the leading edge of information technology

INTERNATIONAL FEDERATION FOR INFORMATION PROCESSING

WG 10.4 — DEPENDABLE COMPUTING AND FAULT TOLERANCE

http://www.dependability.org/wg10.4

47TH MEETING — RINCÓN, PR, USA

JANUARY 26–30, 2005

Reflection on the Pool and Ocean at Sunset

Toulouse, November 2005

Credits: Cover photographs by Brian Randell (front) and Jean Arlat (back)

the leading edge of information technology

INTERNATIONAL FEDERATION FOR INFORMATION PROCESSING

WG 10.4 — DEPENDABLE COMPUTING AND FAULT TOLERANCE

[http://www.dependability.org/wg10.4]

Chairman: Vice Chairmen:

Jean Arlat
LAAS-CNRS
7, Avenue du Colonel Roche
31077 Toulouse Cedex 4
France

Phone: +33 5 61 33 62 33
Fax: +33 5 61 33 64 11
EMail: Jean.Arlat@laas.fr

Takashi Nanya
RCAST
University of Tokyo
4-6-1, Komaba, Meguro-ku
Tokyo 153-8904
Japan

Phone: +81 3 5452 5160
Fax: +81 3 5452 5161
EMail: nanya@hal.rcast.u-tokyo.ac.jp

William H. Sanders
CRHC - Coordinated Science Lab.
University of Illinois at Urbana-Champaign
1308 West Main Street
Urbana, IL 61801
USA

Phone: +1 217 333 0345
Fax: +1 217 244 3359
EMail: whs@crhc.uiuc.edu

Organizers:

47TH MEETING OF IFIP WG 10.4

RINCÓN, PR, USA

January 26-30, 2005

Nicholas S. Bowen
IBM Server Group
11400 Burnet Road - B905/4B018
Austin, TX 78758 – USA
Phone: +1 512 838 3865
Fax: +1 512 838 4025
Email: bowenn@us.ibm.com

T. Basil Smith
M.S. 4S-A26 IBM
19 Skyline Drive
HAWTHORNE, NY 10532 – USA
Phone: +1 914 784 7018
Fax: +1 914 784 6201
Email: tbsmith@us.ibm.com

William H. Sanders
CRHC - Coordinated Science Lab.
University of Illinois
1308 West Main Street
Urbana, IL 61801 – USA
Phone: +1 217 333 0345
Fax: +1 217 244 3359
Email: whs@crhc.uiuc.edu

Carl E. Landwehr
Program Director, Cyber Trust
CISE/CNS
National Science Foundation
4201 Wilson Boulevard
Arlington, VA 22230 – USA
Phone: +1 703 292 8950
Fax: +1 703 292 9059
Email: clandweh@nsf.gov

Bienvenido Velez-Rivera
Dept. Electrical & Computer Eng.
University of Puerto Rico
P.O. Box 9042
Mayaguez, PR 00681– USA
Phone: +1 787 831 3244
Fax: +1 787 833 3331
Email: bvelez@ece.uprm.edu

CONTENTS

Program of the Meeting

Workshop on *Autonomic Web Computing*

Coordinators: **Nicholas S. Bowen**, IBM Systems Group, Austin, TX, USA
T. Basil Smith, IBM Research, Hawthorne, NY, USA
William H. Sanders, UIUC, Urbana-Champaign, IL, USA
Carl E. Landwehr, NSF, Arlington, VA, USA

First Day of Workshop — *Thursday, January 27*

Nicholas S. Bowen
Introduction and Workshop Structure

Session 1 – Platform Infrastructure
Moderator: T. Basil Smith

Steve Hunter, IBM Server Group, Research Triangle Park, NC, USA
IBM BladeCenter as a Dependable Web Infrastructure Platform

Dwight Barron, HP Industry Standard Servers, Houston, TX, USA
HP BladeSystem Reliable Web Services

Rich Oehler, Newisys, Austin, TX, USA
Ideas for a Dependable 'Industry Standard Architecture' Platform

Marc Rougier, Meiosys, Toulouse, France
Non-intrusive Middleware for Continuity of Service: Protection Against System Failures

Session 2 – Autonomic Response to Faults and Attacks
Moderator: William H. Sanders

Nicholas S. Bowen
Autonomic Computing: An Overview

Kimberly Keeton, HP Laboratories, Palo Alto, CA, USA
Automating Data Dependability

Ravishankar K. Iyer, UIUC, Urbana-Champaign, IL, USA
Adaptive Application-Aware Runtime Checking

Second Day of Workshop — *Saturday, January 29*

Session 3 – Security
Moderator: Carl E. Landwehr

Brian A. LaMacchia, Microsoft Corporation, Redmond, WA, USA
Security Attacks Security Attacks and Defenses

George Robert Blakley III, IBM, Round Rock, TX, USA
Security in Autonomic Web Computing

John R. Black, University of Colorado at Boulder, USA
Practical Cryptography and Autonomic Web Computing

Elisa Bertino, Purdue University, West Lafayette, IN, USA
A Flexible Access Control Model for Web Services

Sanjai Narain, Telcordia Technologies Research, Piscataway, NJ, USA
Web Services Security Configuration Challenges

Session 4 – Synthesis and Wrap Up
Moderator: Nicholas S. Bowen

T. Basil Smith
Summary of Session 1

William H. Sanders
Summary of Session 2

Carl E. Landwehr
Summary of Session 3

IFIP WG 10.4 Business Meeting

Jean Arlat, LAAS-CNRS, Toulouse, France

Jean Arlat
Overall Presentation and News

Jean-Claude Laprie, LAAS-CNRS, Toulouse, France
Report on 18th IFIP WCC-2004

Takashi Nanya, University of Tokyo, Japan
Update on IEEE/IFIP DSN-2005

Chandra M.R. Kintala, Stevens Inst. of Technology, Hoboken, NJ, USA
Update on IEEE/IFIP DSN-2006

Takashi Nanya
Update on 48th IFIP WG 10.4 Meeting (Hakone, Japan)

Research Reports — *Sunday, January 30*

Session 1: Moderator: Takashi Nanya

John Rushby, SRI International, Menlo Park, CA, USA
Automated Test Generation with "sal-atg"

Philip Koopman, Carnegie Mellon University, Pittsburgh, PA, USA
Thoughts on Embedded Security

Brian Randell, University of Newcastle upon Tyne, UK
RODIN: Rigorous Open Development Environment for Complex Systems

Paulo Esteves Veríssimo, University of Lisbon, Portugal
On Detours and Shortcuts to Solve Distributed Systems Problems

Session 2: Moderator: Jean Arlat

Priya Narasimhan, Carnegie Mellon University, Pittsburgh, PA, USA
MEAD: Middleware for Embedded Adaptive Dependability

Kevin Driscoll, Honeywell Laboratories, Minneapolis, MN, USA
Byzantine Filtering

Lisa Spainhower, IBM, Poughkeepsie, NY, USA
Upcoming IBM Sponsored/Contributing Activities & Research

Lorenzo Alvisi, University of Texas at Austin, USA
Byzantine Faults in a Rational World

Attendance List

Prof. ABRAHAM Jacob A.
CERC, ACE 6.134, C 8800
University of Texas at Austin
AUSTIN, TX 78712-1014 – USA
Tel : (+1) 512 471 8983
Fax : (+1) 512 471 8967
Email : jaa@cerc.utexas.edu

Prof. ALVISI Lorenzo
Taylor Hall 2.124
University of Texas at Austin
AUSTIN, TX 78712-1014 – USA
Tel : (+1) 512 471 9792
Fax : (+1) 512 232 7886
Email : lorenzo@cs.utexas.edu

Dr. ARLAT Jean
LAAS - CNRS
7, avenue du Colonel Roche
31077 TOULOUSE Cedex 4 – FRANCE
Tel : (+33) 05 61 33 62 33
Fax : (+33) 05 61 33 64 11
Email : Jean.Arlat@laas.fr

Dr BARRON Dwight
Hewlett-Packard Company
20555 SH 249 MS 100210
HOUSTON , TX 77070 – USA
Tel : (+1) 281 514 2769
Fax : (+1) 281 518 5925
Email : dwight.barron@hp.com

Prof. BERTINO Elisa
Department of Computer Science,
Purdue University
250 N University Street
WEST LAFAYETTE, IN 47907-1315 – USA
Tel : (+1)
Fax : (+1) 765 494 0739
Email : bertino@cs.purdue.edu

Dr. BLACK, JR John R.
Dept. CS - 430 UCB
University of Colorado
BOULDER, CO 80309 – USA
Tel : (+1) 303 492 0573
Fax : (+1) 303 492 2844
Email : jrblack@cs.colorado.edu

Dr. BLAKLEY Bob
Security & Privacy IBM
914 Blue Spring Circle
ROUND ROCK, TX 78681 – USA
Tel : (+1) 512 286 2240
Fax : (+1) 512 286 2057
Email : blakley@us.ibm.com

Dr. BOWEN Nicholas S.
IBM Server Group
11400 Burnet Road - B905/4B018
AUSTIN TEXAS, 78758 – USA
Tel : (+1) 512 838 3865
Fax : (+1) 512 838 4025
Email : bowenn@us.ibm.com

Mr. DRISCOLL Kevin R.
MN65-2200 Honeywell Laboratories
3660 Technology Drive
MINNEAPOLIS, MN 55418-1006 – USA
Tel : (+1) 612 951 7263
Fax : (+1) 612 951 7438
Email : kevin.driscoll@honeywell.com

Dr. HEIMERDINGER Walter
MN 65-2200 Honeywell Laboratories
3660 Technology Drive
MINNEAPOLIS, MN 55418-1006 – USA
Tel : (+1) 612 951 7333
Fax : (+1) 612 951 7438
Email : walt.heimerdinger@honeywell.com

Dr. HUNTER Steven W.
Research Triangle
IBM Research
3039 Cornwallis Road
RALEIGH, NC 27709 – USA
Tel : (+1) 919 254 2984
Fax : (+1) 919 543 4268
Email : hunters@us.ibm.com

Prof. IYER Ravishankar K.
CRHC -255 Coordinated Science Lab.
University of Illinois
1308 West Main Street
URBANA, IL 61801 – USA
Tel : (+1) 217 333 2510
Fax : (+1) 217 244 1764
Email : iyer@crhc.uiuc.edu

Dr. KEETON Kimberly
MS 1U-13 Hewlett-Packard Labs.
1501 Page Mill Road
PALO ALTO, CA 94304-1126 – USA
Tel : (+1) 650 857 3990
Fax : (+1) 650 857 5548
Email : kkeeton@hpl.hp.com

Prof. KINTALA Chandra M.R.
Electrical & Computer Engineering
Stevens Institute of Technology
Castle Point on Hudson
HOBOKEN, NJ 07030 – USA
Tel : (+1) 201 216 8057
Fax : (+1) 201 216 8246
Email : chandra@kintala.com

Prof. KOOPMAN Philip
ECE Dept. - HH A-308
Carnegie Mellon University
PITTSBURGH, PA 15213 – USA
Tel : (+1) 412 268 5225
Fax : (+1) 412 268 6353
Email : koopman@cmu.edu

Dr. LAMACCHIA Brian
Microsoft Co.
1 Microsoft Way
REDMOND, WA 98052 – USA
Tel : (+1) 425 703 9906
Fax : (+1) 206 936 7329
Email : bal@microsoft.com

Dr. LANDWEHR Carl E.
Program Director, Cyber Trust
CISE/CNS
National Science Foundation
4201 Wilson Boulevard
ARLINGTON, VA 22230 – USA
Tel : (+1) 703 292 8950
Fax : (+1) 703 292 9059
Email : clandweh@nsf.gov

Dr. LAPRIE Jean-Claude
LAAS - CNRS
7, avenue du Colonel Roche
31077 TOULOUSE Cedex 4 – FRANCE
Tel : (+33) 05 61 33 78 85
Fax : (+33) 05 61 33 64 11
Email : Jean-Claude.Laprie@laas.fr

Prof. MADEIRA Henrique
Dep. Eng. Informatica
Universidade de Coimbra
Pinhal de Marrocos
P - 3030-290 COIMBRA – PORTUGAL
Tel : (+351) 2 39 790 003
Fax : (+351) 2 39 701 266
Email : henrique@dei.uc.pt

Dr. MAXION Roy A.
Dept. of Computer Science
Carnegie Mellon University
PITTSBURGH, PA 15213-3890 – USA
Tel : (+1) 412 268 7556
Fax : (+1) 412 268 5576
Email : maxion@cs.cmu.edu

Prof. MEYER John F.
EECS Dept. - 4111 EECS Bldg
University of Michigan
1301 Beal Ave.
ANN ARBOR, MI 48109-2122 – USA
Tel : (+1) 734 763 0037
Fax : (+1) 734 763 1503
Email : jfm@eecs.umich.edu

Prof. NANYA Takashi
RCAST
University of Tokyo
4-6-1 Komaba, Meguro-ku
TOKYO, 153-8904 – JAPAN
Tel : (+81) 3 5452 5160
Fax : (+81) 3 5452 5161
Email : nanya@hal.rcast.u-tokyo.ac.jp

Dr. NARAIN Sanjai
Room 1N-375
Telcordia Technologies, Inc.
1 Telcordia Drive
PISCATAWAY, NJ 08854 – USA
Tel : (+1) 732 699 2806
Fax : (+1)
Email : narain@research.telcordia.com

Prof. NARASIMHAN Priya
Dept. ECE & CS - HH A303
Carnegie Mellon University
PITTSBURGH, PA 15213 – USA
Tel : (+1) 412 268 8801
Fax : (+1) 412 268 6353
Email : priya@cs.cmu.edu

Mrs OBERNDORF Patricia
Director, Dynamic Systems
Software Engineering Institute
Carnegie Mellon University
4500 Fifth Ave.
PITTSBURGH, PA 15213 – USA
Tel : (+1) 412 268 6138
Fax : (+1) 412 268 5758
Email : po@sei.cmu.edu

Mr. OEHLER Rich
Building 4, Suite 300
Newisys, Inc.
10814 Jollyville Road
AUSTIN, TX 78759 – USA
Tel : (+1) 914 649 8609
Fax : (+1) 512 349 9927
Email : rich.oehler@newisys.com

Prof. ORTIZ Jorge L.
Dept. Electrical & Computer Eng.
University of Puerto Rico
P.O. Box 9042
MAYAGUEZ, PR, 00681 – USA
Tel : (+1) 787 265 3821
Fax : (+1) 787 831 7564
Email : jortiz@ece.uprm.edu

Dr. POWELL David
LAAS - CNRS
7, avenue du Colonel Roche
31077 TOULOUSE Cedex 4 – FRANCE
Tel : (+33) 05 61 33 62 87
Fax : (+33) 05 61 33 64 11
Email : david.powell@laas.fr

Prof. RANDELL Brian
School of Computing Science
University of Newcastle upon Tyne
Claremont Road, Claremont Tower
NEWCASTLE UPON TYNE, NE1 7RU – ,UK
Tel : (+44) 191 222 7923
Fax : (+44) 191 222 8232
Email : brian.randell@ncl.ac.uk

M. ROUGIER Marc
MEIOSYS
42, avenue du Général de Croutte
31100 TOULOUSE – FRANCE
Tel. (+33) 5 34 63 85 00
Fax. (+33) 5 61 40 04 20
Email. mrougier@meiosys.com

Dr. RUSHBY John
Computer Science Laboratory
SRI International
333 Ravenswood Avenue
MENLO PARK, CA 94025 – USA
Tel : (+1) 650 859 5456
Fax : (+1) 650 859 2844
Email : rushby@csl.sri.com

Prof. SANDERS William H.
CRHC - Coordinated Science Lab.
University of Illinois
1308 West Main Street
URBANA, IL 61801 – USA
Tel : (+1) 217 333 0345
Fax : (+1) 217 244 3359
Email : whs@uiuc.edu

Mr. SAYDJARI Omar Sami
Cyber Defense Agency, LLC
3601 43rd Street South
WISCONSIN RAPIDS, WI 54494 – USA
Tel : (+1) 715 424 2642
Fax : (+1) 715 424 2638
Email : ssaydjari@cyberdefenseagency.com

Dr. SCHLICHTING Richard D.
Shannon Laboratory, E221
AT&T Labs Research
180 Park Avenue
FLORHAM PARK, NJ 07932 – USA
Tel : (+1) 973 360 8234
Fax : (+1) 973 360 8077
Email : rick@research.att.com

Dr. SMITH T. Basil
M.S. 4S-A26 IBM
19 Skyline Drive
HAWTHORNE, NY 10532 – USA
Tel : (+1) 914 784 7018
Fax : (+1) 914 784 6201
Email : tbsmith@us.ibm.com

Dr. SOKOL Tammy
Research Triangle
IBM Server Group
11400 Burnet Road - B905/4B018
AUSTIN TEXAS, 78758 – USA
Tel : (+1) 512 838 0795
Fax : (+1)
Email : tsokol@us.ibm.com

Ms. SPAINHOWER Lisa
M/S P314 IBM
2455 South Road
POUGHKEEPSIE, NY 12601 – USA
Tel : (+1) 845 435 6485
Fax : (+1) 845 432 9413
Email : lisa@us.ibm.com

Prof. SURI Neeraj
Fachbereich Informatik/Dept. of Computer Science
Technische Universität Darmstadt
Hochschulst. 10
D-64289 DARMSTADT – GERMANY
Tel : (+49) 6151 16 3513
Fax : (+49) 6151 16 4310
Email : suri@informatik.tu-darmstadt.de

Dr. TAN Kymie
Department of Computer Science
Carnegie Mellon University
PITTSBURGH, PA 15213 – USA
Tel : (+1) 412 268 3266
Fax : (+1) 412 268 5576
Email : kmct@cs.cmu.edu

Mr. VELEZ-RIVERA Bienvenido
Dept. Electrical & Computer Eng.
University of Puerto Rico
P.O. Box 9042
MAYAGUEZ, PR, 00681USA
Tel : (+1) 787 831 3244
Fax : (+1) 787 833 3331
Email : bvelez@ece.uprm.edu

Prof. VERISSIMO Paulo
Bloco C6-Dept. of Informatics
University of Lisboa
Campo Grande
1749-016 LISBOA – PORTUGAL
Tel : (+351) 21 750 01 03
Fax : (+351) 21 750 00 84
Email : pjv@di.fc.ul.pt

Dr. WALTER Chris J.
WW Technology Group
4519 Mustering Drum
ELLICOTT CITY, MD 21042-5949 – USA
Tel : (+1) 410 418 4353
Fax : (+1) 410 418 4355
Email : cwalter@wwtechnology.com

Dr. WEINSTOCK Charles B.
Software Engineering Institute
Carnegie Mellon University
PITTSBURGH, PA 15213 – USA
Tel : (+1) 412 268 7719
Fax : (+1) 412 268 5758
Email : weinstock@sei.cmu.edu

Workshop

Autonomic Web Computing

Session 1

Platform Infrastructures

Moderator and Rapporteur

T. Basil Smith, IBM Research, Hawthorne, NY, USA

IBM eServer BladeCenter as a Dependable Web Infrastructure Platform

Steven W Hunter

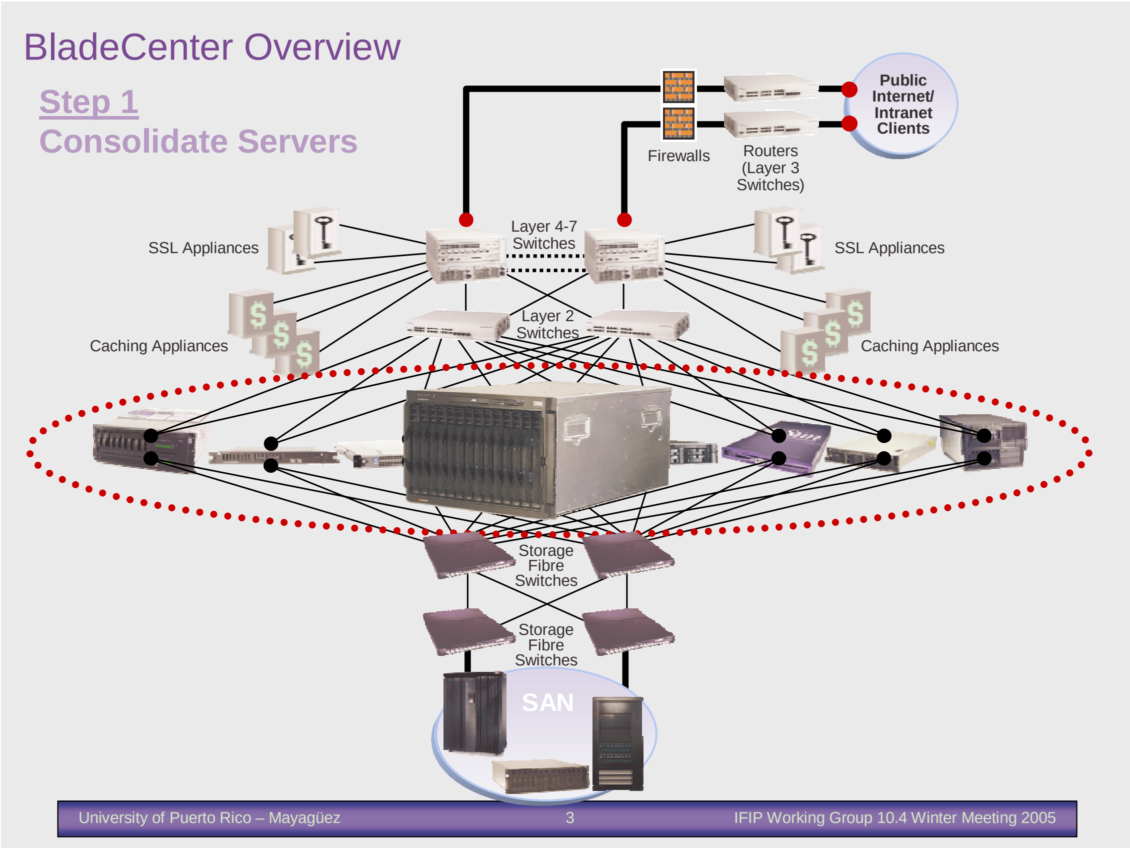
IBM Corporation
1/27/2005

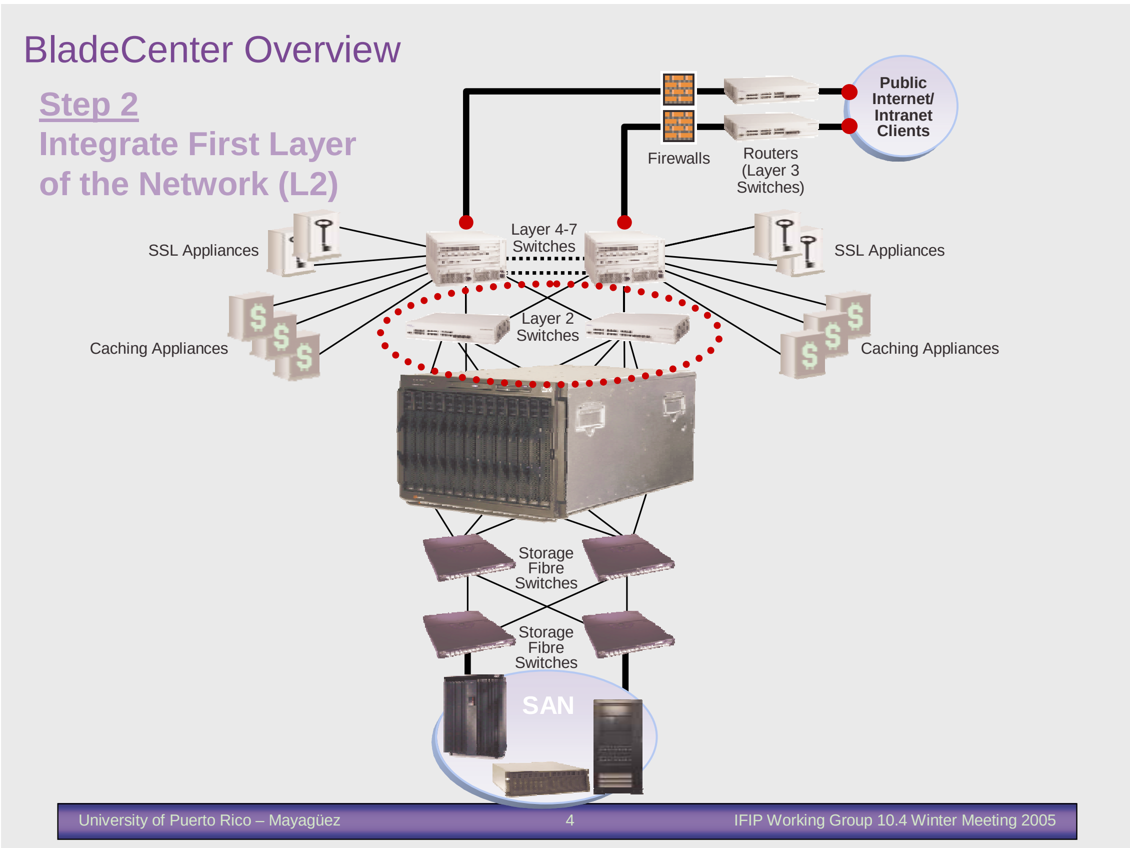
IFIP Working Group 10.4
Winter Meeting 2005
University of Puerto Rico
Mayaguez, Puerto Rico

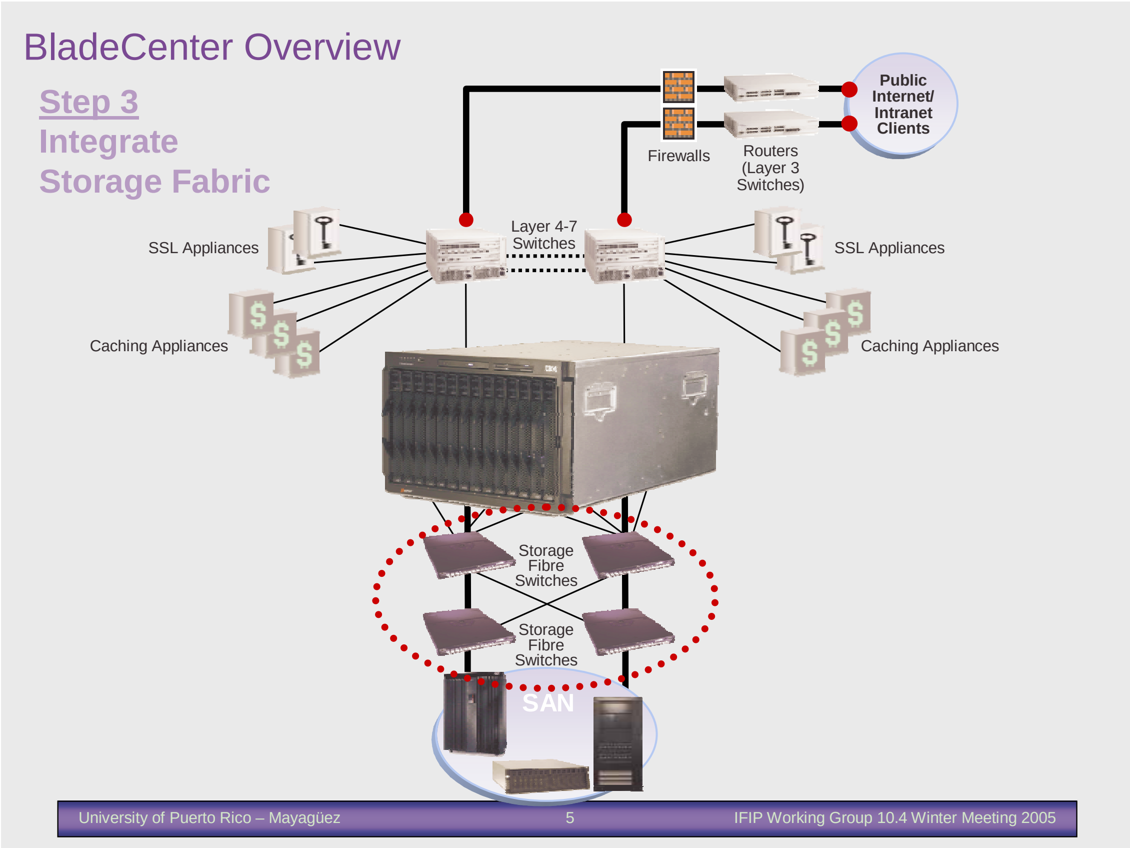
BladeCenter Overview

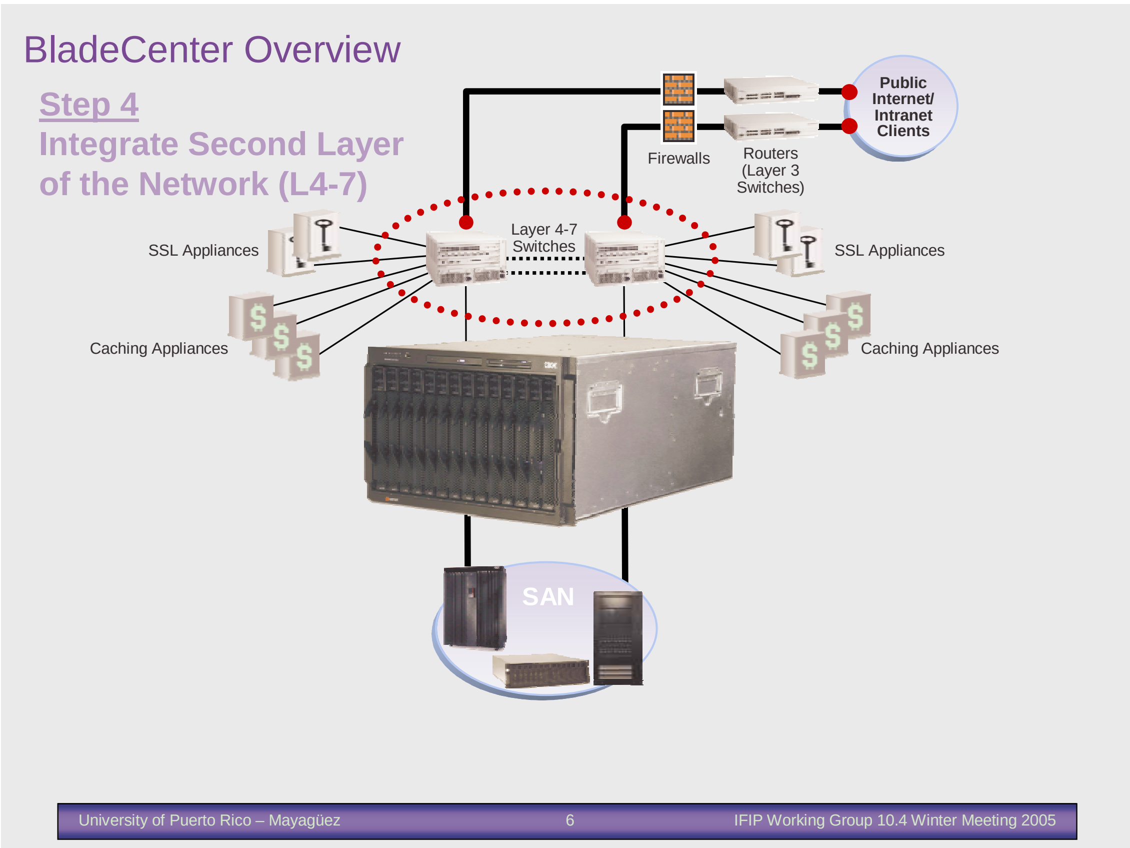
- **Modular, Scalable**
  - ➢ **1 – 14 Processor Blades**
- **Density with Performance**
  - ➢ **7U Mechanical Chassis**
- **Integrated Network Infrastructure**
  - ➢ **Switching with point-to-point blade connections**
- **Affordable Availability**
  - ➢ **Redundant, Hot-swappable blades and modules**
- **Advanced Systems Management**
  - ➢ **Integrated service processor**

University of Puerto Rico – Mayagüez 2 IFIP Working Group 10.4 Winter Meeting 2005


BladeCenter Overview
Step 1
Consolidate Servers
Firewalls
Routers (Layer 3 Switches)
Public Internet/ Intranet Clients
SSL Appliances
Layer 4-7 Switches
SSL Appliances
Layer 2 Switches
Caching Appliances
Caching Appliances
Storage Fibre Switches
Storage Fibre Switches
SAN
University of Puerto Rico – Mayagüez
3
IFIP Working Group 10.4 Winter Meeting 2005

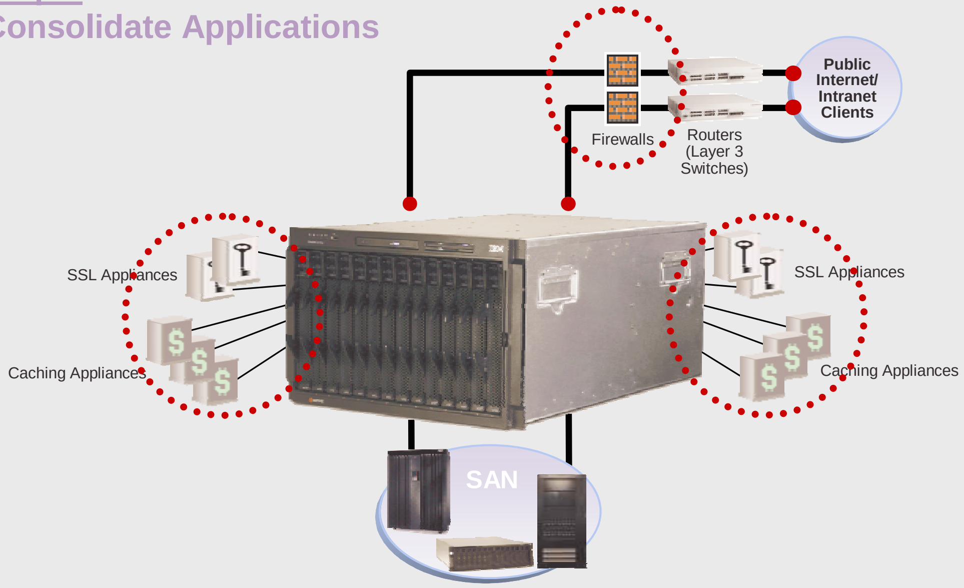

BladeCenter Overview
Step 2
Integrate First Layer of the Network (L2)
Firewalls
Routers (Layer 3 Switches)
Public Internet/ Intranet Clients
SSL Appliances
Layer 4-7 Switches
SSL Appliances
Layer 2 Switches
Caching Appliances
Caching Appliances
Storage Fibre Switches
Storage Fibre Switches
SAN
University of Puerto Rico – Mayagüez
4
IFIP Working Group 10.4 Winter Meeting 2005


BladeCenter Overview
Step 3
Integrate
Storage Fabric
Firewalls
Routers
(Layer 3
Switches)
Public
Internet/
Intranet
Clients
Layer 4-7
Switches
SSL Appliances
SSL Appliances
Caching Appliances
Caching Appliances
Storage
Fibre
Switches
Storage
Fibre
Switches
SAN
University of Puerto Rico – Mayagüez
5
IFIP Working Group 10.4 Winter Meeting 2005


BladeCenter Overview
Step 4
Integrate Second Layer
of the Network (L4-7)
Firewalls
Routers
(Layer 3
Switches)
Public
Internet/
Intranet
Clients
Layer 4-7
Switches
SSL Appliances
SSL Appliances
Caching Appliances
Caching Appliances
SAN
University of Puerto Rico – Mayagüez
6
IFIP Working Group 10.4 Winter Meeting 2005

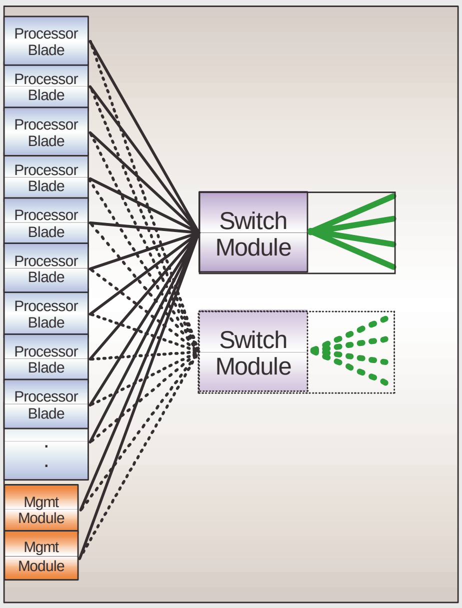
BladeCenter Overview

Step 5
Consolidate Applications



University of Puerto Rico – Mayagüez 7 IFIP Working Group 10.4 Winter Meeting 2005

BladeCenter Overview



Switching Modules
- Fibre Channel
- Ethernet
- Others…

Blade I/O Card
- I/O expansion card matches switch technology in the corresponding slot

University of Puerto Rico – Mayagüez 8 IFIP Working Group 10.4 Winter Meeting 2005

BladeCenter Overview

Gigabit Ethernet Switches (Layer 2)
- **Commodity level networking**
- **Link aggregation**
- **VLAN partitioning and management**

Advanced Switching (Layer 2-7)
- **Load Balancing**
- **Content-based switching**

Fibre Channel Switches
- **Lower cost via integration**
- **Full support of FC-SW-2 standards**

Power (4 x 1800W load-balancing)
- **Upgradeable as required**
- **Redundant and load balancing for HA**

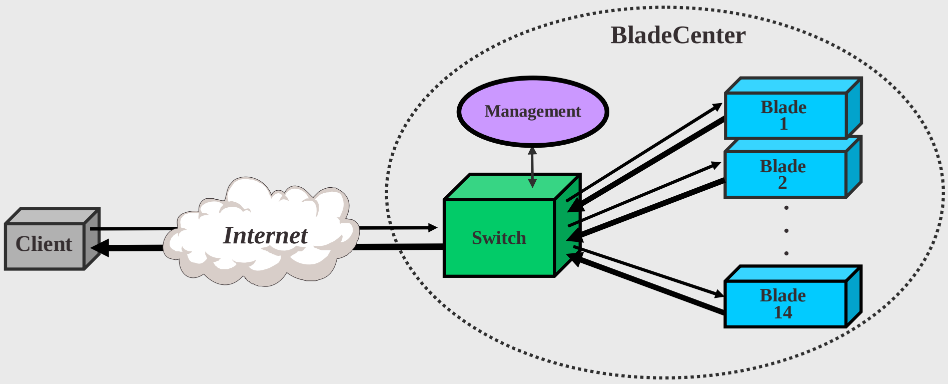
Calibrated, vectored cooling™
- **Fully fault tolerant**
- **Allow maximum processor speeds**

KVM Switches / Management Modules
- **Full remote video redirection**
- **Out-of-band / lights out systems management**

University of Puerto Rico – Mayagüez 9 IFIP Working Group 10.4 Winter Meeting 2005

Autonomic Web Computing with BladeCenter

- **Integrated switching enables autonomic functions through a common control point**
  - **Layer 2 switching provides basic standard functionality**
  - **Layer 4 (load balancing) and Layer 7 (content switching) for advanced web clustering**
  - **Layer 4/7 enables control point for directing traffic to up to fourteen blades**
  - **Web clusters are a popular method of workload management**
- **Examples of autonomic functions include performance, management, health, power, etc.**
  - **Automated workload management supports performance optimization and failover of blades**
  - **VM technology applied to blades to further improve granularity**
  - **Software health addressed with rejuvenation techniques**
  - **Power management can be addressed at multiple levels**

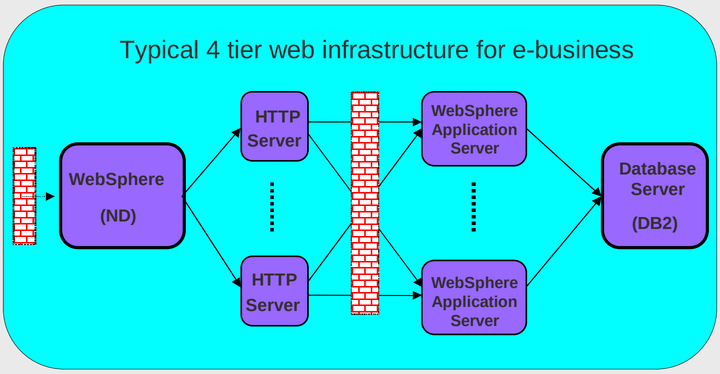


University of Puerto Rico – Mayagüez 10 IFIP Working Group 10.4 Winter Meeting 2005

Autonomic Web Computing with BladeCenter

Multi-Tier Infrastructure
- Front-End Load Balancer
- Web Servers
- Application Servers
- Data Base Server

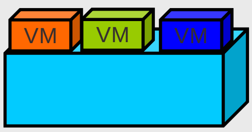
Infrastructure Automation
- Initially configures chassis & network and dynamically configures new and failover blades
- Automatically deploys and configures software stack (OS, middleware & apps) & network VLANs
- Monitors CPU load and predicts need for additional capacity (configures from free pool)



Solution Details
- Opus automatically provisions HTTP and WAS tiers
- IBM Tivoli Intelligent Orchestrator 1.1 (ITITO) policy-based analysis can determine when to schedule provisioning
- Opus utilizes IBM Director, Remote Deployment Manager for bare-metal install of Linux or Windows OS
- Opus workflows to install WebSphere Application Server/IBM HTTP Server/J2EE application, update Load balancer and HTTP Plug-in configuration files

Autonomic Web Computing with BladeCenter

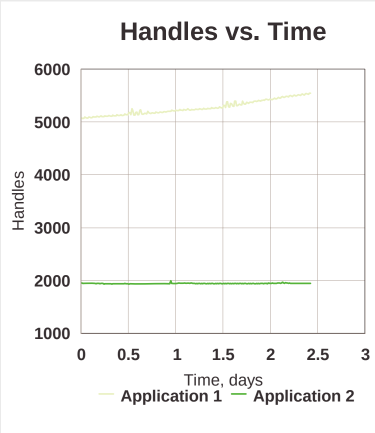
Virtual Machines
- VM technology such as VMware applied to blades for server consolidation
  - Orchestration and provisioning tools also apply to virtual machines
- VMware's VMotion technology enhances failover by transferring the entire system and memory state of a running virtual machine from one ESX Server to another
  - The Systems' disk, including all of its data, software and boot partitions, must be stored on a shared storage infrastructure such as a SAN
  - Keeps track of on-going memory transactions in a bitmap, which is kept small
  - When the memory and system state has been copied to the target server. VMotion:
    1. Suspends the source VM
    2. Copies the bitmap to the target ESX Server
    3. Resumes the VM on the target ESX Server
- The process takes less than 2 seconds (i.e., "hiccup time") on a Gigabit Ethernet network and appears as no more than a temporary network loss to the app, service and/or user.
  - It's necessary to keep this length of time minimal, since it leverages the operation of the TCP protocol for guaranteed delivery of lost packets.

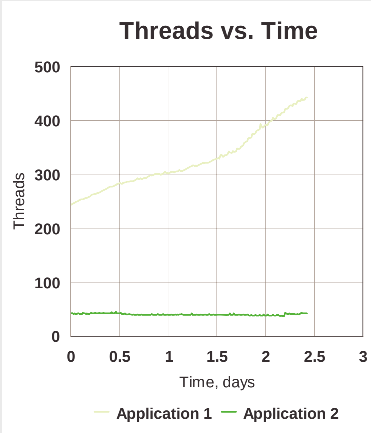


Autonomic Web Computing with BladeCenter

Software Rejuvenation

- System outages are far more likely to be a result of software errors than hardware failures
- Software (OS, middleware, applications, actually, state) ages with time...
  - memory leaks, handle leaks, nonterminated threads, unreleased file-locks, data corruption
  - ...resulting in Bad Things (outages, hangs, ...)
- Software failure prediction and state rejuvenation is a proactive technology designed to mitigate the effects of software aging



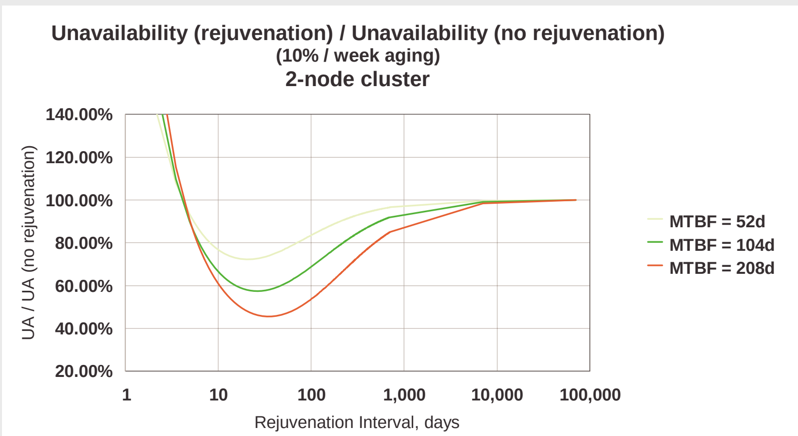


University of Puerto Rico – Mayagüez 13 IFIP Working Group 10.4 Winter Meeting 2005

Autonomic Web Computing with BladeCenter

Software Rejuvenation

- Develop proactive self-healing systems
  - Reduce probability of "Bad Things" due to software aging
  - Detect and predict resource exhaustion
  - Invoke timely corrective action via Software Rejuvenation
  - Resetting of software state to initial level of resource consumption
  - Apply technology to web clustering
  - More info: https://www.research.ibm.com/journal/rd/452/castelli.html

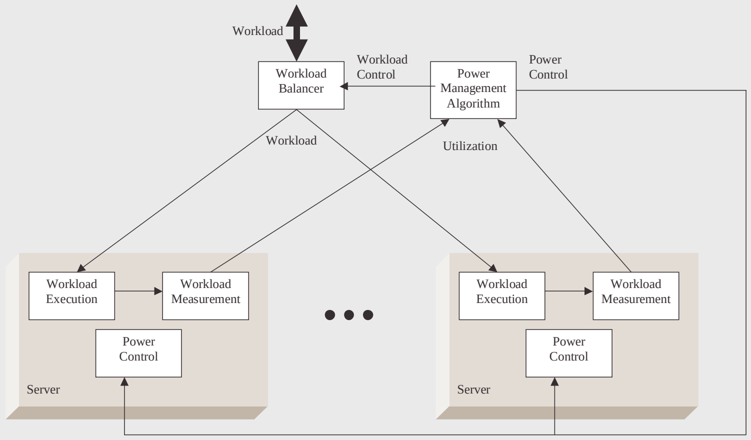


University of Puerto Rico – Mayagüez 14 IFIP Working Group 10.4 Winter Meeting 2005

Autonomic Web Computing with BladeCenter

Power Management

- Predictive algorithm that measures and predicts workload and determines when to place servers in a low power state
- Objective is to minimize energy consumption, unmet demand, and power cycles
  - Automatically adapts to short term and seasonal workload variations
  - Automatically adapts algorithm "gains" to workload dynamics
  - Energy savings of 20% or more can be achieved
  - More info: http://www.research.ibm.com/journal/rd/475/bradley.pdf



Questions?

