LYNX MOSA.ic™

Simplifying Software Systems Through Modular Design

Lynx MOSA.ic™ is a software development framework for rapidly building comprehensible software systems out of independent application modules, delivering the vision of the Modular Open Systems Approach (MOSA).

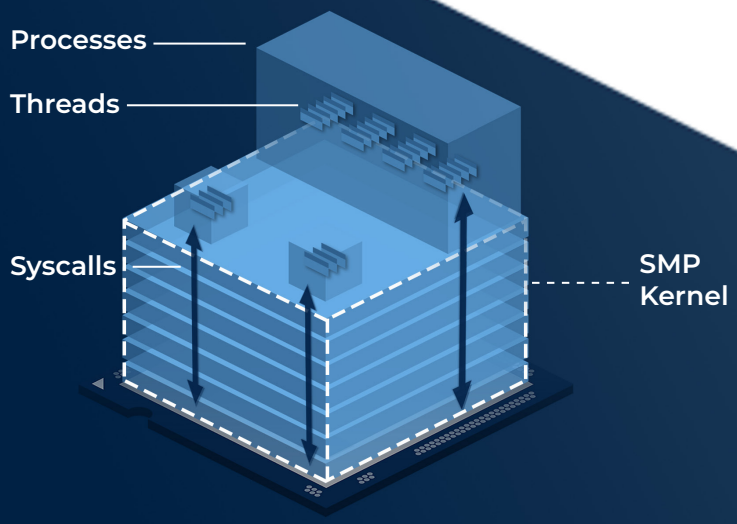
Giving developers deeper insight and increased control over how applications are realized onto modern CPUs, Lynx MOSA.ic™ introduces a new perspective to application development that simplifies the creation, certification, and maintenance of inherently complex software systems.

Simpler Software Systems—
Harnessing CPU virtualization, Lynx MOSA.ic™ is founded on a simpler distributed resource control model compared to a traditional OS-based central resource management model, removing as much complexity as possible between application interfaces and hardware.

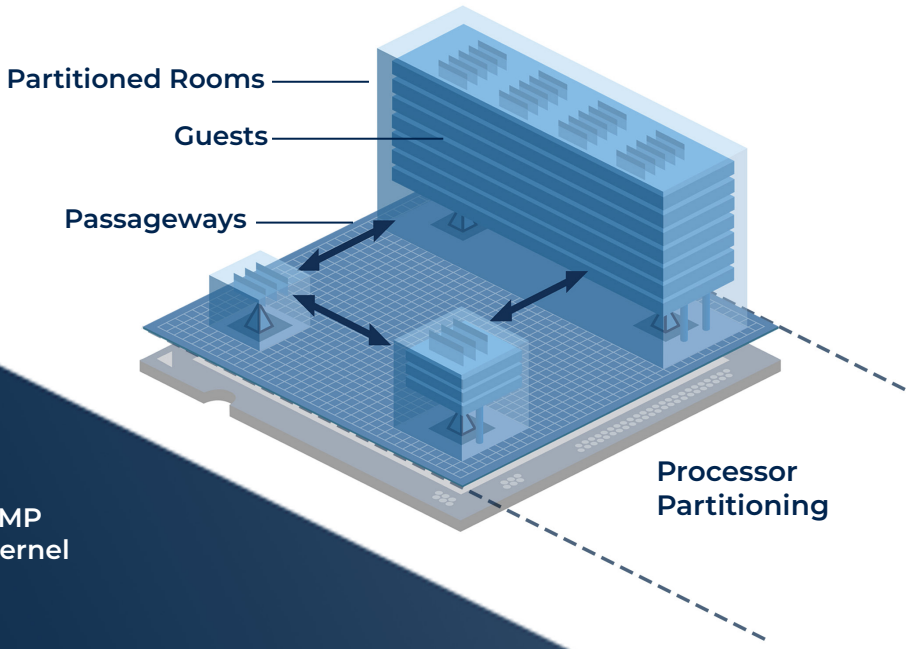
—Through Modular Design
Lynx MOSA.ic™ isolates computing resources into independent distributed environments which are uniquely capable of managing themselves while providing software development tools for building guests at just the right complexity levels demanded by specific target environments.

Clearer Path to Multi-core Certification
Forgoing the traditionally inherited complexities of a centralized resource management model common to OS and hypervisor designs, Lynx MOSA.ic™ adapts to the advances in multi-core virtualization to simplify vital platform abstraction layers. The comprehensible, hardware enforced architecture of Lynx MOSA.ic™ makes inherently complex multi-core system development a viable option when facing the risks of building solutions in highly regulated safety- and security-conscious markets.

Lynx MOSA.ic™ Architecture



Traditional Platform Architecture

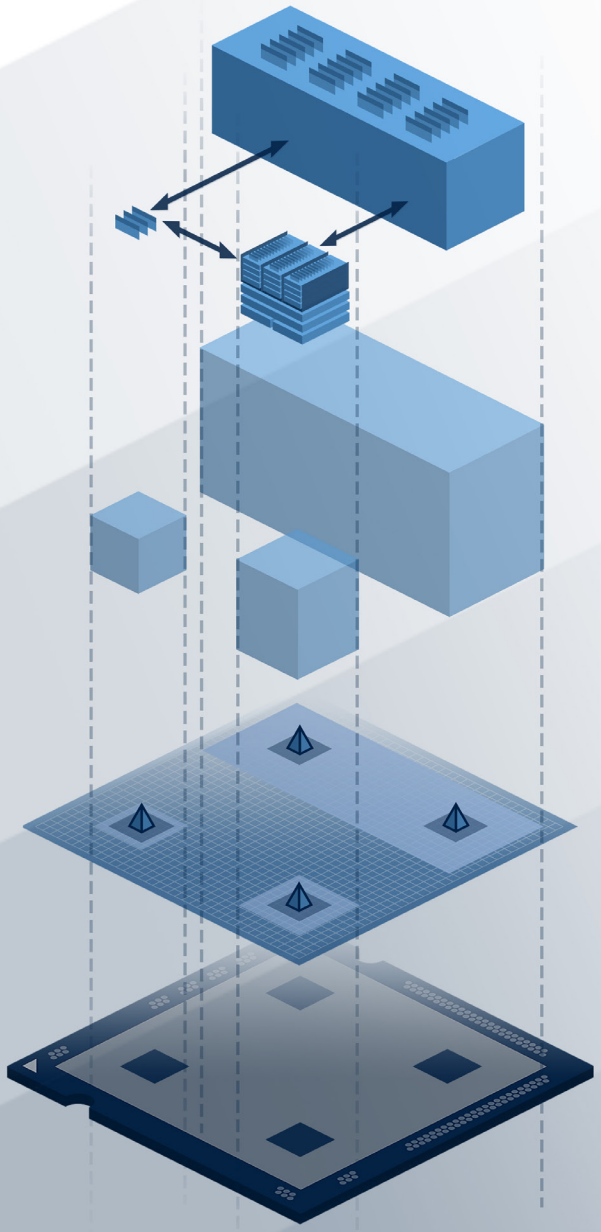


LYNX SOFTWARE TECHNOLOGIES

LYNX MOSA.ic™

Rapidly build robust, comprehensible systems using the Lynx MOSA.ic™ development framework. Construct rooms of partitioned hardware, connect rooms to each other or to external devices, then place legacy, vendor, or competitor guests in rooms.

The Framework is comprised of three distinct classes of tools—**Architecture Design, Module Development, & System Module Integration.**



Module Development & System Module Integration

A cross development kit is included for building guests of varying size, quality, and complexity specific to their target environments. Integration tools connect legacy, competitor, or partner-provided guests together, and can move guests from room to room in current and future designs.

Lynx CDK Guest Support

LSA	• Lynx Simple Application (Bare Metal Application) • LSA.store – Bare-Metal Crypto Module XTS-AES 256 • Z-Scheduling – Real-time scheduling across rooms • Guest IPC – Point-to-Point FIFO • Debug – Lauterbach TRACE 32 Integration
LynxOS -178	• UNIX-like Real Time Operating System • Certs – DO 178 DAL A, FAA Reusable Software Component • APIs – POSIX, FACE, ARINC 653 • Scheduling – Priority Pre-emptive, Cyclic • Debug – Lauterbach TRACE 32, Eclipse IDE Profiling & GDB • Guest IPC – Point-to-Point FIFO, Ethernet UART
Buildroot	• Embedded Linux Toolchain • APIs – POSIX, FACE • Guest IPC – Point-to-Point FIFO, Ethernet, UART • Device Sharing (Intel) – SRIOV, GFX, USB, Storage, Ethernet • Debug – Eclipse IDE GDB

Architecture Design

Control the behavior of the system with an architecture configuration policy. Enforce the policy with a least privilege distributed control plane that partitions hardware to create rooms and passageways for guests.

Rooms—Collections of resources created by the processor partitioning system and defined by the architecture configuration policy.

Room Types	• Bare-metal – Raw 64-bit guest contexts • RTOS – Lightweight context support for real-time scheduling and certified code bases • Legacy OS – Complex Rooms with hardware emulation support for legacy code bases

Passageways—Explicit point-to-point memory regions link rooms together via standard IPC interfaces, maximizing performance and preserving minimal complexity.

Types	• FIFO • Ethernet • Device Emulation

Processor Partitioning System

Partition processor resources with an architecture configuration policy to control the behavior of the system. Enforce the policy with a least privilege distributed control plane that creates rooms and passageways for guests.

Processors	• Arm v8-A • Intel VTx • PPC QorIQ (Coming Soon—2019)

LYNX SOFTWARE TECHNOLOGIES