



High-Performance Cluster Computing in Space

Challenges and Opportunities

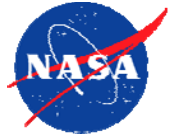
Hans P. Zima

Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA
zima@jpl.nasa.gov

Clusters and Computational Grids for Scientific Computing
CCGSC 2008

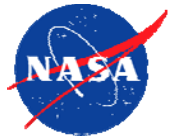
Flat Rock, North Carolina, September 14-17, 2008

Contents



- 1. Requirements and Challenges for Space Missions**
- 2. Emerging Multi-Core Systems**
- 3. High Capability Computation in Space**
- 4. An Introspection Framework for Fault Tolerance**
- 5. Concluding Remarks**

NASA has more than 50 missions exploring our solar system



Spitzer studying stars and galaxies in the infrared

Ulysses studying the sun

Cassini studying Saturn

GALEX surveying galaxies in the ultraviolet

Aqua studying Earth's oceans

Mars Odyssey, rovers "Spirit" and "Opportunity" studying Mars

CALIPSO studying Earth's climate

Two Voyagers on an interstellar mission

Aura studying Earth's atmosphere

MESSENGER on its way to Mercury

Chandra studying the x-ray universe

Hubble studying the universe

New Horizons on its way to Pluto

QuikScat, Jason 1, CloudSat, and GRACE (plus ASTER, MISR, AIRS, MLS and TES instruments) monitoring Earth.

Space Challenges: Environment



◆ Radiation

- *total dose*
- *Single Event Upsets (SEUs)*

◆ Temperature

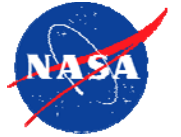
- *wide range (-170C on Europa to >400C on Venus)*
- *short cycles (about 50C on MER)*

◆ Vibration

- *launch*
- *planetary Entry, Descent, Landing (EDL)*

.... these present severe constraints to the compute hardware

Space Challenges: Communication and Guidance



◆ Bandwidth

- *6 Mbit/s max*
- *spacecraft transmitter power typically less than light bulb in a refrigerator*

◆ Latency (one way)

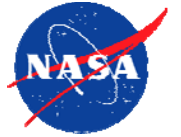
- *20 minutes to Mars*
- *13 hours to Voyager 1*

◆ Navigation

- *Position*
- *Velocity*

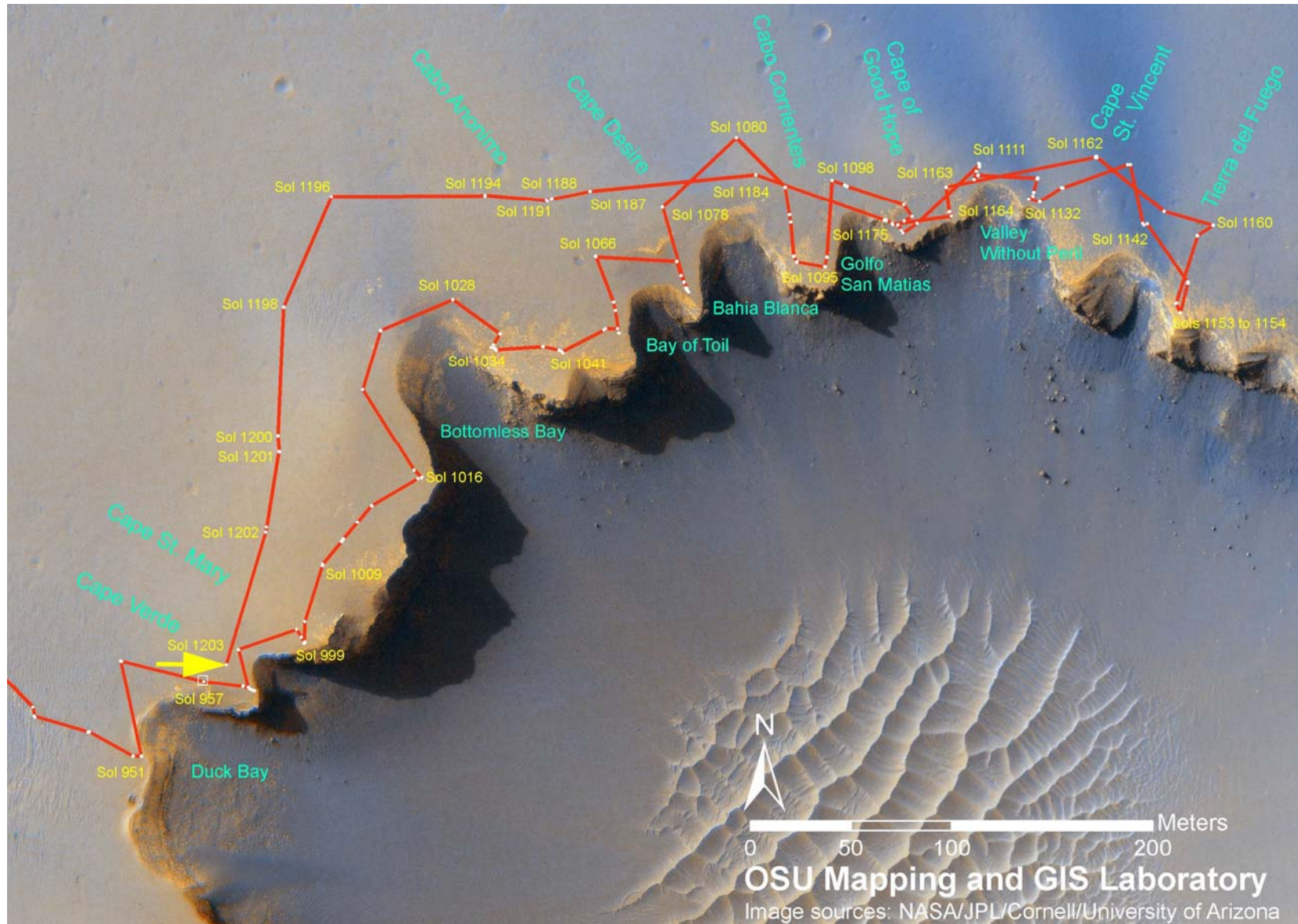
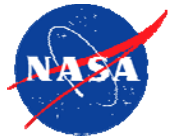
... these present severe constraints to mission operations

