RICAPP: Cherry epics

Matti Hiltunen, PTL
hiltunen@att.com
Main xApps in O-RAN SC Cherry Release

- **QP driver xApp**
- **QuE Predictor (QP) xApp**
- **KPI Monitor xApp**
- **UE metrics**
- **Traffic Steering (TS) xApp**
- **cell metrics**

Languages:
- **C++**
- **Python**

Environments:
- **Measurement Campaign (MC) xApp**
- **HelloWorld (HW) xApp**

Integration Points:
- **E2 (KPM SM)**
- **E2 (TS SM)**
- **A1-P**

**CU/DU/E2 SIM**
Main RICAPP epics for Cherry

• Extend the use of RIC SDK to support O1-PM/FM/CM and A1-P:
  • Implement alarming using new alarm API in SDK
  • Implement statistics collection using new stats API in SDK
  • Exercise new RIC SDK higher-level abstractions for A1-P, O1-CM

• TS use case extensions:
  • More interesting A1-P policy intents
  • E2 CONTROL message (initially using our own SM, later WG-3 SM)
  • QP with real machine learning model and training
  • Integration with O-DU/O-CU or more realistic simulator (Viavi?)
  • Design and integration with ML training outside the RIC

• More comprehensive example xApps.
How to support ML training in Cherry?

Training data: original input features, (predicted value), measured value (after some time period)

Multiple open issues:
• How to communicate the training data to non-RT RIC (e.g., VES file transfer) – WG-2
• New SDK API for xApps to communicate the info? – nearRT-RIC
• Which xApp should collect the training data? – RICAPP
• What features are used for prediction (e.g., are the identities of the current serving cell and the target neighbor cell used) – RICAPP, R-SAC
• Which component will consume the training data in Cherry release – non-RT-RIC - ML FRQAMEWORK
• Reinforcement learning? Large data set needed?
What is RIC SDK?

Enable xApp to:
- Define xApp descriptor for on-boarding and dynamic configuration
- Send and receive messages
- Access data persistence layer (SDL), R-NIB and other database
- Communicate with E2 Nodes
- Report alarms, statistics, and logs

SDK components:
- Libraries, Interfaces and APIs
- Tools
- Documentation and guides
- Code Samples
- Processes and Flows

Help developers:
- Generate skeleton codes
- Setup development and testing environments
- Package, distribute and onboard their xApp

Provide an abstraction layer to insulate xApps from a specific underlying technology, improve portability.

https://wiki.o-ran-sc.org/display/ORANSDK/
RIC SDK Libraries, Interfaces and APIs

- Services provided by other xApps
- App frameworks (Go, C++, Python) with higher-level APIs
- A1, O1, and E2 interfaces
- RAN inventory management (R-NIB, UE-NIB, MC-NIB)
- Messaging (RMR) + shared data layer (SDL) (+ logging, tracing)
- xApp descriptor + life cycle management
- Kubernetes

The RIC SDK implementation could be utilized directly in other projects (e.g., logging, life cycle management) or the RIC SDK API could be utilized with a different underlying implementation – initial discussions underway.