Think your future OS strategy includes VxWorks? Think again.

Leo Forget, Product Manager
Pat Shelly, Field Application Engineer

QNX SOFTWARE SYSTEMS

Changes to the RTOS vendor market

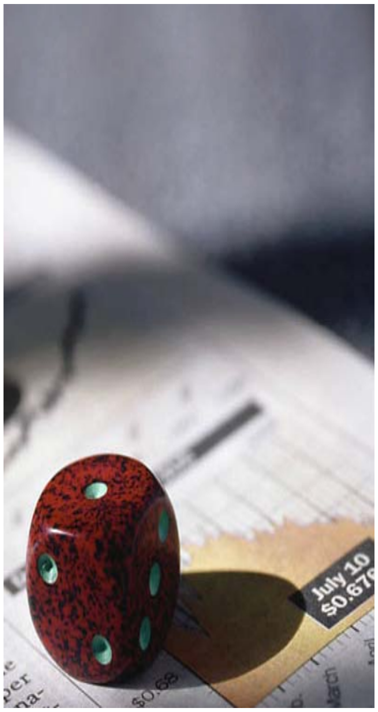
QNX SOFTWARE SYSTEMS

Intel to Acquire Wind River Systems for Approximately $884 Million

SANTA CLARA, Calif., June 4, 2009 – Intel Corporation has entered into a definitive agreement to acquire Wind River Systems Inc, under which Intel will acquire all outstanding Wind River common stock for $11.50 per share in cash, or approximately $884 million in the aggregate. Wind River is a leading software vendor in embedded devices, and will become part of Intel's strategy to grow its processor and software presence outside the traditional PC and server market segments into embedded systems and mobile handheld devices. Wind River will become a wholly owned subsidiary of Intel and continue with its current business model of supplying leading-edge products and services to its customers worldwide.

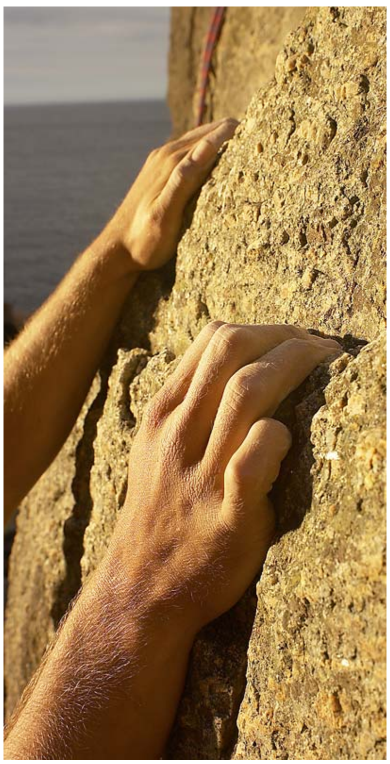
"This acquisition will bring us complementary, market-leading software assets and an incredibly talented group of people to help us continue to grow our embedded systems and mobile device capabilities," said Renee James, Intel vice president and general manager of the company's Software and Services Group. "Wind River has thousands of customers in a wide range of markets, and now both companies will be better positioned to meet growth opportunities in these areas."

Business risks to Wind River customers



- Will Intel continue investments into Wind River to support the latest technology trends?
- Will Wind River continue to support non-Intel silicon?
- Will other silicon vendors cooperate (roadmaps, technical assistance, funding) with their biggest competitor?
- Will the Wind River software ecosystem shrink as software suppliers desert a perceived Intel-only OS?
- Will VxWorks business or support models change as a result of the acquisition?
- Will Intel decide to change the focus of Wind River (for example, from a general embedded company to phone-only focus)?
- Will VxWorks be discontinued by Wind River and/or Intel?

Technical risks to Wind River customers



- Will the solution continue to be sufficient in the face of relentlessly advancing hardware and software complexity?
- Is the development team able to prevent an escalating workload with the current tools and OS technologies?
- Will the current OS solution continue to support newly released hardware?
- Is your current software solution easily scalable and modifiable as you expand your product's feature set?
- Can the vast resources of open source be tapped, but in a safe reliable fashion?

OS migration: QNX is a great fit if you need …

QNX has…

A reliable foundation for critical systems?	A 30 year track record
Hard real-time or time/space partitioning?	Scheduler and adaptive partitioning
A scalable architecture for the future?	Modular microkernel architecture
To build on standards?	POSIX, OpenGL ES, OpenKODE, Eclipse
Certifications?	ISO9001, PSE52, EAL4+, SIL3

OS migration: QNX is a great fit if you need …

QNX SOFTWARE SYSTEMS

	QNX has…
Transparency of source?	Fully published source code and forums
Clean intellectual property?	No viral GPL licensing
Support through lifetime of the product?	Guaranteed availability of past versions
Sophisticated HMI solutions?	Attractive HMI environment
Low RAM and flash requirements?	Solution customizable for smallest size

