Ch.9 Bottom-up/top-down inference in And-Or-Graph

--- Event Recognition and Intent Prediction

Earley-Stolcke parsing algorithm

- S->A B C
- S->D E F
- A->a1 A->a2
- B->b1 B->b2
- C->c1 C->c2
- D->a1 D->a2
- E->b1 E->b2
- F->c1 F->c2

- a1b1c1
- a1b1c2
- a1b2c1
- a1b2c2
- a2b1c1
- a2b1c2
- a2b2c1
- a2b2c2

Earley-Stolcke parsing algorithm

The input is a1 b1 c1

State set 0	State set 1	State set 2	State set 3
$_0$->.S Predicted $_0$S->.A B C $_0$S->.D E F $_0$A->.a1 $_0$A->.a2 $_0$D->.a1 $_0$D->.a2	Scanned $_0$A->a1. $_0$D->a1. Completed $_0$S->A. B C $_0$S->D. E F Predicted $_1$B->.b1 $_1$B->.b2 $_1$E->.b1 $_1$E->.b2	Scanned $_1$B->b1. $_1$E->b1. Completed $_0$S->A B. C $_0$S->D E. F Predicted $_2$C->.c1 $_2$C->.c2 $_2$F->.c1 $_2$F->.c2	Scanned $_2$C->c1. $_2$F->c1. Completed $_0$S->A B C. $_0$S->D E F. $_0$->S.

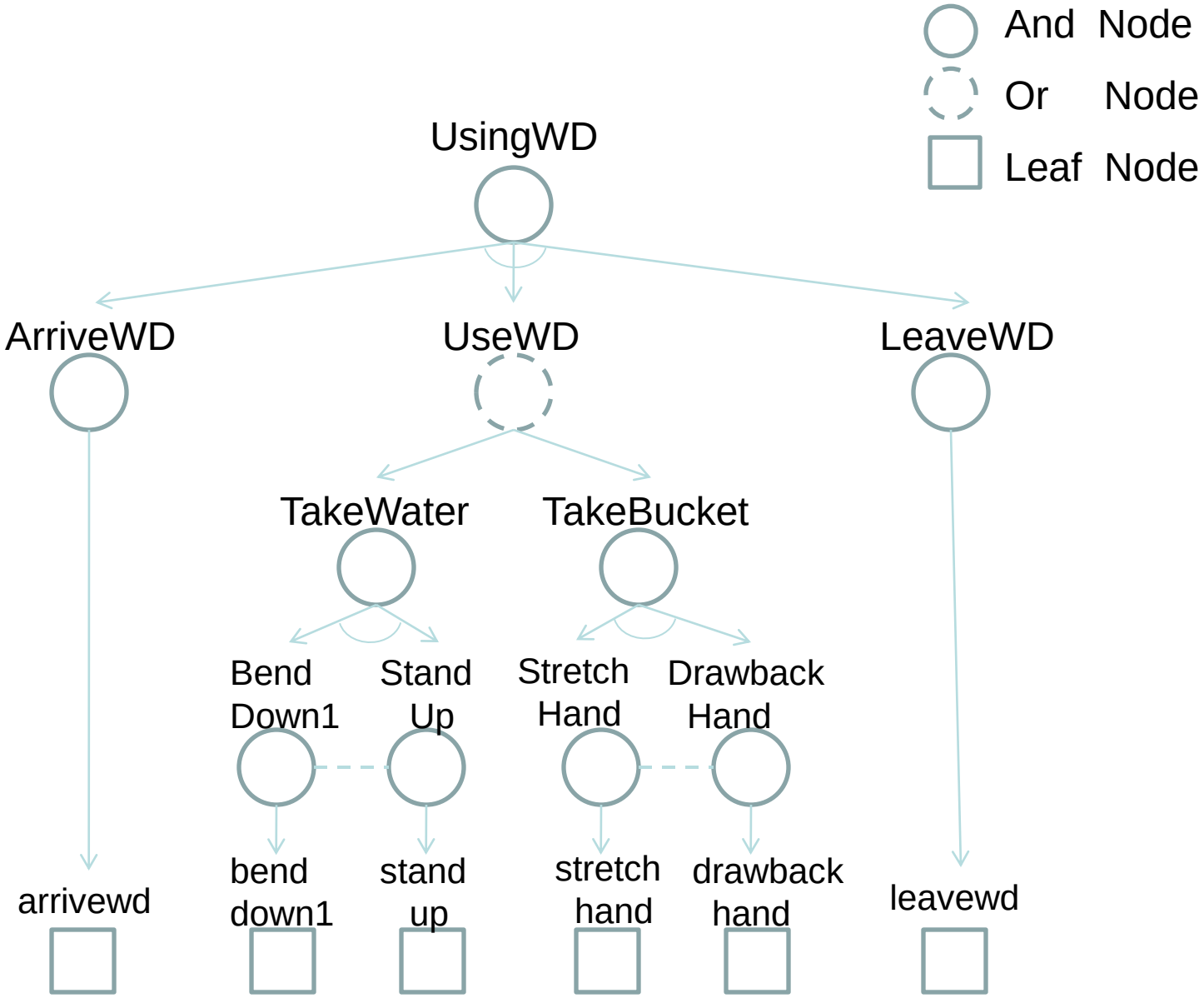
Grammars for Events

- UsingWD→ArriveWD UseWD LeaveWD
- ArriveWD→arrivewd
- UseWD→TakeWater
- UseWD→TakeBucket
- TakeWater→benddown1 standup
- TakeBucket→stretchhand drawbackhand
- LeaveWD→leavewd

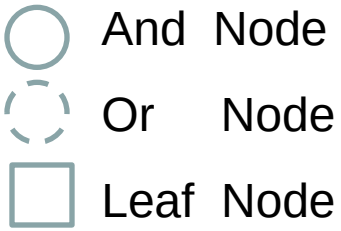
Grammars for Events

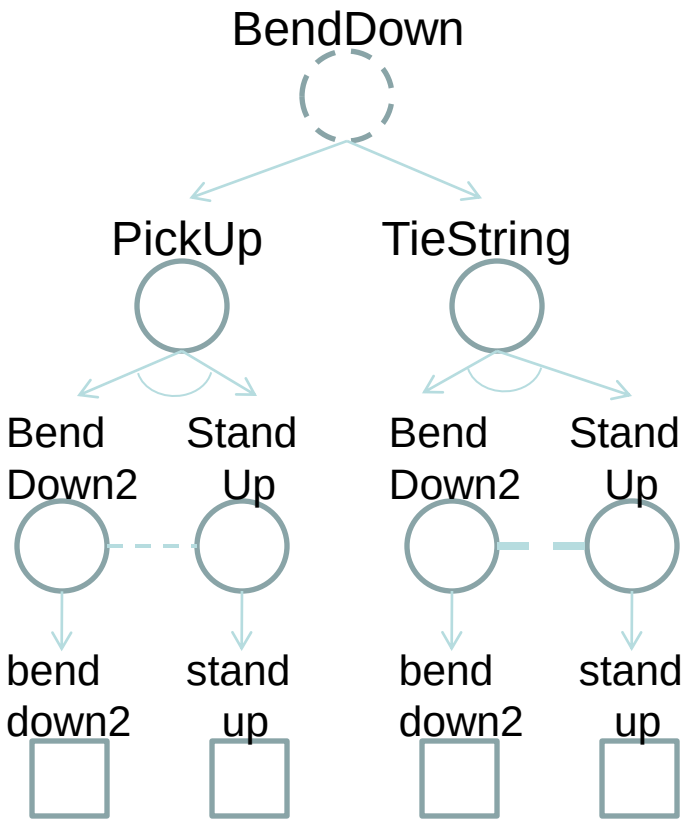
- BendDown→Pickup
- BendDown→TieString
- Pickup→benddown2 standup
- TieString→benddown2 standup

