
AWS
re:Invent

OPN402

Rust-vmm: Secure VM-based isolation made simple

Andreea Florescu
Software Development Engineer
AWS

Samuel Ortiz
Principal Engineer
Intel

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

Intro: VM-based isolation

Intro: rust-vmm

Building custom VMMs

Future plans

Getting involved



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Intro: VM-based isolation



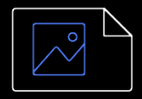
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

VM workloads

- Serverless
- Containers
- General-purpose workloads

VM workloads: Top-down view

Workload	 Process image	 Games
Product	 AWS Lambda	 Amazon EC2
Virtualization	 Firecracker	AWS Nitro System
Virtualization components	- Block device - Network	- Block device - Network - GPU

Intro: rust-vmm



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What is rust-vmm?

- Building blocks for custom VMMs (Virtual Machine Monitors)
- Virtualization components written in Rust
- Open source: https://github.com/rust-vmm

Rust-vmm in production

Virtual Machine Monitors (VMMs)

- Firecracker
- Cloud Hypervisor
- Alibaba Cloud Sandbox
- Enarx

Containers

- libkrun
- Nydus: Dragonfly Container Image Service

Building custom VMMs



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

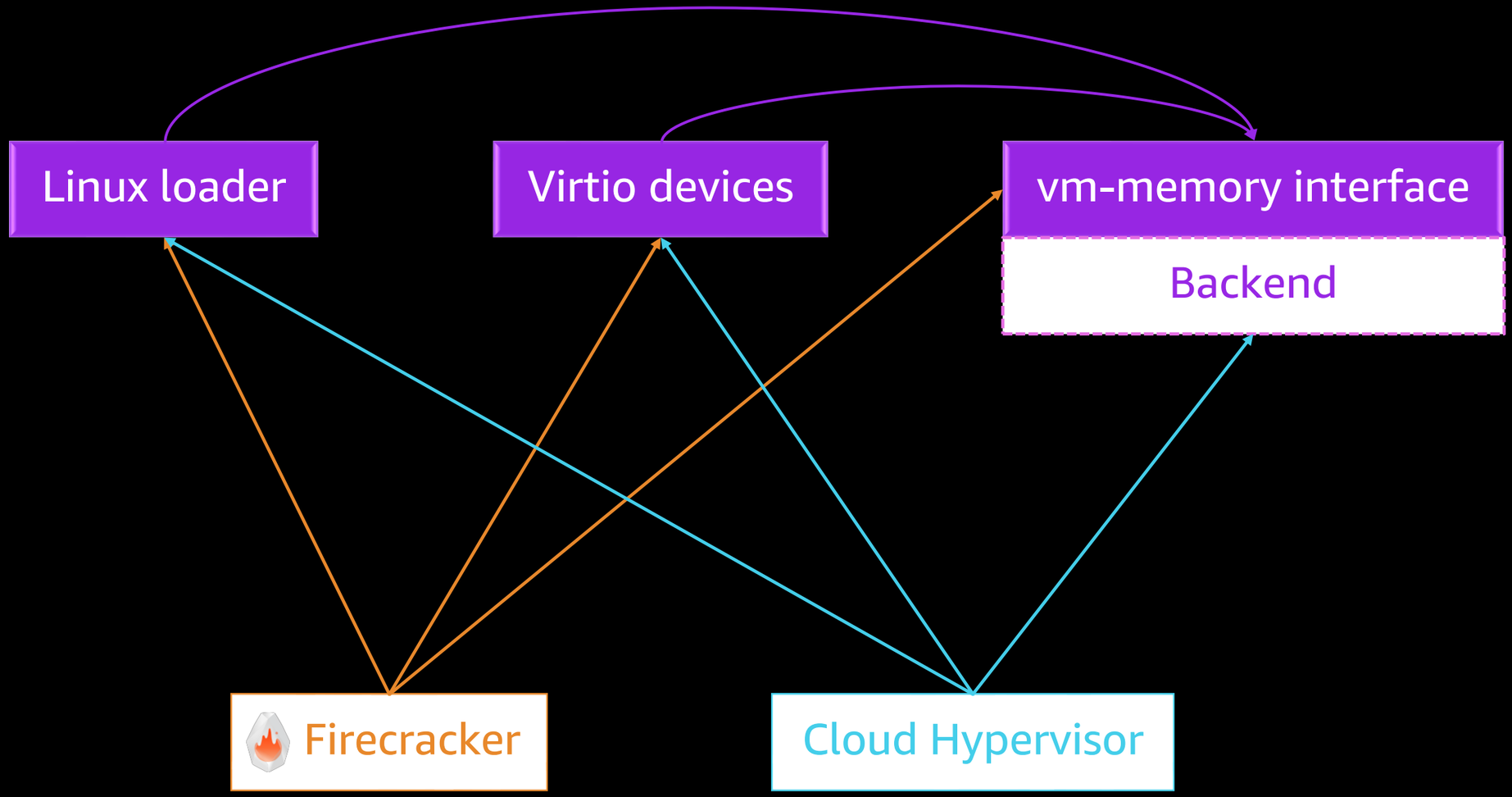
Enabling multiple implementations

- rust-vmm/vm-memory
  - Configure and access physical memory of a Virtual Machine
  - Public interface: *GuestMemory*
  - Optional interface implementation: *GuestMemoryMmap*
- Firecracker
  - Consumer of vm-memory interface
  - Custom GuestMemory implementation

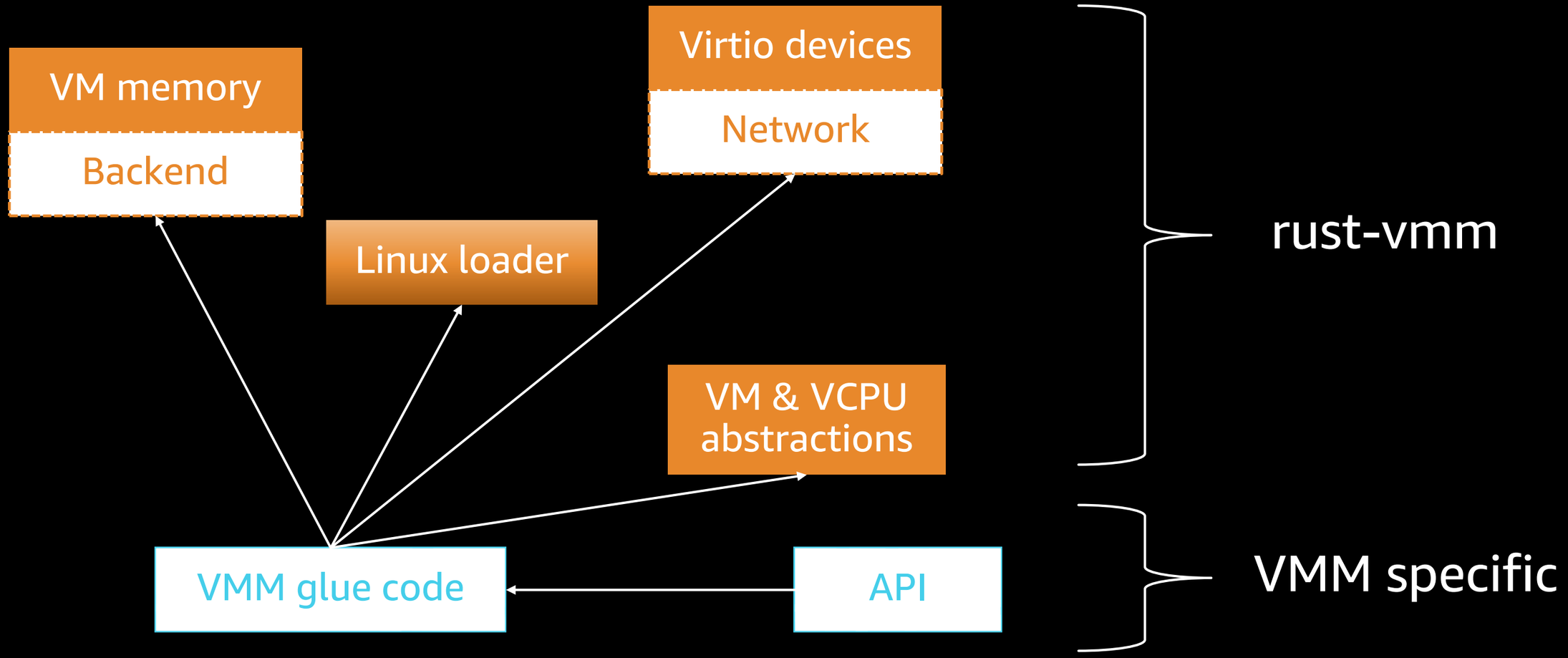
Dependencies between components

- Linux loader
  - Load kernel images in guest memory
  - Requires memory abstractions for writing images
  - Depends on the *vm-memory interface*
- Virtio devices
  - Implementation of virtio devices
  - Communication between driver and device -> shared memory
  - Requires memory abstraction for handling the communication
  - Depends on the *vm-memory interface*

