**abstract\_unity\_linker**

**Subspecies of:** agent  
**Microspecies of:** model  
**Skills:** [network]

Attributes:

* **shape**: attribute of type geometry. Returns the shape of the receiver agent
* **location**: attribute of type point. Returns the location of the agent
* **name**: attribute of type string. Returns the name of the agent (not necessarily unique in its population)
* **index**: constant of type int. Returns the unique index of this agent in its population. Read-only attribute
* **host**: attribute of type model. Returns the agent that hosts the population of the receiver agent
* **peers**: attribute of type list. Returns the population of agents of the same species, in the same host, minus the receiver agent
* **members**: attribute of type container. Returns the list of agents for the population(s) of which the receiver agent is a direct host
* **agents**: attribute of type list. Returns the list of agents for the population(s) of which the receiver agent is a direct or undirect host
* **network\_name**: attribute of type string. Net ID of the agent
* **network\_groups**: attribute of type list. The set of groups the agent belongs to
* **network\_server**: attribute of type list. The list of all the servers to which the agent is connected
* **end\_message\_symbol**: attribute of type string. Symbol to be added at the end of the messages (only when the middleware is not used); it should be the same defined in Unity
* **distance\_player\_selection**: attribute of type float. Maximal distance to select a player agent
* **initialized**: attribute of type bool. Has the world being initialized yet?
* **geometries\_to\_keep**: attribute of type list. List of geometries to keep in Unity
* **background\_geometries**: attribute of type map. Map of background geometries to sent to Unity with the unity properties to use.
* **player\_species**: attribute of type string. Species of the player agent
* **attributes\_to\_send**: attribute of type map. List of attributes to sent to Unity
* **min\_num\_players**: attribute of type int. Number of Unity players required to start the simulation
* **new\_player\_position**: attribute of type map. The new poistion of the player to be sent to Unity - map with key: agent name, value: list of int [x,y]
* **receive\_information**: attribute of type bool. should GAMA receive information from Unity?
* **move\_player\_event**: attribute of type bool. Does the player agent moved from GAMA?
* **init\_locations**: attribute of type list. Init locations of the player agents in the environment - this information will be sent to Unity to move the players accordingly
* **ready\_to\_send\_geometries**: attribute of type list. list of players that are readdy to send geometries to Unity
* **do\_send\_world**: attribute of type bool. Has the agents has to be sent to unity?
* **connect\_to\_unity**: attribute of type bool. Activate the unity connection; if activated, the model will wait for an connection from Unity to start
* **move\_player\_from\_unity**: attribute of type bool. Has the player to move in GAMA as it moves in Unity?
* **ready\_to\_move\_player**: attribute of type list. list of players that are readdy to have their position updated from Unity
* **player\_unity\_properties**: attribute of type list. Properties used to send the player agent geometry to Unity - if nil/empty, the player agents are not sent
* **precision**: attribute of type int. Number of decimal for the data (location, rotation)
* **min\_player\_position\_update\_duration**: attribute of type float. Minimum delay between two transmissions of a player's position from Unity
* **geometries\_to\_send**: attribute of type map. List of geometries to sent to Unity with the unity properties to use. It could be updated each simulation step
* **player\_agents**: attribute of type map. Player agents indexes by their name
* **unity\_properties**: attribute of type list. List of background geometries to sent to Unity.
* **max\_num\_players**: attribute of type int. Maximal number of Unity players
* **use\_middleware**: attribute of type bool. Use of the middleware to connect Unity and GAMA? Direct connection is only usable for 1 player game
* **mailbox**: attribute of type list. The list of messages that can be consulted by the agent

Actions:

* **fetch\_message**: no arguments, returns a result of type message. Fetch the first message from the mailbox (and remove it from the mailing box). If the mailbox is empty, it returns a nil message.
* **disconnect**: no arguments, returns a result of type bool. Disconnects from all the servers previously connected to. Will return true if everything went well, false in case of an error.
* **fetch\_message\_from\_network**: no arguments, returns a result of type bool. Fetch all messages from network to mailbox. Use this in specific case only, this action is done at the end of each step.
* **execute**: (string command), returns a result of type string. Action that executes a command in the OS, as if it is executed from a terminal. use 'command' instead
* **has\_more\_message**: no arguments, returns a result of type bool. Check whether the mailbox contains any message.
* **leave\_group**: (string with\_name), returns a result of type bool. leave a group of agents. The leaving agent will not receive any message from the group. Otherwise, it can send messages to the left group
* **join\_group**: (string with\_name), returns a result of type bool. Allow an agent to join a group of agents in order to broadcast messages to other members or to receive messages sent by other members. Note that all members of the group called : "ALL".
* **connect**: (string protocol, int port, bool raw, string with\_name, string login, string password, bool force\_network\_use, string to, int size\_packet), returns a result of type bool. Action used by a networking agent to connect to a server or to create a server.
* **set\_terrain\_values**: (agent player, string id, field field, matrix matrix, int index\_x, int index\_y), no value returned. send a sub-part of a Terrain in Unity at a given index
* **message\_geometry\_loc**: (geometry geom), returns a result of type map. Action called by the send\_world action that returns the message to send to Unity
* **create\_init\_player**: (string id), no value returned. Create and init a new unity player agent
* **send\_teleport\_area**: (agent player, string id, list geoms), no value returned. send a geometry to be used a teleportation area. If a telepoprtation with the same id already existed,update its geometry. It the send geometry is null or empty, destroy this teleportation area
* **after\_sending\_world**: (map map\_to\_send), no value returned. Action trigger just after sending the world to Unity
* **add\_to\_map**: (map map, geometry geom), no value returned
* **filter\_distance**: (list geometries, agent player), returns a result of type list. Action called by the send\_world action that returns the sub-list of geometries to send to Unity from a given list of geometries according to a max distance to the player
* **add\_geometries\_to\_send**: (container geometries, unity\_property property, map attributes), no value returned. Action allows to define the list of geometries to send to Unity
* **send\_current\_message**: no arguments, no value returned. send the current message to the Unity Client
* **send\_message**: (list players, map mes), no value returned. send a message to the Unity Client
* **end\_of\_game**: (string mes), no value returned. put an end to the game and restart the game
* **send\_init\_data**: (string id), no value returned. Wait for the connection of a unity client and send the paramters to the client
* **new\_player\_location**: (point loc, agent player), returns a result of type point. Action called by the move\_player action that returns the location to send to Unity from a given player location
* **send\_world**: no arguments, no value returned. send the current state of the world to the Unity clients
* **ping\_GAMA**: (string id), no value returned. Ping GAMA to test the connection
* **add\_to\_send\_parameter**: (agent player, map map\_to\_send), no value returned. add values to the parameters sent to the Unity Client
* **build\_invisible\_walls**: (agent player, string id, float height, float wall\_width, list geoms), no value returned. send a list of geometry to be used to build walls at the given height. If walls with the same id already existed,destroy them before creating the new walls
* **player\_position\_updated**: (string id), no value returned. reactivate the reception of player position
* **player\_ready\_to\_receive\_geometries**: (string id), no value returned. enable to receive geometries
* **move\_player**: (agent player, point loc), no value returned. move the player agent
* **move\_player\_external**: (string id, int x, int y, int z, int angle), no value returned. move the player agent
* **loc\_to\_send**: no arguments, returns a result of type point
* **add\_background\_geometries**: (container geometries, unity\_property property), no value returned. Action called by the send\_world action that returns the sub-list of geometries to send to Unity from a given list of geometries according to a max distance to the player
* **create\_player**: (string id), no value returned. Create a new unity player agent
* **send\_player\_position**: (agent player), no value returned. send the new position of the player to Unity (used to teleport the player from GAMA)
* **add\_to\_send\_world**: (map map\_to\_send), no value returned. add values to the world sent to the Unity Client
* **filter\_overlapping**: (list geometries, agent player), returns a result of type list. Action called by the send\_world action that returns the sub-list of geometries to send to Unity from a given list of geometries according to a min proximity to the other geometries to send
* **send\_geometries**: (agent player, bool update\_position, bool is\_init, map geoms), no value returned. send the background geometries to the Unity client
* **message\_geometry\_shape**: (geometry geom), returns a result of type map. Action called by the send\_world action that returns the message to send to Unity
* **send\_parameters**: (agent player), no value returned. Send the parameter to Unity to intialized the connection
* **update\_terrain**: (agent player, string id, field field, matrix matrix, float max\_value, float size\_x, float size\_y, int resolution), no value returned. send a matrix/cell/field to update a Terrain in Unity
* **move\_geoms\_followed**: (string ids, string points, string sep), no value returned. Action called by the Unity Client to move agents
* **enable\_player\_movement**: (agent player, bool enable), no value returned. Enable (true) or disable (false) the movement (teleportation, continuous move) of a player
* **add\_geometries\_to\_keep**: (container geometries), no value returned. Action allows to define the list of geometries to keep in Unity (and not sent)
* **after\_sending\_geometries**: (agent player), no value returned. Action trigger just after sending the background geometries to a Unity client
* **send\_unity\_propetries**: (agent player), no value returned. Send the Unity properties to intialize the possible properties of geometries
* **update\_animation**: (list players, container geometries, list triggers, map parameters), no value returned. Action called by the send\_world action that returns the sub-list of geometries to send to Unity from a given list of geometries according to a max distance to the player
* **send**: (unknown to, unknown contents), returns a result of type message. Action used to send a message (that can be of any kind of object) to an agent or a server.
* **die**: no arguments, no value returned. Kills the agent and disposes of it. Once dead, the agent cannot behave anymore
* **tell**: (string msg, bool add\_name), no value returned
* **error**: (string message), no value returned
* **debug**: (string message, string separator, string end), no value returned
* **\_init\_**: no arguments, no value returned
* **\_step\_**: no arguments, no value returned