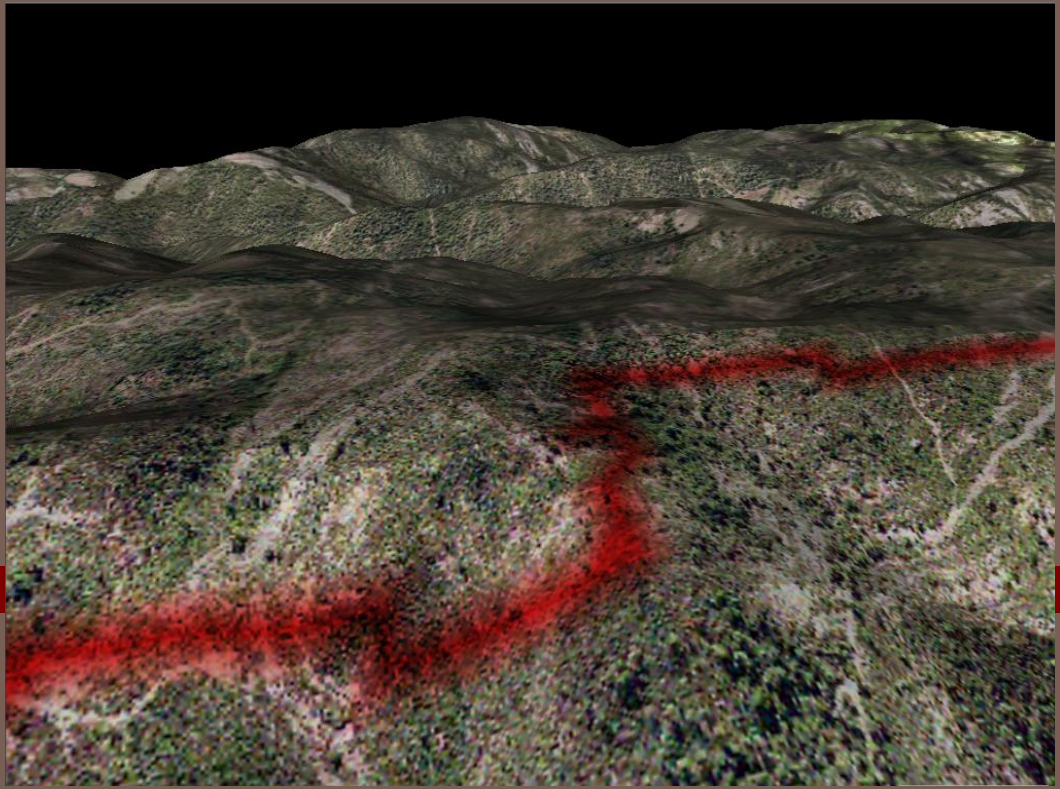


GIS AND SIMULATION, ENVIRONMENTAL SIMULATION

Pau Fonseca i Casas, pau@fib.upc.edu

Outline

- Introduction
- Cellular automaton
- Intelligent Agents
- Formalization
- Examples

Discrete simulation

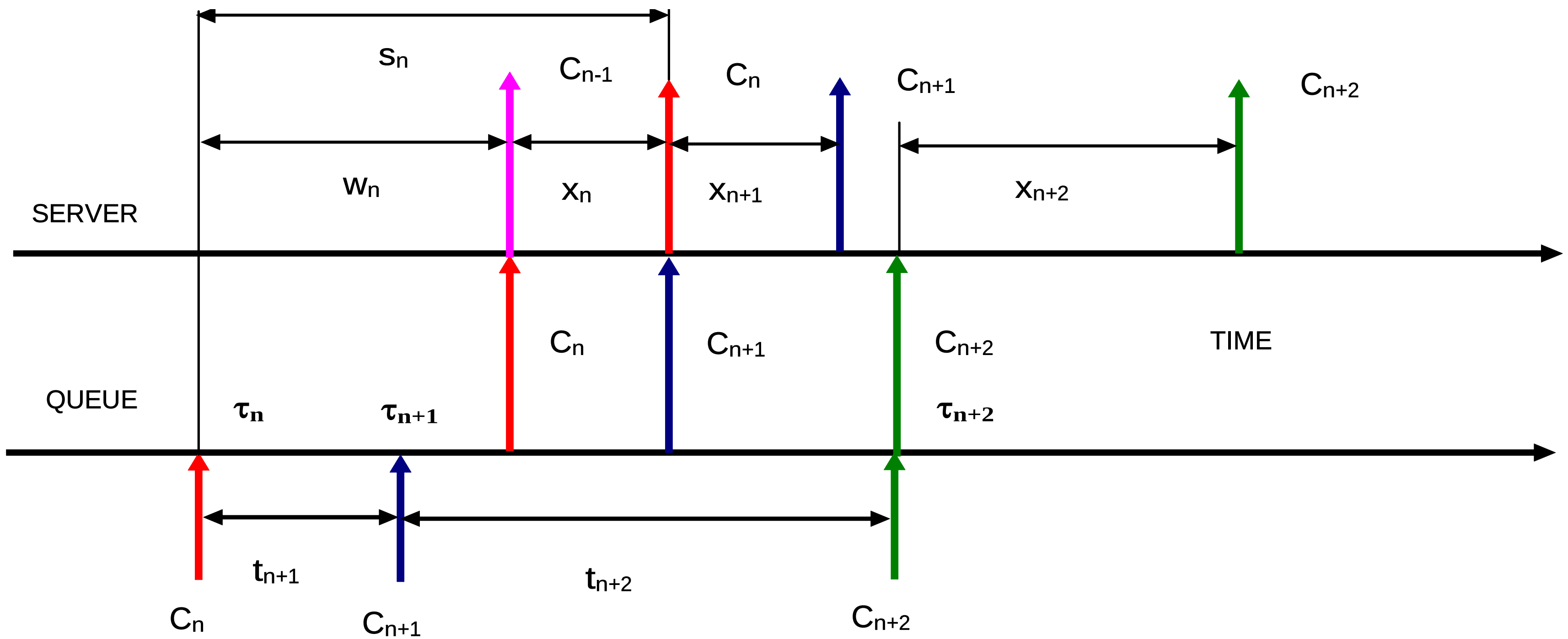
A brief description of the simulation engine

Event Scheduling 1/2

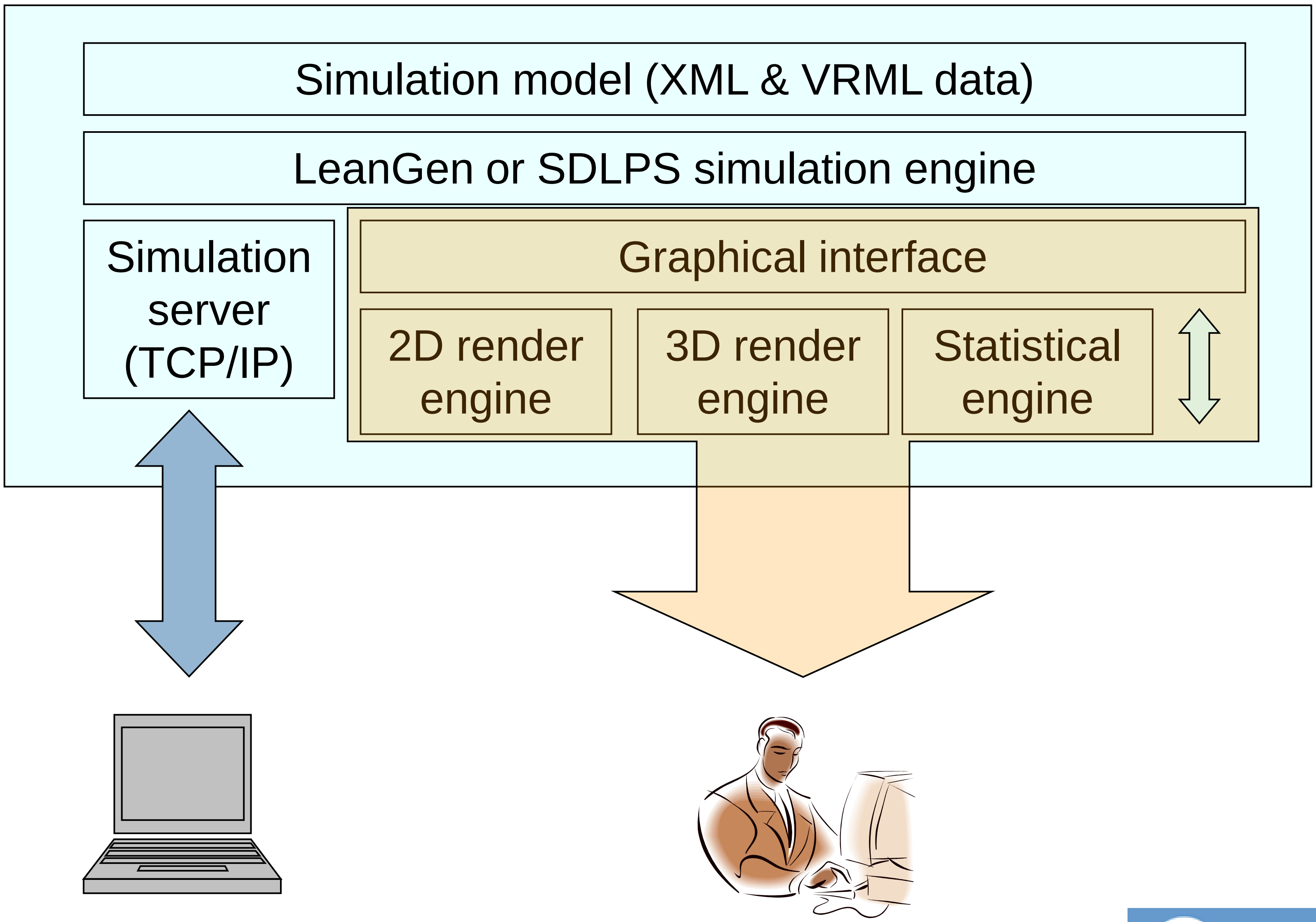
- System modifications only occur in certain time instants.
  - Changes determined by the incoming events.
- Allow detailed model construction.
- Allow different paradigm combination.