Space Challenges: Engineering



- ◆ **Only flight qualified parts are typically used**
 - *systems are at least 5 years out of date when launched (two generations behind commercial state-of-the-art)*
- ◆ **Power and Mass Restrictions**
 - *20-30 W for a flight computer*
- ◆ **Often test of final system possible only when it is flown**
 - *importance of modeling and simulation*
- ◆ **Long mission duration challenges maintainability of ground assets in operations phase**
 - *Voyager is based on custom flight computer designed with MSI parts and ferrite core memory of the late 1960's (programmed in assembler)*

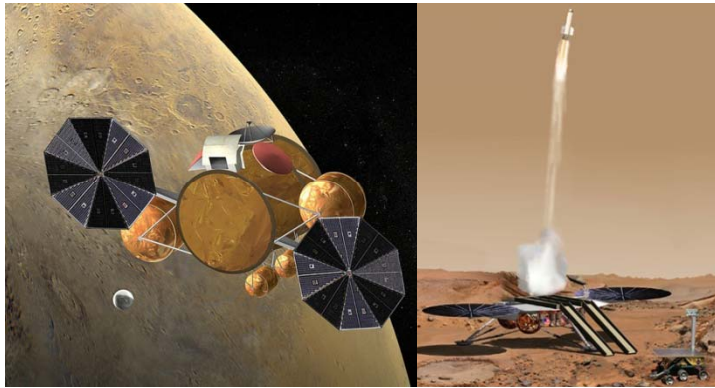
Duck Bay: Site of Opportunity's descent into Victoria Crater



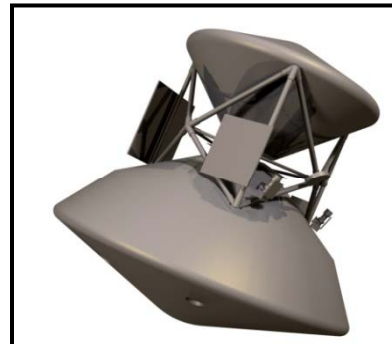
NASA/JPL: Potential Future Missions



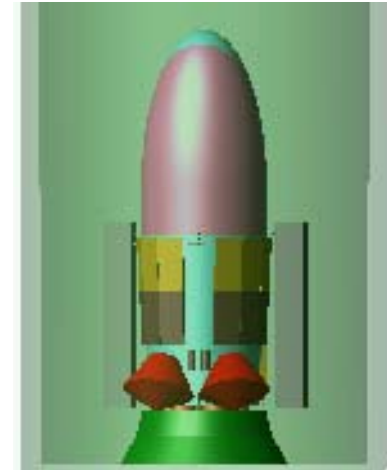
Artist Concept



Mars Sample Return



Titan Explorer



Neptune Triton Explorer



Europa Explorer



Europa Astrobiology Laboratory

Future Mission Applications



◆ New Types of Science

- *Opportunistic science (event detection: e.g., dust devils or volcanic eruptions)*
- *Model-based autonomous mission planning*
- *Smart high resolution sensors (e.g., Gigapixel, SAR,...)*
- *Hyperspectral imaging*

◆ Entry Descent & Landing

- *Flight control through disparate flight regimes*
- *Landing zone identification*
- *Lateral winds*
- *Soft touchdown*

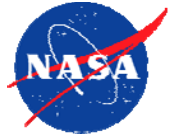
◆ Surface Mobility

- *Terrain traversal, obstacle avoidance*
- *Science Target identification*
- *Image/video Compression*

◆ Communication with Earth is a limiting factor

- *Small bandwidth requires reduction of data transfer volume; on-board data analysis, filtering, and compression*

New Requirements

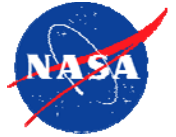


New applications and the limited downlink to Earth lead to two major new requirements:

1. Autonomy

2. High-Capability On-Board Computing

The Traditional Approach will not Scale



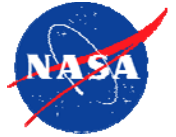
- ◆ **Traditional approach based on radiation-hardened processors and fixed redundancy (e.g., Triple Modular Redundancy—TMR)**
 - ***Current Generation (Phoenix and Mars Science Lab –'09 Launch)***
 - ◆ *Single BAE Rad 750 Processor*
 - ◆ *256 MB of DRAM and 2 GB Flash Memory (MSL)*
 - ◆ *200 MIPS peak, 14 Watts available power (14 MIPS/W)*
 - ***ST8 Honeywell Dependable Multiprocessor***
 - ◆ *COTS system with Rad 750 controller (100 MIPS) and IBM PowerPC 750FX (1300 MIPS)*
 - ◆ *120 MIPS/Watt Performance*
 - ◆ *Fault tolerant architecture*
- ◆ **Rad-hard processors today lag commercial architectures by a factor of about 100 (and growing)**
- ◆ **By 2015: a single rad-hard processor may deliver about 1 GFLOPS—orders of magnitude below requirements**

