Higraphs, a Visual Formalism

- abstract and concrete visual *syntax* + basic *semantics* *refined* in specific formalisms such as Statecharts
- visualizing *complex* information in a *compact* fashion
- visualizing *non-quantitative, structural* information

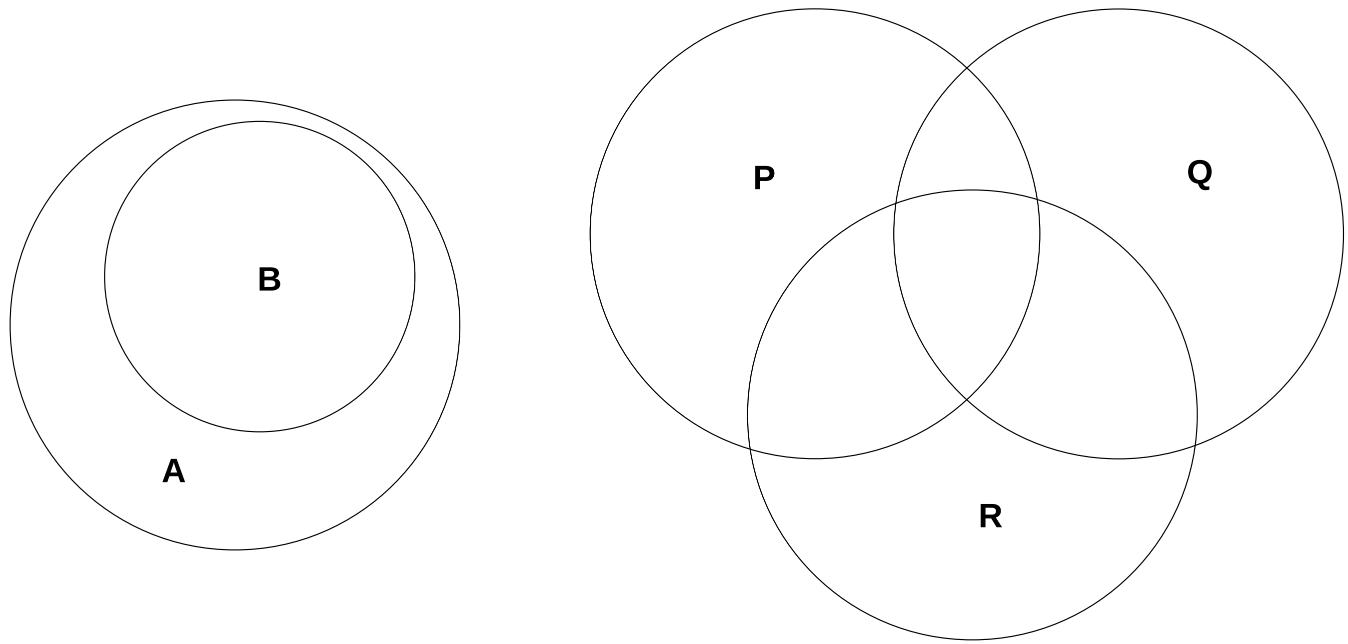
⇒ use *topological, not geometrical* constructs

⇒ combine:

1. Venn diagrams (Jordan curve: inside/outside): enclosure, intersection
2. hypergraphs (extending graphs)

David Harel. **On visual formalisms**. CACM, 31(5):514-530, May 1988.

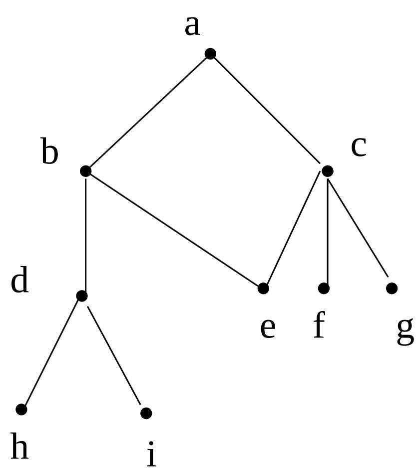
Higraphs 1. Venn diagrams, Euler circles



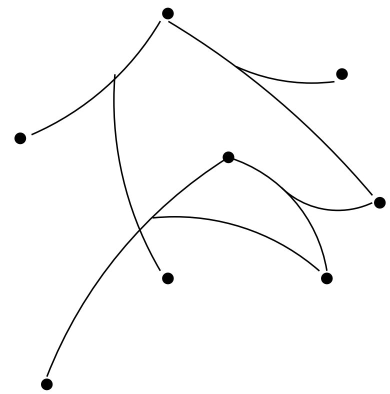
- visual syntax: *topological* notions (as opposed to geometrical) insideness, enclosure, intersection, exclusion

- semantics: *mathematical set operations*
  subset ($B \subset A$), union ($P \cup Q$), intersection ($P \cap Q$), difference ($P \setminus Q$)

Higraphs 2. Hypergraphs



a graph



a hypergraph

- syntax: *topological* notion of *connectedness*
- semantics: *relations* between sets

