundant data: materialized views & indexes
  - Asynchronous maintenance
  - Delta or stacked indexes (from IR world)
- Distributed memory cache (remote memory "closer" than disk)
- I/O Cooling: Blend hot & cold data
- I/O concentration: partition hot & cold data

**** Tape is Dead, Disk is Tape, Flash is Disk, Ram Locality is King (Jim Gray)***

Case Study: TPC-C with SSD

DELL	PowerEdge 2900 Server with Oracle Database 11g Standard Edition One		TPC-C Rev 5.9 Original Report Date June 16 ,2008
Total System Cost	TPC-C Throughput	Price/Performance	Availability Date
$65,910	**97,083 tpmC**	**$.68 / tpmC**	June 16, 2008

Processors	Database Manager	OS	Other Software	Number of Users
1/4/4 Quad Core Intel® Xeon® 5440,2X6MB Cache, 2.83GHZ 1333MHZ FSB	Oracle Database 11g Standard Edition One	Microsoft Windows Server 2003 Standard x64 Edition SP1	Windows Server 2003 Standard Edition w/ COM+ Internet Information Server 6.0 Microsoft Visual C++	76,700



System Component	Server		Each Client	
Processor/Core/Cache	1	1/4/4 Quad Core Intel® Xeon® 5440, 2x6MB Cache, 2.83GHz, 1333	2	2/4/4 Intel® Xeon® w/ 2x4MB L2, 2.0 GHz
Memory		32GB 667 FB-DIMM		4 GB
Disk Controllers	3 1	Dell PERC6/E RAID Integrated PERC5i RAID	1	Onboard SATA
Disk Drives	38 60	73GB SAS 15K 36GB SAS 15K	1	80GB 7.2K SATA
Total Storage	98	4011 GB SAS	1	80GB SATA
Other	2 1	Broadcom NetXtreme II GigE ports CD-ROM	2 1 1	Broadcom on-board ports Dual port Intel Pro 1000 NIC CD-ROM

Slot	Controller	Disks		Capacity		Usage
0	Dell PERC5i	8x73GB,15K,SAS	RAID10	Disk 6	15GB	OS
				279.99GB	260GB	Logs
3	Dell PERC6/E	15x36GB,15K,SAS	RAID0	Disk 2	488.92GB	DB data
		15x36GB,15K,SAS	RAID0	Disk 3	488.92GB	DB data
4	Dell PERC6/E	15x36GB,15K,SAS	RAID0	Disk 4	488.92GB	DB data
		15x36GB,15K,SAS	RAID0	Disk 5	488.92GB	DB data
6	Dell PERC6/E	15x73GB,15K,SAS	RAID0	Disk 0	1016.23GB	DB data
		15x73GB,15K,SAS	RAID0	Disk 1	1016.23GB	DB data

HDD

SSD?

- 98 HDD total
  - 90 data disks (primarily random access)
  - 8 log & O/S disks (primarily sequential access)
- Compute HDD cross-over using fictitious 128GB SSD @ 7,000 IOPS
- 90 HDD to store 2,464GB (short stroked)
  - 106GB static & 2,357GB dynamic (60 day rule)
  - 90 disk HDD budget: $26,910 (disks only at $299)
  - Requires 20 SSDs to support @ up to $1,346 each
- Static content only (no 60 day rule)
  - Artificially more IOPS/GB than legal TPC-C
  - Assuming 325 IOPS/disk would require 5 SSDs at up to $5,382
- **VERY hot I/O workloads approaching breakeven on SSD**
  - But TPC-C is much hotter than most commercial workloads
  - SSD price continues to improve

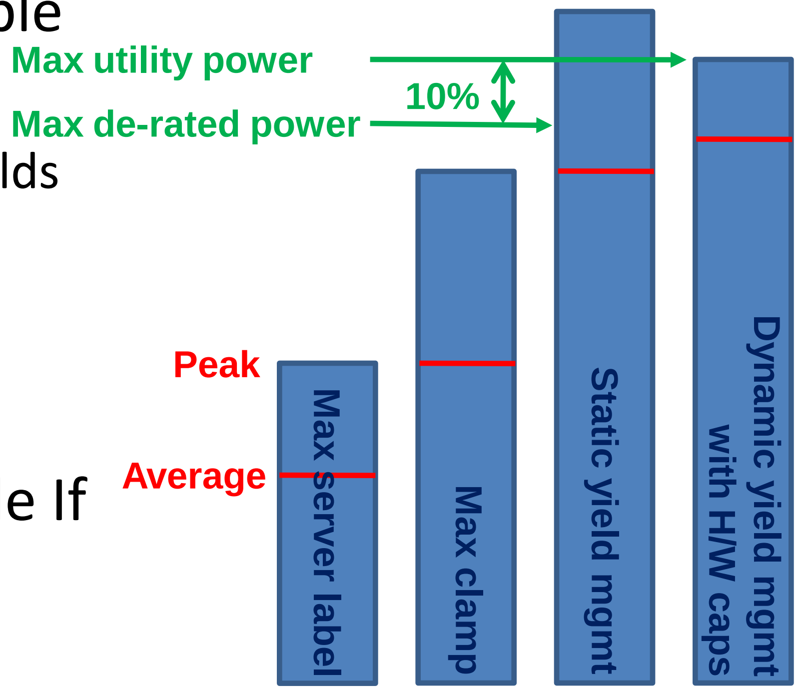
http://www.tpc.org/results/FDR/TPCC/Dell_2900_061608_fdr.pdf

