



Casablanca Release Developer Forum

ONAP Beijing DCAE operational perspective

Pawel Pawlak

June 22, 2018

Agenda

- Objective & scope
- DCAE architecture and its evolution
- Operational assessment criteria
- Conclusion

DCAE assessment goals

- The goal is to perform analysis and assessment of ONAP DCAE from operational perspective.
- Evaluation of working effectiveness and suitability of a system through test methods aimed at **identification of defects, gaps, areas of risk**, measurement of the adequacy of the output, and **assessment of the reliability of the operations**.

Source: <http://www.businessdictionary.com/definition/operational-assessment.html>

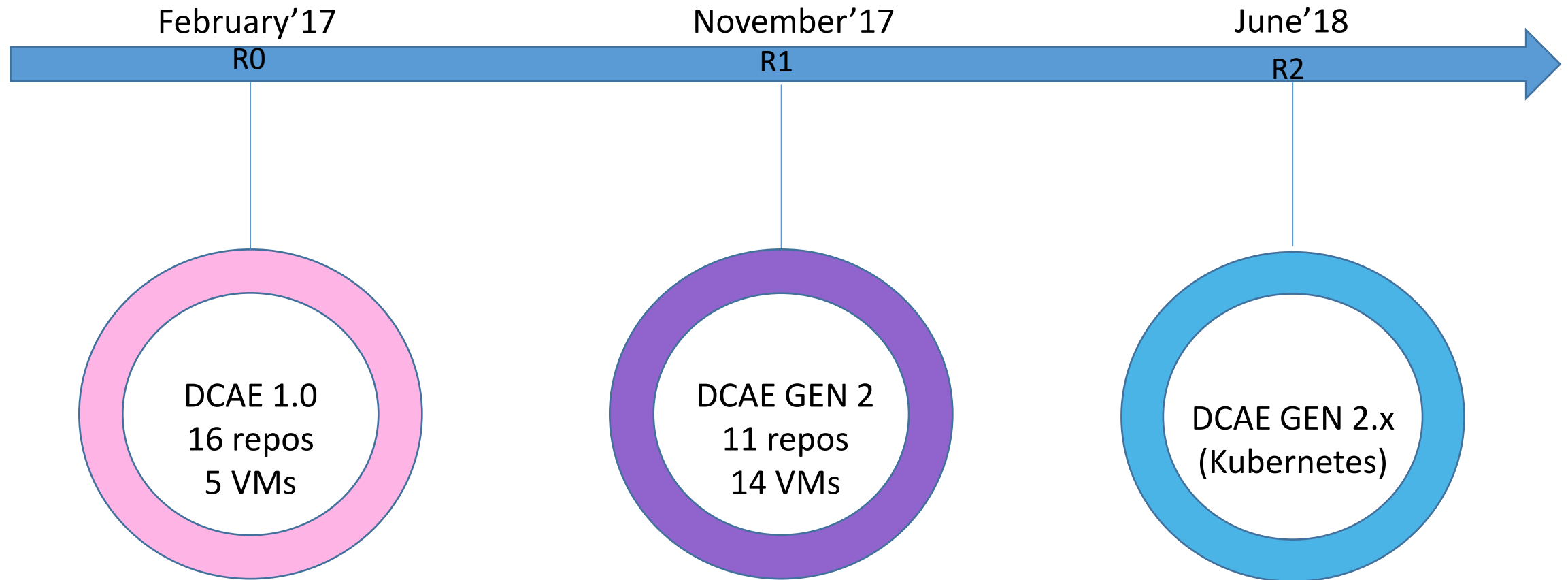
Alarms types – operational perspective

- „VM alarms, OpenStack alarms, what exactly and how to measure, what CSFs/KPIs?”
- „What alarms we can measure from VNFs/VNFCs of VFs?”
- „What alarms we can measure from physical elements (e.g. infra)?”
- „In general we should be able to measure status of all components from all layers but also statuses and utilization of compute network and storage resources...”
- DCAE currently does not monitor other ONAP components but is flexible enough to do so if needed- in R3 we shall expect more collector and analytics to include monitoring of underlying platform components (e.g. OpenStack or Contrail data collector could be used)