Contents

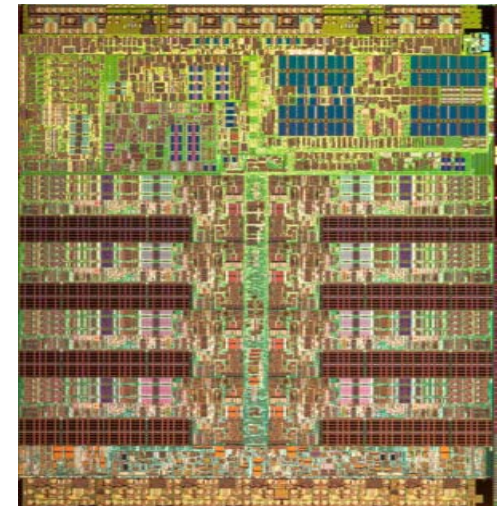


- 1. Requirements and Challenges for Space Missions**
- 2. Emerging Multi-Core Systems**
- 3. High Capability Computation in Space**
- 4. An Introspection Framework for Fault Tolerance**
- 5. Concluding Remarks**

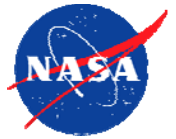
Multicore: An Emerging Technology



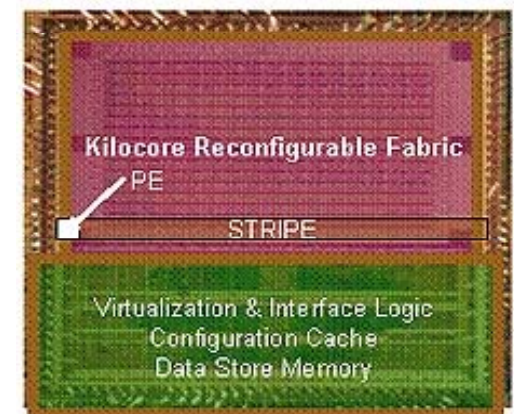
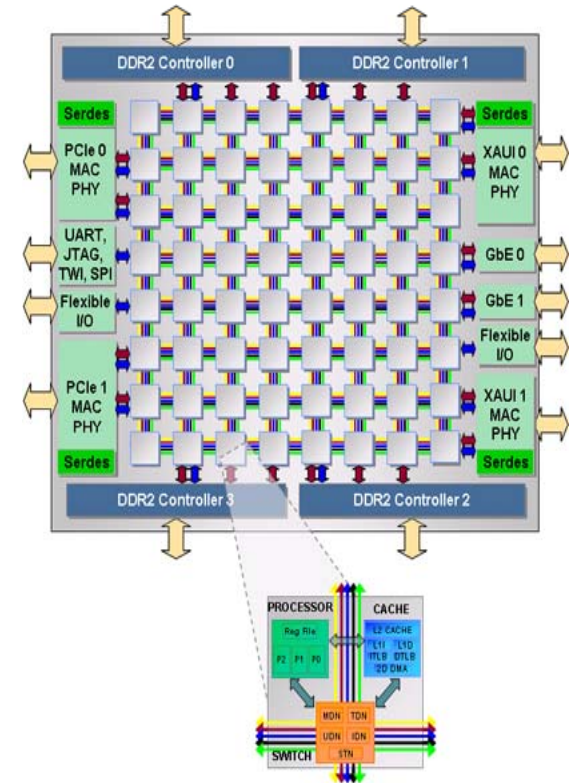
- ◆ **The era of faster sequential processors is over**
 - *CMOS manufacturing technology approaches physical limits*
 - *power wall, memory wall, instruction-level parallelism (ILP) wall*
 - *Moore's Law still in force: number of transistors on chip increasing*
- ◆ **Multicore chip: a single chip with two or more independent processing units**
- ◆ **Multicore technology provides continued performance growth**
 - *improvements by multiple cores on a chip rather than higher frequency*
 - *on-chip resource sharing for cost and performance benefits*
- ◆ **Multicore systems have been produced since 2000**
 - *IBM Power 4; Sun Niagara; AMD Opteron; Intel Xeon;...*
 - *Quadcore systems by AMD, Intel*
 - *IBM/Sony/Toshiba: Cell Broadband Engine*
 - ◆ *Power Processor (PPE) and 8 Synergistic PEs (SPEs)*
 - ◆ *peak performance 230 GF (1 TF expected by 2010)*
- ◆ **1000 cores on a chip possible with 30nm technology**
- ◆ **"Manycore" chips are already emerging ...**



