How to create a knowledge base for Simulator

Created on 2013/09/20

Modified on 2013/10/02

Fortune Truong ([fortune.truong@gmail.com](mailto:fortune.truong@gmail.com))

# 

**Table of Contents**

[Step 1: Create asset-method relationship](#h.m3jlzh9lzapc)

[Overview](#h.34j0yfcaq7bk)

[Outcome](#h.3cjz72yp2vgb)

[Scenario](#h.37ak24zih0ff)

[Step 2: Create precondition and postcondition for a method](#h.7fxnlw1nph6)

[Overview](#h.wykaczebljtn)

[Outcome](#h.pjp7zrlpm72l)

[Scenario](#h.6uau99rsl4d4)

[Step 3: Create initial state and world state for an asset](#h.mh7dcq7rzf33)

[Overview](#h.6knzxy8ot032)

[Outcome](#h.98e6yyn5qnuy)

[Scenario](#h.wst528y2e51n)

[Appendix 1: Problem Troubleshooting](#h.ci9yg2iyiuz6)

[Problem 1](#h.krz9nq1h5fu7)

[Problem 2](#h.3h74faqtnx3)

[Problem 3](#h.13tvbnv0mk7n)

[Appendix 2: Trauma center knowledge base implemetation](#h.akoo5yeopcgd)

# 

# 

# 

# Step 1: Create asset-method relationship

## Overview

* This section shows how to set up the most basic element of the knowledge base, the asset-method relationship.
* Purpose: the system will know which methods (or actions) belongs to an asset, and what is the possible options of that methods upon that asset. In object oriented term, it means what Methods an Object has, and what are the parameters.
* Variations: methods with 2 parameters and 1 parameter.
  + 2 parameters: i.e., move STH from A to B, or change STH from X to Y. This kind of method is natively supported in the system.
  + 1 parameter: i.e., submit a report to Dr R, make a sandwich for Ms Q. In those cases, the method does not really need a precondition, in other word the precondition is always be true. In order to implement those methods in the system, try to make them become a 2 parameter methods by create a fake parameter 1. For instance:
    - A method of “Add Bill blood to sample tube” does not really require any world-state precondition. “Blood tube contains none blood” is a “*always true”* parameter 1, could be used to make the method become 2-parameter method.
    - Identify the terms:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | *(Fake)* | *(Fake)* |
| **Statement** | Add | Bill blood | to | sample tube |  |  |
| **Knowledge base role** | Method | parameter 2 | Predicate | Asset |  |  |
| **Adding fake parameter** | Add | Bill blood | to | sample tube | contains | none |
| **Knowledge base role** | Method | parameter 2 | Predicate 2 | Asset | Predicate 1 | parameter 1 |

* + - From the table, “contains none” is an “always true” parameter which the knowledge base designer implicitly adds to make the 1 parameter statement to become a 2 parameter.

## Outcome

* BE ABLE TO let the user select a method from available methods of an asset, then select the expected parameter for that method.
* NOT BE ABLE TO check the pre and postcondition, the system may show an error after submit the parameter. To troubleshoot this issue, see [Step 2: Create precondition and postcondition](#h.xdnp7wtz7lc)
* NOT BE ABLE TO show the current state of the asset, the system may show an empty box in “Current state of Asset”. To troubleshoot this issue, see [Step 3: Create initial state for an asset](#h.mh7dcq7rzf33) and [Step 4: Create the current world state for an asset](#h.tigrw3nfp13c)

## 

## 

## Scenario

For ease of understanding this step, a scenario will be fully described as following: System simulates a case of moving a patient from Emergency Room to Examination Room in a hospital virtual environment.

|  |  |
| --- | --- |
|  |  |

From the scenario description, explicit data can be extracted are:

* Asset: Patient
* Method: Move\_bed
* Initial location: Emergency Room
* Destination location: Examination Room

And some implicit data which need to be put into account:

* All locations, such as Examination Room, Emergency Room, Intensive Care Room, are put in a domain, named Bed\_Locations.
* Identifier: Two parameters for the method are the place where the bed is and where it need to be moved to. They are [Bed\_From] and [Bed\_To]. It can be imagined by Move\_Bed(Bed\_From, Bed\_To).
* Predicate: Describe the relationship between asset and domain. In this case, [Patient] is [AT] a certain [Bed\_Location], so a predicate name [Bed\_At] is quite appropriate predicate.

