Statecharts aka Harel Charts

- visual formalism
- higraph based (rigour)
- diverse applications;
  in particular: concurrent systems behaviour
Visualising Information

- complex
- non-quantitative, structural
- topological, not geometrical
- Euler
  - graphs (nodes, edges: binary \textit{relation}); hypergraphs
  - Venn diagrams (Jordan curve: inside/outside): enclosure, intersection
Venn diagrams, Euler circles

- **topological notions:**
  - enclosure, exclusion, intersection

- Used to represent *mathematical* set operations:
  - union, intersection, difference
Hypergraph

- topological notion: connectedness
- Used to represent relations between sets.
- Hyperedges: $\subset 2^X \times 2^X$
Higraphs: combining graphs and Venn diagrams

- sets + cartesian product
- hypergraphs
Blobs: set *inclusion, not membership*
Unique Blobs (atomic sets, no intersection)

- empty space has no meaning, intersection must be identified
- atomic blobs are identifiable sets
- other blobs are union of enclosed sets
Unordered Cartesian Product: Orthogonal Components

\[ K = G \times H = (L \cup M) \times (N \cup O \cup P) \]
Meaningless constructs

A

B

C

D
Simple Higraph
Induced Acyclic Graph (blob/orth comp alternation)
Adding (hyper) edges

- hyperedges
- attach to contour of any blob
- inter-level possible (global variables binding)
Clique Example

A

B

C

D

E
Clique: fully connected semantics
Entity Relationship Diagram (is-a)
Higraph version of E-R diagram
Extending the E-R diagram

- employees
  - men
  - women
- secretaries
- others
- pilots
  - can fly
  - married
  - works for
  - paid on

hv@cs.mcgill.ca
McGill, October, 2001
Higraph applications

- E-R diagrams
- data-flow diagrams (activity diagrams)
  edges represent (flow of) data
- inheritance
- StateCharts
Formally

A higraph \( H \) is a quadruple \((B, E, \rho, \Pi)\)

\( B \): set of all unique blobs

\( E \): set of hyperedges \( \subset 2^X \times 2^X \)

\( \rho : B \to 2^B \), the hierarchy function

\( \Pi : B \to 2^{B \times B} \), the partitioning function (equivalence relationship)

\( \rho \) defines the direct descendants of a blob

\( \rho^0(x) = \{x\} \)

\( \rho^+(x) = \bigcup_{i=1}^{+\infty} \rho^i(x) \), cycle free: \( x \not\in \rho^+(x) \)
Simple Higraph

blobs

orthogonal components

Higraphs/State Charts/Object Modelling

McGill, October, 2001

hv@cs.mcgill.ca
Induced Orthogonal Components

\[ B = \{A, B, C, D, E, F, C, G, H, I, J, K, L, M\} \]
\[ E = \{(I, H), (B, J), L, C\} \]
\[ \rho(A) = \{B, C, H, J\}, \rho(G) = \{H, I\}, \rho(B) = \{D, E\}, \rho(C) = \{E, F\}, \]
\[ \rho(J) = \{K, L, M\}, \rho(D) = \rho(E) = \rho(F) = \rho(H) = \rho(I) = \rho(K) = \rho(L) = \rho(M) = 0 \]
\[ \Pi(J) = \{(K, K), (K, L), (L, L), (L, K), (M, M)\} \]

Induces equivalence classes \{K, L\} and \{M\},…
These are the orthogonal components
StateCharts =
state diagrams + depth + orthogonality + broadcast

- Reactive Systems (event driven, react to internal and external stimuli)
- like Petri Nets, CSP, CCS, sequence diagrams, ...
- graphical but formal and rigorous for
  - analysis
  - code generation
- solve FSA problems:
  - flat $\Rightarrow$ hierarchy $\Rightarrow$ re-use
  - represent large number of transitions concisely
  - represent large number of states concisely
  - sequential $\Rightarrow$ concurrent
Depth (XOR)
Orthogonality (AND), flattening $\Rightarrow$ semantics

McGill, October, 2001
hv@cs.mcgill.ca
Higraphs/State Charts/Object Modelling
Broadcasting (output events)
History States

Autonomous

Red

Green

Yellow

Manual

H

resume

t

t

to_manual

McGill, October, 2001
hv@cs.mcgill.ca
Higraphs/State Charts/Object Modelling
26/39
Executable Object Modelling

- analysis $\Rightarrow$ use cases $\Rightarrow$ sequence diagrams
- analysis $\Rightarrow$ use cases $\Rightarrow$ class diagrams
- $\Rightarrow$ Statecharts $\Rightarrow$ sequence diagrams $\Rightarrow$ test use cases
Executable Object Modelling with Statecharts

- OO development: intuitive and rigorous
- fully executable models (simulation)
- code synthesis
Executable Object Modelling with Statecharts

- Structure (classes, multiplicities, relationships)
  ⇒ Object-model diagrams (higraph version of ER-diagrams)

- Behaviour
  ⇒ StateCharts
Automated Railcar System
Scenarios (Use Cases)

- Car approaches terminal
- Car departs from terminal
- Passenger in terminal
Inheritance

- structural or behavioural
- interface subtyping
- Modify states
  - Decompose state in OR or AND components
  - Add sub-states to OR state
  - Add orthogonal components to any state