Multiple Independent Levels of Safety & Security (MILS): High Assurance Architecture

Gordon M. Uchenick
Sr. Mentor/Principal Engineer

OBJECTIVE INTERFACE

What is High Assurance?

- To the FAA:
  - One failure per 10^9 (1 Billion) hours of operation
    - How long *is* a Billion hours? Do the math!
      - $1{,}000{,}000{,}000\,\text{hours} \times \frac{1\,\text{day}}{24\,\text{hours}} \times \frac{1\,\text{year}}{365.25\,\text{days}}$
      - 114,077 *YEARS!*

- For National Security Systems processing our most valuable data under most severe threat:
  - Failure is *Unthinkable*

- ***How do we implement systems that we can trust to be this robust?***

Industry Standards

- RTCA DO-178B, *Software Considerations in Airborne Systems and Equipment Certification*
- ARINC-653, *Avionics Application Software Standard Interface*
- ISO-15408, *Common Criteria for Information Technology Security Evaluation*
- DCID 6/3, *Protecting Sensitive Compartmented Information Within Information Systems*

Assurance Certification Goals

Common Criteria	***MSLS / MLS Separation Accreditation***
Basic Robustness (EAL3)	System High Closed Environment
Medium Robustness (EAL4+)	System High Open Environment
High Robustness (EAL6+)	Multi Level Separation
DCID 6/3 Protection Level 5	***Multi Nation Separation Accreditation***
DO-178B Level A	***Failure is Catastrophic***

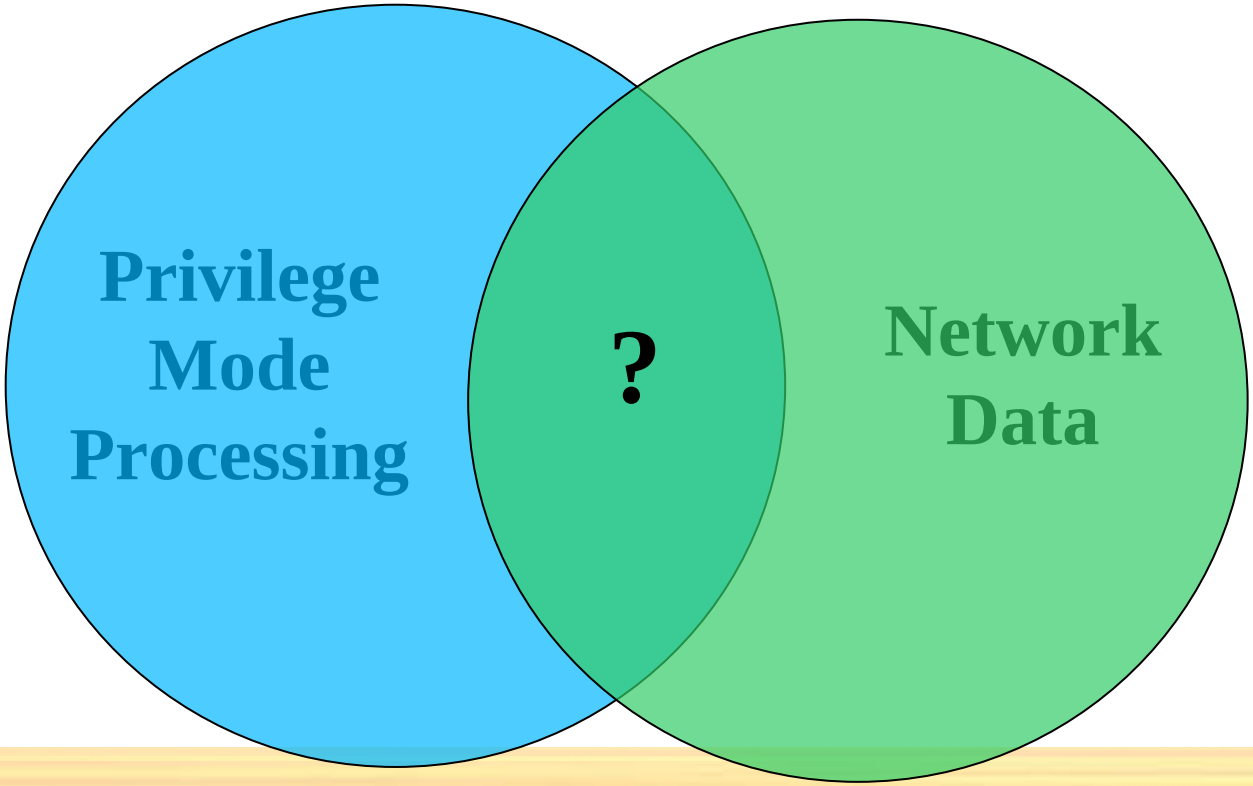
Fail-first, Patch-later

- Most commercial computer security architectures
  - The result of systems software where security was an afterthought
    - Operating systems
    - Communications architectures
  - **Reactive** response to problems
    - Viruses, Worms, and Trojan Horses
    - Hackers and Attackers
    - Problems are only addressed ***after*** the damage has been done

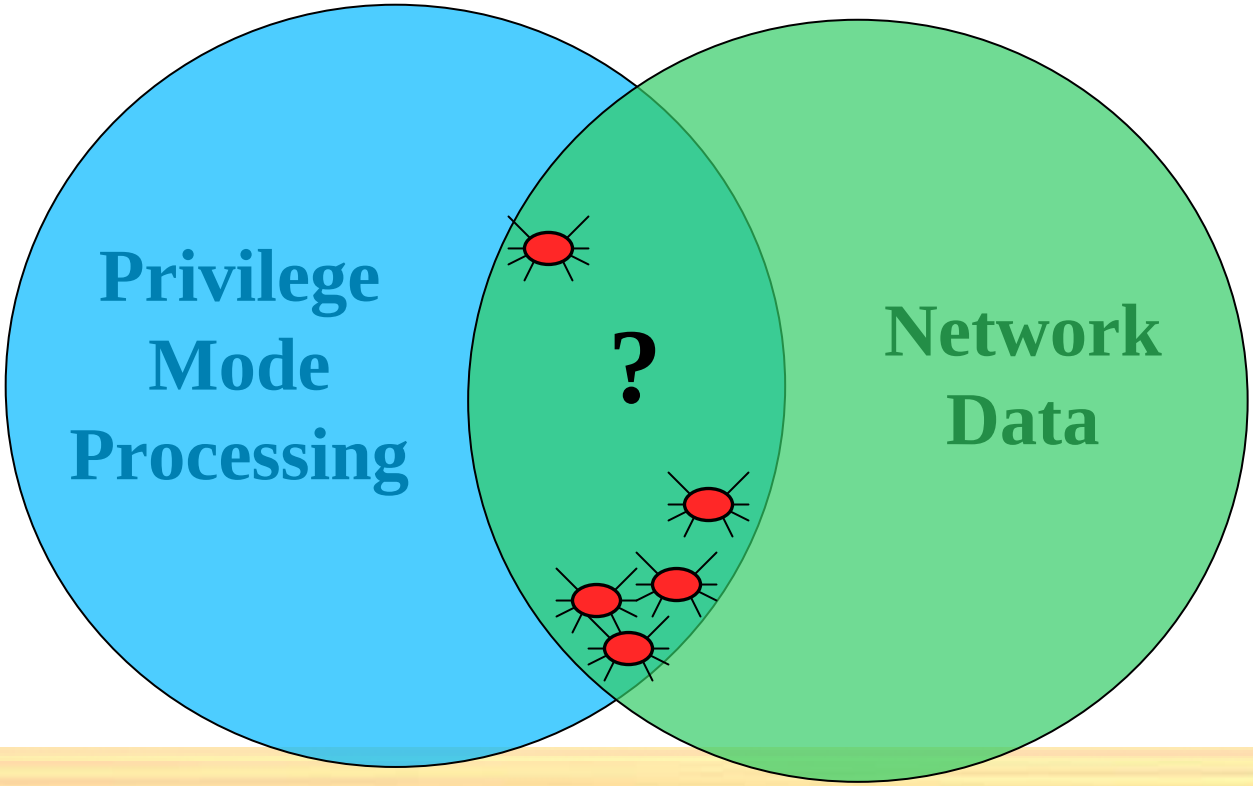
- Inappropriate approach for mission critical systems
  - Does not safeguard information or the warfighter
  - **Proactive** measures are required to ***prevent*** damage