From the ablow analysis, the following data need to be generated in database:

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Patient] |
| 2 | method name | activity\_methods | [Move\_Bed] |
| 3 | asset name, method name | asset\_methods | [Patient, Move\_Bed] |
| 4 | domain name | domain | [Bed\_Locations] |
| 5 | domain name, domain value | domain\_values | [Bed\_Locations, Emerg\_Room]  [Bed\_Locations, Exami\_Room] |
| 6 | domain name, identifier | variables | [Bed\_Locations, Bed\_From]  [Bed\_Locations, Bed\_To] |
| 7 | method name, variable | method\_parameters | [Move\_Bed, Bed\_From]  [Move\_Bed, Bed\_To] |
| 8 | predicate | predicate\_labels | [Bed\_At] |
| 9 | predicate, variable | predicate\_paramenters | [Bed\_At, Bed\_From]  [Bed\_At, Bed\_To] |

## 

# Step 2: Create precondition and postcondition for a method

## Overview

* This section shows how to set up precondition and postcondition for the knowledge base.
* Purpose:
  + Precondition describes what will be checked before the method execution, to make sure the world-state is fulfill the method.
  + Postcondition describes how the world-state need be modified after the effect of the method execution.

## Outcome

* BE ABLE TO check whether the method is executable upon the current world state.
* BE ABLE TO let the Simulator know what need to be modified in the knowledge base.

## Scenario

Take “Move\_Bed” as a scenario since it’s common and simple. If Bed need to be moved from position A to B, precondition of it is “Bed\_At A”, and postcondition is “NOT Bed\_At A” and “Bed\_At B”.

Hint: 0 (ZERO) and 1 (ONE) are used to describe “False” and “True” in postcondition table.

From the ablow analysis, the following data need to be generated in database:

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 to 9 | *(continued from the previous scenario)* | *(continued from the previous scenario)* | *(continued from the previous scenario)* |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Move\_Bed, Bed\_At, Bed\_From] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Move\_Bed, Bed\_At, Bed\_From, 0]  [Move\_Bed, Bed\_At, Bed\_To, 1] |

## 

# 

# 

# Step 3: Create initial state and world state for an asset

## Overview

* This section shows how to tell the simulator know what state an asset supposed to be when the simulation start, and the current state of an asset.
* Purpose:
  + The simulator use initial state to force the virtual world change all asset states to the original state.
  + On the other hand, the world state tell the simulator the current state of all assets, which used for the precondition validation.

## Outcome

* BE ABLE TO let every assets return to initial state, prepare for the simulation.
* BE ABLE TO let the Simulator check the precondition.

## Scenario

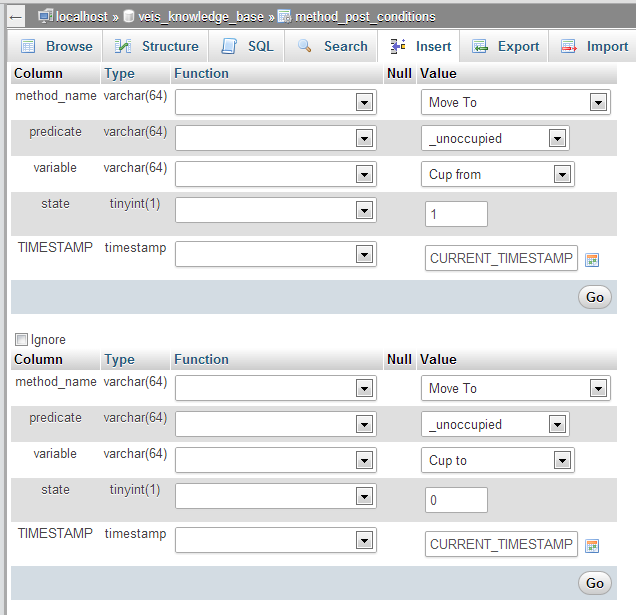
Keep the same scenario, “Move\_Bed”, as the other step for consecutive connection. At the beginning of the simulation, the Bed need to be located at “Emerg\_Room”, and the world state also let the simulation know it is at “Emerg\_Room”.

