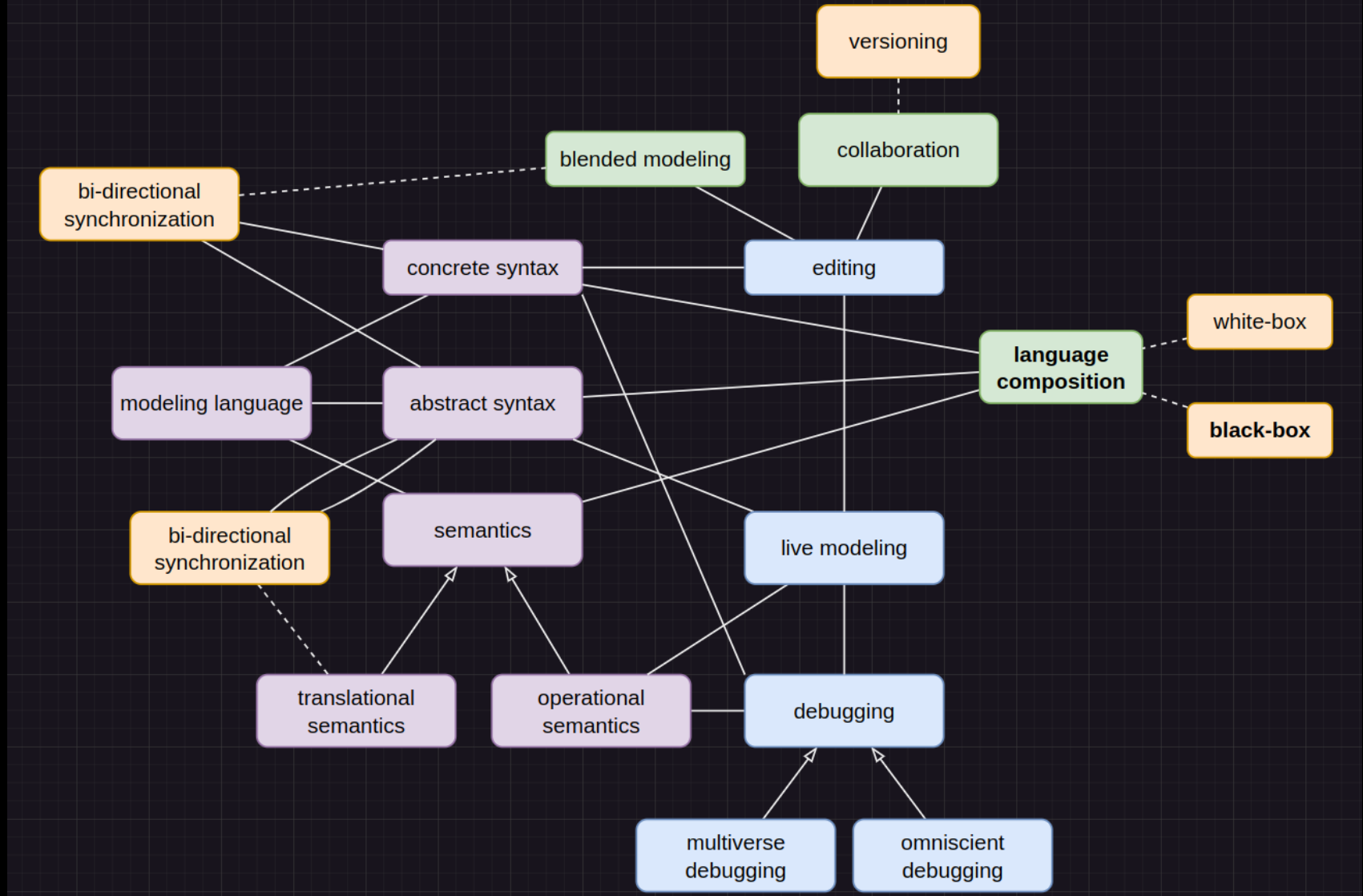

versioning
collaboration
blended modeling
bi-directional synchronization
concrete syntax
editing
white-box
language composition
black-box
modeling language
abstract syntax
bi-directional synchronization
semantics
live modeling
translational semantics
operational semantics
debugging
multiverse debugging
omniscient debugging

