17. Multi-agent systems (MAS)

17.1 What are multi-agent systems?

- "Computer programs with actors situated in an interactive environment, which are each capable of flexible, autonomous, yet socially organized actions that can be, but need not be, goal-oriented or directed towards pre-determined objectives." (Luger, 2005)
- Agents can be
  - Biologic (people and animals)
  - Computational (software agents)
  - Robotic with sensors.
- Agents are entities that carry out tasks on behalf of a user.
- Relatively recent trend to use MAS is in modeling of real-world phenomena:
  - Social simulations, group behavior, tragedies
  - Urbanization, land-use change, etc.

17.2 Properties of intelligent agents

- Intelligent
  - Possession of domain knowledge
  - Ability to operate in changing or exceptional situations.
- Situated
  - Agent receives input from the environment that can be viewed as another agent.
  - Agent can affect changes in the environment.
- Autonomous
  - Agents can operate without intervention of others.
  - Agents have control of their actions.

- Flexible
  - Reactive agents respond appropriately to stimuli coming from the environment or other agents.
  - Proactive agents are goal-oriented and have intentions.
  - Learning agents learn from their own experience, from other agents, from the environment, or from the user.
- Social
  - Agents interact with other agents or human user.
  - Agents communication happens on the knowledge level.
- Heterogeneous
  - Agents have individual characteristics.

17.3 Additional properties

- Agents can be
  - Cooperative or competitive.
  - Benevolent or adversarial.
  - Altruistic or selfish.
  - Trustworthy or betraying.
- Agents may degrade gracefully.
  - When encountering a problem they still can make some progress instead of totally failing.

17.4 Types of Agents

- Reactive agents
  - Simple *reactive* (reflex) agents
    - Rule based (for instance, email filters)
    - Do not accommodate changes
  - *Goal-oriented* agents try to achieve goals
    - Search, planning
    - Do not regard efficiency
  - *Utility-based* agents
    - Maximize the utility function ← presupposes rationality
    - Utility function not always obvious or rational for others.

A or B?
A: 80% chance of winning $4000
B: 100% chance of winning $3000

C or D?
C: 20% chance of winning $4000
D: 25% chance of winning $3000

- Interface agents
  - Aid human user in some specific task, for instance to learn to use a new software package.
- Mobile agents
  - Physical robots
  - Software agents that traverse network to other computers for efficiency
  - Raises many security questions.

17.5 Problem solving in MAS

- Ideal in domains that involve many problem solving methods, multiple viewpoints (e.g., different kinds of knowledge) and multiple entities (e.g., actors of different kinds).
- Agents acting together have more power than a single agent:
  - Each agent has incomplete information and insufficient resources for solving the entire problem.
  - Agents can communicate each other changes in the environment.
  - By collaborating agents can work together to solve a common goal.

- No global controller
  - Not about parallelization but coordination and collaboration.
  - Knowledge and input is decentralized.
  - Reasoning is asynchronous.
- Many expert systems are not agent-based (AB)
  - No direct link to environment
  - No autonomous reaction to changes in the environment, but the user requests the system to do reasoning.

17.6 MAS ≠ OOS

- Objects do not have control over their own behavior.
- Agents do not invoke methods on one another but request actions to be performed.
- Agents have reactive, proactive and social behavior.
- Agents are seen to have individual threads of control.

