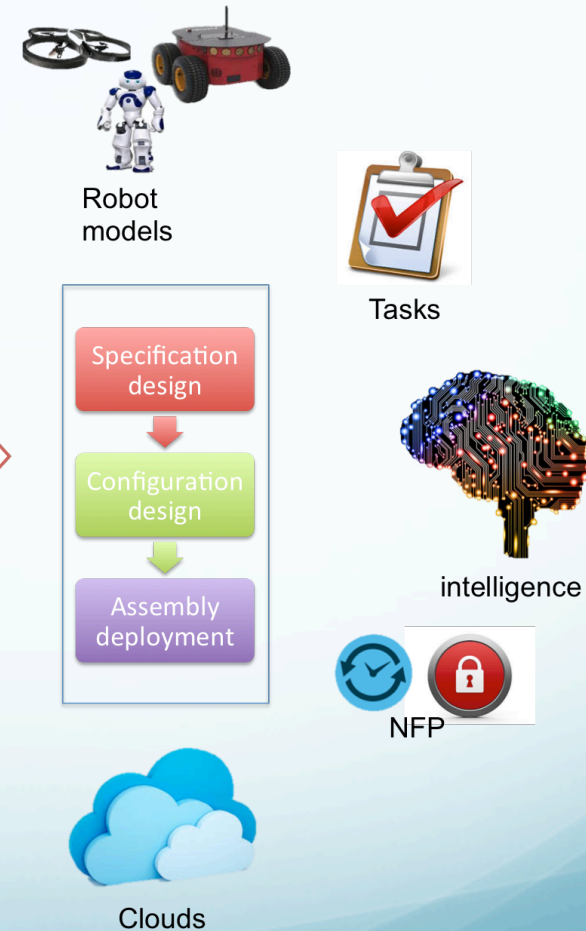


Towards An Architecture-Centric Approach to Manage Variability of Cloud Robotics

Authors: Huaxi (Yulin) Zhang, Lei Zhang, Zheng Fang,
Xianbo Xiang, Marianne Huchard, René Zapata

CRALA

- Propose CRALA : a Cloud Robotics Architecture Language
 1. Targets to specific domain of Cloud robotics
 2. Defines architectures of Cloud Robotic systems
 3. Demonstrates architectures of Cloud Robotic systems



Why Robots + Cloud?

- Provides a shared knowledge database
- Offloads heavy computing tasks to the cloud
 - Cheaper, lighter, easier-to-maintain hardware
 - Longer battery life
 - Less need for software pushes/updates
 - CPU hardware upgrades are invisible & hassle-free
- Skill / Behavior Database
 - Reusable library of “skills” or behaviors
 - Data-mining the history of all cloud-enabled robots

Why Architecture-Centric

- We consider that Cloud robotics should be architecture-centric, because architecture makes robotic system
 1. Universal: Unified interface makes robotic systems excellent compatibility
 2. Extensible: Componentized robotic system could have large-scale extensibility.
 3. Reusable: Reusable architectures and components are achieved by Model-Driven.

What we do

We propose CRALA : a Cloud Robotics Architecture Language:

1. Completely covers different aspects of Cloud robotic system, such as robot, component and Cloud descriptions.
2. More visualized and intuitively models Cloud robotic system's architectures
3. Greatly improves reusability of architectures in different levels.

Three Level Architectures

Specification

- Requirement

Configuration

- Design

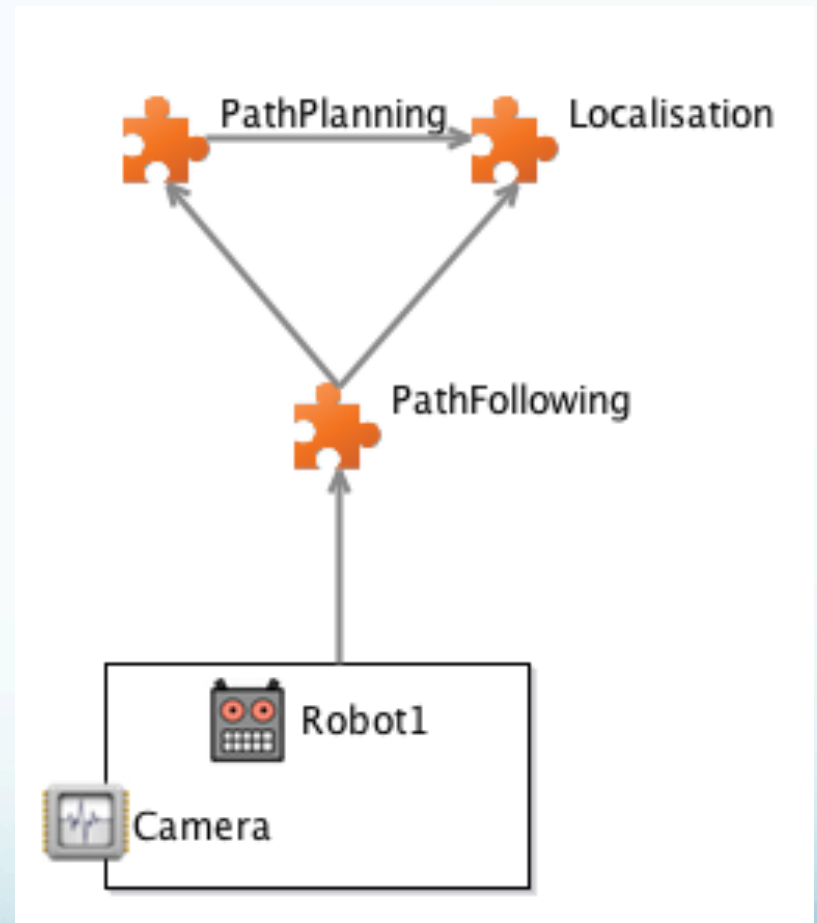
Runtime system

- Deployment in Cloud

1. Describes the functionalities required by Cloud Robotic system.
2. Defines the design details of Cloud Robotic system.
3. Illustrates how architecture configuration is deployed in Cloud.

Specification

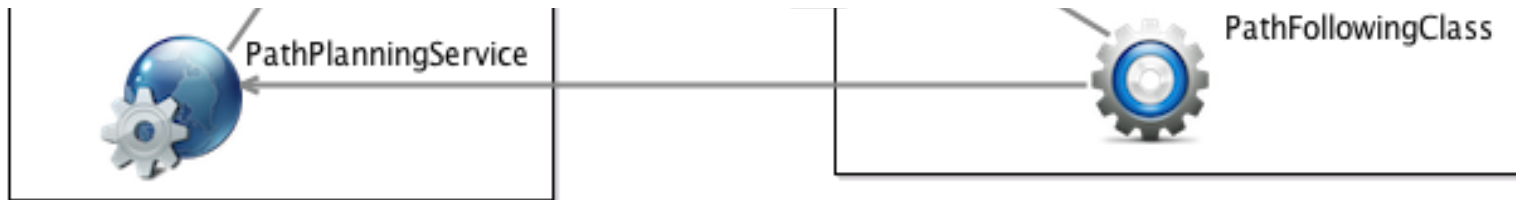
- Architecture specification is composed by *component roles, connections and abstract robots*.
- Component role describes the functionality of a component should play in a system, such as Localisation.



