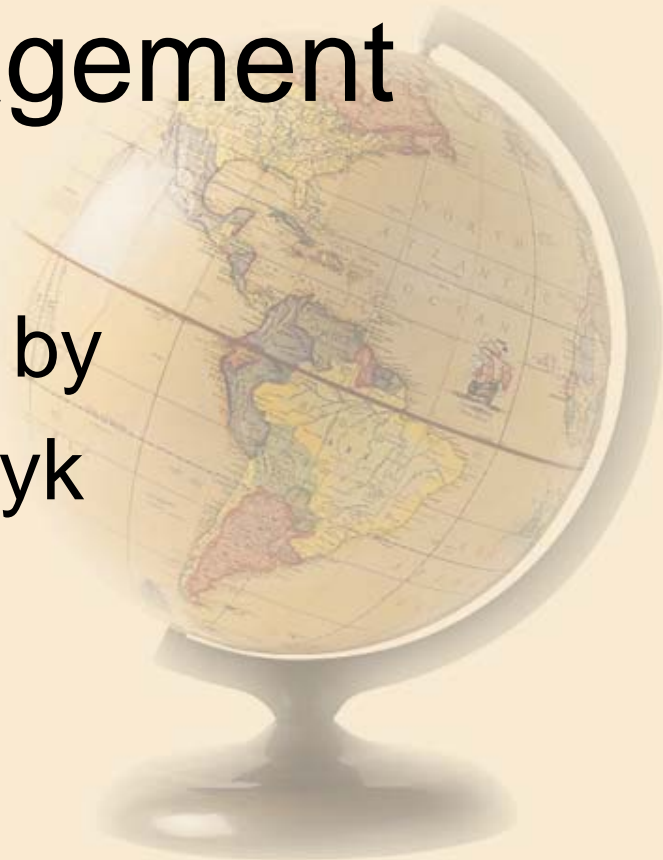


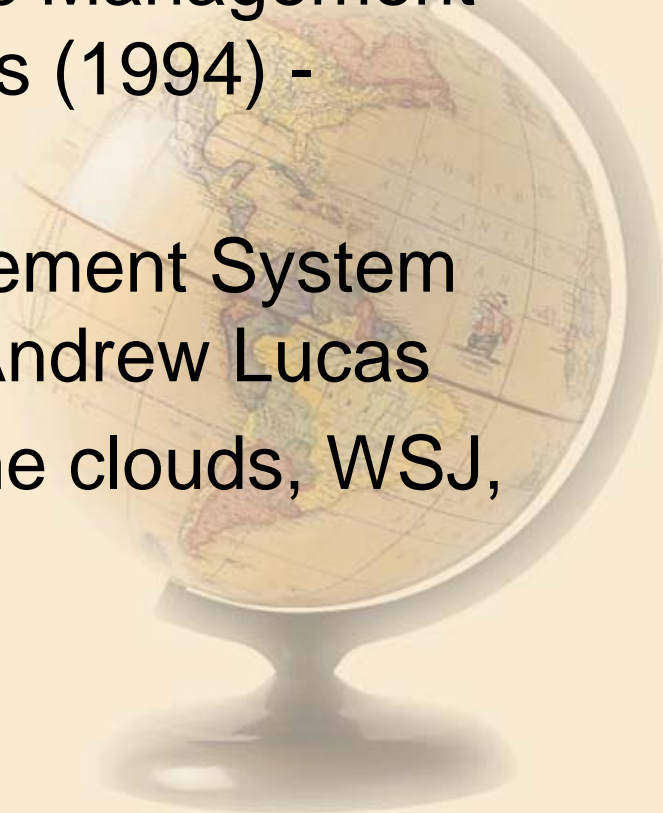
**BDI Agents:
From Theory to Practice
Implementation examples in
Air Traffic Management**

Paper discussion by
Andrzej Borowczyk



Papers discussed

- BDI Agents: From Theory to Practice (1995)
Anand S. Rao, Michael P. Georgeff
- New Techniques for Air Traffic Management for Single and Multiple Airports (1994) -
Andrew Lucas et. al.
- The OASIS Air Traffic Management System (1992) - Magnus Ljungberg, Andrew Lucas
- Help for calculating costs in the clouds, WSJ, March 7, 2007



Agenda

- BDI Belief Desire Intention
- ATM Air Traffic Management
- OASIS Optimal Aircraft Sequencing using Intelligent Scheduling
- Route-planning and cutting costs in aviation



BDI

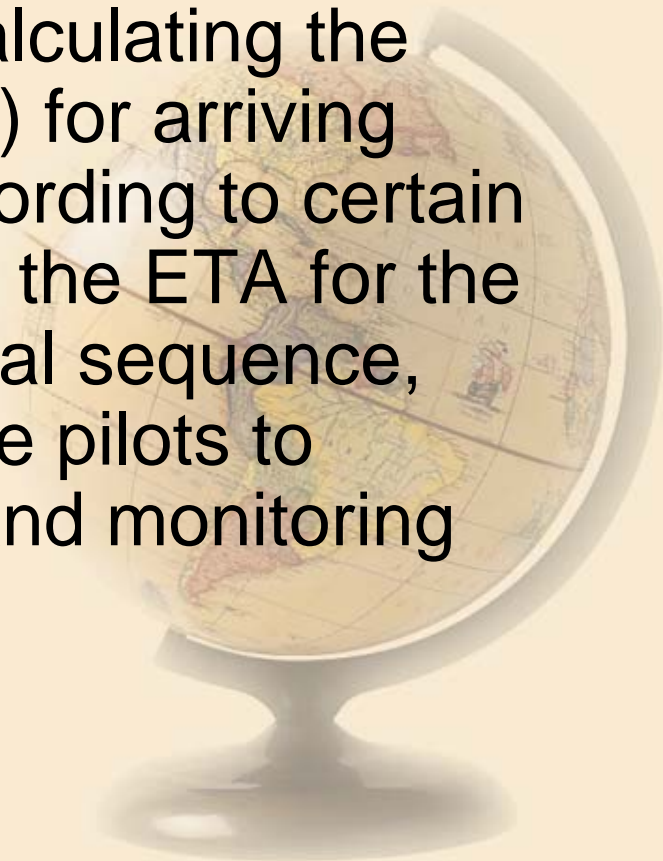


BDI

- **Mental attitudes:**
 - Belief – the information an agent has about its surrounding environment
 - Desire – the aim of the agent
 - Intention (or goals) – the objective that the agent is trying to achieve

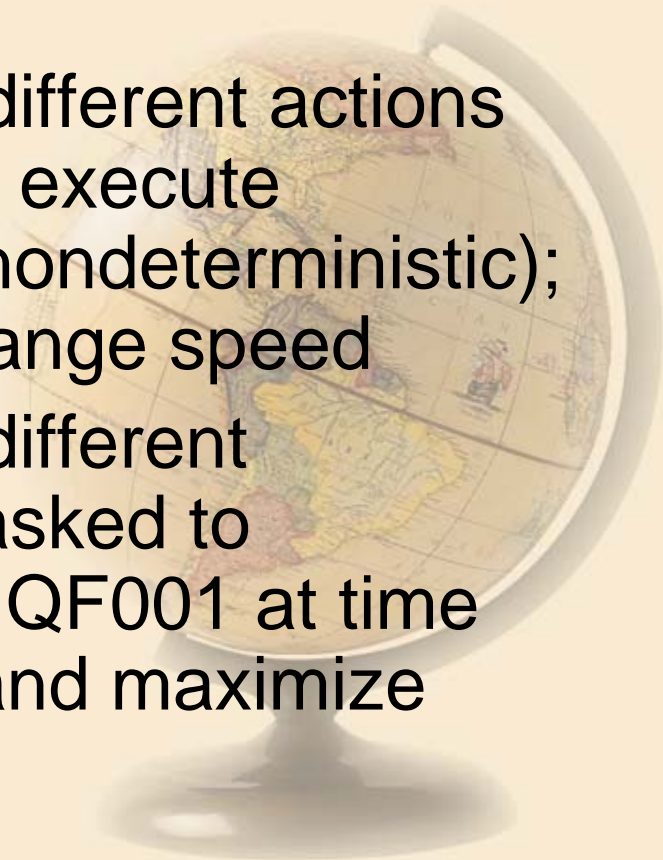


- The necessity of beliefs, desires, and intentions for a system to act appropriately in a class of application domains characterized by various practical limitations and requirements
- Design of an air traffic management system that is to be responsible for calculating the expected time of arrival (ETA) for arriving aircraft, sequencing them according to certain optimality criteria, reassigning the ETA for the aircraft according to the optimal sequence, issuing control directives to the pilots to achieve the assigned ETAs, and monitoring conformance

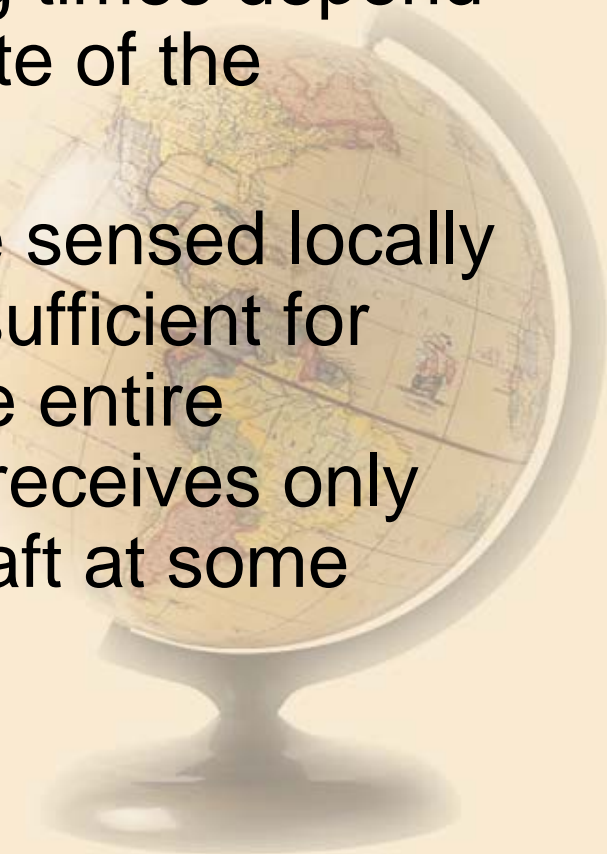


At any instant of time:

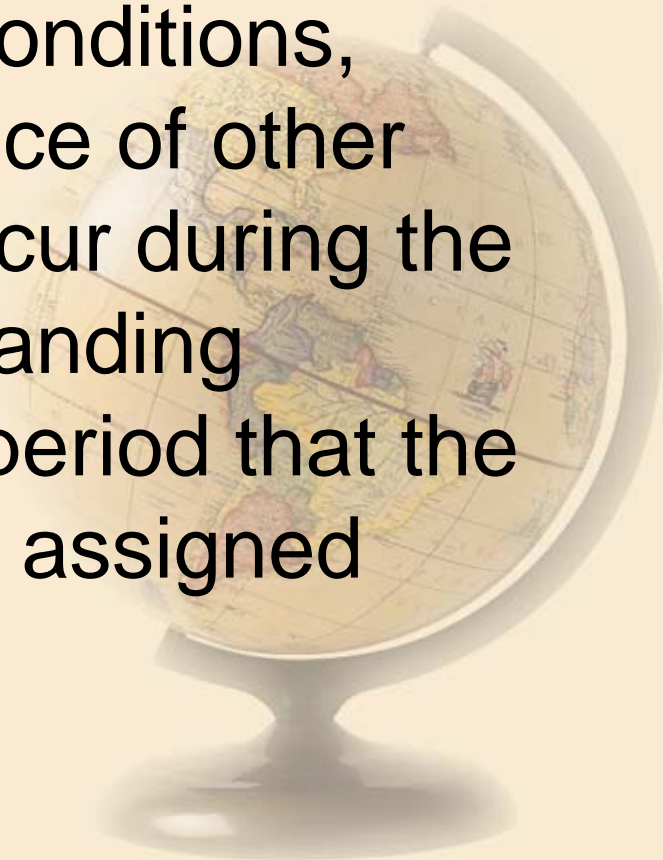
- 1. there are potentially many different ways in which the environment can evolve (formally, the environment is nondeterministic); e.g., the wind field can change over time in unpredictable ways
- 2. there are potentially many different actions or procedures the system can execute (formally, the system itself is nondeterministic); e.g., requesting an aircraft change speed
- 3. there are potentially many different objectives that the system is asked to accomplish; e.g., land aircraft QF001 at time 19:00, land QF003 at 19:01, and maximize runway throughput,



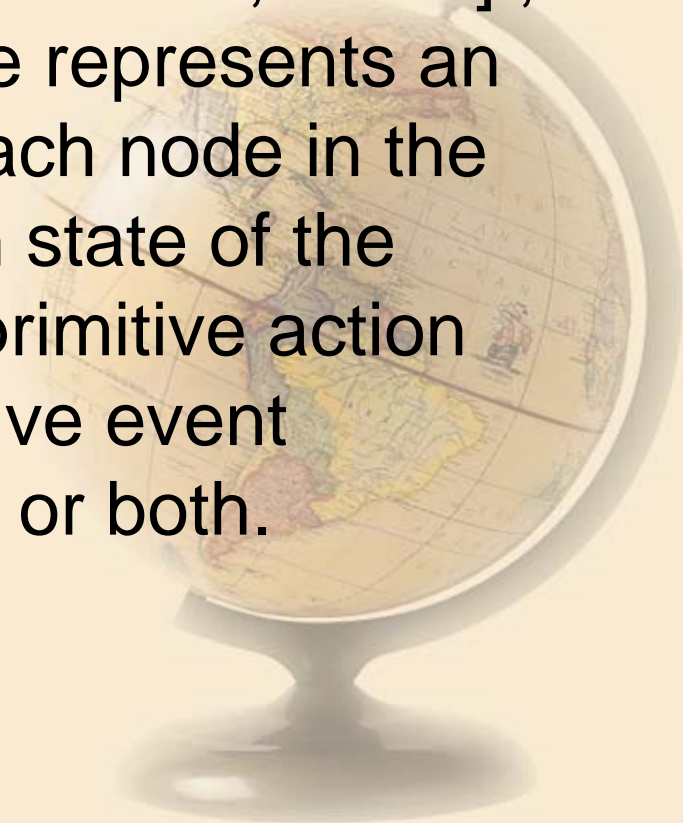
- 4. The actions or procedures that (best) achieve the various objectives are dependent on the state of the environment (context) and are independent of the internal state of the system; e.g., the actions by which the aircraft achieve their prescribed landing times depend on wind field, but not on the state of the computational system.
- 5. The environment can only be sensed locally (i.e., one sensing action is not sufficient for fully determining the state of the entire environment); e.g., the system receives only spot wind data from some aircraft at some times at some locations



- 6. The rate at which computations and actions can be carried out is within reasonable bounds to the rate at which the environment evolves; e.g., changes in wind field, operational conditions, runway conditions, presence of other aircraft, and so on, can occur during the calculation of an efficient landing sequence and during the period that the aircraft is flying to meet its assigned landing time.

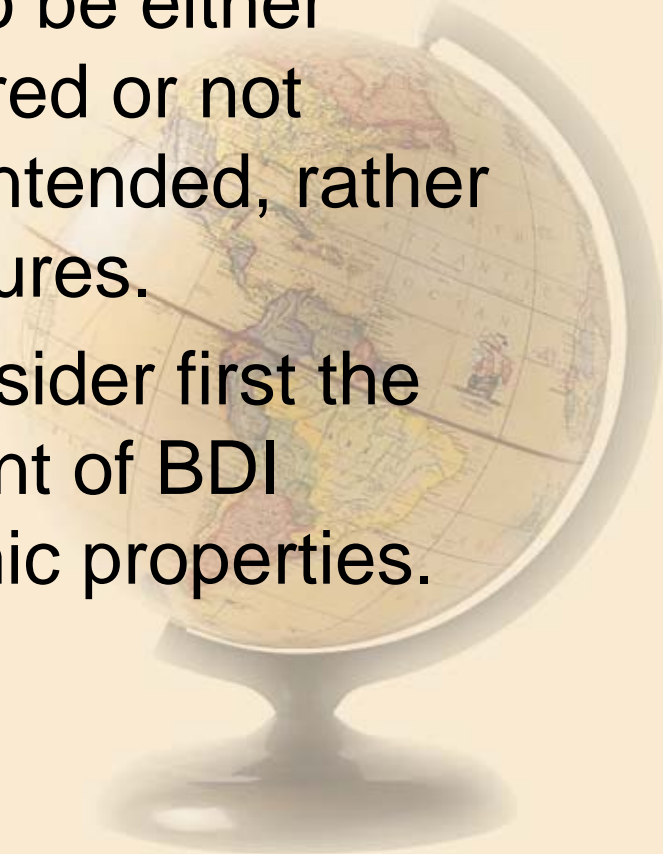


- One way of modelling the behaviour of such a system, given Characteristics (1) and (2), is as a branching tree structure [Emerson, 1990] , where each branch in the tree represents an alternative execution path. Each node in the structure represents a certain state of the world, and each transition a primitive action made by the system, a primitive event occurring in the environment, or both.



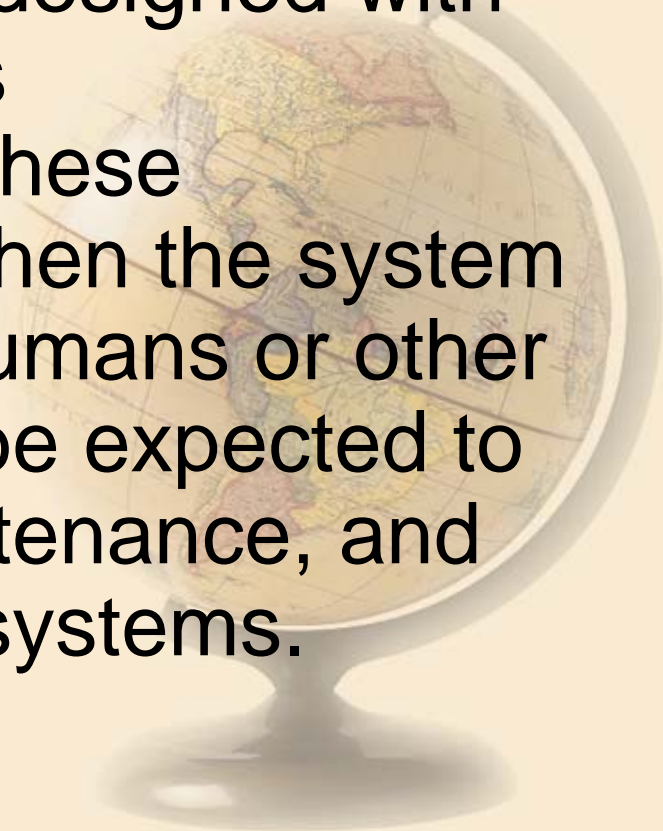
BDI Logics

- Abstract model reducing probabilities and payoffs to dichotomous (0-1) values. That is, propositions are considered to be either believed or not believed, desired or not desired, and intended or not intended, rather than having continuous measures.
- Within such a framework, consider first the static properties we would want of BDI systems and next their dynamic properties.