Event Scheduling 2/2

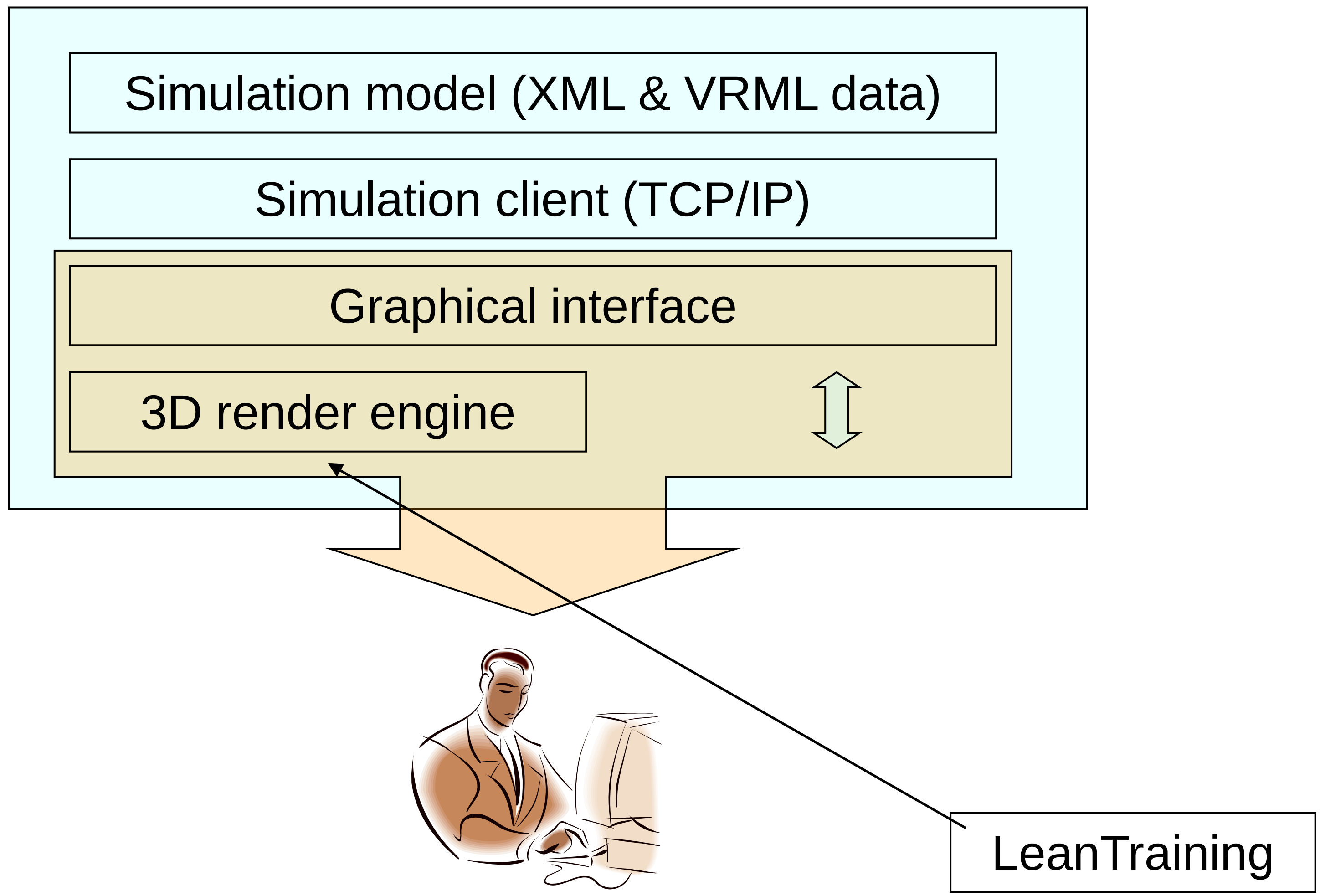
- Event Scheduling: Sample M|M|S



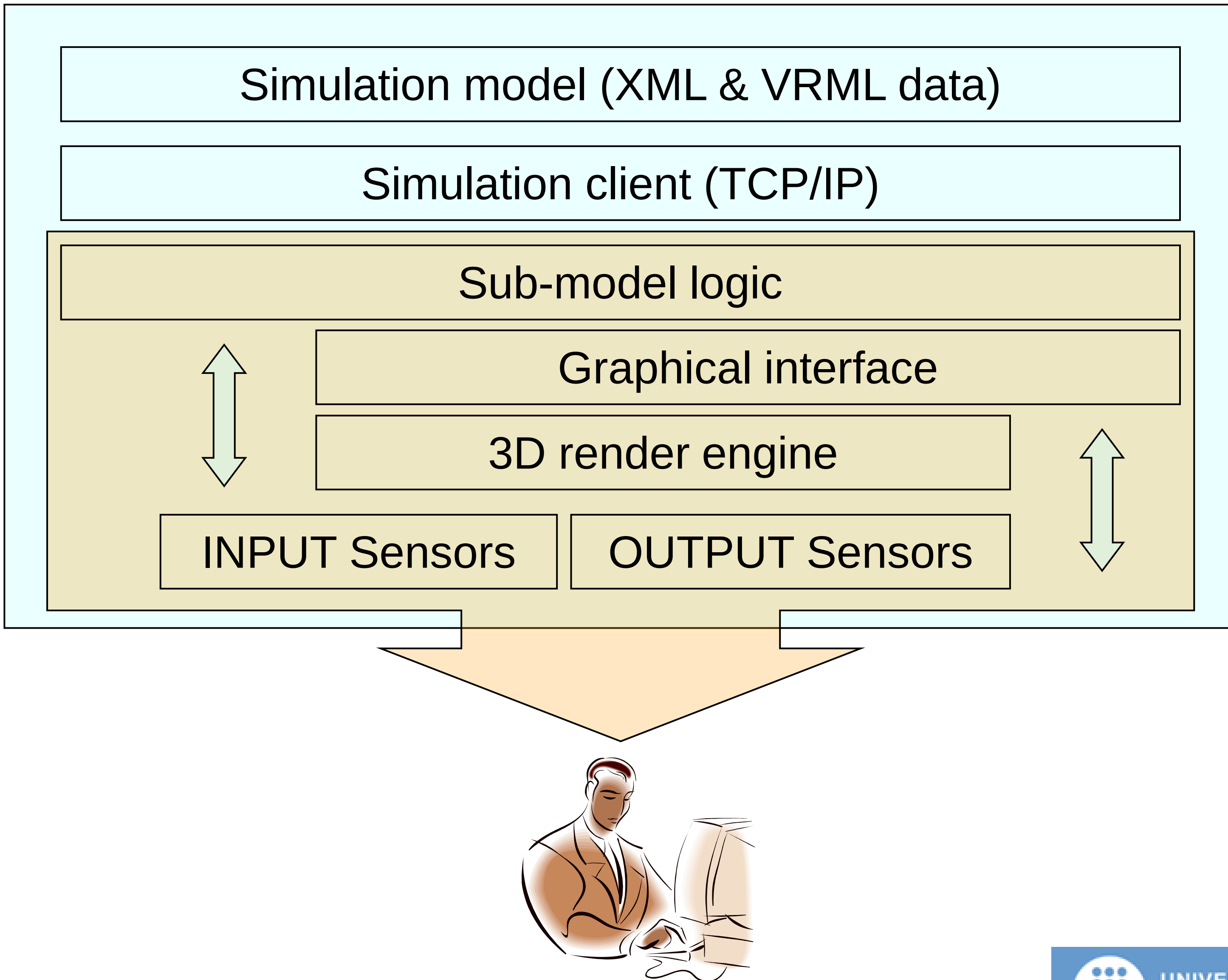
Simulation engine



Simulation client

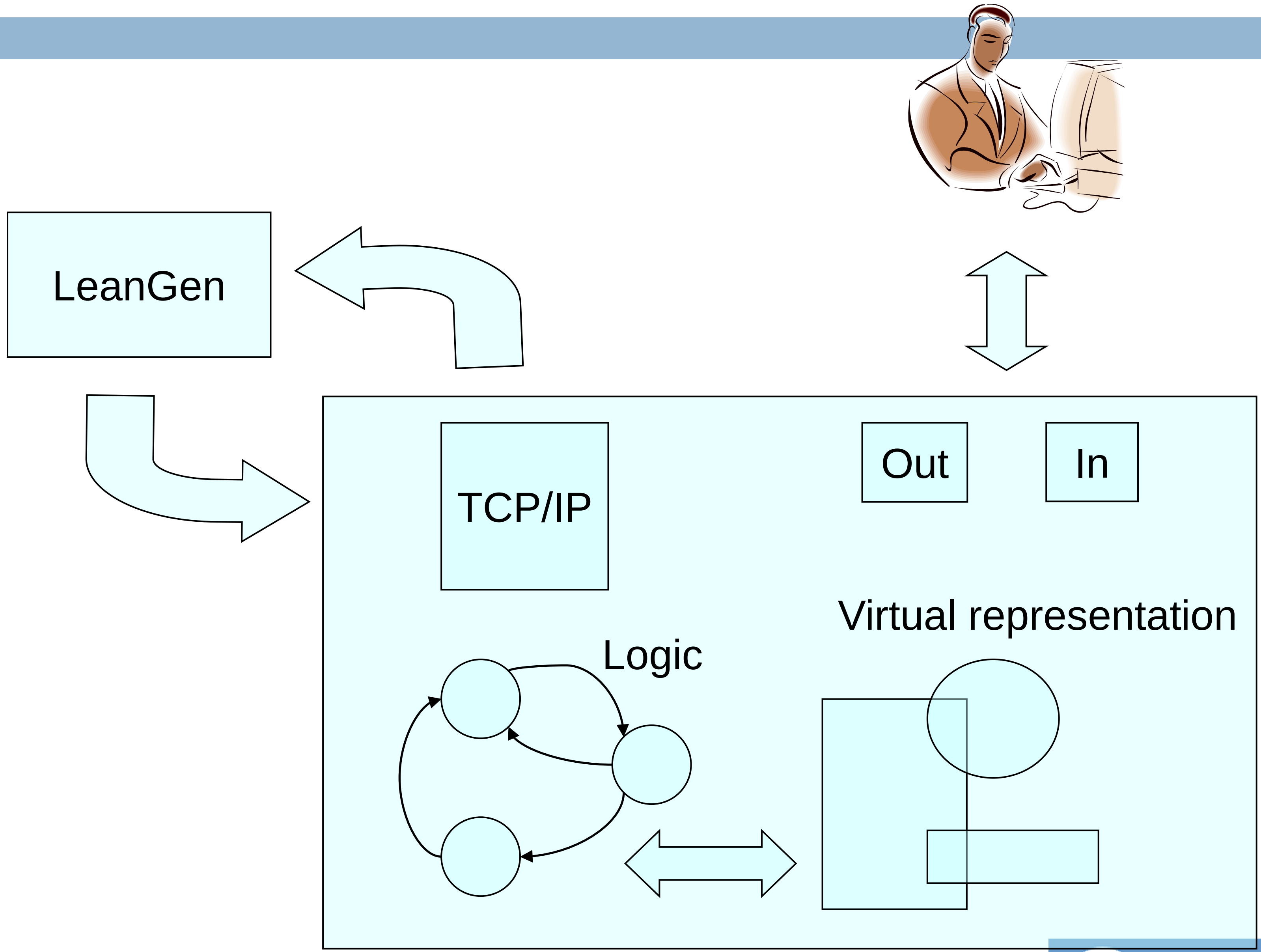


Training client



UNIVERSITAT POLITÈCNICA DE CATALUNYA

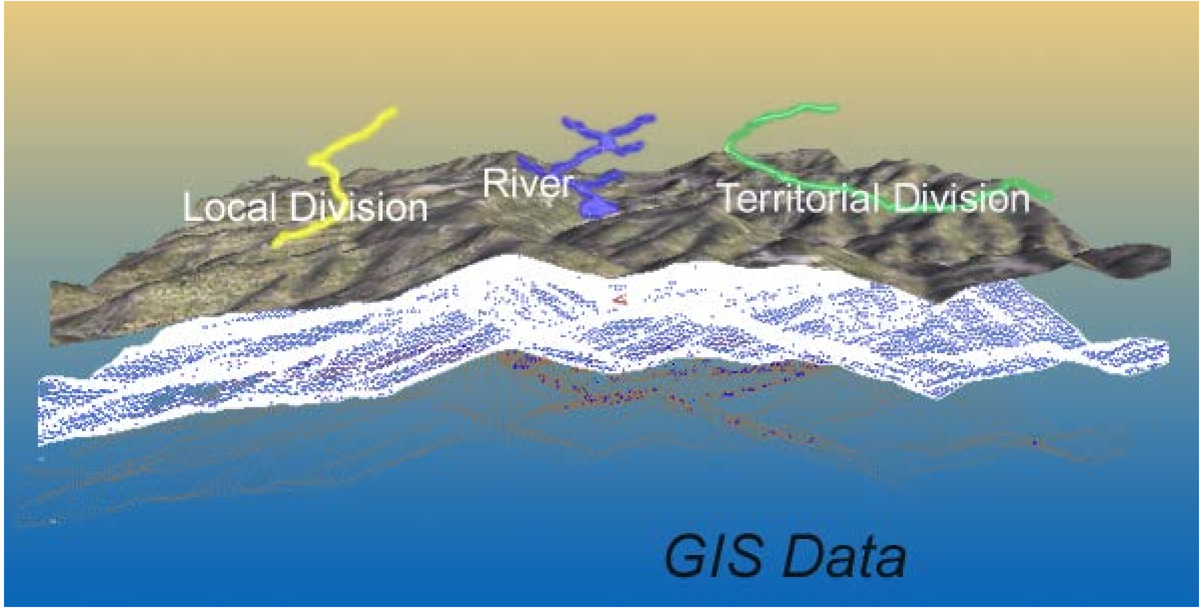
Agent client



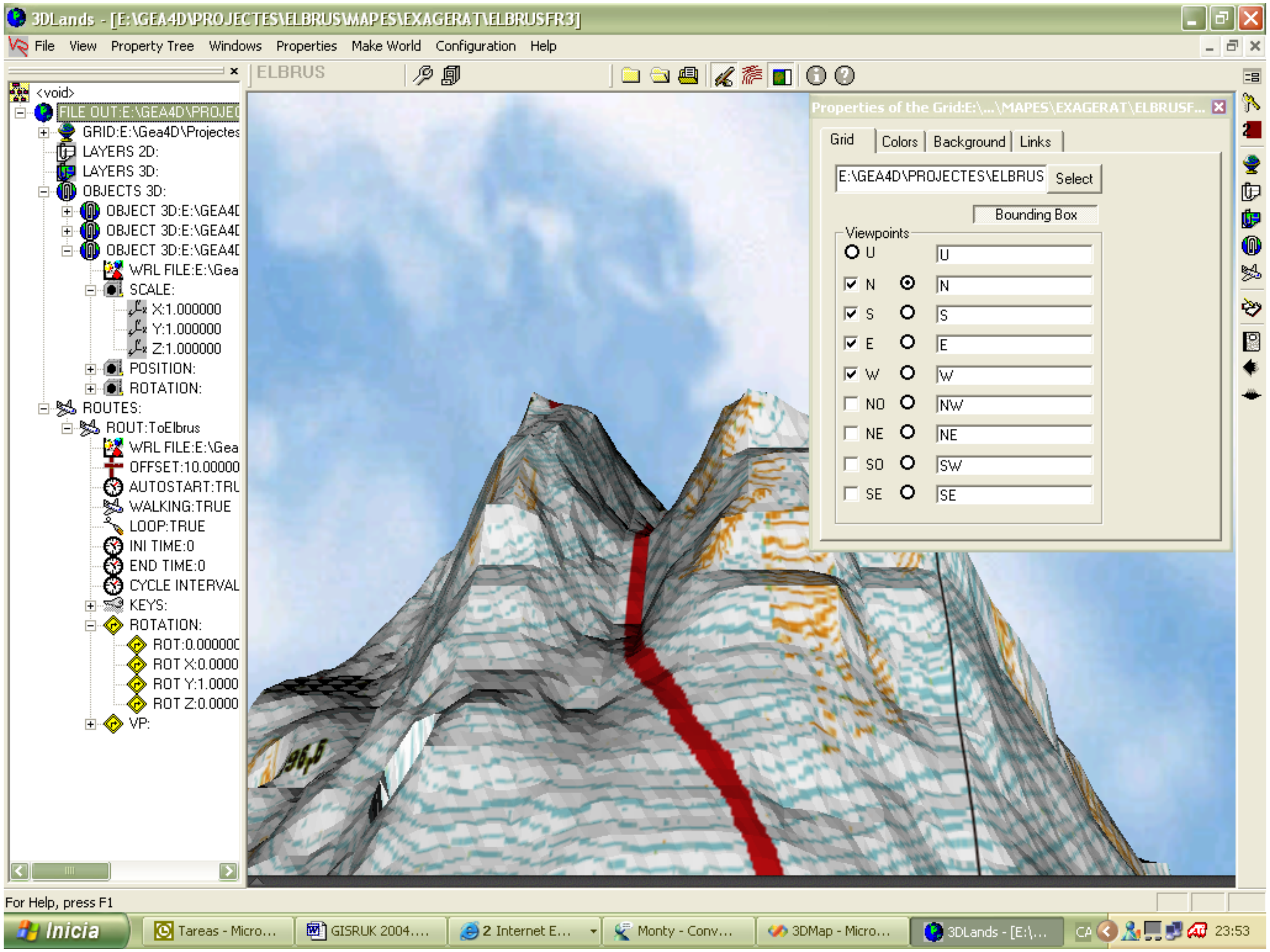
GIS data

GIS data structure and classification.

GIS data



VirtualLands©



UNIVERSITAT POLITÈCNICA DE CATALUNYA

GPS layers integration

Layer	***GPS integration***
DEM	Geo referenced
2DLayers	Geo referenced
3DLayers	Geo referenced
Routes	Track points
2DObjects	Waypoints
3DObjects	Waypoints

- Integration of some Objects or layers with a GPS.
- Enables the training systems

2DLayers (Vectorial layers)

- Point, polylines, texts or lines.
- Usually represent information for the observer, but not represent information for the simulation model (don't have any specific behaviour).

3DLayers (Rasters Layers)

- To represent a fixed population of elements
  - Forest or a city (each tree have his own position over the DEM).
- All the elements have the same virtual representation and the same behaviour.

Objects

- To represent element with an individual or concrete behaviour (2DObjects or 3DObjects).
  - **2DObjects:** lines polylines, texts and points with an individual behaviour, (a point with a touch sensor that shows some information).
  - **3DObjects:** virtual objects that are not associated with any layer. These objects can have own behaviour represented by a script or program connected with the world through EAI.

Routes

- The routes define 3DObjects movements through the virtual Landscape.
- These routes can be defined by the simulator.
- Enables the possibility of move different elements through to Landscape following the simulator logic.

Why use GIS data?

A classification of the models

Interactive/evolutionary models

	Static models	Dynamic models
Interactive	Static interactions. The only changes are in the composition. For instance, systems that are not modified over time. Through simulations, an approximate value can be obtained.	System interaction. Changes in the interactions between the different model components. For instance, an industrial plant.
Evolutionary	Evolutionary selection. Random acquisition of variations that change the composition of types.	Evolutionary system feedback that influences the supply of variation and the speed of evolution. Changes in type depend on the history of the system. For instance, the evolution of a society or wildfire with the interaction of an extinction model.

(Henning 2001)

MAS

		Enviroment	
	Relation with the environment	Discrete	Continuous
Static	Strong	Automaton	Control system
	Weak	Semi-Isolated Evolution	
Dynamic	Weak	Complex Interactions	
	Strong	Ecosystem	

(Marín and Mehandjiev 2006)

Using GIS in Simulation models

- Allows environment modeling.
- Dynamical use of the GIS data.
  - Dynamical modification of the GIS data.
  - Dynamical acquisition of the GIS data.

Cellular automata

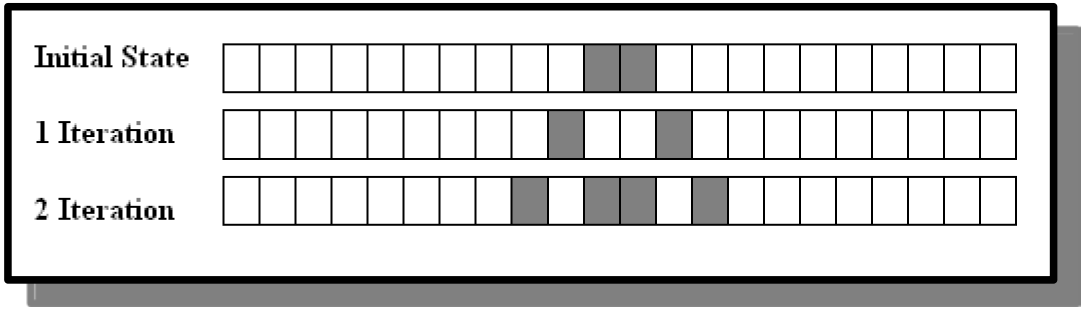
Modeling the environment

Cellular automaton

- Simplifies the use of environment in a simulation model.

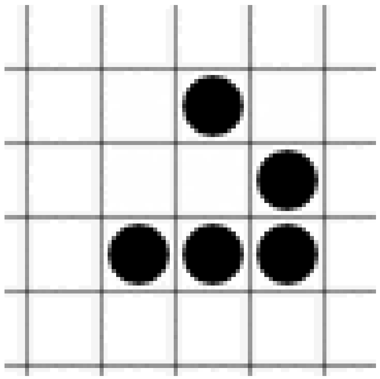
Cellular automaton

- Structure based in:
  - Set of rules.
  - Matrix of data.
- Modify the matrix following the set of rules.



