High-level Vision

What are the tasks (is the scope) of high-level vision?

Vision as silent-movie understanding

- connecting to common-sense knowledge
- understanding goal-oriented behaviour
- vision in context

Vision and acting

- robot vision
- goal-oriented vision, attention control
- spatial and temporal reasoning

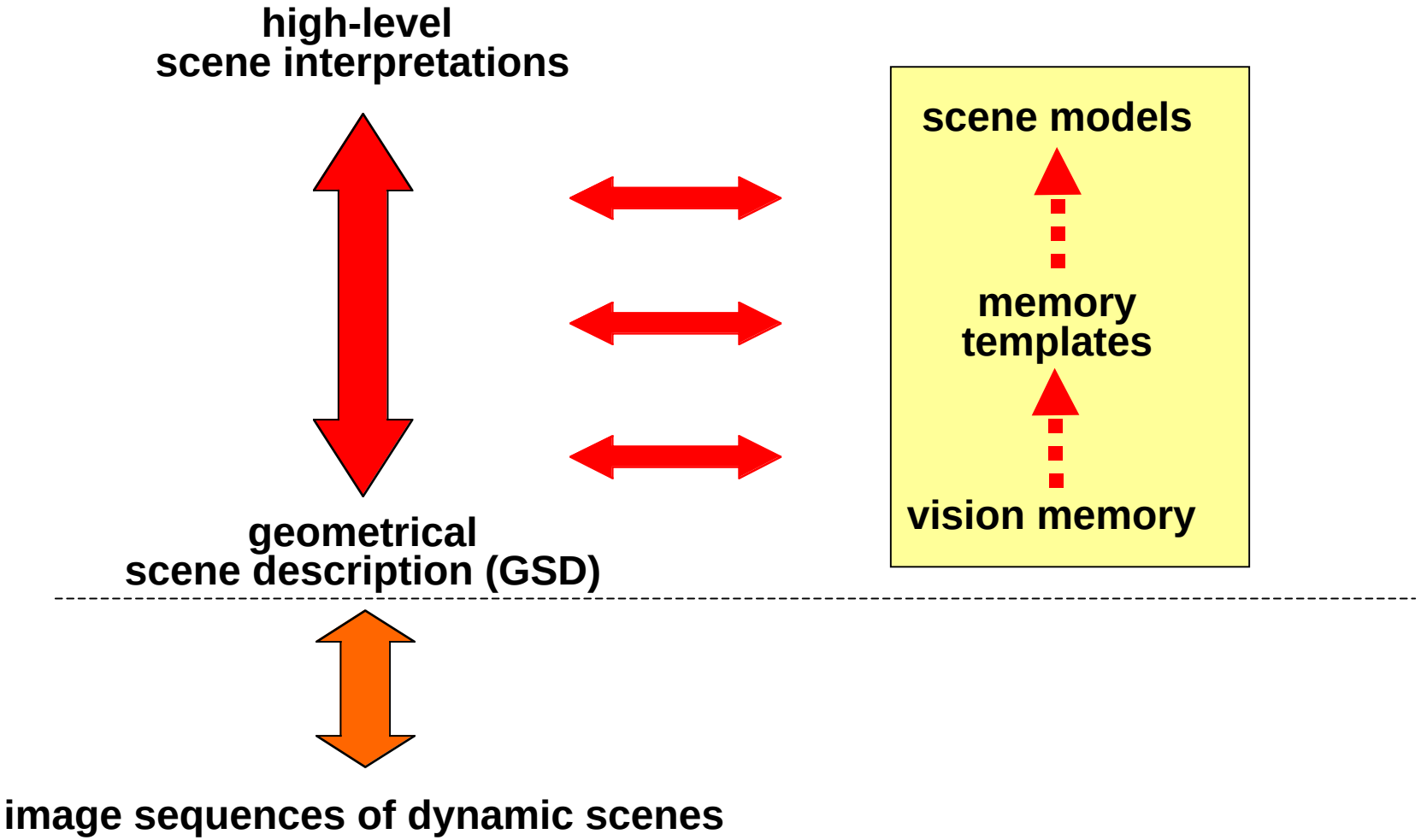
Vision and learning

- discovering reoccurring patterns
- building models
- predicting from experience

Topics of High-Level Vision

- **Representing and recognizing structures consisting of several spatially and temporally related components (e.g. object configurations, situations, occurrences, episodes)**
- **Exploiting high-level knowledge and reasoning for scene prediction**
- **Understanding purposeful behaviour (e.g. obstacle avoidance, grasping and moving objects, behaviour in street traffic)**
- **Mapping between quantitative and qualitative descriptions**
- **Natural-language communication about scenes**
- **Learning high-level concepts from experience**
- **Connecting uncertain knowledge with logic-based reasoning**

Basic Building Blocks for High-level Scene Interpretation



Basic Representational Units

scene	**spatially and temporally coherent real-world section**
geometrical scene description (GSD)	**scene description in terms of object locations in an image sequence**
scene interpretation	**scene description in terms of instantiated scene models (object configurations, occurrences, episodes, purposive actions)**
memory record	**memorized scene interpretation incl. imagery**
memory template	**generalized substructure of memory records**
scene model	**conceptual unit for scene interpretation**

Temporal Decomposition of Scenes

Temporal decomposition

- **by temporal segmentation:**
  constancies in time-dependent properties of an image sequence
- **by model matching:**
  occurrences which obey a model

Compare with spatial decomposition

- **by spatial segmentation:**
  image regions with spatially constant (uniform) properties
- **by model matching:**
  image regions which obey a model

Temporal Relations

Distinguish between relations based on

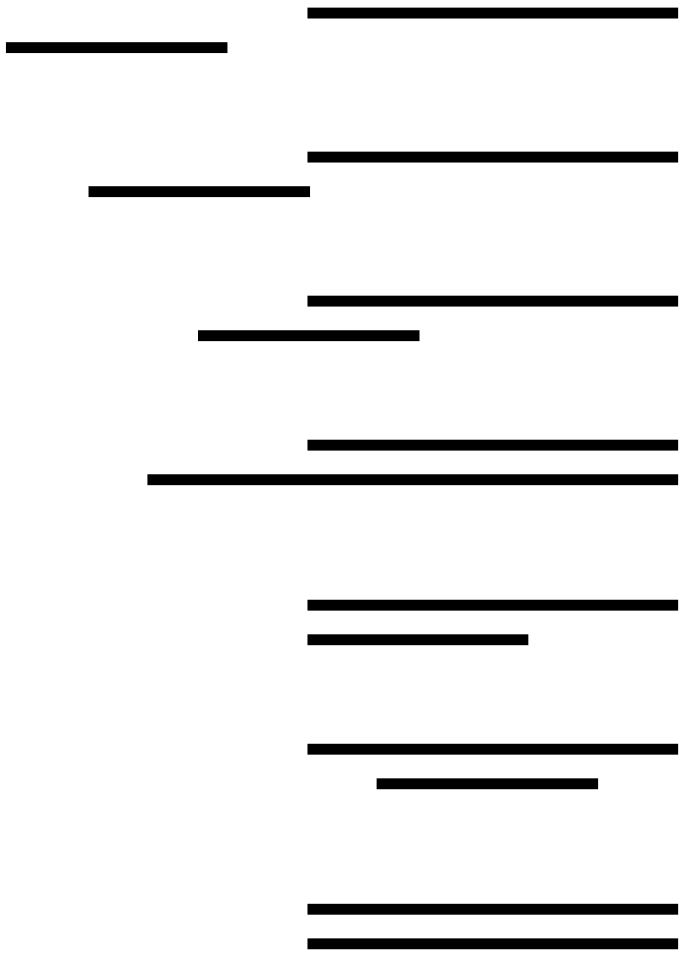
- ***time points***
  - discrete $T \in \{1, 2, 3, \ldots\}$
  - continuous $T \in \Re$
- ***time intervals*** **I_1 "during" I_2**

Distinguish between

- ***quantitative*** **T1 = T2 + 4**
- ***qualitative*** **T1 "after" T2**

relations

Interval Relations in Allen´s Algebra



BEFORE (I1, I2)	< >
MEETS (I1, I2)	m mi
OVERLAPS (I1, I2)	o oi
FINISHES (I1, I2)	f fi
STARTS (I1, I2)	s si
DURING (I1, I2)	d di
EQUAL (I1, I2)	=

Convex Time-point Algebra

Qualitative relations between time points which can be described by the inequality

$$T1 + c \leq T2$$

(T1, T2: time points; c: constant)

"Convex relation":

All intervals satisfying a convex relation can be generated by continuous displacements of the begin and end points of an interval

In Allen´s Algebra:

convex relation e.g. **d v m**

non-convex relation e.g. **b v bi**

Perceptual Primitives

What are basic attributes for the description and temporal segmentation of a time-varying scene?