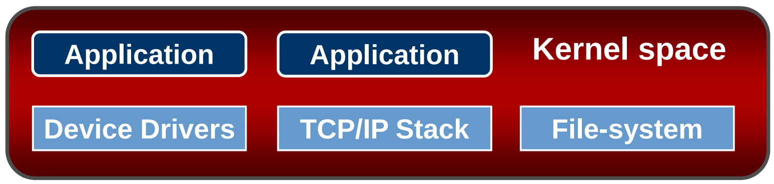
Technology comparison

Microkernel architecture benefits

QNX SOFTWARE SYSTEMS

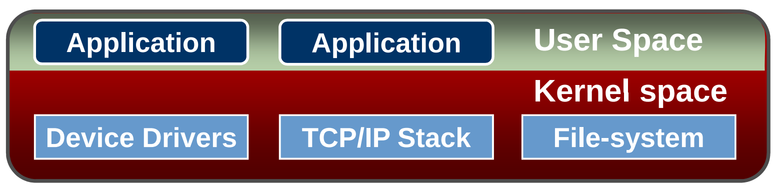
Realtime executive

- No MMU and no protection
- Applications, drivers, and protocols are all in kernel space



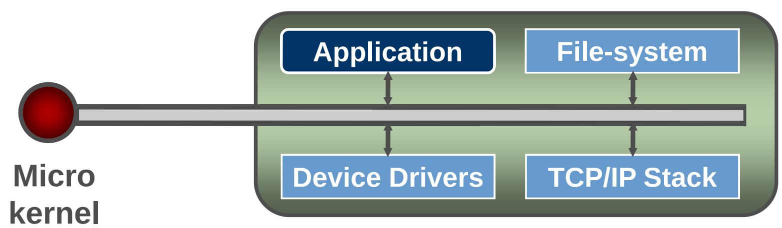
Monolithic kernel
(Microsoft / Unix / etc)

- MMU with partial protection
- Applications are protected



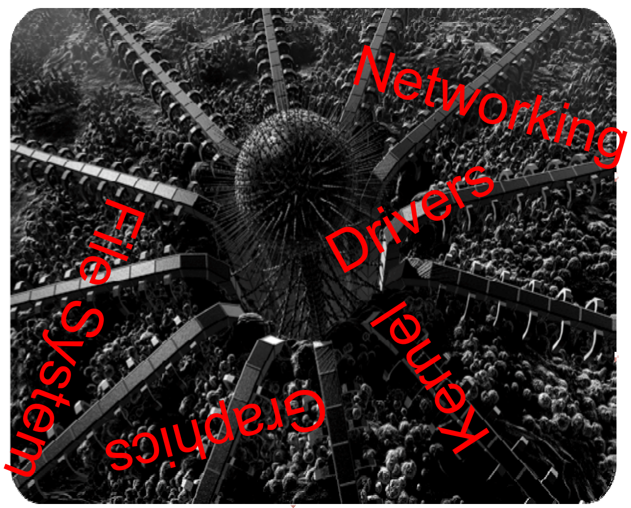
TRUE microkernel
(QNX Neutrino RTOS)

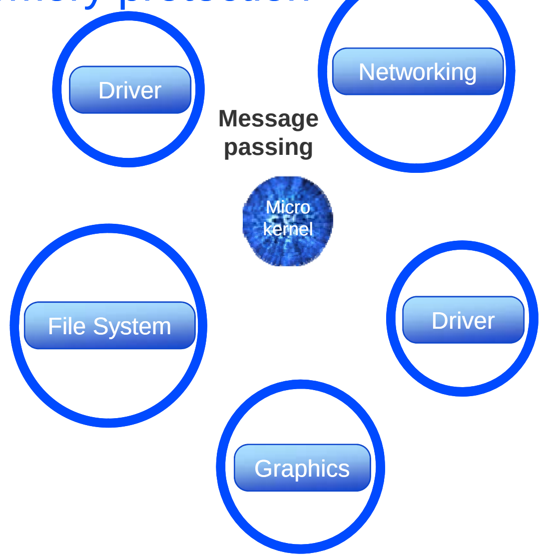
- MMU with full protection
- Applications, drivers, and protocols are protected



QNX reliability (when failure is not an option)

Memory protection





Monolithic kernel

WinCE 3.9 million lines of code
Linux: 5.76 million lines of code
XP: 40 million lines of code

Microkernel

QNX 0.1 million lines of code

Microkernel architecture benefits

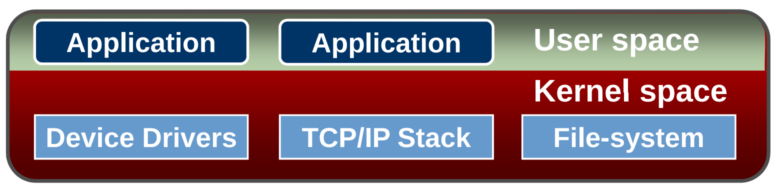
QNX SOFTWARE SYSTEMS

Realtime executive

- No MMU and no protection
- Applications, drivers, and protocols are all in kernel space

