

CLAMP Project Proposal (5/11/17)

Project Name:

- Proposed name for the project: CLAMP
- Proposed name for the repository: clamp

Project description:

CLAMP is a platform for designing and managing control loops. It is used to design a closed loop, configure it with specific parameters for a particular network service, then deploying and undeploying it. Once deployed, the user can also update the loop with new parameters during runtime, as well as suspending and restarting it.

It interacts with other systems to deploy and execute the closed loop. For example, it pushes the control loop design to the SDC catalog, associating it with the VF resource. It requests from DCAE the instantiation of microservices to manage the closed loop flow. Further, it creates and updates multiple policies in the Policy Engine that define the closed loop flow.

The OpenCLAMP platform abstracts the details of these systems under the concept of a control loop model. The design of a control loop and its management is represented by a workflow in which all relevant system interactions take place. This is essential for a self-service model of creating and managing control loops, where no low-level user interaction with other components is required.

At a higher level, CLAMP is about supporting and managing the broad operational life cycle of VNFs/VMs and ultimately ONAP components itself. It will offer the ability to design, test, deploy and update control loop automation - both closed and open. Automating these functions would represent a significant saving on operational costs compared to traditional methods.

Scope

Functionality supported by a Control Loop:

A control loop does several functions:

1. Collect data about VNFs or VNF infrastructure; both fault and performance metrics
2. Use analytics to do:
 - a. root-cause analysis on faults
 - b. threshold of performance metrics
 - c. determining signatures of network conditions
 - d. ...
3. For open loop, send message describing condition to ticketing system for human intervention
4. For closed loop, execute one of many actions to remediate the network condition
 - a. restart vm
 - b. rebuilt vm
 - c. migrate vm
 - d. scale up/down resources
 - e. redirect traffic
 - f. ...
5. For closed loop, detect that a network condition has been corrected through our actions

CLAMP Functions on Control Loop:

1. **Designing the Control Loop:**
 - a. Model based (based on templates)
 - b. Re-usable designs
 - c. Choose from different analytics microservices to include in control loop (for instance, Threshold Crossing (TCA), String Matching, or Holmes (fault correlation))
2. **Configuring the Control Loop:**
 - a. VNF driven, future should be SERVICE driven
 - b. Defining threshold values...
3. **Deploying the Control Loop:**
 - a. Sending template towards DCAE, Policy, ...
4. **Lifecycle Management of the Control Loop:**
 - a. Stopping/re-starting/updates
 - b. Monitoring (visualization of the Loop)(future not in R1)
 - c. Trial mode, production mode, ... (future not in R1)
 - d. Activation (manually and/or triggered by a policy(future not in R1) and/or tied to the vnf startup(future not in R1))

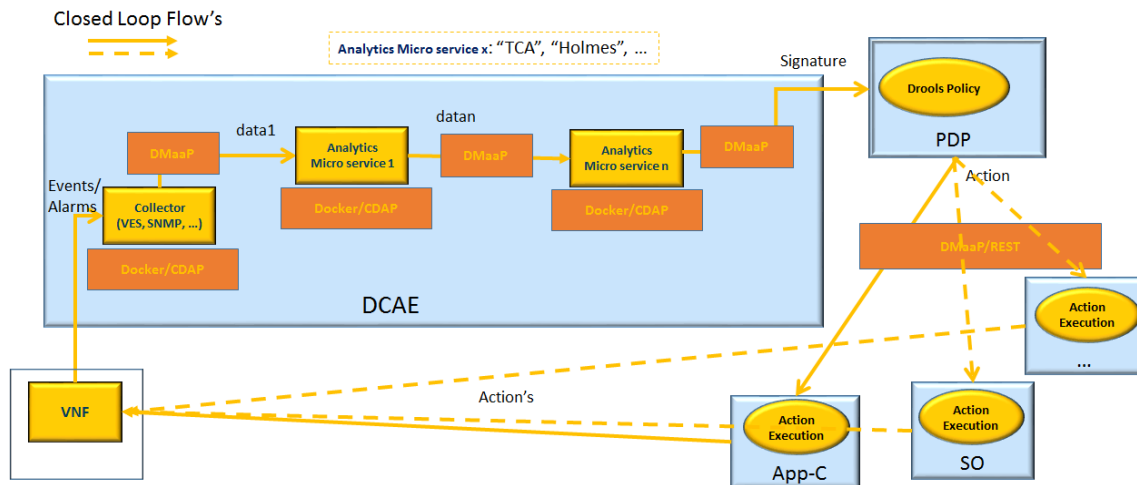
End to End Flows:

Since CLAMP is an end-to-end application, the project will, early on, develop documents describing detailed call-flows and message formats between ONAP components to support

control loop.