Hint: world\_states is NOT in the same database as the knowledge base, it is in veis\_world\_states database.

From the ablow analysis, the following data need to be generated in database:

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 to 11 | *(continued from the previous scenario)* | *(continued from the previous scenario)* | *(continued from the previous scenario)* |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Patient, Bed\_At, Bed\_From, Emerg\_Room] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Patient, Bed\_At, Emerg\_Room] |

Optional: Setup MOVABLE object: modify table method\_service\_calls

Optional: Setup a \_unoccupied post condition for moving abject 

# Appendix 1: Problem Troubleshooting

## Problem 1

System show both parameters in the case it just need to show the second one.

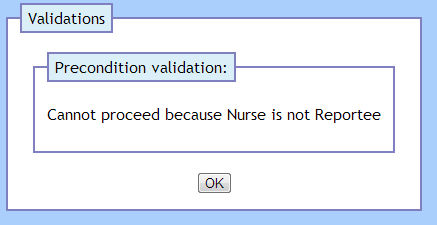
|  |  |
| --- | --- |
| **How it looks like** | **How it supposed to be** |
|  |  |

Solution 1: Check and fix the precondition in precondition table.

Solution 2: Check and fix the current state in world state table.

## Problem 2

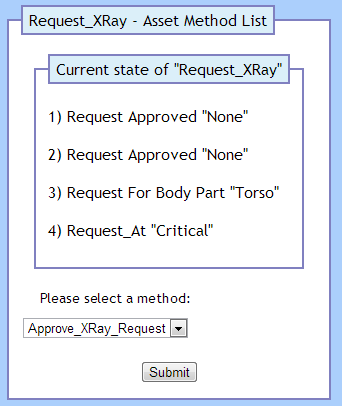
Cannot proceed because [Something] is not [Something]



Solution: Check and fix precondition table

## Problem 3

Duplicate states appeared



Solution 1: Check and fix method\_precondition table, may be one method has a duplicated precondition.

Solution 2: If method\_precondition is alright, it maybe because another method / asset used the same predicate. I.E. Move\_Bed and Move\_Truck, instead of having “At” as a shared predicate, they need to have two different predicate as “Bed\_At” and “Truck\_At”.

# Appendix 2: Trauma center knowledge base implemetation

This appendix shows some cases set up the Trauma center scene.

Case 1: Move bed

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Patient] |
| 2 | method name | activity\_methods | [Move Bed] |
| 3 | asset name, method name | asset\_methods | [Patient, Move bed] |
| 4 | domain name | domain | [bed\_locations] |
| 5 | domain name, domain value | domain\_values | [bed\_locations, EmergencyR]  [bed\_locations, ExaminationR]  [bed\_locations, IntensiveCare]  [bed\_locations, RadiologyR] |
| 6 | domain name, identifier | variables | [bed\_locations, Bed from]  [bed\_locations, Bed To] |
| 7 | method name, variable | method\_parameters | [Move bed, Bed from]  [Move bed, Bed to] |
| 8 | predicate | predicate\_labels | [Bed\_at] |
| 9 | predicate, variable | predicate\_paramenters | [Bed\_at, Bed from  [Bed\_at, Bed to] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Move bed, Bed\_at, Bed from] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Move bed, Bed\_at, Bed from, 0]  [Move bed, Bed\_at, Bed to, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Patient, Bed\_at, Bed From, EmergencyR] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Patient, Bed\_At, EmergencyR] |

