T-Engine: An Open Platform for Real-Time Embedded System Design

Shinsuke Kobayashi
University of Tokyo
YRP Ubiquitous Networking Lab.

Embedded System

T-Engine

- Increasing complexity of application
  - Growing software size is a problem.
- HIGH design cost while a SHORT life cycle
  - Improvement of design cost is required.

Concept of T-Engine

Common design platform

- The common platform does not exist in embedded system design.

Platform based design

- The cost of design can be improved by using the common platform.
- Distribution of software on the platform to reduce the cost of software development.

Technology Goal

- Improvement of the following factors in embedded system design by the standardization of architecture and OS
  - Reusability
  - Productivity
  - Maintainability

  ➡ “Distribution of middleware”
- Realization of interoperability by using HW/SW design platform

T-Engine

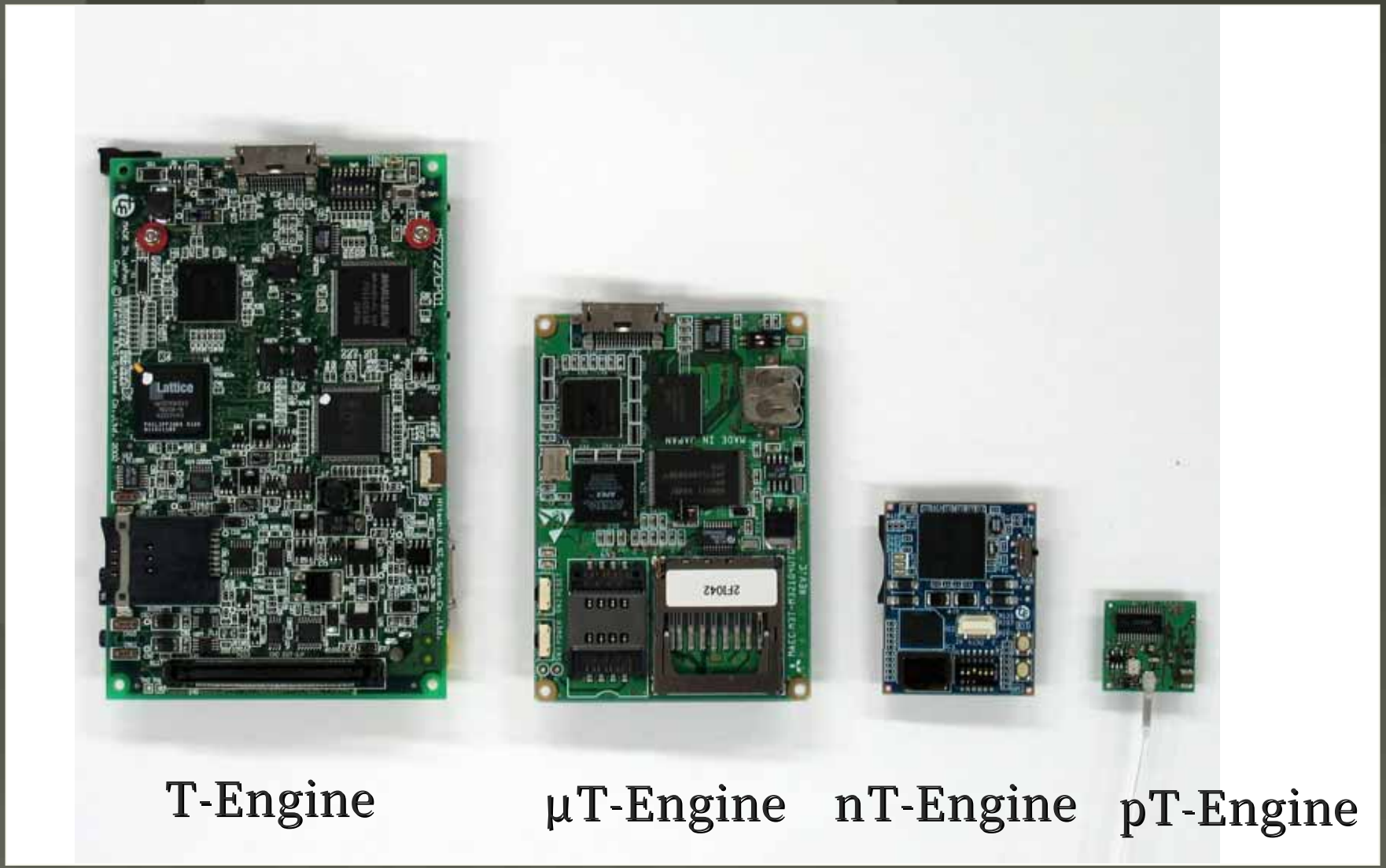
T-Engine

- Open standard
  - Open license for embedded systems
- Commercial-Off-The-Shelf (COTS)
  - Cost effective
- Chip-free architecture
  - Software is isolated from hardware implementation by the layer structure architecture.