AOGs for Events

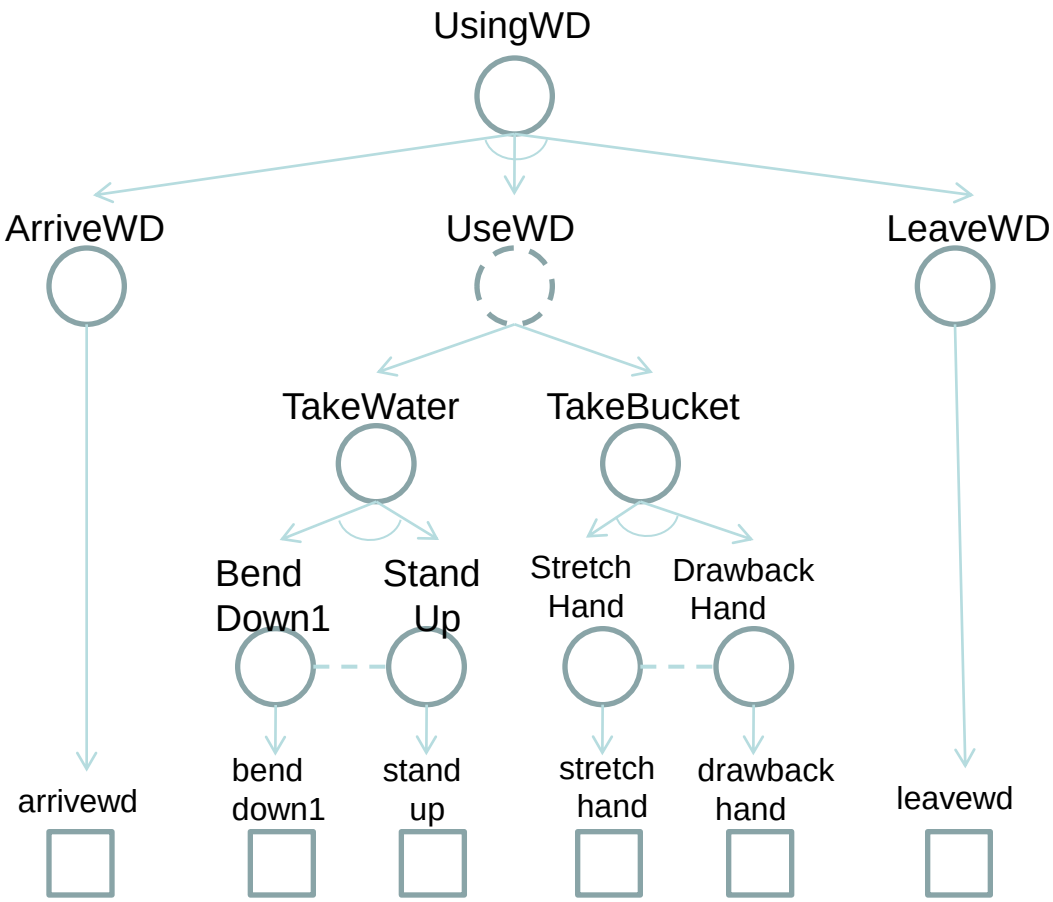


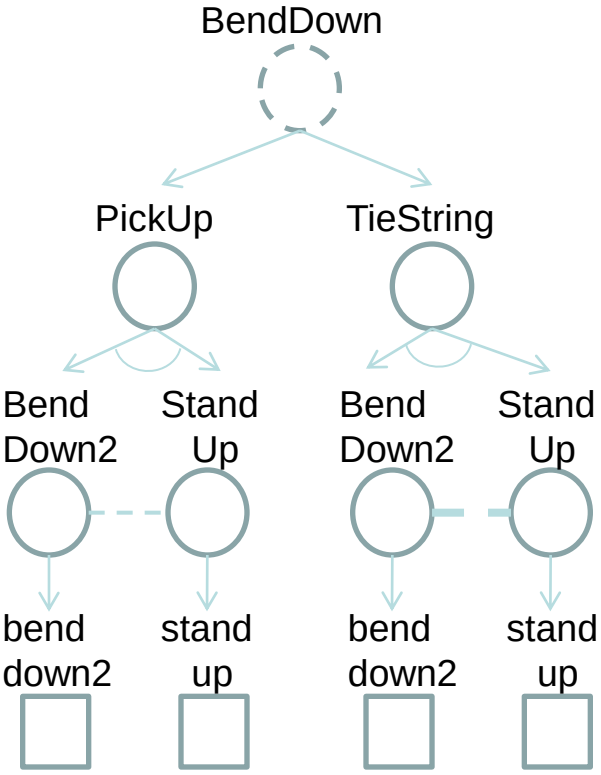
AOGs for Events



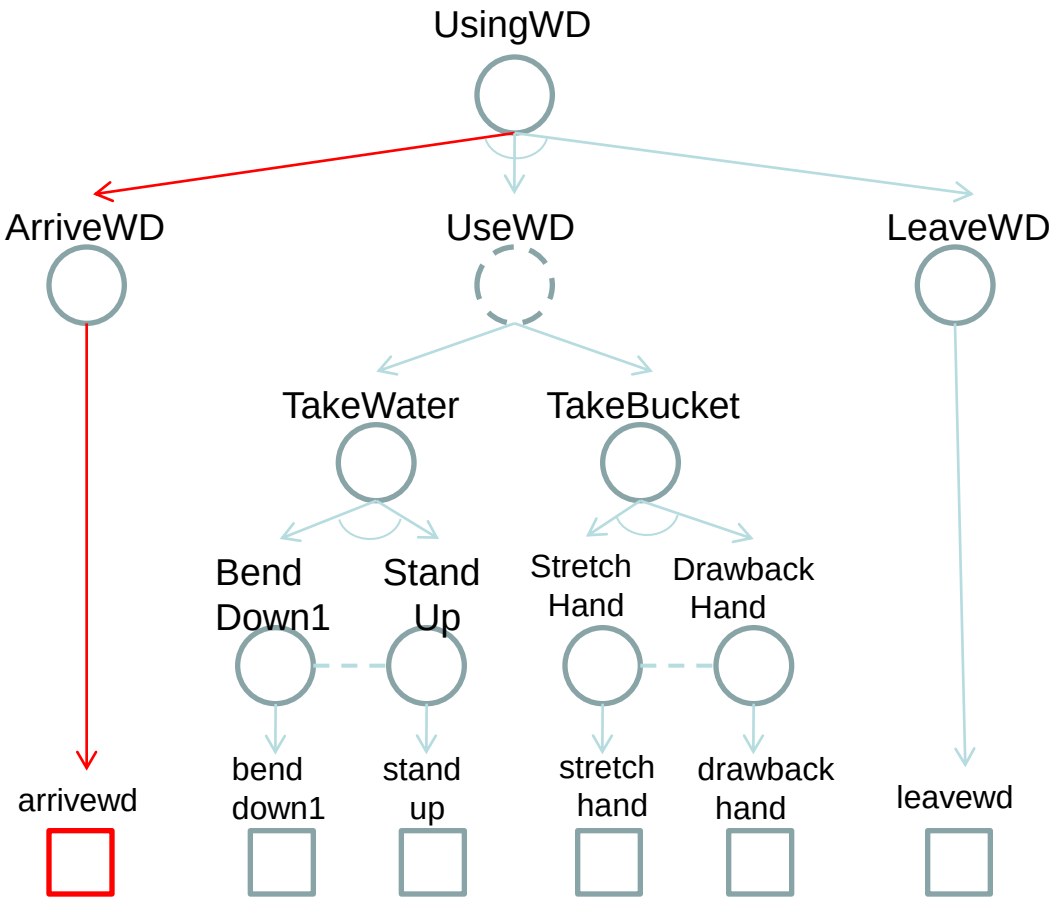


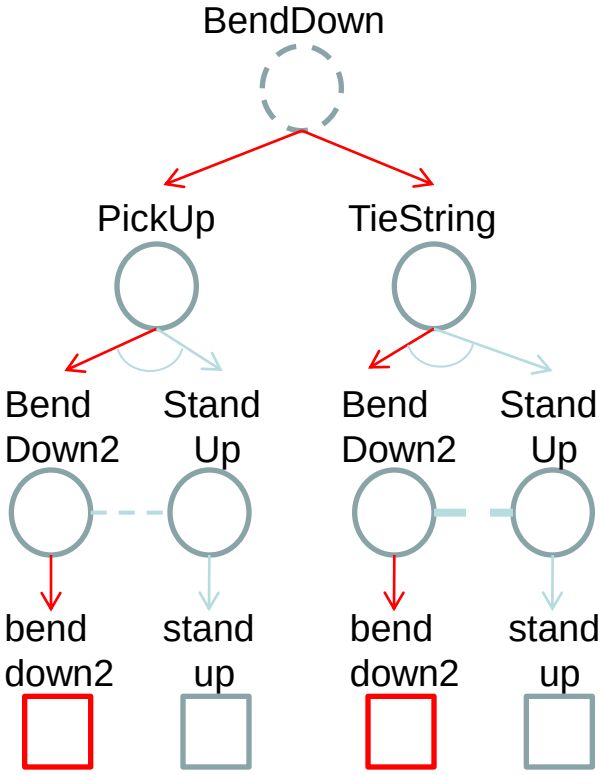
The parsing process



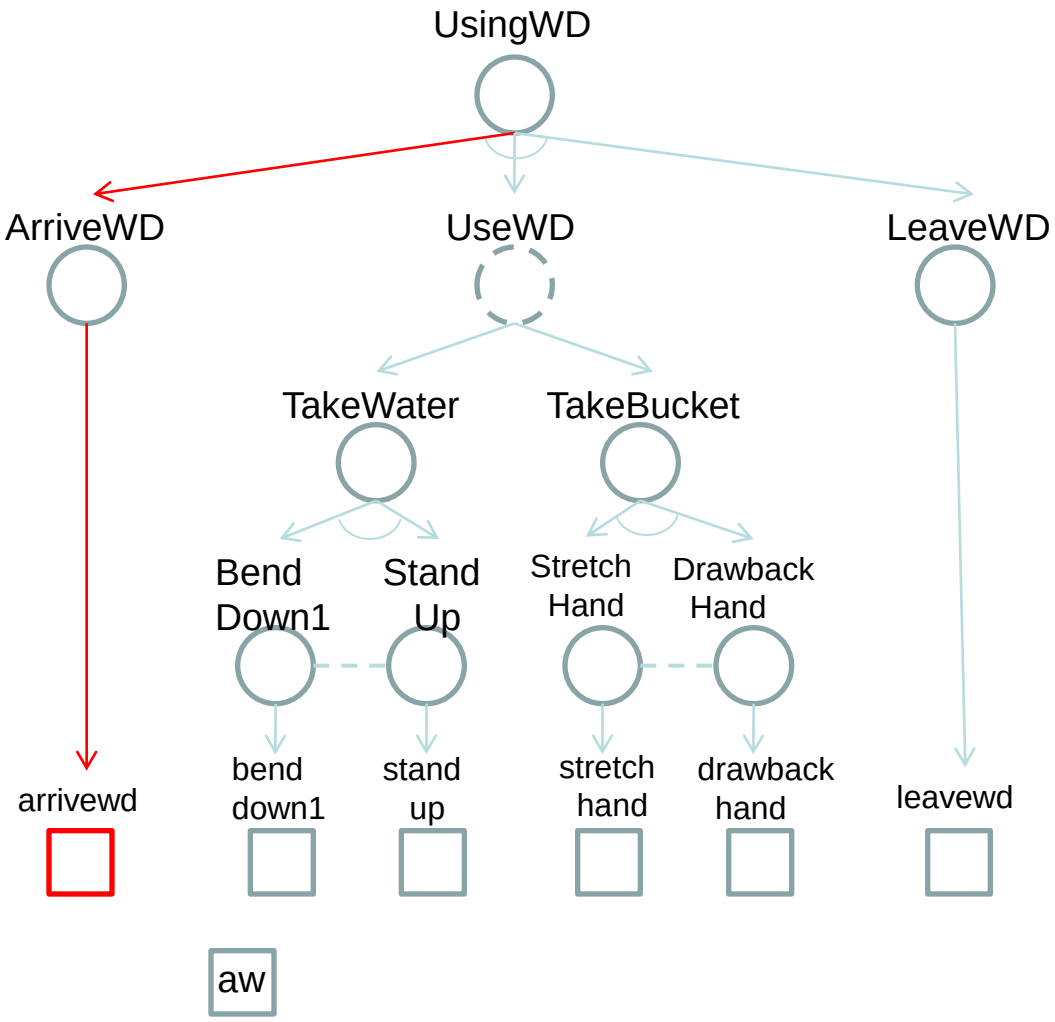


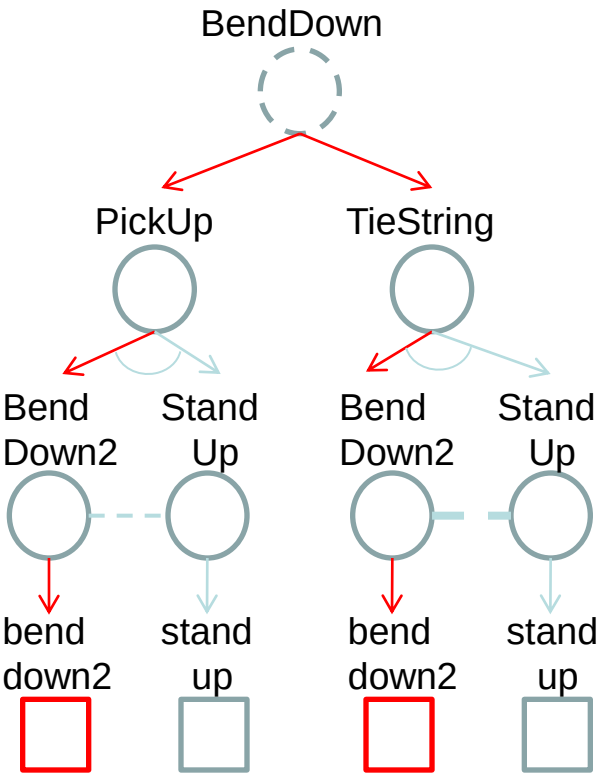
The parsing process



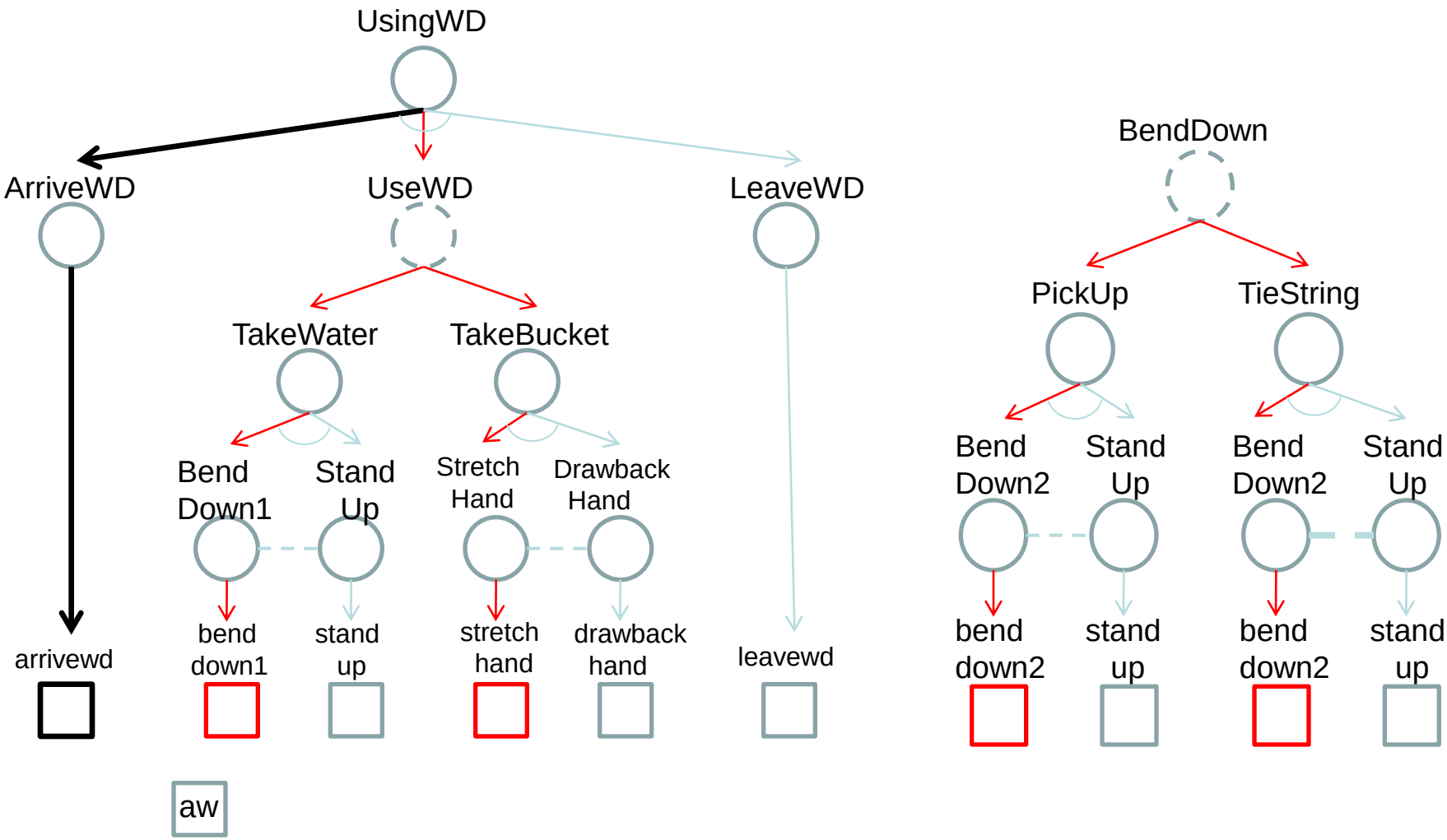


The parsing process

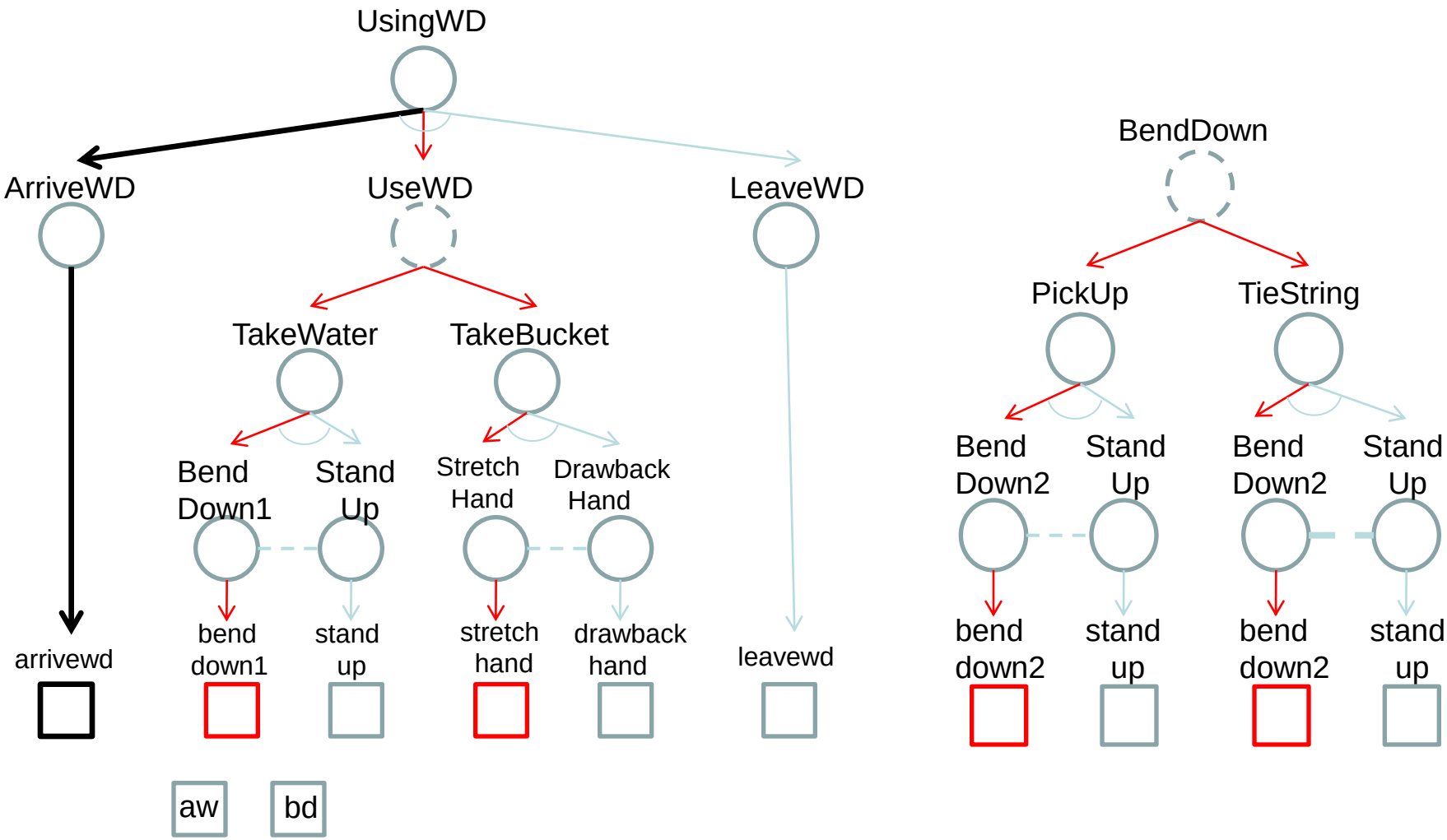




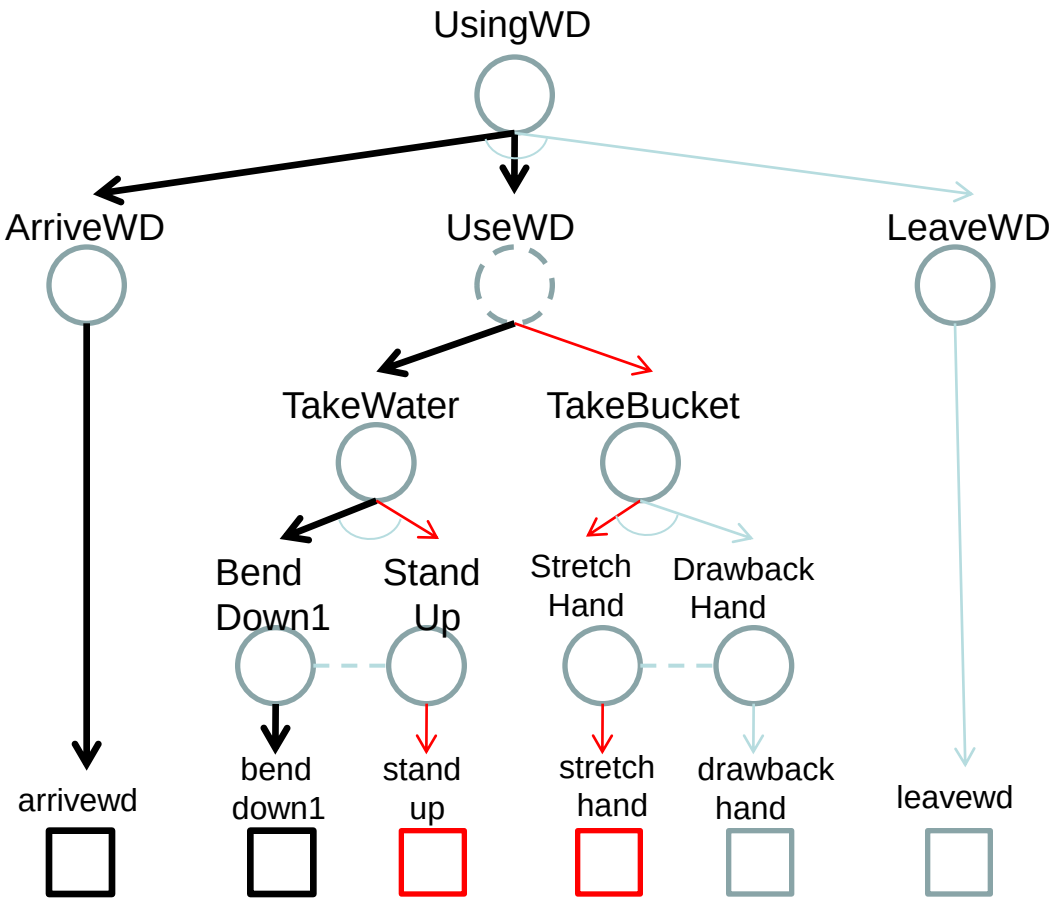
The parsing process

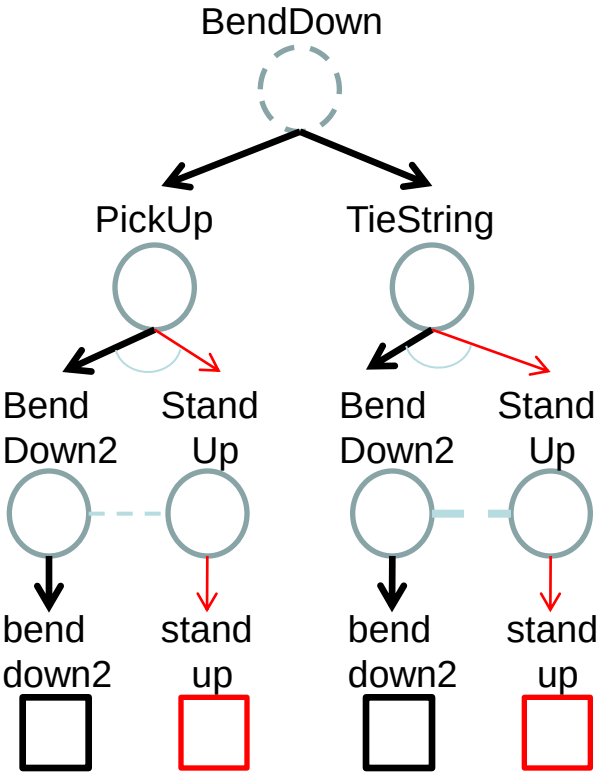


The parsing process

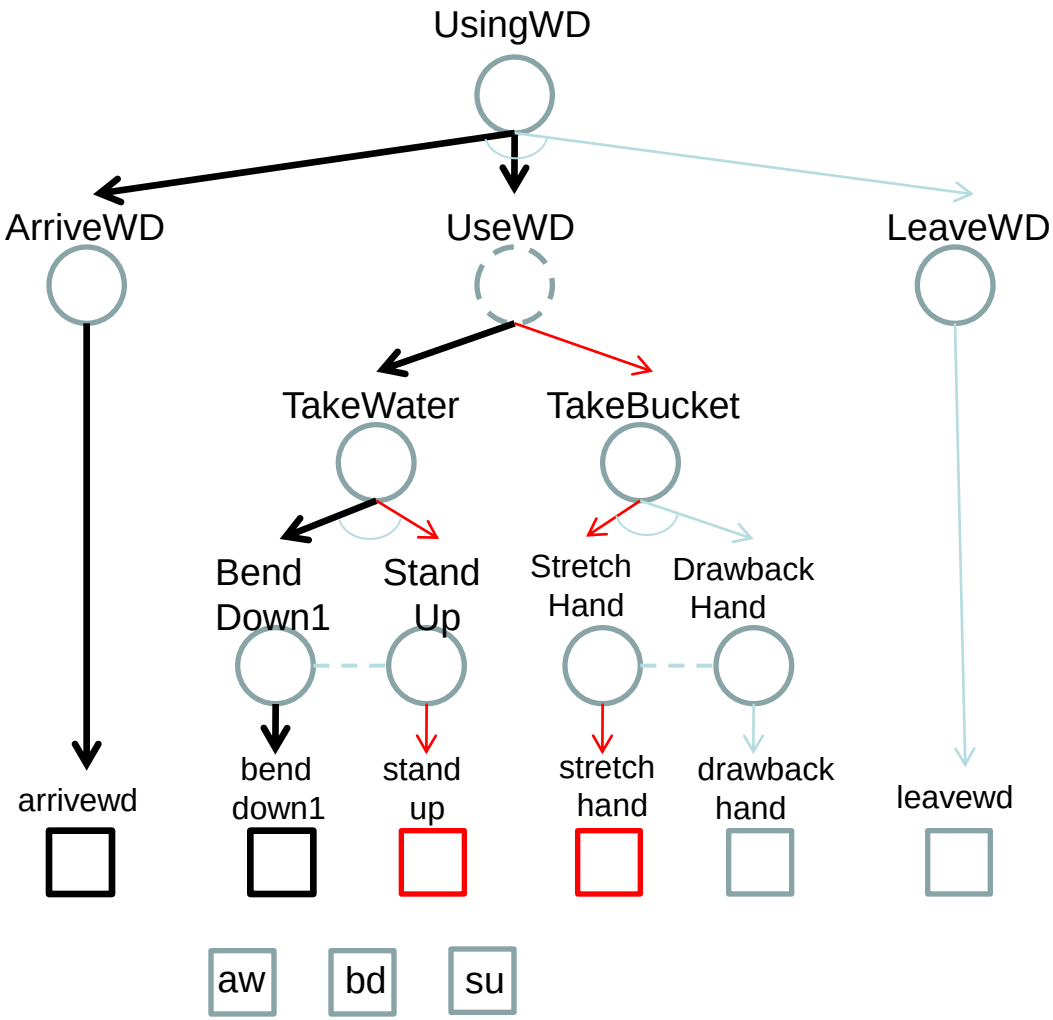


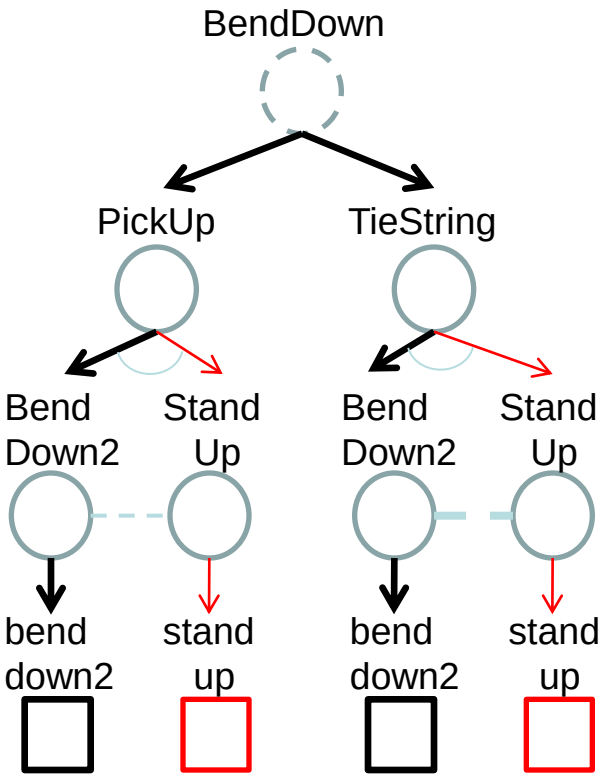
The parsing process



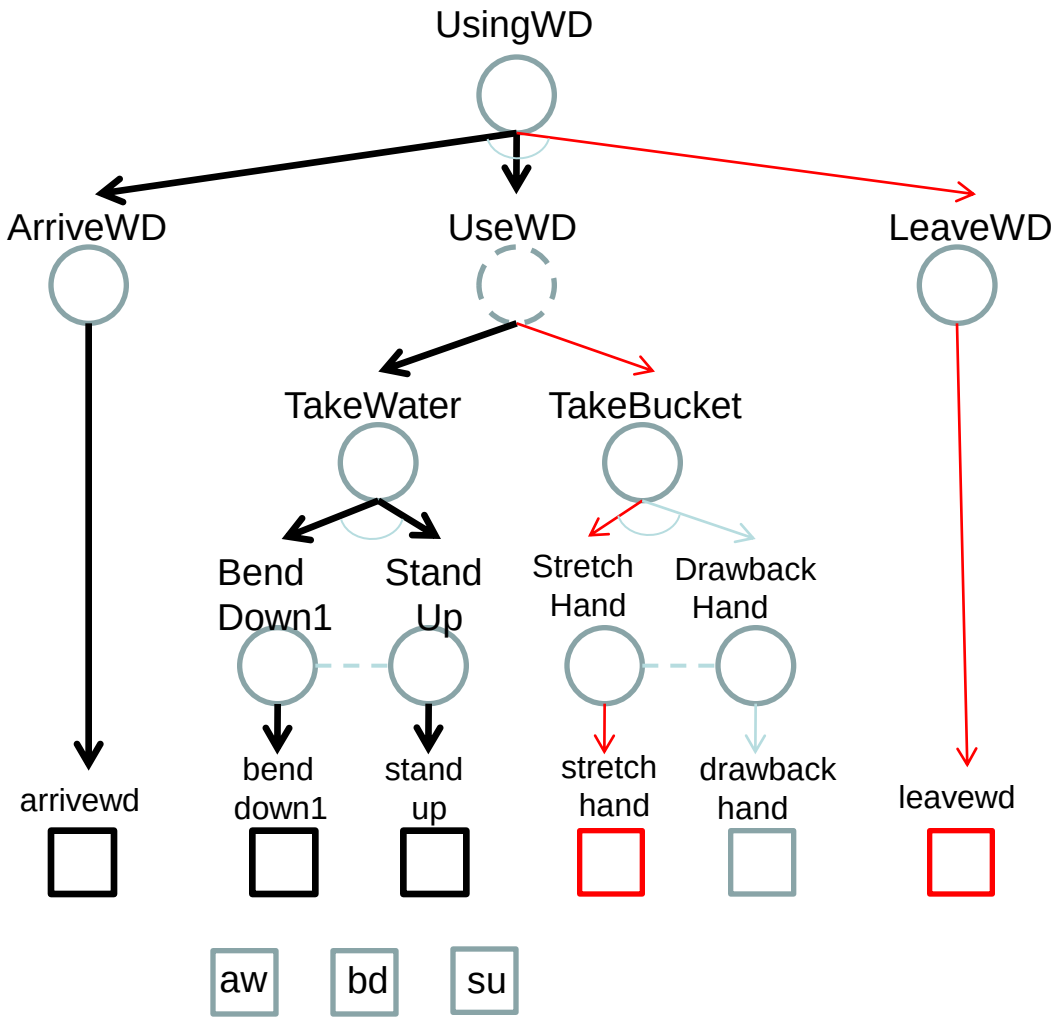


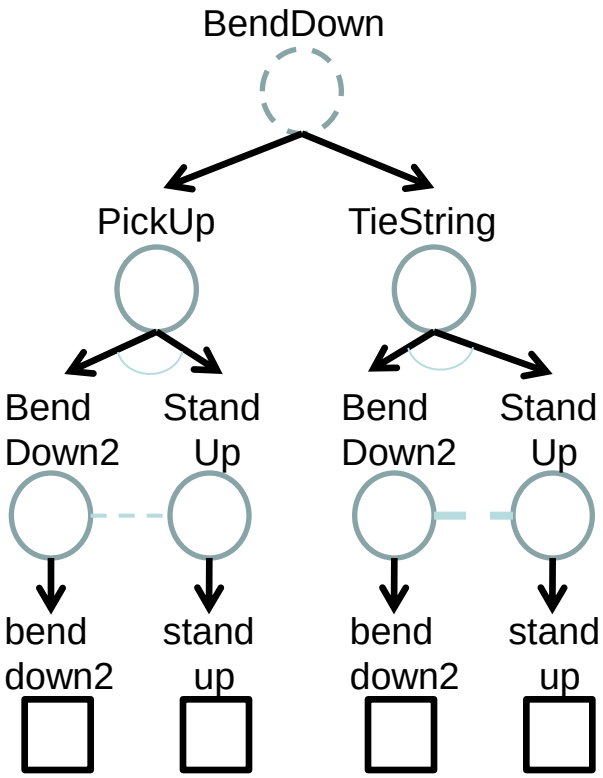
