6.3.3 State properties

Carlos Hernandez

FlexBE states

Remember:

- A state encapsulates an atomic action in the behavior.
- ROS nodes provide **capabilities** through topics, services and ROS actions.

A FlexBE **state**:

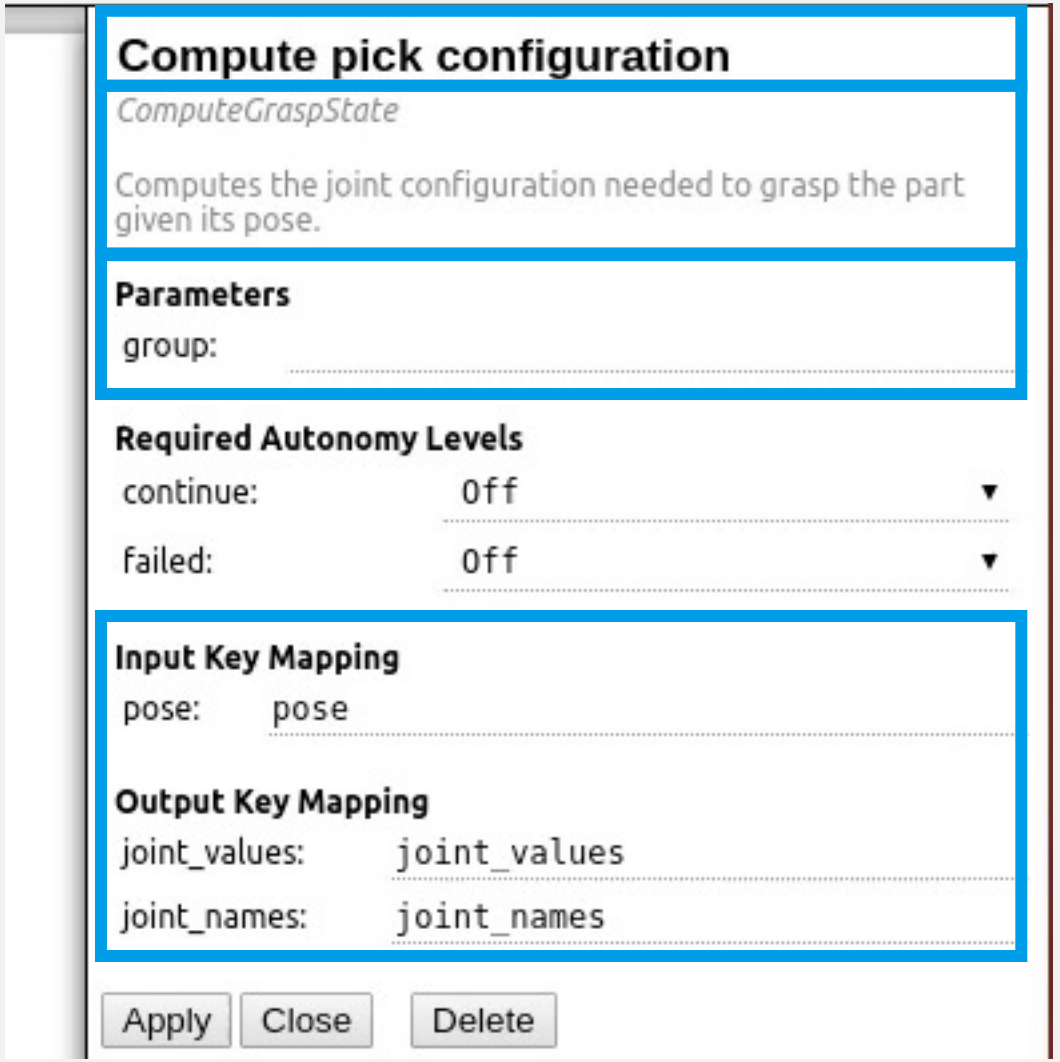
- represents a concrete use of one or more those capabilities to perform a **specific action**.