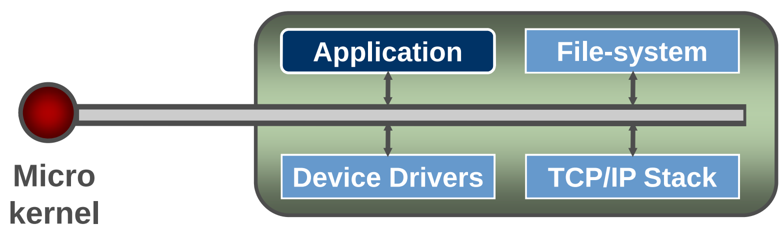
Monolithic kernel
(Microsoft / Unix / etc)

- MMU with partial protection
- Applications are protected



TRUE microkernel
(QNX Neutrino RTOS)

- MMU with full protection
- Applications, drivers, and protocols are protected



Microkernel architecture benefits

QNX SOFTWARE SYSTEMS

Realtime executive

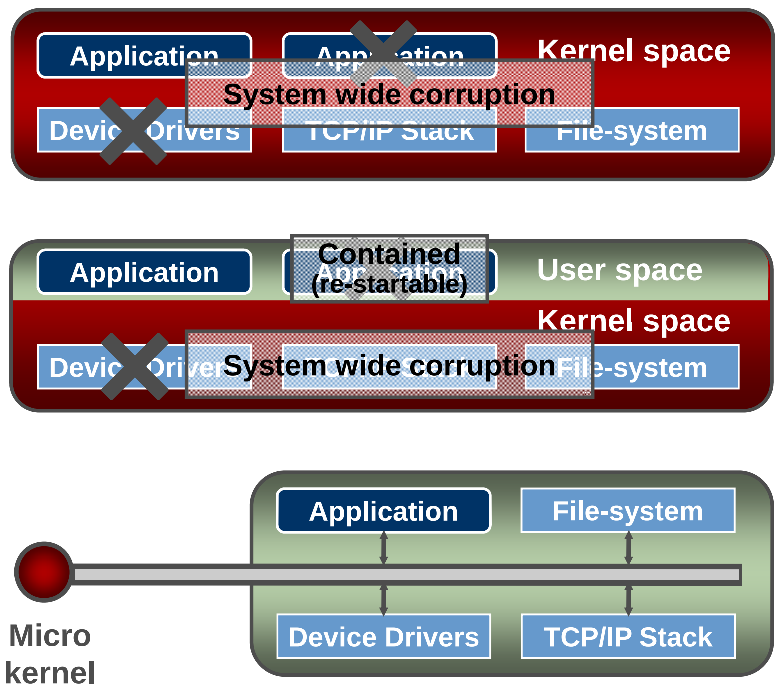
- No MMU and <u>no</u> protection
- Applications, drivers, and protocols are all in kernel space

Monolithic kernel
(Microsoft / Unix / etc)

- MMU with <u>partial</u> protection
- Applications are protected

TRUE microkernel
(QNX Neutrino RTOS)

- MMU with <u>full</u> protection
- Applications, drivers, and protocols are protected



Microkernel architecture benefits

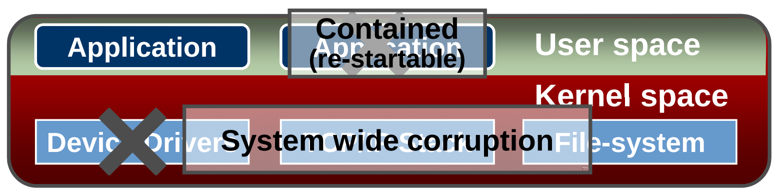
QNX SOFTWARE SYSTEMS

Realtime executive

- No MMU and <u>no</u> protection
- Applications, drivers, and protocols are all in kernel space

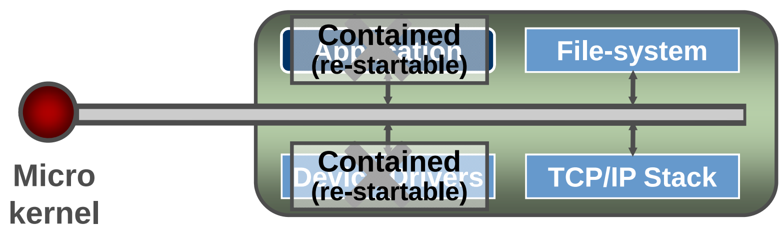
Monolithic kernel
(Microsoft / Unix / etc)

- MMU with <u>partial</u> protection
- Applications are protected



TRUE microkernel
(QNX Neutrino RTOS)

- MMU with <u>full</u> protection
- Applications, drivers, and protocols are protected



Weighing technical risks

QNX SOFTWARE SYSTEMS

	VxWorks	QNX Software Systems
Multicore	✓ Yes	✓ QNX Neutrino RTOS supports three types of multi-core operation; Symmetric, Bound and Asymmetric.
Scalable	☒ Distribution code is not transparent and requires application rewriting	✓ QNX Neutrino RTOS supports scaling up or down easily: TDP for distributed processing, and significant support for multicore silicon
High availability	✓ VxWorks support a 3rd party High Availability solution.	✓ QNX Neutrino RTOS provides features like Adaptive Partitioning and High Availability Monitor (HAM), designed for reliability and robustness
Real-time	✓ VxWorks is hard real-time	✓ QNX Neutrino RTOS is hard real-time
System architecture	☒ While sometimes dubbed a microkernel by Wind River, it is not.	✓ The QNX Neutrino microkernel and its accompanying Process Manager (procnto) contain only the most fundamental OS services (signals, scheduling, messaging, etc.). Virtually all other OS services and user-written code — file systems, device drivers, GUIs, protocol stacks, applications — run in memory-protected user space.
Standards	✓ VxWorks solution adheres to many industry standards including POSIX, Eclipse	✓ QNX pursues and participates in standards; standards promote freedom of choice: POSIX, OpenGL-ES, OpenVG, OpenKODE, Eclipse

