Statechart Refactorings

1. Rename

- Description:
  - Renaming of elements in the whole statechart model
- Pre-checks:
  - Name does not already exists

2. Fold incoming actions

- Description:
  - Moves actions of incoming transitions to entry block of state
- Applicable to:
  - States
- Pre-checks:
  - All incoming transitions have same actions
    - Or check form back to front: if last actions are the same, move them, if not, abort moving
  - Special case: Transition crosses border of parent composite state (transition enters parent composite state)
    - If composite state has entry actions: No folding possible!
- Post-Conditions:
  - Folded actions are inserted before already existing entry actions

3. Fold outgoing actions

- Description:
  - Moves actions of outgoing transitions to exit block of state
- Applicable to:
  - States
- Pre-checks:
  - All outgoing transitions have same actions
    - Or check from front to back: if first actions are the same, move them, if not, abort moving
  - Special case: Transition crosses border of parent composite state (transition leaves parent composite state)
    - If composite state has exit actions: No folding possible!
- Post-Conditions:
  - Folded actions are inserted after already existing exit actions

4. Unfold exit actions

- Description:
  - Moves exit actions from state to all its outgoing transitions
- Applicable to:
  - States
- Pre-checks:
  - Special case: Outgoing transition leaves parent composite state
    - If composite state has exit actions: No unfolding possible!
- Post-Conditions:
  - Unfolded actions are inserted in front of already existing actions (of outgoing transition)

5. Unfold entry actions

- Description:

        - Moves entry actions from state to all its incoming transitions
    - Applicable to:
        - States
    - Pre-checks:
        - Special case: Incoming transition enters parent composite state
            - If composite state has entry actions: No unfolding possible!
    - Post-Conditions:
        - Unfolded actions are inserted after already existing actions (of incoming transition)

6. Group states into composite state

- Description:
    - Puts the selected set of states into a composite state
- Applicable to:
    - States (multiple selection possible)
- Pre-checks:
    - All selected states belong to the same container
    - Name of the new composite state is unique
- Post-Conditions:
    - New composite state has no incoming/outgoing transitions and no entry/exit actions

7. Fold outgoing transitions

- Description:
    - Replaces transitions leaving from states of a composite state and going to the same target state with one transition from the composite state to the target state
- Applicable to:
    - Composite states
- Pre-checks:
    - Only equivalent transitions (events, guards and actions) with same target can be fold
- Post-Conditions:
    - No special

8. Create Submachine state from composite state

- Description:
    - Replaces a composite state by a submachine state. The new statechart is stored via the new file wizard. All transitions to the states inside the composite state are handled appropriately by generating a corresponding interface in the new statechart.
- Applicable to:
    - Composite states
- Pre-checks:
    - When only one state of the composite serves as target from outside
        - Attach initial state to this target
        - Target all transitions to the composite state
    - When several states of the composite serve as targets from outside
        - Create new idle state in composite and attach initial state to it
        - Create incoming events in the new statechart. These events are fired from outside, on the transitions which have the submachine state as target. The new incoming events are used to reproduce an equivalent

execution flow

- Post-Conditions:
  - No special

9. Extract Submachine

- Consists of the steps "Group states into composite state" and "Create submachine state from composite state"

10. Inline Submachine

- Consists of the steps "Create composite state from submachine state" and Ungroup states in a composite state"