HP BladeSystem
Reliable Web Services

January 2005

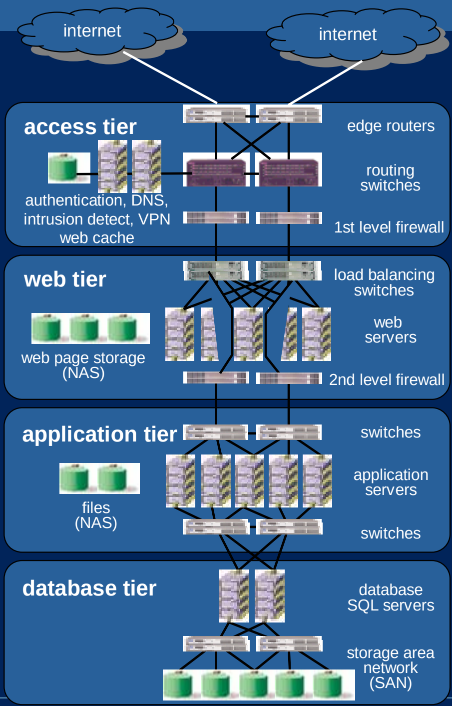
Dwight Barron
HP Fellow
Hardware CTO
Industry Standard Servers

Agenda

- Web Services Architecture
- Adaptive Enterprise Management Architecture
- Infrastructure Trends (aka Blades)
- Key Challenges

1/27/2005 HP Corporation page 2

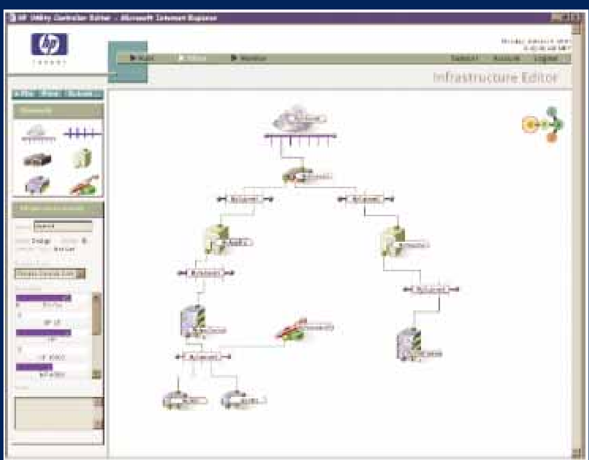
Web Services Architecture



- Established multi-tier architecture
- Increasing complexity of web transactions
- Static content wrapped in multiple layers of dynamic business content
- Tier boundaries blurring
- Web service reliability requires services at all tiers

1/27/2005 HP Corporation page 3

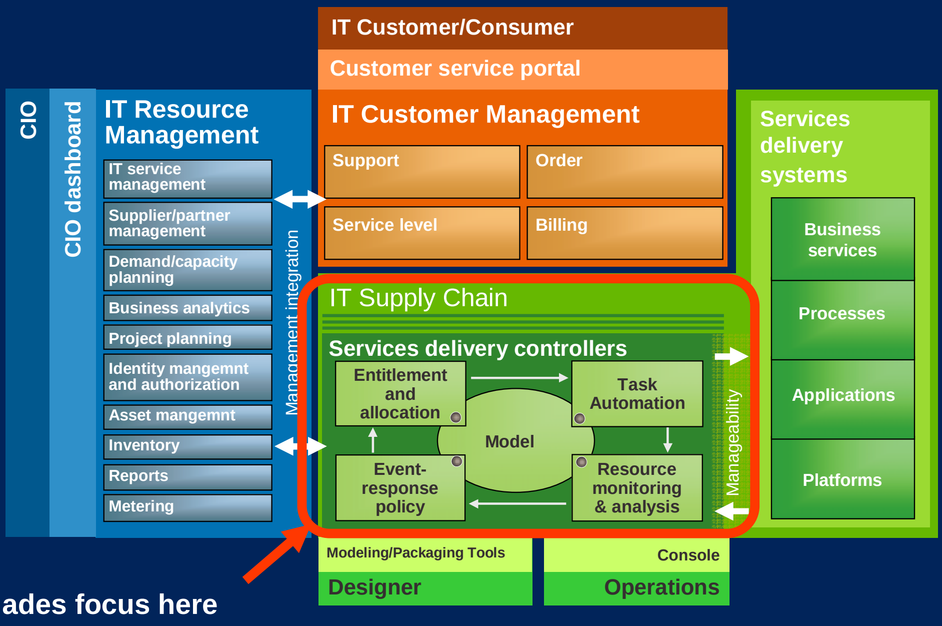
Web Services Model



- Web service elements have been successfully modeled
- Management tools to instantiate, isolate, monitor and dynamically repair web service instances
- Scales to large datacenter
- Most effective at scale of a large datacenter

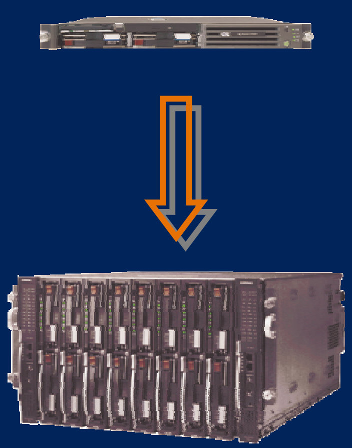
1/27/2005 HP Corporation page 4

IT service delivery business
conceptual architecture detail



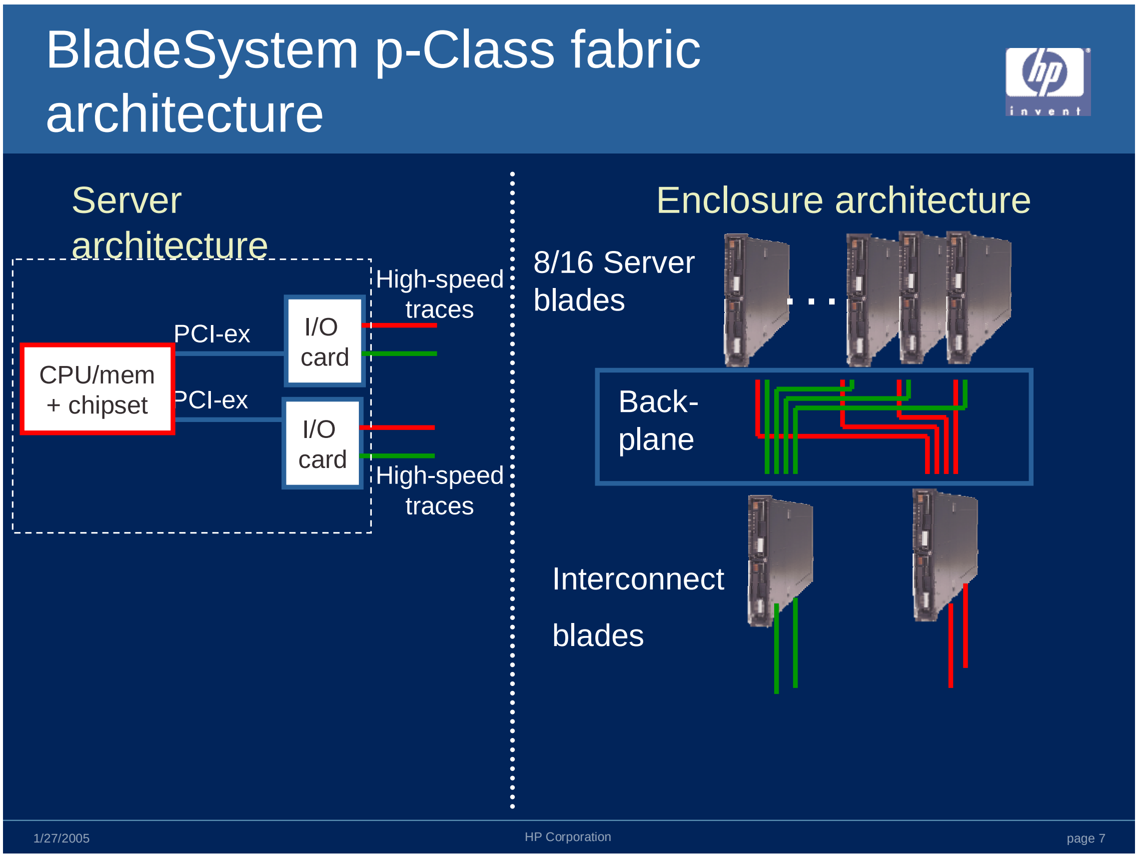
1/27/2005 HP Corporation page 5

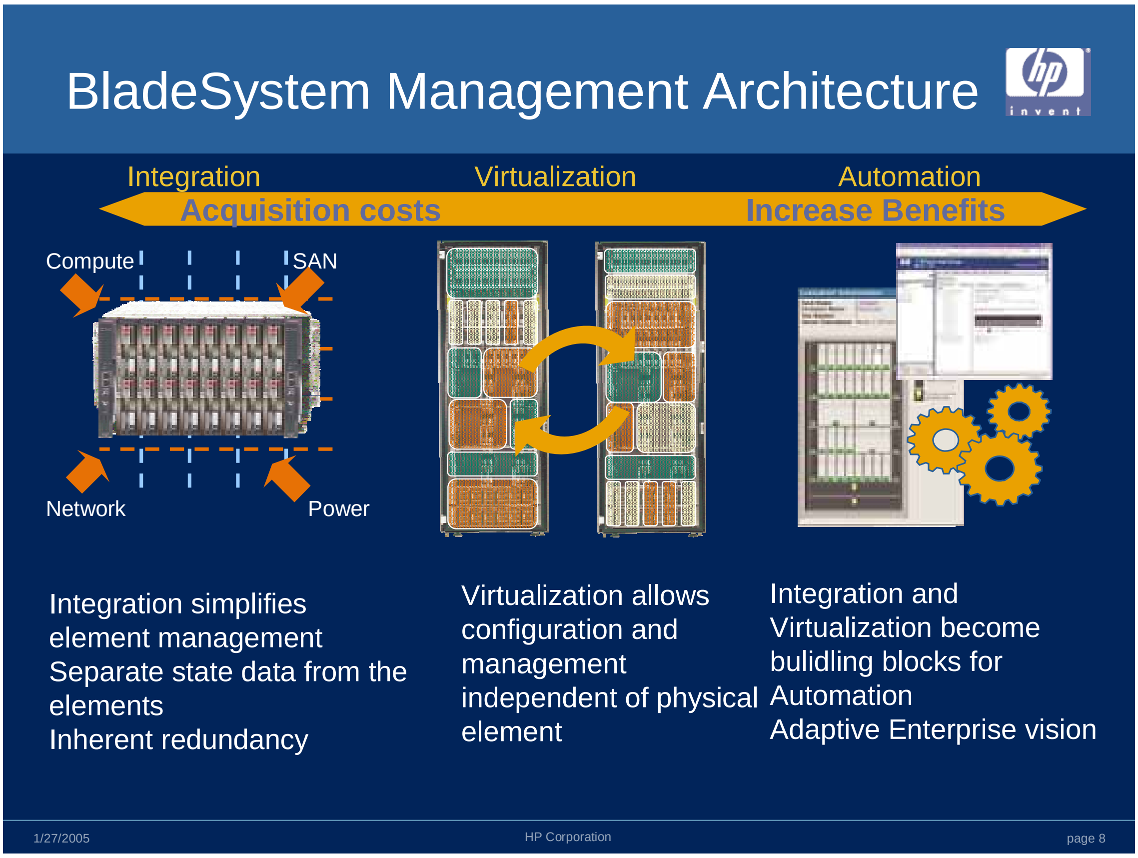
Blades Change Everything



- Complete web services infrastructure in a single chassis
- Even at datacenter scale, >90% of web services infrastructure is in the chassis
  - Servers
  - 1st tier networking
  - 1st tier SAN
  - Direct and network attached storage
  - Power distribution
- Fixed internal topology
- Complex problems become tractable

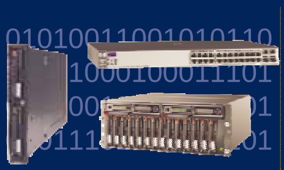
1/27/2005 HP Corporation page 6


BladeSystem p-Class fabric architecture
Server architecture
CPU/mem + chipset
PCI-ex
PCI-ex
I/O card
I/O card
High-speed traces
High-speed traces
Enclosure architecture
8/16 Server blades
Back-plane
Interconnect blades
1/27/2005
HP Corporation
page 7


BladeSystem Management Architecture
Integration
Virtualization
Automation
Acquisition costs
Increase Benefits
Compute
SAN
Network
Power
Integration simplifies element management
Separate state data from the elements
Inherent redundancy
Virtualization allows configuration and management independent of physical element
Integration and Virtualization become bulidling blocks for Automation
Adaptive Enterprise vision
1/27/2005
HP Corporation
page 8

HP BladeSystem Automation

End-to-end provisioning



Provision solutions across compute, network, and storage in minutes

Automated node recovery

Deliver economical high availability via resource pooling and auto-recovery

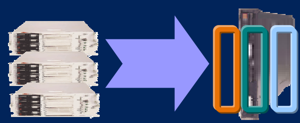
Scheduled re-provisioning

Improve system utilization through scheduled re-provisioning

Dynamic scaling

Dynamically scale infrastructure based on performance needs

Rapid IT consolidation



Consolidate legacy systems to latest performance platforms

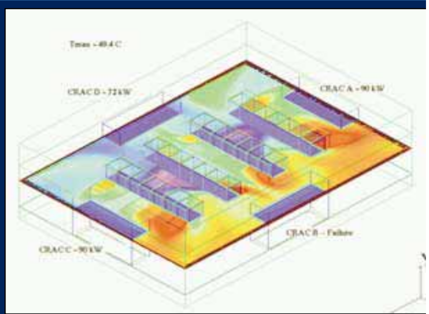
Patch and vulnerability

Quickly assess and respond to potential security vulnerabilities

1/27/2005 HP Corporation page 9

Key (Reliability) Challenges

- Datacenter design and layout
  - Power is fully redundant
  - What about the cooling?
- Interoperable services models
  - Standards work underway
- Storage management
  - SANs require end-end management
  - NAS model is rapidly evolving
- Security, security, security



1/27/2005 HP Corporation page 10

Thank You

Ideas for a Dependable ‘Industry Standard Architecture’ Platform

Newisys, Inc.

Rich Oehler

27 January 2005

Outline

- Our Company - Newisys
- Our Current Products – 2100 and 4300
  - Under-development - Horus
- Industry Standard Architecture Products
  - Attributes
    - Weaknesses
- Dependable Systems
  - Attributes
- Achieving Dependable System Structures
  - Scaling (both Up and Out)
  - I/O Connectivity and Configuration
  - Systems Management
- Performance Projections
- Summary

Newisys, Inc

- Founded in July 2000
  - Designing Enterprise Class, Rack Mounted, Opteron Based Server Systems for the OEM Market
- Entered into a Strategic Alliance with AMD for access to coherent HyperTransport
  - Began design of a custom ASIC (Horus) to enable large SMP (8 to 32 socket) Opteron Systems
- Acquired by Sanmina/SCI in July 2003
- Bringing up systems based on our custom ASIC
- Currently about 110 employees, ~ 90 Eng/PGM
  - Located in Austin TX

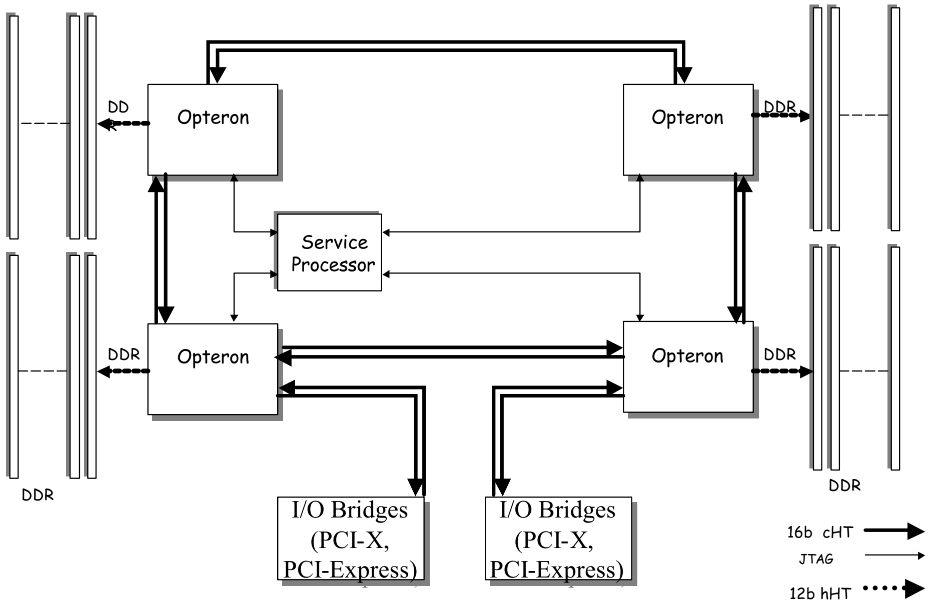
Why Opteron?

- AMD radically changed the system architecture of Industry Standard platforms
- Opteron has 3 point to point links (HyperTransport) on each chip
  - Each link can be used to connect to other Opterons (coherent) or to I/O (non-coherent)
- Opteron has a direct memory interface on each chip

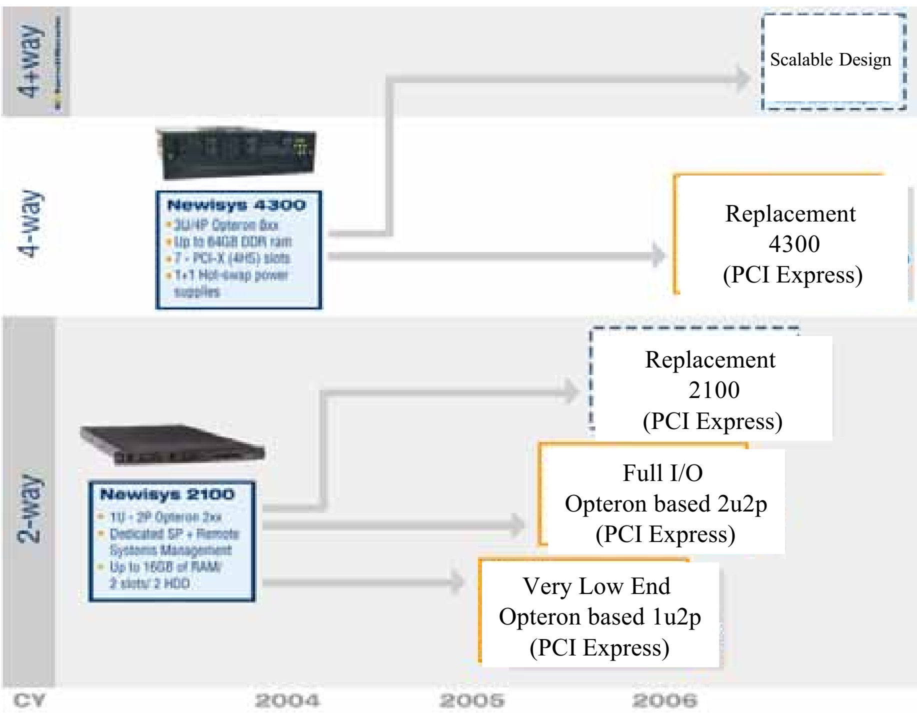
Results:

- Glueless SMP – up to 8 sockets
- Adding Opterons greatly improves scalability
  - More memory capacity and bandwidth
  - More coherency bandwidth
  - More I/O bandwidth

Typical 4 Socket (Quad) Opteron System



Newisys Product Roadmap



Limits of Scalability on Opteron

- Opteron provides for up to 8-socket 'glueless' SMP solution
- Opteron has very good Scaling to at least 4-socket
- Performance of important commercial applications is challenging above 4-socket due to:
  - Link interconnect topology (wiring and packaging)
  - Link loading with less than full interconnect (even less than 3 links)
- Going above 8-socket needs both:
  - Fix to number of addressable sockets
  - Better interconnect topology
- Ever larger Coherency Fabric will increase delays (loading/queuing) and become the major obstacle to good SMP scaling

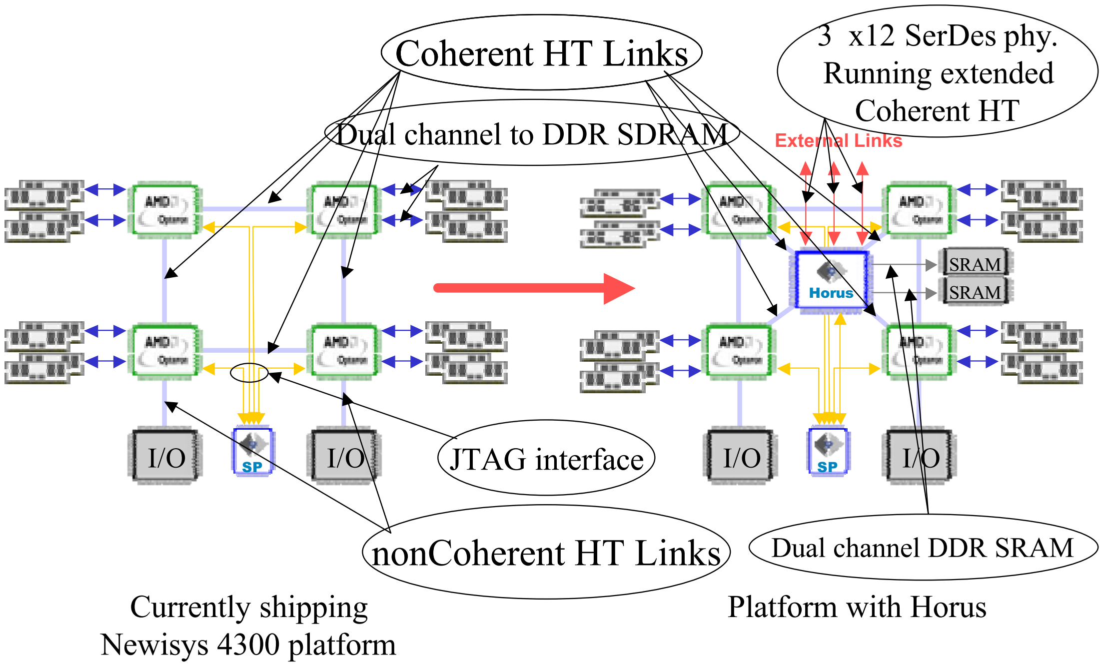
Solving the Fundamental Problem

- Combine multiple four socket quads into a larger coherent domain…
- But local quads have no knowledge of "remote quads" (CPUs, I/O or Memory) outside of the their own local space
- So our approach is to add into each quad a "fifth" socket that abstracts all of the remote quads
  - Acts as a "cache" for local request probing
  - Acts as a "memory controller" for requests to remote memory space and from remote CPUs
  - Acts as a "CPU" for requests from remote nodes
- And to place in all of the Opteron sockets an abstraction of all of the remote resources

Horus – Newisys Custom ASIC

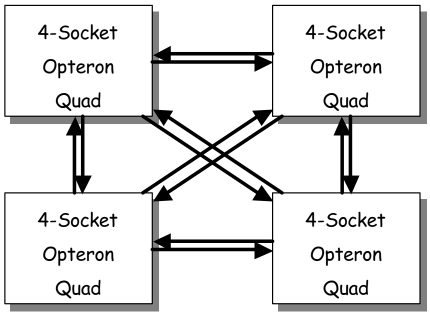
- Defines a coherence mechanism to support two or more 4-socket AMD Opteron quads
  - Built into our standard 4 socket rack building block
  - Industry Standard Servers (Industry Standard Pricing)
- Acts as a Distributed Router in the coherency domain
  - Multiple Horus are connected by an extension of coherent HyperTransport
  - Direct connect (cut through) to non adjacent quads
- Adds facilities to reduce coherency traffic
  - Remote Directory, Remote Data Cache
- Provides a management point and performance optimization point
  - Partitioning between/among quads

Scalable Newisys Opteron Systems



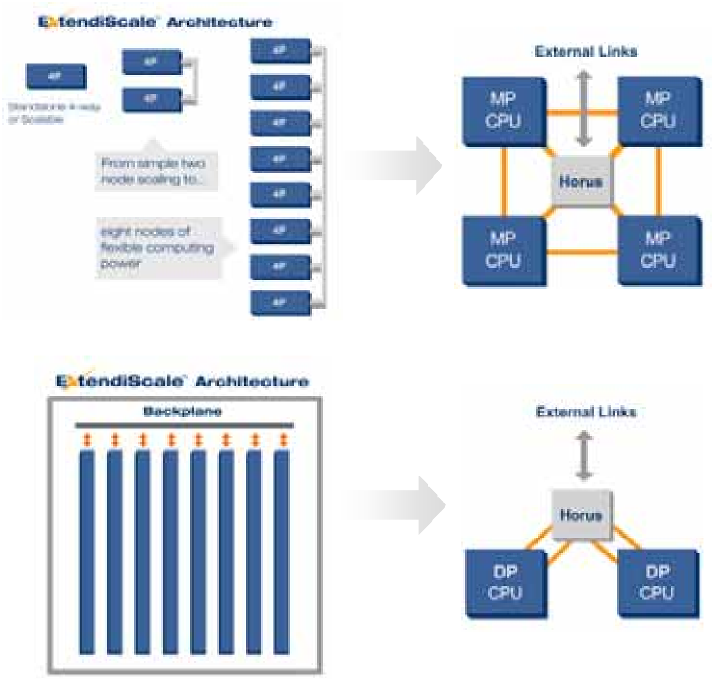
Building Larger Configurations

Typical 16-way



Up to 32 Sockets possible

Newisys ExtendiScale Architecture



- Exceptional performance headroom
- Enables modular systems
  - Traditional 8-64 way CC SMP (Dual core)
  - Blade frame 2-32 way CC SMP (Dual core)
- The ExtendiScale Architecture delivers:
  - Pay as you grow budget flexibility
  - RISC/UNIX replacement at a fraction of the cost
  - Mission Critical ready: Availability, Manageability, Reliability

What makes hardware dependable?

- Hardware that never fails; or if it does, self heals; has no loss of data or incorrect results; or if it does, contains and identifies the error; adjusts to workloads without bogging down; or if it does, can apply additional or spare resources; …
  - Typically (Very?) expensive
  - Certainly custom design
- Are there different design points for dependability? Can Industry Standard Servers be made dependable enough?
  - Certainly lower cost
  - How much dependability is required / sufficient?
    - Software can make up for many hardware deficiencies
      - At what cost? Performance?

Acceptability of Industry Standard Servers

- Industry Standard Servers suffer from
  - silent failures, catastrophic failures, lock up failures
- Newisys is building enterprise class servers out of Industry Standard parts.
  - Our hardware systems are much more reliable than those produced by Taiwan Inc. (better engineering)
  - Our incremental cost is marginal
- Our System Management with an out of band Service Processor fixes even more problems not solved in current Industry Standard parts

Focus on Newisys Opteron Blades

Disclaimer – not currently on our road map

- Built around 2 socket CPU Blades and I/O Blades
- Coherency Fabric connects all CPU Blades
  - Used to configure larger than 2 socket SMP systems
  - Each CPU Blades also develop at least 2 connections to an I/O Fabric based on PCI-Express
- I/O Fabric connects all I/O Blades with connections to each CPU Blade
  - I/O Fabric contains a switch (two for redundancy)
    - Based on Advanced Switching or more specialized solutions
  - I/O Blades can be dedicated or shared

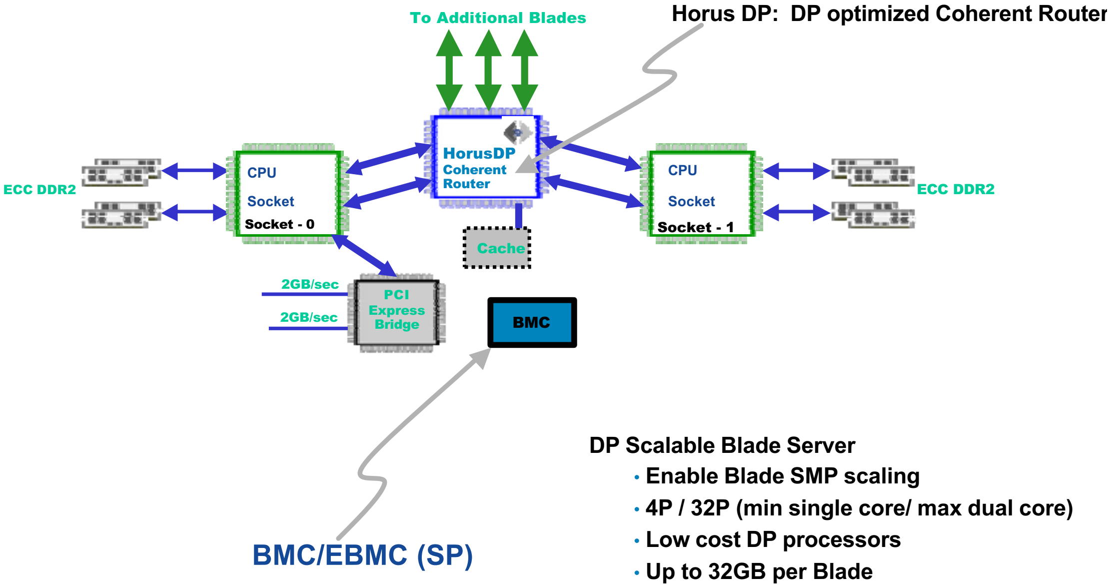
Why Blades?

- Blades are not about power packaging and cooling (although these problems are hard and getting harder and must be solved)
- Blades are not scaled down systems
  - Large and Powerful systems can be built as Blades
- Blades are about defining a uniform set of structures over which many problems are solved in a systematic way
  - Provisioning
  - Configuration (including partitioning)
  - Recovery (including hot swap, fail over, …)
  - Maintenance and Repair
  - Alignment of hardware boundaries with application boundaries
  - …

Why Scale Up?

- For many web applications scale out is the best answer
  - Especially near the edge of the net (tier 1 and 2)
- But for many tier 3 applications, the answer is not obvious
  - Lots of existing large monolithic databases and their associated applications
  - Some problems/applications just don't partition well
    - Pieces are too small, synchronization cost too high
- Newisys Blades can do both scale up and scale out
  - Can be configured/controlled to go from scale out to scale up and back as needed by policy, workload, …

Scalable DP Blade

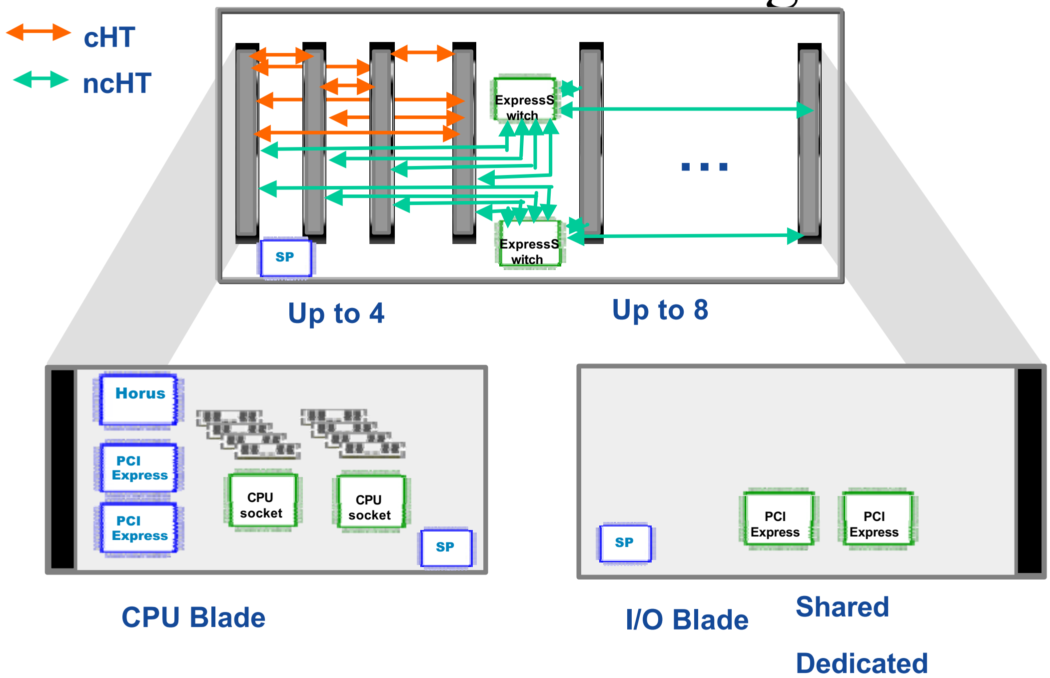


DP Scalable Blade Server
- Enable Blade SMP scaling
- 4P / 32P (min single core/ max dual core)
- Low cost DP processors
- Up to 32GB per Blade

PCI Express Attributes

- Aggregated very high speed I/O lanes
  - Each lane can be 2Gb/second (today)
  - 16, 24, 32 lanes can be bundled together
- ‘Advanced Switching’ Technology exists today
  - Defined to map up to and down from PCI Express
- Several Startups working on direct PCI Express switching
- Controllers / adapters can be
  - Dedicated (1 to 1) with a system
    - Examples: today’s storage, network controllers (HBA)
  - Shared (1 to n) with multiple systems
    - Examples: shared 10Gb Ethernet adapter, shared FC adapter

Blade Mid-Plane Diagram



Virtualization and Hardware Partitioning

- Virtualization (creating many virtual machines / environments) works really well
- When is it not better to virtualize on a really big system
  - Depends on structure of the really big system
  - If virtualized resources don't correspond to equivalent hardware resources, performance issues may result
    - Many of today's OSs can not match physical resources with virtual resources
  - Again, if no correspondence, hardware failure boundaries may impact many virtual environments (sometimes significantly more)
- Matching real system resources with program resource needs leads to
  - Better performance with dedicated resources
  - More robust execution when errors occur

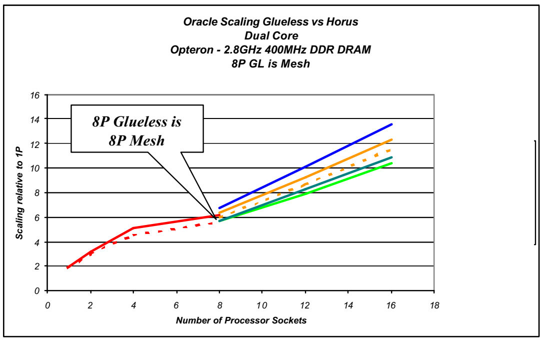
Role of System Management

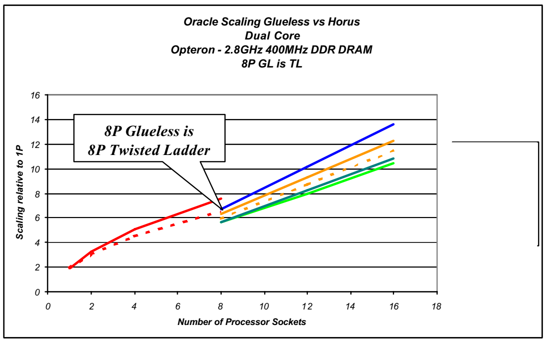
- Separate, out of band management required
- At Several Levels
  - CPU card and I/O card
    - Used for standard environmental controls
    - Also acts as a surrogate during provisioning, configuration and initialization, error detection and recovery
    - Can provide local performance monitoring and local power management
  - At Switch (coherent and non-coherent)
    - Configuration control and performance monitoring
  - At Frame/Rack
    - Overall complex view

Newisys Systems Management

- Horus provides building blocks not a complete solution for a single SMP system
- We use an onboard but independent Service Processor and special interconnect hooks to provide the rest
- There are at least two Service Processors and their system management code, one primary and one fall back in each complex system.
- The system management code deals with configuration control, including partitioning, various RAS issues including watch dog timers and managing the various hardware hooks for Power On/Off, Reset, Hard and Soft IPL, HT Stopping and Restarting, etc.

Scaling – Dual Core





Summary

- Newisys is building robust Industry Standard Servers as well as a Scalability ASIC
- Blades can be built out of Newisys parts that offer
  - SMP scaling through Horus
  - I/O scaling through PCI Express switching
- Newisys Systems Management offers a level of RAS in Industry Standard Serves previously only achievable in RISC/Unix servers
- Dependable Systems can be built out of Newisys building blocks

47th Meeting of IFIP WG10.4
Puerto Rico – Jan 2005

Non-intrusive Middleware for Continuity of Service: Protection Against System Failures

Marc Rougier - Meiosys

About Meiosys

- Independent Software Vendor, founded in 2000
- 35 people, 25 engineers in Toulouse, France and Palo Alto, CA, USA
- Genes are in middleware for distributed, life-critical systems
- Develops linux and Unix-based middleware to increase flexibility and dependability of commodity platforms
- Main topic of R&D today is Record and Replay technology for Fault Tolerance

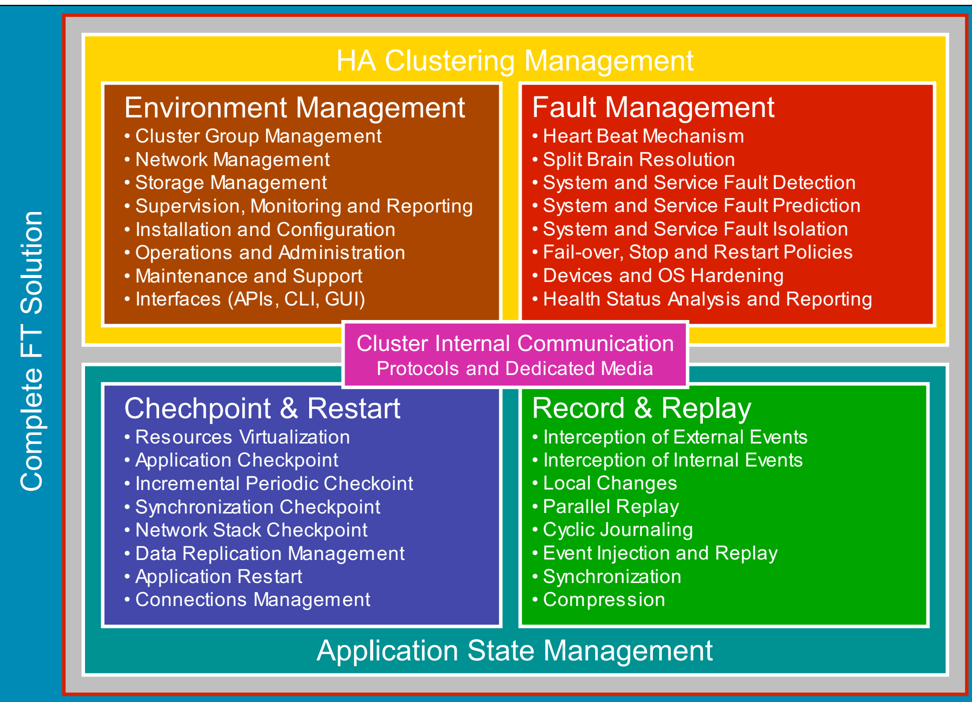
2 Meiosys Confidential

Meiosys FT R&D Objectives

- Mission is to increase the service uptime (at an acceptable cost)
- Focus is to protect against system failures
  - Solution provides a dependable infrastructure…
  - But does not solve all problems (software bugs, human errors, etc)
- Approach is based on
  - Hardware redundancy and
  - Dedicated middleware maintaining operational and back-up systems in-sync
  - Active-Passive and Active-Active mode
- Main challenges
  - Application-transparent: no modification, re-compile nor re-link of the application
  - Runs on commodity equipment (off-the-shelf servers)
  - Performances impact needs to be “acceptable”
  - Needs to be applicable to commercial ISVs applications (DBMS, AS, ERP, etc), new applications (J2EE) and legacy applications
- Main problem: *the non deterministic nature of linux / Unix*

3 Meiosys Confidential

FT Solution: High Level Components



4 Meiosys Confidential

Meiosys FT R&D: Current Status

Complete FT Solution

HA Clustering Management

Environment Management
- Cluster Group Management
- Network Management
- Storage Management
- Supervision, Monitoring and Reporting
- Installation and Configuration
- Operations and Administration
- Maintenance and Support
- Interfaces (APIs, CLI, GUI)

Fault Management
- Heart Beat Mechanism
- Split Brain Resolution
- System and Service Fault Detection
- System and Service Fault Prediction
- System and Service Fault Isolation
- Fail-over, Stop and Restart Policies
- Devices and OS Hardening
- Health Status Analysis and Reporting

Cluster Internal Communication
Protocols and Dedicated Media

Chechpoint & Restart
- Resources Virtualization
- Application Checkpoint
- Incremental Periodic Checkoint
- Synchronization Checkpoint
- Network Stack Checkpoint
- Data Replication Management
- Application Restart
- Connections Management

Record & Replay
- Interception of External Events
- Interception of Internal Events
- Local Changes
- Parallel Replay
- Cyclic Journaling
- Event Injection and Replay
- Synchronization
- Compression

Application State Management

5 Meiosys Confidential

Meiosys FT R&D: Current Focus

Complete FT Solution

HA Clustering Management

Environment Management
- Cluster Group Management
- Network Management
- Storage Management
- Supervision, Monitoring and Reporting
- Installation and Configuration
- Operations and Administration
- Maintenance and Support
- Interfaces (APIs, CLI, GUI)

Fault Management
- Heart Beat Mechanism
- Split Brain Resolution
- System and Service Fault Detection
- System and Service Fault Prediction
- System and Service Fault Isolation
- Fail-over, Stop and Restart Policies
- Devices and OS Hardening
- Health Status Analysis and Reporting

Cluster Internal Communication
Protocols and Dedicated Media

Chechpoint & Restart
- Resources Virtualization
- Application Checkpoint
- Incremental Periodic Checkoint
- Synchronization Checkpoint
- Network Stack Checkpoint
- Data Replication Management
- Application Restart
- Connections Management

Record & Replay
- Interception of External Events
- Interception of Internal Events
- Local Changes
- Parallel Replay
- Cyclic Journaling
- Event Injection and Replay
- Synchronization
- Compression

Application State Management

6 Meiosys Confidential

Technology Modules

MEIOSYS

Interfaces	Enables integration with third-party components in the data center (e.g., Tivoli, OpenView, Unicenter, HA clustering solution, billing systems, etc)
Policy Engine	Takes actions according to reported data; actions can be driven by optimization of resources (TCO), performances or uptime (SLA)
Monitoring	Captures and reports status data related to the behavior and health of the system and of the applications
Relocation	Orchestrates the mobility of the state files (mediation with the management layer, check of nodes consistency, N-stage migration with hand-checks)
R&R	Records and Replay all events which modify the application state (external messages and internal non deterministic events)
Checkpoint	Captures in a « state file » all states constituting the run-time of the applications (memory, IPCs, kernel states including TCP stack, etc)
Virtualization	Maintains a near real-time view of the application structure in a « container » and subsitutes local IDs by relocatable IDs (e.g. PID, Sys V IPC IDs, etc)
Instrumentation	Techniques to interact with applications at run time (e.g. interposition agents, kernel APIs, syscall injection)

7 Meiosys Confidential

Technology Modules

MEIOSYS

Interfaces

Policy Engine

Monitoring

Relocation

Checkpoint

Virtualization

Instrumentation

- Completed (and shipping)
- Enables dynamic, on-demand workload placement
- Maintains full states and network connections
- Thin virtualization layer (<1% runtime overhead)
- Granularity = application-level
- Stateful Application Relocation can be triggered by:
  - Resource optimization policies (consolidation)
  - Performance optimization policies (scale up)
  - High Availability policies (predictive fail-over)

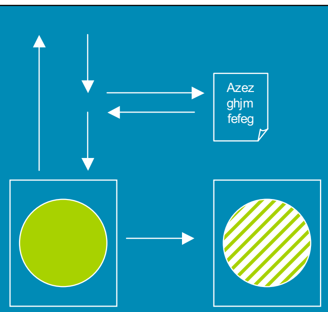
8 Meiosys Confidential

Technology Modules

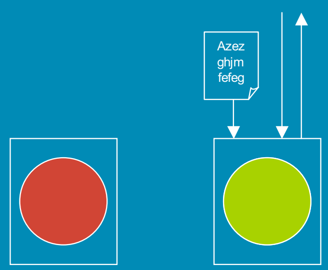
- Interfaces
- Policy Engine
- Monitoring
- Relocation
- R&R
- Checkpoint
- Virtualization
- Instrumentation

Main focus on R&D

Active-Passive Mode: Enables N+K

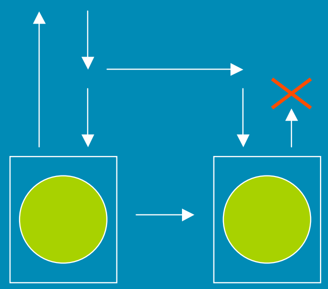


- A passive clone is maintained nearly up-to-date on the back-up node through incremental periodic checkpoint (in sync with the journal)
- Events are logged synchronously on the back-up node in a revolving journal
- Clone is stateful and includes TCP connections

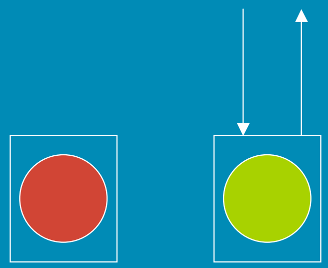


- Upon detection of outage of main node, journal is replayed on backup, so as to bring the clone up-to-date, in sync with external world (no messages are output during replay)
- Then communications are re-established with external world via migrated connections
- During replay, incoming messages are « on hold » (TCP flow control property)

Active-Active Mode: Faster Switch-Over



- Initial synchronization is achieved through a checkpoint
- Events are forwarded optimistically to the back-up node, on the fly
- Events are processed on both nodes but only the master sends messages to external world



- Upon detection of outage of main node, current log is flushed and IOs are switched from shadow mode to operational mode
- Backup node immediatly resumes operations (sub-second switch-over)
- Order to switch-over can come from an external system (not necessarily a fault)

11 Meiosys Confidential

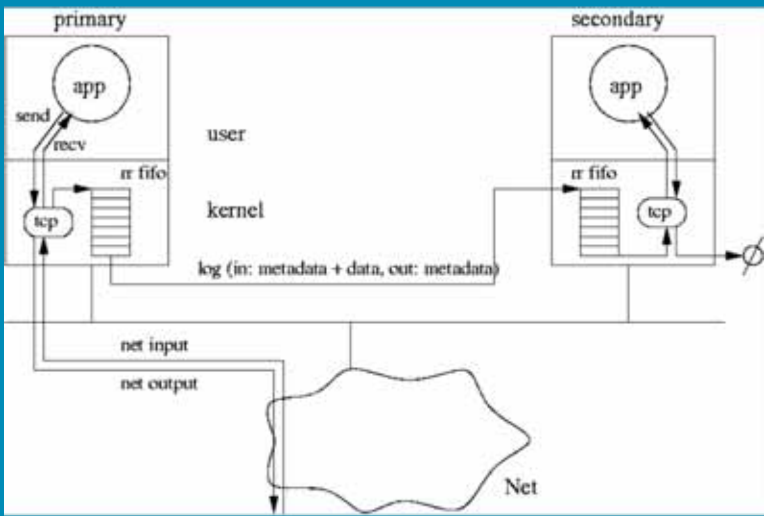
The Challenge of R&R: Non Determinism

- A State can be modified by external and internal events
- External Non Determinist Events (ENDE):
  - Inputs from network (TCP), or shared storage
  - Medium frequency (up to 10 Khz), medium volume (1-10 KB / event)
- Internal Non Determinist Events (INDE):
  - Non-determinist conditions due to OS or HW concurrency:
    - SHM access ordering , FS access order, IPCs, signals, I/Os
  - Random conditions:
    - Date (timestamps), timers, random numbers
  - High frequency (up to 10 Mhz), low volume (~ 10 B / event)
  - Internal NDEs between last external NDE and crash time can be lost
- The challenge is to Record and Replay these events deterministically, to maintain service integrity

12 Meiosys Confidential

R&R of External Events: TCP

- Both nodes have the same virtual IP address. Only primary is visible.
- On primary: network input data, and connection metadata are logged on the fly to secondary.
- On secondary: network output disabled. Shadow sockets are feed and maintained up-to-date from the log and active application replica.

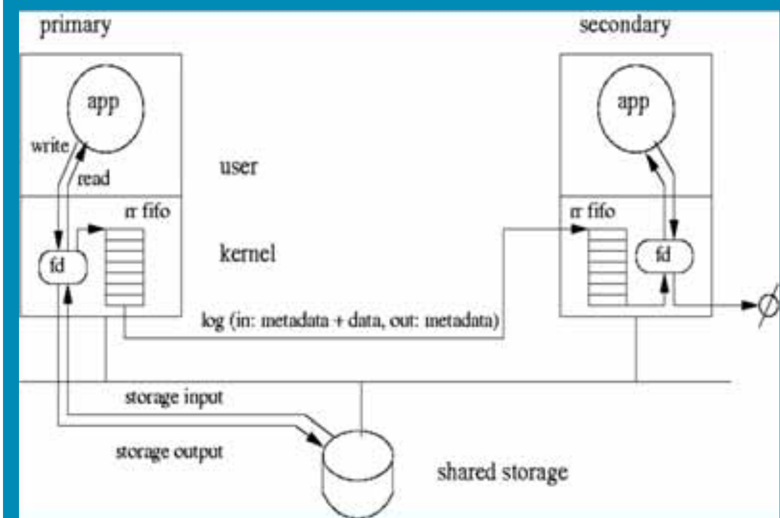


- Switch-over: at end of log, secondary takes over network physical access. Shadow sockets are ready to take over.
- Stand-by reinsertion: TCP sockets are checkpointed and cloned as part of process resources.
- No loss of in-flight messages: ACK'ed by primary only after logging. If crash during logging, retransmit by TCP.

13 Meiosys Confidential

R&R of External Events: Shared Storage

- Only the primary node has physical access to the shared storage
- On primary: inputs and system calls metadata are logged to secondary on the fly
- On secondary: output to storage is disabled
- Storage metadata (shadow file descriptors) are updated on the fly by active application replica and log

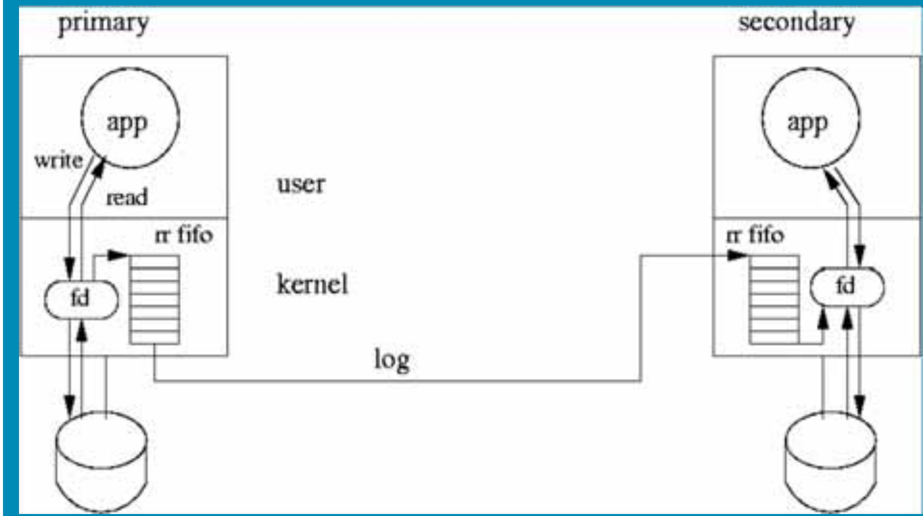


- At switch-over: secondary enables access to storage (procedure depends on type of storage)
- Shadow file descriptors mapped on real storage
- Reinsertion of standby: nothing to be done

14 Meiosys Confidential

R&R of External Events: Unshared Storage

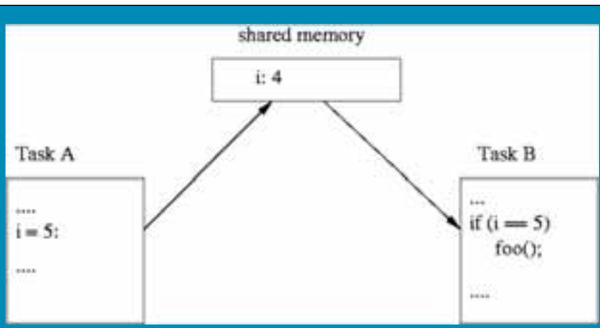
- Storage considered as a local resource
- Only storage access system calls metadata are logged
- At switch-over: the storage is already operational
- Reinsertion of stand-by: requires filesystem snapshot and replication capabilities



R&R of Internal Events: "Easy" Cases

- I/O-related System calls (non deterministic size)
  - Record and Replay the behavior (number of bytes)
  - Or change behavior locally ("semantic change") if more efficient (force number of bytes, hence reducing amount of data to be logged)
- I/O Multiplexing (non deterministic ordering)
  - Record and Replay the behavior (ordering)
  - Or change behavior locally ("semantic change") if more efficient (force ordering, hence reducing amount of data to be logged)
- Date, Timestamps, Random numbers
  - Must be Recorded and Replayed

R&R of Internal Events: “Difficult” Cases



- SHM access:
  - Task A and B running on 2 parallel CPUs in SMP
  - Execution result depends on the ordering of SHM access by A and B
  - Race condition is arbitrated at physical level (CPU-MEM bus controller), beyond the reach of kernel
  - If ordering could be detected, logging each access would multiply unitary cost by 1000 (ref. works by Bacon & Goldstein – Berkeley and IBM Watson, on snooping the CPU-memory bus with specific hardware technology)
- Signal delivery
  - Task A sends a signal to task B
  - Crash occurs after task B receives the signal on Operational node but before task B receives the signal on back-up node
  - Task B needs to receive signal at the same instruction on back-up node

R&R of Internal Events: “Difficult” Cases. Approach: Repeatable Scheduling

- Repeatable Scheduling
  - Definition: ability to reproduce task interleaving at instruction level
  - If a task receives the same interrupts at the same execution points, it will reproduce the same outputs
  - Addresses R&R of several INDEs: signals, SHM, IPCs
  - Transparent to applications (kernel-level solution)
  - BUT:
    1. **It assumes that instruction counters are reliable… which is (generally) false**
    2. **It is not applicable to SMP: does not address hardware parallelism**
- Repeatable Scheduling on SMP architectures with reliable counters
  - Modify resource access control to implement exclusive access during scheduling slice
  - Each CPU logs its scheduling activity
  - Shared resource access log used for global ordering
  - Requires two new algorithms:
    - **Reliable Instructions Counter**
    - **Exclusive SHM Access**

Reliable Instruction Counters

MEIOSYS

- Implement reliable instruction counting mechanism to complement repeatable scheduling on SMP architectures:
  - Hardware counters are available on modern CPUs, with negligible overhead
  - BUT not accurate: count of instructions impacted by pipelining, HW interrupts and exceptions, latency of overflow interrupts, micro-architecture optimizations
  - Forcing the CPU to produce precise instructions count makes it 25 times slower
  - Our approach: an additional software layer brings accuracy at instruction granularity level, compensating hardware inaccuracy
    - Software layer uses breakpoints to stop tasks at the exact location at Replay. Implements a reliable light weight CPU state checksum to handle closed loops
    - Scheduler's routines managing context switch have been extended
    - Record includes capture of signal delivery position
    - Enables to record on N CPUs and replay on M, whatever N and M ("logical CPU")

19 Meiosys Confidential

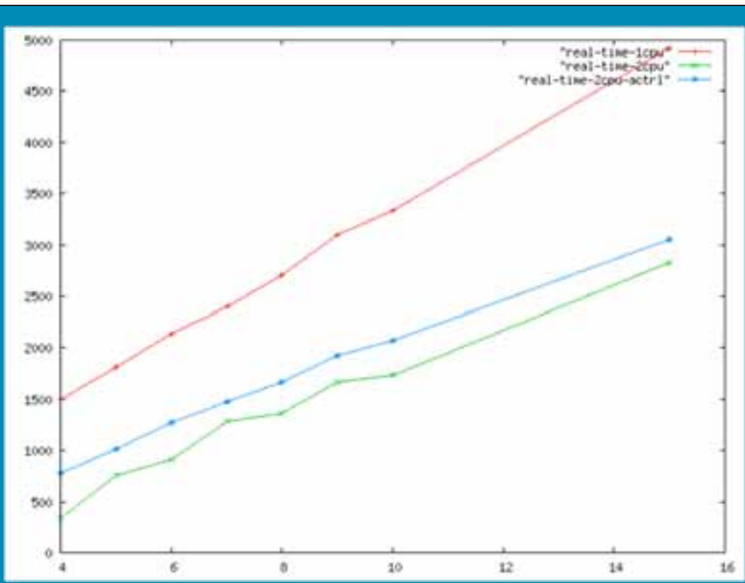
Exclusive SHM Access Control

MEIOSYS

- Implement exclusion mechanism to complement repeatable scheduling on SMP architectures:
  - Provides elected task with exclusive access to each shared memory page, for its scheduling period
  - Access control implemented by extending memory protection and paging mechanisms of MMU at kernel level
  - Allows to block a task if it accesses "in-use" SHM, freeing the slot for other work
  - Remove race conditions at user level
  - Allows reproducible SHM access at very low performance cost in SMP

20 Meiosys Confidential

Exclusive SHM Access Control and Reliable Instruction Counters: Performance Overhead



Performance hit less than 10%, scales gracefully with number of processes

21 Meiosys Confidential

Current Status and Next Steps

- Current Status:
  - On-demand stateful application relocation :
    - Works with transactional apps (Oracle, Weblogic) under heavy load
    - Contributes to increasing uptime thanks to predictive stateful fail-over triggered by fault management systems (system-level and application-level)
  - Active-Passive and Active-Active frameworks, with R&R of TCP and basic logging and fault detection mechanisms; sub-second switch-over
  - Reliable Instructions Counter algorithm
  - Exclusive SHM Access Control algorithm
- Next Steps:
  - Integration of all NDEs into Active-Active framework
  - Integration of a high performance logging infrastructure
    - Low latency interconnect and dedicated protocol
    - Optimization (cached logging “TCP-out committed ”, null logging, etc)
  - Full scale performance benchmarks

22 Meiosys Confidential

MEIOSYS
Optimize and Protect

Thanks you for your attention

www.meiosys.com

Session 2

Autonomic Response to Faults and Attacks

Moderator and Rapporteur

William H. Sanders, UIUC, USA

Autonomic Computing

Autonomic Computing: an overview
January 2005

Nick Bowen
CTO IBM Systems Group Software

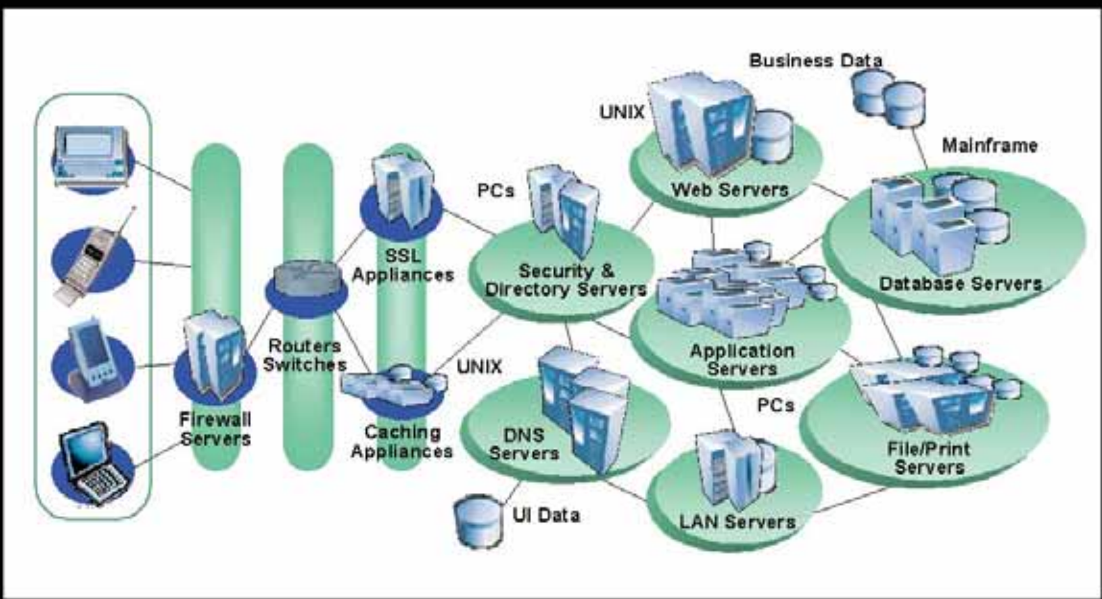
ON DEMAND BUSINESS

Autonomic Computing

31/01/05

Autonomic Computing

Today's Complex Infrastructure



2 Autonomic Computing 31/01/05 ON DEMAND BUSINESS

Autonomic Computing

The Beginnings – "Project eLiza"

Autonomic Computing is the embodiment of the principles and features that IBM designers have been building into our Systems for years.

