Warsaw University of Technology, Poland
RISE Research Institutes of Sweden AB, Sweden
Institute of Mathematics and Computer Science, University of Latvia, Latvia
EurA AG, Germany
Municipality of Vejle, Denmark
Lithuanian Innovation Center, Lithuania
Machine Technology Center Turku Ltd., Finland
Tartu Science Park Foundation, Estonia

# BalticLSC Requirements

## BalticLSC requirements elicitation workshops

| Work package | WP4, WP5 |
|---|---|
| Document number | |
| Document type | |
| Title | BalticLSC Requirements |
| Subtitle | BalticLSC requirements elicitation workshops |
| Author(s) | Michał Śmiałek, Kamil Rybiński, Radosław Roszczyk |
| Reviewer(s) | |
| Accepting | |
| Version | 0.35 |
| Status | Draft |
| Distribution | Internal |

# 1. Functional Requirements

## 1.1 Actors

**App user**

A person performing calculations using the BalticLSC Network.

**Developer**

A person able to create applications for the BalticLSC Network.

**Supplier**

A representative of a body (company, research centre, etc.) that supplies computation power to the BalticLSC Network.

**Administrator**

A person supervising the BalticLSC Network.

**Computation Resource**

An external machine constituting a Computation Node of the BalticLSC Network that performs computations.

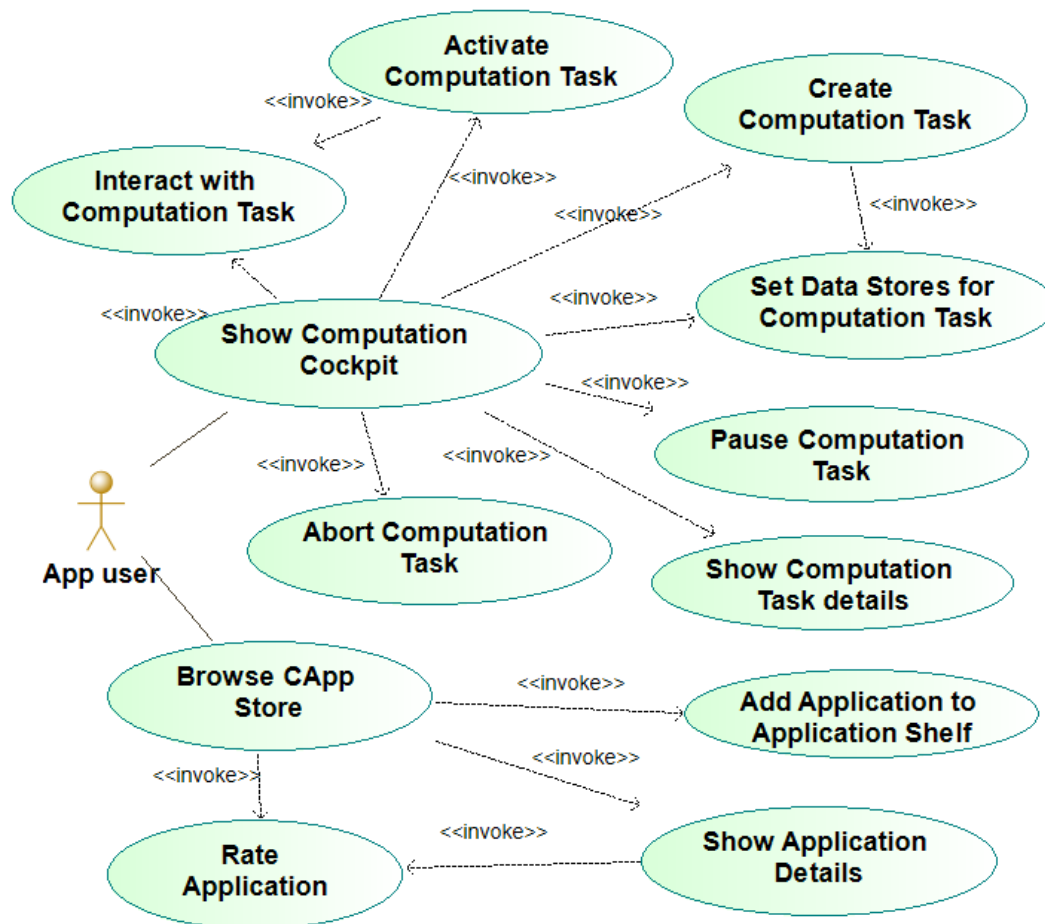## 1.2 Computation Application Usage



*Figure 1. Use Case Diagram for Computation Application Usage*

| Name | Show Computation Cockpit |
|---|---|
| Description | Show the App Shelf with all the Computation Applications that the user can run – retrieved from the CApp Store or created by him/her. Also shows the Computation Rack with all the Computation Tasks created by the user, based on those applications. |

| Name | Create Computation Task |
|---|---|
| Description | Prepare to run (but not run yet) a new Computation Task using a selected Computation Application. Includes specifying the Computation Valuation. |

| Name | Activate Computation Task |
|---|---|
| Description | Start executing an existing Computation Task which is just created or has been paused. Might involve interacting with the task. |

| Name | Interact with Computation Task |
|---|---|
| Description | Performs required interactions (entering parameters, selecting options, etc.) with a selected Computation Task. Available only for the Computation Tasks that need such interactions. |

| Name | Set Data Stores for Computation Task |
|---|---|
| **Description** | Allows to define parameter values for Data Stores as specified by the Computation Application. This use case can be used dynamically during computations to change e.g. ways in which the Computation Application interacts with the App user (pauses for data input etc.). |

| Name | Show Computation Task details |
|---|---|
| **Description** | Show details of a Computation Task including its valuation and configuration. If task execution was started, shows also the task status and statuses of its subtasks. |

| Name | Abort Computation Task |
|---|---|
| **Description** | Stops execution of a running Computation Task. All the computations are aborted and cannot be activated in the future. Appropriate billing for the computations performed so far is calculated. |

| Name | Pause Computation Task |
|---|---|
| **Description** | Stops execution of a running Computation Task. All the computations are halted but can be activated in the future. All the internal data is stored and billing for storage is calculated. |

| Name | Browse CApp Store |
|---|---|
| **Description** | Allows for browsing of verified Computation Applications in the CApp (Computation Application) Store. |

| Name | Add Application to App Shelf |
|---|---|
| **Description** | Add a selected Computation Application to the user's App Shelf, which makes it available in the Computation Cockpit. |

| Name | Show Application Details |
|---|---|
| **Description** | Show detailed information about a Computation Application. |