Experiment: Natural-language traffic scene description (imagine the report of an accident witness)

> **"A white Golf approached the pedestrian crossing from the left. A pedestrian turned off the side walk and crossed the street about 2 meters behind the pedestrian crossing. A red BMW turned into the street from the right and flashed its lights. In the middle of the street the pedestrian stopped, turned around and waved to a woman on the side walk. The white golf braked but hit the pedestrian. The pedestrian flew through the air. The red BMW turned to the right to avoid the pedestrian and hit a tree. The woman laughed."**

The description is based on geometric and photometric attributes and their temporal derivatives:

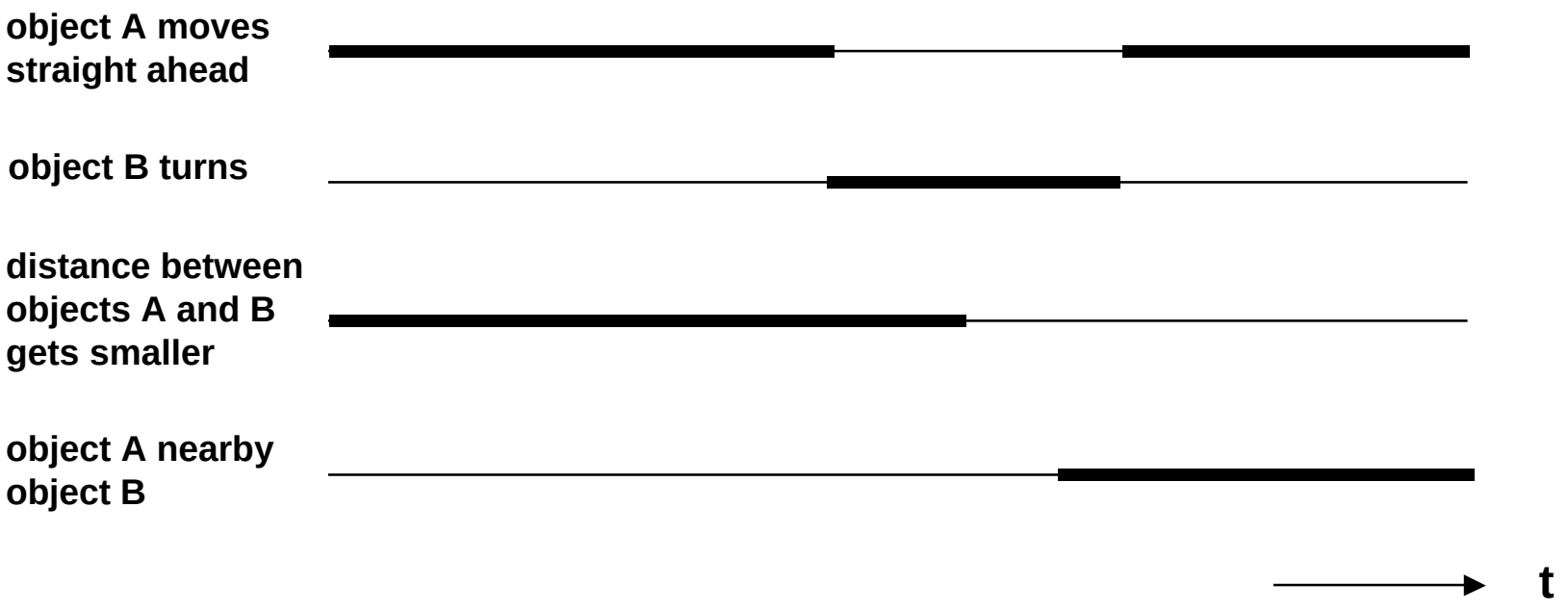
- distance, angle, shape, size
- brightness, colour

(+ change of distance, change of angle, change of shape, etc.)

Qualitative Predicates for Modelling and Recognizing Occurrences

Simple durative predicates applied to perceptual primitives:

- **constant value**
- **value within certain interval**
- **value smaller / larger than threshold**



Example: Criminal Act Recognition

Work by Somboon Hongeng, USC, USA

1. **Video is segmented into primitive occurrences**
2. **Complex occurrence is established if primitive occurrences meet temporal constraints**

Qualitative Predicates for Occurrences in Traffic Scenes

Results of project NAOS: "Natural-language description of object motions in traffic scenes"

exist
move
decelerate, accelerate
turn_left, turn_right
increasing_distance, reducing_distance
along, across
in_front_of, behind, beside
on, above, under, below
at, near_to
between
(and others)

Note that qualitative predicates are often (but must not be) part of natural language.

For technical applications one may use technical (artificial) qualitative predicates, e.g.

v50 (= 45 ≤ v ≤ 55 km/h)

shape_x (= shape_index ≤ 4.174)

Occurrence Models

Basic ingredients:

- relational structure
- taxonomy
- partonomy
- spatial relational language
- temporal relational language
- object appearance models

- An occurrence model describes a class of occurrences by
  - properties
  - sub-occurrences (= components of the occurrence)
  - relations between sub-occurrences

- A primitive occurrence model consists of
  - properties
  - a qualitative predicate

- Each occurrence has a begin and end time point

Occurrence Model for Overtaking in Street Traffic

Predicate:	**ueberholen**
	:is-a occurrence-model
	:local-name ue
Arguments:	**(?veh1 :is-a vehicle)**
	(?veh2 :is-a vehicle)
Time marks:	**(ue.B ue.E)**
Component events:	**(mv1 :is-a (move ?veh1 mv1.B mv1.E))**
	(mv2 :is-a (move ?veh2 mv2.B mv2.E))
	(bh :is-a (behind ?veh1 ?veh2 bh.B bh.E))
	(bs :is-a (beside ?veh1 ?veh2 bs.B bs.E))
	(bf :is-a (before ?veh1 ?veh2 bf.B bf.E))
	(ap :is-a (approach ?veh1 ?veh2 ap.B ap.E))
	(rc :is-a (recede ?veh1 ?veh2 rc.B rc.E))
Temporal relations:	**(ue.B = bh.B)**
	(ue.E = bf.E)
	(ap :during mv1)
	(ap :during mv2)
	(rc :during mv1)
	(rc :during mv2)
	(bh :overlaps bs)
	(bs :overlaps bf)
	(bh :during ap)
	(bf :during rc)

Occurrence Model for Transport Vehicle Behaviour

The occurrence model *transport-load* describes the regular unloading procedure of an automatic transport vehicle

Predicate:	**transport-load**
	:is-a occurrence-model
	:local-name tl
Arguments:	**(?dtv :is-a stacker)**
	(?rm :is-a room)
	(?stat :is-a station)
Time marks:	**(tl.B tl.E)**
Component events:	**(er :is-a (enter-room ?rm ?dtv er.B er.E))**
	(fs :is-a (free-station ?stat fs.B fs.E))
	(ul :is-a (unload ?dtv ?stat ul.B ul.E))
	(ex :is-a (exit-room ?rm ?dtv ex.B ex.E))
Temporal relations:	**(tl.B + 10 ≤ tl.E)**
	(tl.E - 12 ≤ tl.B)
	(er :before ul)
	(ul :before ex)
	(ul :starts-within fs)
	(tl.B = er.B)
	(tl.E 0 ex.B)