- **Self-Configure**
  - ➢Hot Swappable Disks, PCI
  - ➢Wireless System Configuration - SNAP
  - ➢Auto discovery and update of firmware
- **Self-Heal**
  - ➢Virtual IP Takeover
  - ➢LightPath Diagnostics
  - ➢Chipkill ECC Memory, Dynamic bit steering
  - ➢Automatic Deallocation
  - ➢Call Home
  - ➢Virtual Help Desk
- **Self-Optimize**
  - ➢Clustering
  - ➢Dynamic LPAR
  - ➢Workload Management
  - ➢Quality of Service
  - ➢e-Business Mgt Service
- **Self-Protect**
  - ➢Self-protecting kernel
  - ➢Digital Certificates
  - ➢Enhanced encryption
  - ➢LDAP enhancements
  - ➢Security & Privacy Service

Now – A coordinated, systematic approach
The Future – consistent, world-class systems
- instrumented for enterprise level AC

3 Autonomic Computing 31/01/05 ON DEMAND BUSINESS

Autonomic Computing

Autonomic Computing
Focus on business, not infrastructure

Intelligent open systems that:

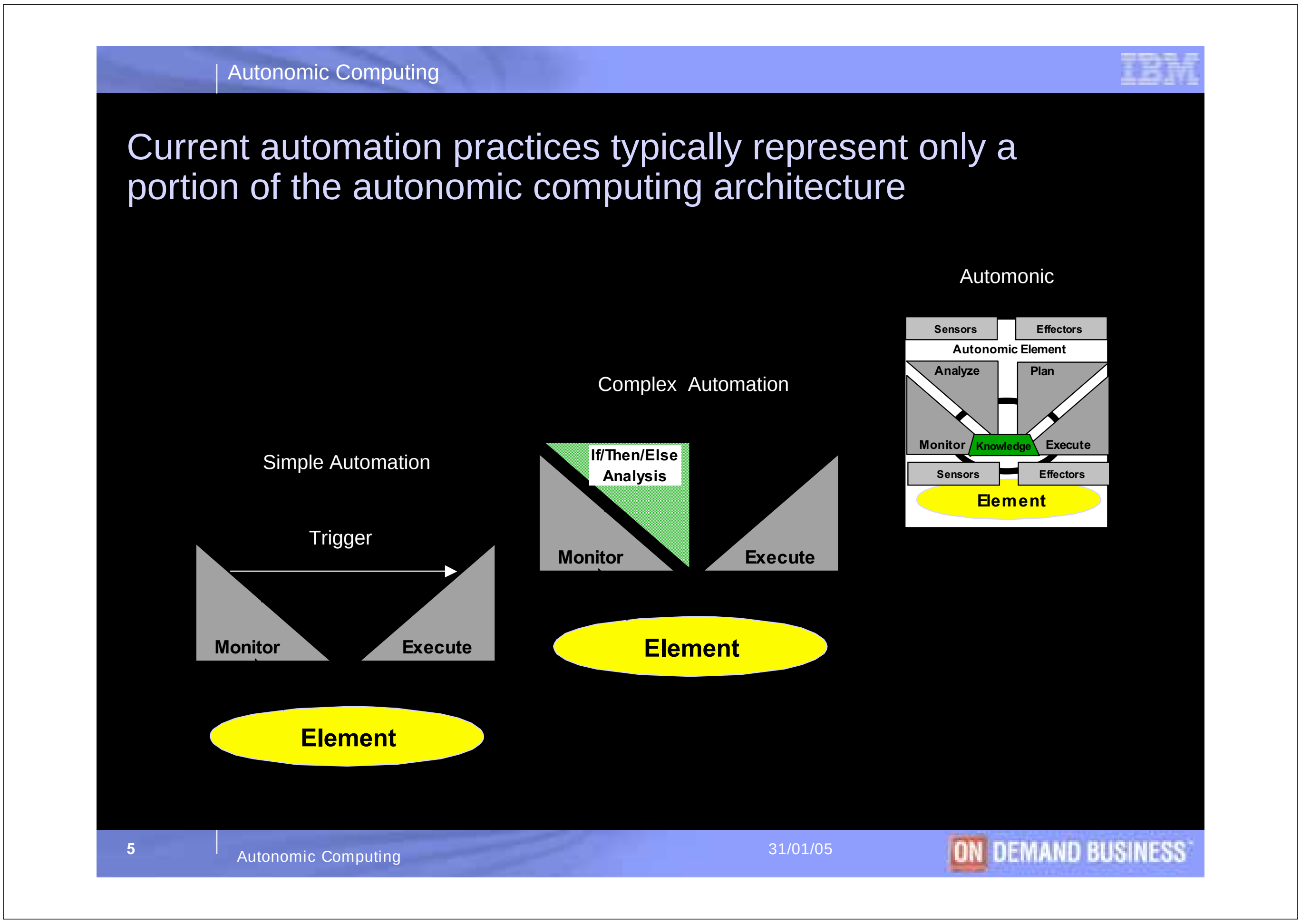
- **Adapt to unpredictable conditions**
- **Continuously tune themselves**
- **Prevent and recover from failures**
- **Provide a safe environment**

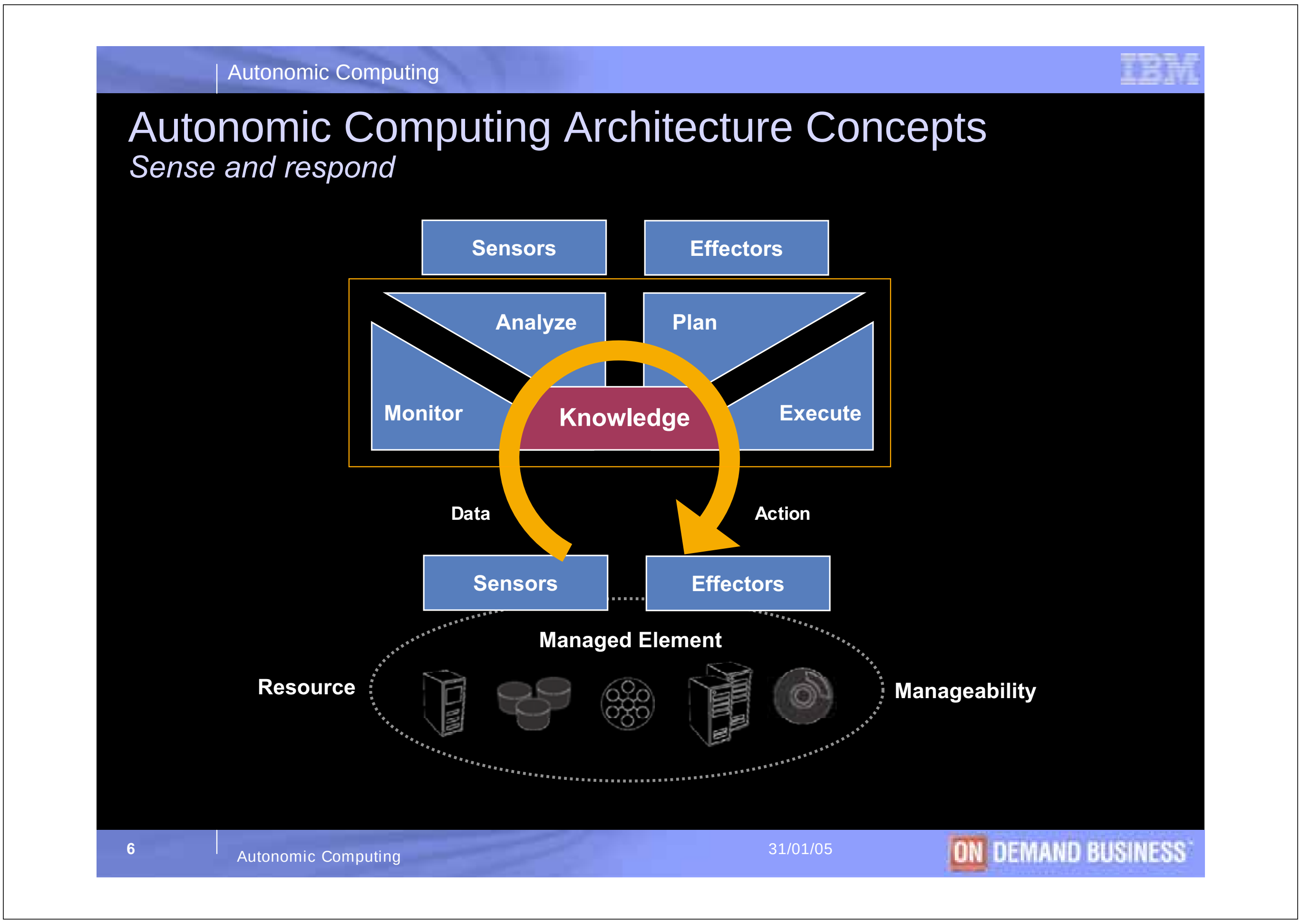
Providing customer value
- **Increased return on IT investment**
- **Improved resiliency and quality of service**
- **Accelerated time to value**

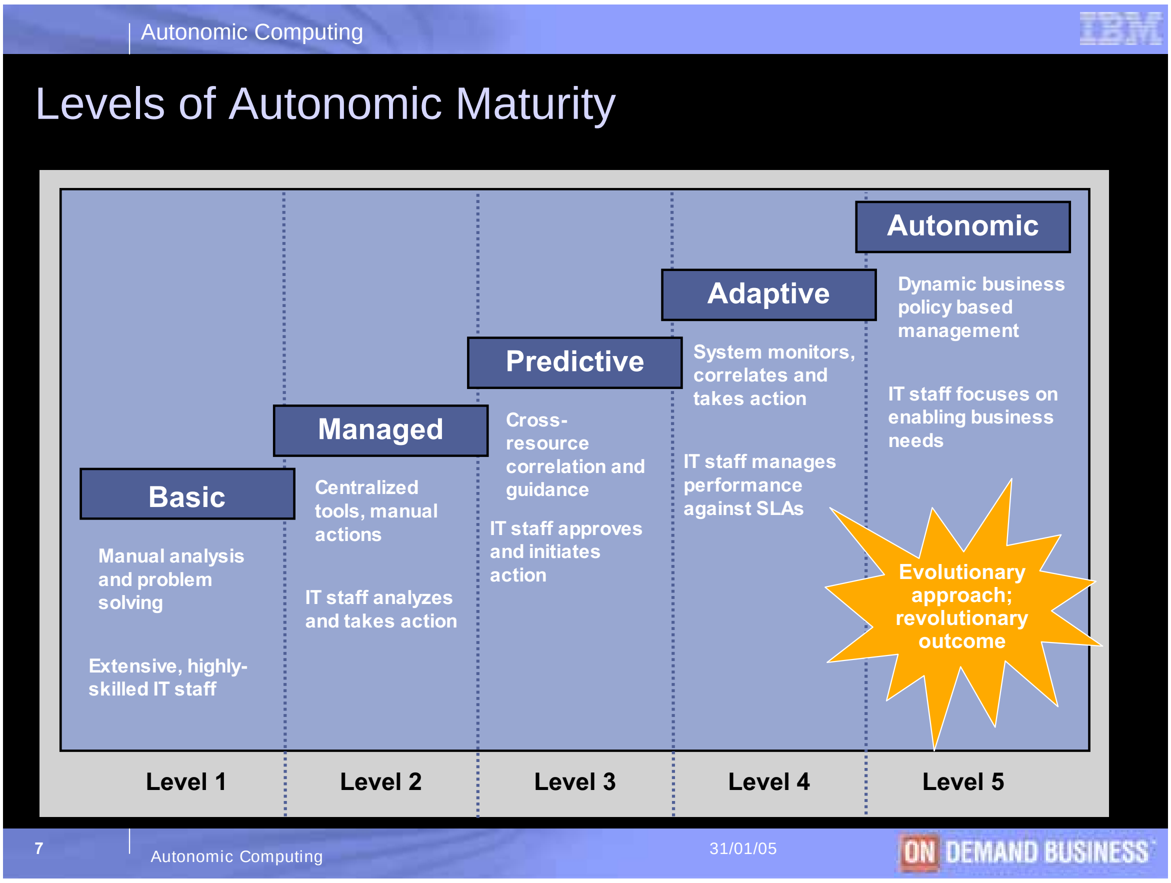
" IBM's autonomic computing initiative will become its most important cross-product initiative (as the foundation of On Demand Business)."

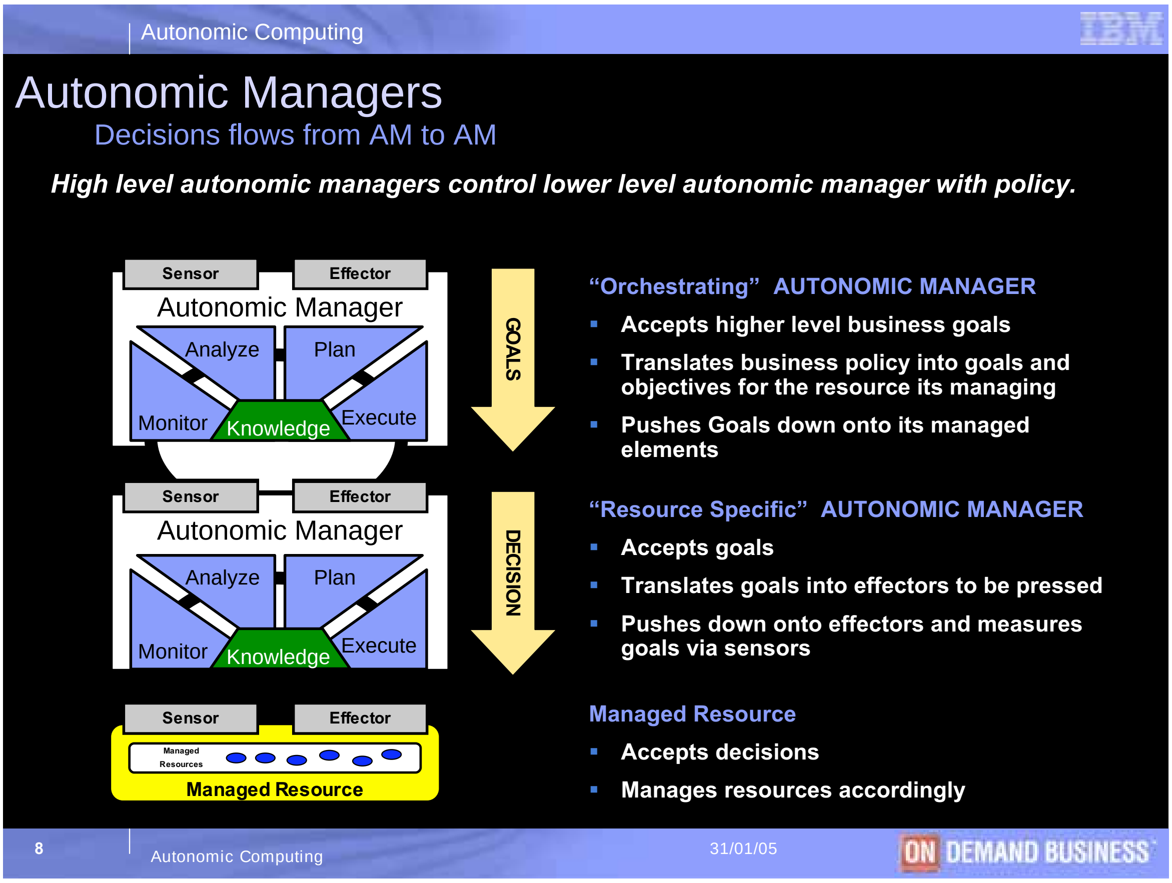
— Thomas Bittman, Gartner

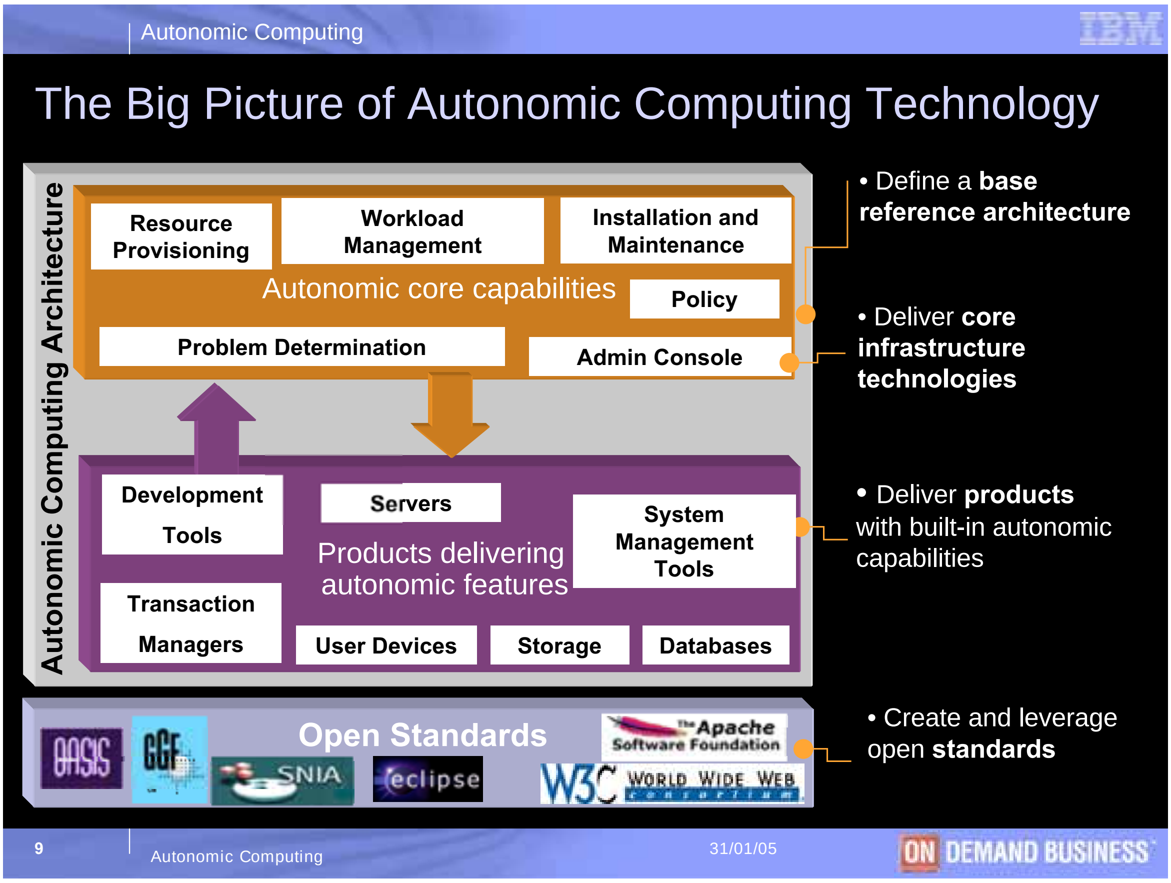
4 Autonomic Computing 31/01/05 ON DEMAND BUSINESS


Autonomic Computing
IBM
Current automation practices typically represent only a portion of the autonomic computing architecture
Automonic
Sensors
Effectors
Autonomic Element
Analyze
Plan
Monitor
Knowledge
Execute
Sensors
Effectors
Element
Complex Automation
If/Then/Else Analysis
Monitor
Execute
Element
Simple Automation
Trigger
Monitor
Execute
Element
5
Autonomic Computing
31/01/05
ON DEMAND BUSINESS


Autonomic Computing
IBM
Autonomic Computing Architecture Concepts
Sense and respond
Sensors
Effectors
Analyze
Plan
Monitor
Knowledge
Execute
Data
Action
Sensors
Effectors
Managed Element
Resource
Manageability
6
Autonomic Computing
31/01/05
ON DEMAND BUSINESS


Autonomic Computing
IBM
Levels of Autonomic Maturity
Basic
Manual analysis and problem solving
Extensive, highly-skilled IT staff
Managed
Centralized tools, manual actions
IT staff analyzes and takes action
Predictive
Cross-resource correlation and guidance
IT staff approves and initiates action
Adaptive
System monitors, correlates and takes action
IT staff manages performance against SLAs
Autonomic
Dynamic business policy based management
IT staff focuses on enabling business needs
Evolutionary approach; revolutionary outcome
Level 1
Level 2
Level 3
Level 4
Level 5
7
Autonomic Computing
31/01/05
ON DEMAND BUSINESS


Autonomic Computing
IBM
Autonomic Managers
Decisions flows from AM to AM
High level autonomic managers control lower level autonomic manager with policy.
Sensor
Effector
Autonomic Manager
Analyze
Plan
Monitor
Knowledge
Execute
GOALS
Sensor
Effector
Autonomic Manager
Analyze
Plan
Monitor
Knowledge
Execute
DECISION
Sensor
Effector
Managed Resources
Managed Resource
"Orchestrating" AUTONOMIC MANAGER
Accepts higher level business goals
Translates business policy into goals and objectives for the resource its managing
Pushes Goals down onto its managed elements
"Resource Specific" AUTONOMIC MANAGER
Accepts goals
Translates goals into effectors to be pressed
Pushes down onto effectors and measures goals via sensors
Managed Resource
Accepts decisions
Manages resources accordingly
8
Autonomic Computing
31/01/05
ON DEMAND BUSINESS

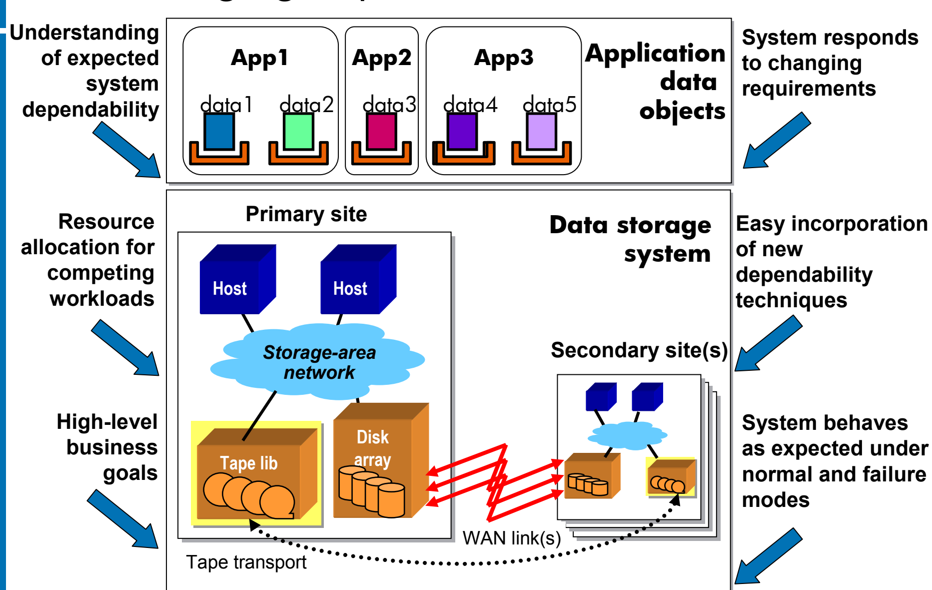

Autonomic Computing
IBM
The Big Picture of Autonomic Computing Technology
Autonomic Computing Architecture
Resource Provisioning
Workload Management
Installation and Maintenance
Autonomic core capabilities
Policy
Problem Determination
Admin Console
Development Tools
Servers
Products delivering autonomic features
System Management Tools
Transaction Managers
User Devices
Storage
Databases
• Define a base reference architecture
• Deliver core infrastructure technologies
• Deliver products with built-in autonomic capabilities
Open Standards
OASIS
GGF
SNIA
eclipse
The Apache Software Foundation
W3C World Wide Web Consortium
• Create and leverage open standards
9
Autonomic Computing
31/01/05
ON DEMAND BUSINESS

Automating data dependability

47th Meeting of IFIP WG 10.4 – "Autonomic Web Computing"
January 26-30, 2005
Rincon, Puerto Rico, USA

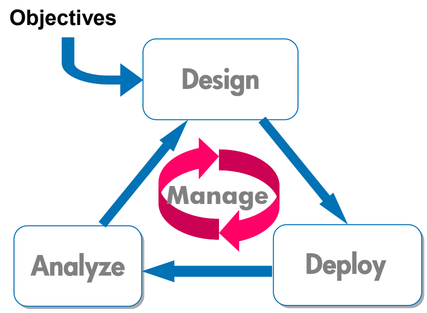
Kim Keeton, Dirk Beyer, Jeff Chase, Arif Merchant, Pano Santos and John Wilkes
Hewlett-Packard Labs and Duke University
{kimberly.keeton,dirk.beyer,arif.merchant,cipriano.santos,john.wilkes}@hp.com, chase@cs.duke.edu

Self-managing dependable data service



January 29, 2005 2

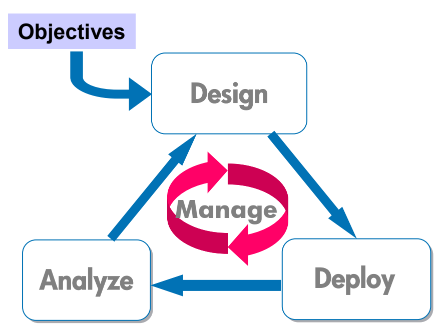
Automated data dependability



- Defining the desired level of service
- Designing
- Deploying
- Analyzing

the system that powers the service

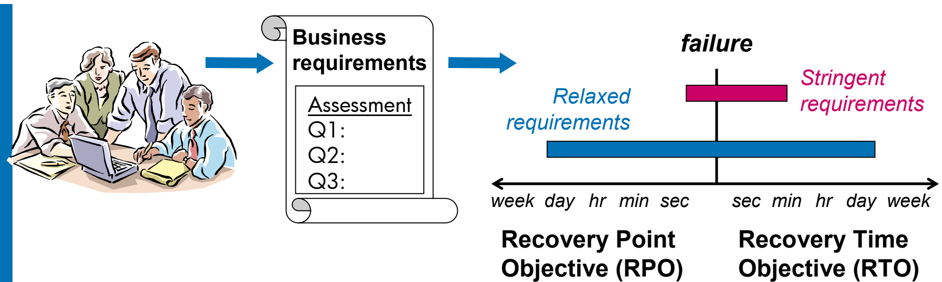
Outline



- Defining the desired level of service
- Designing
- Deploying
- Analyzing

the system that powers the service

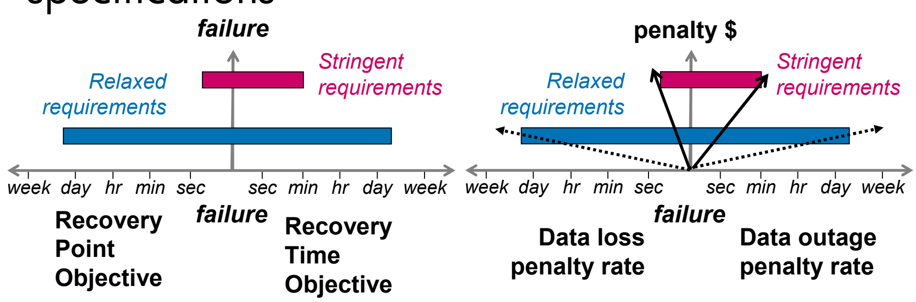
Challenge: expressing dependability goals



- Better (e.g., more quantitative) goals lead to better system designs
  - Users can't always state goals quantitatively
  - Specifying quantitative utility functions even harder
  - Users often possess intangible goals (e.g., manageability, training)
- Challenges:
  - Capturing utility-based goals in a quantitative fashion
  - Expressing intangible goals

January 29, 2005 5

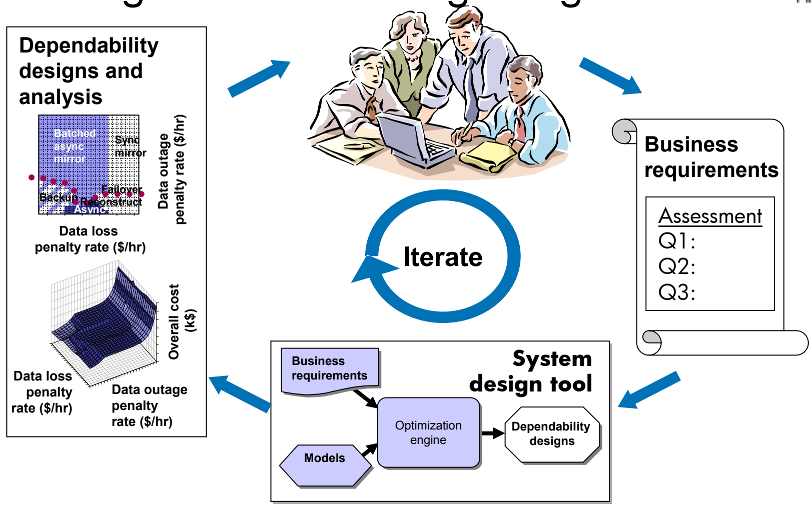
Approach: quantitative utility-based specifications



- Data outage penalty rate ($/hour)
  - How long before the system is back up?
- Data loss penalty rate ($/hour)
  - How much recent data can the system discard?
- Time-varying penalty rates
  - Allow differentiation between short and long durations
  - Allow specification of constraints (RTO/RPO + violation penalties)

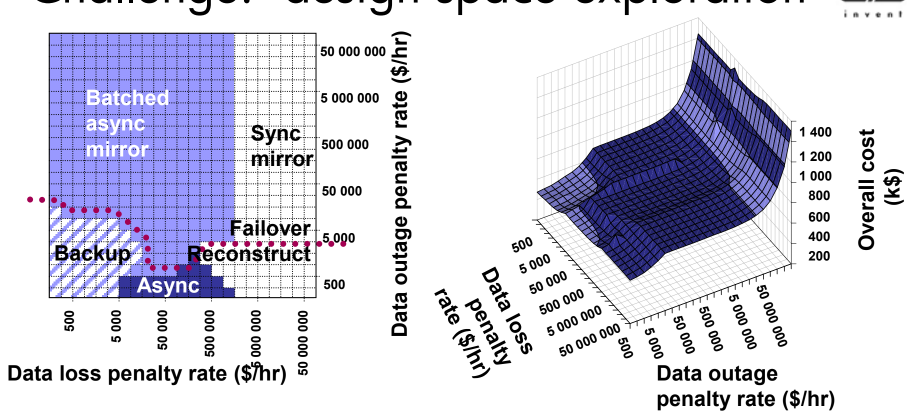
January 29, 2005 6

Challenge: understanding design choices



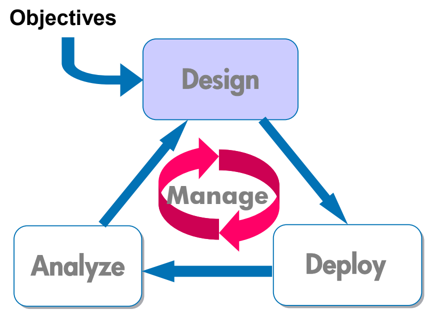
- Challenge: giving users feedback on design choice implications

Challenge: design space exploration



- Representing different choices for different objects
- Illustrating sensitivity to input choices
  - Business requirements, workload characteristics, failure likelihoods
- How to avoid overwhelming user with too much info?

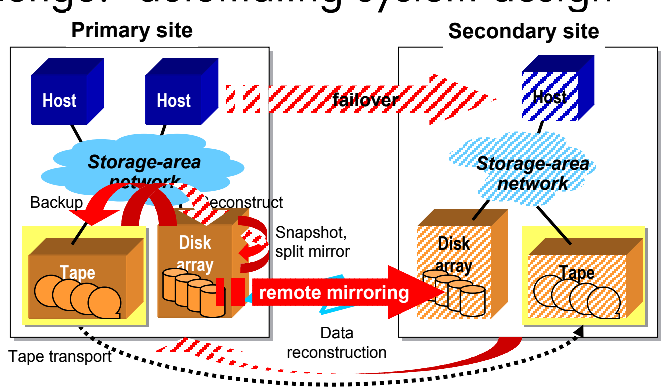
Outline



- Defining the desired level of service
- Designing
- Deploying
- Analyzing

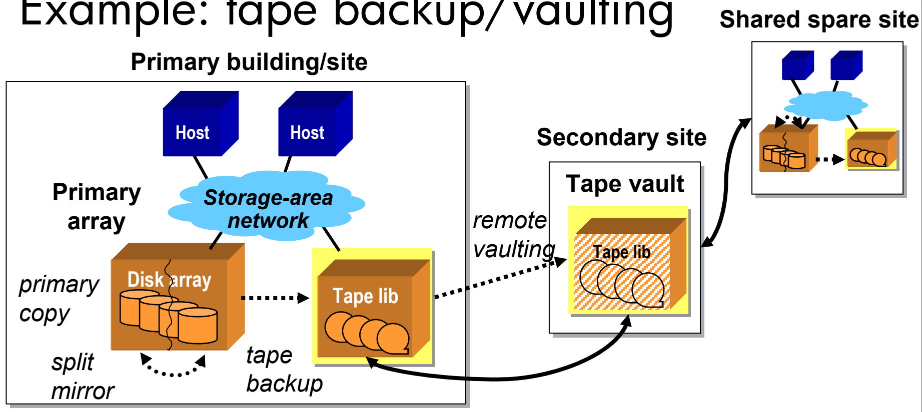
the system that powers the service

Challenge: automating system design



- Automatically designing dependable (storage) system
  - From scratch
  - Based on existing legacy system
- Choosing appropriate techniques to protect workload data, and how to set config parameters
- Allocating physical resources to protection workloads

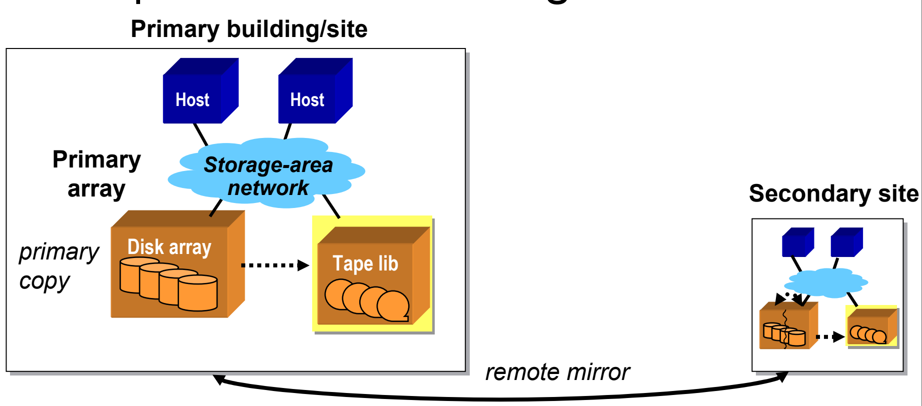
Example: tape backup/vaulting



- Backup configuration questions:
  - How long between successive backups?
  - How often to do full vs. incremental backups?
  - How long should backup window be?
  - How long to keep backups?
- Vaulting configuration questions:
  - How often to ship tapes offsite?
  - How long to delay before shipping?
  - What to ship offsite?

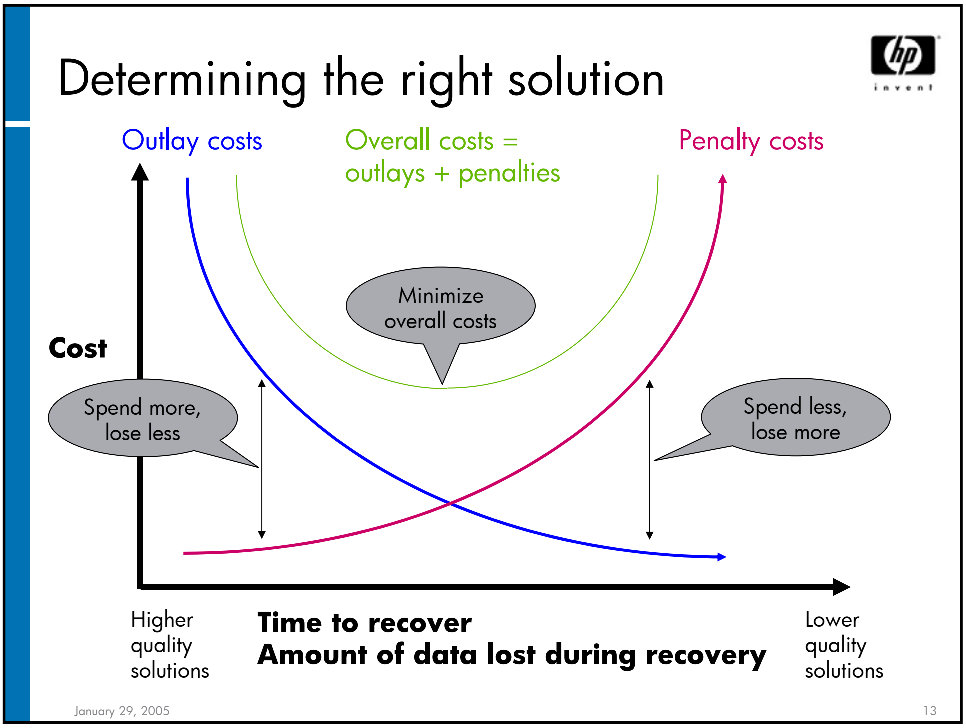
January 29, 2005 11

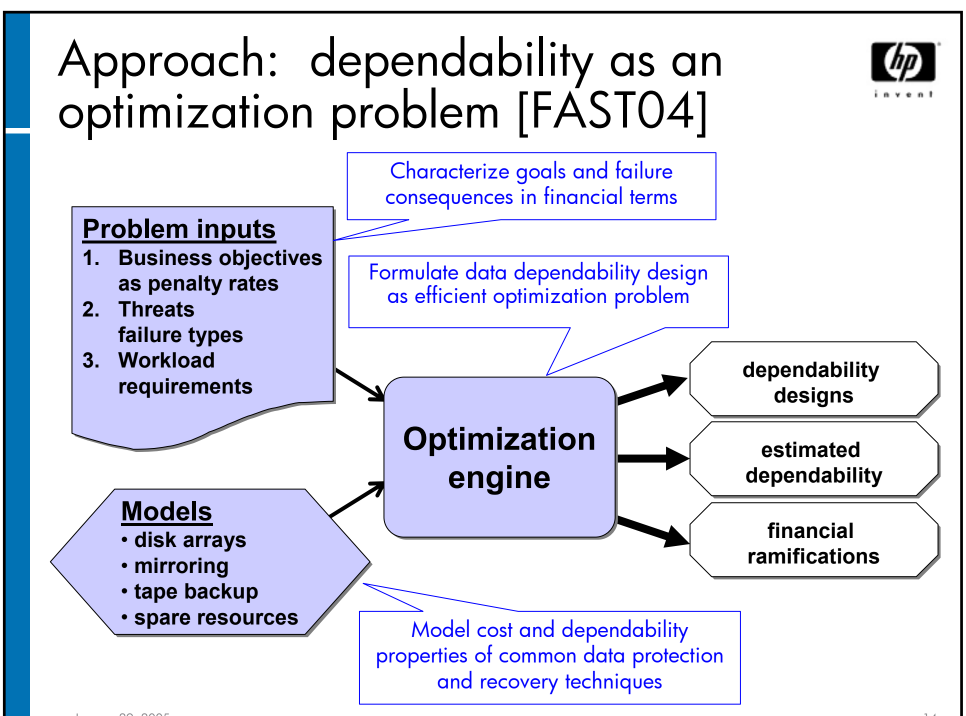
Example: remote mirroring



- Remote mirroring configuration questions:
  - What protocol to use – synchronous or asynchronous?
  - If asynchronous batch protocol, how long to coalesce updates?
  - How many network links to use?

January 29, 2005 12


Determining the right solution
Outlay costs
Overall costs =
outlays + penalties
Penalty costs
Minimize
overall costs
Cost
Spend more,
lose less
Spend less,
lose more
Higher
quality
solutions
Time to recover
Amount of data lost during recovery
Lower
quality
solutions
January 29, 2005
13

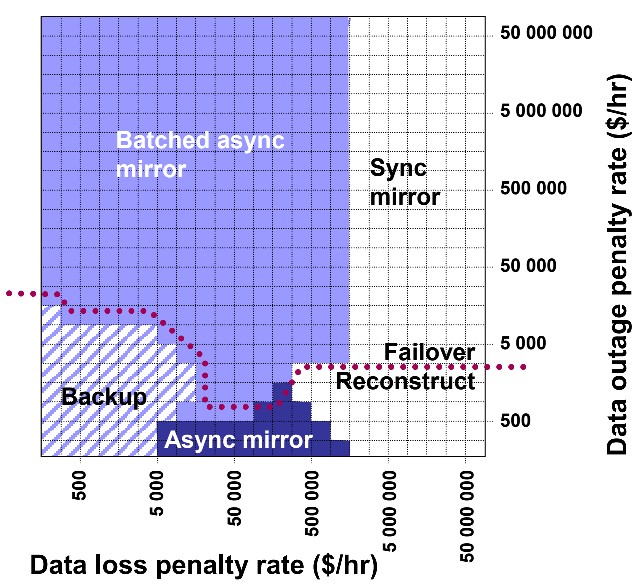

Approach: dependability as an
optimization problem [FAST04]
Characterize goals and failure
consequences in financial terms
Problem inputs
1. Business objectives
as penalty rates
2. Threats
failure types
3. Workload
requirements
Formulate data dependability design
as efficient optimization problem
Optimization
engine
dependability
designs
estimated
dependability
financial
ramifications
Models
• disk arrays
• mirroring
• tape backup
• spare resources
Model cost and dependability
properties of common data protection
and recovery techniques
January 29, 2005
14

Mixed integer programming formulation

- Objective function
  - Minimize overall business cost = outlays + penalties
- Decision variables
  - Binary variables to select an alternative and its configuration
  - Integer variables for number of bandwidth devices (e.g., mirroring links or tape drives)
- Constraints
  - Allowable design alternatives
  - Bandwidth and capacity provisioning
  - Linearization constraints
- Solver prototype
  - Implementation using off-the-shelf optimization engine (CPLEX)

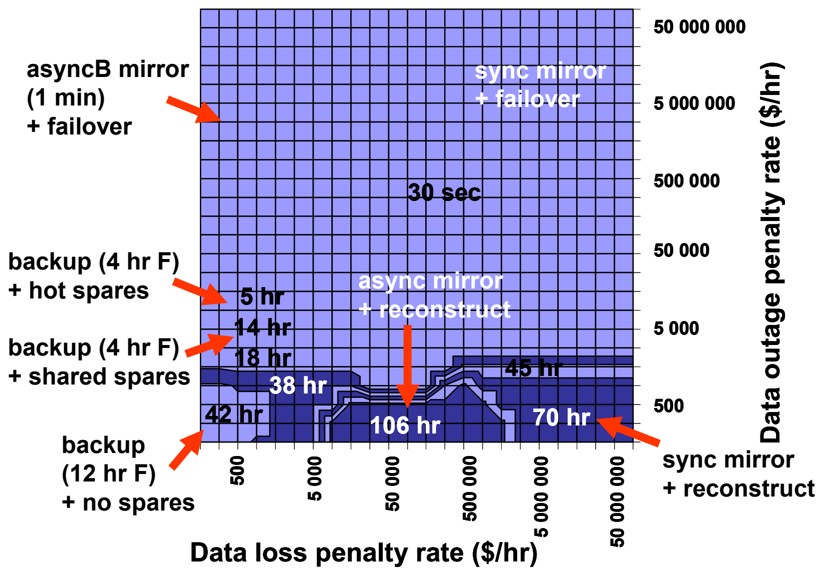
January 29, 2005 15

Design space exploration



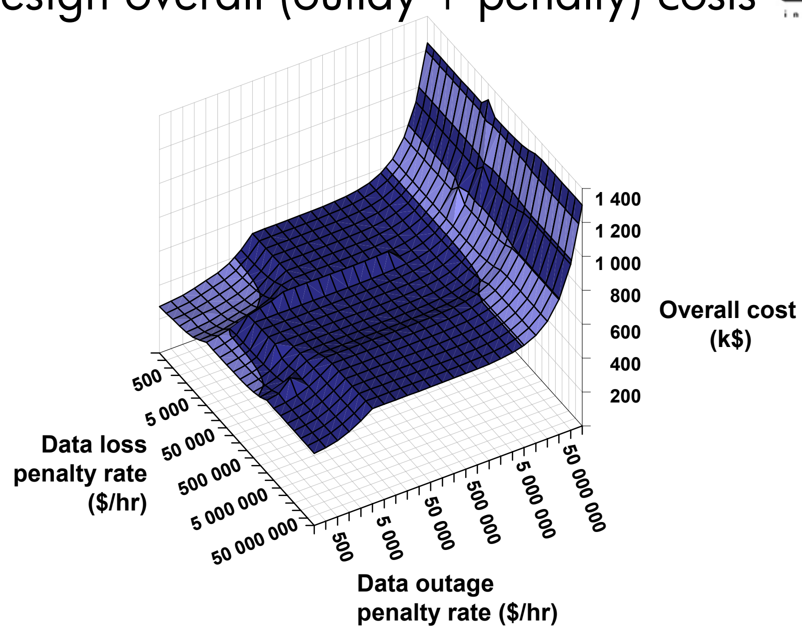
January 29, 2005 16

Design recovery time



January 29, 2005 17

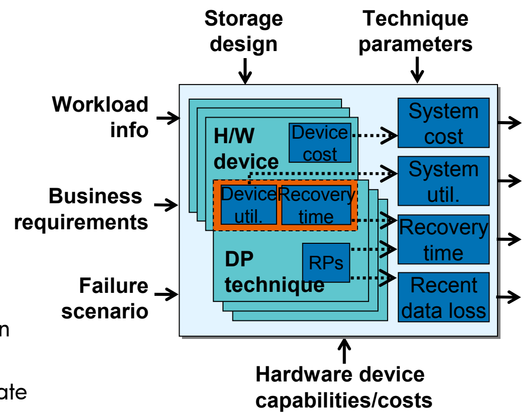
Design overall (outlay + penalty) costs



January 29, 2005 18

Challenge: new dependability techniques

- Issues:
  - Easily incorporate new techniques
  - Complex storage solutions: multiple techniques
- Approach: extensible modeling framework [DSN04]
  - Model secondary copy commonalities
    - Full vs. partial representation
    - Copy frequency, retention
    - Time for updates to propagate
  - Composition rules to evaluate overall solution recovery time and data loss



January 29, 2005 19

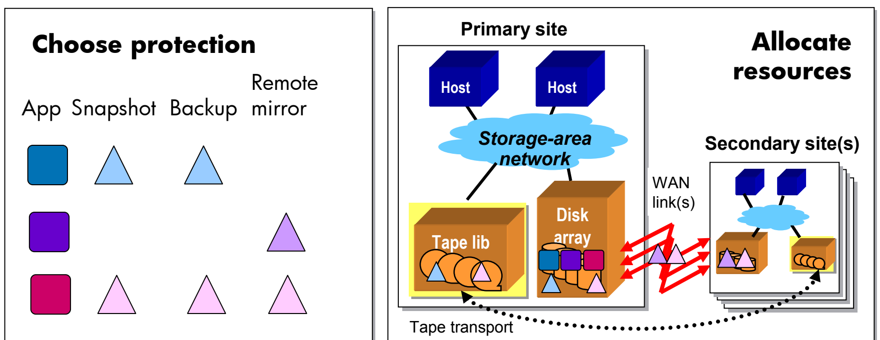
Open questions: new dependability techniques

- "Grammar" to describe reasonable combinations of dependability techniques
- Extending framework to higher-level techniques (e.g., logging, checkpointing)
- Modeling tradeoffs between:
  - Techniques at different layers of stack
    - Block-level replication vs. log shipping
  - Techniques using different resources
    - Recompute vs. store intermediate results

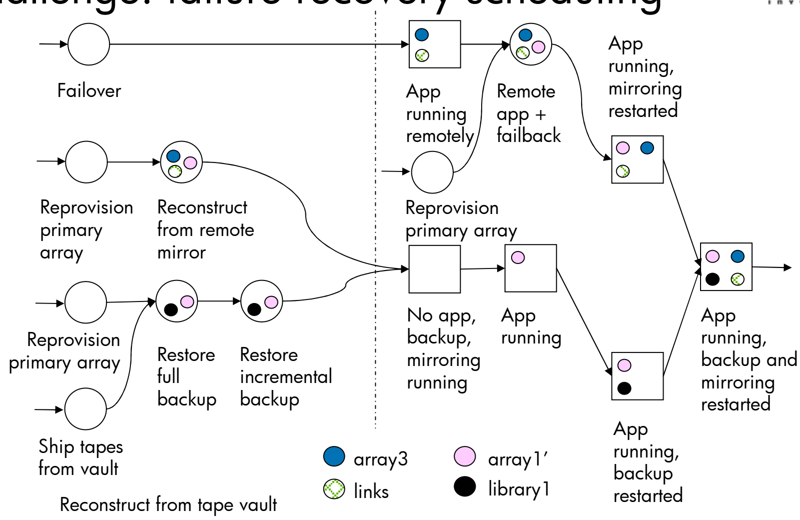
January 29, 2005 20

Challenge: competing data objects

- Must choose protection and recovery alternatives and allocate physical resources per data object
- Potential approaches:
  - Two-phase optimization heuristic
  - Evaluation + randomized search



Challenge: failure recovery scheduling



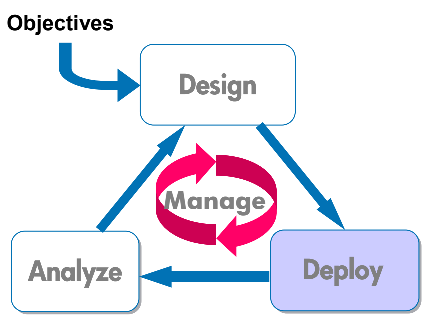
- Choosing the best set of recovery operations
- Determining how to schedule recovery operations and unaffected workloads

End-to-end dependability design

- Goal: end-to-end dependability
  - Business processes and applications are unit of dependability
  - Continuous service operation (“business continuity”)
- Challenges:
  - Provisioning system resources (servers, storage, networks)
  - Effectively using techniques at all levels of application stack
    - Snapshots, checkpointing, logging and replication
    - Failover and recomputation of results
  - Managing interactions and tradeoffs between techniques
  - Translating end-to-end dependability goals into system component goals

January 29, 2005 23

Outline



- Defining the desired level of service
- Designing
- Deploying
- Analyzing

the system that powers the service

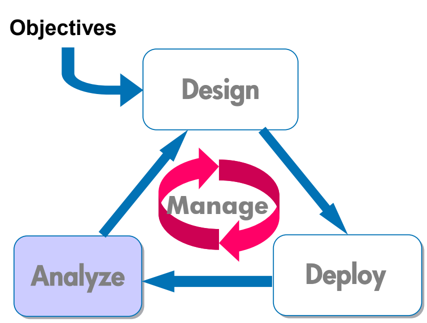
January 29, 2005 24

Deployment challenges

- Implementing dependable storage designs
  - Ex: interacting with backup software to adjust backup frequency
- Implementing recovery operations in response to failures
- Providing online data layout
  - Ex: RAID level selection [Anderson, et al., FAST2002]
- Migrating data in response to system changes

January 29, 2005 25

Outline

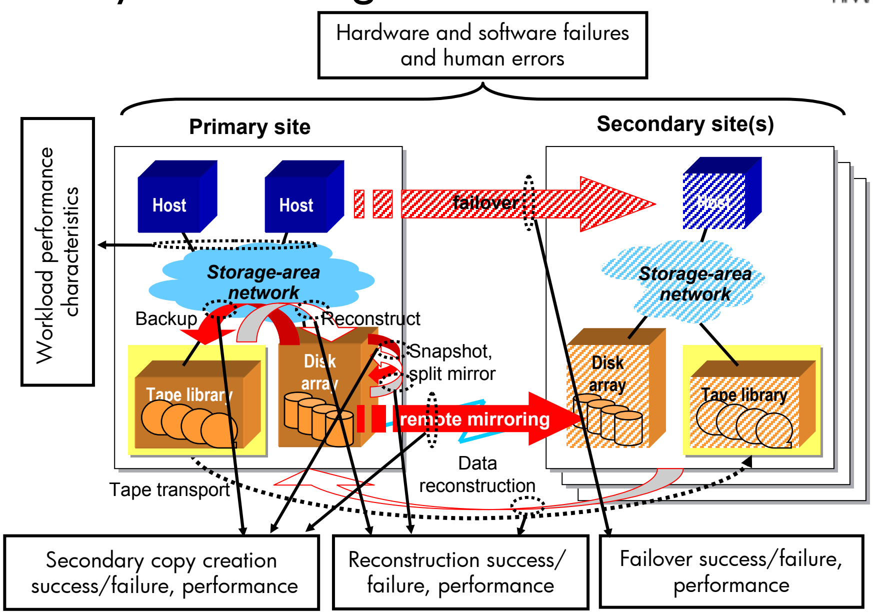


- Defining the desired level of service
- Designing
- Deploying
- Analyzing

the system that powers the service

January 29, 2005 26

Analysis challenges



January 29, 2005 27

Open questions for analysis

- Verifying correct deployment of techniques
- Measuring resource requirements of successful secondary copy creation
- Diagnosing problems when they occur
- Collecting data on recovery behavior
  - Measure disaster drills and naturally occurring problems
  - Proactive small-scale fault injection using virtualization
- Using measurements to iteratively refine models
- Understanding workload characteristics
  - Steady-state behavior, trends and cyclic behavior

January 29, 2005 28

Conclusions

- Designing and managing dependable systems is challenging
  - Competing workload demands
  - Dynamic environments
  - Desire that system meets expectations
  - End-to-end dependability
- Automated data dependability provides starting point
  - Define desired level of service
  - Design, deploy, analyze system behind the service
- Wealth of research opportunities – join us!
- Further details available:
  - http://www.hpl.hp.com/SSP/
  - kimberly.keeton@hp.com

January 29, 2005 29

Backup slides

January 29, 2005 30

Data dependability bibliography

- [FAST04]: "Designing for disasters," K. Keeton, C. Santos, D. Beyer, J. Chase and J. Wilkes, *Proc. 3rd Conference on File and Storage Technologies (FAST),* March 2004.
- [DSN04]: "A framework for evaluating storage system dependability," K. Keeton and A. Merchant, *Proc. Intl. Symposium on Dependable Systems and Networks (DSN),* June 2004.
- [SIGOPS04]: "Lessons and challenges in automating data dependability," K. Keeton, D. Beyer, J. Chase, A. Merchant, C. Santos and J. Wilkes, *Proc. 11th SIGOPS European Workshop,* September 2004.
- [SIGOPS02]: "Automating data dependability," K. Keeton and J. Wilkes, *Proc. 10th SIGOPS European Workshop*, September 2002.