Case 5: Do report the examination report to nurse. Note that it is a 1 parameter method, which requires a fake parameter is “reportee from: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [ExamReport] |
| 2 | method name | activity\_methods | [Add\_Report] |
| 3 | asset name, method name | asset\_methods | [ExamReport, Add\_Report] |
| 4 | domain name | domain | [Actor] |
| 5 | domain name, domain value | domain\_values | [Actor, None]  [Actor, Doctor]  [Actor, Nurse] |
| 6 | domain name, identifier | variables | [Actor, Reportee from]  [Actor, Reportee to] |
| 7 | method name, variable | method\_parameters | [Add\_Report, Reportee from]  [Add\_Report, Reportee to] |
| 8 | predicate | predicate\_labels | [Report\_To] |
| 9 | predicate, variable | predicate\_paramenters | [Report\_To, Reportee from]  [Report\_To, Reportee to] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Add\_Report, Report\_To, Reportee from] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Add\_Report, Report\_To, Reportee from, 0]  [Move bed, Report\_To, Reportee to, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [ExamReport, Report\_To, Reportee from, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [ExamReport, Report\_To, None] |

## 

## 

Case 7: Make a pathology request for patient Bill. Note that it is a 1 parameter method, which requires a fake parameter is “Make rof: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Request\_Pathology] |
| 2 | method name | activity\_methods | [Make\_Request] |
| 3 | asset name, method name | asset\_methods | [Request\_Pathology, Make\_Request] |
| 4 | domain name | domain | [Patient\_list] |
| 5 | domain name, domain value | domain\_values | [Patient\_list, None]  [Patient\_list, Bill Gate]  [Patient\_list, Steve Job] |
| 6 | domain name, identifier | variables | [Patient\_list, Make rof]  [Patient\_list, Make for] |
| 7 | method name, variable | method\_parameters | [Make\_Request, Make rof]  [Make\_Request, make for] |
| 8 | predicate | predicate\_labels | [For Patient] |
| 9 | predicate, variable | predicate\_paramenters | [For Patient, Make rof]  [For Patient, Make for] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Make\_Request, For Patient, Make rof] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Make\_Request, For Patient, Make rof, 0]  [Make\_Request, For Patient, Make for, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Request\_Pathology, For Patient, Make rof, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Request\_Pathology, For Patient, None] |

## 

Case 11: Make a XRay request for a part of a patient. Note that it is a 1 parameter method, which requires a fake parameter is “Make rof: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Request\_XRay] |
| 2 | method name | activity\_methods | [Make\_XRay\_Request] |
| 3 | asset name, method name | asset\_methods | [Request\_XRay, Make\_XRay\_Request] |
| 4 | domain name | domain | [Patient\_part] |
| 5 | domain name, domain value | domain\_values | [Patient\_part, None]  [Patient\_part, Torso]  [Patient\_part, Head] |
| 6 | domain name, identifier | variables | [Patient\_part, Body part rof]  [Patient\_part, Body part for] |
| 7 | method name, variable | method\_parameters | [Make\_XRay\_Request, Body part rof]  [Make\_XRay\_Request, Body part for] |
| 8 | predicate | predicate\_labels | [Request For Body Part] |
| 9 | predicate, variable | predicate\_paramenters | [Request For Body Part, Body part rof]  [Request For Body Part, Body part for] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Make\_XRay\_Request, Request For Body Part, Body part rof] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Make\_XRay\_Request, Request For Body Part, Body part rof, 0]  [Make\_XRay\_Request, Request For Body Part, Body part for, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Request\_XRay, Request For Body Part, Body part rof, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Request\_XRay, Request For Body Part, None] |