Occurrence Model for Placing a Cover

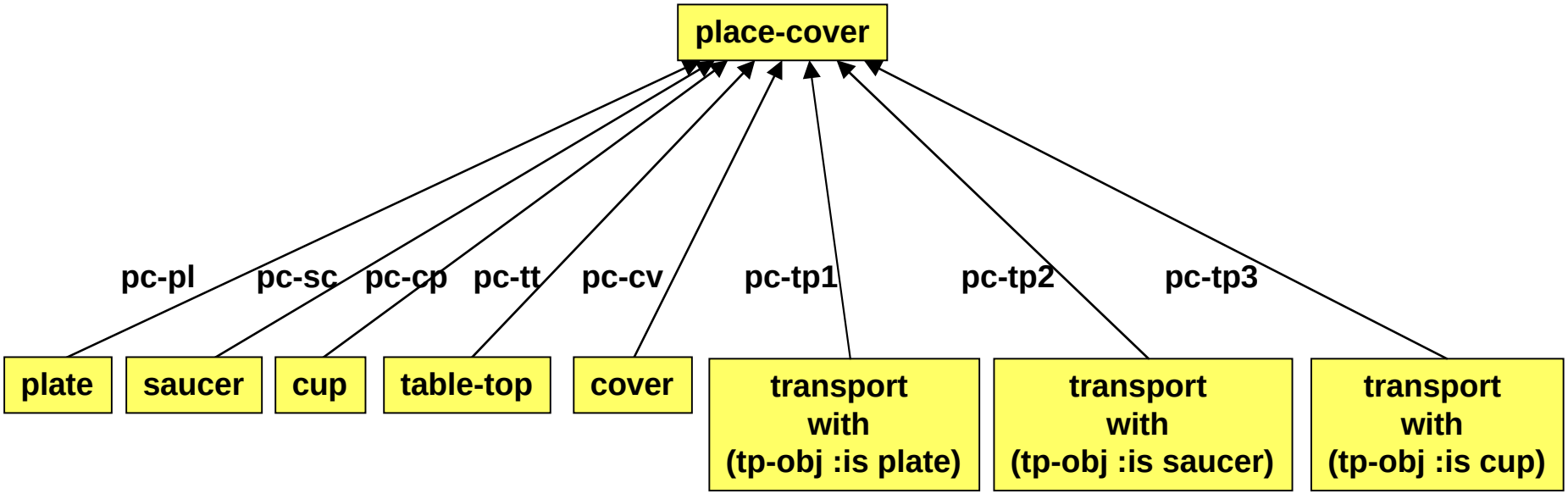
Table-laying scenario of project CogVis:

Stationary cameras observe living room scene and recognize meaningful occurrences, e.g. placing a cover onto the table.

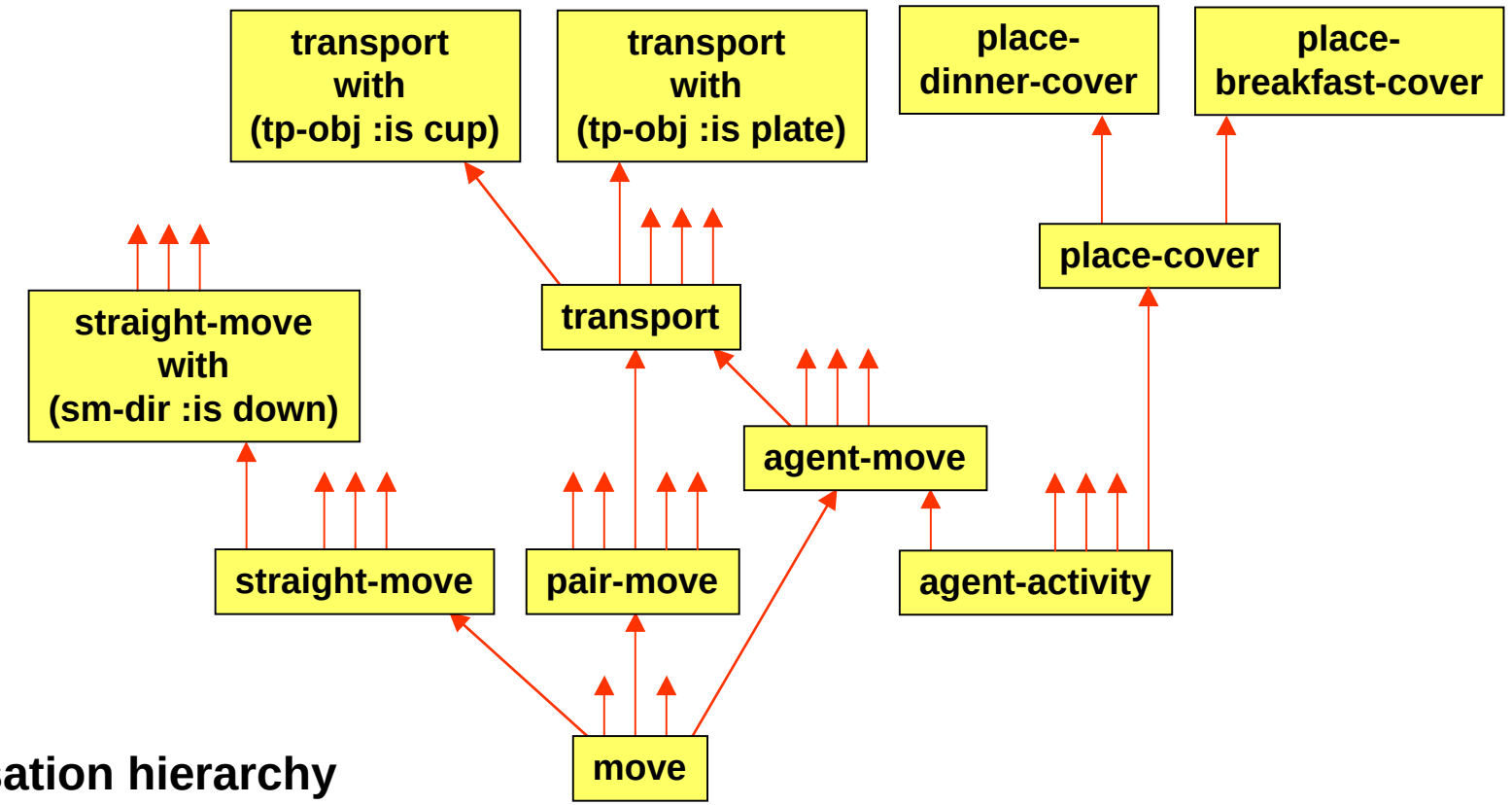
name:	**place-cover**
parents:	**:is-a agent-activity**
parts:	**pc-pl :is plate**
	pc-sc :is saucer
	pc-cp :is cup
	pc-tt :is table-top
	pc-tp1 :is (transport with (tp-obj :is plate))
	pc-tp2:is (transport with (tp-obj :is saucer))
	pc-tp3 :is (transport with (tp-obj :is cup))
	pc-cv :is cover
time marks:	**pc-tb, pc-te :is timepoint**
constraints:	**pc-tp1.tp-ob = pc-cv.cv-pl = pc-pl**
	pc-tp2.tp-ob = pc-cv.cv-sc = pc-sc
	pc-tp3.tp-ob = pc-cv.cv-cp = pc-cp
	pc-cv.cv-tb ≥ pc-tp1.tp-te
	pc-cv.cv-tb ≥ pc-tp2.tp-te
	pc-cv.cv-tb ≥ pc-tp3.tp-te
	pc-tp3.tp-te ≥ pc-tp2.tp-te
	pc-tb ≤ pc-tp1.tb
	pc-tb ≤ pc-tp2.tb
	pc-tb ≤ pc-tp3.tb
	pc-te ≥ pc-cv.cv-tb
	pc-te ≥ pc-tb + 80Δt