These documents will be developed for the three use cases: vFW/vDNS, vVoLTE, and vCPE. See description of vVoLTE use-case at the bottom of the page.

Here below you can find a diagram showing, generically, how a "runtime control loop flow" looks like:



Seed Code:

CLAMP will be based on seed code implementing the design tool and cockpit.

The current functionality of the CLAMP tool includes:

Design

- A set number of templates to define resources that are created and configured as part of a specific control loop.
- Interaction with Policy through APIs: Configuration of policies that define the control loop, directly from the CLAMP tool.
- Interaction with ASDC through APIs to push a fully configured control loop and distribute it to DCAE [represented as TOSCA template]

Runtime

- Interaction with DCAE and Policy through APIs: Life-cycle management of the control loop: deploy, stop, start, undeploy, update configurations

The rich functionality already available in CLAMP will put us in a good position

to implement the ONAP use cases, as described next.

Use-cases - Release 1 content :

Using existing seed code, CLAMP will support the **vFW/vDNS** ONAP Use case in R1.

vFW Control Loop design will consist of:

1. create the blueprint for DCAE to capture traffic level .
2. create the policy that will define threshold signature that DCAE will use.
 - a. DCAE team provides model to Policy team to built the Configuration policy
3. create the policy that will trigger traffic gen to lower traffic.

vDNS Control Loop design will consist of:

1. create the blueprint for DCAE to capture traffic level.
2. create the policy that will define threshold signature that DCAE will use.
 - a. DCAE team provides model to Policy team to built the Configuration policy
3. create the policy that will trigger SO to spun up a new vDNS vm.

For the **vVoLTE** and **vCPE** use-cases, CLAMP plans to support them, but we need more clarifications on those use-cases in order to know the features CLAMP will be able to implement, to support those two use-cases, in R1.

We have developed call flows to describe how these use-cases will be implemented, and require input from other teams. Below see the vVoLTE Call flows.

[CLAMP ONAP vVoLTE Use Case.docx](#)

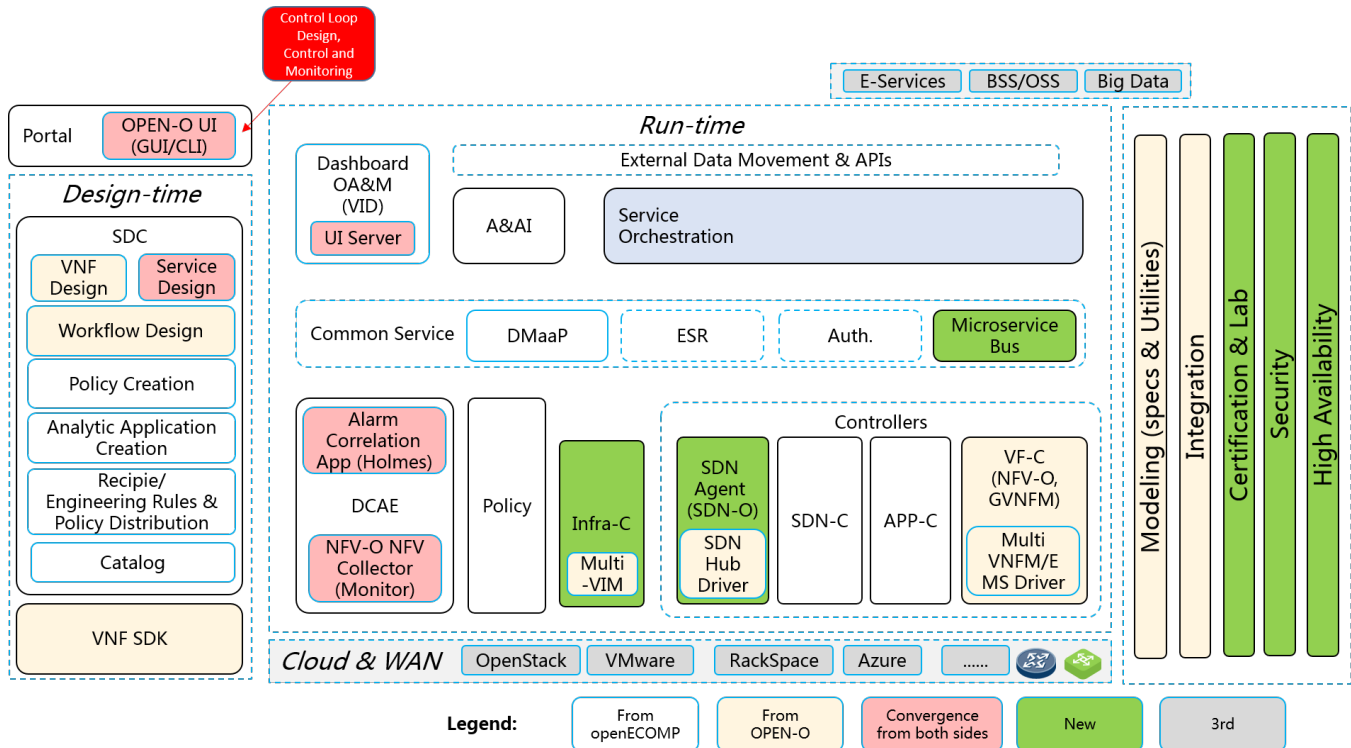
Interfaces:

