

Introduction to Robust Systems

Subhasish Mitra

Stanford University

Email: subh@stanford.edu

Objective of this Talk

- Brainstorm
 - ❖ What is a robust system ?
 - ❖ How can we build robust systems ?
 - ❖ Robust systems research directions
 - ❖ How can EE392U be beneficial ?
- Not covered: Specific solutions

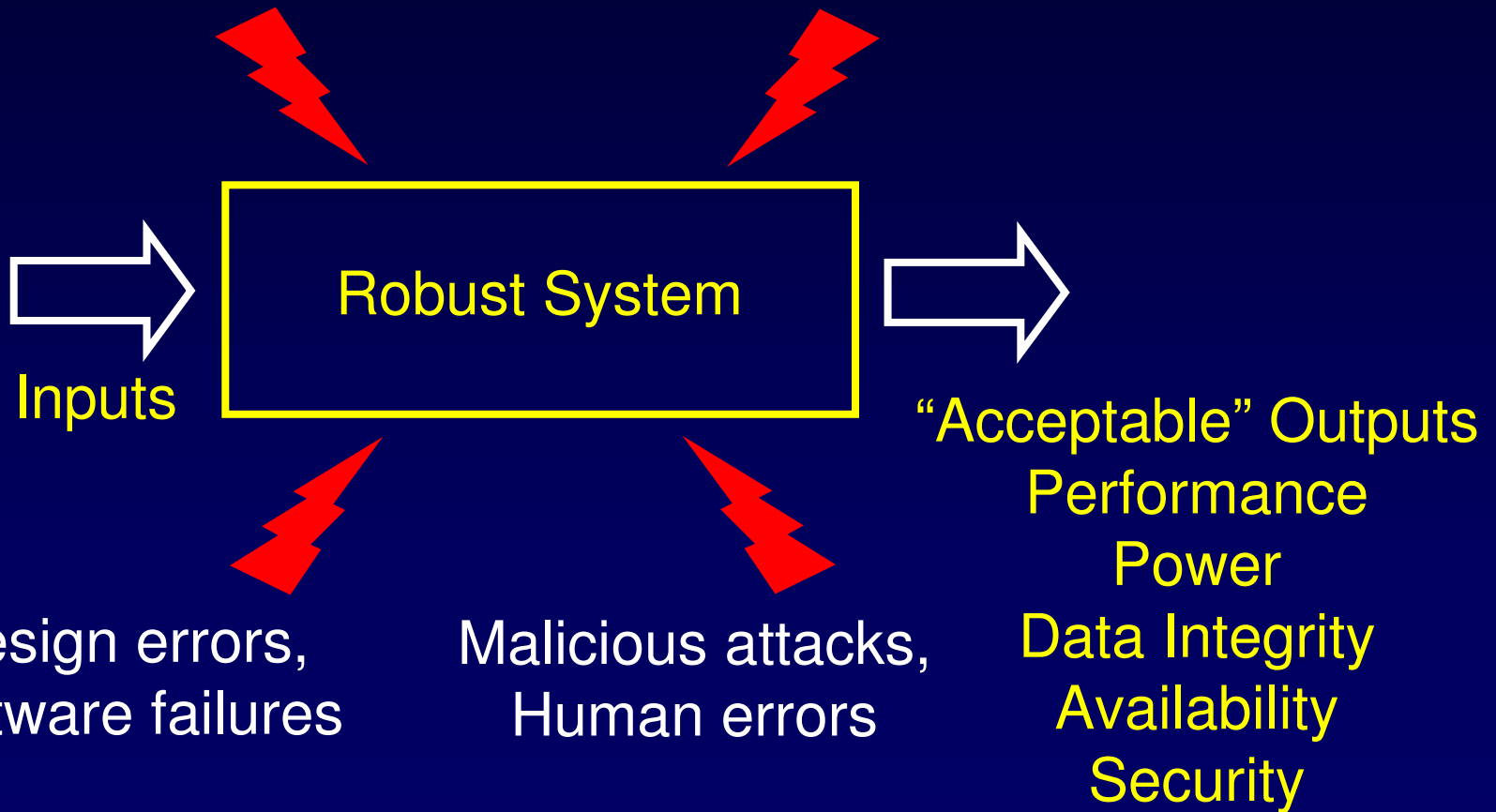
What is a Robust Design ?