- Further details available:
  - http://www.hpl.hp.com/SSP/
  - kimberly.keeton@hp.com

January 29, 2005 31

Related work

- Dependability modeling and simulation techniques [Deavours2002, Haverkort2001, Kaaniche1998]
- System administration literature: operational issues [Chervenak1998, daSilva1993]
- Backup and return-on-investment calculators [Sun, EMC]
- Development, application of new data protection techniques [Rhea2003, Wylie2001]
- Specifying and evaluating dependability requirements [Keeton2002, Wilkes2001, Brown2000]
- Automatic storage design for performance goals [Anderson2002, Alvarez2001]
- Automatic tuning of application computation resources in multi-tier environments: [Janakiraman2004]

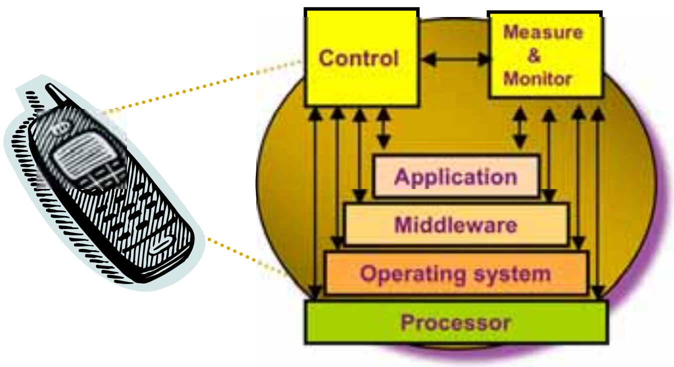
January 29, 2005 32

Adaptive Application-Aware Runtime Checking

Ravi Iyer, Z. Kalbarczyk, N. Nakka, L. Wang, N. Breems et. al
Center for Reliable and High-Performance Computing
Coordinated Science Laboratory
University of Illinois at Urbana-Champaign
www.crhc.uiuc.edu/DEPEND

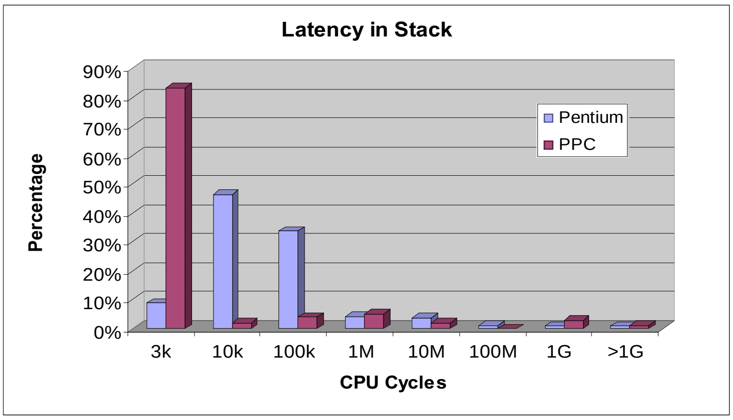
http://www.crhc.uiuc.edu/DEPEND/

The Embedded Environment: Cell Phones



- Modular design of processes lends itself well to small footprint solutions.
- Specialized Applications optimized for memory/performance requirements.
- Specialized/Customized kernels

Crash Latency
Stack Injection *(Linux on Pentium and PowerPC)*



Early detection of kernel stack overflow on PPC major contributor to reduced crash latency

What is Needed?

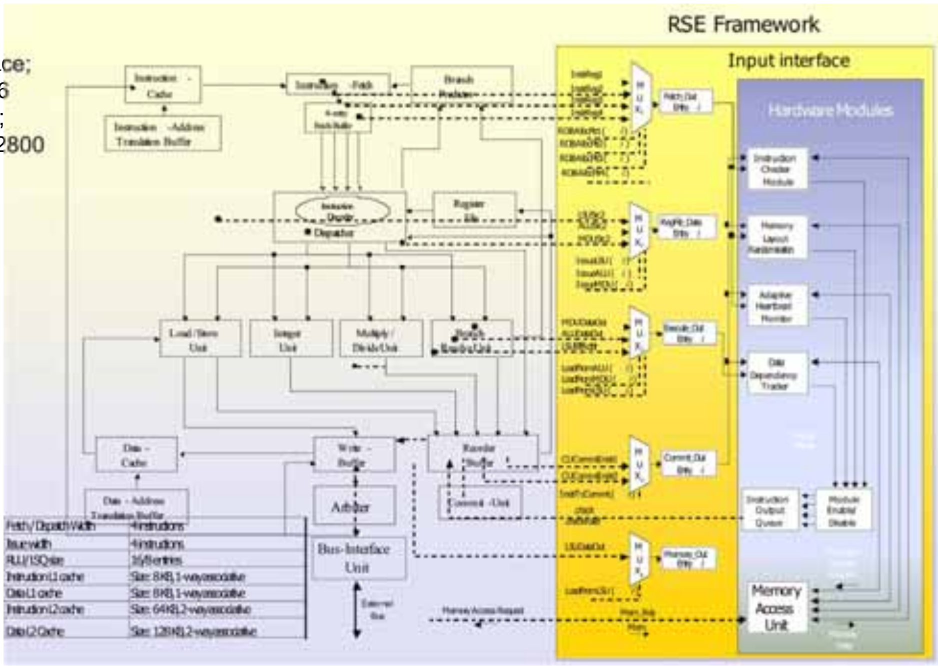
- **A hardware/software framework that adapts dynamically to application needs**
- **Extracting application properties that can be used as an indicator of correct behavior and to drive synthesis of application-aware checks**
- **Instantiating the optimal hardware/software for runtime application checking**
- **Embed the devised checks into the custom hardware or software middleware or operating system**

Adaptive Application Aware Checking in Hardware: Basics

- **Static source-code analysis and profiling provides**
  - **Which checkers to be used and at what points of application execution**
  - **Checkers are adapted to application**
- **Hardware modeling using HDL**
- **Synthesize modules into reconfigurable hardware framework**
- **Modules themselves are runtime reconfigurable**

Adaptive Application Aware Checking in Hardware: Reliability and Security Engine

For Input Interface;
Queue Size = 16
32-bit regs = 80;
Gate Count = 12800



N. Nakka, J. Xu, Z. Kalbarczyk, R. K. Iyer, "An Architectural Framework for Providing Reliability and Security Support", DSN2004.

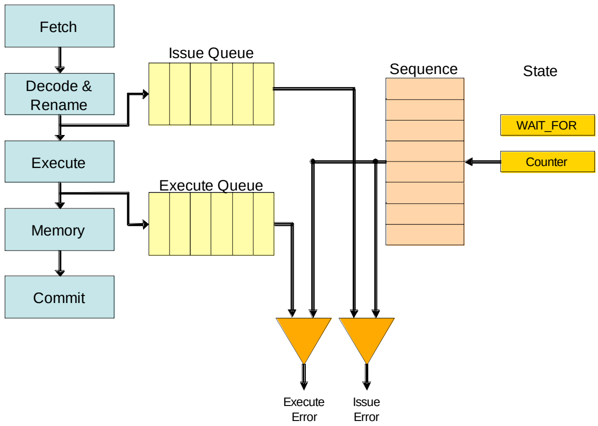
The Processor-Level Framework

- Implemented as an integral part of the processor on the same die
- Embeds hardware modules for reliability, security and recovery that execute in parallel with the instruction execution in the main pipeline
- Provides a generic interface to external processor system through which modules access runtime information for checking
- Application interfaces to framework through CHECK instructions
  - Extension of the ISA
  - Used by the application to invoke specific modules

Detection of Instruction Dependency Violations

- RAW dependency imposes sequential order on execution of instructions
- Errors in processor control logic, binary of instruction can lead to a violation
- Sequence Checker Module (SCM), detects such violations
  - monitors issue and execute events in pipeline
- Representative instruction sequences extracted using static analysis
- CHECKs used to dynamically reconfigure the module with sequences

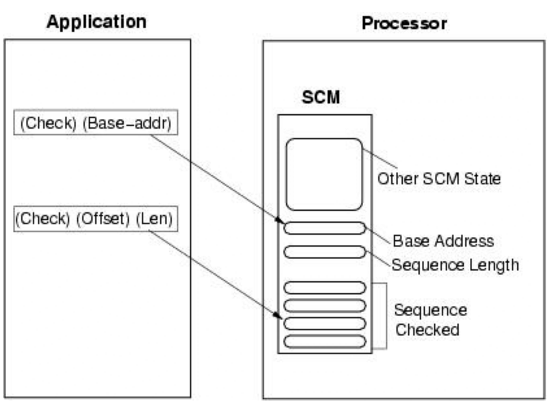
SCM Detection Mechanism



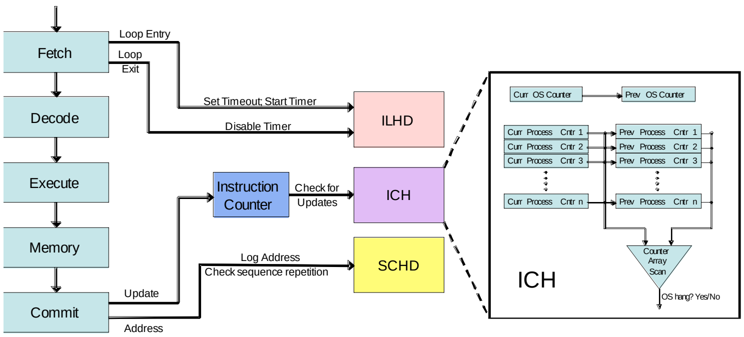
- SCM state for sequence – (*i, e*)
  - *i* : instruction on which event is awaited
  - *e* : event (issue/execute)
- Property – at any instance of time, *at most one* instruction of a dependent sequence can be issued or executed
- Instructions in issue and execute queues matched against instructions of sequence
- *at most* one instruction from the queue should match the correct state
- Error Detected when there is :
  - more than one match
  - a match other than expected state

SCM Reconfiguration Architecture

- Achieved with help of CHECK instructions
- Extracted sequences loaded as part of program image
- At runtime SCM loads sequences into set of registers
- Each sequence has additional registers
  - length, state



Process Crash/Hang Detection (1)



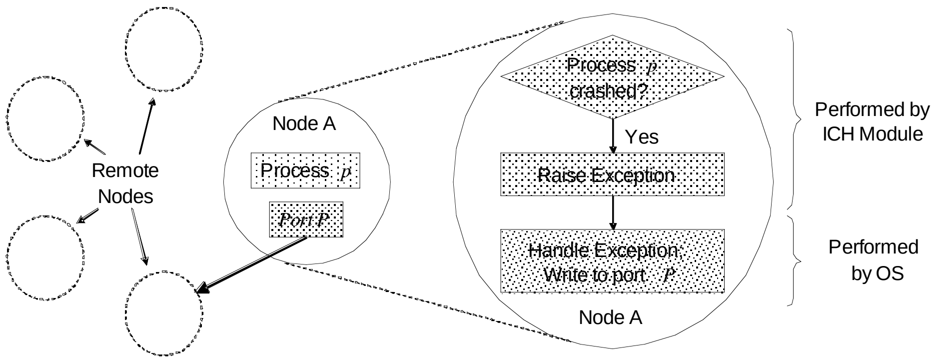
- Infinite Loop Hang Detection (ILHD) by tracking loop entry and exit points
- Sequential Code Hang Detection (SCHD) detects illegal repetition of sequence of instructions
- Instruction Count Heartbeat (ICH) leverages processor performance registers to detect process/OS crashes/hangs

Process/Crash Hang Detection (2)

- Process hang in legal loops
  - Infinite loop Hang Detector (ILHD)
  - Profile-based analysis of application to estimate loop execution time
  - Module reconfigured with timeout for loop as it is entered – CHECK Loop Entry and Loop Exit
- Process hang in illegal loops
  - Sequential code hang detector (SCHD)
  - Parameterize module with length of loop
  - Any loop shorter than given length indicates control error

Process Crash/Hang Detection

- Crash detection
  - Instruction Count Heartbeat (ICH)
  - Uses processor performance counters to detect process and OS crashes
  - Can be extended to support failure detection in distributed systems



Adaptive Application Aware Checking in Software: Runtime Executive (RTE) – Middleware

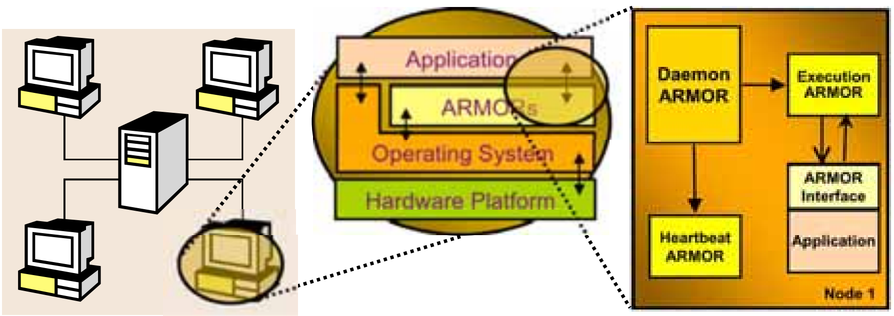
- Reconfigurable statically and dynamically to provide range of customizable error checks to operating system and applications, e.g.,
  - Heartbeats – (i) *adaptive* - the timeout value adapts to changes in the network traffic or node load and (ii) *smart* - the monitored entity excites a set of checks before sending the heartbeat) .
  - Data-Flow Signatures – a pattern of reads and writes to variables in a code block (program object, thread, function, basic block or instruction)
- Self-checking (self-healing)
- Example – reconfigurable ARMOR architecture
  - K. Whisnant, Z. Kalbarczyk, R. Iyer, "A System Model For Dynamically Reconfigurable Software," IBM Systems Journal, Special Issue on Autonomic Computing, March 2003

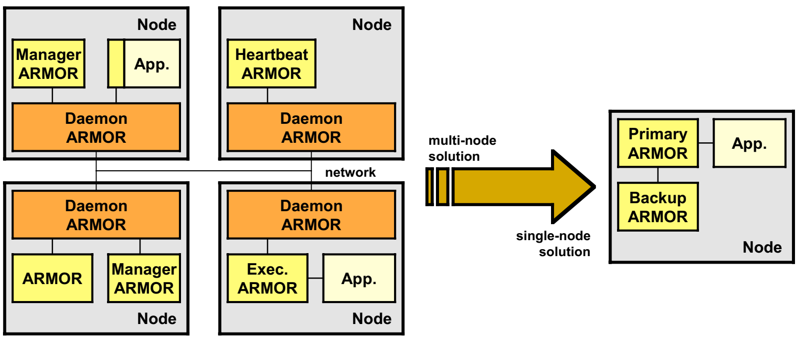
Runtime Executive (RTE): ARMOR Approach

- **Adaptive Reconfigurable Mobile Objects of Reliability:**
  - Multithreaded processes composed of replaceable building blocks called elements
  - Elements provide error detection and recovery services to user applications or operating system.
- **Hierarchy of ARMOR processes form runtime environment:**
  - ARMOR runtime environment is itself self checking
- **ARMOR properties**
  - designed to be reconfigurable
  - resilient to errors by leveraging multiple detection and recovery mechanisms
  - internal self-checking mechanisms to prevent failures from occurring and to limit error propagation.
  - state protected through checkpointing.

Runtime Executive (RTE): ARMOR Approach "Total Solution"

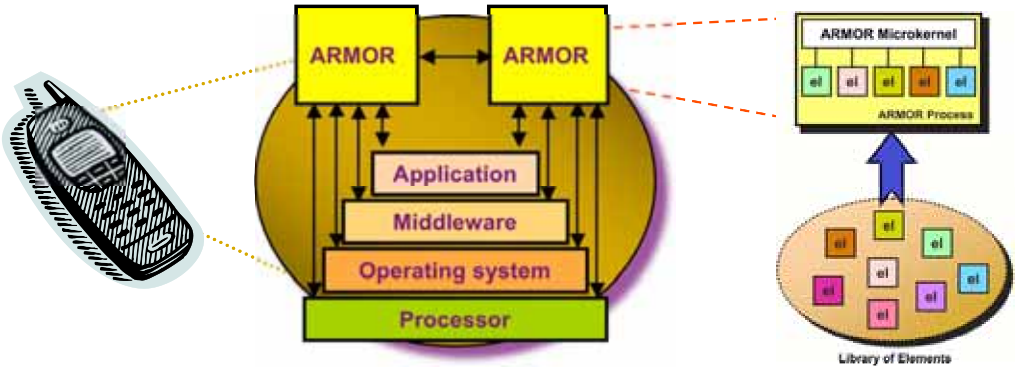
Basic Configuration





Scaling ARMOR Runtime Environment

Runtime Executive (RTE): ARMOR Approach "Embedded Solution"



- Modular design of processes lends itself well to small footprint solutions.
- Special elements optimized for memory/performance requirements.
- Specialized microkernel:
  - Remove support for inter-ARMOR communication through regular messaging
  - Static configuration of elements; no need to dynamically change elements

Support for Adaptation of Error Detection Across System Hierarchy

Application Support
Robust (self-checking) middleware
Runtime executive for fault/error management

Compiler Support
Application and/or OS instrumentation
to customize and invoke FT mechanisms

Operating System Support
Kernel health monitoring,
Application transparent checkpointing

Hardware Support
Reliability and security engine
Application-aware checks
Control logic protection

- Hardware –
  - a common processor-level framework exploiting features (e.g., debug and performance registers) of current processors
- Software
  - robust, self-checking runtime executive for fault management

Session 3

Security

Moderator and Rapporteur

Carl E. Landwehr, NSF, Arlington, VA, USA

Security Attacks and Defenses

Brian A. LaMacchia
Software Architect
Microsoft Corporation

47th Meeting of IFIP WG 10.4
January 29, 2005

Agenda

- Kinds of attacks
  - Infrastructure threats
  - Monetizing attacks
  - Social engineering threats (phishing)
- Defensive techniques
  - Automatic patching
  - Development tools
  - Run-time techniques
  - Leveraging automated feedback from customers

Kinds of Attacks

- Infrastructure attacks
  - OS/local machine
  - Web server
  - Network protocols
- Some techniques becoming more prevalent
  - SQL injections, cross-site scripting
    - Rooted in poor development practices
  - Building hitlists from Google & other public sources
    - Better saturation of vulnerable hosts
- We're *not* hearing about attacks on custom applications (if it's happening it's quiet)

29-JAN-2005 47th Meeting of IFIP WG 10.4 3

Attack Goals Shifting

- We've seen a dramatic shift in the past 12-18 months in the goal of these attacks
  - Used to be malicious behavior
  - Now it's financial
- Exploits are used to install Bots
  - Or the info is sold for $$$
- Networks of controlled exploited machines (BotNets) are then sold
  - Spammers
  - Organized crime

29-JAN-2005 47th Meeting of IFIP WG 10.4 4

Terminology

- Bot
  - Application that performs action on behalf of a remote controller
  - Installed on a victim machine (zombie)
  - Most are open-source
  - Modular (plug in your functionality / exploit / payload)
- BotNet
  - Linkage of “owned” machines into centrally controlled armies
  - Literally, *roBOT NETworks*
- Control Channel
  - Method for communicating with an army
- Herder
  - a.k.a. Bot herder, controller, pimp
  - Owns control channel, commands BotNet army
  - Motivations – money, power

Bots & BotNets

- Bots are prolific
  - Earthlink claims 20% of machines have bots and/or spy-ware
  - May account for 1/3 of all email traffic from comcast.net
- Spam
  - Bots sent 66% of all SPAM traffic on the Internet
  - Bots are rented to spammers
  - Provide mass mailing and anonymity
- Identity theft
  - Some versions include scanners for SSNs and credit card information
- DDoS / Extortion
  - Used for sustained DDoS attacks
  - Used for online extortion against Internet merchants
- Infringement/License violations
  - Scanners for CD keys and content

Monetizing BotNets

- First large-scale monetization done with MyDoom.A
  - Eight days after MyDoom.A hit the Internet, somebody scanned millions of IP addresses looking for the back door left by the worm
  - The attackers searched for systems with a Trojan horse called Mitglieder installed
  - Then used those systems as their spam engines
  - Millions of computers across the Internet were now for sale to the underground spam community

29-JAN-2005 47th Meeting of IFIP WG 10.4 7

BotNet Spammer Rental Rates

>20-30k always online SOCKs4, url is de-duped and updated every
>10 minutes. 900/weekly, Samples will be sent on request.
>Monthly payments arranged at discount prices.

- 3.6 cents per Bot week

>$350.00/weekly - $1,000/monthly (USD)
>Type of service: Exclusive (One slot only)
>Always Online: 5,000 - 6,000
>Updated every: 10 minutes

- 6 cents per Bot week

>$220.00/weekly - $800.00/monthly (USD)
>Type of service: Shared (4 slots)
>Always Online: 9,000 - 10,000
>Updated every: 5 minutes

- 2.5 cents per Bot week

September 2004 postings to SpecialHam.com, Spamforum.biz

29-JAN-2005 47th Meeting of IFIP WG 10.4 8

Current situation

- BotNets themselves unseen; uses are noticed
  - Spam relays
  - Identity theft, credit cards, keystrokes, other PII
  - DDoS attacks
- Ease of writing, deploying Bots is increasing
  - GUIs driven by script kiddies (13 year olds)
  - Many don't know how to program – "personalized" bots
  - Automatic scanning for vulnerable machines
- Threat is escalating
  - Low profile (vs. Slammer / MyDoom / phishing, etc.)
  - Financial opportunity driving activity
  - Model is maturing into tiers – herders, service providers
  - Numbers are increasing
  - Bot technologies are getting better

Bot Pedigree

- Relatively few "families" of Bots
  - Based on open source Bot collaboration efforts
  - Berbew, Gaobot, …
- Custom variants abound
  - Typically see 3 to 5 new variants per week
  - Have seen as many as 50 per day

BotNet use: Data Theft

Bots often have built-in functionality to steal

- Documents or data from an infected computer
- Computer passwords, IRC passwords
- Bank account numbers and passwords
- PayPal account info
- Credit card data
- Keystroke loggers

http://www.lurhq.com/phatbot.html

29-JAN-2005 47th Meeting of IFIP WG 10.4 11

Botnet use: Extortion

Small-scale: Even small BotNets (a few hundred machines) can extort online businesses for money.

- Small site in Kentucky taken down for a week because they refused to pay $10k

http://www.courier-journal.com/business/news2004/05/10/F1-scam10-8568.html

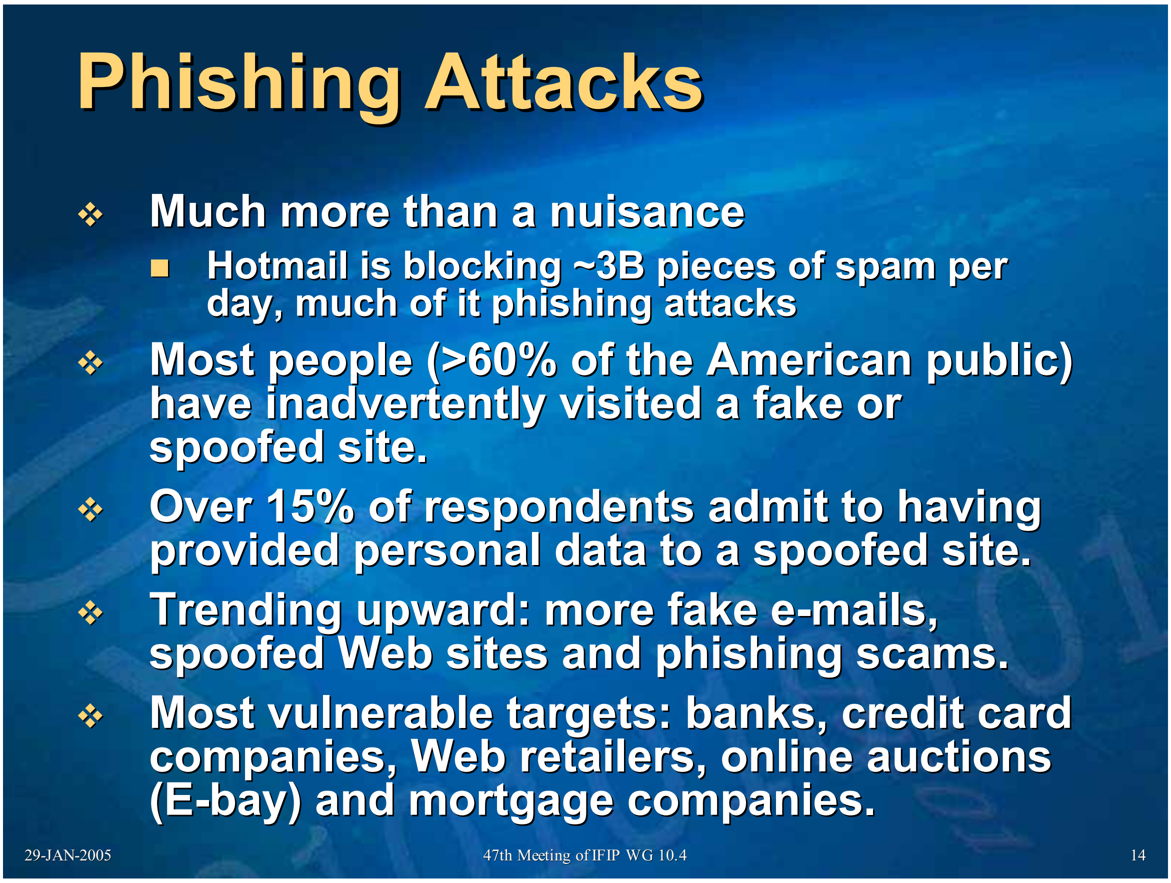
Large-scale: Crime rings extorting business for "protection monies“.

- A number of UK gambling sites have been offered protection for $50k/year

http://www.rense.com/general44/hack.htm

29-JAN-2005 47th Meeting of IFIP WG 10.4 12


Attack Trends
From isolated to networked
Attacker is on the "outside"
From programs to services
Unconstrained input
From multi-user to single user to multi-user
"User as admin" problem
From asynchronous to mass malware
Asymmetry favors attacker
From vandalism to for profit
More dedicated attackers
From specific to general to specific
Value will draw more sophisticated adversaries
29-JAN-2005
47th Meeting of IFIP WG 10.4
13


Phishing Attacks
Much more than a nuisance
Hotmail is blocking ~3B pieces of spam per day, much of it phishing attacks
Most people (>60% of the American public) have inadvertently visited a fake or spoofed site.
Over 15% of respondents admit to having provided personal data to a spoofed site.
Trending upward: more fake e-mails, spoofed Web sites and phishing scams.
Most vulnerable targets: banks, credit card companies, Web retailers, online auctions (E-bay) and mortgage companies.
29-JAN-2005
47th Meeting of IFIP WG 10.4
14

Losses from Phishing

- Estimated economic losses:
  - Small number of people (slightly more than 2%) affected, with an average cost of $115 dollars/victim.
  - Extrapolating to the entire U.S. population, economic impact of fraud close to $500M.

29-JAN-2005 47th Meeting of IFIP WG 10.4 15

Defensive Techniques

- Automated patching
- Development tools
- Run-time techniques
- Leveraging automated feedback from customers

29-JAN-2005 47th Meeting of IFIP WG 10.4 16

First, Some Numbers

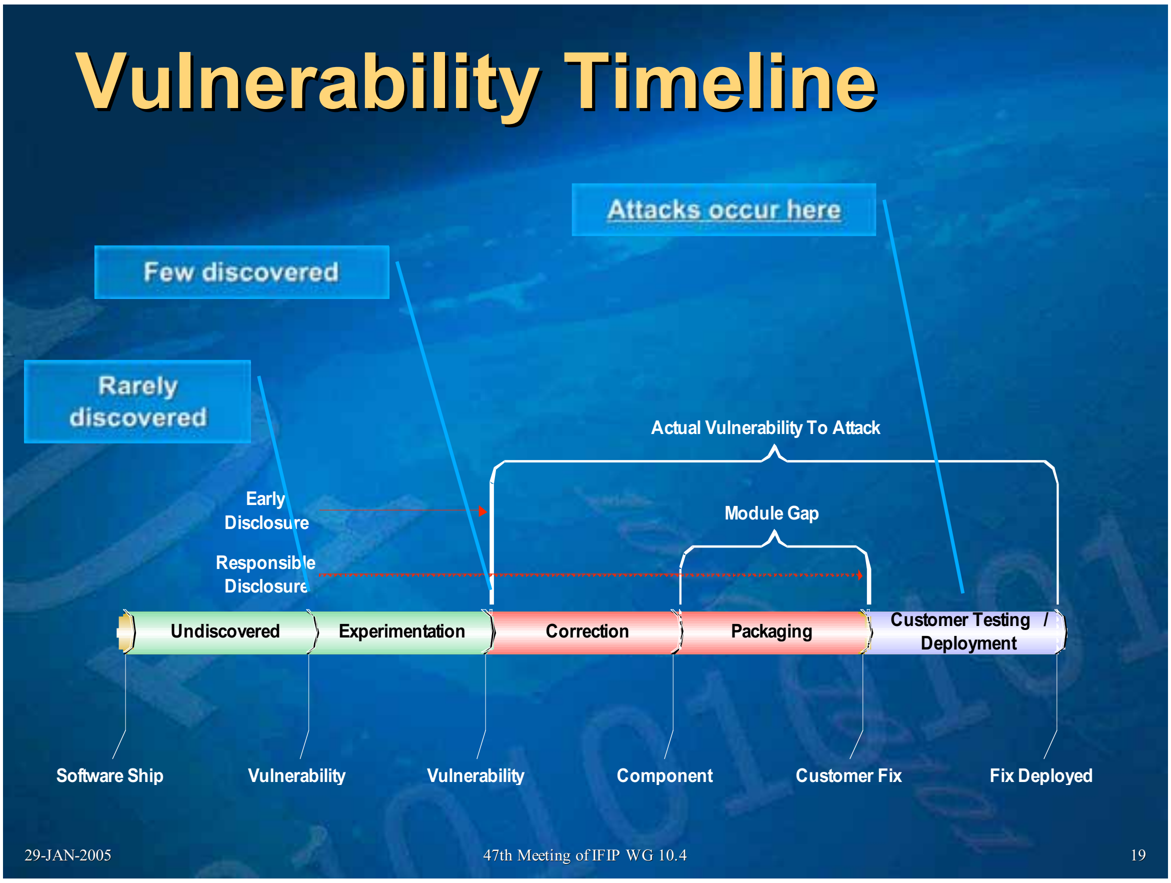
- 656.5M PCs run Windows Client worldwide
  - OEMs shipped 115.4M Windows PCs in 2004
- MS Malicious Software Removal Tool
  - Released 1/11/05 – targets 8 families of malware
  - As of 1/27/2005
    - Run over 104M times
    - Over 177K infected hosts cleaned
- MS Anti-Spyware Beta
  - Over 3M downloads in <2 weeks

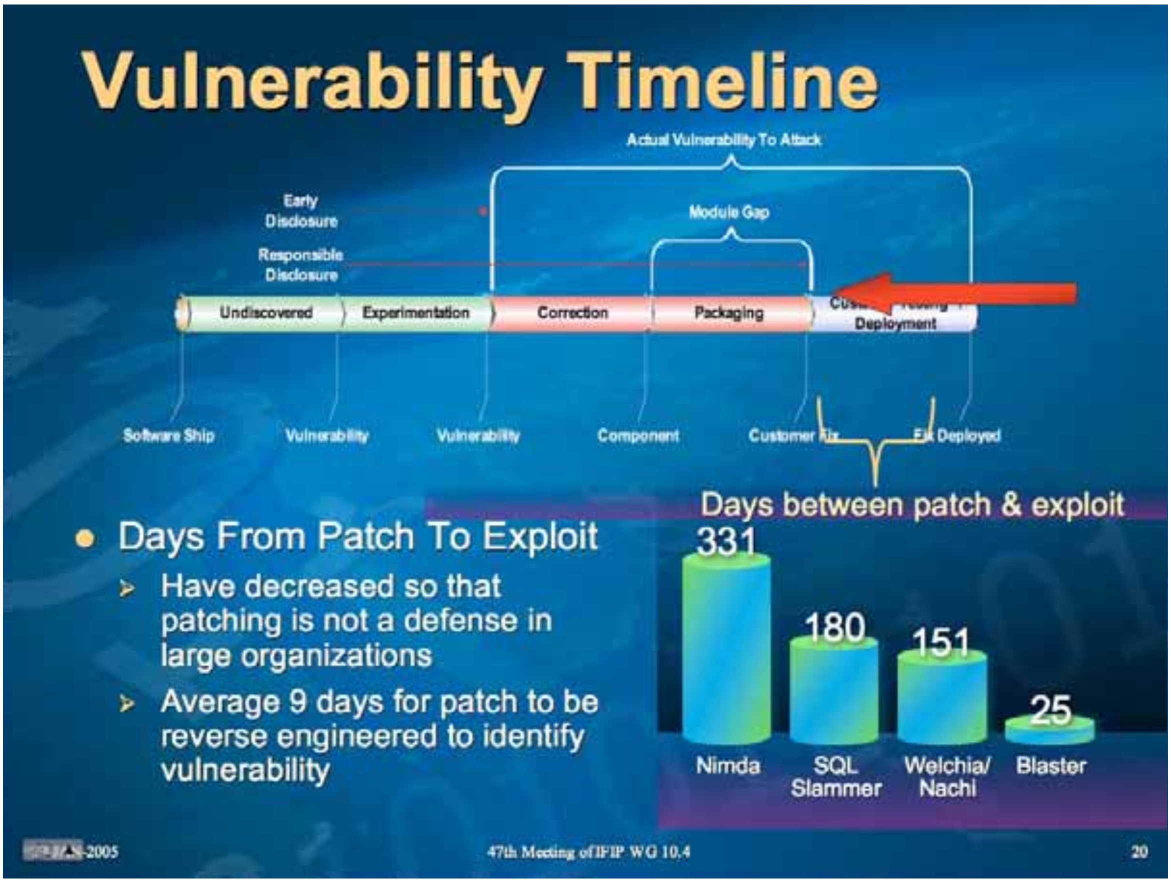
29-JAN-2005 47th Meeting of IFIP WG 10.4 17

Automatic Patching

- Windows Update services 190M PCs
- 140M PCs use Automatic Updates to stay current with patches
- Time to update 95% of XP PCs with a patch via Automatic Update
  - <14 days

29-JAN-2005 47th Meeting of IFIP WG 10.4 18


Vulnerability Timeline
Attacks occur here
Few discovered
Rarely discovered
Actual Vulnerability To Attack
Early Disclosure
Module Gap
Responsible Disclosure
Undiscovered
Experimentation
Correction
Packaging
Customer Testing / Deployment
Software Ship
Vulnerability
Vulnerability
Component
Customer Fix
Fix Deployed
29-JAN-2005
47th Meeting of IFIP WG 10.4
19


Vulnerability Timeline
Actual Vulnerability To Attack
Early Disclosure
Module Gap
Responsible Disclosure
Undiscovered
Experimentation
Correction
Packaging
Deployment
Software Ship
Vulnerability
Vulnerability
Component
Customer Fix
Fix Deployed
Days between patch & exploit
331
180
151
25
Nimda
SQL Slammer
Welchia/ Nachi
Blaster
Days From Patch To Exploit
Have decreased so that patching is not a defense in large organizations
Average 9 days for patch to be reverse engineered to identify vulnerability
47th Meeting of IFIP WG 10.4
20

Development Tools

- Source code defect detection tools
  - PREfix & PREfast (C/C++)
    - Detects defects like bounds violations, resource exhaustion, memory management errors, format string errors, etc.
  - FXCop (MSIL -- .NET managed code)
    - Detects defects in these categories: Library design, Localization, Naming conventions, Performance, Security
- Developers run versions of these tools before checking code into a product tree.
  - We also integrate the tools directly into the build process for automatic scans & bug reporting

Run-time Techniques

- Dynamic input scanning
  - Ex: URL filtering
- Middleware-based isolation
  - JVM, CLR, other host-based VMs
- OS virtualization
  - VMWare/Virtual PC/Xen
  - Hypervisors (IBM sHype, Intel VT)

Leveraging Customer Feedback

- MS Online Crash Analysis
  - Mechanism for reporting errors back to Microsoft, along with some debugging & tracing information (“minidumps”)
  - OCA reports are bucketed by application/module offset information
  - Minidump analysis identifies likely buffer overruns & other issues
  - Potential code defects automatically flagged for developer review

29-JAN-2005 47th Meeting of IFIP WG 10.4 23

Summary

- Attack frequency ↑
- Spyware ↑
- Vandalism → monetary objectives
- Patch reverse engineering time ↓

29-JAN-2005 47th Meeting of IFIP WG 10.4 24

Blatant Workshop Plug

- DIMACS Workshop on Security of Web Services & E-Commerce
  - May 5-6, 2005
  - DIMACS Center, Rutgers Univ. Piscataway, NJ
  - CFP deadline: February 11, 2005

http://dimacs.rutgers.edu/Workshops/Commerce/

29-JAN-2005 47th Meeting of IFIP WG 10.4 25

Questions?

29-JAN-2005 47th Meeting of IFIP WG 10.4 26

IFIP 10.4 Winter Meeting 2005

Security in Autonomic Web Computing

Bob Blakley
Chief Scientist, Security and Privacy, IBM
blakley@us.ibm.com

This Morning's Headline

- Lexus Landcruiser 100 models LX470 and LS430 have been discovered with virus-infected operating systems.
- It is understood the virus could affect the navigation system of the Lexus models
- It transfers onto them via a Bluetooth mobile phone connection.

Challenges

- Accountability
  - Driven by compliance mandates
- Availability
  - Driven by shift from “hard asset value” to “information value” to “process value”
- Privacy
  - Driven by customer perceptions

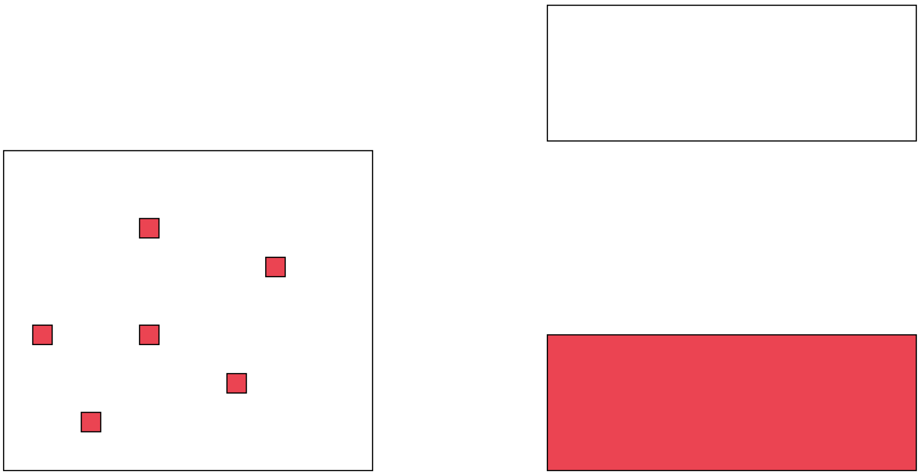
More Challenges

- Breakdown of the TCB
  - Where is the boundary?
  - Drives the requirement for vulnerabillity management
- Introductions
  - Identity of strangers
- Risk aggregation and Risk Diffusion
  - Single points of failure
  - No single point of incentive or responsibility

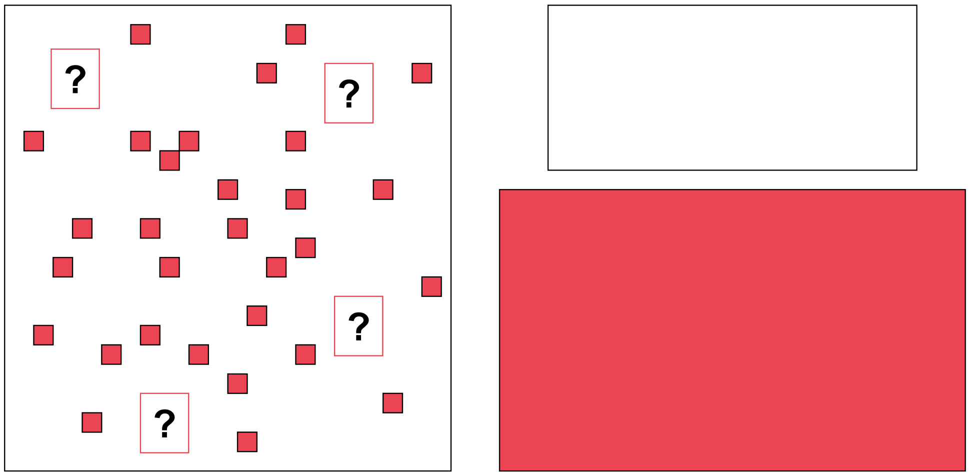
What’s Available?

- Traditional Security Technology
  - Wrong model, not well executed

TCB: Two Options



TCB: One Outcome



What's Available?

- Assurance
  - EAL 4 down are useful
  - But mainly improve documentation and catch obvious flaws
  - EAL 7 would be great…
- Tools
  - It's great that we're gradually phasing out the dumb stuff we've always known was bad for us

What's Available?

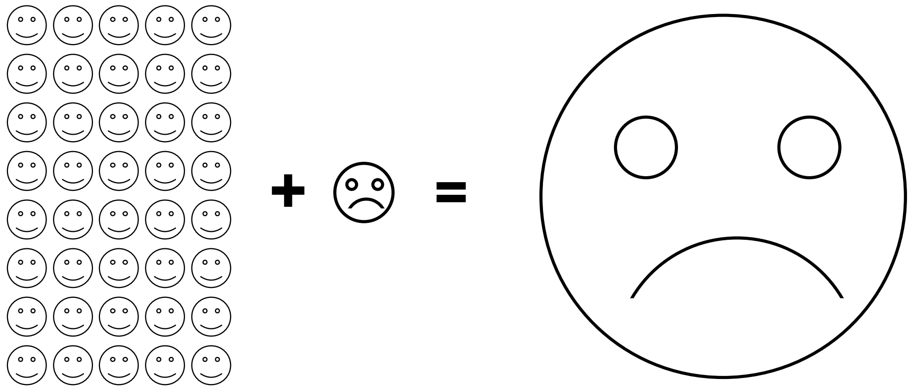
- Assurance
  - EAL 4 down are useful
  - But mainly improve documentation and catch obvious flaws
  - EAL 7 would be great…
- Tools
  - It's great that we're gradually phasing out the dumb stuff we've always known was bad for us
  - Like C++

What's Available?

- New Security Technology
  - Intrusion Detection, Antivirus,
  - Vulnerability Management
  - Kinda like sprinkler systems, these are great if you already *have* a fire and don't care about water damage…

Intrusion Detection

What detection?



Vulnerability Management

1,000,000 bugs
MBTF of each = 1,000,000,000 hours

Attacker has 1,000 hrs/yr available
Defender 100,000 hrs/yr plus expertise, source available

In 1 year, defender finds 100,000 bugs
Defender finds 1

Probability that defender finds attacker’s bug = 0.10

(Ross Anderson: Why Information Security is Hard)

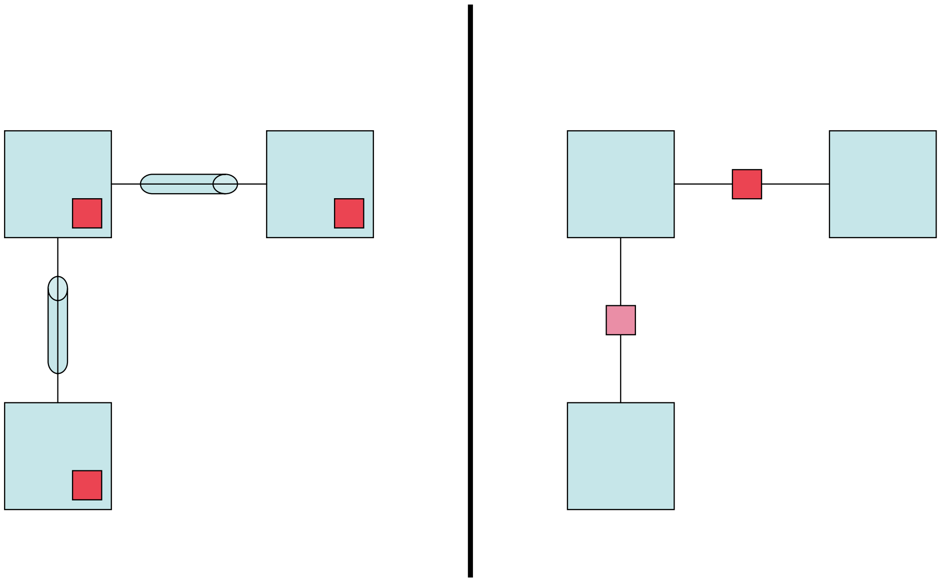
What’s Going To Happen?

- None of this stuff is going to work.
  - Traditional security technology assumes an infrastructure and an environment which don’t exist.
  - New security technologies lock the barn door after the horse is already gone.
  - Vulnerability management is a fool’s game.
- Periodic catastrophes will occur

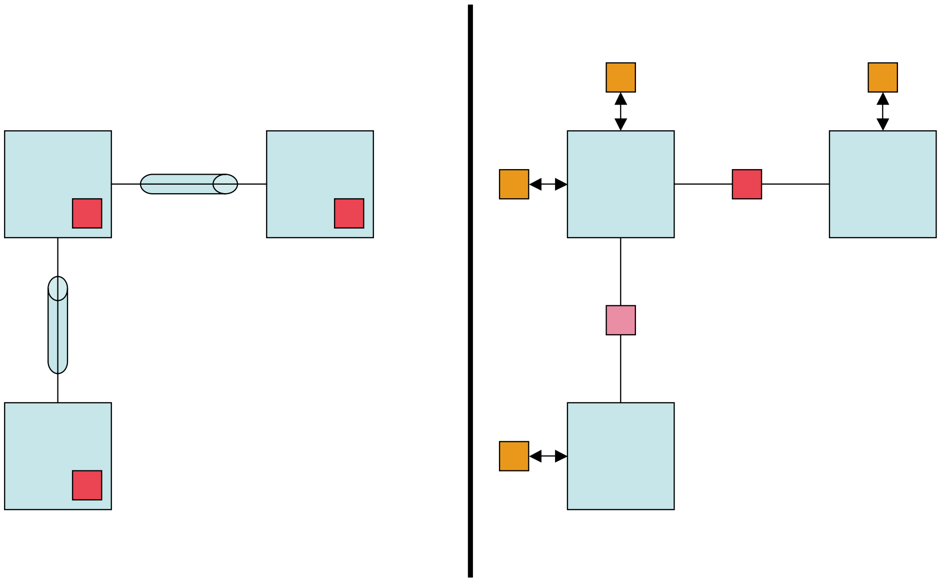
OK, What *Else* Is Available?

- Redundancy (hey, stuff is cheap now!)
- Diversity
- Use of time (need better way to say this…)
- Quick sense/analyze/respond loops
- Legislation/Regulation
  - HIPAA, GLB, etc…
  - Often diagnoses dyspepsia and prescribes leeches…
- New Models
  - Financial
  - Operational
  - Technical

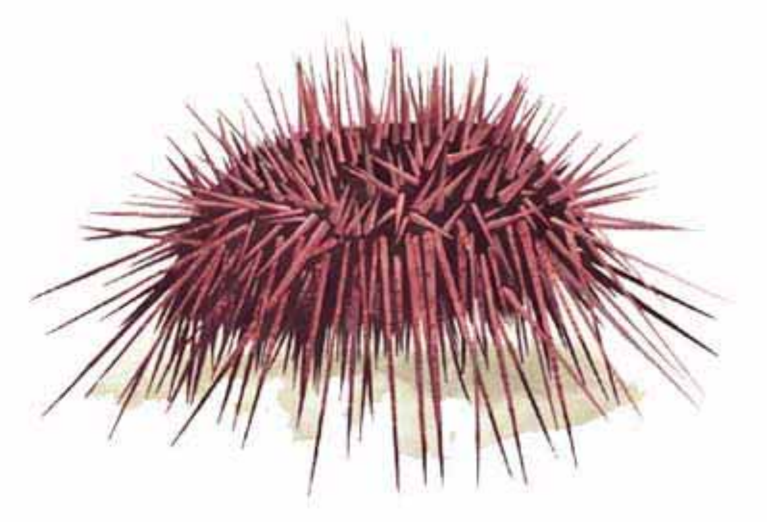
Externalizing Security



Security Services



Y'all Got Questions?



Backup (covered by Brian)

What's Out There?

- Hackers
  - Still lots
- Script Kiddies
  - Lots more
- Bots & Zombies
  - WAAAAY more
- Competitors
  - Hard to tell
- Terrorists
  - Definitely, but there are easier & more spectacular targets
- Nation-States
  - If you have to worry about these, you should be buying more specialized stuff

Why Is It Out There?

- Curiosity
- Fame (viruses)
- Fortune (trojans, spam, phishing)
- Malice (trojans)
  - Some people really hate Microsoft…
  - Which wouldn't be quite so bad if they'd attack Microsoft's servers instead of my client.

How Much Does It Cost?

- A lot
- But not as much as some folks want you to believe

How Bad Is It?

- Volume of attacks still doubles every year
- Time between discovery of vulnerability and release of automated exploit is asymptotically approaching zero
- Propagation of baddies is VERY fast
- Effectiveness of countermeasures against new exploits is pretty poor

Practical Cryptography and Autonomic Web Computing

John Black
University of Colorado, Boulder

IFIP WG 10.4
January 29, 2005
Rincón, Puerto Rico

Issues Exciting to *Theoretical* Cryptographers

- Primes 2 P? // yes [AKS02]
- (Extended) Riemann Hypothesis
- P = NP?
- Factoring $\cdot_p$ RSA?

2

Issues Exciting to *Practical* Cryptographers

- Key Distribution
- Factoring 2 P?
- Secure hashing
- Fast Crypto
- Crypto for Constrained Environments

3

Key Distribution

- Chicken and Egg
  - If we had a secure infrastructure, we could distribute keys securely
- Would solve a lot of major problems
  - ARP and DNS poisoning
  - SSH/SSL/IPSec
    - CA structure is far from ideal trust model
  - DDoS attacks
    - Though privacy types would protest if we traced every IP packet
    - Is the crypto fast enough for this (more later)

4

Factoring 2 P?