Properties of a FlexBE State

- Name
- State implementation

Interface

- Parameters
- Input & Output keys
- Outcomes

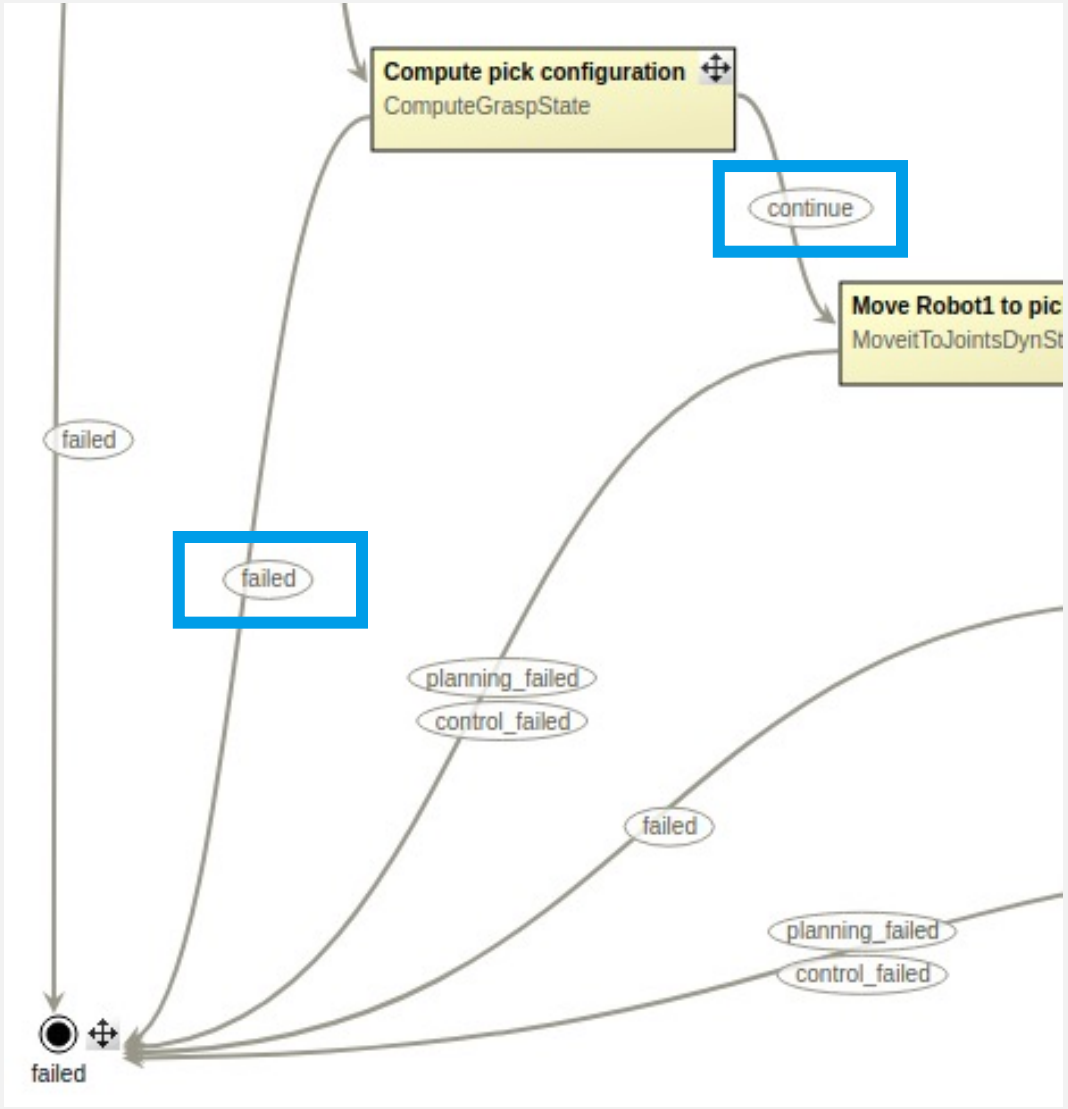


Properties of a FlexBE State

- Name
- State implementation

Interface

- Parameters