Fail-first, Patch-later (cont.)

- **Reactive** approach failures:
  - How many PC anti-virus programs can detect or quarantine malicious device drivers?
    - *None!*
  - What can an Active-X web download do to your PC?
    - *Anything!*

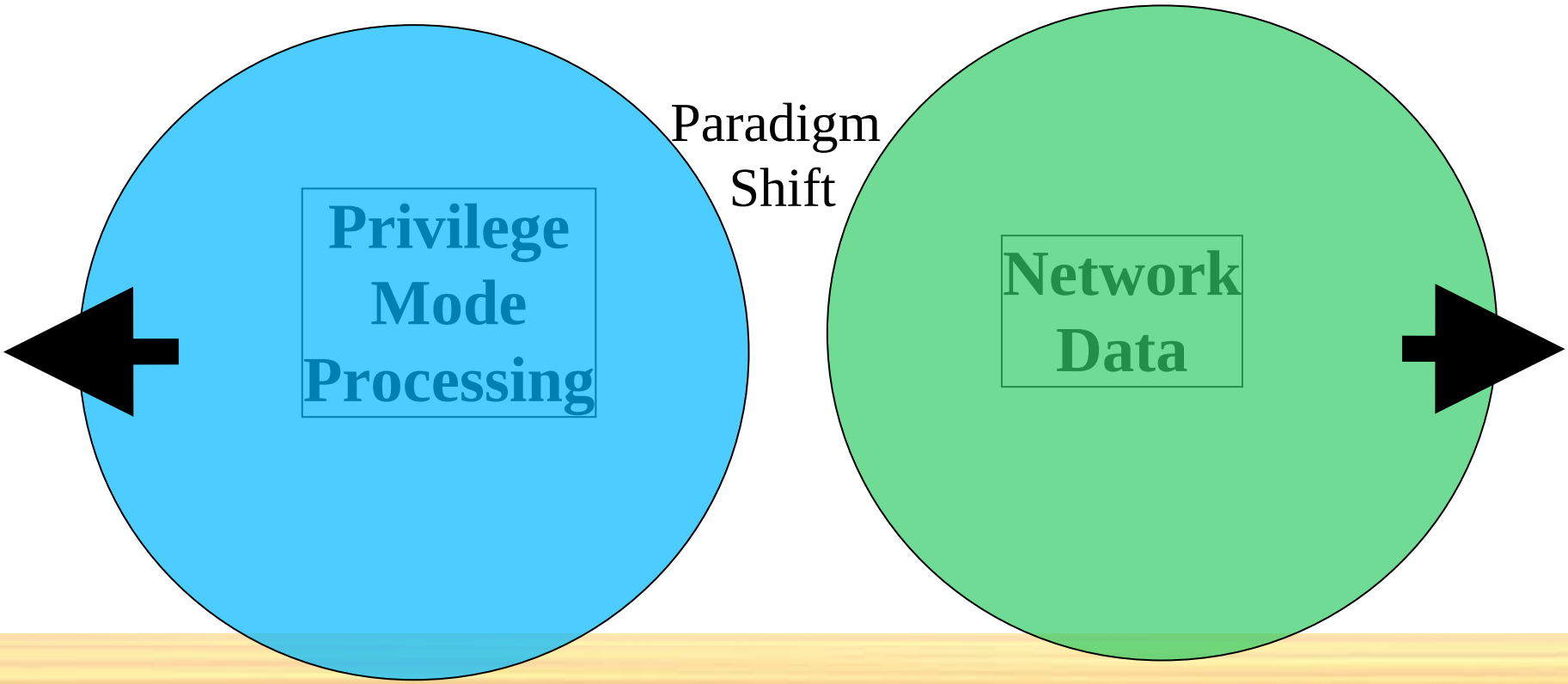


What happens when network data is processed in privilege mode?