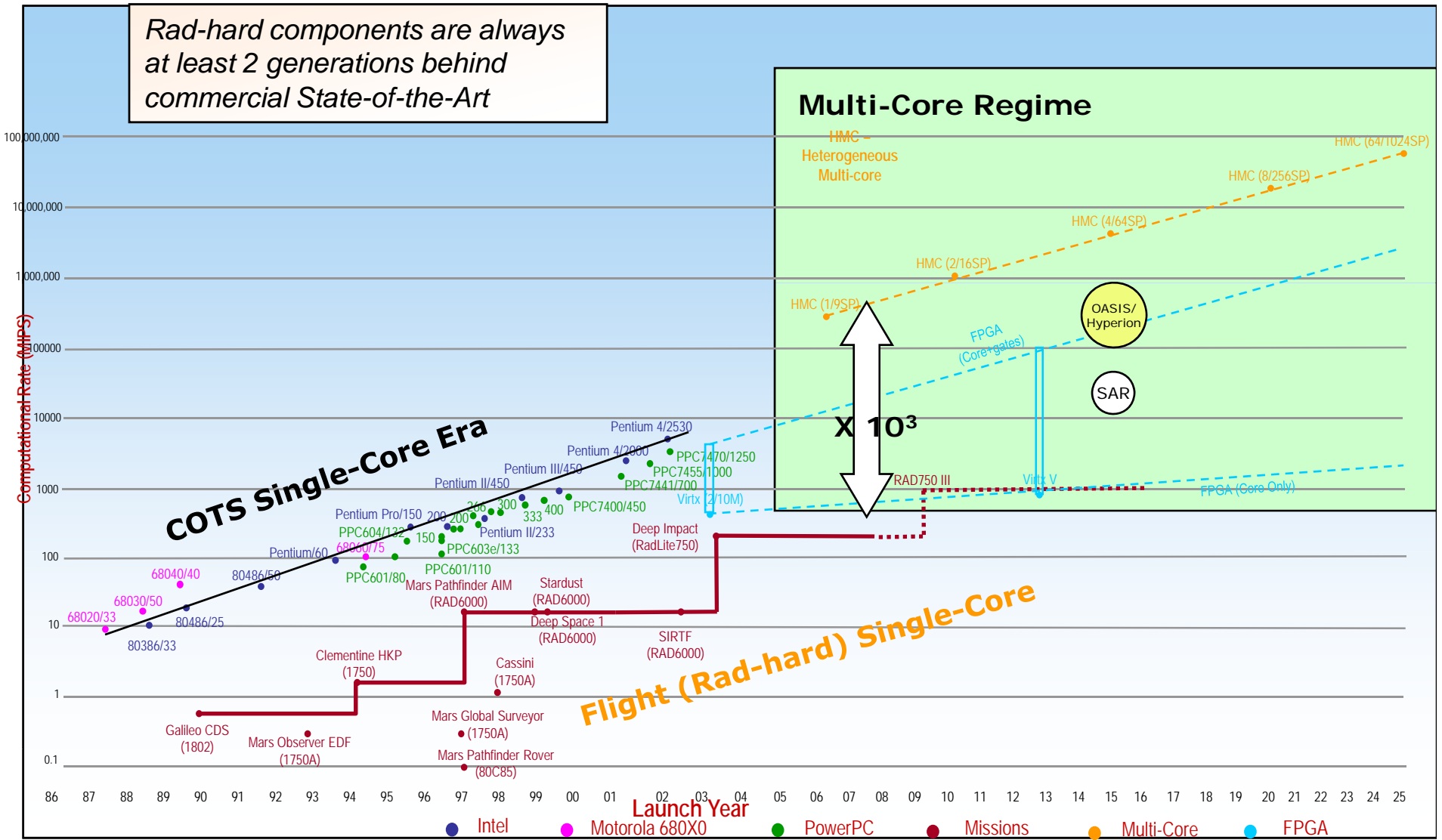
Future Multicore Architectures: From 10s to 100s of Processors on a Chip



- ◆ **Tile64 (Tilera Corporation, 2007)**
 - 64 identical cores, arranged in an 8X8 grid
 - iMesh on-chip network, 27 Tb/sec bandwidth
 - 170-300mW per core; 600 MHz – 1 GHz
 - 192 GOPS (32 bit)—about 10 GOPS/Watt
- ◆ **Kilocore 1025 (Rapport Inc. and IBM, 2008)**
 - Power PC and 1024 8-bit processing elements
 - 125 MHz per processing element
 - 32X32 “stripes” dedicated to different tasks
- ◆ **512-core SING chip (Alchip Technologies, 2008)**
 - for GRAPE-DR, a Japanese supercomputer project
- ◆ **80-core 2 TF research chip from Intel (2011)**
 - 2D on-chip mesh network for message passing
 - 1.01 TF (3.16 GHz); 62W power—16 GOPS/Watt
 - **Note: ASCI Red (1996): first machine to reach 1 TF**
 - ◆ 4,510 Intel Pentium Pro nodes (200 MHz)
 - ◆ 500 KW for the machine + 500 KW for cooling of the room

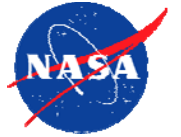


Space Flight Avionics and Microprocessors History and Outlook



Source: Contributions from Dan Katz (LSU), Larry Bergman (JPL), and others

Contents



- 1. Requirements and Challenges for Space Missions**
- 2. Emerging Multi-Core Systems**
- 3. High Capability Computation in Space**
- 4. An Introspection Framework for Fault Tolerance**
- 5. Concluding Remarks**

Multi-Core Challenges for Space



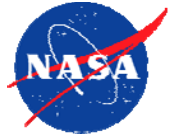
◆ General

- *parallel programming and execution models*
- *complex hardware architectures*
- *porting of legacy codes*
- *programming environments*
- *new methods for exploiting hardware: introspection, automatic tuning, power management*

