DELIBERATIVE ARCHITECTURE: ACCOUNTING FOR OBSERVATIONS
Recap

- Domain description: Define a transition function $\Phi$ from actions and states to sets of states.
  - Effect of actions: $a$ \textit{causes} $f$ if $p_1, \ldots, p_m$
  - Causality: $p_1, \ldots, p_n$ \textit{causes} $f$
  - Executability conditions: executable $a$ if $q_1, \ldots, q_r$.
  - $\Phi(a, s) = \emptyset$ means $a$ is not executable in the state $s$.

- Observations: $f$ \textit{after} $a_1, \ldots, a_n$.
  - These observations (except of the kind \textit{initially} $f$) are actually hypothetical statements.
  - Together with a domain description (or a $\Phi$) they define possible initial states.
  - Real observations vs hypothetical statements (or oracles)
    * $a$ \textit{causes} $f$; $b$ \textit{causes} $g$; $f$ \textit{after} $a$; $g$ \textit{after} $b$ is consistent.
    * But only one of $a$ and $b$ happened in the initial state!
A language for observations

• Motivation:
  – Initially the car was parked. The next day the car was gone. Conclusion: The car must have been stolen.
  – I am at home, i have a car, and my suitcase is unpacked. I need to go to the airport with a packed suitcase. I make the plan to pack and then drive to the airport. After packing I observe that my car is no longer there. I can not just execute the remaining part of my old plan. I need to make a new plan from the current situation I am in.
  – I left home and my light switch was working. I came back and turned on the light and it does not work. I need to diagnose what went wrong; but not just a surface level diagnosis about what component is faulty but also a deeper diagnosis explaining what actions caused the fault.

• Syntax of the language:
  – $f$ at $t$
  – $t_1$ precedes $t_2$
\(- \alpha \text{ occurs}_\text{at} \ a\)
\(- \alpha \text{ between} \ t_1, t_2\)
\(- t \) and \(t_i\);s are situations (or time points), \(\alpha\) is a sequence of actions.

• Example 1:
  \(- O = \{\text{has}_{-}\text{car at} \ t_0; \neg \text{has}_{-}\text{car at} \ t_1; t_0 \text{ proceeds} \ t_1\}\)
  \(- D = \{\text{steal}_{-}\text{car causes} \ \neg \text{has}_{-}\text{car}\}\)
  \(- \) From \((D, O)\) we want to conclude that \text{steal}_{-}\text{car} \text{ between} \ t_0, t_1.\)

• Example 2:
  \(- O_1 = \{\text{at}_{-}\text{home at} \ t_0; \text{has}_{-}\text{car at} \ t_0; \neg \text{packed at} \ t_0; \}\)
  \(- D_1 = \{\text{pack causes} \ \text{packed}; \ \text{executable} \text{ pack if} \ \text{at}_{-}\text{home}; \text{steal}_{-}\text{car causes} \ \neg \text{has}_{-}\text{car}; \text{rent}_{-}\text{car causes} \ \text{has}_{-}\text{car}; \text{drive causes} \ \text{at}_{-}\text{airport}; \ \text{executable} \text{ drive if} \ \text{has}_{-}\text{car}; \text{at}_{-}\text{home causes} \ \neg \text{at}_{-}\text{airport}; \ \text{at}_{-}\text{airport causes} \ \neg \text{at}_{-}\text{home}\}\)
  \(- \) From \((D_1, O_1)\) we can conclude that \text{at}_{-}\text{airport}, \text{packed after} \ \text{pack, drive.}\)
  \(- \) I do the packing.
  \(- O_2 = O_1 \cup \{\text{pack occurs}_\text{at} \ t_1; t_0 \text{ proceeds} \ t_1\}.\)
But then I notice that my car is gone. 
\[ O_3 = O_2 ∪ \{¬\text{has\_car at } t_2; t_1 \text{ proceeds } t_2 \} \]

- From \((D_1, O_3)\) I can no longer conclude that \(\text{at\_airport}, \text{packed after drive}\).
- But from \((D_1, O_3)\) I should be able to conclude that \(\text{at\_airport}, \text{packed after rent \_car; drive}\).

- Query language: The language of observations plus
  \[ f \text{ after } a_1, \ldots, a_n \text{ at } t. \]
  - If \(t = t_C\), the current situation then we just write
    \[ f \text{ after } a_1, \ldots, a_n \]
  - If \(n = 0\) then we write \(f \text{ at } t\).
  - If \(n = 0\) and \(t = t_C\) then we write \(\text{currently } f\).
  - If \(n = 0\) and \(t = t_0\) then we write \(\text{initially } f\).

- \(t_C\) is a special constant and observations are not allowed with respect to this constant.
- A pair \((D, O)\) will now be referred to as a narrative.
Semantics of narratives

- **Defining Φ**
  - An action \( a \) is said to be possibly executable in a state \( s \) w.r.t. a \( D \) if there exists **executable** \( a \) if \( q_1, \ldots, q_r \) in \( D \) such that \( q_1, \ldots, q_r \) hold in \( s \).
  - \( Φ(a, s) = \emptyset \) if \( a \) is not possibly executable in \( s \).
  - Else, \( Φ(a, s) = \{ s' : s' = Cn_R((s \cap s') \cup E_a(s)) \} \) and \( s' \) is an interpretation.

- **Valid states (w.r.t. \( D \))**: States that are closed under the static causal rules in \( D \).

- **Semantic is defined in terms of a trajectory \( τ \) and a situation assignment \( Σ \).**

- **A trajectory \( τ \) of a domain description \( D \) is a sequence of the form**
  
  \[ s_0, a_0, s_1, a_1, \ldots, a_{n-1}, s_n \]

  such that \( s_{i+1} \in Φ(a_i, s_i) \), for \( 0 \leq i \leq n - 1 \). By the action sequence of \( τ \) we refer to the sequence \( a_0, \ldots, a_{n-1} \).

  Intuitively \( τ \) selects one of the possible trajectories defined by \( Φ \) as a possible evolution of the world.

- **A situation assignment with respect to \( D \) is a mapping \( Σ \) from situations into the set of action sequences of \( D \) that satisfy the following properties:**
- $\Sigma(t_0) = []$
- for every $t \in S$, $\Sigma(t)$ is a prefix of $\Sigma(t_C)$.

- For a sequence of actions $a_0, \ldots, a_m$, we say its length is $m$.

- An interpretation $M$ of $(D, O)$ is a pair $(\tau, \Sigma)$, where $\tau$ is a trajectory of $D$, $\Sigma$ is a situation assignment, and $\Sigma(t_C)$ is the action sequence in $\tau$.

- For an interpretation $M = (\tau, \Sigma)$ of $(D, O)$, where $\tau = s_0, a_0, s_1, a_1, \ldots, a_{n-1}, s_n$:
  - $\alpha$ occurs at $t$ is true in $M$ if the sequence $\Sigma(t) \circ \alpha$ is a prefix of $\Sigma(t_C)$;
  - $\alpha$ between $t_1, t_2$ is true in $M$ if $\Sigma(t_1) \circ \alpha = \Sigma(t_2)$;
  - $f$ at $t$ is true in $M$ if $f$ holds in $s_{m+1}$, where $m$ the length of $\Sigma(t)$;
  - $t_1$ precedes $t_2$ is true in $M$ if $\Sigma(t_1)$ is a prefix of $\Sigma(t_2)$.

- An interpretation $M = (\tau, \Sigma)$ is a model of a narrative $(D, O)$ if:
  
  (i) observations in $O$ are true in $M$;
  (ii) there is no other interpretation $M' = (\tau', \Sigma')$ such that $M'$ satisfies condition (i) above and $\tau'$ is a subsequence of $\tau$.

Observe that these models are minimal in the sense that they exclude extraneous actions.
For a model $M = (\tau, \Sigma)$, $\Sigma(t_C)$ is referred to as the *actual line* w.r.t. $M$. It encodes how the world has progressed till the current situation.

- A narrative is *consistent* if it has a model. Otherwise, it is *inconsistent*.
- A query $q$ of the form $\varphi$ after $a_1, \ldots, a_n$ at $t$ is entailed by a narrative $(D, O)$, denoted by $(D, O) \models q$, if $\varphi$ is true in all states $s'_n$, where $M = (\tau, \Sigma)$ is a model of $(D, O)$, $\Sigma(t) = b_1, \ldots, b_m$, the sequence $s_0, b_1, s_1, \ldots, b_m, s_{m+1}$ is a prefix of $\tau$, and $s_0, b_1, s_1, \ldots, b_m, s_{m+1}, a_1, s'_1, a_2, s'_2, \ldots, a_n, s'_n$ is a trajectory of $(D, O)$.
- For other kind of queries (i.e., with syntax same as observations):
  - We already defined when they are true in a model.
  - We say such a query $q$ is entailed by a narrative $(D, O)$, denoted by $(D, O) \models q$, if $q$ is true in every model of $(D, O)$. 
The stolen car example revisited

- \( D = \{ \text{steal} \_\text{car causes } \neg \text{has} \_\text{car} \} \)
- \( O = \{ \text{has} \_\text{car at } t_0; \neg \text{has} \_\text{car at } t_1; t_0 \text{ precedes } t_1 \} \)

- The semantics:
  - States: \( \emptyset, \{ \text{has} \_\text{car} \} \).
  - \( \Phi(\text{steal} \_\text{car}, \emptyset) = \emptyset; \Phi(\text{steal} \_\text{car}, \{ \text{has} \_\text{car} \}) = \emptyset. \)
  - \( \tau = \{ \text{has} \_\text{car} \}, \text{steal} \_\text{car}, \emptyset. \)
  - \( \Sigma(t_0) = []; \Sigma(t_1) = [\text{steal} \_\text{car}]; \Sigma(t_C) = [\text{steal} \_\text{car}]. \)
  - \((D, O)\) has only one model \( M = (\tau, \Sigma) \).
  - Why is \((\tau', \Sigma')\) not a model?
    (where \( \tau' = \{ \text{has} \_\text{car} \}, \text{steal} \_\text{car}, \emptyset, \text{steal} \_\text{car}, \emptyset, \) and
    \( \Sigma'(t_0) = []; \Sigma'(t_1) = [\text{steal} \_\text{car}, \text{steal} \_\text{car}]; \Sigma'(t_C) = [\text{steal} \_\text{car}, \text{steal} \_\text{car}]. \))

- \( D, O \models \text{steal} \_\text{car between } t_0, t_1 \)
  because \( \text{steal} \_\text{car between } t_0, t_1 \) is true in the only model \( M \) of \((D, O)\)
  because \( \Sigma(t_1) \circ \text{steal} \_\text{car} = \Sigma(t_0) \).
An example with multiple models.

- \( O = \{\neg f, \neg g \text{ at } t_0; \ f \text{ at } t_1; \ t_0 \text{ precedes } t_1; \ a_4 \text{ occurs at } t_1\} \)
- \( D = a_1 \text{ causes } g; \ a_2 \text{ causes } f \text{ if } g; \ a_3 \text{ causes } g; \ a_4 \text{ causes } \neg g \text{ if } f\) 
- What are the models of \((D, O)\)? What are the actual lines? Does \((D, O) \models \text{ currently } \neg g\)?

<table>
<thead>
<tr>
<th>actions</th>
<th>(a_1)</th>
<th>(a_2)</th>
<th>(a_3)</th>
<th>(a_4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>states</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(s_1 = {\neg f, \neg g})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(s_2 = {f, \neg g})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(s_3 = {\neg f, g})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(s_4 = {f, g})</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- \(\Phi\) is straightforward.
- \(\Sigma_1(t_0) = [], \Sigma_1(t_1) = [a_1, a_2], \Sigma_1(t_C) = [a_1, a_2, a_4]. \tau_1 = ?\)
- \(\Sigma_2(t_0) = [], \Sigma_2(t_1) = [a_1, a_2, a_3], \Sigma_1(t_C) = [a_1, a_2, a_3, a_4]. \tau_2 = ?\)
- \(\Sigma_3(t_0) = [], \Sigma_3(t_1) = [a_3, a_2], \Sigma_3(t_C) = [a_3, a_2, a_4]. \tau_3 = ?\)