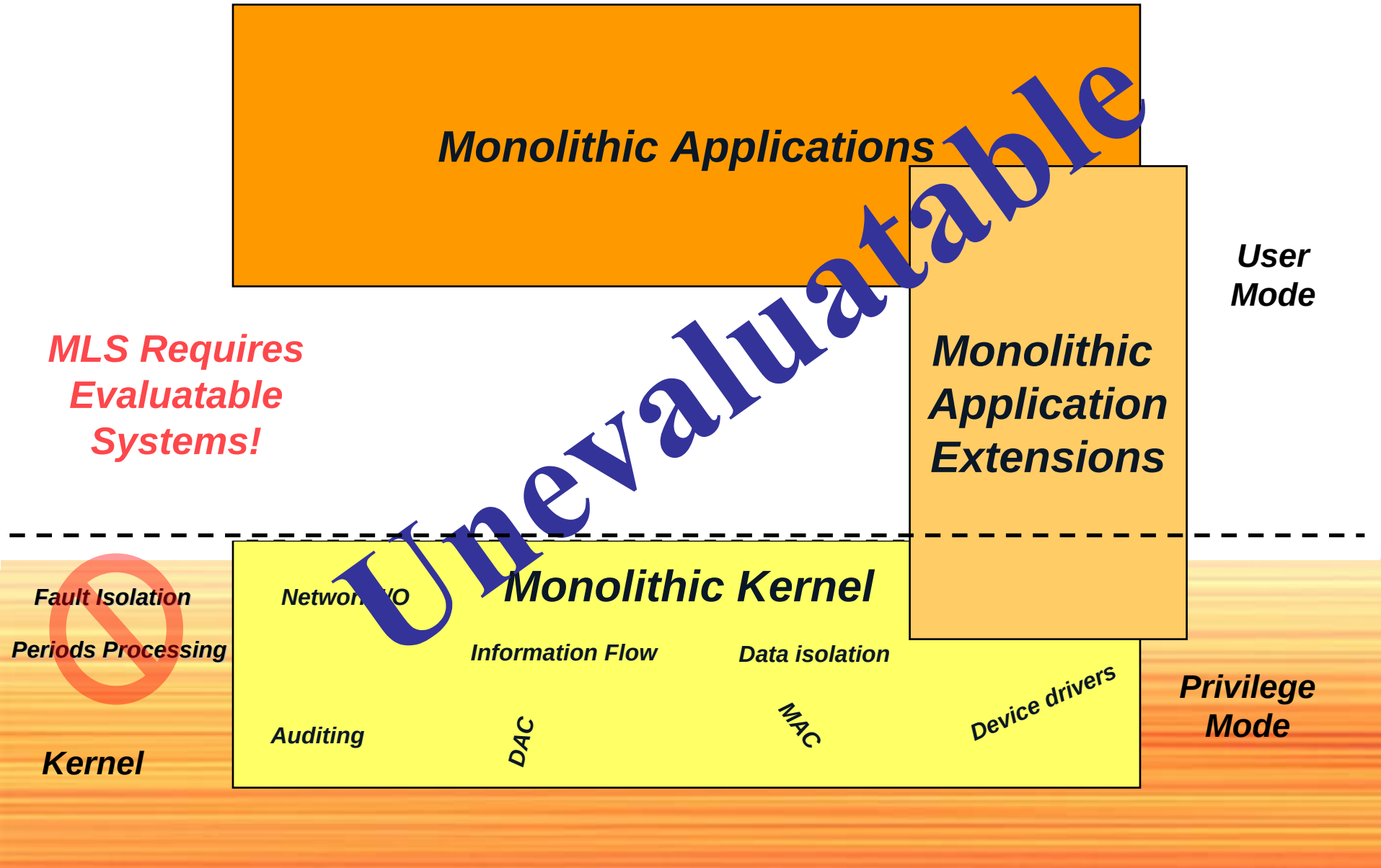
Wild Creatures of the Net: Worms, Virus, . . .

MILS Security Policy



Under MILS Network Data and Privilege Mode Processing are Separated

Where We've Been: Starting Point for Architectural Evolution



Really very simple:

- Dramatically reduce the amount of *safety/security critical code*

So that we can

- Dramatically increase the scrutiny of *safety/security critical code*

The MILS Architecture

Three distinct layers (John Rushby, PhD)

- Separation Kernel
  - Separate process spaces (partitions)
  - Secure transfer of control between partitions
  - Really small: 4K lines of code
- Middleware
  - Application component creation
  - Provides secure end-to-end inter-object message flow
    - Device Drivers, File Systems, Network Stacks, CORBA, DDS
- Applications
  - Implement application-specific security functions
    - Firewalls, Cryptomod, Guards, Mapplet Engine, CDS, Multi-Nation Web Server, etc.

The MILS Architecture (cont'd)

Separation Kernel

- **Microprocessor Based**
  - Time and Space Multi- Threaded Partitioning
  - Data Isolation
  - Inter-partition Communication
  - Periods Processing
  - Minimum Interrupt Servicing
  - Semaphores
    - Synchronization Primitive's
  - Timers

And nothing else!

MILS Middleware

- **Traditional RTOS Services**
  - Device Drivers
  - File Systems
  - Token and Trusted Path
- **Traditional Middleware**
  - CORBA (Distributed Objects)
  - Data Distribution (Pub-Sub)
  - Web Services
- **Partitioning Communication System (PCS)**
  - Global Enclave Partition Comm
    - TCP, UDP, Rapid-IO, Firewire, …
  - Partition Based Attestation

Why Does Neatness Count?

N
E
A
T

Safety and Security enforcing functions must be:

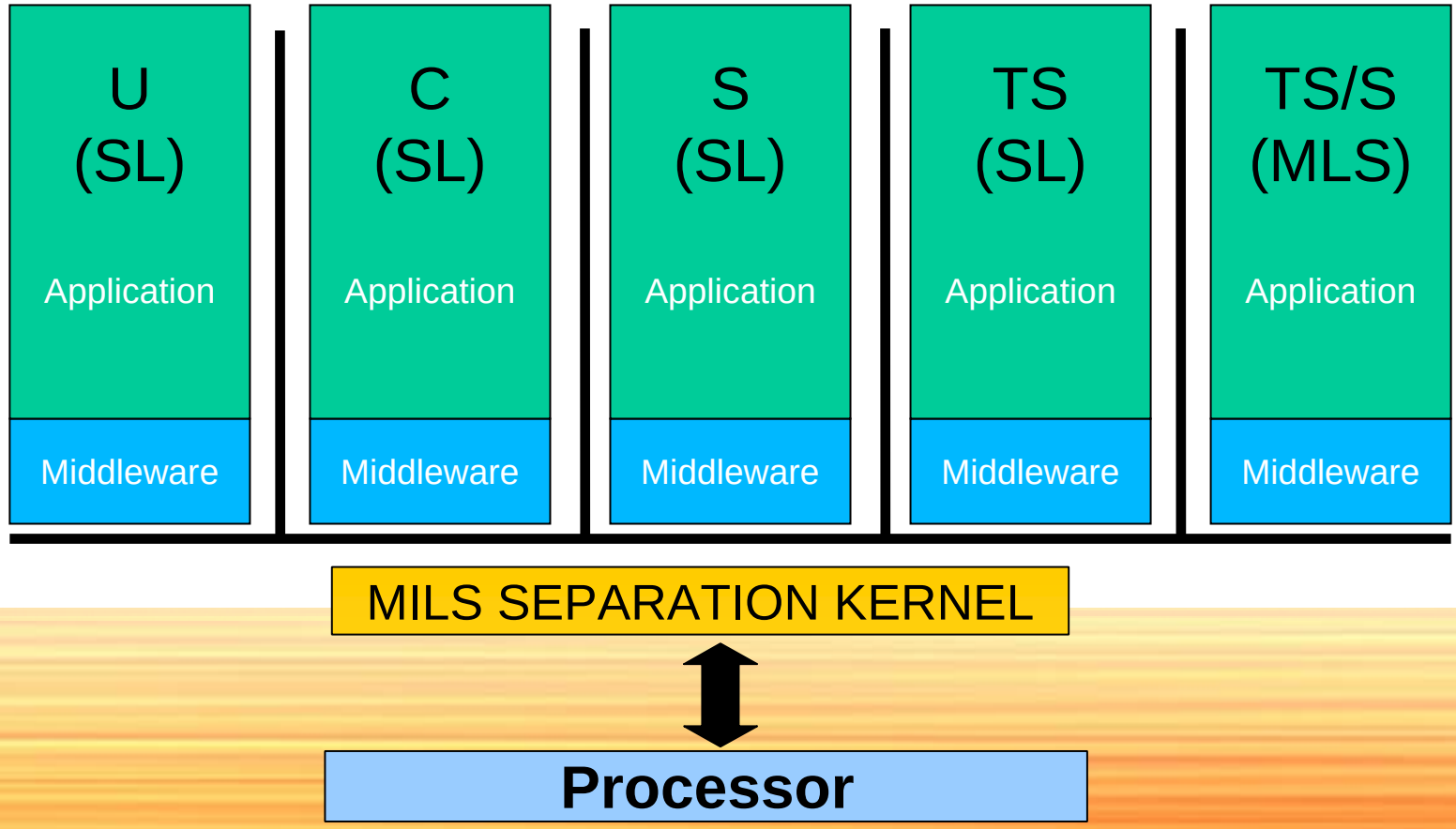
- ***N***on-bypassable
  - Enforcing functions cannot be circumvented
- ***E***valuatable
  - Enforcing functions are small enough and simple enough for mathematical verification
- ***A***lways Invoked
  - Enforcing functions are invoked each and every time
- ***T***amperproof
  - Subversive code cannot alter the enforcing data or functions