Case 15: Make a XRay approval for a part of a patient. Note that it is a 1 parameter method, which requires a fake parameter is “Approve rof: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Request\_XRay] |
| 2 | method name | activity\_methods | [Approve\_XRay\_Request] |
| 3 | asset name, method name | asset\_methods | [Request\_XRay, Approve\_XRay\_Request] |
| 4 | domain name | domain | [Patient\_part] |
| 5 | domain name, domain value | domain\_values | [Patient\_part, None]  [Patient\_part, Torso]  [Patient\_part, Head] |
| 6 | domain name, identifier | variables | [Patient\_part, Body part rof]  [Patient\_part, Body part for] |
| 7 | method name, variable | method\_parameters | [Approve\_XRay\_Request, Body part rof]  [Approve\_XRay\_Request, Body part for] |
| 8 | predicate | predicate\_labels | [Request Approved] |
| 9 | predicate, variable | predicate\_paramenters | [Request Approved, Body part rof]  [Request Approved, Body part for] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Approve\_XRay\_Request, Request Approved, Body part rof] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Approve\_XRay\_Request, Request Approved, Body part rof, 0]  [Approve\_XRay\_Request, Request Approved, Body part for, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Request\_XRay, Request Approved, Body part rof, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Request\_XRay, Request Approved, None] |

Case 18: Move XRay Machine from middle to top

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Machine\_XRay] |
| 2 | method name | activity\_methods | [Move\_XRayMachine] |
| 3 | asset name, method name | asset\_methods | [Machine\_XRay, Move\_XRayMachine] |
| 4 | domain name | domain | [XRay\_locations] |
| 5 | domain name, domain value | domain\_values | [XRay\_locations, XRay\_Middle]  [XRay\_locations, XRay\_Top] |
| 6 | domain name, identifier | variables | [XRay\_locations, XRay from]  [XRay\_locations, XRay to] |
| 7 | method name, variable | method\_parameters | [Move\_XRayMachine, XRay from]  [Move\_XRayMachine, XRay to] |
| 8 | predicate | predicate\_labels | [XRay\_At] |
| 9 | predicate, variable | predicate\_paramenters | [XRay\_At, XRay from]  [XRay\_At, XRay to] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Move\_XRayMachine, XRay\_At, XRay from] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Move\_XRayMachine, XRay\_At, XRay from, 0]  [Move\_XRayMachine, XRay\_At, XRay to, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Machine\_XRay, XRay\_At, XRay From, XRay\_Top] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Machine\_XRay, XRay\_At, XRay\_Top] |

## 

## 

Case 19: Do XRay. Note that it is a 1 parameter method, which requires a fake parameter is “Body part rof: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Machine\_XRay] |
| 2 | method name | activity\_methods | [Do\_XRay] |
| 3 | asset name, method name | asset\_methods | [Machine\_XRay, Do\_XRay] |
| 4 | domain name | domain | [Patient\_part] |
| 5 | domain name, domain value | domain\_values | [Patient\_part, None]  [Patient\_part, Torso]  [Patient\_part, Head] |
| 6 | domain name, identifier | variables | [Patient\_part, Body part rof]  [Patient\_part, Body part for] |
| 7 | method name, variable | method\_parameters | [Do\_XRay, Body part rof]  [Do\_XRay, Body part for] |
| 8 | predicate | predicate\_labels | [XRay For Body Part] |
| 9 | predicate, variable | predicate\_paramenters | [XRay For Body Part, Body part rof]  [XRay For Body Part, Body part for] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Do\_XRay, XRay For Body Part, Body part rof] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Do\_XRay, XRay For Body Part, Body part rof, 0]  [Do\_XRay, XRay For Body Part, Body part for, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Machine\_XRay, XRay For Body Part, Body part rof, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Machine\_XRay, XRay For Body Part, None] |

Case 22: Do report the XRay report to Pathology. Note that it is a 1 parameter method, which requires a fake parameter is “reportee from: none”