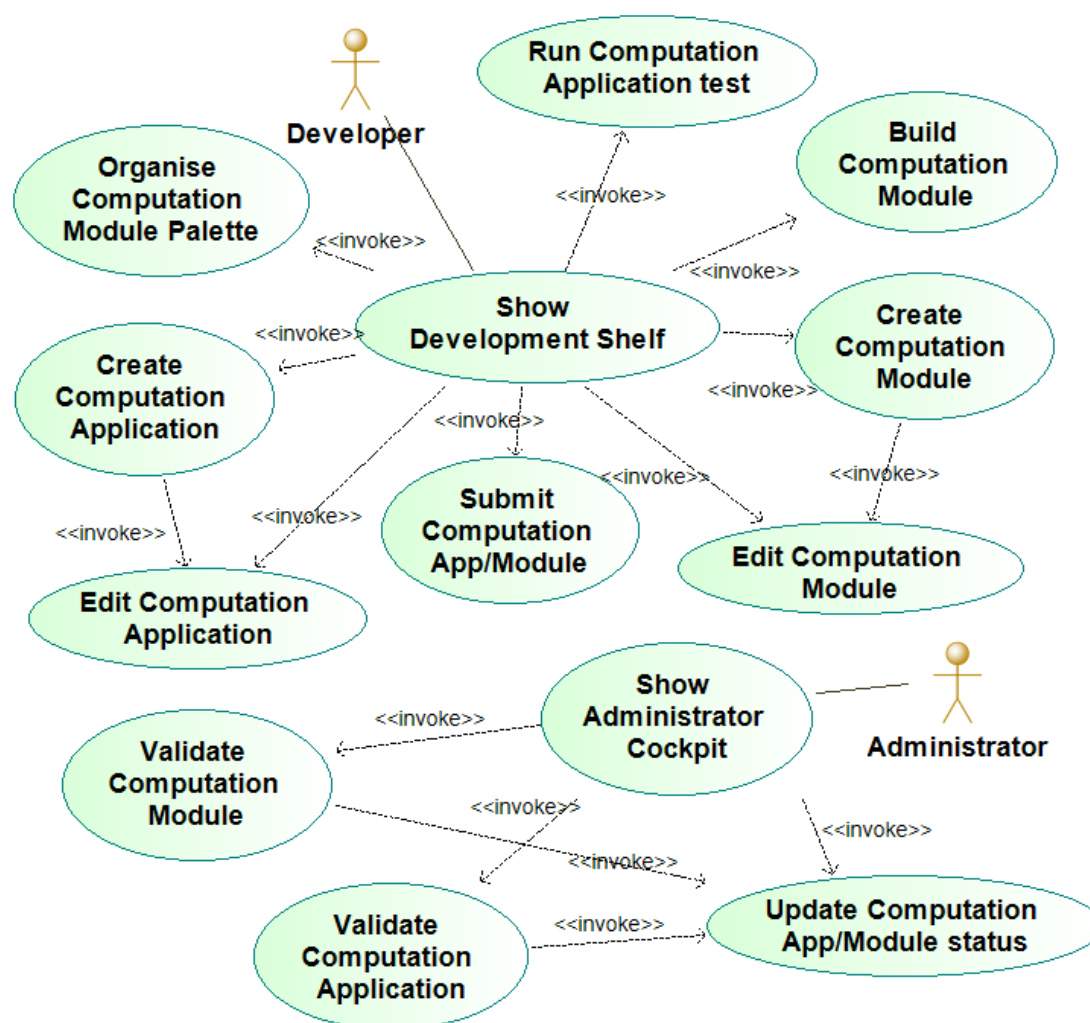| Name | Rate Application |
|---|---|
| **Description** | Allows the user to review a selected Computation Application. |

## 1.3 Computation Application Development



*Figure 2. Use Case Diagram for Computation Application Development*

| Name | Show Development Shelf |
|------|------------------------|
| Description | Allows to browse all the Computation Modules and Applications developed by the current user. |

| Name | Submit Computation App/Module |
|------|-------------------------------|
| Description | Allows for submitting a Computation Module or Application to be verified and made available to other users. |

| Name | Create Computation Application |
|------|-------------------------------|
| Description | Create a new Computation Application entry. |

| Name | Edit Computation Application |
|------|-----------------------------|
| Description | Specify content of a Computation Application using the CAL editor. |

| Name | Create Computation Module |
|---|---|
| Description | Create a new Computation Module entry. |

| Name | Edit Computation Module |
|---|---|
| Description | Specify content of a Computation Module – its description and executables. |

| Name | Build Computation Module |
|---|---|
| Description | Prepare the executable form of a computation module (meant for future versions). |

| Name | Organise Computation Module Palette |
|---|---|
| Description | Select Computation Modules to be available in the Palette for the given Developer. Allows also to organize order of Modules in the Palette. |

| Name | Run Computation Application test |
|---|---|
| Description | Allows for running of developed Applications in restricted test mode on the BalticLSC Network. |

| Name | Show Administrator Cockpit |
|---|---|
| Description | Shows the App/Module Repository which allows to browse all Computation Modules and Applications that have been ever submitted by any user (including active and inactive). |

| Name | Validate Computation Application |
|---|---|
| Description | Show advanced details about a Computation Application, including its implementation (code) and allow to validate its usage in the BalticLSC Network. |

| Name | Validate Computation Module |
|---|---|
| Description | Show advanced details about a Computation Module, including its implementation (code) and allow to validate its usage in the BalticLSC Network. |

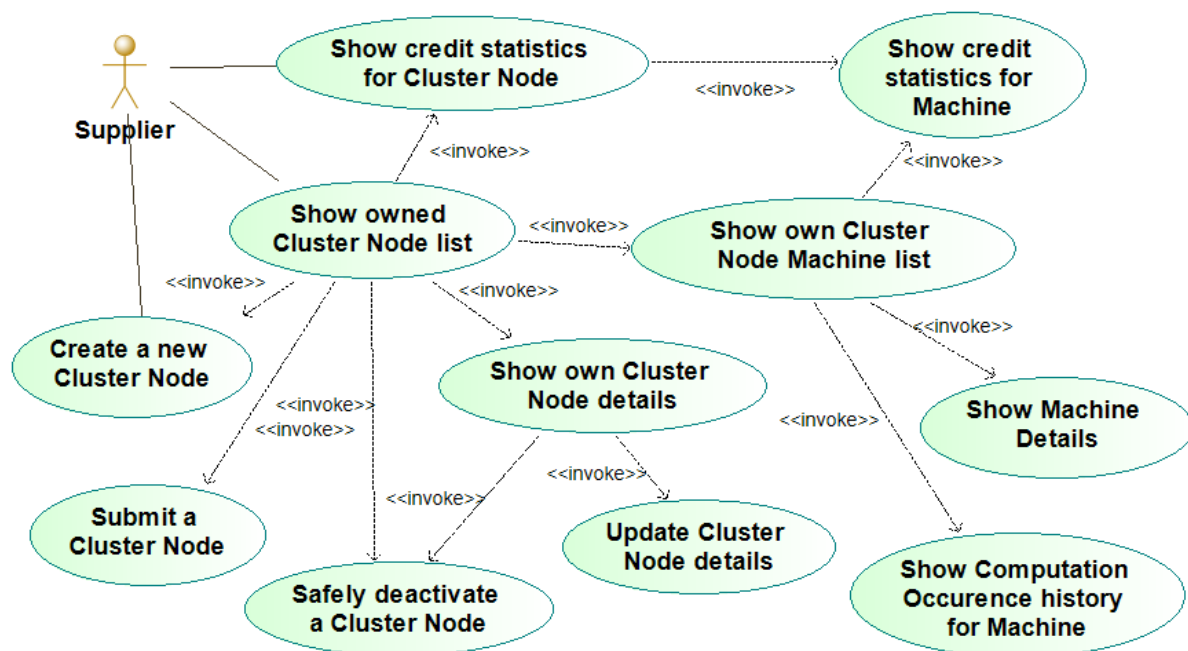| Name | Update Computation App/Module Status |
|---|---|
| Description | Allows for status change, including approval of a Computation Application or Module. |