MILS Architecture Evolution

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture

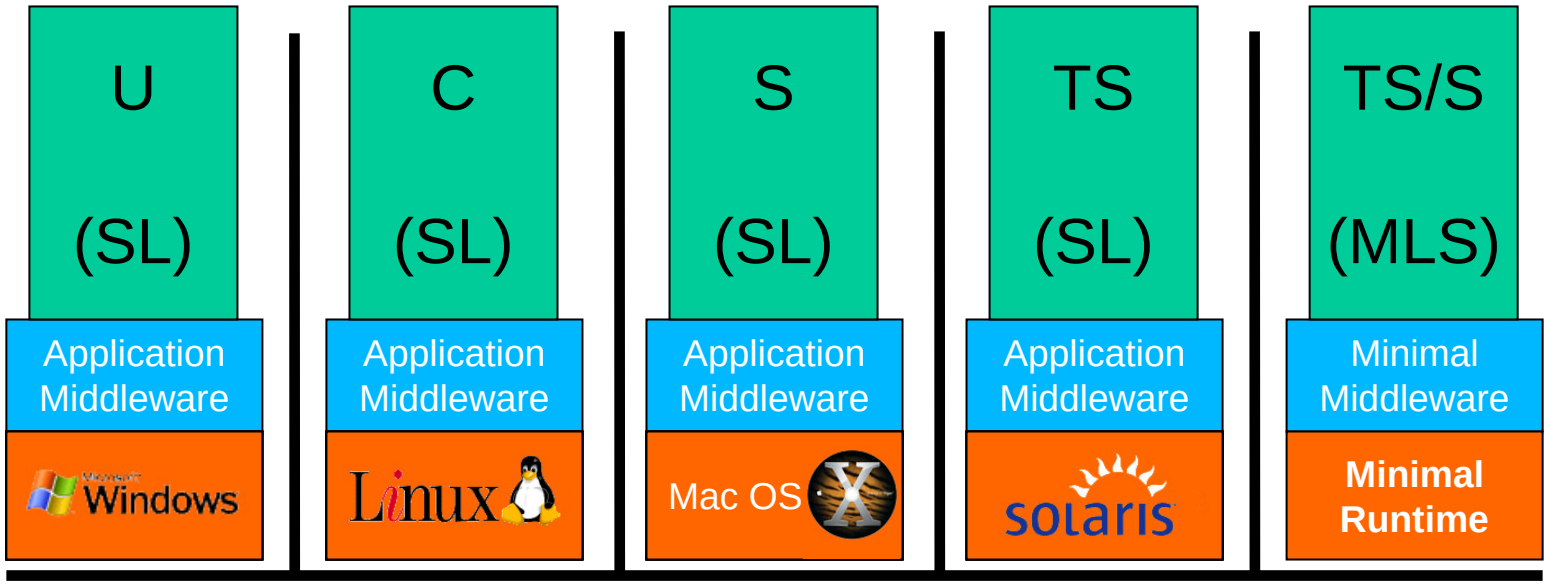
The MILS Architecture

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture



Guest OS Architecture

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture



A MILS Workstation? (later…)

Processor

Distributed Security Requirements

- Extend single node enforcement to multiple nodes
- Do not add new threats to data Confidentiality or Integrity
- Enable distributed Reference Monitors to be **NEAT**
- Optimal inter-node communication
  - Minimizing added latency (first byte)
  - Minimizing bandwidth reduction (per byte)
- Fault tolerance
  - Infrastructure must have no single point of failure
  - Infrastructure must support fault tolerant applications

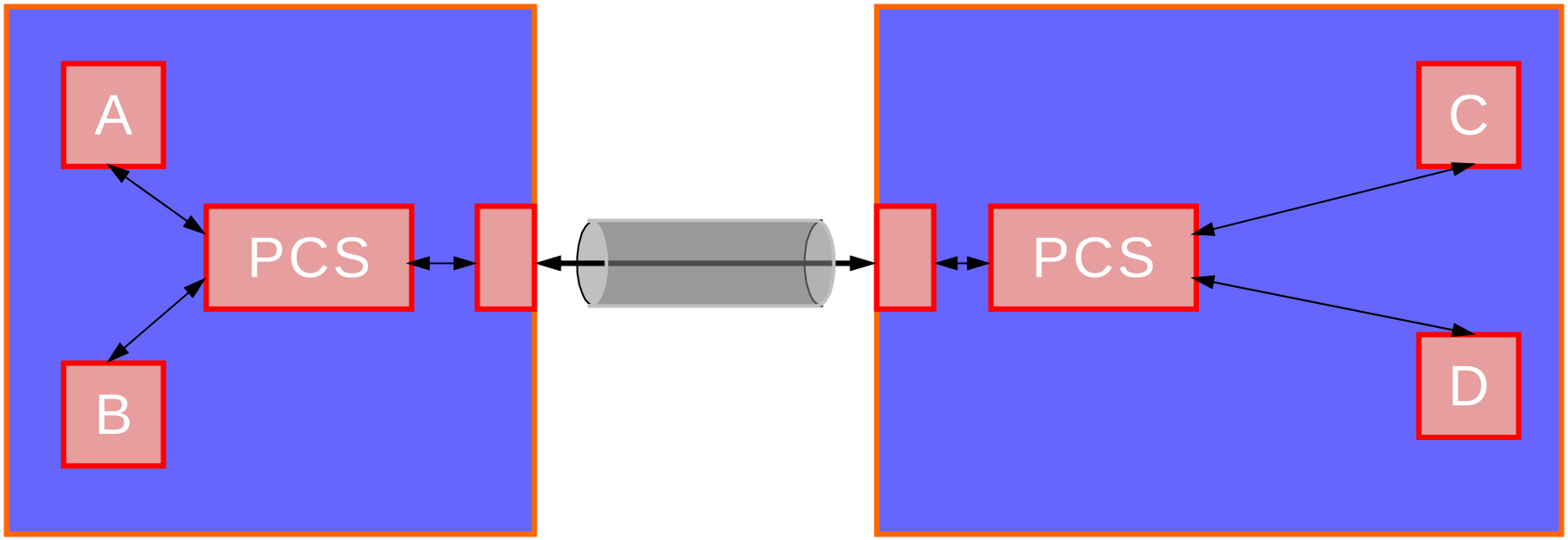
The Partitioning Communications System

- The Partitioning Communications System (PCS) is communications middleware for MILS
- Always interposed in inter-node communications
- Interposed in some intra-node communications also
- Parallels Separation Kernel's policies

