Snapshot of ITO
for PITAC

February 25, 2000

Dr. Shankar Sastry, Director
Information Technology Office
Defense Advanced Research Projecst Agency


Autonomous & Embedded Software
MBIES -J. Sztipanovits
MARS -M. Swinson
PCES -H. Gill
SEC -H. Gill
PCES
TIDES
Distributed Systems & Information Assurance
Fault Tolerant Networking -D. Maughan
Active Networks -D. Maughan
Quorum -G. Koob
Human Computer Interface
Information Management -J. Scholtz
Communicator -G. Strong
Web in a Box -J. Scholtz
TIDES -G. Strong
ITO
Expeditions Project -S. Sastry
-B. Hui
Endeavour
InfoSphere
Portolano
Oxygen
Aura
LCS
Web
BOX
Networking
Next Generation Networking -M. Maeda
Network Management -S. Kumar
SensIT -S. Kumar
Hardware
Data Intensive Systems -R. Graybill
PAC/C -R. Graybill
CPU
Beyond Silicon
Amorphous Computing -S. Kumar

DARPA Has Done Great Things for IT

Current Mission of ITO: Superiority of Armed Forces through Revolutionary Advances in:

- High Performance Computing and Communications Devices
- Networking and Information Assurance
- Embedded Software
- Seamless User Interfaces for the war fighter
- Ubiquitous Computing and Communication Resources

Drivers of IT Research

Computing, Networking, Security have come a long way, but they have a long way to go.

Key drivers:

- wireless and power aware computing devices,
- ubiquitous computing devices,
- embedded computers, (interacting in hard real time with sensors and actuators),
- wideband optical networks,
- MEMS,
- quantum devices,
- system on a chip: billion transistor chip, photonic interconnects, programmable hardware,
- cognitive neurophysiology,
- computational biology.

What Are the Hard Problems??

Wireless:

1. Power/ Energy Aware Computing and Communication: design suites for trading off power/energy consumption. (PAC/C). Design Environments for integrated design across algorithms, instruction sets, and device clock/frequency characteristics.

2. Distributed Computation with sensors which have to trade off on board computation with communication. Thresholding phenomena in performance improvement of networked sensing systems. (SensIT)

3. Secure Ad-hoc networking protocols for insecure and jammable networks. Game theoretic approaches to information assurance in a hostile environment.

What Are the Hard Problems??

Ubiquitous Computing Devices:

1. Hands off interaction with portable or omnipresent computers. Need for voice / speech/ foreign language recognition. (Communicator, TIDES)

2. Operating Systems for small sensors, embedded devices for specialized operation. (Expeditions)

3. Ad hoc networking, content addressable data, queries for intermittently available data stores. (Web in a Box, Expeditions)

4. Dynamic caching of data, data provisioning systems, aggregation of temporally evolving data. (IM, Web in a Box, Expeditions)

5. Collaborative and Hierarchical Decision Making Environments. (Expeditions)

Hard Problems Continued

Embedded Computers and Software:

- Distributed software each performing time critical tasks needed to coordinate with guarantees of overall QoS. (Quorum)
- Verified software for adaptable, time critical operations with multiple distributed processes for physical systems whose mode changes depending on mission priorities. (SEC)
- Model based design of embedded software for hardware-software codesign. The goal is to have embedded software keep up with Moore's law advances in processor speed. (MoBIES)
- Learning and Embedded Intelligence in Robotic Systems (MARS)
- Networked Embedded Systems, on systems like Crusader, F-22 different subsystems have individually designed real time kernels with their own schedules, QoS requirements, etc. Compositionality of the subsystems is unknown resulting in large cost overruns and worse inadequate performance.

Hard Problems Continued

Optical Networking

- WDM is nearing maturity, however optical networking protocols for WDM over IP are not ready yet: routing, congestion control, network management. (NGI)
- Security of high speed networks.
- In more general terms, modeling, estimation and control of traffic at various levels of granularity on WDM networks, ATM networks, and other WAN is in its infancy: QoS for different streams of traffic. (NMS)

MEMS

- Smart matter: the integration of MEMS actuators and sensors with computation and networks. (seedling, amorphous computation)
- SmartDust: usage of MEMS sensors with wireless, GPS, biochemical sensors and ad-hoc networking to enable distributed detection and tracking of bio-hazards (SensIT)
- Embedded Actuators and Sensors. (ISAT Study Area)

Hard Problems Continued

Quantum Devices (Beyond Si)

- New paradigms for secure communication and computation. (quantum teleportation)
- Quantum, DNA, Smart matter models of computation: Amorphous Computing. Challenge problems: quantum and string theoretic simulations of molecules.
- Molecular electronic quantum devices in computational architectures. (joint with MTO/DSO, ISAT Study Area)

System on a Chip

- Integrate functionally different computational elements ASICs, FPGAs, programmable elements using optical interconnects. (joint with MTO)
- Programmable hardware with verified components for morphing computational elements and power aware applications. (Just-in-Time)

Hard Problems Continued

Cognitive Neurophysiology:

- Interfacing computer memory to human memory, models of memory and forgetfulness to augment situation awareness. (ISAT Study Area)
- Learning of information search patterns and language acquisition. (TIDES, Web in a Box)
- Synthesis of speech, gaze, gesture and lip reading for noisy, multi-speaker environments.

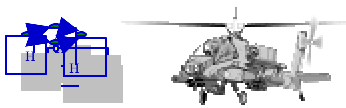
Computational Biology:

- Hidden Markov models for biological models of gene expression and phenotype expression. Putting biological content into phenomenological models.
- Architectures for computation, hardware and software with the fault tolerant and self-organizational character of biological systems.
- Modeling and Control of genetic circuits for applications like suppression of piliation or forced sporulation, multi-grained models of the organism, cell, DNA, gene as a computational element. Modeling of gene expression from gene chip data. (joint with DSO, MTO)

Current ITO Programs

Intelligent Software

- Communicator
- Information Management
- Translingual (TIDES)
- Web-in-a-Box

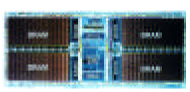


Autonomous and Embedded Systems

- Autonomous Negotiation Targets (ANTS)
- Mobile Autonomous Robot Software (MARS)
- Software Enabled Control (SEC)
- Model-Based Integration of Embedded Software (MoBIES)
- Software for Distributed Robotics (SDR)
- Program Composition for Embedded Systems (PCES)

Networking & Distributed Systems

- Active Networks
- Next Generation Internet (NGI)
- Quorum
- Sensor Information Technology (SensIT)
- Network Modeling and Simulation (NMS)



High Performance Computing Components

- Data Intensive Systems
- PAC/C

Information Survivability

- Tolerant Networks
- Dynamic Coalitions

Expeditions

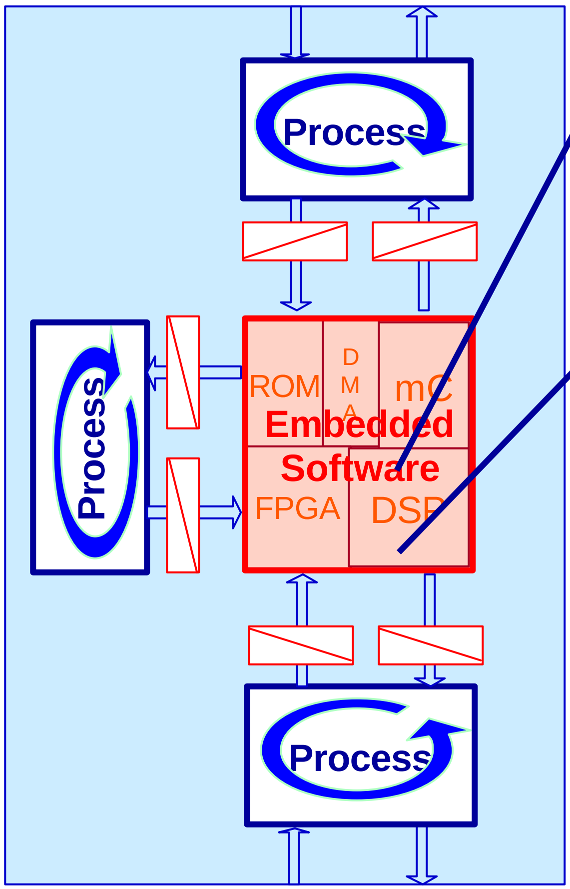
Seedlings

Initiatives in Embedded Software

- **What is the problem?**
- **How can we solve it?**
  - Software and Physics
  - Embracing Change
  - Dynamic Structures

The Technology Challenge

Embedded systems: information systems tightly integrated with physical processes



Problem indicators:

- Integration cost is too high (40-50%)
- Cost of change is high
- Design productivity crisis

Root cause of problems is the emerging new role of embedded information systems:

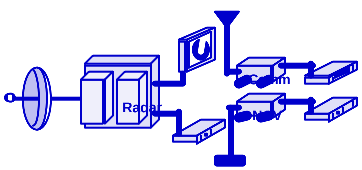
- exploding integration role
- new functionalities that cannot be implemented otherwise
- expected source of flexibility in systems

Problem: Lack of Design Technology aligned with the new role

Problem for Whom?