- Quote from
 - ❖ www.amsup.com/robust_desig/index.htm
- “Not just strong”
 - ❖ “Flexible ! Idiot-proof ! Simple ! Efficient !”
- Consistent high-level performance
 - ❖ Wide range of changing conditions
 - Client & manufacturing related
- Anticipated vs. unanticipated

Robust Computing System

Defects, Process variation,
Degraded transistors

Radiation, Noise



Availability & Data Integrity

- Availability: Probability system operational at time t
 - ❖ Telecom: 99.999% → 5 mins./ year downtime
- Data integrity: no undetected errors
 - ❖ \$20K not interpreted as \$3,616

“Beagle2 mission presumed to be lost” (www.beagle2.com)

[Enterprise Networks / Outsourcing /](#)

EBay: One outage too many

Sign up to receive this and other networking newsletters in your inbox.

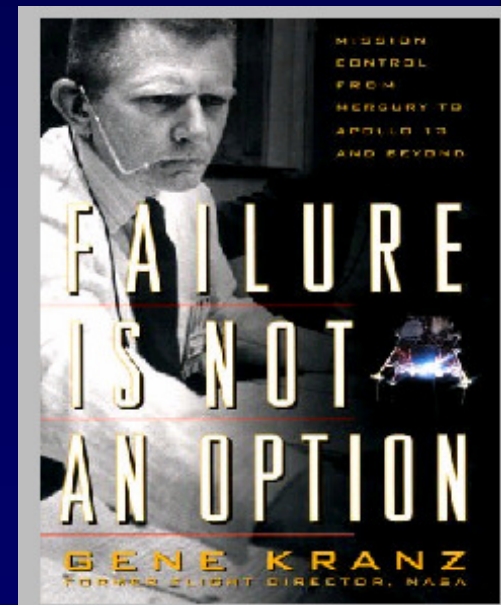
By Tim Wilson

Network World Outsourcing Newsletter, 04/15/02



Relat

More



Safety

- “Active” safety for drive-by-wire systems
- Implantable medical devices
- Nano-robot assisted remote surgery
- “Context-aware” “pro-active” healthcare system

The drive for silicon

Safety and entertainment rev up new chip opportunities

Text by Bernard Levine ■ Illustrations by Charles Mackey -- *Electronic Business*, 11/1/2003

If there's a light at the end of the downturn's tunnel, it's mobile computing. Not necessarily wireless networking, mind you, but rather the proliferation of electronics in automobiles.

Drive-by-wire a Reality

- What about reliability ?

Drive-by-wire needs real-time reliability

By Brian T. Murray, Manager, System Safety Engineer
E. Steele, Senior Staff Research Engineer, Delphi Auto
Systems, Warren, Ohio

[EE Times](#)

June 28, 2001 (9:55 a.m. ET)



Recent advances in dependable embedded-system technology, as well as continuing demand for improved handling and passive and active safety improvements, have led vehicle manufacturers and suppliers to work to develop computer-controlled, by-wire subsystems with no mechanical link to the driver. These include steer-by-wire and brake-by-wire and are composed of mechanically decoupled sets of

Recent

In F

- [High-sp
amps bu
SiGe](#)
- [Mixed S](#)
- [High-vo
support
phone f
Modeling](#)

Security

- Major adversaries
 - ❖ Security thefts
 - ❖ Virus, hacks, spam,
 - ❖ Terrorists

Internet architects zero in on reliability, security

By George Leopold

[EE Times](#)

May 29, 2001 (1:47 p.m. ET)



HERNDON, Va. — As the architects of the future Internet struggle to define underlying technologies for providing a range of new network services, reliability and security are again moving to the top of the agenda.

Power & Performance

Intel technologist cites power as biggest issue

By [Stephan Ohr](#)

[EE Times](#)

February 6, 2001 (7:52 a.m. ET)



SAN FRANCISCO — Moore's Law brings more than increases in the number of transistors per chip; it also brings dramatic increases in power consumption and power density. If current trends continue, you would have a device with 425 million transistors in 2005 and a processor with 1.8 billion transistors by 2010, said Pat Gelsinger, Intel's vice president and chief technology officer. You'd also have a heat generator with the intensity of a nuclear reactor, he

Recent Articles

EET

- [Sensor nets top R&D list for Homeland Security agency](#)
- [Digital camera shipments up 15% in November](#)
- [Boston to host UWB compatibility](#)

Server Reliability Goals

Server Reliability Categories and Criteria

Undetected errors corrupting customer data

- Typical system target: 1000 yr MTBF (114 FITS)