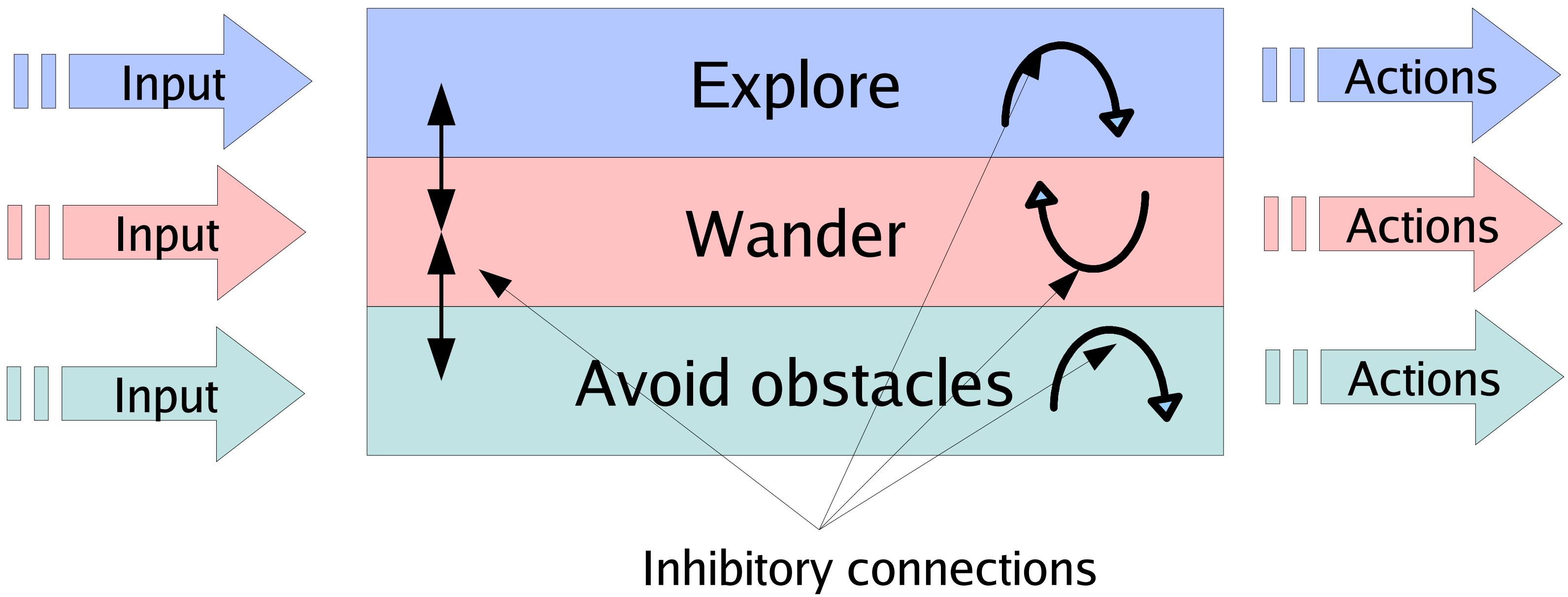
17.7 Implementation of MAS

- MAS a synthesis of AI techniques
  - Agents make decisions, plan, solve problems, communicate, perceive the environment, and learn from experience.
- How to design and implement the structure of intelligent system?
  - Procedural system: hierarchy of procedures
  - Object-oriented approach: classes, objects, events.
  - Knowledge representation issue:
    - Philosophical stance on intelligence
    - Domain specific requirements and constraints:
      - Optimality vs. real-time response

17.8 Architectures

- Brooks *subsumption* architecture (Brooks, 1986)
  - No explicit knowledge representation
  - No centralized intelligence or control
  - Reflexive and asynchronous
  - Hierarchically layered set of modules
    - Each layer is a Finite State Machines augmented with a timer (AFSM).
    - Each layer controls one aspect of agent's behavior.
    - Higher layers build upon the lower layers.
  - System is fast
    - No need to update and maintain the accuracy of the world model.