*Figure 3. Use Case Diagram for Computation Resource Management*

| Name | Create a new Cluster Node |
|---|---|
| Description | Define a new Cluster Node entry. |

| Name | Show owned Cluster Node list |
|---|---|
| Description | Show all Cluster Nodes defined by the current user. |

| Name | Submit a Cluster Node |
|---|---|
| Description | Submit a Cluster Node for verification, after which it will be made available to other users for performing computations. |

| Name | Show own Cluster Node Machine list |
|---|---|
| Description | Show all Machines for a selected Cluster Node. |

| Name | Show credit statistics for Cluster Node |
|---|---|
| Description | Show statistic about credits earned by the user's Cluster Node. |

| Name | Show credit statistics for Machine |
|---|---|
| Description | Show statistic about credits earned by the user's Machine. |

| Name | Show own Cluster Node details |
|---|---|
| Description | Show details about a selected Cluster Node. |

| Name | Show Machine details |
|---|---|
| Description | Show details about a selected Machine. |

| Name | Show Computation Occurrences history for a Machine |
|---|---|
| Description | Show information about all the Computation Occurrences that have been ever executed on a selected Machine. Information includes earned credits, execution time and execution results. |

| Name | Safely deactivate a Cluster Node |
|---|---|
| Description | Stop receiving new Computation Tasks by a Cluster Node and deactivate it after all already assigned Computation Tasks are completed. Optionally: freeze and save all Computation Task and deactivate Cluster Node right away. |

| Name | Update Cluster Node details |
|---|---|
| Description | Allows for modifying information that describes a selected Cluster Node. |

## 1.5 Computation Resource Supervision



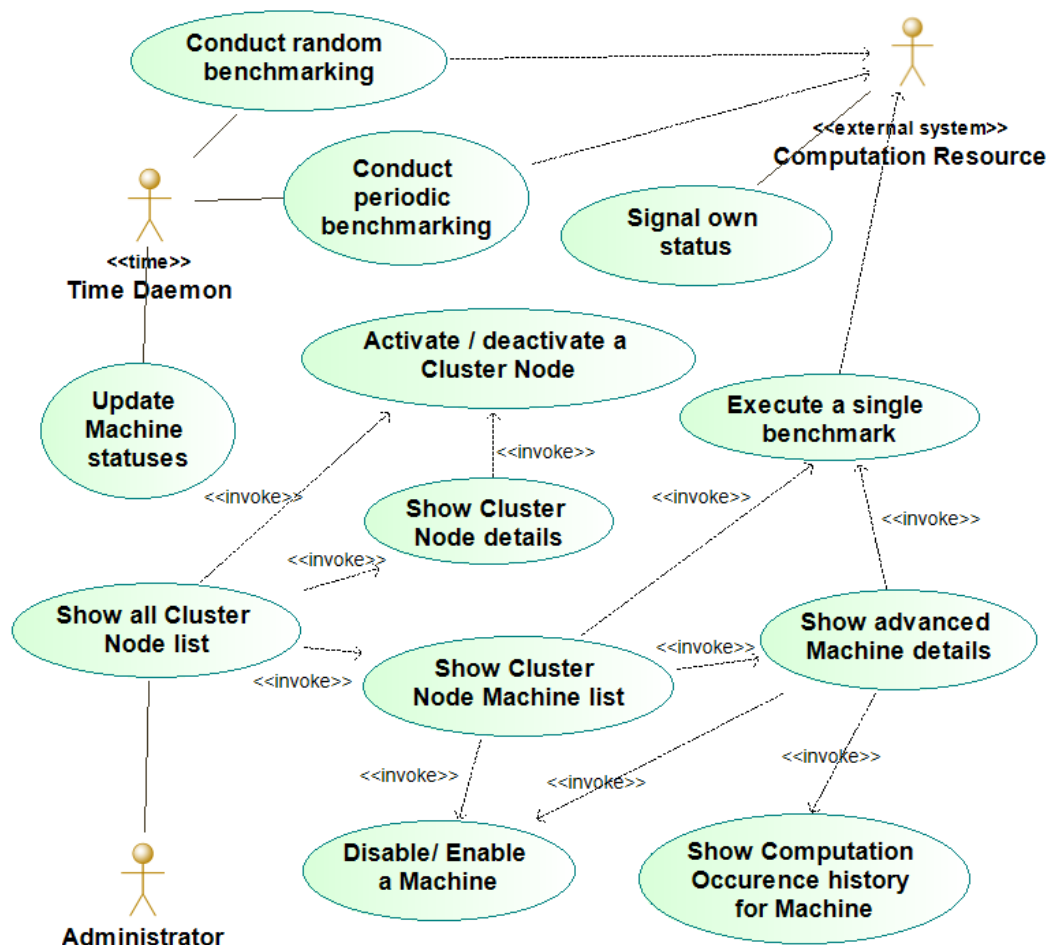*Figure 4. Use Case Diagram for Computation Resource Supervision*

| Name | Show all Cluster Node list |
|---|---|
| **Description** | Shows a list of all submitted Cluster Nodes. |

| Name | Show Cluster Node Machine list |
|---|---|
| **Description** | Shows a list of machines for a selected Cluster Node. |

| Name | Show Cluster Node details |
|---|---|
| **Description** | Show all details about a Cluster Node. |

| Name | Activate / deactivate a Cluster Node |
|---|---|
| **Description** | Allows to make a selected Cluster Node available or unavailable for performing computations by the users other than its owner. |

| Name | Show advanced Machine details |
|---|---|
| **Description** | Shows all details about a Machine, including its benchmark results. |

| Name | Disable / Enable a Machine |
|---|---|
| **Description** | Allows to make a selected Machine available or unavailable for performing computations of users other than its owner. |

| Name | Execute single benchmark |
|---|---|
| **Description** | Choose and run a benchmark on a selected Machine. |

| Name | Conduct random benchmarking |
|---|---|
| **Description** | Run benchmarks on randomly selected set of Machines. |

| Name | Conduct periodic benchmarking |
|---|---|
| **Description** | Periodically run benchmarks on all Machines. |

| Name | Signal own status |
|---|---|
| **Description** | Allows a Cluster Node to inform the BalticLSC Network of its machines' statuses and send information about progress of Computation Occurrences assigned to them. |

| Name | Update Machine statuses |
|---|---|
| **Description** | Mark Machines that didn't inform about its statuses as inactive. |

