

# API-Refrence

---

## Agent - Ontology communication

### Resource Agents

```
# Activate RA
POST http://api.adress/resource/activate/{resourceId}

# Deactivate RA
POST http://api.adress/resource/deactivate/{resourceId}

# Add RA POST http://api.adress/resource/add
body:{
  "name": "string",
  "resourceType": "string",
  "neighbours": [
    "nameOfNeighbour1","nameOfNeighbour2","etc",
  ],
  "capabilities": [
    "Process1","Process2","etc"
  ],
  "isActive": true
}

# Get RA Status
GET http://api.adress/resource/getStatus/{resourceId}
```

### Product Agents

```
# Add PA
POST http://api.adress/spec/add
body: {
  "id": "someUuidOrDistinctName",
  "specification": "Class the spec is an instance of",
  "deadline": "2022-04-28T11:24:21.478Z",
  "features": ["feat1", "feat2", "etc"],
  "objectiveFunction": {
    "timeCoeff": 0,
    "costCoeff": 0,
    "emissionCoeff": 0,
    "qualityCoeff": 0
  }
}

# Delete feature from PA
POST http://api.adress/{spec}/deleteFeature/{feature}
```

```
# Add feature to PA
POST http://api.adress/{spec}/addFeature/{feature}

# Get PA Status
GET http://api.adress/spec/getStatus/{spec}
```

## Automata for initialization

```
# PA Automata
GET http://api.adress/pa/getAutomata

# PA Automata with recommended optimal path
GET http://api.adress/pa/getAutomata/withRecommendation

# RA Automata
GET http://api.adress/ra/getAutomata
```

*All calls contain automatas for all RAs/PAs. Getting Agent-specific Automata could be easily implemented if required*

## Process Executions

```
# Propose new Execution
POST http://api.adress/processExecutions/addProposal
body:{
  "resource": "theResourceTheProcessRunsIn",
  "process": "processToExecute",
  "specification": "specificationInvolved",
  "plannedStartTime": "2022-04-28T11:40:52.875Z",
  "plannedEndTime": "2022-04-28T11:40:52.875Z",
  "proposalGroup": "stringThatAllowsGroupingOfDifferentProposals"
}
# returns executionId as string

# Select Proposal
POST http://api.adress/processExecutions/selectProposal/{executionId}
# This changes the status of the proposal from proposed to planned and
checks if the proposal is executable

# Change Status to Running
POST http://api.adress/processExecutions/executionRunning/{executionId}

# Complete Execution with Success
POST
http://api.adress/processExecutions/executionSuccessful/{executionId}
body: {
  "performance": {
    "emissions": 0,
```

```
    "quality": 0,  
    "costs": 0  
  },  
  "realStartTime": "2022-04-28T11:46:06.973Z",  
  "realEndTime": "2022-04-28T11:46:06.973Z"  
}  
  
# Complete Execution with Error  
POST http://api.adress/processExecutions/executionErrored/{executionId}  
body: {  
  "errorMessage": "string",  
  "restartOfSpecification": true,  
  "restartOfProcess": true  
}
```