Documentation of PAR database (openpardb)

October 2015

This database consists of 11 largely unpopulated tables. The tables can be divided into 2 groups: action tables and object tables. There are two tables that link them, obj_act and obj_capable. In this document, we will describe the tables and their entries.

Contents

1 Actionary

The following tables are mainly related to action PARs and are meant to store both uPARs and iPARs.

1.1 action

This table is the main table related to the action part of PARs. It is linked to other tables to save on storage costs.

Column name	Data type	Description
act_id	SMALLINT	The action id that is used to link this PAR to other tables including action properties and objects. [PRIMARY KEY]
act_name	VARCHAR	The name of the action should be similar to a verb.
act_app_cond	VARCHAR	The applicability conditions field could be a Python expression but is more likely a Python script filename. If the applicability conditions evaluate to false, the action will not be performed.
act_term_cond	VARCHAR	The termination conditions field could be a Python expression but is more likely a Python script filename. Termination conditions indicate when the action has been completed. It may be based on the duration of the action or the state of the world.
act_prep_spec	VARCHAR	The preparatory specifications are condition action pairs. The field could be a Python expression but is more likely a Python script filename.
act_exec_steps	VARCHAR	The executions steps (sub-steps) for this action. Actually a path and python file name that contains them.
act_purpose_achieve	VARCHAR	This field indicates the state of the environment that is trying to be achieved by performing the action. This field could be a Python expression or a Python script filename.

Column name	Data type	Description
parent_id	SMALLINT	This field holds the actions parents id. This is used mainly for inheritance.
act_dur_time_id	SMALLINT	The duration time field holds an id that is referenced in the time_spec table and refers to the duration of the action.
act_obj_num	TINYINT	The object number is the number of objects that are participants in this action. When used for uninstanced PARS, act_obj_num is the maximum amount of objects that an action can have.
act_site_type_id	SMALLINT	The id that indicates the type of site that should be used to properly locate the agent relative to the object participant of the action. For example, the position and orientation a character should stand in to clean a chair vs. to sit in a chair.
wordnet_sense	SMALLINT	The id of the corresponding wordnet frame, or -1 if one does not exist for the action.

1.2 adverb_exp

These fields hold adverbs and adverb modifiers that might be used to describe the action. These might then be used by the corresponding motion generator to modify the action performance accordingly. They might also effect planning (were it to exist).

Column name	Data type	Description
act_id	SMALLINT	The action id that is used to link this PAR to other tables including the main action table. [PRIMARY KEY]
adverb_name	VARCHAR	An adverb that would need to have meaning associated with it in a planner or motion generator.
adverb_mod_name	VARCHAR	An adverb modifier that would need to have meaning associated with it in a planner or motion generator. For example, move more quickly

2 Object Hierarchy

The following tables are mainly related to PAR objects and are meant to include both general object descriptions and specific objects. These also hold the current properties of the world.

2.1 obj_act

This table links PAR action with the PAR object participants.

Column name	Data type	Description
obj_id	SMALLINT	This is the associated objects id. There may be more than one object per action, meaning more than one object participating in a single action. Certainly there may also be more than one action id per object, meaning that a single object participates in more than one action. [PRIMARY KEY]
act_id	SMALLINT	This is the id of the action as referenced in the main action table.
obj_num	TINYINT	If an action has more than one object participant, then the order of the objects has implicit meaning about the type of participation. This field provides us with a means of determining that order aside from simple order in the table.

2.2 object

This is the main object table and is linked to many other object oriented tables.

Column name	Data type	Description
obj_id	SMALLINT	The obj_id that is used to link the information in this table to many of the other tables in the database. [PRIMARY KEY]
obj_name	VARCHAR	The name of the object. The convention is that names with underscores and numbers are instances and those with out are the general parents or classes. For example, Chair would contain general properties of a Chair and Chair_0 would contain information about an actual chair in the environment.
obj_agent	TINYINT	Indicated whether or not this object is also an agent (can execute actions). 1 = yes, 0 = no
parent_id	SMALLINT	The id of the parent object so that it can inherit the properties.
wordnet_sense	SMALLINT	The id of the corresponding wordnet frame, or -1 if one does not exist for the object.