$\leadsto$ graph: edges encode a binary relation $G \subseteq X \times X$

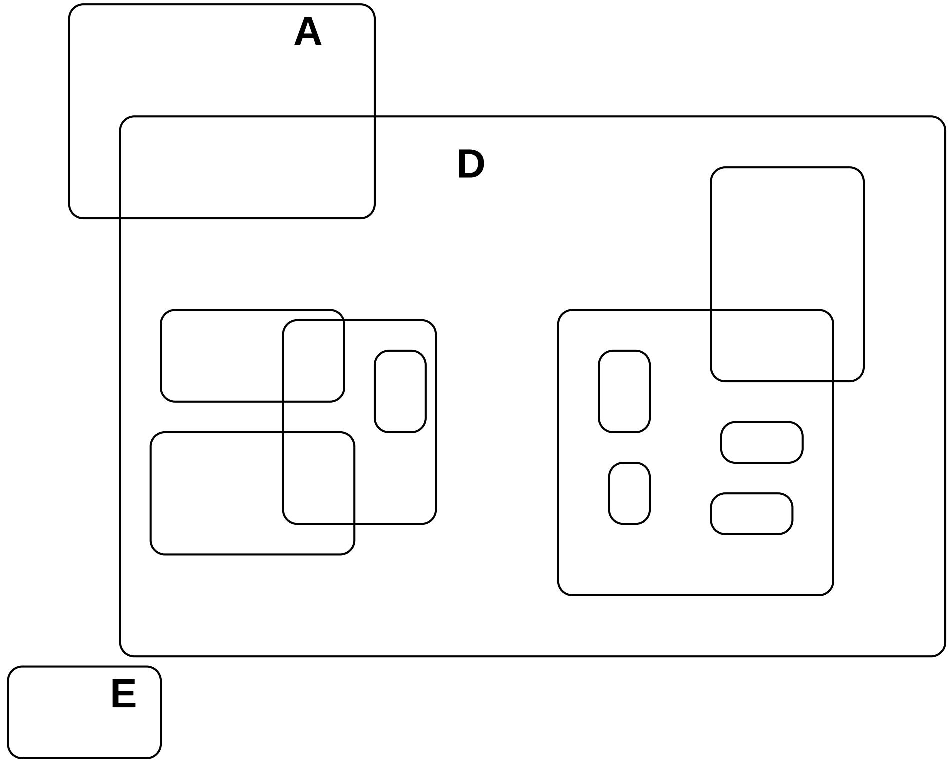
$\leadsto$ hypergraph: hyperedges encode non-binary relation $HG \subseteq 2^X$ (undirected), $HG \subseteq 2^X \times 2^X$ (directed).

Higraphs

Combine:

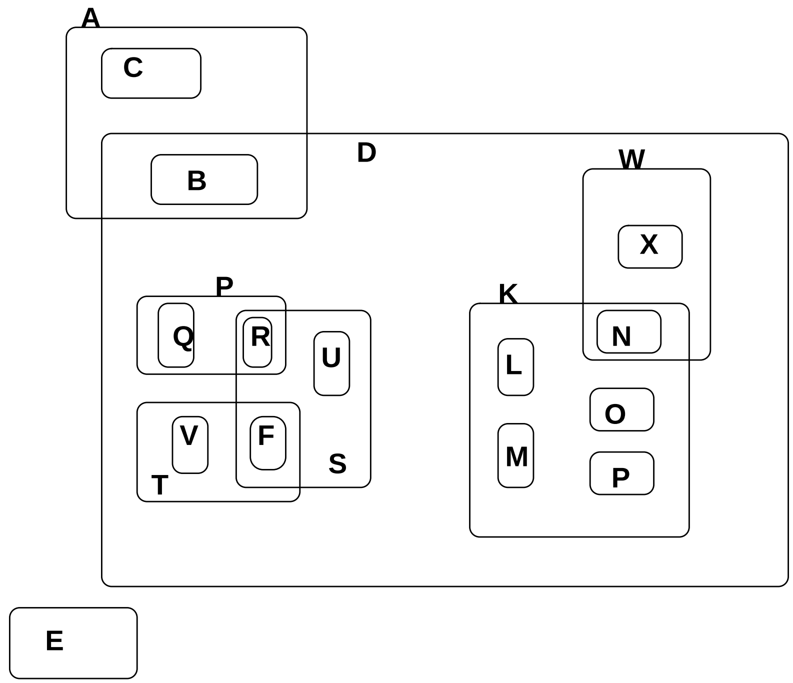
1. sets + cartesian product
2. hypergraphs

Visual syntax: blobs



- **syntax**: blob
- **semantics**: set
- **syntax**: insideness
- **semantics**: subset $\subset$ (not membership $\in$)

Unique Blobs



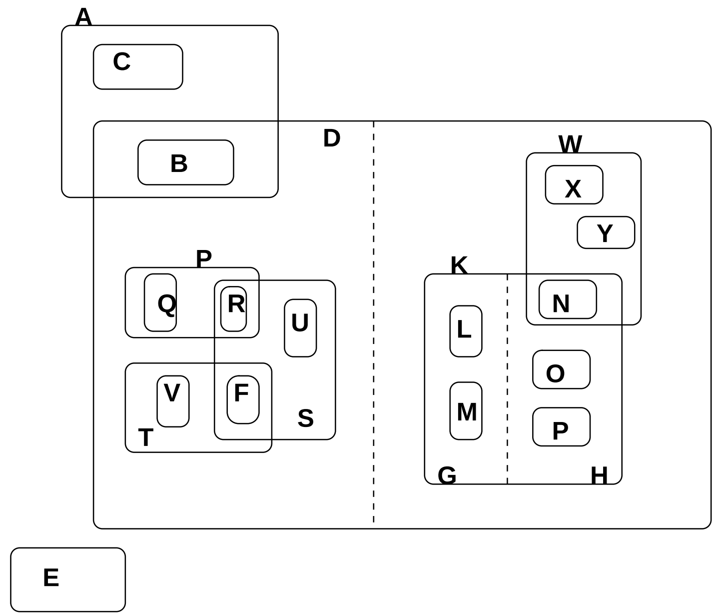
- **syntax**: empty space
- **semantics**: meaningless, must identify intersection explicitly

$\Rightarrow$ atomic blobs are *identifiable* sets (*e.g.*, $A \cap D$ identified as B)

