Software Design, Modelling and Analysis in UML

Lecture 17: Live Sequence Charts II

2012-01-31

Prof. Dr. Andreas Podelski, **Dr. Bernd Westphal**

Albert-Ludwigs-Universität Freiburg, Germany

Contents & Goals

Last Lecture:

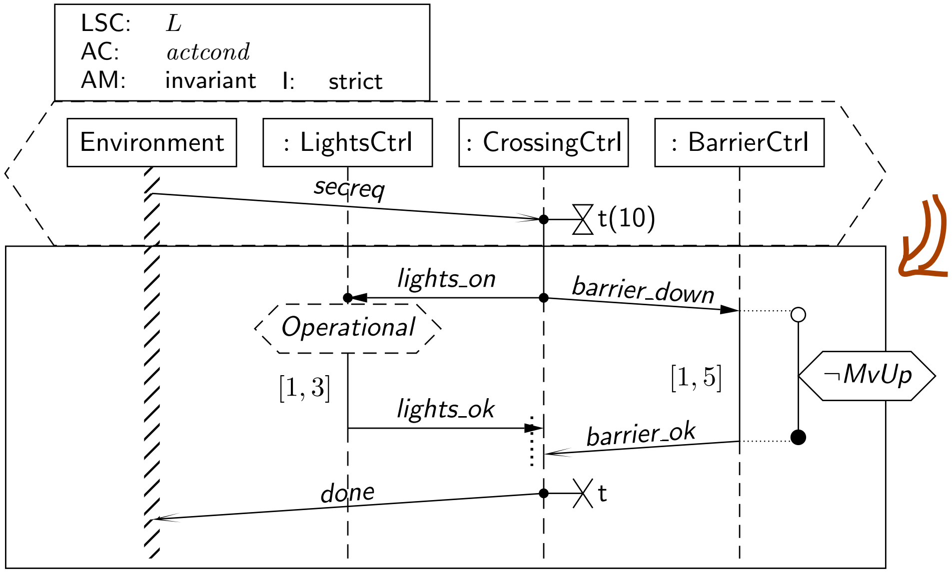
- Reflective vs. constructive description of behaviour
- Live Sequence Charts: syntax, intuition

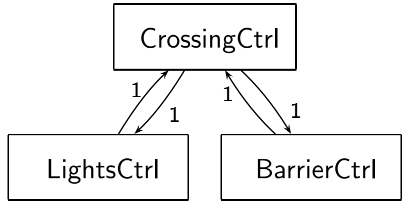
This Lecture:

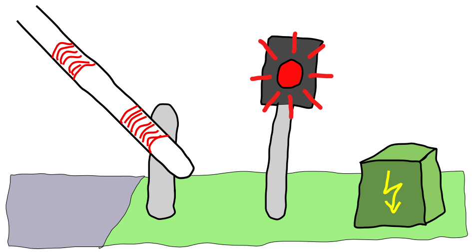
- **Educational Objectives:** Capabilities for following tasks/questions.
  - What does this LSC mean?
  - Are this UML model's state machines consistent with the interactions?
  - Please provide a UML model which is consistent with this LSC.
  - What is: activation, hot/cold condition, pre-chart, etc.?
- **Content:**
  - Symbolic Büchi Automata (TBA) and its (accepted) language.
  - LSC formal semantics.

Recall: Live Sequence Charts Syntax

Recall: Example







- **Whenever** the CrossingCtrl has consumed a 'secreq' event
- **then** it shall finally send 'lights_on' and 'barrier_down' to LightsCtrl and BarrierCtrl,
- if LightsCtrl **is not** 'operational' when receiving that event,
  the rest of this scenario doesn't apply; maybe there's another LSC for that case.
- if LightsCtrl **is** 'operational' when receiving that event,
  it shall reply with 'lights_ok' within 1–3 time units,
- the BarrierCtrl shall reply with 'barrier_ok' within 1–5 time units, during this time (dispatch time not included) it shall not be in state 'MvUp',
- 'lights_ok' and 'barrier_ok' may occur in any order.
- After having consumed both, CrossingCtrl may reply with 'done' to the environment.

Recall: LSC Body – Abstract Syntax

Let $\Theta = \{\text{hot}, \text{cold}\}$. An **LSC body** is a tuple

$$(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$$

where

- I is a finite set of **instance lines**, each associated with a class $C \in \mathscr{C}$
- $(\mathscr{L}, \preceq)$ is a finite, non-empty, partially ordered set of **locations**, each $l \in \mathscr{L}$ is associated with a temperature $\theta(l) \in \Theta$ and an instance line $i_l \in I$,
- $\sim \subseteq \mathscr{L} \times \mathscr{L}$ is an **equivalence relation** on locations, the **simultaneity** relation,
- $\mathscr{S} = (\mathscr{T}, \mathscr{C}, V, atr, \mathscr{E})$ is a signature, (w)
- $\mathsf{Msg} \subseteq \mathscr{L} \times \mathscr{E} \times \mathscr{L}$ (φ) is a set of **asynchronous messages** with $(l, b, l') \in \mathsf{Msg}$ only if $l \sim l'$,

  Not: **instantaneous messages** — could be linked to method/operation calls.
- $\mathsf{Cond} \subseteq (2^{\mathscr{L}} \setminus \emptyset) \times Expr_{\mathscr{S}} \times \Theta$ is a set of **conditions** with $(L, expr, \theta) \in \mathsf{Cond}$ only if $l \sim l'$ for all $l, l' \in L$,
- $\mathsf{LocInv} \subseteq \mathscr{L} \times \{\circ, \bullet\} \times Expr_{\mathscr{S}} \times \Theta \times \mathscr{L} \times \{\circ, \bullet\}$ is a set of **local invariants**,

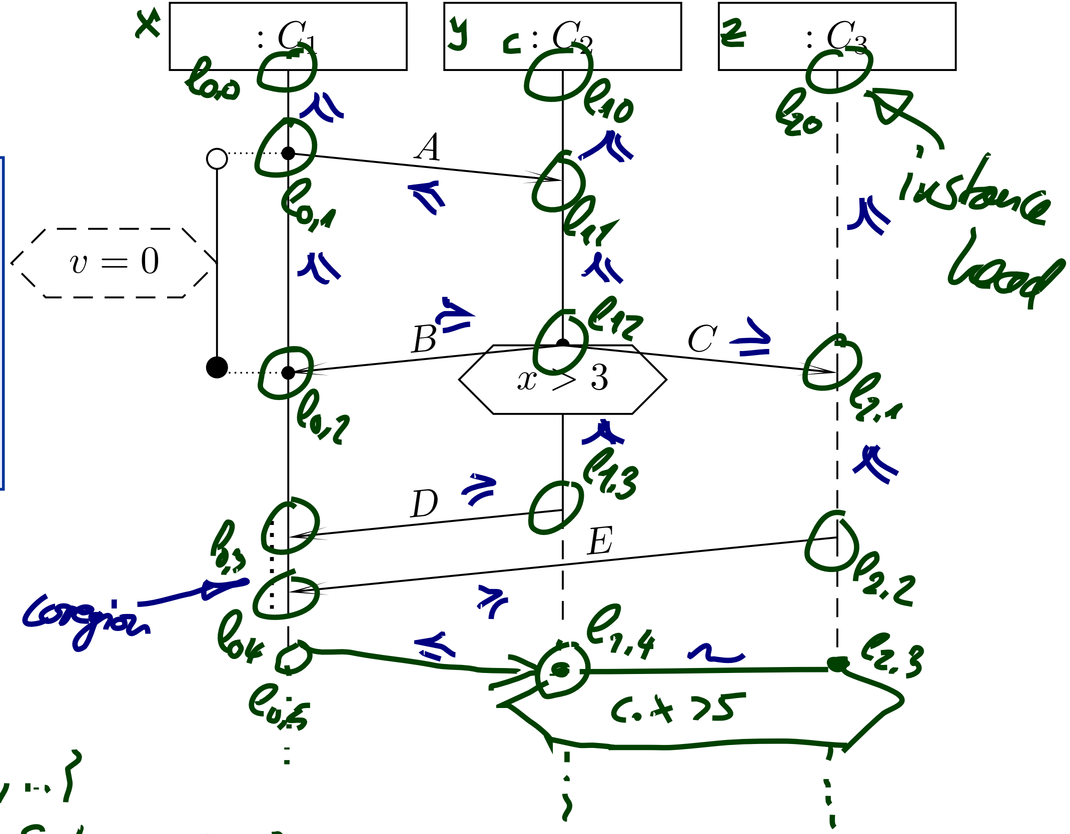
exclusive

inclusive

Example

$$(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$$

$$\mathsf{Msg} \subseteq \mathscr{L} \times \mathscr{E} \times \mathscr{L}$$
$$\mathsf{Cond} \subseteq (2^{\mathscr{L}} \setminus \emptyset) \times \mathit{Expr}_{\mathscr{S}} \times \Theta$$
$$\mathsf{LocInv} \subseteq$$
$$\mathscr{L} \times \{\circ, \bullet\} \times \mathit{Expr}_{\mathscr{S}} \times \Theta \times \mathscr{L} \times \{\circ, \bullet\}$$



$I = \{x, y, z\}$, $\mathscr{C}(x) = C_1, \ldots$

$\mathscr{L} = \{ (\ell_{0,0}, hot), (\ell_{1,0}, cold), \ldots \}$

$\preceq \subseteq \mathscr{L} \times \mathscr{L}: \{ \ell_{0,0} \preceq \ell_{1,0}, \ldots$
$\ell_{0,0} \preceq \ell_{0,1}, \ell_{0,2} \preceq \ell_{0,3}, \ell_{0,2} \preceq \ell_{0,4}, \ldots \}$

$\mathsf{Msg} = \{ (\ell_{0,1}, A, \ell_{1,1}), \ldots \}$

$\sim \, = \{ (\ell_{1,4}, \ell_{2,3}) \}$

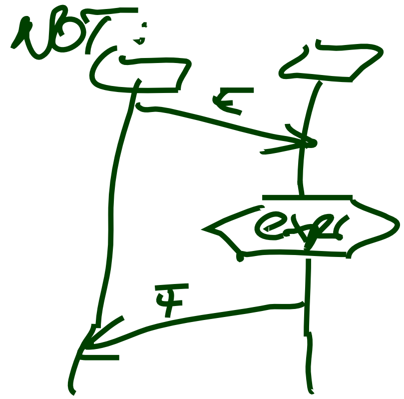
$\mathsf{Cond} = \{ (\{\ell_{1,4}, \ell_{2,3}\}, (x > 5), hot), \ldots \}$

$\mathsf{LocInv} = \{ (\ell_{0,1}, \circ, (v = 0), cold, \ell_{0,2}, \bullet) \}$

Recall: Well-Formedness

Bondedness/no floating conditions: (could be relaxed a little if we wanted to)

- For each location $l \in \mathscr{L}$, **if** l is the location of



  - a **condition**, i.e.

$$\exists\,(L, expr, \theta) \in \mathsf{Cond} : l \in L,$$

  - a **local invariant**, i.e.

$$\exists\,(l_1, i_1, expr, \theta, l_2, i_2) \in \mathsf{LocInv} : l \in \{l_1, l_2\}, \text{ or}$$

  then there is a location l' **equivalent** to l which is the location of

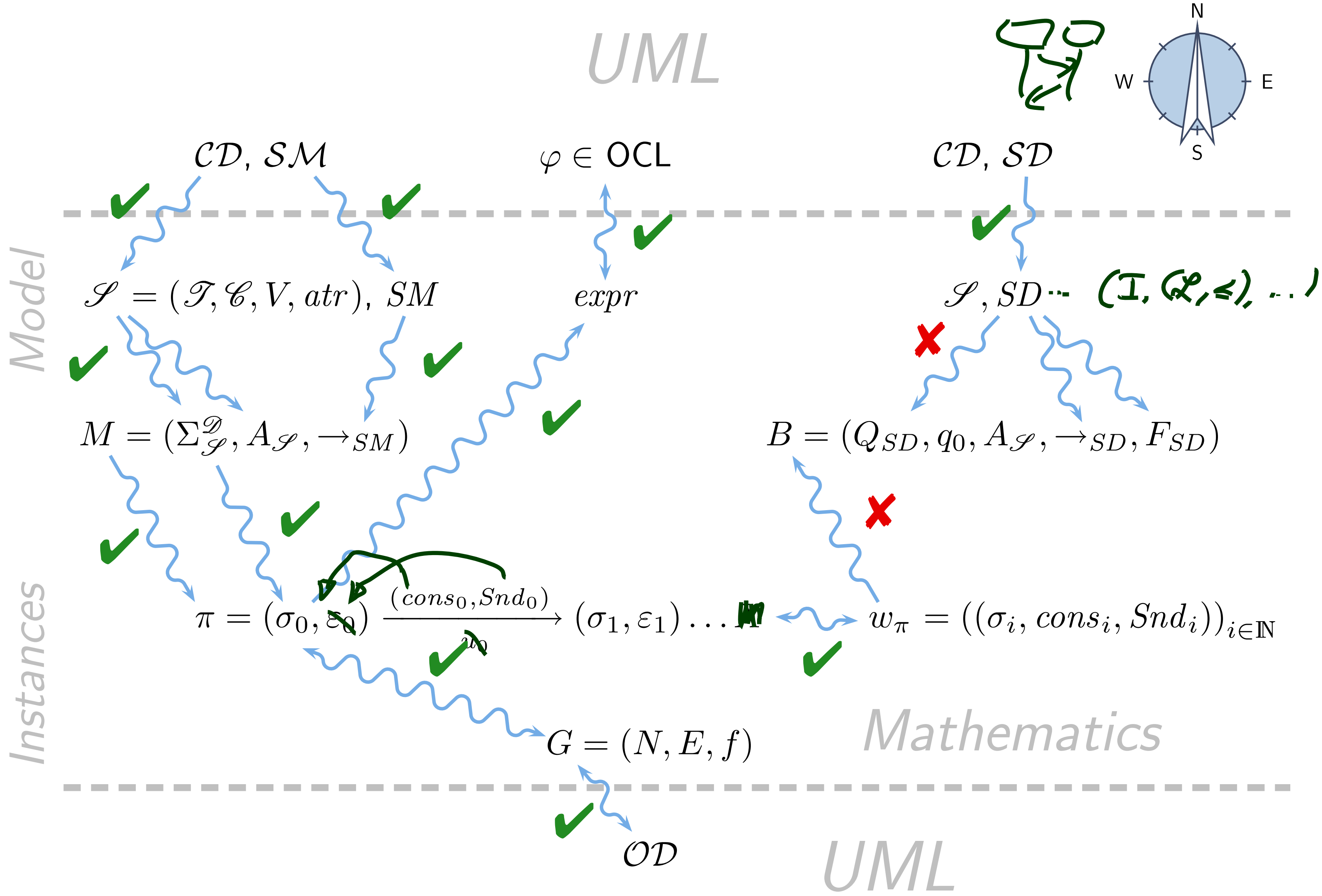
  - a **message**, i.e.

$$\exists\,(l_1, b, l_2) \in \mathsf{Msg} : l \in \{l_1, l_2\}, \text{ or}$$

  - an **instance head**, i.e. l' is minimal wrt. $\preceq$.