◆ Space Critical

- *real-time*
- *fault tolerance*
- *verification and validation*

COTS-Based On-Board Systems

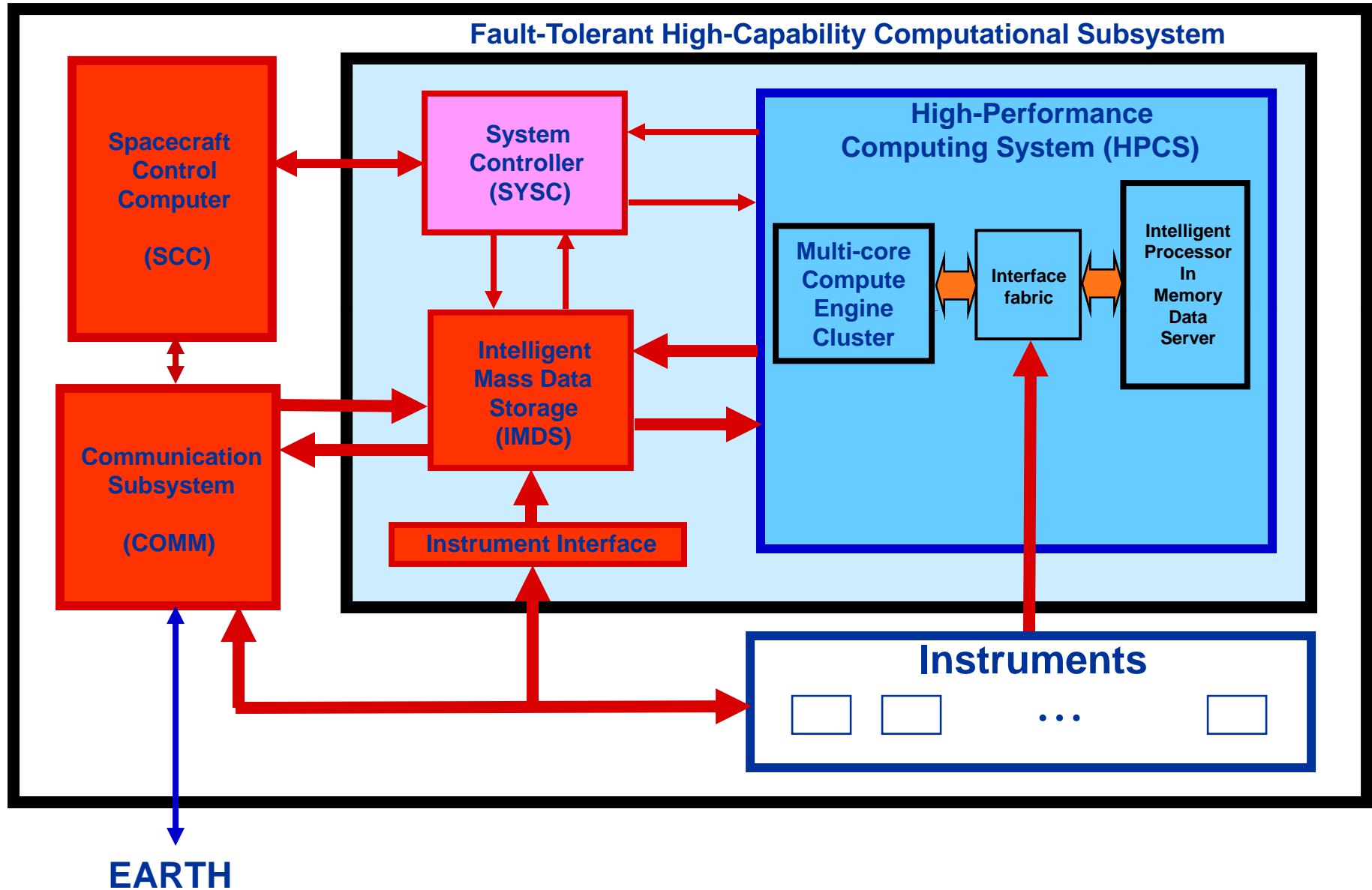


- ◆ **Basic Idea:** augment the traditional, radiation-hardened computational capability of a spacecraft with a commodity high-capability subsystem based on multi-core technology

- ◆ **Earlier approaches with a similar goal (based on state-of-the-art hardware architectures) include:**
 - *Remote Exploration and Experimentation (REE) project at NASA*
 - *ST8 Dependable Multiprocessor (DM) project (Honeywell, U. of Florida)*
 - *both projects were based on standard multiprocessors; none of them led to an actual flight mission*

- ◆ **Key issue:** fault tolerance for the multi-core subsystem

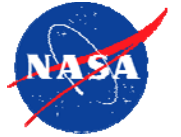
High-Capability On-Board System: An Example



- ◆ **SEUs and MBUs are radiation-induced transient hardware errors, which may corrupt software in multiple ways:**
 - *instruction codes and addresses*
 - *user data structures*
 - *synchronization objects*
 - *protected OS data structures*
 - *synchronization and communication*

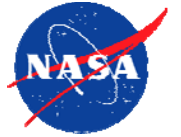
- ◆ **Potential effects include:**
 - *wrong or illegal instruction codes and addresses*
 - *wrong user data in registers, cache, or DRAM*
 - *control flow errors*
 - *unwarranted exceptions*
 - *hangs and crashes*
 - *synchronization and communication faults*

Current Focus



- ◆ **Support for application-oriented, adaptive, and dynamic fault tolerance in the HPCS component**
- ◆ **Assumptions**
 - *HPCS: a homogeneous cluster built from COTS-based multi-core components*
 - *applications are non-critical, using standard software, parallelized based on the Message-Passing Interface (MPI)*
 - *considering the effects of hard and transient faults*
- ◆ **Approach**
 - *replacing fixed redundancy schemes with an application-adaptive approach, exploiting application and system knowledge*
 - *leveraging existing methods for static analysis, profiling*
 - *supported by an introspection framework*