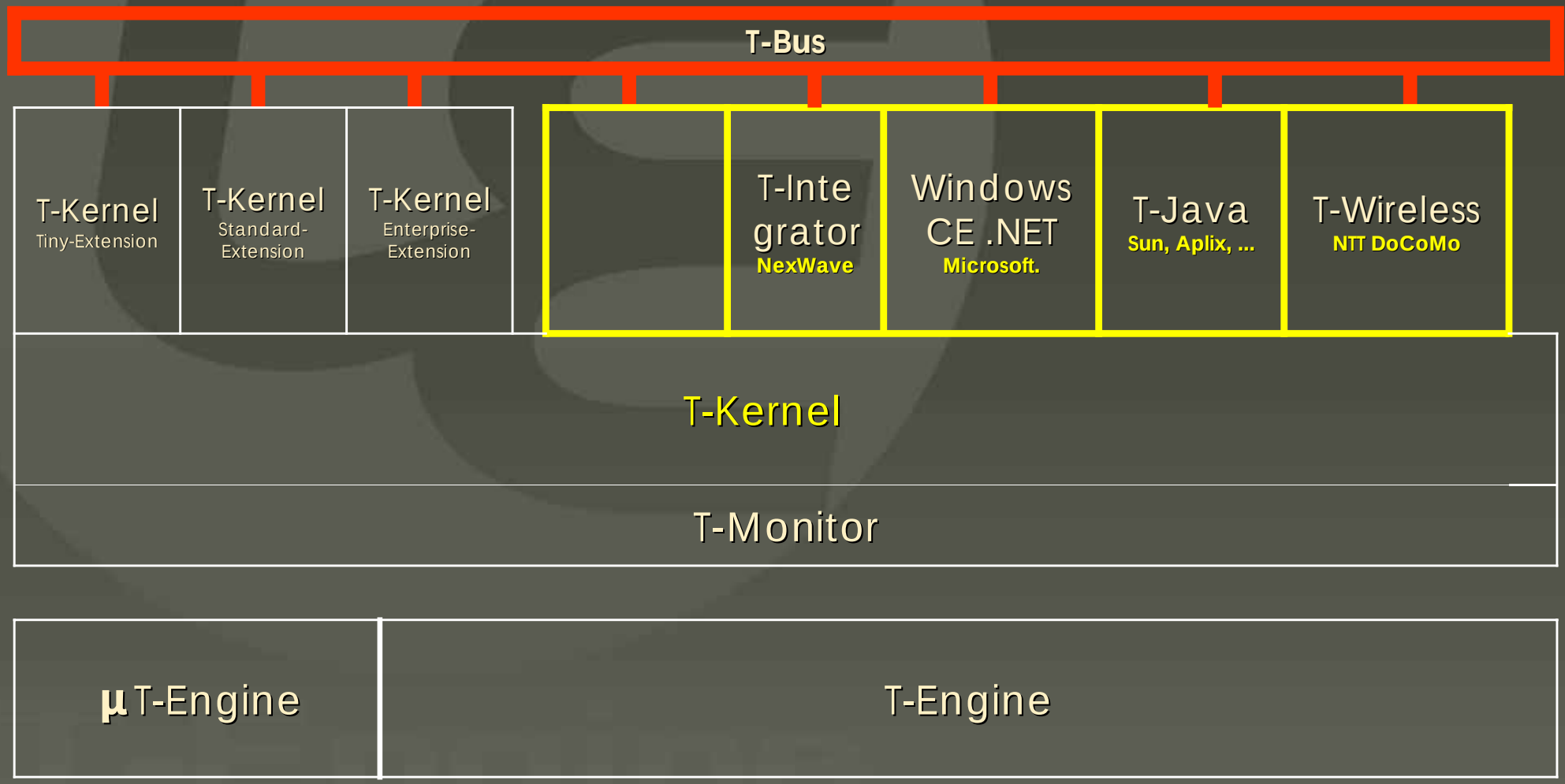
T-Engine Series

- Development platform for embedded system
  - Standard T-Engine
  - µT-Engine (micro)
- Execution platform for ubiquitous computing
  - nT-Engine (nano)
  - pT-Engine (pico)

T-Engine Lineup



T-Engine Software Architecture



T-Engine Boards (1)

Model	h101	d301	n101	n301M	y101	f301
Vendor	Renesas Technology Corp. &Hitachi ULSI Systems Co. Ltd.	Renesas Technology Corp.	NEC Corporation	NEC Corporation	Yokogawa Digital Computer Corporation	FUJITSU LIMITED & Yokogawa Digital Computer Corporation
Specification	Standard T-Engine	µT-Engine	Standard T-Engine	µT-Engine	Standard T-Engine	µT-Engine
CPU	SH7727 (SH3-DSP)	M32104 (M32R)	VR5500-400 (MIPS)	VR4131-200 (MIPS)	ARM720T (ARM)	MB93403 (FR-V)
Clock	96MHz	216MHz	400MHz	200MHz	72MHz	266MHz
RAM	32MB	16MB	64MB	32MB	32MB	64MB
Flash	8MB	4MB	16MB	16MB	8MB	16MB
Release Time	2002/07	2002/09	2002/11	2002/11	2002/12	2003/06

T-Engine Boards (2)

Model	y102	h102	h301	y103	t101
