Input data specification for PARS

Definition:

“input”: Xml based input information into the PARS system

User-Requirements:

1. All input data should be parametrized and not require any programming skills
2. Define spacial points and attach 3D Holograms to them
3. Define Events and their eventhandler from a certain range of events
4. Define the robot’s environment incl. a set of 3D models
5. Possibility to create processes and step patterns

Additional things to think about:

* Does the system need a defined starting state or not?
  + If not: Runtime manipulation of the input would be possible

System Requirements:

|  |  |  |
| --- | --- | --- |
| Nr | Targeted User-Rqmt | Explanation |
| 11 | 1 | Create an input data generator that allows the selection of points and outputs xml-based information |
| 12 | 2 | Have a point definition part in the input that allows the parametrized definition of spacial points |
| 13 | 2 | Have a Hologram definition part in the input that allows the coupling between 3D models and defined points   * Define three points in space and in model and mount it accordingly * Define One origin coord frame mount it accordingly |
| 14 | 3 | Set of pre-defined events that take some input information and have more than one event handler defined |
| 15 | 3 | Set of pre-defined event handlers that allow actions on the robot and on the VR system |
| 16 | 5 | Have a int counter part in the input data that allows it’s manipulation through events and can execute event handlers |
| 17 | 5 | String Values to show in UI at one specific place |

Architecture design:

Input data structure definition

InputData:

* Variables
  + IntVariable  
    *Var to be used for counters*
    - Value : int
    - Name
  + StringVariable
    - Name
    - Value
* SpacialDefinition (Unique Name)
  + PointDefintionFix : IPoint
    - *Some 3D point definition to e.g. mount static models*
    - X,Y,Z
  + PointDefintionRobot : IPoint
    - *3D point that is between two Joints. ScaleValue = 0 -> Joint1, ScaleValue = 1 -> Joint 2 and everything in between.*
    - Joint1, Joint2, ScaleValue
  + CoordFrameDefinitionAngle : ICoordFrame
    - IPoint, Alpha, Beta, Gamma
  + CoordFrameDefinitionIPoint : ICoordFrame
    - IPoint, IPoint, IPoint, IPoint
* HologramDefintions
  + HologramDefinition\_Sphere:
    - IPoint, Radius
  + HologramDefinition\_Zylinder:
    - IPoint, IPoint, Radius
  + HologramDefinition\_General
    - Model, ICoordFrame
* UIText
  + UITextField
    - StringVariable
* Events:
  + TriggerDefinitions (EventHandler1, EventHandler2)

*Events can only trigger once directly but can be reset by an event handler*

* + - TimeTrigger
      * TimeSpan
    - DistanceTrigger
      * IPoint
      * IPoint
      * Distance
    - VarTrigger
      * IntVarName *ToCheck*
      * IntVarName *TriggerValue*
  + EventHandlerDefintions:
    - IncrementCounterEventHandler

*Increments the given counter*

* + - * Counter
    - ChangeStringValueEventHandler
      * StringValueVarName
      * NewValue
    - ResetEventEventHandler

*Resets the given Event so that it can trigger again*

* + - * EventID
    - MoveRobotEventHandler
      * IPoint
    - ChangeModelEventHandler

*Changes Model definitions but uses the same IPoints*

* + - * HologramDefinitionID
      * NewModel