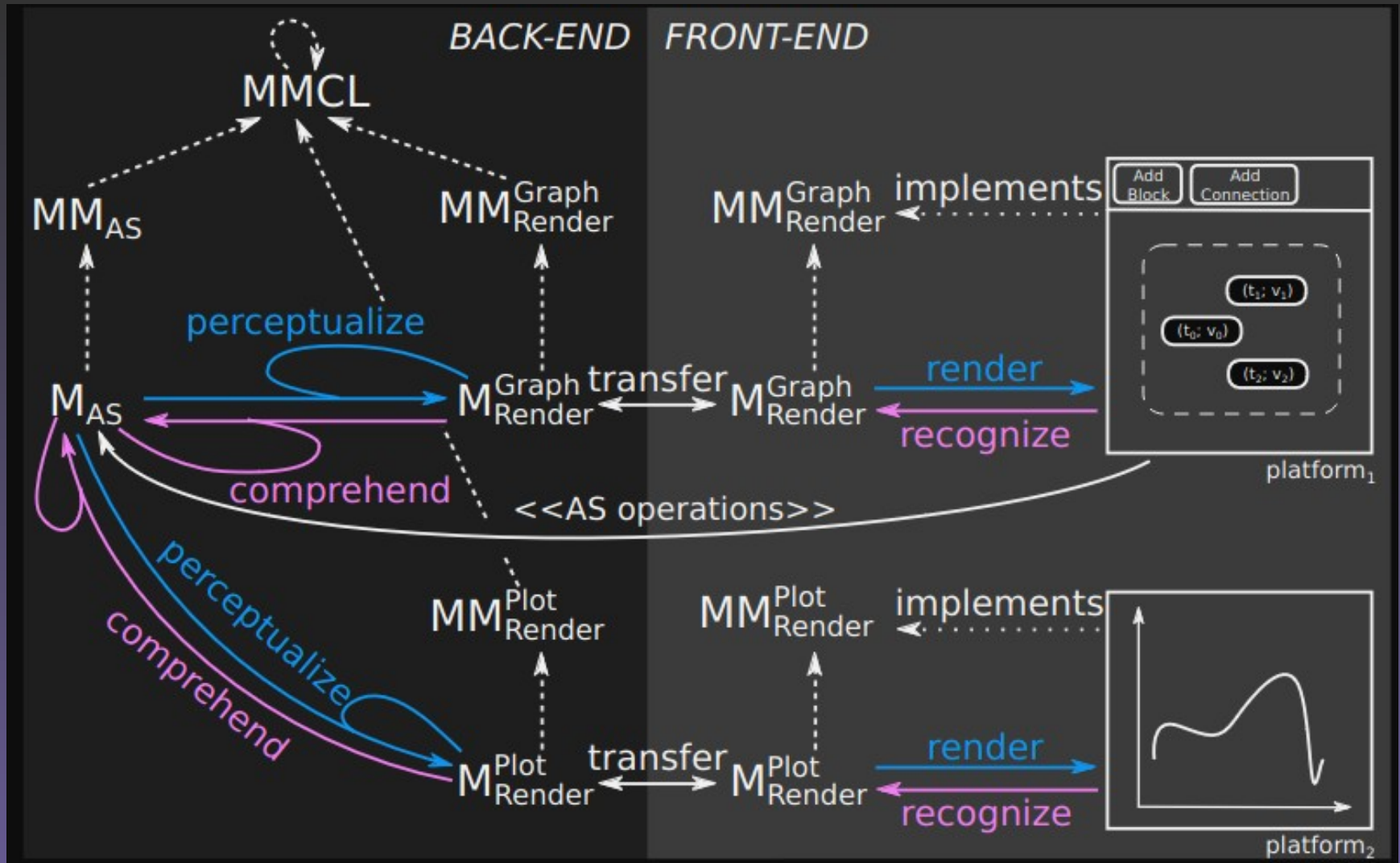
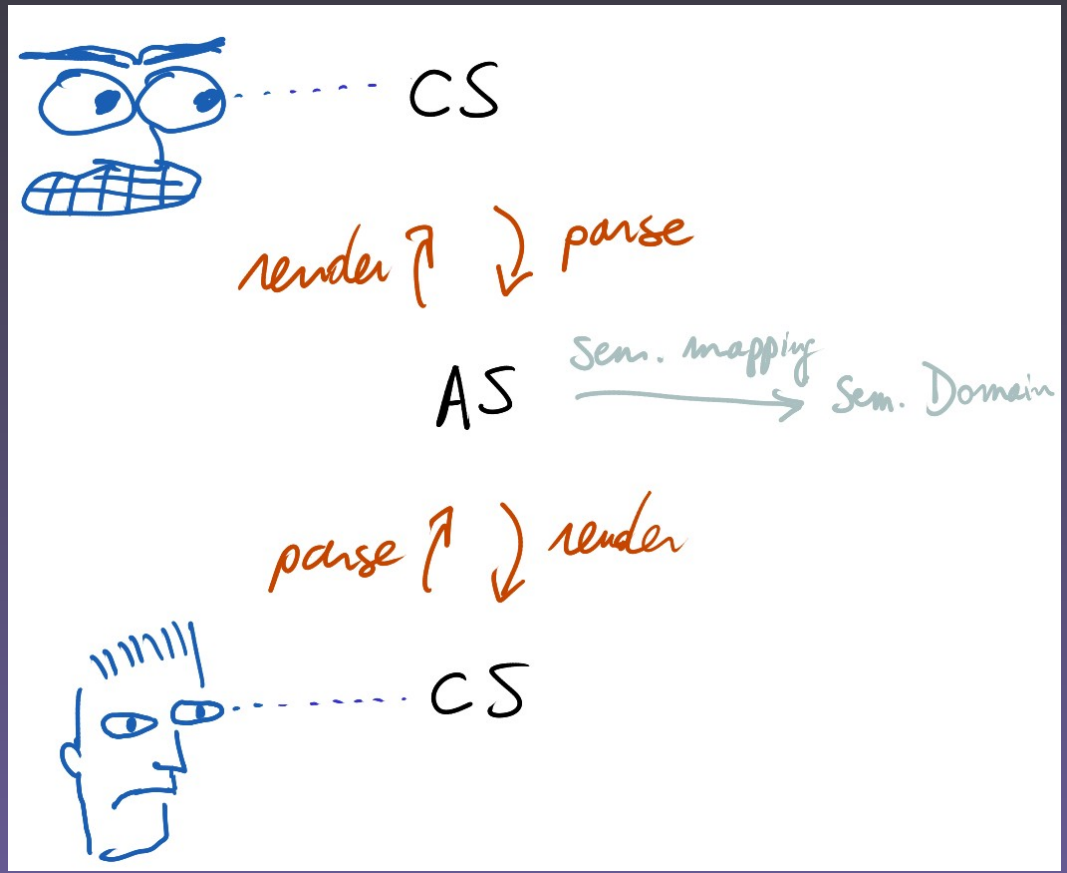
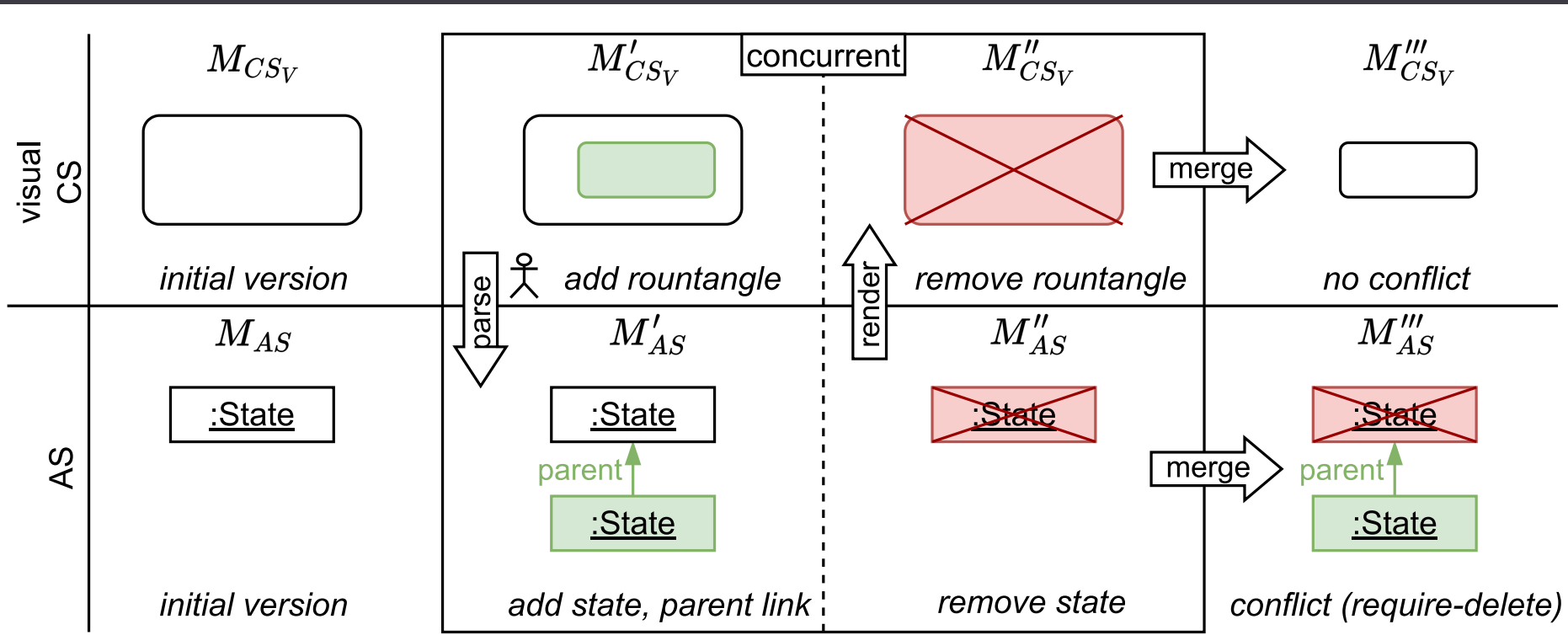