2.3 obj_capable

This table associates objects with capabilities. For agents this is actions they are capable of performing (checked in applicability conditions). Note that there may be many entries for each object and action.

Column name	Data type	Description
obj_id	SMALLINT	The obj_id that is used to link the information in this table to other tables including the main object table. [PRIMARY KEY]
action_id	SMALLINT	Ids for the actions that the objects can participate in.

2.4 obj_prop

This table associates objects with properties. Properties can be rather general or generic. The meanings of the properties and their values come from how they are processed by functions, motion generators and planners. The values of the properties come from property tables, such as obj_status.

Column name	Data type	Description
prop_id	SMALLINT	The prop_id that is used to link the information in this table to the other tables in the database. [PRIMARY KEY]
obj_id	SMALLINT	The obj_id that is used to link the information in this table to the other tables in the database. [PRIMARY KEY]
table_name	VARCHAR	The name of the property.
prop_value	SMALLINT	The value of the property for this object. This is connected to the id of the table with the same name as table_name.

2.5 obj_status

This table associates objects with their statuses. The meanings of the statuses would come from how they are processed by functions, motion generators and planners. This is an example of a generalized property table.

Column name	Data type	Description
id_value	SMALLINT	The status_id that is used to link the information in this table to the other tables in the database. [PRIMARY KEY]
name_value	VARCHAR	The name of the status current status. For example, on or off.

2.6 property_type

This table contains the names of all known properties in the PAR system. Properties can be of two types, continuous integer or a set of string values.

Column name	Data type	Description
prop_id	SMALLINT	The unique property id for each system [PRIMARY KEY]
prop_name	VARCHAR	The name of the property. For properties that are created in the database and are sets (such as obj_status) the property must match the name of the table
is_int	TINYINT	If this is set to 1, then the property is an integer and does not have a corresponding set, otherwise, the corresponding set should be in the database.

2.7 site

This table associates objects with sites. In essence, sites are oriented points or regions that can be used by motion generators to enable agents to interact with objects.

Column name	Data type	Description
site_type_id	SMALLINT	An id for this site, linked to the site_type table [PRIMARY KEY]
obj_id	SMALLINT	The obj_id that is used to link the information in this table to the other tables in the database. This also links to actions to determine the placement of an agent relative to the object for doing the action. For example, where to stand when cleaning vs about to sit for a chair [PRIMARY KEY]
site_pos_x	FLOAT	The x position of the site.
site_pos_y	FLOAT	The y position of the site.
site_pos_z	FLOAT	The z position of the site.
site_orient_x	FLOAT	The x orientation of the site.
site_orient_y	FLOAT	The y orientation of the site.
site_orient_z	FLOAT	The z orientation of the site.
site_shape_id	SMALLINT	The shape of site represented. This links to site_shape.

2.8 site_type

This table associates sites with the kind of site. While not necessarily needed for an agent to operate, it does provide a way to categorize the connections between site definitions and actions.

Column name	Data type	Description
site_type_id	SMALLINT	An id for this type of site. Used in site as a key and action to define the type of site. [PRIMARY KEY]
site_name	VARCHAR	The human readable name of the site

2.9 site_shape

This table defines the shape of a region that is connected to a site. A region is a volumetric space that an action is meant to operate on. While not enforced, we generally consider three types of regions: spheres, cylinders, and boxes. Spheres only need one size defined,

cylinders two, and boxes three.

Column name	Data type	Description
site_shape_id	INT	An id for this region's shape, linked to the site table [PRIMARY KEY]
site_type	VARCHAR	The human readable type of the region (box, cylinder, or sphere).
first_coord	FLOAT	The numerical value of the first coordinate. Generally treated as a size.
second_coord	FLOAT	The numerical value of the second coordinate. Generally treated as a size.
third_coord	FLOAT	The numerical value of the third coordinate. Generally treated as a size.