- ***DoD (from avionics to micro-robots)***
  - *Essential source of superiority*
  - *Largest, most complex systems*
- ***Automotive (drive-by-wire)***
  - *Key competitive element in the future*
  - *Increasing interest but low risk taking*
- ***Consumer Electronics (from mobile phones to TVs)***
  - *Problem is generally simpler*
  - *US industry is strongly challenged*
- ***Plant Automation Systems***
  - *Limited market, conservative approach*

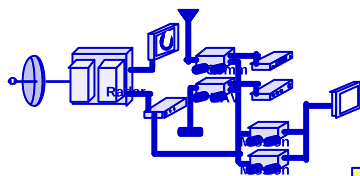
DoD Example: Avionics Systems



Independent Avionics — 64 KB

DEDICATED SUBSYSTEMS

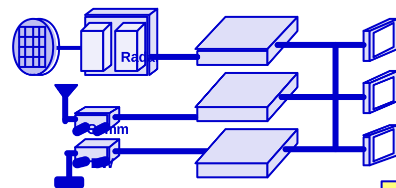
- •Digital Fire Control/NAV
- •PT-PT Wiring
- •Mechanically Controlled Sensors/FLT Controls/ Displays
- •Crew-Dominated Operation



Federated Avionics — 1 MB

FEDERATED SUBSYSTEMS

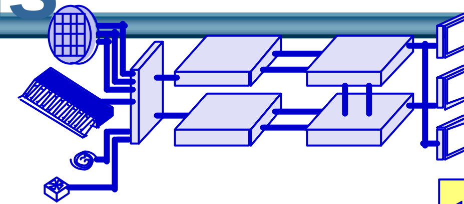
- •Functionally Integrated Data Processing
  - -NAV/WD/Air Data Sensors
  - -Flight Control
- •Beam Steering Sensors
- •Fly By Wire
- •Dedicated Digital Processing
- •Crew-Assisted Operations
  - - Weapon Delivery
  - - Automated TF/TA
  - - EW Response



Integrated Avionics — 100 MB

INTEGRATED SYSTEMS

- •Aircraft-Wide Information Integration
  - - Sensors/Stores/ Vehicle/ Propulsion
- •Modular Electronics
- •Massive Data Bases
  - - Terrain, Threat
- •Digital Sensor Processing
  - - Sensor Fusion
  - - Hyperspectral Imaging
- •Integrated Diagnostics/ System Fault Tolerance
- •System Data Security
- •Limited UAV Autonomy



Advanced Avionics — 1 GB

SYSTEM of SYSTEMS

- •Platform Exploitation of Global Information
  - - Information Mining
  - - At-A-Distance Reconfiguration
- •Autonomous Vehicle Emphasis
  - - Air & Space
- •Air Crew/ Ground Crew Monitoring & Management
- •Automated Functions
  - - ATR (Multi-Sensor)
  - - Failure Prognostics
  - - Route/ Sensor/ Weapon/ Vehicle Coordination
  - - Bistatic Sensing (Air/ Space)
  - - Threat Evasion

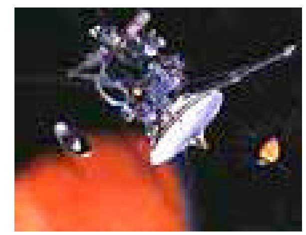
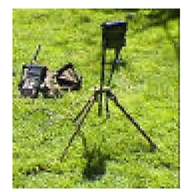
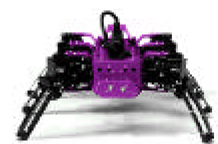
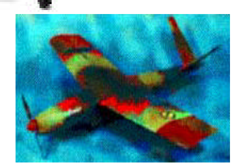
1958 **1950's - 60's** **1970's - 80's** **1990's - 00's** **2000 →**

Source: AFRL

Technology Themes

- **Software and Physics**
  - Establish composability in SW for physical characteristics; System/software co-design and co-simulation environments; New methods for system/code composition
- **Embracing Change**
  - Adaptive Component Technology; Adaptable composition frameworks; QoS middleware for embedded systems
- **Dealing with Dynamic Structures**
  - Property prediction without assuming static structures; Monitoring, controlling and diagnosing variable structure systems;



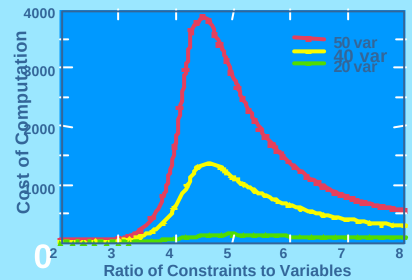
Why Should We Do It?

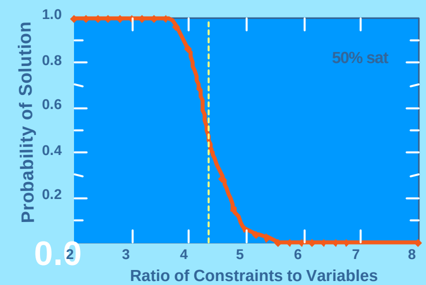
- ***Themes 1 & 2:***
  - *These problems hurt: cost, schedule, performance*
  - *The trend of IT becoming the universal integrator for systems continues and both unstoppable and necessary.*
  - *We have already started the work, have preliminary results and know what to do.*
- ***Theme 3:***
  - *The wave is coming:*
    - *Tremendous progress in MEMS, photonics and, communication technology -* ***we need to build systems now!***
    - *Identified applications with very high expected ROI (prognostic health management, unmanned vehicle fleets, active materials,..)*

Why Can We Make a Difference?

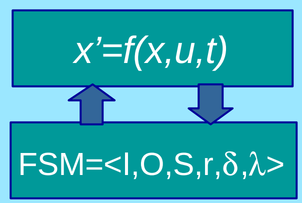
New, critical insights in fundamentals:

Phase transitions have been found in computational requirements for solving fundamental "intractable" problems.



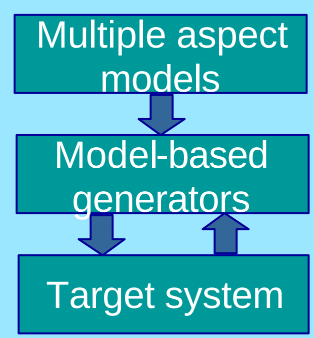


Emerging theory of hybrid systems provides a new mathematical foundation for the design and verification of embedded systems



- model checking
- compositional synthesis
- simulation

Revolutionary changes in software creation: model-based generators, aspect languages, DSL-s offer new foundation for design automation and adaptation.

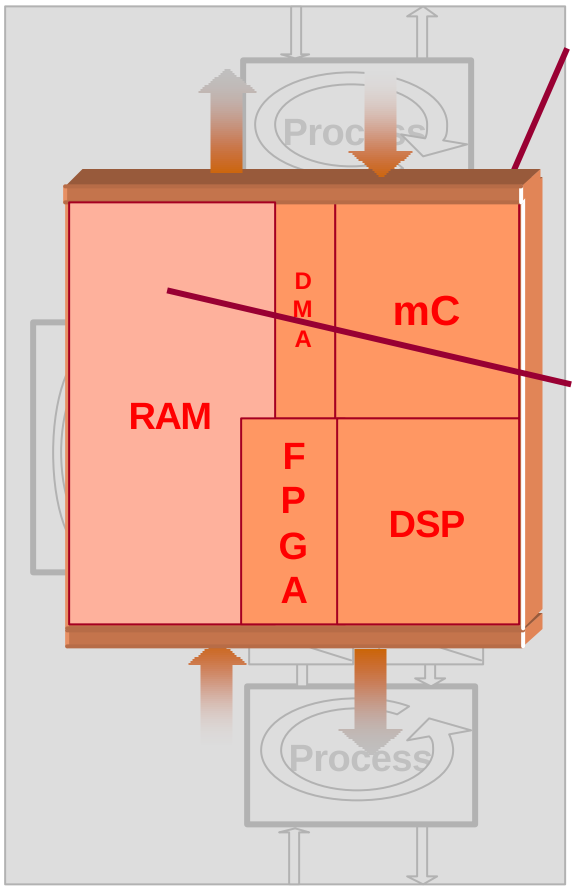


- formal modeling
- verification tools
- automated code synthesis

Theme 1:

Software and Physics

Embedded software: defines physical behavior of a complex nonlinear device



Embedded System: a physical process with dynamic, fault, noise, reliability, power, size characteristics

Embedded Software: designed to meet required physical characteristics

Hard Design Problem:

- Both continuous and discrete attributes (a lot)
- Every module has impact on many attributes (throughput, latency, jitter, power dissipation,..)
- Modules contend for shared resources
- Very large-scale, continuous-discrete, multi-attribute, densely-connected optimization problem