The parsing process



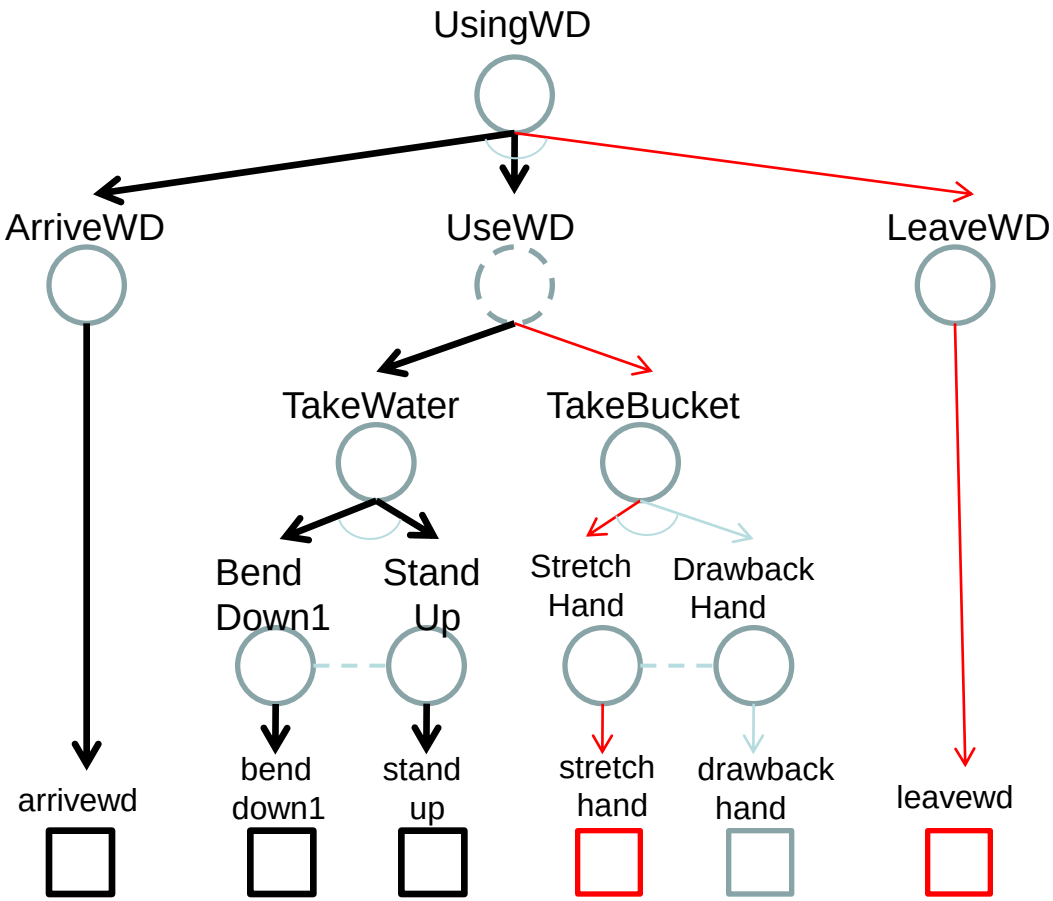


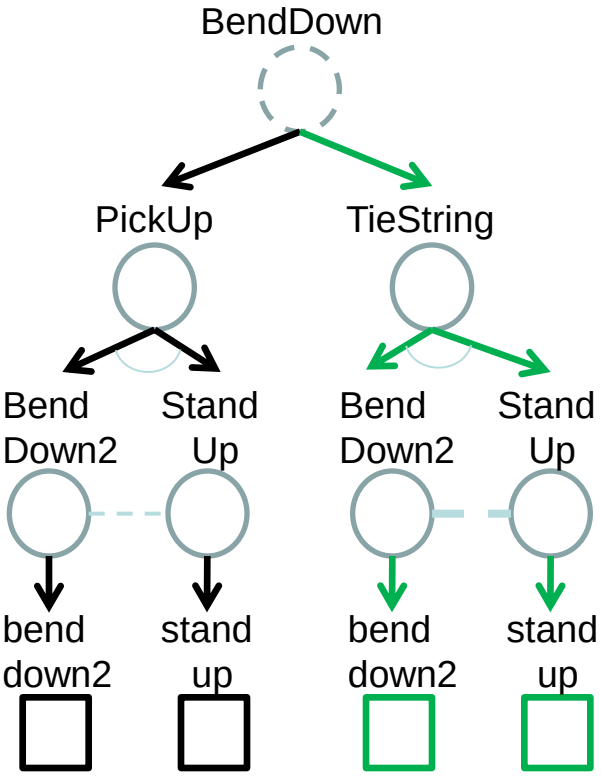
The parsing process



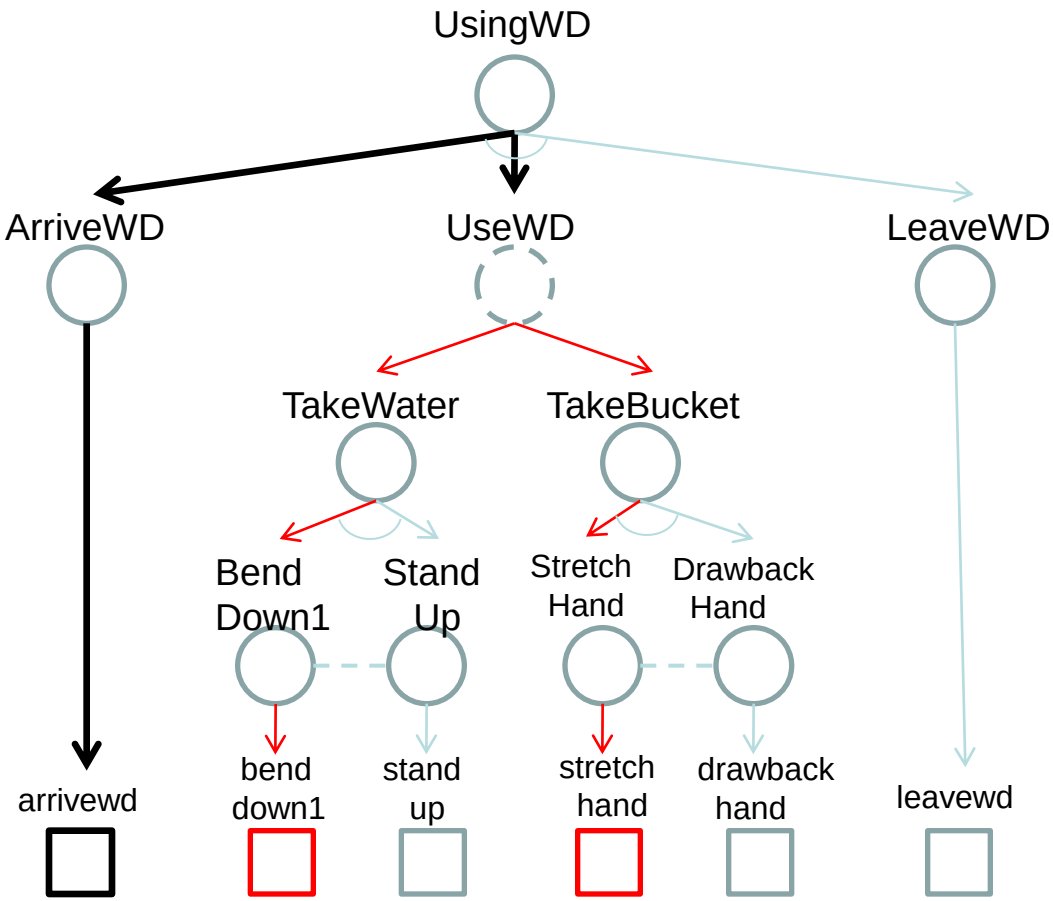


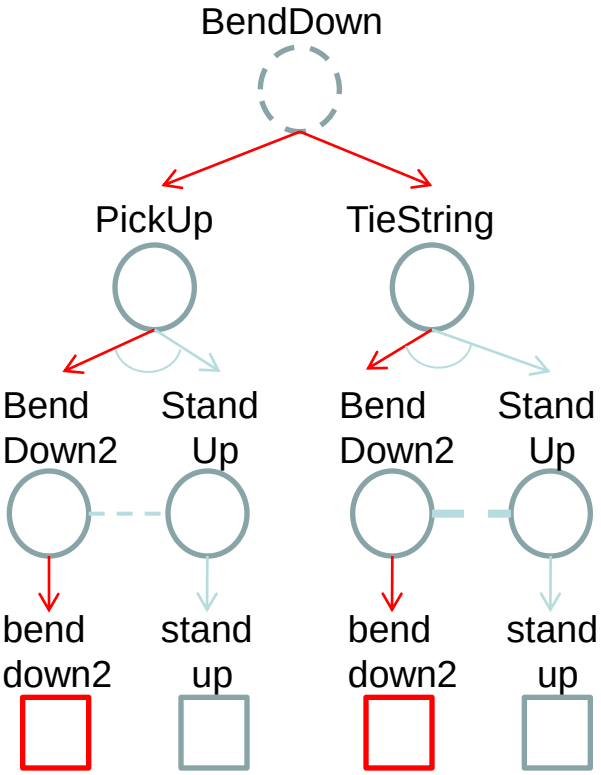
The parsing process



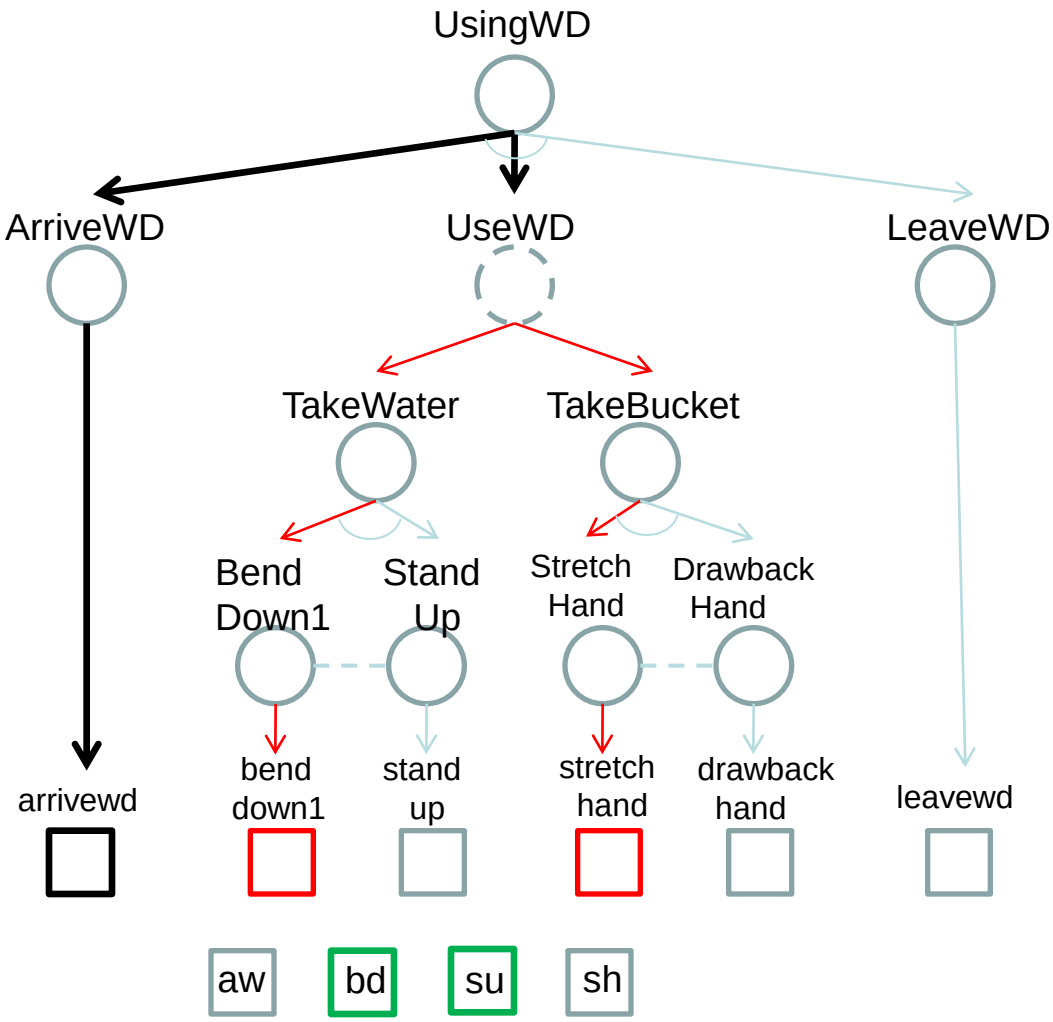


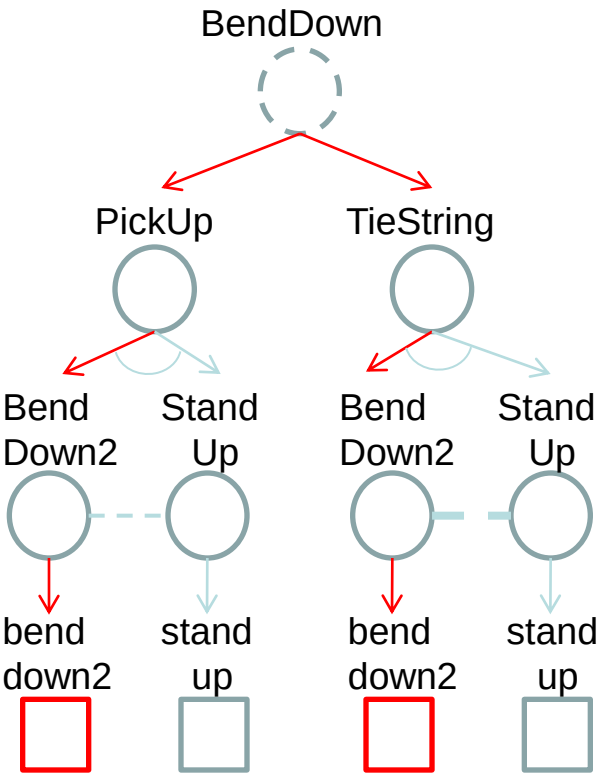
The parsing process



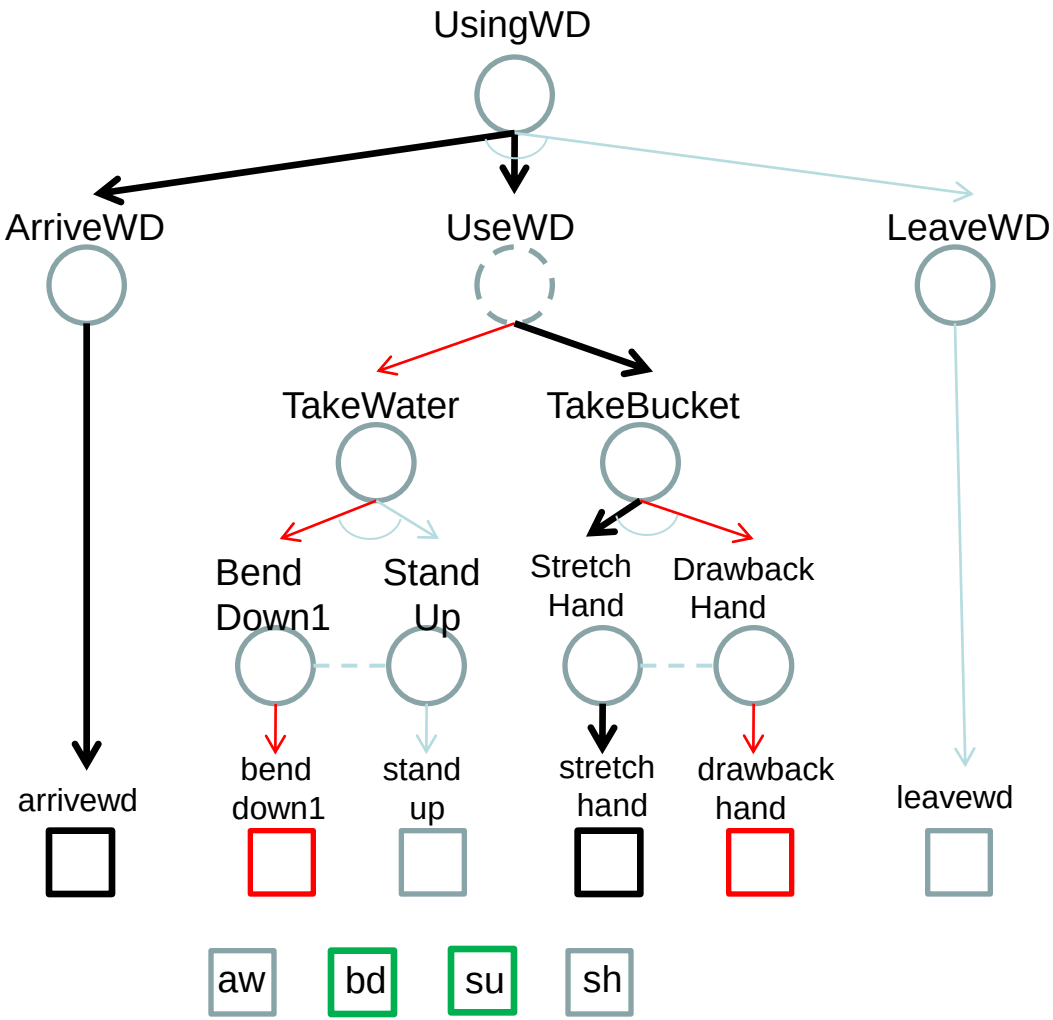


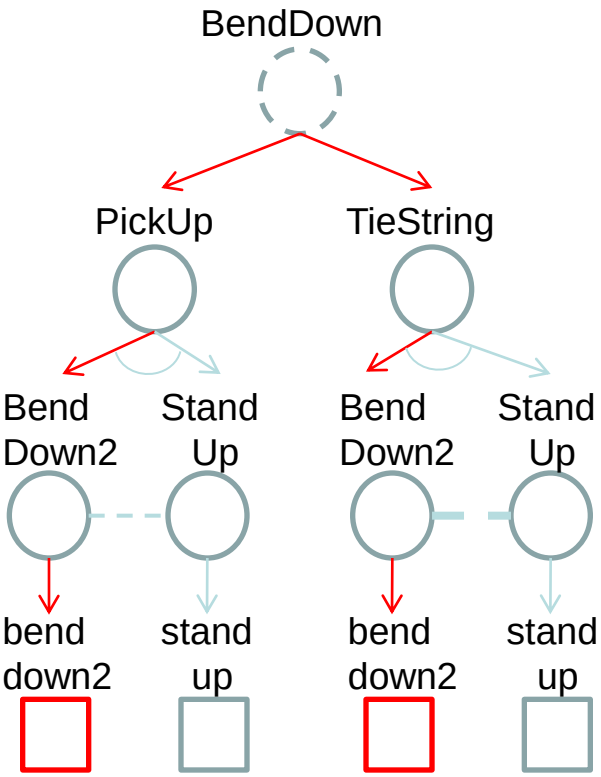
The parsing process



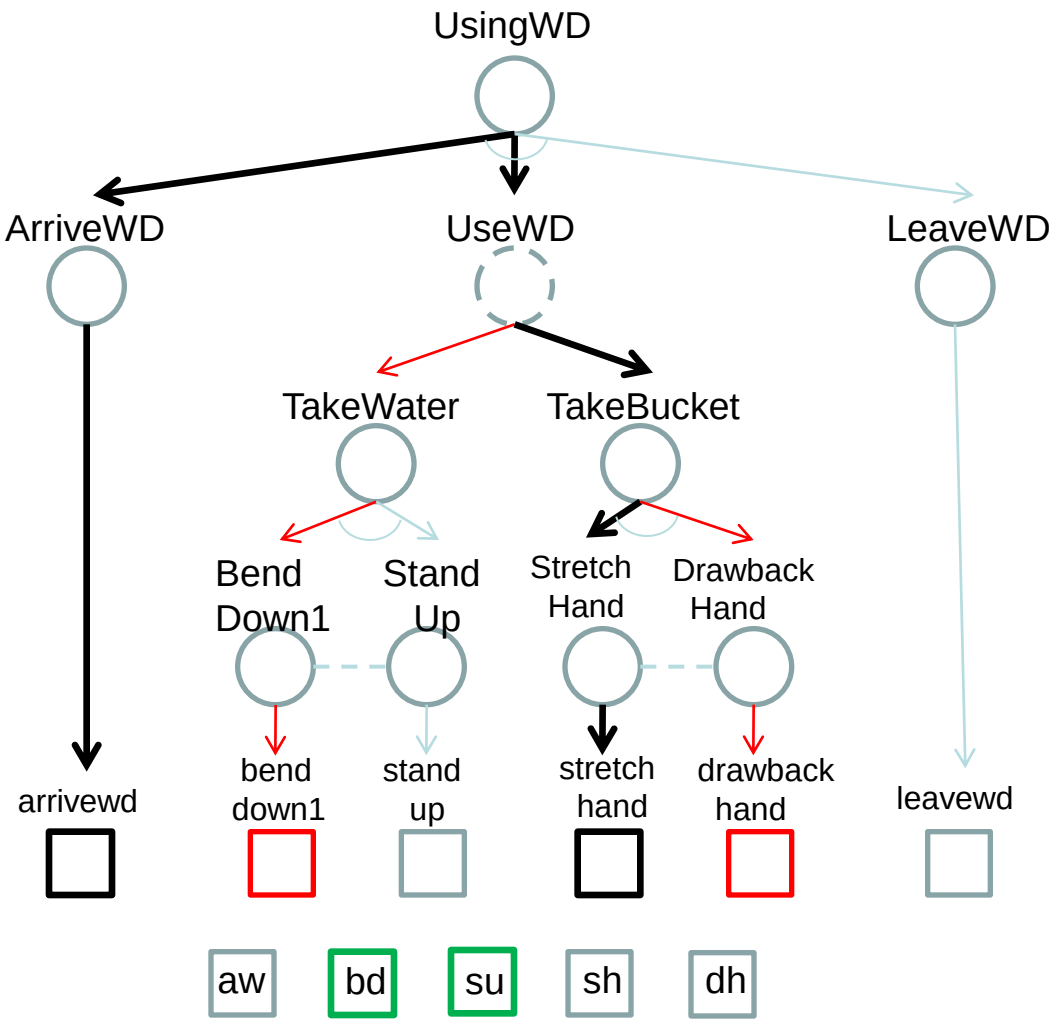


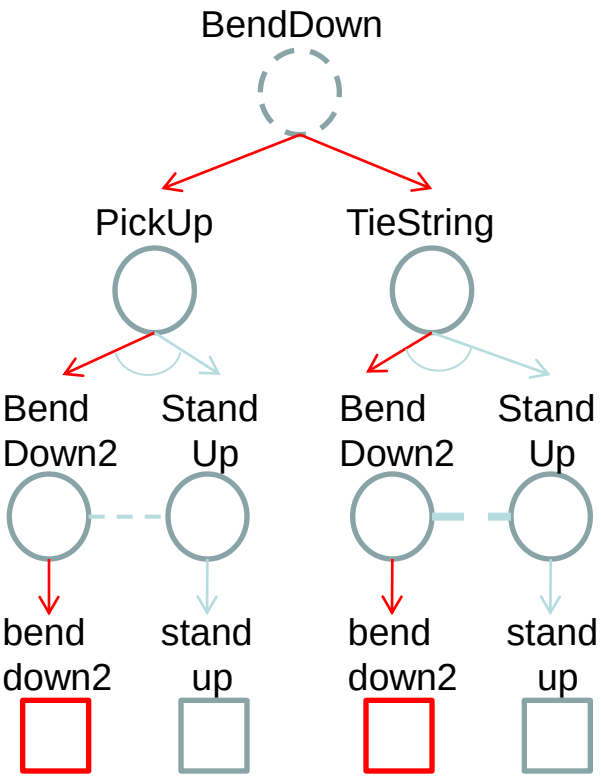
The parsing process



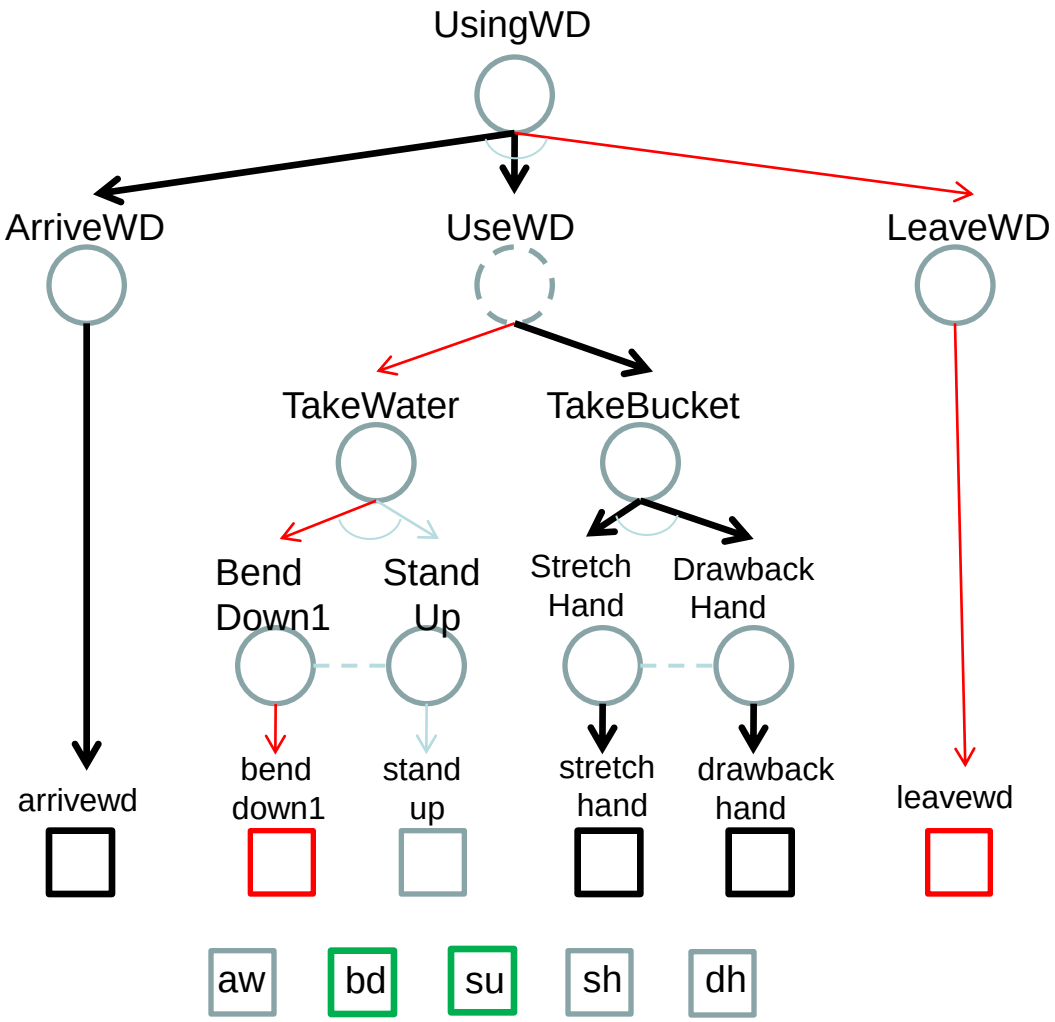


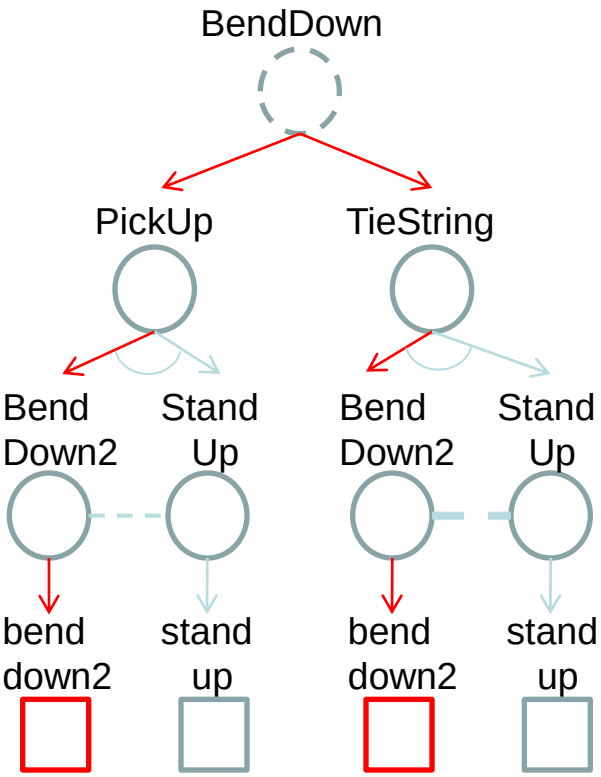
The parsing process



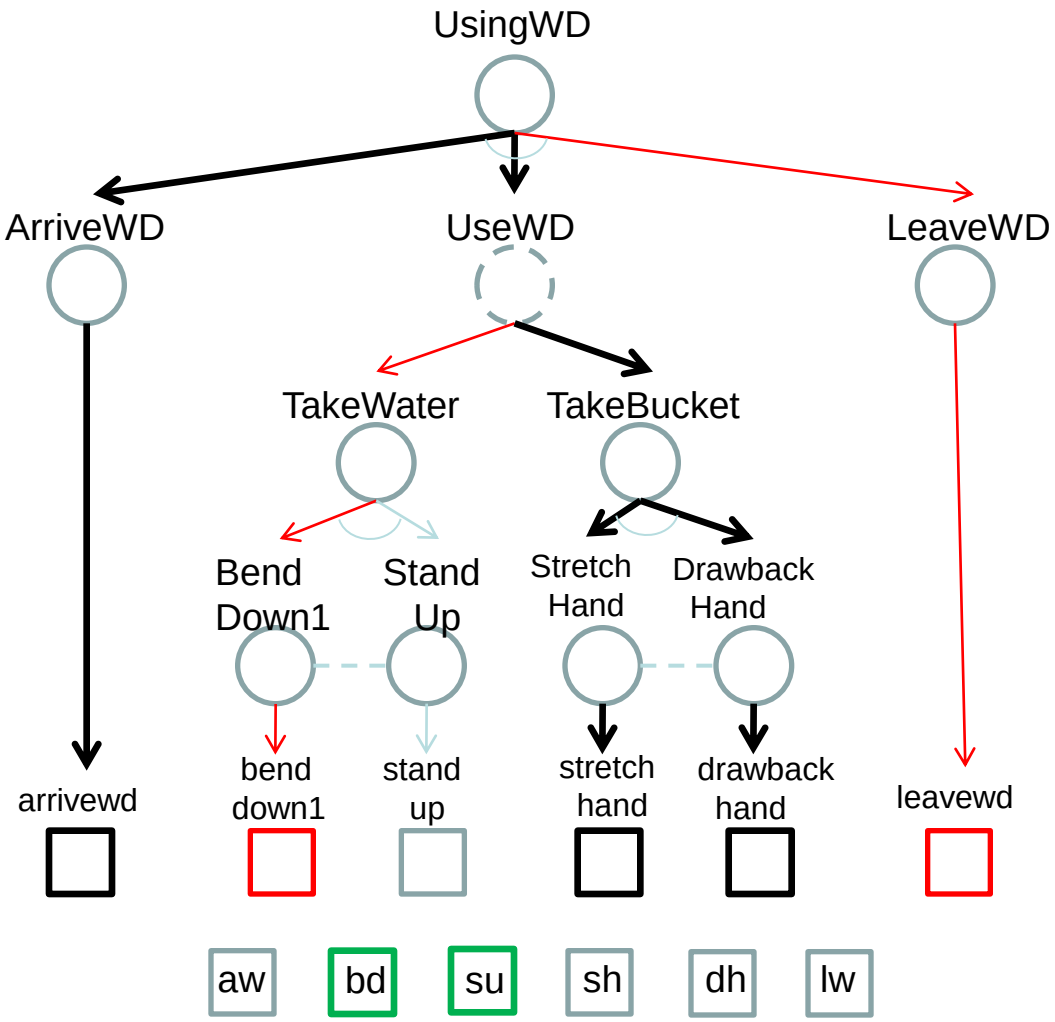


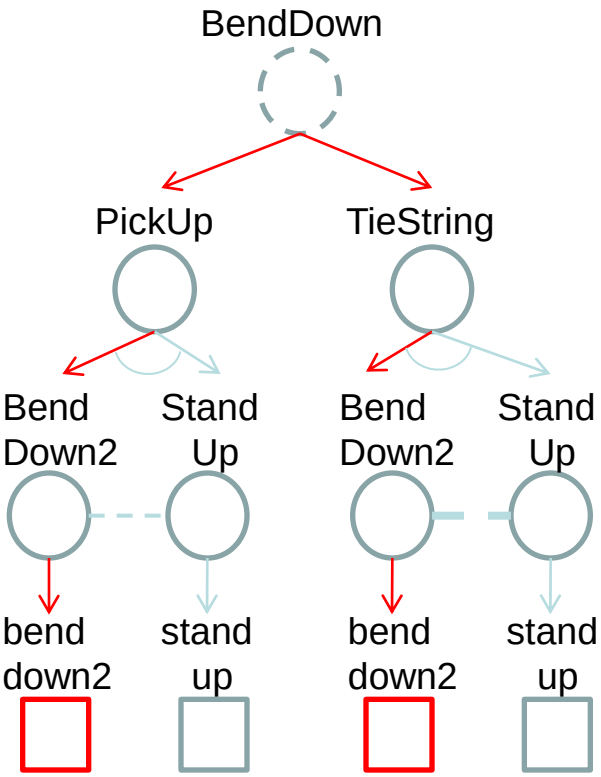
The parsing process



