

Web and Enterprise Computing: Availability, Availability, Availability

Ling Liu
Georgia Institute of
Technology

Panel Statement

- The following three systems evolve in different directions in the context of Web and Enterprise Computing? Why? What? How?
 - Front end systems
 - Richer semantics and user-oriented functionality
 - IT backend systems
 - Higher availability, reliability, and security
 - Grid systems
 - Evolve as a Platform and infrastructure for web and enterprise computing

Enterprise Computing: What is it?

- Automation of Tasks & Processes
 - Financials
 - Human Resources
 - Enterprise Resource Planning
 - Customer Relationships
 - E-business
 - Content, Learning & Knowledge Management
- Co-management and Conflation of data about People, Places, and Things
- Deriving value from the ability to see and manage data and information



Enterprise Computing Systems in the past

- Goal: Improve performance, cost-performance
- Assumptions
 - Humans are perfect
 - Software will eventually be bug free
 - Hardware is expensive but relatively reliable (~100 years between failures), and will continue to increase
 - Maintenance costs irrelevant vs. Purchase price

Enterprise Computing Systems today

- Services as a model for future of IT
- Availability is now vital metric for services
 - near-100% availability is becoming mandatory
 - for e-commerce, enterprise apps, online services, ISPs
 - but, service outages are frequent
 - 65% of IT managers report that their websites were unavailable to customers over a 6-month period
 - 25%: 3 or more outages
 - outage costs are high
 - social effects: negative press, loss of customers who “click over” to competitor

Source: InternetWeek 4/3/2000

Jim Gray: Trouble-Free Systems

- Manager
 - Sets goals
 - Sets policy
 - Sets budget
 - System does the rest.
- Everyone is a CIO
- Build a system
 - Used by millions of people each day
 - Administered and managed by a ½ time person.
 - On hardware fault, order replacement part
 - On overload, order additional equipment
 - Upgrade hardware and software automatically.

*“What Next?
A dozen remaining IT problems”
Turing Award Lecture,
FCRC,
May 1999
Jim Gray
Microsoft*

[Dave Patterson, HPC 2002]



IBM Research (10/15/'01)

- **Overview:** Computing is too hard. It's time we stop our preoccupation with faster and more powerful and start making them smarter.
- **The Solution:** "Autonomic Computing" a systemic view of computing modeled after a self-regulating biological system; largely self-managing, self-diagnostic. User perspective:
 - **Flexible**
 - **Accessible**
 - **Transparent**

[Source: www.research.ibm.com/autonomic/, Dave Patterson, HPC 2002]

Bill Gates M/S (1/15/2002): "Trustworthy Computing"

- **Trustworthiness** is a fundamental challenge that spans entire computing ecosystem, from individual chips to global Internet services
 - **Availability:** System outages should become a thing of the past (SW architecture that supports redundancy and automatic recovery)
 - **Privacy:** Users should be in control of how their data is used
 - **Security:** should be easy for developers to understand and build into their apps

[Source: "Microsoft Makes Software Safety a Top Goal," by John Markoff, N.Y. Times, 1/17/02, Dave Patterson, HPC 2002]

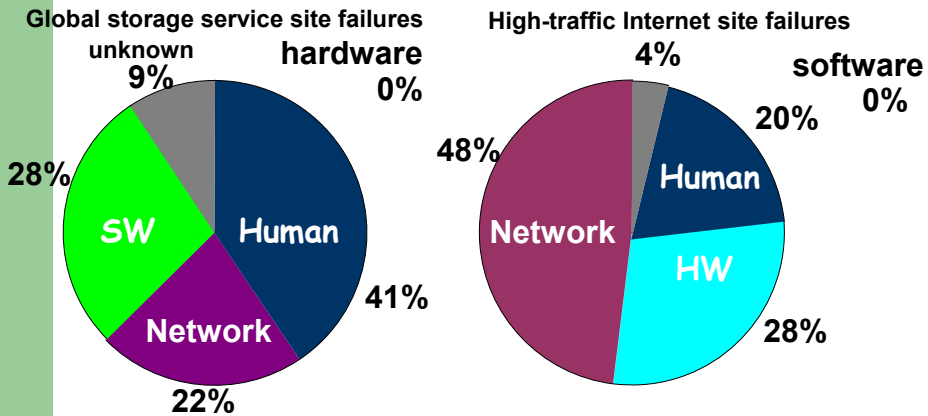
New research goals for Next generation Enterprise Computing

- **Availability**
 - 24x7 delivery of service to users
- **Extensibility (Evolutionary Growth)**
 - allow easy system expansion over time without sacrificing availability or maintainability
- **Recoverability (Maintainability)**
 - Self configuring, self-adaptive, and self-healing
- **Security/Privacy**
 - dependability

Challenge: Availability

- **Availability:**
failures are common [Dave Patterson, HPC 2002]
 - Well designed and manufactured HW: >1% fail/year
 - Well designed and tested SW: > 1 bug / 1000 lines
 - Well trained people doing difficult tasks: up to 10%
 - Well run co-location site (e.g., Exodus):
 - 1 power failure per year, > 1 network outage per year
 - Denial of service attacks => routine event

Internet Site Failures



Human error largest cause of failure in the more complex service, significant in both
Network problems largest cause of failure in the less complex service, significant in both
[Dave Patterson, HPC 2002]

Modern Enterprise Computing Philosophy

“If a problem has no solution, it may not be problem,

but a fact, not to be solved, but to be coped with over time”

— Shimon Peres (“Peres’s Law”)

Problems v.s. Facts

- **Availability is a fact to cope with**
- **Security/Privacy are facts we need to learn to deal with**
- **People/HW/SW failures are facts, not problems**
 - Recovery/repair is how we cope with above facts
- **More Fact: Cost of Ownership is 5-10X HW/SW, if necessary, sacrifice disk/DRAM space and processor performance is a practical optimization**

Challenges (cont)

- **Evolution**
 - In back-end system tiers,
 - software upgrades difficult, failure-prone, or ignored, back-end scalability difficult, operator intensive
 - In front end tier:
 - application service over WWW, daily changes are common, 1U-PC cluster front-ends scale, evolve well
- **Maintainability in backend systems**
 - system maintenance environments are unforgiving
 - human operator error is single largest failure source
- **Security/Privacy**
 - front end more vulnerable than backend