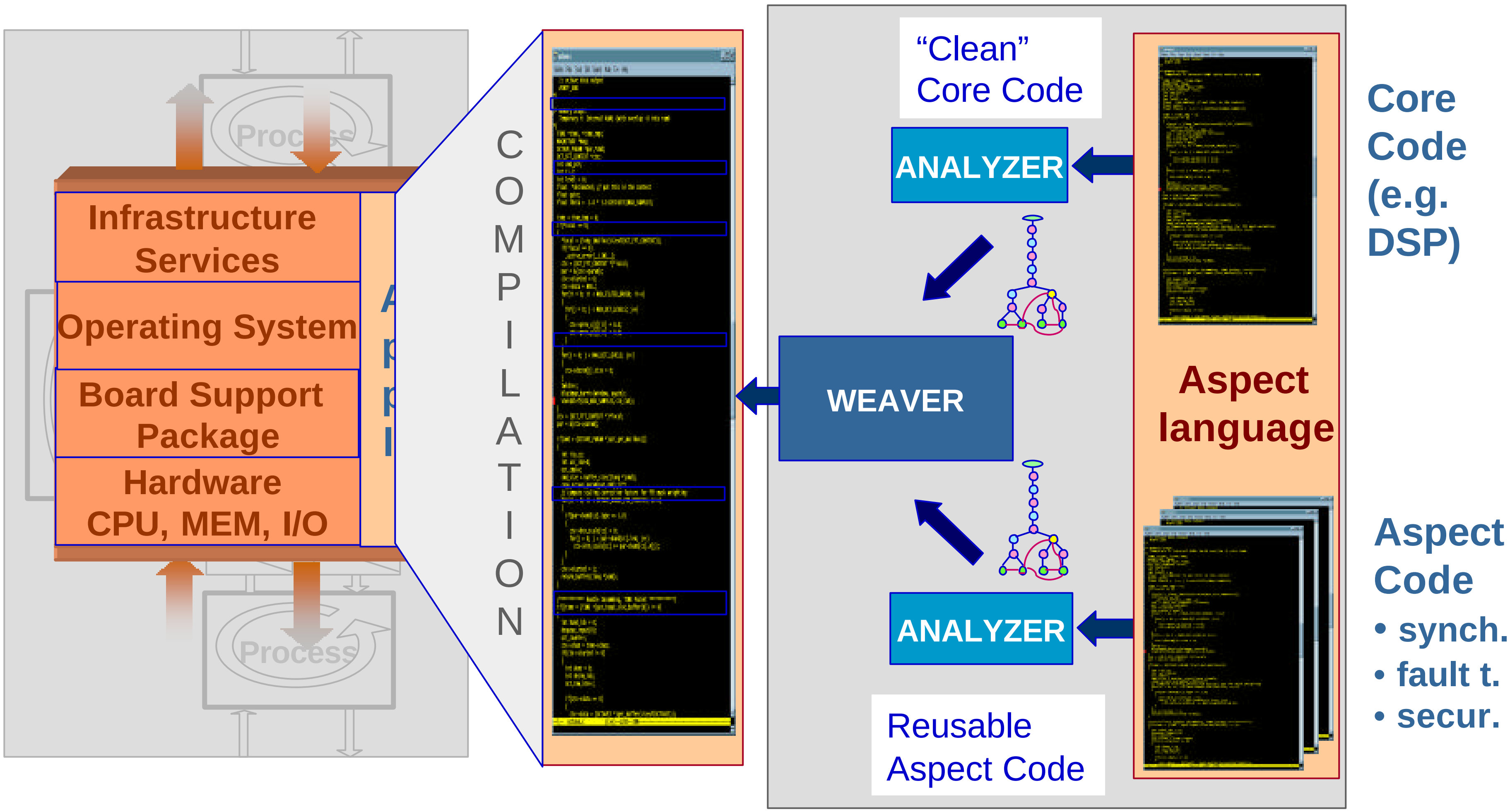
Primary challenge: Cost-cutting physical constraints destroy composability

Technology Gaps

1. System/software co-design, co-simulation and analysis environments
   a. Composition of design automation tool environments
   b. Reusable components for design automation tools
2. New methods for system/code composition
   a. Model-based system composition
   b. Aspect-oriented programming
   c. Domain-specific languages
3. Frameworks and middleware to provide higher level programming abstractions
4. Hybrid optimization and analysis methods

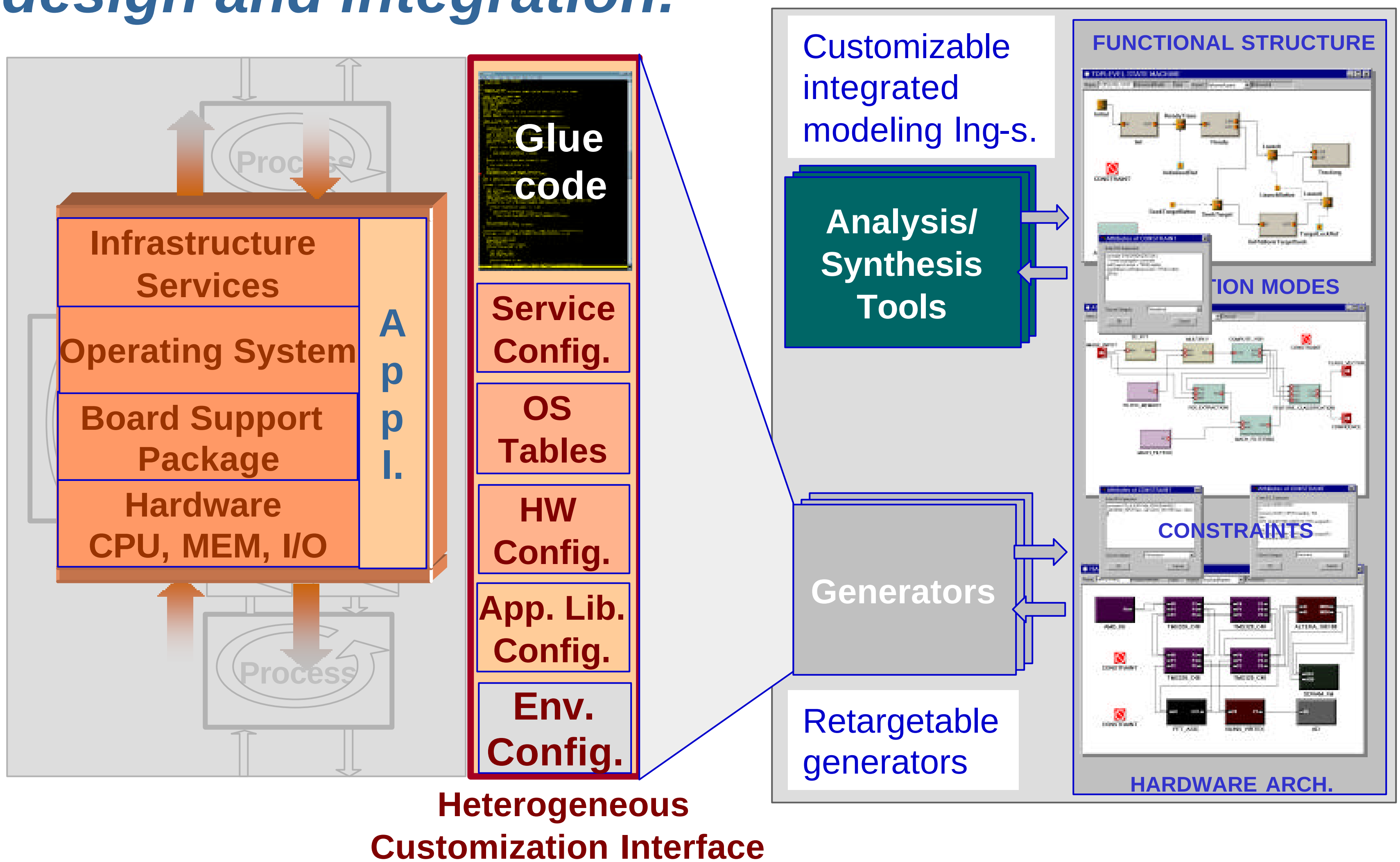
ITO: Program Composition for Embedded Systems (PCES)

Aspect languages will change programming:



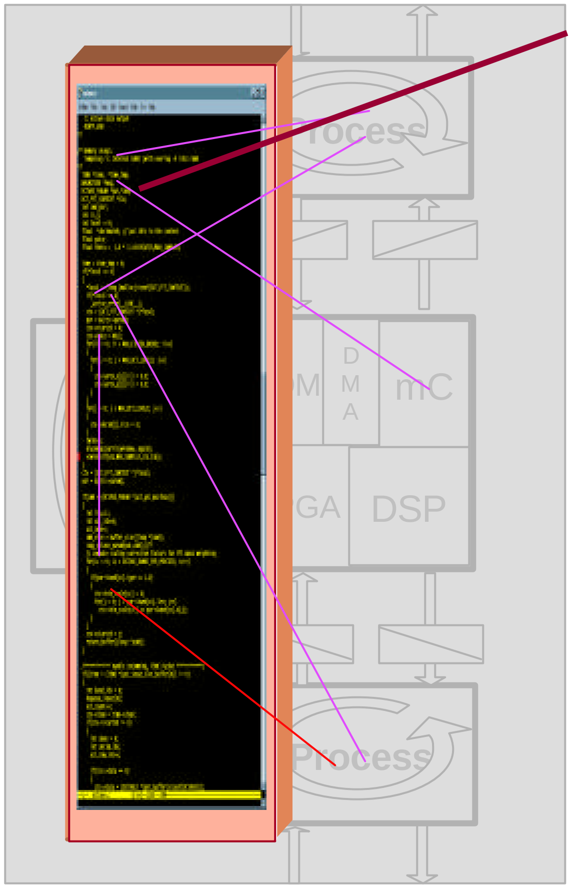
ITO: Model-Based Integration of Embedded Software (MoBIES)

Model-based integration will change system design and integration:



Theme 2: Embracing Change

Source of change: environment, requirements



Hard Problem: due to its integration role, system-wide constraints accumulate in software:
- process properties - algorithms, speed, data types
- algorithms, speed, data types - resource needs
- shared resources - speed, jitter,..

..scattered all over the software.