Parts Structure

- associational structure between aggregats and their parts
- probabilistic information may be added

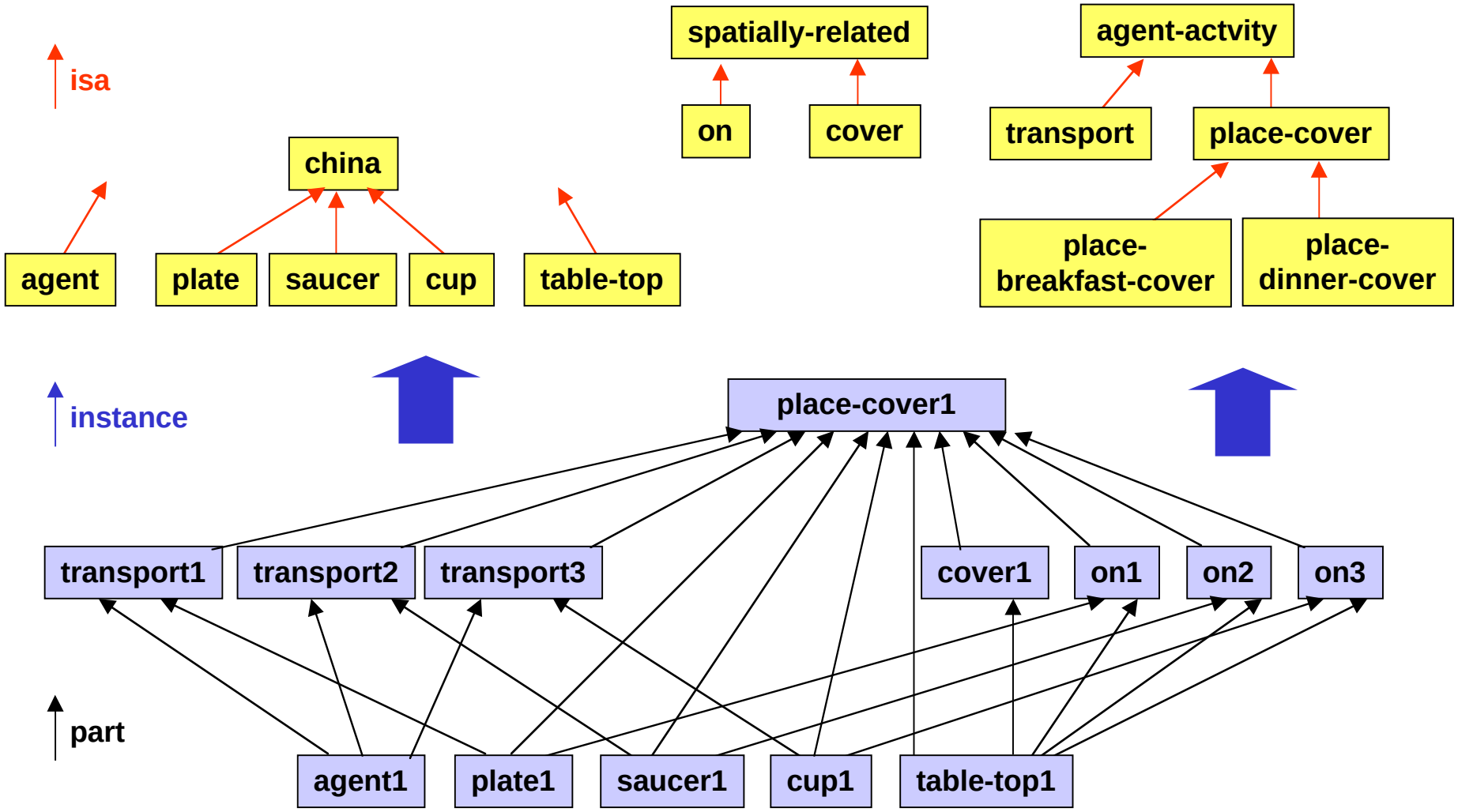


Concept Hierarchy

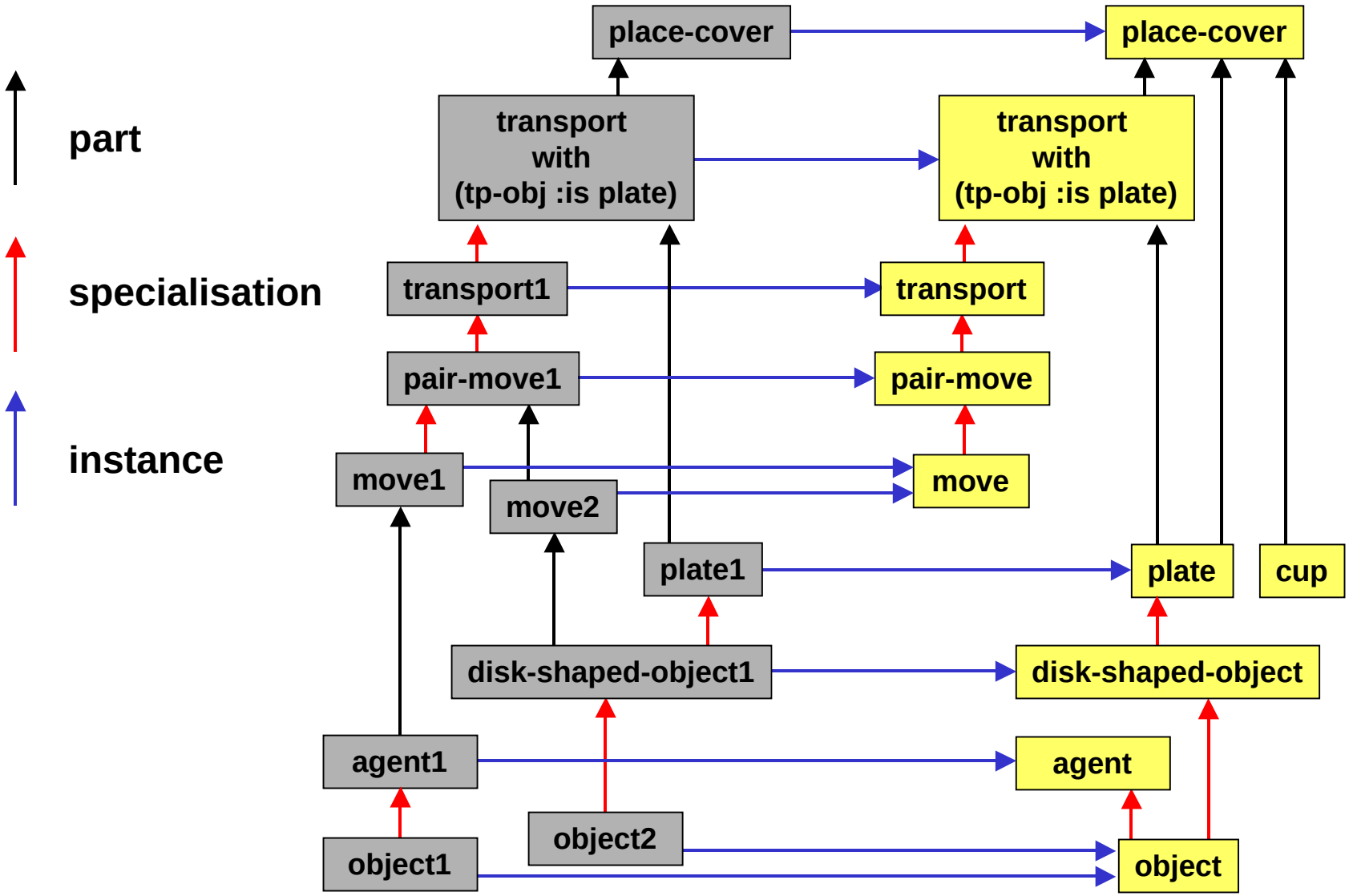


- specialisation hierarchy
- nodes are concept expressions
- multiple inheritance