- Input & Output keys
- Outcomes



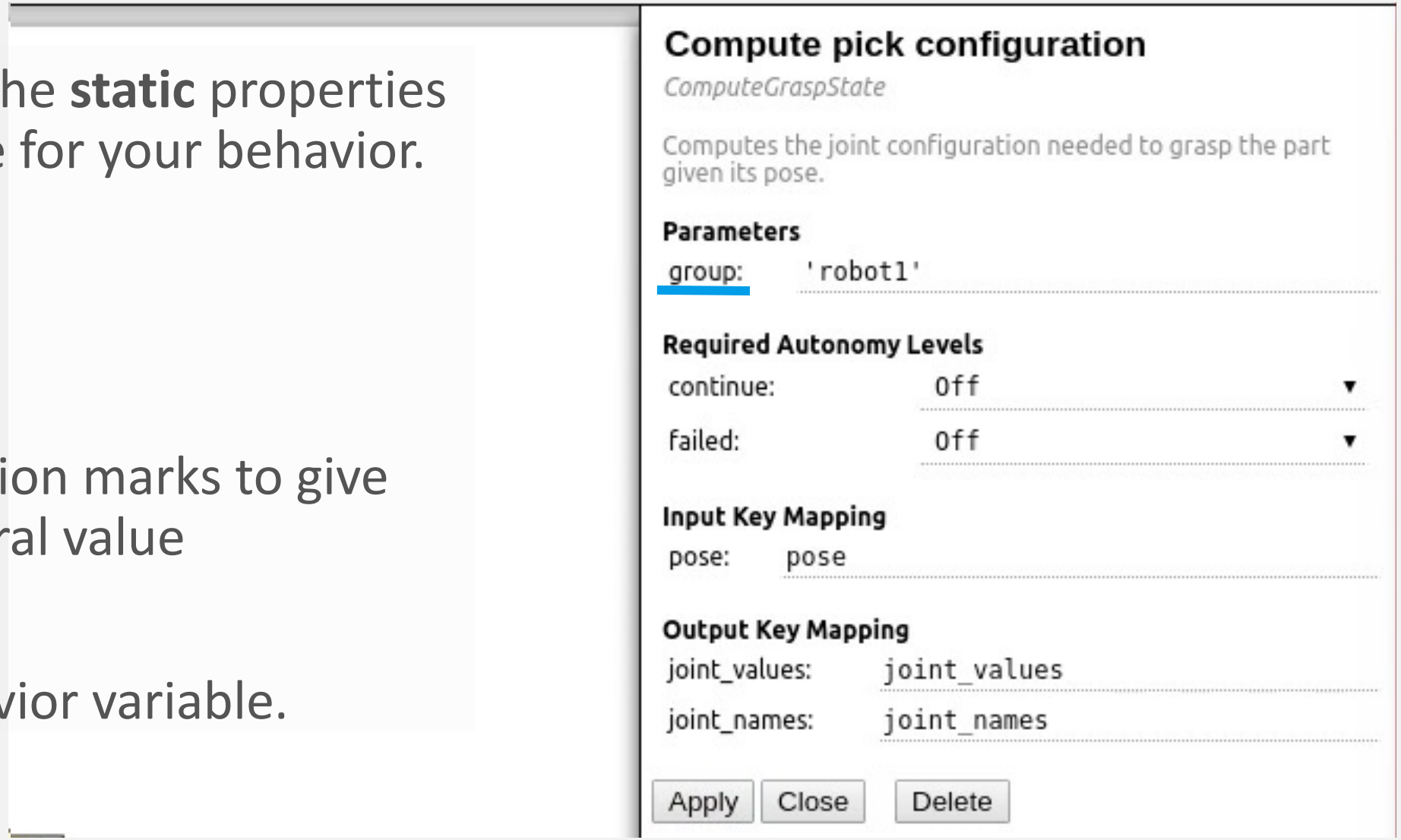
State parameters

- Configure the **static** properties of the state for your behavior.

- Use quotation marks to give them a literal value

  or

- use a behavior variable.