## 1.6 Computation Application Requests
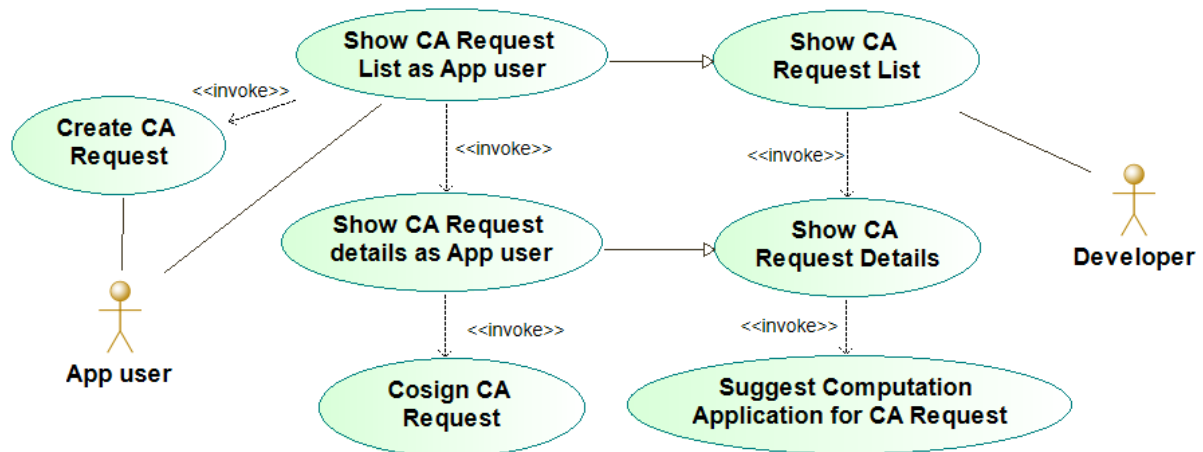


*Figure 5. Use Case Diagram for Computation Application Requests*

| Name | Create CA Request |
|------|-------------------|
| **Description** | Allows to create a Computation Application Request specifying details of a Computation Application user needs. |

| Name | Show CA Request List/Show CA Request List as End-user |
|------|-------------------------------------------------------|
| **Description** | Show a list of all created Computation Application Requests. |

| Name | Show CA Request Details/Show CA Request Details as End-user |
|------|-----------------------------------------------------------|
| **Description** | Show the definition of a selected Computation Application Request. |

| Name | Cosign CA Request |
|------|-------------------|
| **Description** | Mark the current user as being interested in a Computation Application Request created by another user. |

| Name | Suggest Computation Application for CA Request |
|------|-----------------------------------------------|
| **Description** | Allows to assign computation applications as fulfilling a selected Computation Application Request. |

# 2. Data Model

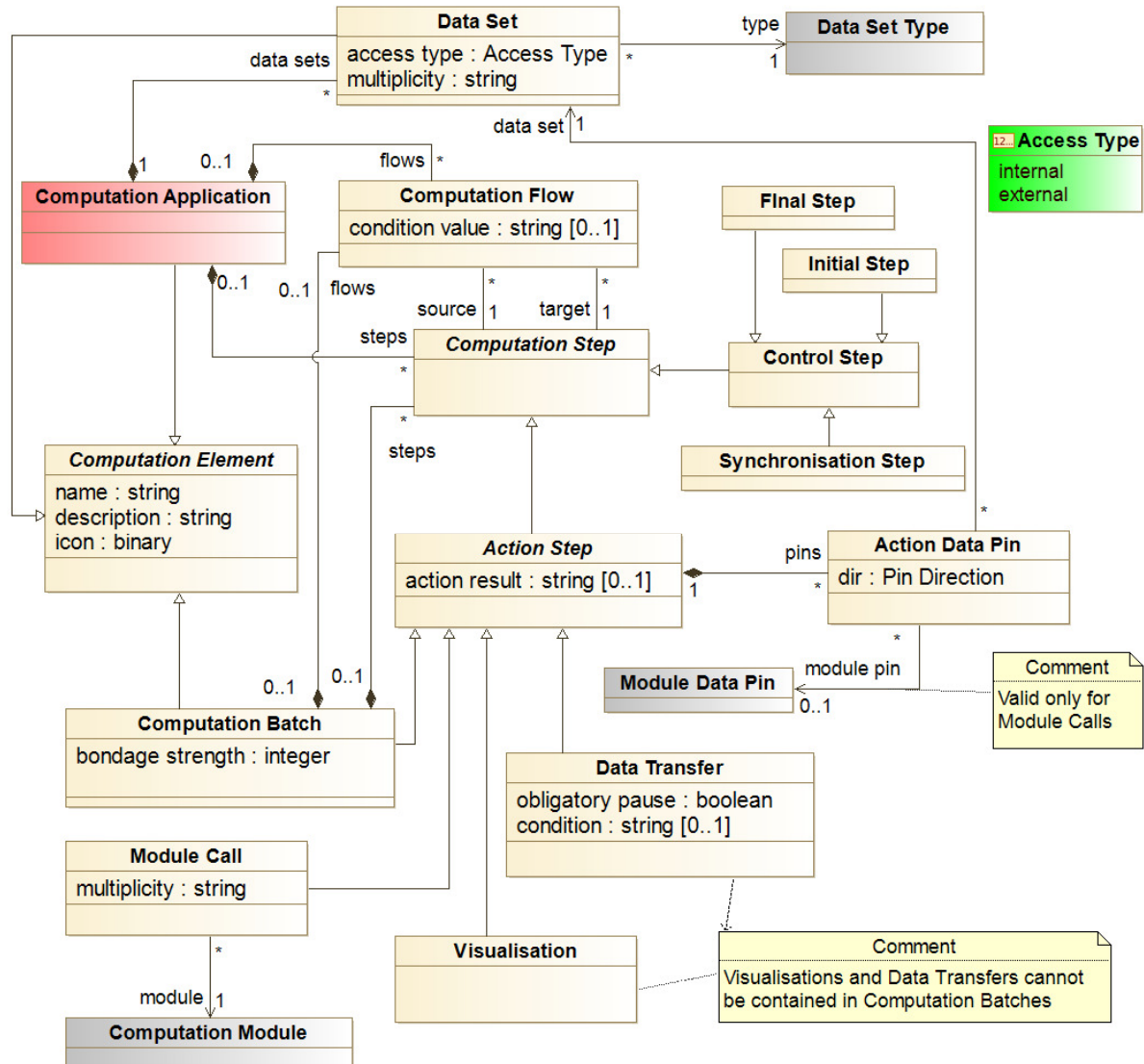## 2.1 CAL: Computation Applications



*Figure 6. Class Diagram for Computation Applications*

**Computation Application**

A program in the CAL language that can be run by the user. Contains all the Computation Steps and Computation Flows, defining the way computations should be executed. It also contains declarations of Data Sets that will be used as inputs or outputs for the computations.