The following interfaces will be implemented by other components but used for CLAMP. Therefore, CLAMP has these as dependencies:

- Query and update services in [SDC](#)
- Query and update control loop templates in [SDC](#) (future not in R1)
- [Policy](#) interfaces for creating/updating configuration and operational policies
- [DCAE](#) interfaces for deploying/undeploying control loops
- Runtime messaging interface between [DCAE](#) and [Policy](#)

Architecture Alignment

- Below we show how the CLAMP application fits into ONAP. The red figure below shows the CLAMP application components. There is a design portion and an operations component, which are both deployed within ONAP portal.

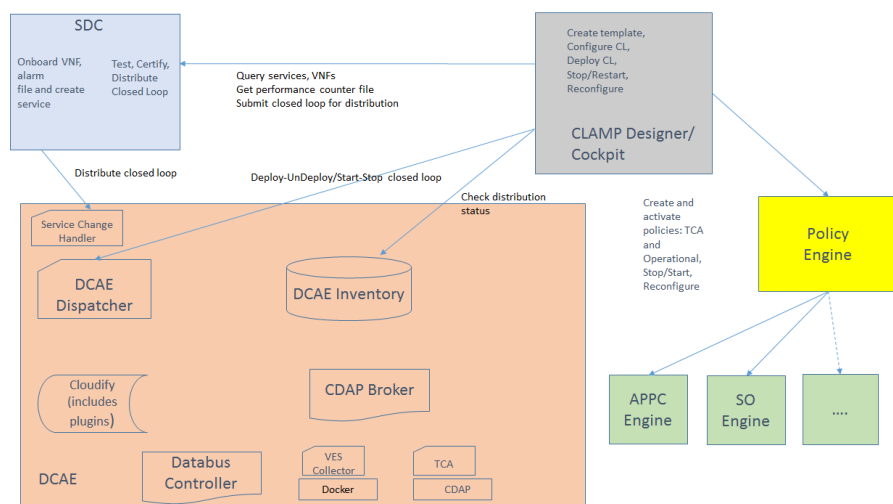


- **CLAMP** is separated in 2 areas, which are currently (in seed code) both supported by a single application:

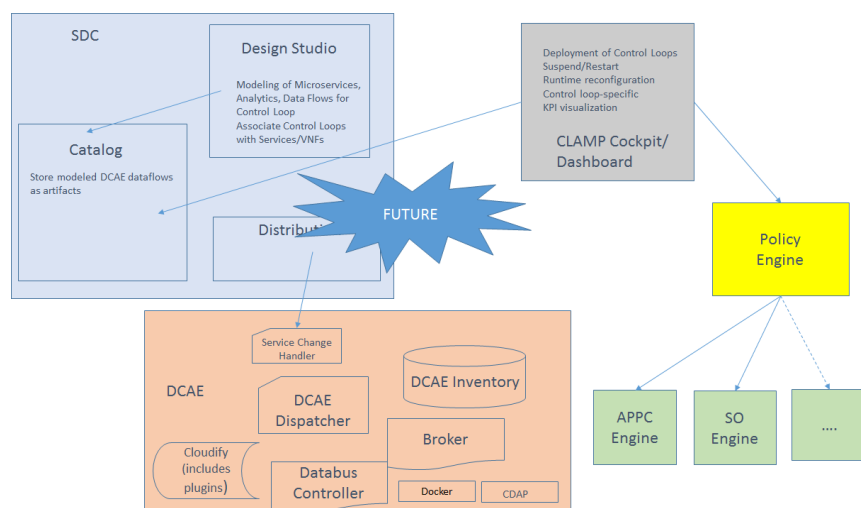
1. **Design Time** (Cockpit/UI to define the templates)
 - a. Templates are pushed to SDC. The template format is **TOSCA** blueprint, those blueprints will be pushed/provisioned, by [SDC](#), to [DCAE](#) orchestration engine.
 - b. policies (configuration and operational policies) are pushed/provisioned towards the Policy Component of ONAP. (those policies will be triggered by [DCAE](#) during Closed Loop operations).
 - i. The [DCAE](#) team needs to provide models to [Policy](#) team in order for the Configuration policy to be built.
2. **Run time** ([DCAE-Policy](#), grabbing events and triggering policies based actions)
 - a. In the first release of CLAMP, the triggering to deploy (and then effectively start the closed loop) a blueprint will be manual (via CLAMP cockpit) an automatic deployment based on an event will come in future release.
 - b. The CLAMP cockpit will support the following action at runtime:
 - i. start (start the provisioned Closed Loop on [DCAE](#))
 - ii. stop (stop a provisioned Closed loop on [DCAE](#))