Relational Structure for Placing-a-cover



Model-based Interpretation



Temporal Constraint Net for Convex Time-Point Algebra

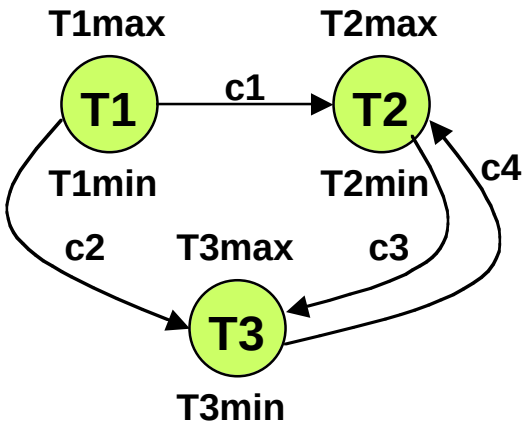
Unary temporal constraints: $T_{min} \leq T \leq T_{max}$

Binary temporal constraints: $T1 + c \leq T2$

Convex interval relations may be expressed by inequalities:

I1 during I2 => $I2.B \leq I1.B$, $I1.E \leq I2.E$

The temporal relations of an occurrence model are represented by a constraint net:

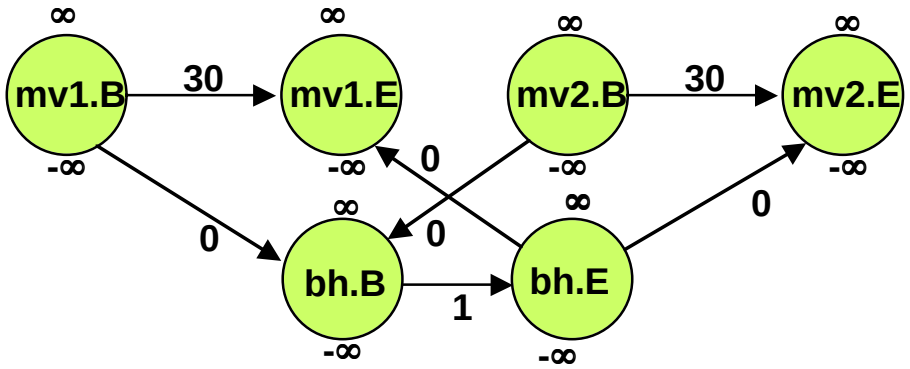


For general models one typically has
$T_{min} = -\infty$, $T_{max} = \infty$

Occurrence Recognition by Constraint Propagation (1)

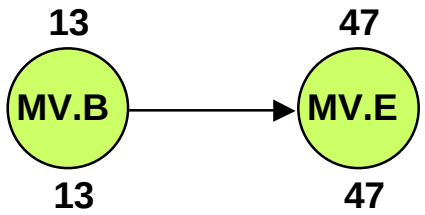
Matching an occurrence model with a time-varying scene:

1. **Initialize constraint net of occurrence model**



2. **Compute qualitative scene predicates**

z.B. (move obj1 13 47)
(behind obj1 obj2 20 33)
...



Occurrence Recognition by Constraint Propagation (2)

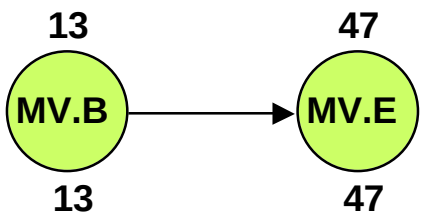
3. Instantiate predicates (sub-occurrences) in occurrence model

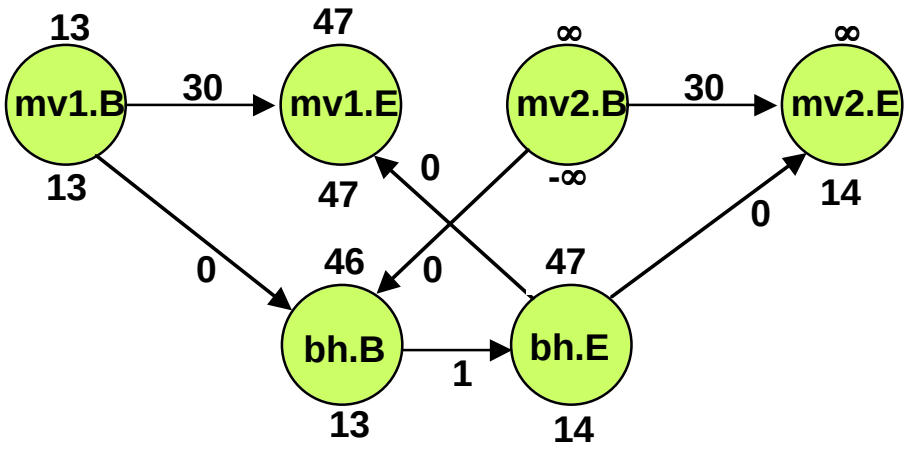
propagate minima and maxima of time points through constraint net:

- minima in edge direction **T2min´= max {T2min, T1min+c}**

- maxima against edge direction **T1max´= min {T1max, T2max-c}**

Example: MV in scene instantiates mv1of model





Occurrence Recognition by Constraint Propagation (3)

4. Consistency and completeness test

A (partially) instantiated model is inconsistent, if for any node T one has: Tmin > Tmax

=> search for alternative instantiations or terminate with failure

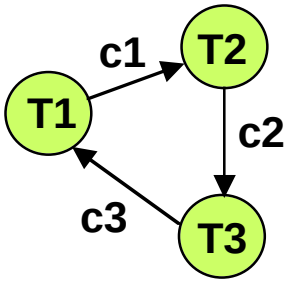
An occurrence has been recognized if the occurrence model is instantiated with sufficient completeness and the instantiation is consistent.

Note:

- **Incremental occurrence recognition follows an evolving scene**
- **A-posteriori occurrence recognition is carried out after observing a scene (choice of order)**
- **Partially instantiated models may be used for scene prediction**

Convergence and Complexity

Consider cycles in constraint net:



$\sum_i c_i > 0$ => **model is inconsistent**

$\sum_i c_i \leq 0$ => **consistency test complete after one cycle**

- **a consistency test of M binary constraints requires M steps (~ number of edges in constraint net)**
- **for occurrence recognition, a consistency test must be carried out for each instantiation of each of the N model nodes**
- **number of edges in model is ~ N^2**

=> complexity of occurrence recognition is $O(N^3)$

- **complexity may increase with alternative instantiations**

Generalization of Temporal Relations

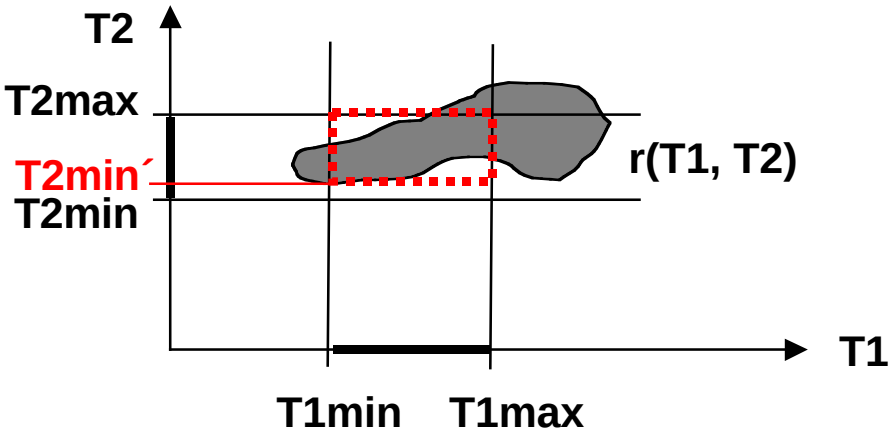
In principle, the constraint propagation procedure may be applied to arbitrary temporal relations.

<u>Requirement</u>:

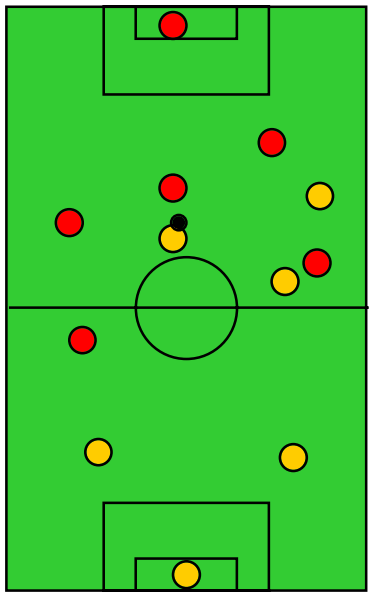
Compute extreme values T2min´ and T2max´ from T1min, T1max and r

$$T2min' = \max \{T2min, \min_{\substack{T1min \le T1 \le T1max \\ T2min \le T2 \le T2max}} r(T1, T2) \}$$

$$T2max' = \min \{T2max, \max_{\substack{T1min \le T1 \le T1max \\ T2min \le T2 \le T2max}} r(T1, T2) \}$$



Recognizing Intentions and Plans



Intention recognition in soccer games (Retz-Schmidt 91):

"Brandt dribbelt, um dem gegnerischen Tor nahe zu kommen"

("Brandt dribbles to get close to the opposing goal")

"Meier läuft zu Brandt, um ihn am Torschuß zu hindern"

("Meier runs to Brandt to prevent him from shooting a goal")

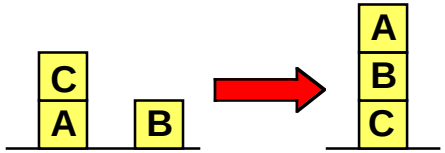
- **model-based representation of plans and counter plans**
- **partial instatiation allows predictions and explanations**

Intention recognition has been used in robot soccer (RoboCup)

Definition of Planning

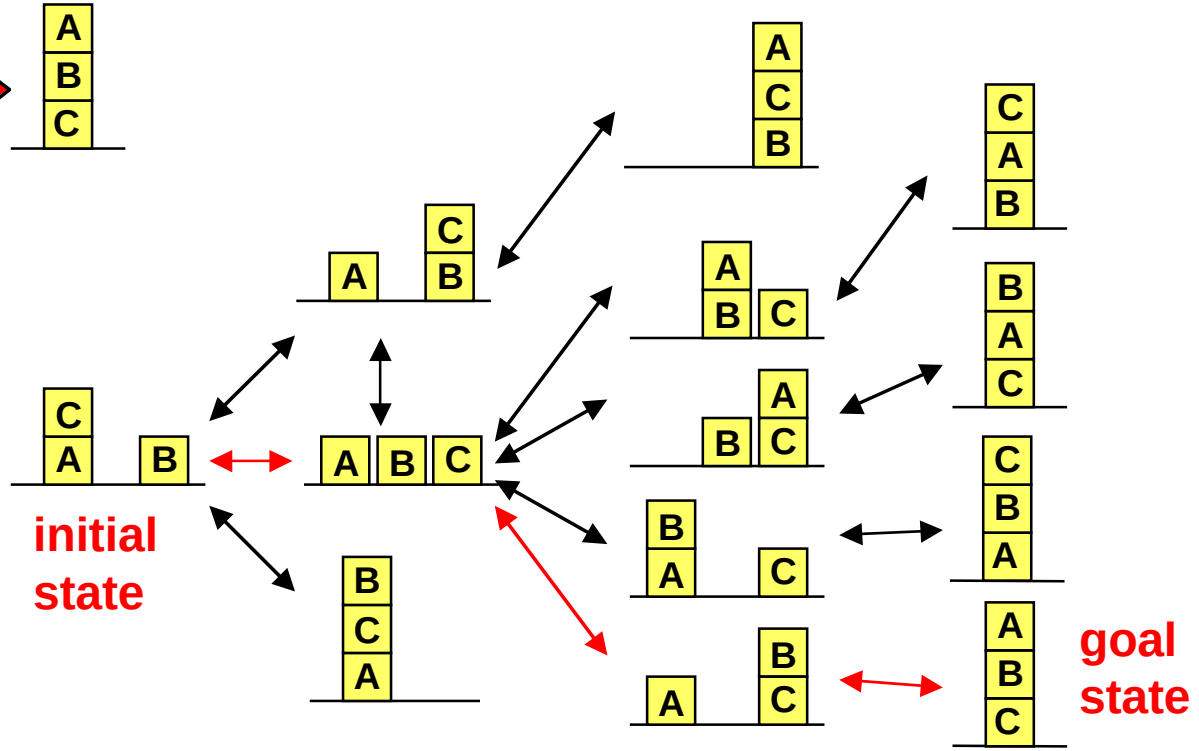
Planning is the construction of an action sequence which transforms an initial state into a goal state.

Example:



Search results in the plan:

move (C, Table)
move (B, C)
move (A, B)



Plan Recognition

Given:
- observed actions
- knowledge about likely goals of actor

⇒ predict further actions

⇒ plan own actions (cooperative or adversary)

Example ("smart room" or service robotics scenario):

Observations: tea-time: person gets up - person walks to door - ...

Predictions: ... - person goes to kitchen - person prepares tea

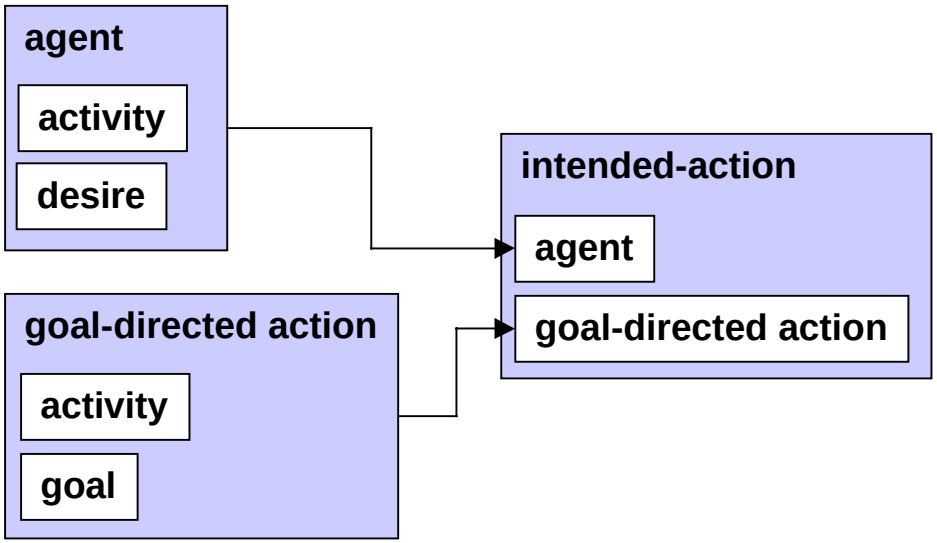
Plan recognition by

- matching partial action sequences to plan models (same principle as occurrence recognition)
- generating likely plans from the initial action sequence

