

Considerations for Intent-Based Management Architecture(s)

Marinos Charalambides - UCL

Laurent Ciavaglia - Nokia (presenter)

IRTF NMRG Meeting, 25 July 2019

General Design Considerations

- Define the problem statement, challenges and requirements **first**
- Focus on intent-* specific aspects, not on yet another holistic management framework
- Design principles and methodology
 - Aim for simplicity ; flexibility : extensibility and integrability
 - Consider Service Based Architecture approach and principles

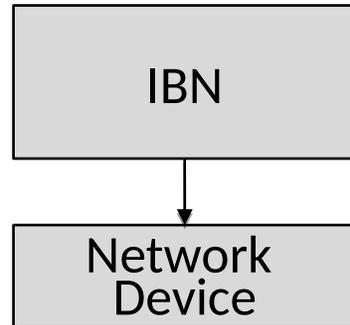
Other Considerations for the Design of IBN

Objective: simplify the configuration of complex infrastructures

- Intents should be agnostic of implementation details (importance of right abstractions)
- What should be the lower boundary of an IBN architecture (what should it configure)?
- How IBN co-exists with legacy management systems and emerging management technologies
- Automatic decomposition of intents
- Continuous enforcement of intents

Determining the Lower IBN Boundary

Option 1: Configure network device



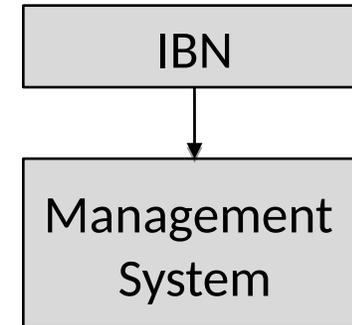
IBN architecture includes:

- Elaborate resource representations (device specific)
- Complex management logic/algorithms

Resulting solution

- **Bloated functionality**
- **Difficult to extend**

Option 2: Configure management system



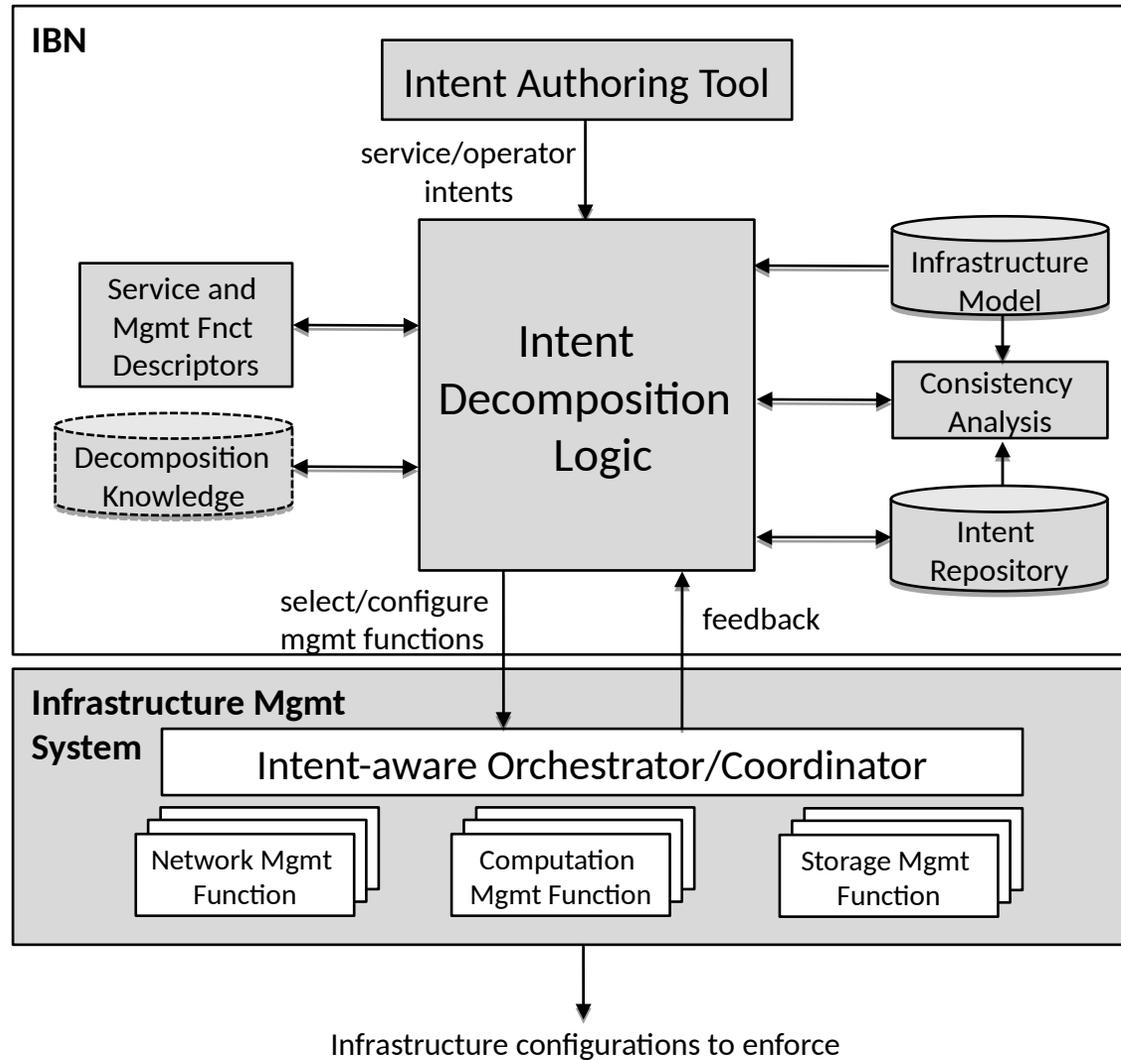
IBN architecture includes:

- Abstract representation of infrastructure resources
- Abstract representation of mgmt functions

Resulting solution

- **Simpler**
- **Extensible (wrt new resources/functions)**
- **Keep legacy mgmt systems in place**
- **Interoperate with NFV/SDN**

Thoughts on IBN Architecture Components



Architecture Components (1/2)

- Intent authoring tool
 - High-level technical requirements that drive management functions
 - Service- and operator-oriented intents
- Decomposition logic
 - Resolve intent to relevant service and select appropriate mgmt function(s)
 - Methods/techniques should not be prescribed (e.g. simple mapping -> elaborate reasoning)
- Decomposition facilitated by
 - Abstract infrastructure model – resources and commodities
 - Service and management function descriptors – representative attributes
 - Optional knowledge base – prior knowledge, best practices

Architecture Components (2/2)

- Intent repository
 - Store/retrieve intents – both high-level and low-level representations
- Consistency analysis
 - Detect/resolve inconsistencies, e.g. specification errors, competing goals, ...
 - Carried out during decomposition phase
- Interface to management system
 - Selected management functions and execution parameters
 - Orchestrator/coordinator as 'contact point' in infrastructure management system
 - Feedback – feasible configurations? Has intent stopped being met due to current conditions?