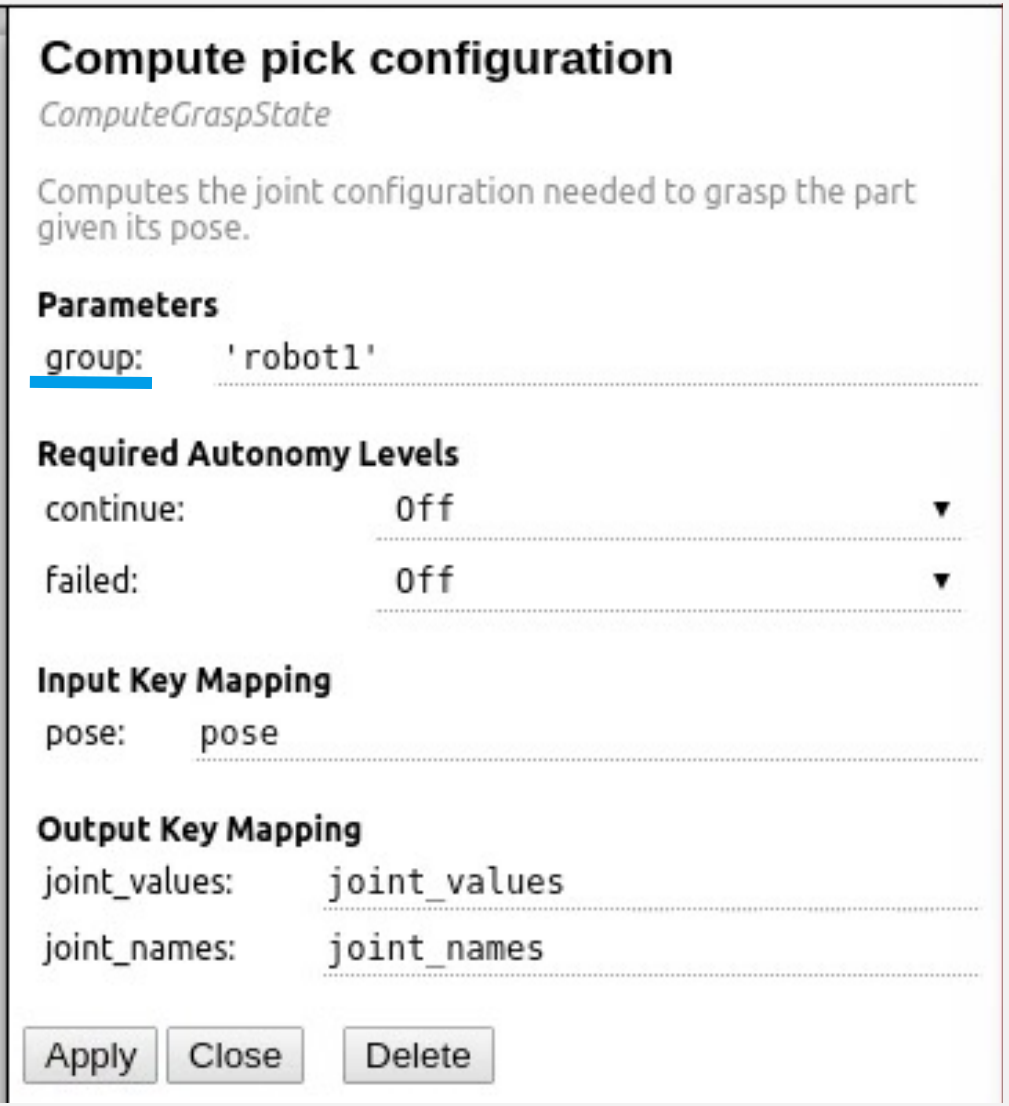


State parameters

- Use quotation marks to give them a literal value

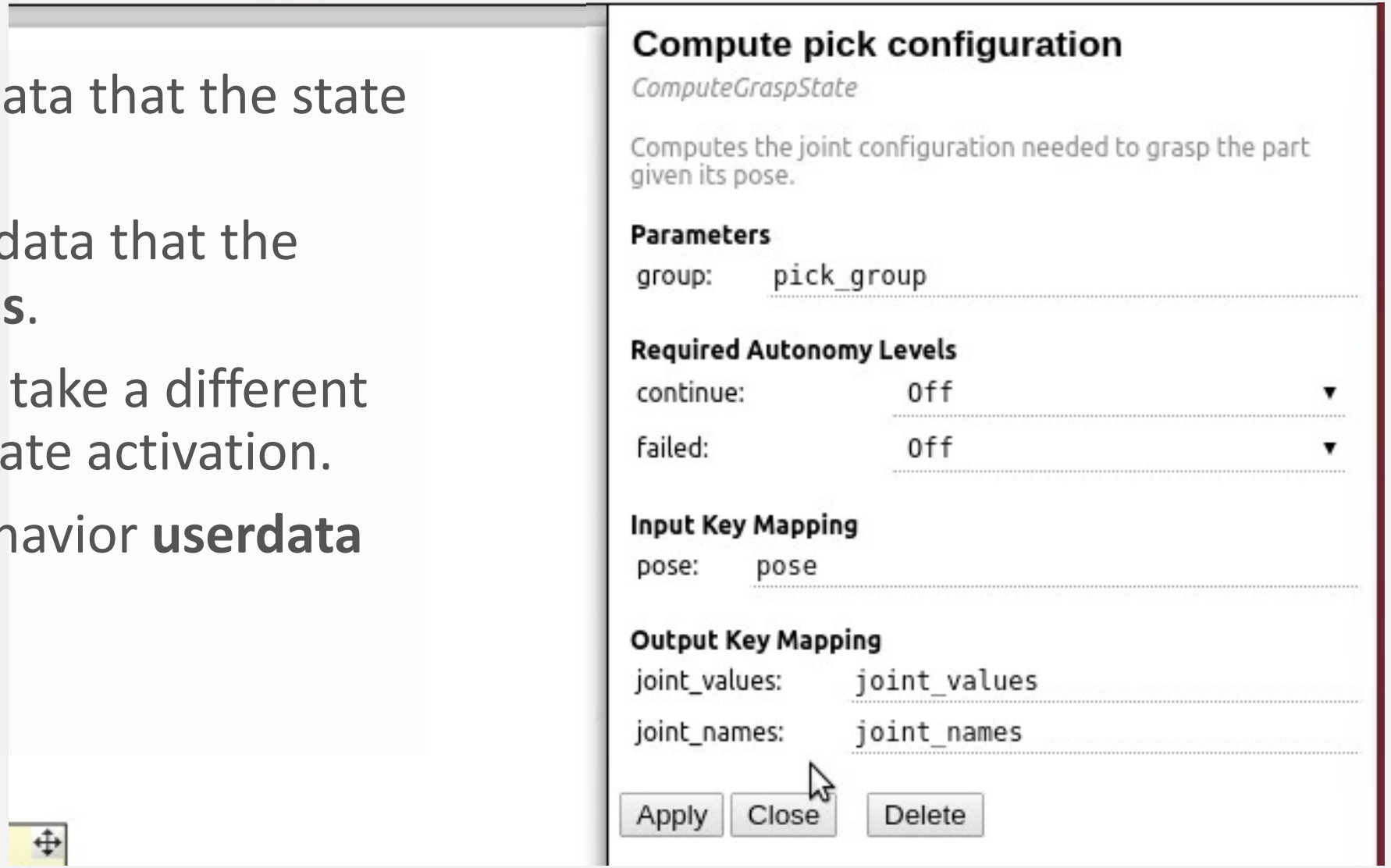
  or

- use a behavior variable.