**Computation Step**

A single step in the flow of computations. Specializes into concrete steps (see below).

**Computation Flow**

Specifies flow of computations between Computation Steps. Can contain a condition value that guards flow over the current Computation Flow, based on the result of condition value of the source Computation Flow.

**Control Step**

A type of Computation Step that does not perform any data processing, but controls flow of execution.

**Initial Step, Final Step**

Type of control steps that determine the starting and final point of computations. There can be one of each with any Computation Application or Computation Batch.

**Synchronisation Step**

A type of control step that synchronises parallel computations. Waits for all the action steps on incoming flows to finish, and then runs all the action steps on the outgoing flows in parallel

**Action Step**

A type of Computation Step the performs data processing and ends with an action results that can be used to control the flow of execution in further processing.

**Module Call**

A type of action step that executes a specific Computation Module.

**Computation Batch**

A type of action step that executes a batch, containing specific data processing execution. It contains Computation Steps and Computation Flows. Cannot contain Data Transfers and visualisations. It defines bondage strength that determines assignment of internal Module Calls to Cluster Nodes.

**Data Transfer**

A type of action that transfers data between some Data Sets. The way the operation is performed depends on the Data Stores associated with the Data Sets. If the input Data Store is marked as "user interaction" then the user receives a signal to access an appropriate data entry form. In other case, data is transferred from some preset storage (file, database, etc.). Data can be transferred to a preset store (internal or external) of a type that matches the input data store.

**Visualisation**

A type of action that allows to visualize data. Probably not implemented in the early version.

**Data Set**

A declaration of some data to be used as input or output for Action Steps. Data sets comply with specific Data Set Types, and during runtime should be associated with specific Data Stores.

**Action Data Pin**

A declaration of some Data Set as input and/or output to the given Action Step.
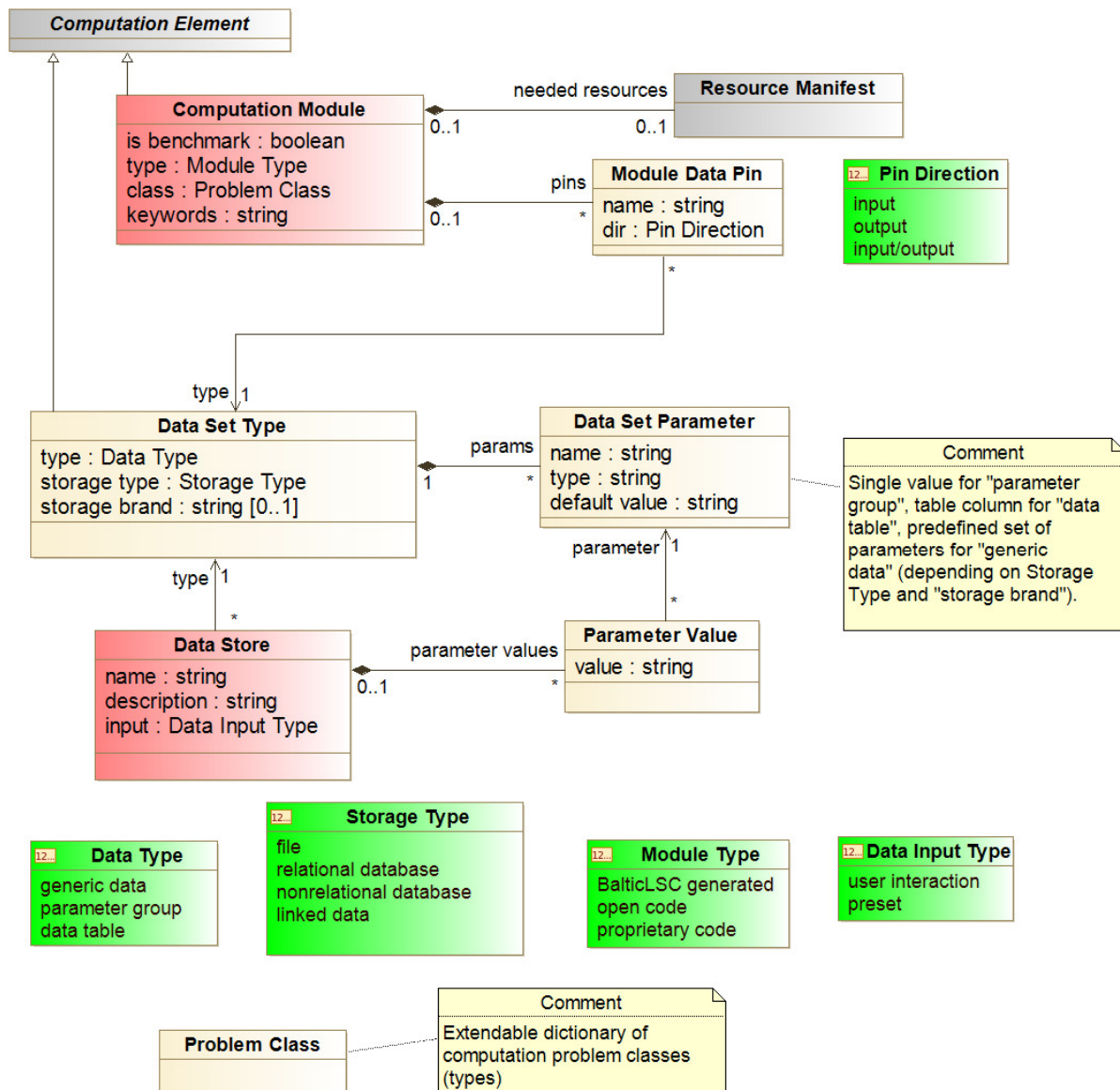
## 2.2 CAL: Computation Modules and Data Sets



*Figure 7. Class Diagram for Computation Modules and Data Sets*

### Computation Module

A declaration of an executable module that can perform computations according to its internal code. It contains a Resource Manifest that declares resources needed to run the module. It also contains one or more Module Data Pins that declare types of data that need to be input and/or output from the module. Modules can be assigned to a specific Problem Class and keywords that facilitate searching.

### Module Data Pin

A declaration of some Data Set Type as input and/or output to the given Computation Module.

### Data Store

A declaration of a specific external source/destination for data. It contains a set of parameter values that determine the contents of the Data Store, determined by the associated Data Set Type. Depending on the storage type and data type, these parameter values can contain specific values for parameter groups associated with computations, column names for tables or access parameters for generic data sets.

## Data Set Type

A declaration of a predefined type for instantiating various data sets and data stores used in computations. These sets types can be defined and accessed by computation users. They can contain references to data tables, parameter groups or generic data.

## Data Set Parameter

A declaration of a single parameter within a Data Set Type.