Enabling laser-focused implementations

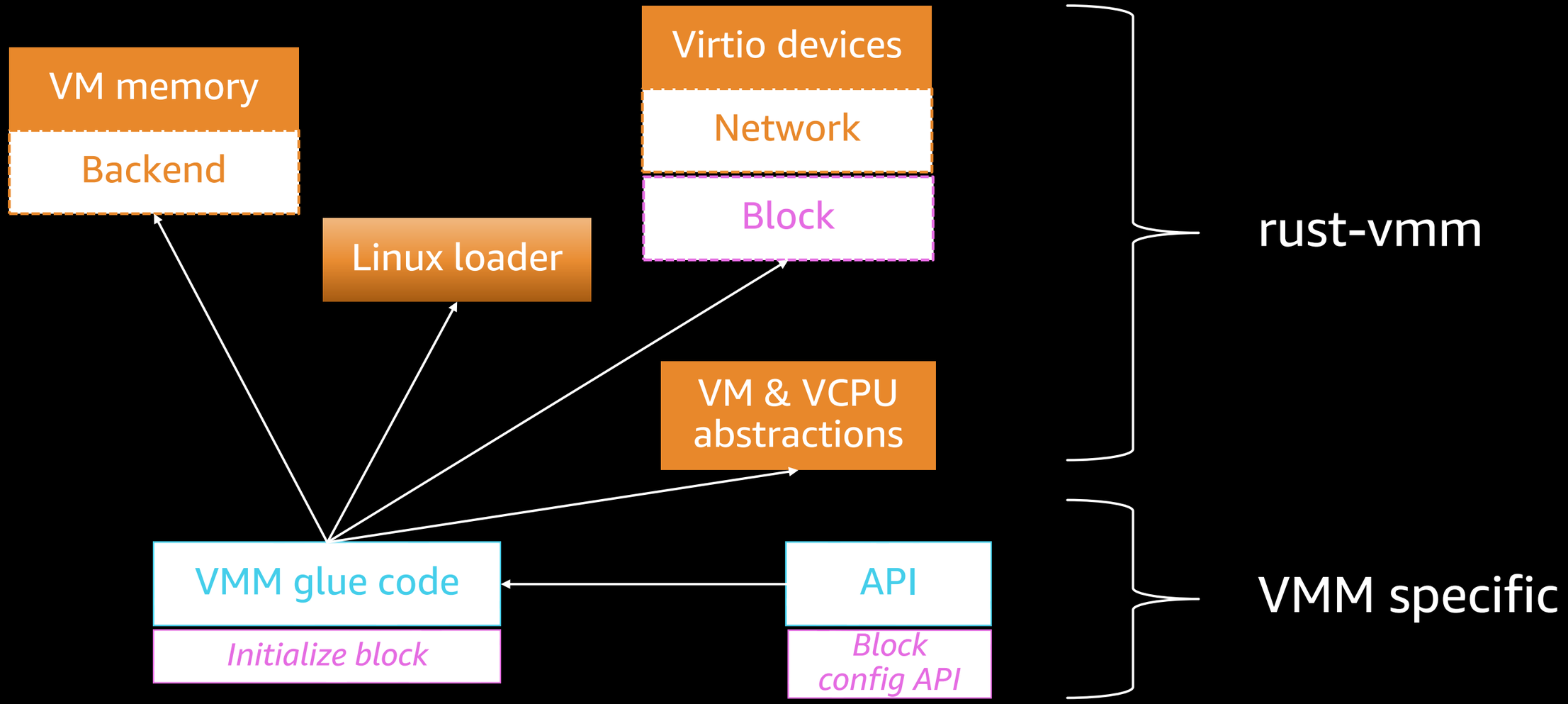


Http server

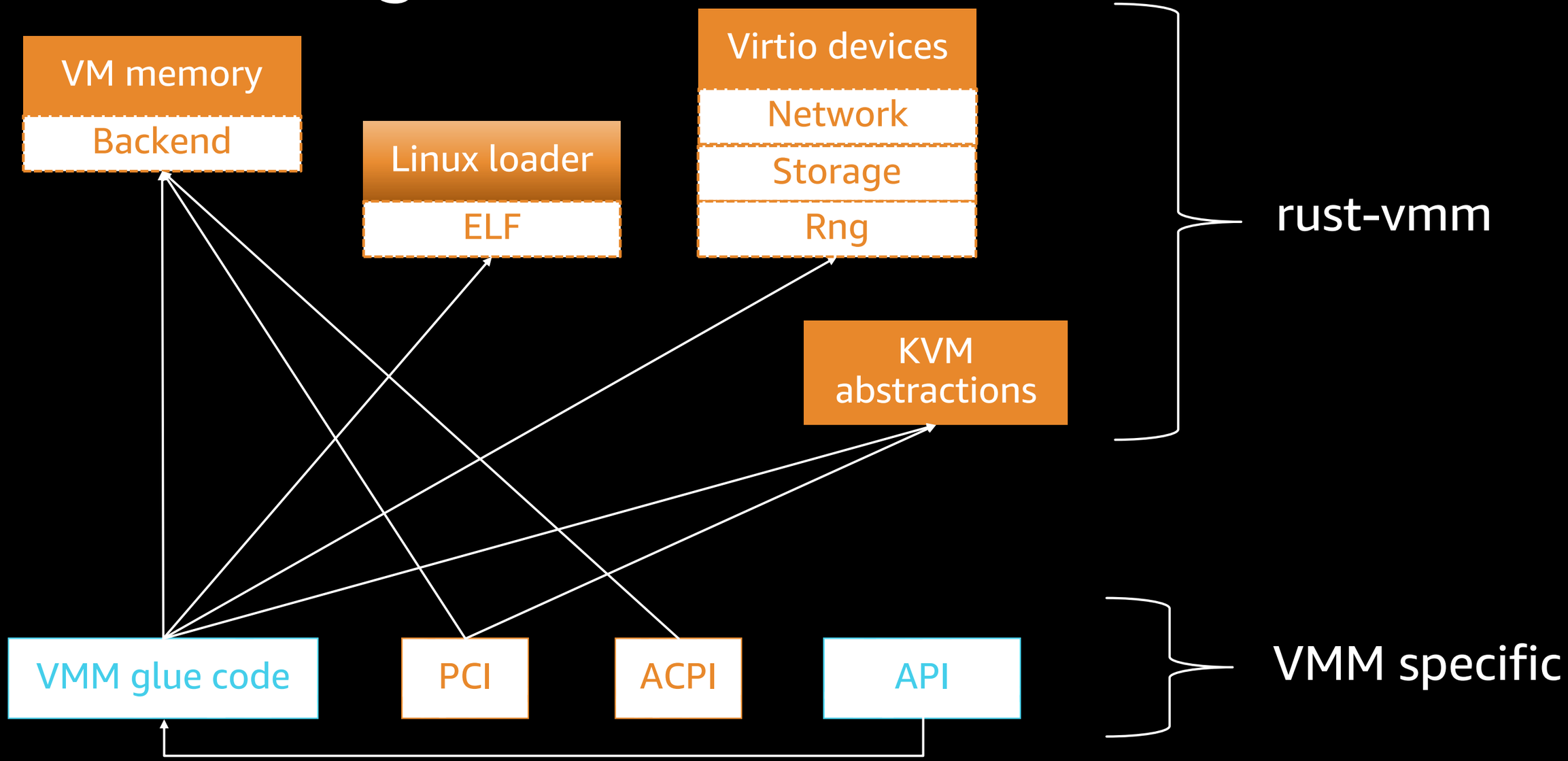

VM memory
Backend
Virtio devices
Network
Linux loader
VM & VCPU abstractions
VMM glue code
API
rust-vmm
VMM specific

HTTP server + storage

REQUIRED CHANGES



Windows guest



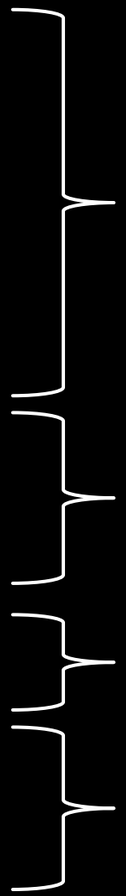
Future plans



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Virtualization components

- vhost-user-backend
- Virtio
  - Block
  - Network
  - Vsock
- VCPU abstractions
- ACPI
- Hypervisor abstraction
- PCI
- VFIO



IO (storage, networking, etc.)

Windows workloads & multi-platform

Microsoft Hypervisor

GPU workloads

Security and testing

- Threat model documentation
- Continuous fuzzing
- Extend performance tests

Getting involved



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Contribute to rust-vmm

- New virtualization components
  - Design discussion via rust-vmm/community/issues
  - Maintainers create a repository and set up the CI
  - Incremental updates and publish to crates.io

- Existing components
  - Good first issues
  - Help wanted
  - Issues without owners

Use rust-vmm

- Create laser-focused VMMs
  - vmm-reference as a starting point
  - Mix and match existing rust-vmm components
- Production ready/developer preview components
  - rust-vmm/community
  - crates.io

Let’s chat!

- Slack

  https://bit.ly/2VRfGc6
- Email

  rust-vmm@lists.opendev.org

Thank you!

Andreea Florescu
fandree@amazon.com

Samuel Ortiz
samuel.ortiz@intel.com

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.