DCAE function

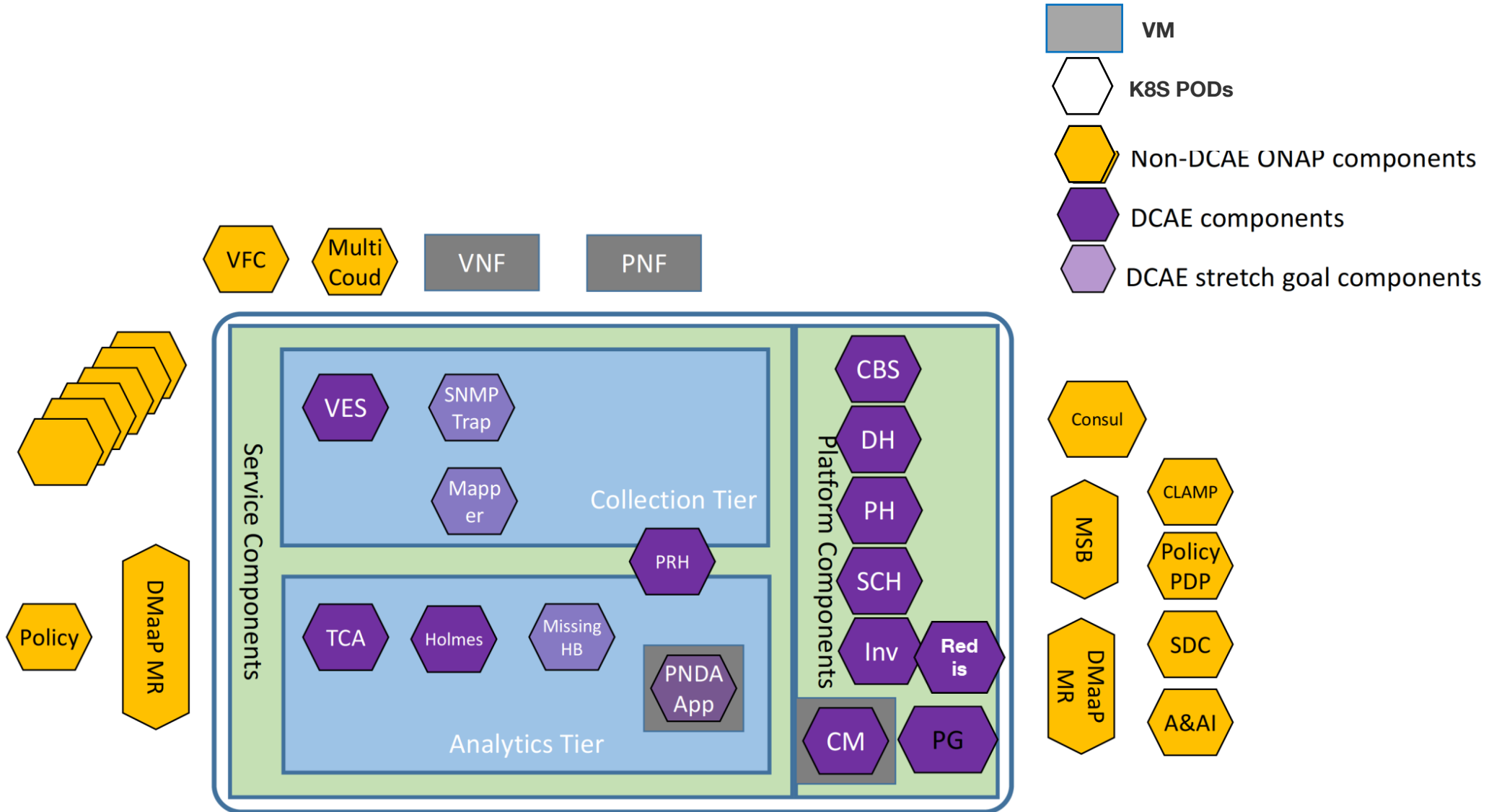
- DCAE is the key ONAP subsystem that supports closed loop control and higher level correlation for business and operations activities
 - DCAE collects performance, usage and configuration data;
 - Provides computation of analytics;
 - Aids in troubleshooting and management;
 - Publishes event, data, and analytics to the rest of the ONAP system for FCAPS functionality.

