Ericsson
Software
Technology

OSC NONRTRIC
rApp Manager

NONRTRIC Team

rApp Manager

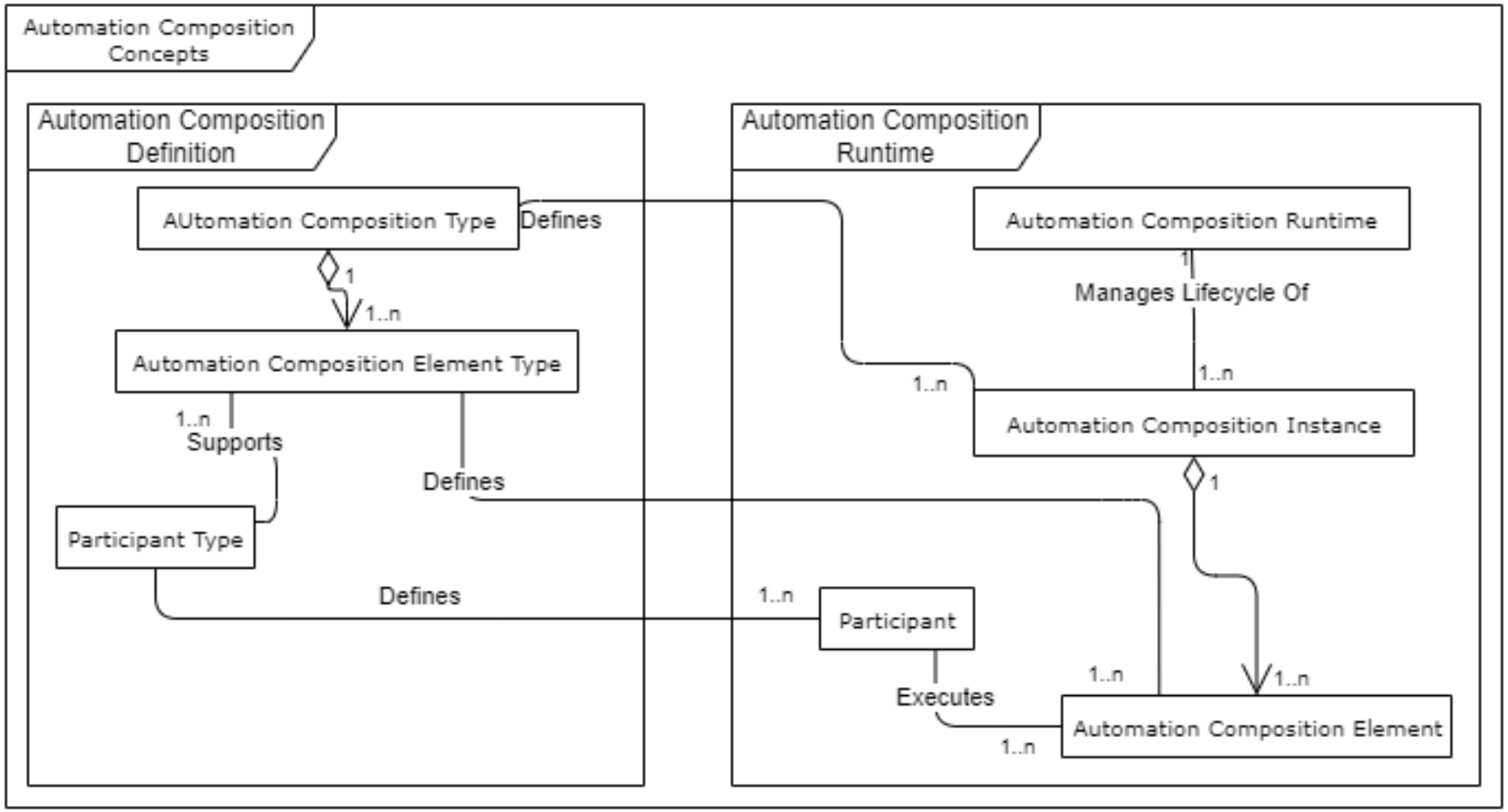
- It manages the lifecycle of rApp.
- Uses ONAP ACM as backend.
- rApp definition uses an ASD package as an example.
- ASD package contains the details required to create and integrate the required services/components.

Assumptions

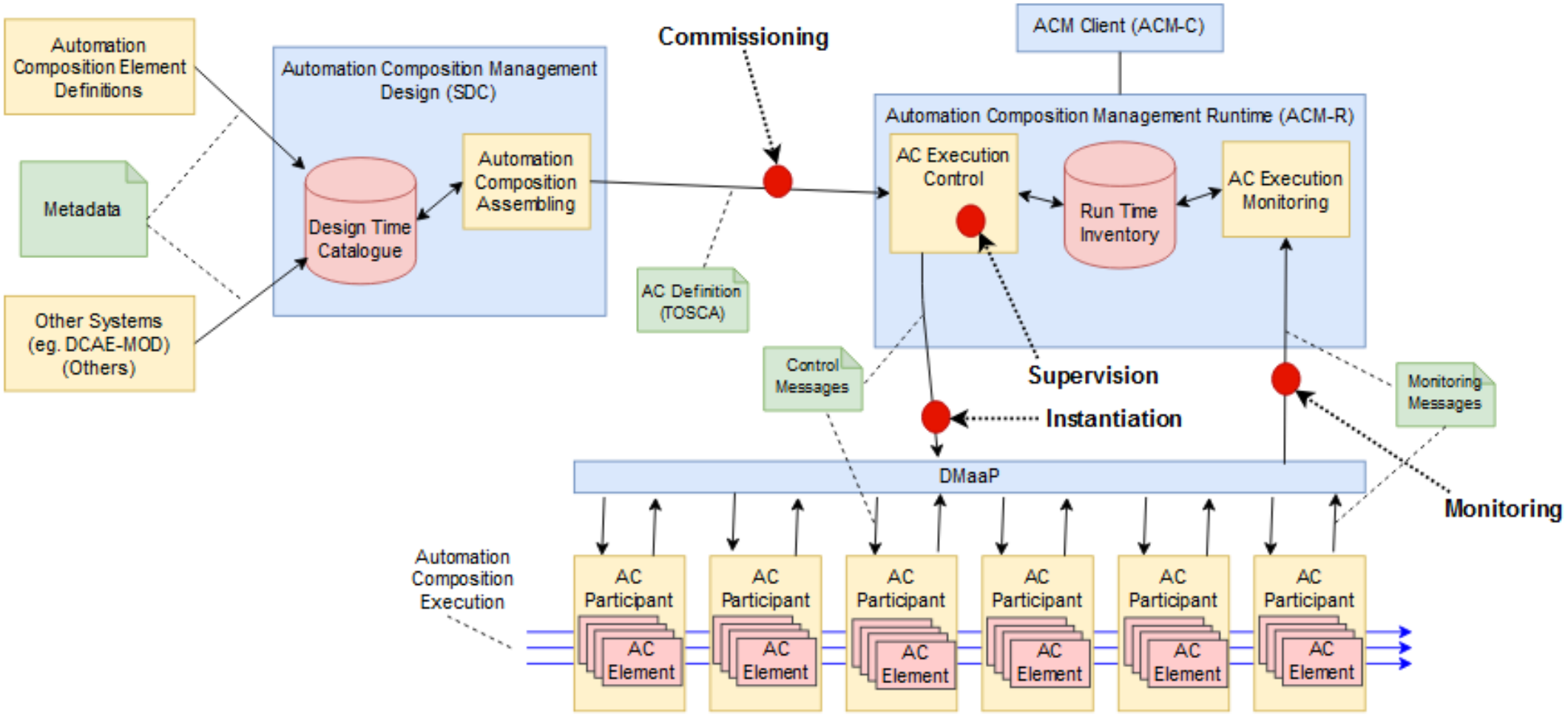
- Each ASD package contains only one rApp.
  - One rApp can have any number of rApp instances
- ASD package do not contain any kubeconfig file.
- ASD package contains the configuration required for ACM and SME.
- ACM definition get lifecycle managed during priming/depriming of the rApp.
- rApp Manager gets registered as AMF and lifecycle managed as part of start/stop of the application.
- ASD package contains the SME Provider/Service API/Invoker definitions.
- ASD package need to have one ACM composition definition.

rApps and ACM (Automation Composition Mgmt.)

- An ‘App’ is a ‘bundle of stuff’ composed together, not just a micro-service.
- ONAP ACM-R is a manager that life-cycle manages a bundle of other parts/stuff (a single 'Composition') together, rather than a user managing the parts individually.
- It takes a particular format of package to describe/package all the parts as a single 'Automation Composition’.
- Here 'parts' are called 'Automation Composition Elements' and can be microservices, workflows, helm charts (services, configurations, etc ...)
- Each 'Automation Composition Elements' *type* requires a plug-in (ACM 'Participant’) to handle parts of that type in each composition
- ACM Runtime delegates/orchestrates support for each 'Automation Composition Element' to appropriate participant.
- rApp is very similar to an 'Automation Composition’
    - rApp may have 0...n microservice, and 0..n other 'AC Elements'

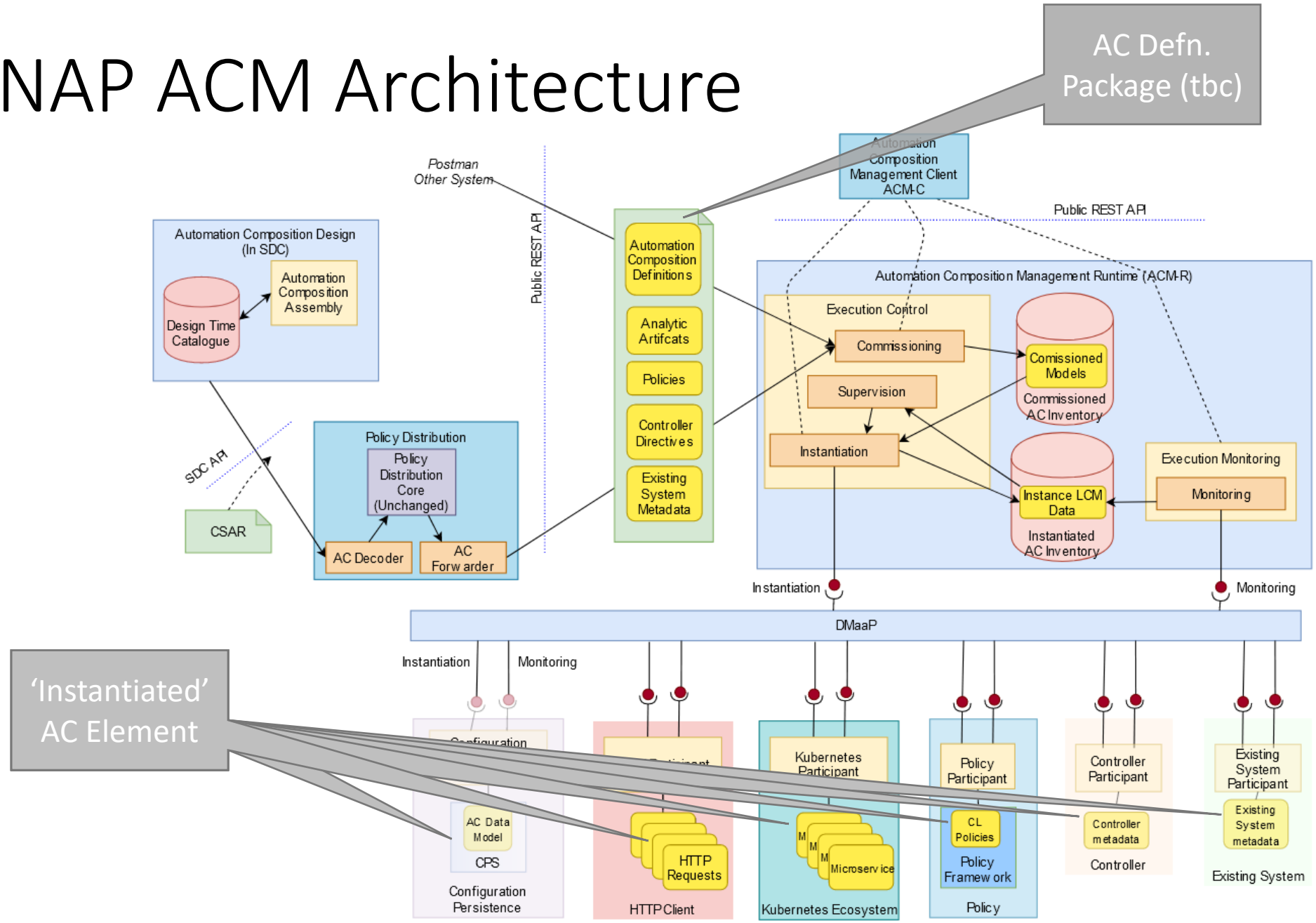


ONAP ACM Architecture



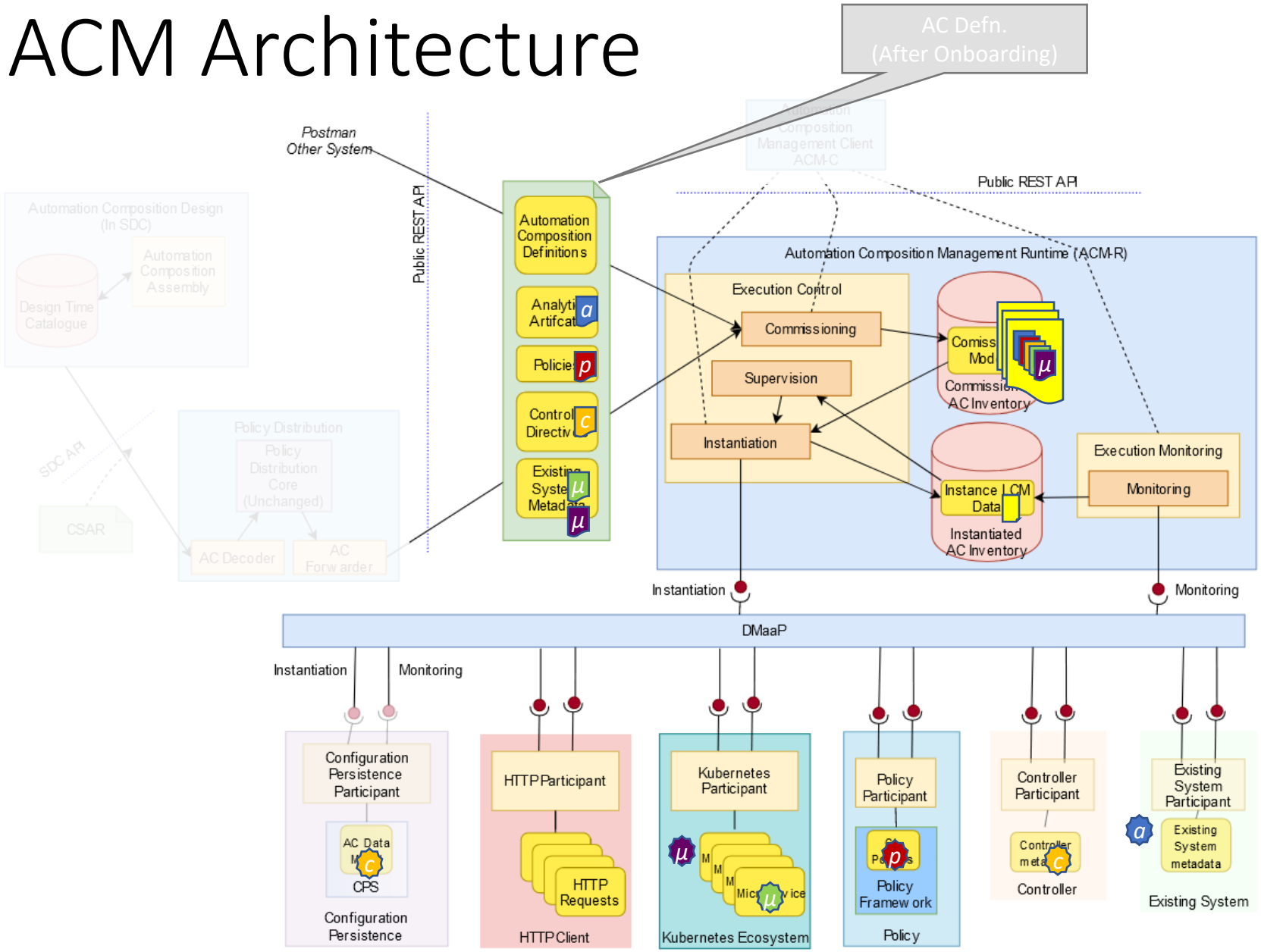
ONAP ACM: https://wiki.onap.org/display/DW/Automation+Composition+Management%3A+Architecture+and+Design

ONAP ACM Architecture



ONAP ACM: https://wiki.onap.org/display/DW/Automation+Composition+Management%3A+Architecture+and+Design

ONAP ACM Architecture



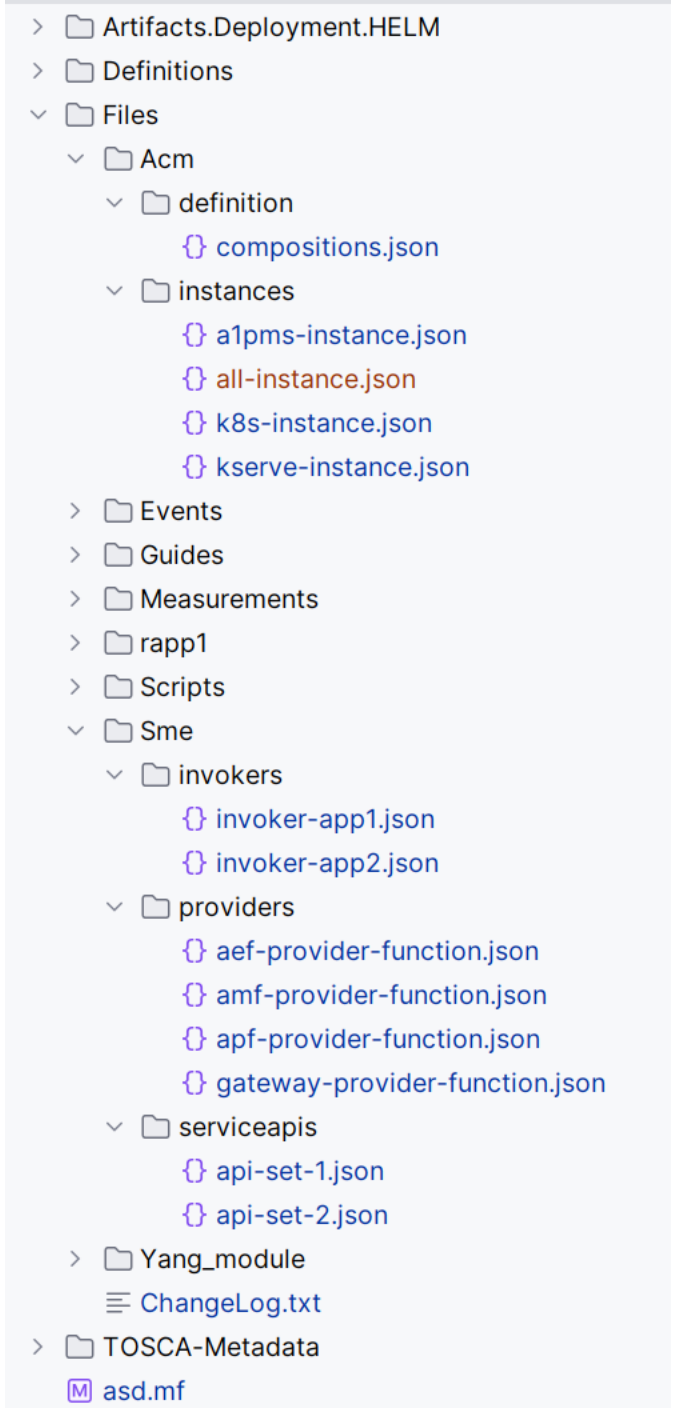
ONAP ACM: https://wiki.onap.org/display/DW/Automation+Composition+Management%3A+Architecture+and+Design

SME (Service Management and Exposure)

