

A Plan of an NICT P4 Testbed

Eiji Kawai

Director, ICT Testbed Research, Development and Operations Laboratory
National Institute of Information and Communications Technology



Data-Plane Programmability

- Next-generation open networking technology
 - Programmable packet processing pipelines
 - P4 programming language
 - Stateful packet processing
- Research on the data-plane programmability
 - Programmable data-plane architecture
 - Data-plane programming frameworks
 - Compiler extensions for data-plane programming
 - Verification of data-plane programs
 - Use cases (Orchestration, Novel protocols, Telemetry, Security, etc.)

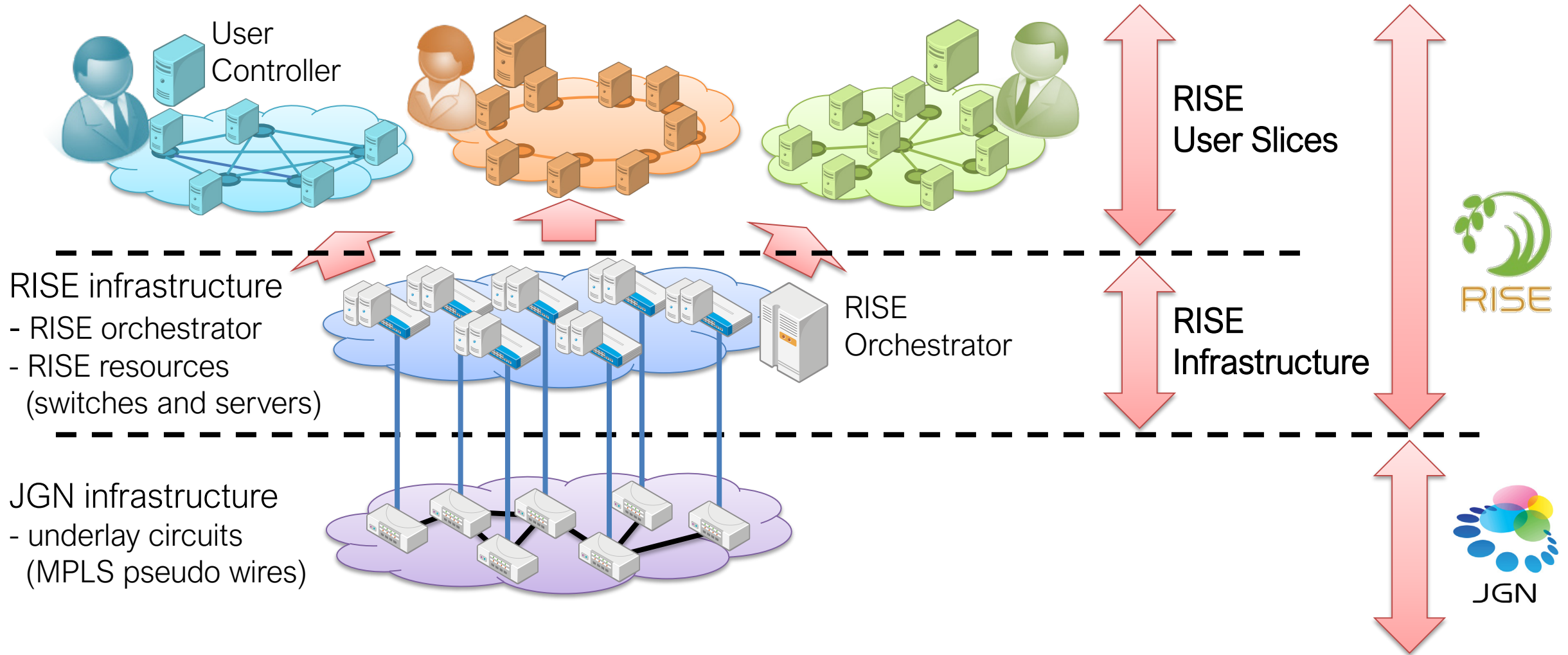


How to Design a Data-plane Programmable Testbed



- Operatability
 - for operation cost effectiveness
 - boundary/interface between the users and the operator
 - how to help the users to debug
- Multi-tenancy
 - for infrastructure cost effectiveness
 - time-division, physical space-division, logical space-division
 - translation/conversion

Example: RISE (Control-plane Programmable Testbed)



Prototyping an NICT P4 Testbed

- Switches
 - Software-based (ex. P4 bmv2, PISCES, SONiC, ...)
 - Hardware-based (ex. NetFPGA, Smart NIC, Tofino, ...)
- We are prototyping a P4 testbed with software switches (bmv2).
 - Based on the software-based RISE environment
 - Replace software OpenFlow switches with bmv2 instances
 - Enhance the RISE orchestrator
- Many open issues
 - Debugging
 - “My program doesn’t work in the testbed which worked in my lab.”
 - Multi-tenancy
 - How to virtualize the hardware-based nodes

Thank you!!

Contact info:

Eiji Kawai, eiji-ka@nict.go.jp