DCAE platform evolution

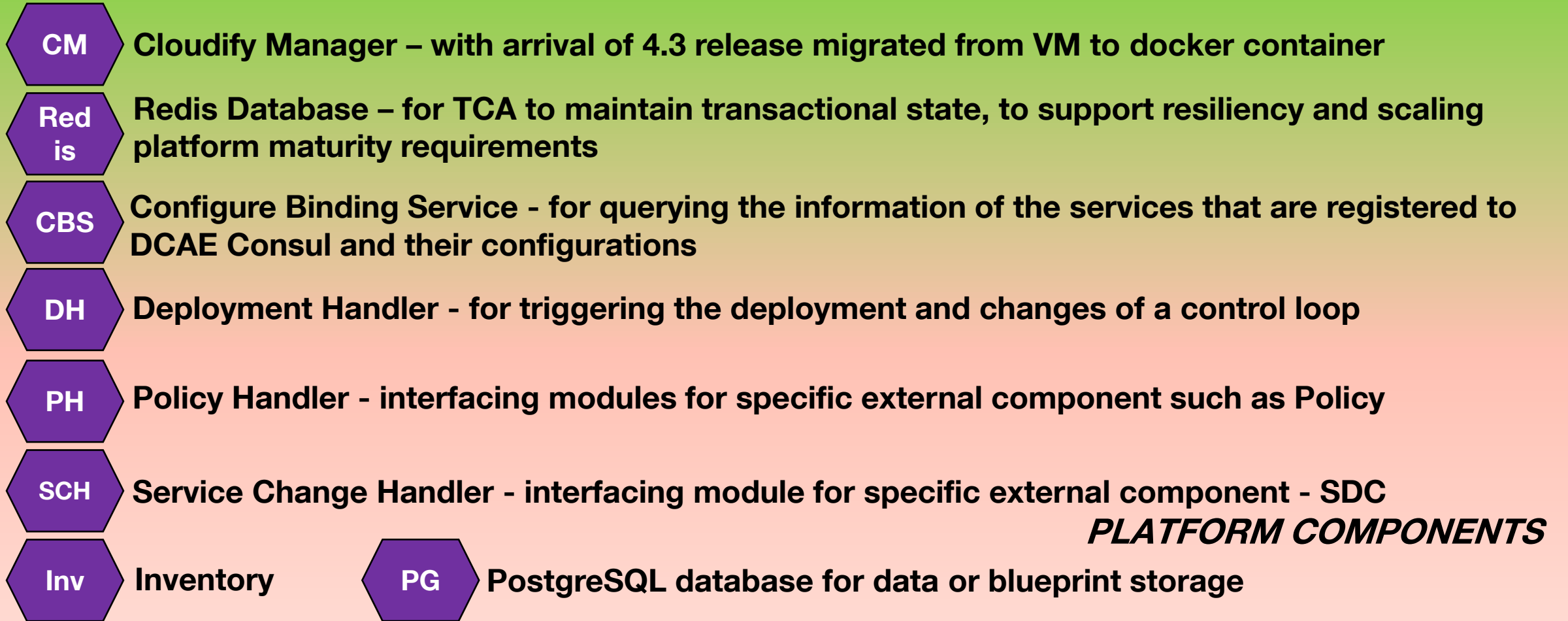


Infra requirements initially very high with R1 (CDAP with Hadoop cluster) – but going down due to design assumptions change (R2 use cases focus don't require high capacity analytics)