Condition for managing change:
- constraints need to be explicitly represented
- effects of changes need to propagated by tracking constraints

Flexibility is essentially a SYSTEM-WIDE CONSTRAINT MANAGEMENT PROBLEM

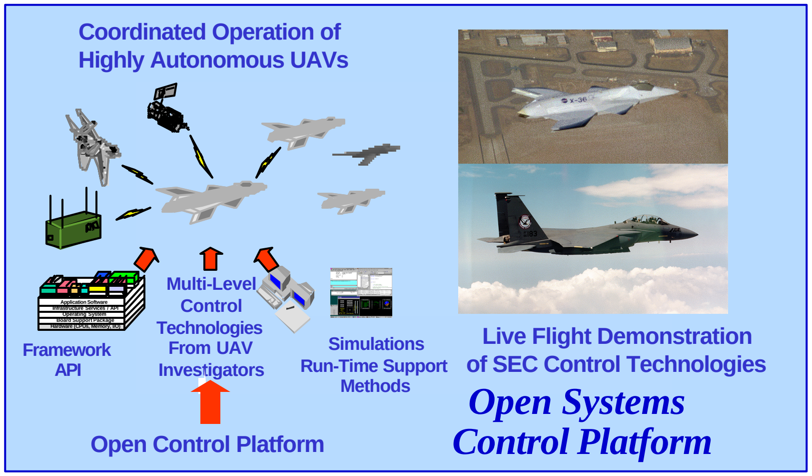
Technology Gaps

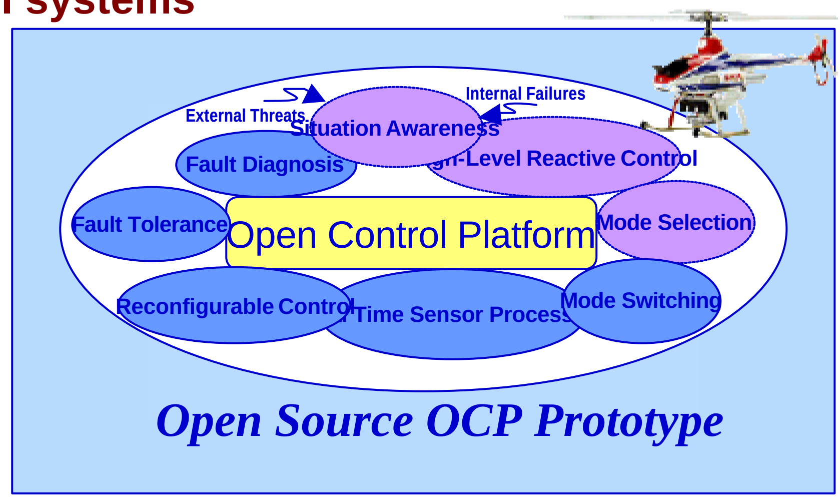
1. Adaptive Components for Embedded Systems
   a. embedded, active models, constraints, generators
   b. adaptive,self-monitoring, embedded software
2. Methods to control flexibility
   a. parametric design
   b. constraint languages
3. Adaptable composition frameworks and QoS middleware
4. Programming Methods to achieve flexibility
   a. dynamic languages
   b. programming through learning

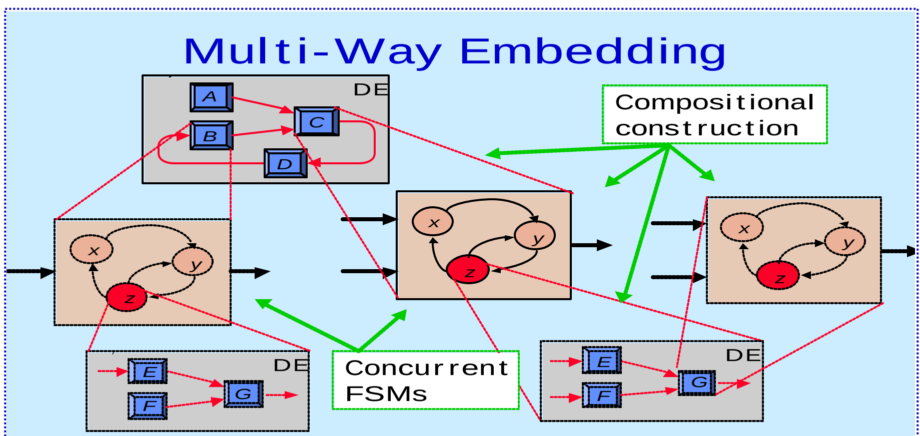
ITO: Software Enabled Control (SEC)

TECHNOLOGY GOALS:

- Control systems that we haven't been able to control before
- Increase automation for extreme maneuvers, tightly coordinated actions
- **Middleware for embedded control systems**





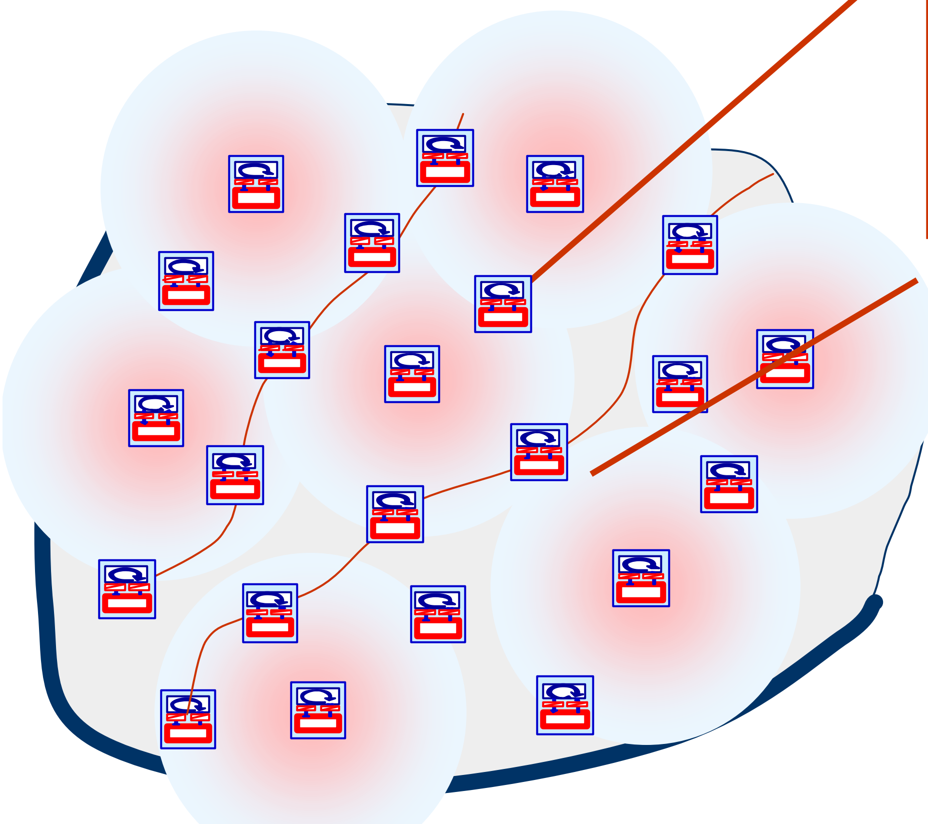


Coordinated Multi-Modal Control:

- **Control middleware (reusable)**
- Open systems, open source
- Reconfigurable hybrid (discrete and continuous) control loops
- Real-time data services for active (predictive) state models

Theme 3: Dealing With Dynamic Structures

Networked embedded systems will change again everything:



LARGE number of tightly integrated, spatially and temporally distributed physical/information system components with reconfigurable interconnection.

Hard Problems:

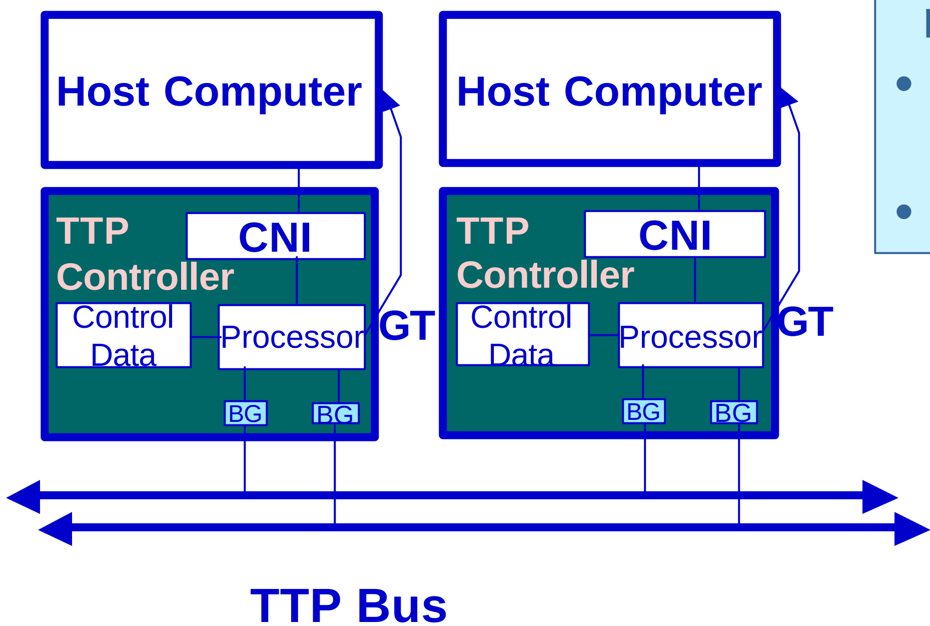
- **Design culture changes:** from analysis and synthesis of static structures to dynamic structures
- **Notion of correctness changes:** from "getting it right" to "keeping it right
- **Richness of interactions changes:** from stable data oriented to dynamic hybrid interactions

Essentially a dynamic aggregation problem

A European Approach to Networked Embedded Systems: Time-Triggered Architecture

Designed for:
- Automotive, drive-by-wire apps.
- Distributed architecture
- Safety critical, hard real-time

- Time division multiple access protocol
- Synchronous
- Fault Tolerant Global Time
- Distributed redundancy management
- Composability for jitter, faults, reliability
- Low-cost implementation