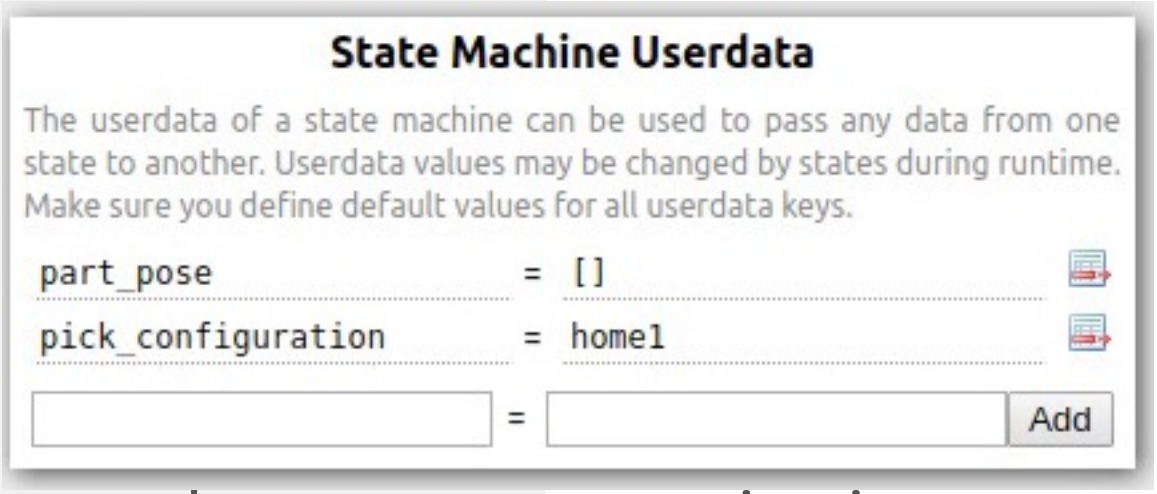


Input and Output Key Mappings

- **Inputs** keys: data that the state **requires**.
- **Output** keys: data that the state **produces**.
- **Dynamic**: can take a different value every state activation.
- Values are behavior **userdata** variables.