Contents



- 1. Requirements and Challenges for Space Missions**
- 2. Emerging Multi-Core Systems**
- 3. High Capability Computation in Space**
- 4. An Introspection Framework for Fault Tolerance**
- 5. Concluding Remarks**

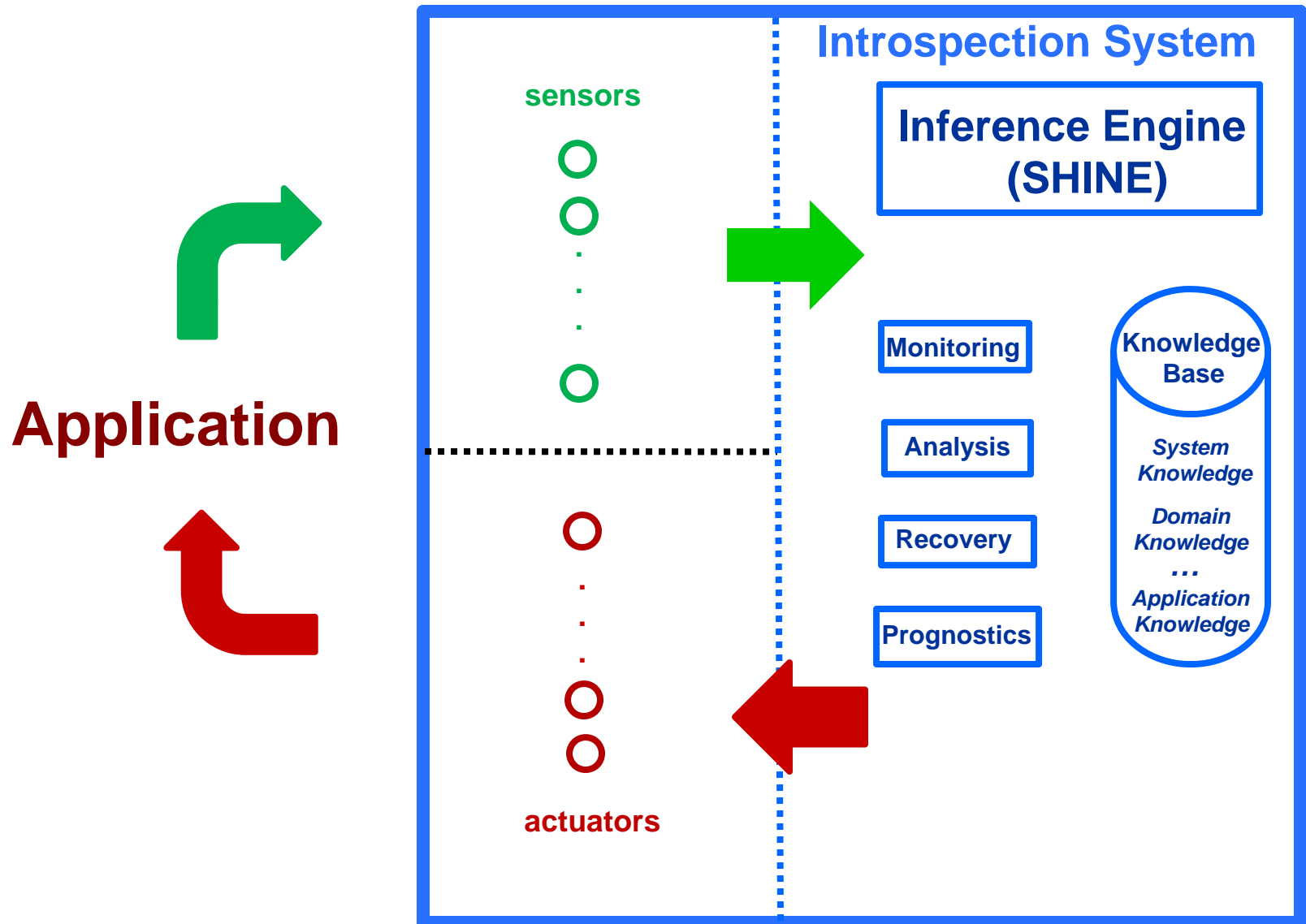
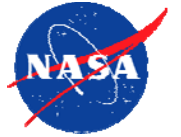
A Framework for Introspection



Introspection...

- ◆ provides *dynamic* monitoring, analysis, and feedback, enabling system to become self-aware and context-aware:
 - *monitoring execution behavior*
 - *reasoning about its internal state*
 - *changing the system or system state when necessary*
- ◆ exploits adaptively the available threads
- ◆ can be applied to different scenarios, including:
 - *fault tolerance*
 - *performance tuning*
 - *power management*
 - *behavior analysis*
 - *intrusion detection*

An Introspection Module (IM)



The Spacecraft Health Inference Engine (SHINE)



- A tool for building and deploying real-time rule-based reasoning systems for detection, diagnostics, prognostics, and recovery