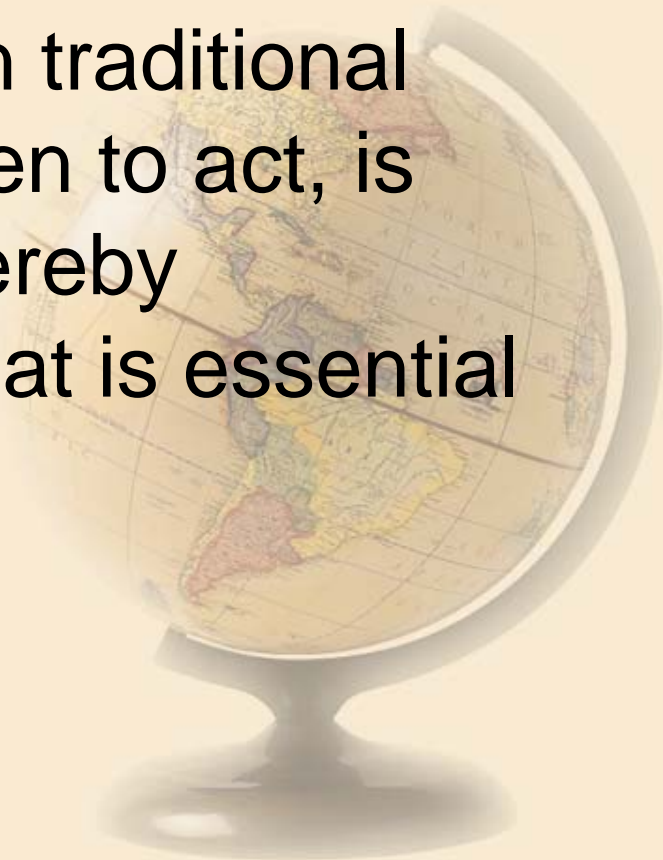
Abstract architecture

- While it is not necessary that a system that is specified in terms of beliefs, desires and intentions be designed with identifiable data structures corresponding to each of these components, it is useful when the system must communicate with humans or other software agents and can be expected to simplify the building, maintenance, and verification of application systems.



Abstract architecture

- the architecture cannot be simply based on a traditional theorem-proving system
- the time taken to reason in traditional way and thus the time taken to act, is potentially unbounded, thereby destroying the reactivity that is essential to an agent's survival



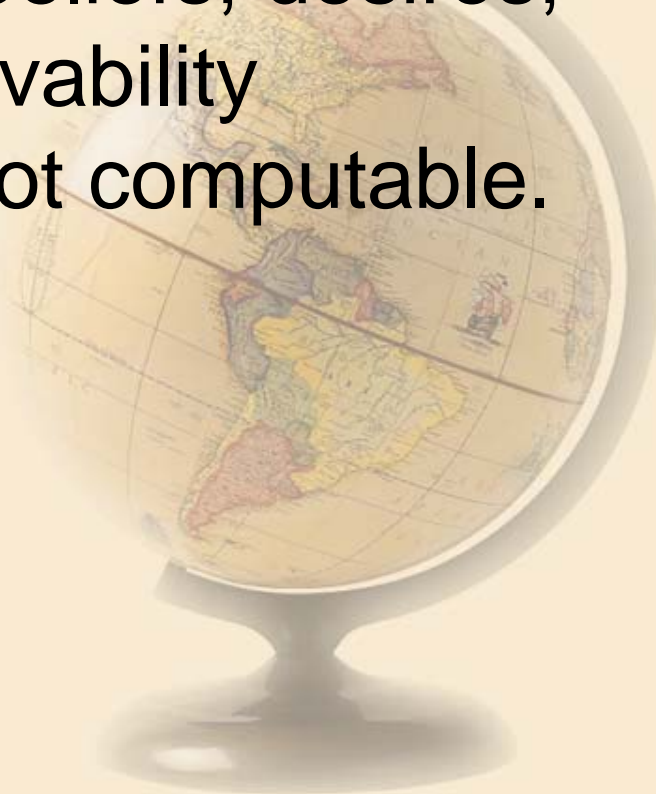
Abstract architecture

- BDI-interpreter
- initialize-state();
- repeat
- options := option-generator(event-queue);
- selected-options := deliberate(options);
- update-intentions(selected-options);
- execute();
- get-new-external-events();
- drop-successful-attitudes();
- drop-impossible-attitudes();
- end repeat



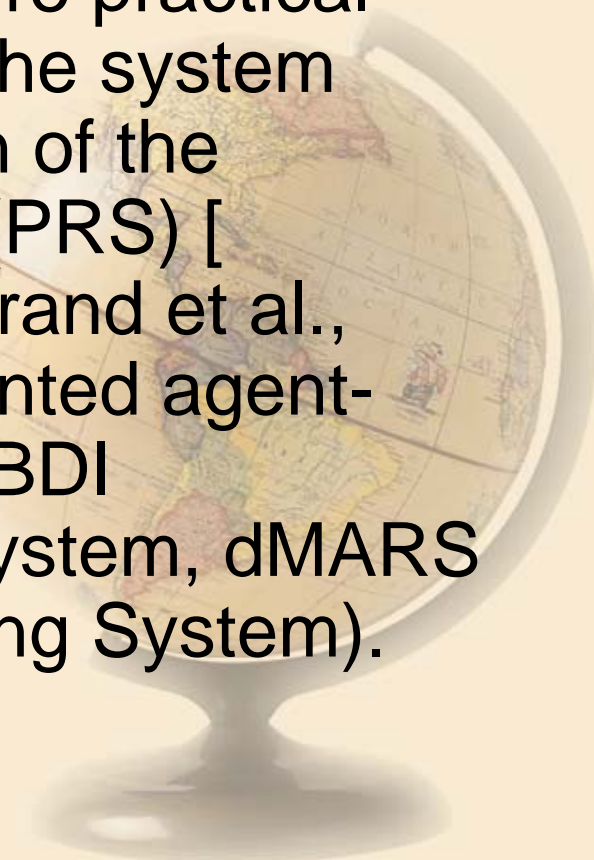
Abstract architecture

- It is not a practical system for rational reasoning. The architecture is based on a (logically) closed set of beliefs, desires, and intentions and the provability procedures required are not computable.



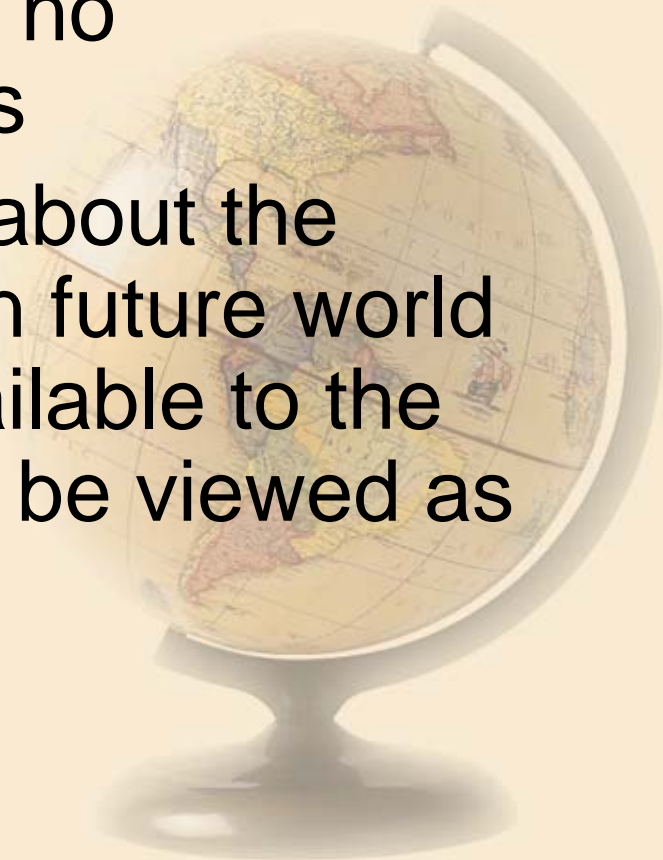
Abstract architecture

- therefore make a number of important choices of representation which, while constraining expressive power, provide a more practical system for rational reasoning. The system presented is a simplified version of the Procedural Reasoning System (PRS) [Georgeff and Lansky, 1986; Ingrand et al., 1992], one of the first implemented agent-oriented systems based on the BDI architecture, and a successor system, dMARS (distributed MultiAgent Reasoning System).