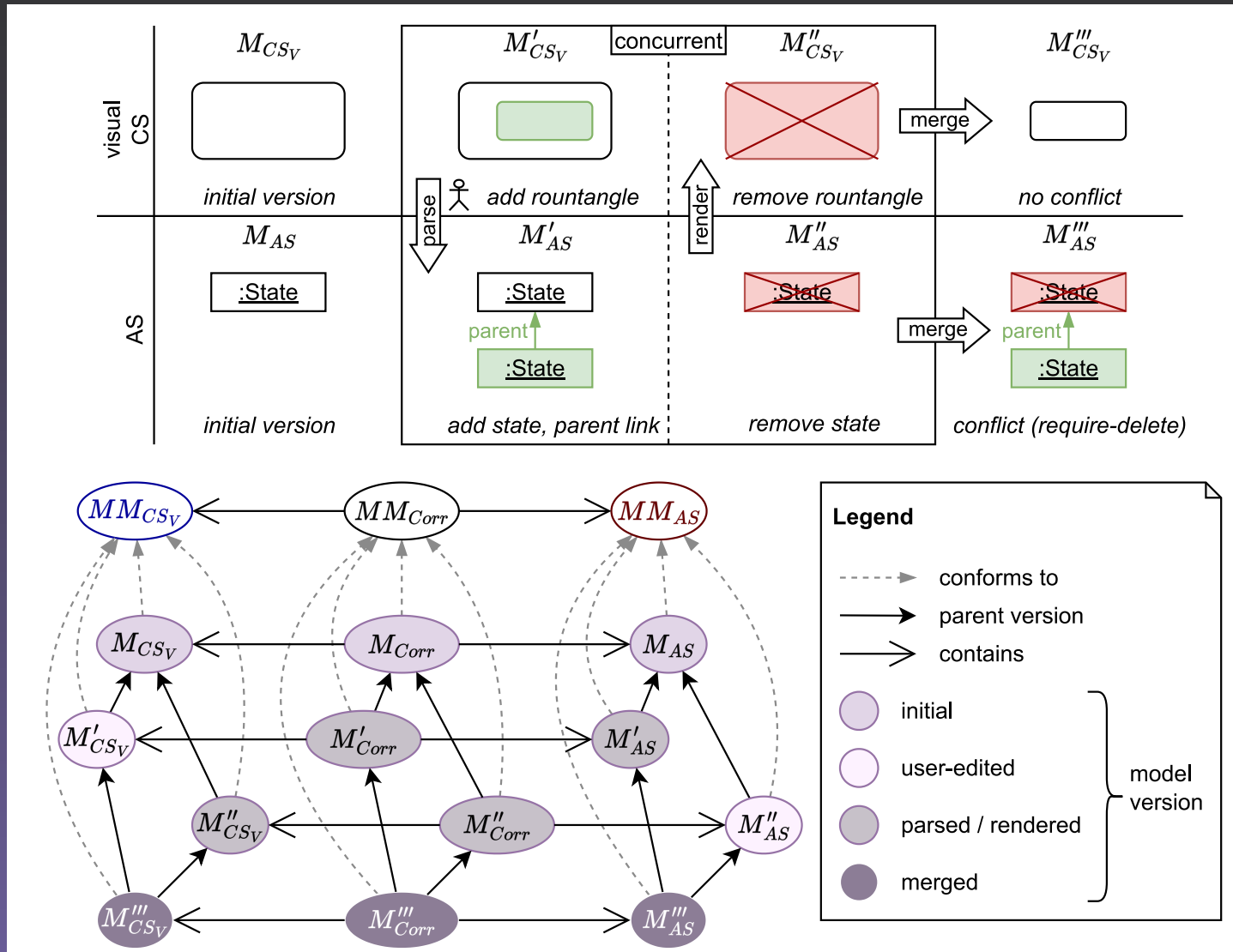
Fig. 3: Modelshare in our concrete syntax framework.

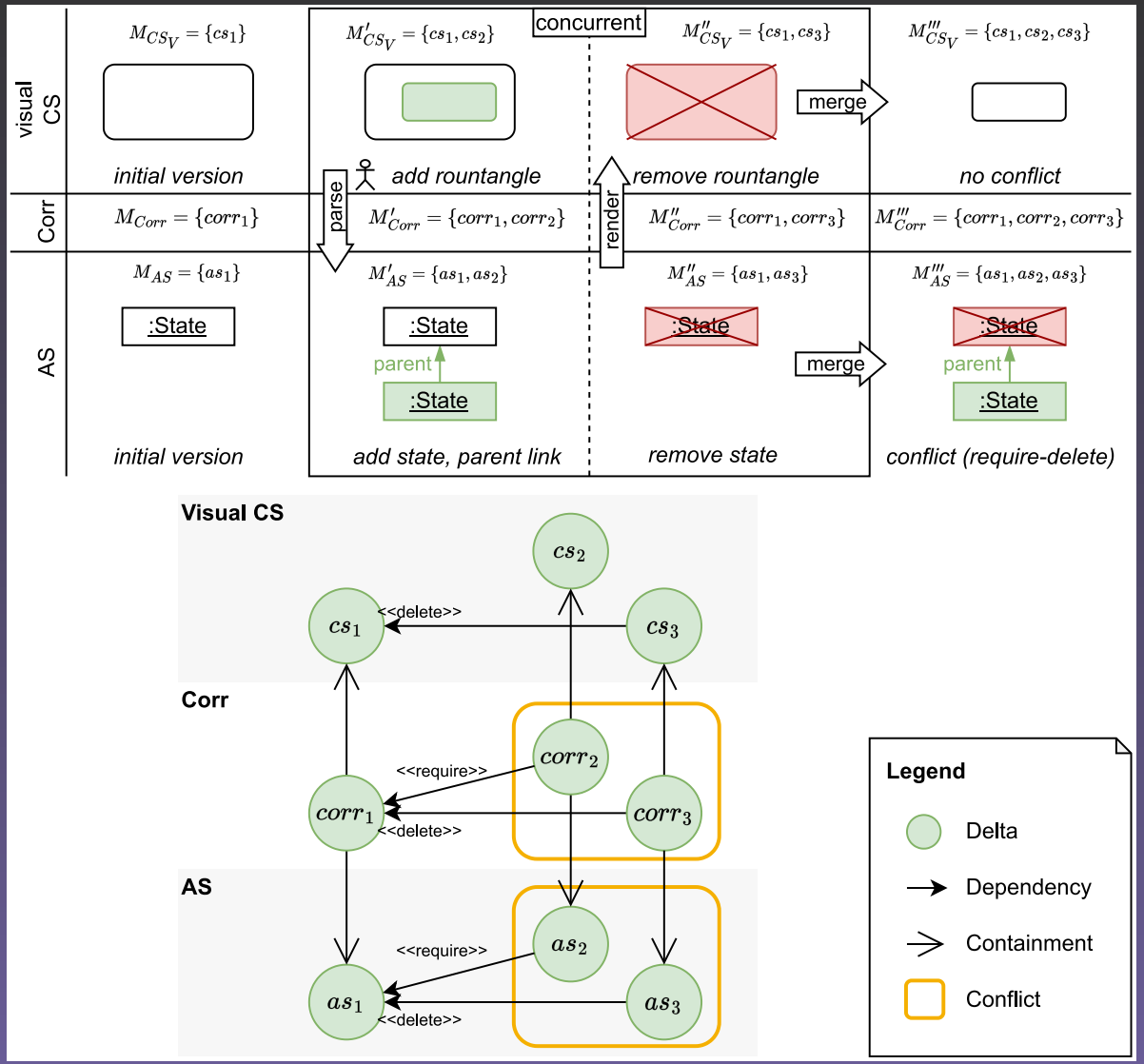
Blended modeling[1]

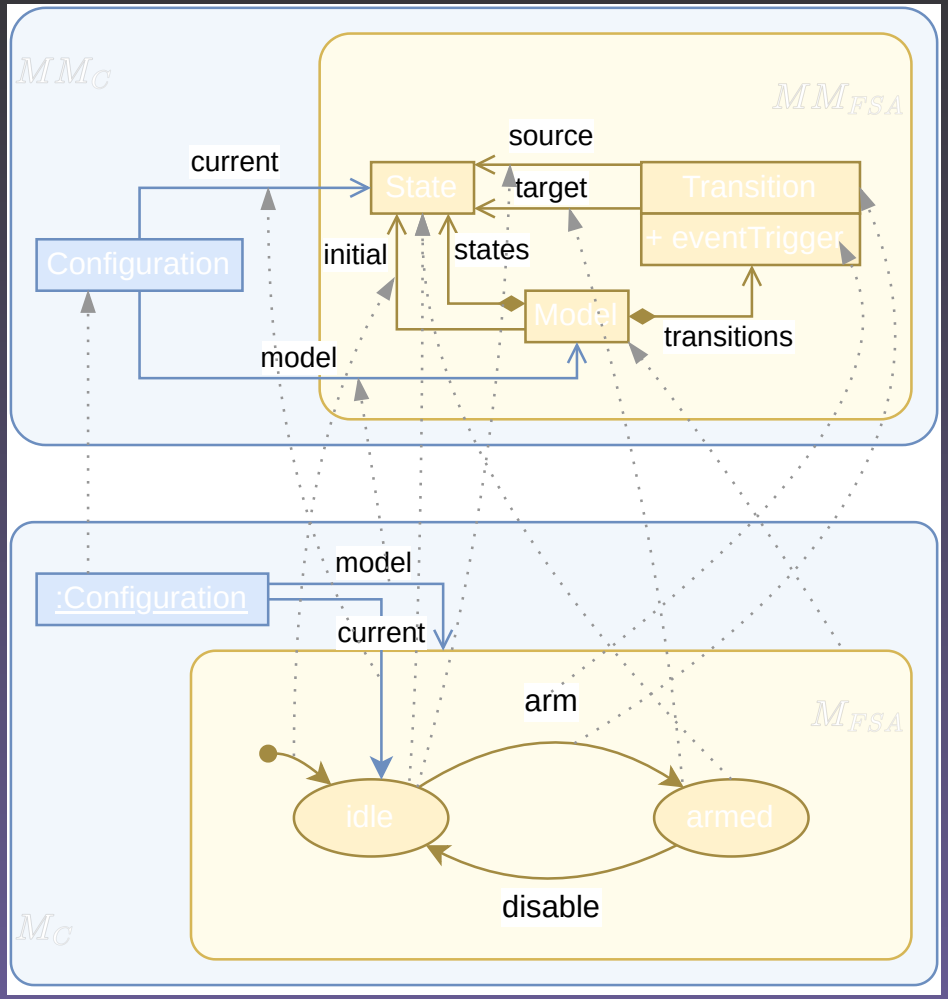


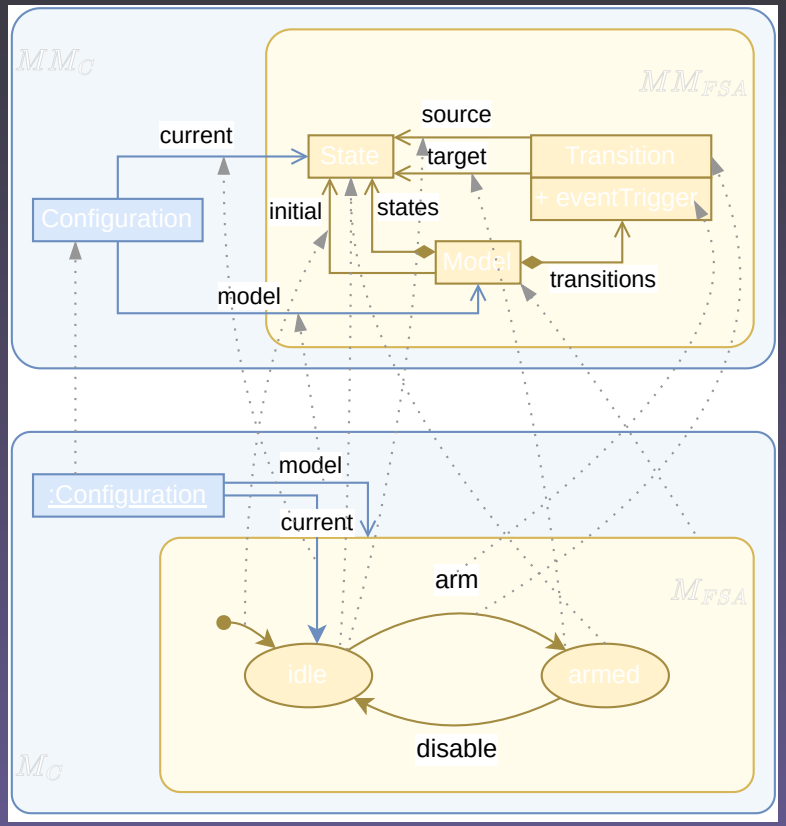
[1] Federico Ciccozzi, Matthias Tichy, Hans Vangheluwe, Danny Weyns: **Blended Modelling - What, Why and How**. MoDELS (Companion) 2019: 425-430

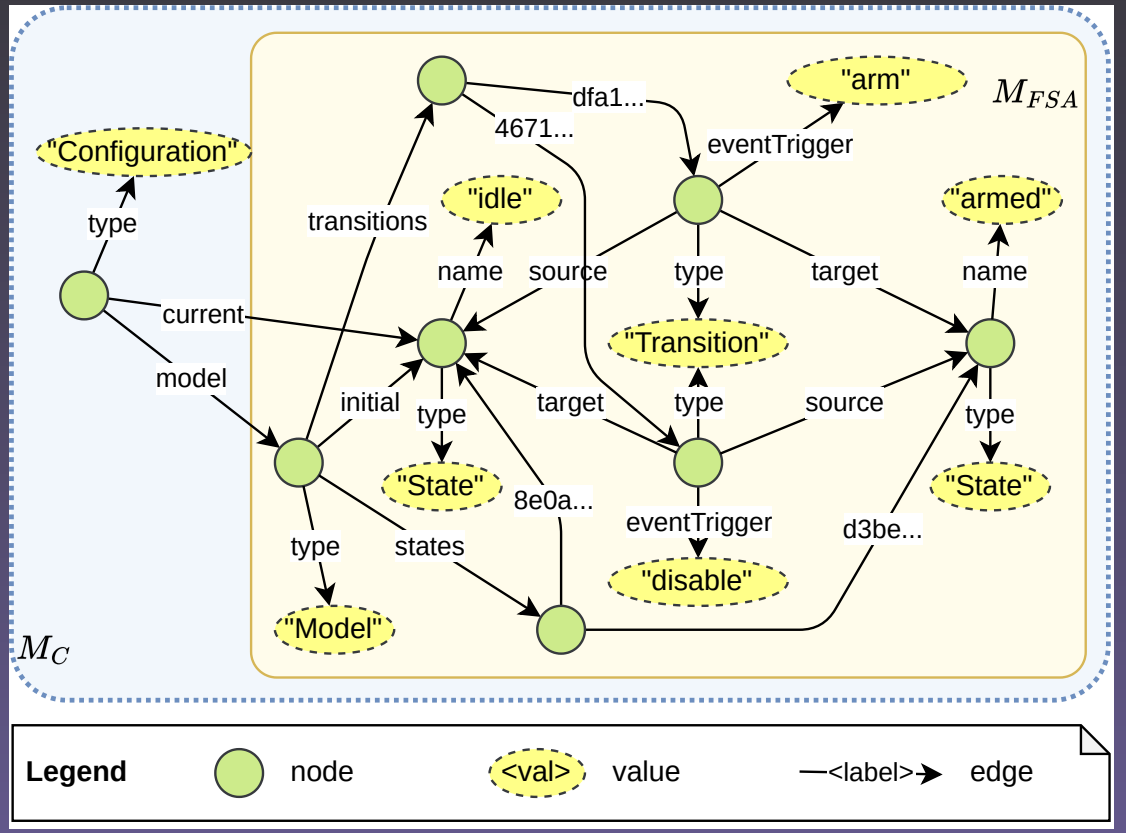

visual CS
AS
M_{CS_V}
M'_{CS_V}
concurrent
M''_{CS_V}
M'''_{CS_V}
merge
initial version
parse
add rountangle
render
remove rountangle
no conflict
M_{AS}
M'_{AS}
M''_{AS}
M'''_{AS}
:State
:State
:State
:State
parent
merge
parent
:State
:State
initial version
add state, parent link
remove state
conflict (require-delete)


M_{CS_V}
M'_{CS_V}
concurrent
M''_{CS_V}
M'''_{CS_V}
visual CS
merge
initial version
parse
add rountangle
render
remove rountangle
no conflict
M_{AS}
M'_{AS}
M''_{AS}
M'''_{AS}
:State
:State
:State
:State
AS
parent
merge
parent
:State
:State
initial version
add state, parent link
remove state
conflict (require-delete)
MM_{CS_V}
MM_{Corr}
MM_{AS}
M_{CS_V}
M_{Corr}
M_{AS}
M'_{CS_V}
M'_{Corr}
M'_{AS}
M''_{CS_V}
M''_{Corr}
M''_{AS}
M'''_{CS_V}
M'''_{Corr}
M'''_{AS}
Legend
conforms to
parent version
contains
initial
user-edited
parsed / rendered
merged
model version


visual CS
Corr
AS
concurrent
$M_{CS_V} = \{cs_1\}$
$M'_{CS_V} = \{cs_1, cs_2\}$
$M''_{CS_V} = \{cs_1, cs_3\}$
$M'''_{CS_V} = \{cs_1, cs_2, cs_3\}$
merge
initial version
add rountangle
remove rountangle
no conflict
parse
render
$M_{Corr} = \{corr_1\}$
$M'_{Corr} = \{corr_1, corr_2\}$
$M''_{Corr} = \{corr_1, corr_3\}$
$M'''_{Corr} = \{corr_1, corr_2, corr_3\}$
$M_{AS} = \{as_1\}$
$M'_{AS} = \{as_1, as_2\}$
$M''_{AS} = \{as_1, as_3\}$
$M'''_{AS} = \{as_1, as_2, as_3\}$
:State
parent
merge
initial version
add state, parent link
remove state
conflict (require-delete)
Visual CS
Corr
AS
cs_1
cs_2
cs_3
<<delete>>
$corr_1$
$corr_2$
$corr_3$
<<require>>
<<delete>>
as_1
as_2
as_3
<<require>>
<<delete>>
Legend
Delta
Dependency
Containment
Conflict


MM_C
MM_{FSA}
source
current
State
target
Transition
+ eventTrigger
Configuration
initial
states
Model
transitions
model
model
:Configuration
current
arm
M_{FSA}
idle
armed
disable
M_C


MM_C
MM_{FSA}
current
source
State
target
Transition
+ eventTrigger
Configuration
initial
states
Model
transitions
model
:Configuration
model
current
arm
M_{FSA}
idle
armed
disable
M_C

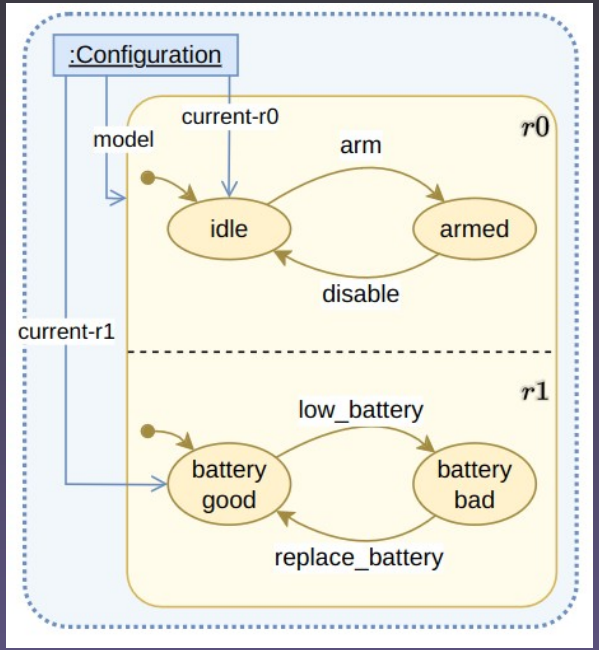

dfa1...
"arm"
M_{FSA}
4671...
eventTrigger
"Configuration"
"idle"
"armed"
type
transitions
name
source
type
target
name
current
"Transition"
model
initial
type
target
type
source
type
"State"
8e0a...
"State"
eventTrigger
type
states
d3be...
"disable"
"Model"
M_C
Legend
node
<val>
value
—<label>→
edge

Additional benefits

- “Who set this variable”?

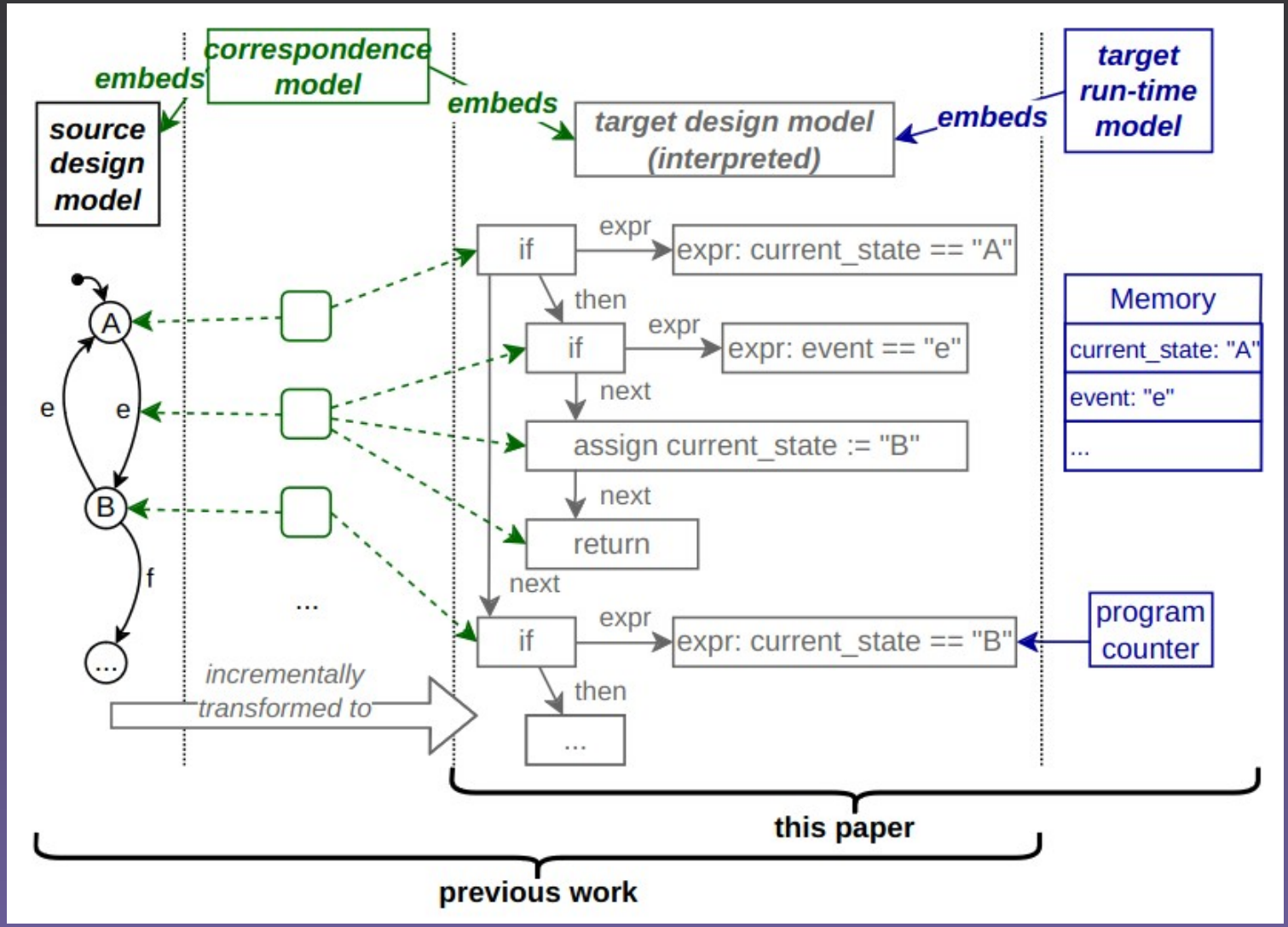
- Parallel independence

- Hibernating executions

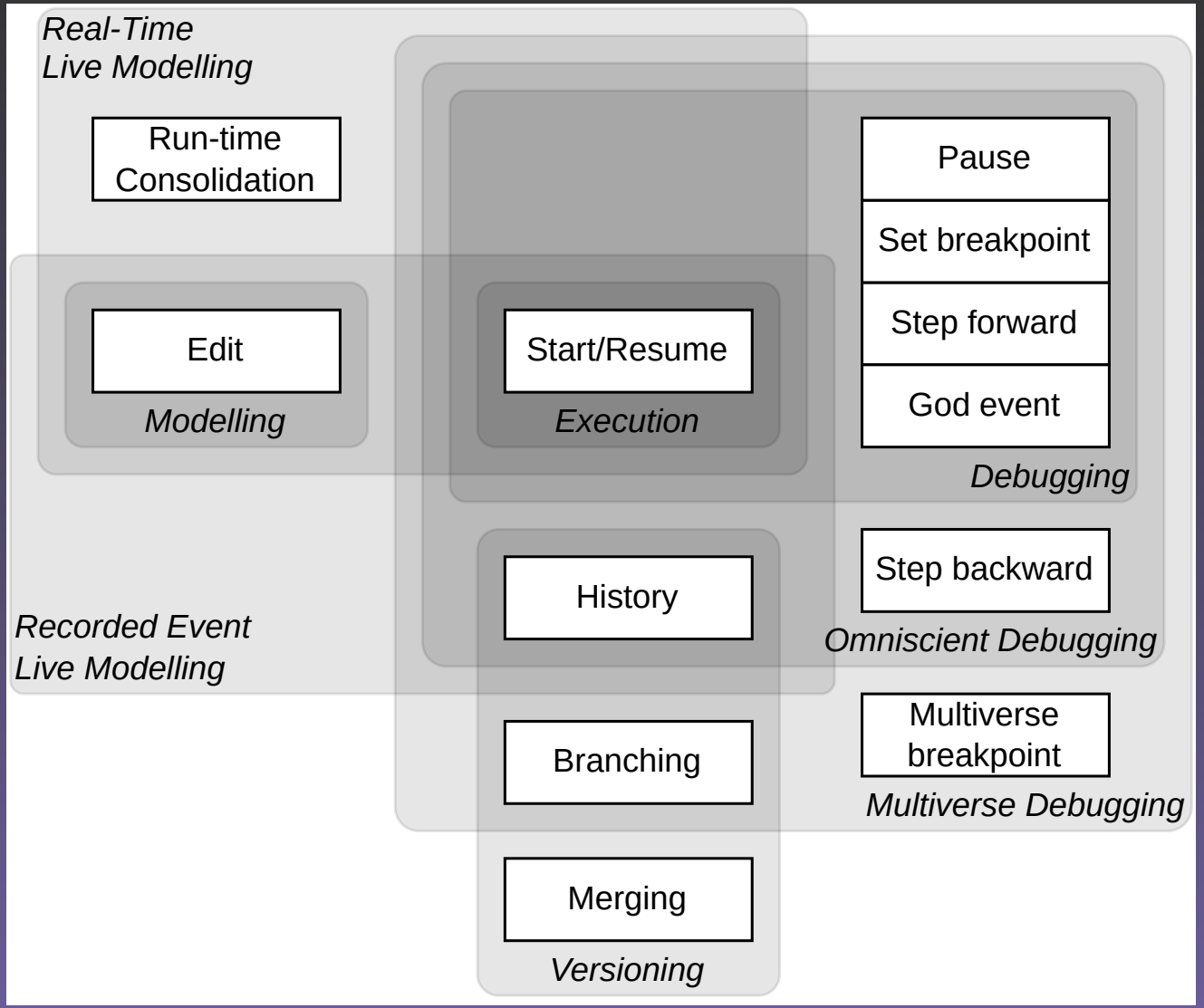


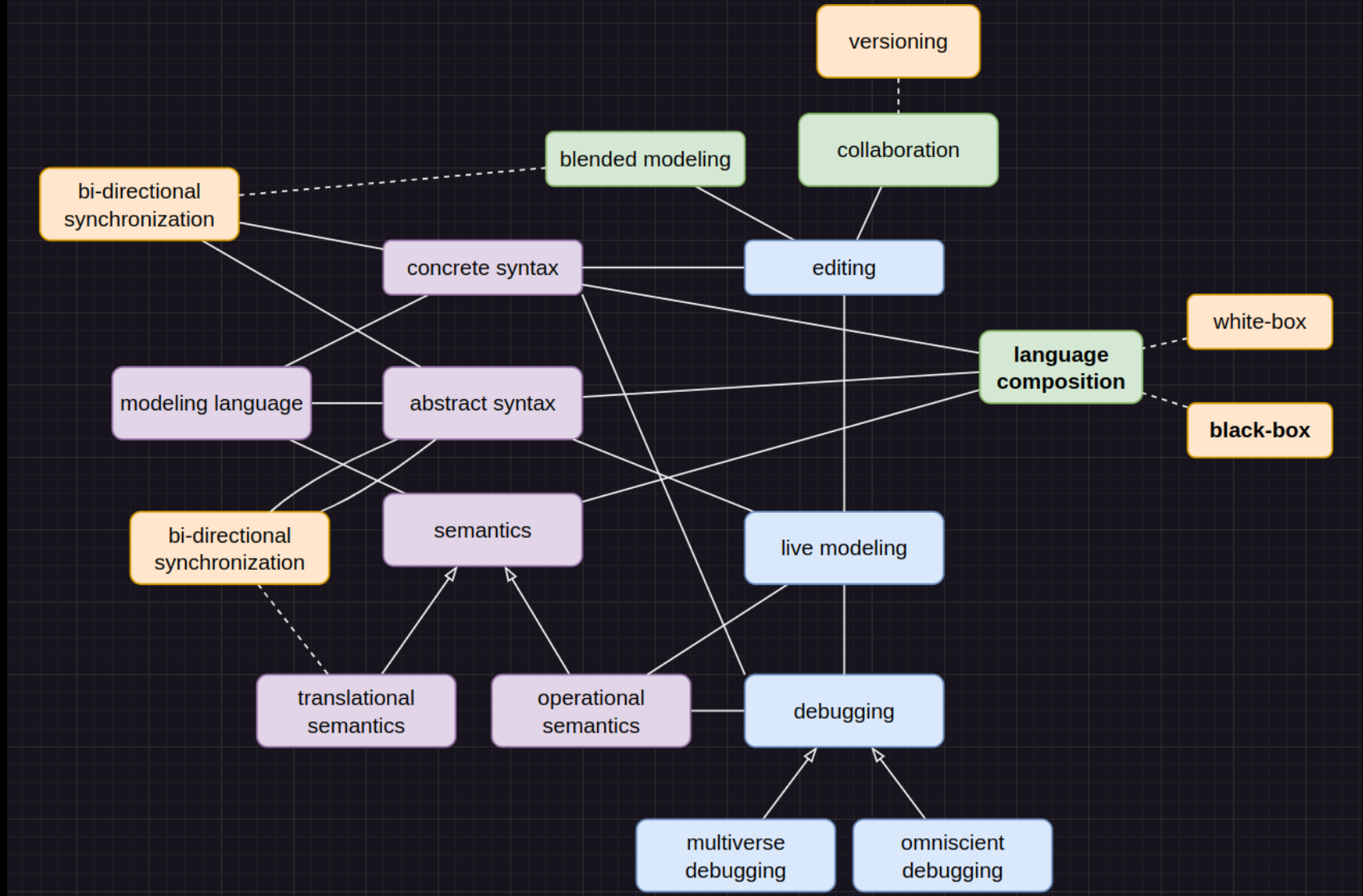
Scalability issues to be addressed

- Collapsing long-running simulation traces

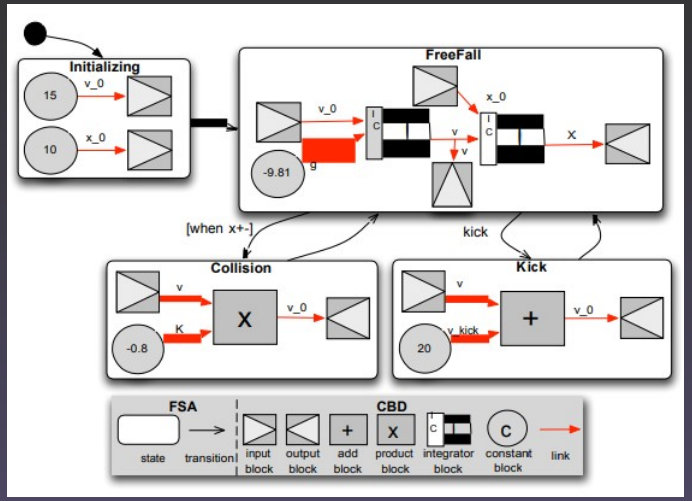

correspondence model
embeds
source design model
embeds
target design model (interpreted)
embeds
target run-time model
A
B
e
e
f
...
incrementally transformed to
if
expr
expr: current_state == "A"
then
if
expr
expr: event == "e"
next
assign current_state := "B"
next
return
next
if
expr
expr: current_state == "B"
then
...
Memory
current_state: "A"
event: "e"
...
program counter
this paper
previous work

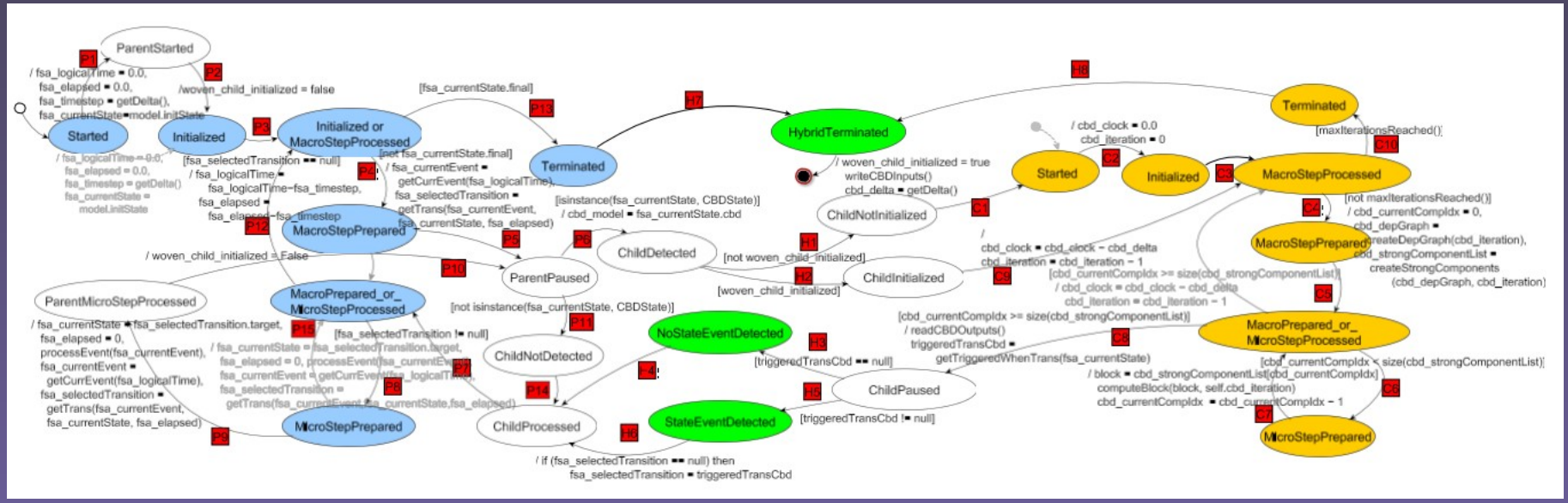
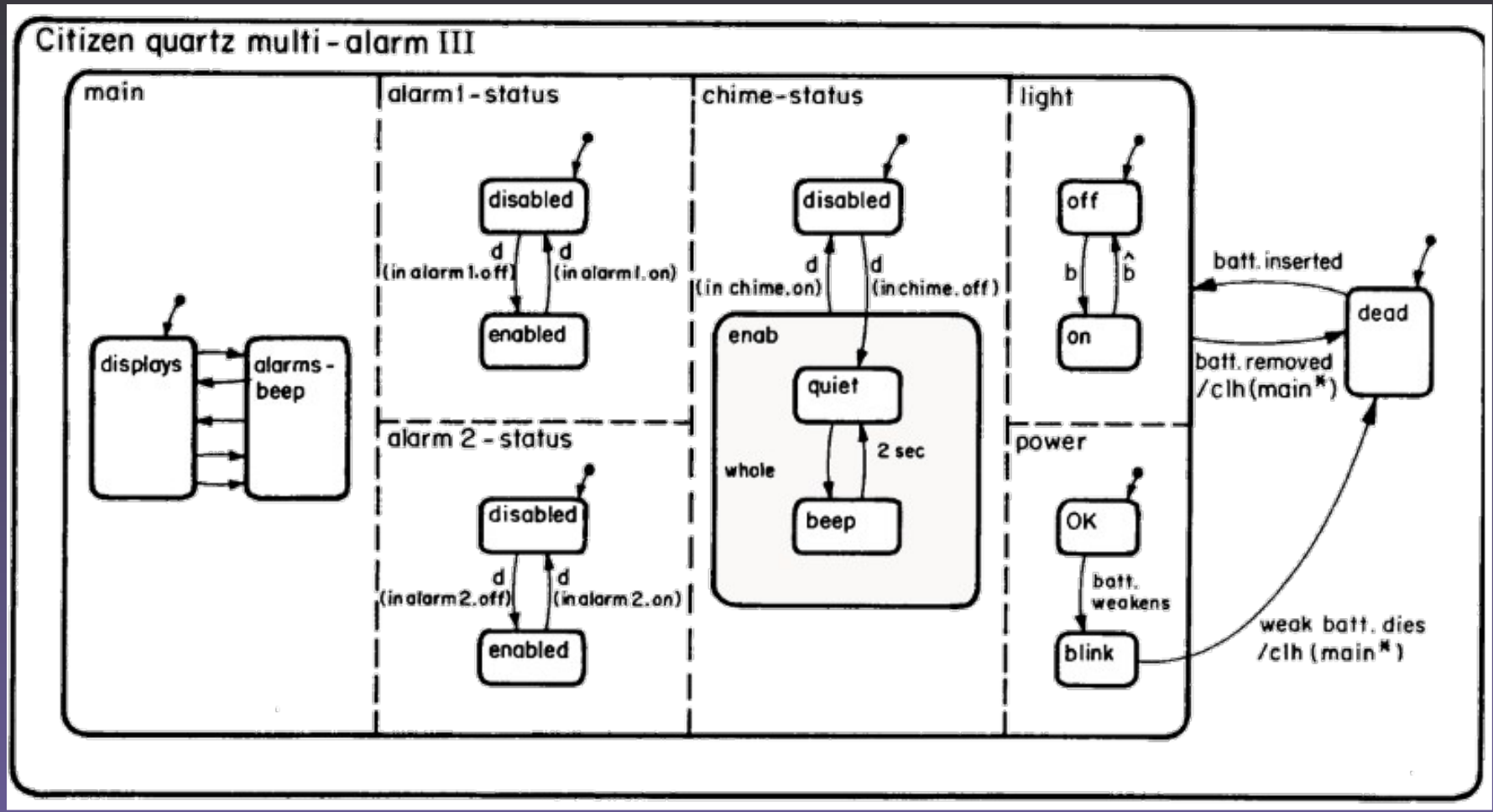
Collaboration, Debugging and Live Modeling




versioning
collaboration
blended modeling
bi-directional synchronization
concrete syntax
editing
white-box
language composition
black-box
modeling language
abstract syntax
bi-directional synchronization
semantics
live modeling
translational semantics
operational semantics
debugging
multiverse debugging
omniscient debugging

CBD within FSA





Statecharts