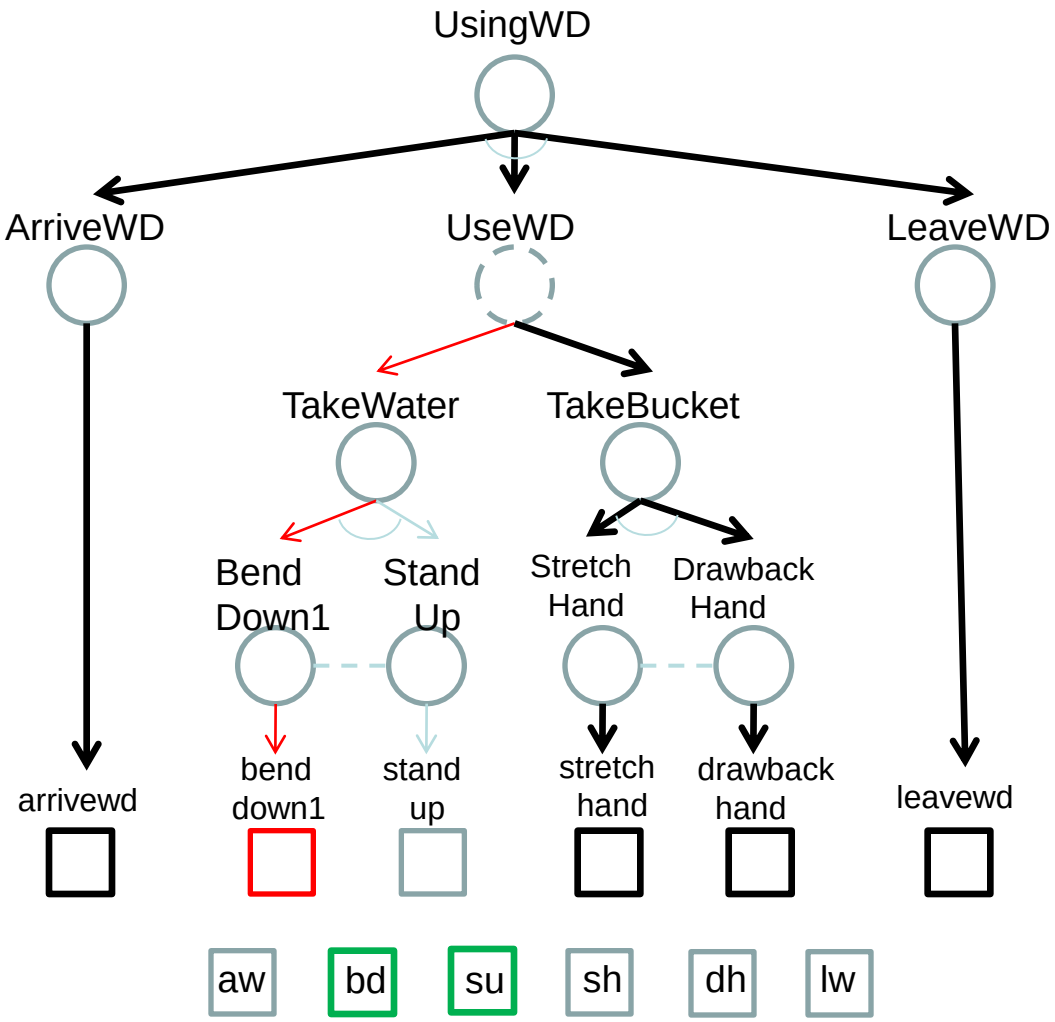


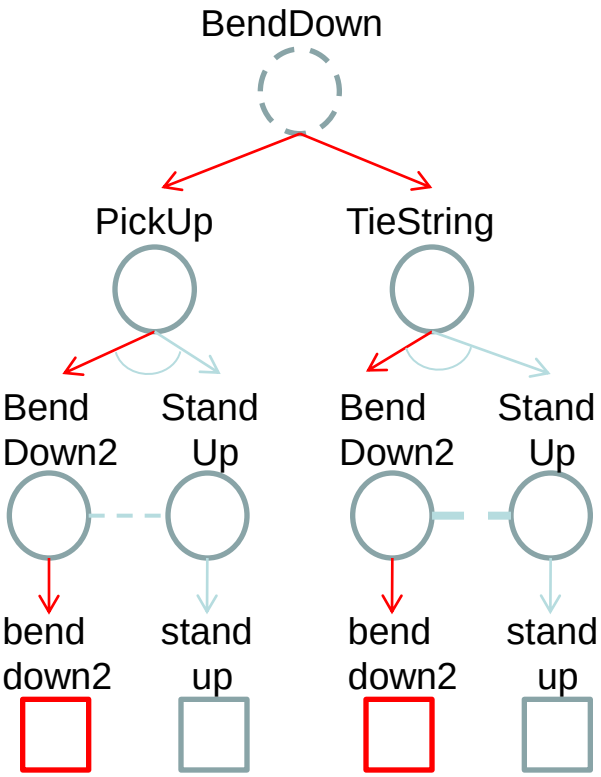
The parsing process



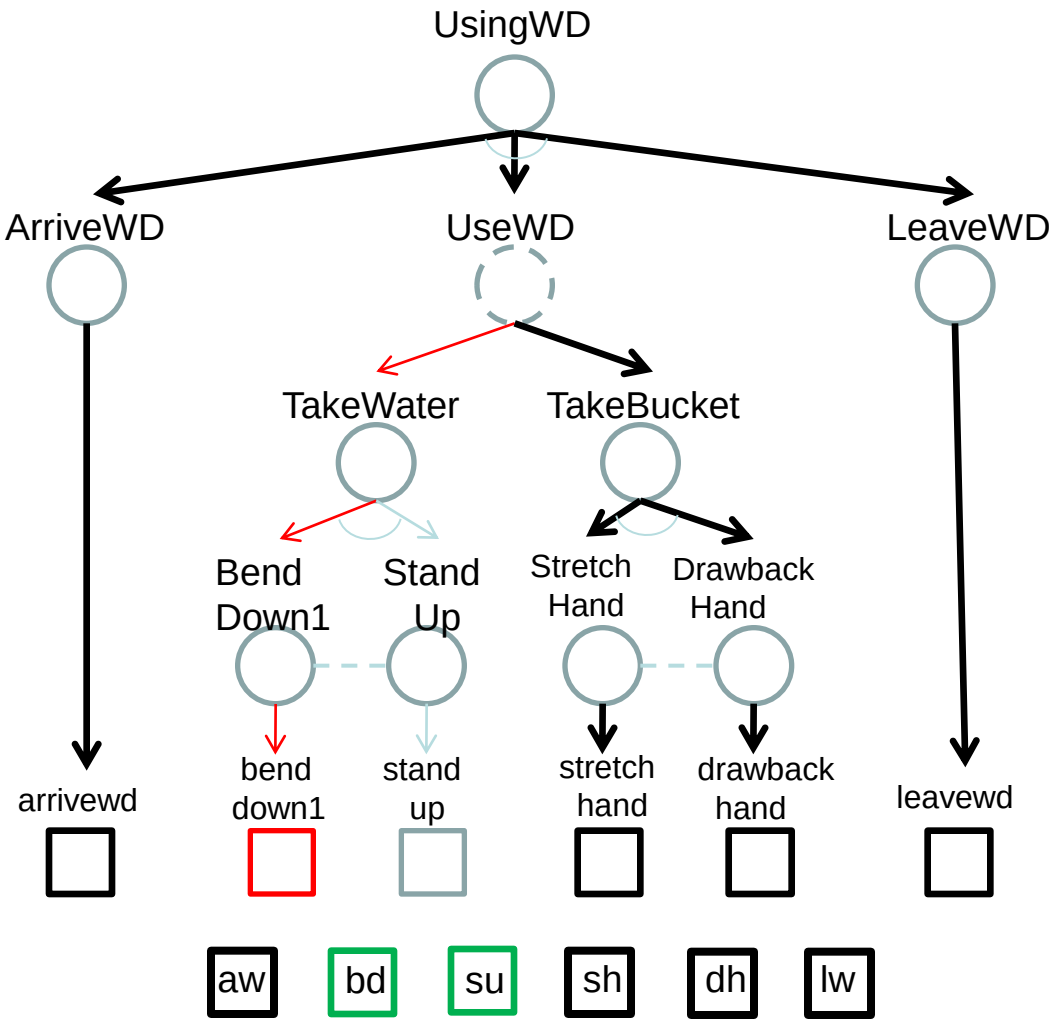


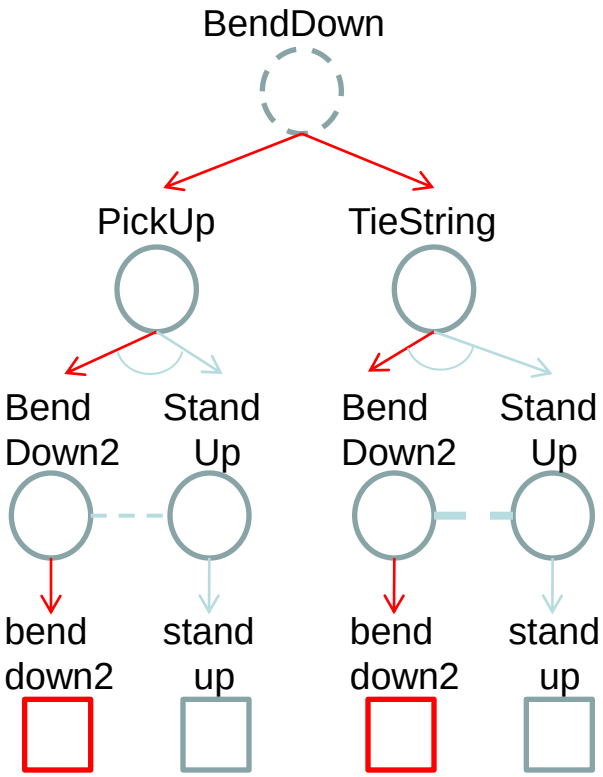
The parsing process





The parsing process





Goal: Recognize events in daily scenes

For example, in an office scene

Work by Mingtao Pei, UCLA

Challenges

1. Events happen over an extended time period
   - Variant time-span
   - Could be interrupted
   - Multiple routes
   - Intention and prediction

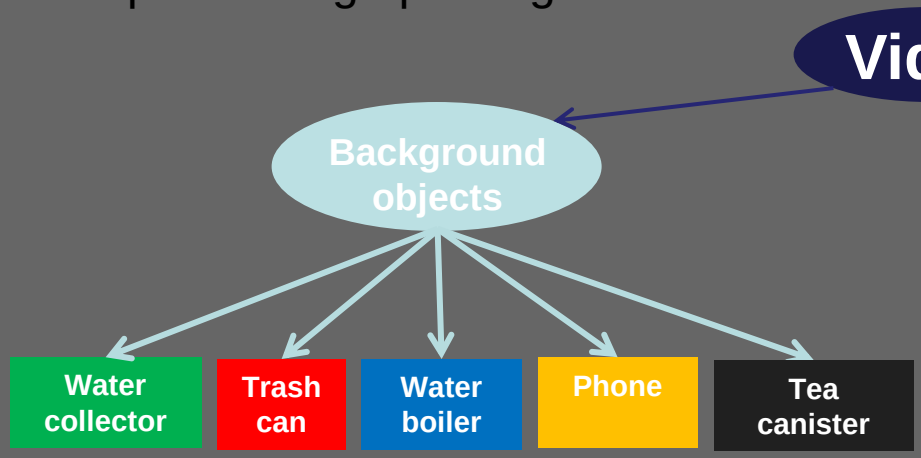
2. Actions are hard to recognize
   - Subtle and similar
   - No salient motion/pose at most of the time
   - Contextual OBJCTS are key!!

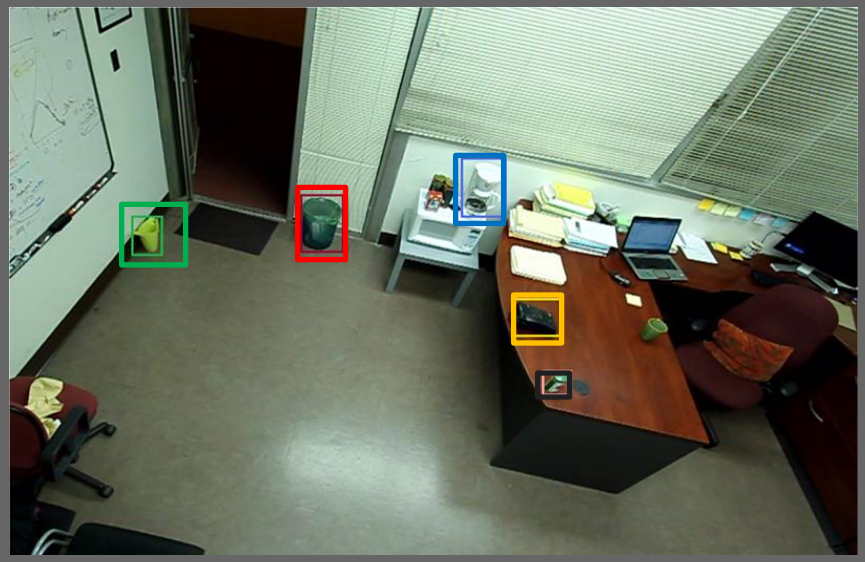
Use laptop | Read book | Dump water | Use microwave