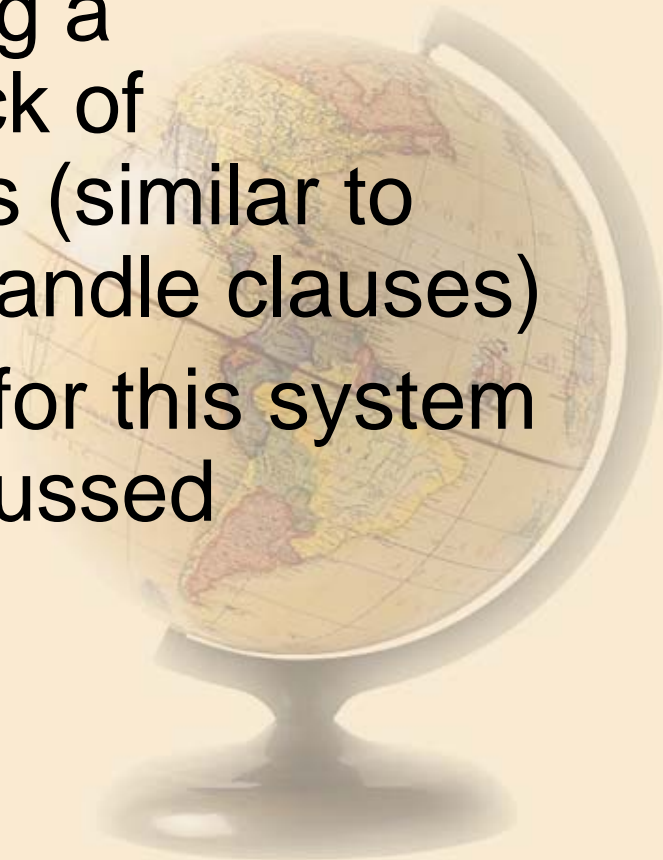
Abstract architecture

- represent only beliefs about the current state of the world and consider only ground sets of literals with no disjunctions or implications
- represent the information about the means of achieving certain future world states and the options available to the agent as plans, which can be viewed as a special form of beliefs



Abstract architecture

- each intention that the system forms by adopting certain plans of action is represented implicitly using a conventional run-time stack of hierarchically related plans (similar to how a Prolog interpreter handle clauses)
- The main interpreter loop for this system is identical to the one discussed previously.



Abstract architecture

- the procedures appearing in the interpreter must be fast enough to satisfy the real-time demands placed upon the system. One way of tailoring and thus improving the process of option generation is to insert an additional procedure, post-intention-status, at the end of the interpreter loop in order to delay posting events on the event queue regarding any changes to the intention structure until the end of the interpreter loop



Air Traffic




Air Traffic

- Air traffic worldwide continues to grow and so does congestion
- It was already being predicted in the 90's (1994, Andrew Lucas)
- International Air Transport Association (IATA) undertook a study on European air traffic congestion



IATA study result

- 1. Air traffic congestion and the resulting constraints on growth will cause annual losses of US\$10 billion to European national economies by the year 2000;
 - 2. Sufficient airspace capacity exists to accommodate the expected growth in air traffic until 2000, but only if it is used efficiently; and
 - 3. At least 10 major European airports will be seriously congested between 1995 and 2000 unless their capacity is enhanced (by expansion of existing infrastructure or more efficient air traffic management).
- 

- Many of Asia's airports are already congested, Hong Kong's Kai Tak airport runway is now at 90% capacity during peak periods.
- This growth will add further to the present levels of congestion and resulting delay.
- The challenge is to match the demand with the available capacity.
- The alternatives are to increase capacity, restrict demand or achieve more efficient utilization of the existing infrastructure. Increasing capacity requires enormous capital expenditure, for example the US\$14 billion cost of Kansai.



- The European Civil Aviation Conference (ECAC) estimated that an increase in the capacity of up to 30% can be achieved through more efficient traffic management.
- The Lincoln Laboratory, MIT, has demonstrated, through a computer simulation of aircraft arriving at Boston Logan Airport, that an increase of at least 13% in terminal throughput is achievable through efficient sequencing.

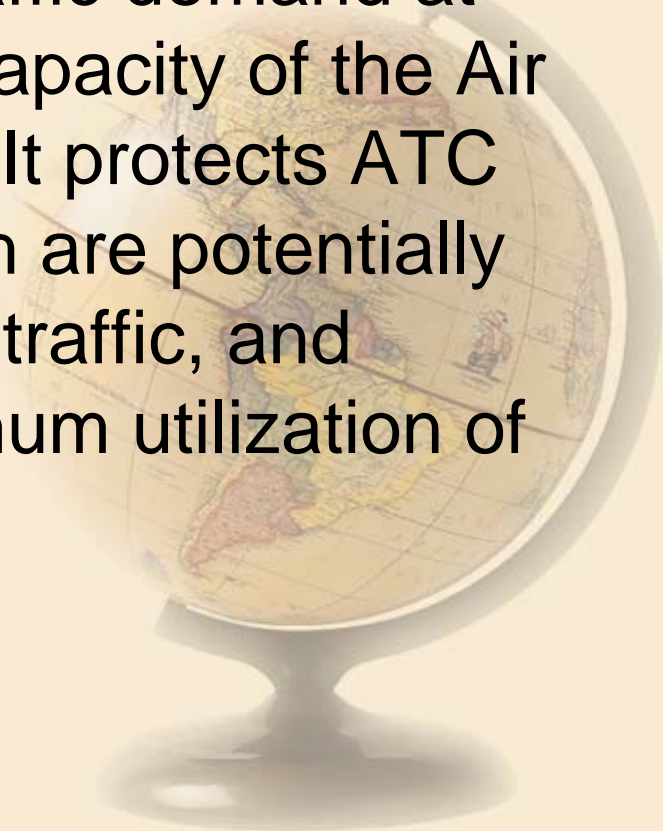


- Commercial imperatives mean that currently available infrastructure must be made to work as efficiently as possible to fully exploit its potential capacity. Similarly, any new infrastructure must operate at optimal efficiency as soon as it is brought on stream.
- In response, governments and aviation authorities world wide are investing in new air traffic control (ATC) and air traffic management (ATM) systems.



Air Traffic Management

- Air traffic management is concerned with ensuring an optimum flow of air traffic to or through areas within which traffic demand at times exceeds the available capacity of the Air Traffic Control (ATC) system. It protects ATC from overload situations which are potentially dangerous to the safety of air traffic, and should also ensure the maximum utilization of ATC capacity in all situations.



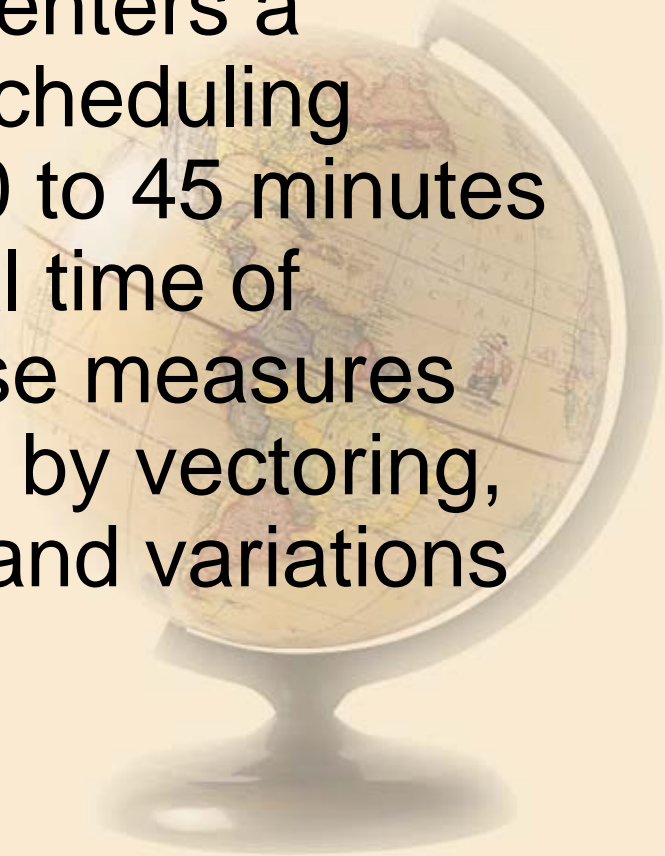
ATM's four main functions

- **Strategic Planning:** long term projection of traffic demand and then structuring the traffic to match the available capacity of the ATC system
- **Pre-Tactical and Tactical Planning:** analyses the available capacity of the departure, enroute, and arrival airspace, and takes tactical measures if the predicted demand exceeds the available capacity. The tactical measures include restrictions in flight profile, slot allocation for arrival or departure airports and re-routing of traffic.



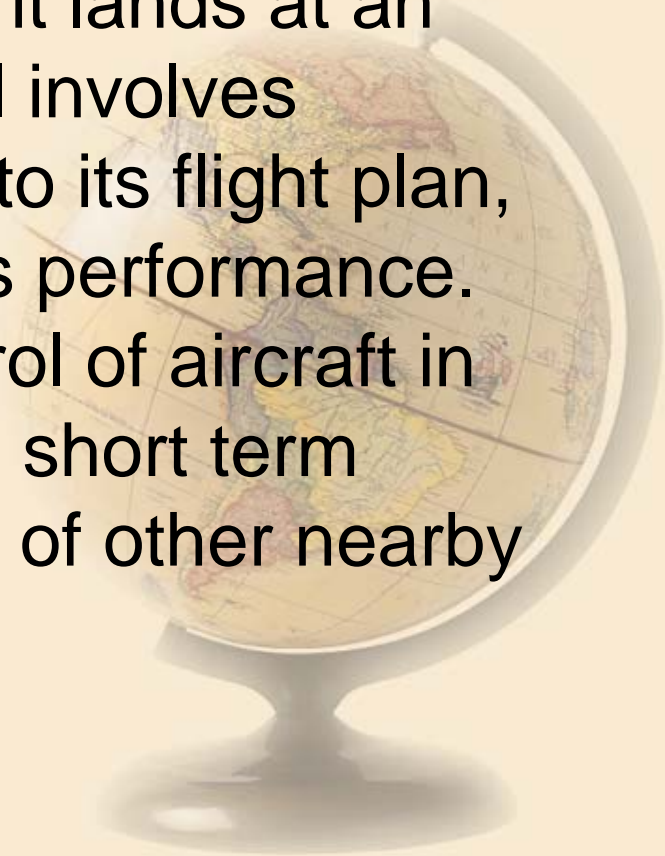
ATM's four main functions

- **Short Term Planning:** This involves conflict resolution measures performed 45 minutes before a flight enters a control sector or optimal scheduling strategies implemented 20 to 45 minutes before the expected arrival time of aircraft at the airport. These measures include re-routing of traffic by vectoring, speed control advisories, and variations to the normal flight profile.