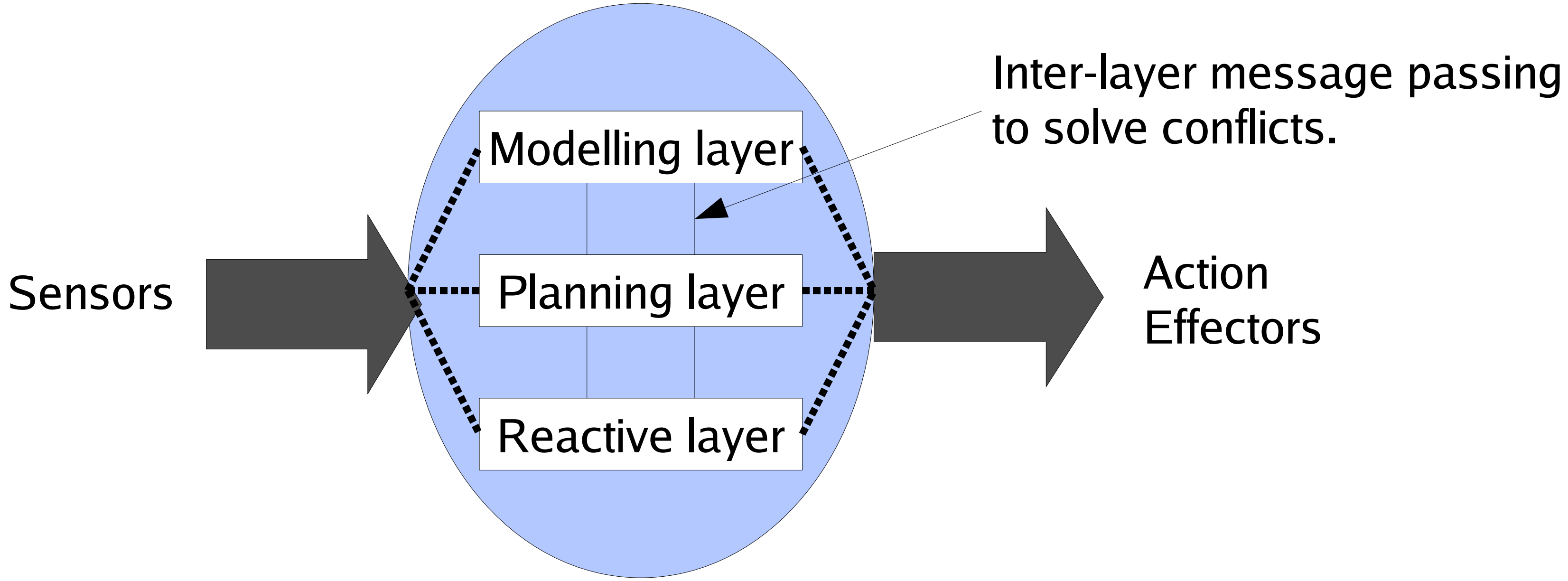
- When AFSM's input signal exceeds predetermined threshold, output is activated (triggered):
  - Input from sensors.
  - Output to actuators or to inputs of another AFSM.
- The input can be inhibited, which prevents the layer from triggering.



- *Beliefs Desires Intentions* (BDI) architecture
  - Rao & Georgeff, 1991
  - Beliefs
    - Statements about the environment the agent considers true (agent's knowledge).
    - Actions that the agent can perform.
    - Properties of the beliefs (for example, if they can change)
  - Desires
    - Agent's long-term goals (desired states of environment)
  - Intentions
    - Agent's deliberation or plan to reach the goal.
    - Intention to carry out an action.
    - Intention to reach a particular state.
  - Architecture ensures that beliefs, goals and intentions evolve rationally.
    - Events are responded in timely manner.
    - Beliefs are maintained consistently.
    - Plans are selected and executed rationally.

- *TouringMachines* (Ferguson, 1992)
  - Inspired by Brooks subsumption architecture.
  - Store and manipulate explicit representations of attitudes, beliefs and intentions.
  - Design of intelligent agent architectures for dynamic, and unpredictable environments (e.g., power plants, factories, space stations):
    - Deliberative
      - Embedded with means-ends reasoning, planning, natural language understanding, etc.
      - Maintenance of complete, up-to-date world model is costly.
    - Non-deliberative
      - Simple design: direct coupling of perception to action.
      - Minimal memory results in sub-optimal performance.
      - Actions hardwired.

- Operating in real world requires an architecture that can
  - Deal with multiple events at the same time at several levels, both in time and space.
    - Rational, goal-directed reasoning to accomplish the task.
    - Reasoning about events and their effects taking place in the environment, and predicting what is likely to happen in the future.
  - React to unforeseen events and recover from poor decisions
  - On top of accomplishing the task it was assigned!



17.9 Applications for MAS

- Interacts with a human, or replaces humans.
- Personal travel agent (PTA)
  - Collection of interacting agents booking flights, accommodations, car rental or rail travel by coordinating their actions to form a trip plan with no conflicts.
- Personalized services
  - Stock market byers
  - Shopping assistants
  - Automated tutoring
- Automated process control
- Information management
  - Filtering (e.g., email filters)
  - Information retrieval

- Mixed-initiative agents
  - Collaborative systems in which humans and automated systems work together towards a common goal.
  - Human-machine interaction and agent-agent interaction. Integrates:
    - knowledge representation
    - problem solving and planning
    - knowledge acquisition and learning
    - multi-agent systems
    - discourse theory
    - HCI

- Examples (AI magazine, Summer 2007):
  - TRIPS (Ferguson & Allen): collaborative problem-solving assistant based on BDI.
  - DiamondHelp (Rich & Sidner): general task-guidance system that assists a user in programming a washing machine or a thermostat.
  - Project Execution Assistant, PexA (Myers et al.): manages knowledge worker's time commitments, e.g., meetings and appointments.
  - GTrans (Fox & Zhang): planning
  - MAPGEN (Bresina & Morris): mission-critical planning of ground operations for Mars Rover.
  - STC (Cheetham & Goebel): helps call takers to diagnose problems with home appliances.