Vendor	Yokogawa Digital Computer Corporation	Renesas Technology Corp. &Hitachi ULSI Systems Co. Ltd.	Renesas Technology Corp. &Hitachi ULSI Systems Co. Ltd.	Yokogawa Digital Computer Corporation	TOSHIBA CORPORATION
Specification	Standard T-Engine	Standard T-Engine	µT-Engine	Standard T-Engine	Standard T-Engine
CPU	MC9328MX1 (ARM9 core) (Motorolla)	SH7751R (SH4)	SH7145 (SH2)	ML7101 (ARM9 core) (OKI)	TX4956 (MIPS4)
Clock	200MHz	240MHz	50MHz	TBD	400MHz
RAM	64MB (SDRAM)	64MB (SDRAM)	1MB (SRAM)	TBD	128MB (SDRAM)
Flash	16MB	8MB	1MB	TBD	16MB
Release Time	2004/04E	2003/05E	TBD	TBD	2004/5E

T-Kernel

T-Kernel

- Real-time OS for embedded systems
- Open standard
  - The infrastructure for embedded system design
- Not only specification but also source code
  - Distribution of software by T-Engine Forum

Objects of T-Kernel

- Task
- Synchronization, Communication
  - Semaphore, Event flag, and Mailbox
- Extended synchronization, Communication
  - Mutex, Message buffer, and Rendezvous port
- Memory pool manager
  - Fixed/Variable size memory pool
- Time manager
  - Cyclic handler, and Alarm handler

Real-time

Real-time

- The time constraints to execute tasks are kept.