ATM's four main functions

- **Monitoring and Control:** This involves the continuous monitoring of the aircraft from the time it becomes airborne until it lands at an airport. Monitoring and control involves verifying if an aircraft is flying to its flight plan, and if necessary, modifying its performance. Thus the monitoring and control of aircraft in one sector may well affect the short term planning and tactical planning of other nearby sectors.

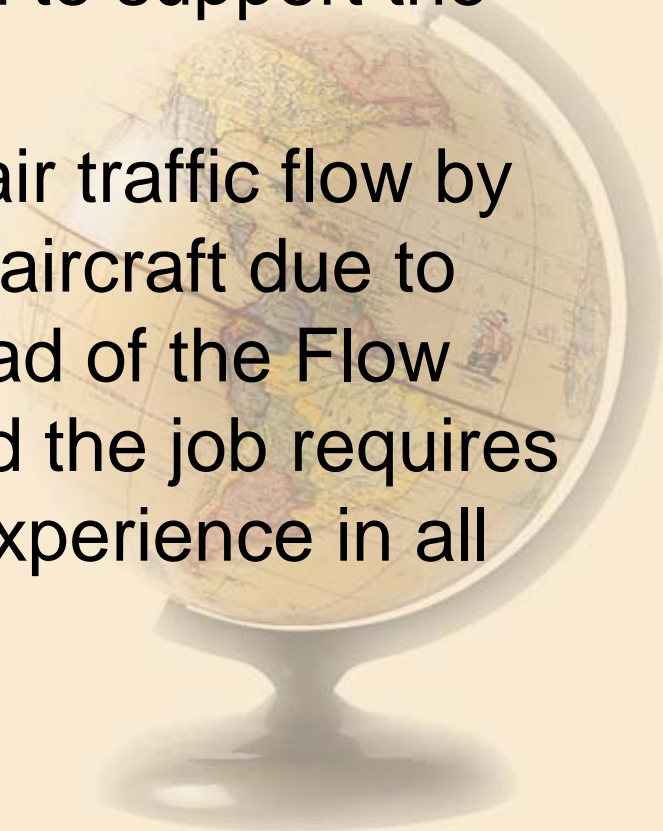


OASIS



OASIS

- OASIS (Optimal Aircraft Sequencing using Intelligent Scheduling) is a real-time artificial intelligence system developed to support the Flow Director.
- Flow Director - regulates the air traffic flow by speeding up or slowing down aircraft due to land at an airport. The workload of the Flow Director is extremely high, and the job requires a very high level of skill and experience in all aspects of air traffic control.



Flow Director

- Flow directors in "Pushing Tin" (1999)



- The design of OASIS is agent-oriented: the major components of the system are independent agents, each solving a part of the overall problem. The system's flexibility results from this co-operative problem solving approach.



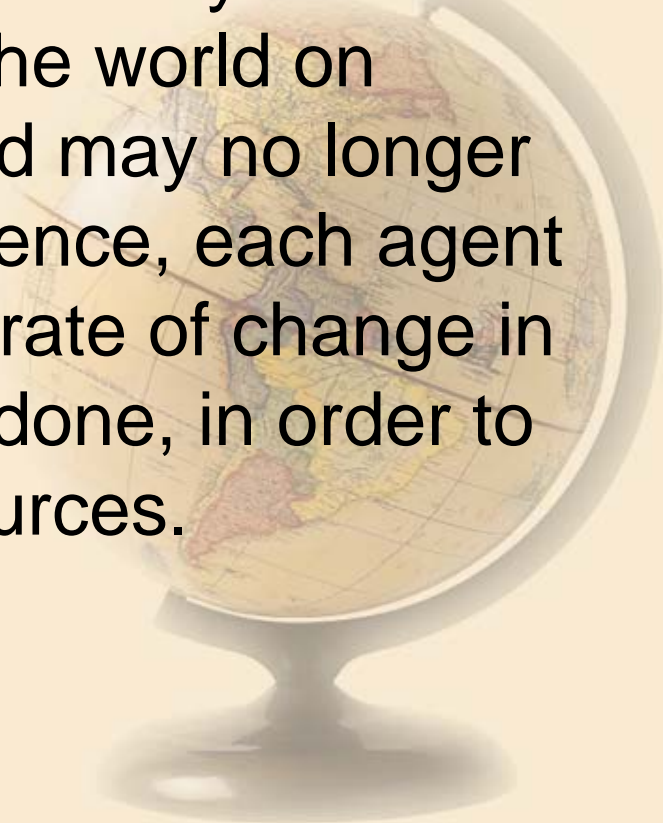
Conceptual Design of OASIS

- OASIS has been designed by sub-dividing the air traffic management task into its major parts and designing separate agents to solve each of those sub-problems. Each agent solves its part of the task independently, and cooperate with the others to produce the overall system behavior.
- Agents communicate with each other and with the environment using messages. Messages are sent and received asynchronously, and are assumed to have assured delivery. However, no guarantees are given or assumed about the processing of the message once it has reached its recipient.

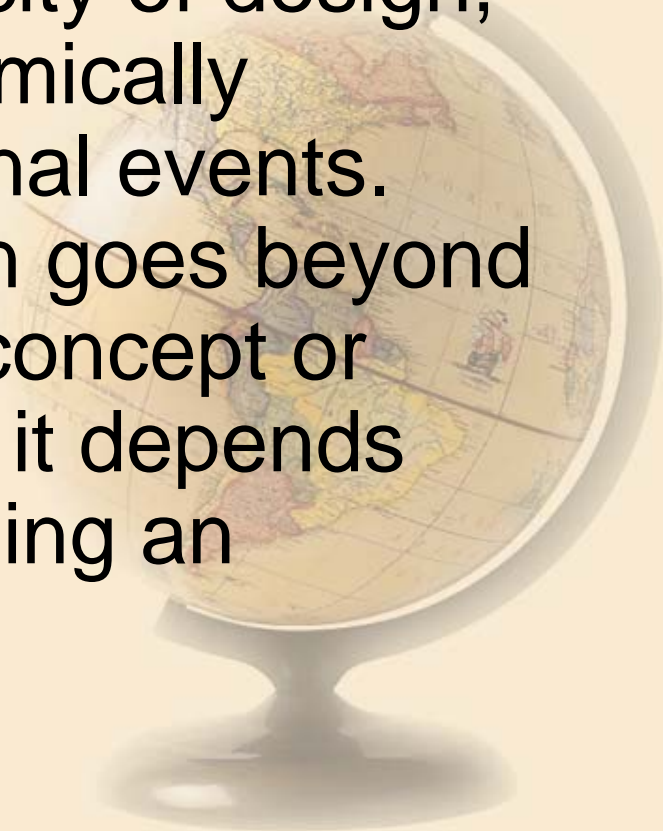
- Agents have integrity. Facts, goals, and intentions that are part of the internal state of the agent cannot be manipulated from the outside.
- The internal state of an agent is private. The only way for an agent to find out the belief, goal or intention of another agent is to send a message to that agent asking for the information.



- As the environment changes, agents must decide how to act. If that deliberation continues for too long, the agent may find that the facts, goals, and state of the world on which the deliberation is based may no longer reflect the current situation. Hence, each agent must be able to reflect on the rate of change in the world and the tasks to be done, in order to effectively use its limited resources.



- This design enables us to tailor each individual agent to the sub-problem it is solving. It allows for simplicity of design, high robustness and dynamically variable reactivity to external events. This agent-oriented design goes beyond the traditional subroutine concept or object-oriented design, as it depends crucially on each agent being an autonomous reasoner



OASIS is designed using two classes of agents

- First, those that handle inter-aircraft coordination and reasoning, called global agents;
- Second, those that perform computation or reasoning relevant to each aircraft individually, called aircraft agents.