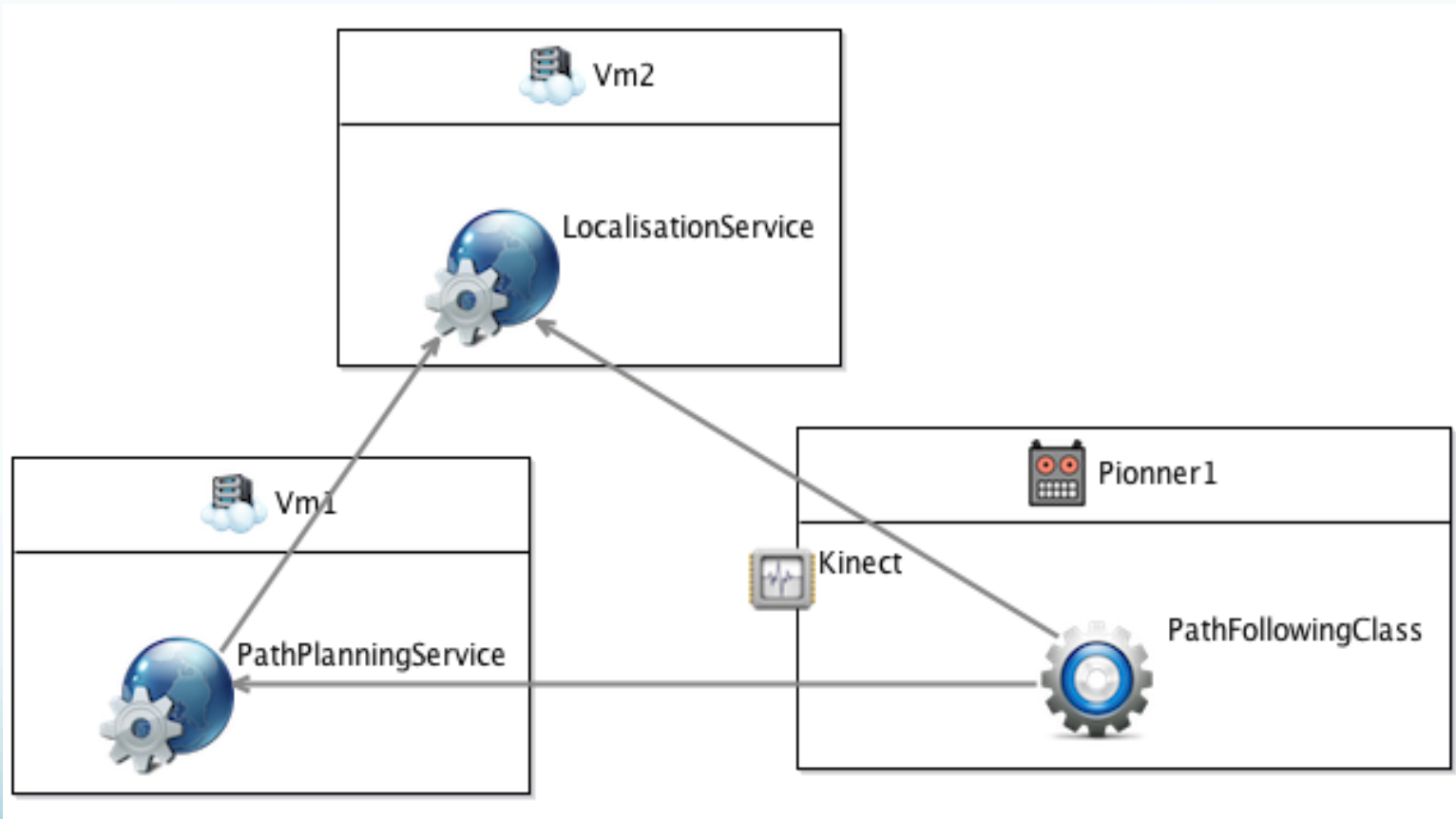
Configuration



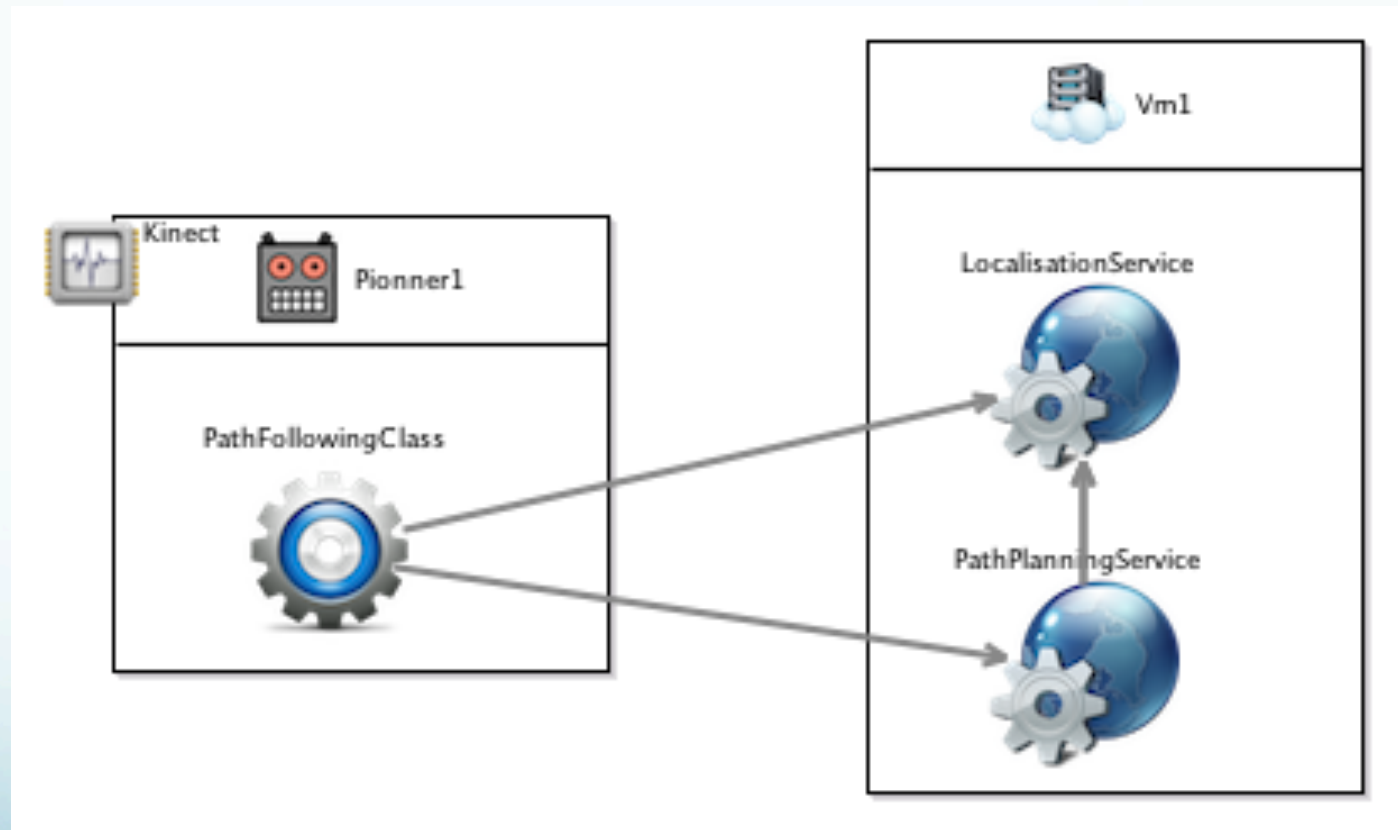
- Architecture configuration is composed by *virtual machines (VM)* and *robots*.
- Component or services are grouped and then located in different VMs or robots according to different designs.



