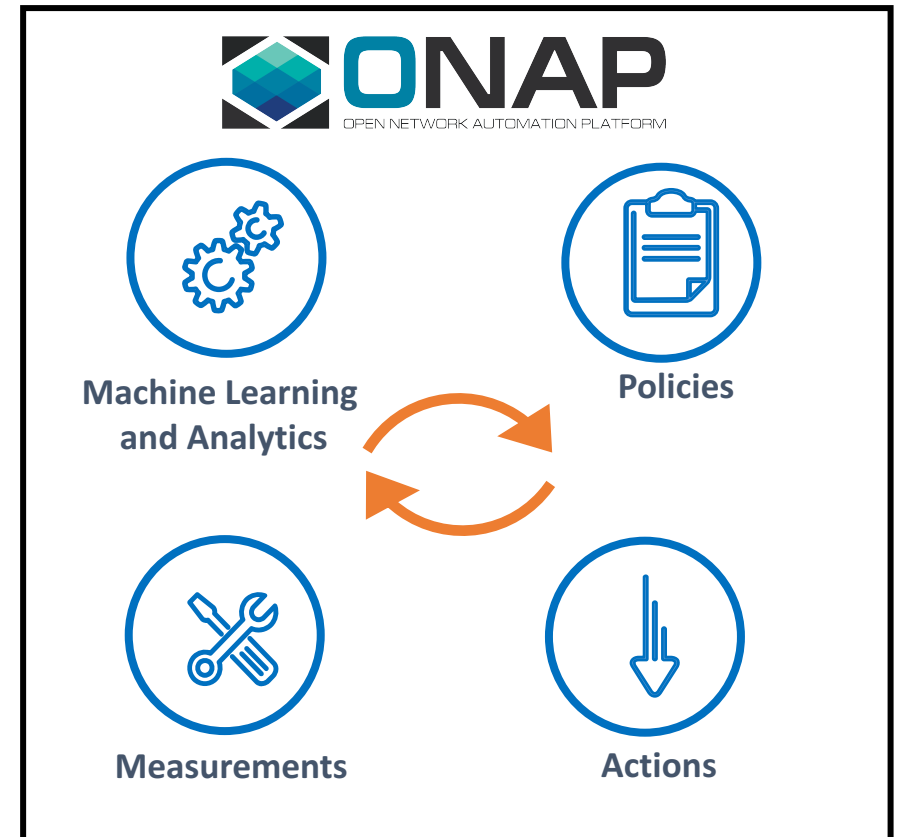# ONAP Control Loops

- ONAP Whitepaper
  - https://www.onap.org/wp-content/uploads/sites/20/2017/12/ONAP_CaseSolution_Architecture_120817_FNL.pdf
  - Section 6 talks about control loops

- Design -> Create -> Collect -> Analyze -> Detect -> Publish -> Respond.

- Control Loops are not just for ONAP services
  - Applicable to PNF and ONAP components themselves

- Control Loop Sub Committee Logistics
  - https://wiki.onap.org/display/DW/Control+Loop+Subcommittee

ONAP Platform Component Control Loop Responsibilities

- SDC – service design
  - Onboarding of DCAE micro services and DCAE templates into DCAE Inventory
  - Control Loop design via DCAE-DS
- DCAE – data collection, analytics and events
  - DCAE does the heavy lifting when it comes to collection, analysis and correlation of runtime KPI's, events, alarms etc.
  - DCAE micro services are policy-driven allowing an operations team to modify policies during runtime
  - DCAE micro services generate control loop events. They do NOT perform any actions
- Policy Platform
  - Supports design time definition of policies for DCAE Micro services and Control Loop events
  - Enforces Control Loop events
  - Supports runtime changes to Control Loop Policies for DCAE micro services and Control Loop operational policies
- SO/VF-C/APPC/etc.
  - Orchestrators and controller are instructed to perform actions via the Policy Platform
- CLAMP
  - Runtime monitoring of Control Loops

ONAP Control Loops – future roadmap

- Adaptive Policies and integration with Machine Learning/Artificial Intelligence
- Integrating Feedback Loops during runtime
- Closed Loop vs Open Loop – dynamically creating loops during runtime

- **Definition: Actor**: An actor is an component in the ONAP Platform that exposes a set of API's and their payloads such that a Control Loop designer can build a policy-driven set of actions to take when a Control Loop Event occurs

- **Gap**:
  - There is no standard way for custom actors to be integrated.
  - Costs a lot of development effort and hard coding in CLAMP/Policy.
  - Desire is to have a simple high level abstraction for service designers to create control loops.
  - Cannot programmatically utilize VNF vendor recommendations for control loops. eg. How to map that to an actor API

- **Proposal**: Inventory of Control Loop actors and their API payloads
  - Design Time Feature integrated in SDC
  - Default actors: SO, APPC and VF-C
  - Custom Actors:
    - Example: Ticketing systems, Q-chat robots, custom internal Controllers
  - Stretch goal: How to enable integration of VNF capabilities with actors
    - eg. vFW Use Case uses ModifyConfig APPC API to perform a REST call to the packet generator VNF
- Call for help to develop a Use Case for this functionality
  - Where does the inventory live?
  - How to model actors/

- **Gap**:
  - Current API's for SO, VF-C, APP-C are inconsistent and differ tremendously in complexity
  - Example: SO API's for scale out require multiple A&AI calls to pull together required information for the API
  - The amount of work required for developing Policy Templates to support is large due to API complexity
  - Inconsistency in that some API's are HTTP (SO, VF-C) while some are Dmaap-based (APPC)

- Proposal: Design a simple Control Loop Actor API so companies can easily design and build Control Loop Operational Policies

- Call for Help:
  - Can Dmaap be used?
  - Minimum details required to specify an action
    - Actor
    - Recipe
    - Payload
    - A&AI details

Bio

- Pamela Dragosh has 27 years experience in designing and building software platforms in AT&T Research. Her projects have ranged from speech recognition, text-to-speech, digital rights management, music encryption, big data, location-based services, software defined networking, and policy platform. Pam has open sourced several projects including XACML 3.0 Policy Engine, OpenAZ Apache, and is currently the Project Technical Lead in ONAP for the Policy Framework Project and Chair of Control Loop Sub Committee.