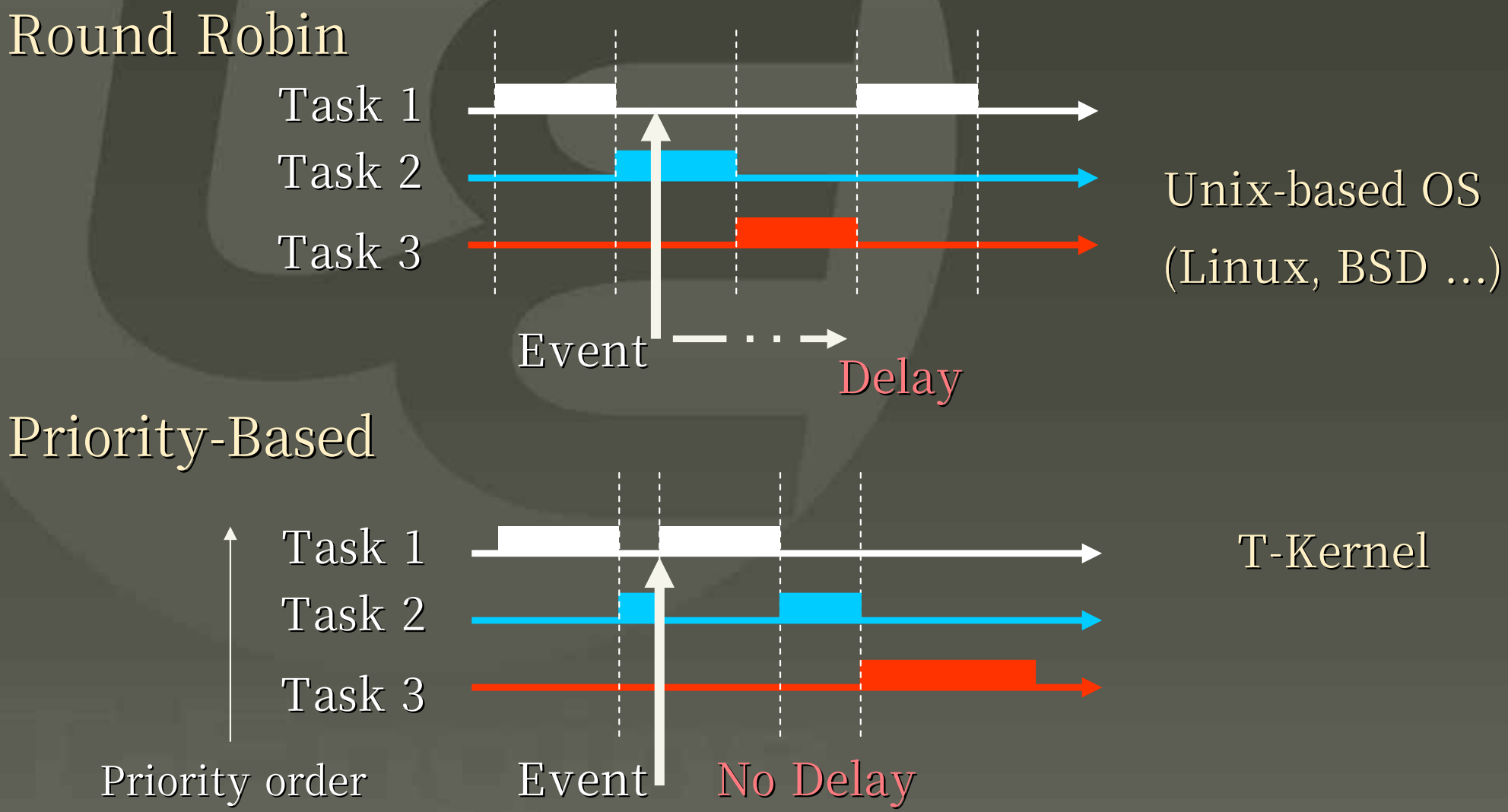
Real-time application

- Real-time execution required.
- Ex. Engine control, Attitude control …

Real-time OS

- The function to keep real-time execution
- High speed context switching and task switching
- Priority-based scheduling

Priority-Based Scheduling vs. Round Robin Scheduling



Why Can T-Kernel Be Used In Critical Applications?

- High performance of real-time execution
  - Several micro-seconds for context switching
- High reliability of source code
- Safe intellectual property
  - The other's (dangerous) codes are not included.
- Open multi-vendor
  - Independent from a particular vendor
- Community
  - Many engineers, and education programs

Design Reuse Framework

Design Method Using T-Engine

- Rapid prototyping
  - T-Engine can be used as a prototype board.
  - A target system can be evaluated by designer in early phase of design.
- Design reuse
  - A lot of middleware is provided by T-Engine platform.
  - The cost of design and test is reduced by the design reuse.