- Efficient factoring breaks RSA (and others)
- Twinkle
  - Spinning Mirrors
- Integer Factorization Circuits, TWIRL
  - 512-bit RSA modulus: 10 mins, $10K
  - 1024-bit modulus: < 1 yr, $10M
- Quantum Computers

5

Secure Hashing

- Important, useful objects
- Thin theoretical foundations
  - Blockcipher-based methods from 80's
  - Few proofs
- Differential attacks [Wang et al, 2004]
  - SHA-0, MD5, and others "broken"
- SHA-1 appears safe still [Rijmen05]
  - Can break 53-round SHA-1 with < 2^{80} work

6

SHA-1

512 bits

To obtain SHA-0, remove this

M_1	M_2	...	M_m

for i = 1 **to** m **do**

$$W_t = \begin{cases} t\text{-th word of } M_i & 0 \le t \le 15 \\ (\mathrm{W}_{t-3} \text{ © } \mathrm{W}_{t-8} \text{ © } \mathrm{W}_{t-14} \text{ © } \mathrm{W}_{t-16}) \ll 1 & 16 \le t \le 79 \end{cases}$$

$$A \leftarrow H_0^{i-1}; \quad B \leftarrow H_1^{i-1}; \quad C \leftarrow H_2^{i-1}; \quad D \leftarrow H_3^{i-1}; \quad E \leftarrow H_4^{i-1}$$

for t = 1 **to** 80 **do**

$$T \leftarrow A \ll 5 + g_t(B, C, D) + E + K_t + W_t$$

$$E \leftarrow D; \quad D \leftarrow C; \quad C \leftarrow B \gg 2; \quad B \leftarrow A; \quad A \leftarrow T$$

end

$$H_0^i \leftarrow A + H_0^{i-1}; \quad H_1^i \leftarrow B + H_1^{i-1}; \quad H_2^i \leftarrow C + H_2^{i-1};$$

$$H_3^i \leftarrow D + H_3^{i-1}; \quad H_4^i \leftarrow E + H_4^{i-1}$$

end

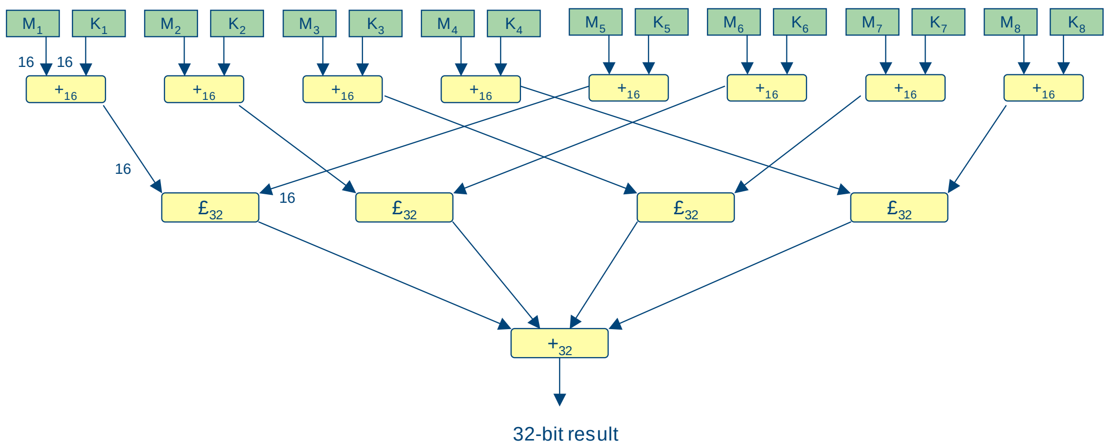
return $H_0^m \ H_1^m \ H_2^m \ H_3^m \ H_4^m$ ← 160 bits

Fast Parallelizable Crypto

- Slow and serial crypto is an impediment
  - High-end web servers
  - Routers
- Recent research has sought to find fast (and parallelizable, often) algorithms
  - AES is much faster than DES was
  - HMAC, UMAC, Poly1305
  - Many AE schemes (OCB, CWC, EAX, etc)
- Proof-of-correctness now a requirement
  - Some are skeptical about the value of this, but none suggests it's better than no proof at all

8

The Heart of the UMAC algorithm



The above represents **three** MMX instructions (2 **`paddw`**'s and a **`pmaddwd`**)

Crypto in Constrained Environments

- We can do standard crypto on a laptop
  - But a cell phone has a lot fewer cycles to spare
    - Indeed, they've blown it a few times already
  - Sensor nodes have ever fewer (and radio constraints)
  - RFIDs present an extreme challenge
- We need simple algorithms, even if they don't provide industrial-strength cryptography
  - TinySec [KSW04] is a start

10

And What Virtually No Cryptographers Find Exciting…

- Software Engineering and Education
  - In my opinion, where a lot of security problems start
- Software Engineering:
  - Security was not “built in” from the start
    - More examples than non-examples
  - Software not built according to “best practices”
    - Every vulnerability is a bug, so security is really a *quality* problem
    - Code is not agile, so when something breaks (eg, PKCS #1) it’s hard to plug in something new

11

Education

- Students emerge with a degree in Computer Science with little to no training in security
  - Not a standard part of most curricula
  - Not enough knowledgeable people available to train students
- On the crypto side, two important themes
  - 1) Don’t create your own crypto; you’ll get it wrong
    - Example: Internet Chess Club
  - 2) Perfectly good crypto primitives get misused and are rendered worthless
    - Example: MS Office

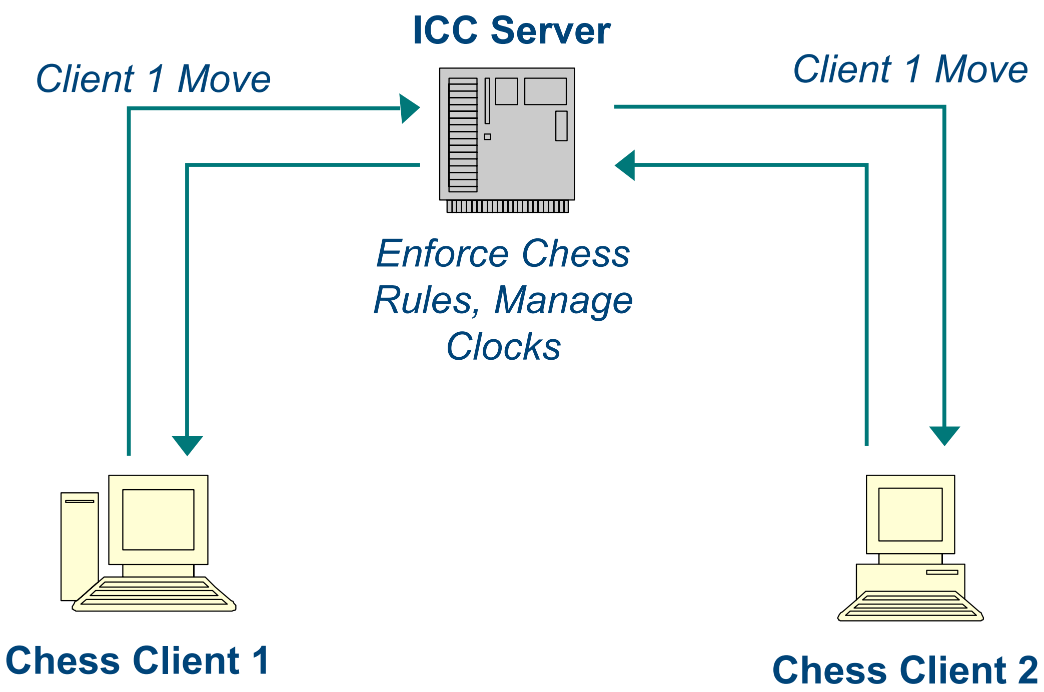
12

Internet Chess Club

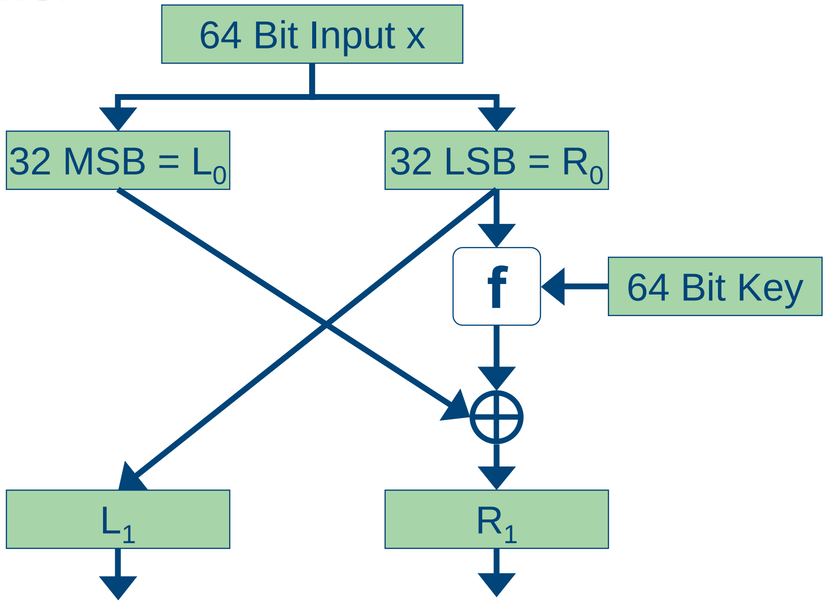
- Over 30,000 members
- Pay Site ($60/year)
- Madonna, Nicholas Cage, Will Smith, Sting, even Kasparov
- Best choice for online chess
- Written by and run by a CMU CS Professor
  - Specializes in theory, interested in cryptography

13

Basic Idea



Block Cipher



16 Rounds

$$f(V,K,r) = S[V_0 + V_1 + K_{r \bmod 8} \bmod 256] + V^2$$

Mode of Operation

- Pad formed by XOR of two LCGs

$$x_{n+1} = 3x_n + 1 \bmod 43060573$$

$$y_{n+1} = 17y_n \bmod 2413871$$

$$\text{pad} = x_n \oplus y_n$$ *(just low byte)*

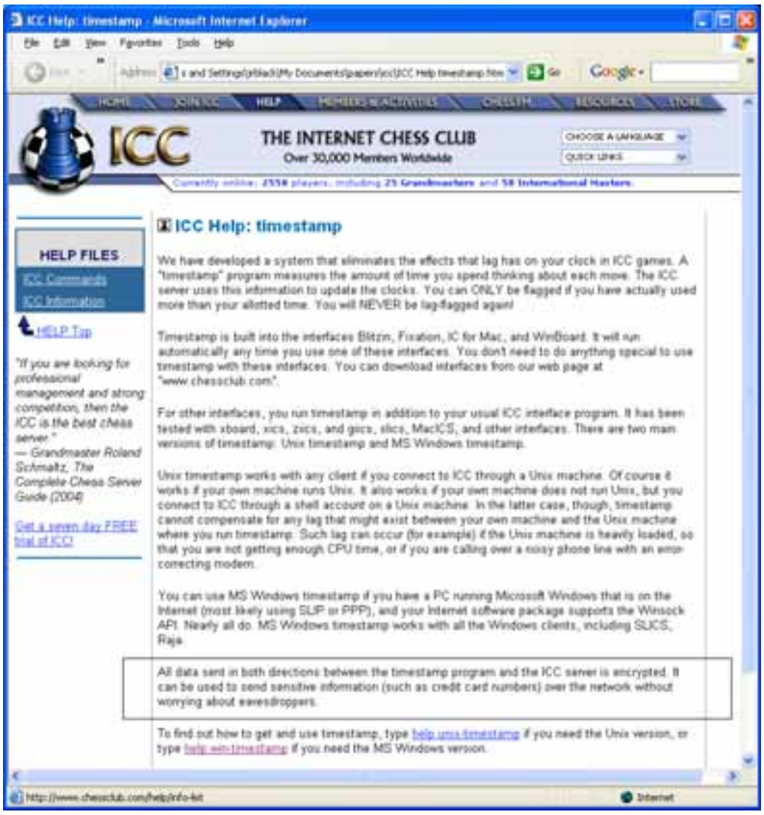
- Given 10 pad bytes, we get the rest
- 1.1 secs on Martin's laptop

16

Key Exchange

- Seeds for symmetric keys exchanged ***in the clear!!!***
- We sniff the connection (pcap) and read all the traffic trivially
  - Get CC #s
  - Get usernames and passwords
- Active attacks would be even MORE damaging

17

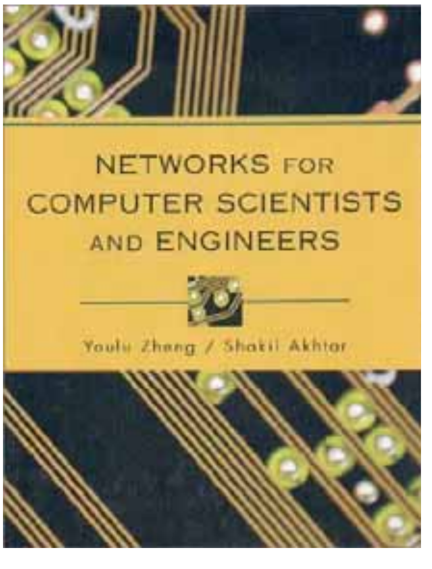


MS Office

- Office 95
  - Just xor'ed password with plaintext over and over
- Later RC4 employed, but exportable versions forced to use 40-bit key
  - Easily broken by brute-force
- Office 2000, XP, 2003 use 128-bit RC4
  - But use the same IV (seed) each time

19

Closing Example: Pedagogy



20

10.2 DIGITAL CERTIFICATE AND PUBLIC KEY INFRASTRUCTURE (PKI) 461

10.1.8 Write Your Own Encryption Algorithm

People are often discouraged from writing a personal encryption algorithm because of a fear that a small bug in the code will render their decrypted messages meaningless. On the other hand, trusting the security of your transmissions to "experts" can also be a questionable practice.

If you follow the principles outlined here, writing your own encryption system should be easy. For practice, the laboratory manual (part of the Instructor's manual and CD accompanying the book) provides an encryption program written in X86 assembler code. The program incorporates several encryption steps to produce a multiple product cipher and chooses steps that are aimed at thwarting various attack methods. Here are the steps contained in the sample program and some suggestions for designing an encryption system:

Encryption Algorithm (Cont) [ZA 2002]

9. Every so often, change the order of the steps in the algorithm.

10. Insert some random snow, especially at the start.

14. Make sure that changing even a single bit in the key or in the ciphertext will produce garbage.

15. Insert some useful garbage, such as a dummy message, and rescramble the whole thing with a simple, eventually breakable message.

Moral

- Security Education is sorely inadequate
- Even if we did more, there would still be vulnerabilities, but it wouldn't be nearly this bad

23

A flexible access control model for web services

Elisa Bertino
CERIAS and CS&ECE Departments
Purdue University

Outline

- Motivations
- Overview of Ws-Attribute Based Access control (Ws-ABA)
- Underlying technologies
  - Digital identity management
  - Trust negotiation system
- Access control model
- System architecture
- Conclusions and future work

Web Services

- A Web service is a Web-Based application that can be
  - Published
  - Located
  - Invoked
- Compared to centralized systems and client-server environments, a Web service is much more *dynamic* and *security* for such an environment poses unique challenges

Promises of Web Services

- Interoperability across lines of business and enterprises
  - Regardless of platform, programming language and operating system
- End-to-end exchange of data
  - Without custom integration
- Loosely-coupled integration across applications
  - Using Simple Object Access Protocol (SOAP) and XML

Why HTTPS Is not Enough for Web Services

- HTTPS is protocol-level security
  - Point-to-point: lasts only for the duration of the connection
  - Does not secure solutions that use other protocols
  - "All or nothing" encryption only
  - Does not support other security mechanisms

Building Blocks for Web Service Security

- XML Encryption
  - Encrypt all or parts of an XML message
  - Separation of encryption information from encrypted data
- XML Signature
  - Apply to all or parts of a document
  - Facilitates production of composite documents while preserving the signature
  - Multiple signature with different characteristics over the same content
- SAML
  - XML format for exchanging authentication, authorization, and attribute assertions
- WS-Security
  - Originally defined by Microsoft, IBM, and Verisign
  - It defines how to attach signature, encryption, and security tokens to SOAP messages

Web Services: Access Control

An important issue is represented by the development of suitable access control models, able to restrict access to Web services to authorized users.

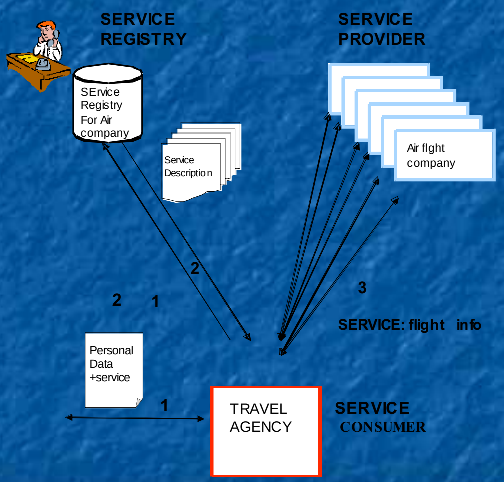
Web services are quite different with respect to objects typically protected in conventional systems, since they consist of software modules, to be executed, upon service requests, according to a set of associated input parameters.

security technologies commonly adopted for Web sites and traditional access control models are not enough!

An Important Requirement: to be Policy-based

- A *policy* is a set of capabilities, requirements, preferences and general characteristics about entities in a system
- The elements of a policy (*policy assertions*) can express:
  - Security requirements or capabilities
  - Various Quality of Service (QoS) characteristics

An Example



- Suppose to have a travel agency selling flight tickets to generic customers offering a service, whose goal is to offer competitive flight tickets fare to requesting customers.
- As sketched (arrow 1), a customer request is sent by including also a set of attributes describing relevant properties of the customer and his/her preference or needs, to customize service release.
- The agency, in turn, forwards customer requests to flight companies.

Ws - Attribute Based Access Control

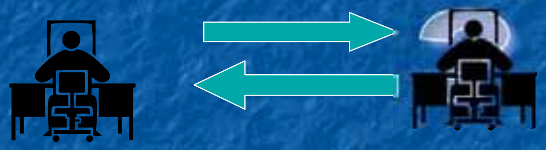
- Implementation independent access control model for Web services, for use within the SOAP standard, characterized by capabilities for negotiating service parameters
- The goal of *Ws-Aba*, is to express, validate and enforce application-based access control policies without assuming pre-established trust in the users of web services

Underlying Technologies
Digital Identity Management

- What is digital identity?
  - *Digital identity can be defined as the digital representation of the information known about a specific individual or organization*
- The term *DI* usually refers to two different concepts:
  - *Nym* – a nym gives a user an identity under which to operate when interacting with other parties. Nyms can be strongly bound to a physical identity
  - *Partial identity* – partially identities refer to the set of properties that can be associated with an individual, such as name, birth-date, credit cards. Any subset of such properties represents a partial identity of the user

Underlying Technologies
Trust Negotiation

- **Interactions between strangers**
  - In conventional systems user identity is known in advance and can be used for performing access control
  - In open systems partecipants may have no pre-existing relationship and may not share a common security domain



- **Mutual authentication**
  - Assumption on the counterpart honesty no longer holds
  - Both participants need to authenticate each other

Underlying Technologies Trust Negotiation

- A promising approach for open systems where most of the interactions occur between strangers
- The **goal**: establish trust between parties in order to exchange sensitive information and services
- The **approach**: establish trust by verifying **properties** of the other party

Ws-Aba access control model

- Access conditions
  - expressed in terms of *partial identities*
  - take into account also the parameters characterizing web services
- Concept of *access negotiation*
  - Web service negotiation in Ws-Aba deals with the possibility for trusted users to dynamically change their access requests in order to obtain authorizations

Ws-Aba access control policies

- An access control policy is defined by three elements:
  - A service identifier
  - A set of parameter specifications
    - A parameter specification is a pair
      - Parameter-name, parameter-value-range
  - A set of conditions against partial identities
- A WS-policy specification of our policy language has been developed

Ws-Aba access control policies examples

- Policy Pol1
  - (FlightRes; Discount[0,30]; Age > 65)
  - It authorizes subjects older than 65 to reserve a flight with a discount up to 30%;
- Policy Pol2
  - (FlightRes;{Fare [Standard, Gold], Discount[0,50]}; {Partnernship=TravelCorporation, Seniority >3, Age>65})
  - It authorizes subjects that are older than 65 and have a 3 year seniority and have a partnership with TravelCorporation to get a fare between standard and gold and a discount up to 50%

Ws-Aba: how it works

- Access requests are received
  - ✓ specified by constraining service parameters, and subject partial identities
  - ✓ Note: a subject before releasing partial identity information may require to establish trust by using trust negotiation
- The system extracts the corresponding access control policies, in order to establish whether the subject request can be:
  - ✓ accepted as it is
  - ✓ must be rejected
  - ✓ has to be negotiated
- A request negotiation results in eliminating and/or modifying some of the service parameters specified within an access request that made it not immediately acceptable

Access responses in Ws-Aba

- Upon an access request three replies are possible:
  - The submitted attributes match with a policy for the specified service request and the specified service parameters are acceptable by the policy

    Request is granted
  - The submitted attributes do not match with any policy for the specified service request

    Request is rejected
  - The submitted attributes match with a policy for the specified service request but the specified service parameters are not acceptable by the policy

Access responses in Ws-Aba - example

- Policy Pol1 - (FlightRes; Discount[0,30]; Age > 65)
- Policy Pol2 - (FlightRes;{Fare [Standard, Gold]; Discount[0,50]}; {Partnernship=TravelCorporation, Seniority >3, Age>65})
- Requests:
  - <[Partnership:TravelCorporation, Seniority:5, Age:70]; FlightRes; [Fare:Gold, Discount:30]>
    - It complies with Pol2 and can be fully accepted
  - <[Age:70; FlightRes; [Discount:50]>
    - It complies with Pol1; however it must be negotiated since the parameter value is outside the range specified in Pol1
  - <[University:Milano; FlightRes; [Discount:30]>
    - It is rejected since it does not match the subject specification of any policy

Certificates supported

- WS-Aba accepts SOAP messages for service invocation
- To promote interoperability and flexibility we do not restrict our system to a specific implementation, we adopt a specific proposal to connect our system to the PKC infrastructure: X.509 AC

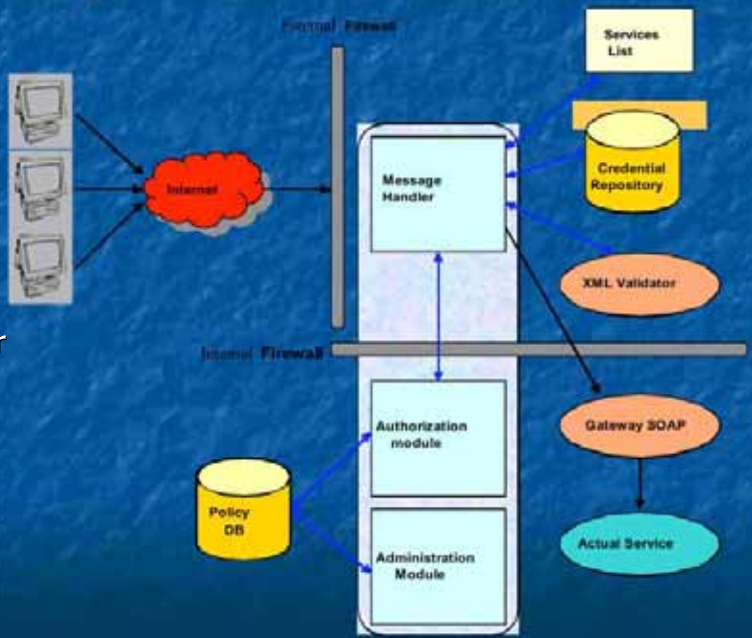
Identity and attributes: X.509 AC

X.509 AC provides a binding between attributes and an identity. It is composed of two nested elements: the former describing the conveyed information, that is, the AttributeCertificateInfo element and the Signature element, carrying the signature

```
<element name="Attributes" type="ac:AttributesType"/>
<complexType name="AttributesType">
   <sequence>
      <element ref="ac:ServiceAuthenticationInformation" minOccurs="0"/>
      <element ref="ac:AccessIdentity" minOccurs="0"/>
      <element ref="ac:ChargingIdentity" minOccurs="0"/>
      <element ref="ac:Group" minOccurs="0"/>
      <element ref="ac:Role" minOccurs="0"/>
      <element ref="ac:Clearance" minOccurs="0"/>
      <element ref="ac:GenericAttribute" minOccurs="0" maxOccurs="unbounded"/>
   </sequence>
   <attribute name="Id" type="ID" use="optional"/>
</complexType>
```

WS- Aba System Architecture

- Three main modules:
  - Message Handler
  - Authorization module
  - Authorization management



Open issues

- Policy selection:
  - If a request complies with several policies, how do we choose a policy to apply?
- Negotiation of parameters:
  - How can subjects negotiate service parameters?
- Delegation:
  - How to manage delegated access requests?
- Cached policies:
  - How and where keep track of previous access requests?
- Policy protection:
  - How to protect UDDI registries where AC policies are stored?

Future work

- Delegation mechanisms for credentials
- Automated mechanisms supporting negotiations of parameters
- Automated mechanisms for policy configurations – for making policies active or passive depending on specific events and context conditions
- Granularity levels of policies: policies that apply to group of services
- Authorization derivation rules, allowing authorizations on a service to be automatically other services

Web Services Security Configuration Challenges

Sanjai Narain
Senior Research Scientist
Telcordia Technologies
narain@research.telcordia.com
(732) 699 2806

Prepared for IFIP WG 10.4, January 26-30
Rincon, PR

Deploying Web Services Security Infrastructure

Challenge is assembling building blocks to satisfy end-to-end requirements on security *and* availability

Some problems

How to precisely specify this plan? Constraints are *global*.

How to reconcile constraints, and synthesize component configurations?

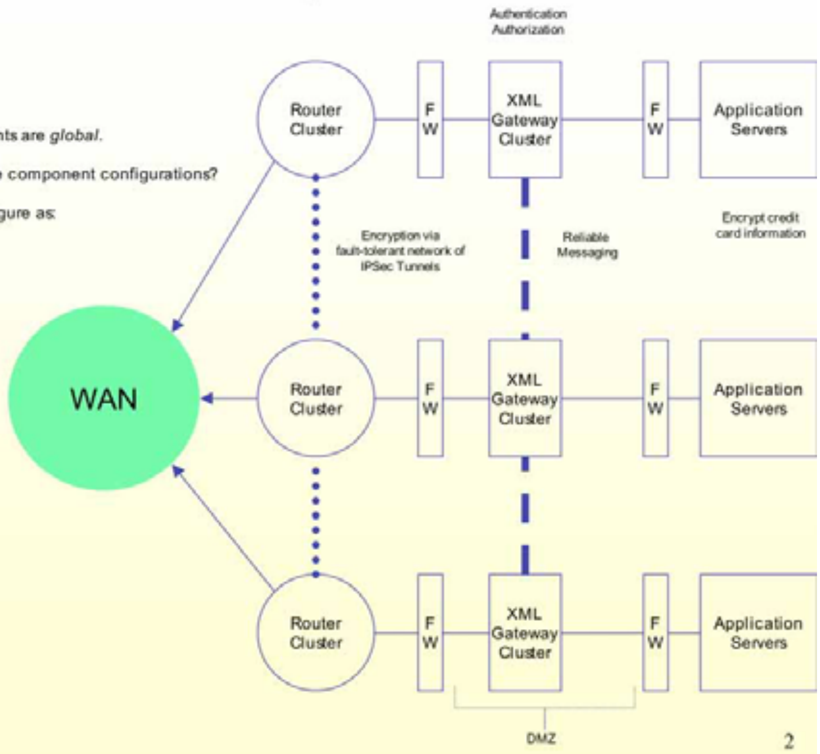
New site needs to be added. How to reconfigure as:
- Requirements change?
- New site is added or deleted?

How to reason about this?
- How much defense in depth is there?
- Are there single points of failure?

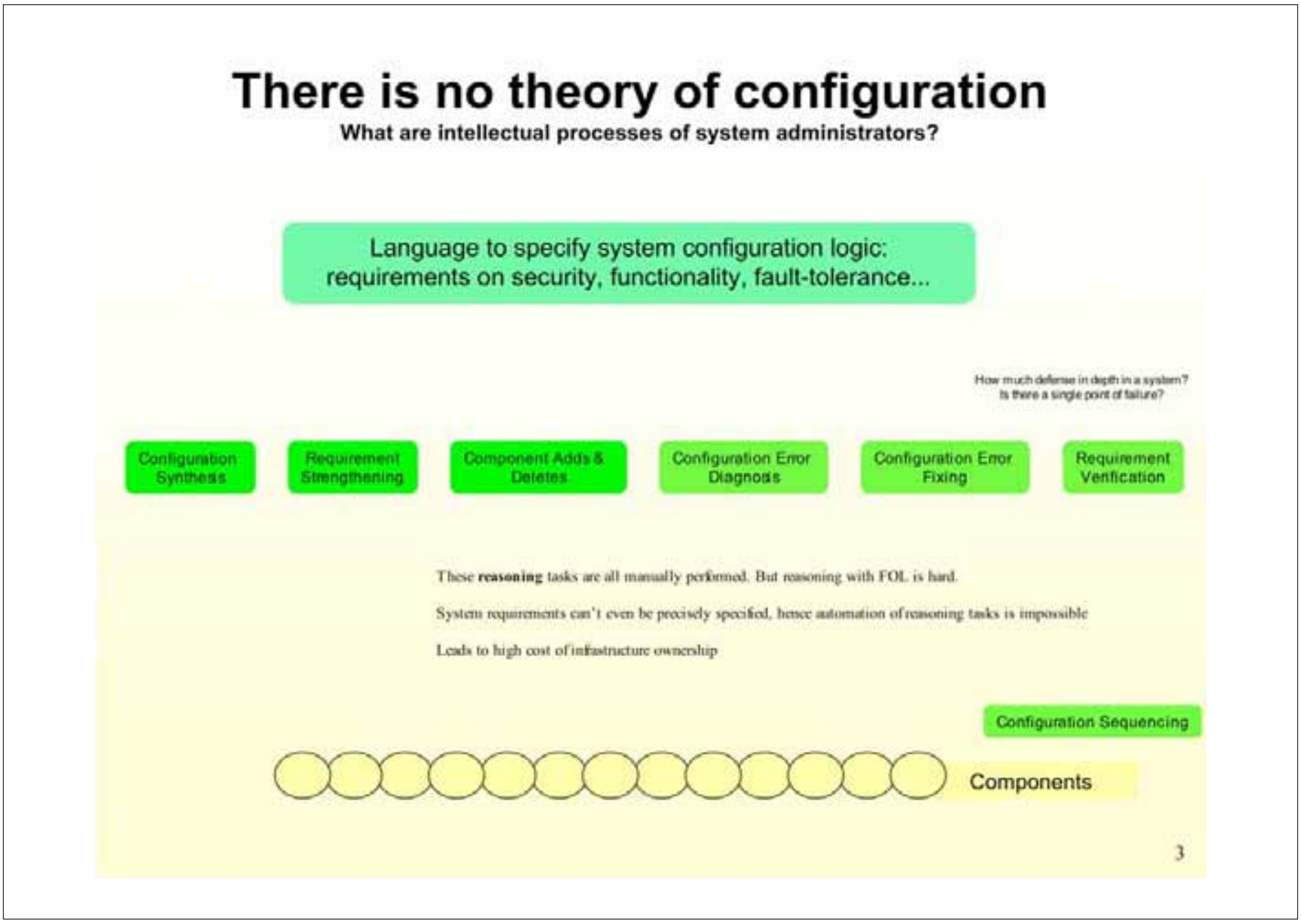
How to diagnose configuration errors?

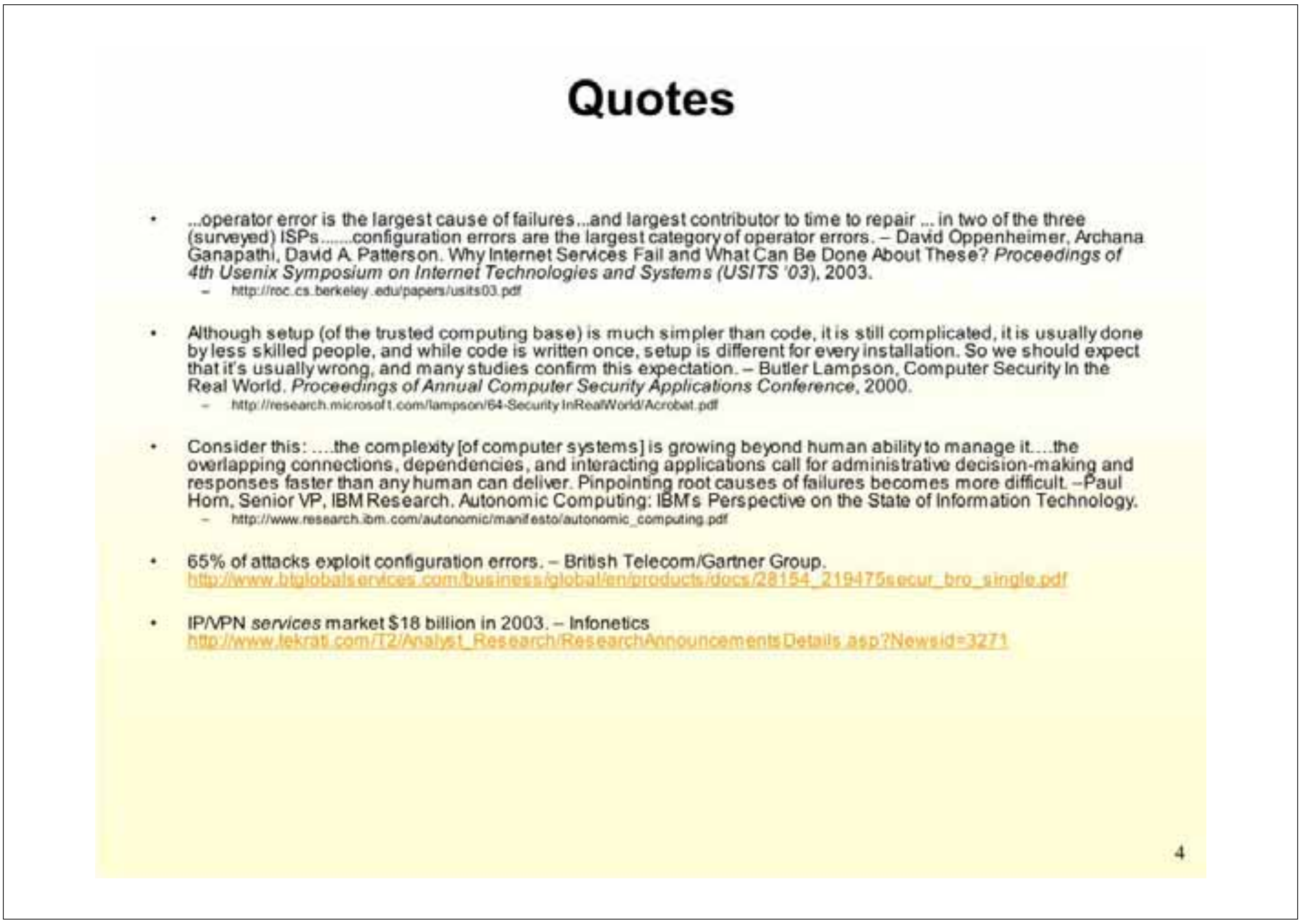
How to troubleshoot these?

How to sequence configurations?

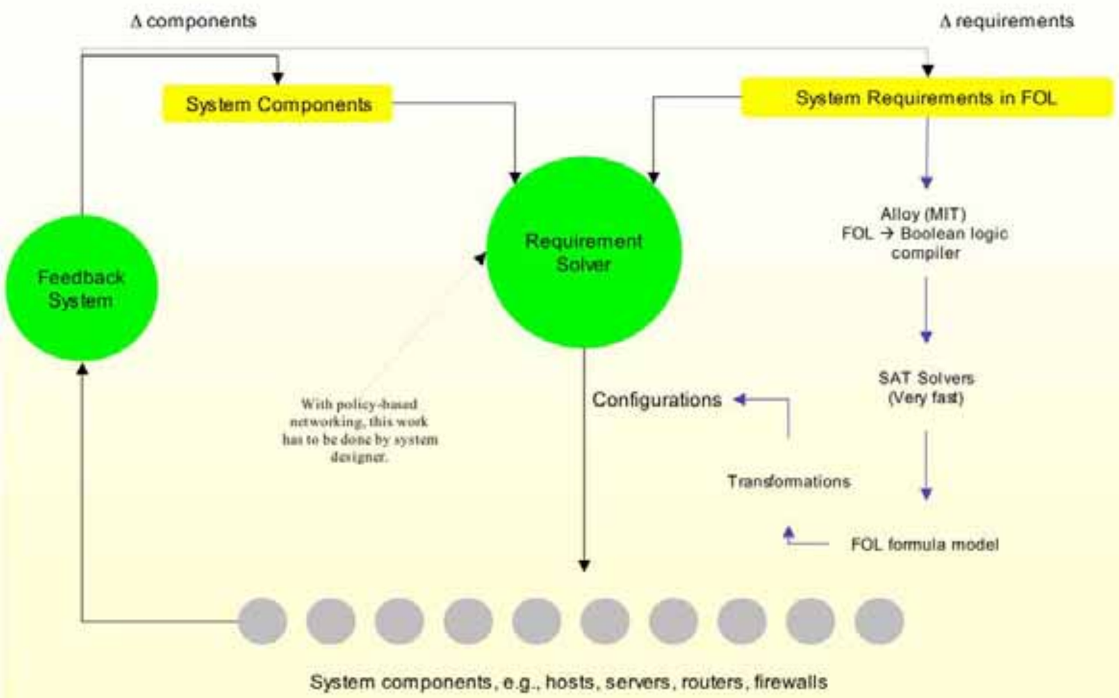


2


There is no theory of configuration
What are intellectual processes of system administrators?
Language to specify system configuration logic:
requirements on security, functionality, fault-tolerance...
How much defense in depth in a system?
Is there a single point of failure?
Configuration Synthesis
Requirement Strengthening
Component Adds & Deletes
Configuration Error Diagnosis
Configuration Error Fixing
Requirement Verification
These **reasoning** tasks are all manually performed. But reasoning with FOL is hard.
System requirements can't even be precisely specified, hence automation of reasoning tasks is impossible
Leads to high cost of infrastructure ownership
Configuration Sequencing
Components
3


Quotes
• ...operator error is the largest cause of failures...and largest contributor to time to repair ... in two of the three (surveyed) ISPs.......configuration errors are the largest category of operator errors. – David Oppenheimer, Archana Ganapathi, David A. Patterson. Why Internet Services Fail and What Can Be Done About These? *Proceedings of 4th Usenix Symposium on Internet Technologies and Systems (USITS '03)*, 2003.
– http://roc.cs.berkeley.edu/papers/usits03.pdf
• Although setup (of the trusted computing base) is much simpler than code, it is still complicated, it is usually done by less skilled people, and while code is written once, setup is different for every installation. So we should expect that it's usually wrong, and many studies confirm this expectation. – Butler Lampson, Computer Security In the Real World. *Proceedings of Annual Computer Security Applications Conference*, 2000.
– http://research.microsoft.com/lampson/64-SecurityInRealWorld/Acrobat.pdf
• Consider this: ....the complexity [of computer systems] is growing beyond human ability to manage it.....the overlapping connections, dependencies, and interacting applications call for administrative decision-making and responses faster than any human can deliver. Pinpointing root causes of failures becomes more difficult. –Paul Horn, Senior VP, IBM Research. Autonomic Computing: IBM's Perspective on the State of Information Technology.
– http://www.research.ibm.com/autonomic/manifesto/autonomic_computing.pdf
• 65% of attacks exploit configuration errors. – British Telecom/Gartner Group.
http://www.btglobalservices.com/business/global/en/products/docs/28154_219475secur_bro_single.pdf
• IP/VPN *services* market $18 billion in 2003. – Infonetics
http://www.tekrati.com/T2/Analyst_Research/ResearchAnnouncementsDetails.asp?Newsid=3271
4

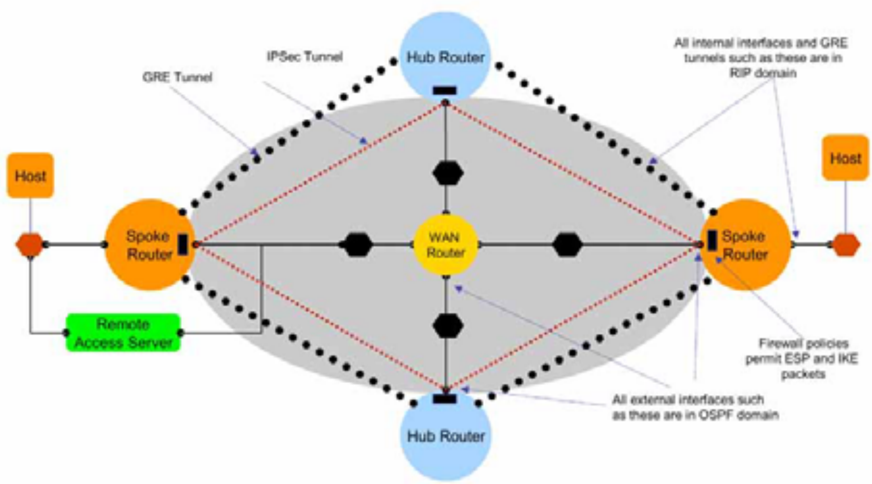
New Concept: Requirement Solver



5

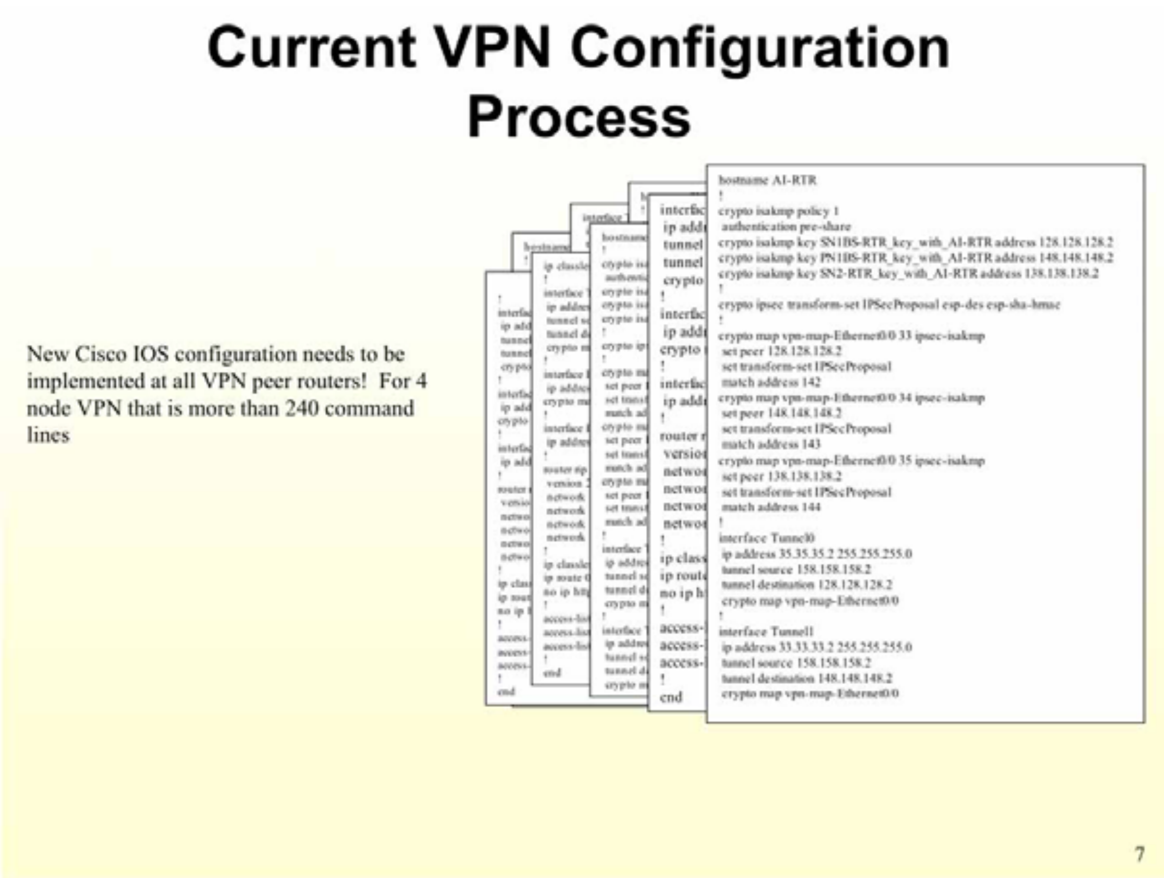
Fault-Tolerant VPN

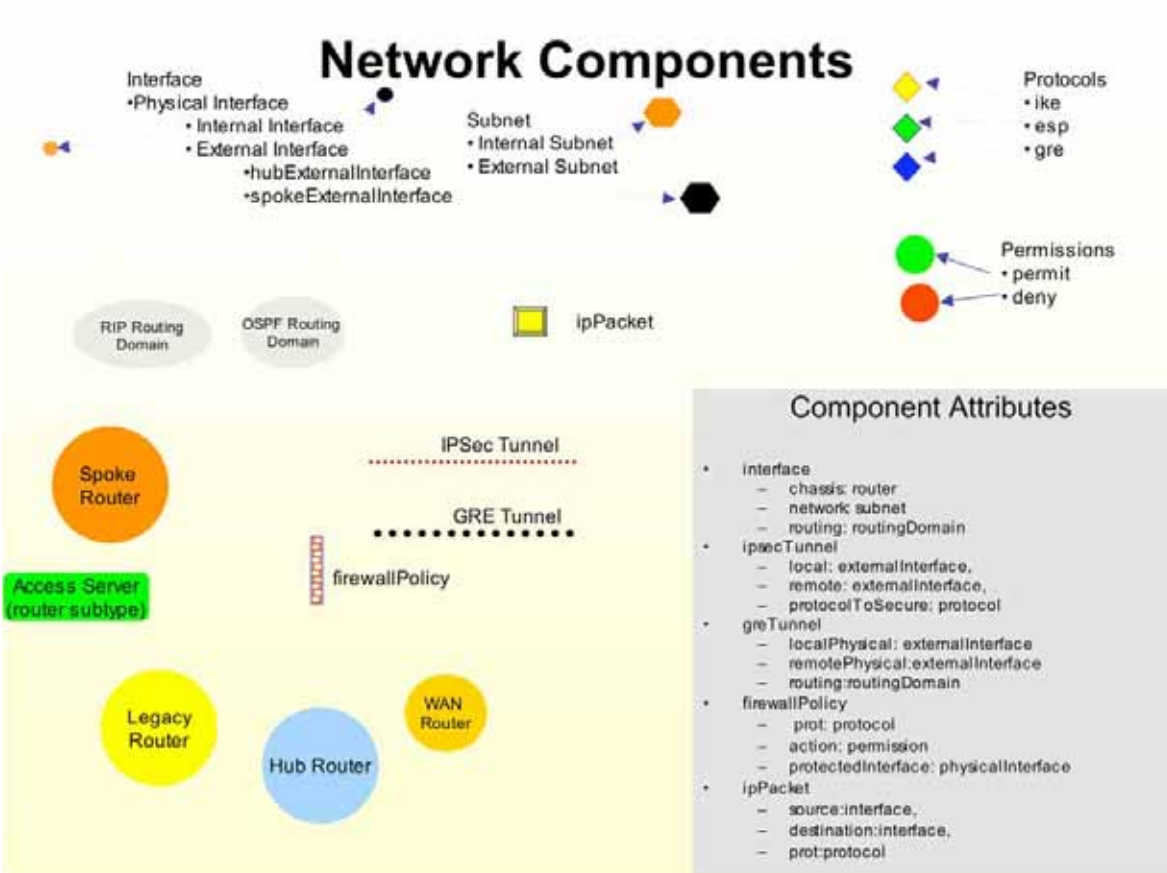
Illustrates composition of FT systems into larger FT system



- Full mesh of IPSec tunnels does *not* scale
- Linearly-scaling solution can have single point of failure

6


Current VPN Configuration Process
New Cisco IOS configuration needs to be implemented at all VPN peer routers! For 4 node VPN that is more than 240 command lines
hostname AI-RTR
!
crypto isakmp policy 1
authentication pre-share
crypto isakmp key SN1BS-RTR_key_with_AI-RTR address 128.128.128.2
crypto isakmp key PN1BS-RTR_key_with_AI-RTR address 148.148.148.2
crypto isakmp key SN2-RTR_key_with_AI-RTR address 138.138.138.2
!
crypto ipsec transform-set IPSecProposal esp-des esp-sha-hmac
!
crypto map vpn-map-Ethernet0/0 33 ipsec-isakmp
set peer 128.128.128.2
set transform-set IPSecProposal
match address 142
crypto map vpn-map-Ethernet0/0 34 ipsec-isakmp
set peer 148.148.148.2
set transform-set IPSecProposal
match address 143
crypto map vpn-map-Ethernet0/0 35 ipsec-isakmp
set peer 138.138.138.2
set transform-set IPSecProposal
match address 144
!
interface Tunnel0
ip address 35.35.35.2 255.255.255.0
tunnel source 158.158.158.2
tunnel destination 128.128.128.2
crypto map vpn-map-Ethernet0/0
!
interface Tunnel1
ip address 33.33.33.2 255.255.255.0
tunnel source 158.158.158.2
tunnel destination 148.148.148.2
crypto map vpn-map-Ethernet0/0
7


Network Components
Interface
•Physical Interface
• Internal Interface
• External Interface
•hubExternalInterface
•spokeExternalInterface
Subnet
• Internal Subnet
• External Subnet
Protocols
• ike
• esp
• gre
Permissions
• permit
• deny
RIP Routing Domain
OSPF Routing Domain
ipPacket
IPSec Tunnel
GRE Tunnel
firewallPolicy
Spoke Router
Access Server (router subtype)
Legacy Router
Hub Router
WAN Router
Component Attributes
• interface
– chassis: router
– network: subnet
– routing: routingDomain
• ipsecTunnel
– local: externalInterface,
– remote: externalInterface,
– protocolToSecure: protocol
• greTunnel
– localPhysical: externalInterface
– remotePhysical:externalInterface
– routing:routingDomain
• firewallPolicy
– prot: protocol
– action: permission
– protectedInterface: physicalInterface
• ipPacket
– source:interface,
– destination:interface,
– prot:protocol

List of Network Requirements

RouterInterfaceRequirements

1. Each spoke router has internal and external interfaces
2. Each access server has internal and external interfaces
3. Each hub router has only external interfaces
4. Each WAN router has only external interfaces

SubnettingRequirements

5. A router does not have more than one interface on a subnet
6. All internal interfaces are on internal subnets
7. All external interfaces are on external subnets
8. Every hub and spoke router is connected to a WAN router
9. No two non-WAN routers share a subnet

RoutingRequirements

10. RIP is enabled on all internal interfaces
11. OSPF is enabled on all external interfaces

GRERequirements

12. There is a GRE tunnel between each hub and spoke router
13. RIP is enabled on all GRE interfaces

SecureGRERequirements

14. For every GRE tunnel there is an IPSec tunnel between associated physical interfaces that secures all GRE traffic

AccessServerRequirements

15. There exists an access server and spoke router such that the server is attached in "parallel" to the router

FirewallPolicyRequirements

16. Each hub and spoke external interface permits esp and ike packets

Human administrators reason with these in different ways to synthesize initial network, then reconfigure it as operating conditions change.

Can we automate this reasoning?

9

Configuration Synthesis: Physical Connectivity and Routing

RouterInterfaceRequirements

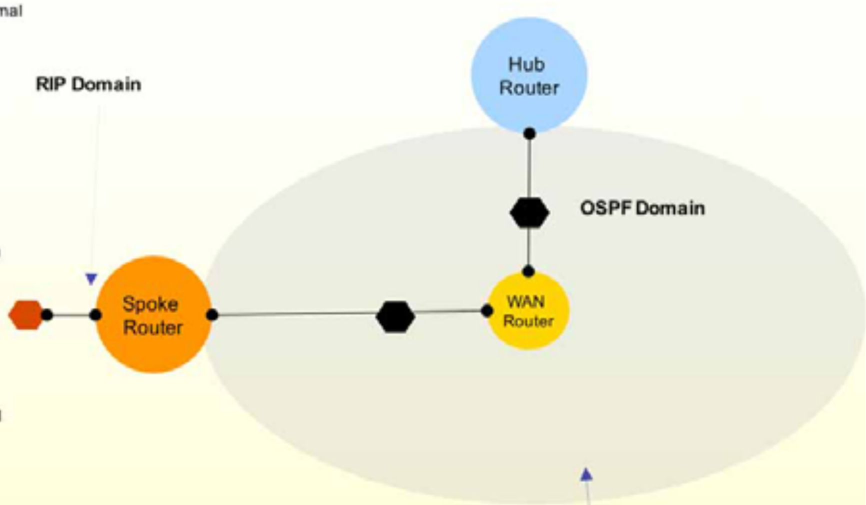
1. Each spoke router has internal and external interfaces
2. Each access server has internal and external interfaces
3. Each hub router has only external interfaces
4. Each WAN router has only external interfaces

SubnettingRequirements

5. A router does not have more than one interface on a subnet
6. All internal interfaces are on internal subnets
7. All external interfaces are on external subnets
8. Every hub and spoke router is connected to a WAN router
9. No two non-WAN routers share a subnet

RoutingRequirements

10. RIP is enabled on all internal interfaces
11. OSPF is enabled on all external interfaces



- To synthesize network, satisfy R1-R11 for
  - 1 hub router,
  - 1 WAN router,
  - 1 spoke router,
  - 1 internal subnet,
  - 2 external subnets
  - 1 internal interface,
  - 4 external interfaces,
  - RIP domain,
  - 1 OSPF domain

Requirement Solver generates solution. Note that Hub and Spoke routers are not directly connected, due to Requirement 9

10

Strengthening Requirement: Adding Overlay Network

RouterInterfaceRequirements

1. Each spoke router has internal and external interfaces
2. Each access server has internal and external interfaces
3. Each hub router has only external interfaces
4. Each WAN router has only external interfaces

SubnettingRequirements

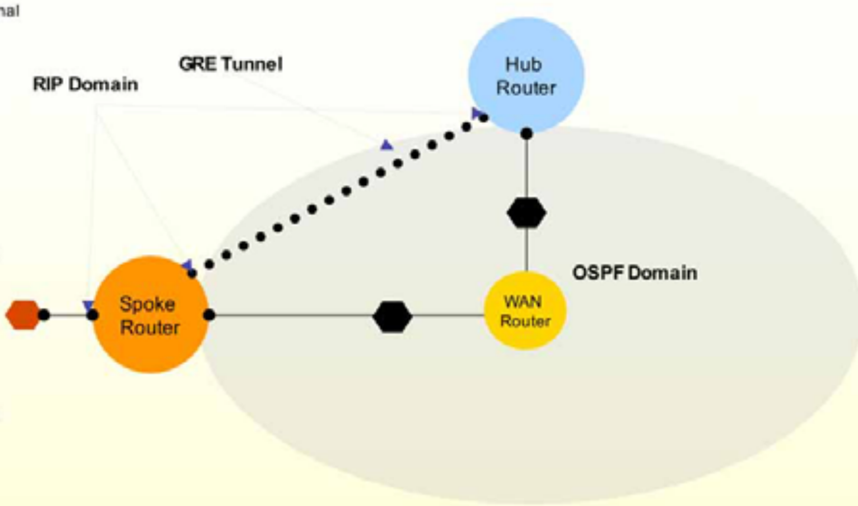
5. A router does not have more than one interface on a subnet
6. All internal interfaces are on internal subnets
7. All external interfaces are on external subnets
8. Every hub and spoke router is connected to a WAN router
9. No two non-WAN routers share a subnet

RoutingRequirements

10. RIP is enabled on all internal interfaces
11. OSPF is enabled on all external interfaces

GRERequirements

12. There is a GRE tunnel between each hub and spoke router
13. RIP is enabled on all GRE interfaces



To synthesize network, satisfy R1-R13 for
- previous list of components &
- 1 GRE tunnel

NOTE: GRE tunnel set up and RIP domain extended to include GRE interfaces automatically!