- **syntax**: non-atomic blobs (contain other blobs)
- **semantics**: *union* of enclosed sets (*e.g.*, $K = L \cup M \cup N \cup O \cup P$)

syntax: *orthogonal* components
semantics: *unordered* cartesian product $\otimes$

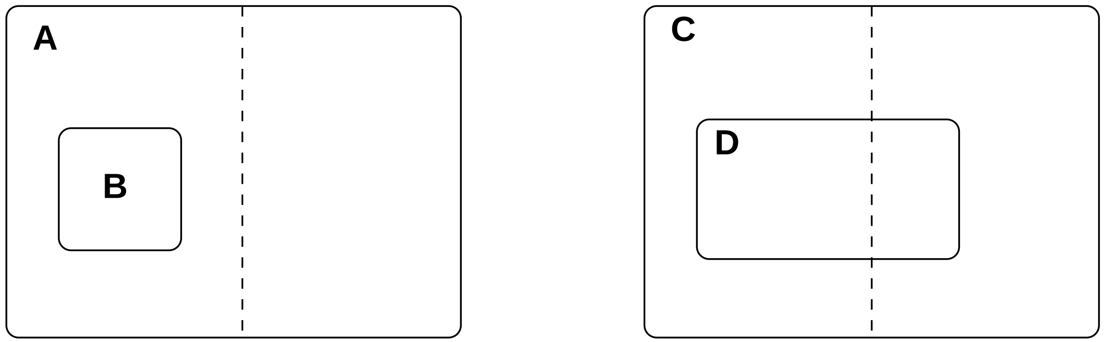
syntax:



semantics:

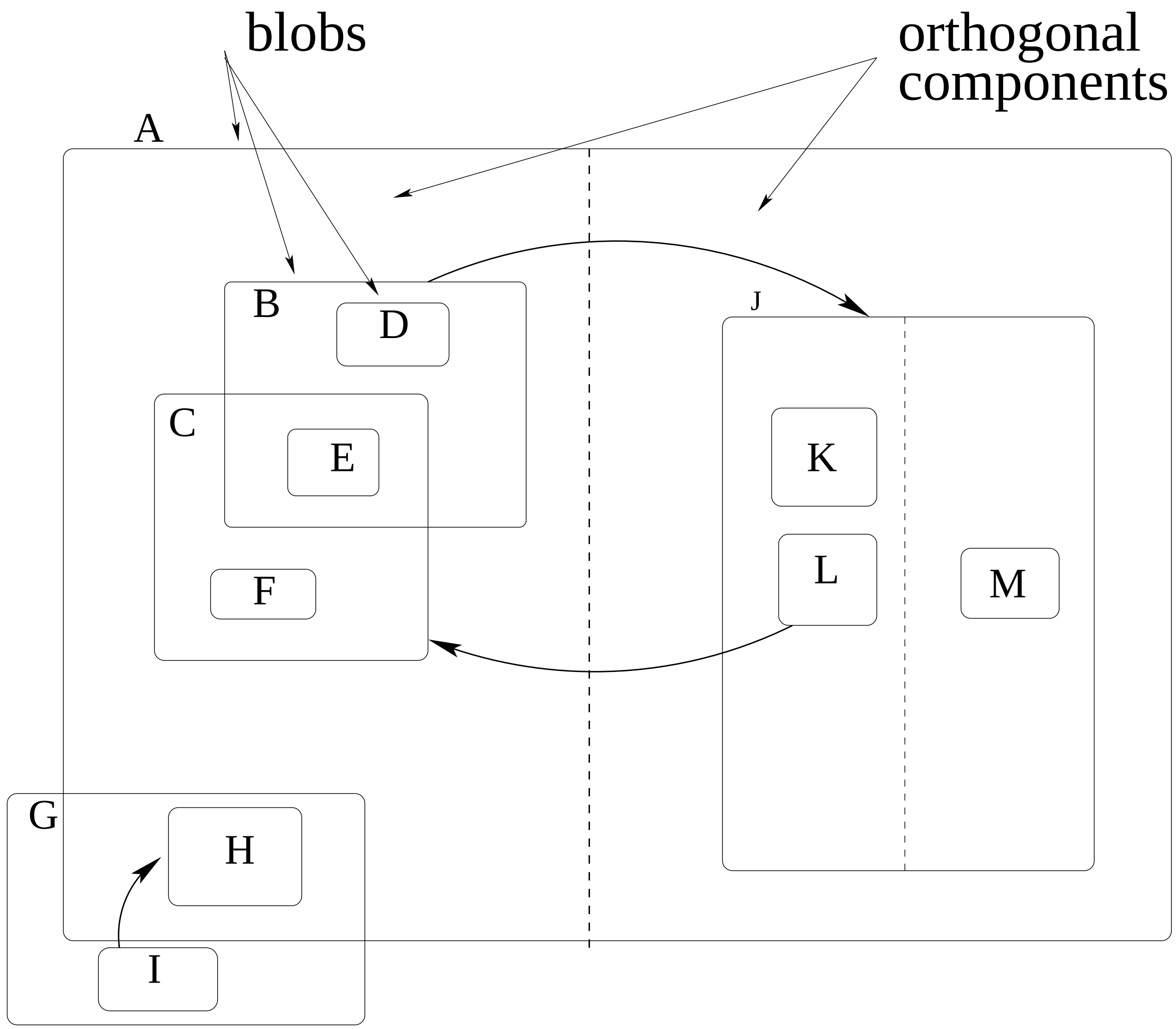
$$K = G \otimes H = (L \cup M) \otimes (N \cup O \cup P)$$

Meaningless constructs



- syntactically possible, semantically nonsense
- alternative semantics might give meaning to these constructs

Simple Higraph

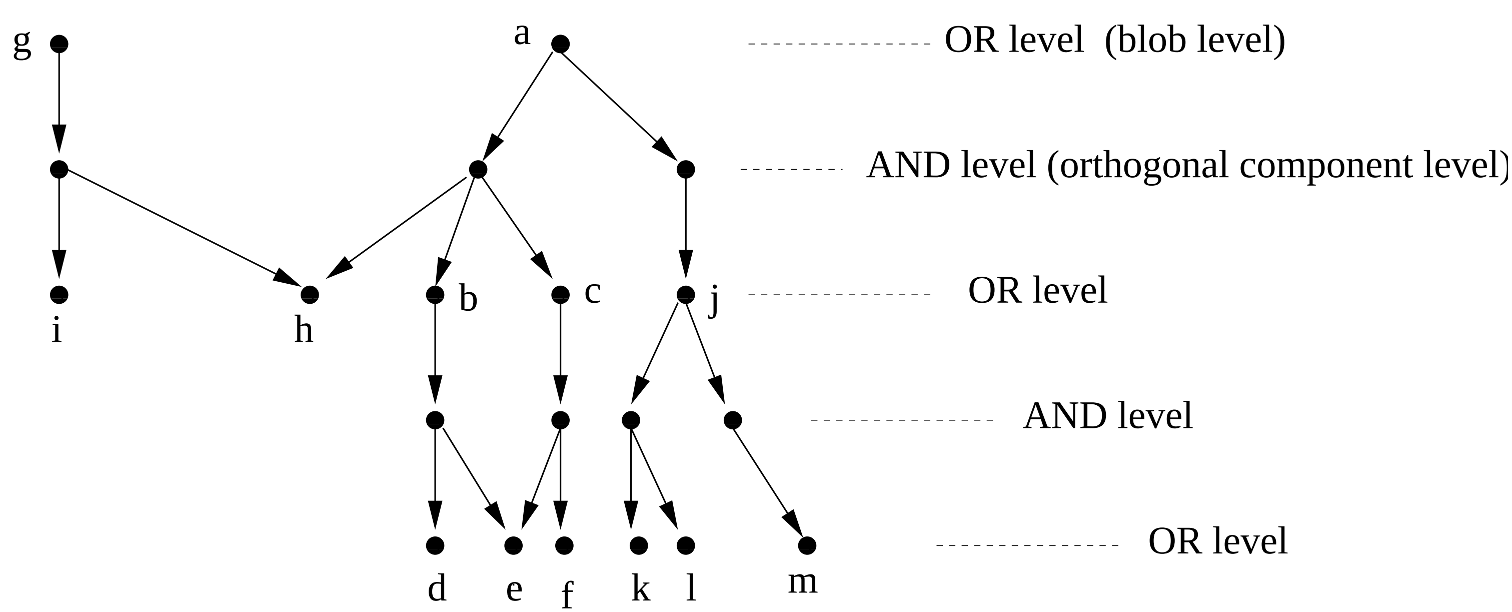


AND/OR levels

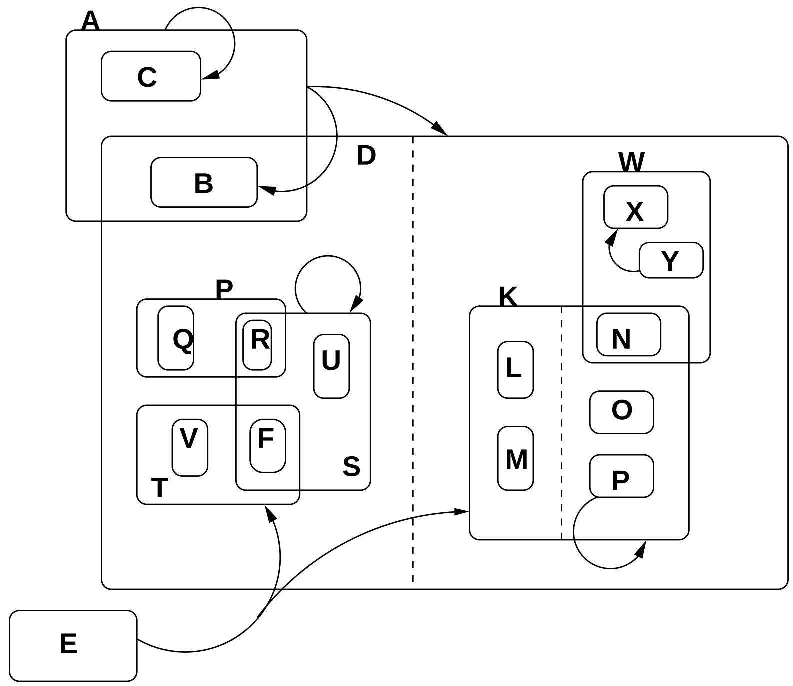
or: meaning of blobs (*e.g.*, K and L) in an orthogonal component:
$K \cup L \Rightarrow$ in K **or** in L.

and: meaning of orthogonal components (*e.g.*, $A1$ and $A2$):
$A1 \otimes A2 \Rightarrow$ in $A1$ **and** in $A2$.

Induced Acyclic Graph (blob/orth comp alternation)

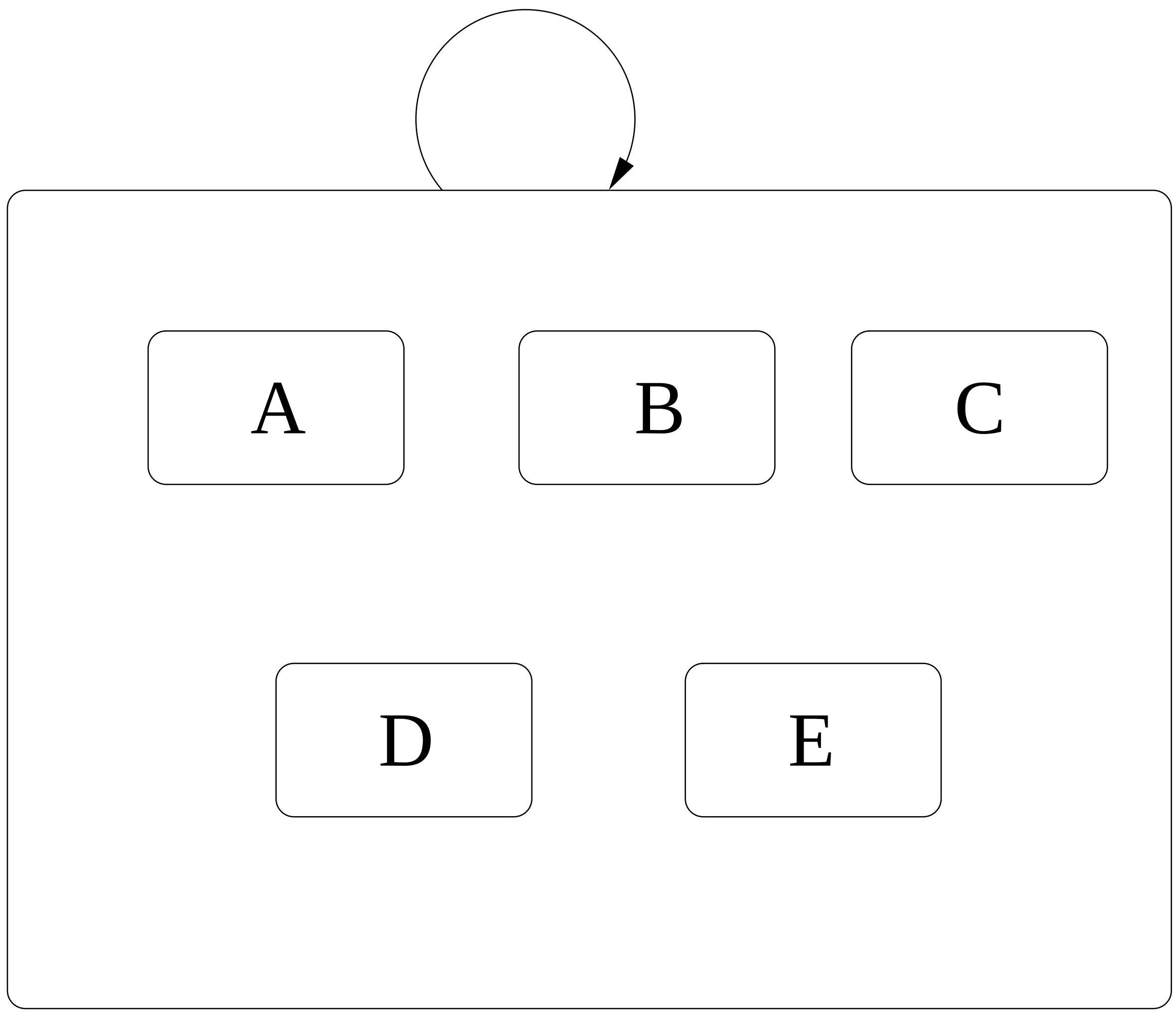


Adding (hyper) edges

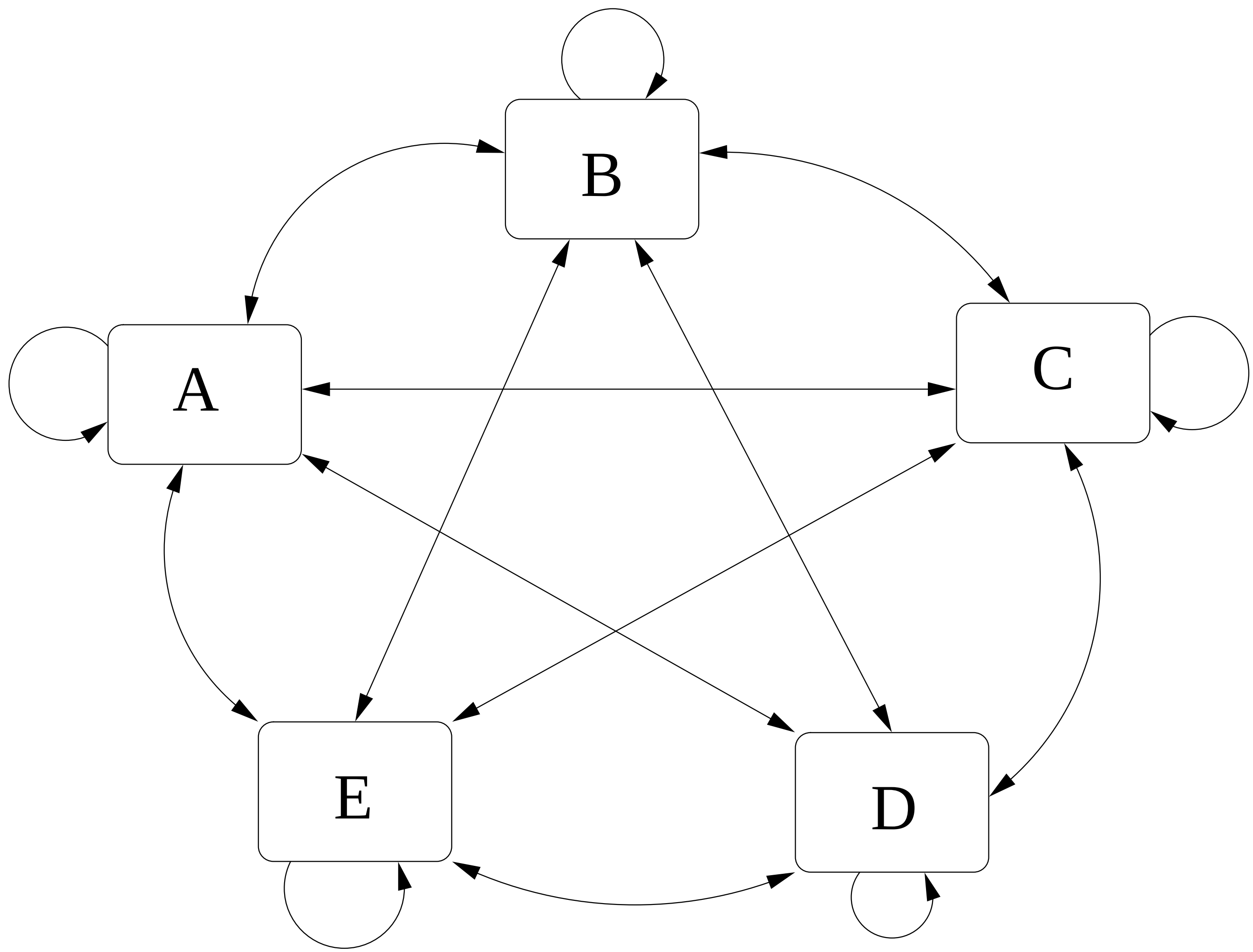


- syntax: *hyper*edges
- syntax: attach ends to contour of any blob, inter-level possible
- semantics: known for blobs and hyperedges, but *combination* ?

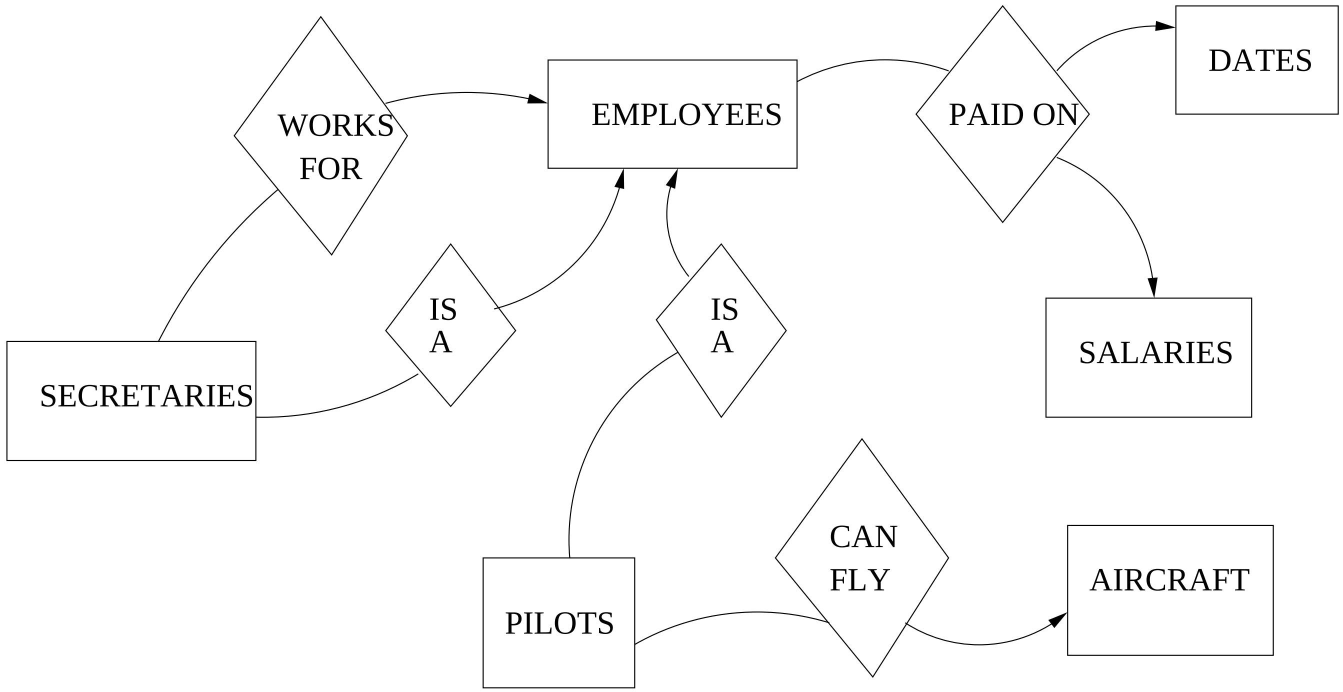
Syntax . . . Semantics ?



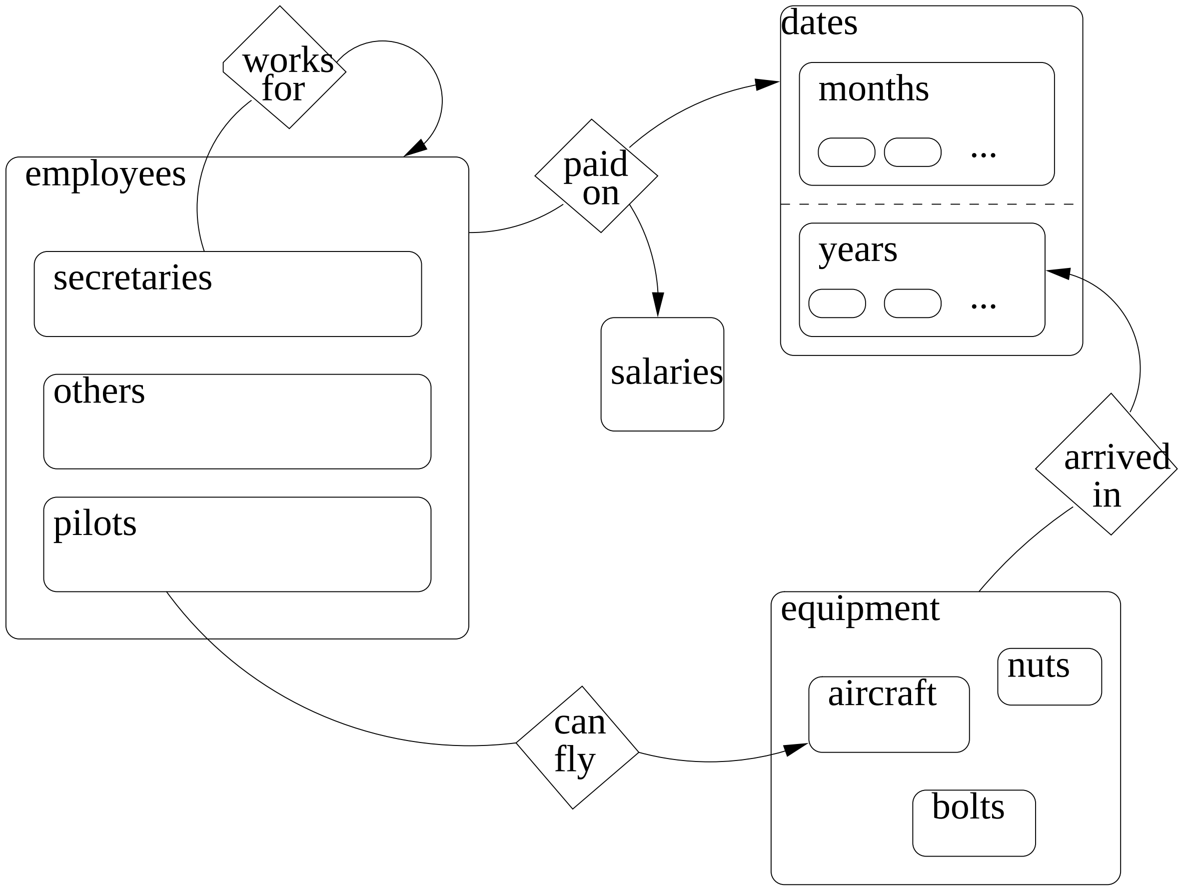
Fully connected semantics (clique)



Entity Relationship Diagram

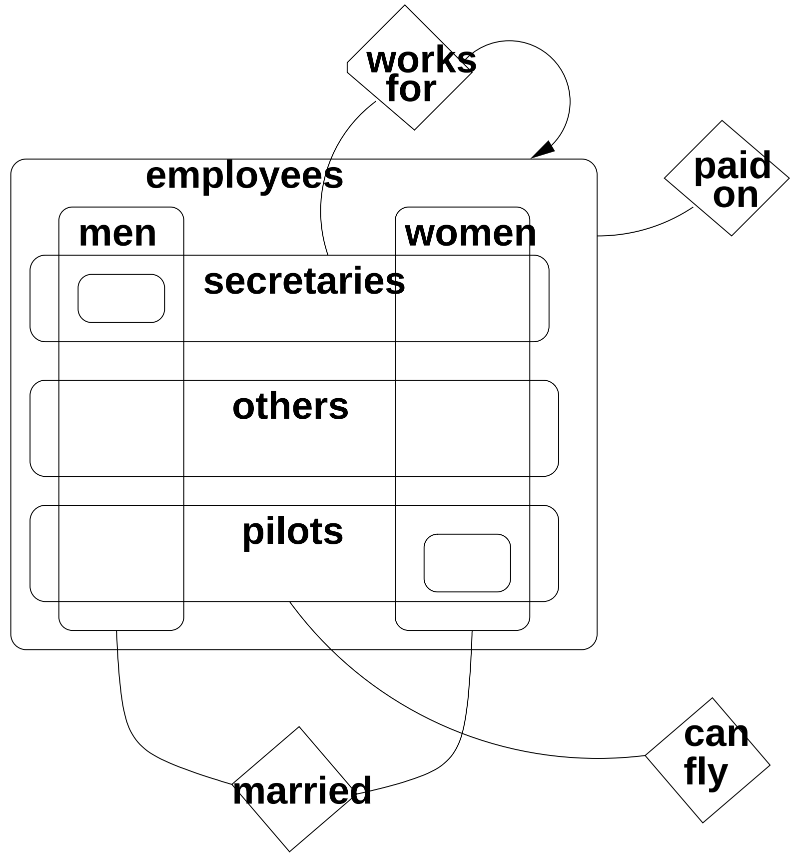


Higraph version of E-R diagram



replace `is-a` relationship by *insideness*

Extending the higraph is easy



try this in E-R . . .

Formally (syntax)

A higraph H is a quadruple

$$H = (B, E, \sigma, \pi)$$

B: finite set of all unique *blobs*

E: set of hyperedges

$$\subseteq X \times X, \quad \subseteq 2^X, \quad \subseteq 2^X \times 2^X$$

The subblob (direct descendants) function σ

$$\sigma : B \rightarrow 2^B$$

$$\sigma^0(x) = \{x\},\ \sigma^{i+1} = \bigcup_{y \in \sigma^i(x)} \sigma(y),\ \sigma^+(x) = \bigcup_{i=1}^{+\infty} \sigma^i(x)$$

Subblobs$^+$ cycle free

$$x \notin \sigma^+(x)$$

The partitioning function π associates *equivalence relationship* with x

$$\pi : B \rightarrow 2^{B \times B}$$

Equivalence classes π_i are (by definition) *orthogonal components* of x

$$\pi_1(x), \pi_2(x), \ldots, \pi_{k_x}(x)$$

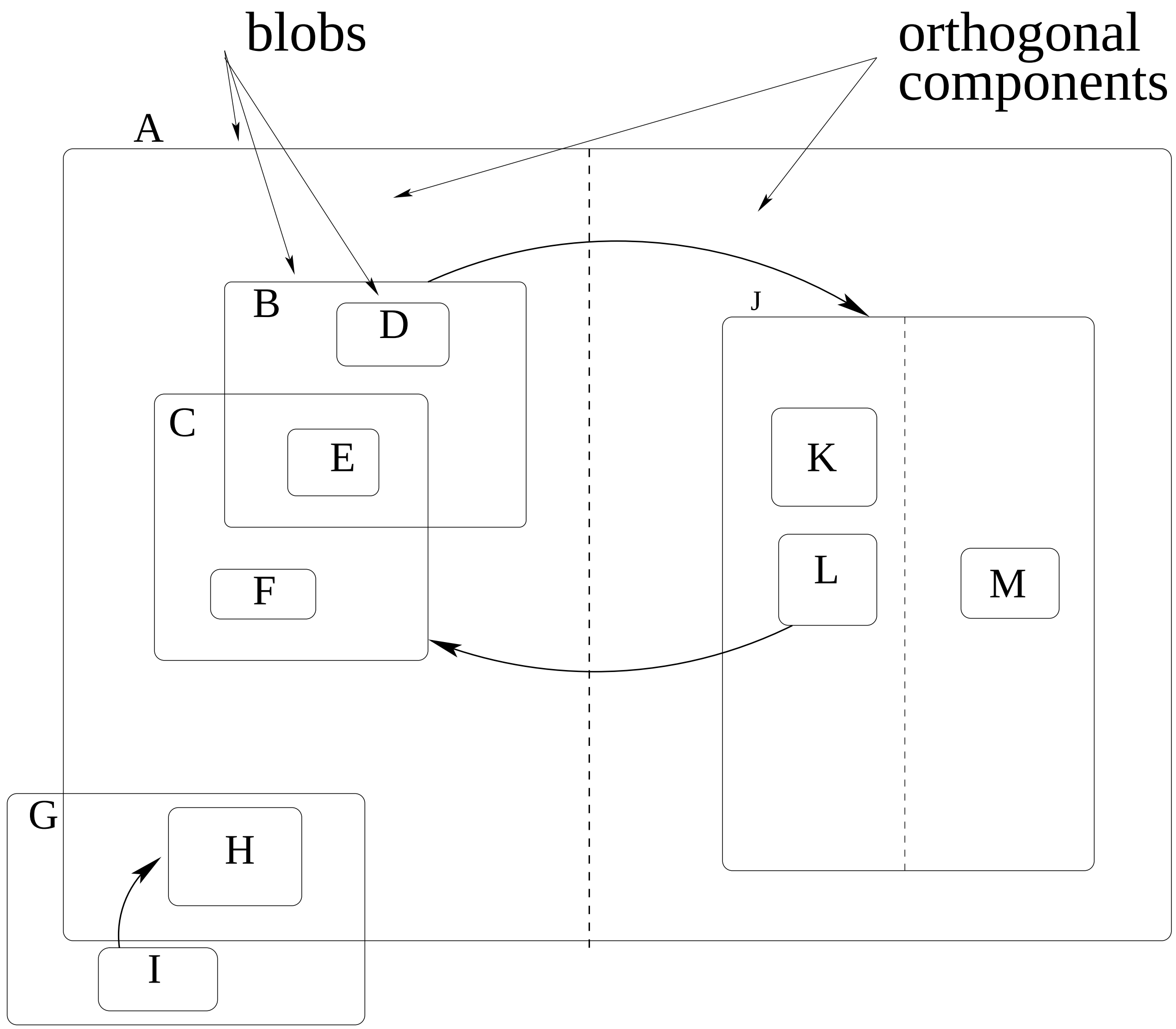
$k_x = 1$ means a single orthogonal component (no partitioning)

Blobs in different orthogonal components of x are *disjoint*

$$\forall y, z \in \sigma(x) : \sigma^+(y) \cap \sigma^+(z) = \emptyset$$

unless in the same equivalence class

Simple Higraph



Orthogonal Components induced by π

$$B = \{A, B, C, D, E, F, C, G, H, I, J, K, L, M\}$$

$$E = \{(I, H), (B, J), (L, C)\}$$

$$\rho(A) = \{B, C, H, J\}, \rho(G) = \{H, I\}, \rho(B) = \{D, E\}, \rho(C) = \{E, F\},$$

$$\rho(J) = \{K, L, M\}$$

$$\rho(D) = \rho(E) = \rho(F) = \rho(H) = \rho(I) = \rho(K) = \rho(L) = \rho(M) = \emptyset$$

$$\pi(J) = \{(K, K), (K, L), (L, L), (L, K), (M, M)\}$$

Induces *equivalence classes* $\pi_1(J) = \{K, L\}$ and $\pi_2(J) = \{M\}$, ...
These are the *orthogonal components*

Higraph applications

- E-R diagrams
- object-model diagrams
  Example use in
  David Harel and Eran Gery. **Executable object modeling with statecharts**. IEEE Computer, pages 31-42, 1997.
- UML activity diagrams
- Statecharts