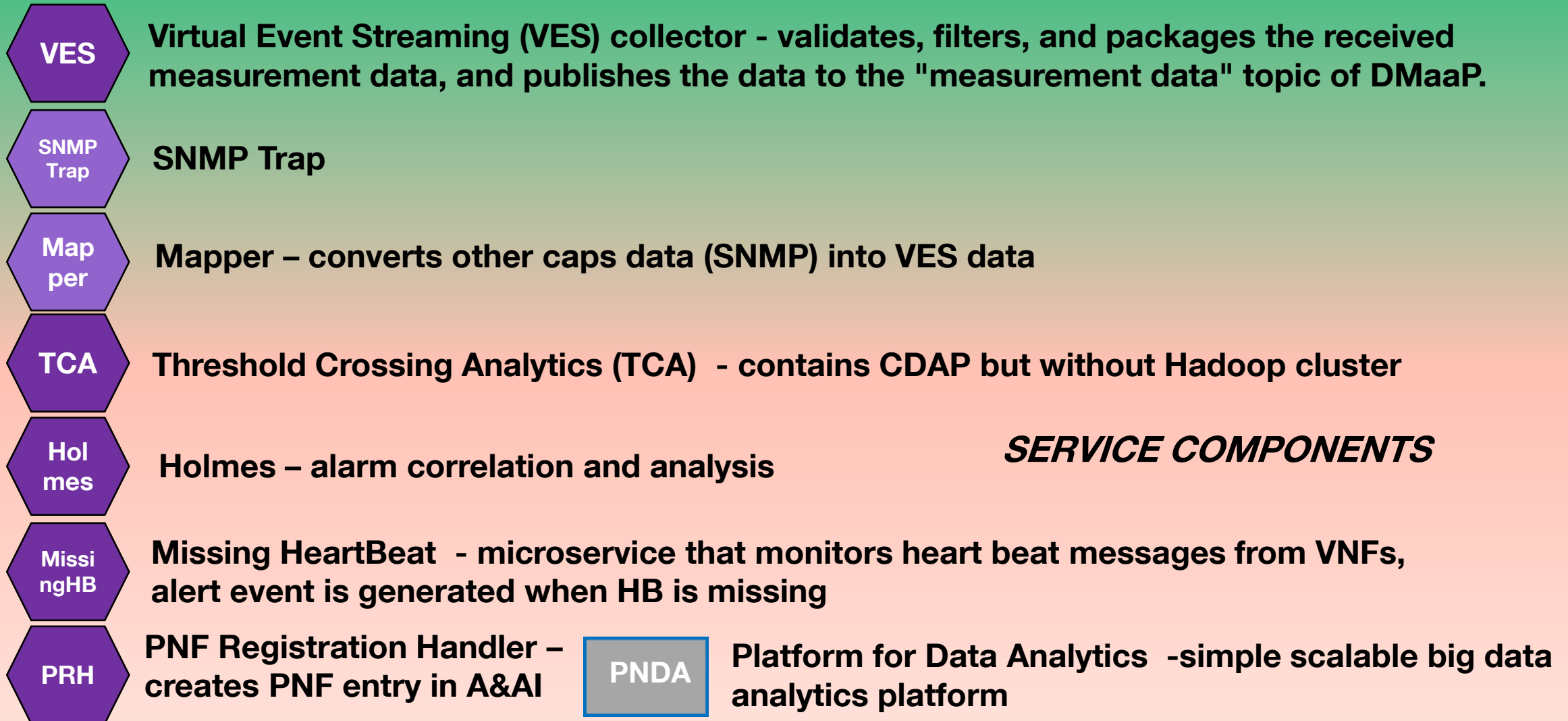
DCAE R2 architecture



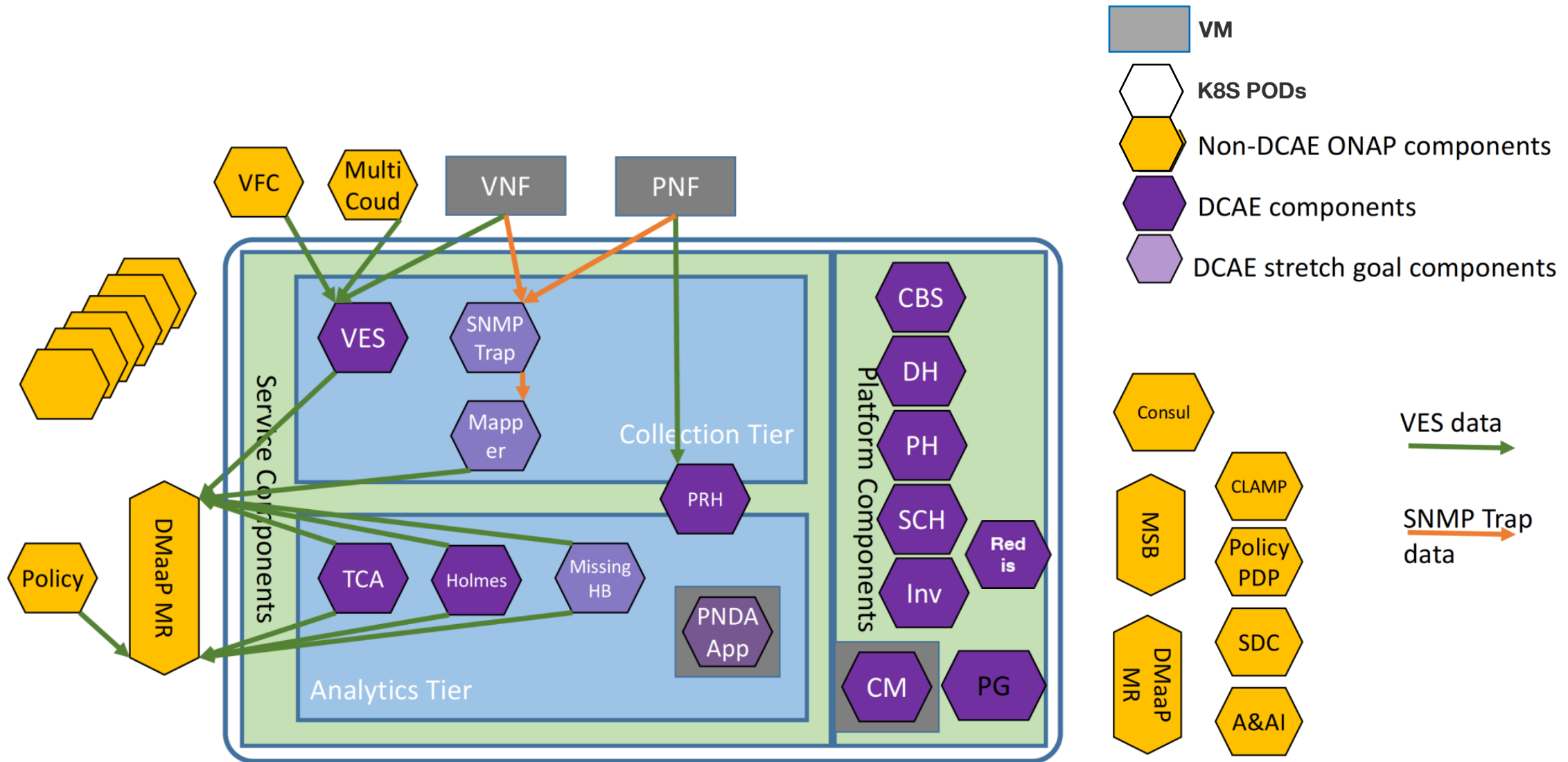
DCAE R2 architecture



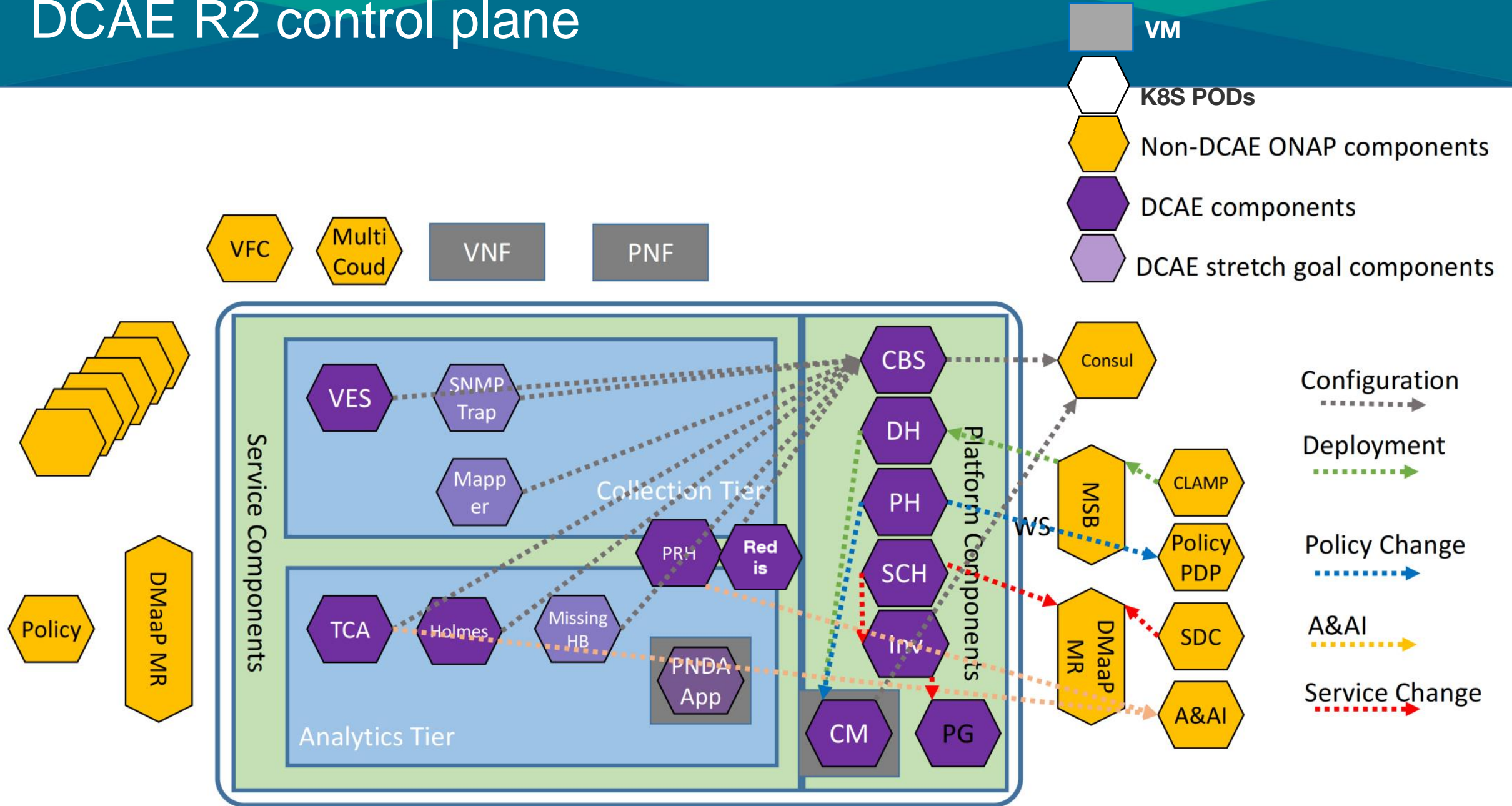
DCAE R2 architecture



DCAE R2 data plane



DCAE R2 control plane



R2 DCAE deployment – high level view

- Primary deployment mechanism becomes Kubernetes based (and not HEAT based as in R1, however HEAT is still supported) – under OOM team responsibility – HELM chart (used to specify how each POD and each service should be specified and deployed) is used for packaging, configuration and templating.
- For DCAE service and deployment components are more tight with control loops or organization operations time business requirements comparing with other ONAP components.

R2 – DCAE deployment process 1/2

Phase 1: Bootstrap process starts from ONAP initialization (with HELM chart) – 3 components are deployed this way:

- Cloudfify Manager
- Redis DB Cluster
- Bootstrapper

Phase 2: Deployment of remaining components by Cloudfify Manager called by Bootstrapper (has ONAP wide configuration parameters from HELM chart):

- Deployment Handler
- Policy Handler
- Service Change Handler
- Inventory API
- Config Binding Service
- Postgres Cluster