Detected errors causing system termination

- Corrupt data/control state in a global resource
- Target: 25 yr MTBF (4500 FITS unrecovered)

Detected errors causing application or partition termination

- Corrupt data/control state in a local resource
- Target: 10 yr MTBF (11400 FITS unrecovered)

MTBF = Mean Time Between Failures

Taken from Bossen, IRPS 2002

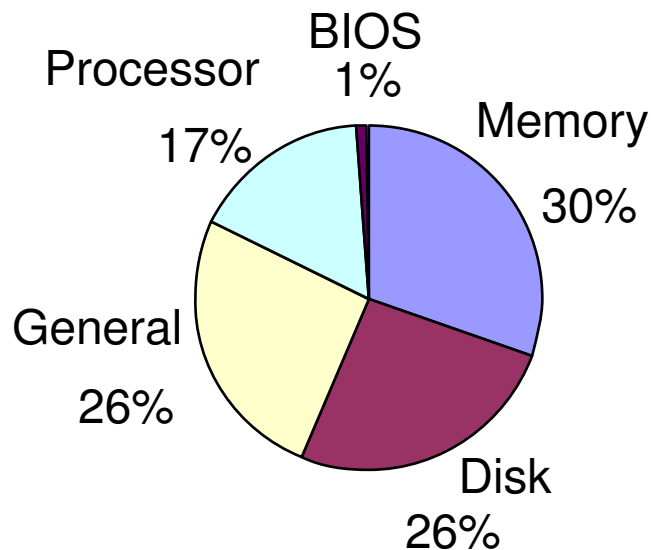
Causes of System Failures ?

- Depends on who you talk to
 - ❖ Application domains
 - PCs vs. servers
 - Medical devices, automotive, ...
 - ❖ System configurations (& costs)
 - Hardware costs & lifetime
 - Single vs. clusters
 - Application-specific vs. general-purpose

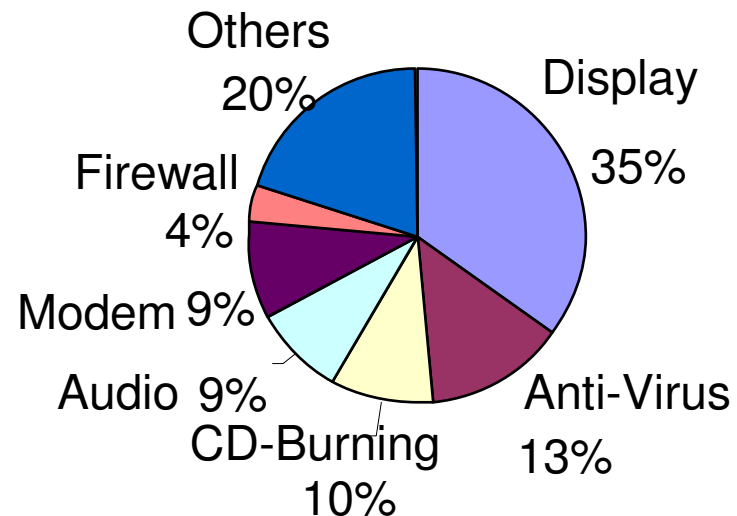
Windows XP Failures

- [Murphy, ACM Queue, Nov. 2004]
- 5% Microsoft software bugs, 12% hardware, 83% 3rd party – What you call a bug ?

Hardware failures



3rd party driver crashes

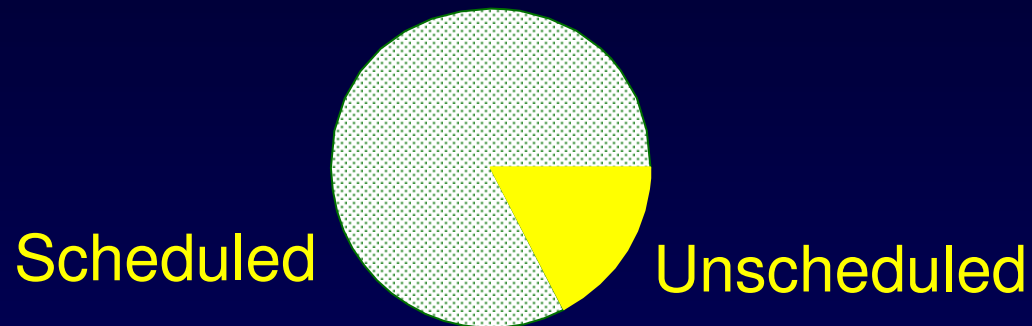


Windows XP Failures: Observations

- 3rd party drivers crashes – bug definition ?
- Increasing hardware failures – aging hardware?
- Good processor reliability enablers
 - ❖ Short PC lifetime
 - ❖ Speed & voltage guardbands during design
 - Price: power & performance cost
 - ❖ Classical scaling was sufficient
 - ❖ Inexpensive test & reliability screens
 - ❖ **BUT, progressively harder in sub-65nm**