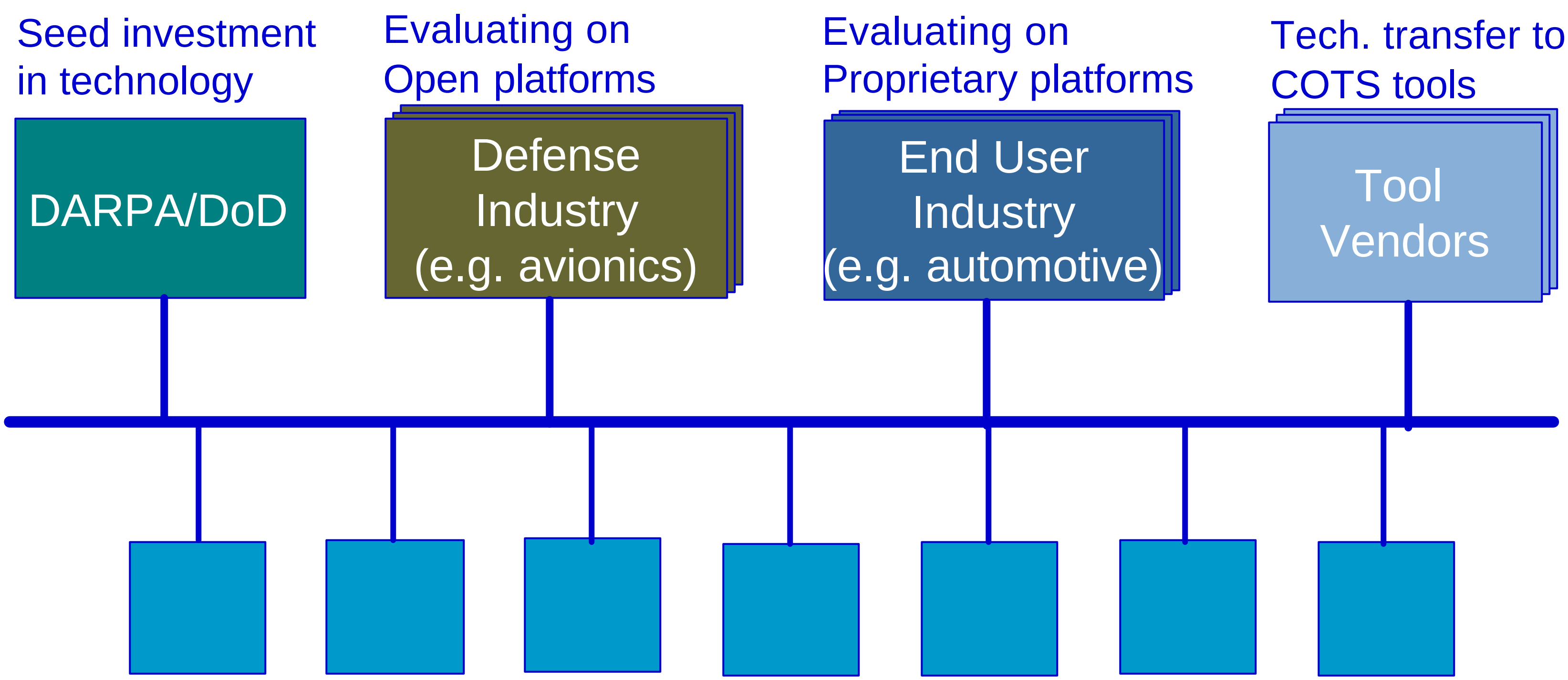
Limitations:
- Static Scheduling
- Overhead cost
- No support for event-driven op.

Same concept as Safebus (in 777 avionics) - but the money is in cars -

How Should We Do It?

- Scope is too big to act alone
- Technology vendors cannot survive on defense industry alone
- Non-defense "end-user" industry has huge stake in progress

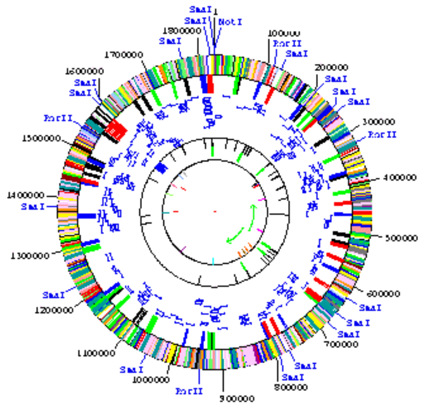
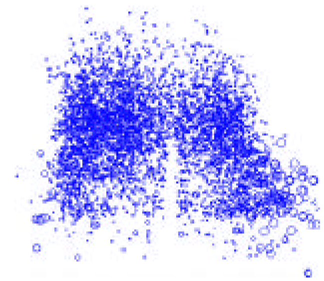
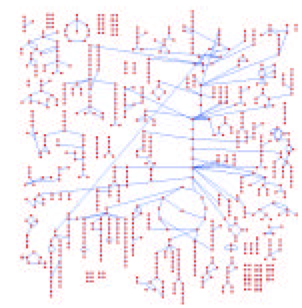
- Evaluate technology on OPEN defense related platforms
- Get buy-in from end-user industry to influence and evaluate technology
- Help tool vendors to absorb technology



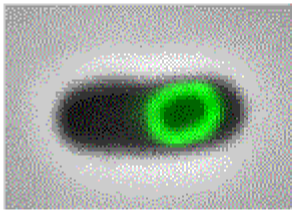
Technology developers (universities, research organizations)

Biofutures:Initiatives in Biocomputing

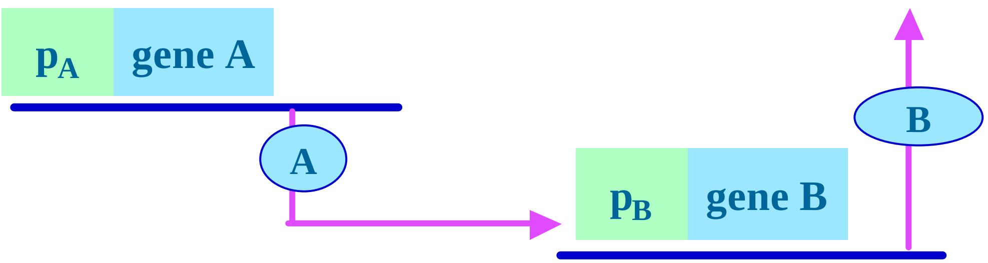
- “Omics”

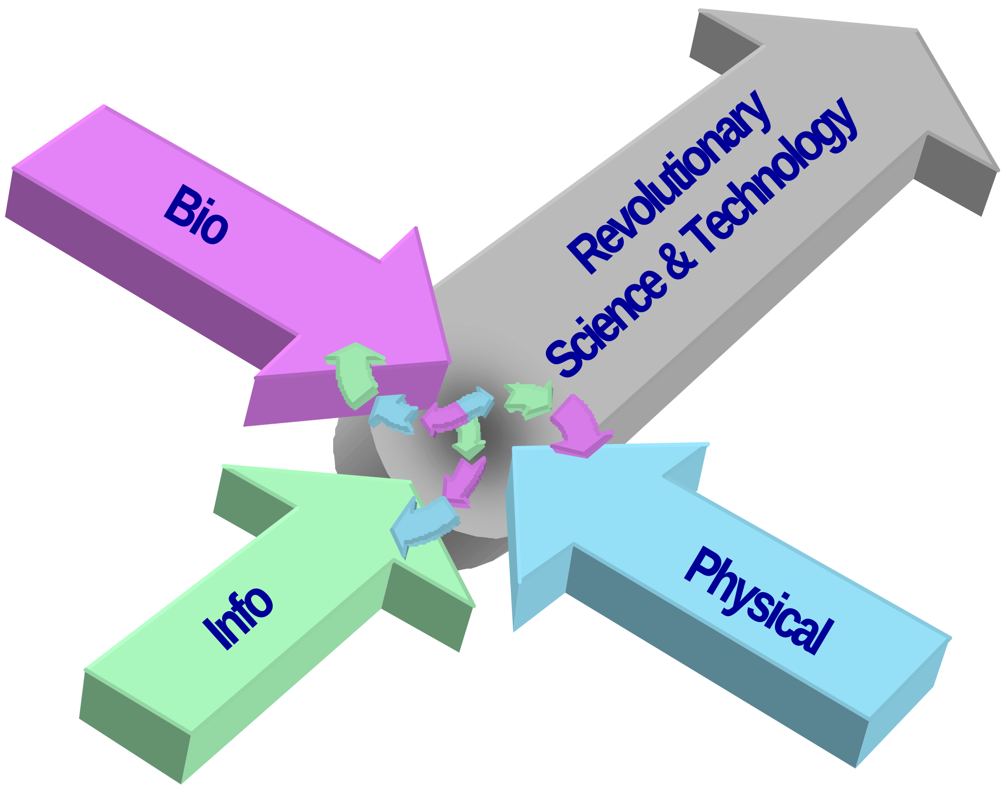
- Single cell biochemistry



- System dynamics



[Bio:Info:Physical]