Game of life

- The **Game of Life** is a cellular automaton devised by the British mathematician John Horton Conway in 1970. It is the best-known example of a cellular automaton.
- Glider gun and glider



Cellular automaton

- Only one matrix of data.
- Only one set of rules.
- The space is discrete.
- The space of states can be huge.
- Not all the states can be reachable → maps of states space.

m:n-CA^k

Extending the capabilities of the cellular automata

m:n-CA^k

- A multi n dimensional cellular automaton is a cellular automaton generalization composed by m layers with n dimensions each one.
- Aim:
  - Allows multiple layers.
  - Allows vectorial layers (continuous space).
  - Allows multiple set of rules (evolution functions).

m:n-CA^k

- The representation is:

$$m:n-AC^{k}$$

- Where
  - m: is the automaton number of layers.
  - n: is the different layers dimension.
  - k: is the number of main layers (1 by default).

m:n-CA^k

- Defined over the mathematical topology concept.
- 1:n-AC ^{1} is the common cellular automata if the topology used is the discrete topology defined over N or Z.
- The implementation, as is usual, can be a matrix.

State of the automata

- $E_m[x_1,..,x_n]$, layer m state in $x_1,..,x_n$ position
  - E_m is a function describing cell state in position $x_1,..,x_n$ of layer m.
- $EG[x_1,..,x_n]$, automata status in $x_1,..,x_n$ position.
  - EG returns automata global state in position georeferenced by coordinates $x_1,..,x_n$.

$$\Psi(E_1[x_1..x_n],\cdot^{m-2)},E_m[x_1..x_n]) = EG[x_1..x_n]$$

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Evolution function Λ_m

- Evolution function allows global automaton state change through cells value modification.
- Defined for the layer m to modify its state through the state of others layers using combination function Ψ, and vicinity and nucleus functions.
- Is only defined in main layers.

Evolution function Λ_m

- Function defined for the layer m to modify its state through the state of others layers using combination function Ψ, and vicinity and nucleus functions.
- A m:n-AC automaton only presents one main layer, an m:n-AC^k automaton presents k main layers.

Vicinity topology

- Topology defining the set of points (neighbourhood) for layer m, to be considered for Λ_m calculus.
- Vicinity function $vn(x_1,..,x_n)$:
  - Function returning minimum open set of vicinity topology containing point $x_1,..,x_n$, and including maximum points that accomplishes the restriction and minimum points not accomplishing the restriction.

Nucleous topology

- Topology defining the set of points (neighborhoods) for layer m, to be modified by Λ_m calculus
- Nucleus function $nc(x_1,..,x_n)$
  - Function returning minimum open set of nucleus topology containing point $x_1,..,x_n$, and including maximum points that accomplishes the restriction and minimum points not accomplishing the restriction.

Vicinity function example

- <u>Over Z</u>
- vn(x_1,..,x_n) = {(x_{1-1},x_{2-1},x_{n-1}), (x1-1,x2-1,Xn), (x1-1,x2-1,xn+1), (x1-1,x2,xn-1), (x1-1,x2,Xn), (x1-1,x2,xn+1), (x1-1,x2+1,xn-1), (x1-1,x2+1,Xn), (x1-1,x2+1,xn+1), (x1,x2-1,xn-1), (x1,x2-1,Xn), (x1,x2-1,xn+1), (x1,x2,xn-1), (x1,x2,Xn), (x1,x2,xn+1), (x1,x2+1,xn-1), (x1,x2+1,Xn), (x1,x2+1,xn+1), (x1+1,x2-1,xn-1), (x1+1,x2-1,Xn), (x1+1,x2-1,xn+1), (x1+1,x2,xn-1), (x1+1,x2,Xn), (x1+1,x2,xn+1), (x1+1,x2+1,xn-1), (x1+1,x2+1,xn), (x1+1,x2+1,xn+1)}
- <u>Over R</u>
- vn(x_1,..,x_n)=returns the open set centered in the point x_1,x_2,x_3 for the topology that defines the vicinity. $B(x,r) = \{y \in \Re^m / d(x,y) < r\}$

UNIVERSITAT POLITÈCNICA DE CATALUNYA

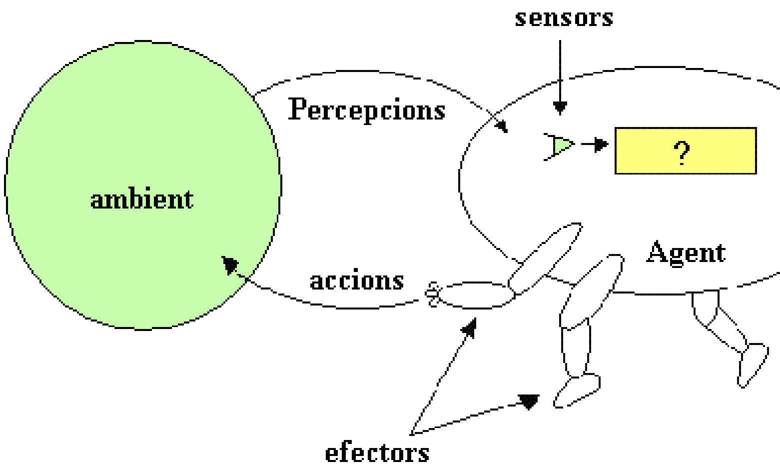
Nucleus function example

- <u>Over Z</u>
- $nc(x_1,..,x_n) = \{(x_1,..,x_n)\}$
- <u>Over R</u>
- $nc(x_1,..,x_n)$=returns the open set centered in the point x_1,x_2,x_3 for the topology that defines the nucleus.

Intelligent agents

Reacting to the environment

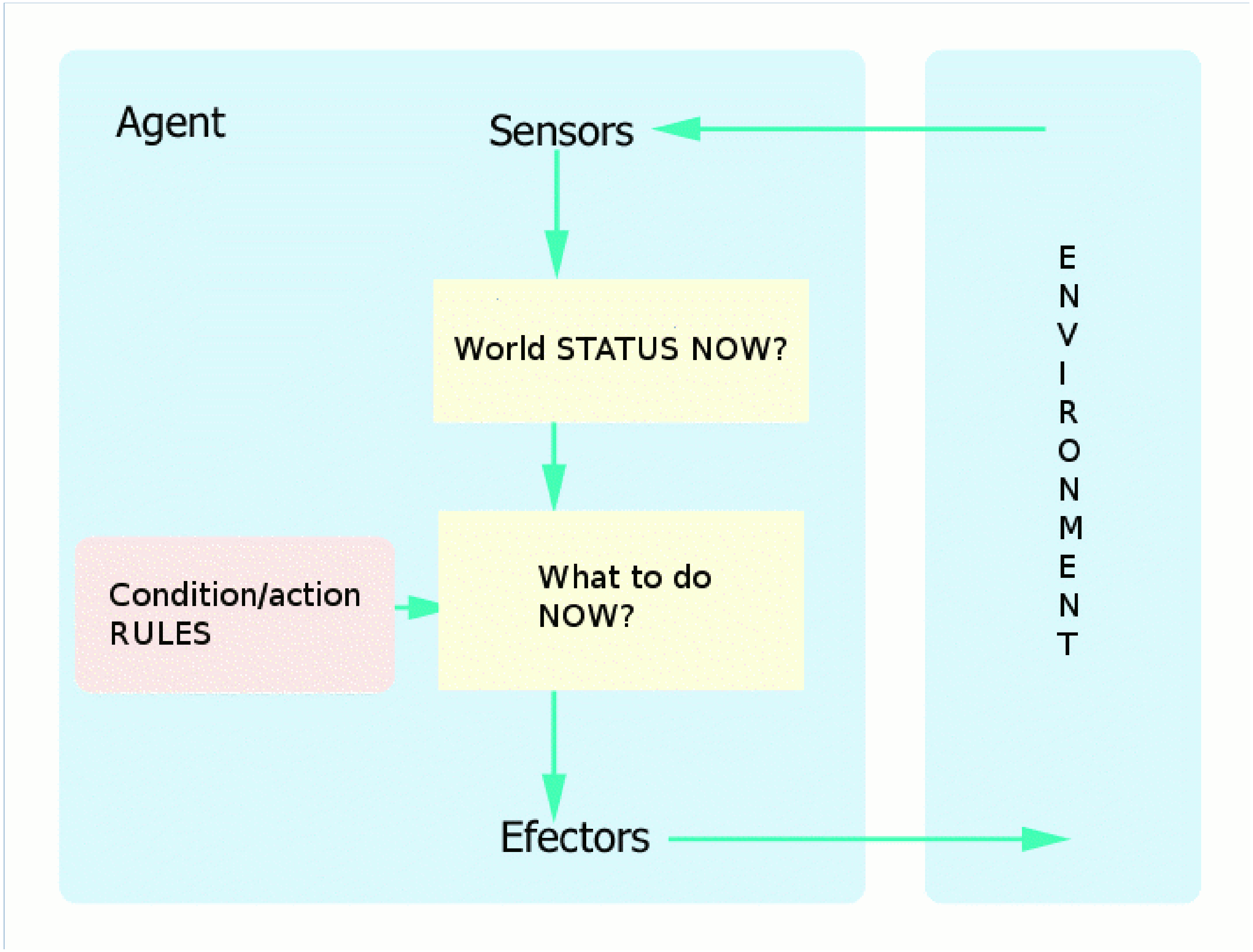
Intelligent agent



- An structure reacting to the environment through his actions, and perceiving the environment through the sensors.