## Parameter Value

A specific value specified within a Data Store. Can be blank if the Data Store is of type "user interaction".
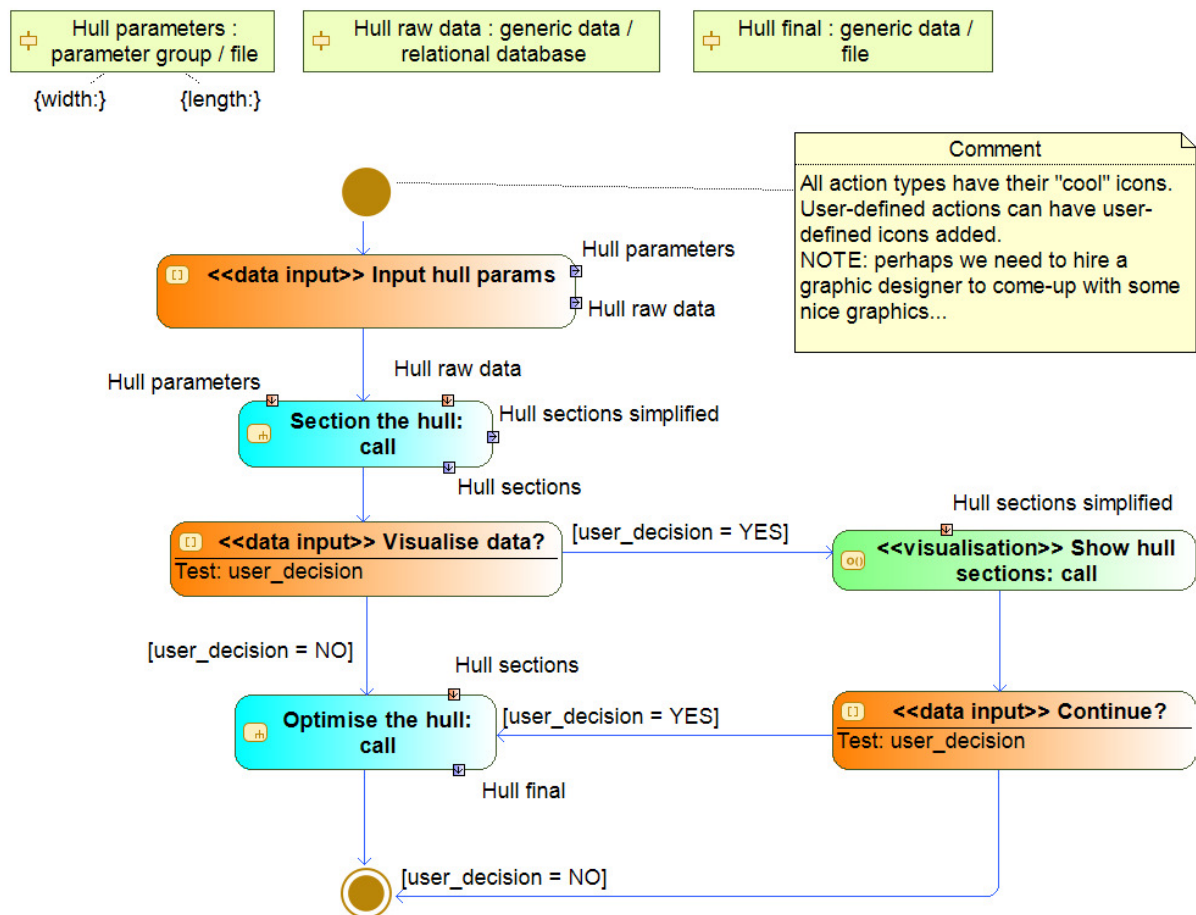
## 2.3 CAL Example
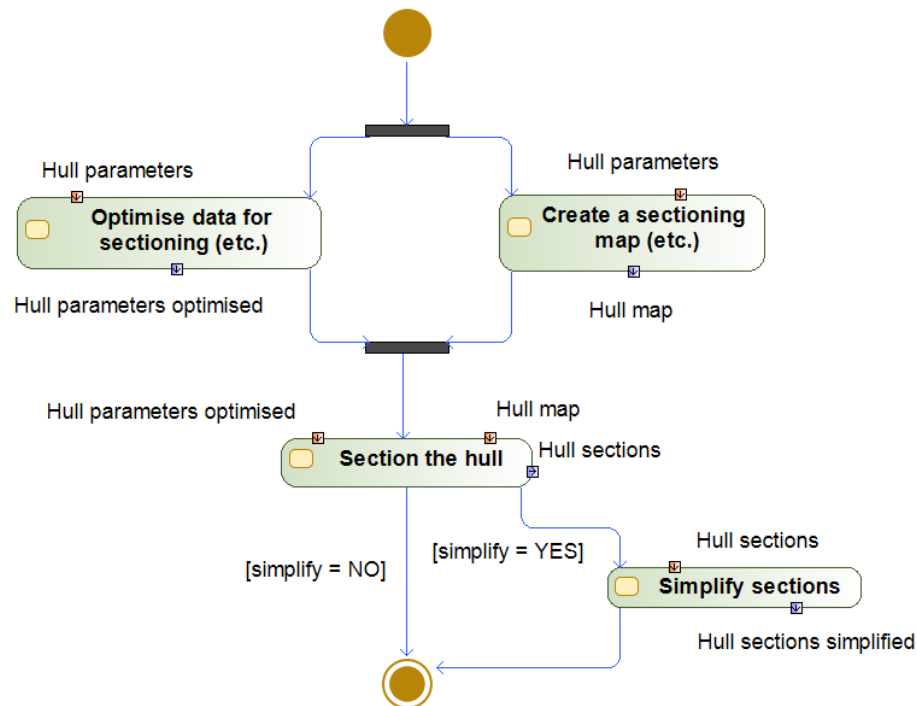
*Figure 8. CAL Example – main program*

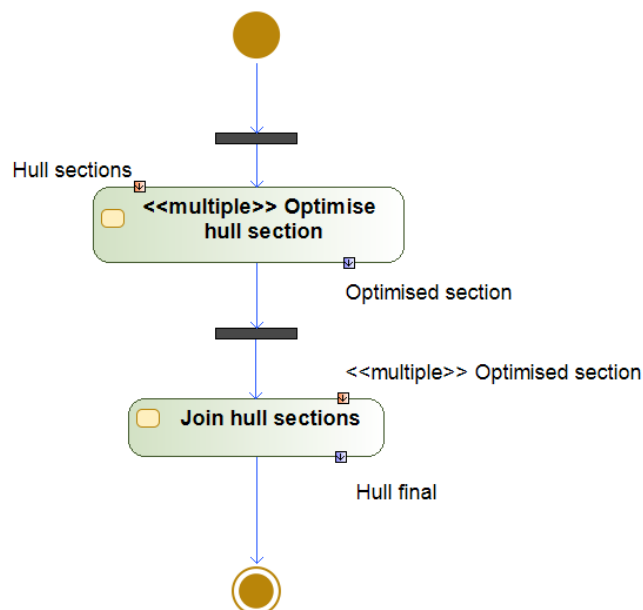*Figure 9. CAL Example – "Section the hull"*