Bio

- Biological Sciences and Technology

Info

- Computer Science and Information Technology

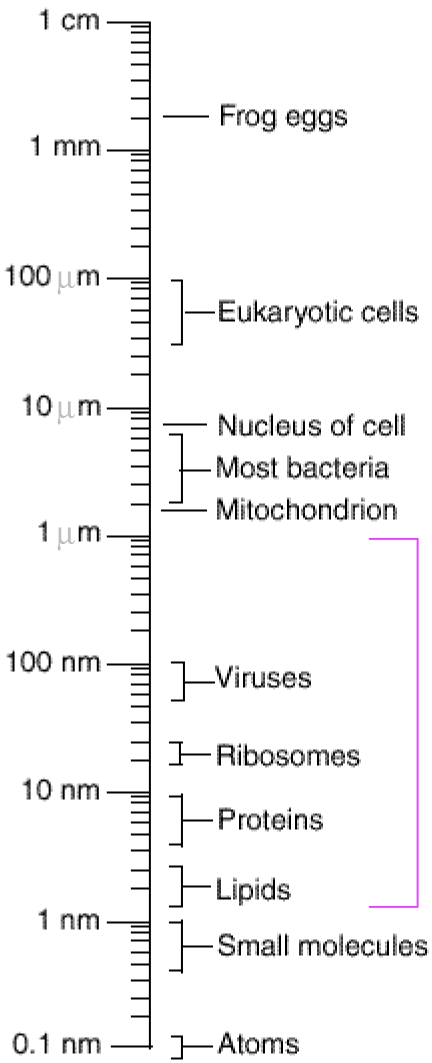
Physical

- Microelectronics, Optoelectronics, Sensors, Actuators, and Microsystems

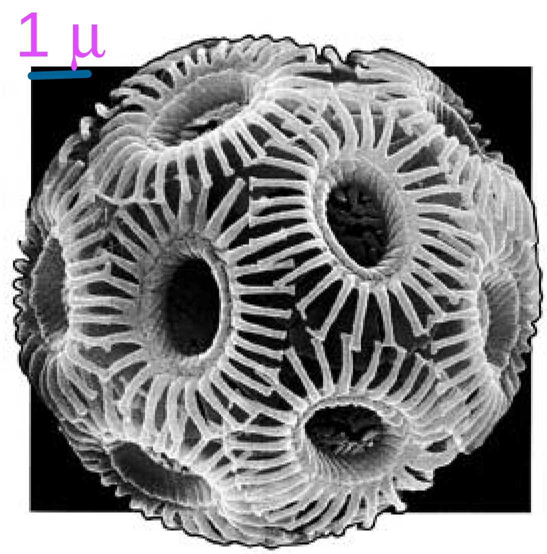
[Bio] « [Physical]

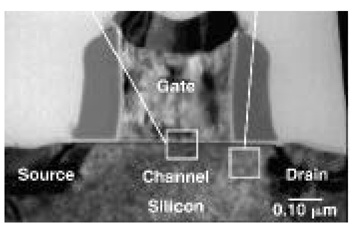
- **Physical Opportunities for Bio**
  - Interfaces to micro/nano scale objects
  - Integrate multiple functions
  - Scale to larger structures
- **Bio Opportunities for Physical**
  - Functions beyond Physical
  - Reliable functions, unreliable devices
  - Self-replicating/organizing, ... devices
- **Bio Challenges for Physical**
  - Interfaces to new kinds of objects
  - Integrate new functions
  - Scale to new structures
- **Physical Challenges for Bio**
  - Engineering devices based on Bio
  - Engineering collections of Bio devices
  - Engineering interfaces between Bio and Physical

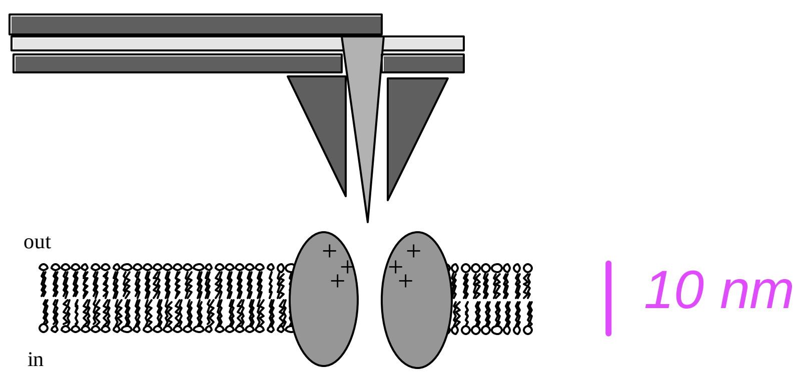
Physical Maps to Bio



CMOS elements



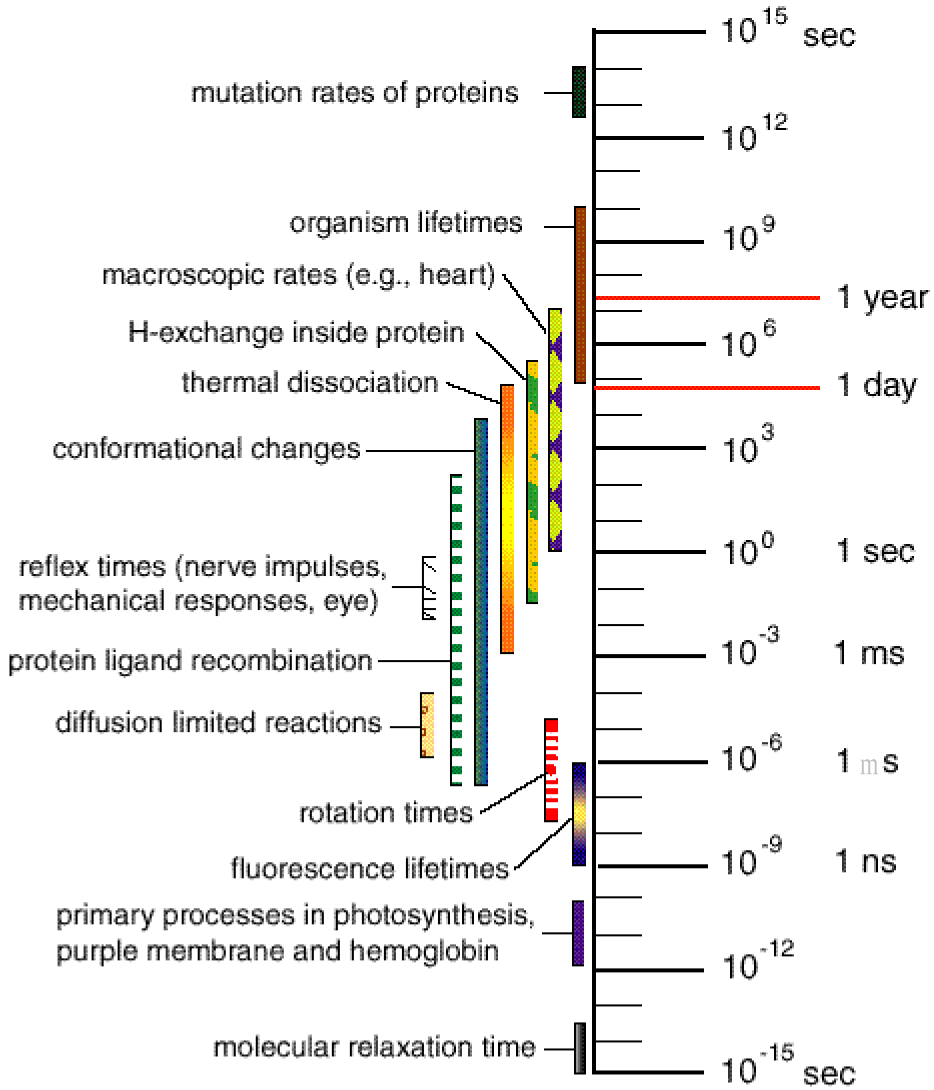




[Bio] « [Info]

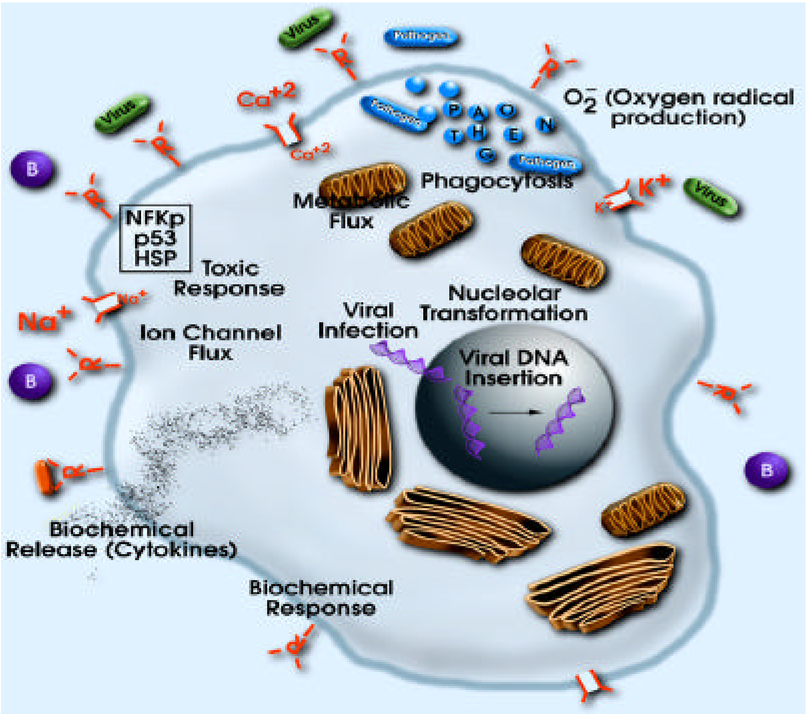
- **Info Opportunities for Bio**
  - Complex systems design
  - Scalable parallel and distributed systems
  - Computational complexity theory
- **Bio Opportunities for Info**
  - Beyond frontier of Info
  - Reliable systems, unreliable sub-systems
  - Self-replicating / organizing systems
- **Bio Challenges for Info**
  - Bio system complexity
  - Bio systems scale
  - Bio systems computational theory
- **Info Challenges for Bio**
  - Filling the gap from lowest to highest levels
  - Recognizing effective system structures
  - Realizing system scaling and stability

Info *vs.* Bio in the Speed Domain



High performance computing

Bio vs. Info at the Single Cell level



~100 μ

- **Specified by**
  - 4 x 109 base pairs (232)
  - 105 proteins (217)
- **Features**
  - Regulation & adaptation
  - Hierarchical self-assembly
  - Parallel processing
  - "Just-in-time" processing
- **Multicellular systems are even cleverer**

Teraflop device in Sandia occupies 10^{14} m^2

Example Areas of Interest to DSO/ITO/MTO

- Development of biologically inspired algorithms and models for computation
- Biologically inspired engineering of complex artificial systems across vastly different size scales from nanometers to meters
- Cellular engineering of synthetic production of materials and chemicals, e.g., application of concepts of developmental biology to biosynthesis of composite materials
- Computational neural science applied to human-systems interaction, addressing both measurement of neural phenomena to control systems and systems control of neural phenomena