Graceful Degradation & Admission control

- No economic "head room" quantity sufficient
  - Even at 25-50% hardware utilization, spikes will exceed 100%
  - EHS average-to-peak load spread over 6x
- Prevent overload through admission control
  - Service login typically more expensive than steady state
- Graceful degradation mode prior to admission control
  - Find less resource-intensive modes to provide degraded services

Power Yield Management

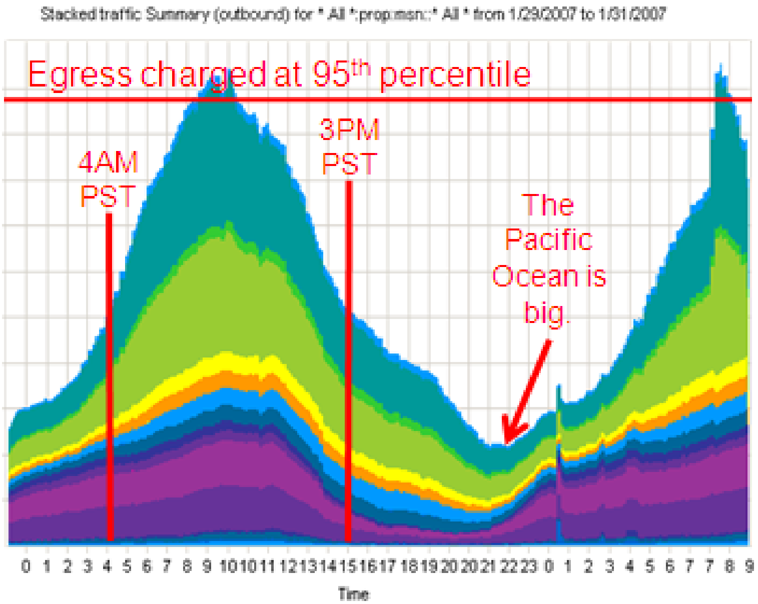
- "Oversell" power, the most valuable resource:
  - e.g. sell more seats than airplane holds
- Overdraw penalty high:
  - Pop breaker (outage)
  - Overdraw utility (fine)
- Considerable optimization possible If workload variation is understood
  - Workload diversity & history helpful
  - Graceful Degradation Mode to shed workload



Power Provisioning in a Warehouse-Sized Computer, Xiabo Fan, Wolf Weber, Luize Borroso

Resource Consumption Shaping

- Essentially yield mgmt applied to full DC
- Network charge: base + 95th percentile
  - Push peaks to troughs
  - Fill troughs for "free"
  - Dynamic resource allocation
    - Virtual machine helpful but not needed
  - Symmetrically charged so ingress effectively free



David Treadwell & James Hamilton / Treadwell Graph

- Power also often charged on base + peak
  - Server idle to full-load range: ~65% (e.g. 158W to 230W)
  - S3 (suspend) or S5 (off) when server not needed
- Disks come with both IOPS capability & capacity
  - Mix hot & cold data to "soak up" both
- Encourage priority (urgency) differentiation in charge-back model

Summary

- Hosted services & utility computing have huge economies of scale
  - Many server workloads will migrate to cloud
- Most difficult aspect of high-scale services is managing multi-datacenter distributed, partitioned, redundant, data stores & caches
  - This really is a database problem
- With partitioning & synchronous redundancy
  - Recover Oriented Computing management technique effective and used extensively in services
- Conclusion: DB world should invest more in making common service design patterns easy
  - This also makes auto-management much more tractable

More Information

- **These slides:**
  - http://mvdirona.com/jrh/TalksAndPapers/JamesHamilton_SMDB2009.ppt
- **Designing & Deploying Internet-Scale Services:**
  - http://mvdirona.com/jrh/talksAndPapers/JamesRH_Lisa.pdf
- **Architecture for Modular Data Centers:**
  - http://mvdirona.com/jrh/talksAndPapers/JamesRH_CIDR.doc
- **Where does the power go and what to do about it:**
  - http://mvdirona.com/jrh/TalksAndPapers/JamesHamilton_AFCOM2009.pdf
- **Recovery-Oriented Computing:**
  - http://roc.cs.berkeley.edu/
  - http://www.cs.berkeley.edu/~pattrsn/talks/HPCAkeynote.ppt
  - http://www.sciam.com/article.cfm?articleID=000DAA41-3B4E-1EB7-BDC0809EC588EEDF
- **Autopilot: Automatic Data Center Operation:**
  - http://research.microsoft.com/users/misard/papers/osr2007.pdf
- **Perspectives Blog:**
  - http://perspectives.mvdirona.com
- **Email:**
  - James@amazon.com