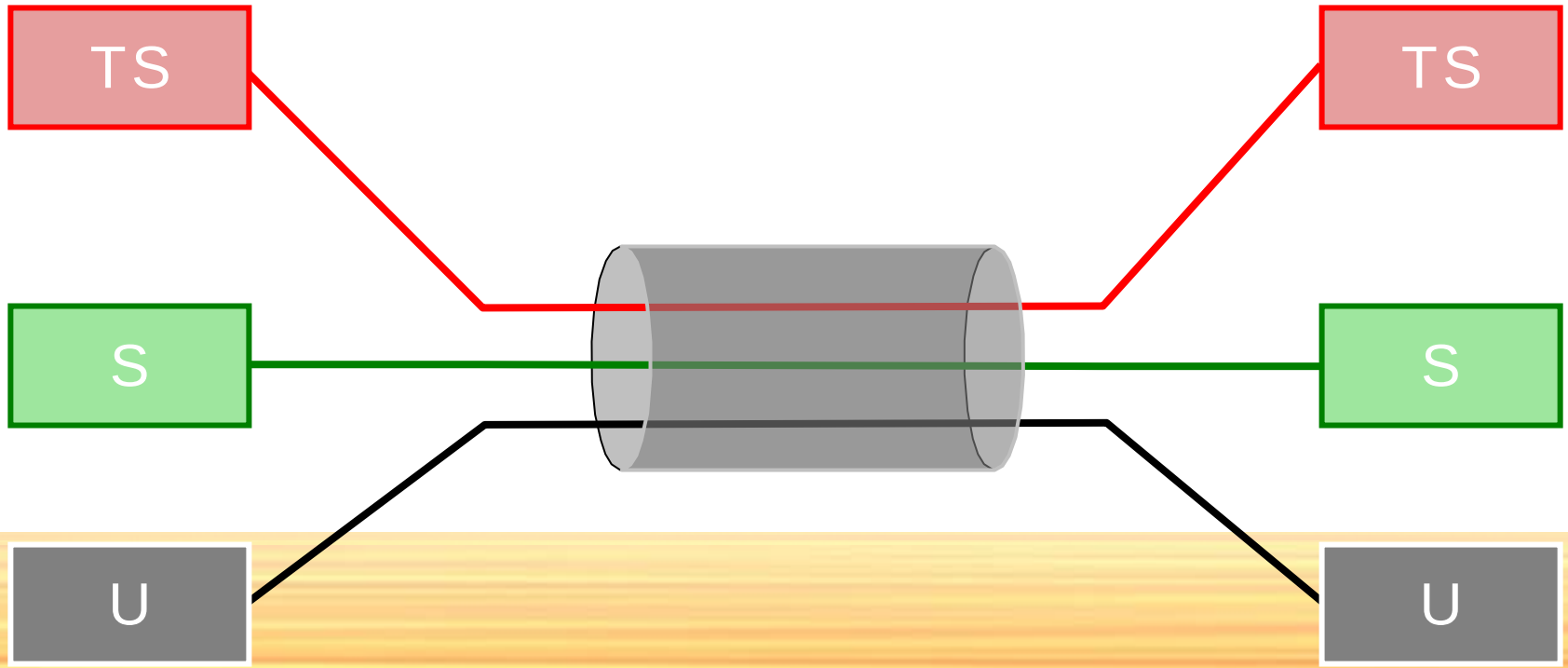
PCS Specific Requirements

- Strong Identity
  - Nodes within enclave
- Separation of Levels/Communities of Interest
  - Need cryptographic separation
- Secure Configuration of all Nodes in Enclave
  - Federated information
  - Distributed (compared) vs. Centralized (signed)
- Secure Loading: signed partition images
- Secure Clock Synchronization
- Suppression of Covert Channels
  - Bandwidth provisioning & partitioning
  - Network resources: bandwidth, hardware resources, buffers

Inter-node Communication

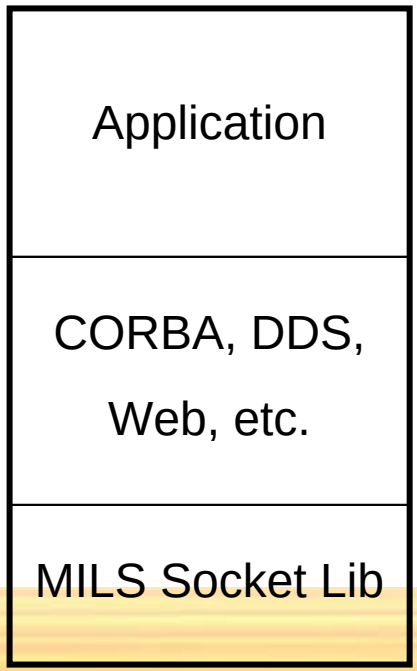


Partitioning the Channel

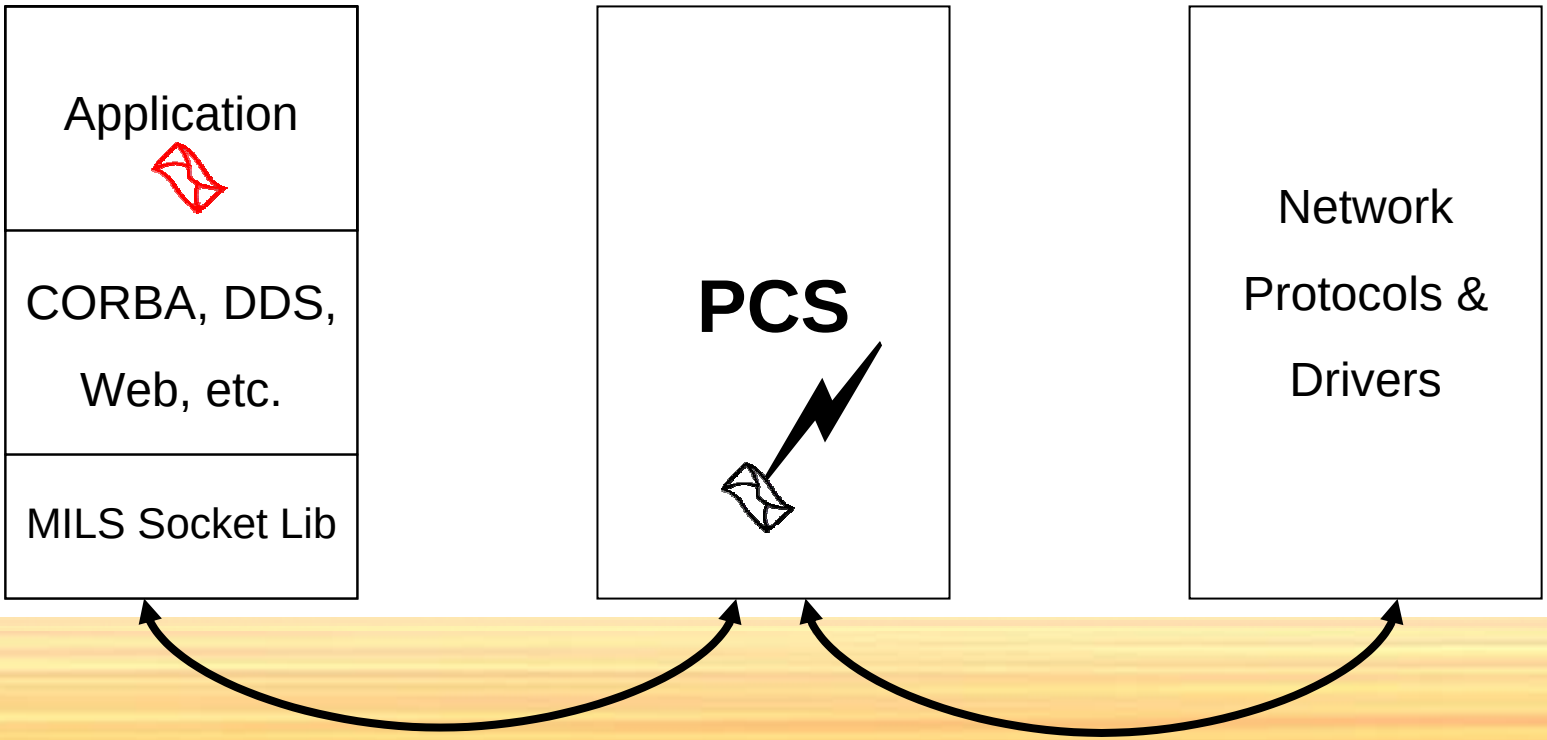


Network Middleware Libraries

- Network middleware provides libraries for application use
  - e.g.,
    - Real-time CORBA
    - Data Distribution Service
    - DBMS libraries
    - Web-based libraries (.NET, Web Objects, etc.)
  - Run in application partitions
  - Provide application with higher level interface to network libraries (eg. Socket libraries)
- Some applications use socket libraries directly

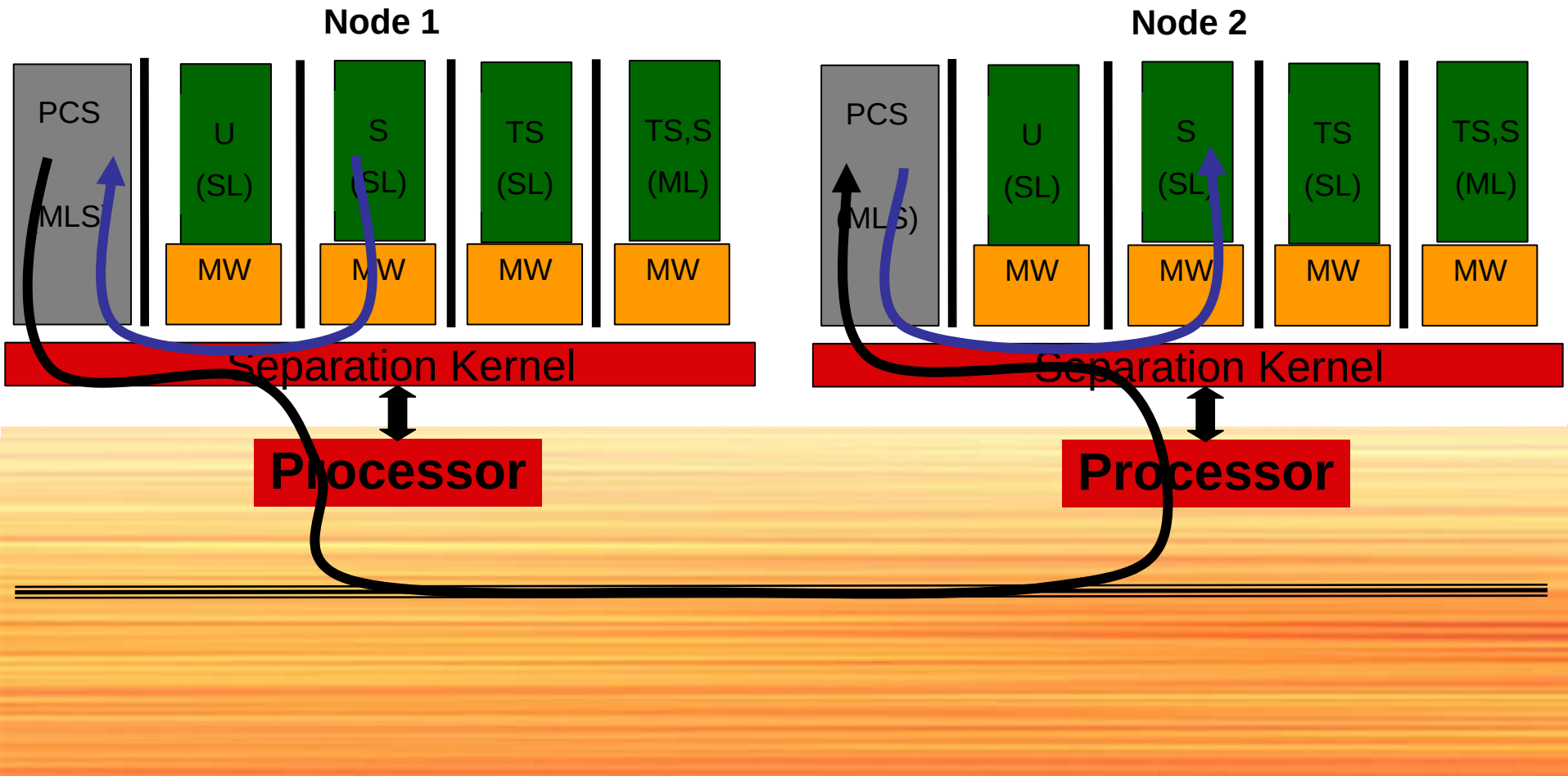


System Architecture with PCS



PCS Cross-Node Information Flow

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture



Real-time MILS CORBA

- Real-time CORBA can take advantage of PCS capabilities
  - Real-time CORBA + PCS = Real-time MILS CORBA
  - Additional application-level security policies are enforceable because of MILS SK and PCS foundation
- Real-time MILS CORBA represents a single enabling application infrastructure

RT CORBA & MILS Synergy

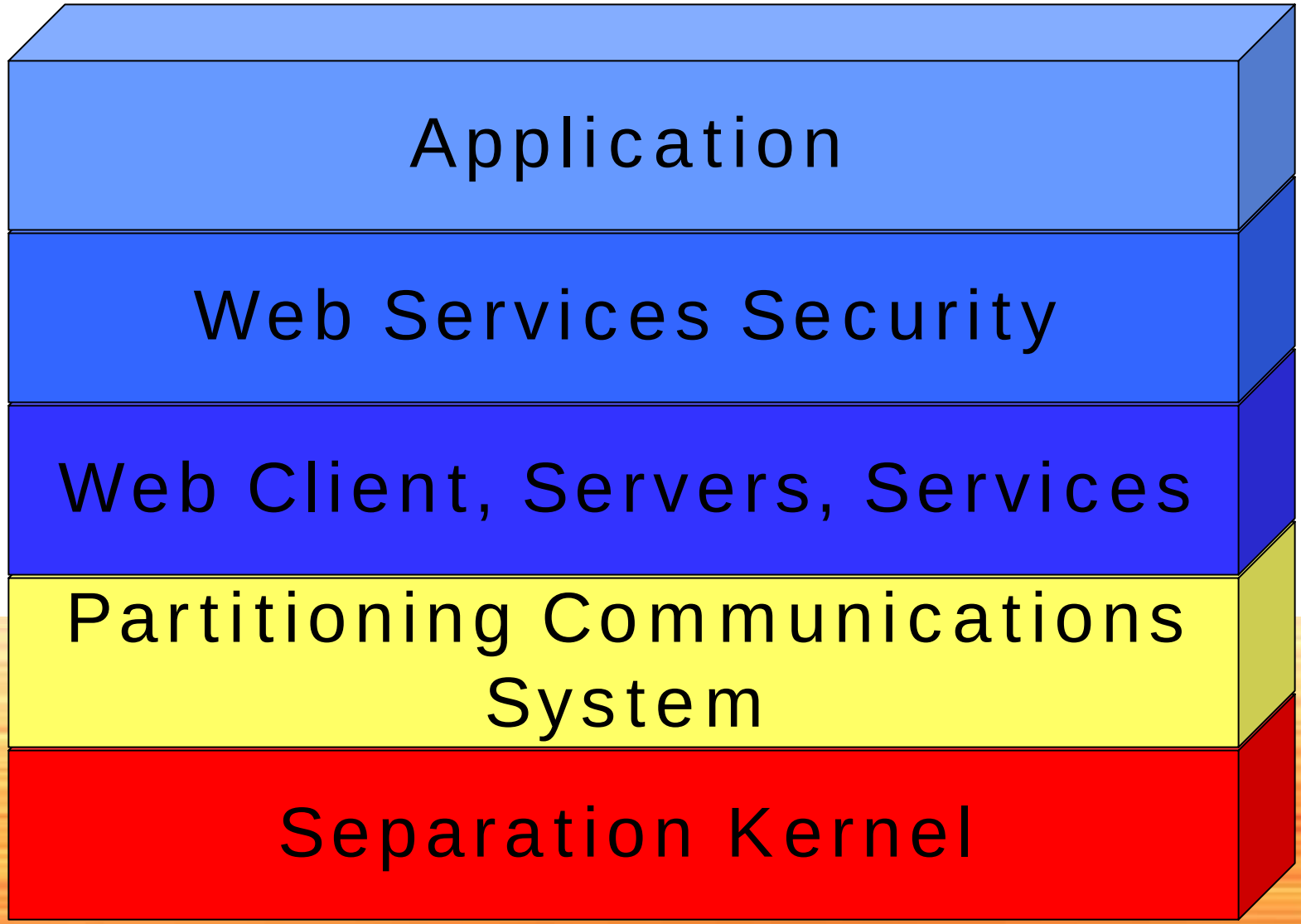
- Synthesis yields an unexpected benefit
  - Flexibility of Real-time CORBA allows realization of MILS protection
  - **MILS is all about location awareness**
    - Well designed MILS system separates functions into separate partitions
    - Takes advantage of the MILS partitioning protection
  - **Real-time CORBA is all about location transparency**
    - The application code of a properly designed distributed system built with Real-time CORBA will not be aware of the location of the different parts of the system.
    - CORBA flexibility allows performance optimizations by rearranging what partitions each system object executes in.
    - System layout can be corrected late in the development cycle
  - Combination of MILS and Real-time CORBA allows system designer
    - ***Rearrange system functions to take advantage of protection without introducing new threats to data confidentiality and integrity***

MILS DDS

- OMG Data Distribution Specification
  - Data-centric publish-subscribe
- PCS protects DDS implementations from
  - Attack by other partitions
  - Network attacks
  - Covert channels
- DDS can take advantage of PCS capabilities
  - PCS + DDS = MILS DDS
  - Application-level security policies are enforceable because of MILS SK and PCS foundation

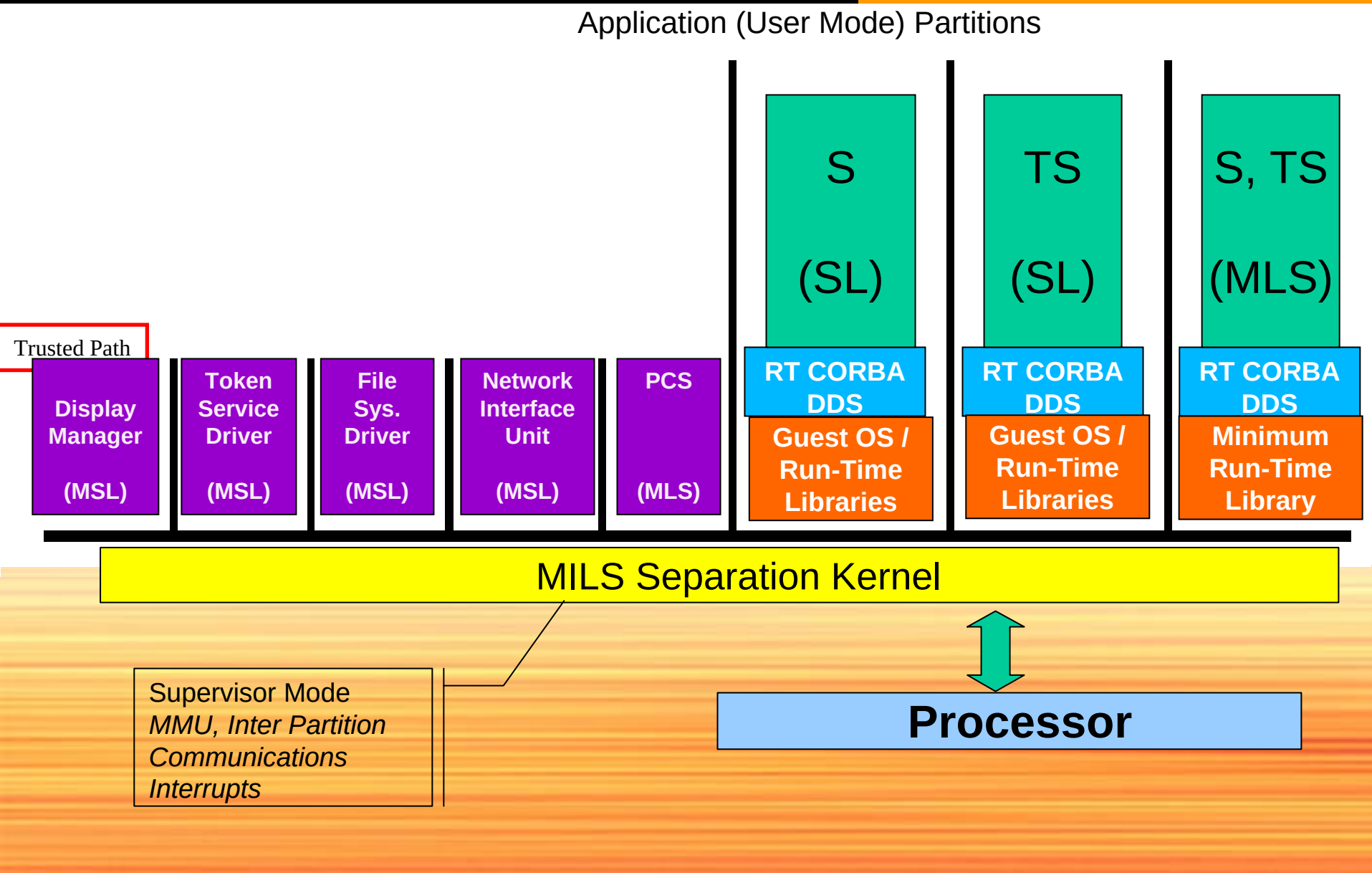
Web Services Overview

- The Web is all about the user interface
- Web Services are all about providing dynamic services driven from and to feed the user interface
- Programmable application logic accessible using standard Internet protocols


Application
Web Services Security
Web Client, Servers, Services
Partitioning Communications System
Separation Kernel

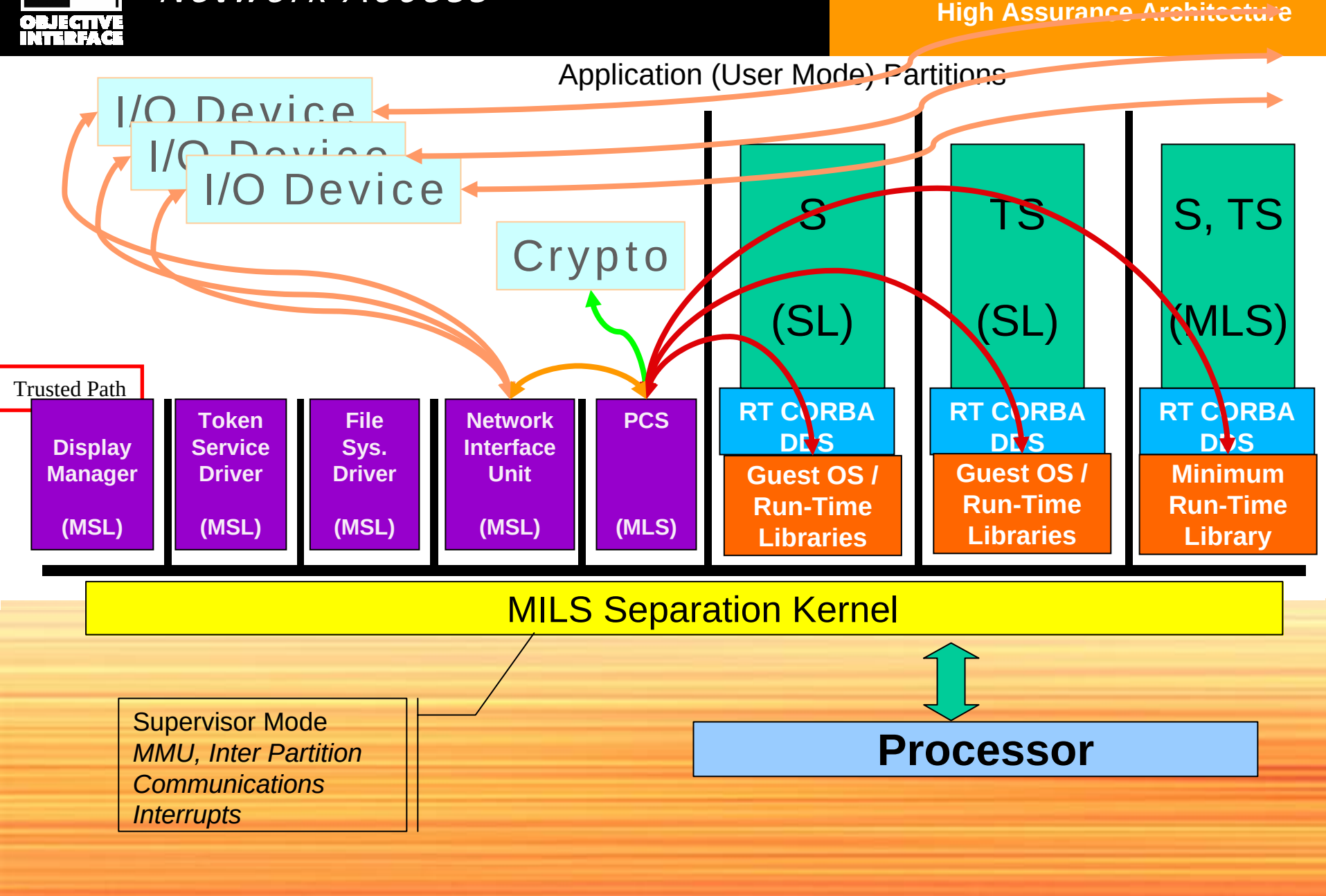
High Assurance MILS Workstation

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture



MILS Workstation Network Access

Multiple Independent Levels Of Safety And Security (MILS): High Assurance Architecture



The Whole Point of MILS

Really very simple:

- Dramatically reduce the amount of *safety/security critical code*

So that we can

- Dramatically increase the scrutiny of *safety/security critical code*

To make

- Development, certification, and accreditation more practical, achievable, and affordable.