Examples, continued

- Engineering novel chemical and biological reagents for micro-scale assays
- Platforms for studying and mimicking biological processes such as synthesis, signaling, regulation and control
- Modeling and simulation of the dynamic behavior of complex biological systems (from the molecular level to the population level)
- Techniques and micro-scale platforms for nondestructively interrogating cells and subcellular components in real-time to help decipher the equivalent of the device physics of biological systems
- Application of language modeling to problems in biological discovery for identifying therapeutic and diagnostic targets based on genomic data

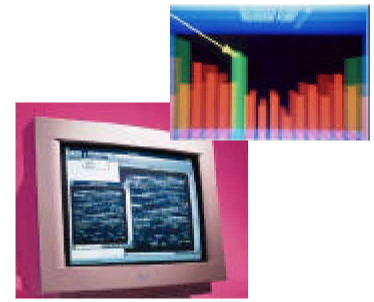
ITO BioExpeditions

High-speed DNA & protein analysis

- Rapid identification of biological organisms
- Greatly speed up almost all areas of biological research

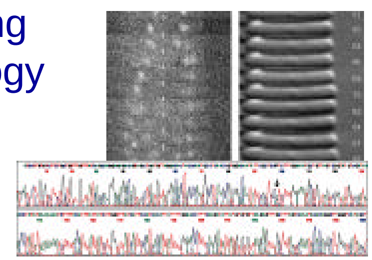
Dynamic cellular behavior modeling & computational tools development

- Greatly increase the understanding of biological systems
- Facilitate the ability to easily “program” biological artifacts



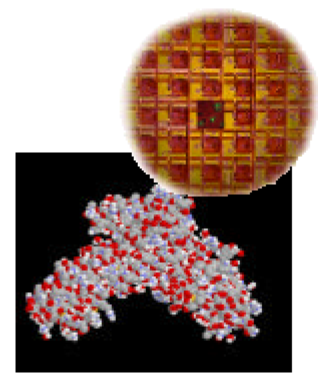
Extraction of useful information from systematically generated data

- Programming of biological cells resulting in truly predictive biology
- Rapid identification of genes to be targeted in new pathogens



Measurements at the molecular level

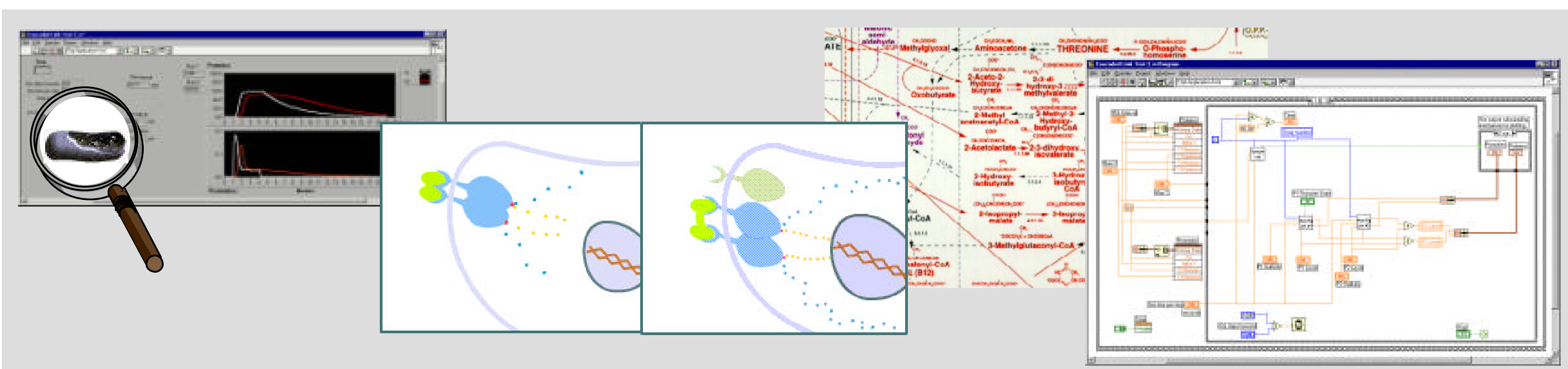
- Develop technology that will enable the detection of extremely small concentrations of target molecules of interest to DoD



BioSpice: Cellular Info Processing

- **Extension of the Bio/Spice genetic regulatory network modeling system to analyze the dynamic behavior of the Caulobacter cell cycle regulatory network**
  - The Bio/Spice modeling system will be allowed to deal with multiple spatial compartments and spatially localized proteins
  - The enhanced systems would be applied to Caulobacter cell cycle regulatory systems and to other bacterial switching nodes

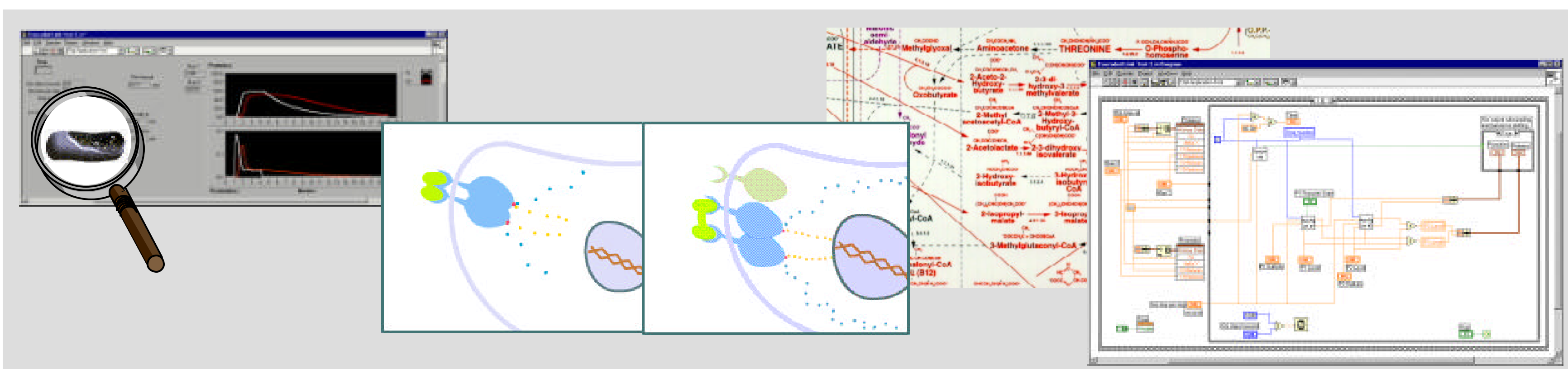
This approach would provide an "Alpha Test Site" for the Bio/Spice modeling system



BioSpice: Cellular Info Processing

- **Development of new computational tools to search through genetic regulatory network data and conjoin it with other kinds of genomic data.**
  - Study of genetic networks using interaction matrices, Markov chain models
  - Study of networks using protein mass spectrometry
  - Use of peptide aptamers
- **Can enhance ability to do biology as well as improving methods for other research**

This can result in the ability to build cellular information handling devices



ITO: Expeditions

The next grand challenge for IT research

To make computers pervasive, easy to interact with, ubiquitous and able to organize for collaborative or hierarchical situation assessment and decision making:

Involves pointwise advances in program areas with some overlap with existing programs as well as architectural development.

- MEMS and hardware devices
- Scalable computing architectures
- Networked-oriented operating systems
- Distributed file systems
- Data management systems
- Security/privacy
- User interfaces
- Collaboration applications
- Intelligent learning systems
- Program verification
- Methodologies for HW/SW design/evaluation

Expeditions: Ubiquitous Computing

Portolano/Workscape Expedition in Invisible Computing

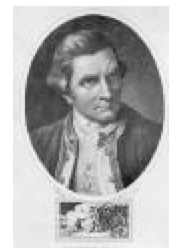
University of Washington

MIT Project Oxygen

L C S

lab @ MIT

Endeavour Expedition: 21st Century Computing to the eXtreme



UC Berkeley

The Aura Project

From Workstations to Invisible Computing

Carnegie Mellon University

Infosphere Project

Georgia Tech / Oregon Grad

Ubiquitous Computing

U Washington - Portolano/Workscape

- **Invisible user interfaces**
  - user-centered and task focused
  - devices of all shapes, sizes, and functions
- **Universal connectivity**
  - diverse physical layers
  - middle-ware and application development to support distr. systems
  - power thrifty, intermittent connections
- **Intelligent services**
  - instrument physical environment, people, and objects
  - automatic agents to make them all communicate
  - off-load as many as possible of users' tasks/concerns

Ubiquitous Computing

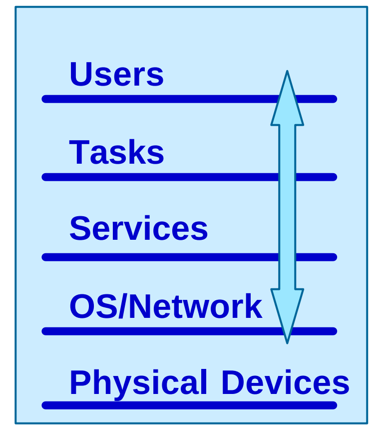
- **A pervasive global information grid is supported by**
  - invisible software services, computers, sensors, and actuators
  - embedded throughout the environment
  - automatically and optimally self-configured to support users' tasks
- **Adaptation: Body computers and personalized agents negotiate with room computers to provide access to all personal information.**
- **Collaboration: Geographically-distributed team members interact with computers and each other using audio and visual inputs in their native languages in real time.**
- **Mobility: As you move outdoors, information and services appear on available facilities without losing task context.**