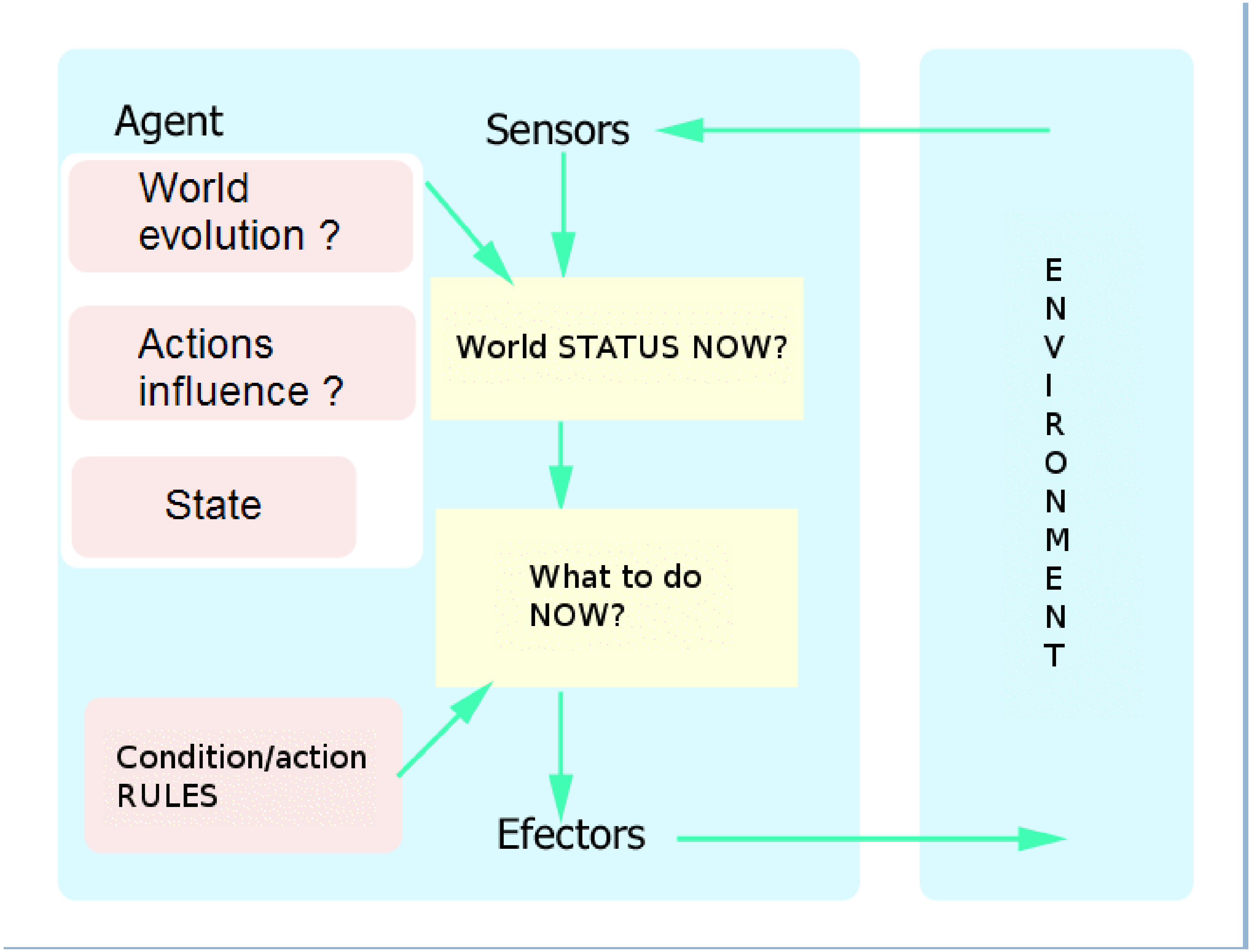
UNIVERSITAT POLITÈCNICA DE CATALUNYA

Reflexive intelligent agent

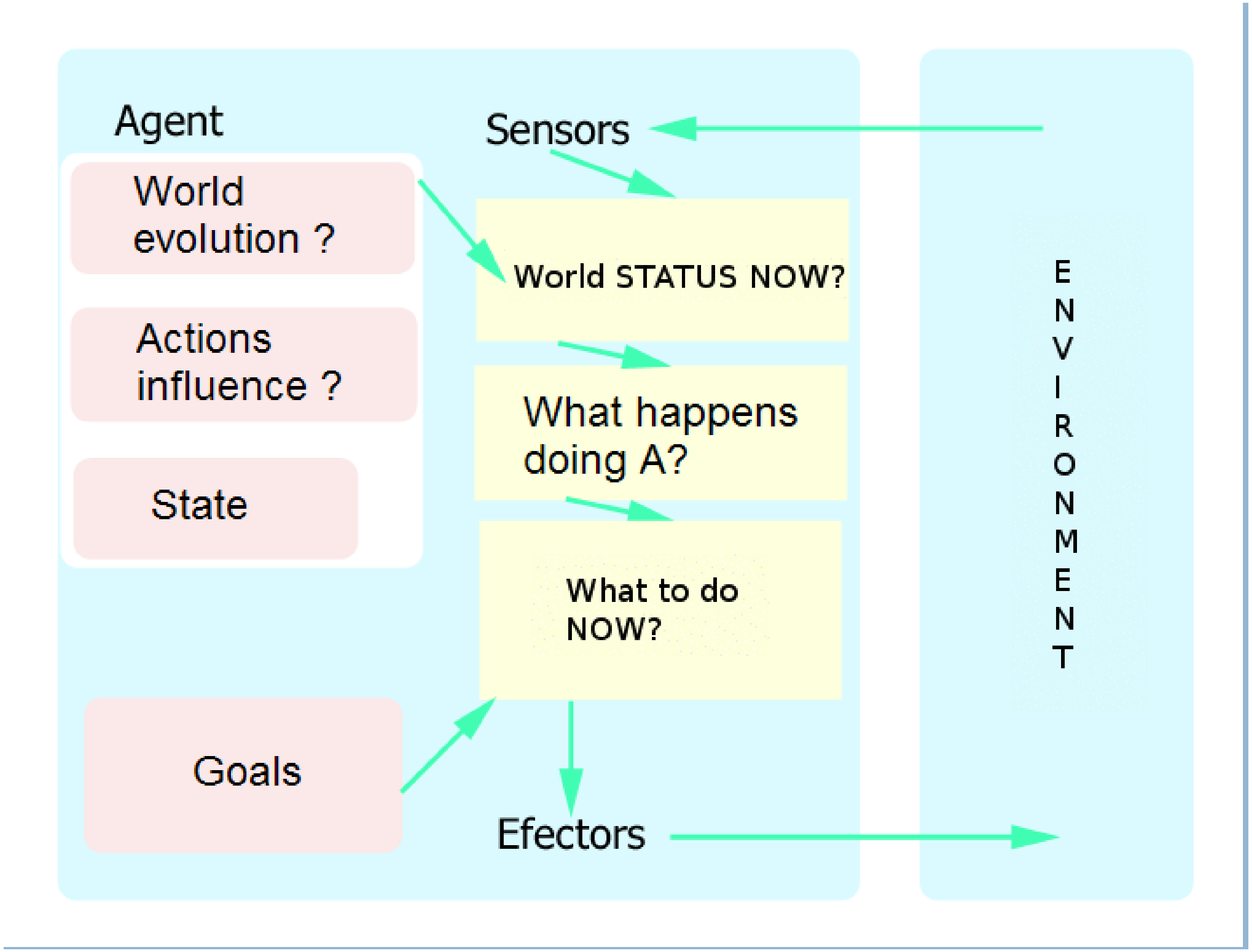


UNIVERSITAT POLITÈCNICA DE CATALUNYA

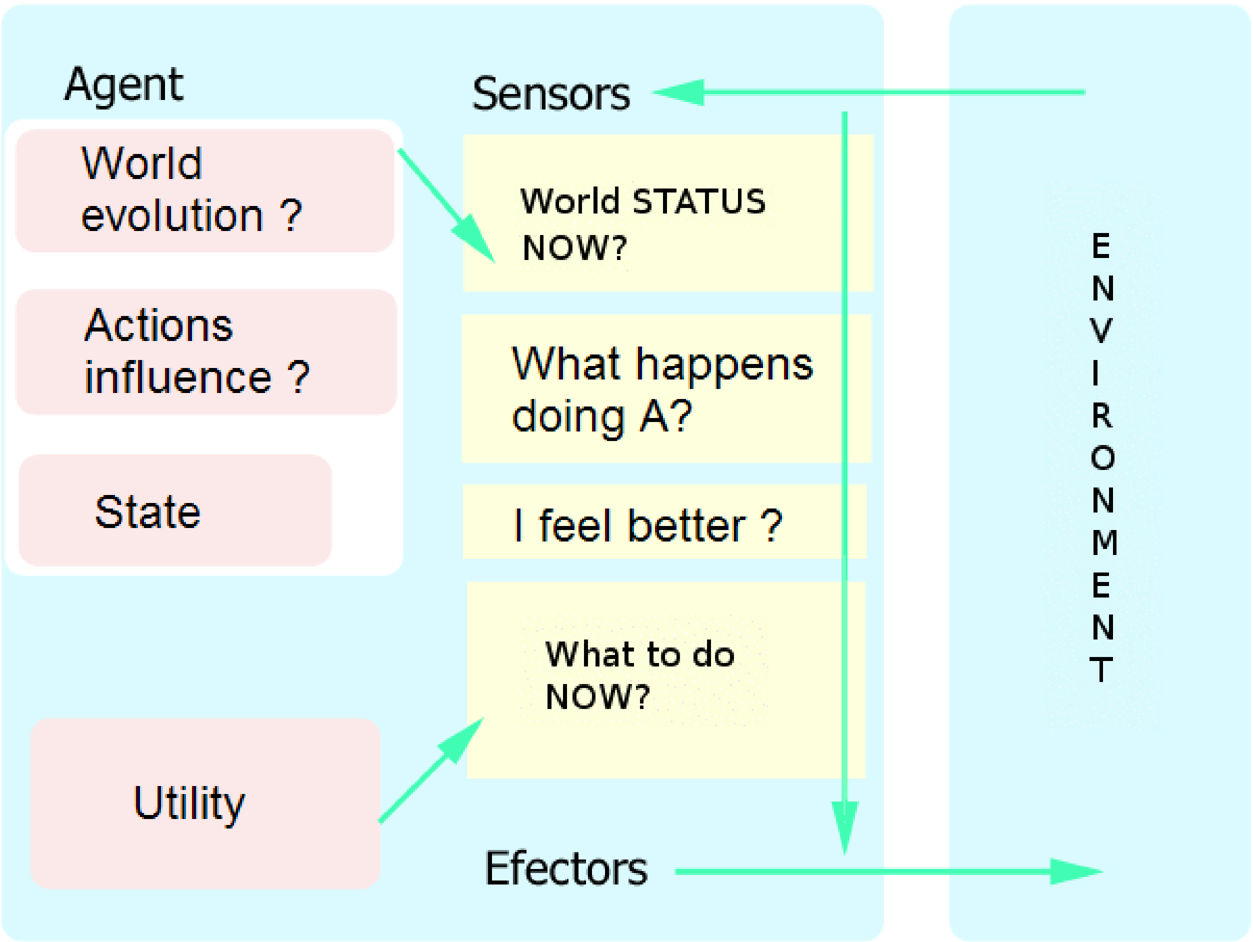
Model based reflexive agents



Goal based agents



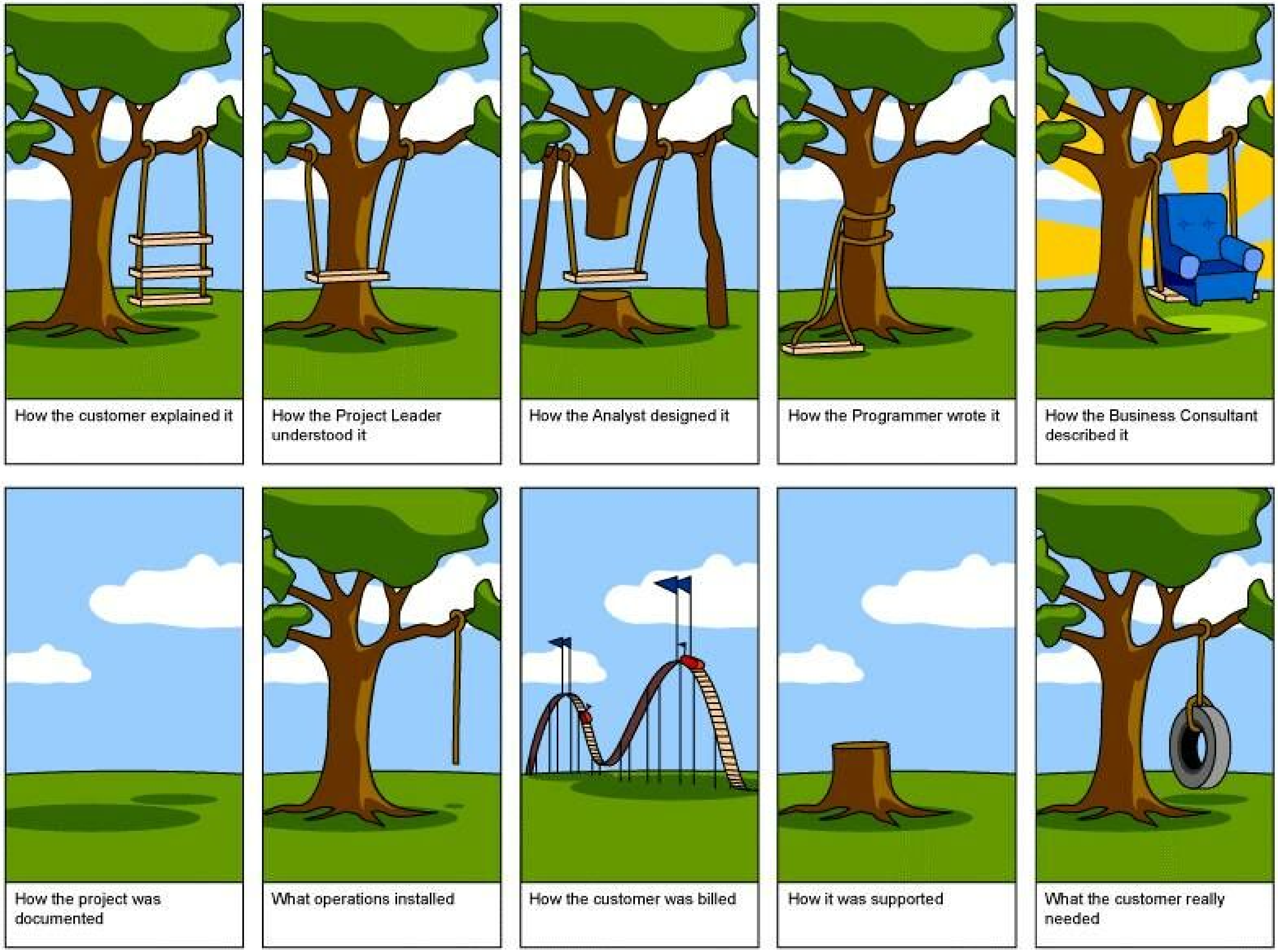
Utility based agents



Specification

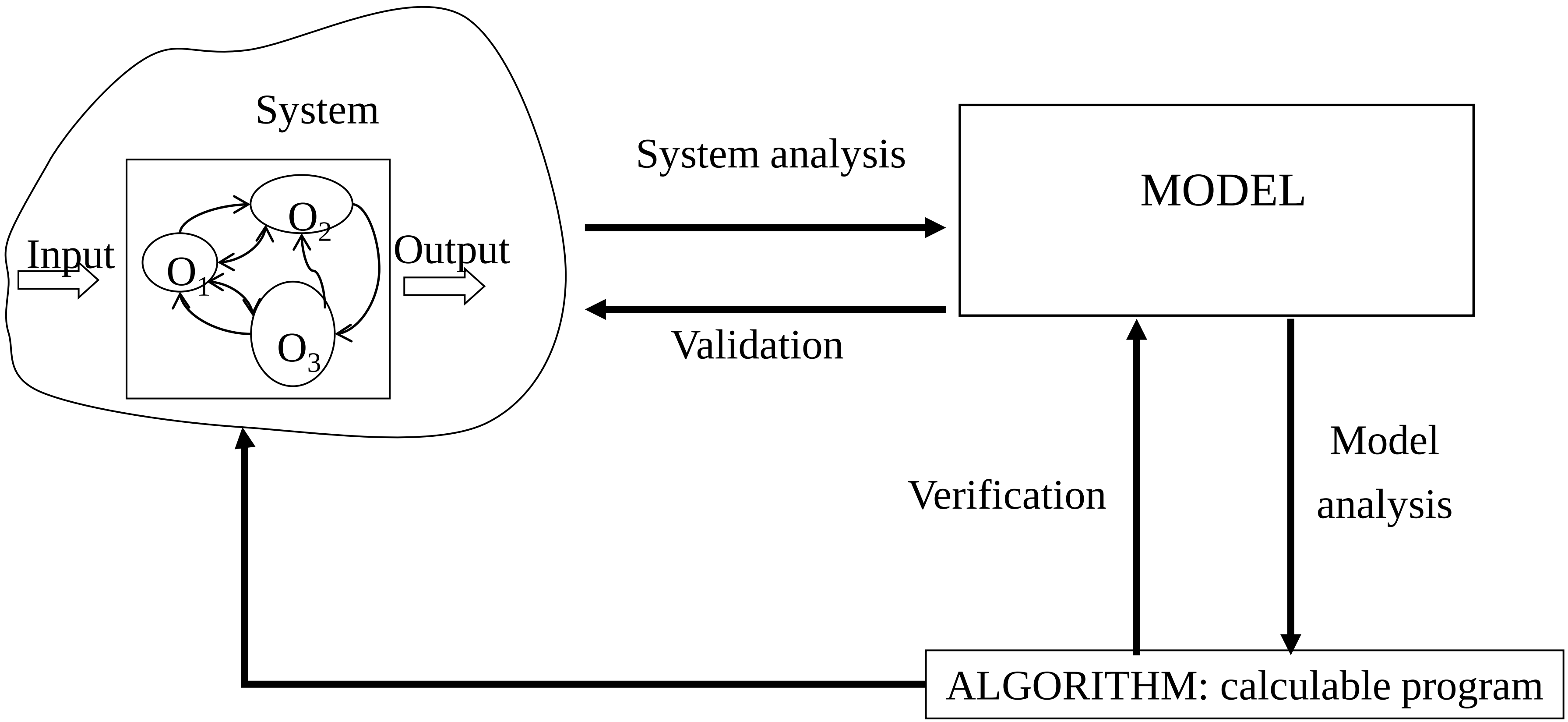
A brief note of how to formalize simulation models

Why?



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Why?



Obtaining answers, data from the executions of the model.

Implementing the results?

Alternatives

- DEVS, Petri Nets, SDL, SysMPL, UML,...
- What is the tool we have?
- What is the personnel involved in the project?
- We have components to reuse?
- Are this formalisms and languages ready to represent the needed structures for our models?

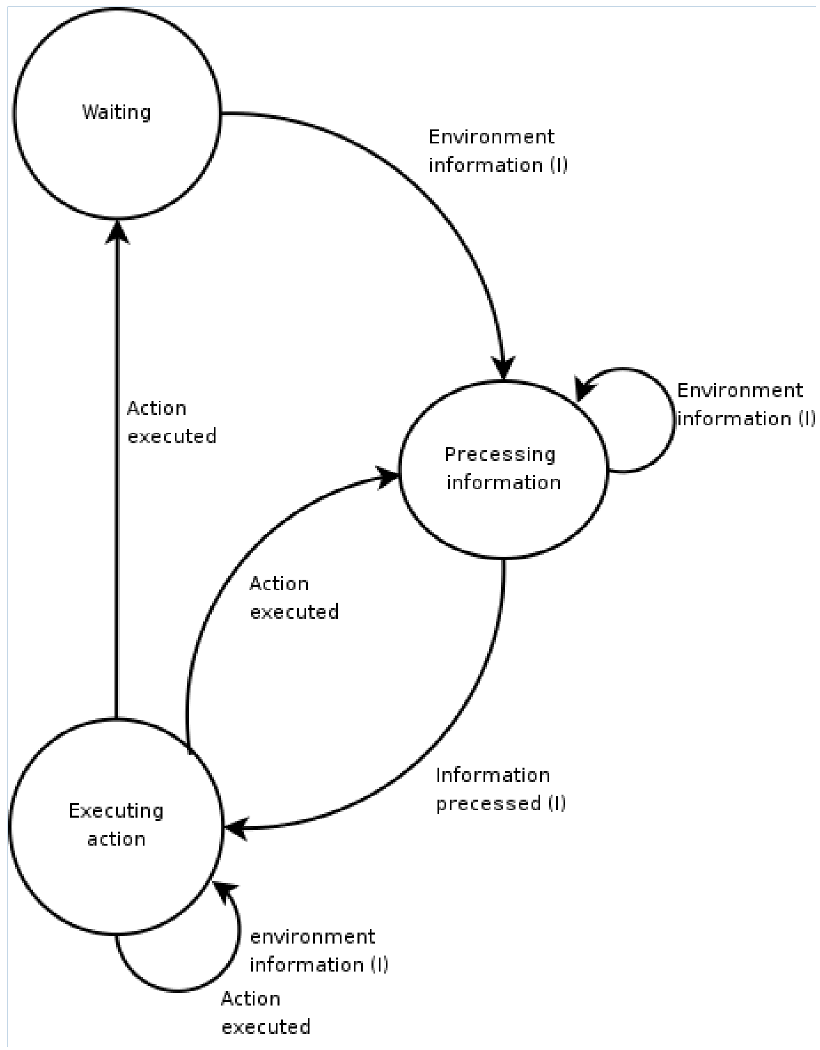
Tendencies

- More and more the languages and formalisms are increasing the interest in this area
  - Implies the support to represent cellular automata or intelligent agents
  - CELL-DEVS

SDL language

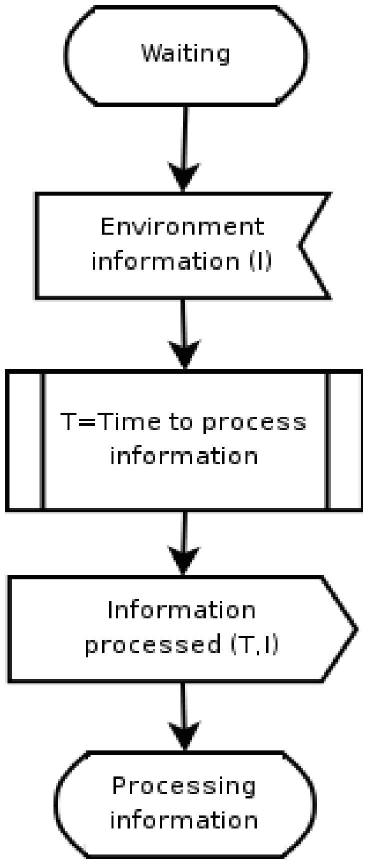
- Object-oriented, formal language defined by The International Telecommunications Union as recommendation Z.100.
- Intended for the specification of complex, event-driven, real-time, and interactive applications involving many concurrent activities that communicate using discrete signals.