Models for Intention Recognition

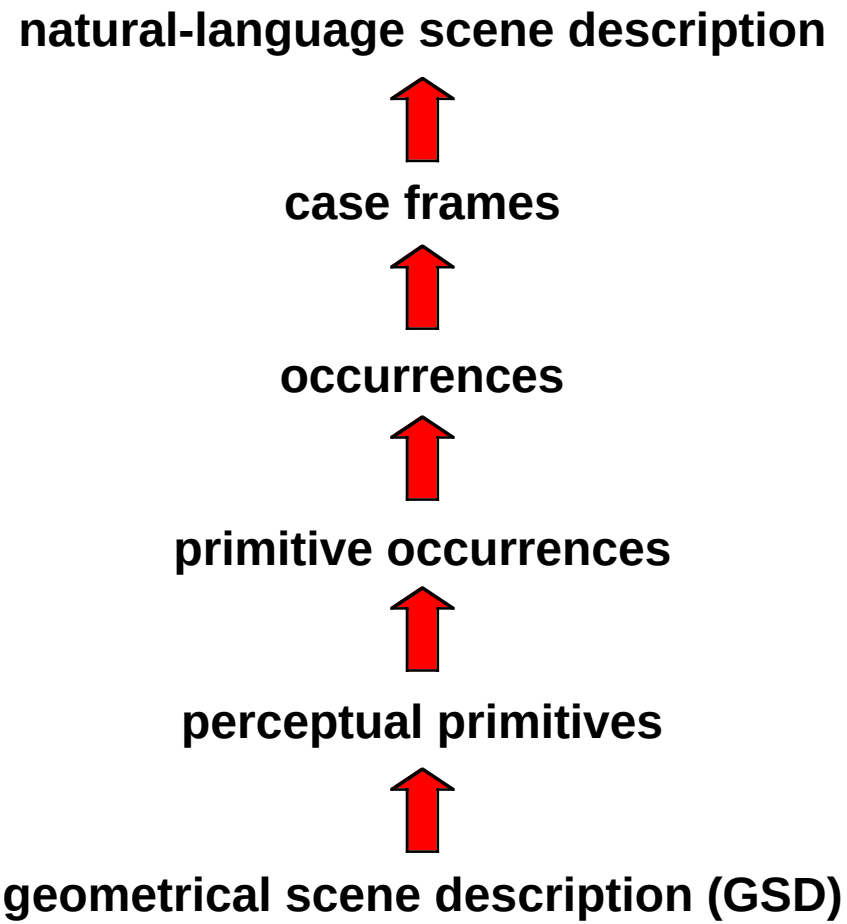
Intended actions may be described by aggregates which connect observable actions with (unobservable) intentions of an actor.

```
name:          scene-intended-place-cover
parents:       :is-a scene-intended-action
parts:         sipc-pc :is-a scene-place-cover
               sipc-ag :is-a scene-agent
                              with (sipc-ag.desire = sipc-pc.goal)
               constraints:   (temporal, spatial and other constraints on parts)
```



If an action is known to be goal-directed and an agent performs such an action, the agent is ascribed the intention to attain the goal.

From Scene Data to a Natural-language Scene Description

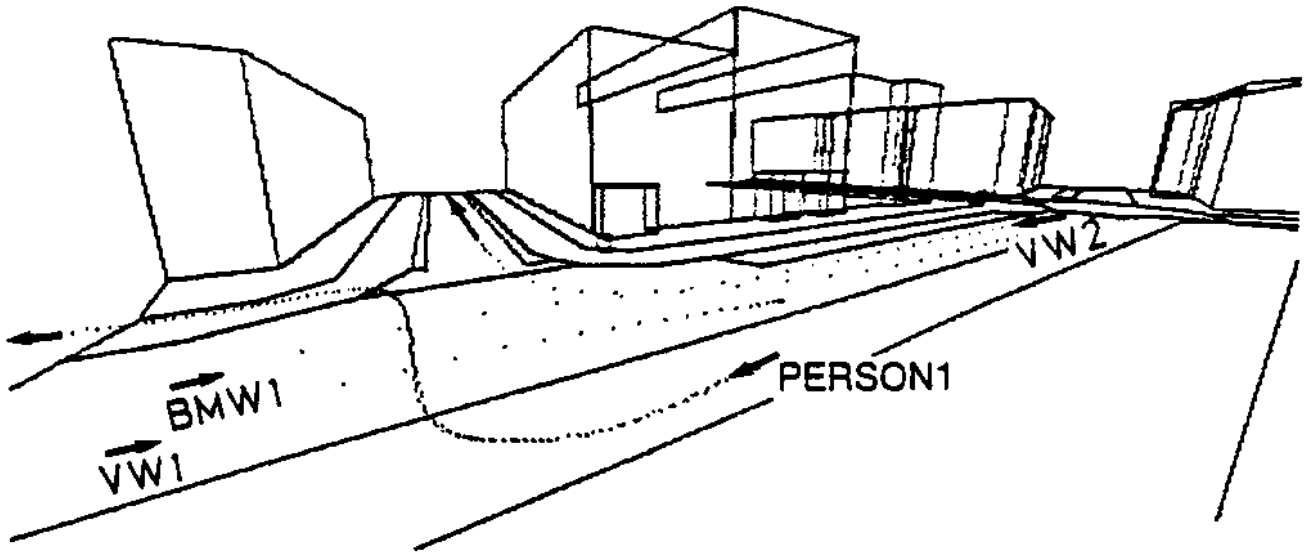


Geometrical Scene Description (GSD)

Quantitative description of all objects in a time-varying scene:

- **name of all objects (class or identity)**
- **position of all objects at all times (location and orientation)**
- **illumination (if required for high-level description)**

