

# CAMEL FAQ

This is FAQ section dedicated to CAMEL. It includes a standard set of questions-answers related to how specific parts of use-cases can be modelled via CAMEL. This set can be dynamically extended. In particular, the user can just pose the question in the related section of this page or in a new section that he/she can create. Then, the CAMEL experts will be happy to answer this question to assist in the further dissemination of the CAMEL modelling knowledge.

## Section 1. Deployment & Requirement

### 1.1 How do I specify my component's resource requirements?

Answer: I specify a requirement set which points to the resource requirement. This resource requirement can include a set of features or attributes. Features group together attributes and can be annotated via the MDS (e.g., CPU). Attributes are also annotated by MDS and are associated to a certain value, thus representing an attribute constraint (e.g., `hasMaxNumberOfCores = 4`).

### 1.2 How can I indicate that two or more components should be co-hosted?

Answer: I specify a *LocationCoupling* that references these components and has as type (see *LocationCouplingType* enumeration) the value of `SAME_HOST`.

### 1.3 Do I need to specify hosting relations for my components (e.g. with VMs)?

Answer: Not any more. The system takes care of that once (initial) deployment reasoning is performed.

### 1.4 How do I specify a non-functional requirement for my component?

Answer: I need to model an SLO which refers to the constraint that has to be specified. This is usually a metric constraint for which I need to refer to the metric's context as well as to the constraint's threshold and comparison operator. In case I need to specify a constraint on a metric variable, instead of the metric context I need to refer to that variable.

## Section 2. Metric & Scalability

### 1.1 How do I specify utility functions?

Answer: I need to define a metric variable that models this function. Such a variable should include the specification of the actual formula that needs to be assessed. Such a formula can be a mathematical expression that can involve other metrics or metric variables.

### 1.2 Can a utility function contain a normal metric (e.g., raw CPU utilisation)?

Answer: Yes, it can. See previous answer.

### 1.3 What kinds of metrics exist & how they can be measured?

Answer: There are raw metrics (e.g., raw CPU utilisation) which can be measured directly from sensors as well as composite metrics (e.g., average CPU utilisation) which can be measured via formulas (which as explained above can include other metrics or even metric variables).

### 1.4 How do I specify a scalability rule?

Answer: As a mapping between an event and a scaling action. An event can be a non-functional event or an event pattern, i.e., a logical or time-based combination of other events. Only horizontal scaling actions can be supported by the Melodic platform currently. Such actions can be specified by referring to the component to be scaled as well as the positive (or negative) number of instances of that component that need to be created (or destroyed).

## Section 3. Data & Deployment

### 1.1 How do I specify the characteristics of my data / data sources?

Answer: As features and/or attributes (see also answer on Question 1.1). Features can be used for grouping attributes together (when they refer to the same entity). Both features and attributes can be annotated via the MDS. The attributes actually specify a certain constraint over a specific data (source) characteristic.

### 1.2 How should I model a data processing component?

Answer: As a software component for which a *ClusterConfiguration* has to be specified. The latter indicates both which big data processing framework will be used to execute this component (as an annotation from the MDS), specific configuration parameters that need to be passed to this framework as well as the component's download URL. As the component that we need to model is a data processing one, we need to also

associate it with certain data which can be either consumed or produced by it. We can also indicate if that component is long-lived and whether we can co-host instances of that component in the same VM. We definitely need to also associate this component with the requirements that should hold for it (e.g., resource ones).

### **1.3 How should I model a data component?**

Answer: As a software component which manipulates/manages a data source. Such a component will need to be associated with the way it needs to be configured (usually a *ScriptConfiguration*) as well as the requirements that should hold for it (e.g., resource, OS, location, etc.).

### **1.4 How do I specify location requirements for my data / data sources?**

Answer: Such requirements need to be associated with the (data) component that manages / encapsulates the respective data source (see *RequirementSet* class and *locationRequirement* reference in that class).