Reflexive agent specification



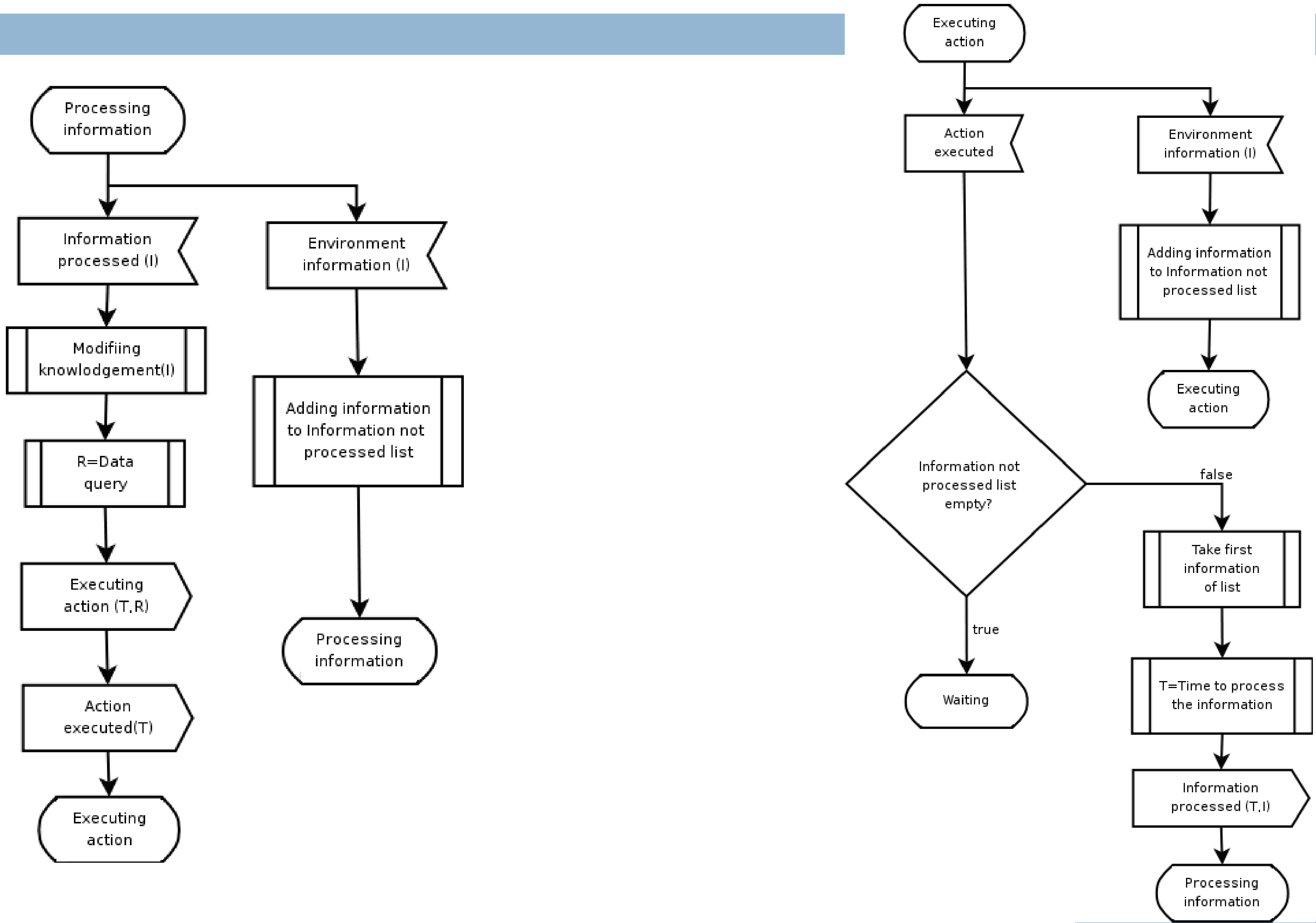
Reflexive agent specification

- Time to process information represent the delay due to the understanding of what happens in the world



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Reflexive agent specification



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Examples

The wildfire model and the slap avalanche model

Wildfire model.

- Motivations:
  - Dangerous environment.
  - Difficult to experiment.
  - Simulations involves naturals resources and personel.

- To develop an experimental framework to simulate a wildfire
  - Propagation.
  - Extintion.

- Working with:
  - CREAF data.
  - *Bombers de la Generalitat* (fireman).

Wildfire propagation (over R)

- Implemented using SDLPS.
- BEHAVE model.
- Raster data describing the landscape.

GIS Data

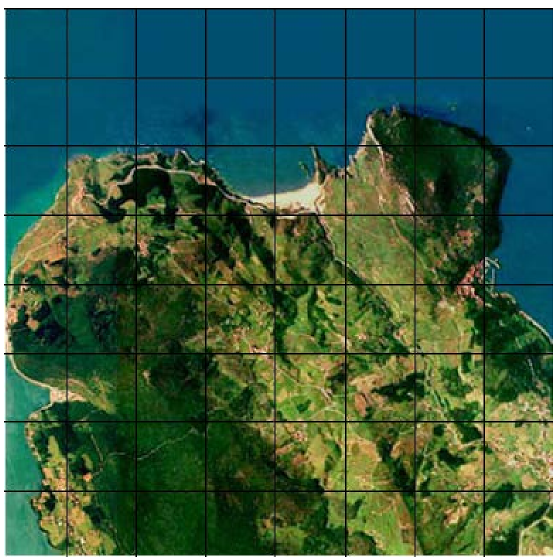
- Input data files:
  - **Mapa**: file containing the DEM (Digital Elevation Model).
  - **Model**: file that represents the propagation model implemented for each cell.
  - **Slope, Aspect**: files that stores the slope and his direction. These files are calculated using the DEM. (Mapa files)
  - **M1, M10, M100, Mherb, Mwood**: files that contains the combustible description.
- The results files are two files:
  - **ignMap.dtm**: Stores ignition time.
  - **flMap.dtm**: Stores flames elevation.

UNIVERSITAT POLITÈCNICA DE CATALUNYA

GIS data: IDRISI32.

- 1987, Research program of Clark's University.

- We use the IDRISI32 file format.
  - One file for the data.
  - Other for the information related to the data.



20	30	10	90	178	12	89	12
34	23	56	12	342	56	34	38
12	12	34	15	76	87	12	32
...							

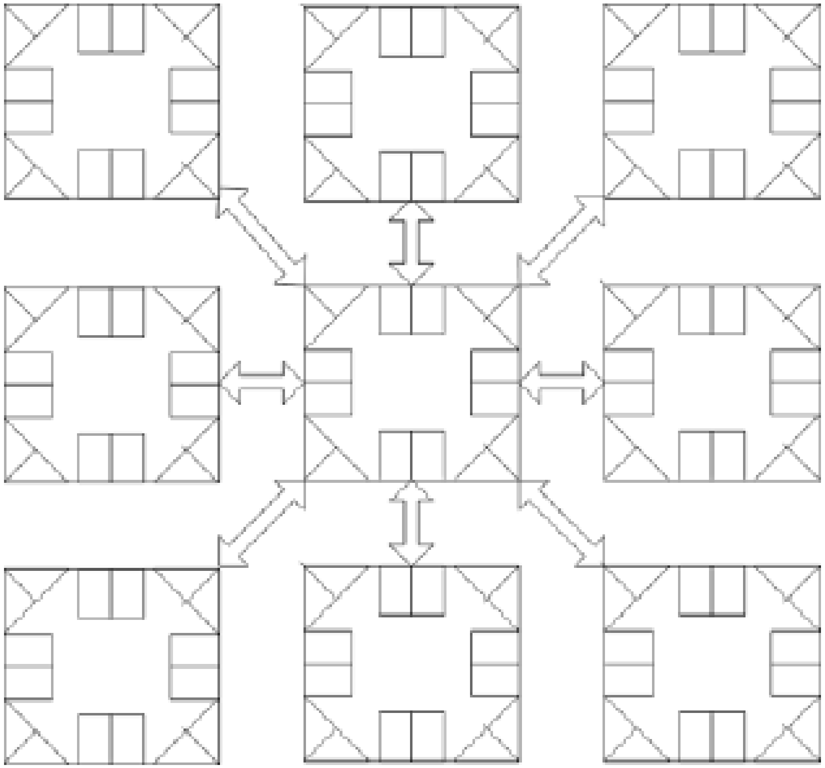
Example, definition

- The Λ_1 function works with Moore neighborhood therefore vicinity function and nucleus function are:
- **vn(x_i,x_j)**= $\{p_{i-1,j-1},p_{i,j-1},p_{i+1,j-1},p_{i-1,j},p_{i+1,j},p_{i-1,j+1},p_{i,j+1},p_{i+1,j+1}\}$
- **nc(x_i,x_j)**= x_i,x_j

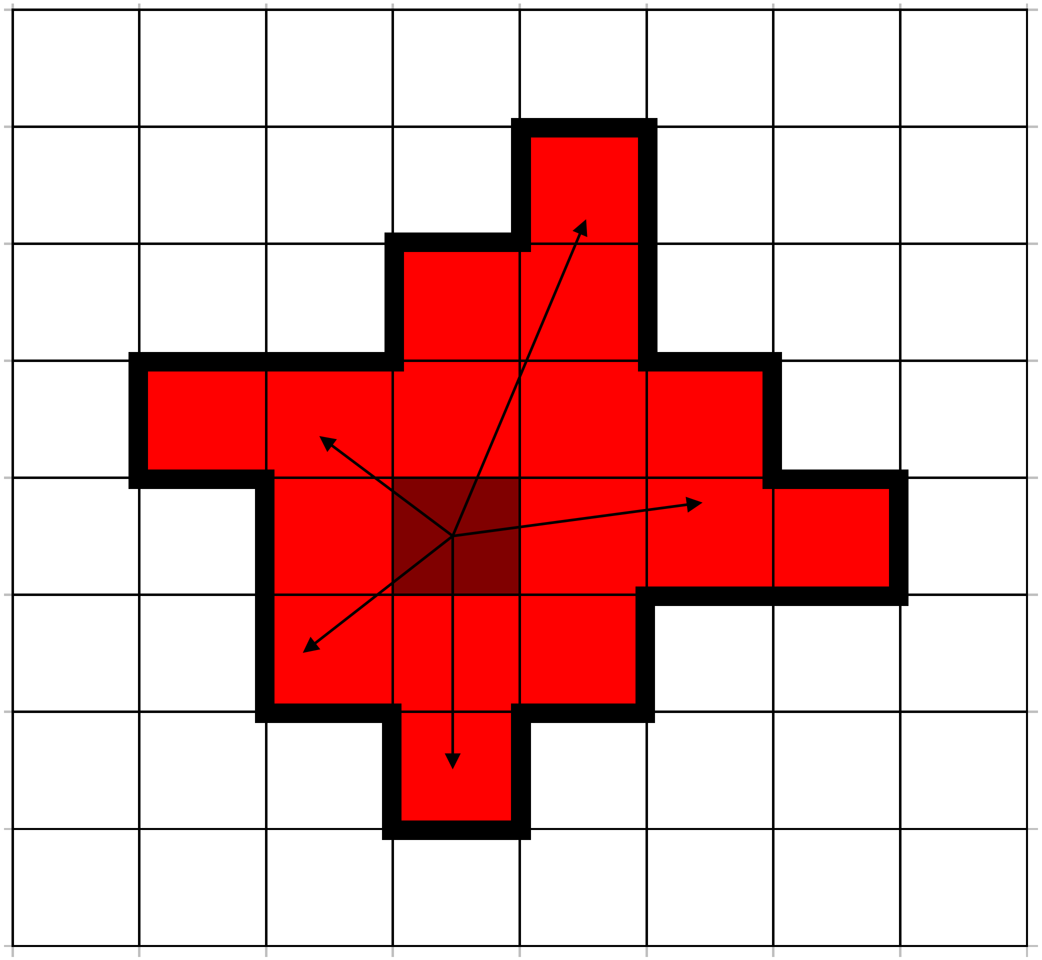
Propagation model events

- The events that lead propagation model are:
- **EBurn**: Associate to ignite fire into simulation cell.
- **EPropagation**: Programmed time for fire propagation to neighbor cell.
- **EExtinguish**: Programmed time to put out fire in a cell.
- **dataUpdate**: Event that represent a modification in the data used to calculate spread time. When this event is received is necessary to recalculate propagation model, (for instance a modification of the wind speed or direction).

Moore neighbourhood



BEHAVE model

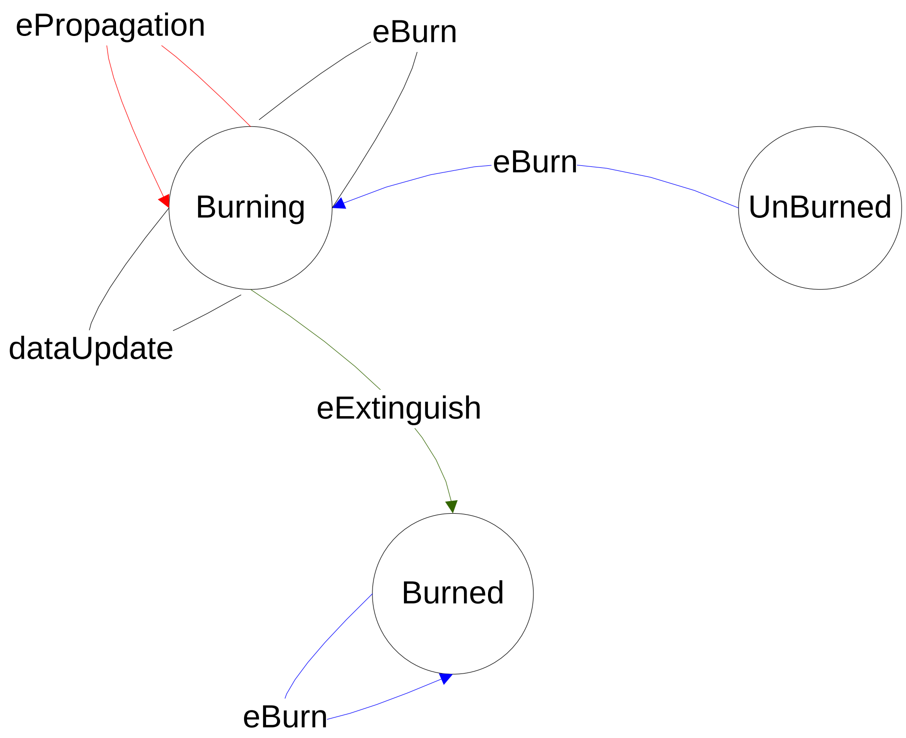


BEHAVE model

- The BEHAVE library, Andrews 1996.
- Based in a cellular automaton and a discrete simulation model.
- From a set of raster layers and an initial point the model calculates the ignition time and the elevation of the flames on each cell.
- In our model:
  - The fire starts in a know cell.
  - The results are calculated to the neighborhood cells.
  - Analyze what is the cell with the lowest ignition time.
  - Recalculate the results for this cell.
  - This loop is repeated while exist cells in the model.

