AUTOSAR ADAPTIVE DEPLOYMENT WITH OCI CONTAINERS FOR EMBEDDED USING VXWORKS

Peter Kleiner, Wind River
2022-05-03

DISCLAIMER

The content set forth herein does not constitute in any way a binding or legal agreement or impose any legal obligation or duty on Wind River.

This information is provided for discussion purposes only and is subject to change for any or no reason.

This document cannot be reproduced without the explicit permission of Wind River.

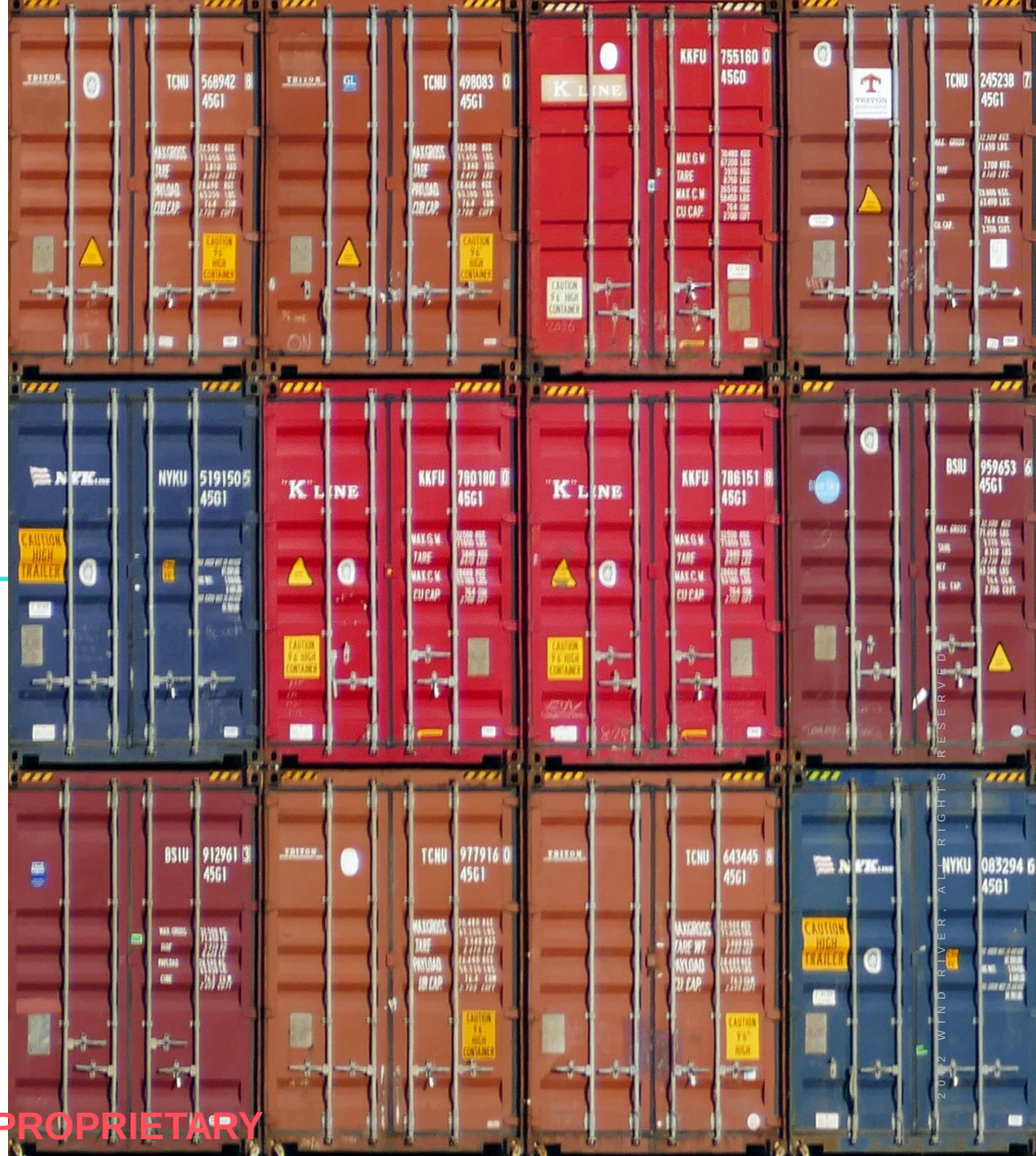
Agenda

1 Container technology

2 Container in Embedded with VxWorks

3 Application to AutoSAR Adaptive

WHAT IS A CONTAINER?



A WAY TO ENABLE SOFTWARE TO RUN RELIABLY WHEN MOVED

WNDRVR

CONTAINER TECHNOLOGY IS A METHOD OF PACKAGING AN APPLICATION SO IT CAN BE RUN WITH ISOLATED DEPENDENCIES

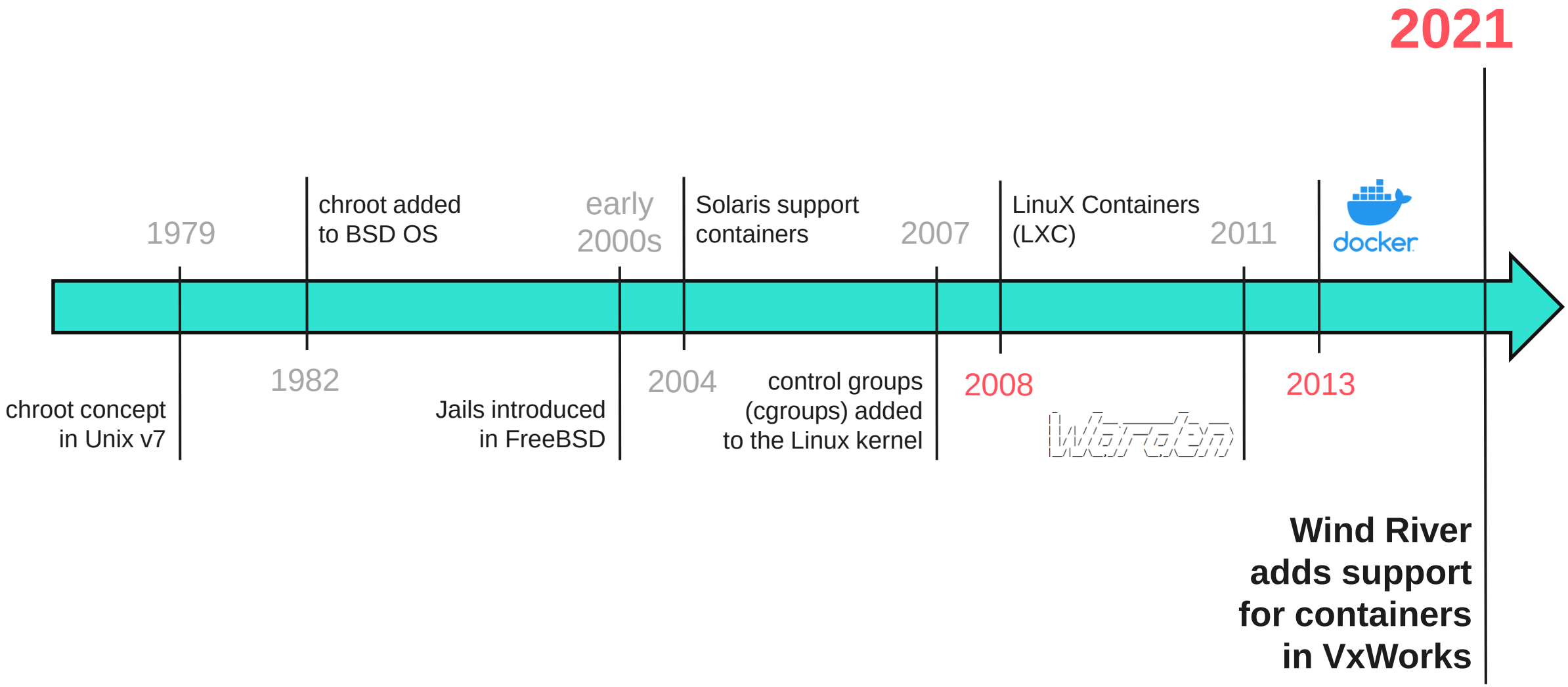
* https://www.section.io/engineering-education/history-of-container-technology/

STANDARD OPEN TECHNOLOGY

A BRIEF HISTORY OF CONTAINER TECHNOLOGY

WE ARE PAST THE PEAK OF INFLATED EXPECTATIONS

WNDRVR



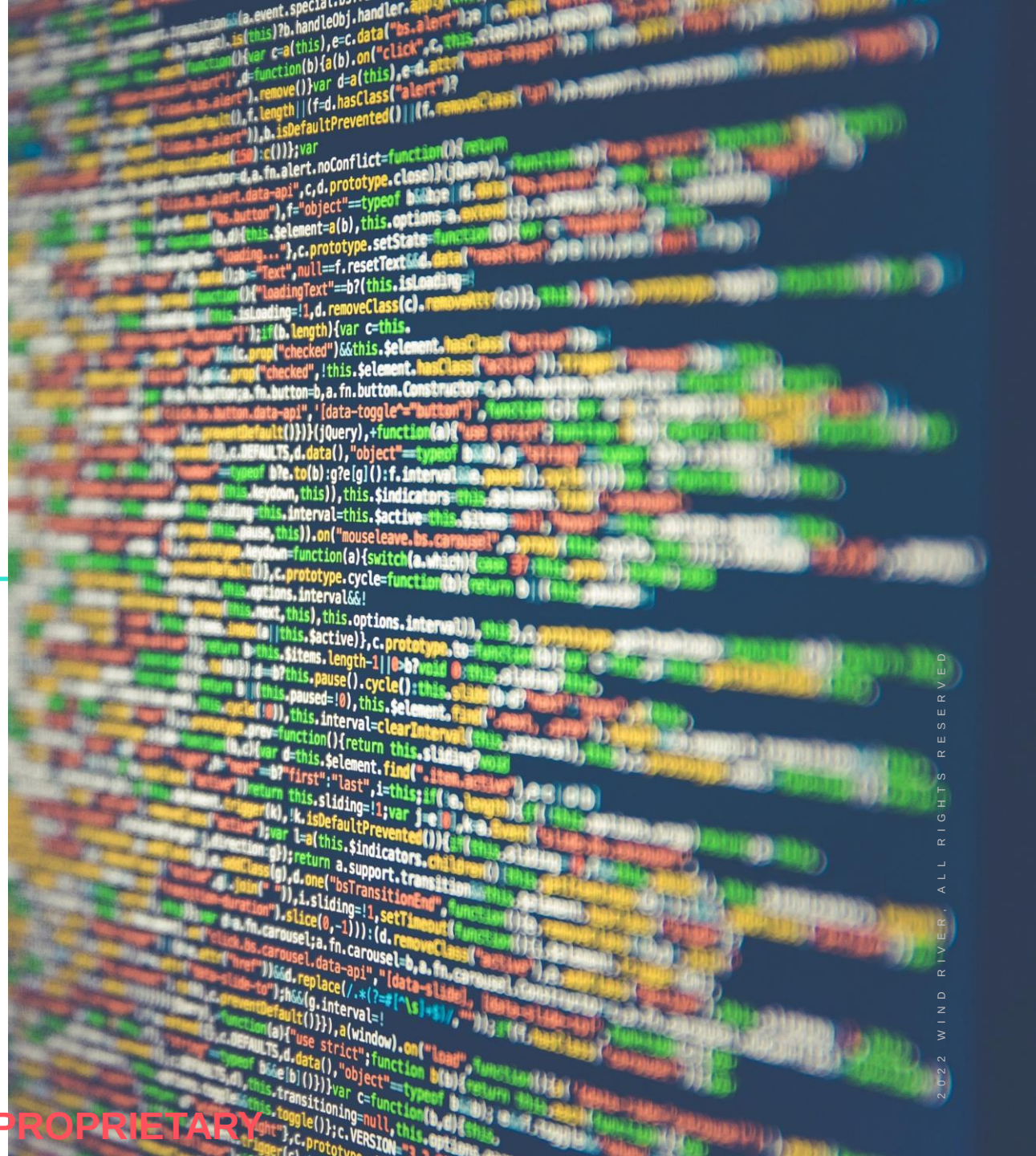
PERCENTAGE OF GLOBAL ORGANIZATIONS RUNNING CONTAINERIZED APPLICATIONS IN PRODUCTION BY 2025 *

85%

* https://www.gartner.com/doc/reprints?id=1-24237Z3C&ct=200831&st=sb/

SIMPLIFIED MANAGEMENT

EASILY MANAGE DEPLOYED SOFTWARE



WNDRVR

LIFECYCLE IN EMBEDDED

- Fixed function/monolithic approach
- Traditionally disconnected (or with limited connectivity)
- Very long deployment life
  - 8 to 10 years for an industrial robot *
  - 20 to 36 years for the average aircraft **
- Risk aversion
- Deployment models changed over the years with limited commonalities
  - Technology or systems limitations, etc.

* motioncontrolsrobotics.com – Robot life cycle faq
** Key Findings on Airplane Economic Life, Boeing

LIFECYCLE AT AWS –

AWS ENGINNERS DEPLOY CODE EVERY

11.7s

* invent 2019

CONTAINER IN EMBEDDED WITH VXWORKS

Wind River provides OCI Container support for

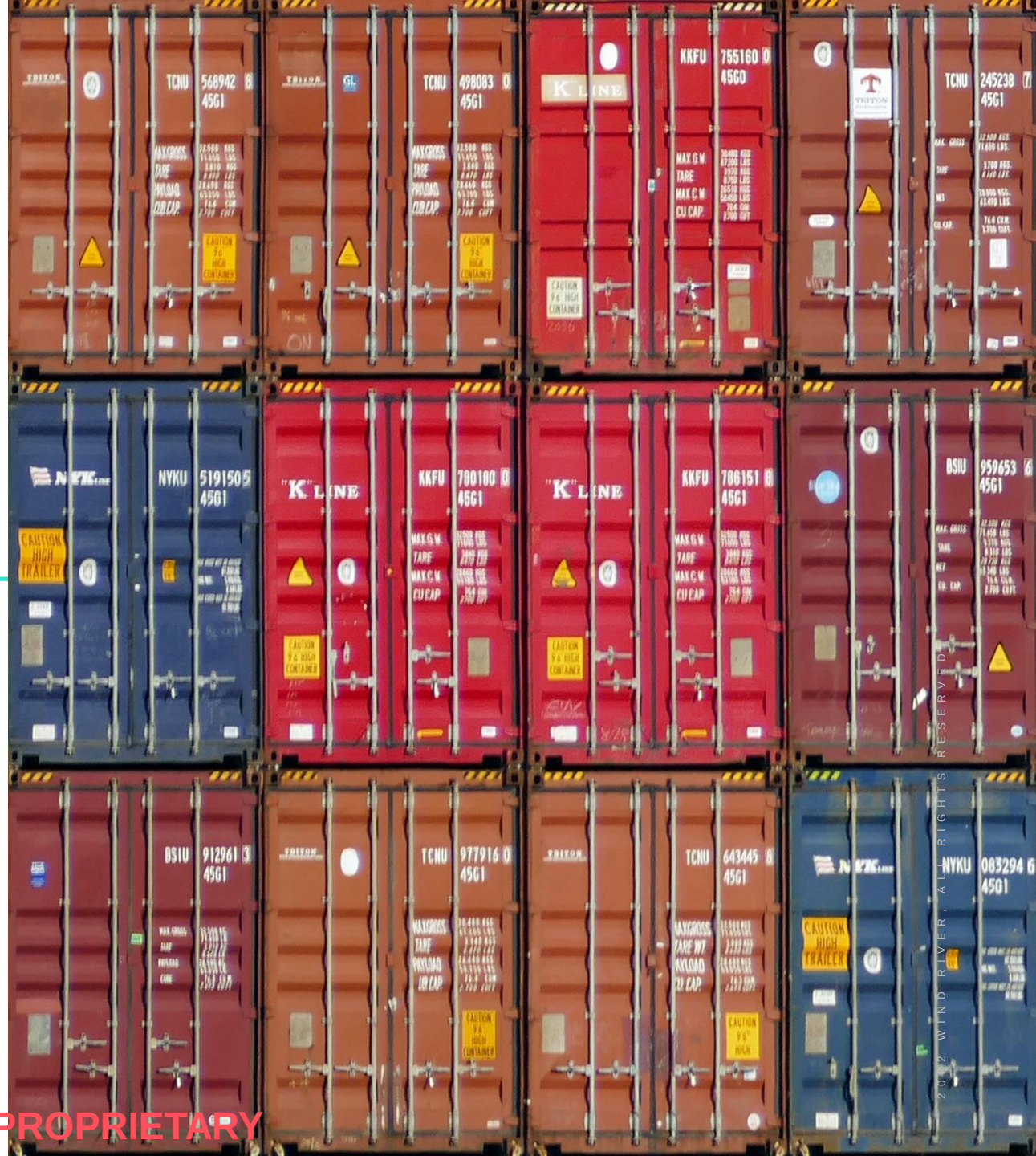
- Safe Real Time OS VxWorks

VxWorks

- Yocto based Wind River Linux

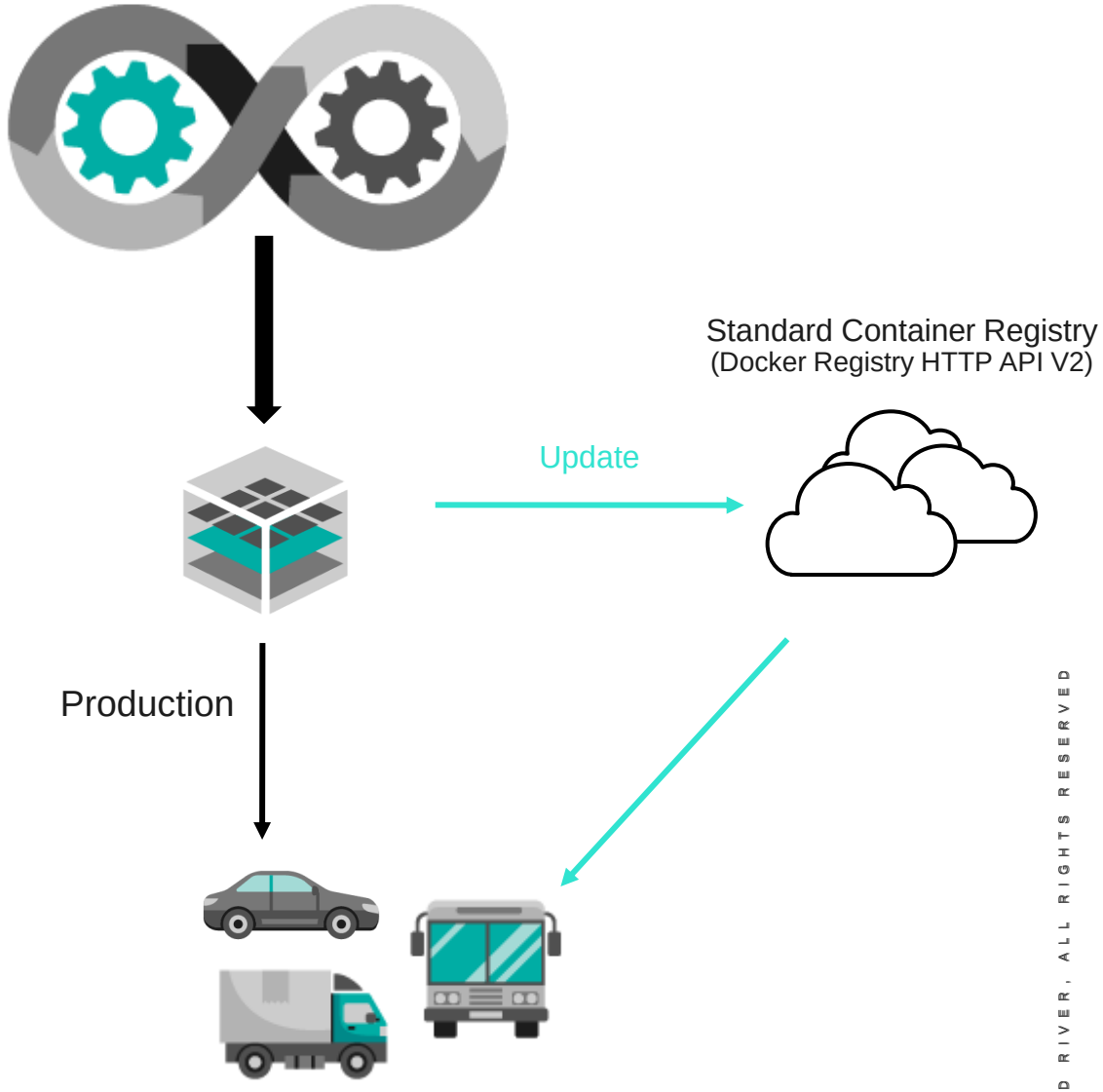
Wind River Linux

WNDRVR

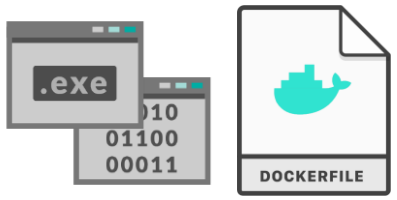


WHAT ABOUT VXWORKS?

- Compliant with OCI specifications
- Runtime
  - Image parsing/validation
  - Instantiation of the container
  - Execution of the application
- Manager
  - Logic for pulling containers from registry
  - Command line tools for development/testing



WHAT ABOUT VXWORKS?




```
FROM scratch
WORKDIR /usr
WORKDIR /vxbin
COPY helloworld.vxe /vxbin
ENTRYPOINT ["helloworld.vxe"]
LABEL com.windriver.vxworks.rtp.rtpStackSize 0x400000
LABEL com.windriver.vxworks.rtp.rtpPriority 50
LABEL com.windriver.vxworks.rtp.rtpOptions 0x80
LABEL com.windriver.vxworks.rtp.rtpTaskOptions 0x01000000
VOLUME "/ram0/usr:/usr"
VOLUME "/shm:/shm:--:iodev"
VOLUME "/romfs:/romfs:--:iodev"
USER 1001:1000
STOPSIGNAL SIGTERM
```

HOW DO CONTAINERS HELP?

- Easy to “tear down” and restart
- Can bundle multi-systems applications (e.g. VxWorks Arch1/Arch2 or VxWorks + Linux)
- Micro services/micro function architecture
  - Can upgrade parts of the system only (useful for security bugs)
  - Consider a progressive approach
- Large ecosystem of tools

WHY LEVERAGE THE IT INFRASTRUCTURE?

- No need to re-invent the wheel
  - Use what is there for the IT operations of you or your clients
- Easy to integrate
  - Many components are open and community driven (e.g. K8s)
  - Talent exists to help transition
- Quicker to deliver
- Reduces amount of development and testing

CONTAINER TECHNOLOGY IS A METHOD OF PACKAGING AN APPLICATION SO IT CAN BE RUN WITH ISOLATED DEPENDENCIES

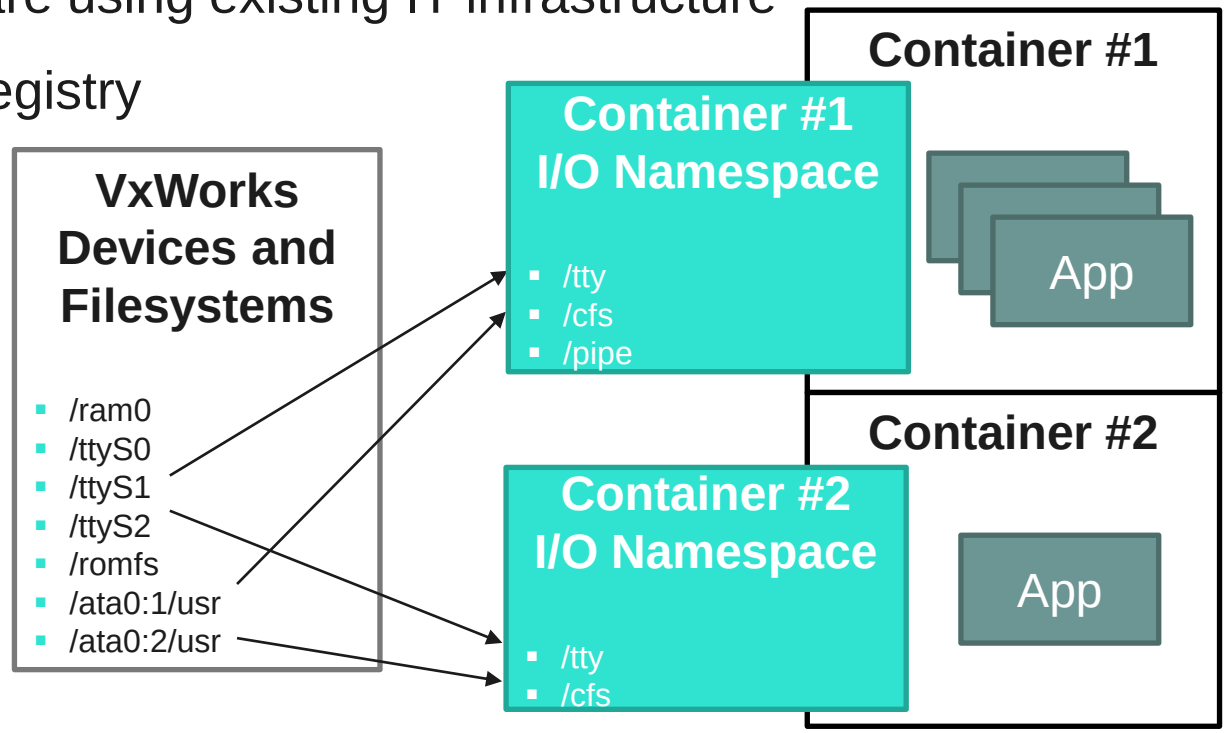
* https://www.section.io/engineering-education/history-of-container-technology/

BUILD ONCE AND RUN ANYWHERE

- Lightweight engine
  - In VxWorks, due to careful design of the engine, little to no runtime performance impact for the application
    - Startup time could be impacted if container image remains packed
  - Small footprint
- Easier to update various layers of the application
- Reduces impact when OS gets updated

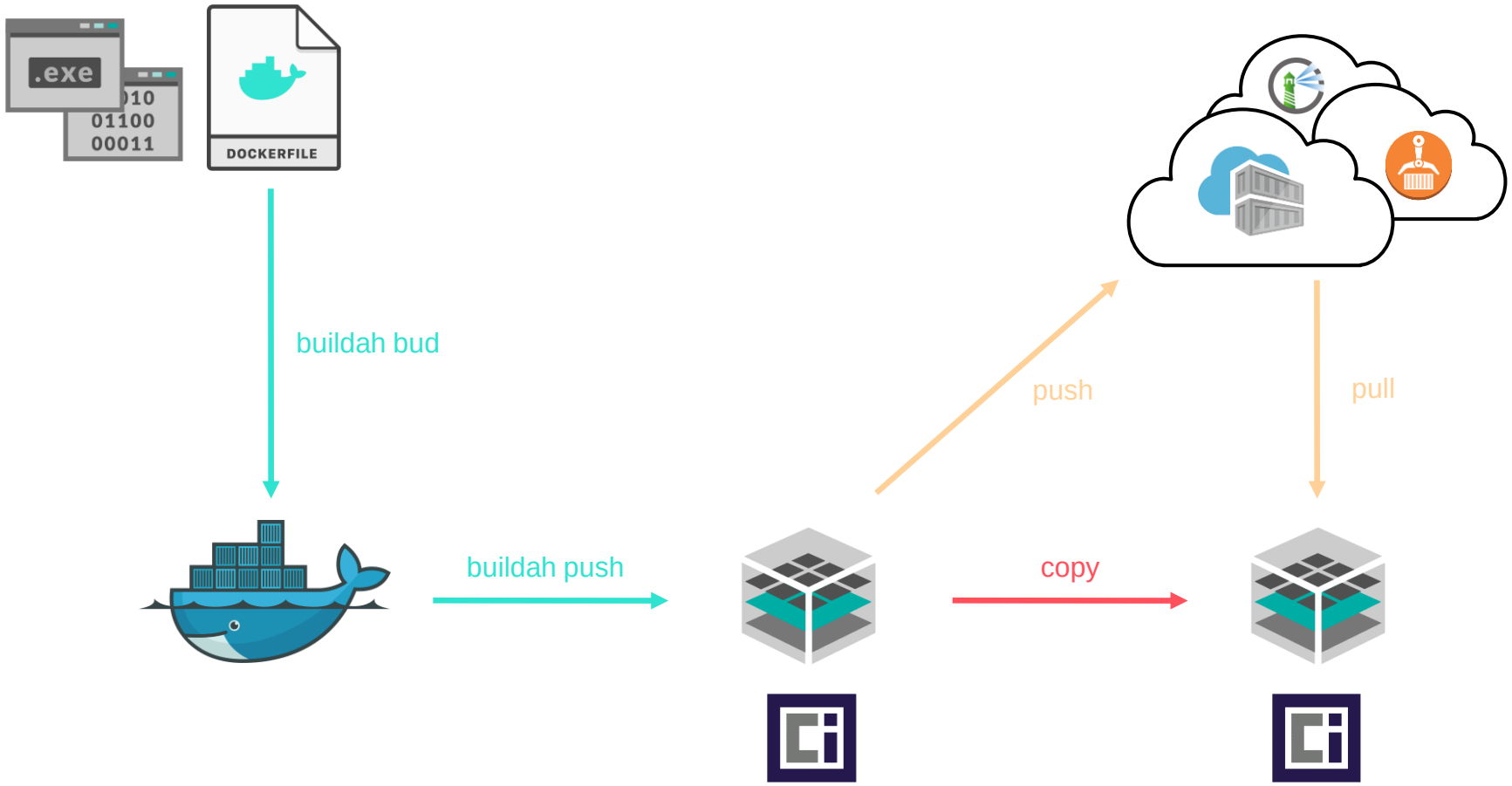
ISOLATION

- Standard packaging for applications across architectures and operating systems
  - No Wind River proprietary tools
  - Use standard Dockerfile
- Quick update of deployed software using existing IT infrastructure
- Authentication of the container registry
- Device isolation
- Overlay filesystem



COMMON PROCESSES

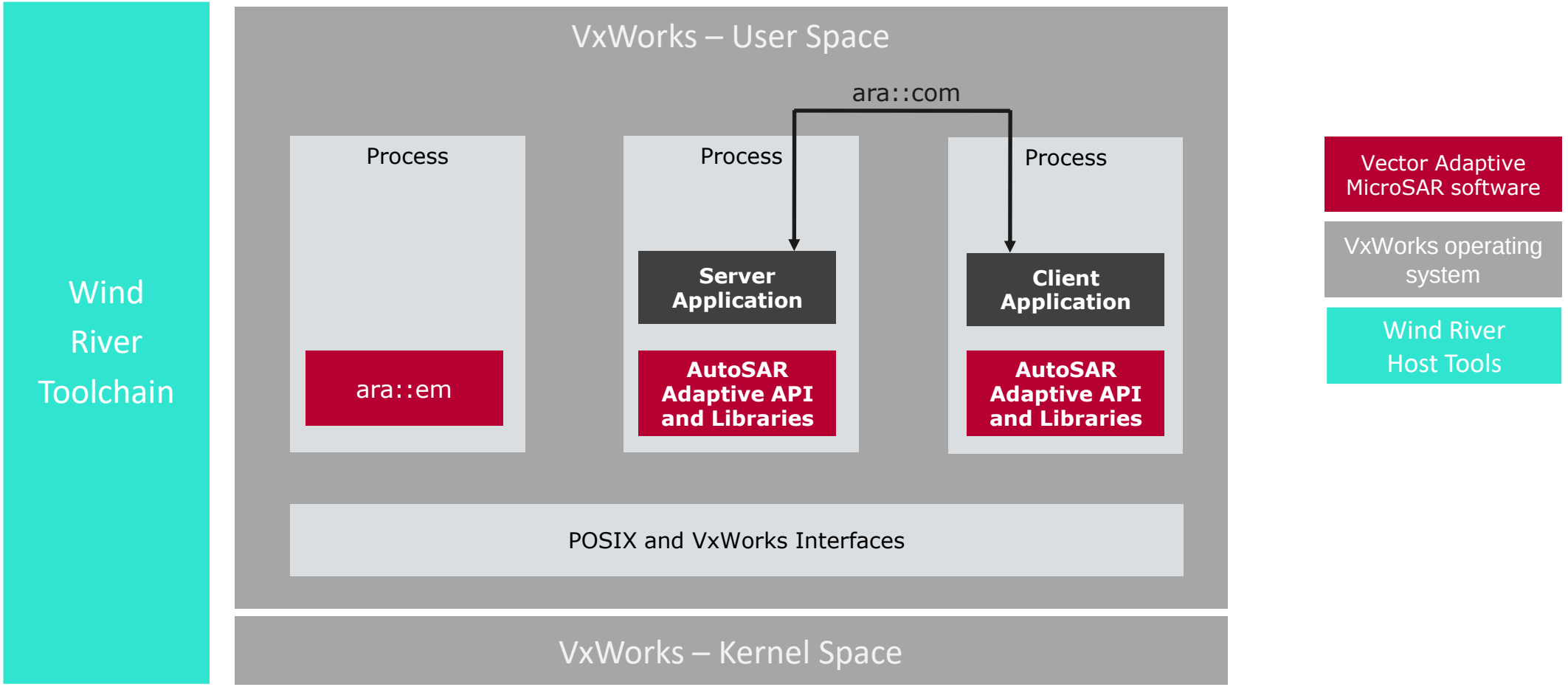
COMMON TOOLS



CONTAINER AND AUTOSAR ADAPTIVE

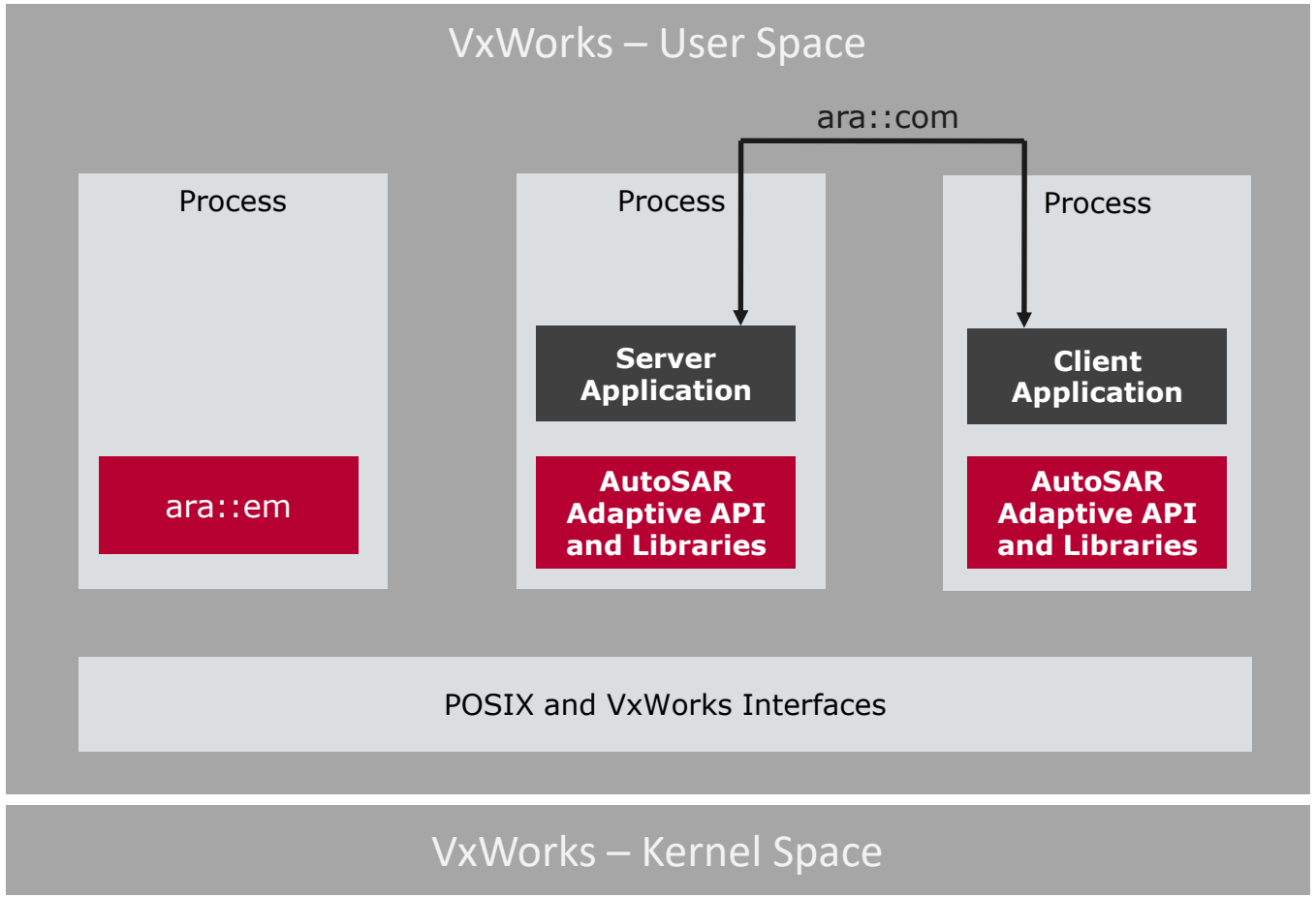
WNDRVR

ADAPTIVE MICROSAR AS VXWORKS PROCESSES PARTNER SOLUTION: VECTOR - WIND RIVER



ADAPTIVE MICROSAR AS VXWORKS PROCESSES CLOUD NATIVE DEVELOPMENT

Wind River Toolchain In Studio



Virtual Lab – Digital Twin in Studio

Vector Adaptive MicroSAR software

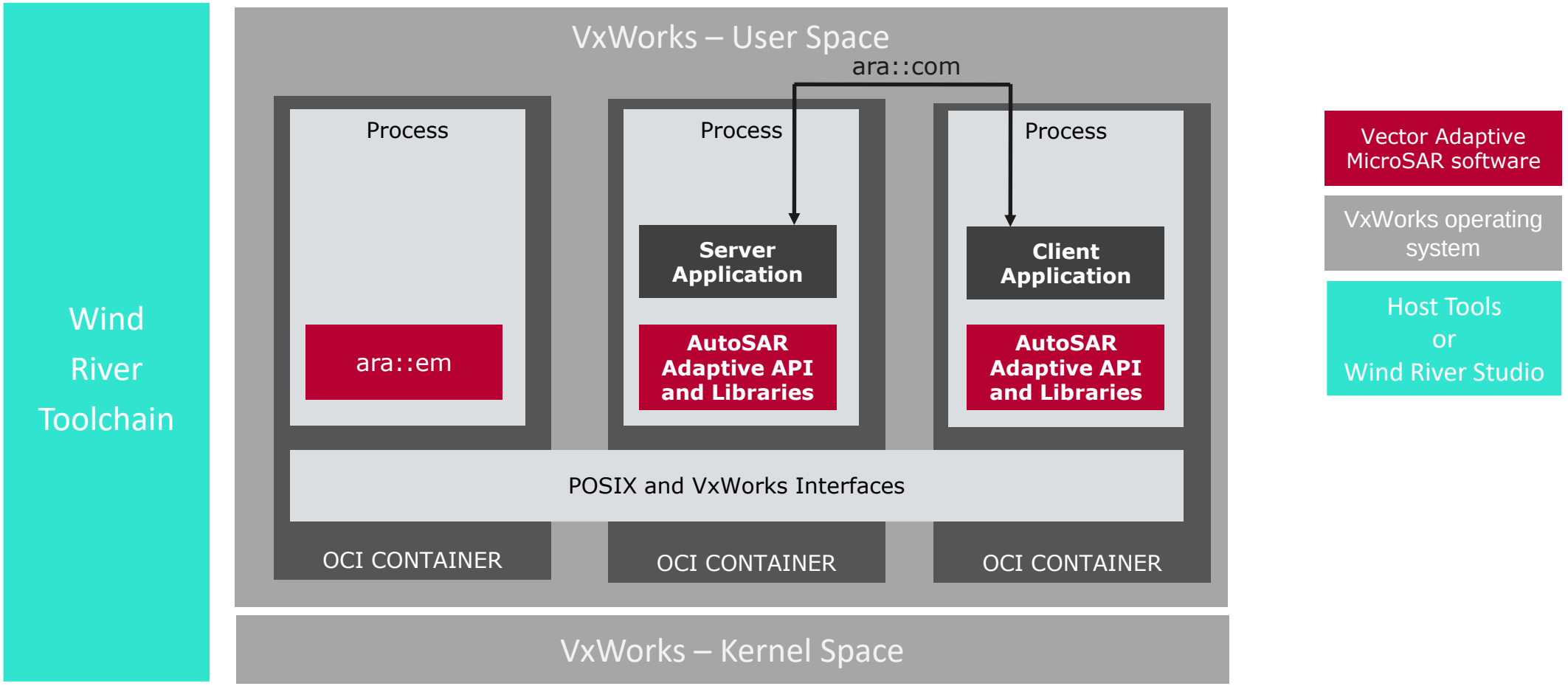
VxWorks operating system

Wind River Studio

WNDRVR

WIND RIVER PROPRIETARY

AUTOSAR ADAPTIVE APPLICATION DEPLOYED AS CONTAINER USING VXWORKS CONTAINER RUNTIME



CONSIDERATIONS

- ara::em needs to updated to spawn containers instead of processes
  - Identify container to be used for set of applications
  - Start/Stop container instead of processes
- File based sharing needs to be known and mapped properly
  - SOME/IP settings
- Adaptive SW Update needs to be made ready for container
- Container creation is packaging exercise post Adaptive application build
- JSON file for ARA::EM and Dockerfile contain overlapping information

Q&A

Visit us at
www.windriver.com

WNDRVR
WHAT'S NEXT