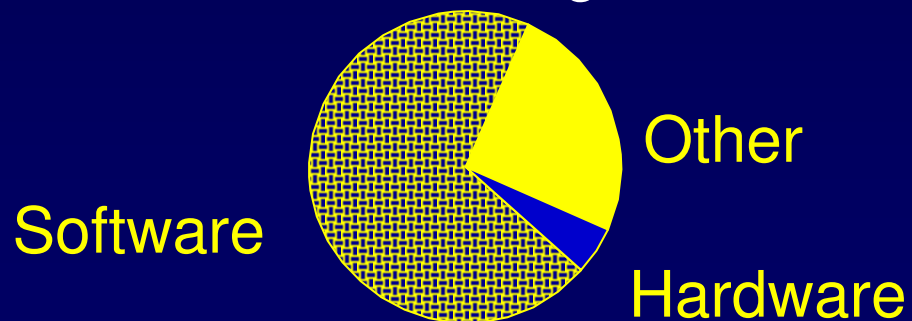
Causes of Server Unavailability – Data from the Past

Total Outage Cause



For 24x7 must address both scheduled & unscheduled

Unscheduled Outage Cause



Often operator error predominates as source of downtime

Ack: Lisa Spainhower, IBM

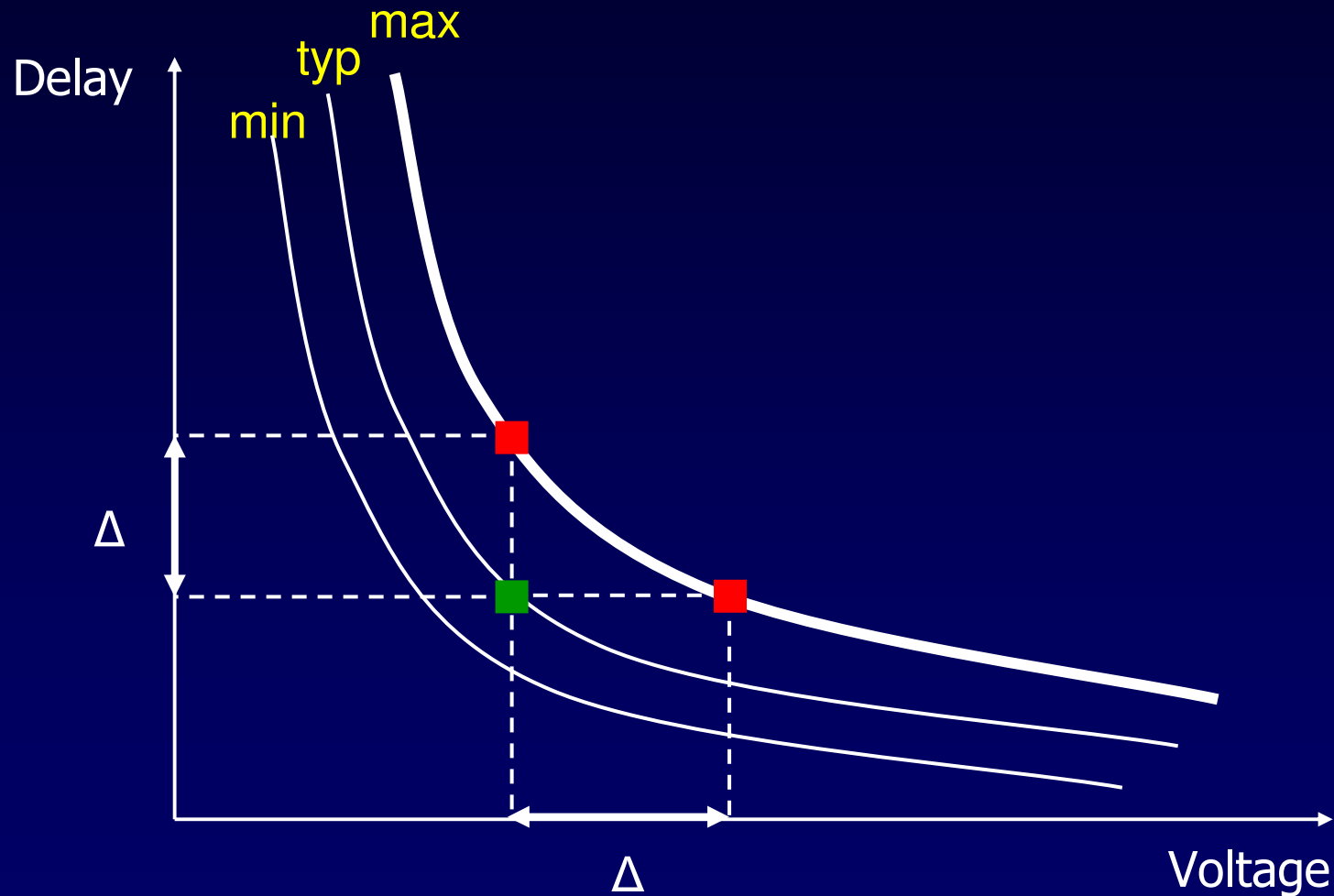
Server Failures: Observations

- Most software bugs “soft”
 - ❖ Heisenbugs – Gone after reboot / restart
 - ❖ Repair time is “key” here
- Operator errors – major issue going forward
- High hardware reliability