Note: if messages in a chart are **cyclic**, then there doesn't exist a partial order (so such charts don't even have an abstract syntax).

Course Map


UML
N
W
E
S
$\mathcal{CD}, \mathcal{SM}$
$\varphi \in$ OCL
$\mathcal{CD}, \mathcal{SD}$
Model
$\mathscr{S} = (\mathscr{T}, \mathscr{C}, V, atr),\ SM$
$expr$
$\mathscr{S}, SD$
$(I, \mathcal{L}, \leq), \ldots)$
$M = (\Sigma_{\mathscr{S}}^{\mathscr{D}}, A_{\mathscr{S}}, \to_{SM})$
$B = (Q_{SD}, q_0, A_{\mathscr{S}}, \to_{SD}, F_{SD})$
Instances
$\pi = (\sigma_0, \varepsilon_0) \xrightarrow[u_0]{(cons_0, Snd_0)} (\sigma_1, \varepsilon_1) \ldots$
$w_\pi = ((\sigma_i, cons_i, Snd_i))_{i \in \mathbb{N}}$
$G = (N, E, f)$
Mathematics
$\mathcal{OD}$
UML

Live Sequence Charts Semantics

TBA-based Semantics of LSCs

Plan:

- Given an LSC L with body

$$(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$$

- Construct a TBA $\mathcal{B}_L$ — taking the **cuts** of L as states.
- Define $\mathcal{L}(L)$ **in terms of** $\mathcal{L}(\mathcal{B}_L)$,
  in particular taking activation condition and activation mode into account.

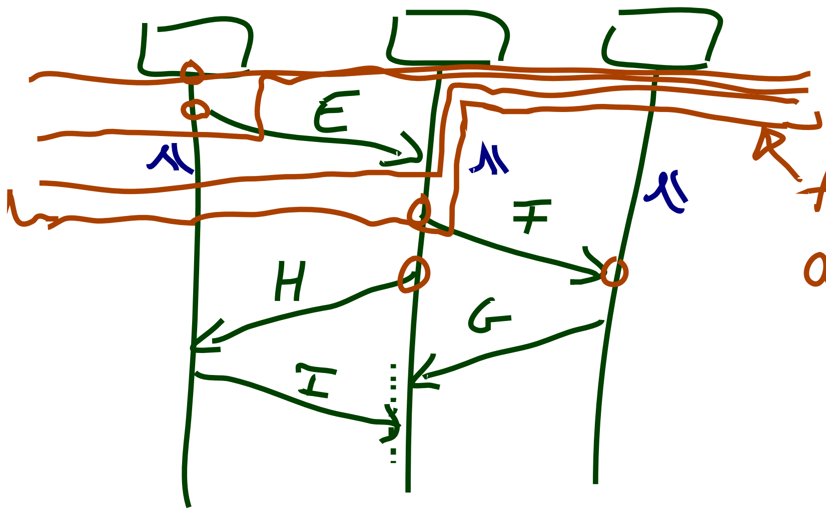
Formal LSC Semantics: It's in the Cuts

- Let $(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$ be an LSC body.
- A non-empty set

$$\emptyset \neq C \subseteq \mathscr{L}$$

  is called a **cut** of the LSC body if and only if

  - it is **downward closed**, i.e. $\forall\, l, l' : l' \in C \wedge l \preceq l' \implies l \in C$,
  - it is **closed** under **simultaneity**, i.e. $\forall\, l, l' : l' \in C \wedge l \sim l' \implies l \in C$, and
  - it comprises at least **one location per instance line**, i.e. $\forall\, i \in I\; \exists\, l \in C : i_l = i$.



front of cut (maximal elements)

ok: send H alone,
recv. F alone,
both at same time

three firedsets

Formal LSC Semantics: It's in the Cuts

- Let $(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$ be an LSC body.
- A non-empty set
$$\emptyset \neq C \subseteq \mathscr{L}$$
is called a **cut** of the LSC body if and only if
  - it is **downward closed**, i.e. $\forall\, l, l' : l' \in C \wedge l \preceq l' \implies l \in C$,
  - it is **closed** under **simultaneity**, i.e. $\forall\, l, l' : l' \in C \wedge l \sim l' \implies l \in C$, and
  - it comprises at least **one location per instance line**, i.e. $\forall\, i \in I\; \exists\, l \in C : i_l = i$.
- A cut C is called **hot**, denoted by $\theta(C) = \mathsf{hot}$, if and only if at least one of its maximal elements is hot, i.e. if
$$\exists\, l \in C : \theta(l) = \mathsf{hot} \wedge \nexists\, l' \in C : l \prec l'$$
Otherwise, C is called **cold**, denoted by $\theta(C) = \mathsf{cold}$.

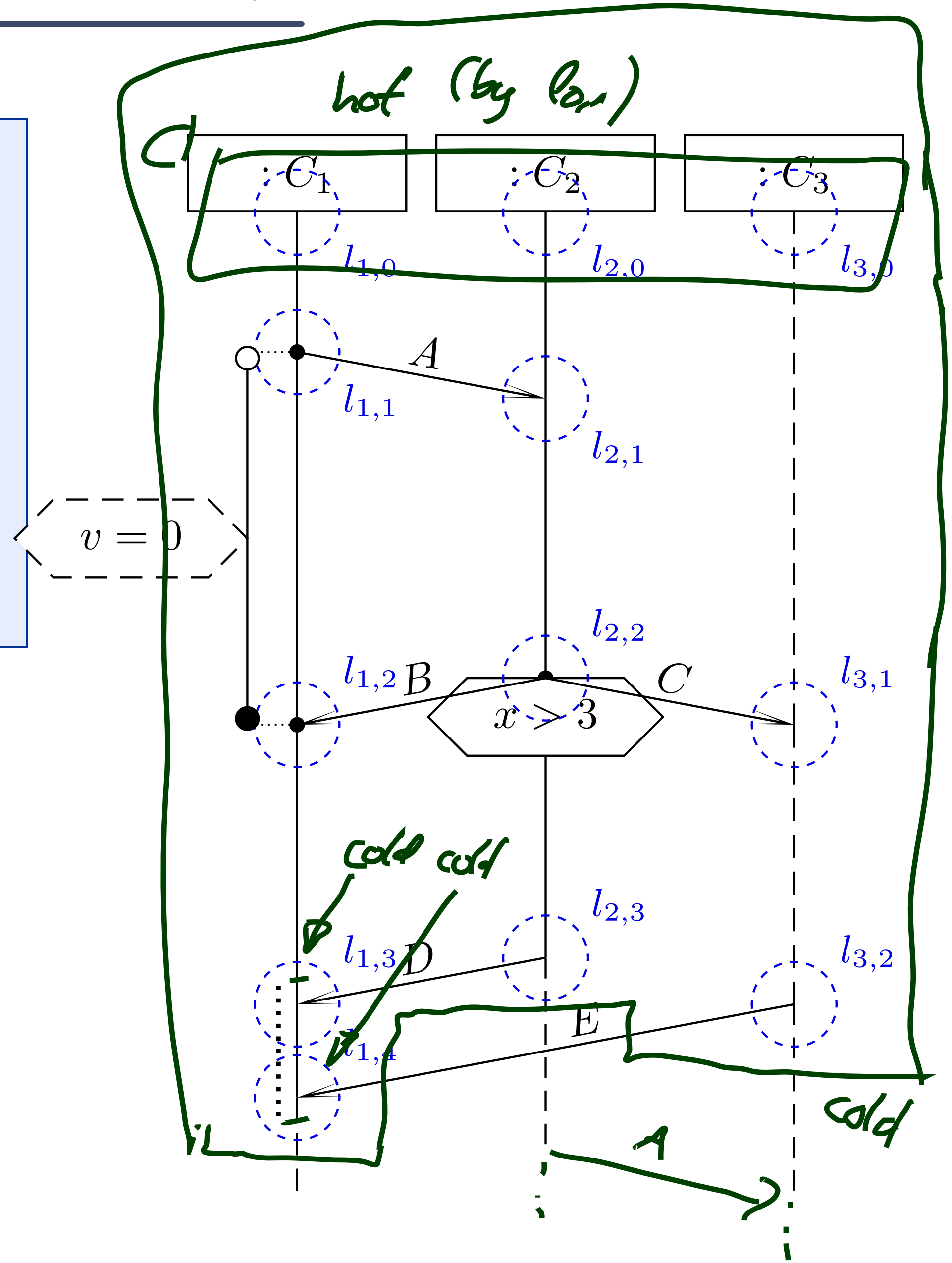
Examples: Cut or Not Cut? Hot/Cold?

(i) **non-empty** set $\emptyset \neq C \subseteq \mathscr{L}$,

(ii) **downward closed**, i.e.
$\forall\, l, l' : l' \in C \wedge l \preceq l' \implies l \in C$

(iii) **closed** under **simultaneity**, i.e.
$\forall\, l, l' : l' \in C \wedge l \sim l' \implies l \in C$

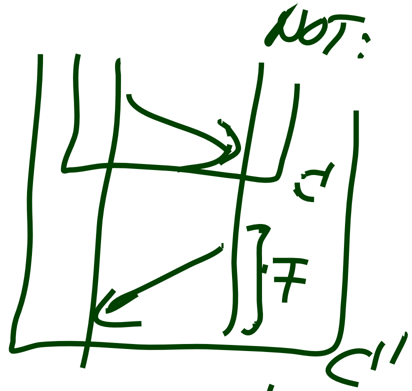
(iv) at least **one location per instance line**, i.e.
$\forall\, i \in I\; \exists\, l \in C : i_l = i$,



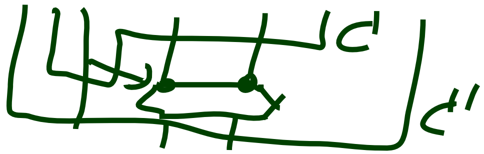
- $C_0 = \emptyset$
- $C_1 = \{l_{1,0}, l_{2,0}, l_{3,0}\}$
- $C_2 = \{l_{1,1}, l_{2,1}, l_{3,0}\}$
- $C_3 = \{l_{1,0}, l_{1,1}\}$
- $C_4 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{3,0}\}$
- $C_5 = \{l_{1,0}, l_{1,1}, l_{2,0}, l_{2,1}, l_{3,0}\}$
- $C_6 = \mathscr{L} \setminus \{l_{1,3}, l_{2,3}\}$
- $C_7 = \mathscr{L}$

A Successor Relation on Cuts

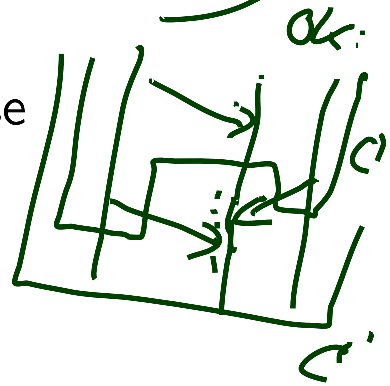
The partial order of $(\mathscr{L}, \preceq)$ and the simultaneity relation "$\sim$" induce a **direct successor relation** on cuts of $\mathscr{L}$ as follows:



- Let $C, C' \subseteq \mathscr{L}$ bet cuts. C' is called **direct successor** of C **via fired-set** F $\subseteq \mathscr{L}$, denoted by $C \leadsto_F C'$, if and only if



  - $F \neq \emptyset$,
  - $C' \setminus C = F$,
  - for each message reception in F, the corresponding sending is already in C,
  - locations in F, that lie on the same instance line, are pairwise unordered, i.e.



∈ I, instance line of l

$$\forall\, l, l' \in F : l \neq l' \wedge i_l = i_{l'} \implies l \not\preceq l' \wedge l' \not\preceq l$$

- ~~Note:~~ F is ~~immediately~~ closed under simultaneity. $(\sim)$

- In other words: locations in F are direct $\preceq$-successors of locations in C, i.e.

$$\forall\, l' \in F\; \exists\, l \in C : l \prec l' \wedge \nexists\, l'' \in C : l' \prec l'' \prec l$$

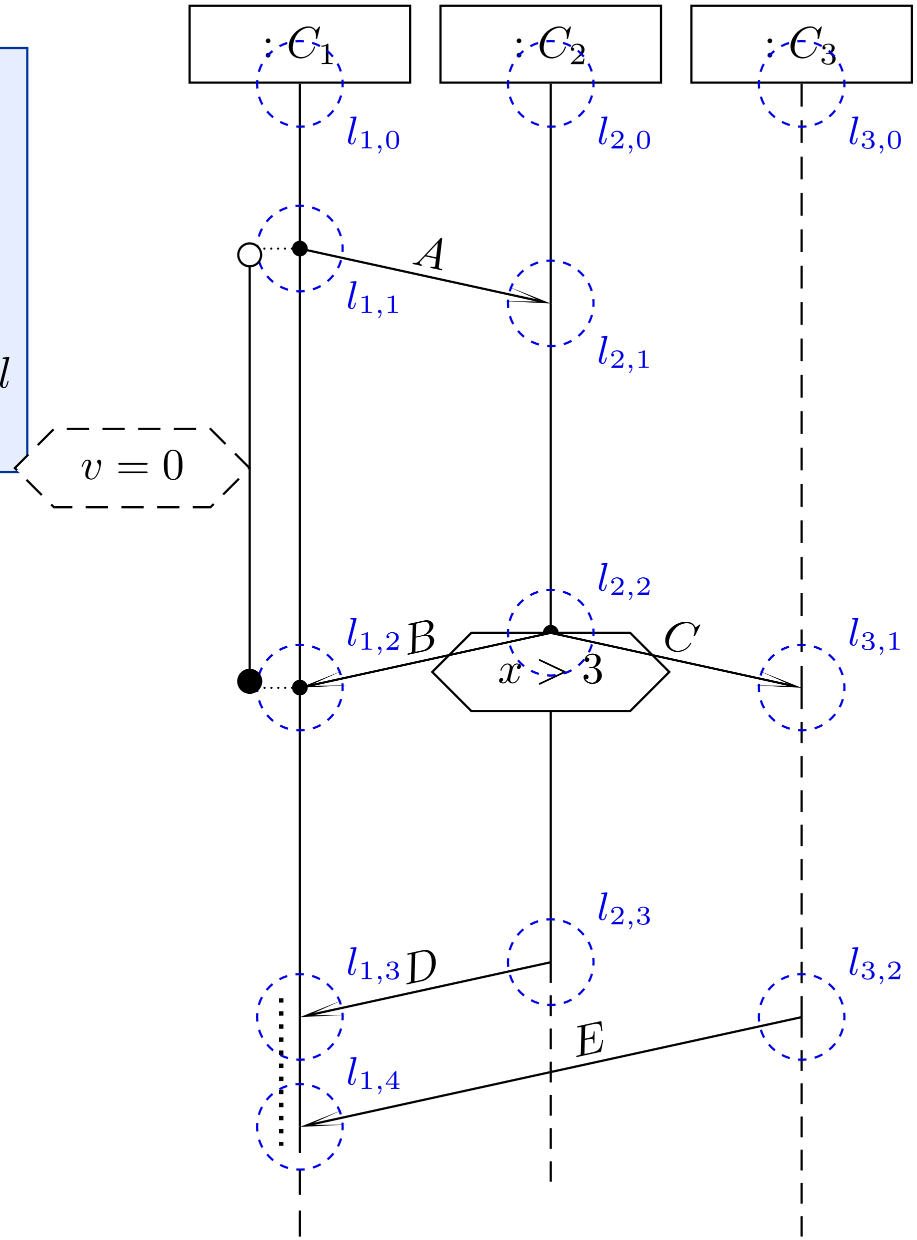
Successor Cut Examples

(i) $F \neq \emptyset$,

(ii) $C' \setminus C = F$,

(iii) message send before receive,

(iv) locations on same instance line unordered, i.e.
$\forall\, l, l' \in F : l \neq l' \wedge i_l = i_{l'} \implies l \not\preceq l' \wedge l' \not\preceq l$



Idea: Accepting Words by Advancing the Cut

Let $w = (\sigma_i, cons_i, Snd_i)_{i \in \mathbb{N}_0}$ be a word over $\mathscr{S}$ and $\mathscr{D}$.