Weighing commercial risks

	VxWorks	QNX Software Systems
Focus	? Wind River has a long history of discontinuing products and changing strategies. How will Intel acquisition affect company focus?	✓ QNX has a 29 year track record in the embedded software industry.
Reliability	? VxWorks can execute user-written code in kernel space. While this model enables the programmer to optimize performance, it makes the kernel vulnerable to poorly written code. The programmer can easily introduce potential kernel faults, creating the need for a significant amount of kernel testing prior to deployment.	✓ QNX Neutrino RTOS field-proven in many life-critical applications: nuclear power plants, military and space applications, medical equipment, etc.
Tools	✓ Wind River provides an Eclipse based development environment.	✓ QNX uses industry standard Eclipse based IDE, tightly integrated with many sophisticated QNX-specific features
Support	✓ Wind River has a solid support organization.	✓ QNX support focused on direct customer problem resolution and customer involvement
Licensing terms	☒ Licensing models are complex and expensive.	✓ Simpler, more flexible, lower barrier to entry.
Hardware	✓ VxWorks currently runs on a wide variety of hardware platforms.	✓ QNX Neutrino RTOS runs on a wide variety of hardware platforms

OS migration – what is involved?

OS migration phases

QNX SOFTWARE SYSTEMS

1. Assess

Thoroughly understand your product's design and software components and identify potential porting issues

2. Port

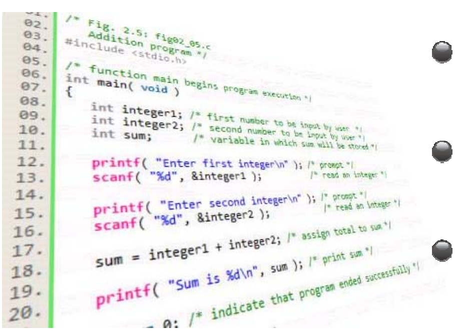
Make code changes as necessary to build and run your product on QNX Neutrino RTOS

3. Optimize

Take advantage of the QNX Neutrino RTOS's modularity and isolation to increase system reliability and facilitate trouble shooting

Assessment phase – understanding the system

QNX SOFTWARE SYSTEMS



1. Assess

- Interface analysis
- Task analysis
- Hardware access analysis
- Third party software analysis
- API analysis

Assessment phase – understanding the issues

QNX SOFTWARE SYSTEMS

1. Assess

- Memory accessibility
- Porting from VxWorks API to QNX Neutrino API
- Exception handling
- Modularity enforcement
- Device driver infrastructure

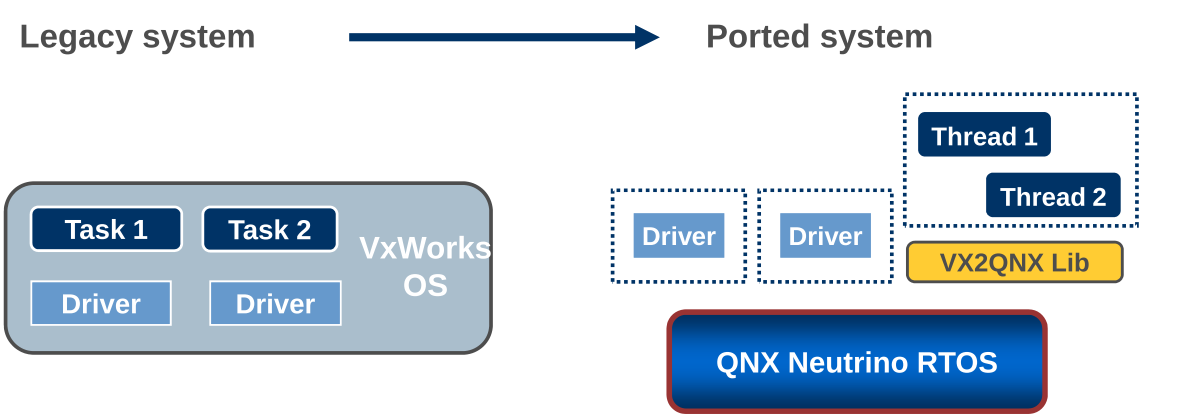
Porting phase – elements of porting

QNX SOFTWARE SYSTEMS

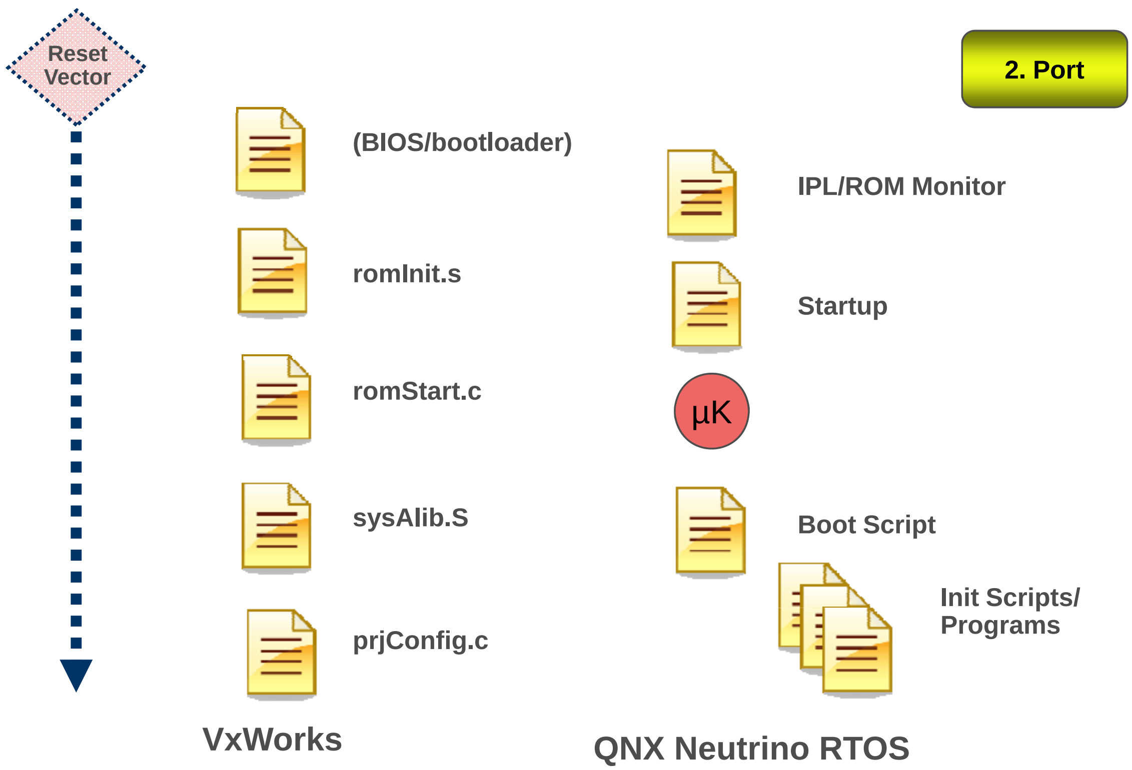
A typical porting project will involve effort in the following areas:

2. Port

- Board bring up / startup
- Hardware input/output (drivers)
- Networking
- Applications
- Build environment



Porting phase – board bring up/ startup



Porting phase – hardware input/output

2. Port

VxWorks:

- ISRs are dealt via the intArchLib.
- Device drivers tightly bound into operating system

QNX Software Systems:

- Hardware access in ISRs needs to be properly setup in terms of privity/permissions, memory addressing and memory accessing before an ISR can be used
- Device drivers can be started and stopped as standard processes
- Drivers can be developed and debugged just like any other application

Porting phase – hardware input/output

2. Port

Conclusions:

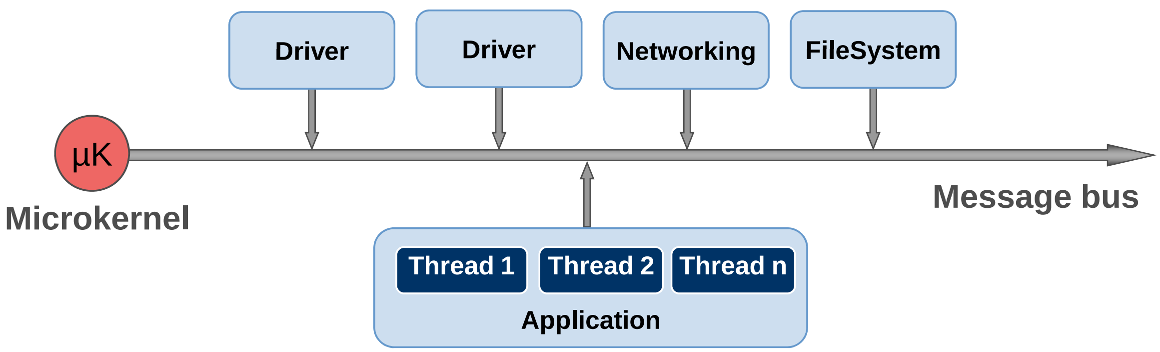
- Simple VxWorks ISRs that clear hardware interrupt source and perform a semGive are relatively easy to port to QNX Neutrino RTOS
- At the application layer, the ANSI-C/POSIX compatible APIs are nearly identical between VxWorks and QNX Neutrino RTOS
- Device drivers operation is very different between VxWorks and QNX Neutrino RTOS, and drivers must be rewritten

Porting phase – applications

2. Port

Execution model: mapping tasks

- Run the applications a single process under the new OS. Every task in the original legacy application becomes a thread in the new application process. Drivers run in their protected memory spaces. Application is protected from driver and OS.