 - ❖ Enabled by hardware redundancy, BUT
 - Redundancy expensive
 - Hardware failure rates increasing
 - Performance & power scaling slowdown

Why Worry About Hardware Reliability?

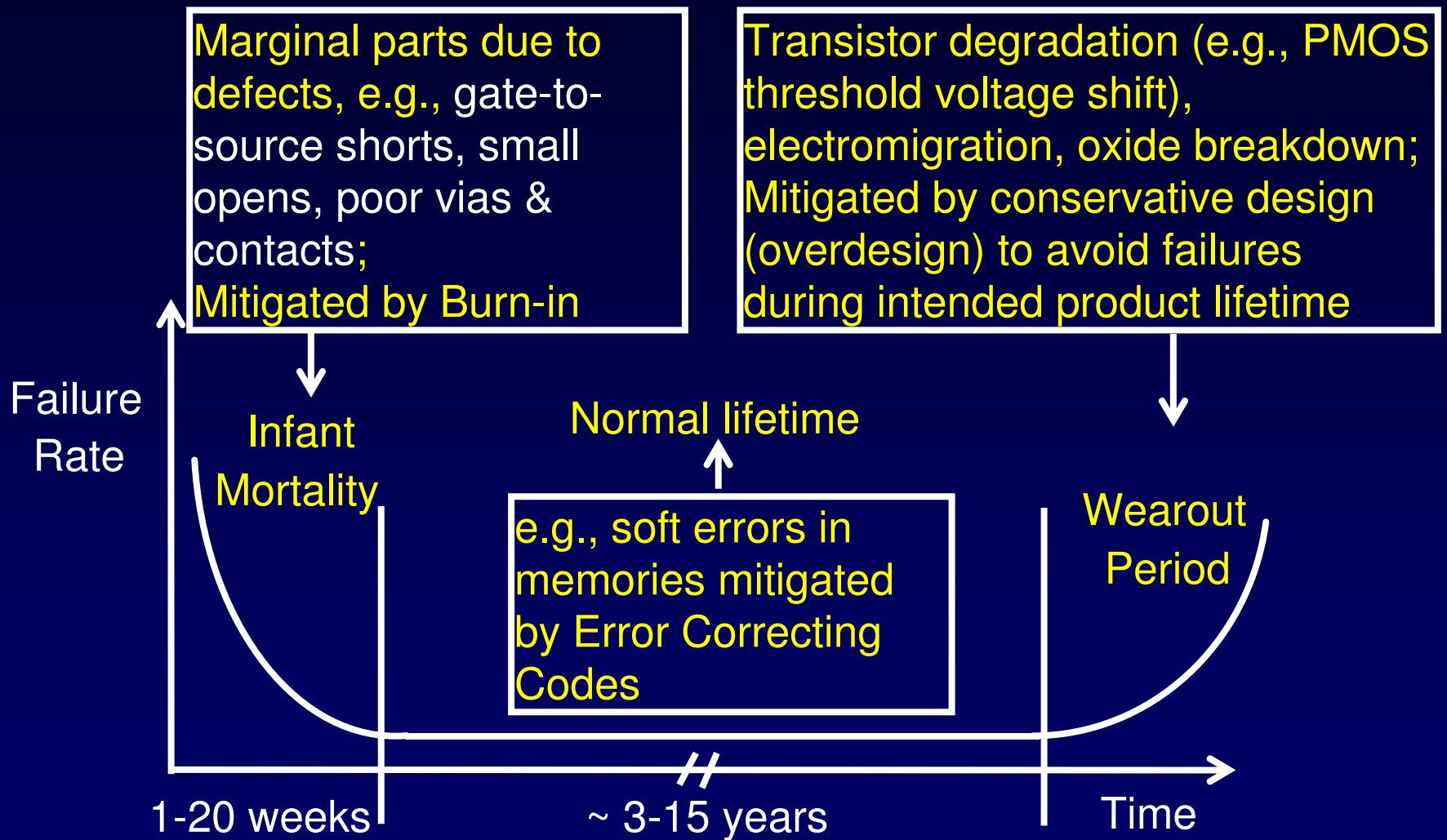
- Major process variation
 - ❖ Worst-case design impractical
- Perfect design verification + test not enough
 - ❖ Manufacturing process imperfect
 - ❖ Testing imperfect: Warranty failures
 - ❖ Transient errors during system operation
 - e.g., noise, radiation induced soft errors
 - ❖ “Aging”: e.g., slow transistors with time