Intuitively (and for now **disregarding** cold conditions),
an LSC body $(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$ is **supposed** to **accept** w
(under valuation β) if and only if there exists a sequence

which maps instance lines to objects

$$C_0 \rightsquigarrow_{F_1} C_1 \rightsquigarrow_{F_2} C_2 \cdots \rightsquigarrow_{F_n} C_n$$

and indices $i_1 < \cdots < i_n$ such that

- C_0 consists of the instance heads,
- for all $1 \leq j < n$,
  - for all $i_j \leq k < i_{j+1}$, $(\sigma_k, cons_k, Snd_k)$ satisfies (under β) the **hold condition** of C_{j-1},
  - $(\sigma_{i_j}, cons_{i_j}, Snd_{i_j})$ satisfies (under β) the **transition condition** of F_j,
- C_n is cold, $C_n = \mathscr{L}$
- for all $i_n < k$, $(\beta_k, \mu_{i_j}, t_{i_j})$ satisfies (under β) the **hold condition** of C_n.

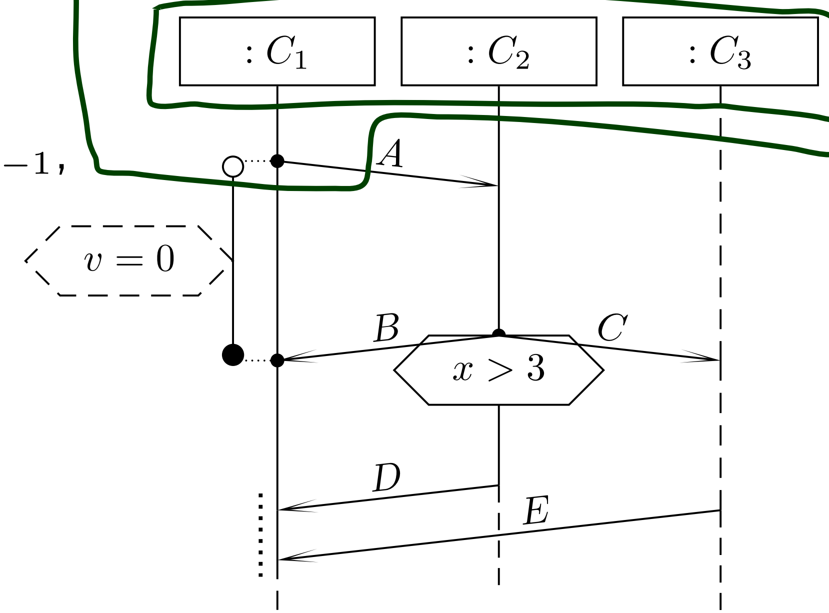
trivial

w: X! X? Y! Y? "hold"

traces to C_1 [A! X!] "hold"

traces. [A?

C_1 C_0



Excursus: Symbolic Büchi Automata (over Signature)

Definition. A **Symbolic Büchi Automaton** (TBA) is a tuple

$$\mathcal{B} = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \to, Q_F)$$

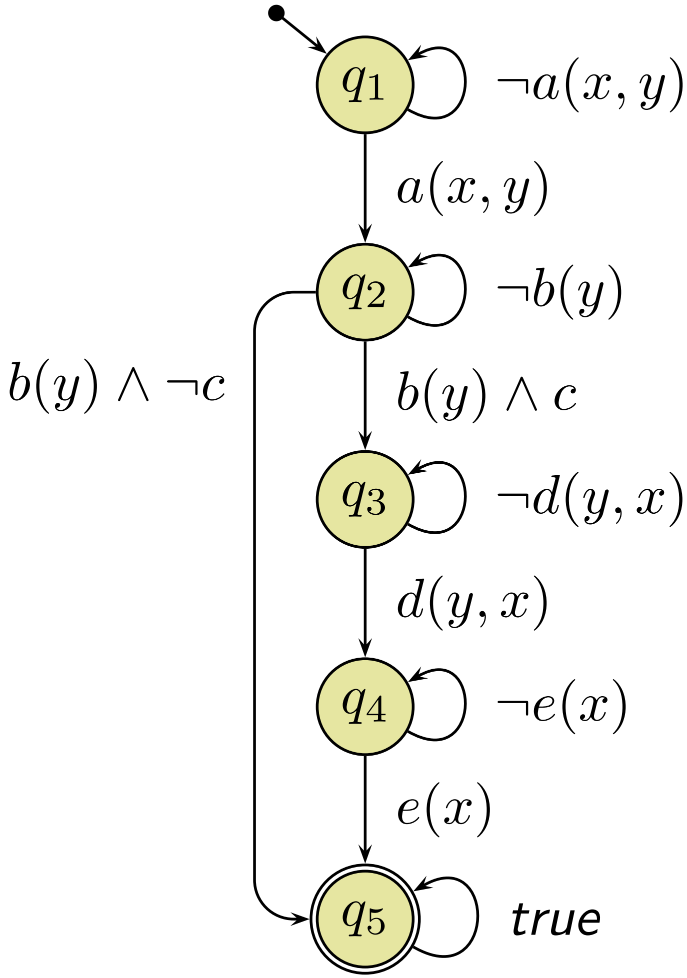
where

- $Expr_{\mathcal{B}}$ is a set of expressions over logical variables from X,
- Q is a finite set of **states**, q_{ini} $\in Q$ the initial state,
- $\to \subseteq Q \times Expr_{\mathcal{B}} \times Q$ is the **transition relation**.

  Transitions $(q, expr, q')$ from q to q' are labelled with a constraint $expr \in Expr_{\mathcal{B}}$ over the ~~signals and the~~ variables.
- $Q_F \subseteq Q$ is the set of **fair** (or accepting) states.

TBA Example

$$(Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$$



$Expr_{\mathcal{B}} = \{ a(x,y),$
$\neg a(x,y),$
$c, \ldots \}, \; x,y \in X$

$Q = \{ q_1, \ldots, q_5 \}$

$q_{ini} = q_1$

$Q_F = \{ q_5 \}$

Word

Definition. Let $Expr_{\mathcal{B}}$ be a set of expressions over logical variables X. and let Σ be the set of interpretation functions of $Expr_{\mathcal{B}}$, i.e.

$$\Sigma = Expr_{\mathcal{B}} \times (X \to \mathscr{D}(X)) \to \{0, 1\}.$$

For $\sigma \in \Sigma$, we write $\sigma \models_\beta expr$ if and only if $\sigma(expr, \beta) = 1$.

A **word** over $Expr_{\mathcal{B}}$ is an infinite sequence of interpretations of $Expr_{\mathcal{B}}$

$$(\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega.$$

w: $\sigma_0 \models_\beta a(x,y)$, $\beta = \{x \mapsto 1, y \mapsto 27\}$

$\sigma_1 \models_\beta c$, $\sigma_1 \models e(x)$ (nothing else)

$\vdots$

Run of TBA over Word

Definition. Let $\mathcal{B} = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$ be a TBA and

$$w = (\sigma_i)_{i \in \mathbb{N}_0} \in \Sigma^\omega$$

a word over $Expr_{\mathcal{B}}$.

An infinite sequence

$$\varrho = q_0, q_1, q_2, \ldots \in Q^\omega$$

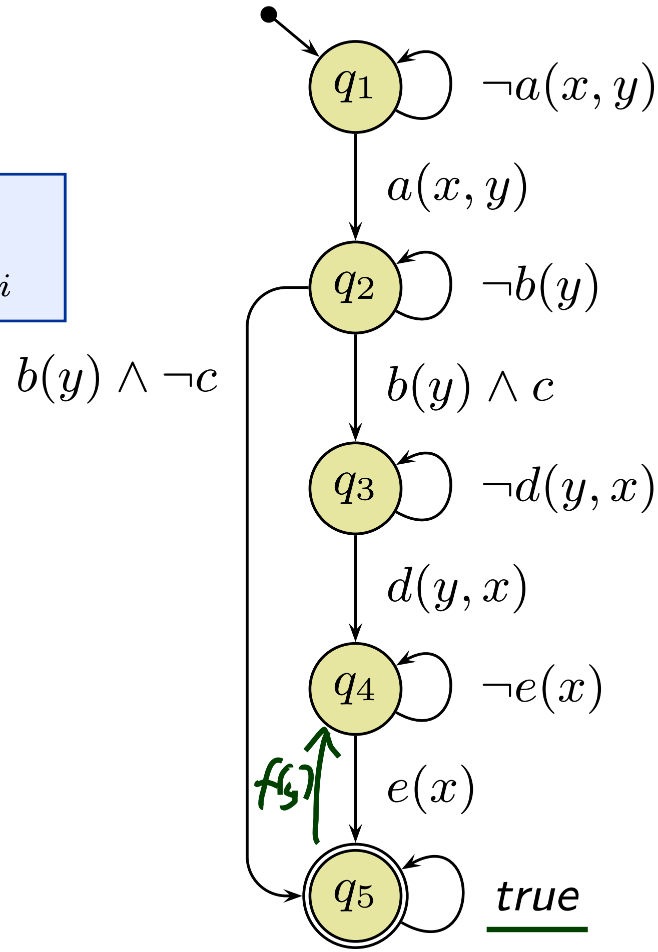
is called **run** of $\mathcal{B}$ over w under valuation $\beta : X \rightarrow \mathscr{D}(X)$ if and only if

- $q_0 = q_{ini}$,
- for each $i \in \mathbb{N}_0$ there is a transition $(q_i, \psi_i, q_{i+1}) \in \rightarrow$ such that

$$\sigma_i \models_\beta \psi_i.$$

Run or Not Run Examples

$$\varrho = (q_i)_{i \in \mathbb{N}_0}, \qquad q_0 = q_{ini},$$
$$\forall\, i \in \mathbb{N}_0\ \exists\, (q_i, \psi_i, q_{i+1}) \in \rightarrow : (\sigma_i, cons_i, Snd_i) \models_\beta \psi_i$$



w: $\sigma_0 \not\models_\beta a(x,y)$ $(\Rightarrow \sigma_0 \models_\beta \neg a(x,y))$

$\sigma_1 \not\models_\beta a(x,y)$

$\sigma_2 \models_\beta a(x,y)$, $\sigma_2 \models_\beta e(x)$

$\sigma_3 \models_\beta b(y) \wedge \neg c$

$\sigma_4 \models_\beta e(x) \wedge d(y,x)$

$\vdots$

$\varrho = q_1\ q_1\ q_1\ q_2\ q_5\ q_5\ q_5 \ldots$

by by by by by ⋱

σ_0 σ_1 σ_2 σ_3 σ_4

The Language of a TBA

Definition.
We say $\mathcal{B} = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$ **accepts** w (under valuation $\beta : X \rightarrow \mathscr{D}(X)$) if and only if $\mathcal{B}$ **has a run**

$$(q_i)_{i \in \mathbb{N}_0}$$

over w such that fair (or accepting) states are **visited infinitely often**, that is,

$$\forall\, i \in \mathbb{N}_0 \;\exists\, j > i : q_j \in Q_F.$$

We call the set $\mathcal{L}_\beta(\mathcal{B})$ of words over $\mathscr{S}$ that are accepted by $\mathcal{B}$ under β the **language of $\mathcal{B}$**.

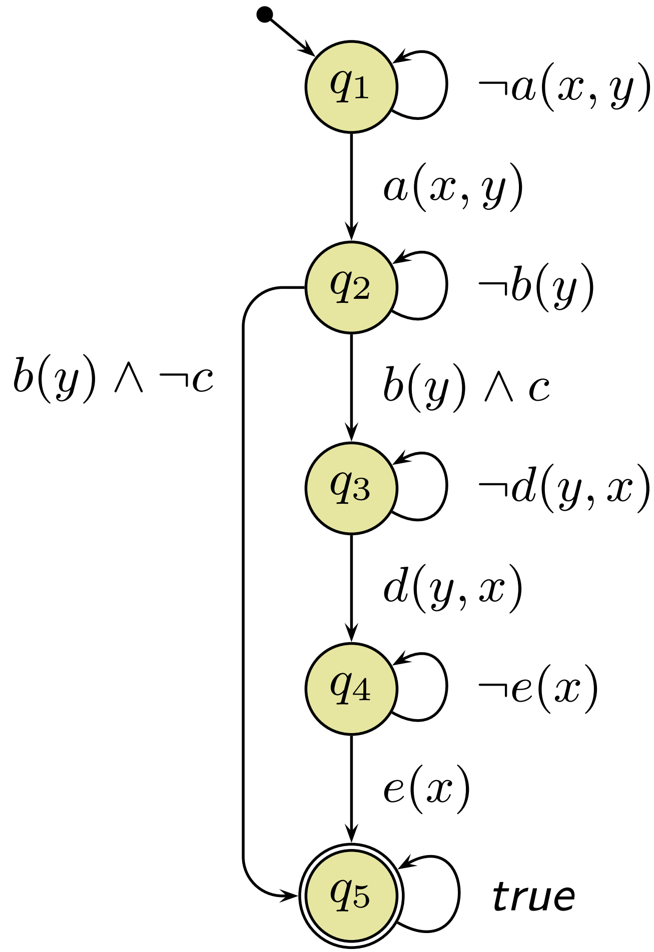
Language of the Example TBA

$\mathcal{L}_\beta(\mathcal{B})$ consists of the words

$$(\sigma_i, Snd_i, cons_i)_{i \in \mathbb{N}_0}$$

where there exist $0 \leq n < m < k < \ell$ such that

- for $0 \leq i < n$, $\sigma_i \not\models_\beta a(x,y)$
- $\sigma_n \models_\beta a(x,y)$
- for $n < i < m$, $\sigma_i \not\models_\beta b(y)$
- $\sigma_m \models_\beta b(y) \wedge c$ and
    - for $m < i < k$, $\sigma_i \not\models_\beta d(y,x)$
    - $\sigma_k \models_\beta d(y,x)$
    - for $k < i < \ell$, $\sigma_i \not\models_\beta e(x)$
    - $\sigma_\ell \models_\beta e(x)$, or
- $\sigma_m \models_\beta b(y) \wedge \neg c$



Back to Main Track: Live Sequence Charts Semantics

Recall Idea: Accepting Words by Advancing the Cut

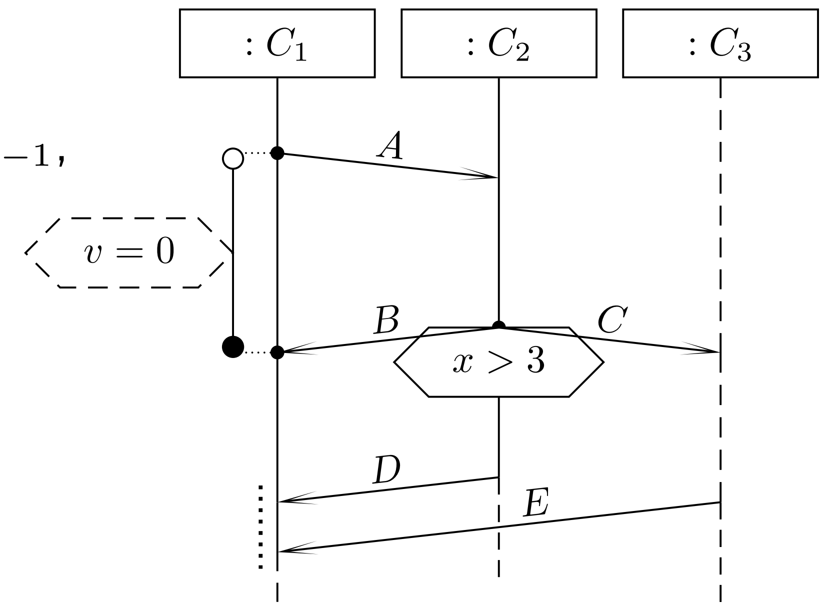
Let $w = (\sigma_i, cons_i, Snd_i)_{i \in \mathbb{N}_0}$ be a word over $\mathscr{S}$ and $\mathscr{D}$.

Intuitively (and for now **disregarding** cold conditions),
an LSC body $(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$ is **supposed** to **accept** w (under valuation β) if and only if there exists a sequence

$$C_0 \rightsquigarrow_{F_1} C_1 \rightsquigarrow_{F_2} C_2 \cdots \rightsquigarrow_{F_n} C_n$$

and indices $i_1 < \cdots < i_n$ such that

- C_0 consists of the instance heads,
- for all $1 \leq j < n$,
  - for all $i_j \leq k < i_{j+1}$, $(\sigma_k, cons_k, Snd_k)$ satisfies (under β) the **hold condition** of C_{j-1},
  - $(\sigma_{i_j}, cons_{i_j}, Snd_{i_j})$ satisfies (under β) the **transition condition** of F_j,
- C_n is cold,
- for all $i_n < k$, $(\beta_k, \mu_{i_j}, t_{i_j})$ satisfies (under β) the **hold condition** of C_n.



Language of LSC Body

The **language** of the body

$$(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$$

of LSC L is the language of the TBA

$$\mathcal{B}_L = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$$

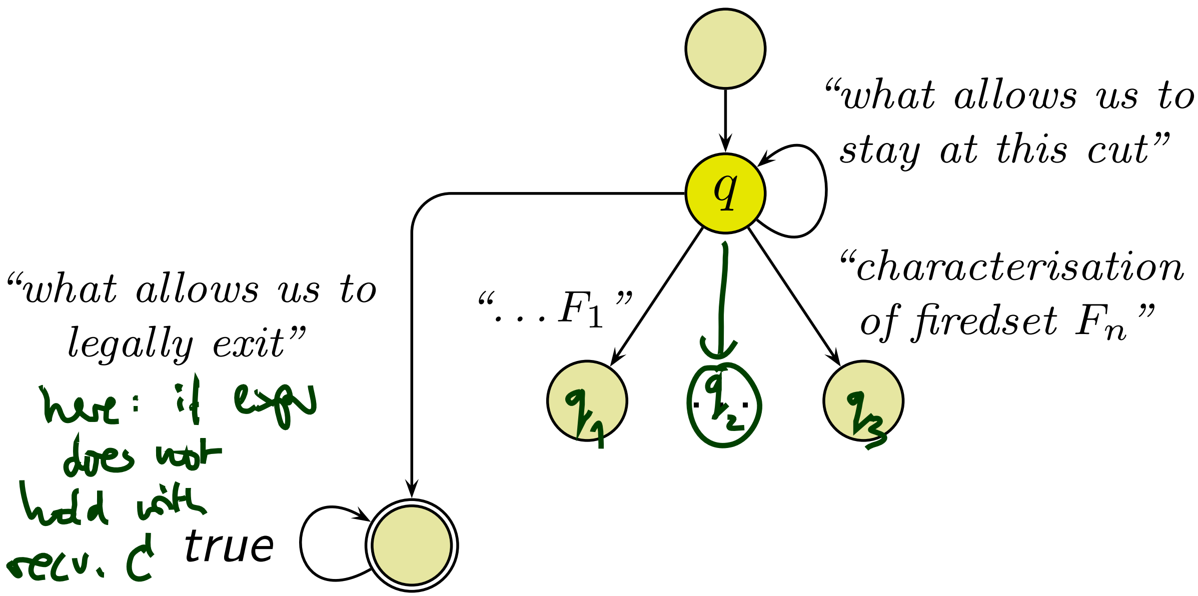
with

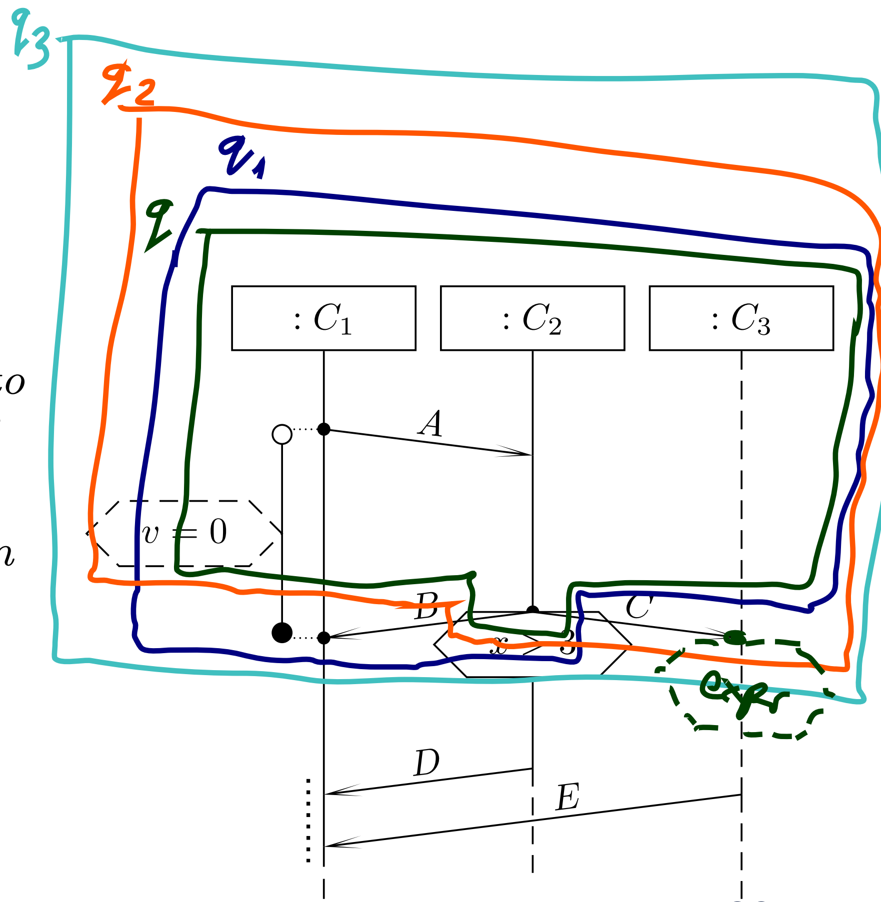
- $Expr_{\mathcal{B}} = Expr_{\mathscr{S}}(V, \mathscr{E}(\mathscr{S}))$
- Q is the set of cuts of $(\mathscr{L}, \preceq)$, q_{ini} is the **instance heads** cut,
- Q $F = \{C \in Q \mid \theta(C) = \text{cold}\}$ is the set of cold cuts of $(\mathscr{L}, \preceq)$,
- $\rightarrow$ as defined in the following, consisting of
  - **loops** (q, ψ, q),
  - **progress transitions** (q, ψ, q'), and
  - **legal exits** $(q, \psi, \mathscr{L})$.

Language of LSC Body: Intuition

$\mathcal{B}_L = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$ with

- $Expr_{\mathcal{B}} = Expr_{\mathscr{S}}(V, \mathscr{E}(\mathscr{S}))$
- Q is the set of cuts of $(\mathscr{L}, \preceq)$, q_{ini} is the **instance heads** cut,
- $F = \{C \in Q \mid \theta(C) = \text{cold}\}$ is the set of cold cuts,
- $\rightarrow$ consists of
  - **loops** (q, ψ, q),
  - **progress transitions** (q, ψ, q'), and
  - **legal exits** $(q, \psi, \mathscr{L})$.





Signal and Integer Expressions

Let $\mathscr{S} = (\mathscr{T}, \mathscr{C}, V, atr)$ be a signature and X a set of logical variables.

The **signal and integer expressions** $Expr_{\mathscr{S}}(V, \mathscr{E}(\mathscr{S}))$ over $\mathscr{S}$ are defined by the grammar:

$$\psi ::= \textsf{true} \mid expr \mid E^!_{x,y} \mid E^?_x \mid \neg\psi \mid \psi_1 \vee \psi_2,$$

where $expr \in Expr_{\mathscr{S}}$, $E \in \mathscr{E}$, $x, y \in X$.

consumes (x, E)

send (x, E, y)

Satisfaction of Signal and Integer Expressions

Let $(\sigma, cons, Snd) \in \left(\Sigma_{\mathscr{S}}^{\mathscr{D}} \times 2^{\mathscr{D}(\mathscr{C}) \times Evs(\mathscr{E},\mathscr{D})} \times 2^{\mathscr{D}(\mathscr{C}) \times Evs(\mathscr{E},\mathscr{D}) \times \mathscr{D}(\mathscr{C})}\right)$ be a letter of a word over $\mathscr{S}$ and $\mathscr{D}$ and let $\beta : X \to \mathscr{D}(\mathscr{C})$ be a valuation of the logical variables in X.

- $(\sigma, cons, Snd) \models_\beta$ *true*
- $(\sigma, cons, Snd) \models_\beta \neg\psi$ if and only if not $(\sigma, cons, Snd) \models_\beta \psi$
- $(\sigma, cons, Snd) \models_\beta \psi_1 \vee \psi_2$ if and only if
  $(\sigma, cons, Snd) \models_\beta \psi_1$ or $(\sigma, cons, Snd) \models_\beta \psi_2$

sender ID destination id

- $(\sigma, cons, Snd) \models_\beta expr$ if and only if $I[\![expr]\!](\sigma, \beta) = 1$
- $(\sigma, cons, Snd) \models_\beta E^!_{x,y}$ if and only if $(\beta(x), (E, \vec{d}), \beta(y)) \in Snd$

event

- $(\sigma, cons, Snd) \models_\beta E^?_x$ if and only if $(\beta(x), (E, \vec{d})) \in cons$

e.g. d.x>5

Satisfaction of Signal and Integer Expressions

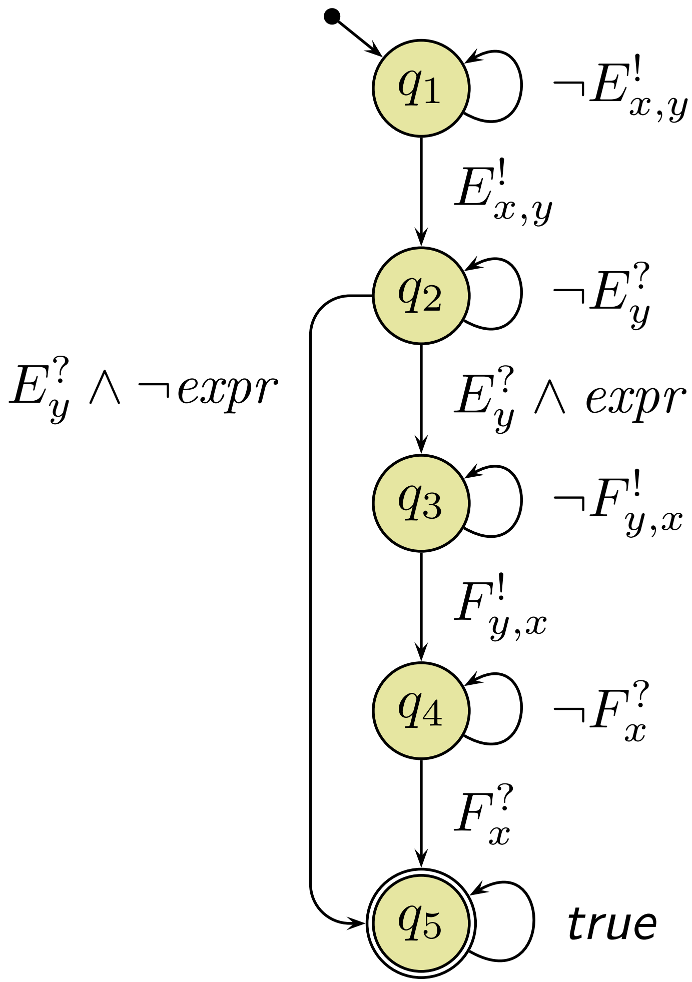
Let $(\sigma, cons, Snd) \in \left(\Sigma_{\mathscr{S}}^{\mathscr{D}} \times 2^{\mathscr{D}(\mathscr{C}) \times Evs(\mathscr{E}, \mathscr{D})} \times 2^{\mathscr{D}(\mathscr{C}) \times Evs(\mathscr{E}, \mathscr{D}) \times \mathscr{D}(\mathscr{C})}\right)$ be a letter of a word over $\mathscr{S}$ and $\mathscr{D}$ and let $\beta : X \to \mathscr{D}(\mathscr{C})$ be a valuation of the logical variables in X.

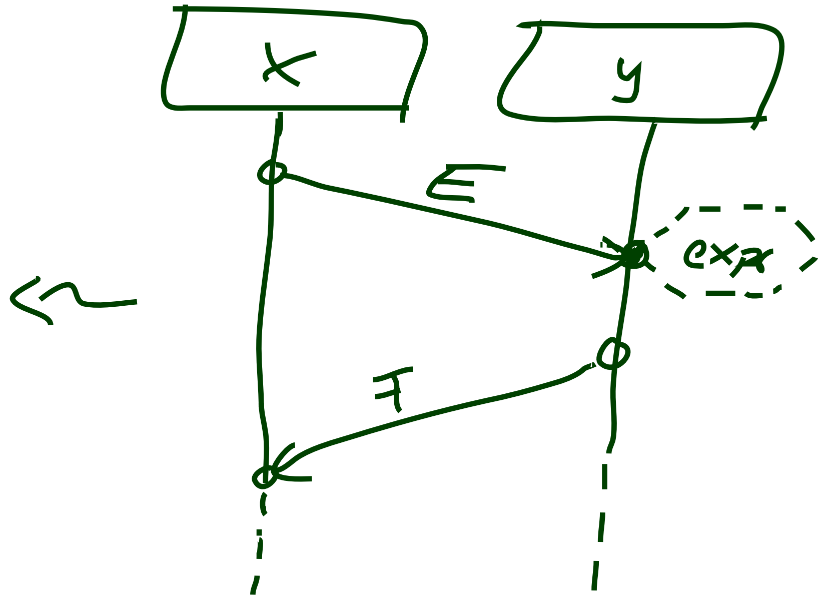
- $(\sigma, cons, Snd) \models_\beta$ *true*
- $(\sigma, cons, Snd) \models_\beta \neg\psi$ if and only if not $(\sigma, cons, Snd) \models_\beta \psi$
- $(\sigma, cons, Snd) \models_\beta \psi_1 \vee \psi_2$ if and only if
  $(\sigma, cons, Snd) \models_\beta \psi_1$ or $(\sigma, cons, Snd) \models_\beta \psi_2$
- $(\sigma, cons, Snd) \models_\beta expr$ if and only if $I[\![expr]\!](\sigma, \beta) = 1$
- $(\sigma, cons, Snd) \models_\beta E^!_{x,y}$ if and only if $(\beta(x), (E, \vec{d}), \beta(y)) \in Snd$
- $(\sigma, cons, Snd) \models_\beta E^?_x$ if and only if $(\beta(x), (E, \vec{d})) \in cons$

Observation: if the semantics has "**forgotten**" the sender at consumption time, then we have to disregard it here (straightforwardly fixed if desired).

Other view: we could choose to disregard the sender.

Example: TBA over Signal and Integer Expressions





Some Helper Functions

starting or ending

- **Messages of a location**:

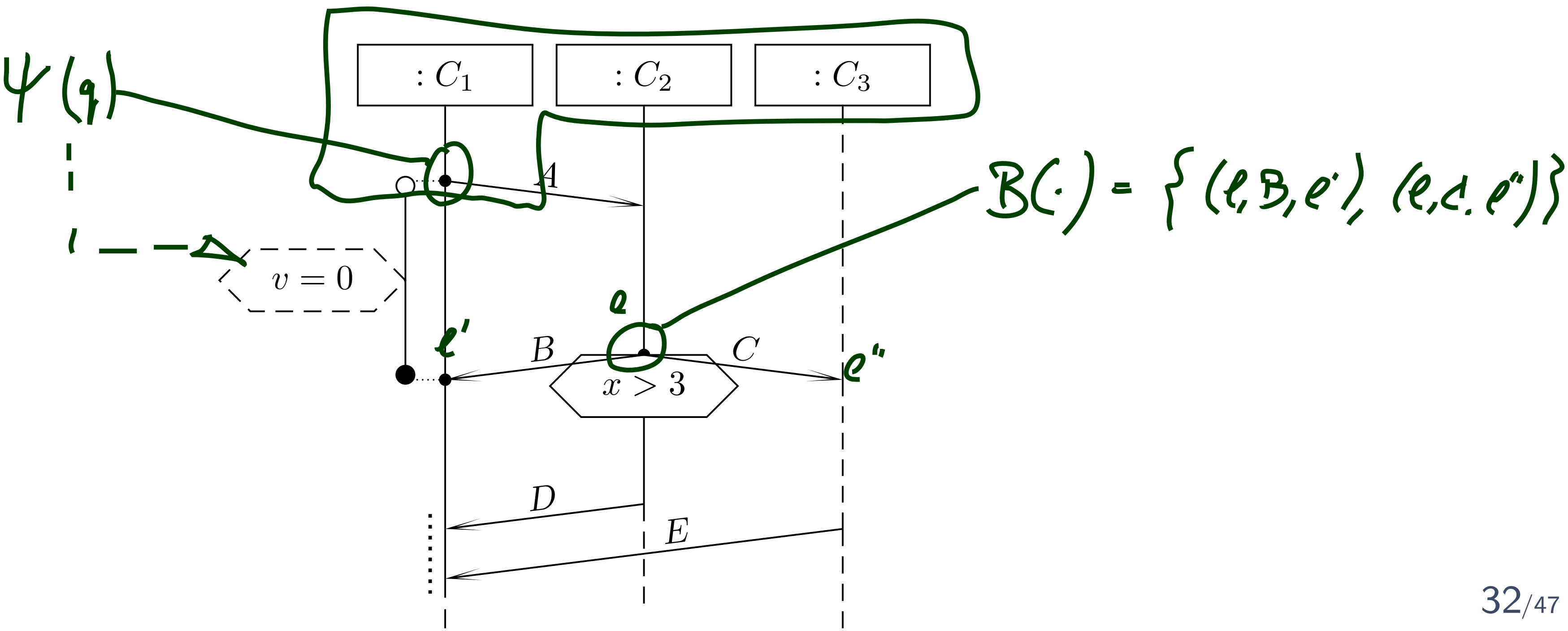
$\{(l', B, l'') \in Msg \mid l' = l \vee l'' = l\}$

$$B(l) := \cancel{\{b \in B \mid \exists\, l' : (l, b, l') \in \mathsf{Msg} \vee (l', b, l) \in \mathsf{Msg}\}},$$

$$B(\{l_1, \dots, l_n\}) := B(l_1) \cup \dots \cup B(l_n).$$

- **Constraints** relevant **at** cut q:

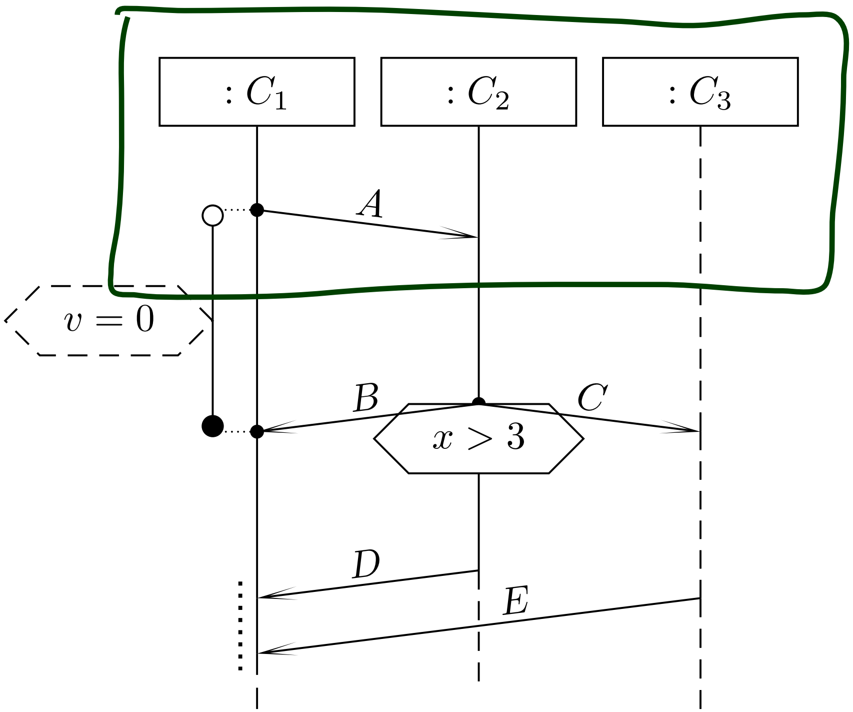
$$\psi(q) = \{\psi \mid \exists\, l \in q, l' \notin q \mid (l, \psi, \theta, l') \in \mathsf{LocInv} \vee (l', \psi, \theta, l) \in \mathsf{LocInv}\},$$



Some More Helper Functions

- **Constraints** relevant when moving from q **to** cut q':

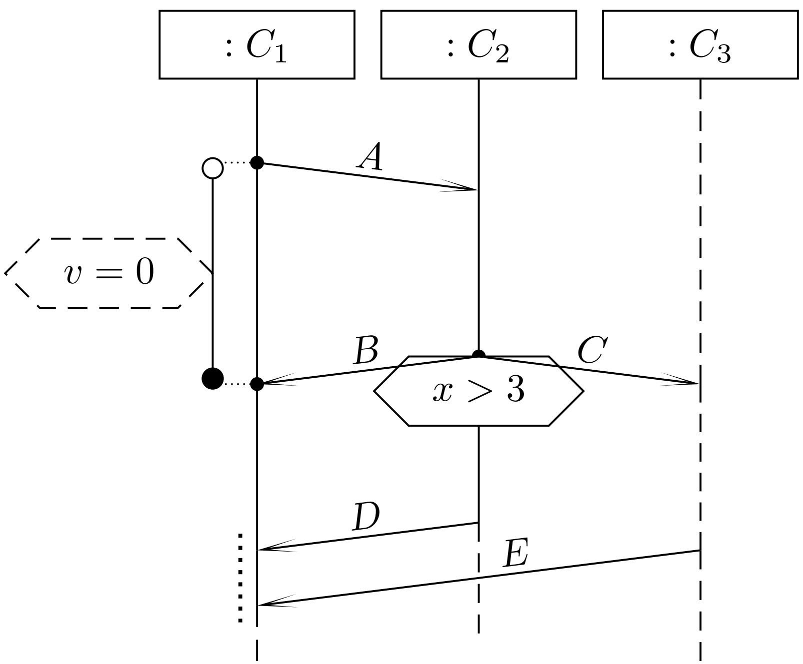
$$\begin{aligned}\psi(q, q') = \{\psi \mid \exists\, l \in q' \setminus q, l' \in \mathscr{L}, \theta \in \Theta \mid \\ (l, \bullet, expr, \theta, l') \in \mathsf{LocInv} \vee (l', expr, \theta, l, \bullet) \in \mathsf{LocInv}\} \\ \cup \{\psi \mid \exists\, l \in q, l' \notin q', \theta \in \Theta \mid \\ (l, expr, \theta, l') \in \mathsf{LocInv} \vee (l', expr, \theta, l) \in \mathsf{LocInv}\} \\ \cup \{\psi \mid \exists\, L \subseteq \mathscr{L}, \theta \in \Theta \mid (L, \psi, \theta) \in \mathsf{Cond} \wedge L \cap (q' \setminus q) \neq \emptyset\}\end{aligned}$$



Even More Helper Functions

- **Cold constraints** relevant when moving from q **to** cut q':

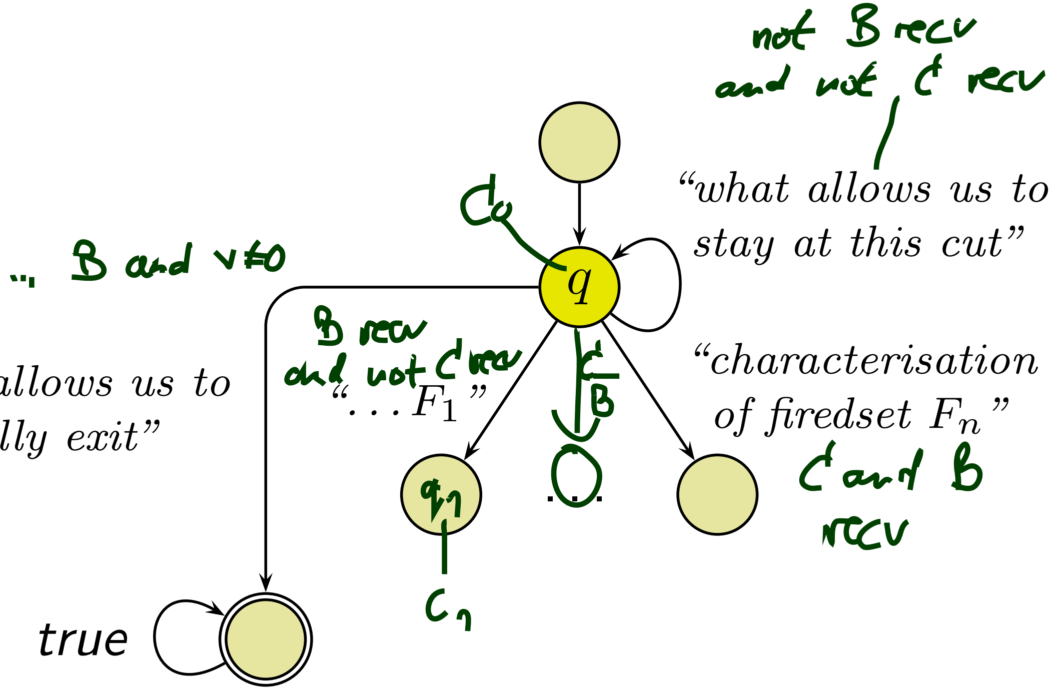
$$\begin{aligned}\psi_{\mathsf{cold}}(q,q') = \{\psi \mid \exists\, l \in q' \setminus q, l' \in \mathscr{L} \mid \\ (l, \bullet, expr, \mathsf{cold}, l') \in \mathsf{LocInv} \vee (l', expr, \mathsf{cold}, l, \bullet) \in \mathsf{LocInv}\} \\ \cup \{\psi \mid \exists\, l \in q, l' \notin q' \mid \\ (l, expr, \mathsf{cold}, l') \in \mathsf{LocInv} \vee (l', expr, \mathsf{cold}, l) \in \mathsf{LocInv}\} \\ \cup \{\psi \mid \exists\, L \subseteq \mathscr{L} \mid (L, \psi, \mathsf{cold}) \in \mathsf{Cond} \wedge L \cap (q' \setminus q) \neq \emptyset\}\end{aligned}$$

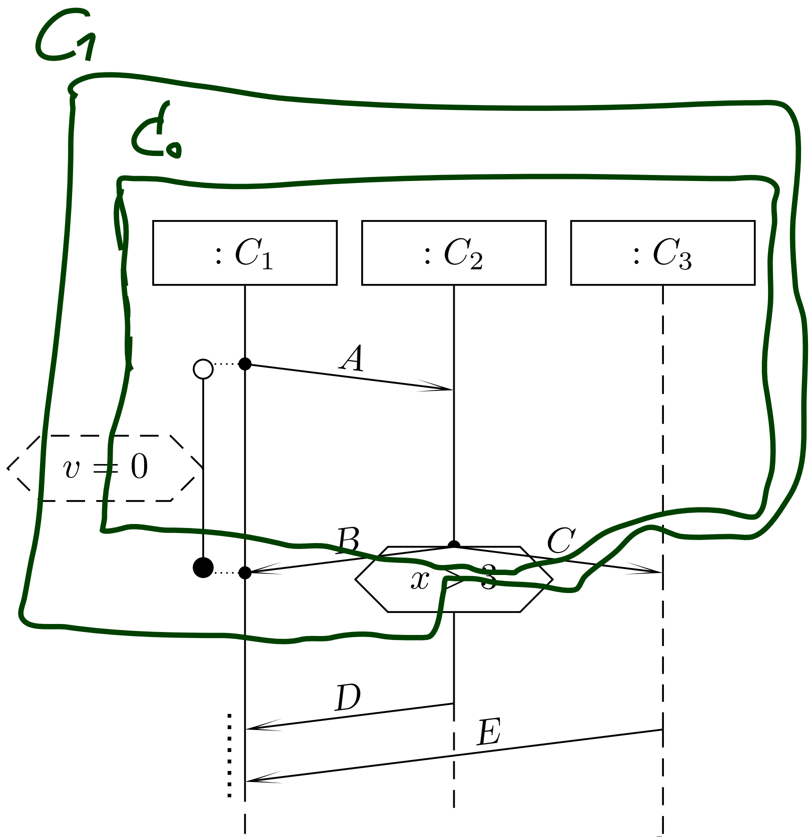


Recall: Intuition

$\mathcal{B}_L = (Expr_{\mathcal{B}}, X, Q, q_{ini}, \rightarrow, Q_F)$ with

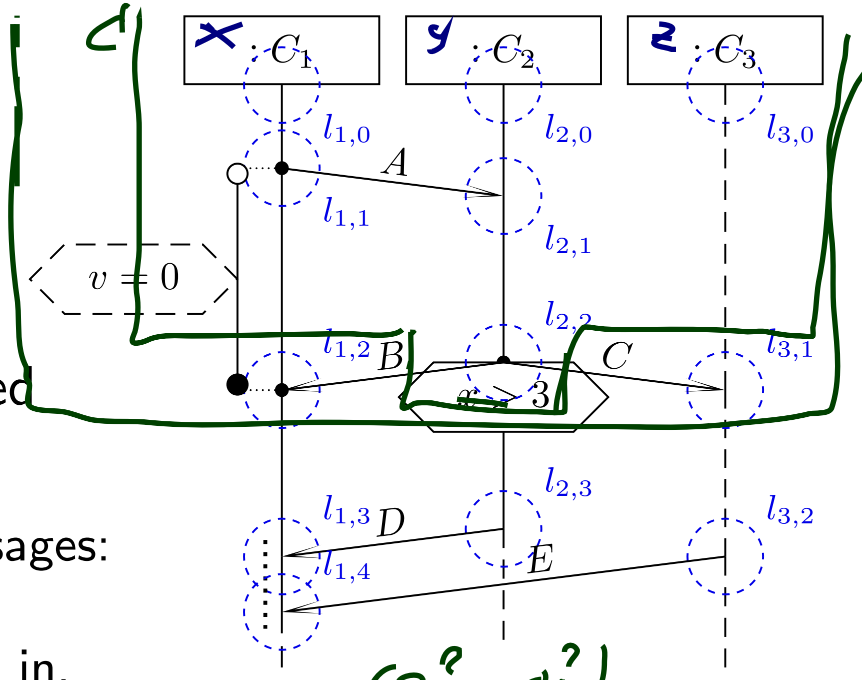
- $Expr_{\mathcal{B}} = Expr_{\mathscr{S}}(V, \mathscr{E}(\mathscr{S}))$
- Q is the set of cuts of $(\mathscr{L}, \preceq)$, q_{ini} is the **instance heads** cut,
- $F = \{C \in Q \mid \theta(C) = \text{cold}\}$ is the set of cold cuts,
- $\rightarrow$ consists of
  - **loops** (q, ψ, q),
  - **progress transitions** (q, ψ, q'), and
  - **legal exits** $(q, \psi, \mathscr{L})$.





Loops

one of three flow cuts → C'



$\neg(B_x^{?} \wedge C_z^{?})$
$\wedge\ v =$

- How long may we **legally** stay at a cut q?
- **Intuition**: those $(\sigma_i, cons_i, Snd_i)$ are allowed to fire the self-loop (q, ψ, q) where
  - $cons_i \cup Snd_i$ comprises only irrelevant messages:
    - **weak mode**: (permissive)
      no message from a direct successor cut is in,
    - **strict mode**:
      no message occurring in the LSC is in,
  - σ_i satisfies the local invariants active at q

  And nothing else.

  conjoin all constraints in the set $\psi(q)$

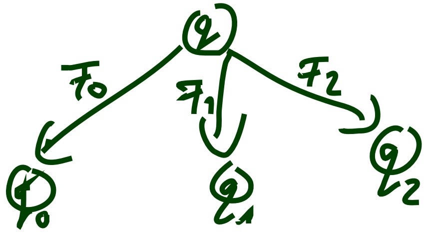
- **Formally**: Let $F := F_1 \cup \cdots \cup F_n$ is be the union of the firedsets of q.
  - $\psi := \neg(\underbrace{\bigvee_{F} B(F)}_{= \textit{true} \text{ if } F = \emptyset}) \wedge \bigwedge \psi(q).$ } weak mode

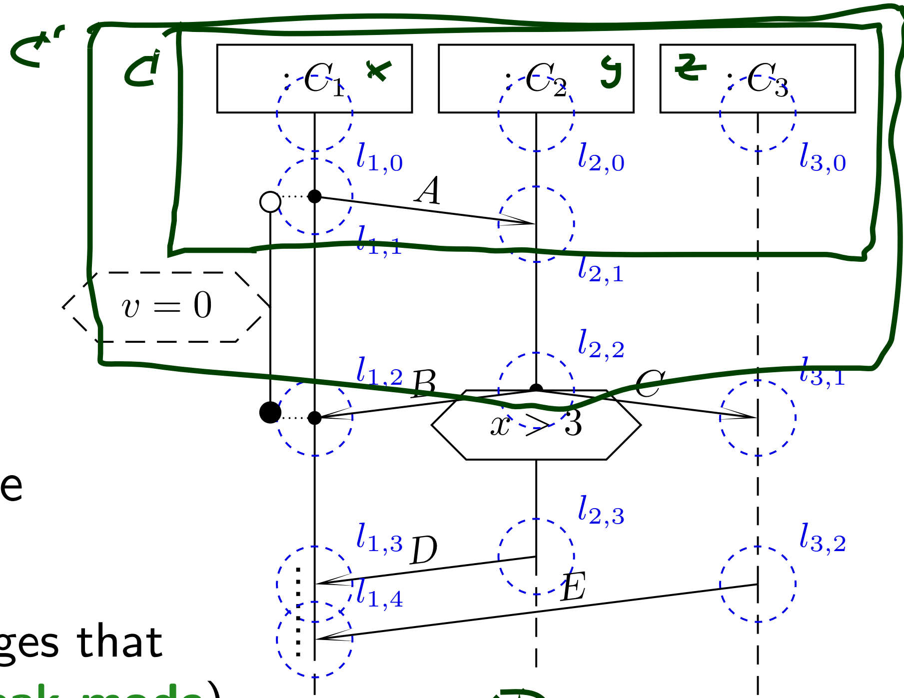
$\bigvee_{F} \{ (l_{2,2}, B, l_{1,2}), (l_{2,2}, C', l_{3,1}) \}$

$F_{(l_{1,2}, l_{3,1})} := B^{2}_{i(l_{2,2}), i(l_{1,2})} \vee C^{2}_{i(l_{2,2}), i(l_{3,1})}$

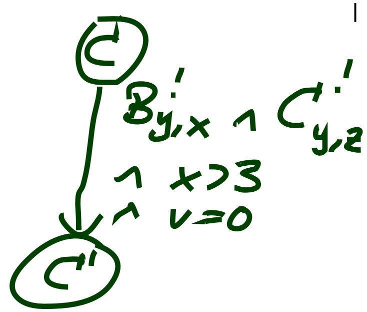
strict: add $\neg(\bigvee Msg)$
(no message from LSC is allowed)

Progress





- When do we move from q to q'?
- **Intuition**: those $(\sigma_i, cons_i, Snd_i)$ fire the progress transition (q, ψ, q') for which there exists a firedset F such that $q \rightsquigarrow_F q'$ and
  - $cons_i \cup Snd_i$ comprises exactly the messages that distinguish F from other firedsets of q (**weak mode**), and in addition no message occurring in the LSC is in $cons_i \cup Snd_i$ (**strict mode**),
  - σ_i satisfies the local invariants and conditions relevant at q'.



- **Formally**: Let $F_0, F_1, \dots, F_n$ be the firedset of q and $q \rightsquigarrow_F q'$ (unique).
  - $\psi := \underbrace{\bigwedge_F B(F_0)}_{\text{the msgs. in firedset } F_0} \wedge \underbrace{\neg(\bigvee_F (B(F_1) \cup \cdots \cup B(F_n)) \setminus B(F_0))}_{\text{and no other firedset}} \wedge \underbrace{\bigwedge \psi(q, q')}_{\text{respect conditions and loc. invariants relevant at } q'}$,

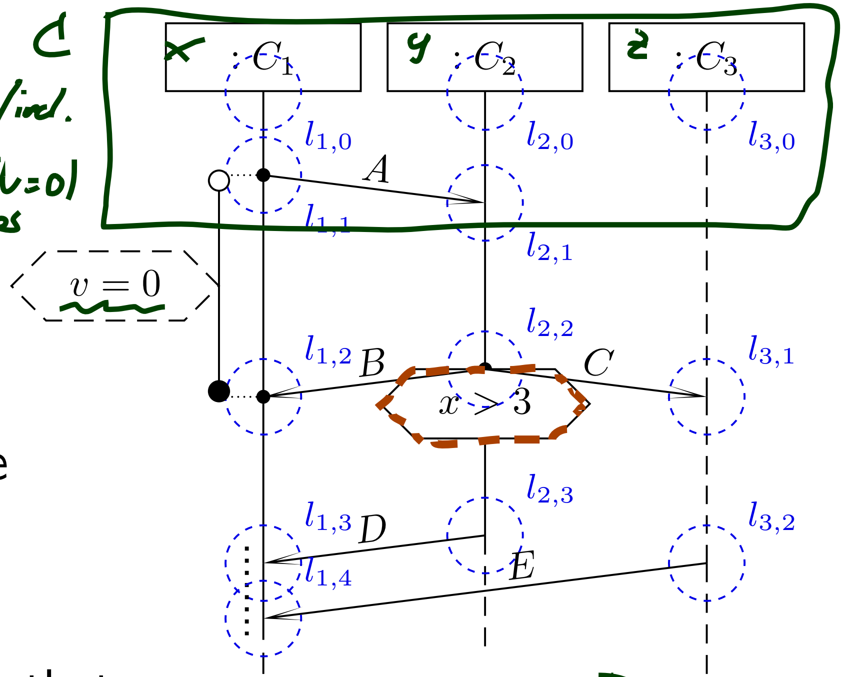
weak mode

Legal Exits

$w = (\sigma_0, cons_0, Snd_0)$
$(\sigma_1, cons_1, cons_1)$
$\vdots$

A!
$\vdots$
B?

C
excl./incl.
loc. inv ($v=0$) applies



- When do we take a legal exit from q?
- **Intuition**: those $(\sigma_i, cons_i, Snd_i)$ fire the legal exit transition $(q, \psi, \mathscr{L})$ for which there exists a firedset F and some q' such that $q \rightsquigarrow_F q'$ and
  - $cons_i \cup Snd_i$ comprises exactly the messages that distinguish F from other firedsets of q (**weak mode**), and in addition no message occurring in the LSC is in $cons_i \cup Snd_i$ (**strict mode**).
  - σ_i does not satisfy one cold constraint (condition or loc. inv)

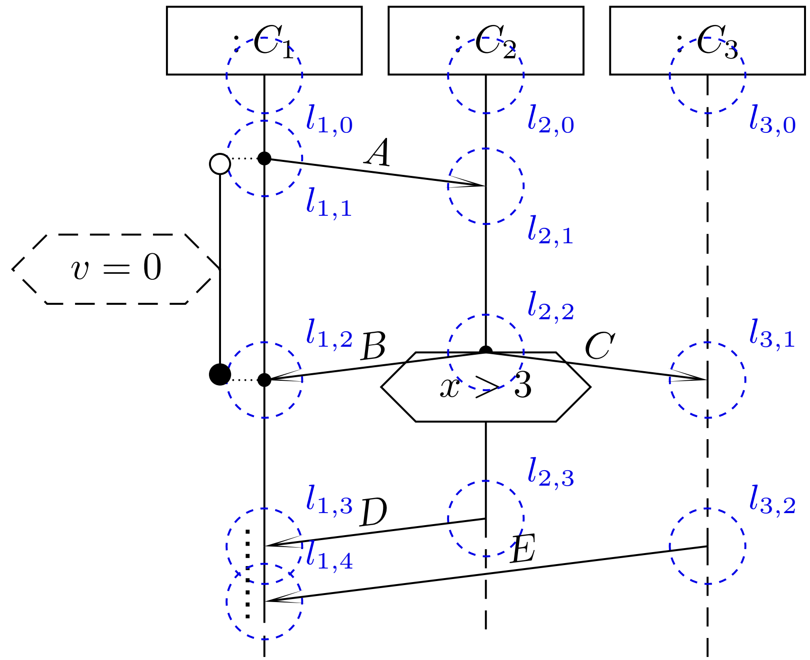
$B!_{x,x} \wedge C!_{3,2} \wedge (v \neq 0 \vee x \leq 3)$

q

true

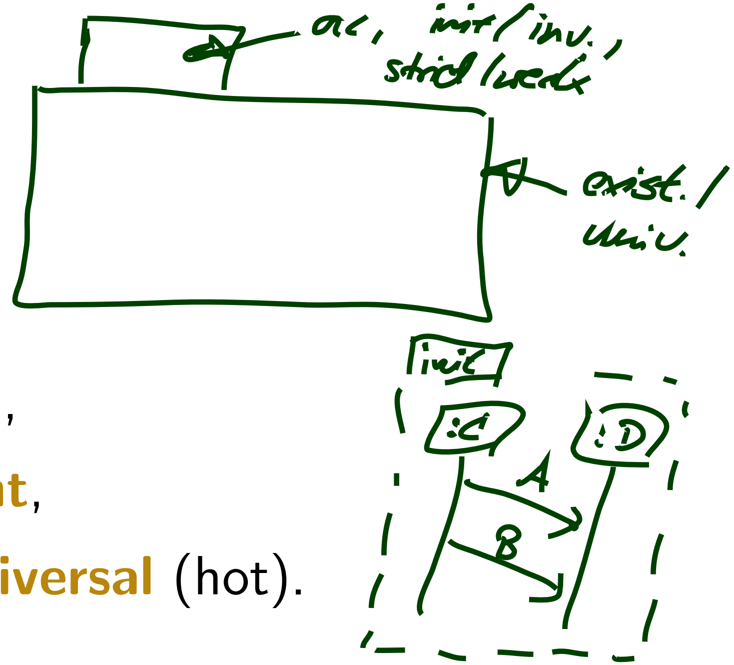
- **Formally**: Let $F_1, \dots, F_n$ be the firedset of q with $q \rightsquigarrow_{F_i} q_i'$.
  - $\psi := \underbrace{\bigvee_{i=1}^{n} \bigwedge B(F_i) \wedge \neg(\bigvee(B(F_1) \cup \dots \cup B(F_n)) \setminus B(F_i))}_{\text{we could move from } q \text{ to } q' \text{ with firedset } F_i} \wedge \underbrace{\bigvee \psi_{\mathsf{cold}}(q, q_i')}_{\text{but also cold constraint doesn't hold}}$,

Example



$:C_1$
$:C_2$
$:C_3$
$l_{1,0}$
$l_{2,0}$
$l_{3,0}$
A
$l_{1,1}$
$l_{2,1}$
$v = 0$
$l_{2,2}$
$l_{1,2}$
B
C
$l_{3,1}$
$x > 3$
$l_{2,3}$
$l_{1,3}$
D
$l_{3,2}$
E
$l_{1,4}$

Finally: The LSC Semantics



A **full LSC** L consist of

- a **body** $(I, (\mathscr{L}, \preceq), \sim, \mathscr{S}, \mathsf{Msg}, \mathsf{Cond}, \mathsf{LocInv})$,
- an **activation condition** (here: event) $ac \in B$,
- an **activation mode**, either **initial** or **invariant**,
- a **chart mode**, either **existential** (cold) or **universal** (hot).

A set W of ~~timed~~ words over ~~B and V~~ **satisfies** L, denoted $W \models L$, iff L

- **universal** (= hot), **initial**, and

$$\forall\, w \in W\ \forall\, \beta : X \to \mathrm{dom}(w_0) \bullet w \text{ activates } L \implies w \in \mathcal{L}(\mathcal{B}_L).$$

- **universal** (= hot), **invariant**, and

$$\forall\, w \in W\ \forall\, k \in \mathbb{N}_0\ \forall\, \beta : X \to \mathrm{dom}(w_k) \bullet w/k \text{ activates } L \implies w/k \in \mathcal{L}(\mathcal{B}_L).$$

- **existential** (= cold), **initial**, and

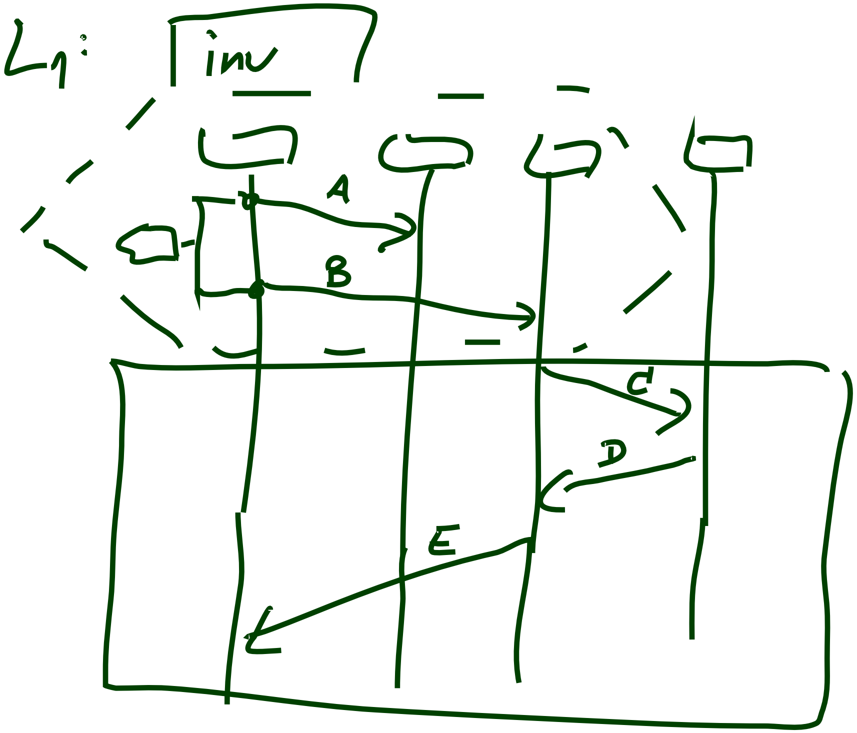
$$\exists\, w \in W\ \exists\, \beta : X \to \mathrm{dom}(w_0) \bullet w \text{ activates } L \wedge w \in \mathcal{L}(\mathcal{B}_L).$$

- **existential** (= cold), **invariant**, and

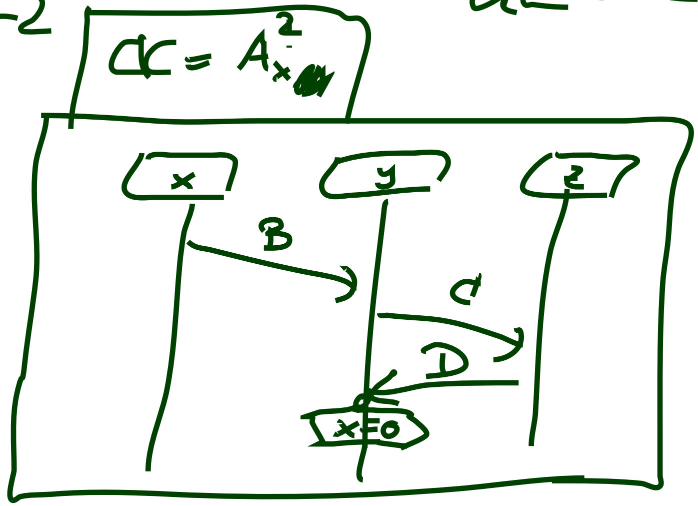
$$\exists\, w \in W\ \exists\, k \in \mathbb{N}_0\ \exists\, \beta : X \to \mathrm{dom}(w_k) \bullet w/k \text{ activates } L \wedge w/k \in \mathcal{L}(\mathcal{B}_L).$$

suffix of w including k-th element

L_1:



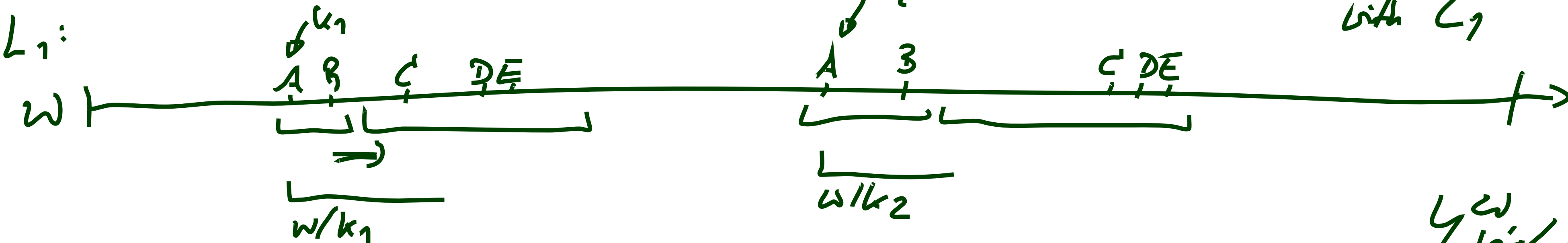
L_2



$ac := E^1_{x,y} \mid E^?_x$

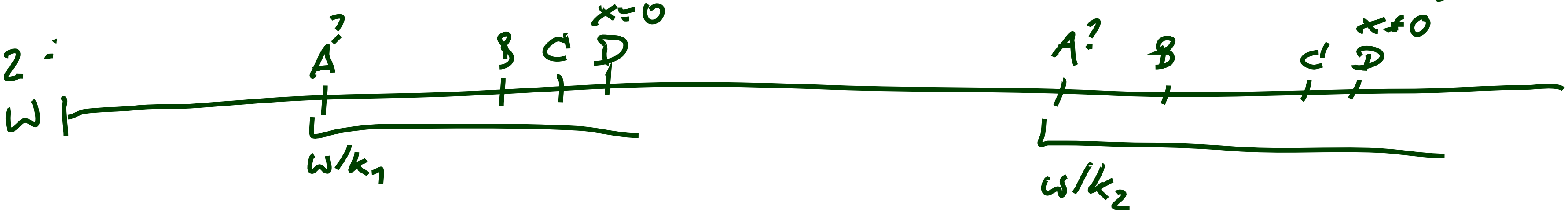
w consistent with L_1

L_1:



w violates L_2

L_2:



Back to UML: Interactions

Interactions as Reflective Description

- In UML, reflective (temporal) descriptions are subsumed by **interactions**.
- A UML model $\mathcal{M} = (\mathscr{CD}, \mathscr{SM}, \mathscr{OD}, \mathscr{I})$ has a set of interactions $\mathscr{I}$.
- An interaction $\mathcal{I} \in \mathscr{I}$ can be (OMG claim: equivalently) **diagrammed** as
  - **sequence diagram**, **timing diagram**, or
  - **communication diagram** (formerly known as collaboration diagram).

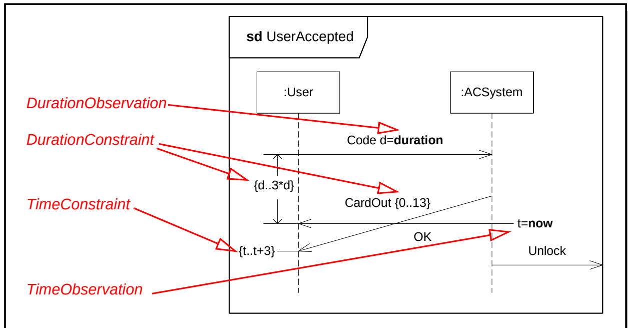


Figure 14.26 - Sequence Diagram with time and timing concepts [OMG, 2007b, 513]

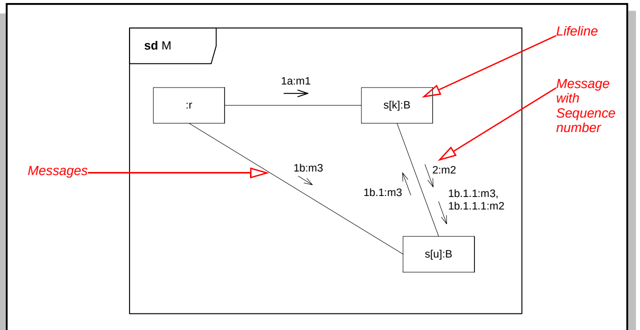


Figure 14.27 - Communication diagram [OMG, 2007b, 515]

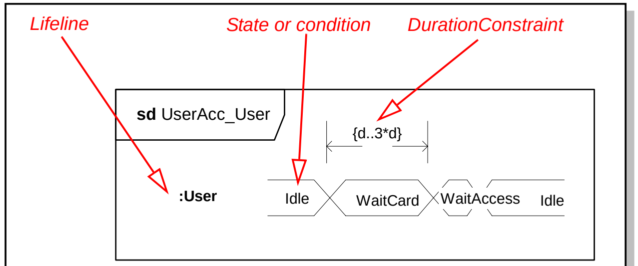


Figure 14.30 - Compact Lifeline with States [OMG, 2007b, 522]

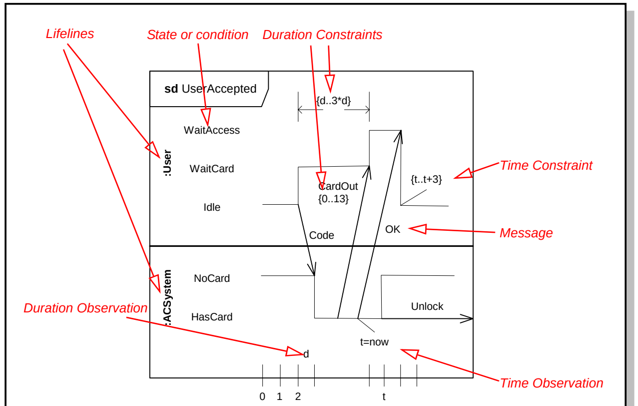
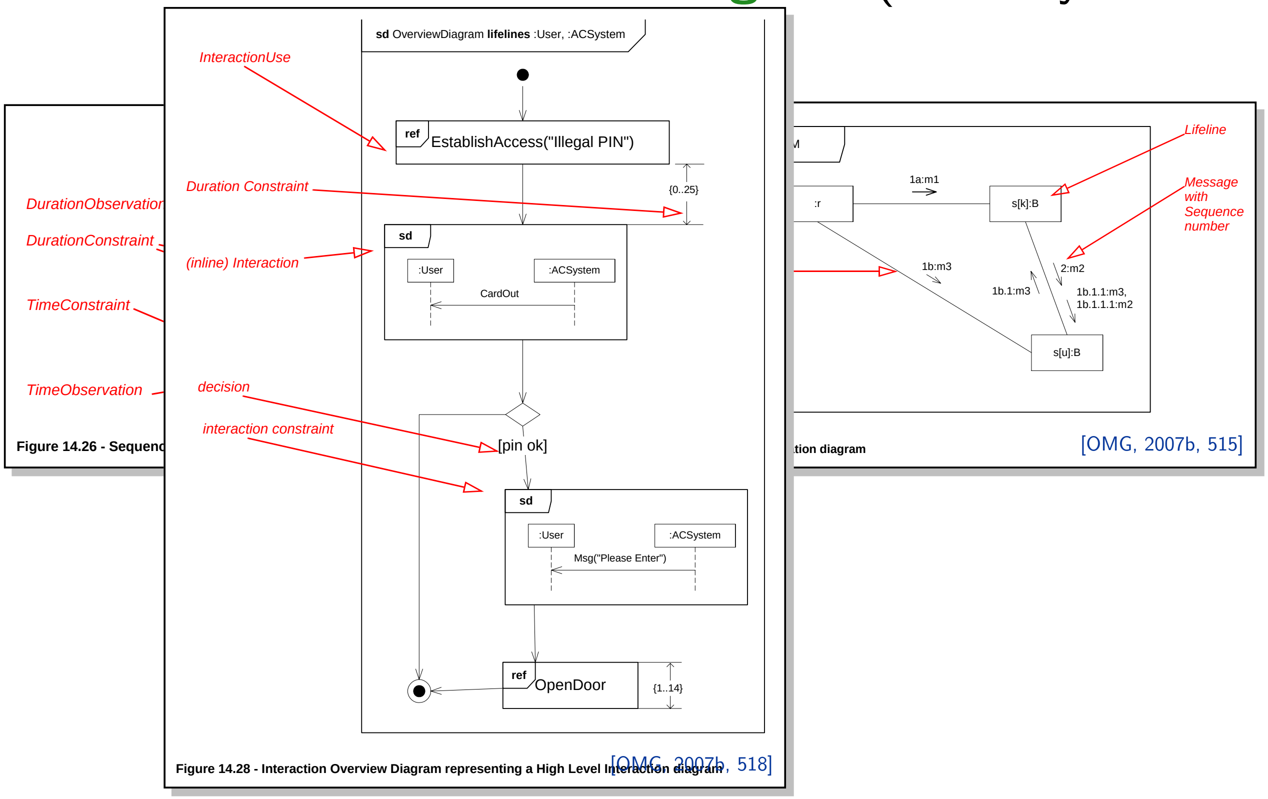
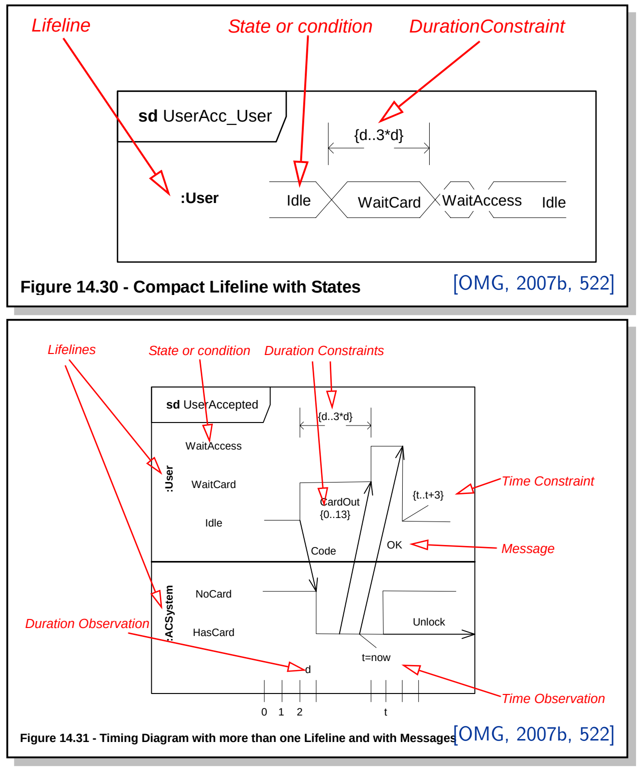


Figure 14.31 - Timing Diagram with more than one Lifeline and with Messages [OMG, 2007b, 522]

Interactions as Reflective Description

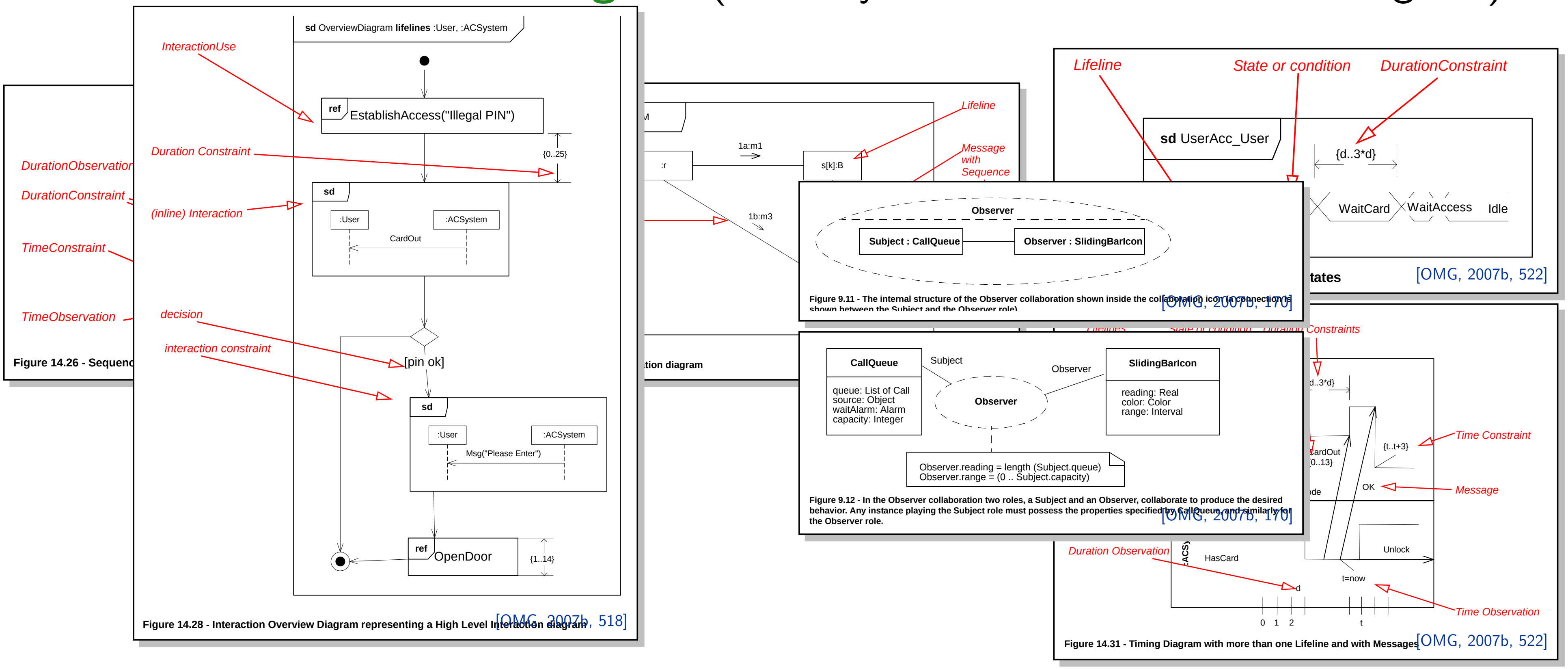
- In UML, reflective (temporal) descriptions are subsumed by **interactions**.
- A UML model $\mathcal{M} = (\mathscr{CD}, \mathscr{SM}, \mathscr{OD}, \mathscr{I})$ has a set of interactions $\mathscr{I}$.
- An interaction $\mathcal{I} \in \mathscr{I}$ can be (OMG claim: equivalently) **diagrammed** as
  - **sequence diagram**, **timing diagram**, or
  - **communication diagram** (formerly known as collaboration diagram).





Interactions as Reflective Description

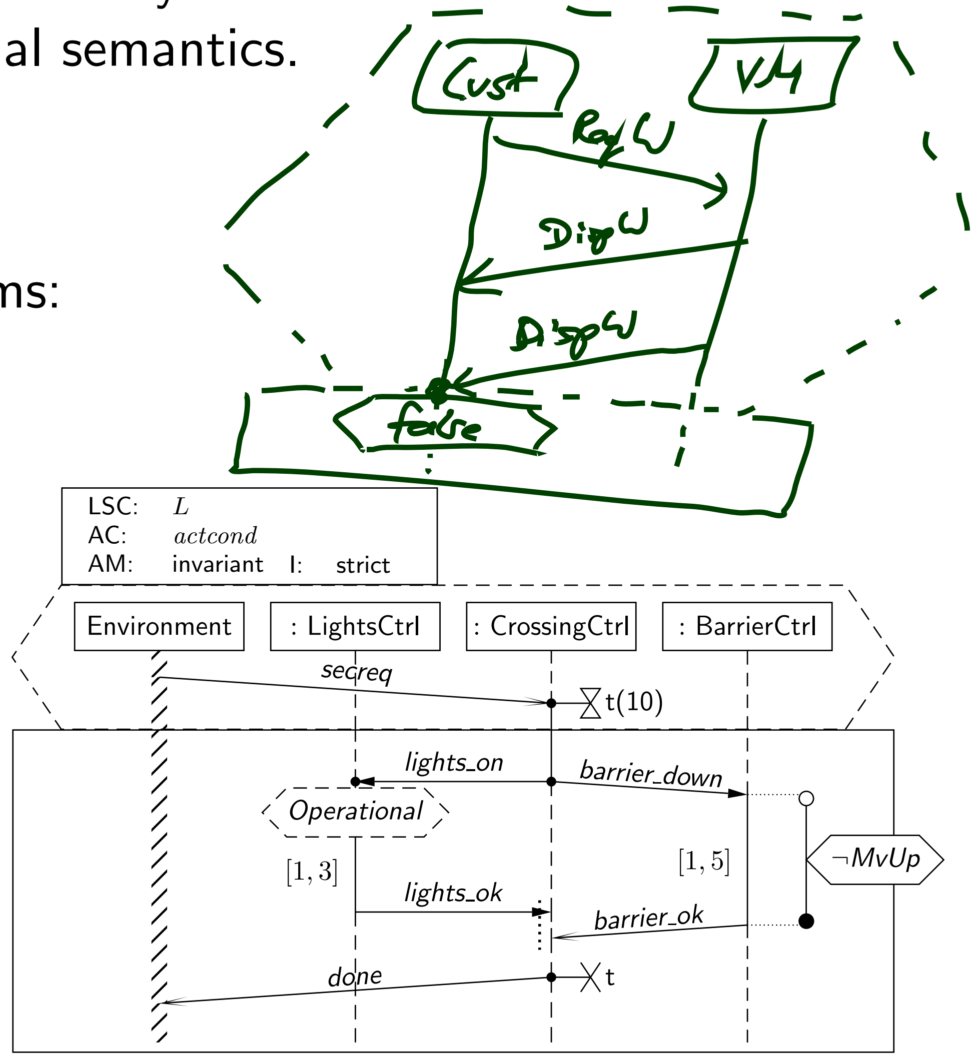
- In UML, reflective (temporal) descriptions are subsumed by **interactions**.
- A UML model $\mathcal{M} = (\mathscr{CD}, \mathscr{SM}, \mathscr{OD}, \mathscr{I})$ has a set of interactions $\mathscr{I}$.
- An interaction $\mathcal{I} \in \mathscr{I}$ can be (OMG claim: equivalently) **diagrammed** as
  - **sequence diagram**, **timing diagram**, or
  - **communication diagram** (formerly known as collaboration diagram).



Why Sequence Diagrams?

Most Prominent: Sequence Diagrams — with **long history**:

- **Message Sequence Charts**, standardized by the ITU in different versions, often accused to lack a formal semantics.
- **Sequence Diagrams** of UML 1.x



Most severe **drawbacks** of these formalisms:

- unclear **interpretation**:
  example scenario or invariant?
- unclear **activation**:
  what triggers the requirement?
- unclear **progress** requirement:
  must all messages be observed?
- **conditions** merely comments
- no means to express
  forbidden scenarios

Thus: Live Sequence Charts

- **SDs of UML 2.x** address **some** issues, yet the standard exhibits unclarities and even contradictions [Harel and Maoz, 2007, Störrle, 2003]
- For the lecture, we consider **Live Sequence Charts** (LSCs) [Damm and Harel, 2001, Klose, 2003, Harel and Marelly, 2003], who have a common fragment with UML 2.x SDs [Harel and Maoz, 2007]
- **Modelling guideline**: stick to that fragment.

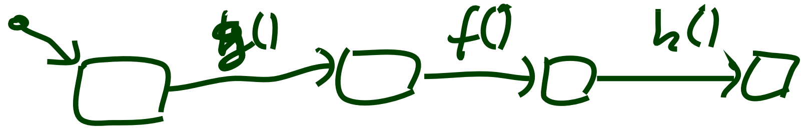
Side Note: Protocol Statemachines

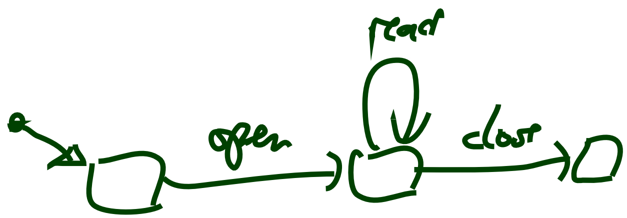
Same direction: **call orders** on operations

- **"for each C instance, method $f()$ shall only be called after $g()$ but before $h()$"**

Can be formalised with protocol state machines.

PSM:





References

References

[Damm and Harel, 2001] Damm, W. and Harel, D. (2001). LSCs: Breathing life into Message Sequence Charts. *Formal Methods in System Design*, 19(1):45–80.

[Harel and Maoz, 2007] Harel, D. and Maoz, S. (2007). Assert and negate revisited: Modal semantics for UML sequence diagrams. *Software and System Modeling (SoSyM)*. To appear. (Early version in SCESM'06, 2006, pp. 13-20).

[Harel and Marelly, 2003] Harel, D. and Marelly, R. (2003). *Come, Let's Play: Scenario-Based Programming Using LSCs and the Play-Engine*. Springer-Verlag.

[Klose, 2003] Klose, J. (2003). *LSCs: A Graphical Formalism for the Specification of Communication Behavior*. PhD thesis, Carl von Ossietzky Universität Oldenburg.

[OMG, 2007a] OMG (2007a). Unified modeling language: Infrastructure, version 2.1.2. Technical Report formal/07-11-04.

[OMG, 2007b] OMG (2007b). Unified modeling language: Superstructure, version 2.1.2. Technical Report formal/07-11-02.

[Störrle, 2003] Störrle, H. (2003). Assert, negate and refinement in UML-2 interactions. In Jürjens, J., Rumpe, B., France, R., and Fernandez, E. B., editors, *CSDUML 2003*, number TUM-I0323. Technische Universität München.