*Figure 10. CAL Example – "Optimise the hull"*
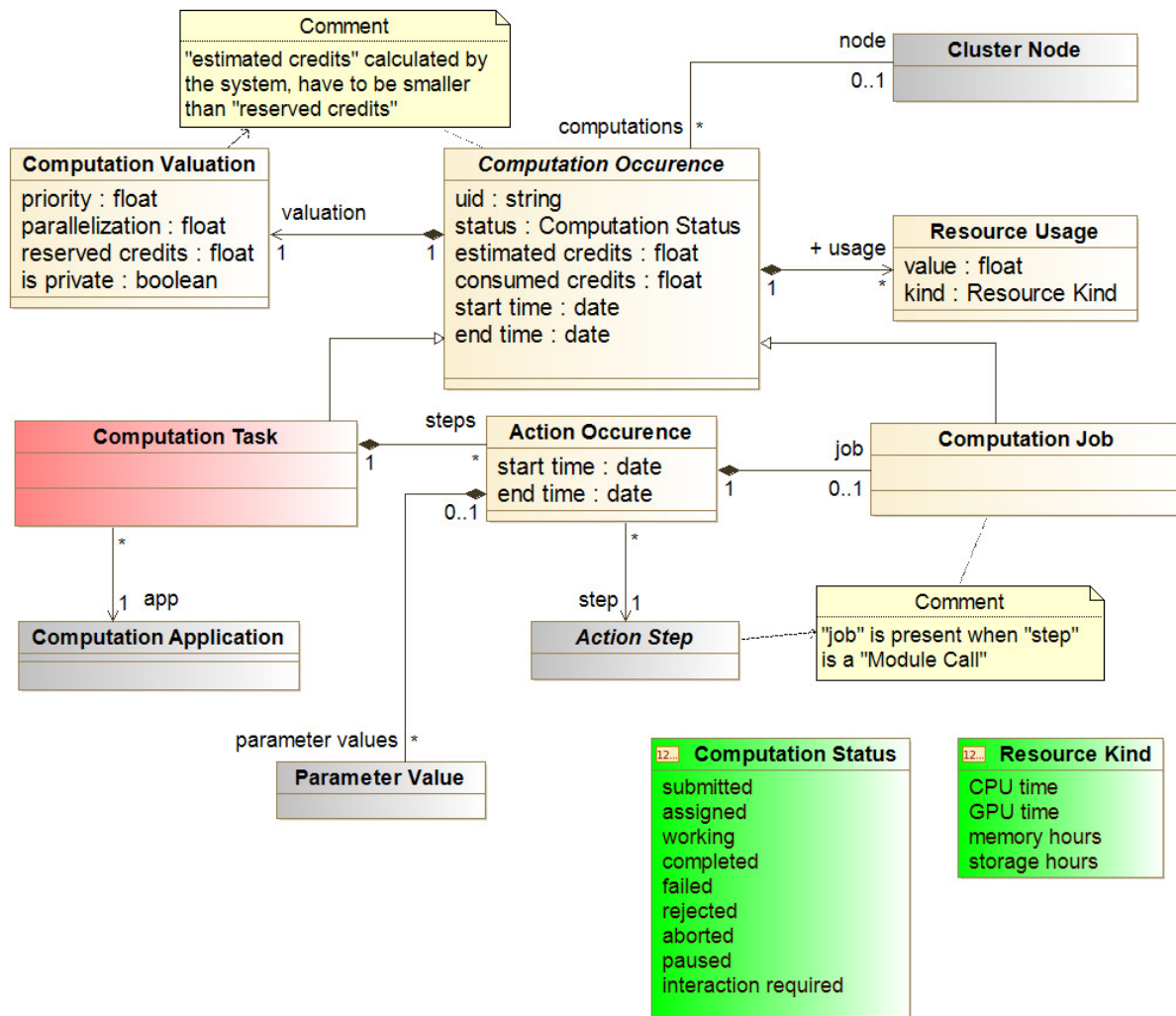
## 2.4 Computation Execution and Billing



*Figure 11. Class Diagram for Computation Execution and Billing*

### Computation Occurence

A generic type that defines parameters for Computation Tasks and Computation Jobs. Each Computation Occurrence has a Computation Valuation to determine priorities for the given occurrence. It also records its Resource Usage, changes its status according to computation progress and records credit consumption. Normally, each Computation Occurrence is assigned to a specific Cluster Node for its execution.

### Computation Task

An instance of a Computation Application and kind of Computation Occurrence. It serves recording all the Action Occurrences associated with the given instance of computations.

### Action Occurence

An instance of Action Step within a Computation Application. It records start and end time and specific parameter values for the given action. In case of Action Steps being Module Calls, it also records the Computation Job.

### Computation Job

A kind of Computation Occurrence that is associated with a specific Module Call. Computation Jobs are instantiated along execution of Module Calls and assigned to specific Cluster Nodes for execution.

**Computation Valuation**

A set of parameters that determine priorities for the given Computation Occurrence. Based on these parameters, the occurrences are brokered to nodes of varying computation power.

**Resource Usage**

A record of usage of a specific resource of a specific Resource Kind.
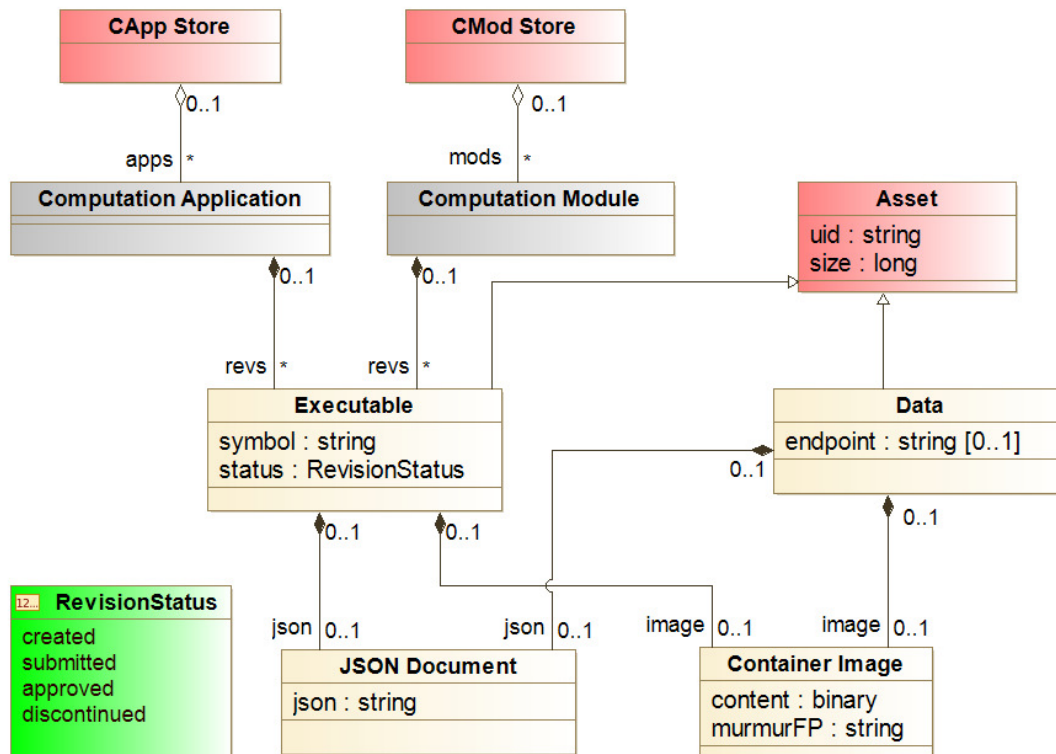
## 2.5  Assets and Stores



*Figure 12. Class Diagram for Assets and Stores*

**CApp Store**

A central (but possibly distributed) store that contains verified Computation Applications that can be used by the users to conduct their computations.