Process variation: Power & Performance Impact

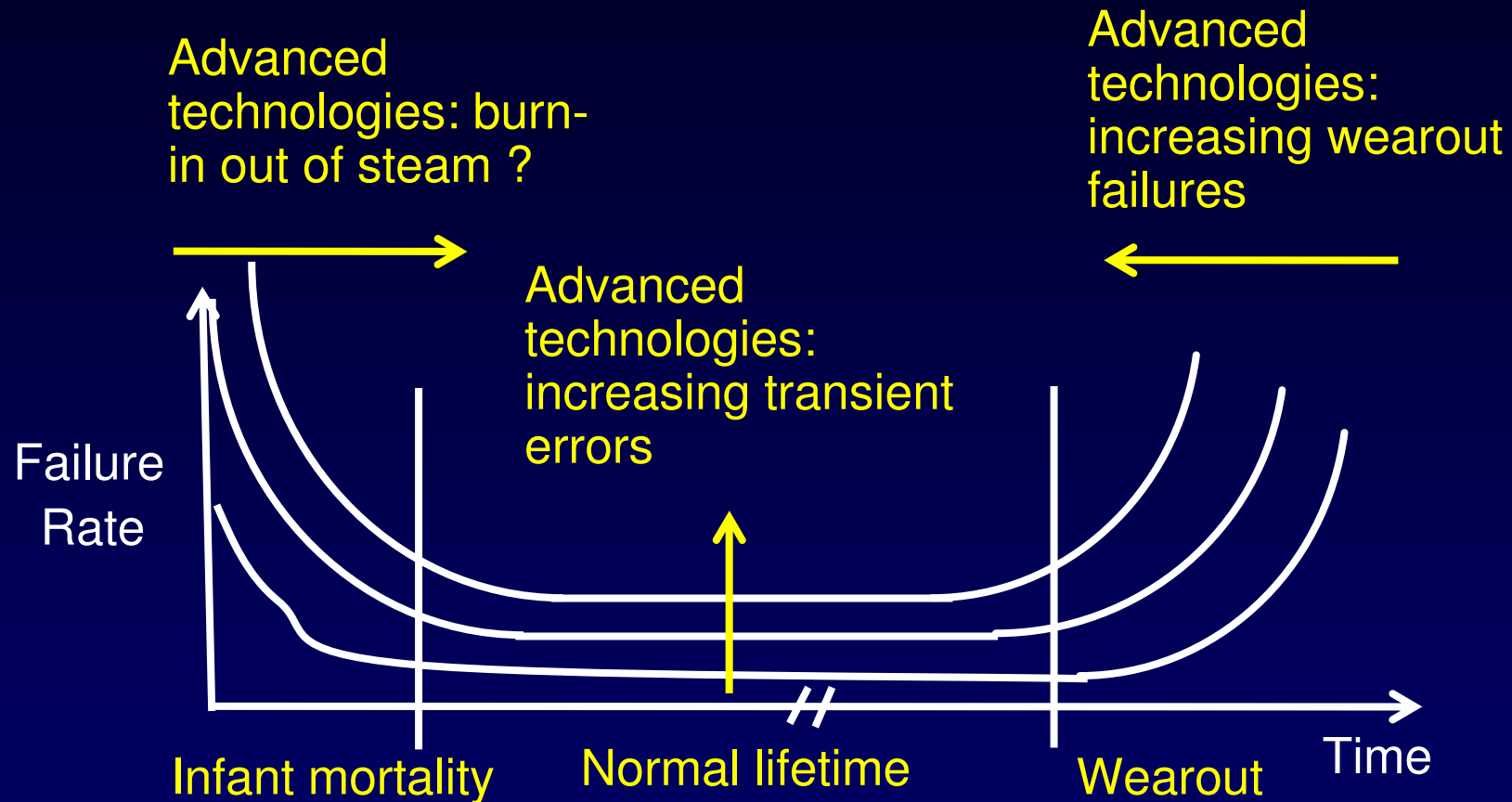


- Ack: Prof. Giovanni De Micheli

Bathtub Curve



(Scary?) Bathtub: Future Technologies



- Exciting opportunities for new system design techniques to cope with failures

Related EE392U Seminars

- Larry Votta, Distinguished Engineer, SUN – Oct. 3
- “Why Do Systems Fail ?” – Oct. 31
 - ❖ Lisa Spainhower, Distinguished Engineer, IBM
- “Estimating the Risk of Releasing Software,” – Nov. 7
 - ❖ Brendan Murphy, Microsoft Research
- “Reliable Design from Unreliable Components” – Nov. 14
 - ❖ Shekhar Borkar, Fellow, Intel
- Columbia Disaster Talk – Nov. 28
 - ❖ Prof. Greg Kovacs, Stanford

How to Build Robust Systems ?

Avoidance

- Conservative design
- Design validation
- Thorough hardware & software test
- Infant mortality screen for hardware
- Transient error avoidance
- Proper interfaces to minimize operator errors

Correct by Construction Simply Not sufficient
Several challenges in future

How to Build Robust Systems? Tolerance

- Error detection during system operation
 - ❖ Permanent & correlated hardware failures ?
 - ❖ Bohrbugs vs. Heisenbugs ?
- On-line monitoring & diagnostics
- Self-recovery & Self-repair
- Automated self-managing systems
- Major Challenge: PROVE these WORK !