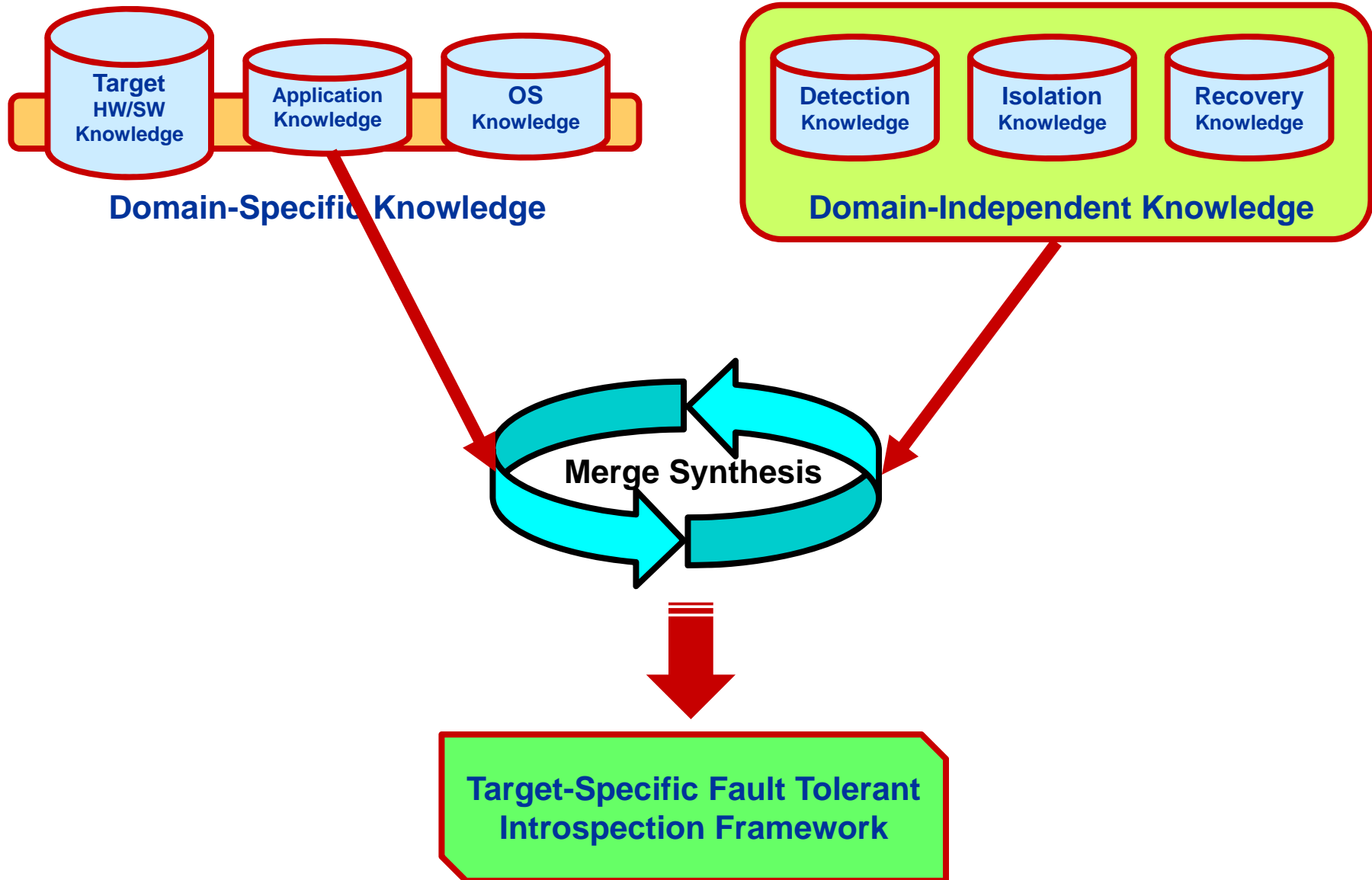
- **Outperforms commercial products by orders of magnitude**
 - *Inference speed is achieved using graph transformations based on data flow analysis*
 - *Rules are statically analyzed for all interactions*

- The underlying structure is mapped into temporally invariant dataflow elements for execution on sequential or parallel hardware

- The final representation is either executed in a development environment or can be translated to a target language (C/C++)

- **Deliveries**
 - *NASA (Deep Space Network, applied to five NASA missions)*
 - *Military (Lockheed JSF program, F-18 with 25+ flights)*
 - *Aerospace (Northrup, Lockheed, Boeing)*
 - *Commercial (ViaChange, Vialogy, VIASPACE, Aerosciences, etc.)*

Knowledge Synthesis

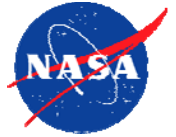


Application-Oriented Introspection-Based Fault Tolerance in the HPCS: A Research Agenda



- ◆ **Current focus on transient and hard faults**
- ◆ **Goal: reducing overhead of fixed-redundancy schemes**
- ◆ **Based on a (mission-dependent) fault model**
 - *classifies faults (fault types, severity)*
 - *specifies fault probabilities, depending on environment*
 - *prescribes recovery actions*
- ◆ **Addressing fault detection, analysis, isolation, recovery**
- ◆ **Exploiting knowledge from different sources**
 - *static analysis and profiling*
 - *properties of target system hardware and software*
 - *application domain (libraries, data structures, data distributions)*
 - *user-provided assertions and invariants*
- ◆ **Leveraging existing technology**
 - *fixed-redundancy for small critical areas in a program*
 - *Algorithm-Based Fault Tolerance (ABFT)*
 - *naturally fault-tolerant algorithms*
 - *integration of high-level generator systems such as CMU's "SPIRAL"*

Introspection versus Traditional V&V



◆ Introspection

- focuses on **execution time** monitoring, analysis, recovery
- actual work considers transient and hard faults, not design errors