<u>Example</u>: Synthetical street scene of project NAOS

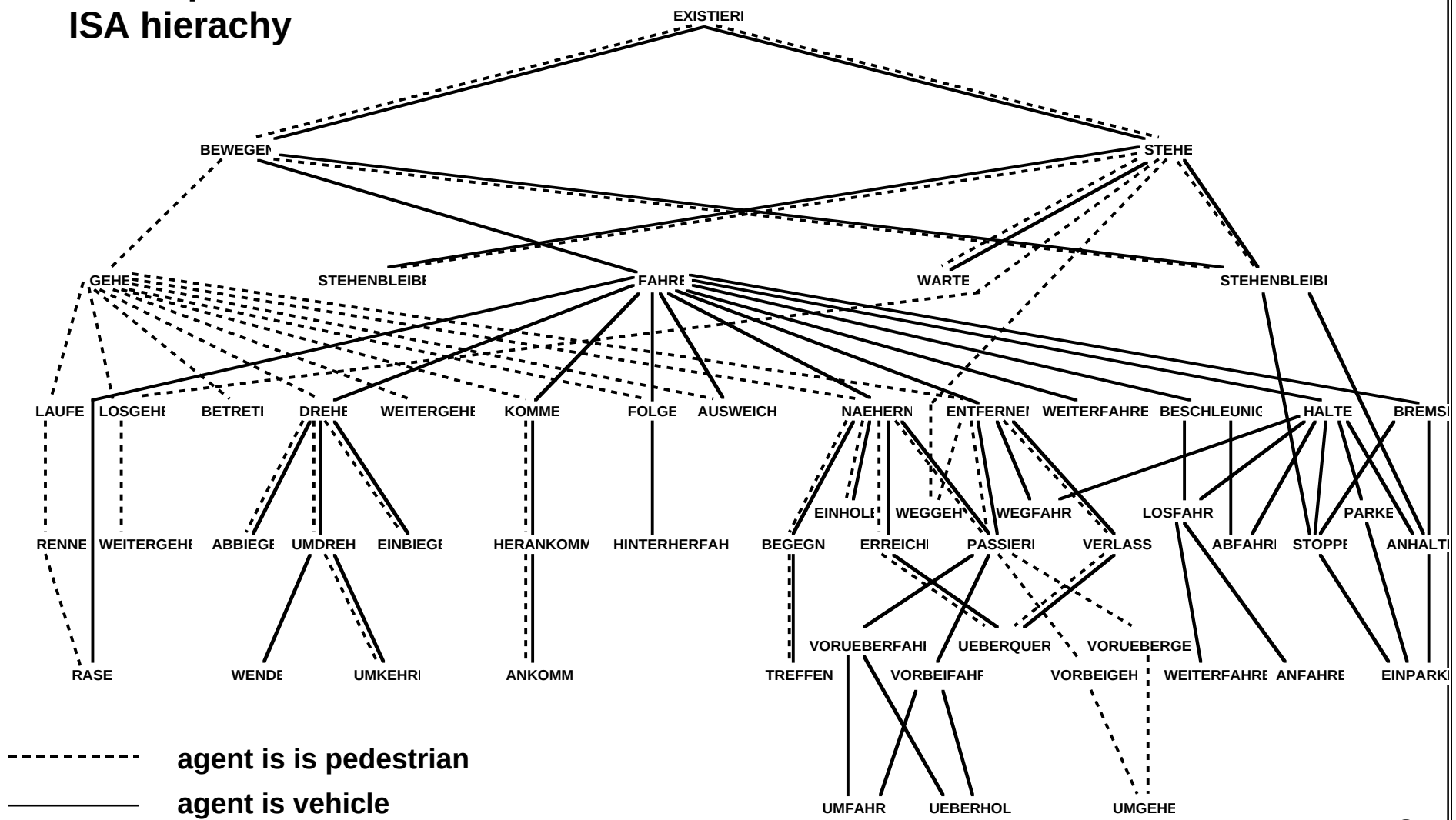


Typical data of a GSD

```
                location          orientation        time
(LAGE VW2 (779. 170. 0.) (-1.0 0.0 0.0) 0)
(LAGE VW2 (753. 170. 0.) (-1.0 0.0 0.0) 1)
(LAGE VW2 (727. 170. 0.) (-1.0 0.0 0.0) 2)
(LAGE VW2 (701. 170. 0.) (-1.0 0.0 0.0) 3)
(LAGE VW2 (675. 170. 0.) (-1.0 0.0 0.0) 4)
(LAGE VW2 (649. 170. 0.) (-1.0 0.0 0.0) 5)
(LAGE VW2 (623. 170. 0.) (-0.999 0.037 0.0) 6)
(LAGE VW2 (596. 171. 0.) (-1.0 0.0 0.0) 7)
(LAGE VW2 (570. 171. 0.) (-1.0 0.0 0.0) 8)
(LAGE VW2 (544. 171. 0.) (-1.0 0.0 0.0) 9)
(LAGE VW2 (518. 171. 0.) (-0.999 0.0383 0.0) 10)
(LAGE VW2 (492. 172. 0.) (-1.0 0.0 0.0) 11)
(LAGE VW2 (466. 172. 0.) (-1.0 0.0 0.0) 12)
(LAGE VW2 (440. 172. 0.) (-0.999 0.0383 0.0) 13)
(LAGE VW2 (414. 173. 0.) (-1.0 0.0 0.0) 14)
(LAGE VW2 (388. 173. 0.) (-0.999 0.037 0.0) 15)
(LAGE VW2 (361. 174. 0.) (-1.0 0.0 0.0) 16)
(LAGE VW2 (335. 174. 0.) (-0.999 0.038 0.0) 17)
•
•
•
```

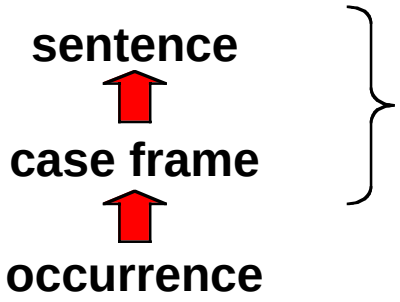
Hierachy for object motions in street traffic

mix of part_of and ISA hierachy



Generating a natural-language description

Principle:



techniques of language-oriented AI

Problems:

- Which occurrences should be selected for verbalization?
- Which deep cases should be filled?
- Which addtitional time or location information is required?
- In which order should the information be presented?

Solution:

Speech planning based on hearer simulation

informing a hearer <=> enabling a hearer to imagine the scene

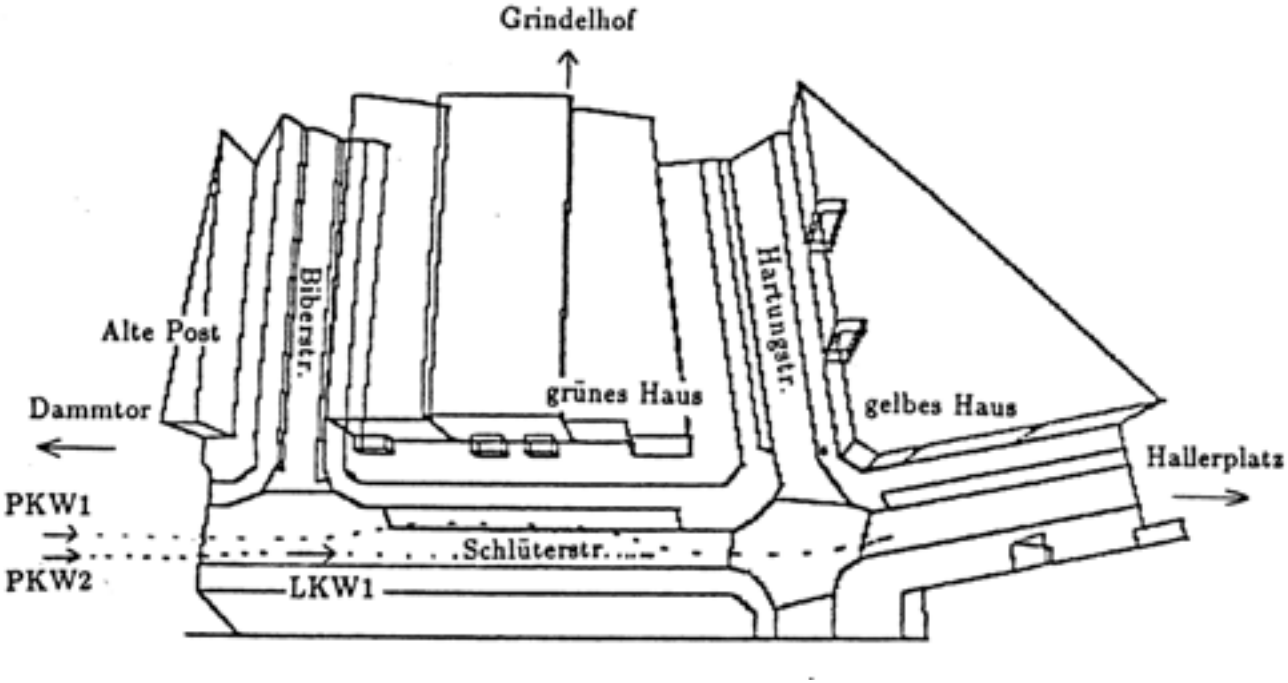
Standard plan for generating natural-language scene descriptions

- rules which assure that the hearer will be able to image the scene
- summary + descriptions of all object trajectories, each in chronological order
- no explicit hearer simulation

Description of an object trajectory

1. Each time interval is described by the most special occurrence
2. The first occurrence begins at the beginning of the scene
3. The next occurrence follows in temporal order
4. Location information is given my prepositional expressions as required
5. Temporal information is given by prepositional expressions or references to other occurrences as required

Example of an automatically generated traffic scene description



DIE SZENE ENTHAELT DREI BEWEGTE OBJEKTE: ZWEI PKWS UND EINEN LKW.

EIN GELBER PKW FAEHRT IN RICHTUNG HALLERPLATZ. DABEI UEBERHOLT ER DEN LKW AUF DER SCHLUETERSTRASSE. DER GELBE PKW RAST VON DER ALTEN POST VOR DAS GELBE HAUS. ER ERREICHT DIE HARTUNGSTRASSE. ER HAELT AN. ER HAELT.

EIN SCHWARZER PKW ERREICHT DIE SCHLUETERSTRASSE. ER NAEHERT SICH DEM LKW VON DER ALTEN POST. DER SCHWARZE PKW FAEHRT IN RICHTUNG HALLERSTRASSE.

DER LKW FAEHRT VON DER ALTEN POST VOR DAS GRUENE HAUS. DABEI STOPPT ER VOR IHM. ER HAELT. ER FAEHRT IN RICHTUNG DAMMTOR WEITER. ER ENTFERNT SICH VON DEM GELBEN PKW. DER LKW HAELT AN. ER HAELT.

Selecting prepositions for trajectory location information

<u>Idea</u>: Simulate natural language conventions by algorithms

In CITYTOUR (Wahlster 87) location expressions are generated depending on the trajectory of the observer: