Cost

May 15, 2020

Chapter 1

Cost

All elements for the cost model are in the Cost NTO and can be found in OGIT under „ogit/Cost“.

The cost model defines the cost structure of a vendor through a hierarchy of cost elements.

Planning templates describe assumptions about how many instances of a certain service with a certain parameterization the vendor will have to operate. Applying a price function to the cost model yields the price model.

Cost element for a Service

There are three levels of cost elements:

- Top level, e.g. „Microsoft for MicProject in EU“
- Middle level, e.g. „OS Hosting“, „CPU“, „Memory“
- Bottom level, e.g. „HR“, „Licensing“, „Housing“, „Depreciation“

Between the different levels, for each cost element there could be two types of edges:

- Incoming
- outgoing

Each type of edge could have two types of connections:

- consistsOf
- contributesTo

A service, e.g. DBHosting has associated to it different types of costs. Let’s call each one of them a cost element.

Each cost element could have an incoming and an outgoing edge, which connects it to another cost element and contains it’s weight indicating how much it contributes/consists in relation to another cost element.

E.g.: from the picture below we can understand that the CPU cost is consisting in 30% of HR cost. 5% of the HR cost contribute to the CPU costs.

Since currently in OGIT there are no edges with weights, the following approach to represent it through entities was taken:

Each service offering is to be offered for a service with attributes defined for certain values or ranges. E.g.: price defined for Memory of size 14188 – 1572864.

Within one order, it could be specified the ordering of a service with attributes containing the concrete values of a range.

An implementation template consists of some interconnected MARS-Nodes.

Together with the values specified by the offering and the order it becomes a concrete implementation.

A *Planning Template* is an estimated planning of cost elements referring to:

- A. an Implementation Template,
- B. Planning Parameter with estimated average parameters for an order and concrete value of a parameter for an Offering and the planned number of service instances (volume attribute of entity PlanningTemplate.
- C. A PlanningTemplate will consist of a recursive set of sub-templates, connected through the edge has, where each one of them will represent a service.

As there could be some services that don't have any parameters, the additional edge plansCostFor from Planning↩ Template to Service is required.

A Planning Parameter will contain a concrete value of a Parameter and the statistics derived from the expected distribution as attributes.