◆ Verification and Validation (V&V)

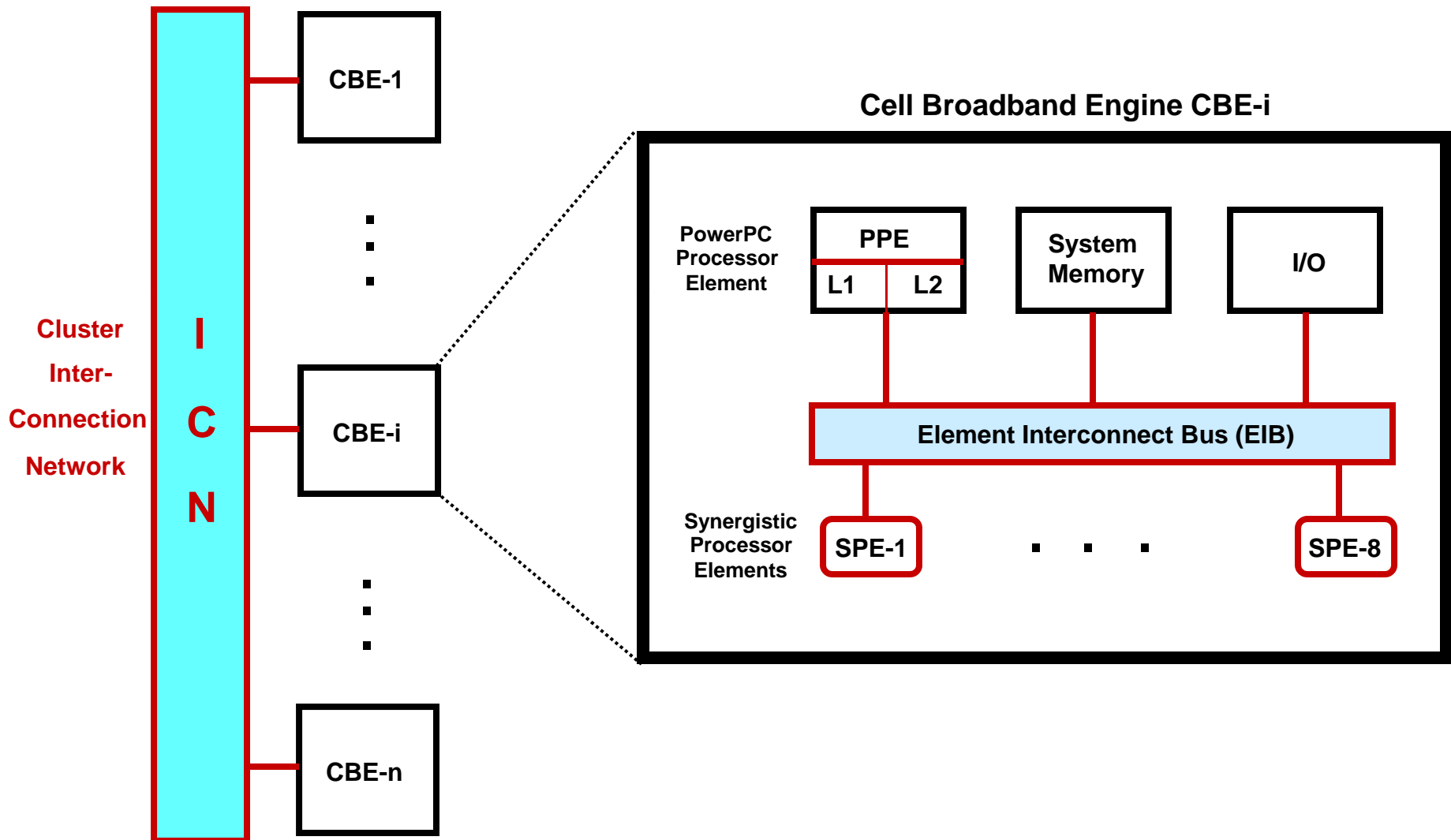
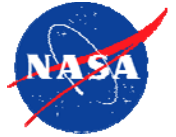
- focuses on design errors
- is applied **before** actual program execution
- cannot deal with dynamic events, such as SEUs, or anomalies occurring during execution

◆ Both verification and test are **not complete**:

- problems may be undecidable or intractable
- tests can prove existence of faults, not their total absence

◆ Introspection can complement traditional V&V technology

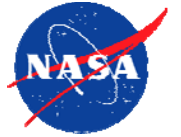
Implementation Target Architecture: Cluster of Cell Broadband Engines



Fault tolerance must be applied across all levels of the system hierarchy:

SPE → PPE → CBE → Cluster

Concluding Remarks



- ◆ **Deep-space missions will require on-board high-capability computing to support autonomy and science processing**
- ◆ **Many challenges remain—in hardware and software**
 - *on-chip fault tolerance support (e.g., isolation)*
 - *programming languages/models, intelligent tools, porting of legacy codes*
 - *fault tolerance across all system levels*
- ◆ **Introspection framework**
 - *supports execution-time monitoring, analysis, and recovery*
 - *exploits the abundance of threads in future multi-core systems*
 - *applicable to different scenarios, such as performance tuning, power management*
- ◆ **Future Work**
 - *completing a prototype implementation on the Cell (and possibly ST8 architecture)*
 - *applying introspection fault tolerance to mission codes (Synthetic Aperture Radar)*
 - *integration of introspection into a coherent V&V approach*

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology