

AAI R8 Architecture Review

- Brief Project Overview (brief as it should be known)
 - Active and Available Inventory (AAI) is the ONAP subsystem that provides real-time views of [Resources](#) and [Services](#) and their relationships. AAI not only forms a registry of active, available, and assigned assets, it also maintains up-to-date views of the multidimensional relationships among these assets, including their relevance to different components of ONAP. This project targets a logically centralized reference point for service and resource details serving other ONAP components and non-ONAP systems to enable fulfillment, closed loop, reporting, and other operational use cases. A&AI is critical to ONAP as the existing sources of truth do not provide a cross domain view and are not designed to serve this information to multiple clients.
- New component capabilities for Honolulu, i.e. the functional enhancements.
- New or modified interfaces
 - Most of the changes are related to the functional use cases; some of the functional use cases are test-only, they use existing types and edge rules.
- If they are modified, are they backwards compatible?
 - All are backward compatible
- Interface naming (point to an example)
 - see REST API SPEC
- Reference to the interfaces.
 - <insert for honolulu>
 - <https://docs.onap.org/projects/onap-aai-aai-common/en/latest/platform/index.html>
- What are the system limits?
 - Multithreaded concurrent operations on the same objects can cause duplicates in the graph - we have implemented stickiness in the haproxy that sits between AAI and clients to keep a client bound to the same node which can mitigate this, but we do not force locking on updates which means that malicious or careless clients can cause duplicates in the graph. We provide cleanup utilities for data grooming, but we recommend that clients are careful not to send multiple requests on the same object within microseconds of one another.
 - Operators should pay special attention to giving cassandra enough space, because we have known issues where things go pretty bad (like unhelpful or just wrong error messages) when the disks fill up on the cassandra cluster nodes.
- Involved use cases, architectural capabilities or functional requirements.
 - [AAI R8 Honolulu Release: Functional Use Cases](#)
- Non-functional requirements (Platform Maturity etc)
 - [AAI R8 Honolulu Release: Global Requirements](#)