PaaS-independent approach to provision appropriate cloud resources for SCA-based applications
Motivating example

- **ComputePrice** SCA-based application
Provisioning resources for service deployment

- Deploying a service on PaaS steps:
  1. Identifying the needed platform resources,
  2. Packaging these resources with the service to be deployed,
  3. Instantiating and initializing the platform and service resources.

- A dedicating resources for each service to be deployed to host and run its service in a specialized MC.
Provisioning resources for service deployment
Provisioning resources for SCA-application deployment

- Deploying an SCA-based application on PaaS steps:

1. Slice the application to elementary entities,
   - SCA Slicer and Service aggregator tool

2. Package entities in service micro-containers,
   - MC deployment framework

3. Deploy obtained MCs on PaaS.
Provisioning resources for SCA-application deployment

1. Send an SCA project
2. Slice an SCA project and initialize the service list
3. Send the service list
4. Loop
   - For each element: Process SCA annotation, Process binding
   - Send java services
5. Alt [first services]
   - Send service client properties
   - Send java services
   - For each service create generic client
   - Send the generic client
6. Send the java properties
7. Send file properties
SCA-based application provisioning

- Step 1: *ComputePrice* application slicing
SCA-based application provisioning

- Step 2: *ComputePrice* application packaging on MCs
SCA-based application provisioning

- Step 3: ComputePrice application deployment on CF
Experiments

- Time response experiments (Apache Tuscany Vs MCs)
Experiments

- Memory consumption experiments (Apache Tuscany Vs MCs)