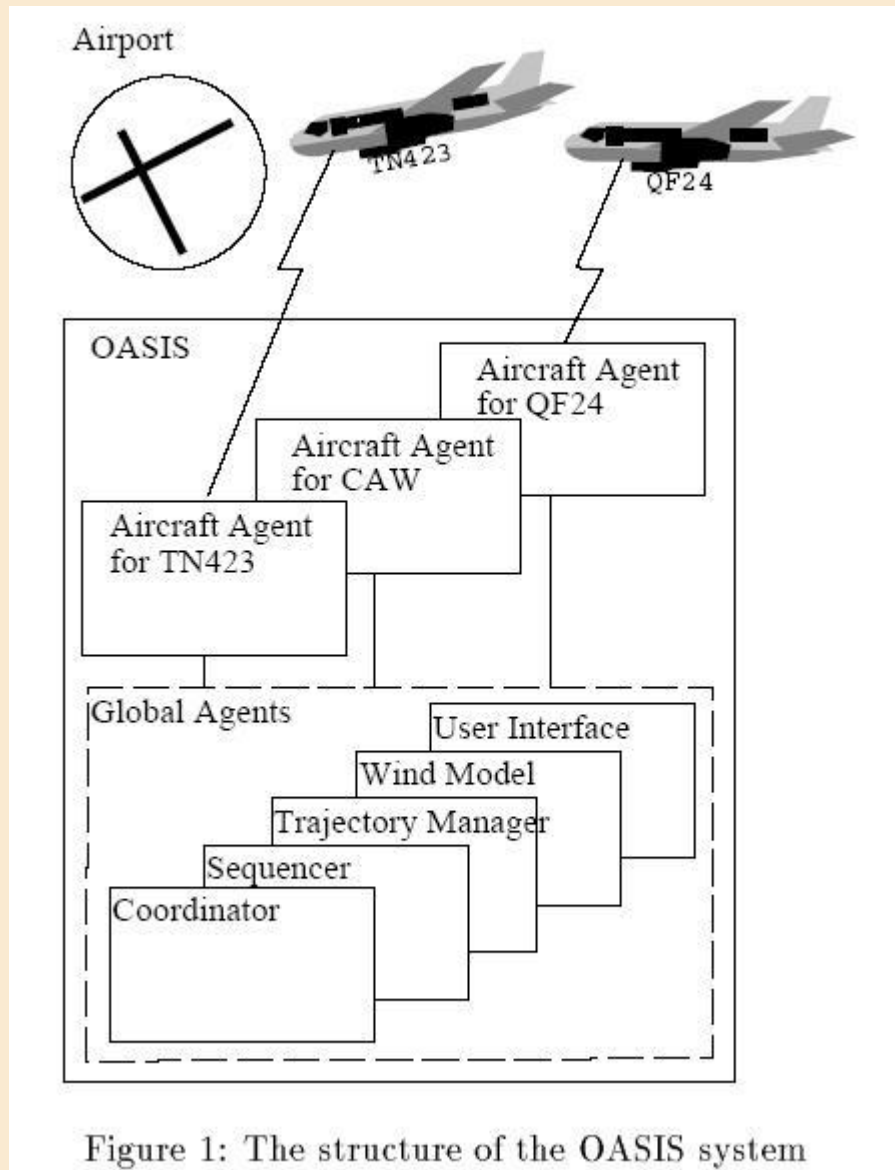


Figure 1: The structure of the OASIS system

Five global agents in OASIS

- **COORDINATOR** - serves as the task manager, co-ordinating the activities of the other global and aircraft agents
- **SEQUENCER** - uses search techniques to arrange the aircraft in a least delay/cost sequence
- **TRAJECTORY CHECKER** - verifies that instructions proposed by the system do not cause aircraft to violate statutory separation requirements



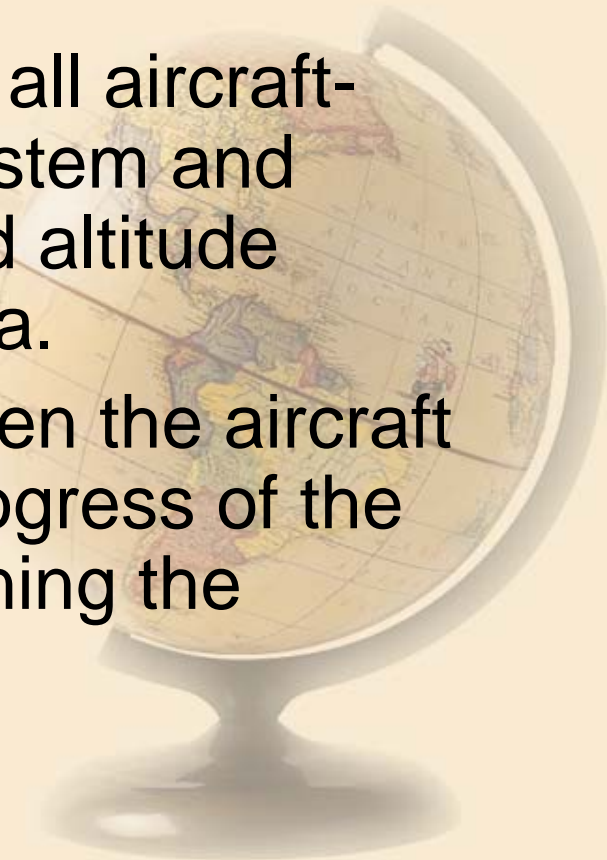
Five global agents in OASIS

- **WIND MODEL** - uses wind observations made by individual aircraft agents for predicting the wind field that aircraft are likely to encounter
- **USER INTERFACE** agent serves as the single point of communication with the Flow Director, managing all user interactions.



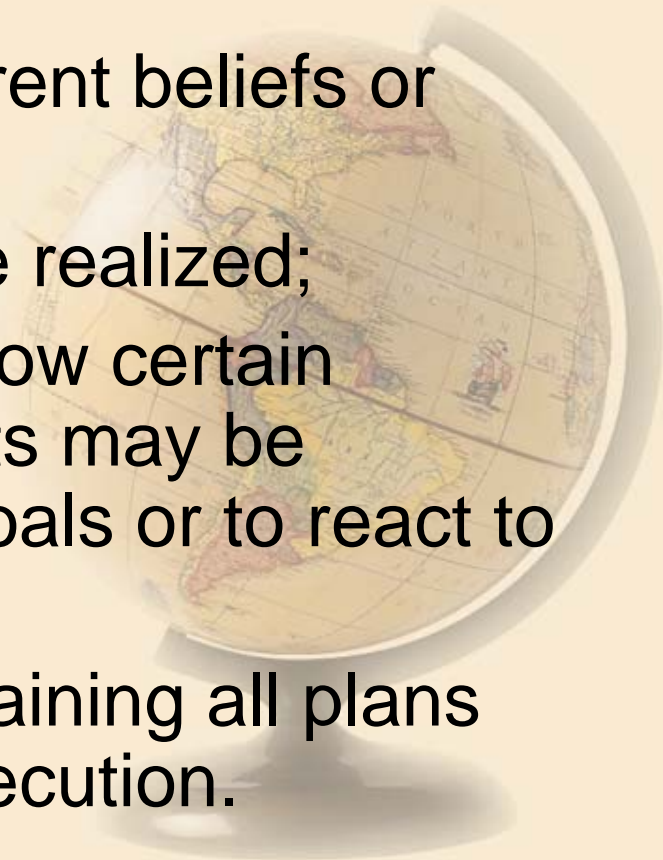
Aircraft agents in OASIS

- The system assigns an agent to each approaching aircraft intending to land at the airport.
- The AIRCRAFT agents contain all aircraft-specific data required by the system and assimilates position, speed, and altitude reports from real-time radar data.
- AIRCRAFT agent estimates when the aircraft will land, monitoring that the progress of the aircraft is as planned, and planning the trajectory of the aircraft.



Procedural Reasoning in OASIS

- PRS is designed to be used as a situated real-time reasoning system. As shown in Figure 2, PRS has:
- (1) a database containing current beliefs or facts about the world;
- (2) a set of current goals to be realized;
- (3) a set of plans describing how certain sequences of actions and tests may be performed to achieve given goals or to react to particular situations; and
- (4) an intention structure containing all plans that have been chosen for execution.



Procedural Reasoning in OASIS

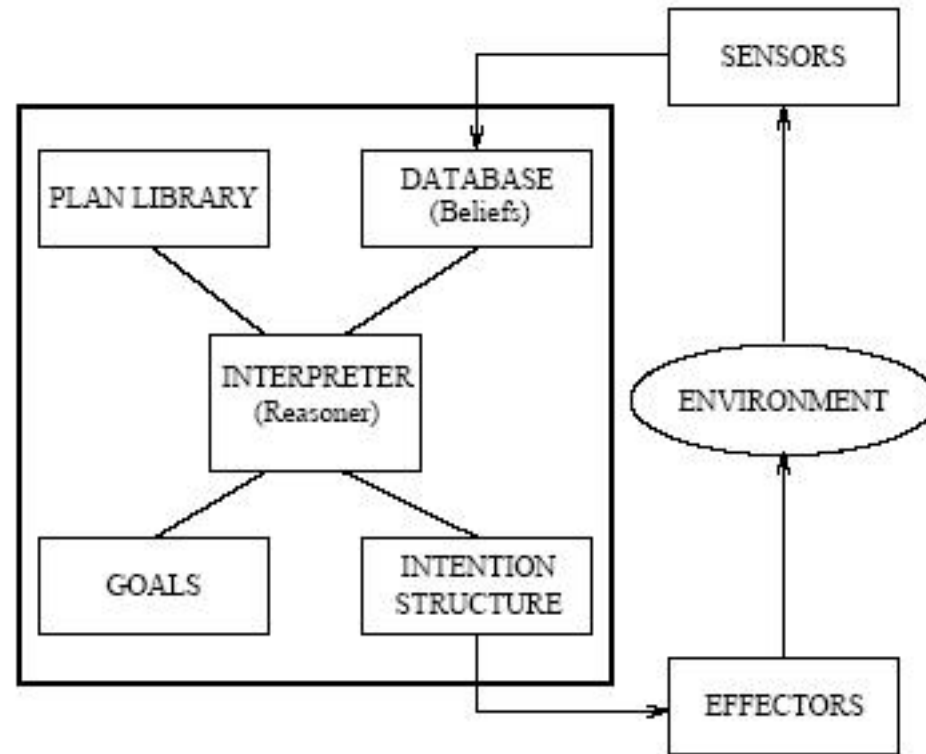
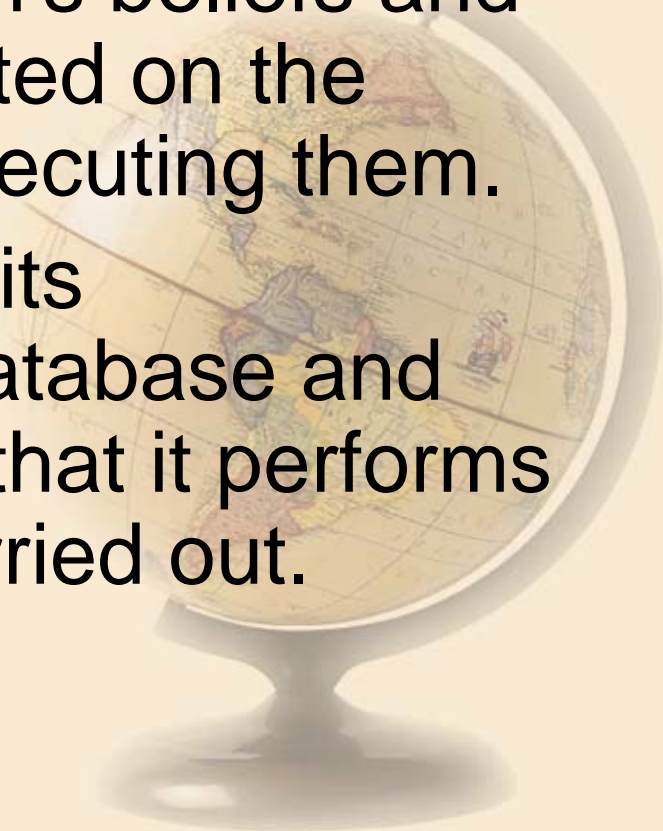