States diagram

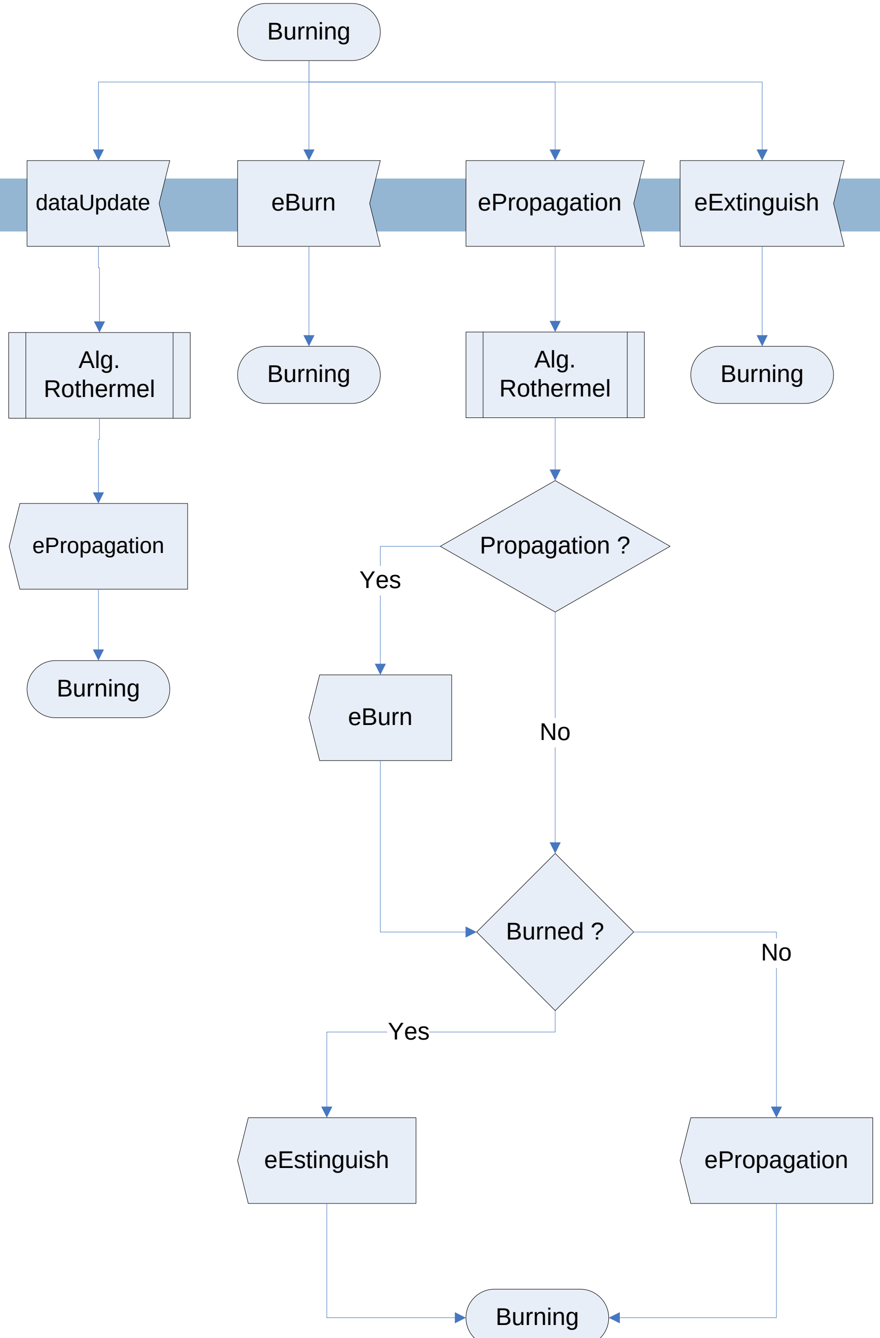
- **EBurn**: Associate to ignite fire into simulation cell.
- **EPropagation**: Programmed time for fire propagation to neighbor cell.
- **EExtinguish**: Programmed time to put out fire in a cell.
- **dataUpdate**: Event that represent a modification in the data used to calculate spread time. When this event is received is necessary to recalculate propagation model, (for instance a modification of the wind speed or direction).



UNIVERSITAT POLITÈCNICA DE CATALUNYA

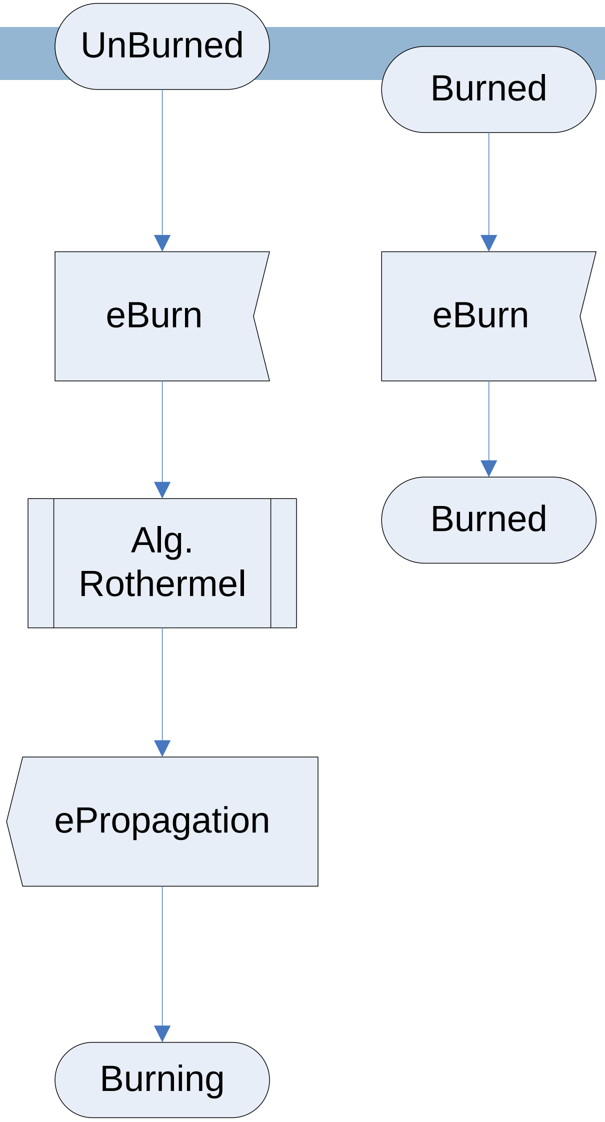
Process diagram

- The third level of the SDL formalism.

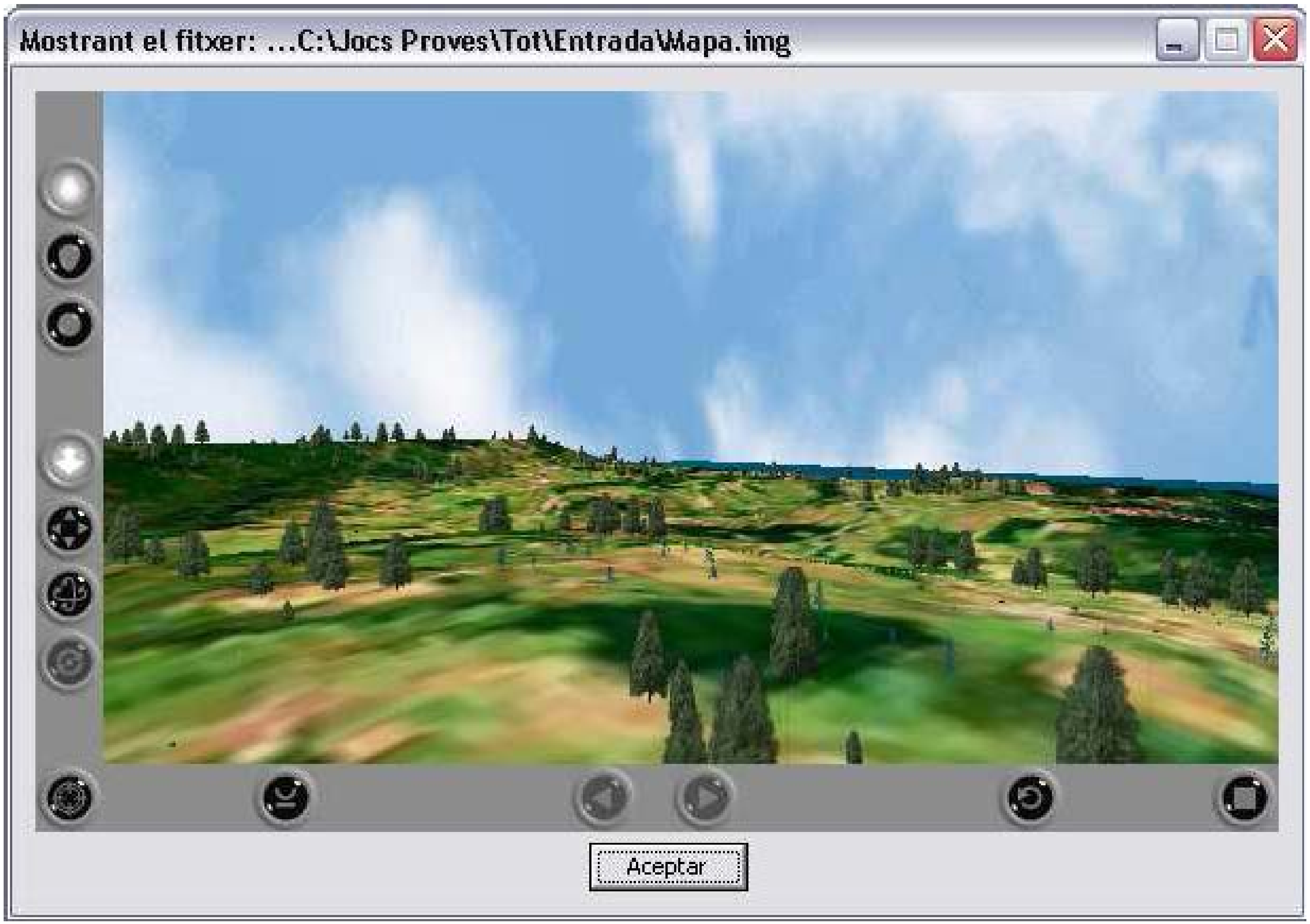


UNIVERSITAT POLITÈCNICA DE CATALUNYA

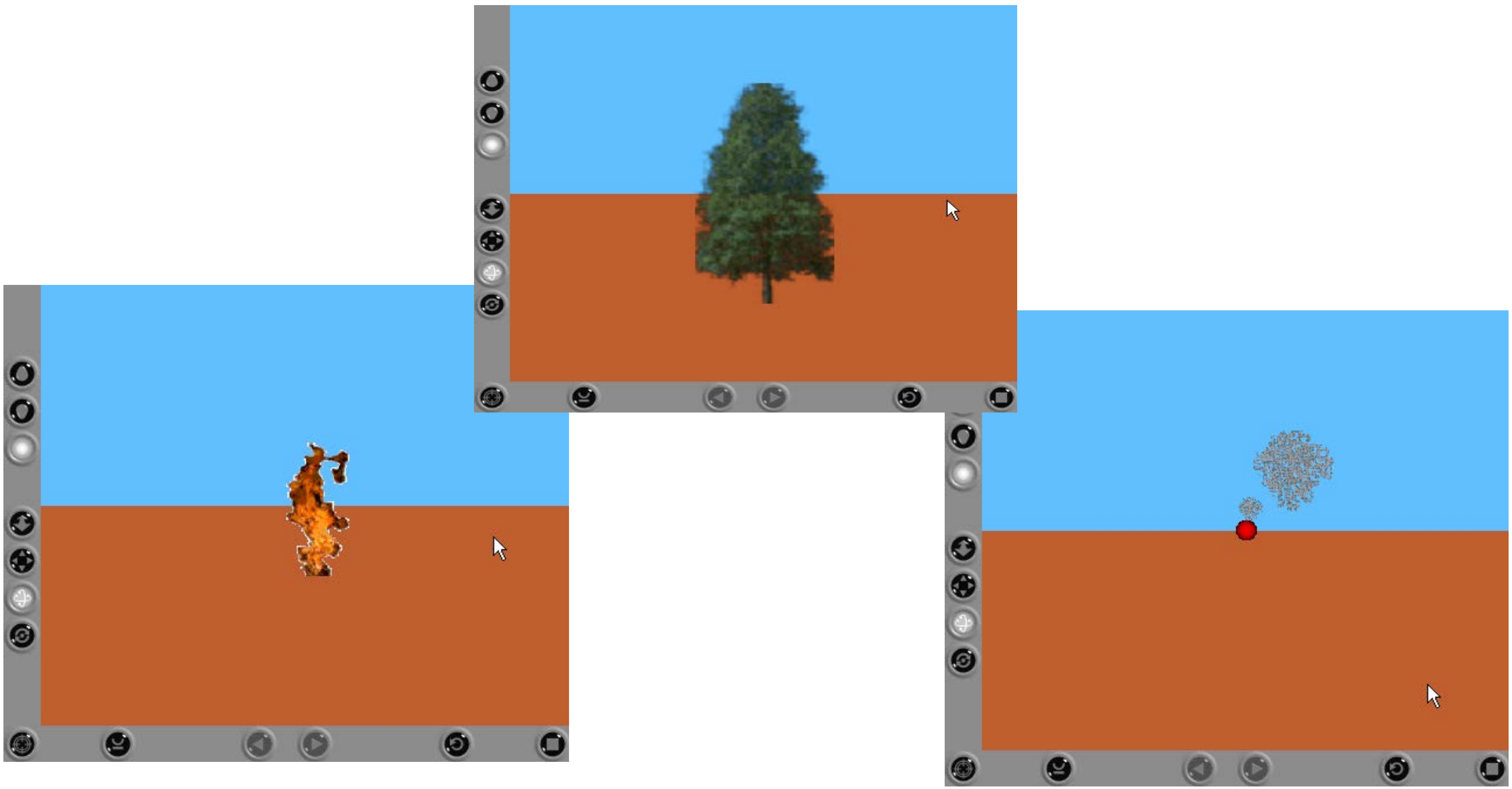
Process diagram



Simulation model



Visual efects: objects

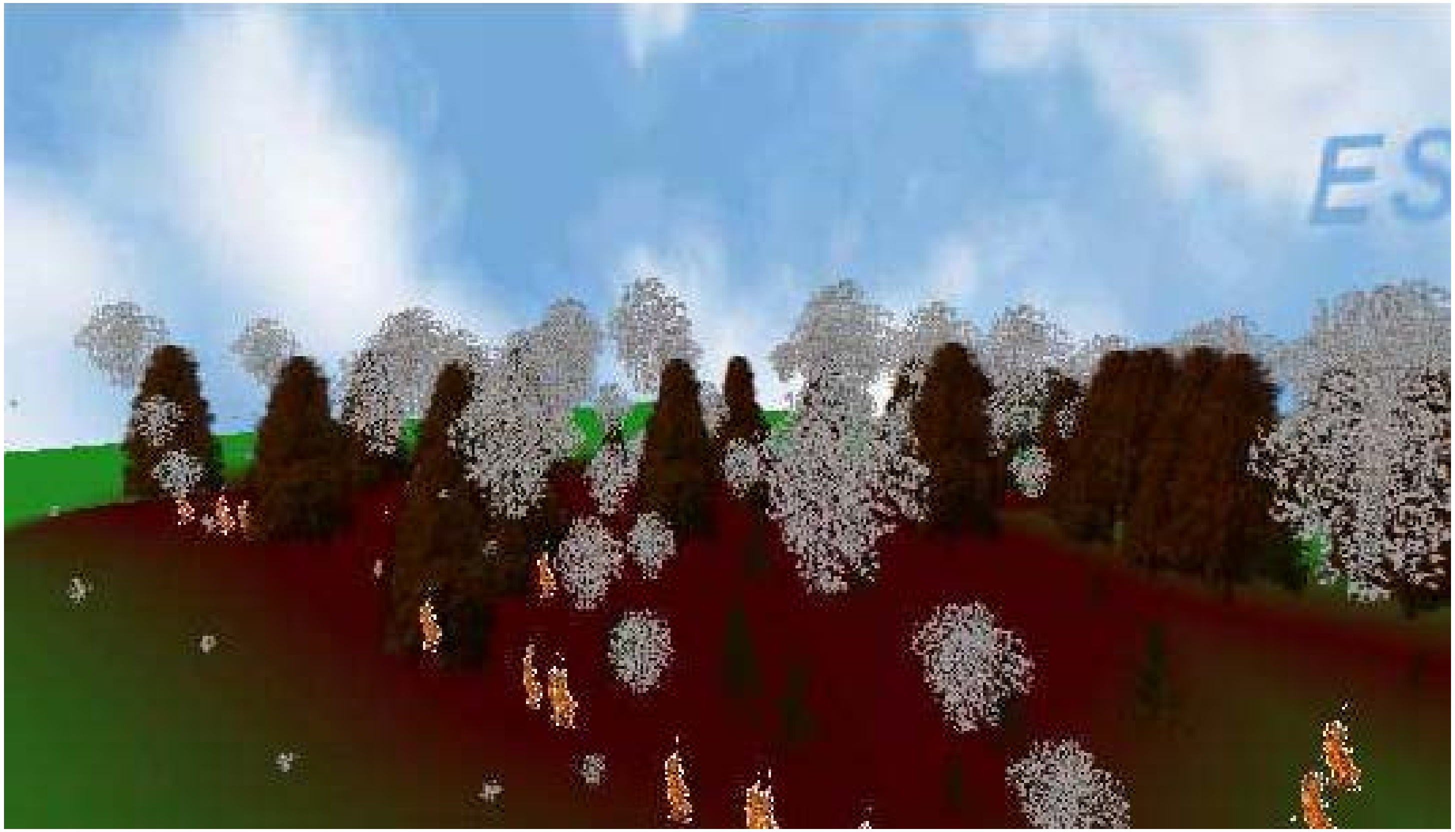


UNIVERSITAT POLITÈCNICA DE CATALUNYA

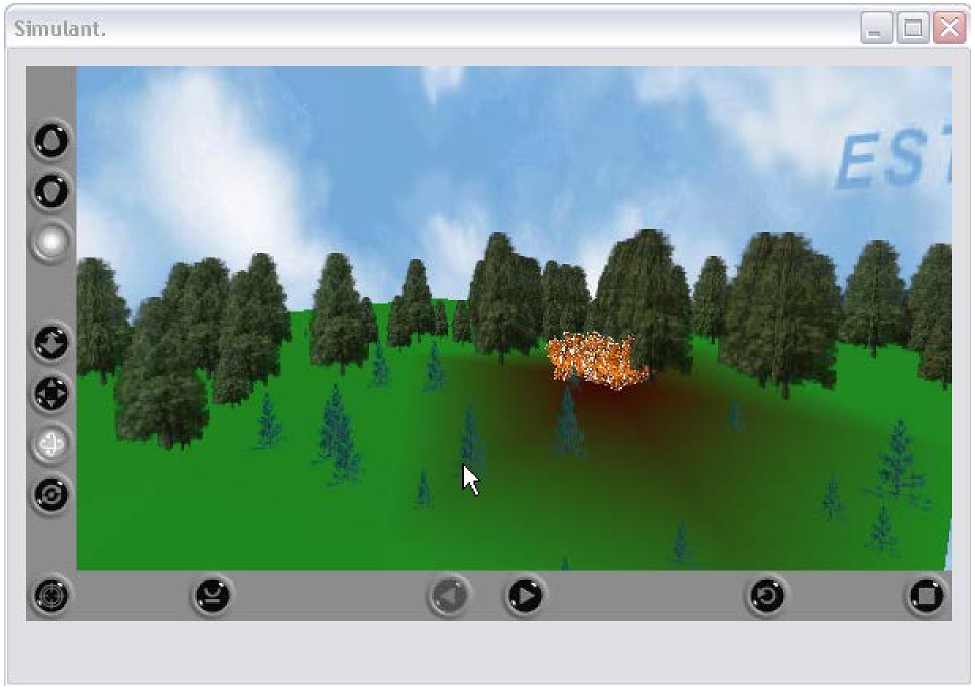
Visual efects: example

Visual efects: example

- Wildfire in action



Visual efects: example

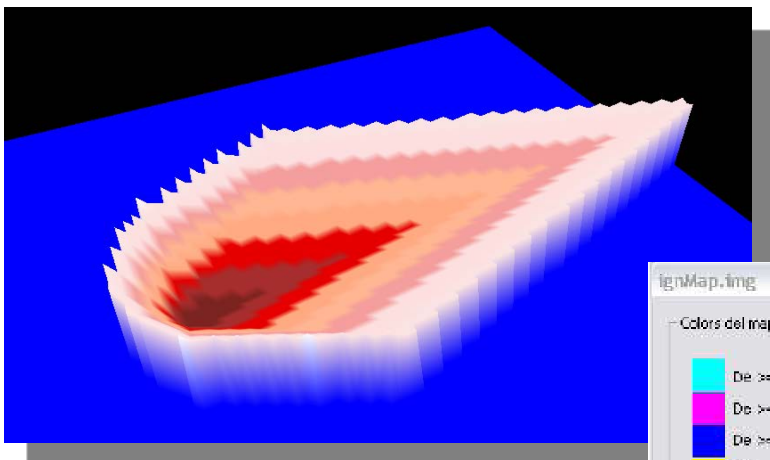


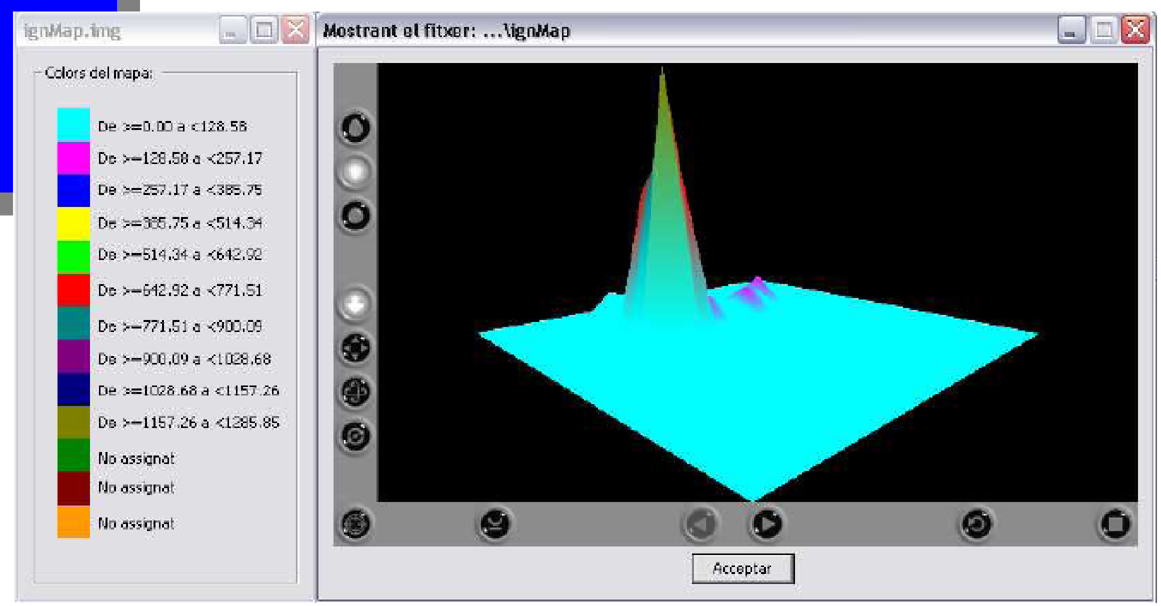
- More cells implies better results, better representation but more CPU time.

UNIVERSITAT POLITÈCNICA DE CATALUNYA

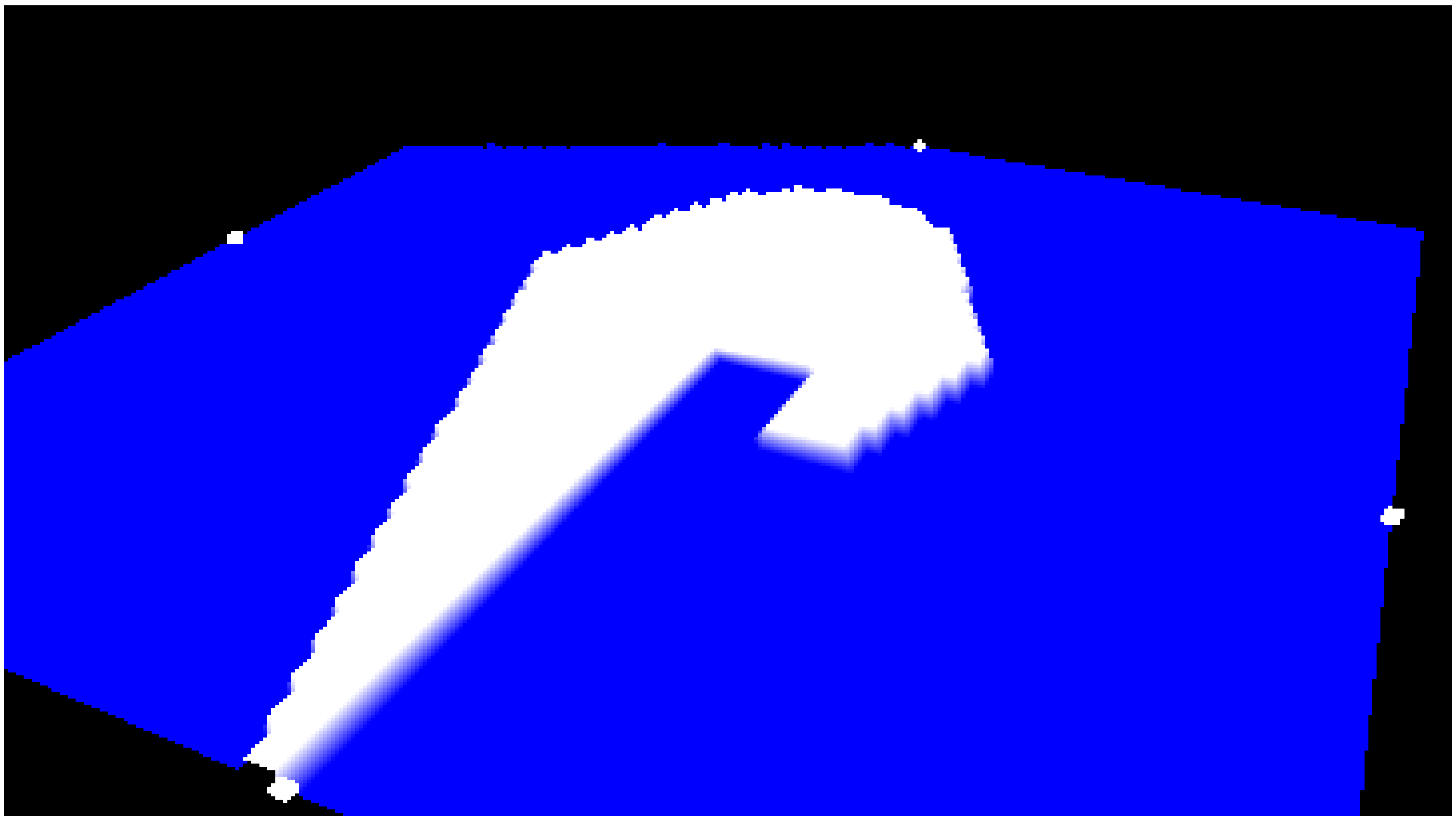
Results

- Height and temperature of the flames.





flMap.dtm (with IAgents)



Examples

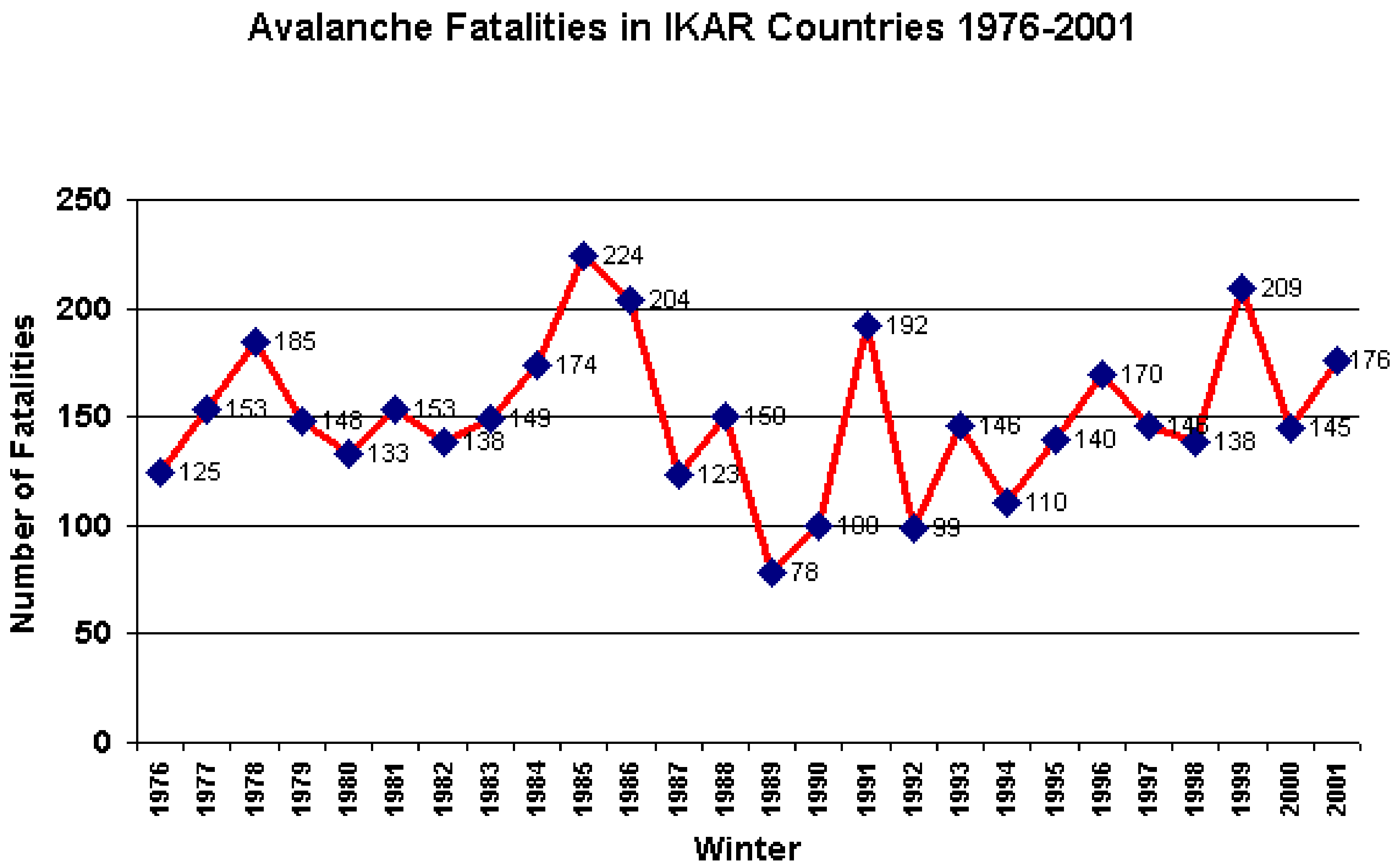
Using cellular automata to represent an slap avalanche