- Task issue
  - Division of responsibility
  - Complementarity:
    - Humans use intuition, common sense, creativity.
    - Computers have computational strength, speed, quick storage and retrieval of large quantities of information.
  - Expertise:
    - An expert assistant can guide a novice user.
    - Delegate routine tasks to the agents.
- Control issue
    - Strategies of shifting the initiative and control between the human and the agent: who does what and when?
    - Depends on the qualification of the participants, but also on the task.

- Awareness issue
  - Shared understanding of the (evolving) state of the problem solving process.
  - Need to share some facts and beliefs, and have shared understanding of the goals.
- Communication issue
  - Protocols that facilitate exchange of knowledge.
  - Multimodal interfaces.
  - As efficient and natural as possible for human users → spoken natural language.
  - As complete and unambiguous as possible for the agent.

- Personalization issue
  - Adaptation of agent's knowledge and behavior to the human user's strategies and preferences.
- Architecture issue
  - Design principles and methodologies for different types of roles and behaviors.
  - For instance:
    - Separate communication and control.
    - Guarantee asynchronous operation of agents.
    - Reuse of components
    - Uniform interface
- Evaluation issue
  - Compared to fully automated, fully manual, and alternative mixed-initiative systems.

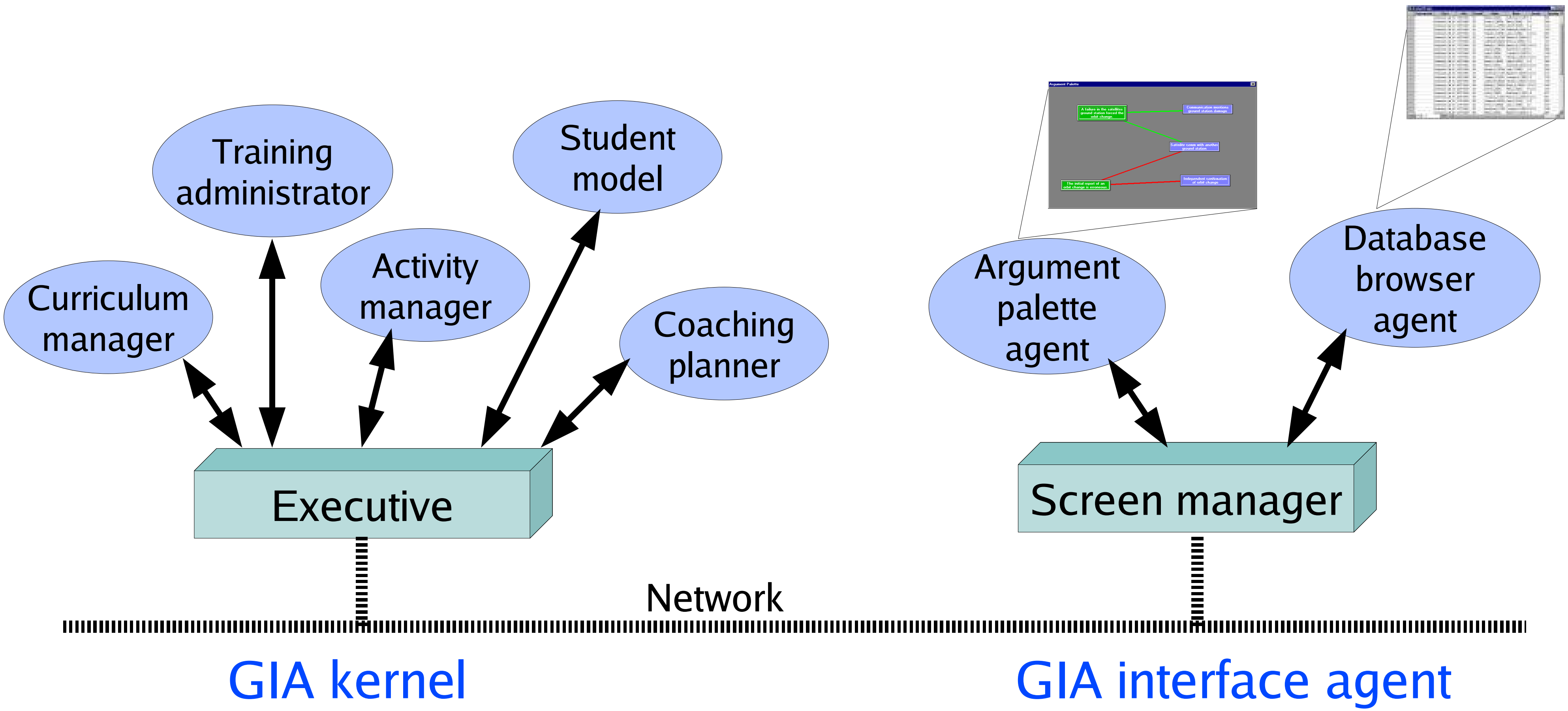
- Air traffic management system (Kinny *et al.*, 1996)
  - Assists human flow controller in determining landing time assignments on multiple runways.
  - Objectives: Meet the timetables, maximize runway usage.
  - Constraints: maintain safety and minimize congestion.
  - Real-time application in a complex and uncertain environment:
    - wind and weather
    - runway conditions
    - delayed departures and arrivals, etc.
  - The system has been tested in Sydney airport.