Figure 2: The structure of the Procedural Reasoning System

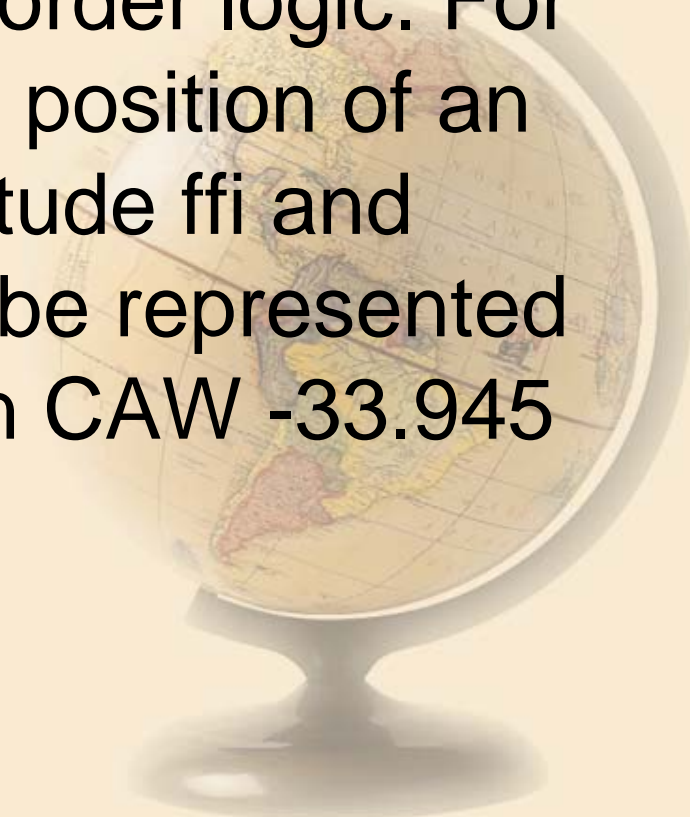
Procedural Reasoning in OASIS

- An interpreter manipulates these components by selecting appropriate plans based on the system's beliefs and goals, placing those selected on the intention structure, and executing them.
- The system interacts with its environment through its database and through the basic actions that it performs when its intentions are carried out.



Goals and Beliefs in OASIS

- The beliefs of PRS provide information on the state of the environment. They are represented in a first-order logic. For example, the fact that the position of an aircraft called CAW is latitude ϕ and longitude λ can be represented by the statement (position CAW (ϕ, λ)).

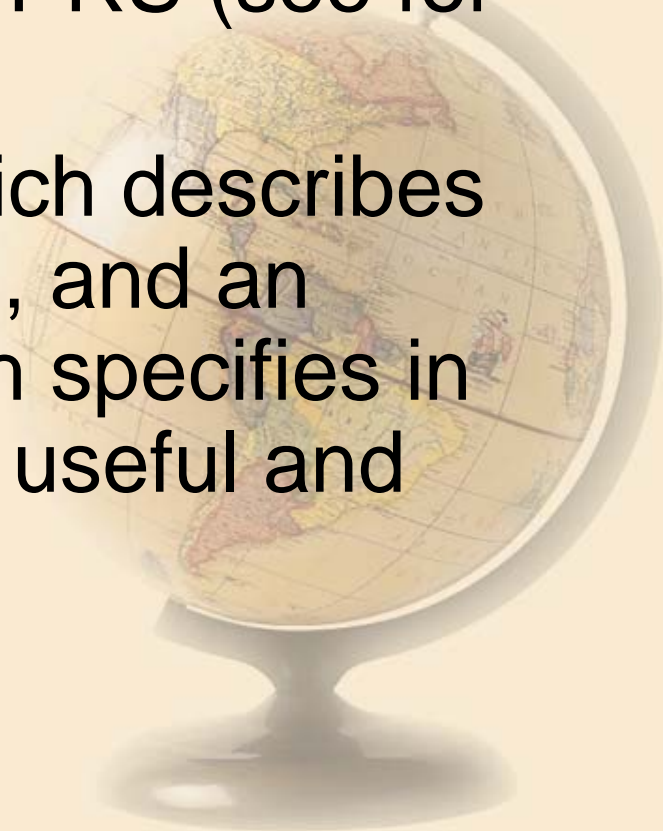


Goals and Beliefs in OASIS

- The goals of PRS are descriptions of desired tasks or behaviors. In the logic used by PRS, the goal to achieve a certain condition C is written $(! C)$; to test for the condition is $(? C)$; and to wait until the condition is true is $(“ C)$. For example, the goal to decrease the speed of an aircraft, CAW, to 150 knots could be expressed as $(! (\text{reduced-speed CAW } 150))$; to test for it could be expressed as $(? (\text{reduced-speed CAW } 150))$; and to wait for the condition to become true could be expressed as $(“ (\text{reduced-speed CAW } 150))$.

Plans in OASIS

- Knowledge about how to accomplish given goals or react to certain situations is represented by plans in PRS (see for example, Figure 3)
- Each plan has a body, which describes the steps of the procedure, and an invocation condition, which specifies in what situations the plan is useful and applicable



Plans in OASIS

Name: Verify Aircraft Progress

Invocation: (*fact (arrived \$checkpoint \$time))

Context: (and (*fact (est-arr-time \$checkpoint \$est))
(*fact (arrival-time-threshold \$limit)))

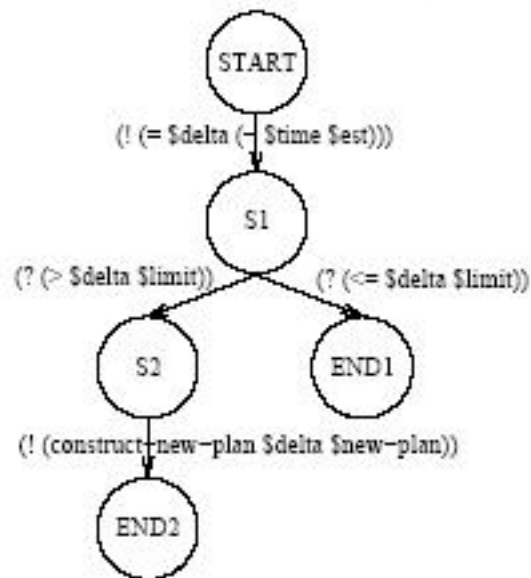
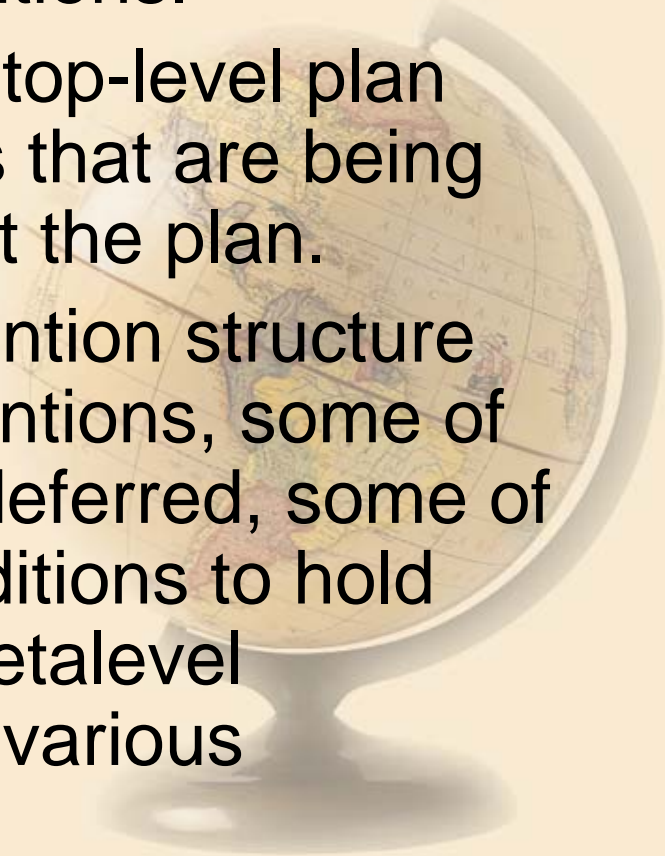


Figure 3: Plan for reacting to an aircraft arriving at a checkpoint

Intention Structure in OASIS

- The intention structure contains all plans the system has chosen for execution, either immediately or at some later time. The adopted plans are called intentions.
- A single intention consist of a top-level plan together with all the sub-plans that are being used in attempting to carry out the plan.
- At any given moment, the intention structure may contain several such intentions, some of which may be suspended or deferred, some of which may be waiting for conditions to hold and some of which may be metalevel intentions to choose between various alternative plans.