New Ideas:
Computing to the eXtreme

- Systems Architecture for Vastly Diverse Computing Devices (MEMS, cameras, displays)
- Wide-area “Oceanic” Data Information Utility
- Sensor-Centric Data Management for Capture and Reuse (MEMS + networked storage)
- Negotiation Architecture for Cooperating Components (Composable system architecture)
- Tacit Knowledge Infrastructure to support High-Speed Decision-Making
- UI Design Tools

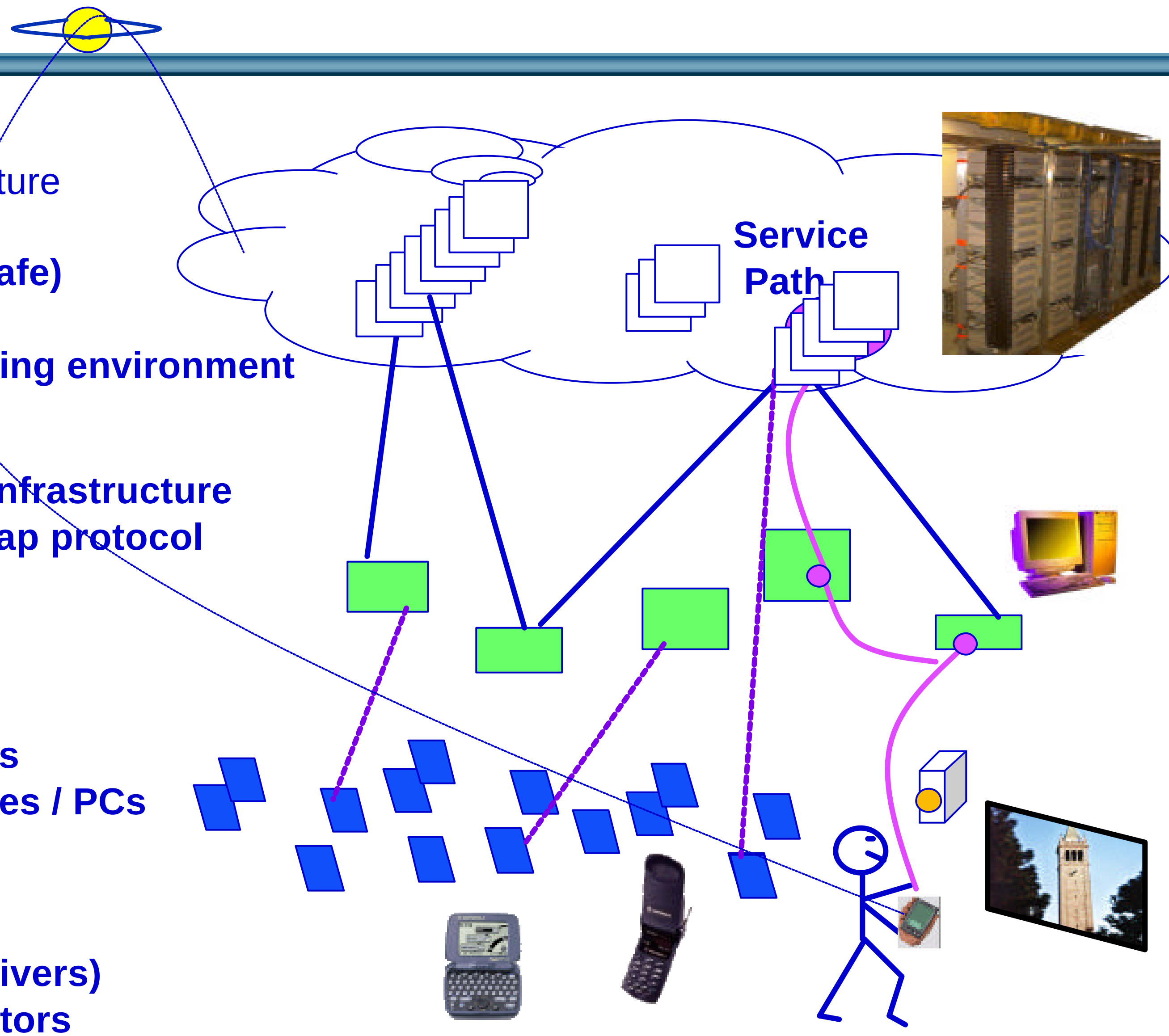
New Ideas:
Power/ Energy Aware Computing

- **Goal: Maximize computation per battery charge**
  - Processor speed setting to adapt to computer architectural bottlenecks (e.g. cache, memory bus) and non-ideal battery behavior
- **Impact:**
  - Over four orders of magnitude difference in energy to perform the same task as a function of User Interface, Data Representation, Compute versus transmit trade-off
- **Approach: Embedded Network Proxies**
  - Reduce bits transmitted
  - Adaptive scheduling to smooth demand. Delivered battery capacity determined by peak rather than average demand



Small Device Operating Systems

- Base Scalable Infrastructure
  - **highly available**
  - **persistent state (safe)**
  - **databases, agents**
  - **service programming environment**
- Active Proxies
  - **connected to the infrastructure**
  - **soft-state, bootstrap protocol**
  - **transcoding**
- Ubiquitous Devices
  - **billions**
  - **sensors / actuators**
  - **PDAs / smartphones / PCs**
  - **heterogeneous**
- Service Paths
  - **aggregate flows (rivers)**
  - **transcoding operators**

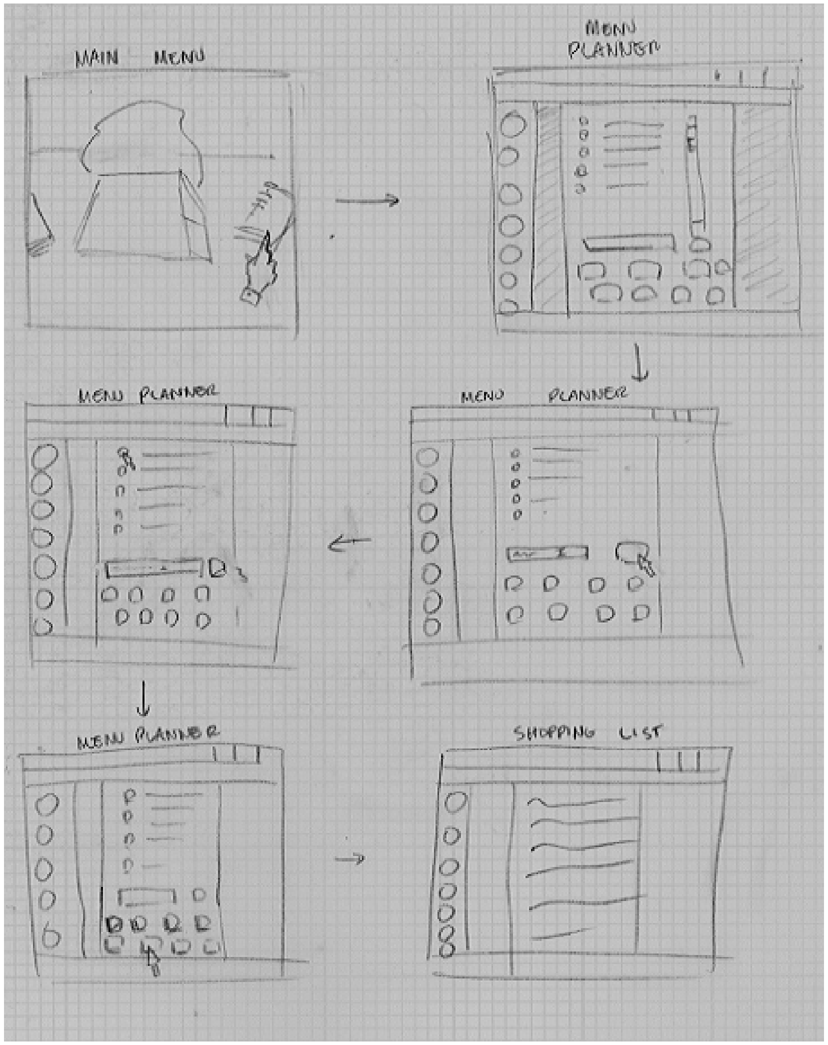


Ubiquitous Devices Þ Ubiquitous Storage

- **Consumers of data move, change from one device to another**
- **Properties** REQUIRED **for Pervasive Computing substrate:**
  - Strong Security: data must be encrypted whenever it is in the infrastructure
  - Coherence: too much data for naïve users to keep coherent “by hand”
  - Automatic replica management and optimization: huge quantities of data cannot be managed manually
  - Simple and automatic recovery from disasters: probability of failure increases with size of system
  - Utility model: world-scale system requires cooperation across administrative boundaries

Multimodal Design Tools Should Support

- **Rapid production of “rough cuts”**
  - don’t handle all cases
  - informal techniques
    - sketching/storyboarding
    - “Wizard of Oz”
  - iterative design
    - user testing/fast mods
- **Generate initial code**
  - UIs for multiple devices
  - designer adds detail & improves interaction
  - programmers add code



UC Berkeley - Endeavour Project

Application 1: Commandscape

- **Spontaneous formation of a command post, for example**
  - example, Temporary Military Base (take over an old building)
- **Some issues**
  - Collection of information
  - Distribution of commands
  - Making sense of large amounts of data

U Washington - Portolano/Workscape

- Command posts, war rooms of the future to be developed in cooperation with ISO.

Application 2: Labscape

- **Lab results are hard to reproduce**
  - need to track everything
- **May need to collaborate with remote groups**
- **Tacit collection and distribution**
- **Data representation and presentation is key to making
  use of complex data-sets**

Infosphere

Problem: too many sources, too much information