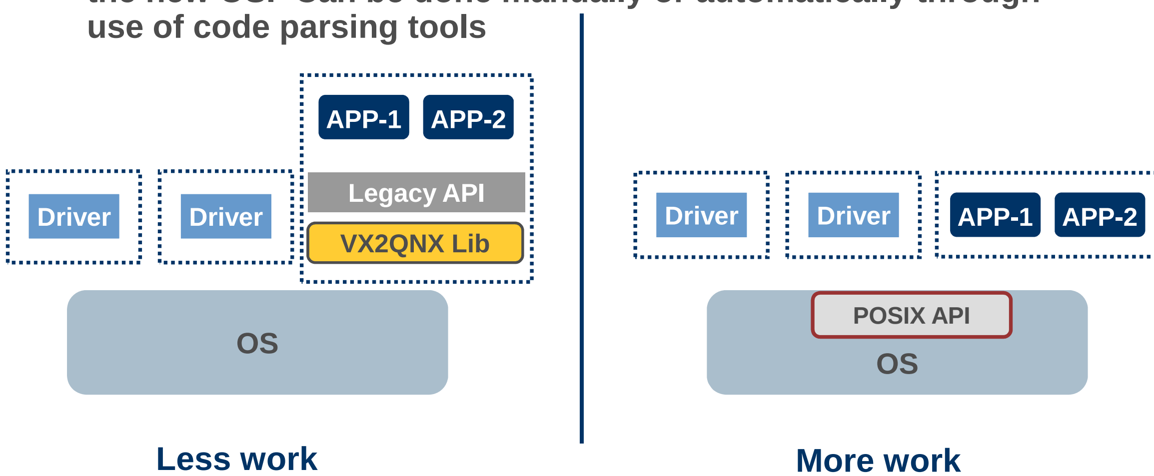


Porting phase – applications

API model : mapping APIs

2. Port

- Two main strategies for legacy application code
  - Develop porting library that provides legacy API while implementing it using underlying API calls of new OS
  - Replace legacy functions with appropriate native OS calls for the new OS. Can be done manually or automatically through use of code parsing tools



Porting phase – build environment

QNX SOFTWARE SYSTEMS

2. Port

C/C++ compiler and tool chain	WIND RIVER Diab	GNU
OSes available	vxWorks only	QNX Neutrino RTOS and vxWorks
Type checking and warnings	Medium	High
Library support	May have STL or other library incompatibilities	Standard
Build result	App or OS Image	Application Image

Optimization phase

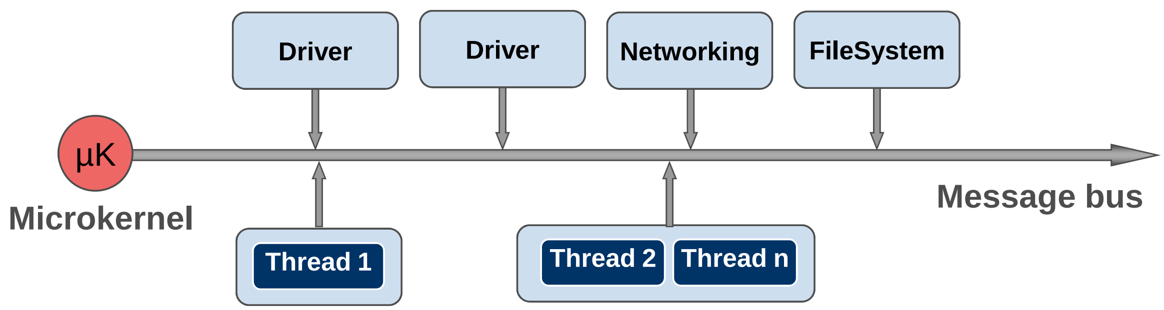
3. Optimize

- Refactoring for increased modularity
- Memory usage optimization
- Performance optimization

Optimization phase 1) modularity

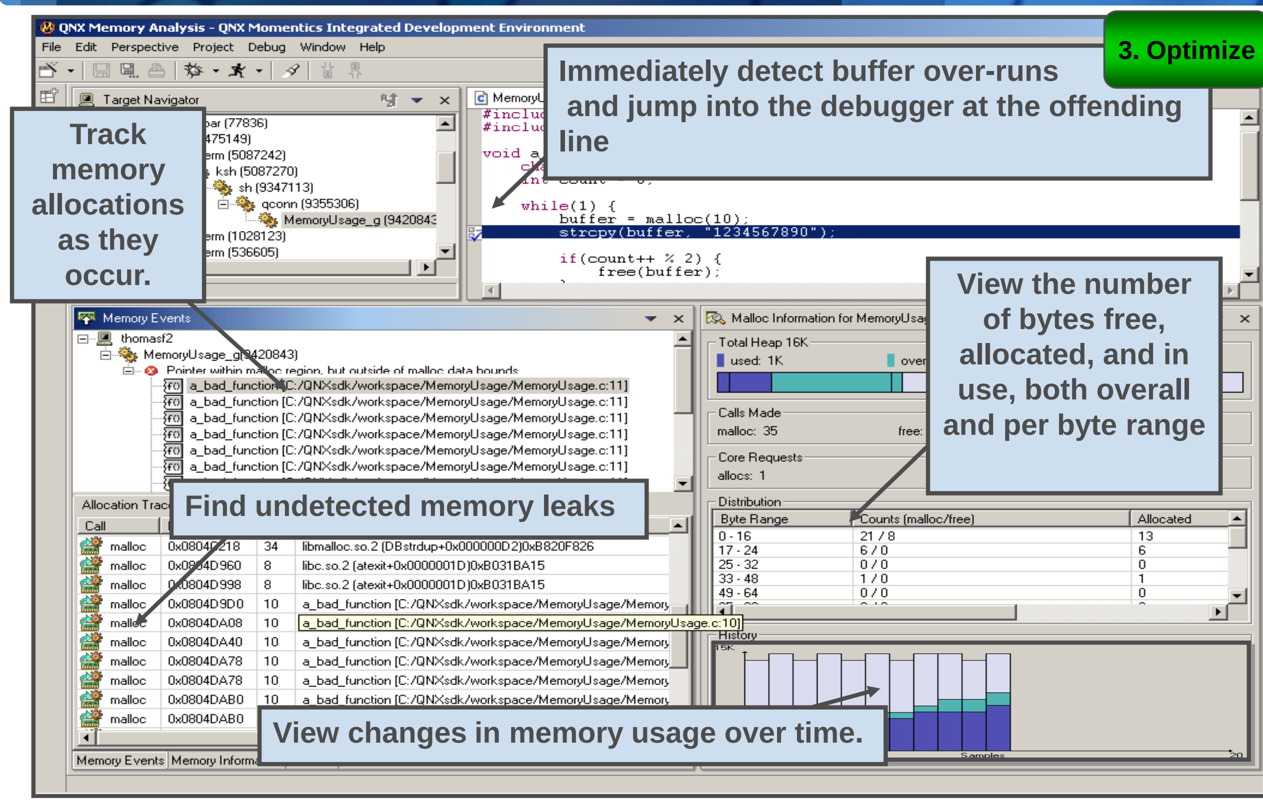
3. Optimize

- Group logical threads into separate processes to increase system modularity and reliability.



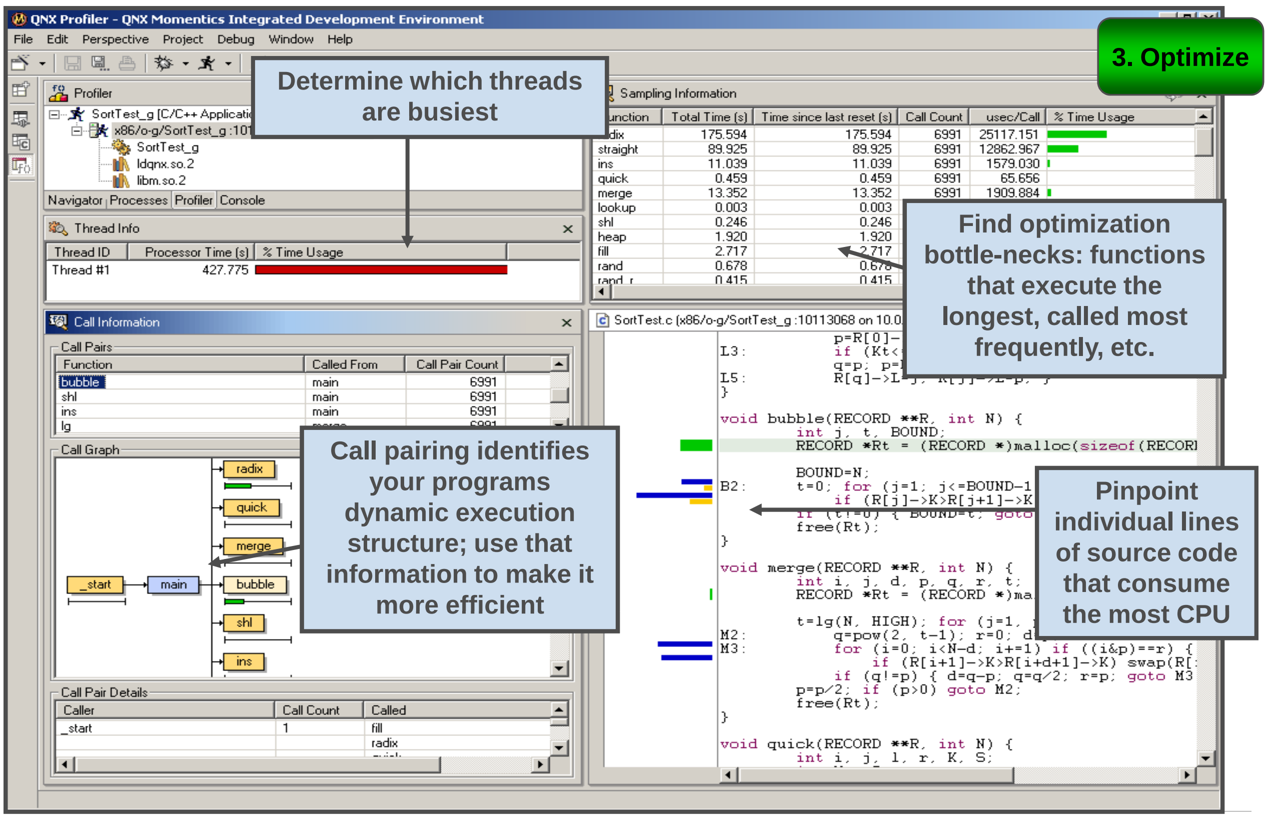
- Use IPC or shared data to communicate between processes

Optimization phase 2) memory usage



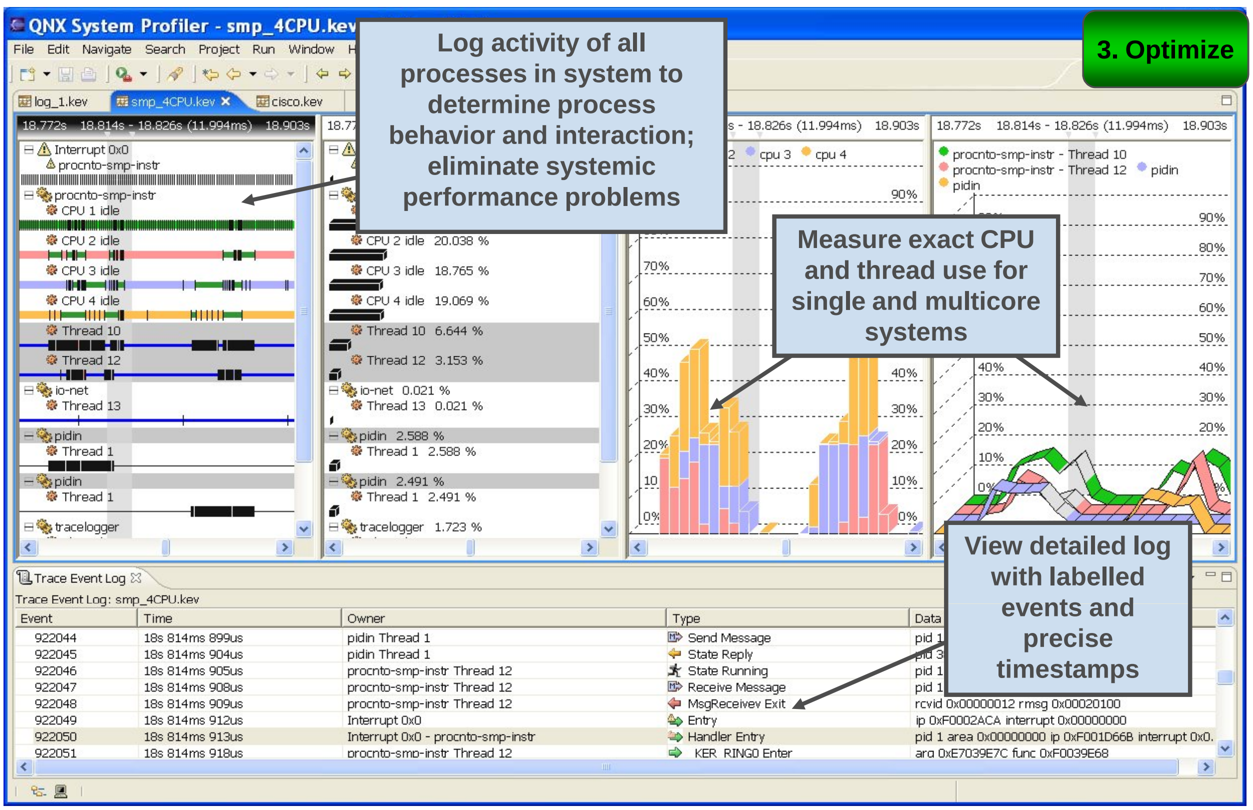
Optimization phase 3) performance optimization

QNX SOFTWARE SYSTEMS



Optimization phase 3) performance optimization

QNX SOFTWARE SYSTEMS



VxWorks migration Foundry27 project

QNX SOFTWARE SYSTEMS

- Comprehensive VxWorks porting guide provides
  - Detailed OS comparison
  - Porting guidelines, tips and tricks
- Vx2QNX Porting Library
  - Covers majority of core VxWorks API
  - Provides code compatibility with legacy code at the application layer.
  - Complete VxWorks system is encapsulated inside one process under QNX Neutrino RTOS
    - Task in VxWorks → Thread in QNX Neutrino RTOS
  - Provided as source - Use as reference for porting and/or deployment as a compatibility layer
- Forum support
  - Interact with QNX engineers

Contact QNX

- North America
  - T: + 1-800-676-0566
  - F: + 1-613-591-3579
- International
  - T: + 1-613-591-0931
  - F: + 1-613-591-3579
- Online
  - info@qnx.com
  - www.qnx.com
  - www.foundry27.com

Thank you for joining us.

Questions?