**CMod Store**

A central (but possibly distributed) store that contains verified Computation Modules that can be used by the users to write their Computation Applications.

**Asset**

Some data or executable that is stored in the asset repository within the system. It defines its unique ID and size.

**Executable**

An asset that contains an executable unit for execution within the system. It can be included as part of a Computation Application or a Computation Module. Normally, in the first case, the executable contains a JSON Document, and in the second case – a Container Image. Each executable specifies the symbol for the current revision and its Revision Status.

## Data

An asset that contains some data used for processing within the system. If it contains some parameters, then these are held with a JSON Document, if it contains generic data, then it is held in a Container Image.

## JSON Document

A file containing a JSON-based specification for some asset.

## Container Image

A file containing a container-based content for some asset.

## 2.6  Computation Accounts

*Figure 13. Class Diagram for Computation Accounts*

## Computation Account

A record for a specific user of BalticLSC. It gathers all the information about the user and determines the user's access rights.

### Resource Shelf

Gathers all the information about Nodes owned and managed by the given Computation Account, including Node Price Policies (structure of the policies not certain at this point – no business models available).

### Computation Cockpit

Gathers all the information regarding computations performed by the given Computation Account. This includes the App Shelf that gathers all the Computation Applications chosen the user and Computation Rack that gathers all the Computation Tasks run by the user.

### Development Shelf

Gathers all the information regarding development activities for the given Computation Account. This includes the developed Computation Applications and Computation Modules.

### CA Request

A request to develop/update some specific application.
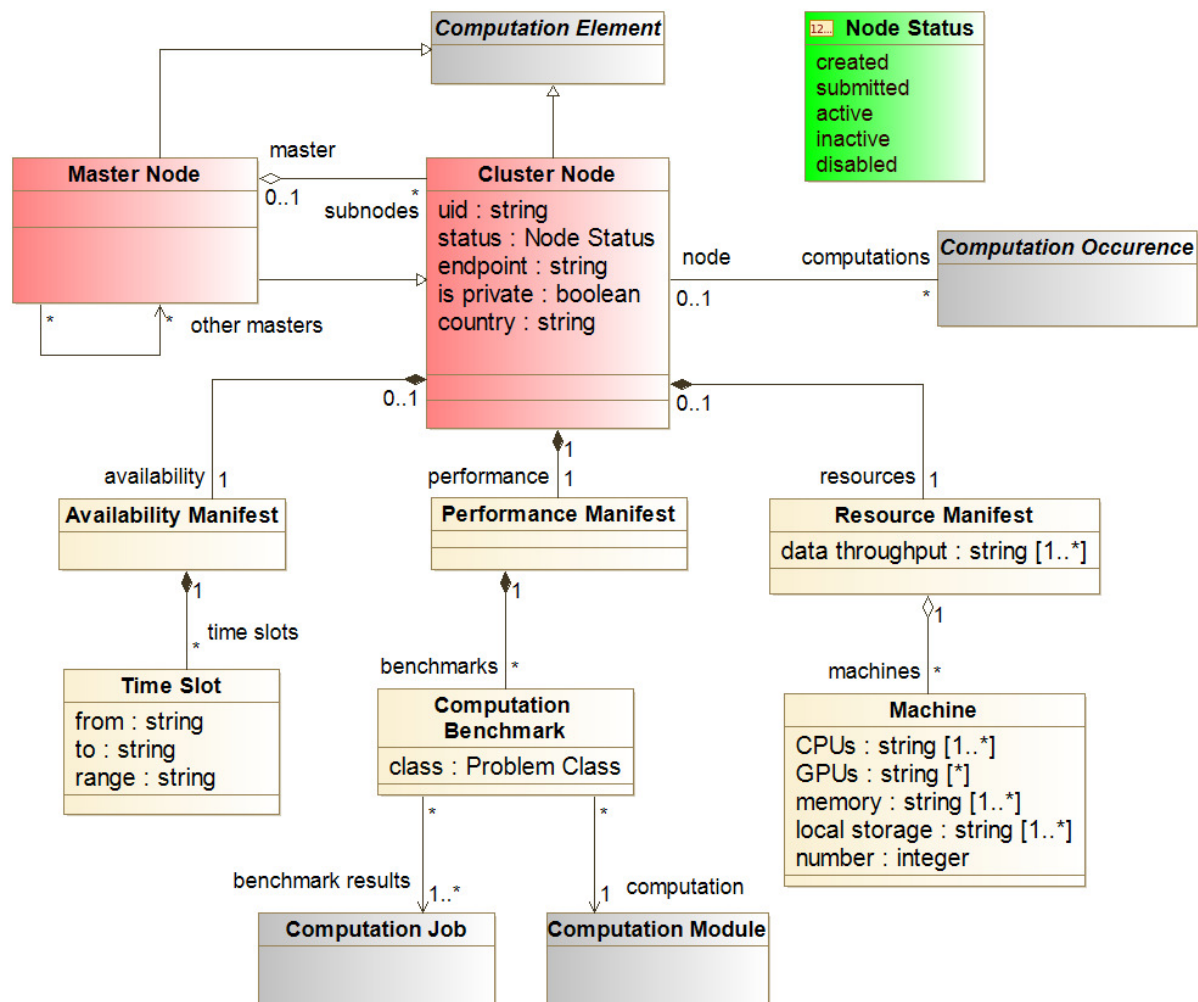
## 2.7  Computation Resources



*Figure 14. Class Diagram for Computation Resources*

## Cluster Node

A register entry with information about a specific Cluster Node (a set of machines organized into a computation cluster). The Cluster Node gathers information on availability, performance and resources (machines) declared for the described cluster.

## Master Node

A distinguished Cluster Node that gathers several subnodes and communicates with other Master Nodes to organize the network.

## Availability Manifest

A description of available Time Slots for a given Cluster Node. The Time Slots perhaps need to have more information than just "from-to" (e.g. what if a running job exceeds the "to"?).

## Performance Manifest

A set of Computation Benchmark results defining the computation power of a given Computation Node.

## Resource Manifest

A set of machine declarations, available as part of the given Cluster Node.