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [Report\_XRay] |
| 2 | method name | activity\_methods | [Add\_XRay\_Report] |
| 3 | asset name, method name | asset\_methods | [Report\_XRay, Add\_XRay\_Report] |
| 4 | domain name | domain | [Actor] |
| 5 | domain name, domain value | domain\_values | [Actor, Doctor]  [Actor, Nurse] |
| 6 | domain name, identifier | variables | [Actor, Reportee from]  [Actor, Reportee to] |
| 7 | method name, variable | method\_parameters | [Add\_XRay\_Report, Reportee from]  [Add\_XRay\_Report, Reportee to] |
| 8 | predicate | predicate\_labels | [XRay\_Report\_To] |
| 9 | predicate, variable | predicate\_paramenters | [XRay\_Report\_To, Reportee from]  [XRay\_Report\_To, Reportee to] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Add\_XRay\_Report, XRay\_Report\_To, Reportee from] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Add\_XRay\_Report, XRay\_Report\_To, Reportee from, 0]  [Add\_XRay\_Report, XRay\_Report\_To, Reportee to, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [Report\_XRay, XRay\_Report\_To, Reportee from, None] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [Report\_XRay, XRay\_Report\_To, None] |

## 

Case 00: Move the coffee from the table to the machine

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [CoffeeCup] |
| 2 | method name | activity\_methods | [Move\_To] |
| 3 | asset name, method name | asset\_methods | [CoffeeCup, Move\_To] |
| 4 | domain name | domain | [CoffeePosition] |
| 5 | domain name, domain value | domain\_values | [CoffeePosition, Table]  [CoffeePosition, Machine] |
| 6 | domain name, identifier | variables | [CoffeePosition, Cup\_from]  [CoffeePosition, Cup\_to] |
| 7 | method name, variable | method\_parameters | [Move\_To, Cup\_from]  [Move\_To, Cup\_to] |
| 8 | predicate | predicate\_labels | [CoffeeMoveTo] |
| 9 | predicate, variable | predicate\_paramenters | [CoffeeMoveTo, Cup\_from]  [CoffeeMoveTo, Cup\_to] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Move\_To, CoffeeMoveTo, Cup\_from] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Move\_To, CoffeeMoveTo, Cup\_from, 0]  [Move\_To, CoffeeMoveTo, Cup\_to, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [CoffeeCup, CoffeeMoveTo, Cup\_from, Table] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [CoffeeCup, CoffeeMoveTo, Table] |

## 

## 

Case 01: Fill the coffee cup from the machine

|  |  |  |  |
| --- | --- | --- | --- |
| **Step** | **Insert** | **Into table** | **Examples** |
| 1 | asset name | asset | [CoffeeCup] |
| 2 | method name | activity\_methods | [Fill\_Coffee] |
| 3 | asset name, method name | asset\_methods | [CoffeeCup, Fill\_Coffee] |
| 4 | domain name | domain | [CoffeeCupState] |
| 5 | domain name, domain value | domain\_values | [CoffeeCupState, Empty]  [CoffeeCupState, Coffee] |
| 6 | domain name, identifier | variables | [CoffeeCupState, Is]  [CoffeeCupState, IsFill] |
| 7 | method name, variable | method\_parameters | [Fill\_Coffee, Is]  [Fill\_Coffee, IsFill] |
| 8 | predicate | predicate\_labels | [CoffeeFill] |
| 9 | predicate, variable | predicate\_paramenters | [CoffeeFill, Is]  [CoffeeFill, IsFill] |
| 10 | method\_name,  predicate,  variable | method\_preconditions | [Fill\_Coffee, CoffeeFill, Is] |
| 11 | method\_name,  predicate,  variable,  state | method\_post\_condition | [Fill\_Coffee, CoffeeFill, Is, 0]  [Fill\_Coffee, CoffeeFill, IsFill, 1] |
| 12 | asset\_name,  predicate,  variable\_name,  value | asset\_initial\_state | [CoffeeCup, CoffeeFill, Is, Empty] |
| 13 | asset\_name,  predicate\_label,  value | veis\_world\_states.world\_states | [CoffeeCup, CoffeeFill, Empty] |

## 