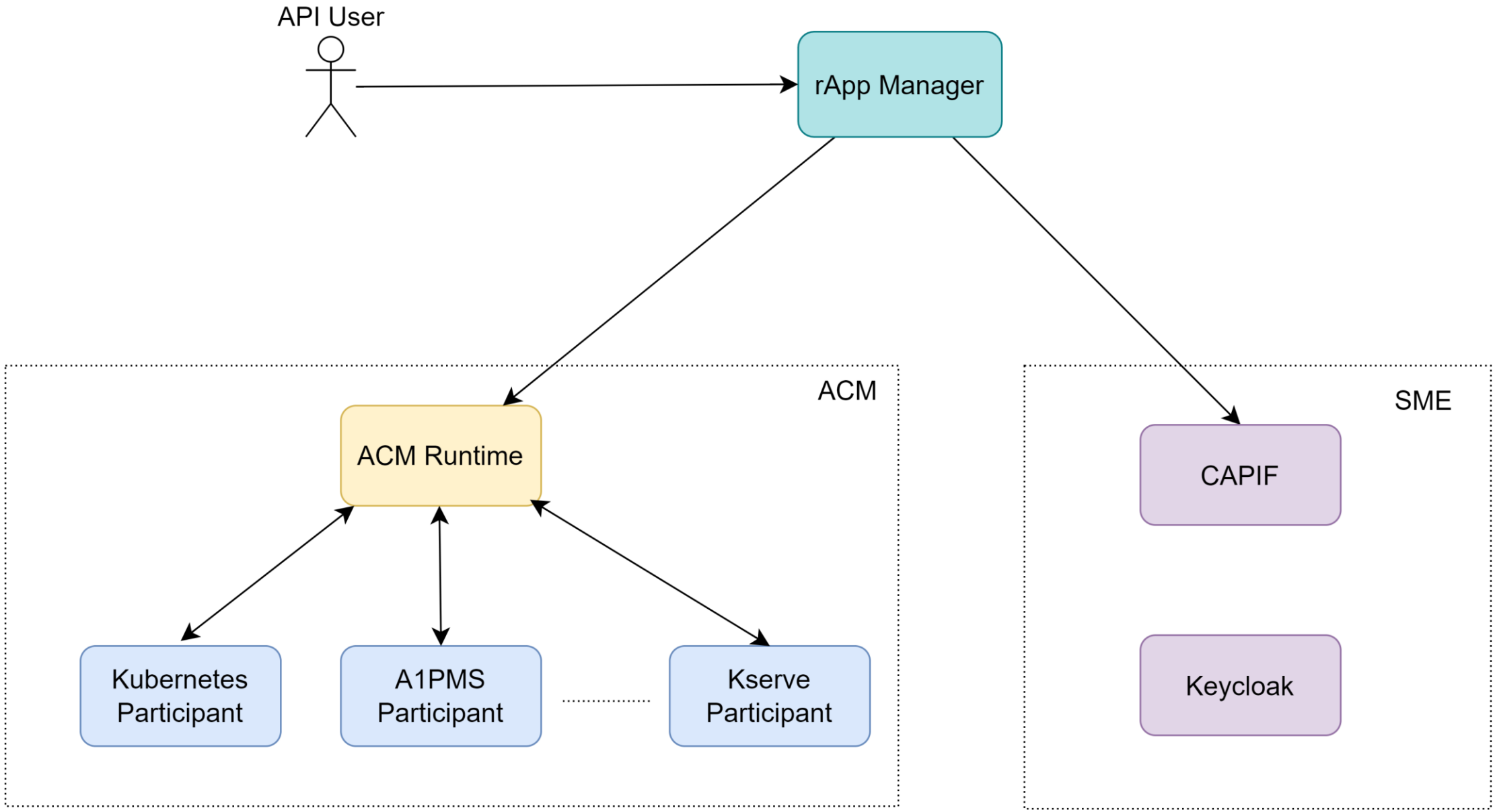
- Provider
  - AEF
    - It is the provider of the service APIs and is also the service communication entry point of the service API to the API invokers. Provides access control, logging, charging, provides authentication and authorization support.
  - APF
    - It is responsible for the capability to publish the service API information of the API provider to the CAPIF core functions to enable the discovery of APIs by the API invoker.
  - AMF
    - It is the entity which registers and maintains registration information of the API provider domain functions.
- Service Api
  - It is the list of API’s exposed by the provider.
- Invoker
  - It is an entity which invokes the service API’s.

rApp Package Definition

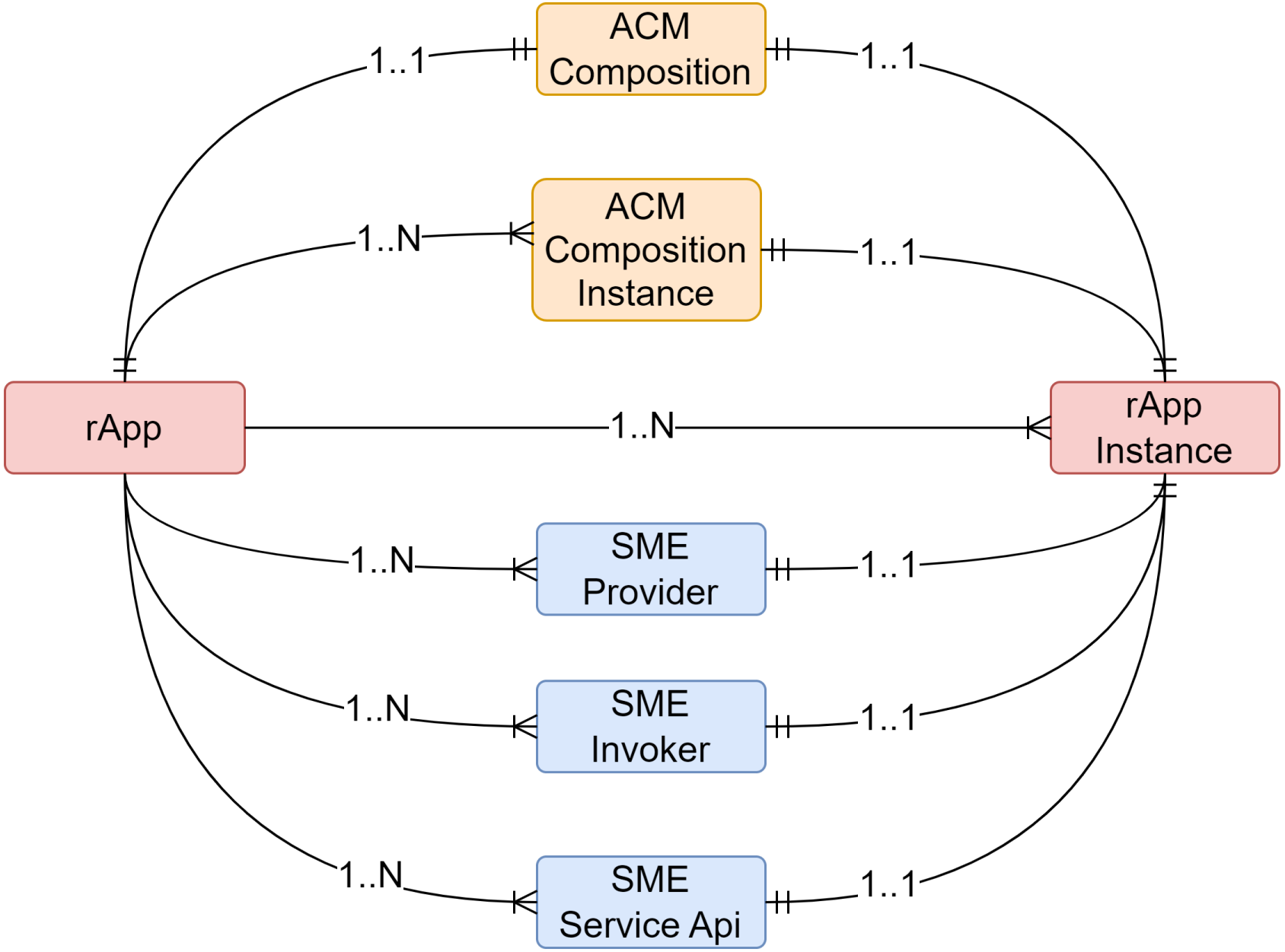
(This serves as a prototype and is prone to changes)