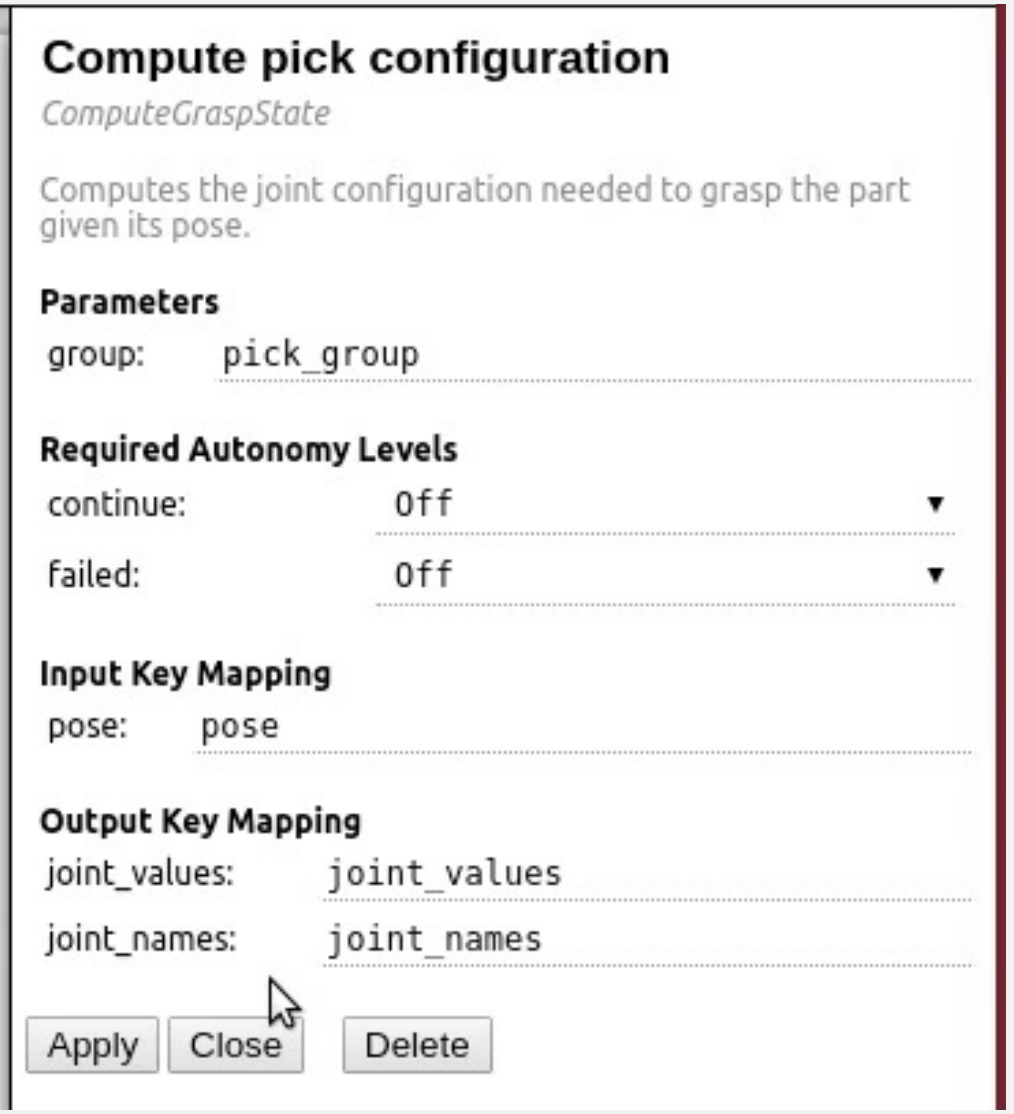


Input and Output Key Mappings



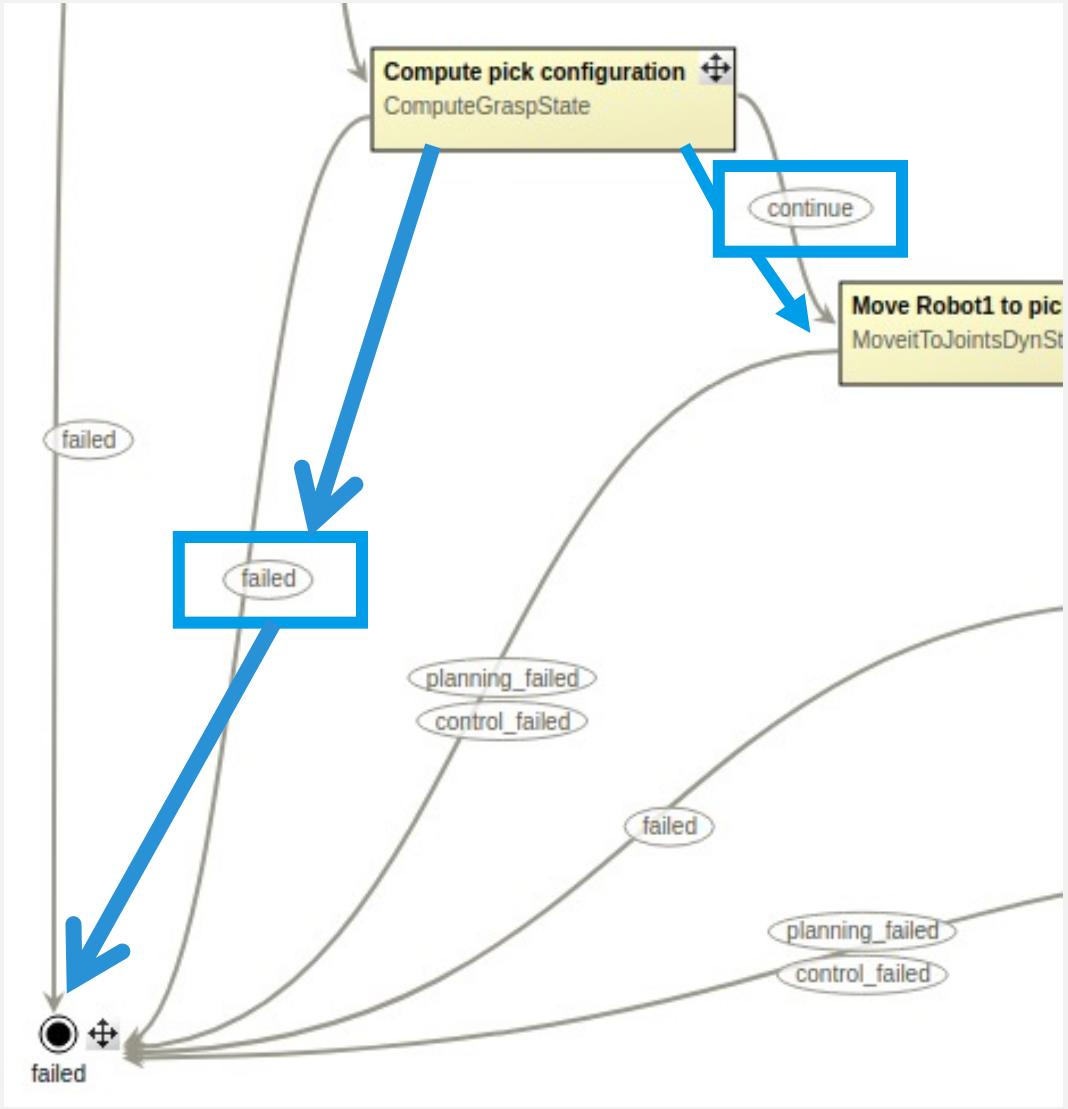
value every state activation.

- Values are behavior **userdata** variables.



