CLAMP R7 - Architecture Review (done with Architecture team the 28th July 2020)

1. **Project Overview**
   CLAMP is the place where you manage the runtime of the Control loop in ONAP:
   a. you can configure the parameters of the µS composing the Control Loop:
      i. this is achieved by creating/updating/deleting policies (configuration policies).
      ii. the above policies will be associated to the deployed µS
      iii. Associate a Control Loop to a Service
   b. you can configure the Operation to be taken by the Control Loop:
      i. this is achieved by creating/updating/deleting policies (operational policies).
   c. you can deploy/un-deploy Control Loop flow (blueprint) to DCAE.
   d. in separated/independent Docker containers, you have the "Control Loop Dashboard" which is an ELK stack dedicated to display DMAAP messages related to runtime control loop messages (from DCAE/Policy/APPC).

2. **New component capabilities for Guilin, i.e. the functional enhancements**
   a. None (waiting finalization of DCAE-MOD interface design to continue Self Serve Control-Loop)

3. **New or modified interfaces**
   a. None.

4. **Interface naming**
   CLAMP supports the following interfaces:
   a. **Control Loop Life Cycle Management** User interface (LCM UI) for, refer to CLAMPE1 in architecture description (ARC CLAMP Component Description - Guilin and ARC CLAMP Component Description - Guilin (R7) Release (New)):
      i. selecting the Control Loop flow.
      ii. selecting the Service/VNF to be associated with the Flow
      iii. entering configuration policy parameter.
      iv. entering operational policy parameter.
      v. manage lifecycle of DCAE blueprint (Control Loop flow).
   b. **Control Loop dashboard** User Interface based on "Kibana" (ELK stack, completely independent from LCM UI), refer to CLAMPE2 in architecture description (ARC CLAMP Component Description - Guilin and ARC CLAMP Component Description - Guilin (R7) Release (New)).

5. **Reference to the interfaces**
   a. for the UI see readthedocs (obviously still under development for Dublin): CLAMP latest user guide
   b. internal interface are available via swagger: clamp swagger pdf

6. **What are the system limits**
   a. none so far, CLAMP is not a in the path of runtime call, so it is not heavily hit by massive amount of call. So auto scaling is not really required but could be added.
   b. DB redundancy/HA relies on kubernetes and persistent volume. mariaDB cluster can be a future improvement
   c. Application redundancy/HA relies on kubernetes.

7. **Involved use cases, architectural capabilities or functional requirements**
   a. scaling use cases (Test only for CLAMP)
   b. Self Serve Control Loop: Self Serve Control Loops v2 (design only in Guilin for CLAMP)
   c. Support of Native Policy (Test Only for CLAMP)
   d. Support of Policy Filtering (Test Only for CLAMP)

8. **Listing of new or impacted models used by the project (for information only)**
   a. None.