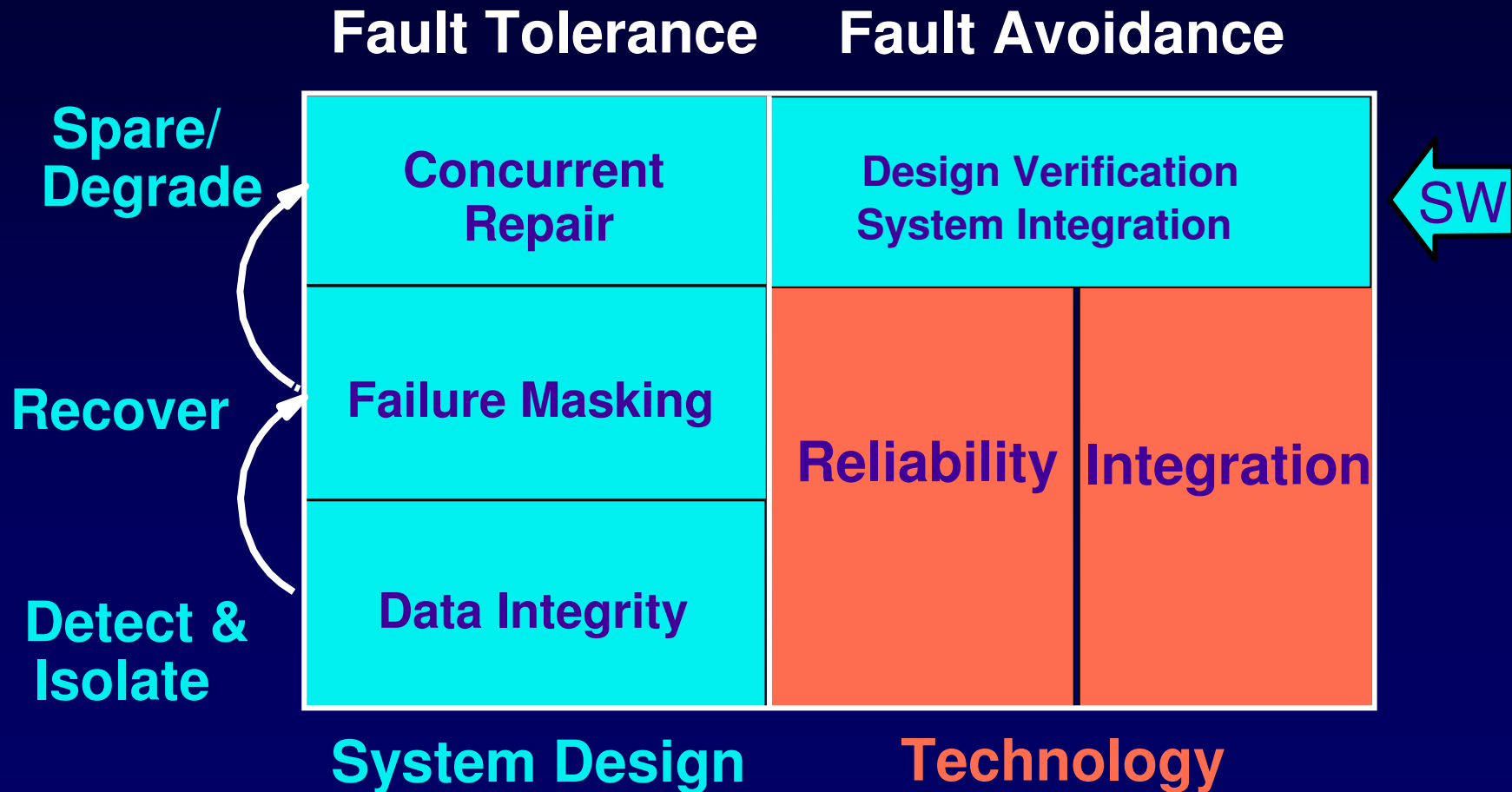
Classical fault-tolerance very expensive
Classical fault-tolerance inadequate

Fault-Tolerant Computing

fault-tol·er·ant \'fölt-'täl(-ə)-rənt\
adj : able to function in the
absence of a major component



High Availability Building Blocks



Related EE392U Seminars

- Larry Votta, Distinguished Engineer, SUN – Oct. 3
- System effects & error protection – Oct. 10
 - ❖ Prof. Ravi Iyer, University of Illinois at Urbana Champaign
- “Fault Tolerance in Space Environments” – Oct. 17
 - ❖ Dr. Philip Shirvani, nVidia
- Trusted systems: Prof. Hector Garcia Molina – Oct. 24
- “Why Do Systems Fail ?” – Oct. 31
 - ❖ Lisa Spainhower, Distinguished Engineer, IBM
- “Estimating the Risk of Releasing Software,” – Nov. 7
 - ❖ Brendan Murphy, Microsoft Research

Robust Systems as Research Area – CRA Recommendations

- Trouble-free systems
 - ❖ PCs – “zero administration”
 - ❖ Large-scale systems
 - Millions of users
 - Administered by single person
 - ❖ Self-diagnosing, self-healing, self-evolving
- Dependable and survivable systems
 - ❖ Secure, safe, reliable, available

Importance in Revolutionary Nanotechnology

- Revolutionary nanotechnologies
 - ❖ e.g., Molecular electronics
- Well-acknowledged fact
 - ❖ Regular structures
 - ❖ Defect prone
 - ❖ Errors during normal operation
 - ❖ ~ 5-10% faulty
- Must be self-healing !

Broad Research Directions: Looking for Interested Students

- Understand failures for various applications
 - ❖ PCs, large servers, embedded (e.g., cars, digital home), space systems, healthcare
 - ❖ Expertise required
 - Experimental data collection
 - Simulation & modeling
 - Circuits, architecture, systems, HCI

Broad Research Directions: Looking for Interested Students

- New robust system design techniques
 - ❖ Failure avoidance & resilience support
 - ❖ Expertise required
 - Circuit / logic design
 - Architecture
 - Compiler & operating systems
 - Human-Computer interaction

Broad Research Directions: Looking for Interested Students

- Robust system prototypes
 - ❖ PROVE that the system works !
 - How ?
 - Not just simulation
 - Build real system prototypes – How ?
- Nanotechnology architectures
 - ❖ Built-in defect & fault-tolerance ?
 - Conventional methods work for very low failure rates