- (Air traffic management continued)
  - *Agents:*
    - *Coordinator creates and deletes aircraft agents.*
    - *Sequencer* communicates with the aircraft and the flow controller to determine landing time assignment.
    - *Wind model* maintains a 4-D model of wind conditions.
    - *Aircraft agents*

- *Aircraft agents*
  - Associated to a particular flight.
  - Exist while the aircraft is under the directions of the flow controller.
    - *Predictor* computes the expected time of arrival ← flight plan, wind conditions and performance profile.
    - *Monitor* compares radar data to ETA and notifies predictor and planner about possible deviations from ATA.
    - *Planner* derives a set of plans to allow the aircraft to land at the time determined by the sequencer → plan is sent to flow controller that chooses which plan to follow (LTA).
  - Plan consists of future trajectory, air-speed and altitude profile.
  - *Performance profile:* e.g., speed, climb performance, fuel burn.

- Intelligent tutoring systems
  - Active focus in research since 1970.
  - Systems for public education and industrial training.
  - Full-scale systems are few; take a lot of time and money to develop.
    - Custom software applications → each system built from scratch → shared tools do not exist.
    - Systems are not modular to support sharing and reuse.
    - Systems are not designed to evolve.
    - Hard to incorporate commercial software because of the lack of proper interfaces.
  - Conditions for a teaching environment to be effective:
    - Correct and logical guidance system that provides sensible feedback, demonstrations and explanations.
    - User interface is optimal and accessible.

- *Generic Instructional Architecture* (GIA)
  - Cheikes, 1995
  - Agent = "any piece of software that communicates by means of sending messages encoded in an Agent Communication Language (ACL)."
  - ACL =
    - *Protocol layer* defines the syntax and methods for information change.
    - *Content layer* contains the actual information being exchanged.
    - *Domain specific* ontology layer that grounds the expression in the content language.


Training administrator
Student model
Activity manager
Curriculum manager
Coaching planner
Executive
Argument palette agent
Database browser agent
Screen manager
Network
GIA kernel
GIA interface agent

17.10 Agent-based modeling

- To model global consequences of local behaviors and interactions of individuals = emergent behavior.
- Domains in which observational studies or controlled experimentation is difficult or impossible.
- Models consist of environment and population of agents.
  - Agents are defined by their behaviors and attributes.
  - Agents can be animals in an ecosystem, humans or households in social context, institutions in society or vehicles in traffic.
- Models can be spatially explicit, i.e., the agents are tied to a specific geographic or physical location.
- In some spatially explicit models agents can move.

- ABMs are relatively new in modelling.
- Traditional methods use system dynamic models, equation-based models, statistical methods, evolutionary models, cellular automata, etc.
- Example study on H*ow building a new power plant affects the ecosystem of a sea-bed*.
  - Traditionally in equation-based system:
    - Define properties of the seabed and the power plant as variables (e.g., emission rate, lead contamination level of fish, strength of currencies).
    - Write equations that determine the relationships between the variables.
    - Also the inputs and outputs are defined in equations.
  - ABMs
    - Model each actor in the system as an agent (e.g., each fish, plant, or plankton, and the power plant itself).
    - For each agent define its behavior, inputs and outputs.

17.11 Modeling domains for ABMs

- Ecological economics
  - Natural resource management
    - Common-pool resource (e.g., irrigation system, nature parks)
  - Collaboration under environmental risk
- Economics
  - Innovation diffusion
  - Games (e.g., ultimatum game, Prisoner's dilemma)
- Geography
  - Land-use and land-cover change
  - Migration and deforestation
  - Urban sprawl
  - Traffic models
- Archaeology & anthropology
  - Decay of ancient civilizations
  - Language evolution & language change

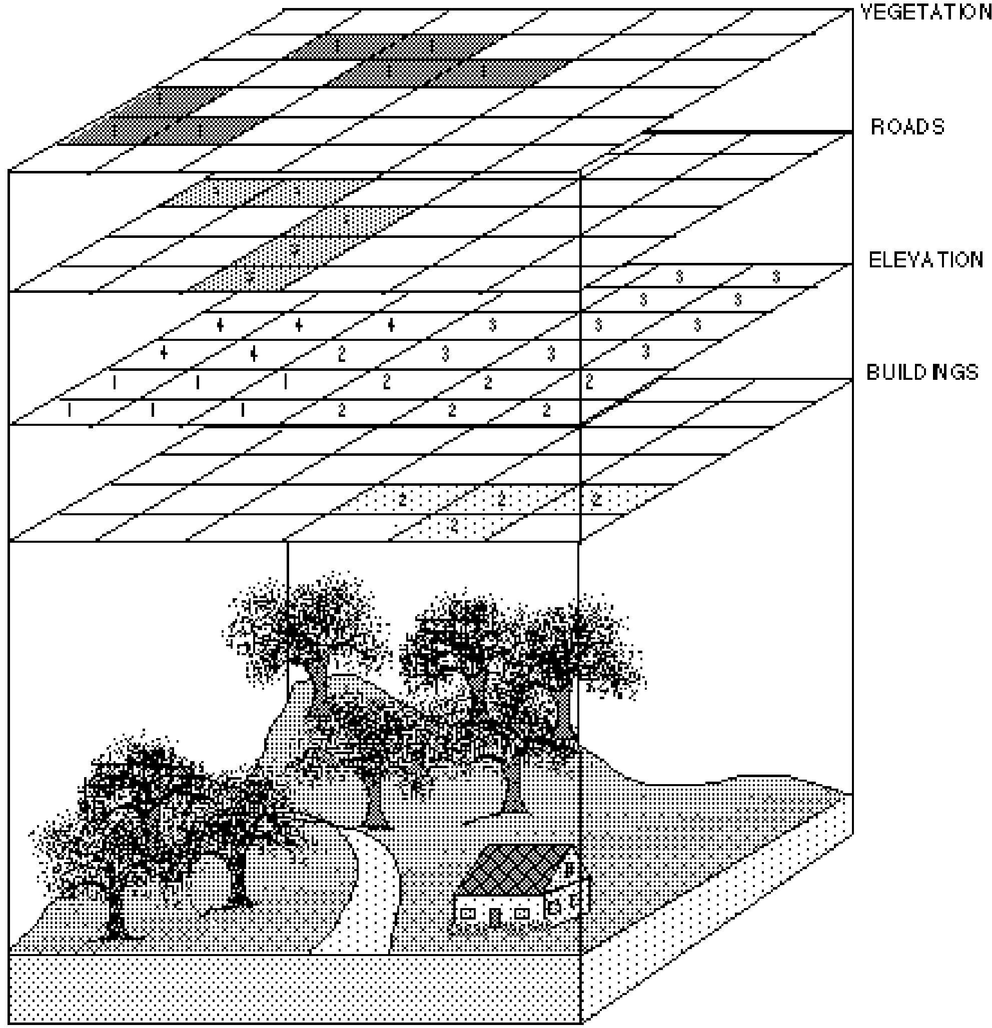
17.12 ABM of pedestrian traffic

- STREETS — a model of pedestrian behavior in urban areas (Schelhorn et al., 1999)
- Background
  - Knowing how people move important for understanding the way city centers work — indicator of the vitality and viability of the city center.
  - City planning, retail industry and emergency services are also interested in how people move in different parts of the center at different times of the day.
  - Pedestrian activity a product of
    - Configuration of street network
    - Location of attractions (shops, offices, public buildings, etc.)

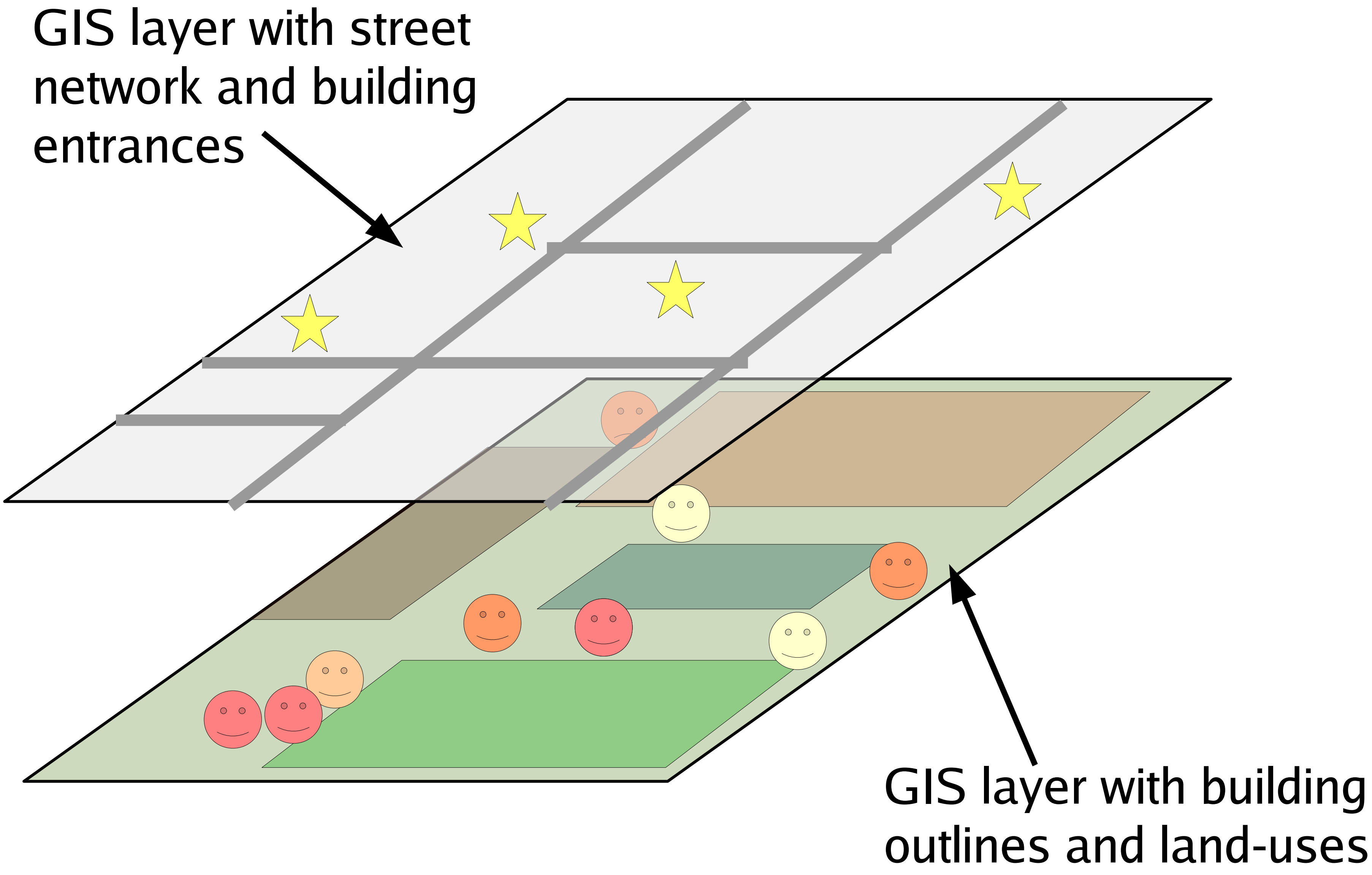
- Model proceeds in two phases:
  1. Use socio-economic data to populate the city area with pedestrian agents.
  2. Run the agent-based model to simulate the movement of population around the urban district.
     - ← Spatial configuration
     - ← Activity schedules
     - ← Distribution of land-uses
- Agent characteristics:
  - Socio-economic
    - Age and gender
    - Define the activity schedule: the sequence of locations the agent visits.
  - Behavioral
    - Speed, visual range and fixation
    - Defines how focused the agent is in following the schedule.

- Entering the agents to the simulation
  - At gateways representing car parks, bus and railway stations, and bus stops.
- Urban district
  - In geographic information system (GIS) format:
    - Building outlines with abstract land-use categories
    - Network data representing streets and building entrances
    - Walkability of spaces

Example of GIS raster representation:



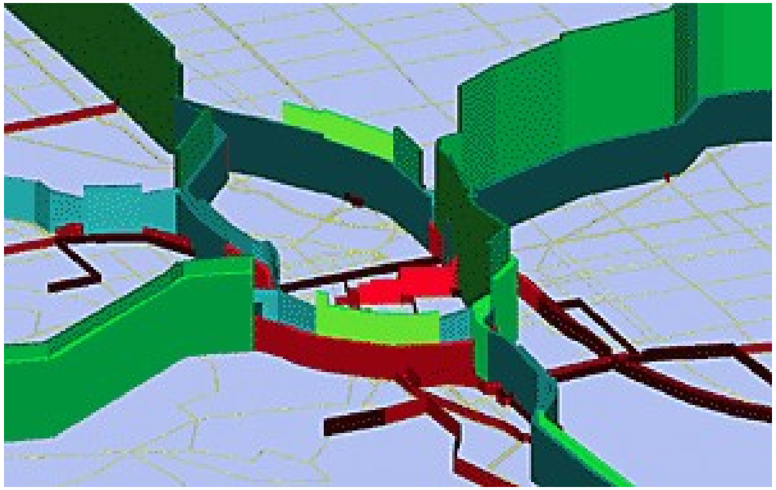
Sketch of the STREET model implementation



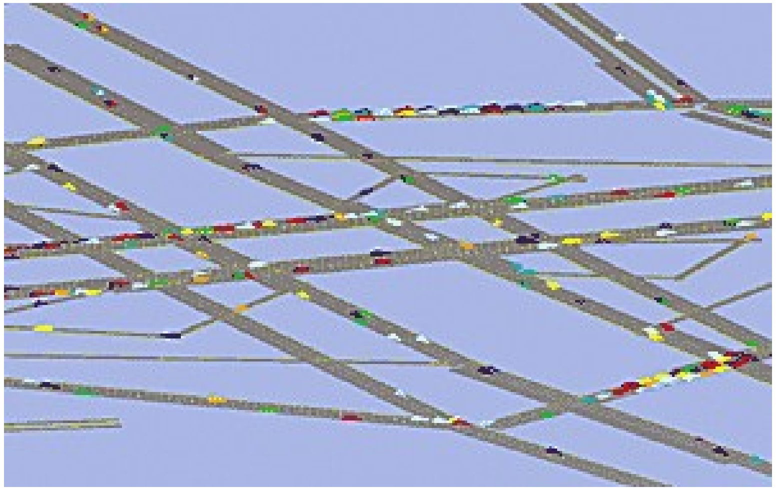
17.13 ABM of road traffic

- TRANSIMS (Beckman, 1997)
- Complete replication of a city's street system, households, travelers and vehicles.
- Enables:
  - Study of traffic patterns throughout the city for 24-hour periods.
  - Tracking of individual travelers.
- Ultimate goals:
  - Study environmental impact of changes in traffic patterns, i.e., congestion due to road construction or accidents.
  - Environmental impact measured in air-pollution levels.
- Applied to Albuquerque, New Mexico, and Helsinki and other major Finnish cities.

Pictures from TRANSIMS simulator



The height of the bars reflect the relative density of traffic on those roads.



Tracking of individual vehicle's movements is based on analysis of simulated populations and data collected from census or other regional databases.