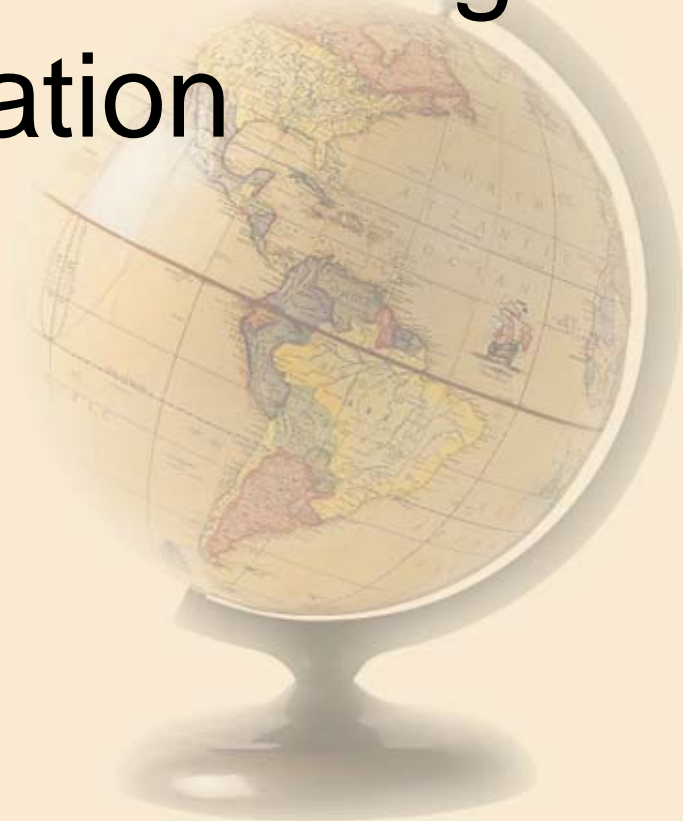


Route-planning in aviation

- Currently (2007) developed software is redrawing routes for airplanes in order to save money (millions of dollars)
- Cost of a route include:
 - overfly fees (varies widely from country to country)
 - fuel bills (depends on changing conditions such as winds, takeoff weight, etc.)
 - navigation charges
 - time



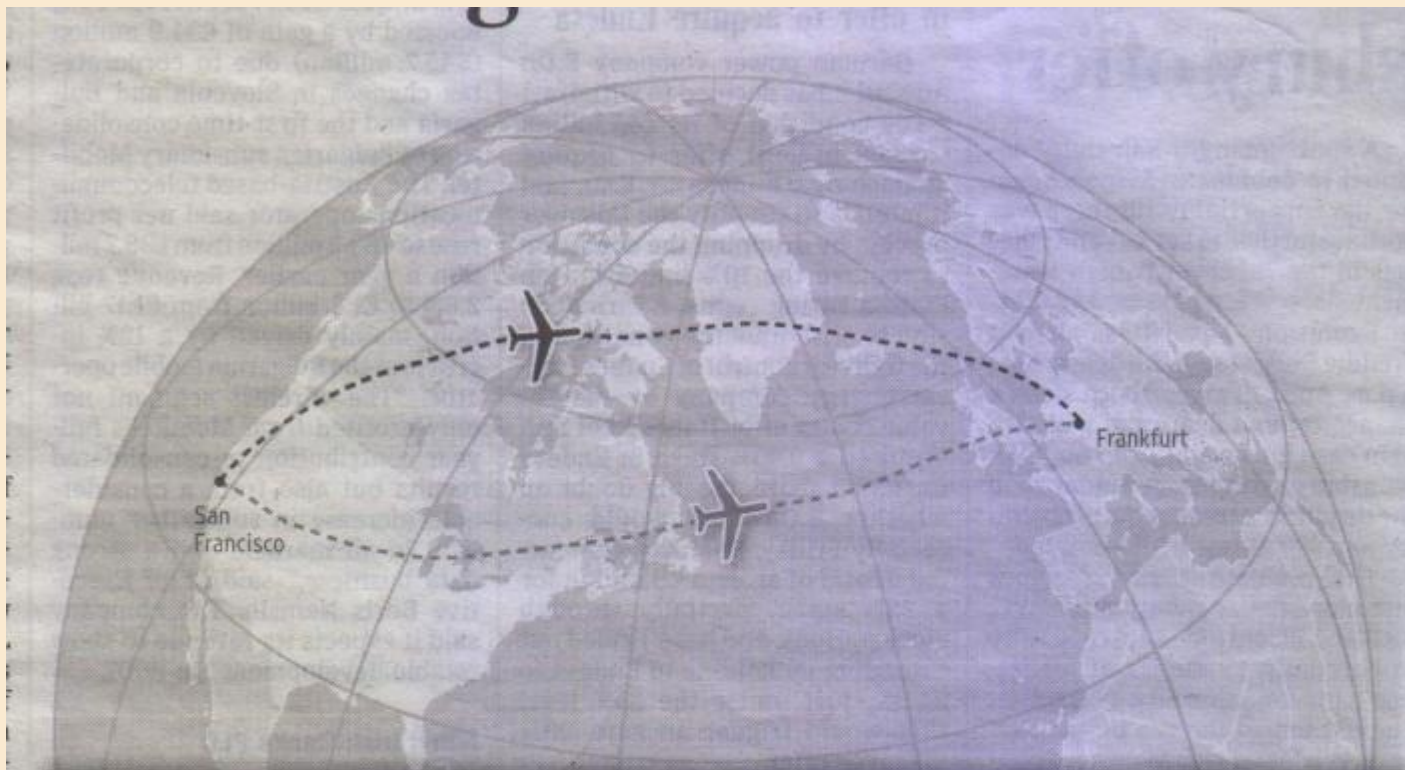
Route-planning and cutting costs in aviation



Economical motivation

- After cutting staff, reducing fleets and taking other steps to cope with a massive industry downturn that begun in 2001, airlines finally are realizing that software can be used to use existing resources more efficiently.
- \$20 billion a year spend by carriers for navigation user fees





Sky less traveled

Route-planning software plots alternative flight paths to save airlines money. How the technology changed United Airlines' San Francisco to Frankfurt route:

Source: United Airlines

SAN FRANCISCO-FRANKFURT

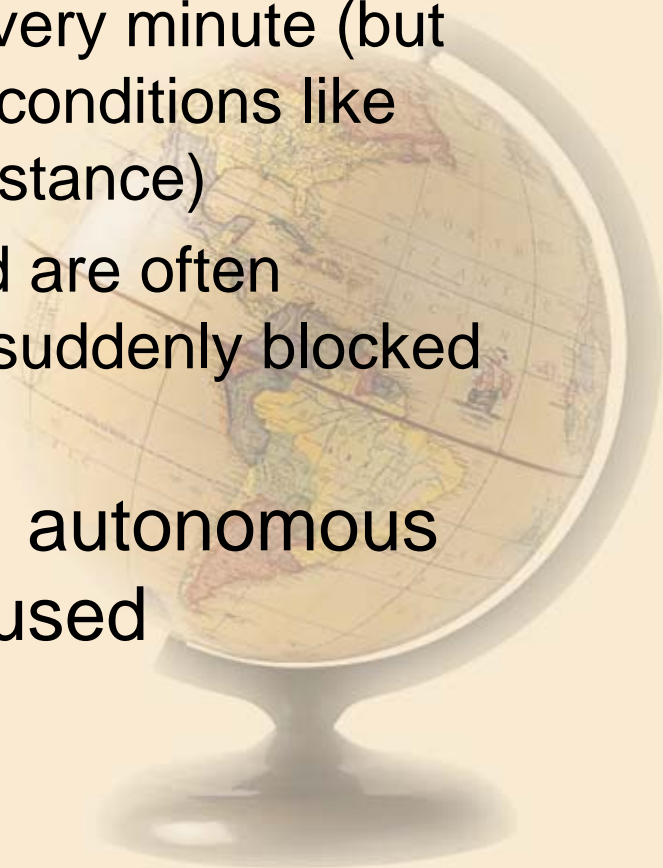
Old Route		New Route	
5,143 nautical miles	Distance	5,222 nautical miles	
10 hours	Approx. flight time	9 hours, 59 minutes	

By flying over the U.S. longer, United pays Canada less in overfly charges and, depending on winds and the aircraft speed, saves time and fuel. Savings on one recent flight: (Average total savings for this rerouting: \$1,433)

Overfly charges	\$884
Fuel	\$790
Total savings	\$1,674

Calculating an efficient route

- Cost depends on both stable and changing conditions:
 - e.g. overfly fees don't change every minute (but they often depend on changing conditions like takeoff weight, overfly time or distance)
 - e.g. wind conditions change and are often unpredictable, airspace maybe suddenly blocked on a planned route
- thus a proactively responding, autonomous system e.g. an bdi should be used



Calculating an efficient route

- Minimizing over flight fees must be balanced against additional fuel costs if an alternative route is less direct
- software systems track: weather, airport locations and runways, weight and performance of each airplane, temporarily blocked airspace and the location of air routes which change daily due to winds



Calculating an efficient route

- Software has to churn through multiple scenarios, including minimum time, minimum fuel and minimum cost
- determine which is the best solution for the maximum payload given up-to-the minute information (e.g. on wind and weather)
- this quite natural to think of bdi here



This works

- British Airways. London, U.K. – Sao Paulo, Brazil. Instead of flying in straight line and over Portugal, Spain and France, the airline takes a northerly track over the Atlantic. Saves \$5,767 per one-way flight in European flight fees, uses one ton less of fuel and is 18 minutes shorter in duration



Thank, You.