11

Strengthening Requirement: Adding Security For Overlay Network

RouterInterfaceRequirements

1. Each spoke router has internal and external interfaces
2. Each access server has internal and external interfaces
3. Each hub router has only external interfaces
4. Each WAN router has only external interfaces

SubnettingRequirements

5. A router does not have more than one interface on a subnet
6. All internal interfaces are on internal subnets
7. All external interfaces are on external subnets
8. Every hub and spoke router is connected to a WAN router
9. No two non-WAN routers share a subnet

RoutingRequirements

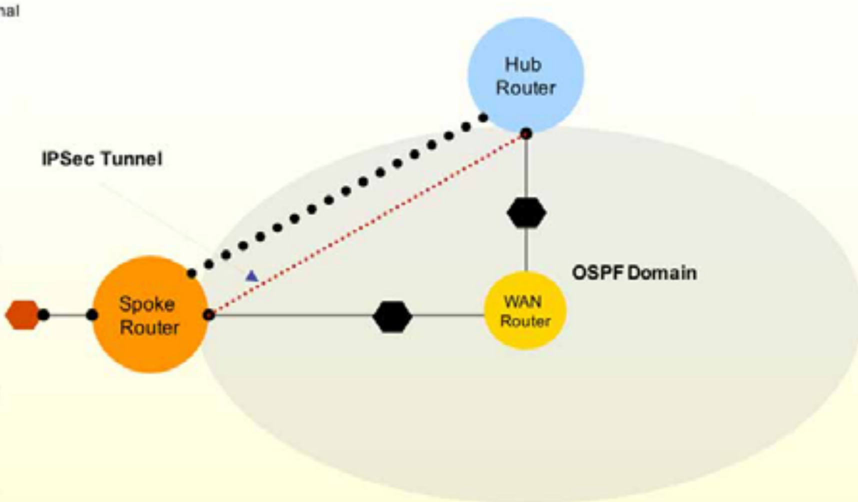
10. RIP is enabled on all internal interfaces
11. OSPF is enabled on all external interfaces

GRERequirements

12. There is a GRE tunnel between each hub and spoke router
13. RIP is enabled on all GRE interfaces

SecureGRERequirements

14. For every GRE tunnel there is an IPSec tunnel between associated physical interfaces that secures all GRE traffic



To synthesize network, satisfy R1-R14 for
- previous list of components &
- 1 IPSec tunnel

NOTE: IPSec tunnel securing GRE tunnel set up automatically

12

Strengthening Requirement: Adding Remote Access Service

RouterInterfaceRequirements

1. Each spoke router has internal and external interfaces
2. Each access server has internal and external interfaces
3. Each hub router has only external interfaces
4. Each WAN router has only external interfaces

SubnettingRequirements

5. A router does not have more than one interface on a subnet
6. All internal interfaces are on internal subnets
7. All external interfaces are on external subnets
8. Every hub and spoke router is connected to a WAN router
9. No two non-WAN routers share a subnet

RoutingRequirements

10. RIP is enabled on all internal interfaces
11. OSPF is enabled on all external interfaces

GRERequirements

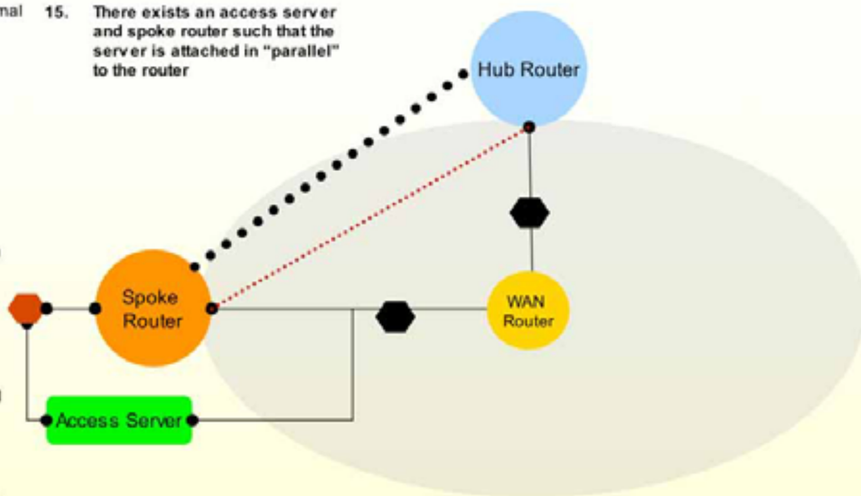
12. There is a GRE tunnel between each hub and spoke router
13. RIP is enabled on all GRE interfaces

SecureGRERequirements

14. For every GRE tunnel there is an IPSec tunnel between associated physical interfaces that secures all GRE traffic

AccessServerRequirements

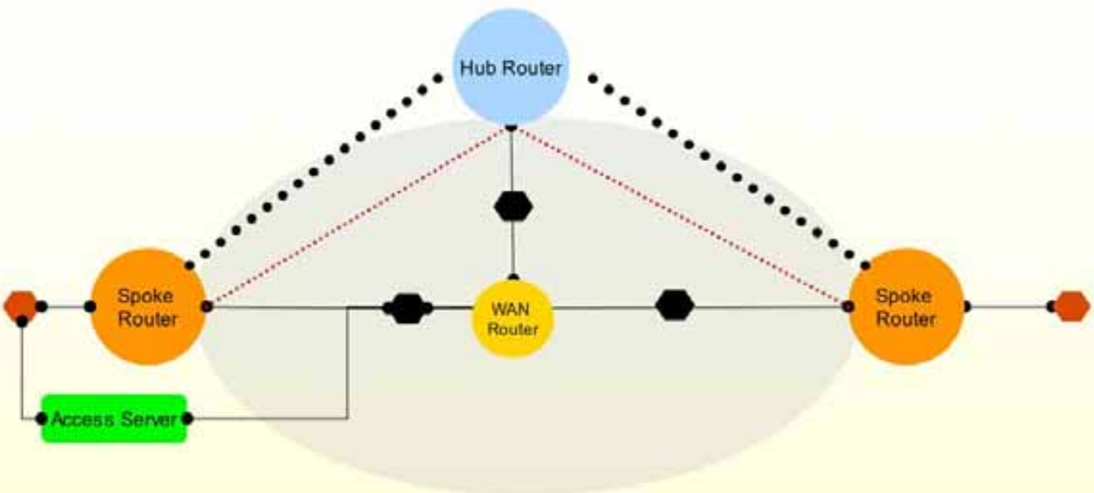
15. There exists an access server and spoke router such that the server is attached in "parallel" to the router



- To synthesize network, satisfy R1-R15 for previous list of components and 1 additional access server.
- Note: Access server interfaces placed on correct interfaces and RIP and OSPF domains correctly extended with internal and external interfaces, respectively

13

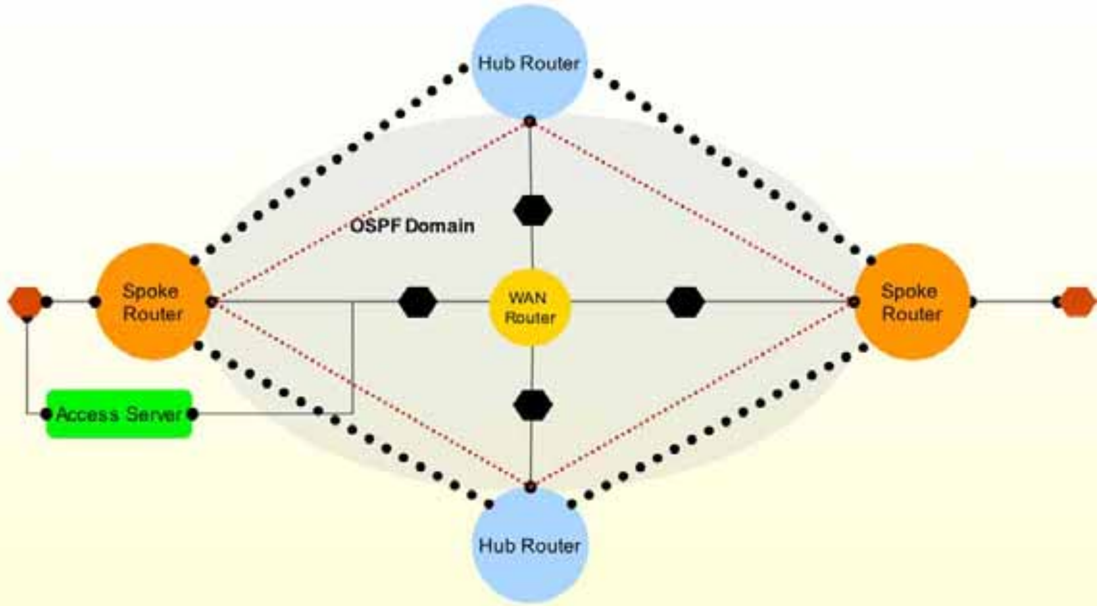
Component Addition: Adding New Spoke Router



- To add another spoke router satisfy requirements R1-R16 for previous components and one additional spoke router and related components
- Note: New subnets, GRE and IPSec tunnels set up, and routing domains extended *automatically*

14

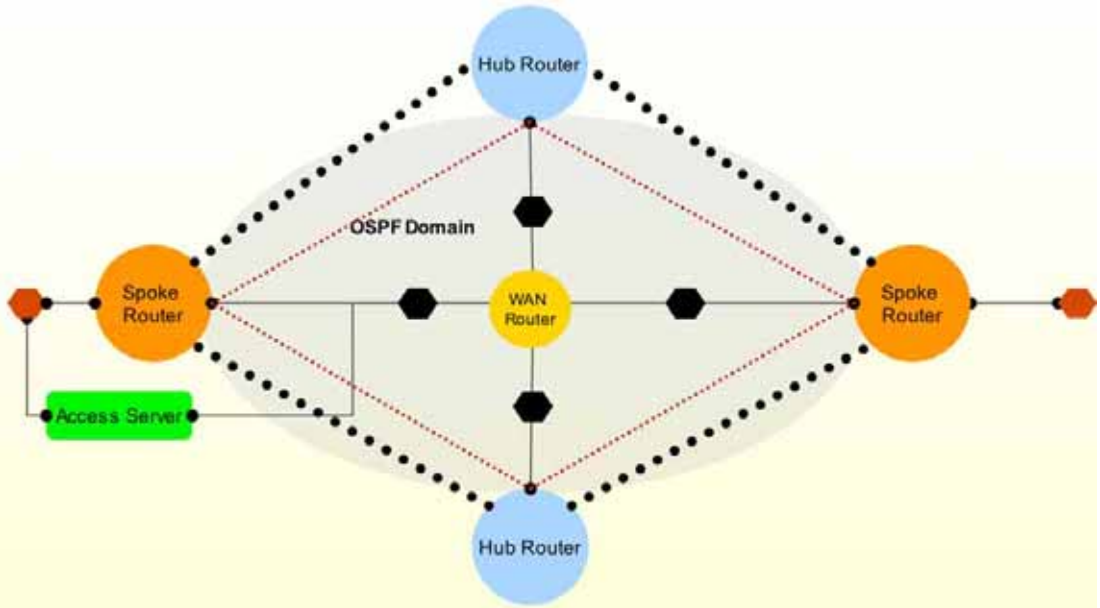
Component Addition: Adding New Hub Router



- To add another hub router satisfy requirements R1-R16 for previous components and one additional hub router (and related com[ponents)
- New subnets, GRE and IPSec tunnels set up, and routing domains extended *automatically*

15

Verification: Adding Firewall Requirements & Discovering Design Flaw



- Symptom: Cannot ping from one internal interface to another
- Define Bad = ip packet is blocked
- Check if R1-R16 & Bad is satisfiable
- Answer: WAN router firewalls block ike/ipsec traffic
- Action: Create new policy that allows WAN router firewalls to pass esp/ike packets

16

Summary & Future Directions

- Configuration plays central role in web services infrastructure synthesis & management
- We need a theory of configuration to automate synthesis and realize "autonomic" behavior
- Fundamental problems:
  1. Specification languages
  2. Configuration synthesis
  3. Incremental configuration (requirement strengthening, component addition)
  4. Configuration error diagnosis
  5. Configuration error troubleshooting
  6. Verification
  7. Configuration sequencing
  8. Distributed configuration
- Proposed formalization of 1-7 via Alloy and SAT solvers
- Future directions:
  - Scalable *algorithms* to solve above problems.

17

Thank You

18

Session 4

Synthesis and Wrap Up

Moderator and Rapporteur

Nicholas S. Bowen, IBM Systems Group, Austin, TX, USA

T. Basil Smith
Platform Issues

- Building integrated HW platforms such as Blade Offerings exposes weaknesses and ad hoc nature of current web practices
  - Control points example:
    - Critical component (sensing and actuation)
    - Each subsystem/vendor has unique interface, little thought to survivability, security of interface (as if each system expected direct VT100 attachment to serial port)
  - Virtualization Concepts now immature – but essential for tractability
    - Processor/Memory (compute core) fairly advanced
    - Disk – there but interoperability and inconsistencies are just as bad as unvirtualized resources
    - Network – vendor tool specific
  - Fragments of solutions
    - Work Load Balancing, Software Rejuv, VLAN's, Virtual Machines (e.g., VMware)
    - Some critical pieces seem to have made progress
      - Initial bare metal provisioning is example
    - Much more needs to be done – lots of pieces means lots of manual work (the non-autonomic part of the problem) e.g., initial provisioning and patching often different tools
- Approach to achieving tractability and scalability elusive
  - Simplicity vs flexibility and complexity

Platform Issues

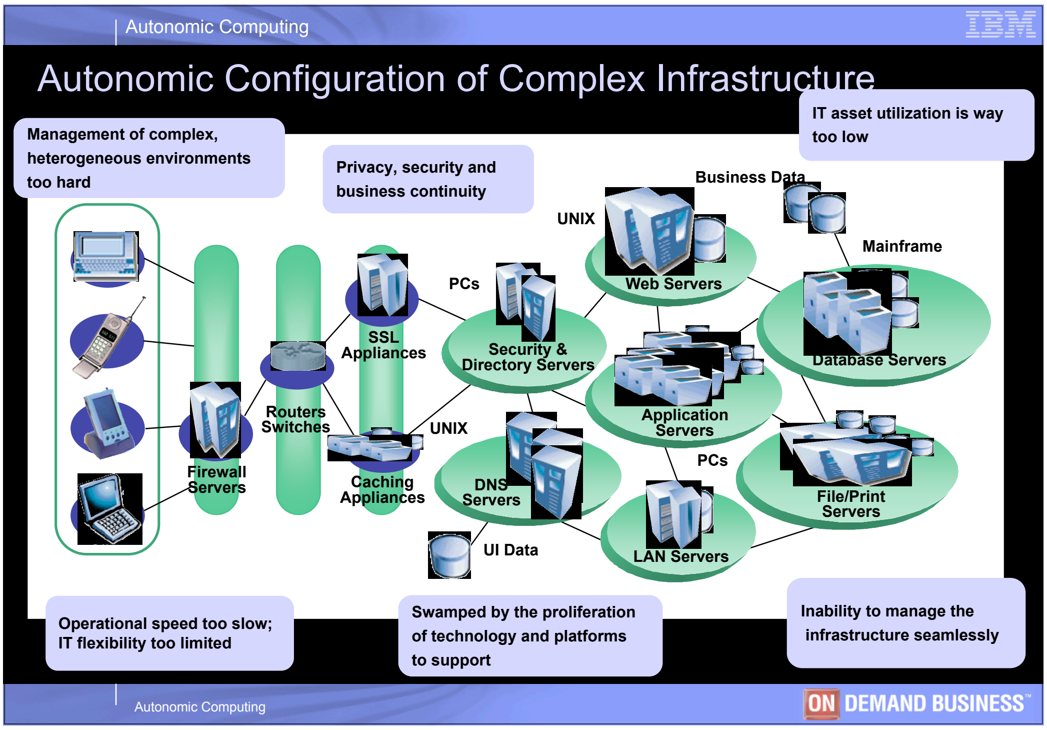
- Some consensus
  - "Service Processor" infrastructure seems a common feature
    - IBM, HP, Newisys all have service processors as key control component
    - The cluster of service processors and service processor redundancy not addressed
    - Security and manageability of service processor cluster needs to be addressed (are security attempt simply amateur, or are they effective)
  - Separating the disk from the computer core common theme
    - SAN and NAS attached storage
    - Magnifies management complexity issues – many difficult end-to-end problems.
  - Role of VLAN in multitier Web seemed:
    - Well understood
    - A complete mystery
    - Obviously critical security and control point

Platform Issues

- Some basic issues:
  - Pervasiveness of “Fail-Stop” assumptions
    - What design attributes are included and need to be included to back this assumption
      - Matched pairs for computational core for example
      - OS checking (never mind when the processor is brain dead, what about brain dead OS)
  - What are basic failure rates, failure modes, failure correlations
    - Lots of uncertainty going forward as:
      - Increasing circuit densities may or may not increase transient error rates
      - Critical SW failure rates and modes are unknown now with more uncertainty looking forward (e.g., how frequently does Windows fail and what fraction of those failures corrupt critical components of file system, or how frequently does firmware in RAID subsystem lose all the data in the RAID subsystem)
      - What about the backplanes in these integrated systems
      - How often does management subsystem mistakenly turn off all elements in system
      - What are the basic HW failure rates

Platform Composibility Issues

- General composibility problem with constrained perfectly virtualized resources is very hard (HP UDC made stab are regularizing resources – fixed wiring constraints)
  - With constraints this is still a form of classically impossible problem if best solution is required, even good is hard
- With general imperfectly virtualized resources things may be intractable


Autonomic Computing
IBM
Autonomic Configuration of Complex Infrastructure
Management of complex, heterogeneous environments too hard
Privacy, security and business continuity
IT asset utilization is way too low
Business Data
UNIX
Mainframe
Web Servers
PCs
SSL Appliances
Security & Directory Servers
Database Servers
Application Servers
Routers Switches
Firewall Servers
UNIX
PCs
Caching Appliances
DNS Servers
File/Print Servers
LAN Servers
UI Data
Operational speed too slow; IT flexibility too limited
Swamped by the proliferation of technology and platforms to support
Inability to manage the infrastructure seamlessly
Autonomic Computing
ON DEMAND BUSINESS

Open Questions – Autonomic Response to Faults and Attacks – William H. Sanders

- What is the definition of Autonomic? Does it matter?
- What kind of faults and attacks can be tolerated autonomically?
- How does one specify the desired (security and dependability) properties in an autonomic web computing infrastructure?
- How can high-level dependability and security requirements be translated to low level configuration decisions?
- What measurement data should be collected to feed into the analysis module?
- Are existing failure/attack detection techniques sufficient?
- What analysis techniques are useful? Do useful ones exist?
- Can measurements be used to use to iteratively refine the models that are used for analysis?
- How can we benchmark/evaluate the quality of an autonomic web computing infrastructure?

On Security Issues

Carl Landwehr
NSF

Autonomic Web Computing - Security

1. What kinds of attacks are prevalent today and what kinds are expected in the future?

2. What techniques are currently available to defend e-commerce sites against these attacks?

Brian L & Bob B

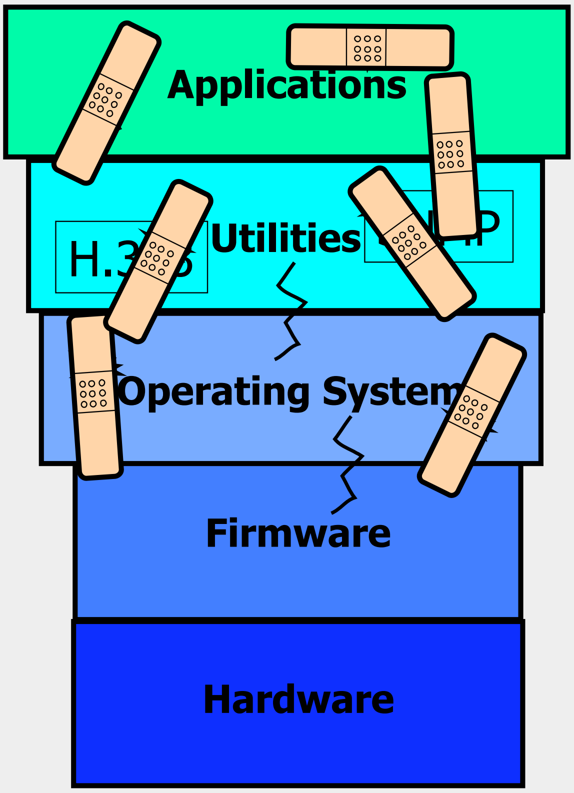
3. How are security configurations for web services specified, configured, and verified? To what extent can these functions be automated?

Elisa B & Sanjain N

4. What is the distance between theory (e.g. in cryptographic protections) and mechanisms actually in use?

John B

Autonomic Security?



We built it - can we fix it?

- How must it be?
  - What are the limits?
- How might it be?
  - What are the possibilities

Some Limits

- Mathematical/logical
  - Access control questions in some models are undecidable (HRU, 1976)
  - Obfuscation is impossible (BGIRSVY, 2001)
  - One time pads can support unbreakable ciphers
  - Shannon's theorem bounds channel capacity
- Physical
  - Reading a quantum-entangled photon alters its state
  - The speed of light limits the rate of information transmission
- Economic
  - Rational consumers don't spend money on undetectable properties
- Social
  - Perfection is not of this world

Observation: the economic and social limits have limited security more than the mathematical and physical ones

University of Puerto Rico – Mayagüez 5 IFIP Working Group 10.4 Winter Meeting 2005

Some Current Assumptions

- **Internet protocols can't be substantially changed or replaced**
- **Operating systems will have 50 million lines of code or more**
- **Security must be reactive**

University of Puerto Rico – Mayagüez 6 IFIP Working Group 10.4 Winter Meeting 2005

We need to think further out

- **Couldn't we at least:**
  - **Create and deploy mechanisms to allow us to identify where a message originated with a good degree of certainty**
  - **Figure out how to build system interfaces that real people (users and developers) can understand and use**
  - **Learn how to organize systems so that even when imperfect, they are not prone to catastrophic failure under attack**

We are a long way from the limits

We need to think of more possibilities

Discussion?

IFIP WG 10.4

Business Meeting

47th IFIP WG 10.4 Meeting

Rincón of the Seas Grand Caribbean Hotel, PR, USA

Wednesday January 26 — Sunday January 30, 2005

Saturday January 29, 2005

Agenda

- IFIP World Computer Congress — WCC'2004 (J.-C. Laprie)
- IEEE/IFIP DSNs - DSN-2005, DSN-2006 (T. Nanya, C.Kintala)
- IEEE Trans. on Dependable and Secure Computing
- Future WG Meetings — 48, 49, … 50 (T. Nanya)
- TC-10 Conference at WCC'2006
- Other Supported Events
- [Membership -- restricted to WG members]

Top3: Fault Tolerance for Trustworthy and Dependable Information Infrastructures

- **Monday 23 August 2004 (Afternoon)**

  13h30 - 15h — Setting up the Scene — Alain Costes
  - *Brief Addresses by the IFIP WG10.4 Past and Current Chairs*
    Algirdas Avizienis, Jean-Claude Laprie, Hermann Kopetz, Jean Arlat
  - *Dependable Systems of the Future: What Is Still Needed?*
    Algirdas Avizienis (UCLA, USA and Vytautas Magnus U., Kaunas, Lithuania)
  - *Dependability and Its Threats: A Taxonomy*
    Algirdas Avizienis, Jean-Claude Laprie (LAAS-CNRS), Brian Randell (U. Newcastle, UK)

  15h30 - 17h30 — Contributions, Advances and Trends — Jacob Abraham
  - *Current Research Activities on Dependable Computing and Other Dependability Issues in Japan*
    Yoshihiro Tohma (Tokyo Denki U.) , Masao Mukaidono (Meiji U); Japan
  - *Dependable Computing at Illinois*
    Ravishankar Iyer, William Sanders, Janak Patel, Zbigniew Kalbarczyk (UIUC, USA)
  - *Wrapping the Future*
    Tom Anderson, Brian Randell, Alexander Romanovsky (U. Newcastle, UK)
  - *From the University of Illinois via JPL and UCLA to Vytautas Magnus University: 50 Years of Computer Engineering by Algirdas Avizienis*
    David Rennels, Milos Ercegovac (UCLA, USA)

Top3: Cont'

- **Tuesday 24 August 2004 (All day)**

  10h30 - 12h — Dependability and Predictability of Embedded Systems — Hiro Ihara
  - *Airbus Fly-by-Wire: A Total Approach to Dependability*
    Pascal Traverse, Isabelle Lacaze, Jean Souyris (Airbus, France)
  - *Unique Dependability Issues for Commercial Airplane Fly By Wire Systems*
    Ying C. Yeh (Boeing Corporation, Seattle, WA, USA)
  - *The Fault-Hypothesis for the Time-Triggered Architecture*
    Hermann Kopetz (U. Technology, Vienna, Austria)

  13h30 - 15h — Focuses on Communications, Security, and Software Verification — Yoshi Tohma
  - *Communications Dependability Evolution Between Convergence and Competition*
    Michele Morganti (Siemens Mobile Communications, Milan, Italy)
  - *Intrusion Tolerance for Internet Applications*
    Yves Deswarte, David Powell (LAAS-CNRS, France)
  - *Static Program Transformations for Efficient Software Model Checking*
    Shobha Vasudevan, Jacob A. Abraham (U. Texas at Austin, USA)

  15h30 - 17h — Further Challenges and Perspectives — Bill Sanders
  - *Architectural Challenges for a Dependable Information Society*
    Luca Simoncini (U. Pisa and PDCC), Andrea Bondavalli (U. Florence and PDCC), Felicita Di Giandomenico, Silvano Chiaradonna (ISTI-CNR, Pisa and PDCC); Italy
  - *Experimental Research in Dependable Computing at Carnegie Mellon University*
    Daniel P. Siewiorek, Roy A. Maxion, Priya Narasimhan (Carnegie Mellon U., Pittsburgh, USA)
  - *Systems Approach to Computing Dependability In and Out of Hitachi: Concept, Applications and Perspective*
    Hirokazu Ihara (Hiro Systems Laboratory Tokyo, Japan), Motohisa Funabashi (Hitachi Ltd, Kawasaki, Japan)

Friendly Dinner

IEEE/IFIP International Conference on Dependable Systems and Networks

Yokohama, Japan (June 28 - July 1, 2005)

- General Chair: Takashi Nanya (University of Tokyo, Japan)
- Conference Coordinator: Tohru Kikuno (Osaka University, Japan)
- DCCS Program Chair: Andrea Bondavalli (University of Florence, Italy)
- PDS Program co-Chairs: Boudjwin Haverkort (Univ. of Twente, The Netherlands) Dong Tang (Sun Microsystems, CA, USA)

Philadelphia, PA, USA (June 25-28, 2006)

- General Chair: Chandra Kintala (Stevens Inst. of Technology, Hoboken, NJ, USA)
- Conference Coordinator: David Taylor (Univ. of Waterloo, Canada)
- DCCS Program Chair: Lorenzo Alvisi (University of Texas, Austin, USA)
- PDS Program Chair: Aad Van Moorsel (University of Newcastle Upon Tyne, UK)

IEEE Transactions on Dependable and Secure Computing

http://computer.org/tdsc

-> Quarterly Journal - Three Issues (2004) already out

- 2nd Editorial Board Meeting at UIUC in Dec. 2005
- More than 100 submissions already received
- Think of submitting a paper!

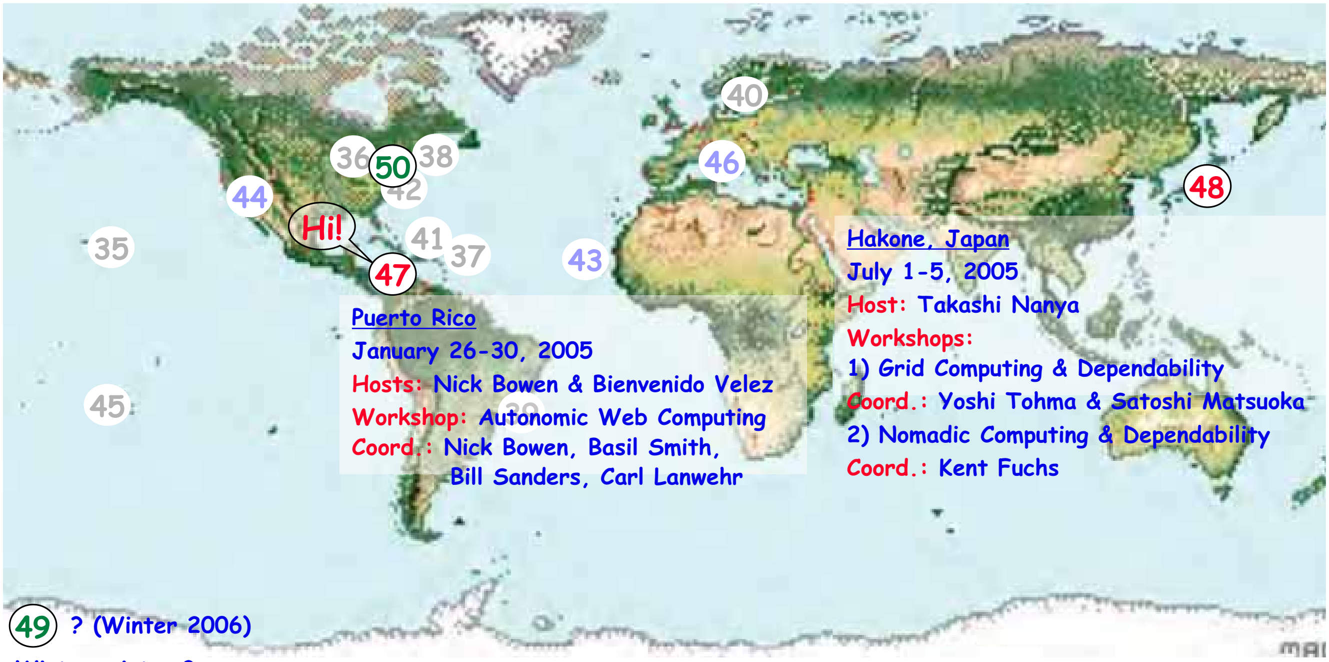
(Some) Proposals for Workshop Topics

- Autonomic Web Computing
- Nomadic Computing and Dependability (Kent)
- Grid Computing and Dependability (Yoshi)

—> at meeting 48 (Hakone)

- Security and Operational Challenges for Service Providers Networks (Farnam) —> tentatively, with 50th meeting linked to DSN-2006 ?
- Dependability in Robotics and Autonomous Systems (David Powell)
  [Possibly in connection with Int. Advanced Robotics Programme WG on Robot Dependability]

Future Meetings



Winter-winter ?

Cabo San Lucas, Mexico [Initially proposed for Winter 2005]

Hosts: Kane Kim & Phil Koopman

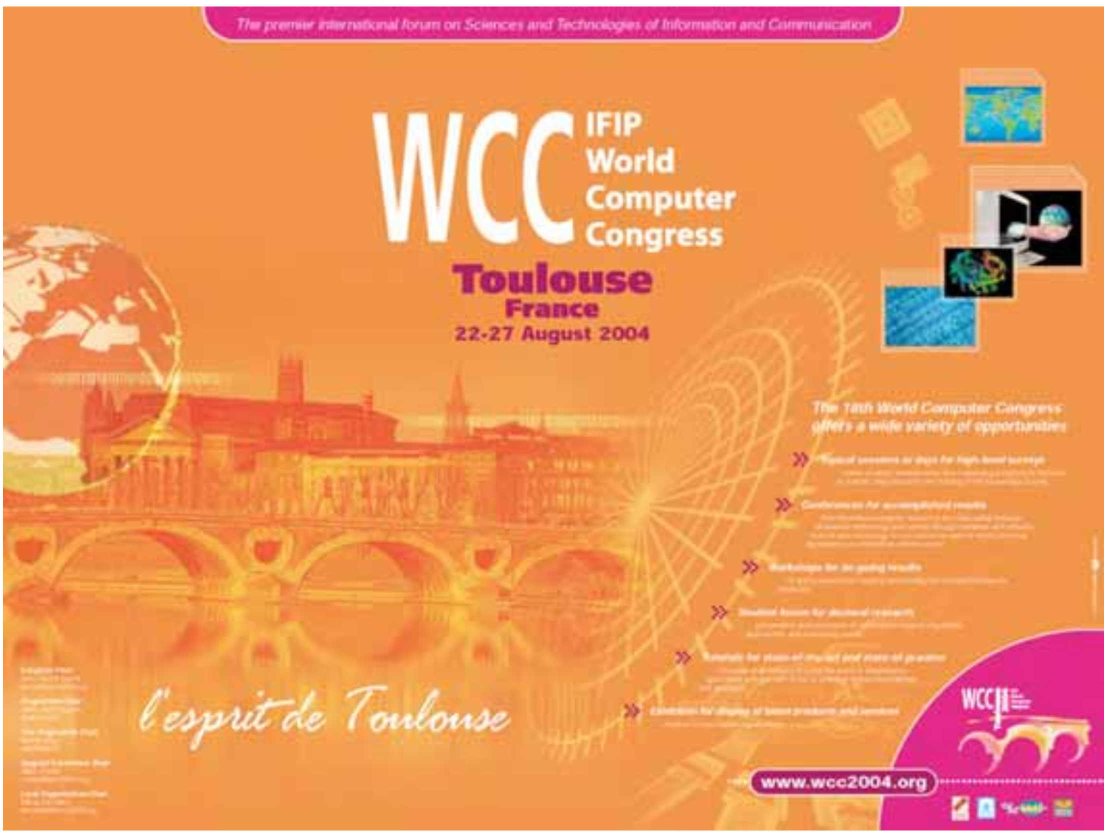
TC-10 Conference at IFIP WCC-2006
Biologically Inspired Cooperative Computing

- Chairs: Franz Rammig (Chair TC10) & Mauricio Solar (U Sant. Chile)
- Program Chairs: Yi Pan (U. Georgia) & Hartmut Schmek (U. Karlsruhe)
- Not bio-informatics -> Four Streams:
  - (1) Modelling and Reasoning about Collabarative Self-Organizing Systems (10.1)
  - (2) Collaborative Sensing and Processing Systems (10.3)
  - (3) Robustness and Dependability in Collaborative Self-Organizing Systems (10.4)
  - (4) Design and Technology of Collaborative Self-Organizing Systems (10.5)

Other (in cooperation) Events

- **SAFECOMP-2004** (23rd International Conference on Computer Safety, Reliability and Security), Potsdam, Germany, **September 21-24, 2004** — http://www.safecomp.org
- **SRDS-2004** (22nd Symp. on Reliable Distributed Systems), Florianopolis, SC, Brazil, **October 18-20, 2004** — http://www.SRDS2004.ufsc.br
- **WORDS-2005** (10th Int. Workshop on Object-oriented Real-time Dependable Systems), **Sedona, AZ, USA, February 2-4, 2005** — http://asusrl.eas.asu.edu/srlab/activities/words05/words05.htm
- **EDCC-2005** (5th European Dependable Computing Conference), Budapest , Hungary, **April 20-22, 2005** — http://sauron.inf.mit.bme.hu/EDCC5.nsf
- **4th IARP/IEEE-RAS/EURON** Workshop on Technical Challenges for Dependable Robots in Human Environments, **Nagoya, Japan, June 16-18, 2005**
- **SAFECOMP-2005** (24th International Conference on Computer Safety, Reliability and Security, Norway, **September 28-30, 2005** — http://www.safecomp.org
- **LADC-2005** (2nd Latin-American Symposium on Dependable Computing), Salvador, Bahia, Brazil, **October 25-28, 2005** — http://www.lasid.ufba.br/ladc2005
- **PRDC-2005** (11th Int. Symp. Pacific Rim Dependable Computing), Changsha, China, **December 12-14, 2005** — http://sc.hnu.cn/newweb/communion/prdc2005/presentation.htm


WCC
Toulouse
2004
IFIP
World
Computer
Congress
IFIP
ifip
see
LAAS
EUROPA


The premier international forum on Sciences and Technologies of Information and Communication
WCC
IFIP
World
Computer
Congress
Toulouse
France
22-27 August 2004
The 18th World Computer Congress
offers a wide variety of opportunities
l'esprit de Toulouse
www.wcc2004.org

Congress Structure

Day	Programme				
Sunday 22 August	Tutorials for state-of-the-art and state-of-practice Workshops for on-going research				
Monday 23 August To Thursday August 26	Plenary Keynote Addresses				Exhibition for display of latest products and services
	Co-Located Conferences for accomplished results	Topical Days for high-level surveys and prospective views	Student Forum for doctoral research		
Friday 27 August	Workshops for on-going research				

Programme contents

- 5 keynotes
- 9 co-located conferences, 367 papers from 48 countries (out of 900+ submissions from 60 countries), 15 invited talks, 7 panels
- 14 topical days, 91 invited talks, 7 panels
- Student forum, 43 papers
- 10 workshops, 109 papers, 6 invited talks, 6 panels
- 20 proceedings volumes, 14 at congress, 6 post-congress

Conferences

- TCS: Theoretical Computer Science
  - TCS-Algorithms: Track 1 — Algorithms, Complexity and Models of Computation
  - TCS-Logic: Track 2 — Logic, Semantics, Specification and Verification
- SEC: Information Security
  - SEC.ISM: Information Security Management
  - SEC.ISE: Information Security Education
  - SEC.I-Net: Privacy and Anonimity in Networked and Distributed Systems
- CARDIS: Smartcard Research and Advanced Applications
- DIPES: Distributed and Parallel Embedded Systems
- AIAI: Artificial Intelligence Applications and Innovations
  - Symposium on Professional Practice in AI
- HESSD: Human Error, Safety and System Development
- PRO-VE: Virtual Enterprises
- I3E: e-Commerce, e-Business, e-Government
- HCE: History of Computing in Education

Topical Days

Top1 Semantic Integration of Heterogeneous Data
Top2 Virtual Realities and New Entertainment
Top3 Fault Tolerance for Trustworthy and Dependable Information Infrastructures
Top4 Abstract Interpretation
Top5 Multimodal Interaction
Top6 Computer Aided Inventing
Top7 Emerging Tools and Techniques for Avionics Certification
Top8 The Convergence of Bio- Info- and Nano-Technologies
Top9 E-Learning
Top10 Perspectives on Ambient Intelligence: Infrastructure, Governance, Applications and Ethics
Top11 TRaIn: The Railway Infrastructure — A grand challenge for computing science: towards a domain theory for transportation
Top12 Open Source Software in Dependable Systems
Top13 Critical Infrastructures Protection

Workshops

Ws2	Technology Enhanced Learning
Ws3	Certification and Security in inter-organizational e-services
Ws4	Formal Aspects in Security and Trust
Ws5	EduTech
Ws6	Architecture Description Languages
Ws7	Broadband Satellite Communication Systems
Ws8	Challenges of Mobility
Ws9	High Performance Computational Science and Engineering
Ws10	International Summit on Computing Professionalism
Ws11	Prep-WITFOR 2005 Workshops

Attendees

Totals

Delegates	1087
Exhibitors	218

Number countries	71
France	295
Germany	85
United Kingdom	75
USA	70
Italy	45
Brazil	32
Japan	35
Spain	35
…	

Academia	865
Industry	152
(incl. Exhib.	370)
Gov. Agencies	40

Scientific success

Organisational success

Interaction

The President's Report to IFIP General Assembly 2004 in Toulouse

IFIP World Computer Congress 2004 in Toulouse: a large success!

… an event which will be long remembered (besides the material products as 21 books and their electronic images) in IFIP and in the participants memories as one of the best organised IFIP World Computer Congresses ever.

The President's Report to IFIP General Assembly 2004 in Toulouse

IFIP World Computer Congress 2004 in Toulouse: a large success!

… an event which will be long remembered (besides the material products as 21 books and their electronic images) in IFIP and in the participants memories as one of the best organised IFIP World Computer Congresses ever.

The International Conference on Dependable Systems and Networks (DSN2005)

Pacific Convention Center (Pacifico), Yokohama, Japan
June 28(Tue) - July 1(Fri), 2005

Access

From Narita Airport to Yokohama St.
90 min. by airport limousine bus, or JR Narita Express

Yokohama St. to Minatomirai St.
3 min. by subway

Conference site, Hotels:
1 ~ 10 min. by foot from Minatomirai St.

Conference Site

Papers & Panel

- DCC: 205 (including 1 panel)
- PDS: 95

- PDS-PC meeting: Feb.17, 18 at Zaandam

- DCC-PC meeting: Feb.21, 22 at Pisa
- SC meeting: Feb. 22 at Pisa

Workshops

Workshops Chair: Nuno Ferreira Neves
accepted all the three submitted

1. 1. Hot Topics in System Dependability, organized by George Candea (Stanford Univ.), David Oppenheimer (UCB)

2. 2. Dependable Software - Tools and Methods, organized by Takuya Kayatama (JAIST, Japan), Yutaka Kikuchi(Univ.Tokyo)

3. 3. Assurance of Networking Systems Dependability Service Level Agreements, organized by Saida Benlarbi (Alcatel, Canada) , Kishor Trivedi(Duke univ., USA), Khaled El-Emam(TrialStat , Canada)

Proposing one more

4. Dependability in Automotive Electronics: X-by-Wire, organized by Masaharu Asano (Nissan, Japan), Herman Kopetz (Wien Tech. Univ.)

Industry session

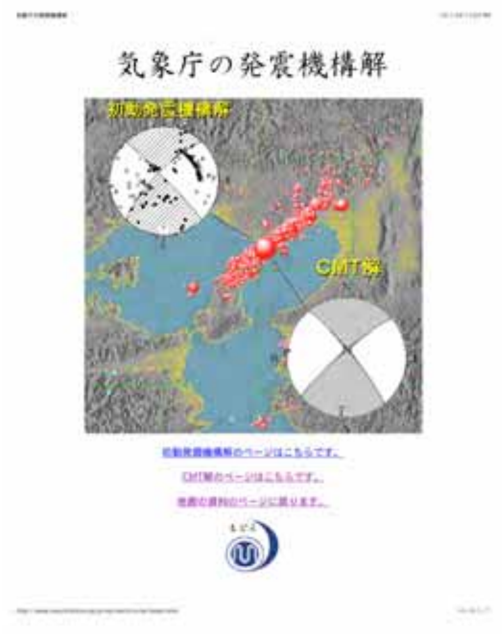
reviewed lightly by subset of DCC-PC or PDS-PC
presented in separate track from DCC and PDS
published in Vol.2
Submission:Mar.1 Notice:Mar.21, Camera-ready:Apr.21

Ansaldo Segnalamento Ferroviario, railway interlocking systems
IBM zSeries systems RAS group
Sun microsystems, HPCS RAS group
JR(Japan Railway), Reliability group
Fujitsu, Server system group
NEC, System Platform group
Hitachi,
Samsung, and more . . .

Keynote Speaker

Dr. Mitsuyuki Hoshiba
(Japan Meteorological Agency)

"Tsunami warning system"



Other technical programs

Tutorials
Chair: Zbigniew Kalbarsczyk (Univ. of Illinois, USA)
will be finalized in SC meeting on Feb.22

Student Forum
Chair: Philip Koopman (CMU, USA)
Submission: Apr.1

Fast Abstracts
Chair: Matti A. Hiltunen (AT&T, USA)
Submission: Apr.1

Social Events

- Reception: on June 28 at Pacifico
- Excursion: late afternoon on June 30
  - Japanese Garden
  - No performance
  - Tokyo Bay Cruising & Banquet

Excursion

“Japanese Garden”

鶴翔閣 正面入口

横浜能楽堂 05.1.30 1:03 AM

City of YOKOHAMA トップメニュー 検索

横浜能楽堂

NO performance

2月の舞台
能「清経」（宝生流）佐野由於

りください。

Registration Fee

25% lower than 2004 !!

Advance/Member	**:￥55,000**	**$529**	**Euro 407**
(Florence, 2004	:￥74,250	$714	Euro 550)
Advance/Student	**:￥30,000**	**$288**	**Euro 222**
(Florence, 2004	:￥40,500	$389	Euro 300)

100 ￥ = 104 $ = 135 Euro

On-site : 20% higher than advance rate
Non-member: 25% higher than member rate

Hotels

- Reserved blocks of 6 hotels
- Located within waking distance

Hotel name	:distance	single(yen) ,	twin(yen)
Intercontinental	: next door,	18,700,	23,100
Panpacific	: 2min.,	20,000,	24,000
Royal Park	: 5min.,	18,700,	28,600
Washington	:10min.,	11,500,	19,000
Navios Yokohama	: 7 min.,	9,000,	17,000
Breeze Bay	:12min.	9,000,	15,000


YOKOHAMA Convention & Visitors Bureau http://www.welcome.city.yokohama.jp/


YOKOHAMA Convention & Visitors Bureau http://www.welcome.city.yokohama.jp/

See you in Yokohama

in June !

Sheraton Society Hill, Philadelphia, PA

http://www.starwoodhotels.com/sheraton/search/hotel_detail.html?propertyID=166

Saturday June 24 -

Wednesday June 28, 2006

Hotel Information:

Meeting Rooms

- Number of Meeting Rooms: 10
- Largest Meeting Room seats: 950
- Internet access in rooms and meeting rooms
- Philadelphia is trying to get city-wide wireless hot-spot facility

- Social Event Possibilities:
  - Exclusive tour and dinner in Philadelphia Museum of Art
  - Cruise and dinner on Spirit of Philadelphia
  - Baseball game
  - … ???

Update at WG10.4 Puerto Rico

Page 3

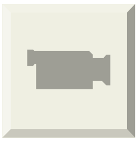
Local Attractions

- Independence Hall, Liberty Bell - 0.3 mi/0.4 km
- Betsy Ross House, Constitution Center - 0.3 mi/0.4 km
- Philadelphia Museum of Art - 2.0 mi/3.2 km
- Penn's Landing, Spirit of Philadelphia - 0.1 mi/0.1 km
- Independence Seaport Museum - 0.1 mi/0.1 km
- Horse-Drawn Carriage Tours - 0.1 mi/0.2 km
- Downtown - 0.6 mi/1.0 km
- Philadelphia Orchestra - 2.0 mi/3.2 km
- Sesame Place - 23.0 mi/37.0 km
- Atlantic City - 50.0 mi/80.5 km
- New Jersey State Aquarium - 3.0 mi/4.8 km
- Philadelphia Sports Teams: Eagles, Phillies, Flyers, 76ers

Update at WG10.4 Puerto Rico

Page 4

Philadelphia, PA:



Video available at
http://www.pcvb.org/mtgplanners/cs_video.asp

Update at WG10.4 Puerto Rico

Philadelphia: Accessibility

- Philadelphia International Airport: 10 miles from hotel
  - Daily flights from/to most cities in U.S. and Europe
- "Liberty Shuttle" for transportation tøfrom City Hotels ($8.00 one way)
- Newark International (80 miles) and JFK (110 miles)
- Amtrak trains from EWR and NYC/NJ to Philadelphia

Update at WG10.4 Puerto Rico

Estimated costs

Rooms (Reservation cut-off date June 2)	$159
Breaks	$15
Luncheons	$45
Meeting space	Complimentary or charge based on hotel room bookings
A/V + Internet + Computer + …	$25,0000
Reception	$40 - $50
Social + Banquet	$125 - $150
On-time Registration	Member: $640-$670 Non-Member: $750-$800 Student: $250-$300

Update at WG10.4 Puerto Rico

Page 7

Organization Schedule

- Had to prepare a draft TMRF for preliminary approval by IEEE CS before hotel contract was signed in November 2004
  - Several issues with IEEE CS; process took 6 months
- Funding calls: any help would be most appreciated
- Filling-in the other committee positions: suggestions welcome
- Print CFP by June'05
- Decide Social Event
- WG10.4 meeting location possibilites: Cape May, NJ or Pocono Mountains in PA
- And so on …

Update at WG10.4 Puerto Rico

Page 8

IFIP WG10.4 48th Meeting

July 1 (Fri) - July 5 (Tue), 2005
(immediately following DSN2005)

Hakone (in Fuji-Hakone National Park)
Hotel de Yama
(http://www.odakyu-hotel.co.jp/yama-hotel/english/)
Hakone Lakeside since 1947
2 hours from Yokohama








Schedule

July 1st (Fri) : Yokohama => Hakone, Evening Reception

July 2nd (Sat) : Workshop Grid Computing & Dependability
(Chaired by Yoshi Tohma, Satoshi Matsuoka)

July 3rd (Sun) : Excursion & Banquet

July 4th (Mon):Workshop Nomadic Computing & Dependability
(Chaired by Kent W. Fuchs) + Business meeting

July 5th(Tue) :Workshop Nomadic Computing & Dependability
(or Research Reports)* -- ending at noon

One-day excursion

- O-waku Valley
- Mt.Fuji
- Sake Cellar

O-waku Valley

Hotel & Registration Fee

- Hotel rate (tax included):
  single:16320 yen, 157 $, 121 euro
  twin:19935 yen, 192 $, 148 euro

- Registration Fee: 400 Euro or $ (tentative)

Research Reports

Session 1

Moderator

Takashi Nanya, University of Tokyo, Japan

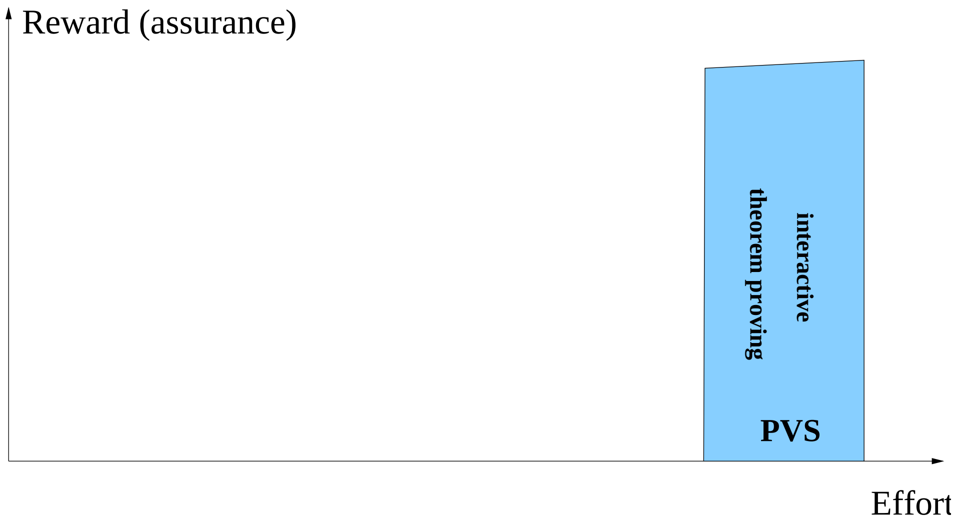
IFIP WG 10.4 Winter Meeting, Rincon PR 30 Jan 2005

Automated Test Generation with sal-atg

John Rushby
with Grégoire Hamon and Leonardo de Moura

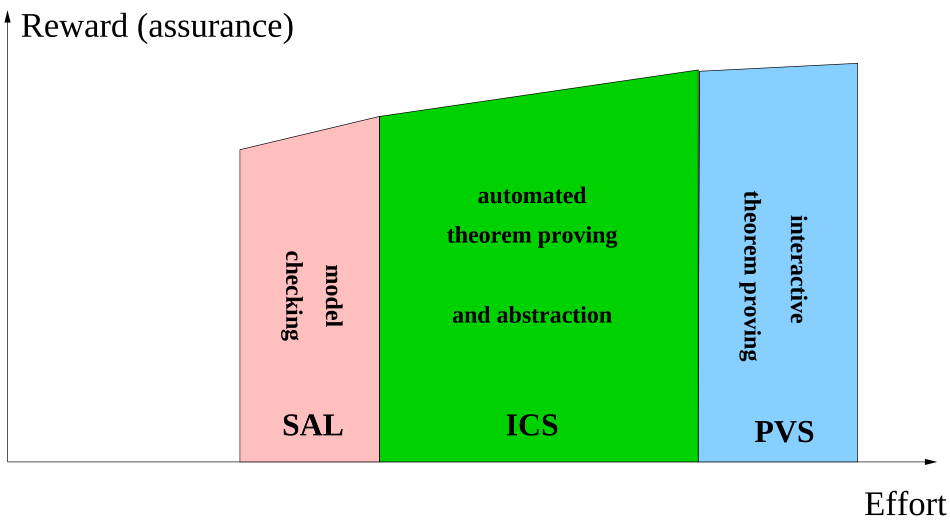
Computer Science Laboratory
SRI International
Menlo Park CA USA

Full Formal Verification is a Hard Sell: The Wall



John Rushby, SR I sal-atg: 2

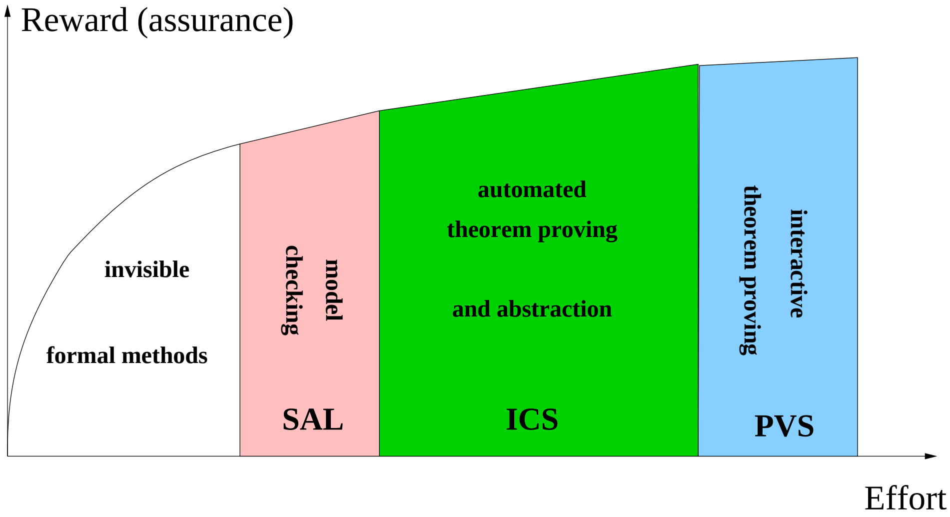
Newer Technologies Improve the Value Proposition



But only by a little

John Rushby, SR I sal-atg: 3

The Unserved Area Is An Interesting Opportunity



Conjecture: reward/effort climbs steeply in the invisible region

Invisible Formal Methods

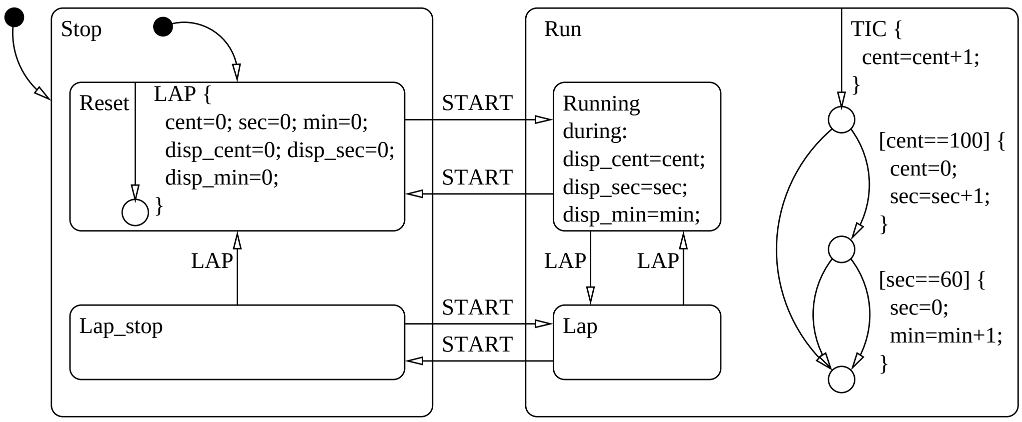
- Use the technology of formal methods
  - Theorem proving, constraint satisfaction, model checking, abstraction, symbolic evaluation
- To augment traditional methods and tools
  - Compilers, debuggers
- To automate traditional processes
  - Testing, reviews, debugging
- Or to create new capabilities
  - Strong static analyzers, autocode by constraint solving
- To do this, we must unobtrusively (i.e., invisibly) extract
  - A formal specification
  - A collection of properties
- And deliver a useful result in a familiar form

Invisible FM Example: Generating Unit Tests

- Necessity and costs of testing well understood
- Automation could be a huge win
- In model based development (MBD), we have an executable model of the system (e.g., in Simulink/Stateflow)
- Generate tests by structural coverage in the model
- Model also provides the oracle
- It is well known that model checkers can be used as test generators

Example: Stopwatch in Stateflow

Inputs: START and LAP buttons, and clock TIC event



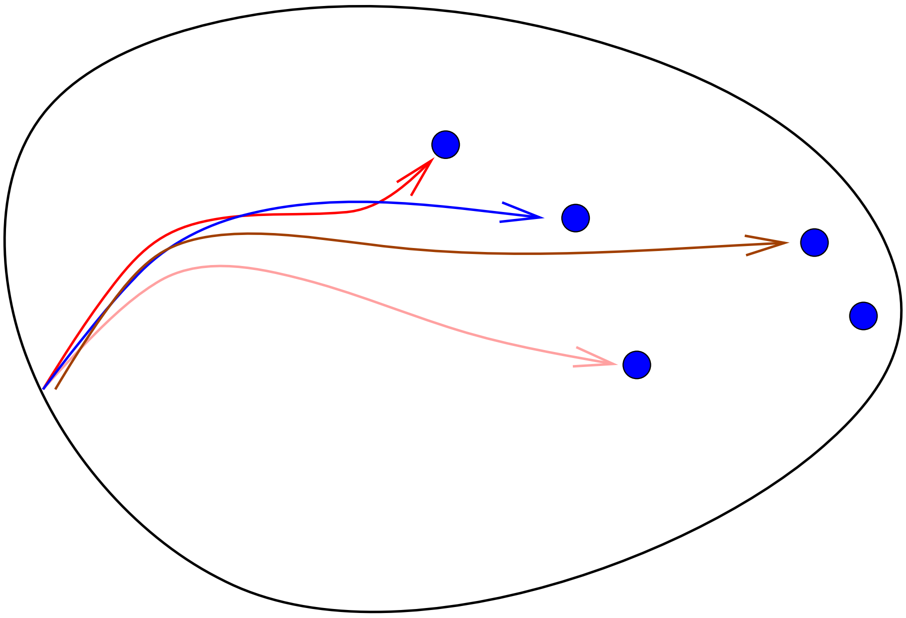
Example test goals: generate input sequences to exercise Lap_stop to Lap transition, or to reach junction at bottom right

Generating Tests Using a Model Checker

- Add trap variables go TRUE when a test goal is satisfied
  - E.g., jabr that goes TRUE when junction at bottom right is reached
  - Trap variables can be inserted automatically during translation from the MBD language to the model checker (Our translator from Stateflow to SAL does this)
- Model check for “always not jabr”
- Counterexample will be desired test case
- Trap variables add negligible overhead ('cos no interactions)
- For finite cases (e.g., numerical variables range over bounded integers) any standard model checker will do
  - Although many pragmatic issues concerning symbolic vs. bounded vs. explicit vs. . . for this application
  - Otherwise need infinite bounded model checker as in SAL

John Rushby, SR I sal-atg: 8

Tests Generated Using a Model Checker



John Rushby, SR I sal-atg: 9

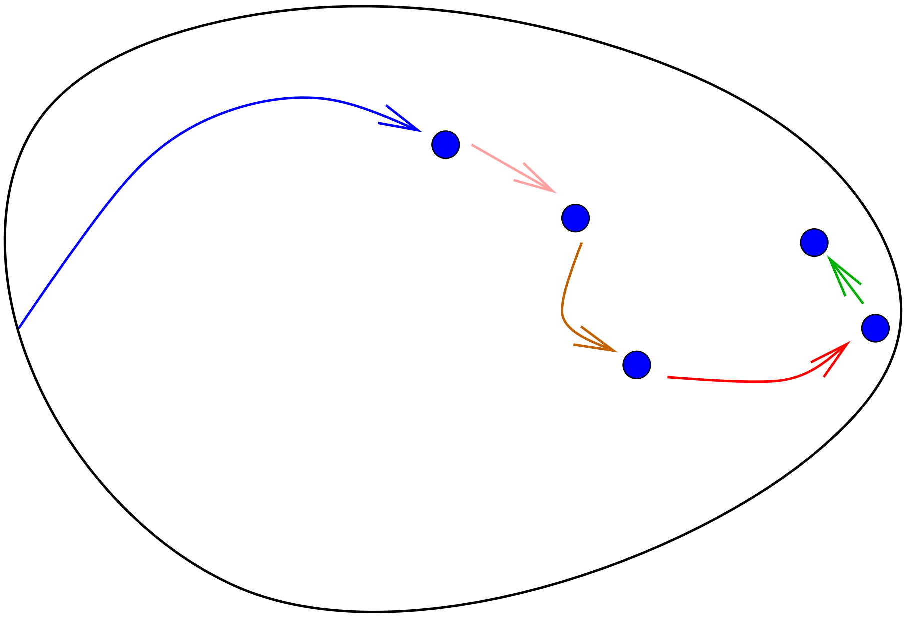
Problems Using OTS Model Checker as Test Generator

- Each test goal is treated separately: model checker is called repeatedly and performs much redundant work
- Test set has many short tests
  - Each incurs a startup cost during execution
  - Total length is large, so high execution cost
  - Much redundancy among the tests (wasteful)
  - Few long tests (so deep bugs undetected)
- Model checker may be unable to reach deep test goals

A Better Way

- Instead of starting each test from the the start state, we try to extend the test found so far
- Extending tests allows a bounded model checker to reach deep states at low cost
  - 5 searches to depth 4 much easier than 1 to depth 20
- Could get stuck if we tackle the goals in a bad order
- So, simply try to reach any outstanding goal and let the model checker find a good order
  - Can slice the model after each goal is discharged
  - A virtuous circle: the model will get smaller as the remaining goals get harder
- Go back to the start (or another earlier state) when unable to extend current test

An Efficient Test Set



Less redundancy, and longer tests tend to find more bugs

The SAL Automated Test Generator: sal-atg

- SAL is scriptable in Scheme
- `sal-atg` implements the method described in a few hundred lines of Scheme
  - (Re)starts use either symbolic or bounded model checking
    - ⋆ Parameterized choice and search depth
  - Extensions use bounded model checking
    - ⋆ Parameterized incremental search depth
  - Optional slicing after each extension or each restart
  - Customizable output to drive test harness

Example

- `sal-atg stopwatch clock stopwatch_goals.scm -ed 5 --incremental`
  In 5 seconds, generates single test case of length 17 that covers the states and transitions of the Statechart
- `sal-atg stopwatch clock stopwatch_goals.scm -ed 5 -id 0 --incremental --smcinit`
- Takes 106 seconds to cover flowchart as well: adds test of length 101 for middle junction and one of length 6,001 for jabr

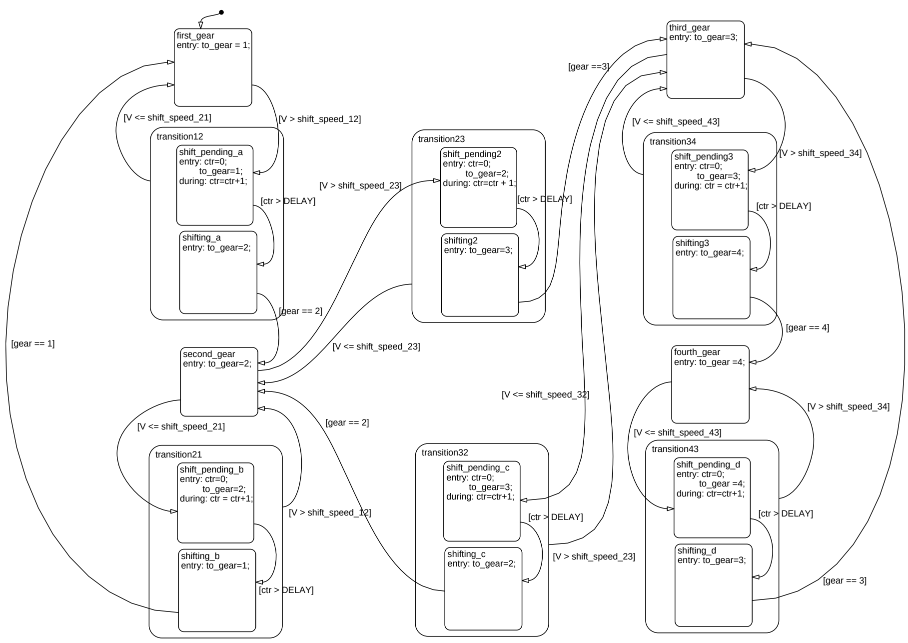
Experimental Results

- Rockwell Collins has developed a series of flight guidance system (FGS) examples for NASA
- SAL translation of largest of these kindly provided by UMN
- Model has 490 variables (576 state bits), 196 reachable control states, and 313 transitions
  - Takes 61 seconds to generate single test case of length 45 that covers all states
  - Takes 98 seconds to generate a single test of length 55 that covers all transitions
- Without extensions, get 73 tests to cover transitions: 1 of length 3, 9 of length 2, and the rest of length 1
  - Poor mutant detection
- We are in the process of testing our tests

Test Engineering with Automation

- Generating tests just to achieve structural coverage is a poor strategy
- Traditional test engineers develop tests to explore interesting cases, requirements, fault hypotheses
- We need to give them a way to do this using automation
- Specify the desired tests rather than constructing them
- Develop an observer module that sets a variable TRUE when a test has achieved some purpose
- Tell `sal-atg` to search for conjunction of each trap variable with the purpose
- In general, `sal-atg` can search for arbitrary conjunctions
  - E.g., product of structural coverage on control states and boundary coverage on some data structure

Example Shift Scheduler



Shift Scheduler

- One input is the gear currently selected by the gearbox
- Tests often change this discontinuously (e.g., 1, 3, 4, 2)
- Can easily establish the test purpose to change only in single steps, and to change at every step

Please Try It Out

- Main FM tools home page: http://fm.csl.sri.com
- SAL home page: http://sal.csl.sri.com
- SAL-atg (next week): http://sal.csl.sri.com/pre-release

Thoughts on Embedded Security

Philip Koopman
koopman@cmu.edu
http://www.ece.cmu.edu/~koopman

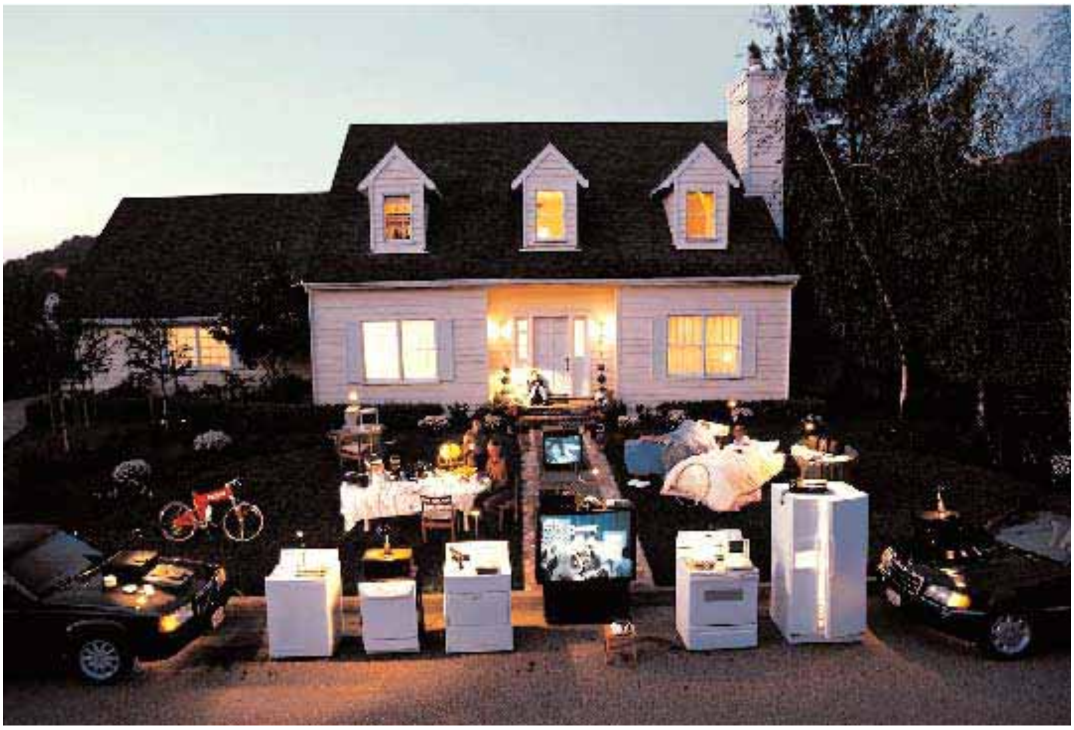
Carnegie Mellon

Electrical & Computer ENGINEERING

1

Small Computers Rule The Marketplace

- Everything here has a computer – but no Pentiums



Must We Worry About Security?

- **Consider the lowly thermostat**
  - Koopman, P., "Embedded System Security,“ *IEEE Computer*, July 2004.
- **Trends:**
  - Internet-enabled
  - Connection to utility companies for grid load management
- **Proliphix makes an Internet Thermostat**
  - (But it we’re not saying that system has these vulnerabilities!)

3

Waste Energy Attack

- **“I’m coming home” function**
  - Ability to tell thermostat to warm up/cool down house if you come home early from work, or return from a trip
  - Save energy when you’re gone; have a comfy house when you return
  - Implement via web interface or SMS gateway
- **Attack: send a false “coming home” message**
  - Causes increase in utility bill for house owner
  - If a widespread attack, causes increased US energy usage/cause grid failure
  - Easily countered(?) – if designers think to do it!
    - Note that playback attack is possible – more than just encryption of an unchanging message is required!

4

Discomfort Attack

- **Remotely activated energy saver function**
  - Remotely activated energy reduction to avoid grid overload
  - Tell house "I'll be home late"
  - Saves energy / prevents grid overload when house empty
- **Attack: send a false "energy saver" command**
  - Will designers think of this one?
  - Some utilities broadcast energy saver commands via radio
    - In some cases, air conditioning is completely disabled
    - Is it secure??
  - Consequences higher for individual than for waste energy attack
    - Possibly broken pipes from freezing in winter
    - Possibly injured/dead pets from overheating in summer

5

Energy Auction Scenario

- **What if power company optimizes energy use?**
  - Slightly adjust duty cycles to smooth load (pre-cool/pre-heat in anticipation of hottest/coldest daily temperatures)
  - Offer everyone the chance to save money if they volunteer for slight cutbacks during peak times of day
  - Avoid brownouts by implementing heat/cool duty cycle limits for everyone
- **You could even do real time energy auctions**
  - Set thermostat by "dollars per day" instead of by temperature
    - More dollars gives more comfort
  - Power company adjusts energy cost continuously throughout day
  - Thermostats manage house as a thermal reservoir

6

Direct Energy Auction Attacks

- **What if someone broke into all the thermostats?**
  - Set dollar per day value to maximum, ignoring user settings
    - Surprise! Next utility bill will be unpleasant
  - Turn on all thermostats to maximum
    - Could overload power grid
  - Pulse all thermostats in a synchronized way
    - Could synchronized transients destabilize the power grid?

7

Indirect Energy Auction Attack

- **What if someone just broke into the auction server?**
  - If you set energy cost to nearly-free, everyone turns on at once to grab the cheap power
  - Guess what – enterprise computer could have indirect control of thousands of embedded systems!
    - A key point is the computer's authority over release of energy
  - Someday soon, almost "everything" will be "embedded," at least indirectly

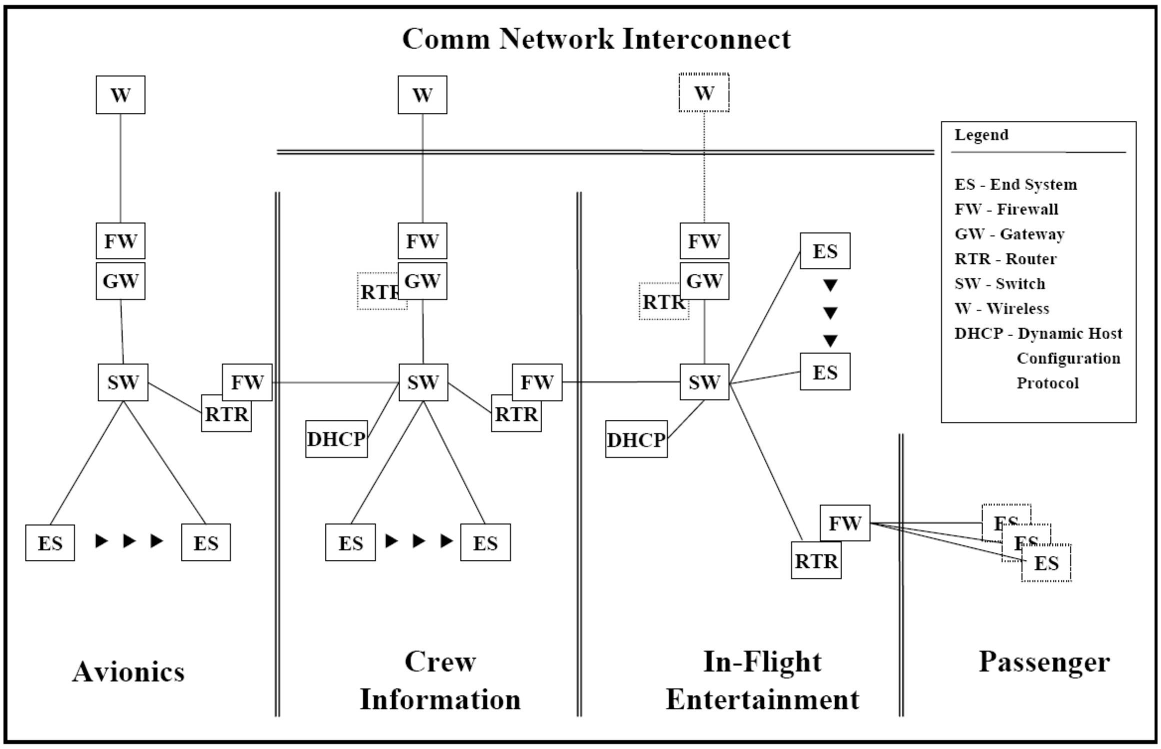
8

Could There Be Safety Critical Stuff Like This?

- **Medtronic pacemaker**
  - July 1, 2001 – VP Dick Cheney gets an Internet Pacemaker (Medtronic GEM® III DR)
  - Uses phone link to connect to secure web-based monitoring system, available to patient, physician, nurses, etc.
  - "Medtronic has taken significant measures to protect the confidentiality and security of patients' healthcare information. The company has partnered with technology experts to build a secure system that employs multiple levels of security and encryption technology. The system is designed to address healthcare privacy and security laws and regulations. Access for clinicians and patients requires registration and is password protected so that only registered users will have access to patient information." *http://www.medtronic.com/newsroom/news_20020102.html*

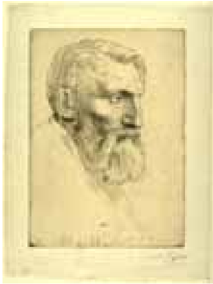
Vice President Dick Cheney looks relatively chipper after having a Medtronic defibrillator/pacemaker implanted in his shoulder.

9



Wargo & Chas, 2003, proposed Airbus A-380 architecture

10



RODIN
Rigorous Open Development Environment for Complex Systems

Specific Targeted Research Project , EU IST FP6

Brian Randell (on behalf of Sascha Romanovsky)
University of Newcastle upon Tyne, UK

January 2005

Participants

University of Newcastle upon Tyne, UK (Coordinator) - Sascha Romanovsky
Aabo Akademi University, Turku, Finland - Kaisa Sere
ClearSy System Engineering, France - Thierry Lecomte
Nokia Corporation, Finland - Colin Willcock
Praxis Critical Systems Ltd, UK - Adrian Hilton
VT Engine Controls Ltd, UK - John Brightman
Swiss Federal Institute of Technology, Zurich, Switzerland - Jean-Raymond Abrial
University of Southampton, UK - Michael Butler

***Start**: September 1, 2004*
***End**: August 31, 2007*
***Total cost:** 4,397,850.00 Euros*
***EC contribution:** 3,171,000.00 Euros*

***Web site:** rodin.cs.ncl.ac.uk*

January 2005

Industrial Interest Group

Adelard, UK
Alstom Transportation, France
AWE Aldermaston, UK
DGA, France
Escher Technologies, UK
Gemplus, France
IBM UK
I.C.C.C. Group, Czech Republic
QinetiQ, UK
RATP, France
STMicroelectronics, France
VTT, Finland

January 2005

Objectives

The overall objective is the creation of a methodology and supporting open tool platform for the cost-effective rigorous development of dependable complex software systems and services

Main Advances aimed for in:

- Formal Design Methods

- Fault Tolerance

- Design Abstractions

- Tool platform

January 2005

Formal Design Methods.

Mastering complexity requires design techniques that support clear thinking and rigorous validation and verification. **Formal design** methods do so.

Fault Tolerance.

Coping with complexity also requires architectures that are tolerant of faults and unpredictable changes in environment. This is addressed by **fault tolerance** design techniques.

Dependability consideration should start from the early stages of system development.

The aim is to deal with faults in the system environment, faults of the individual components, and component mismatches, as well as errors affecting several interacting components.

January 2005

Design Abstractions.

We will tackle complex architectures: our systems approach will support the construction of appropriate **abstractions** and provide techniques for their structured refinement and decomposition.

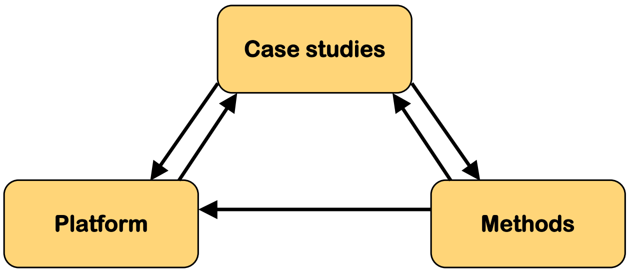
Tool platform.

Tool support for construction, manipulation and analysis of models is crucial and we will concentrate on a comprehensive **tool platform** which is openly available and openly extendable and has the potential to set a European standard for industrial formal method tools.

January 2005

Workpackages:

WP1. Research drivers (case studies)
WP2. Methodology
WP3. Open tool kernel
WP4. Modelling and verification plug-ins
WP5. Dissemination and exploitation
WP6. Project management
WP7. Project review and assessment



January 2005

WP1. Research drivers

The methods and platform will be validated and assessed through industrial case studies:

Case study 1: Formal Approaches to Protocol Engineering (Nokia)
Case study 2: Engine Failure Management System (VT Engine Controls)
Case study 3: Formal Techniques within an MDA Context (Nokia)
Case study 4: CDIS Air Traffic Control Display System (Praxis)
Case study 5: Ambient Campus (U. of Newcastle)

January 2005

WP2. Methodology

To produce the RODIN methodology for rigorous development of complex systems.

To make advances in the basic research areas related to formal system modelling and mapping of models, software reuse, and formal reasoning about system fault tolerance, reconfiguration, mobility and adaptivity.

This includes development of templates for fault tolerant design methods (exception handling, atomic actions, compensation), as well as for reconfigurability, adaptivity and mobility.

WP3. Open tool kernel

To develop a set of *basic kernel tools* implemented on a certain *platform container* that can be extended by the *plug-ins* being developed in WP4.

Openness of the platform is the prime aim.

Generality of the platform.

Based on the use of *Eclipse*.

WP4. Modelling and verification plug-ins

To develop a range of tools to support the application of the RODIN methodology being developed in WP2.

1. **Linking UML and B**
2. **Petri net-based model checking**
3. **Constraint-based model checking and animation**
4. **Model-based testing**
5. **Code Generation**

January 2005

Novel Aspects

- **pursuit of a systems approach**
- **combination of formal methods with fault tolerance techniques**
- **development of formal method support for component reuse and composition**
- **provision of an open and extensible tools platform for formal development**

January 2005

Expected Project Results

A collection of reusable development templates (models, architectures, proofs, components, etc.) produced by the case studies

A set of guidelines on a systems approach to the rigorous development of complex systems, including design abstractions for fault tolerance and guidelines on model mapping, architectural design and model decomposition

An open tool kernel supporting extensibility of the underlying formalism and integration of tool plug-ins

A collection of plug-in tools for model construction, model simulation, model checking, verification, testing and code generation

January 2005

RODIN Presentations to date

I. Johnson, C. Snook, A. Edmunds & M. Butler
Rigorous development of reusable, domain-specific components, for complex applications.
***CSDUML'04 - 3rd International Workshop on Critical Systems Development with UML*, October 2004, Lisbon**

C. Schröter, V. Khomenko.
Parallel LTL-X Model Checking of High-Level Petri Nets Based on Unfoldings.
***Proc. CAV'2004*, Alur, R. and Peled, D.A. (Eds.). Springer-Verlag, Lecture Notes in Computer Science 3114. 2004. pp. 109-121.**

January 2005

Relevant Prior Publications

J.-R. Abrial. *The B-Book: Assigning programs to meanings*. Cambridge University Press,.1996.

A. Avizienis, J.-C. Laprie, C. Landwehr, B. Randell. Basic Concepts and Taxonomy of Dependable and Secure Computing. *IEEE Trans. on Dependable and Secure Computing*. 1, 1, 2004.

M. J. Butler. Stepwise Refinement of Communicating Systems. *Science of Computer Programming*, 27, 1996.

M.C. Gaudel, V. Issarny, C. Jones, H. Kopetz, E. Marsden, N. Moffat, M. Paulitsch, D. Powell, B. Randell, A. Romanovsky, R.J. Stroud, F. Taiani. *Final Version of DSoS Conceptual Model (CSDA1).* CS-TR: 782, School of Computing Science, University of Newcastle, July 2003.

C. Jones, A formal basis for some dependability notions. In *Proceedings of the 10th Anniversary Colloquium of UNU/IIST Formal Methods at the Crossroads: From Panacea to Foundational Support,* Lisbon, Portugal, 2002 Aichernig, B.K. and Maibaum, T. (Eds.) LNCS 2757. 2003.

C. Jones. *Systematic Software Development using VDM*. 1990.

M. Leuschel, M. Butler. ProB: A Model-Checker for B. *Proc. FM 2003: 12th Intl. FME Symposium*. Pisa, September, LNCS 2805, 2003.

A. Romanovsky, C. Dony, J.L. Knudsen, A. Tripathi (Eds.). Advances in Exception Handling Techniques, *LNCS-2022*, 2001.

K. Sere, E. Troubitsyna. Safety Analysis in Formal Specification. In J. Wing, J. Woodcock, J. Davies (Eds.), *FM'99 - Formal Methods. Proc. of World Congress on Formal Methods in the Development of Computing Systems*, Toulouse, France, LNCS 1709, 1999.

January 2005

Since September 2004

Kick-off meeting:

October 4-6, 2004. Newcastle upon Tyne

Work to date:

- Defining the evaluation criteria and traceable requirements documents for the case studies

- Making final decisions on RODIN platform architecture

- Finalising Event B language

January 2005

Originally given at Dagsthul Seminar on Atomicity, April 2004

On Detours and Shortcuts to solve distributed systems problems

Paulo Esteves Veríssimo

Navigators Group,
LaSIGe, Laboratory for Large-Scale Informatic Systems
Univ. Lisboa
pjv@di.fc.ul.pt
http://www.di.fc.ul.pt/~pjv

FC/UL

Problem Motivation

- Design and deployment of distributed applications is faced with the confluence of antagonistic aims:
  - between what is required by applications, and what is given by the supporting infrastructure/ environment
- Current and future large, massive-scale pervasive and/or ubiquitous computing systems will amplify this:
  - very high numbers of players, very large distances, geographical scope, topology and interconnections no longer a given, ill-defined COTS component properties
- Key lies with a changing notion of service guarantees:
  - on what have always been the fundamental issues, e.g., consistency, synchronism, reliability, availability, predictability, security, ...

Problem Motivation

- Take the security dimension
- Many services, beyond mere performance, have to enjoy security properties
- So we should prevent any security breaches
  - But we cannot prevent or detect all attacks/vulnerabilities
  - Even if we could, this would be impractical or too expensive
- Then what if we tolerate them?
  - But it is hard to define a fault model for a hacker...

Grand challenges put by this scenario?

withstanding uncertainty whilst achieving predictability

- Uncertainty:
  - is a common denominator of current systems
  - uncertain synchrony, fault model, and even topology
- Predictability:
  - systems are required to fulfill more and more demanding goals which imply predictability or determinism, e.g, timeliness, security
- Reconciling them means:
  - strong attributes (e.g. on ordering, agreement, timely termination of algorithms) can be secured in settings where usually very little is assumed and very little is expected from
  - current view has been to weaken attributes down to the little that one can expect to get from uncertain environments

The usual path

- If you want efficient/performant solutions to F/T
  - assume controlled failure modes (omissive, fail-silent, etc.)
- If you want to build timely services (even soft RT)
  - assume synchronous models, or at least partially sync
- They only work to the coverage of the assumptions
  - which must be substantiated, else we risk pitfalls such as the "well-behaved hacker" syndrome

Taking detours...

- OBJECTIVE:
  - solve most non-timed problems with highest possible coverage
- tone down determinism
- tone down liveness expectations
- use weaker semantics than ABCAST/Consensus
- tone down allowed fault severity
- OBJECTIVE:
  - solve timed problems with highest possible coverage
- sync, parsync models (coverage ☹)

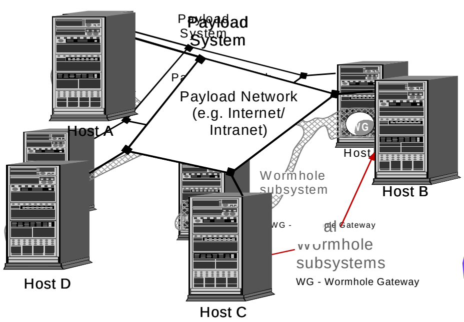
Shortcuts vs. detours

- we propose to render the solution simpler (without changing the problem!)

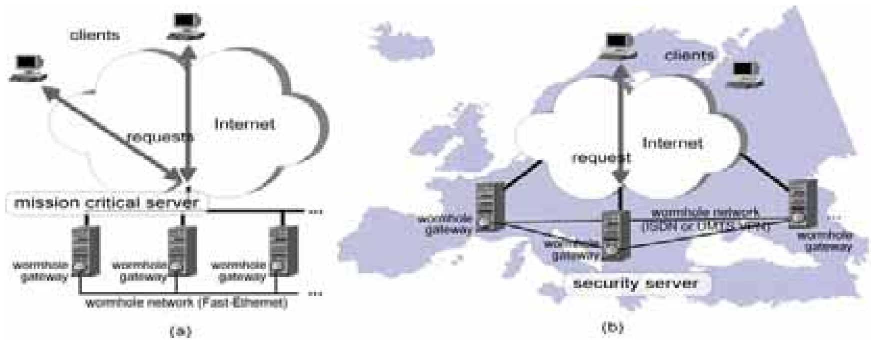
- Architectural hybridization
- Wormholes model

Wormholes

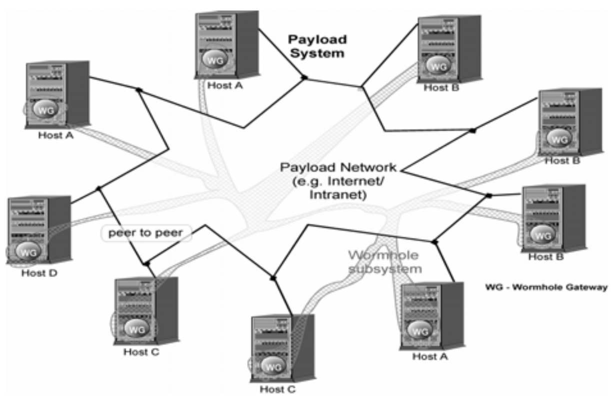
- New design philosophy for architecting and programming distributed systems:
- constructs with privileged properties that endow systems with the capability of evading the uncertainty or weakness of the environment (``taking a shortcut'') for certain crucial steps of their operation, in order to achieve overall strong properties otherwise impossible or complex or expensive



Example of deployment of systems with wormholes



Example of deployment of systems with wormholes



Taking shortcuts i.s.o. detours

- OBJECTIVE:
  - solve most timed or non-timed problems with highest possible coverage
- enforce hybrid behaviour ("strong" and "weak" components) by *architectural hybridization*
- implement strong q.b. components (*trusted-trustworthy*)
- overcome algorithmic hardness (e.g., w.r.t. asynchronism, maliciousness, etc.) through computing models aware of the above (e.g. *Wormholes*)

A (necessarily brief) birds-eye view of some results

Navigators FC/UL

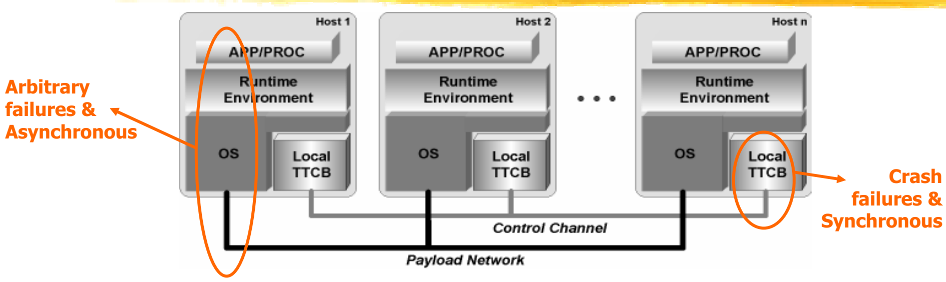
Trusted Timely Computing Base (TTCB)

- **Properties:**
  - trusted and timely execution assistant; trusted timing failure detector
  - secure (can only fail by crashing)
  - real-time (capable of timely behavior)
  - correct processes can interact securely with the TTCB
- **Assists the execution of fault-tolerant algorithms:**
  - provides a trusted environment for crucial steps
- **Can be built (there is a prototype)**

Correia, Veríssimo, and Neves. The Design of a COTS Real-Time Distributed Security Kernel. European Dependable Computing Conf., *EDCC-4*, October 2002

Navigators FC/UL

System Model



- **TTCB is a distributed security kernel that provides a minimal set of trusted and timely services, such as**
  - local authentication
  - agreement on a fixed sized block of data (TBA)
  - globally meaningful timestamps

Navigators FC/UL

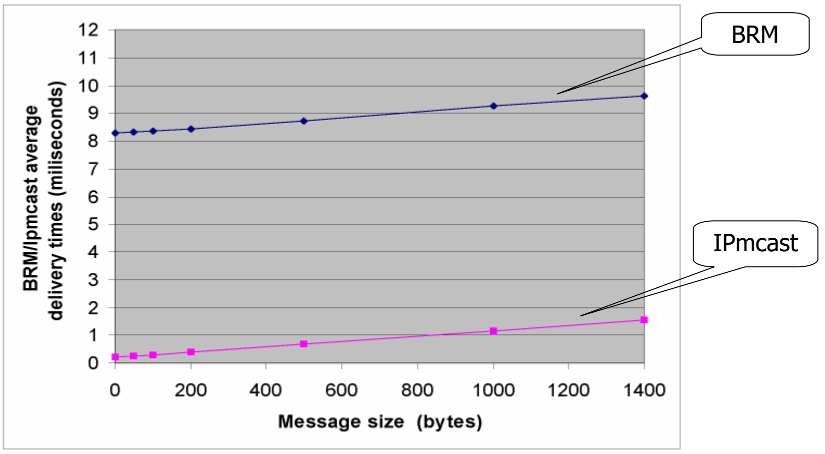
Efficient Byzantine-Resilient Reliable Multicast on a Hybrid Fault Model

www.navigators.di.fc.ul.pt/docs

Correia, M., Lung, L.C., Neves, N.F., Veríssimo, P.: Efficient Byzantine-Resilient reliable multicast on a hybrid failure model. In: Proc. of the 21st Symposium on Reliable Distributed Systems, Suita, Japan (2002)

Navigators FC/UL

Measurements



Typical values in earlier works: ~50ms

Navigators FC/UL

Conclusion

- **Reliable multicast with Byzantine faults requires:**
  - asynchronous system: $n \geq 3f+1$ [Bracha&Toueg]
  - synchronous system: no limit ($n \geq f+2$) [Lamport et al.]
- **We follow a wormhole-aware model:**
  - payload is asynchronous and byzantine-on-failure
  - TTCB is synchronous and crash-on-failure
- **We achieve:**
  - $n \geq f+2$ without asymmetric crypto (signatures)
  - Efficiency: few phases, high performance

Navigators FC/UL

Low Complexity Byzantine-Resilient Consensus

Distributed Computing Journal, 2004/2005

Navigators FC/UL

Termination & FLP result

- **FLP result:**
- ***impossible to deterministically solve consensus in an asynchronous system***
- **Usual solutions:**
- ***randomization, weak synchronous assumptions (e.g., partial synchronous models or unreliable failure detectors)***
- **Our approach:**
- ***avoid violation of safety properties***
- ***ensure termination by finding a way to circumvent the FLP impossibility result***
- **Our assumption**

  eventually there will be a round where at least 2f+1 processes manage to locally call the TTCB on time

Navigators FC/UL

Performance Comparison

- **Use *latency degree* [Schiper 97] criteria extended to include current implementation of TTCB agreement**

Protocol	Latency degree	Requirements
Dwork et al.	**7**	
Dwork et al.	**4**	**signed messages**
Malhki & Reiter	**9 or 6**	**signed messages**
Kihlstrom et al.	**4**	**signed messages**
Block consensus	**1**	**TTCB**
General consensus	**1 or 2**	**TTCB**

Navigators FC/UL

Solving Vector Consensus with a Wormhole

submitted

Navigators FC/UL

Our approach in the FLP scene

- **FLP result:**
  - ***impossible to deterministically solve consensus in an asynchronous system***
- **Usual solutions:**
  - ***randomization, weak synchronous assumptions (e.g., partial synchronous models or unreliable failure detectors)***
- **Our approach:**
  - ***avoid violation of safety properties***
  - ***ensure termination by finding a way to circumvent the FLP impossibility result***
- **Our assumption**

  the algorithm running on the payload is fully asynchronous

Navigators FC/UL

Performance Comparison

- **Use *latency degree* [Schiper 97] criteria extended to include current implementation of TTCB agreement**

Protocol	LatDeg	MSign	GVer	Artifact
DS [15]	5	5	3	Failure detectors
BHRT [1]	3	3	2	Failure detectors
Our protocol	4	1	1	Wormhole

	Crash			Byzantine			
Protocol	LatDeg	MSign	GVer	LatDeg	MSign	GVer	SDeg
DS [15]	5 + 2f	5 + 2f	3 + f	5 + 2f	5 + 2f	3 + f	f
BHRT [1]	3 + f	3 + f	2 + f	3 + f	3 + f	2 + f	f
Our protocol	4	1	1	4 + 2f	1	1 + f	0

Navigators FC/UL

Main Achievements

- **Fully asynchronous payload algorithm**
- **Low complexity**
- **Consensus without FDs:**
- ***Instead failure detectors, uses low level agreement service***
- ***Does not exclude processes, uses all processes that behave correctly at any given time***
- ***Difficult to construct failure detectors in Byzantine systems***
- ***Reliable Byzantine failure detection: an open problem***

Navigators FC/UL

Worm-IT : group communication system for a Byzantine asynchronous environment

submitted

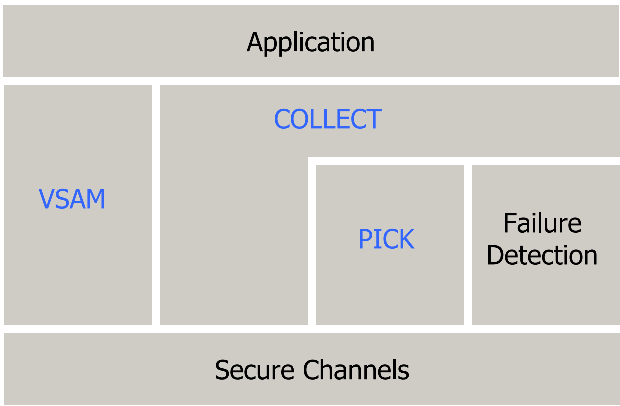
Navigators FC/UL

Worm-IT

- **A group communication system for a Byzantine asynchronous environment**
  - Dynamic Membership Service
  - View-Synchronous Atomic Multicast
- **Intrusion tolerant**
- **The system uses a wormhole that offers a few secure and timely operations**
  - Trusted Timely Computing Base
- **Resilience: f out of 3f+1 (optimal for asynchronous systems)**

Navigators FC/UL

Protocol Stack



Navigators FC/UL

Main Achievements

- Exemplifies how a reasonably complex system can be built with a wormhole
- Make decisions in a distributed way
- Good performance since it does not resort to public key cryptography

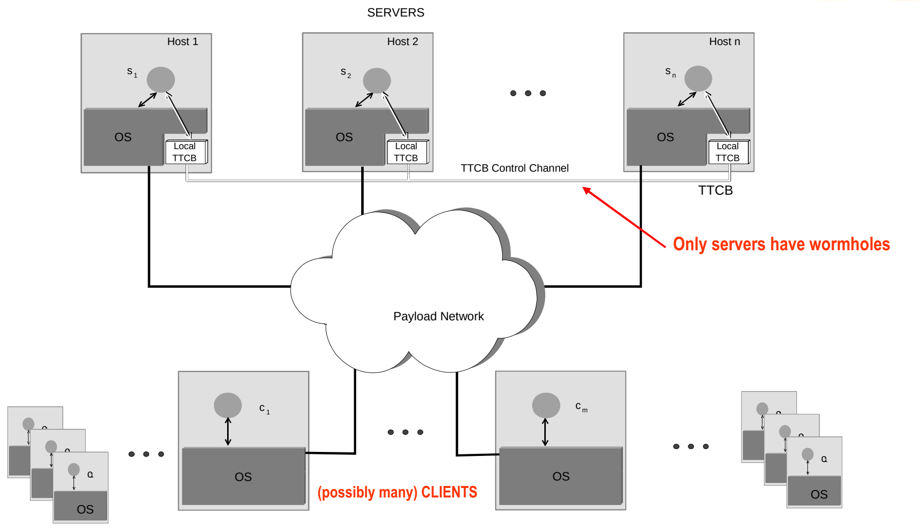
Navigators FC/UL

State machine replication on atomic multicast

IEEE SRDS 04, Florianopolis, Brasil 2004

Navigators FC/UL

System architecture



Navigators FC/UL

Main Achievements

- **First SMA service for practical byzantine distributed systems with resilience f out of 2f+1**
  - Lower number of replicas reduces cost of hardware + cost of designing different replicas (for fault independence)
- **Low time complexity**
- **Probable good performance since it does not resort to public key cryptography**

Navigators FC/UL

Some Recent Publications (urls)

- ***Modeling Wormholes***
- *Uncertainty and Predictability: Can they be reconciled?* Paulo Veríssimo. **Future Directions in Distributed Computing, pages to appear, Springer-Verlag LNCS 2584, month to appear, 2003**
- *The Timely Computing Base Model and Architecture*. Paulo Veríssimo, António Casimiro. **IEEE Transactions on Computers - Special Issue on Asynchronous Real-Time Systems, vol. 51, n. 8, Aug 2002**
- *The Timely Computing Base: Timely Actions in the Presence of Uncertain Timeliness*. Paulo Veríssimo, António Casimiro, C. Fetzer. **In Proceedings of the 1st International Conference on Dependable Systems and Networks, New York, USA, June 2000.**
- **The Timely Computing Base. Paulo Veríssimo and António Casimiro.** Technical Report DI/FCUL TR 99-2, Department of Informatics, University of Lisboa, **May 1999.** ***(original paper, improved in TOCS02)***
- ***Implementing Wormholes***
- *Measuring Distributed Durations with Stable Errors*. António Casimiro, Pedro Martins, Paulo Veríssimo, Luís Rodrigues. **Proceedings of the 22nd IEEE Real-Time Systs Symposium, London, UK, December 2001**
- *How to Build a Timely Computing Base using Real-Time Linux*. António Casimiro, Pedro Martins, Paulo Veríssimo. **in Proceedings of the 2000 IEEE International Workshop on Factory Communication Systems, Porto, Portugal, September 2000.**
- *Timing Failure Detection with a Timely Computing Base*. António Casimiro, Paulo Veríssimo. **3rd Europ. Research Seminar on Advances in Distr. Sys (ERSADS'99), Madeira Island, Portugal, April 23-28, 1999**
- *The Design of a COTS Real-Time Distributed Security Kernel, Miguel Correia, Paulo Veríssimo, Nuno Ferreira Neves,* ***Fourth European Dep. Comp. Conf., Toulouse, France, October 2002 © Springer-Verlag.***

Navigators FC/UL

Some Recent Publications (urls)

- ***Using Wormholes***
- ***Using the Timely Computing Base for Dependable QoS Adaptation***. António Casimiro, Paulo Veríssimo. **Proceedings of the 20th IEEE Symp. on Reliable Distributed Systems, New Orleans, USA, October 2001**
- ***Generic Timing Fault Tolerance using a Timely Computing Base***. António Casimiro, Paulo Veríssimo. **Procs of the Intern'l Conference on Dependable Systems and Networks, Washington D.C., USA, June 2002**
- ***Efficient Byzantine-Resilient Reliable Multicast on a Hybrid Failure Model***, *Miguel Correia, Lau Cheuk Lung, Nuno Ferreira Neves, Paulo Veríssimo.* ***Proc's of the 21st Symp. on Reliable Distributed Systems (SRDS'2002), Suita, Japan, October 2002***
- ***How to Tolerate Half Less One Byzantine Nodes in Practical Distributed Systems***
  Miguel Correia, Nuno Ferreira Neves, Paulo Veríssimo
  In Proceedings of the 23rd IEEE Symposium on Reliable Distributed Systems. Florianopolis, Brasil, pages 174-183, October 2004
- ***Low Complexity Byzantine-Resilient Consensus***
  Miguel Correia, Nuno Ferreira Neves, Paulo Veríssimo, Lau Cheuk Lung
  Distributed Computing, Accepted for publication, 2004. On-line first: http://www.springerlink.com/index/10.1007/s00446-004-0110-7

Navigators FC/UL

- **Navigators group:**
  - http://www.navigators.di.fc.ul.pt/

Session 2

Moderator

Jean Arlat, LAAS-CNRS, Toulouse, France

Carnegie Mellon

My Background

- **Prior research on dependable enterprise systems**
  - Developed systems that provide "out-of-the-box" reliability to CORBA/Java applications
    - No need to change application or ORB code
  - Eternal: Fault-tolerant CORBA/Java support
  - Immune: Secure CORBA/Java support
- **Helped to establish Fault-Tolerant CORBA standard and founded company to sell fault-tolerant products based on my PhD research**
- **Lessons learned [IEEE TOCS 2004]**
  - It's hard for users to (re)configure the fault-tolerance of their systems to suit the applications' needs
  - There needs to be a way of mapping high-level user requirements to low-level implementation mechanisms

2

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Motivation for MEAD

- **Middleware is increasingly used for applications, where dependability and quality of service are important**
  - Fault-Tolerant CORBA and Fault-Tolerant Java standards
- **But ……**
  - These standards provide a laundry list of "fault-tolerance properties"
  - No insight into how these properties ought to be set
  - No insight into how fault-tolerance and fault-recovery can be configured to meet an application's performance or reliability requirements
- **One focus of MEAD**
  - Providing advice on configuring fault-tolerance for distributed applications
  - Being able to determine this configuration at deployment-time
  - Being able to re-determine and enforce configurations at runtime
  - Being able to perform (re)configuration proactively, where possible
  - Middleware merely a vehicle for exploring proactively configurable fault-tolerance

3

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Research Focus

- **Overall objectives of the MEAD system**
  - Automated, adaptive (re)configuration of fault-tolerance [WADS 2004]
  - Proactive fault-recovery for distributed applications [DSN 2004]
    - Exploiting system information for faster recovery
  - Static analysis of application and middleware code to extract application-level insights and communicate them to the MEAD runtime [SRDS 2004]
  - Zero-downtime, live upgrades of the application
    - Dependency tracking at runtime and development-time
    - Staggered quiescence of different parts of the system
- **Target applications**
  - Embedded printing applications (HP Labs)
  - Unmanned aerial vehicles (BBN & Boeing)
  - Shipboard computing platforms (Raytheon & Lockheed Martin)
  - Automotive telematics systems (General Motors)

4

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

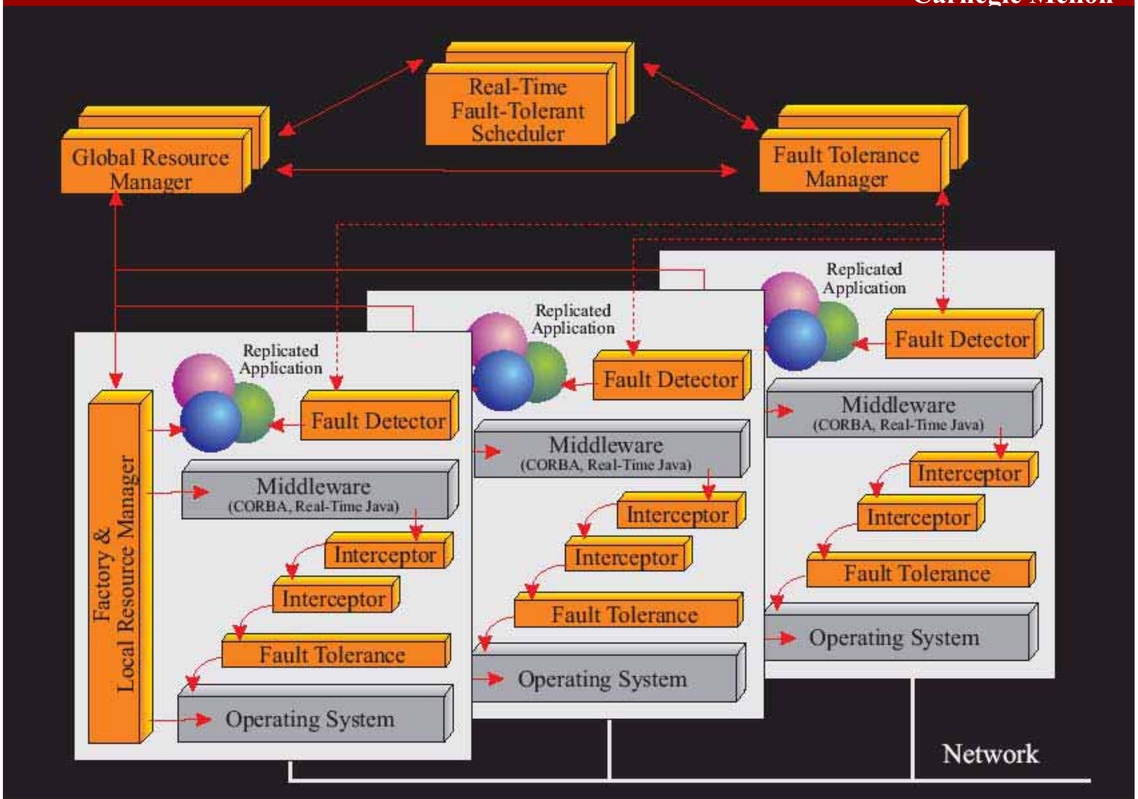
And Now For Something Completely Different ....

- **Why MEAD?**
- **Legendary ambrosia of the Vikings**
- **Believed to endow its imbibers with**
  - Immortality (⇨*dependability*)
  - Reproductive capabilities (⇨*replication*)
  - Wisdom for weaving poetry (⇨*cross-cutting aspects of performance and fault tolerance*)

5

MEAD: Middleware for Embedded Adaptive Dependability

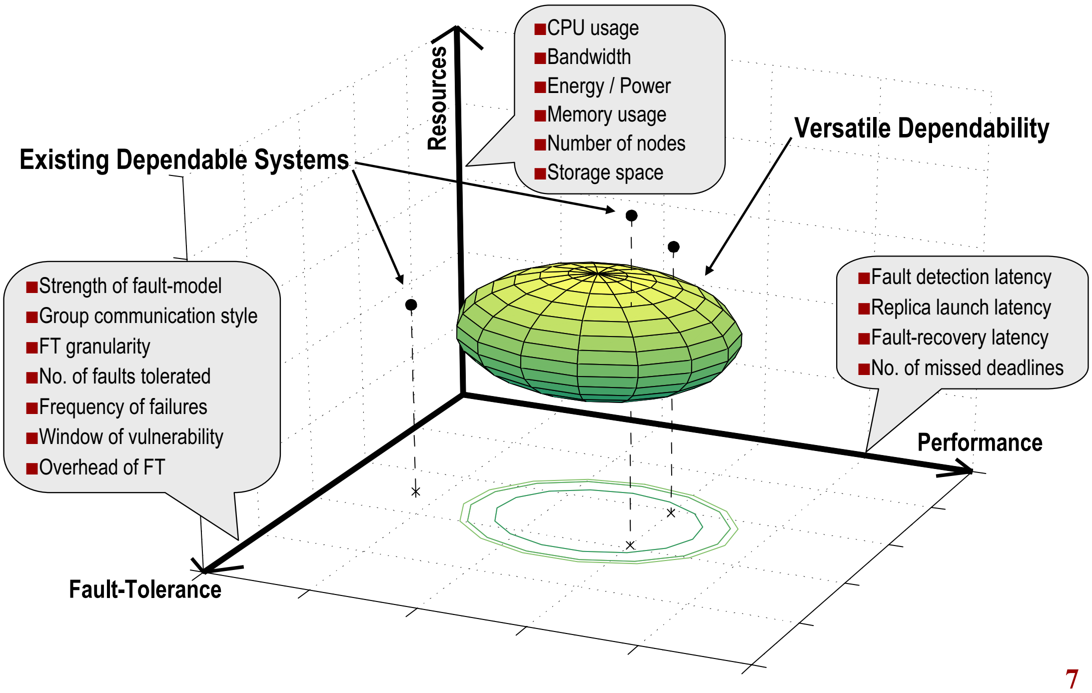
Carnegie Mellon



MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Versatile Dependability



7

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

“Knobs” of the MEAD System

- **Application**
  - Frequency of requests
  - Size of requests/responses
  - Size of State
  - Application Resources

High Level Knobs

- **External Properties**
  - Scalability
  - Availability
  - Real-Time Guarantees

Low Level Knobs

- **Fault-tolerance infrastructure (MEAD)**
  - Replication (Active, Passive)
  - Number of Replicas
  - Checkpointing Frequency

8

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Fault-Tolerance Advisor

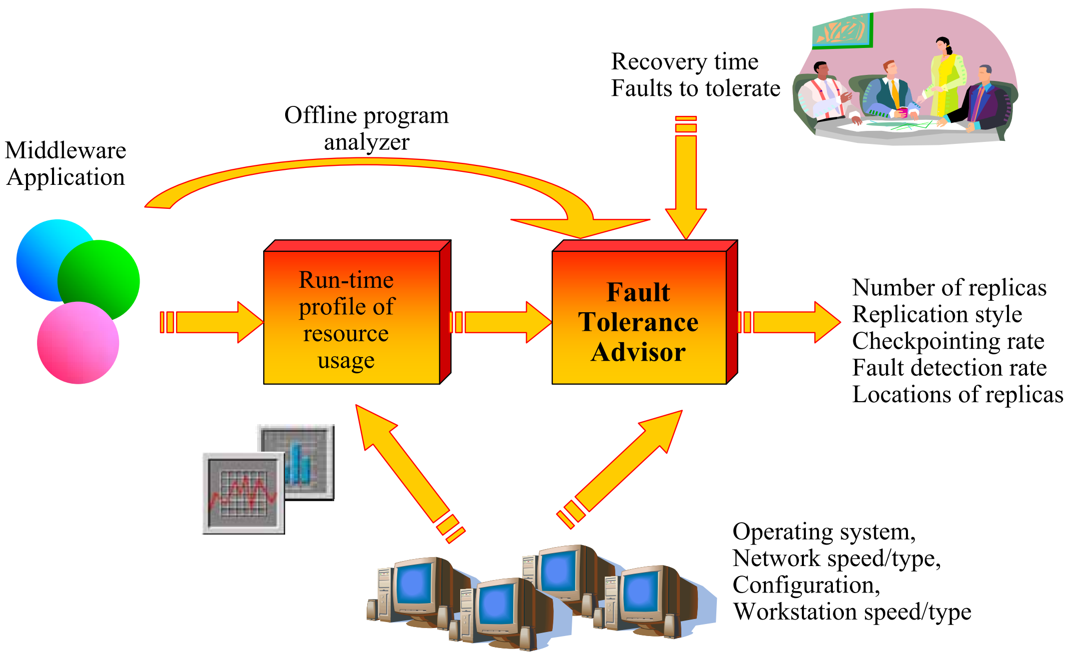
- **Configuring fault tolerance today is mostly ad-hoc**
- **To eliminate the guesswork, we deployment/run-time advice on**
  - Number of replicas
  - Checkpointing frequency
  - Fault-detection frequency, etc.
- **Input to the Fault-Tolerance Advisor**
  - Application characteristics (through program analysis)
  - System reliability characteristics
  - System's and application's resource usage
- **Fault-Tolerance Advisor works with other MEAD components to**
  - Enforce the reliability advice
  - Sustain the reliability of the system, in the presence of faults

9

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Fault-Tolerance Advisor

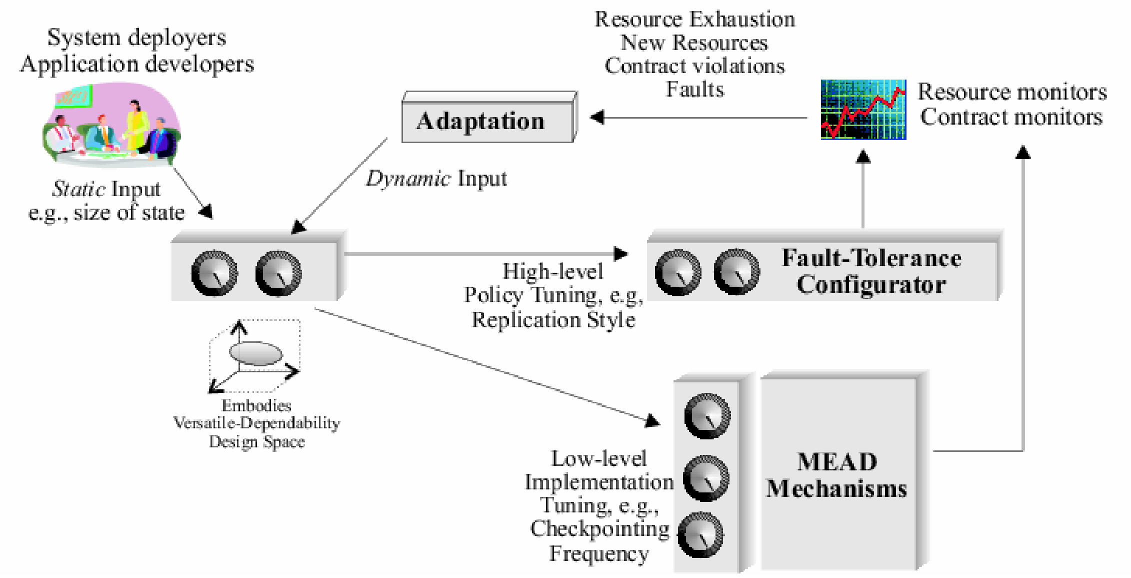


10

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Run-Time Adaptation



11

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Mode-Driven Fault-Tolerance Adaptation

- **Most applications have multiple modes of operation**
  - Example: the unmanned aerial vehicle (UAV) application exhibits
    - Surveillance mode
    - Target recognition mode
- **Each mode might require different fault-tolerance mechanisms**
  - The critical elements in the path might differ
  - The resource usage might differ, e.g., more bandwidth used in some modes
  - The notion of distributed system “state” might be different
- **MEAD aims to provide the “right mode-specific fault-tolerance”**
  - Based on the Fault-Tolerance Advisor’s inputs
  - In response to (omens heralding) mode changes

12

MEAD: Middleware for Embedded Adaptive Dependability

Proactive Fault-Tolerance

- **Involves predicting, with some confidence, when a failure might occur, and compensating for the failure even before it occurs**
  - For instance, if we knew that a processor had an 80% chance of failing within the next 5 minutes, we could perform process-migration
- **Our goal in MEAD is to**
  - Lower the impact faults have on real time schedules
  - Implement proactive dependability in a transparent manner
- **Proactive dependability has two aspects:**
  - Fault prediction: Reducing the unpredictable nature of faults
  - Proactive recovery: Reducing fail-over times and number of failures experienced at the application-level (primary focus in MEAD)
- **Complements, but does not replace, the classical reactive fault-tolerance schemes since we cannot predict every fault**

Carnegie Mellon

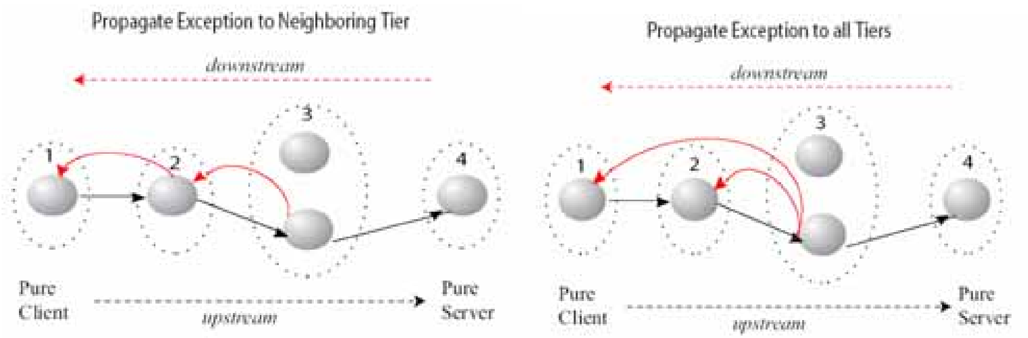
Benefits

- **Provides a framework for proactive recovery that is transparent to the client application**
- **Proactive recovery can**
  - Significantly reduce failover times, lowering the impact of a failure on real-time schedules
  - Reduce the number of failures experienced at the application level
  - Exploit knowledge of system topology to provide advance warning of failures to other servers “further down the line” (multi-tiered applications)
  - Request the recovery manager to launch new replicas so that a consistent number of replicas are retained in the group (useful for active replication where a certain number of servers are required to reach agreement)
- **Caveat**
  - Not applicable to every kind of fault, of course

Carnegie Mellon

Ongoing: Topology-Awareness

- **Curbing the spread of propagating faults or invoking faster recovery based on**
  - System topology,
  - Application's interconnections,
  - Application's normal fault-free behavior
- **Could also help sequence recovery actions across nodes**



·motivation ·**architecture** · evaluation ·future directions 15

MEAD: Middleware for Embedded Adaptive Dependability

Carnegie Mellon

Ongoing: Live Software Upgrades

- **Live software upgrades**
  - Software upgrades currently involve downtime ("scheduled maintenance")
  - Also, can cause a cascade of upgrades rippling through the system
- **Development-time preparation for live upgrades**
  - Exploiting program analysis
  - Identify the state before and after the upgrade, and the transition path
  - Prepare the application for upgrades
  - Identify potential points for scheduling upgrades
  - Building component-based applications to be born upgradeable
- **Runtime handling of live upgrades**
  - Determining quiescence
  - Run-time dependency tracking in a distributed system
  - Staggering out upgrades without incurring downtime

16

MEAD: Middleware for Embedded Adaptive Dependability

Looking Ahead …….

- **OMG (CORBA standards body) in the process of drafting an RFP for RT-FT middleware**
- **Consider performance, configurability and fault-tolerance**
  - To avoid point solutions that might work well, but only for well-understood applications, and only under certain constraints
  - To allow for systems that are subject to dynamic conditions, e.g., changing constraints, new environments, overloads, faults, ……
- **Expose interfaces that support the**
  - Capture of the application's fault-tolerance and timing needs
  - Tuning of the application's fault-tolerance configurations
  - Query of the provided "level" of fault-tolerance and quality-of-service
  - Scheduling of fault-tolerance activities (fault-recovery)

Current Release of MEAD

- **Features**
  - Active replication, warm passive replication, resource monitoring
  - Focus on CORBA applications (upcoming – CCM and EJB)
  - Tunable parameters: number of replicas, replication style, checkpointing frequency
- **Obtaining MEAD**
  - `/groups/pces/uav_oep/mead_cmu/release/` on `users.emulab.net`
- **MEAD User Support**
  - Manual: http://www.ece.cmu.edu/~mead/release/index.html
  - Problem-reporting
    - http://www.ece.cmu.edu/~mead/release/mead-support-request.html
  - You can also email us at mead-support@lists.andrew.cmu.edu

Teaching Students These Skills

- **Mixed class of students – software engineering, electrical engineering, computer science**
- **Semester-long project – pick a middleware platform (CORBA, J2EE, .NET, …..)**
- **Baseline**
  - Distributed application with reliability, scalability and timing requirements
- **Fault-tolerant baseline**
  - Evaluate the fault-tolerance (as compared with the non-fault-tolerant version)
- **"Real-time" fault-tolerant baseline**
  - Make the fault-tolerant baseline application exhibit timing/latency guarantees
- **Scalable real-time fault-tolerant final system**
  - Make your fault-tolerant real-time baseline application maintain performance, even with 1000 threads, 100 processes, etc.
- **Understand the fault-tolerance vs. real-time vs. performance trade-offs**
- **http://www.ece.cmu.edu/~ece749**

Carnegie Mellon

Summary

- **MEAD's configurable fault-tolerance**
  - Born out of lessons learned in deploying previous fault-tolerant systems
- **Advisor to take the guesswork out of configuring fault-tolerance**
- **"Knobs" for the appropriate expression of a user's requirements**
- **Offline program analysis to extract application-level knowledge**
- **Proactive fault-recovery mechanisms**

Carnegie Mellon

Electrical & Computer ENGINEERING

For More Information

http://www.ece.cmu.edu/~mead

Tudor Dumitras, Aaron Paulos, Soila Pertet, Charlie Reverte, Joe Slember, Deepti Srivastava

21

MEAD: Middleware for Embedded Adaptive Dependability

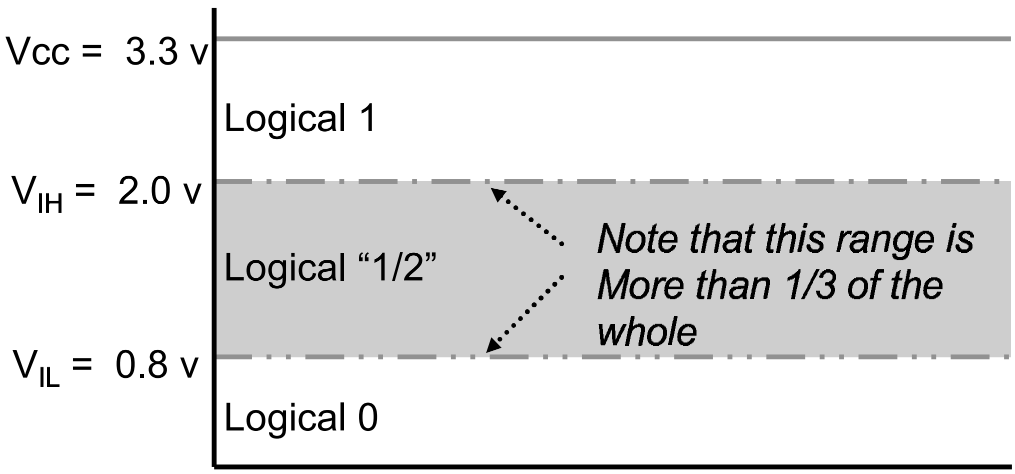
Byzantine Filtering

- From WG10.4 in Siena
  (and our 2003 SAFECOMP and 2004 IEEE DASC papers)
  - Byzantine fault propagation "physics" and example
  - Combating Byzantine Generals' fault propagation
    - Masking (blocks Byzantine signals via dominant logic)
      - Two-of-Three voter example
      - Can be done only with ***completely*** independent sources
        (***completely*** independent sources are very rare)
    - Filtering (converts a Byzantine signal to non-Byzantine)
      - Buried within all real Byzantine tolerance mechanisms
      - Needs to be tested to determine coverage
  - Byzantine filter testing idea
  - But, can this be done with a practicable number of tests?
    - How can proof-of-coverage testing be reduced?
      - An answer that reduces amplitude test range NEW
- Braided Ring: A network to exploit Byzantine filtering NEW

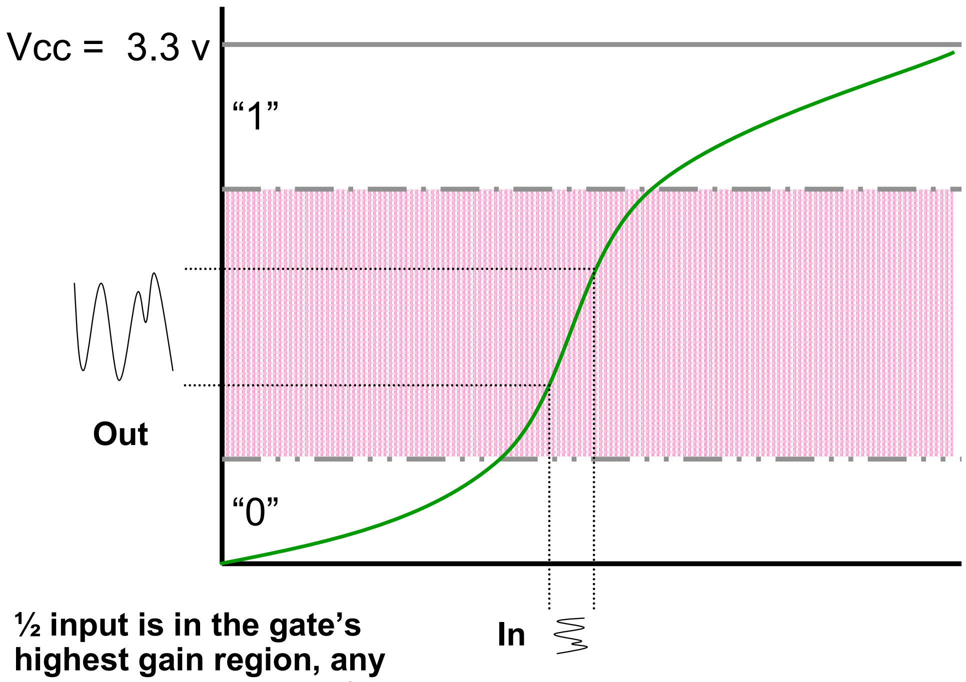
Digital Circuitry Behavior

There is no such thing as digital circuitry, …
there is just analog circuitry driven to extremes.

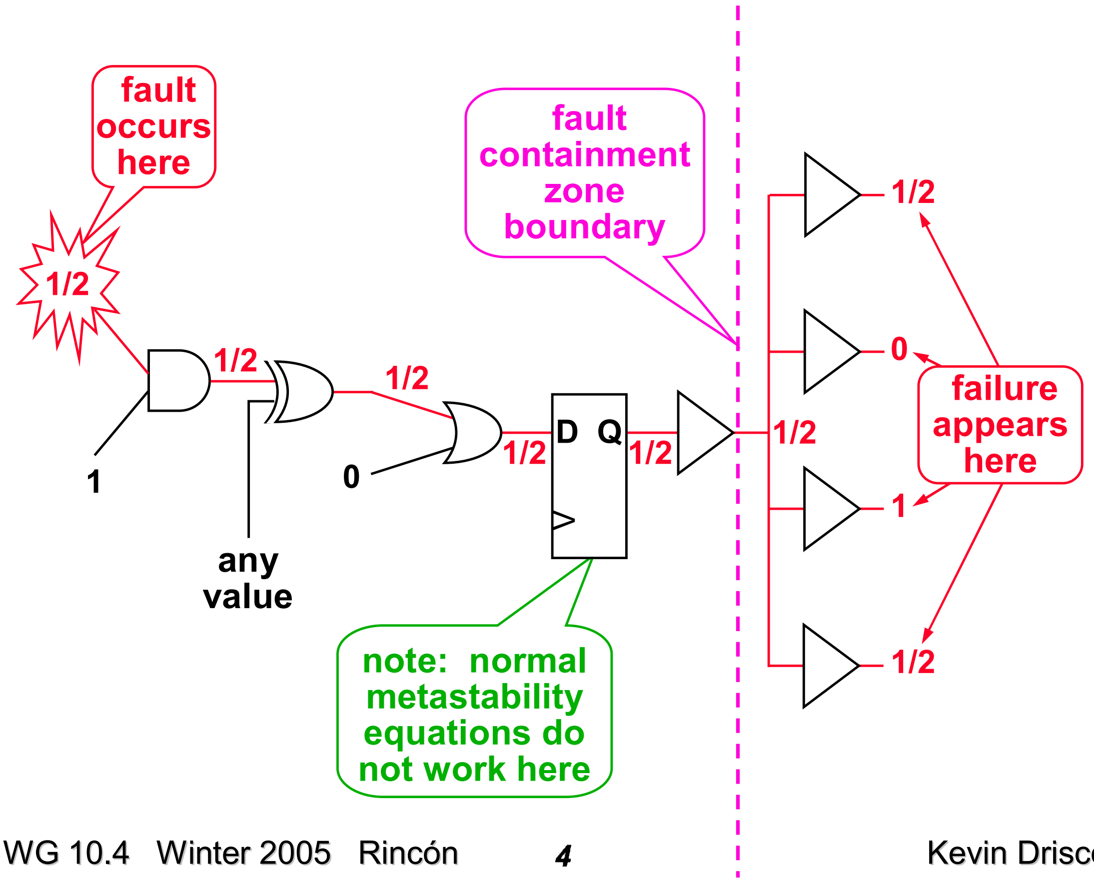
This allows the possibility of a digital logic signal being "1/2".



Logic Gate Transfer Function with "1/2" Noise



½ input is in the gate's highest gain region, any noise is greatly amplified

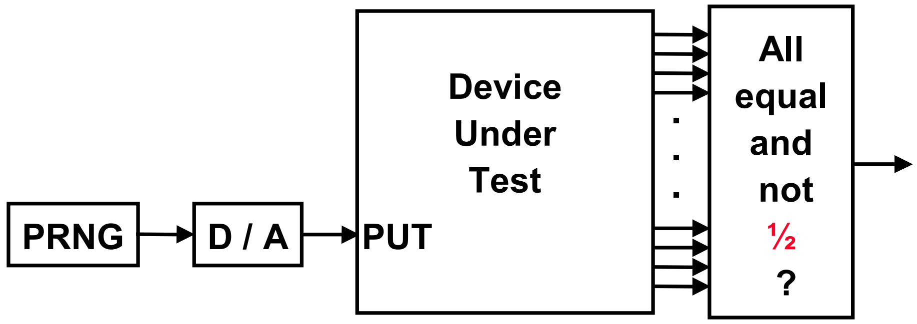


Byzantine Filtering

- Bit-level (waveform) filtering
  - Implemented with any combination of:
    - Schmitt triggers
    - Synchronizers
      (same as used to mitigate metastability)
    - Glitch filters
    - . . . (almost any technique to reduce noise)
  - Perfect coverage impossible
  - Need to determine coverage of implementation
    - Typical pessimistic system Byzantine failure probability is 10^{-5} (10 nodes, 10 critical components in each node with 10^{-7} probability of failure)
    - Typical system requires < 10^{-10} probability of failure
    - Typical coverage needs to be 0.99999

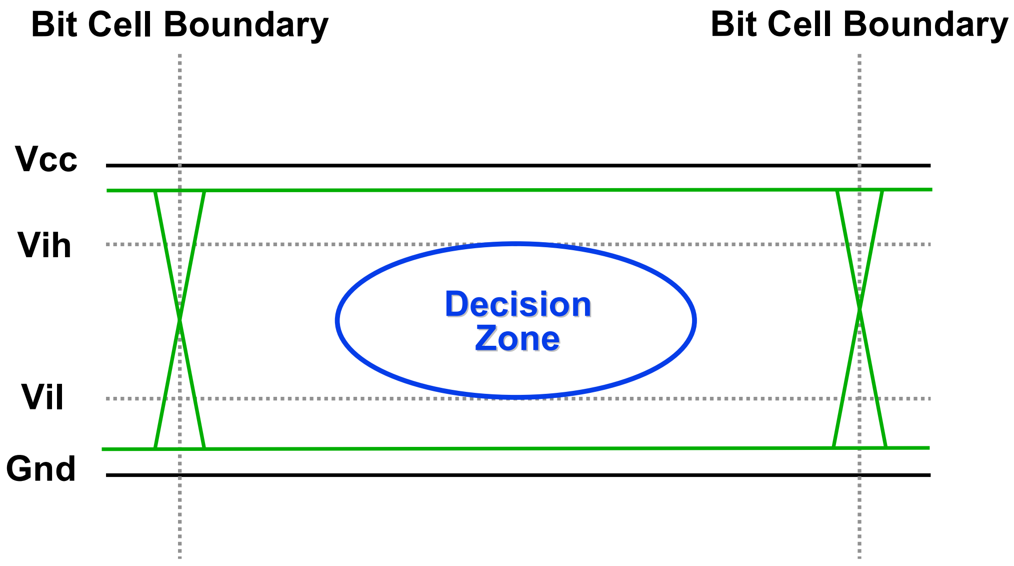
"Byzantine" Fault Injection

Concept:
Create a suitably representative set of faulty waveforms



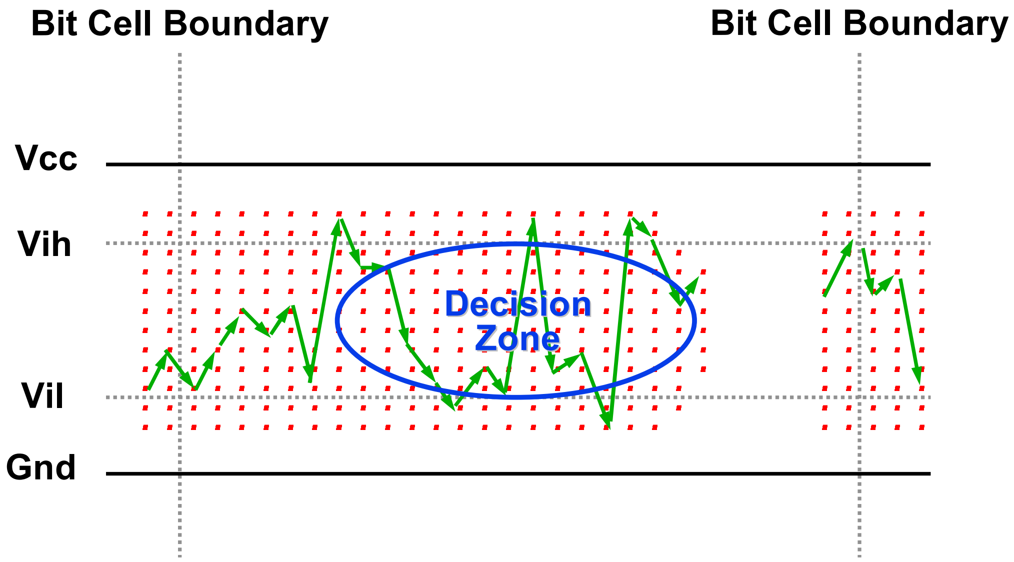
Acronyms:
PRNG = Pseudo Random Number Generator
D / A = Digital to Analog Converter
PUT = Port Under Test

Bit Cell Decision Threshold



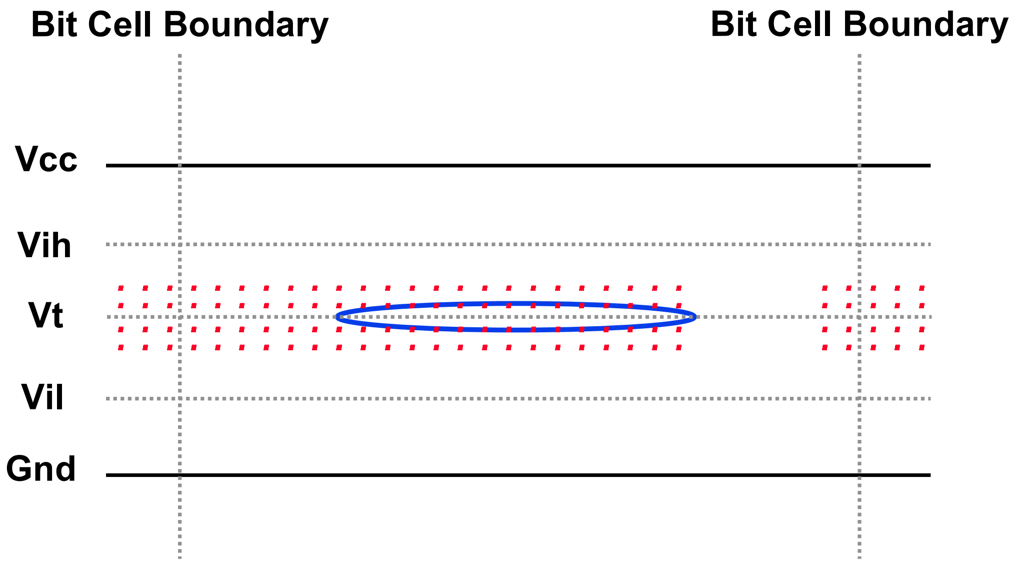
WG 10.4 Winter 2005 Rincón *7* Kevin Driscoll

Remove Test Sample Points Past Hold Time

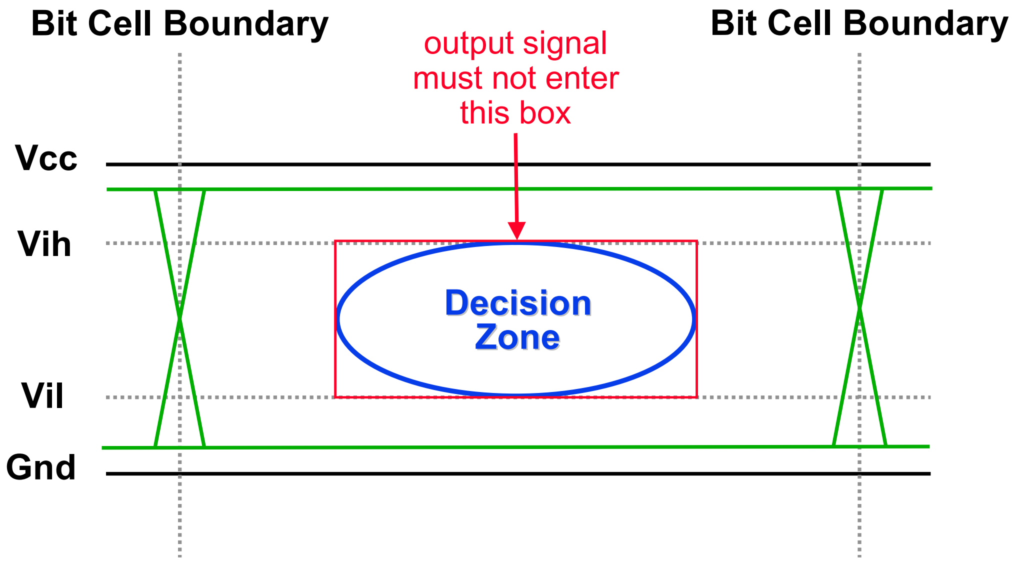


WG 10.4 Winter 2005 Rincón *8* Kevin Driscoll

Test Amplitude Reduction with Known Threshold

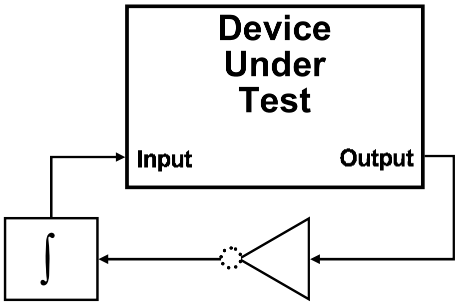


"1/2" DUT Output Signal Rejection



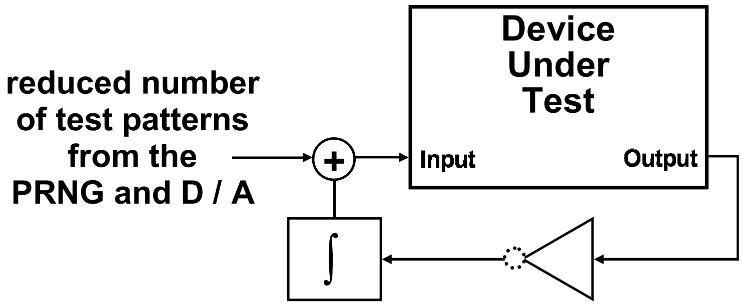
How to Find a Device's Input Threshold

- Connect a device's output back to its input such that the loop has an odd number if inversions in it.
  - This creates an oscillator.
- Add an integrator with a very large time constant.
  - This filters out the oscillations.
- Integrator's output settles on a value which is the input's threshold voltage.



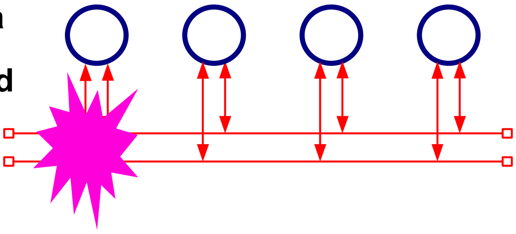
Completion of the Tester

- Add the pseudo-exhaustive bit pattern trajectories to the input feedback (with a reduced number of amplitude test points).
- Either latch the integrator's output before applying the test patterns or make sure the test pattern's are "DC balanced" over the time constant of the integrator.

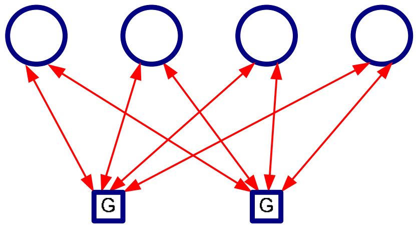


The Path of Low-Cost Dependable Systems

- Redundant Bus
  - Not sufficient due to spatial proximity faults (unavoidable via routing)
  - Serious issues with babbling and masquerade failures
    - Local Guardians not truly independent

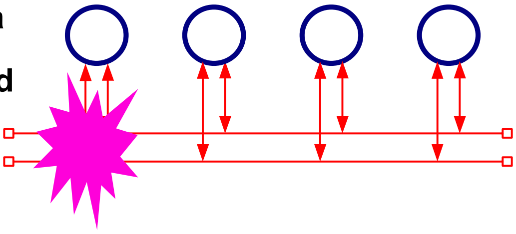


- Redundant Star
  - Independent guardians (G)
  - Reshaping in Guardians performs Byzantine filtering
  - Dual architecture does not allow arbitrarily faulty components
  - What dependability level can be reached?
    - Weakest link principle says 10^{-6}
  - 10^{-6} is not good enough
    - Try argue bizarre fault mode has lower probability
    - Or use triplex (not low cost)

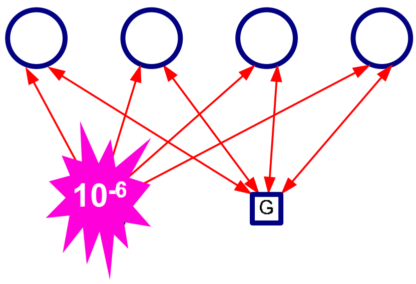


The Path of Low-Cost Dependable Systems

- Redundant Bus
  - Not sufficient due to spatial proximity faults (unavoidable via routing)
  - Serious issues with babbling and masquerade failures
    - Local Guardians not truly independent

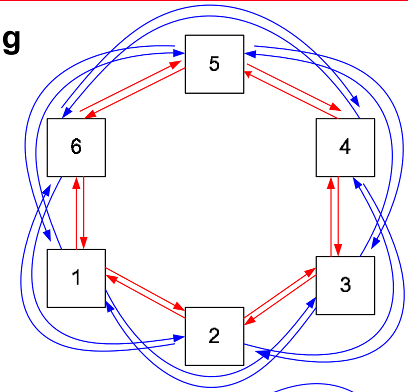


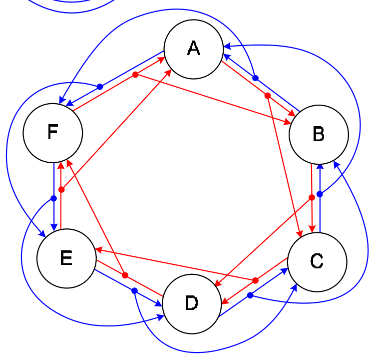
- Redundant Star
  - Independent guardians (G)
  - Reshaping in Guardians performs Byzantine filtering
  - Dual architecture does not allow arbitrarily faulty components
  - What dependability level can be reached?
    - Weakest link principle says 10^{-6}
  - 10^{-6} is not good enough
    - Try argue bizarre fault mode has lower probability
    - Or use triplex (not low cost)



Braided Ring

- Begins with traditional braided ring
  - Each node has links to two nearest neighbors and links to two next nearest neighbors
  - Each link is 2x unidirectional
  - Four link paths from each source to each destination (used for availability only)
- Adds these new ideas
  - Eliminate half of transmitters and ~1/4 of wire length
  - Uses Byzantine filtering during bit regeneration in each node
  - Does a bit-for-bit compare of each node's output vs input
    - Miscompares set a failed flag in the tail of bad messages
    - Nearly 100% coverage of regeneration errors
  - …

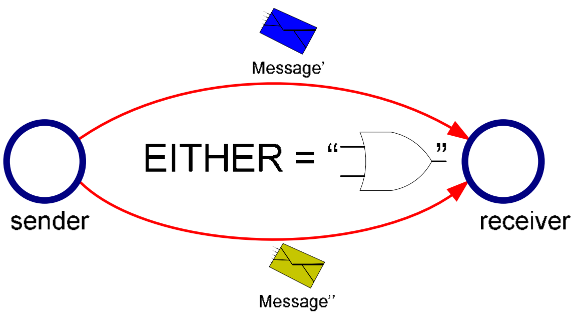




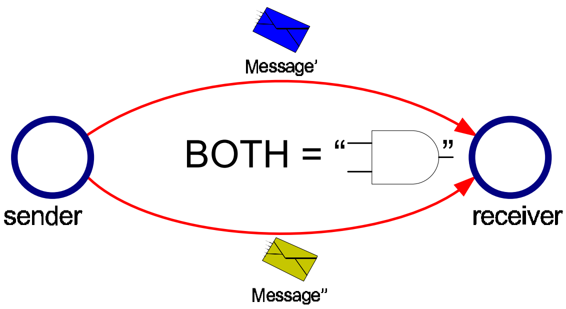
WG 10.4 Winter 2005 Rincón *15* Kevin Driscoll

Availability versus Integrity

- The Availability "OR"
  - If miscompare, arbitrarily select one
  - Goal is readiness for correct service



- The Integrity "AND"
  - If miscompare, reject both
  - Goal is absence of improper alterations

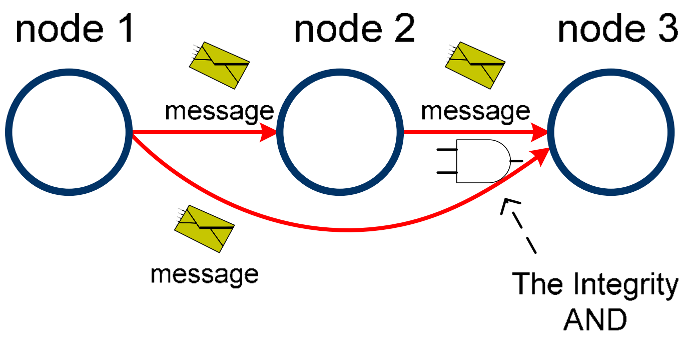


Simple 2x replication gives you availability or integrity but not both!

WG 10.4 Winter 2005 Rincón *16* Kevin Driscoll

Checking for Node Failures

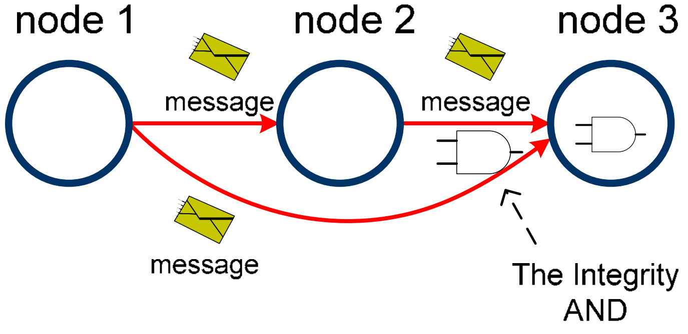
- Errors of components are detected by doing a bit-for-bit compare of node's input versus output



WG 10.4 Winter 2005 Rincón **17** Kevin Driscoll

Checking for Node Failures

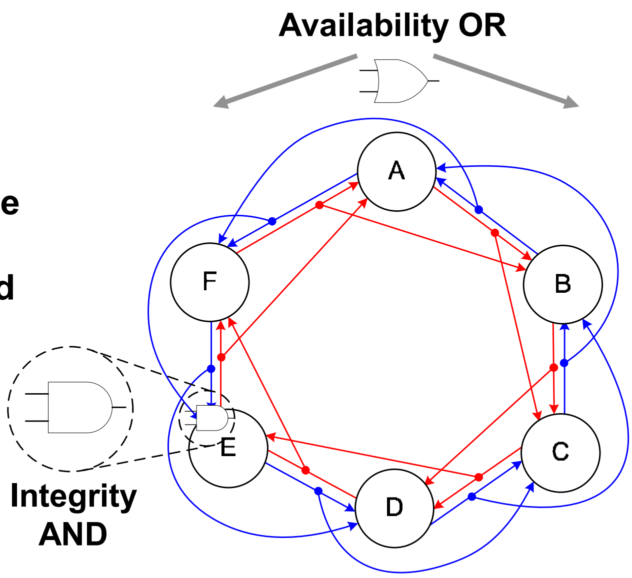
- Errors of components are detected by doing a bit-for-bit compare of node's input versus output
- Comparison of protocol behavior (timing)



WG 10.4 Winter 2005 Rincón **18** Kevin Driscoll

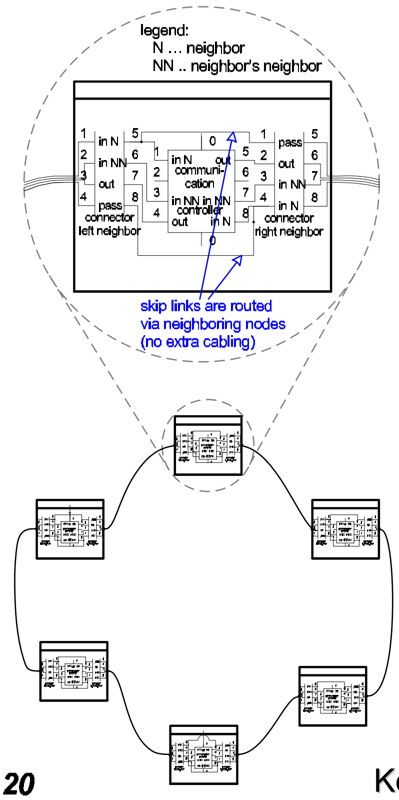
Braided Ring is a Full Coverage Architecture

- Full coverage data propagation
  - “true” 10^{-9}
- Neighboring nodes perform guardian function
- No need for separate silicon for guardian
  - Saves silicon and thus cost



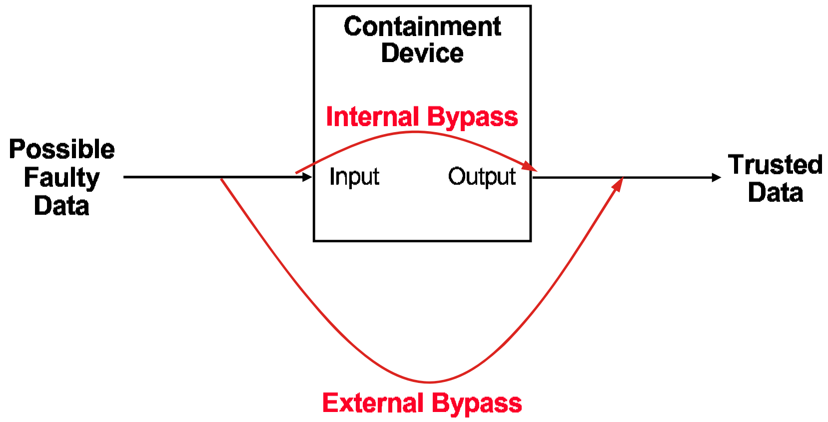
One of Many Ways to Cable the Braided Ring

- “Skip” links to next nearest neighbors can be routed via nearest neighboring nodes
- Useful when bundling cable costs are a significant part of wiring costs



Detection Containment Bypasses

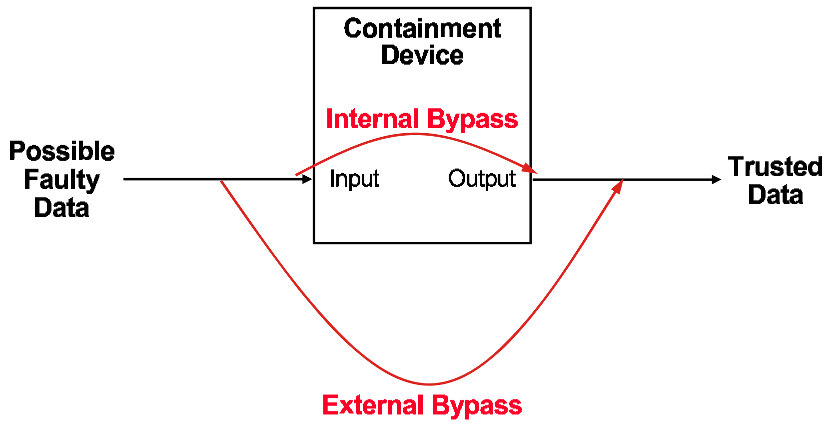
- Need to detect internal and external bypasses of devices entrusted to do fault (error) containment.



WG 10.4 Winter 2005 Rincón **21** Kevin Driscoll

Detection Containment Bypasses

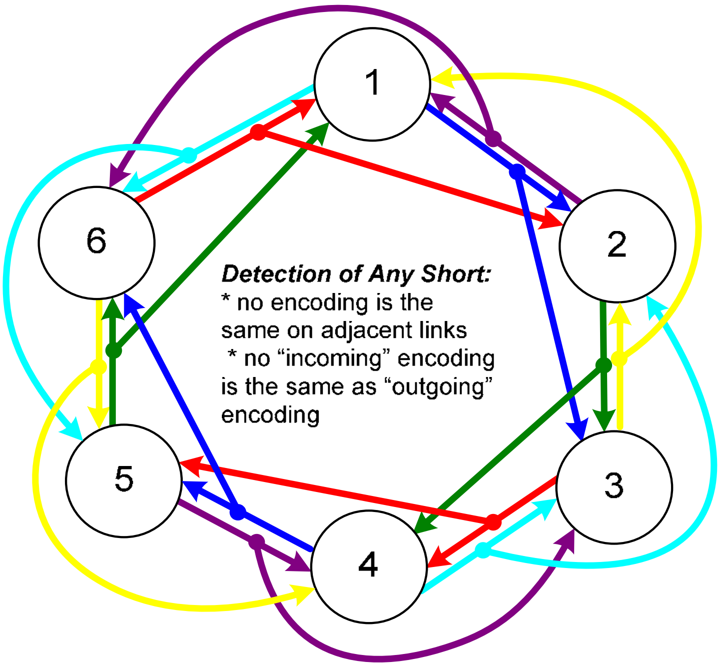
- Need to detect internal and external bypasses of devices entrusted to do fault (error) containment.



- Solution: “Encrypt” each link differently

WG 10.4 Winter 2005 Rincón **22** Kevin Driscoll

Ring Example Using the Minimum 6 Keys



legend: different color represents different encoding

Benefits of a Braided Ring

- **Compared to a bus topology system**
  - **Survives a proximately fault**
  - **Babble and masquerade faults stopped by neighbors**
    - **No problem with untrustable local guardians**
    - **No need to add another integrated circuit for guardianship**
    - **No electrical fault isolation needed for network interface**
- **Compared to a star topology system**
  - **No need for additional (triplex) central components**
    - **Less cost**
    - **Less unreliability**
  - **Less costly wiring**
    - **Cable has to go only to nearest neighbor, not all the way to a central star**
- **Optimally cheap Byzantine solution?**

Lisa Spainhower

WG 10.4 : Upcoming IBM sponsored/ contributing activities & research

- SELSE (System Effects of Logic Soft Errors) April 5& 6, 2005 UIUC **[http://www.crhc.uiuc.edu/SELSE]**
- 3P3AD (3rd Proactive Problem Prediction, Analysis and Determination Conference) April 26, 2005 Yorktown Hts., NY
- Autonomic Computing Benchmarking – Configuration Complexity **[http://www.research.ibm.**
- Autonomic Computing as originally conceived **[IEEE Computer, pp. 41-50, January 2003]**

Call for Participation
Workshop on

System Effects of Logic Soft Errors

University of Illinois at Urbana-Champaign, April 5th & 6th, 2005

* What are the metrics to describe LSER?
* How are the mitigation techniques chosen for a given design?
* How are the metrics used to select the mitigation technique?
* How is system level derating predicted and measured?
* Are there favored techniques or will there in general be a combination of device, circuit and microarchitectural mitigation techniques for a given application?
* How does system level derating enter into the choice of mitigation techniques?
* What are the most significant LSER related findings from case studies?

3rd Proactive Problem Prediction, Avoidance, and Diagnosis Conference: Predictive Techniques for Self-healing and Performance Optimization

April 26, 2005
IBM Auditorium
Yorktown Heights, NY

Anomaly detection and classification
Performance and resource analysis
Text mining and pattern/rule derivation
Surveys of predictive techniques
Machine learning and pattern recognition algorithms
Correlation technology
Performance Optimization
Prediction of imminent field problems
Financial Futures
Portfolio value at risk analysis
Log analysis
Environmental and thermal analysis
System configuration analysis

Byzantine Faults in a Rational World

Amitanand Aiyer, Allen Clement, Jean-Philippe Martin, Carl Porth, Mike Dahlin and Lorenzo Alvisi

LASR
UT Austin

Two motivating observations

- Dependability more pressing need than performance
- Distributed systems increasingly span multiple administrative domains

How should nodes be modeled?

Traditionally, a node is modeled as either::

- *Correct:* the node follows its specification
- *Faulty:* the node deviates from its specification
  - benign
  - Byzantine

A new classification

A node is either:

- *Altruistic*: the node follows the assigned protocol
- *Rational*: the node is not malicious, but will deviate from the assigned protocol to maximize its benefits and minimize its costs
- *Byzantine*: the node deviates from assigned protocol even when not “in its interest” because of malfunction, misconfiguration, or malice

Nodes may be subject to benign faults

Our goal

Develop the theory and practice of building distributed systems that tolerate both rational and Byzantine behavior

Our approach

- Adapt low-level BFT primitives (state machine replication, quorum replication, reliable broadcast) to tolerate rational behavior
  - create suite of building blocks
  - avoid ad-hoc reasoning for each application
- Develop end-to-end BRFT applications on top of these primitives
  - challenge: integrate low-level BRFT mechanism with end-to-end incentive structure of the application

Our assumptions

1. Byzantine nodes are few, but no bounds on the number of rational nodes
2. Cost: bandwidth, storage, computation, power, etc.
3. Long term repeated interactions
   - only way to achieve equilibrium in Prisoner's dilemma
4. Strong identities and restricted membership
   - prevent Sybil attack
   - enable *internal* and *external* disincentives to deter misbehavior
   - reasonable for our target applications

Our target application

Peer-to-peer backup system

- stresses BRFT in multiple dimensions
  - multiple resources integration
  - requires achieving BRFT at different timescales
  - range of provisioning may require to break simple symmetry between pairs of nodes
  - applicable to deployment scenarios with different trust models
- useful!
  - lab, dorm, Box Populi

Incentive compatible backups

- System links storage available to a node with storage contributed by the node
- To enforce quotas
  - peers publish signed lists of teh data they store and of the data that is stored on their behalf
  - *receipts* used to detect and prove lies
  - ¬*witnesses* provide incentives against "passive-aggressive" nodes
  - witnesses implemented as BRFT replicated state machines

Status: protocols

- Studied two protocols:
  1. Lamport's Byzantine agreement with using unforgeable signatures
  2. Srikanth and Toueg's Byzantine agreement without signature s (a.k.a. the *echo* protocol)
- Proved both protocols are vulnerable to the "tragedy of the commons"
- Derived and proved incentive compatible versions of these protocols
- Working on BRFT state machine replication

Status: application

- Authors are trusting their iTunes library (or whatever else is vital to them) to initial prototype
- On schedule for lab-wide deployment in 2 weeks (about 20 users)
- Working on dorm deployment in 6 weeks

White Mansion Overlooking the Sandy Beach and the Ocean