R2 – DCAE deployment process 2/2

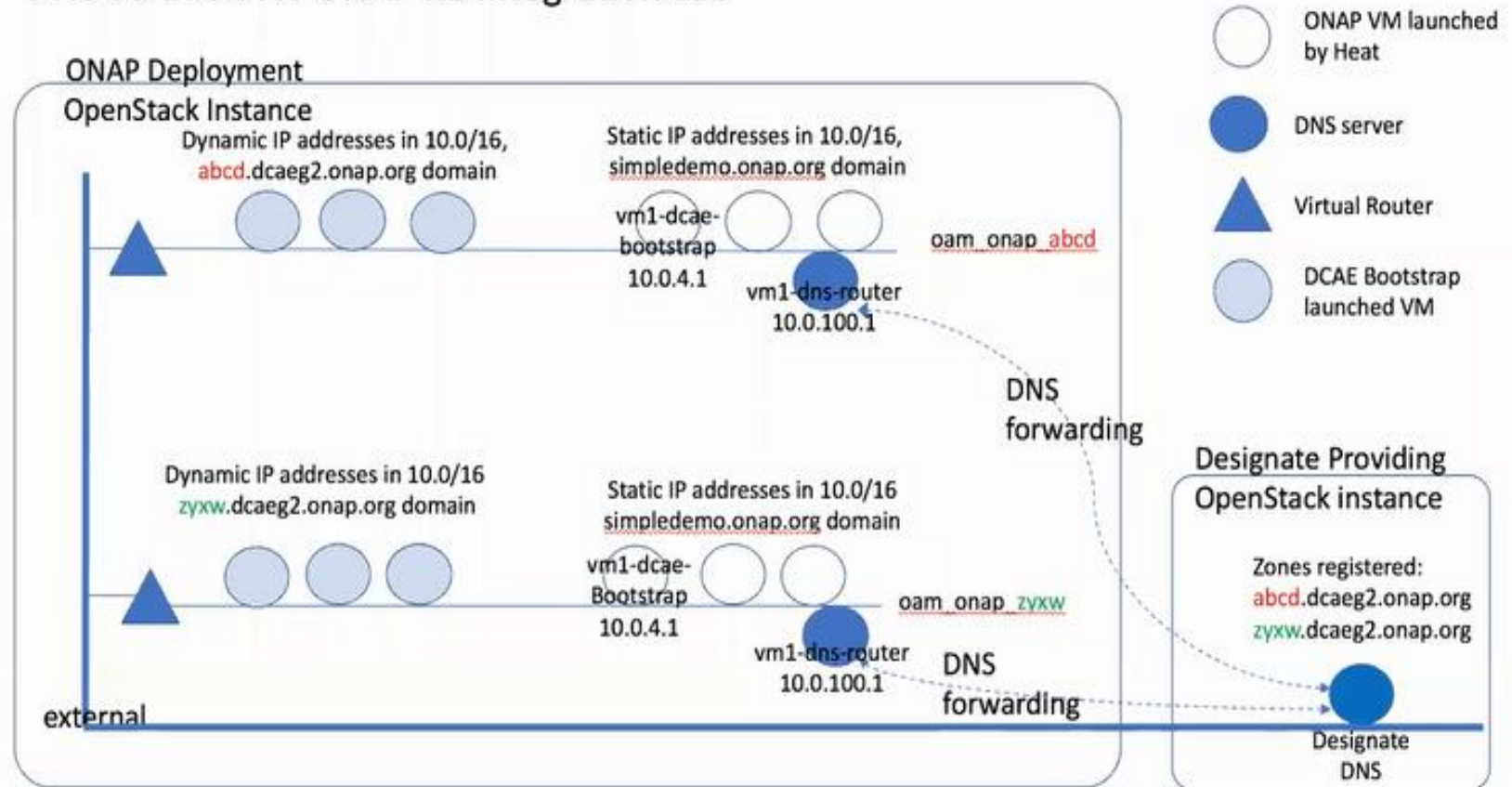
Phase 3: Deployment of remaining components that could be done in several ways:

- Through Cloudfify Manager using blueprints
- For more static things by Bootstrapper (example: PNF Registration Handler)
- For more dynamic things by Deployment Handler (could be triggered by CLAMP)
- Components deployed in Phase 3:
 - VES collector
 - TCA analytics (as docker)
 - Holmes rule management
 - Holmes engine management
 - PNF Registration Handler
- And not actively engaged in user stories in R2: Mapper, SNMP Trap collector and Missing Heart Beat

DCAE DNS solution in R1, and R2 (Kubernetes DNS proxy)

- **R1 - the need to use designate DNS to assure that both groups (with static and dynamic IP addresses) can communicate with each other using host names**
- **R2 – no more need for designate as now OOM based, everything is Kubernetes cluster**

DNS Solution for ONAP R1 Integration Lab



DCAE operational criteria 1/2

- **Resiliency**

- Mainly met by moving to Kubernetes resilience support in R2 - 2 failures observed during the OpenLab R1 usage, not measured – to be addressed how to measure it. Hadoop cluster is very fragile!

- **Scalability**

- Stateless platform and service components are individually scalable by scaling Kubernetes ReplicaSet (Configure Binding Service, Deployment Handler, Policy Handler, Service Change Handler, Inventory, VES, PNF Registration Handler) – in R2 does not scale automatically, on OOM level it is defined how the scaling to be done.
- Approaches for stateful components by:
 - pushing states to scalable stores: ONAP Consul cluster, Postgre Database,
 - pushing states to persistent volume (assuming the underlying technology is cross cluster),
 - individual scalability approach – VES - TCA scalability can be supported by scaling flows (collector - analytics pairs), - each flow is associated by its own DMaaP topic.

- **Usability**

- Solution is complex and not easy to understand, documentation is in general available but limited, but more guidelines would be appreciated to facilitate learning process

- **Manageability**

- Event and Error Logging Framework
- Filebeat sidecar container packed with function container for shipping logs to centralized Elasticsearch, Logstash, and Kibana (ELK) stack

DCAE operational criteria 2/2

- **Code security**

- DCAE has passed CLM scanning – several items needed further exchanges on DCAE exposure, but false positives as library usage nit in part assumed as risky.

- **Access management**

- As target with AAF component – but today nothing implemented apart from login/password per portal access, for R3 integration with AAF planned (2 ways: certificates management but also provisioning of DMaaP topics – who can publish, subscribe to the topic).

- **Performance and stability**

- Partial implementation of identified methods – R2 stability to be proven

- **Easiness to integrate with other components**

- Multiple APIs available (VES, Deployment Handler, Config Binding, Internal APIs for inventory)
- Both VNF and PNF support – only VES format supported

- **Easiness to deploy**

- Release 2 decreases the required resources to deploy DCAE
- Automatic deployment – no more manual tasks
- Deployment automation options simplify the process and gives flexibility
- Introduction of PNDA (but not yet in R2) is an option to replace CDAP that had high infra requirements,
- multisite deployment in a sense of multisite redundancy but also DCAE supports hierarchical or multi-tiers. Edge sites communication with central site – data communication published to DMaaP. R3 target – use case edge automation

- **Easiness to maintain**

- Although there is simple components monitoring assured by DCAE Consul, troubleshooting guidelines don't exist, so additional effort is needed in this area, R2 addresses this part

Summary

- **DCAE component plays an important role in overall ONAP platform acting as central point for Data Collection, Analytics, and Events generation in ONAP.**
- **The evolution of DCAE in Beijing release gives perspective for this platform to be mature enough in R3 which is to simplify the deployment (FLINK – as part of use case does not work very well with DMaaP, Prometheus, Data File Collector, 3GPP PM Mapper support and further enhancements) or R4 for operational deployment.**
- **Additional effort should be taken in DCAE solution operational troubleshooting – creating guidelines would be helpful.**
- **DCAE integration with an existing OSS environment (OCEANE, GNOC tools) in Orange**
- **Knowledge transfer and trainings must be considered for operational teams to understand DCAE architecture, TOSCA model, Kubernetes, Openstack, HEAT template and HELM chart.**
- **Availability of Sandbox, OpenLab for Ops is crucial to enhance discovery capabilities and knowledge transfer efficiency.**



Thank you