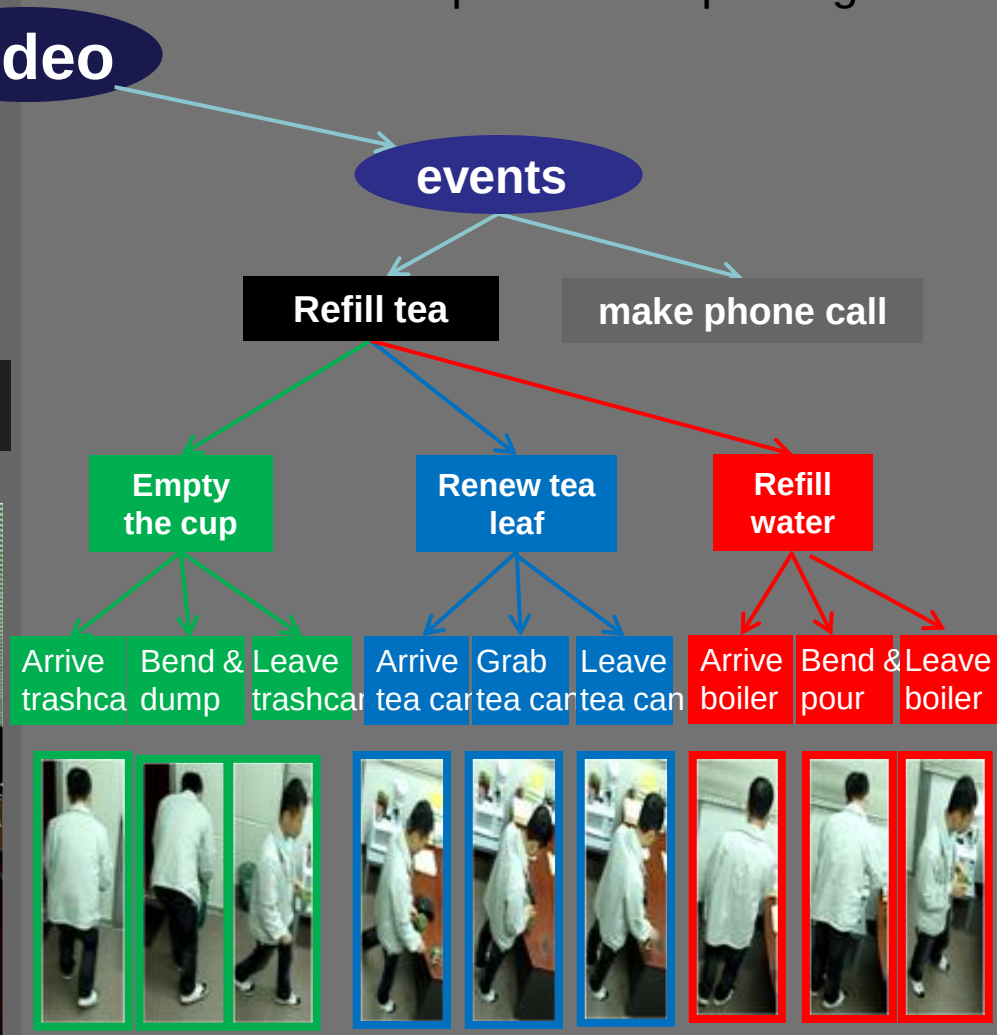
Overview of our approach

Spatial image parsing

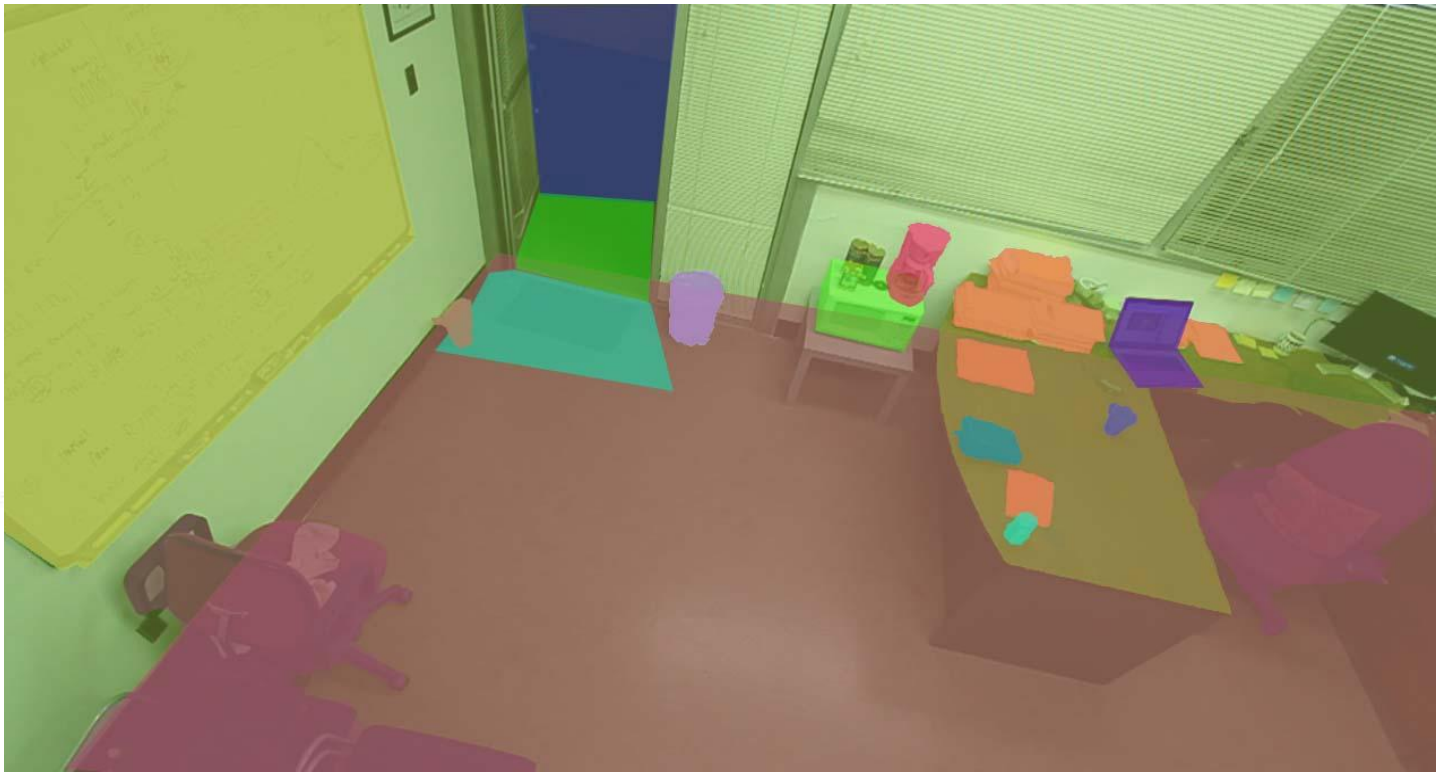
Temporal event parsing



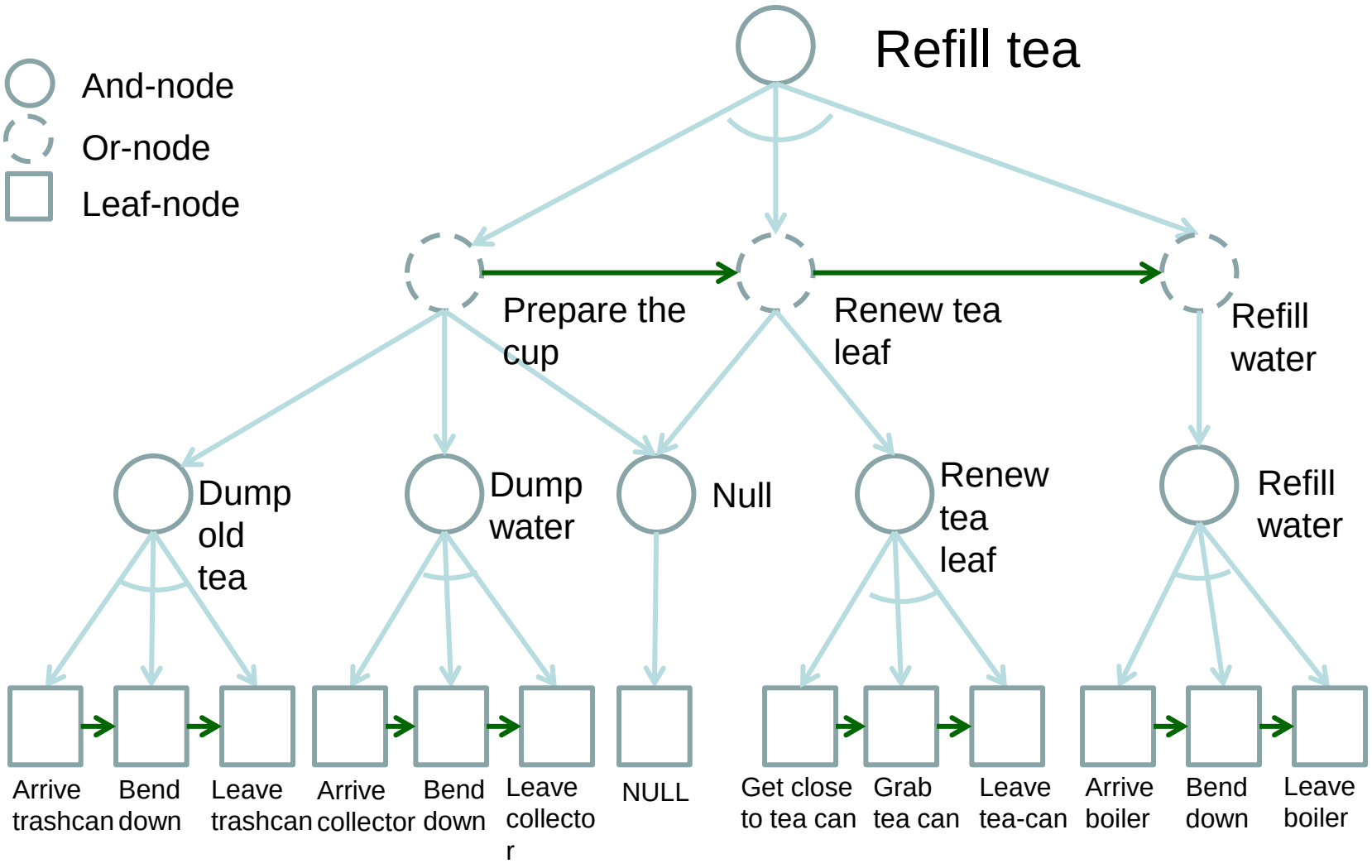




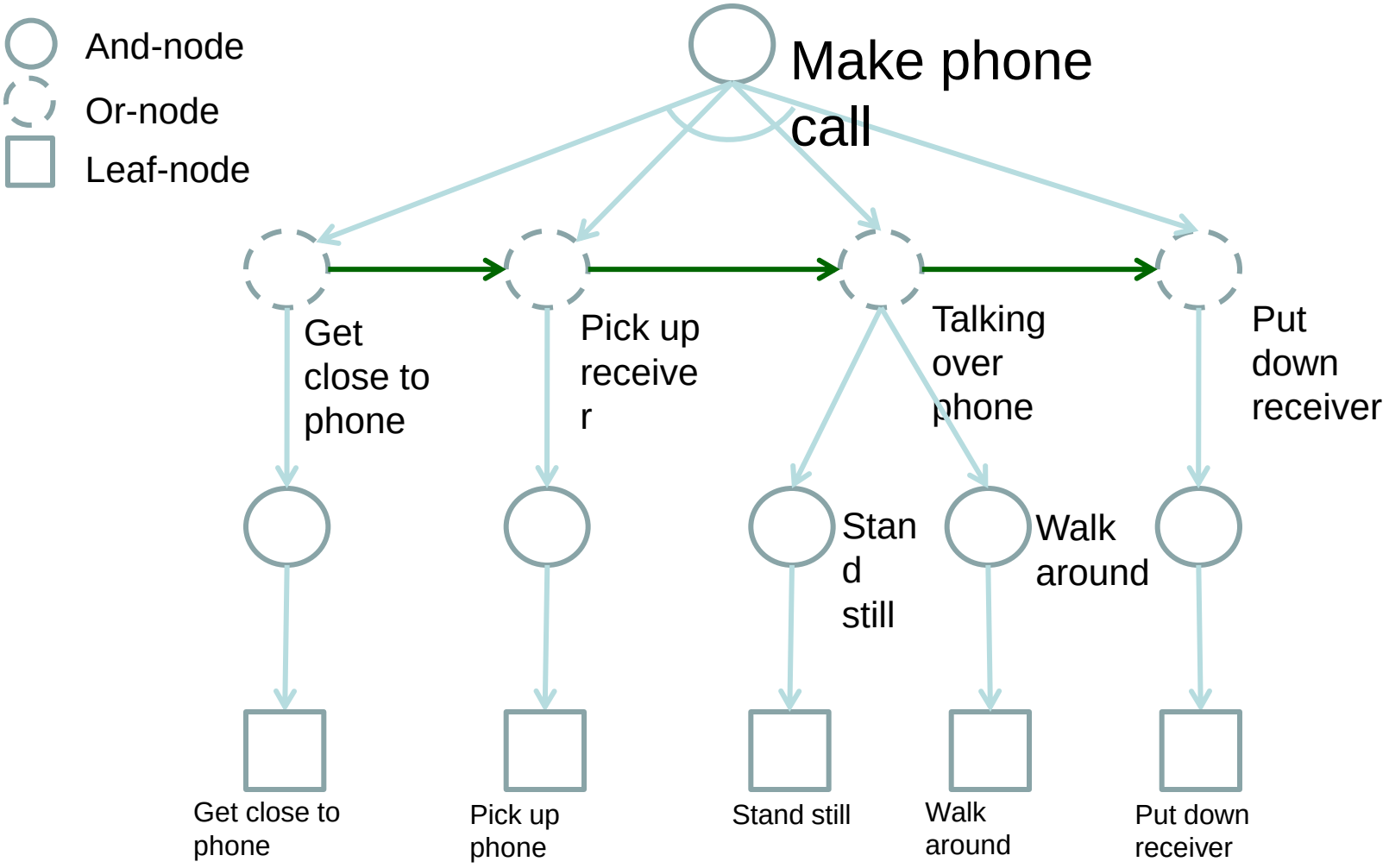
Scene parsing



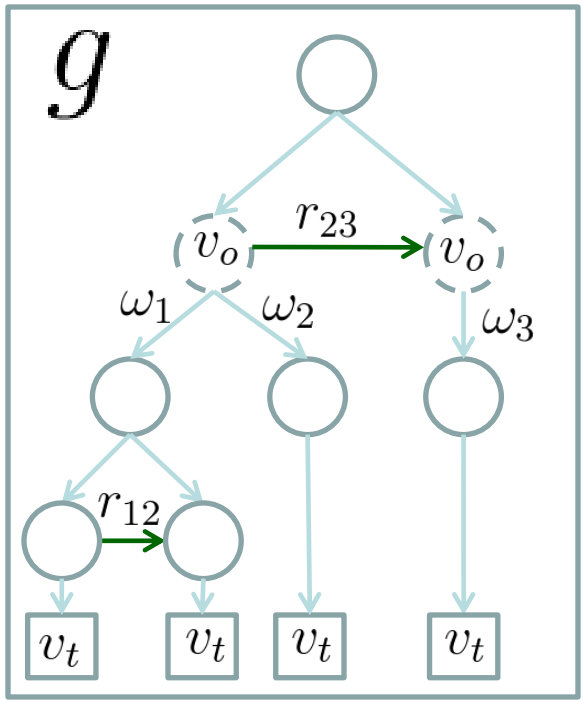
Event as temporal And-Or-Graph



Event as temporal And-Or-Graph



Formulation



$$p(g) = \frac{1}{Z} exp\{score(g)\}$$

Gramma

r

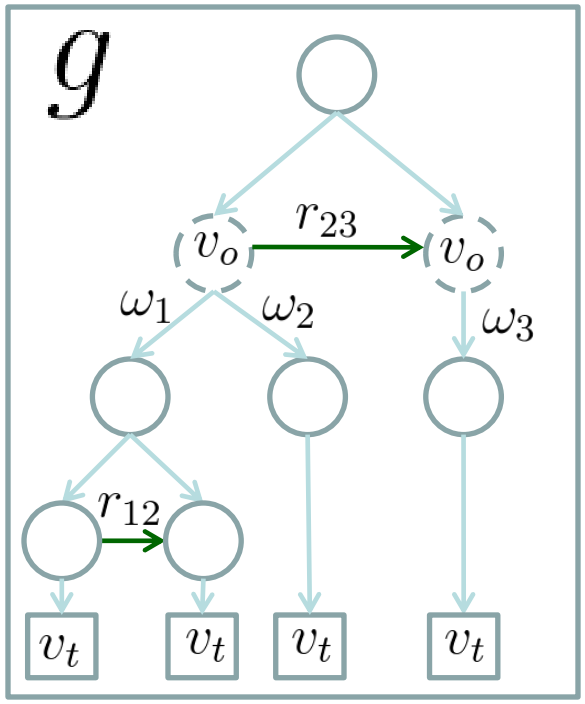
$$score(g) = \sum_{v_t \in T(g)} \lambda_{v_t} \alpha(v_t) + \sum_{v \in V_o(g)} \lambda_v \omega(v) + \sum_{(i,j) \in E(g)} \lambda_{ij} r_{ij}(v_i, v_j)$$

Data term

Or node Frequency term

Temporal Relations

Formulation



$$p(g) = \frac{1}{Z} exp\{score(g)\}$$

Gramma

r

$$score(g) = \sum_{v_t \in T(g)} \lambda_{v_t} \alpha(v_t) + \sum_{v \in V_o(g)} \lambda_v \omega(v) + \sum_{(i,j) \in E(g)} \lambda_{ij} r_{ij}(v_i, v_j)$$

Data term

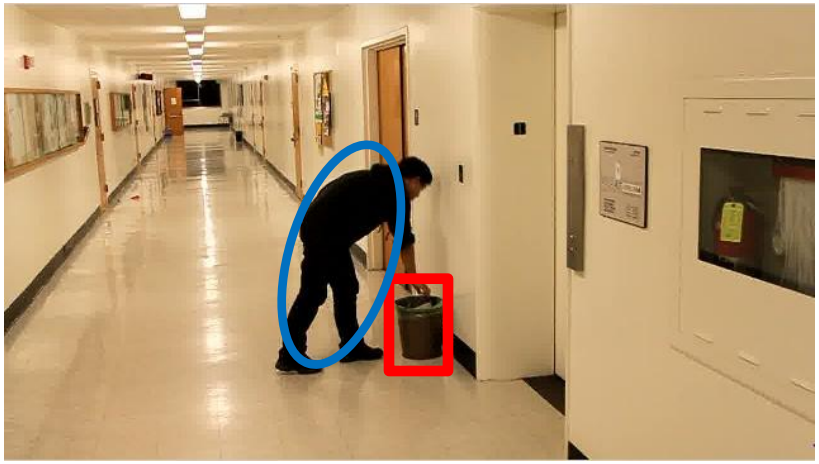
Or node Frequenc

y

Temporal Relations

$$\alpha(v_t) = \sum_{i \in \mathcal{F}} \beta_i, h_i(v_t) - dist(P_{person}, P_{obj})$$

Combing action and contextual object



Bend down action

+

Close to Trash Can

=

Event1:
Drop waste

Reach out action

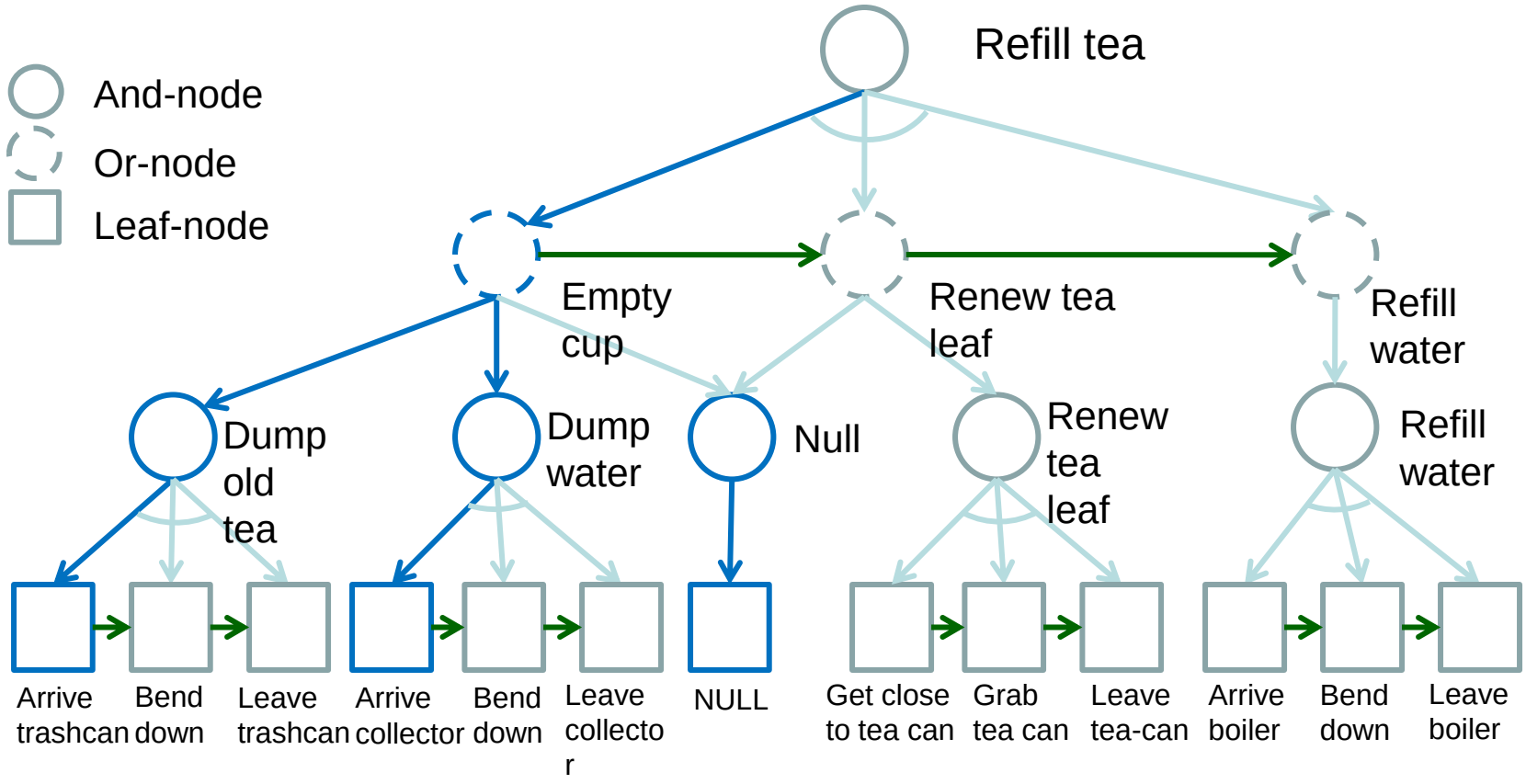
+

Close to Elevator

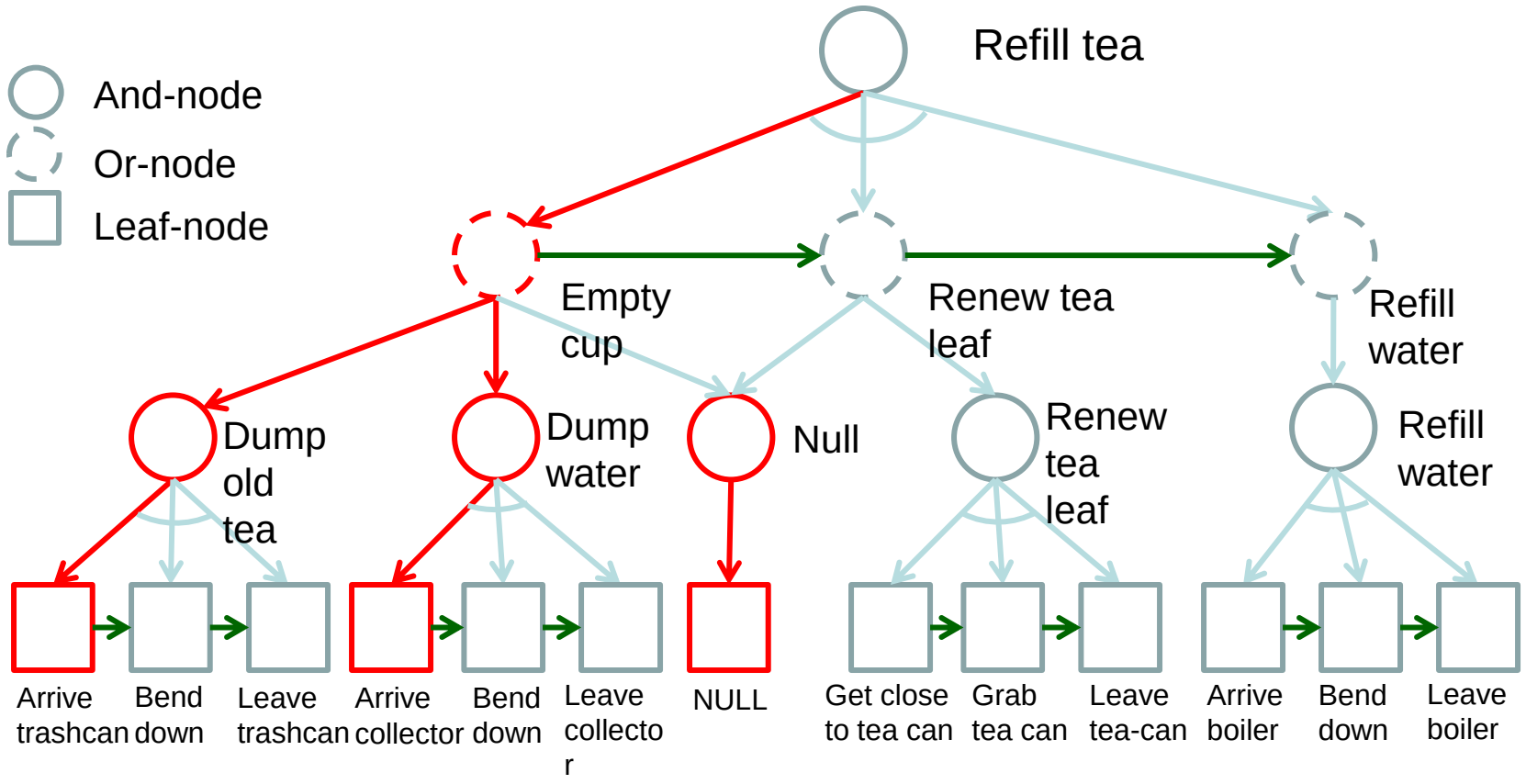
=

Event2:
Use elevator

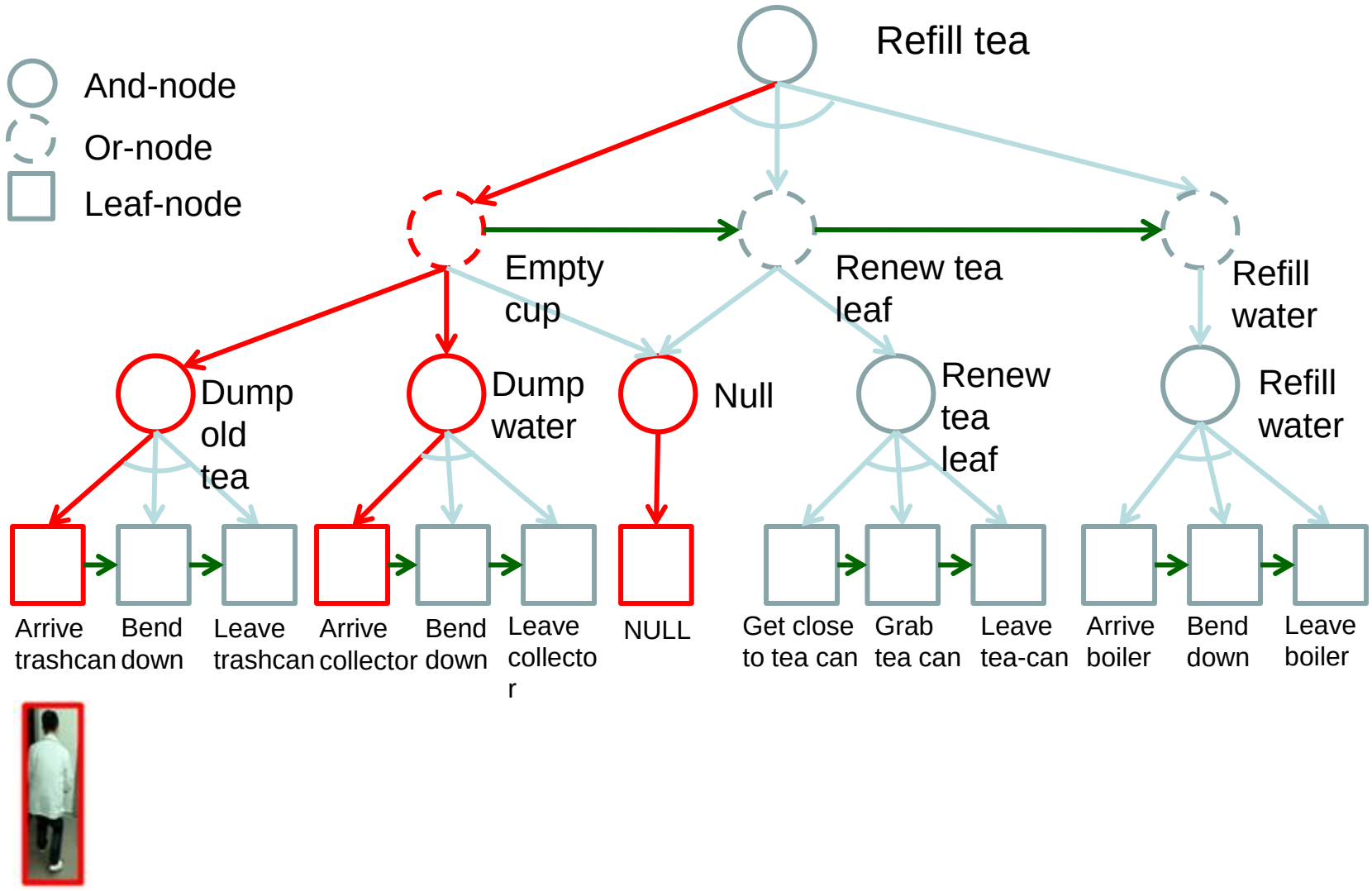
Parsing process (Earley Parser [Earley 1970])



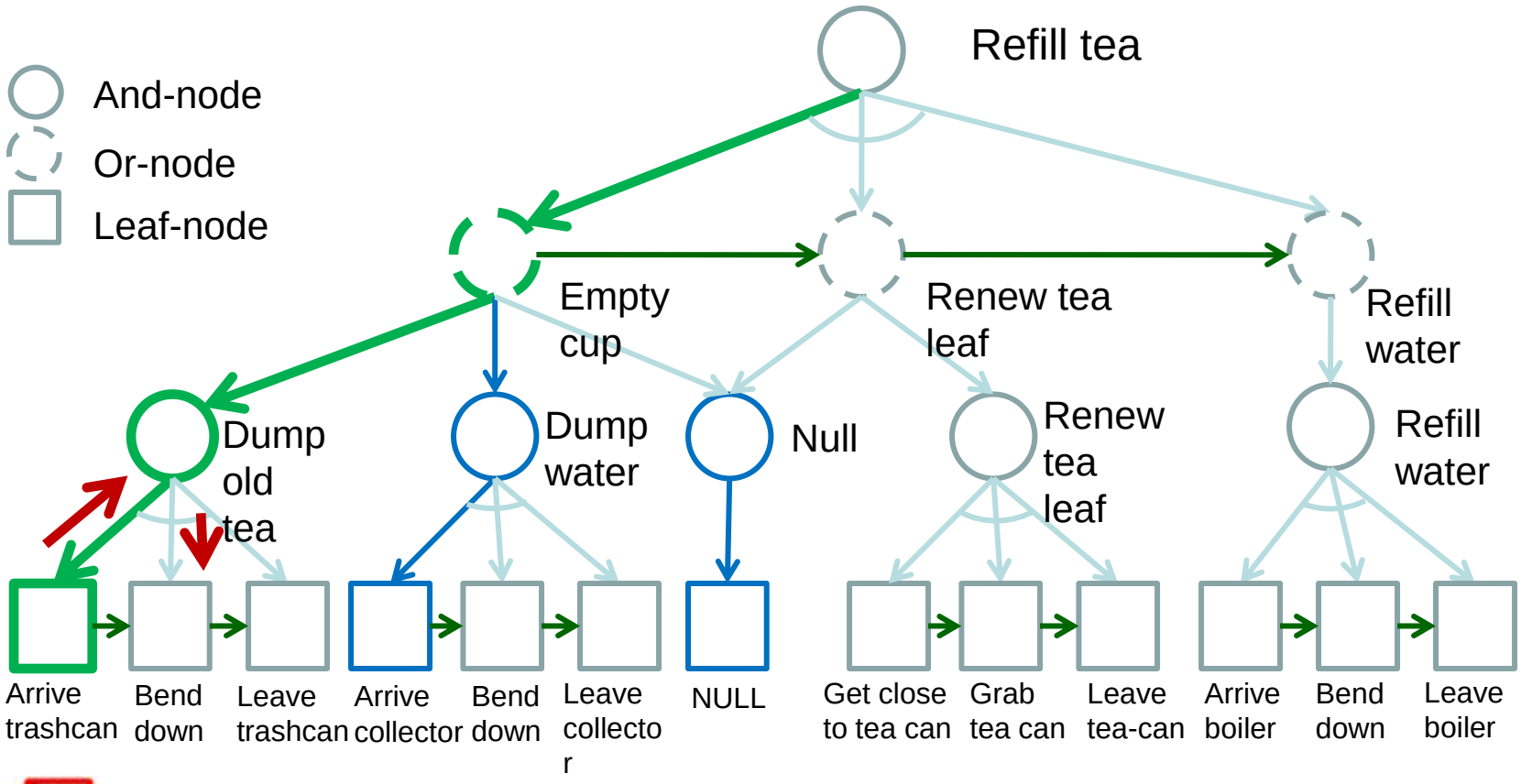
Parsing process (Earley Parser [Earley 1970])



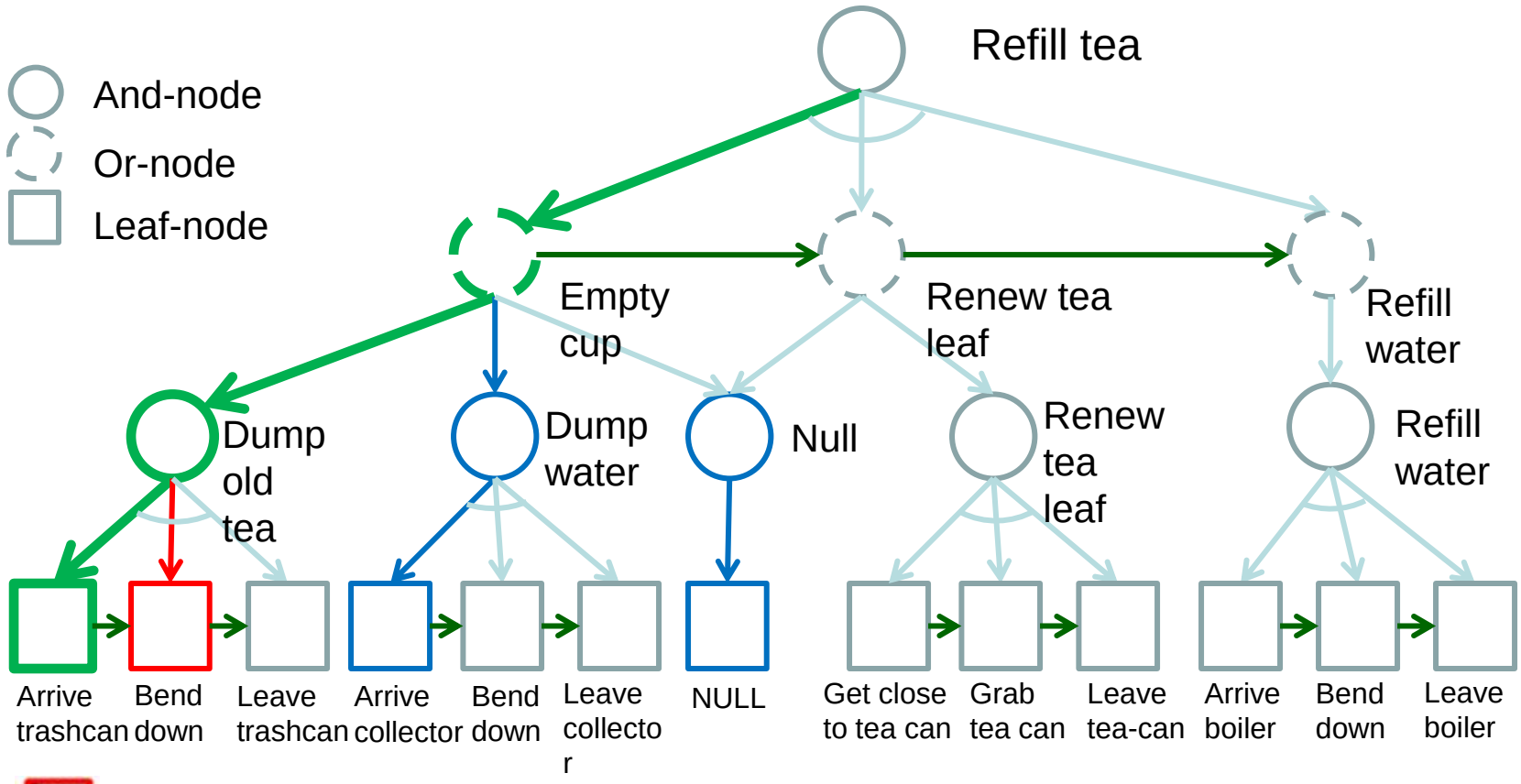
Parsing process (Earley Parser [Earley 1970])



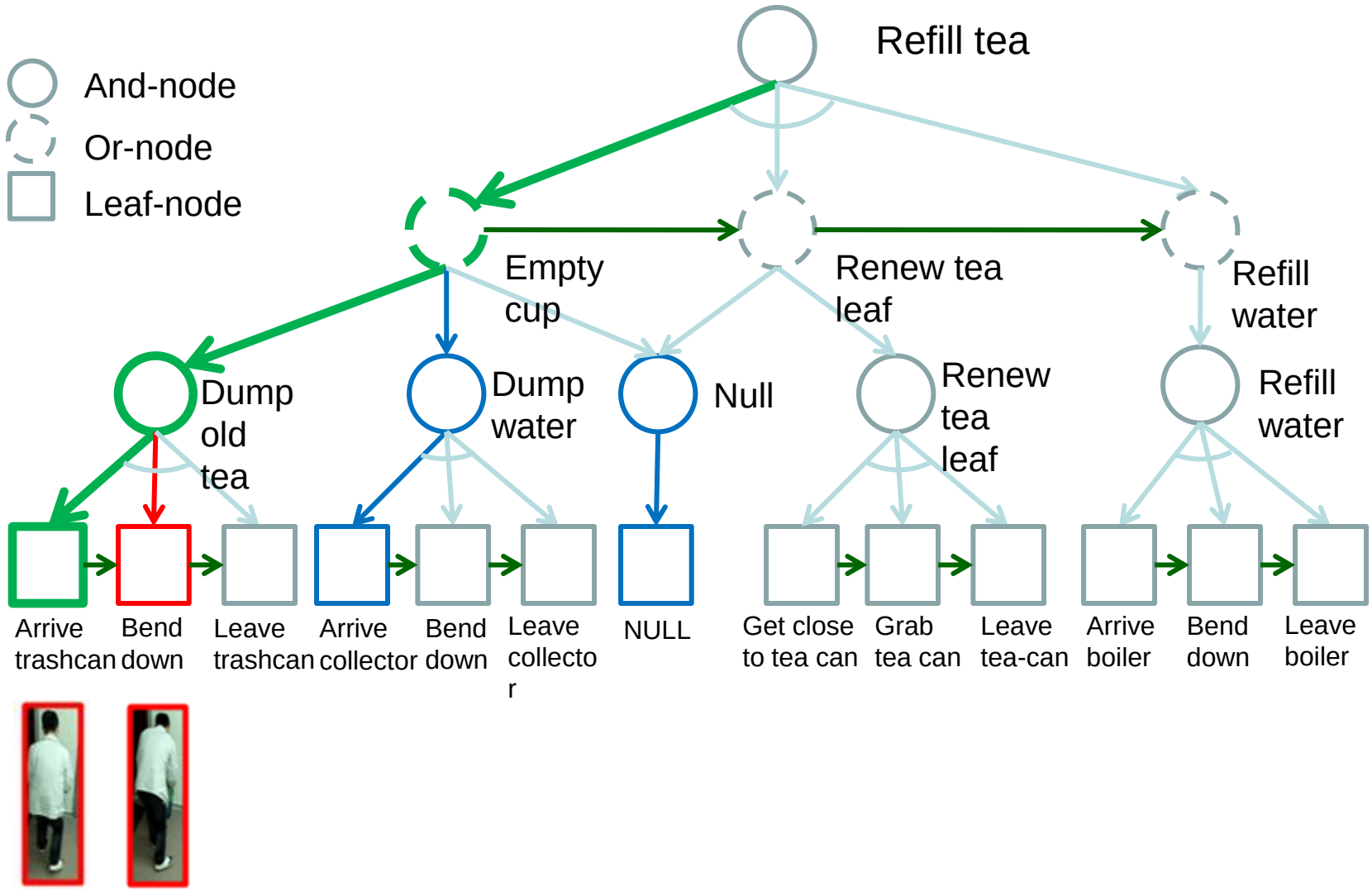
Parsing process (Earley Parser [Earley 1970])



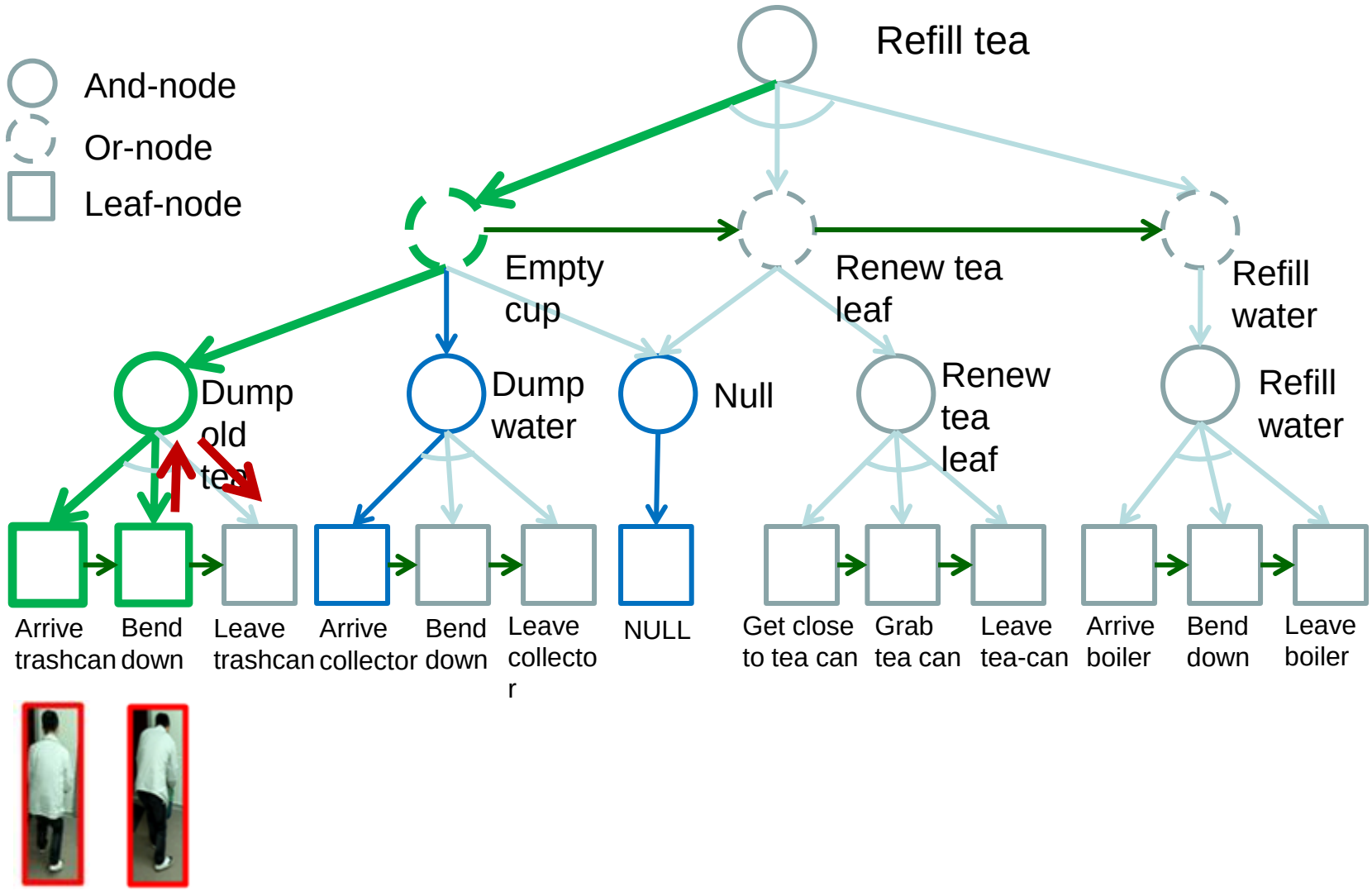
Parsing process (Earley Parser [Earley 1970])



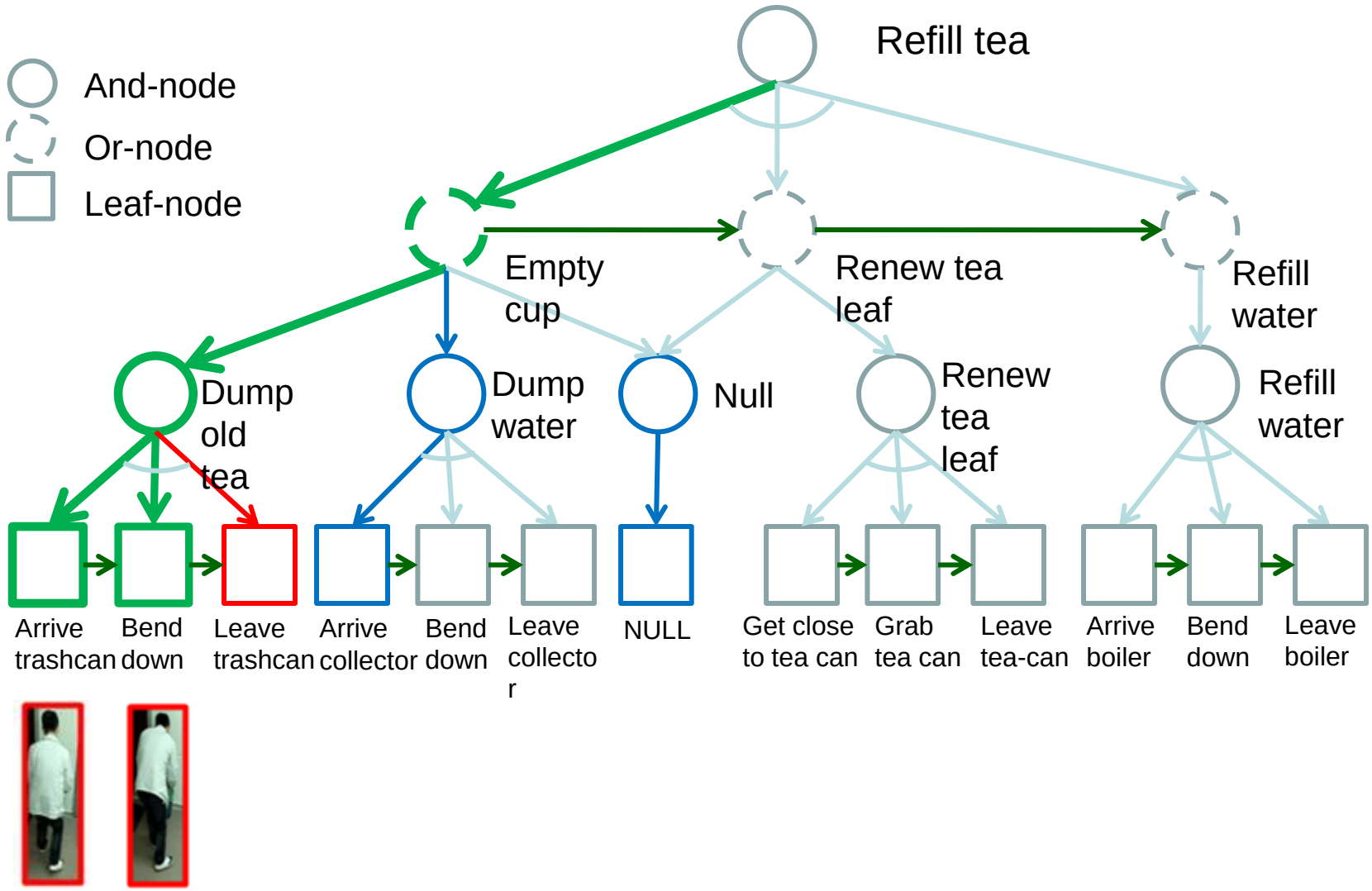
Parsing process (Earley Parser [Earley 1970])



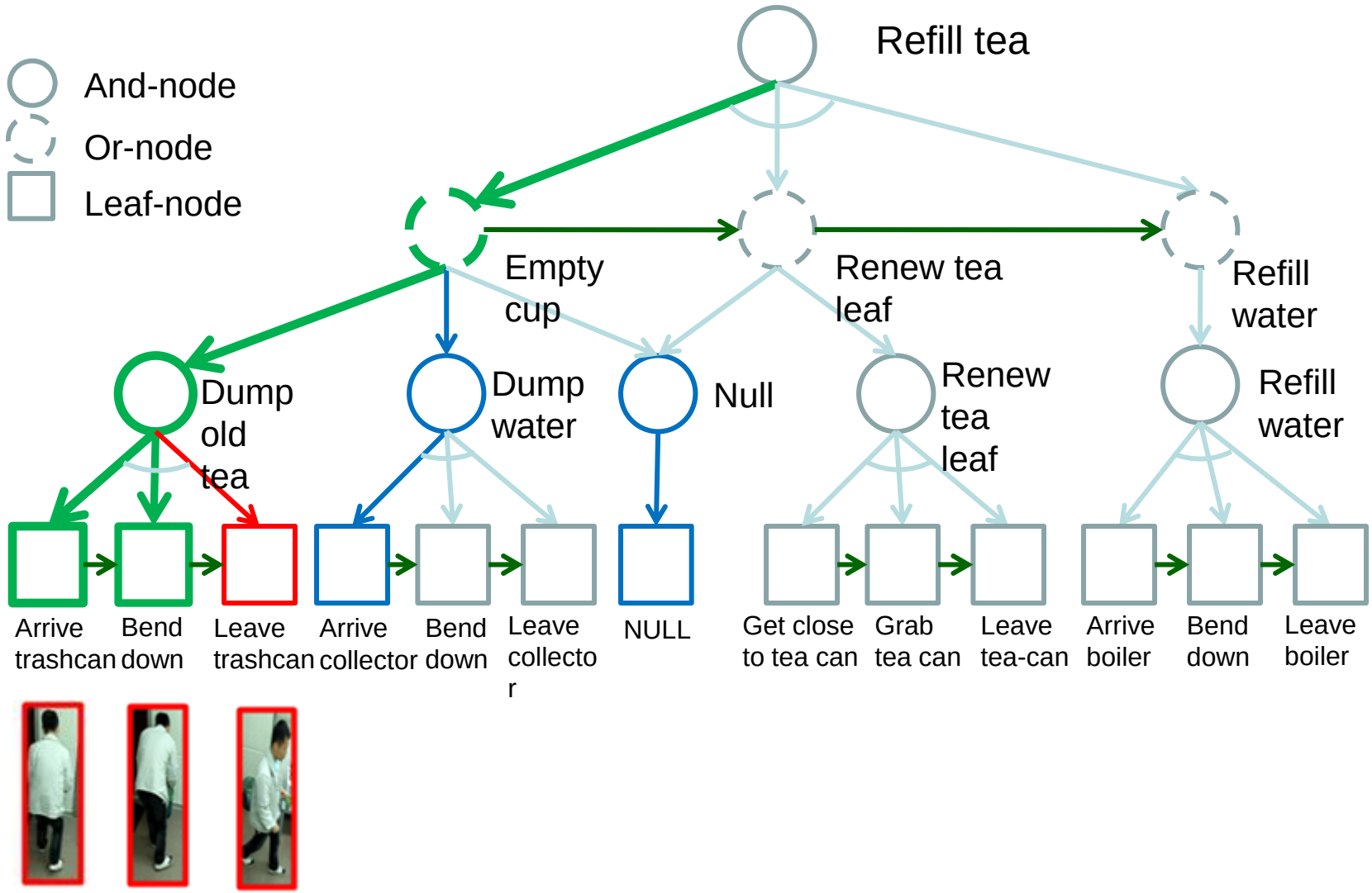
Parsing process (Earley Parser [Earley 1970])



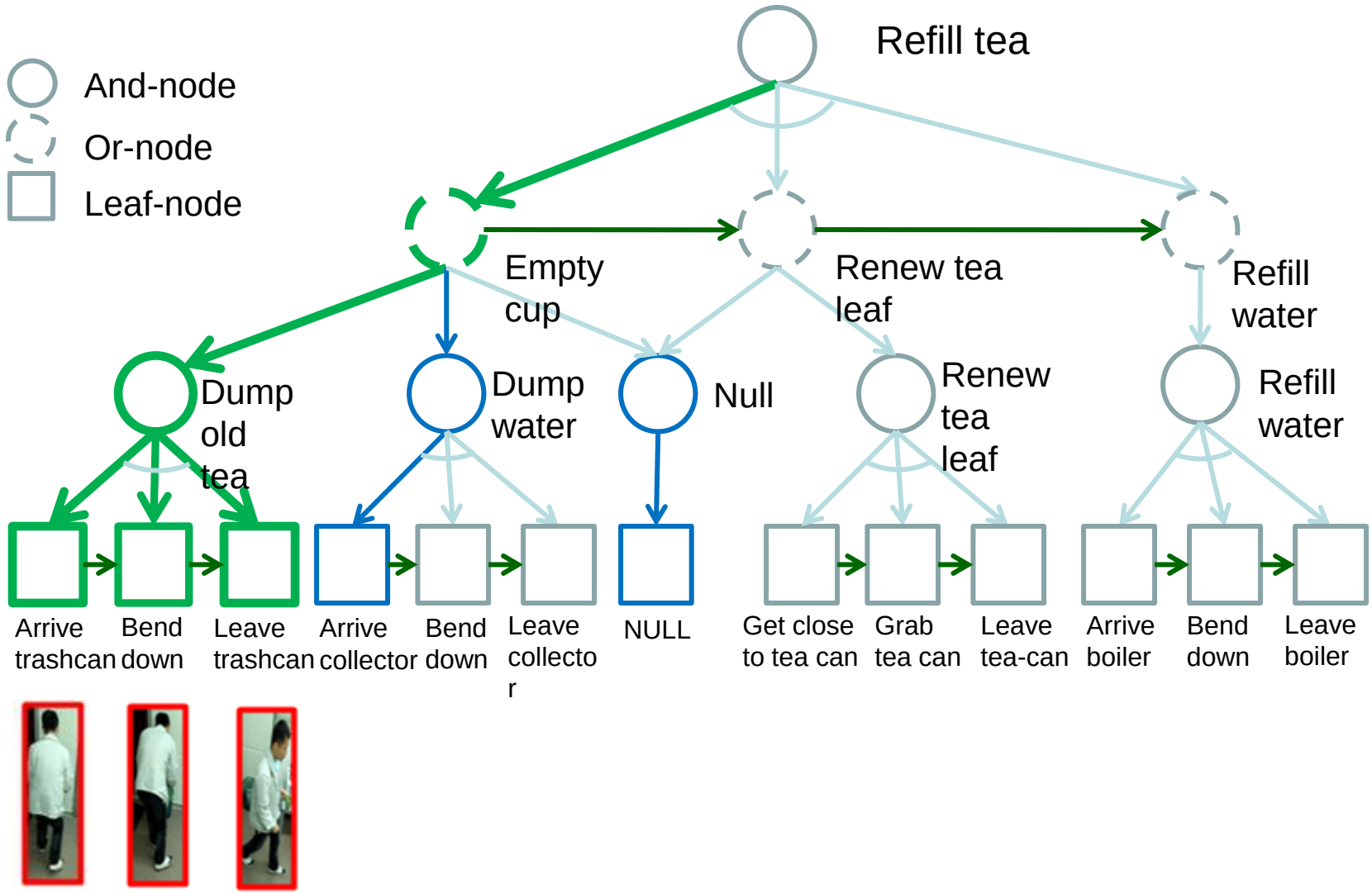
Parsing process (Earley Parser [Earley 1970])



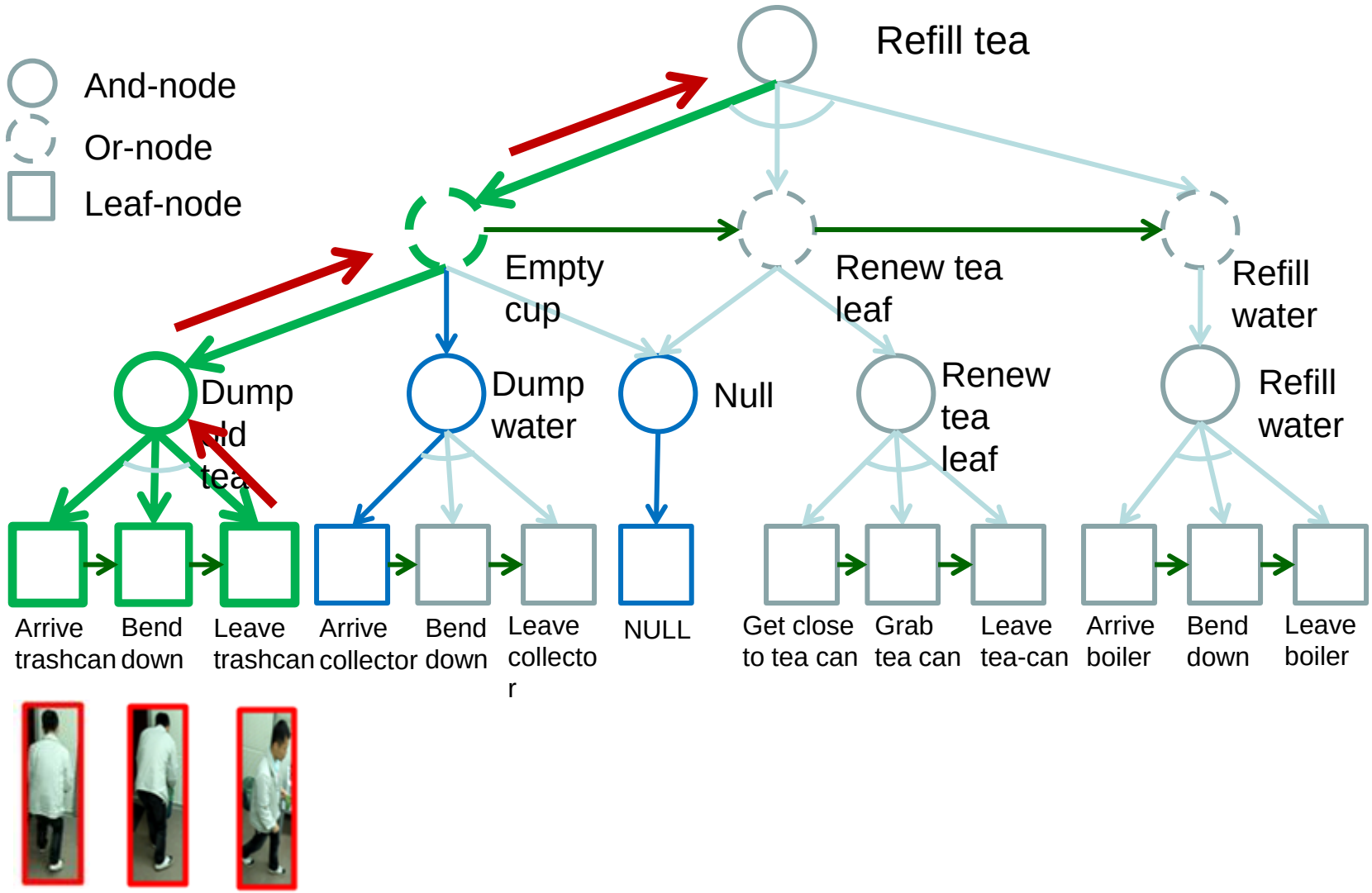
Parsing process (Earley Parser [Earley 1970])



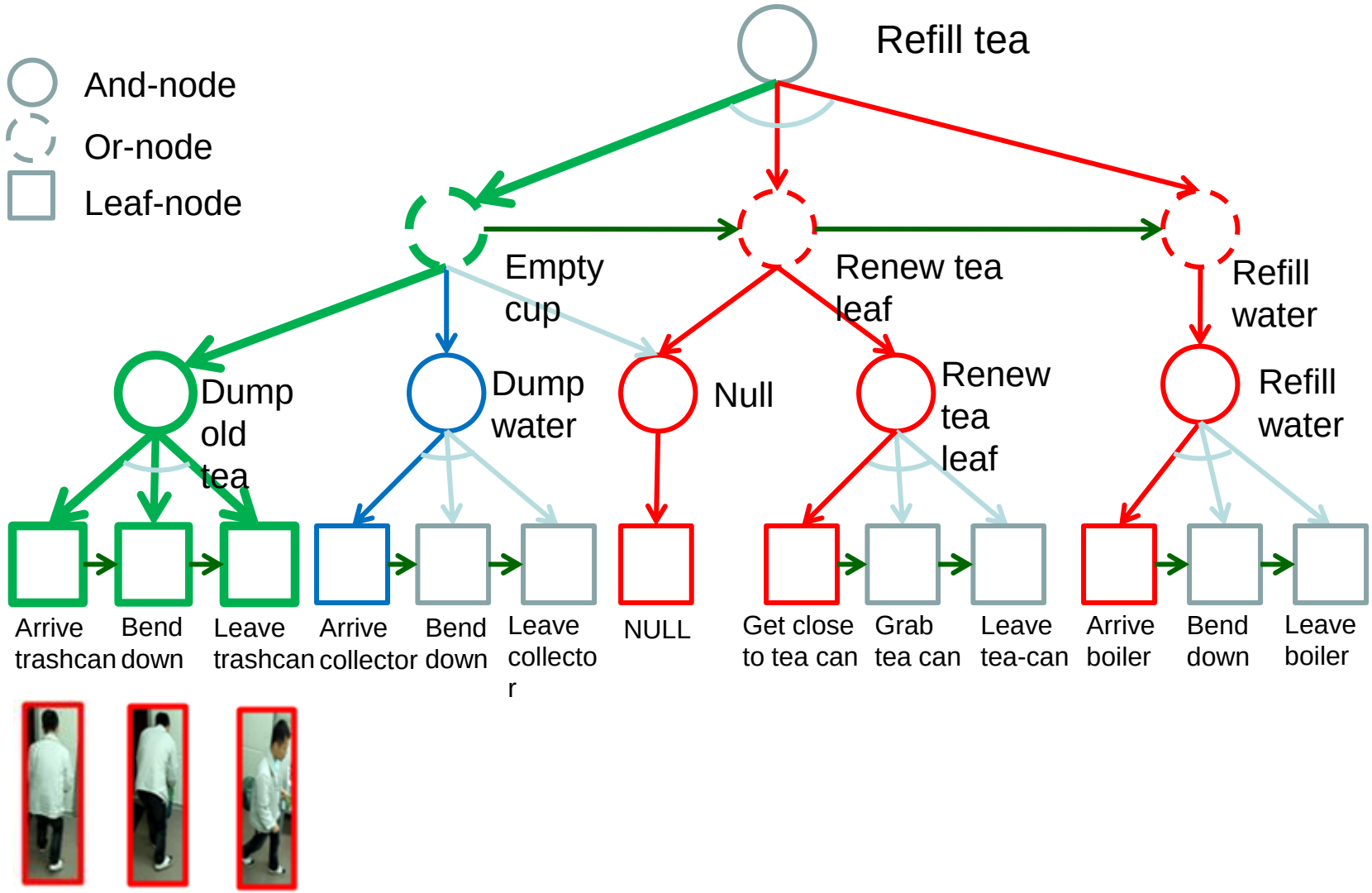
Parsing process (Earley Parser [Earley 1970])



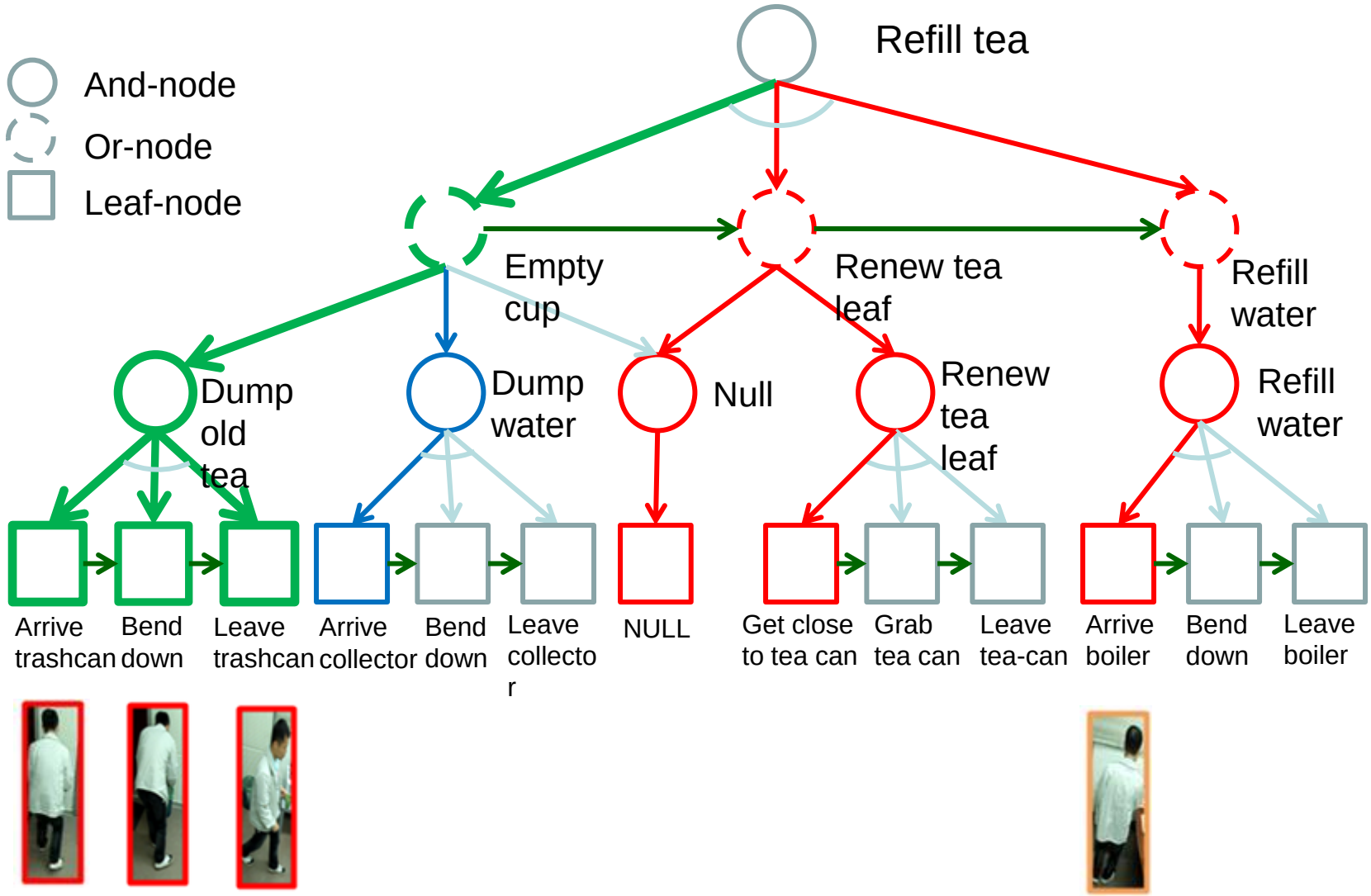
Parsing process (Earley Parser [Earley 1970])



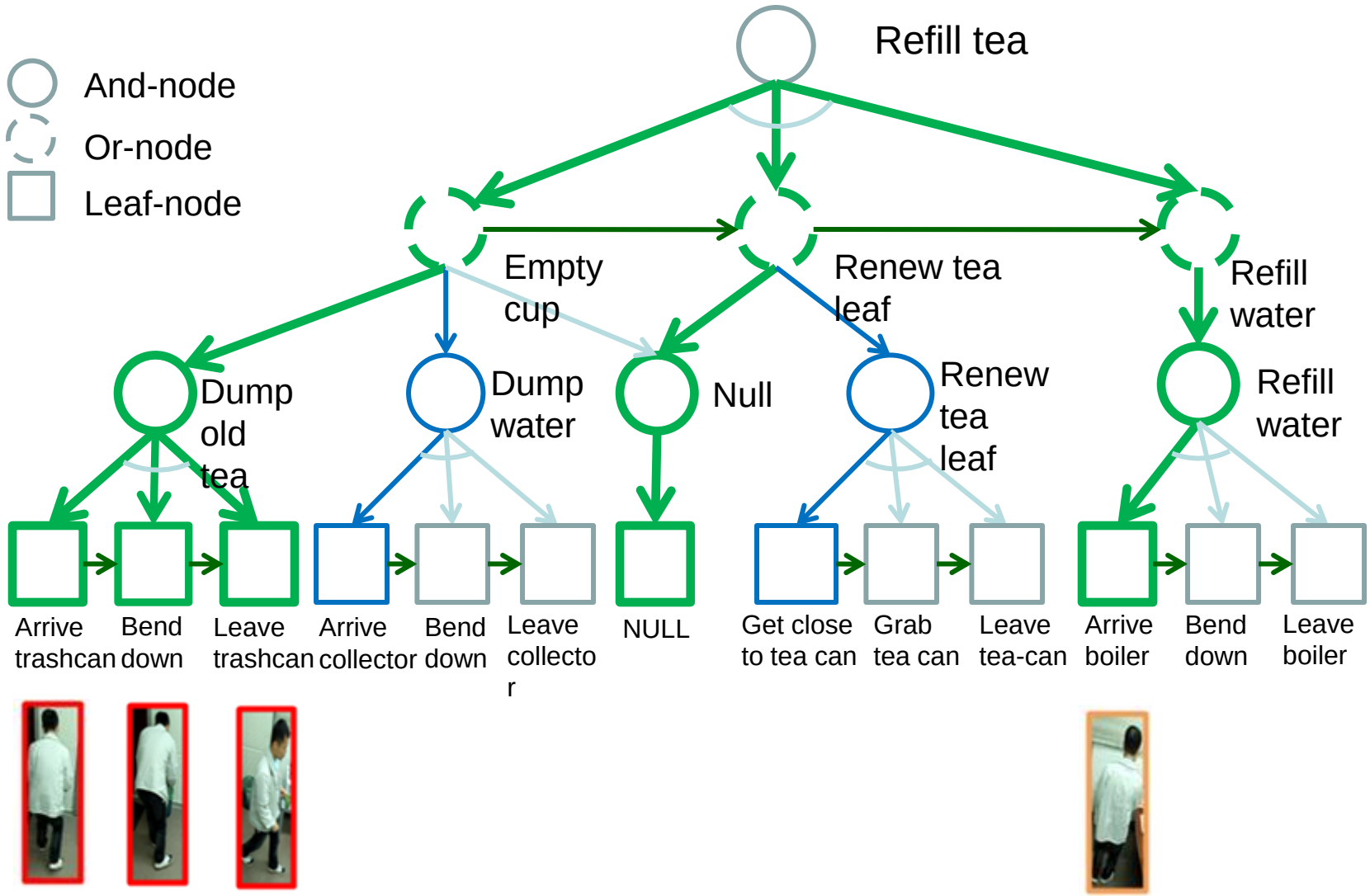
Parsing process (Earley Parser [Earley 1970])



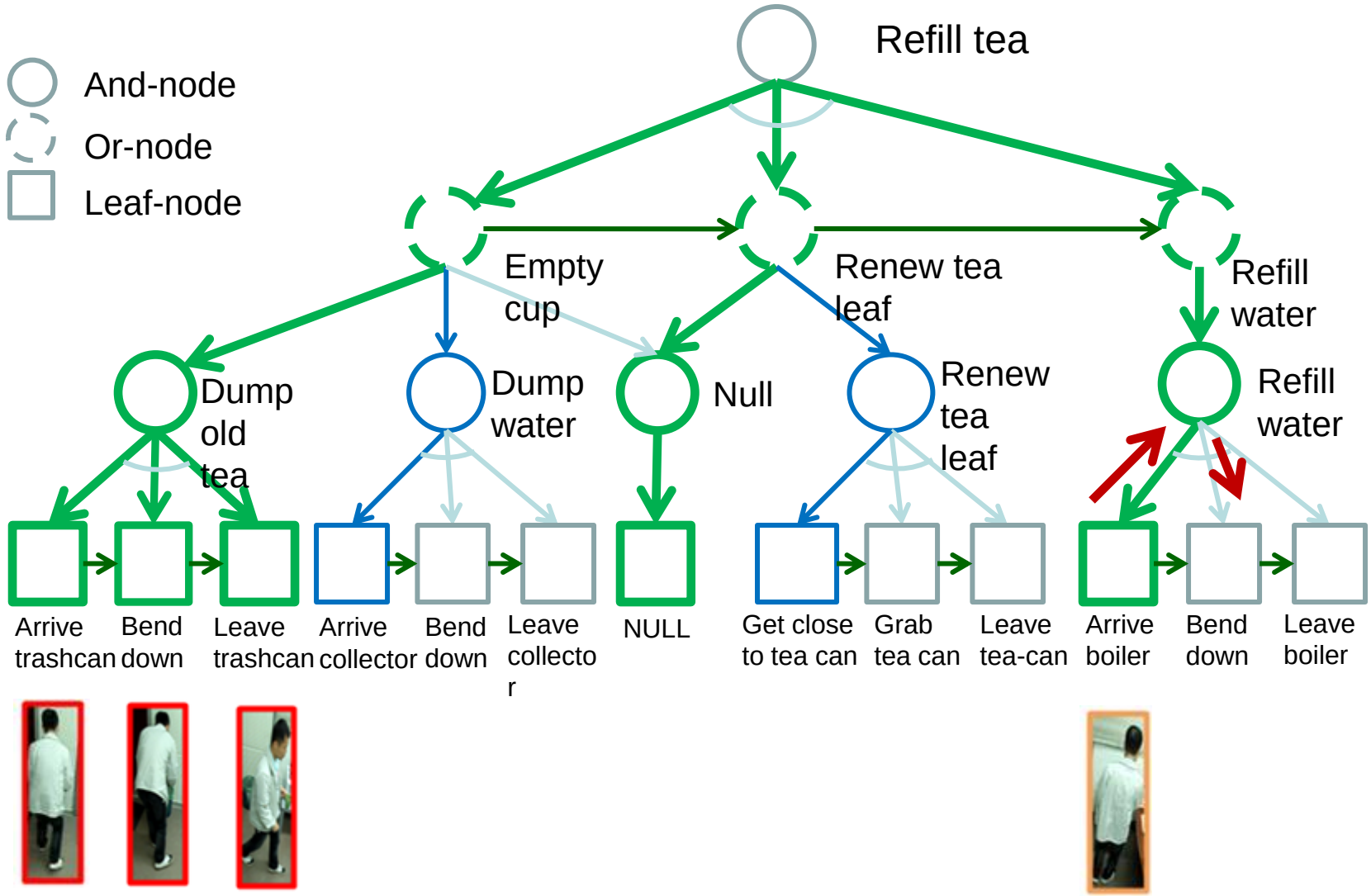
Parsing process (Earley Parser [Earley 1970])



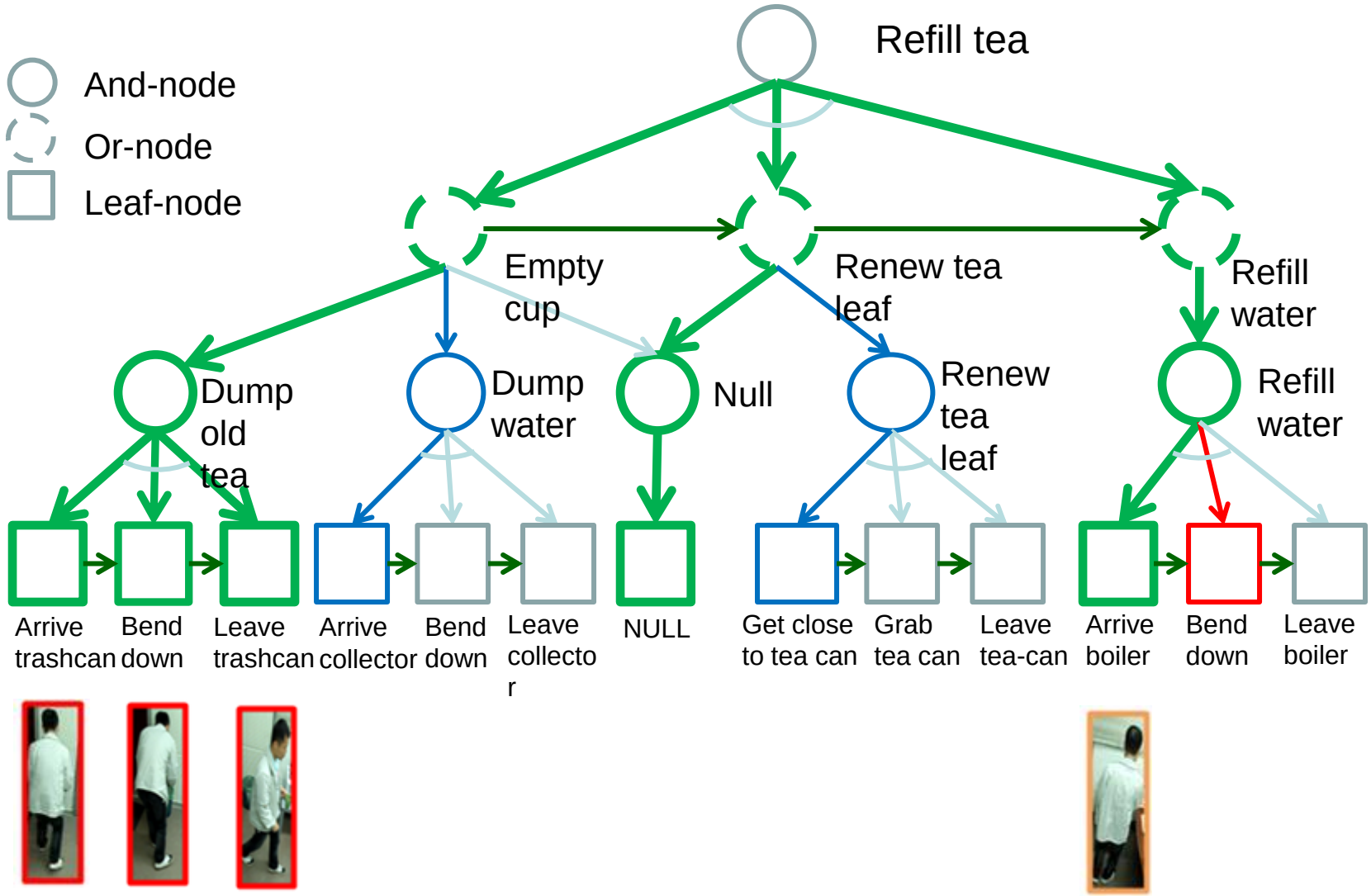
Parsing process (Earley Parser [Earley 1970])



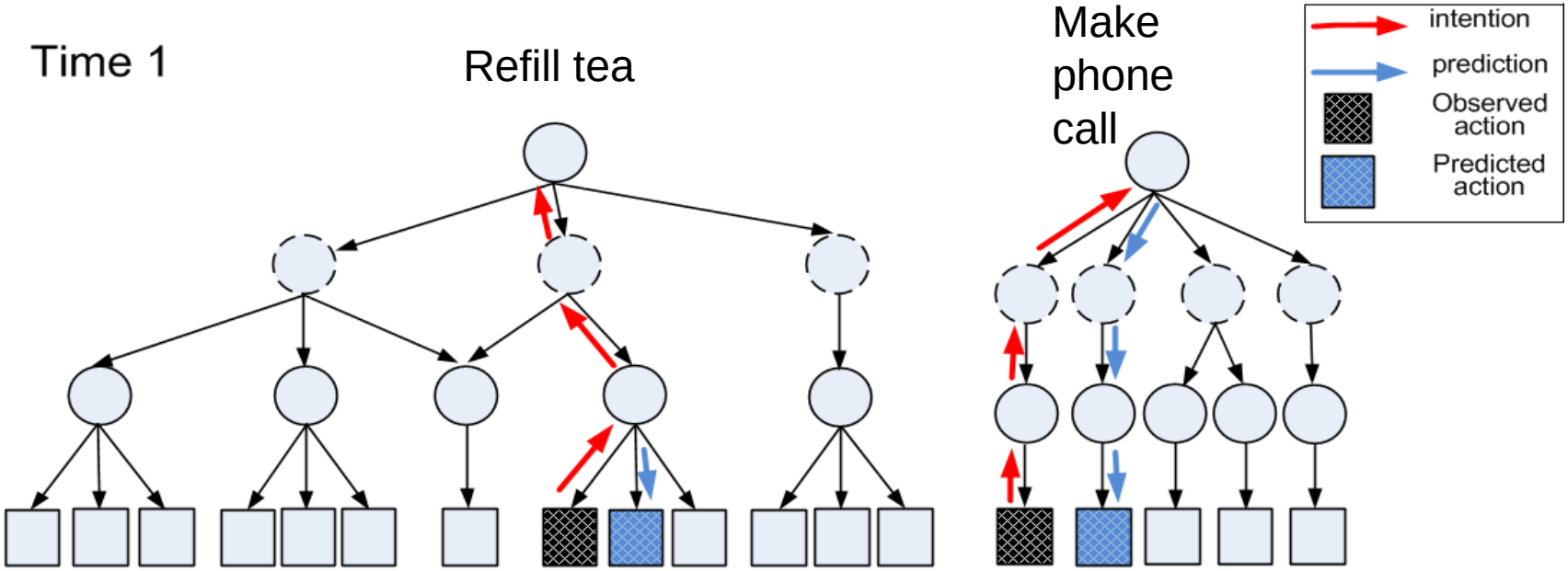
Parsing process (Earley Parser [Earley 1970])



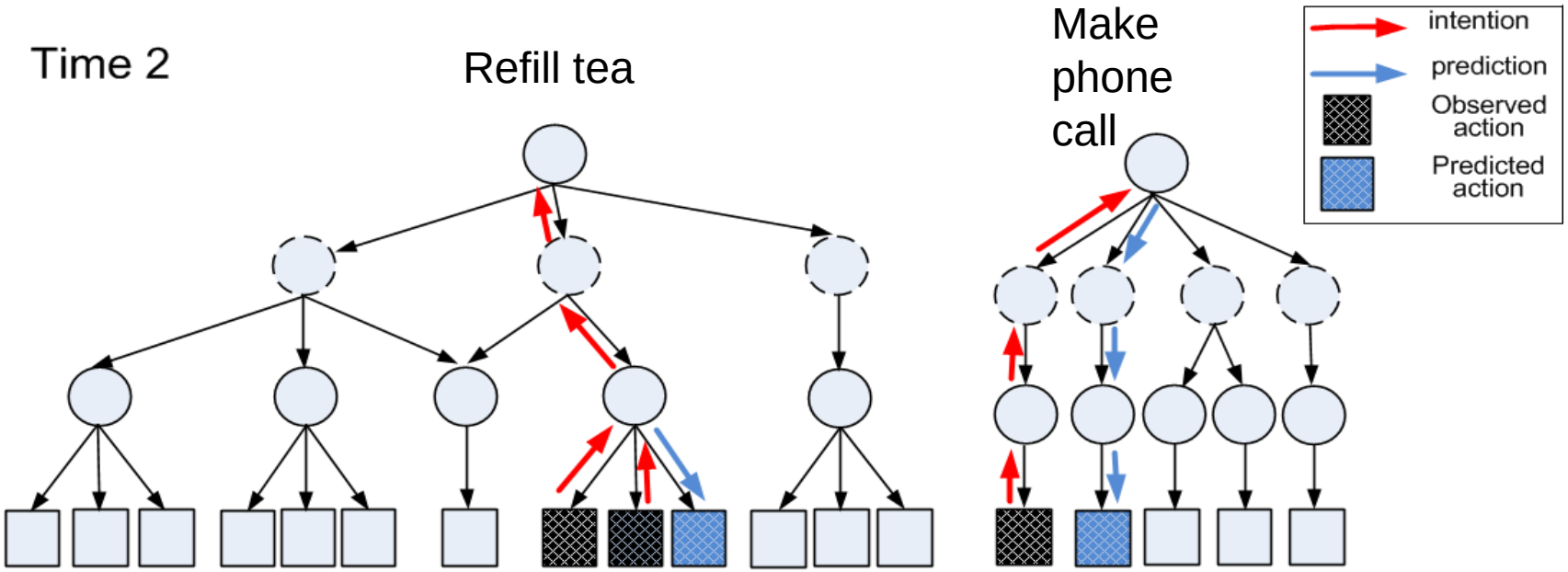
Parsing process (Earley Parser [Earley 1970])

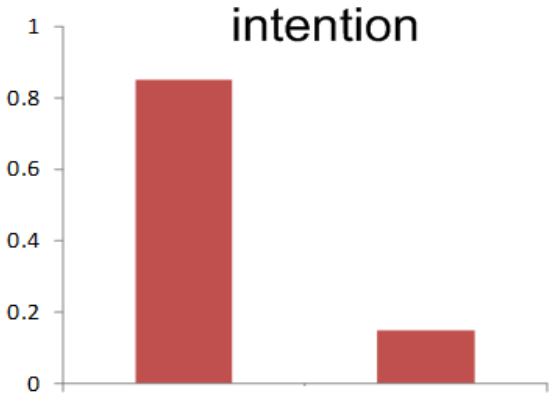


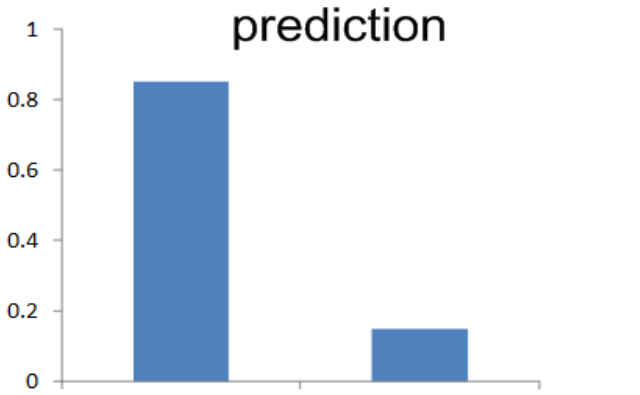
Intention and prediction



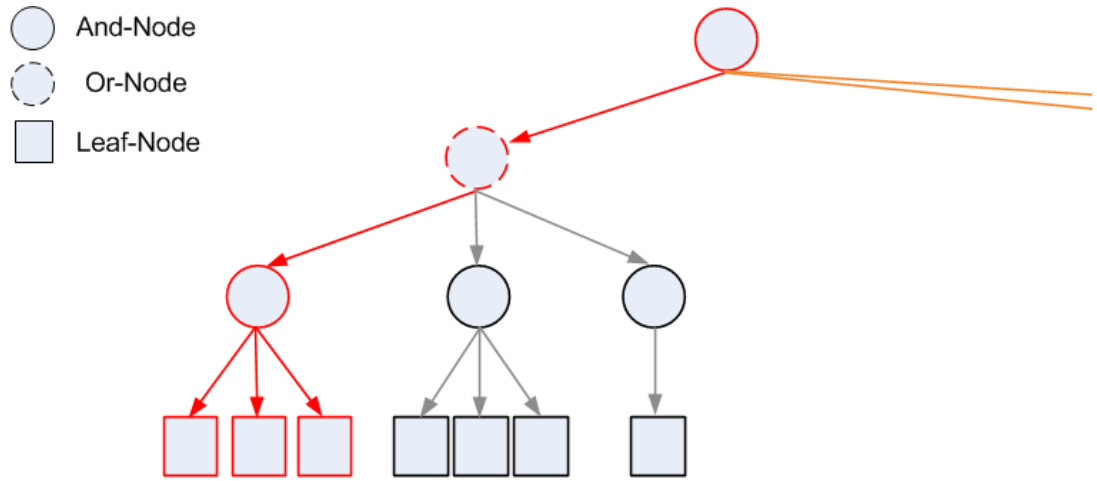
Intention and prediction







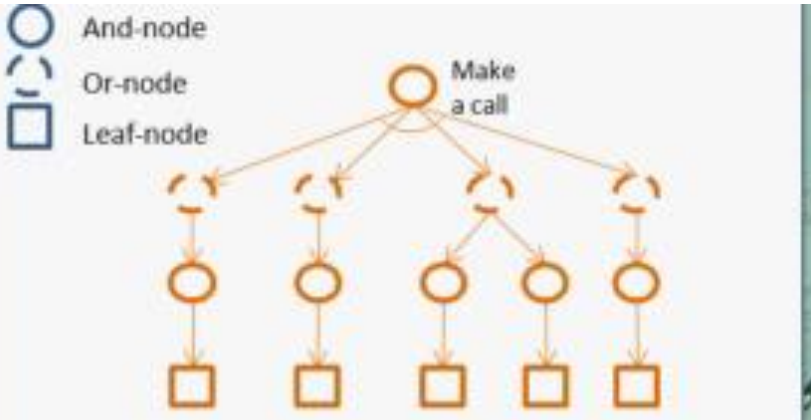
Handle event interruption

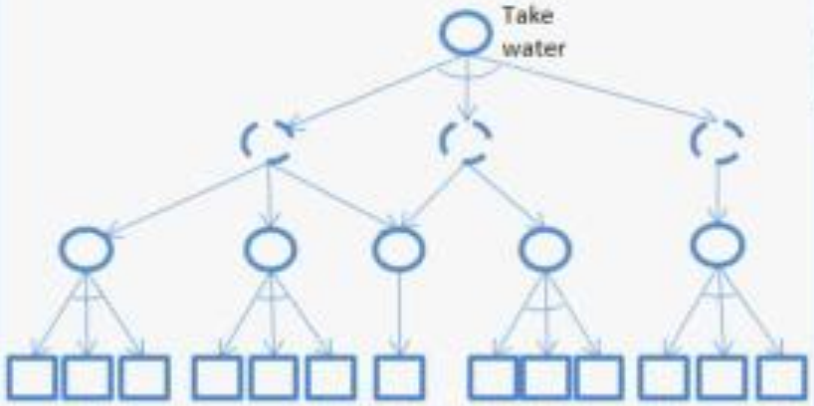


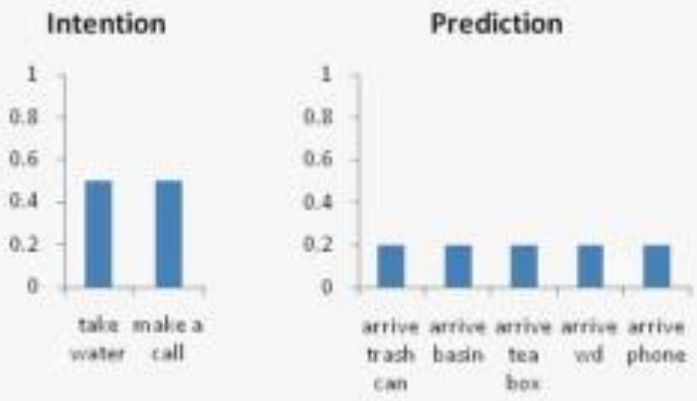
First Partial parse tree of take water

Parse tree of take a phone

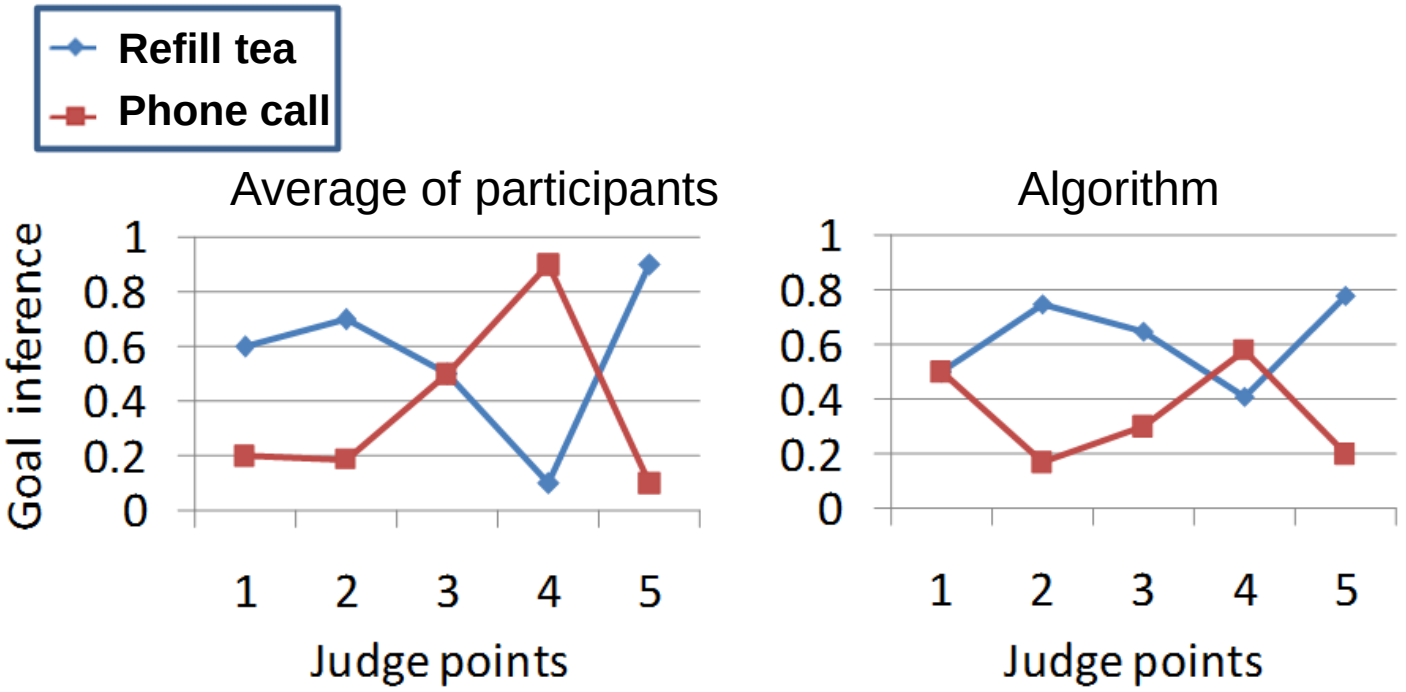
Second Partial parse tree of take water


And-node
Or-node
Leaf-node
Make a call


Take water

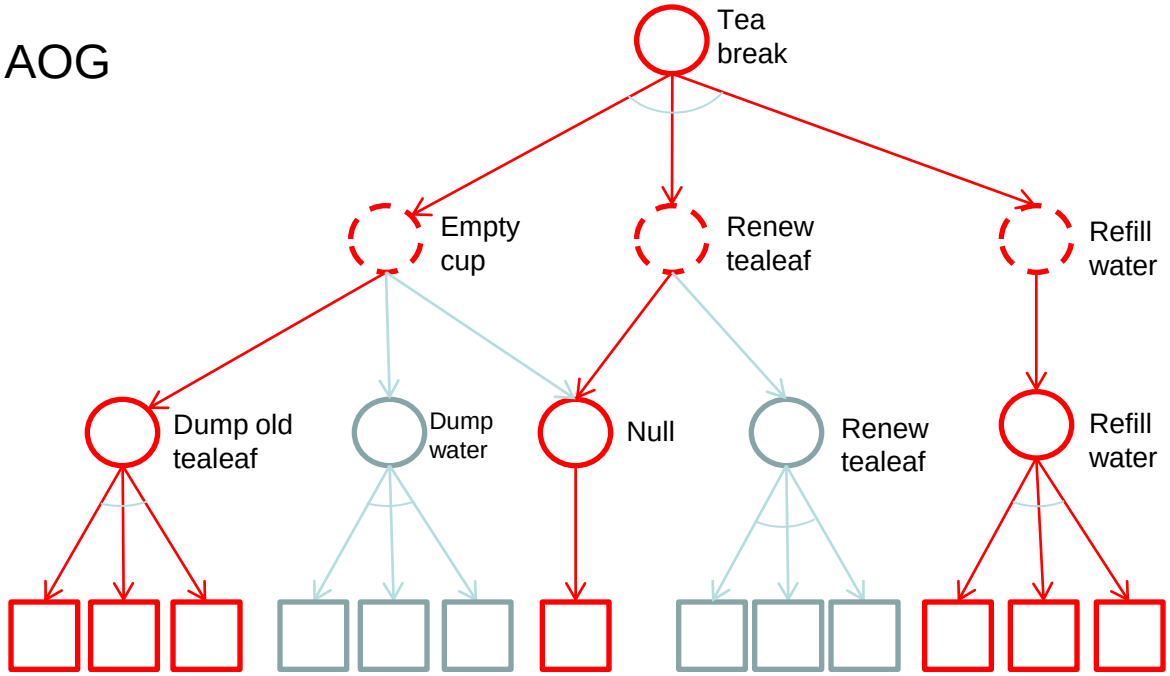

Intention
1
0.8
0.6
0.4
0.2
0
take water
make a call
Prediction
1
0.8
0.6
0.4
0.2
0
arrive trash can
arrive basin
arrive tea box
arrive vd
arrive phone

Comparison with human prediction [Baker, Saxe and Tenenbaum 2009]



Synthesize new events by sampling the AoG

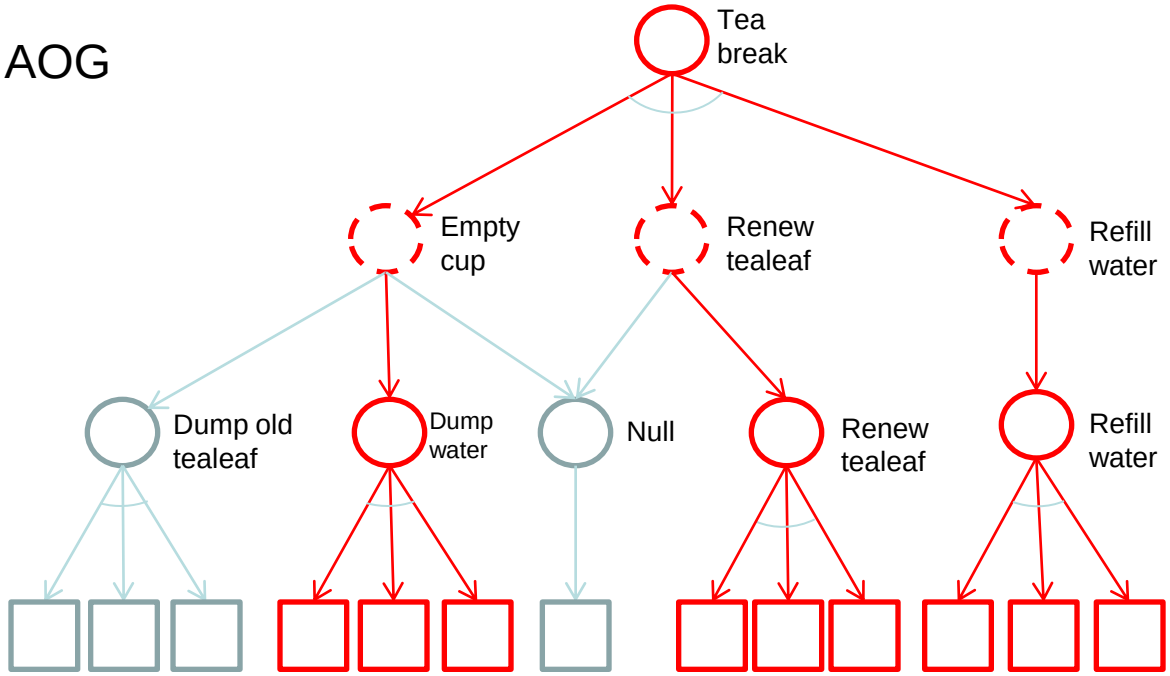
Sample from AOG



Synthesized Event

Event synthesis

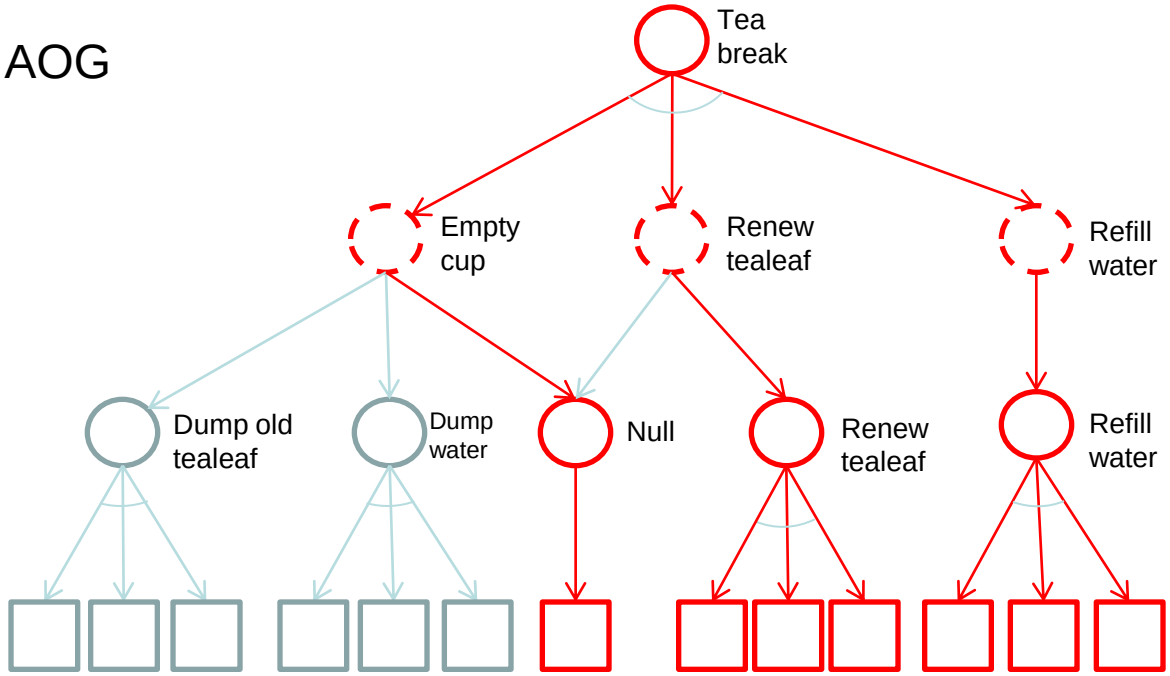
Sample from AOG



Synthesized Event

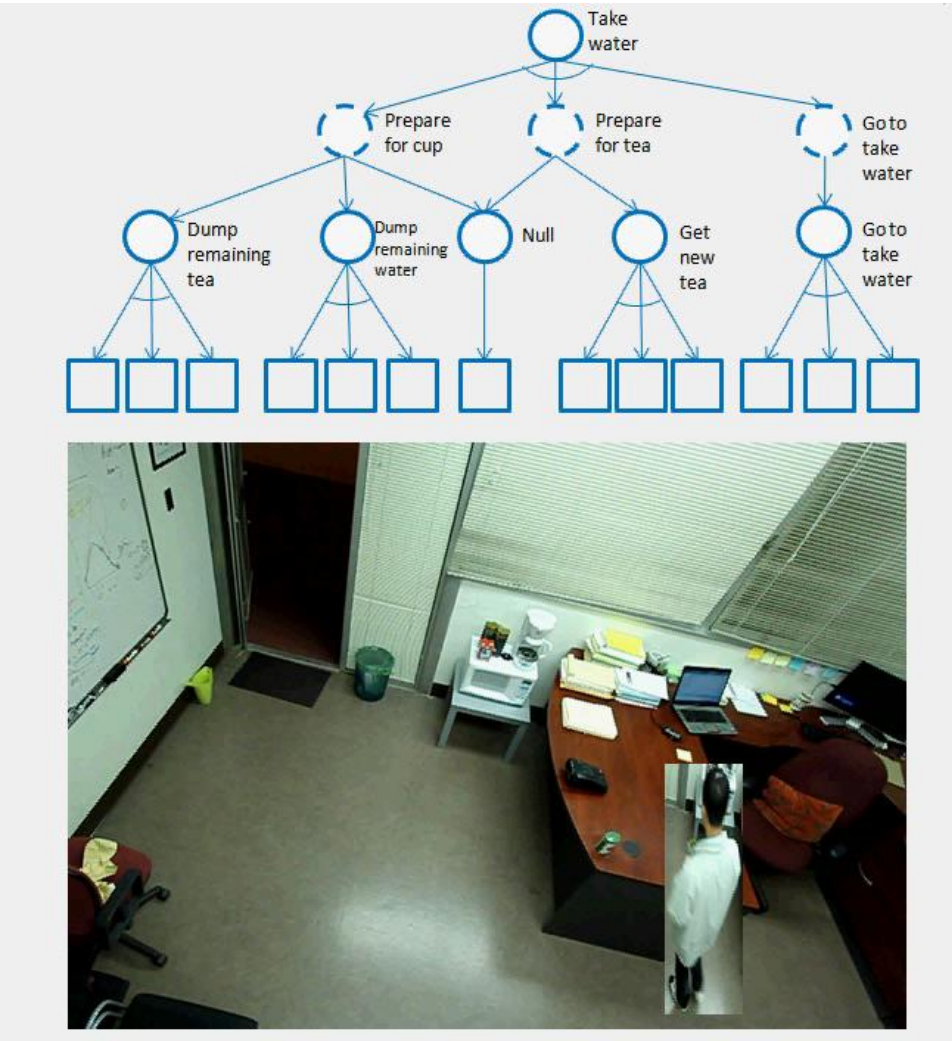
Event synthesis

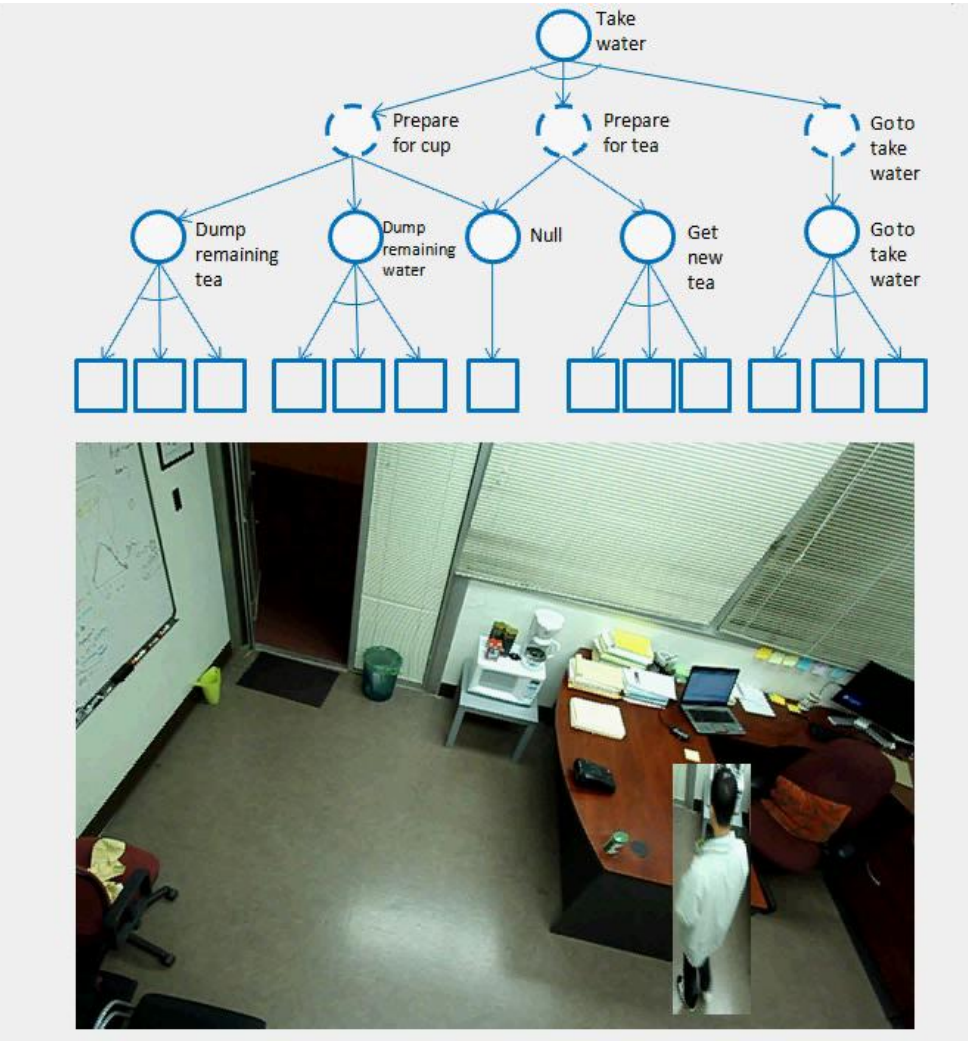
Sample from AOG



Synthesized Event

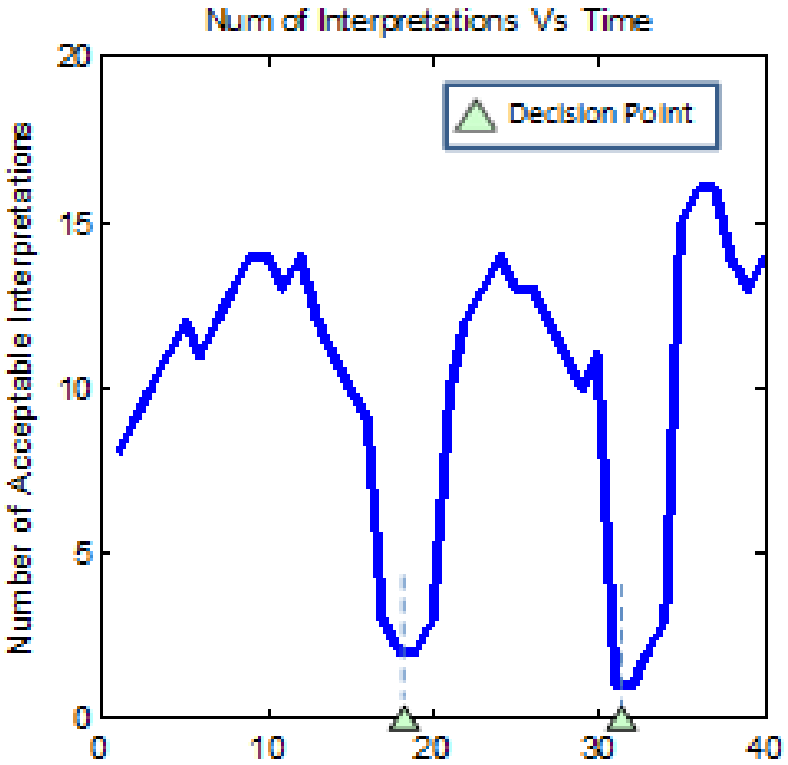
Event synthesize examples*





* We use stored foreground windows and a background frame to synthesize new videos

Computation complexity of parsing



Pickup phone

Reach water boiler

- Initially the number of interpretations above a threshold grows rapidly over time.
- At certain decisive moments, i.e. when informative actions are observed, large number of unlikely interpretation drops below the threshold and hence is pruned.