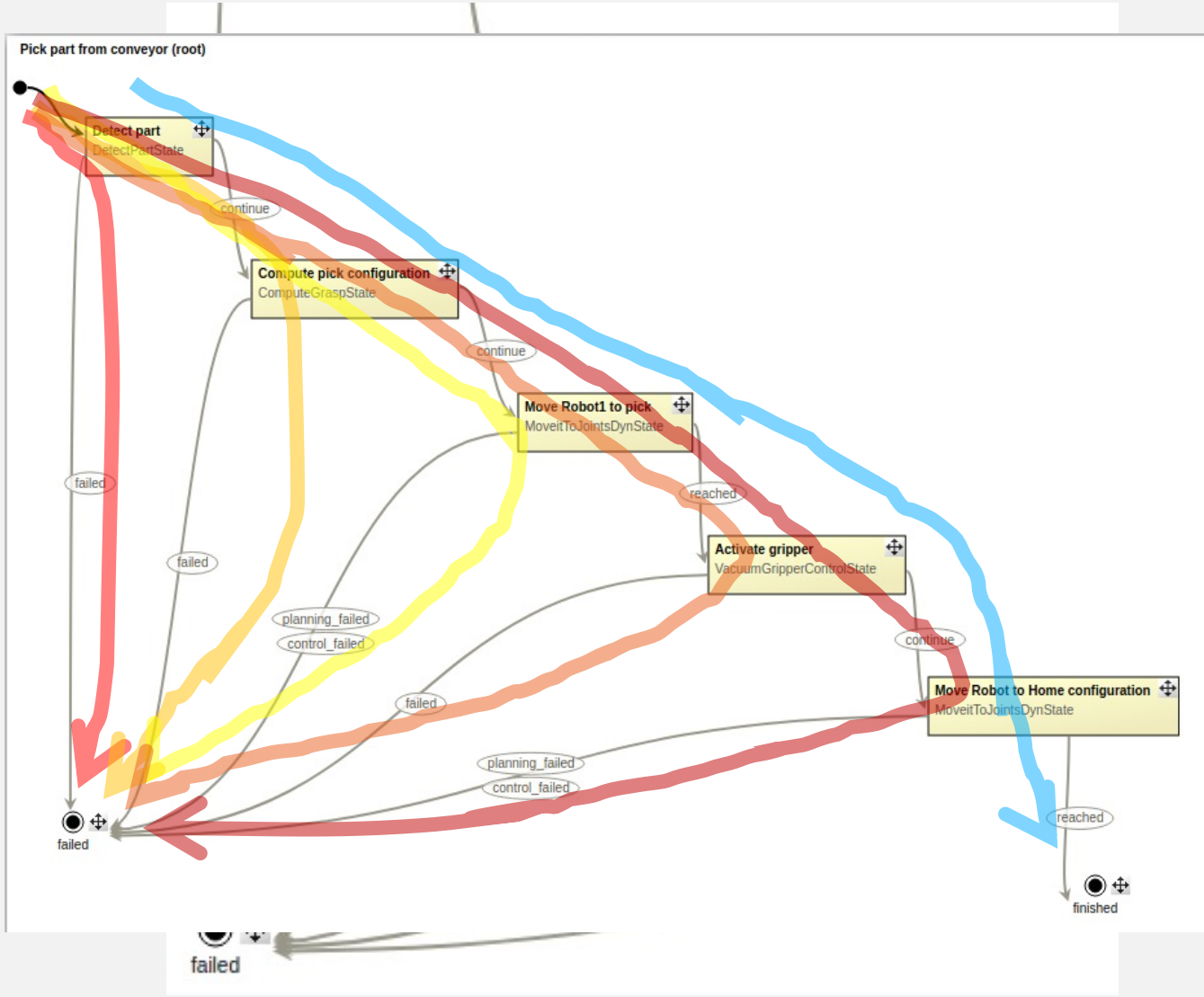
State Outcomes

- Define possible results of execution.
- Possible transitions from the state.
- Shape the possible execution flows in the state machine.



State Outcomes

- Define possible results of execution.
- Possible transitions from the state.
- Shape the possible execution flows in the state machine.



Properties of a FlexBE State

- Name
- State implementation
- Parameters
- Input & Output key Mappings
- Outcomes