Middleware Distribution

Middleware distribution framework

- Device drivers can be easily designed due to that hardware, and especially interfaces are standardized.
- Porting is easy because each T-Engine has the same interfaces.

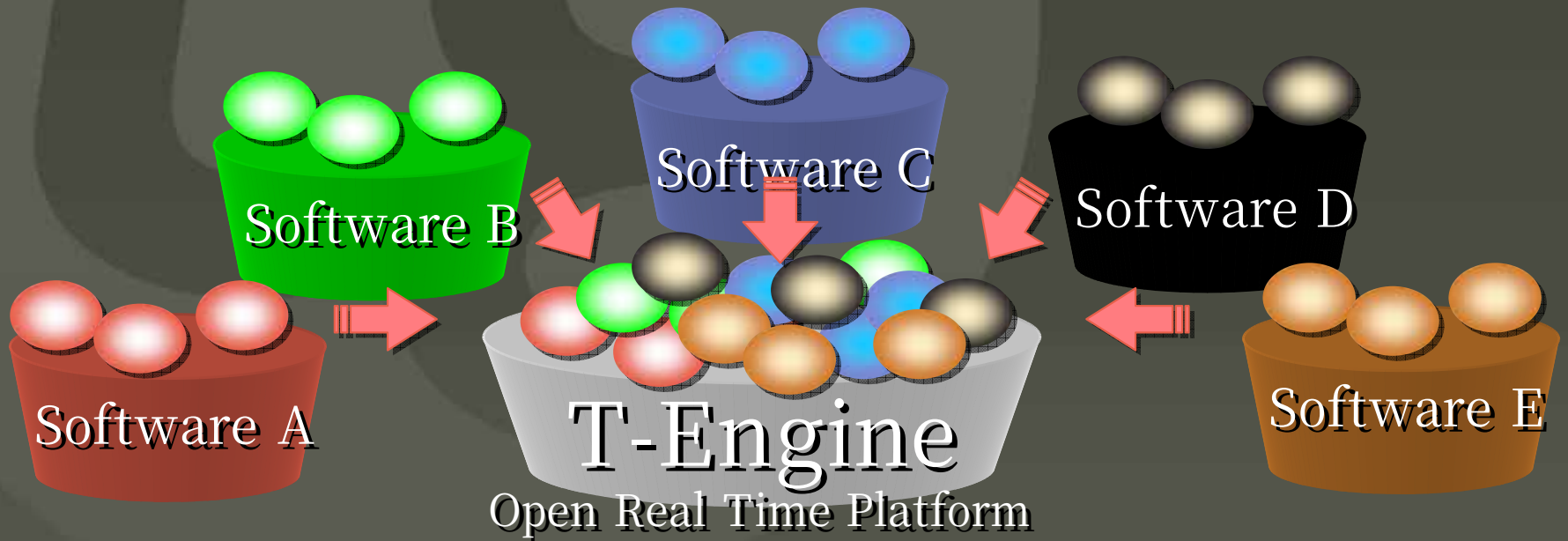
Hitachi h101
(SH)

NEC n101
(MIPS)

YDC y101
(ARM)

Software Design Reuse

- Common platform for embedded software reuse



- The productivity of design can be improved by reusing software design

Middleware Examples : Ported Extensions

- T-Wireless (NTT DoCoMo, Inc.)
  - 3G Mobile Communication Middleware that works on T-Kernel
- T-Java (Sun Microsystems Inc, Aplix Corporation)
  - Java Execution Environment for T-Kernel
- T-Integrator (NexWave Solution)
  - Middleware for Consumer Electronics by NexWave
- Windows CE .NET (Microsoft Corporation)
  - Windows CE .NET for T-Kernel

Conclusion

T-Engine

- Open standard platform for embedded system design

T-Kernel

- Well-defined real-time OS for embedded system
- High performance of real-time execution

T-Engine Forum

T-Engine Forum

http://www.t-engine.org