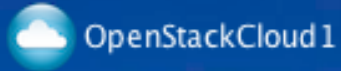
Configuration Example 1



Configuration Example 2



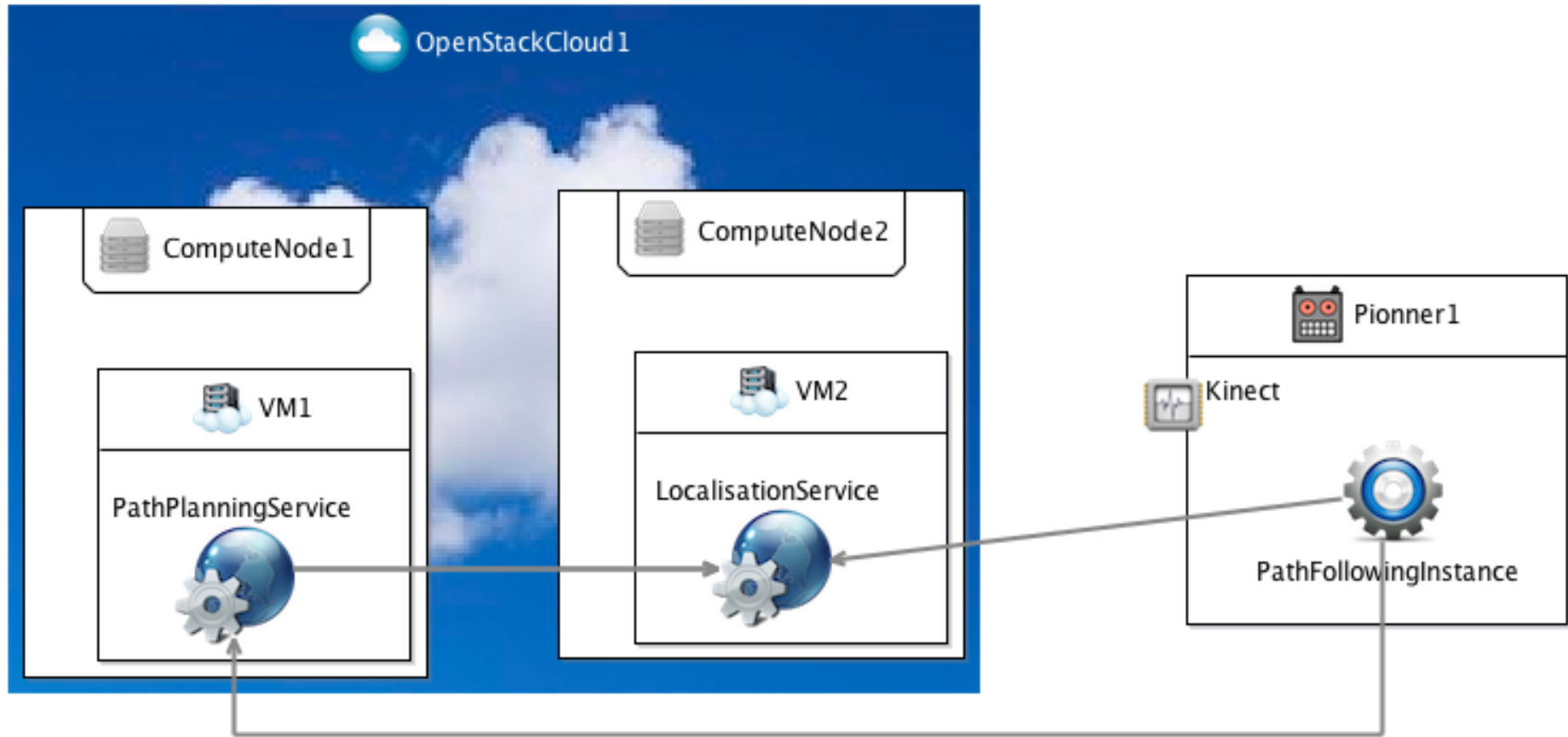
Runtime System



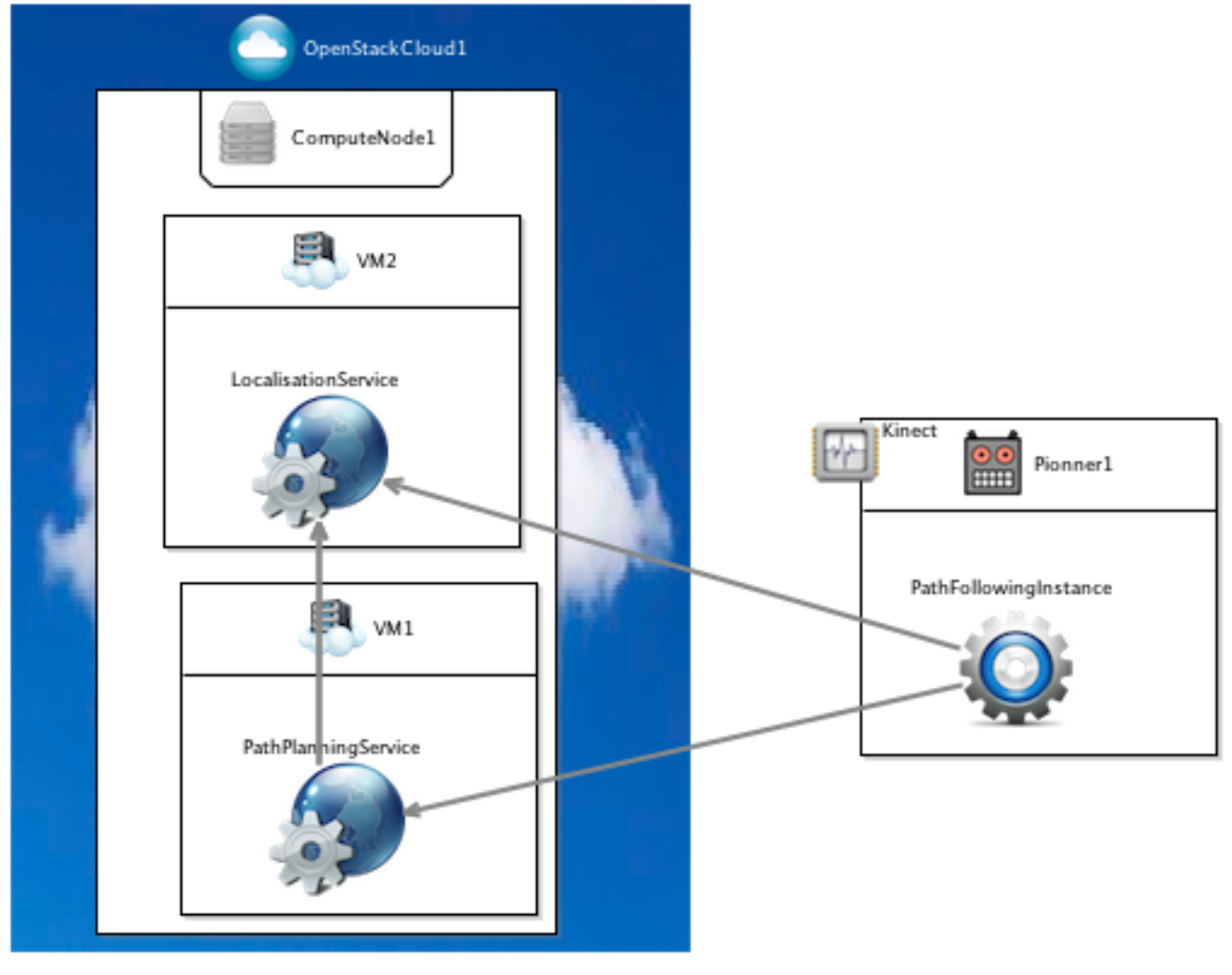
- Architecture of runtime system illustrates how configuration is deployed in Cloud.
- Explicitly describe the deployment of VMs in different Cloud's physical machines according to Cloud scheduler.



Runtime System Example 1

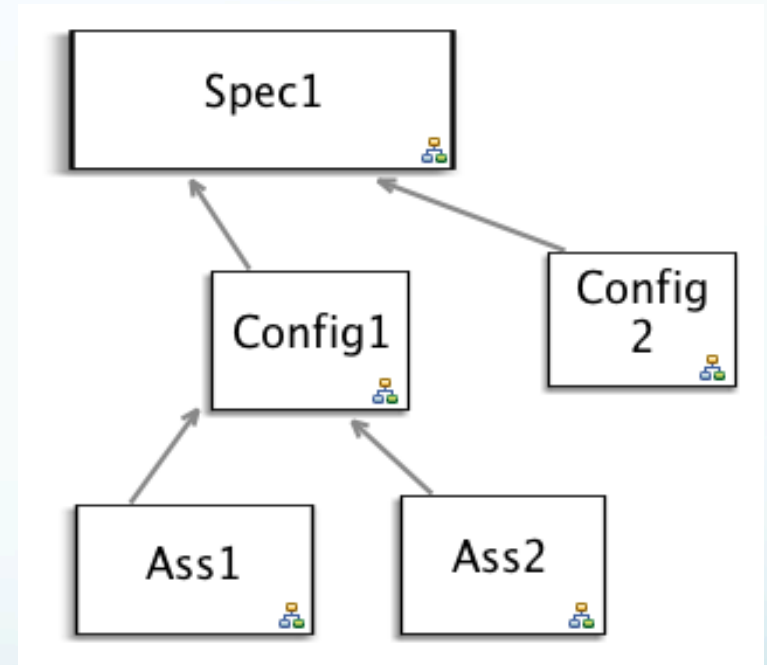


Runtime System Example 2



Architecture Reuse

- Maximize the reusability of Cloud robotic system's architectures in different abstraction levels.
- Every level architecture could be reused.
- Support the variety of system design.



Thank you