- **CLAMP** depends on the following ONAP projects

- [SDC](#) : Rest based interface exposed by the SDC, Distribution of service to DCAE
- [DCAE](#) : Rest based interface exposed by DCAE, Common Controller Framework, DCAE microservices onboarded (TCA, Stringmatch, Holmes (optional))
- [Policy](#) : Rest based interface (the Policy team provide a "jar" to handle the communication), both XACML and Drools PDP, APIs to App-C /VF-C/SDN-C
- [DMaaP](#) : Message bus within DCAE and cross-ONAP



There is a plan to move the design part of CLAMP towards the "DCAE design Studio" inside SDC, but this **won't be in R1**, see below:



CLAMP depends upon the following open source projects:

- AJSC 6.0** (container framework based on Spring and developed by "AT&T Common platform" Team)
- AJSC-Camunda 6.0** (Camunda Workflow integration to AJSC container developed by "AT&T Common platform" Team)

Resources

•Primary Contact Person:

NGUEKO Gervais-Martial (AT&T - gn422w@intl.att.com)

Other Contact/Contributor Person (LAST NAME in capital followed by first name !!):

- SHACHAM Ron (AT&T - rshacham@research.att.com)
- JANA Rittwik (AT&T - rjana@research.att.com)
- CLOSSET Christophe (AT&T - cc697w@intl.att.com)
- CLOSE Pierre (AT&T - pc457b@intl.att.com)

- GAO Xue (AT&T - xg353y@intl.att.com)
- BHATNAGAR Aayush (Reliance Jio - aayush.bhatnagar@ril.com)
- NGUYEN Trung (Amdocs - Trung.Nguyen@amdocs.com)
- MIAO Xin (Fujitsu - xin.miao@us.fujitsu.com)
- MIRZA Yusuf (IBM - ymirza@ae.ibm.com)

Committers/Maintainers(LAST NAME in capital followed by first name !!):

- DETERME Sebastien (AT&T - sd378r@intl.att.com)
- HAUTOT Eddy (AT&T - eh552t@intl.att.com)
- Xinyuan Wang(ZTE - wang.xinyuan1@zte.com.cn)

Other Information's

- The proposal is coming from the AT&T CLAMP project. Before publishing the codebase as part of release 1, AT&T will make sure that all proprietary trademarks, logo's, etc... are removed.
- The code will also goes through Fossology , BlackDuck and other scan to ensure licensing issues are not present.

Key Project Facts

Project Name:

- JIRA project name: *CLAMP (suggestion to be decided)*
- JIRA project prefix: *CLAMP- (suggestion to be decided)*

Repo name:

- *org.onap.clamp (suggestion to be decided)*

Lifecycle State: incubation

Primary Contact: NGUEKO Gervais-Martial, AT&T

Project Lead: NGUEKO Gervais-Martial

mailing list tag [clamp] *(suggestion to be decided)*

Committers & Contributors see Resources section

*Link to TSC approval:

Link to approval of additional submitters:

Associated files

CLAMP Closed Loop for ONAP vCPE Use Case Release 1:



CLAMP Flows for...in ONAP v2.pptx

CLAMP Closed Loop for ONAP vVoLTE Use Case, Release 1:



CLAMP ONAP vVo... Use Case.docx