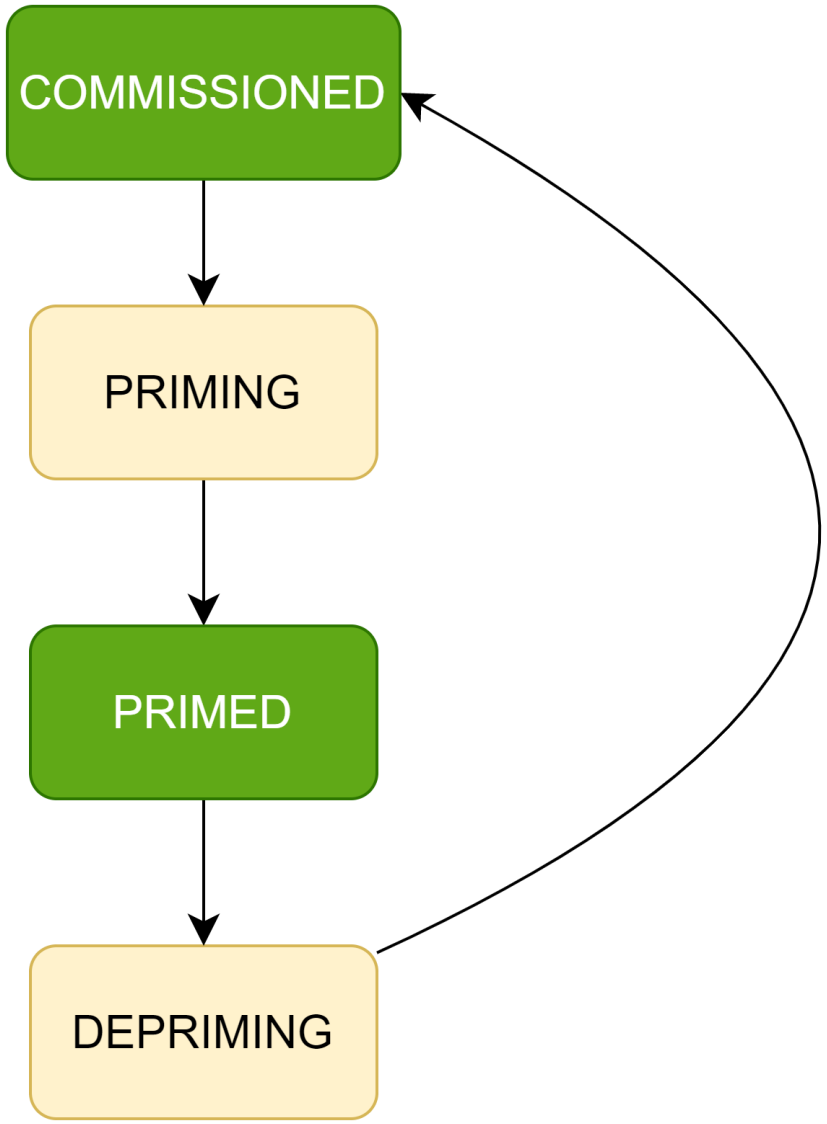
Architecture



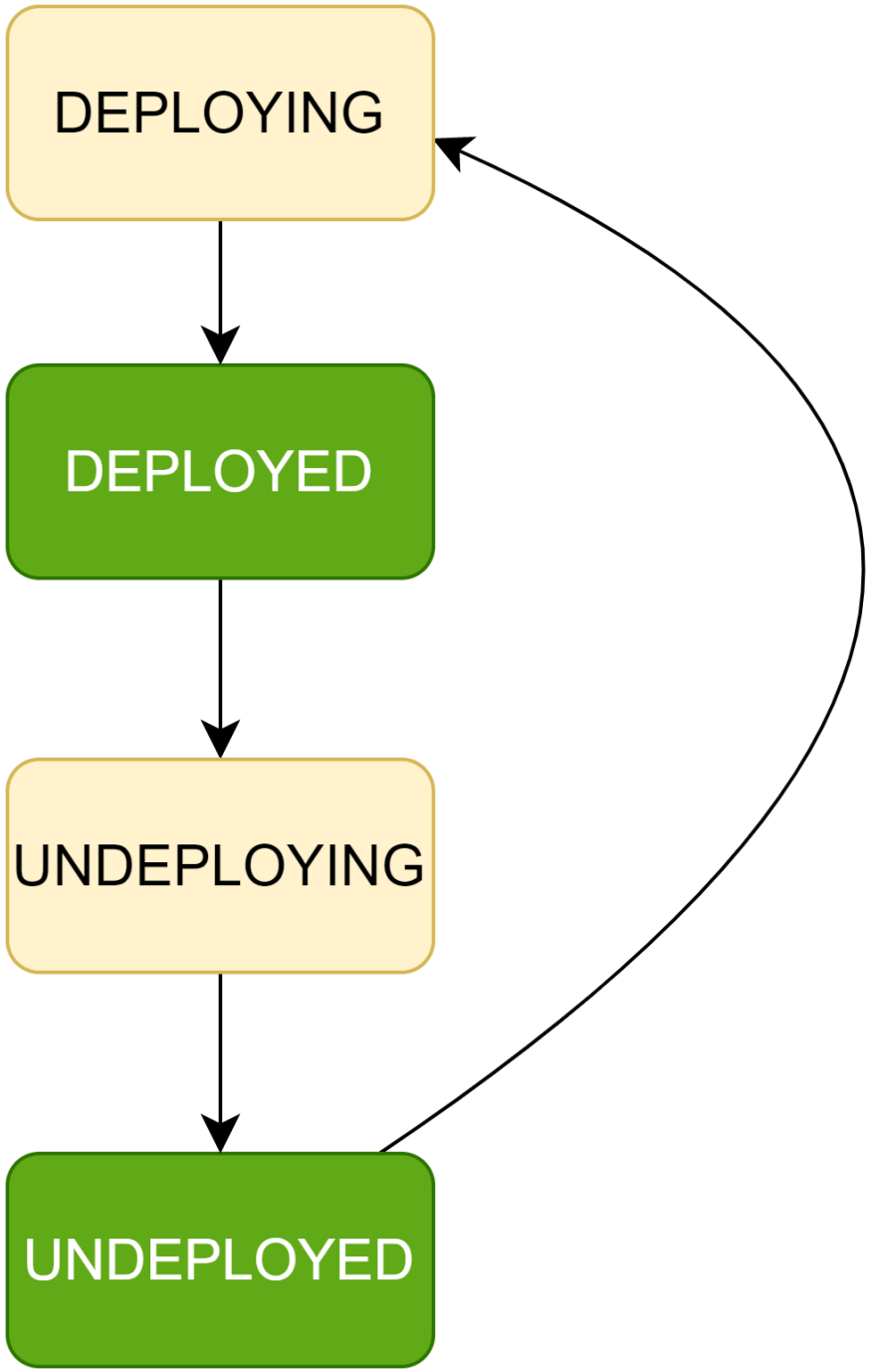
rApp Entity Relationship



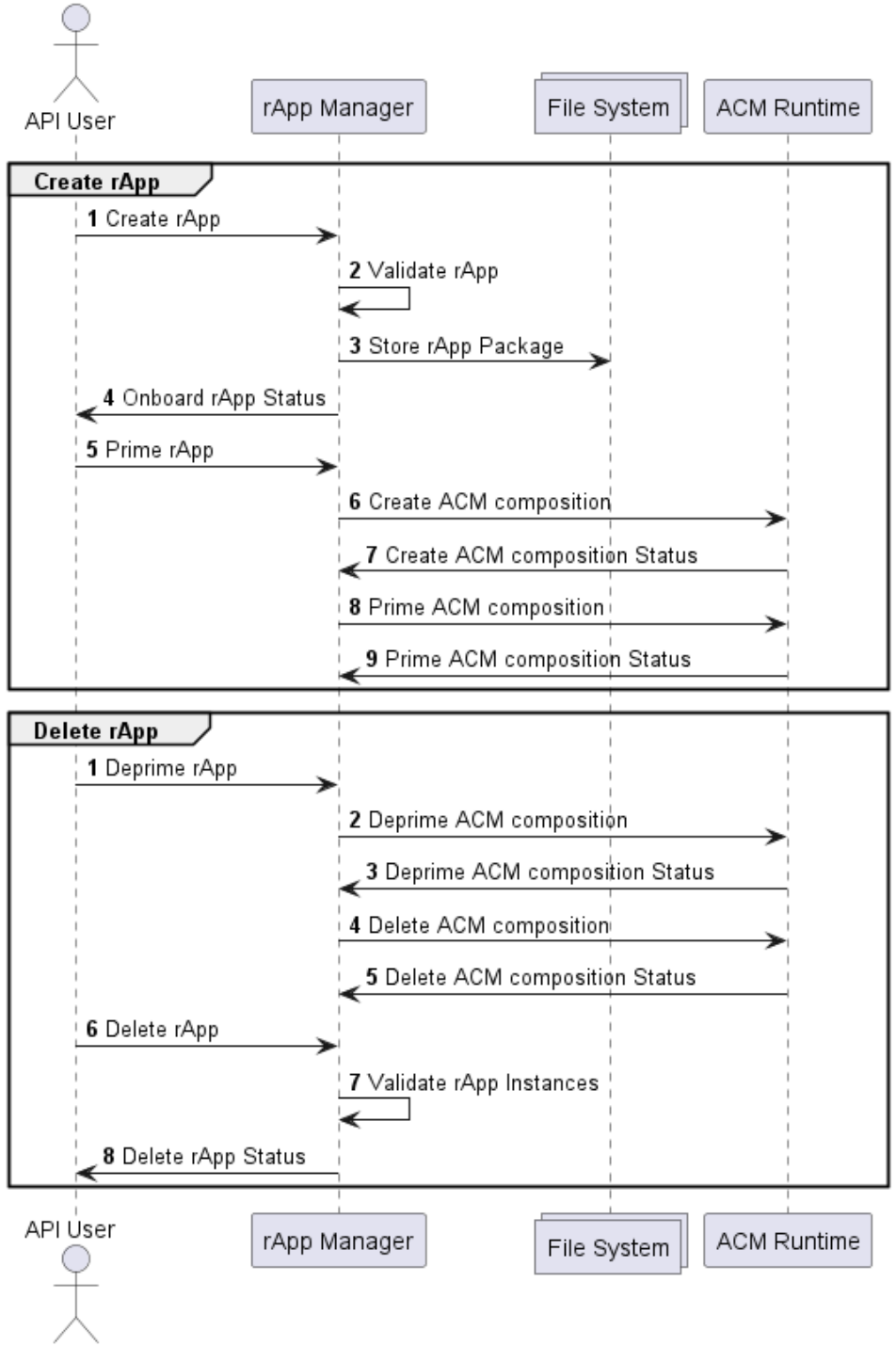
rApp States



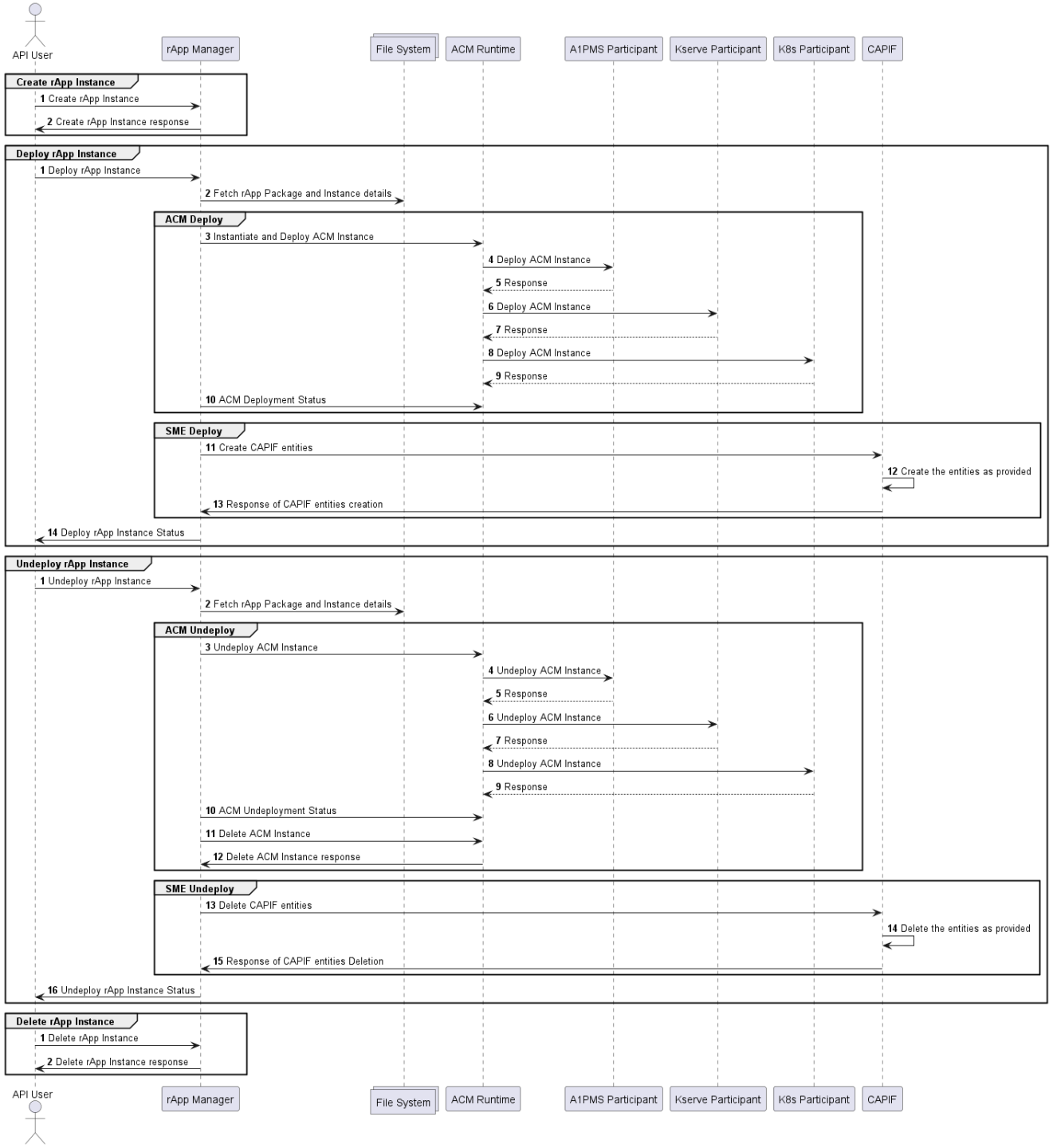
rApp Instance States



rApp flow


API User
rApp Manager
File System
ACM Runtime
Create rApp
1 Create rApp
2 Validate rApp
3 Store rApp Package
4 Onboard rApp Status
5 Prime rApp
6 Create ACM composition
7 Create ACM composition Status
8 Prime ACM composition
9 Prime ACM composition Status
Delete rApp
1 Deprime rApp
2 Deprime ACM composition
3 Deprime ACM composition Status
4 Delete ACM composition
5 Delete ACM composition Status
6 Delete rApp
7 Validate rApp Instances
8 Delete rApp Status
API User
rApp Manager
File System
ACM Runtime

rApp Instance flow



Ericsson Software Technology