Avalanche

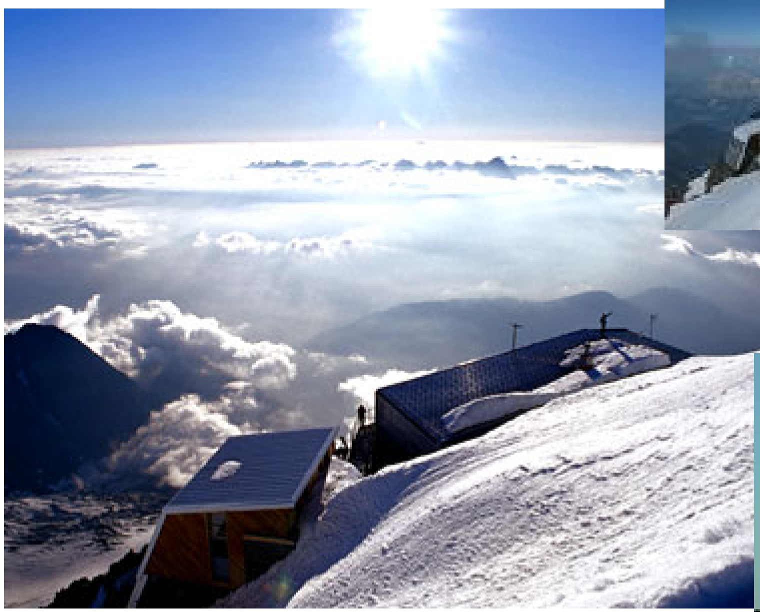
Two main types of **snow** avalanche:

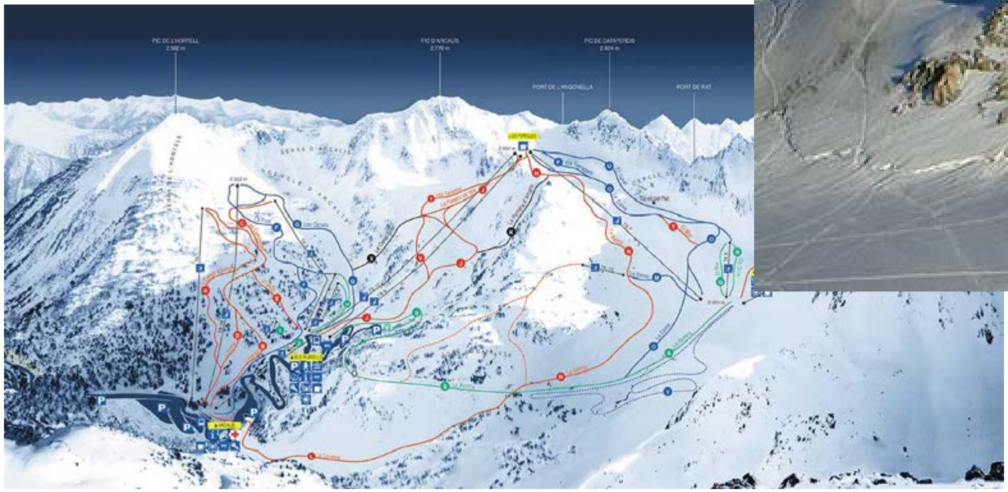
- **Loose-snow** avalanche originates at a point and propagates downhill by successively dislodging increasing numbers of poorly cohering snow grains, typically gaining width as movement continues down slope.
- **Slab avalanche**, occurs when a distinct cohesive snow layer breaks away as a unit and slides because it is poorly anchored to the snow or ground below

Avalanche fatalities in IKAR Countries



Some photos





Avalanche Model data

Name	Type	Description	Qtt	Source	Modifiable
Height	Raster	Layer representing the height of the environment.	1	ICC	No
Thickness of the snow	Raster	Represents the thickness of the "slab snow"	1	Meteocat	Yes
Floor features	Raster	Represents the kind of surface (rocks, sand, snow, ice,..). Each surface has his own specific rough parameter.	1	Meteocat Creaf	No
Snow that causes the slab features	Raster	Density, compactness of the snow.	1	Meteocat	Yes*
Obstacles	Raster	Represents the obstacles that have the environment (small rocks, big rocks, houses, trees,…)	N	Creaf	Yes
Crack	Vectorial	Line representing the breakdown of the ice.	1	Input data	Yes, at beginning.
State of the snow	Raster	Shows the state of the terrain, **empty**, **static** and **dynamic**	1	Meteocat	Yes

Avalanche Model

- $6+N:2\text{-}AC^{4+N}$ on Z^2

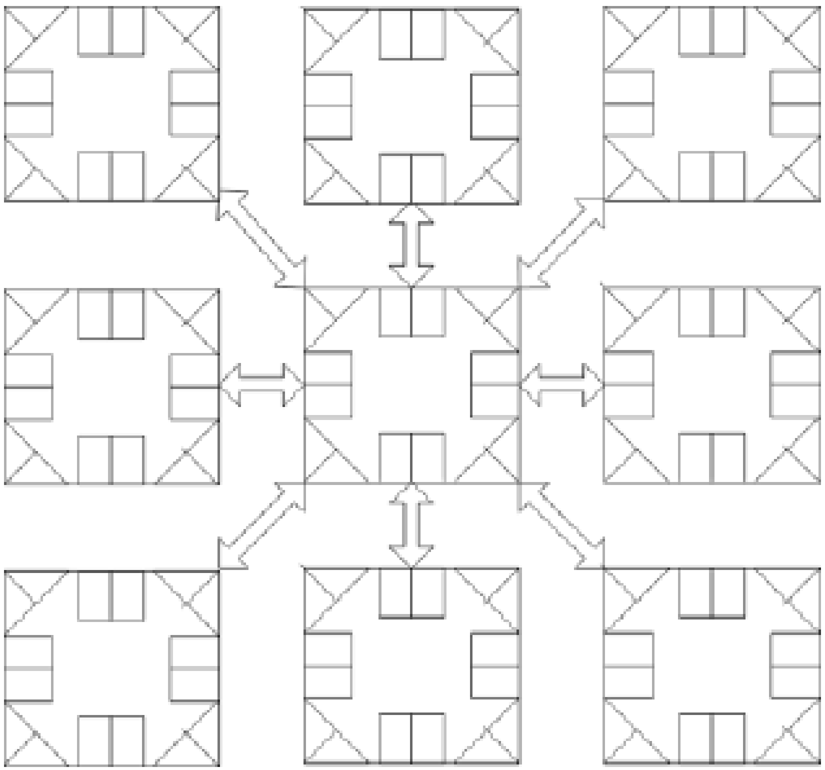
Vicinity and nucleus function

- Vicinity function: $vn(x_1,x_1) = \{(x_{1-1},x_{2-1}), (x_{1-1},x_2), (x_{1-1},x_{2+1}), (x_1,x_{2-1}), (x_1,x_2), (x_1,x_{2+1}), (x_{1+1},x_{2-1}), (x_{1+1},x_2), (x_{1+1},x_{2+1})\}$
- Nucleus function: $nc(x_1,x_1)= \{(x_1,x_1)\}$

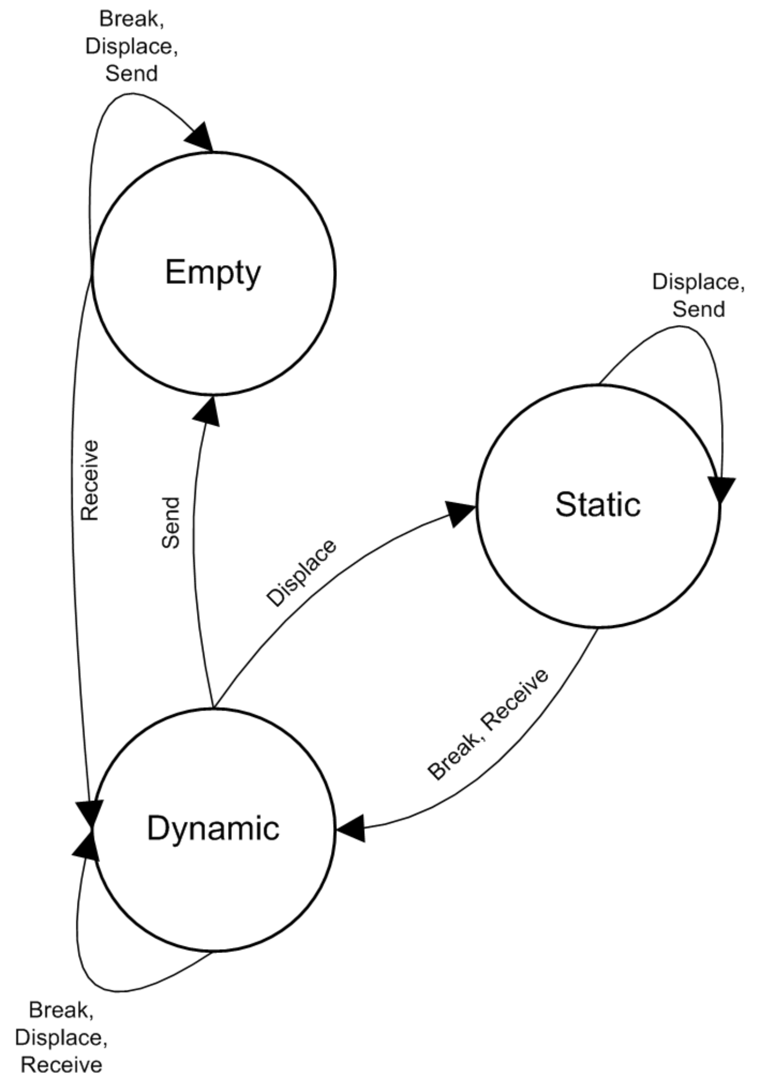
Evolution functions

- $E_2[i]$: Thickness of the snow. The function that rules this layer is "Modify information(p)"
- $E_4[i]$: Density, compactness of the snow, in our case is 0.5 (Mears 1976).
- $E_6[i]$: State of the snow. The function is defined in the next diagrams.
- $E_N[i]$: Obstacles. The function that defines the obstacles we use in the model.

Moore neighbourhood

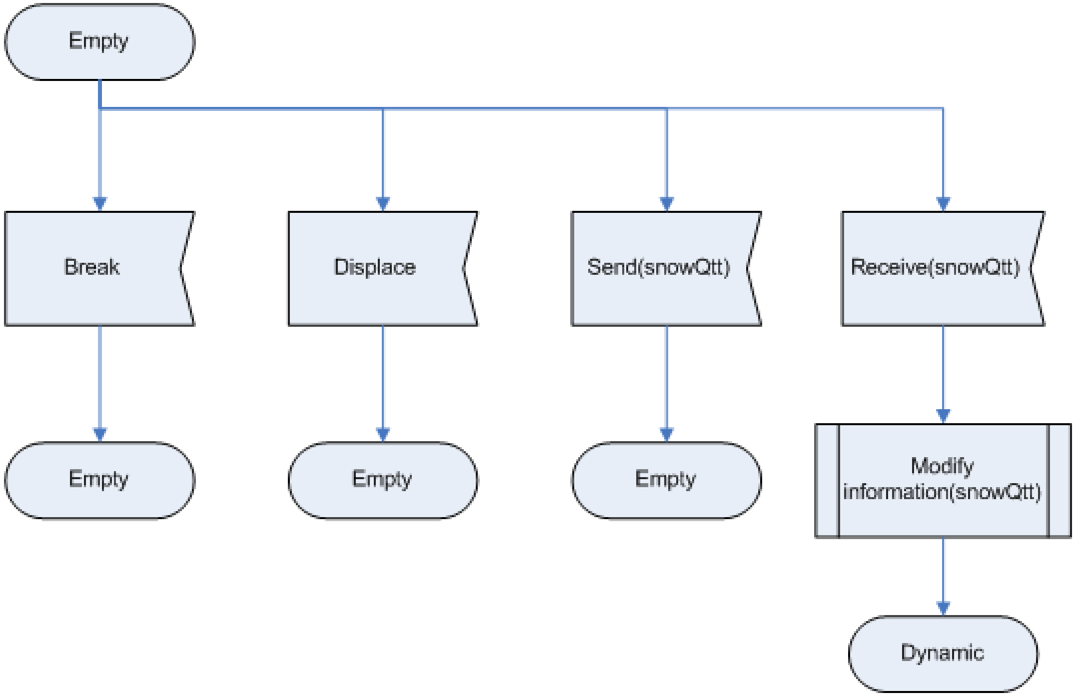


Λ:state of the snow

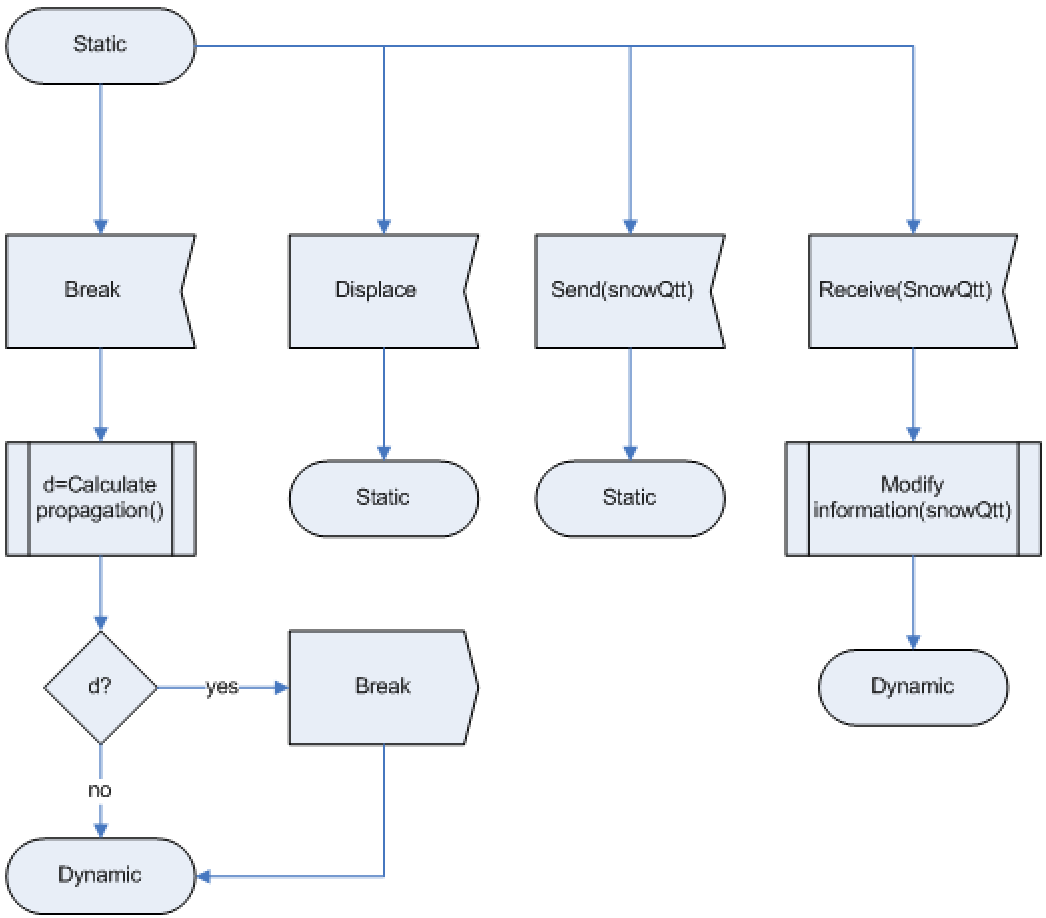


UNIVERSITAT POLITÈCNICA DE CATALUNYA

Empty process

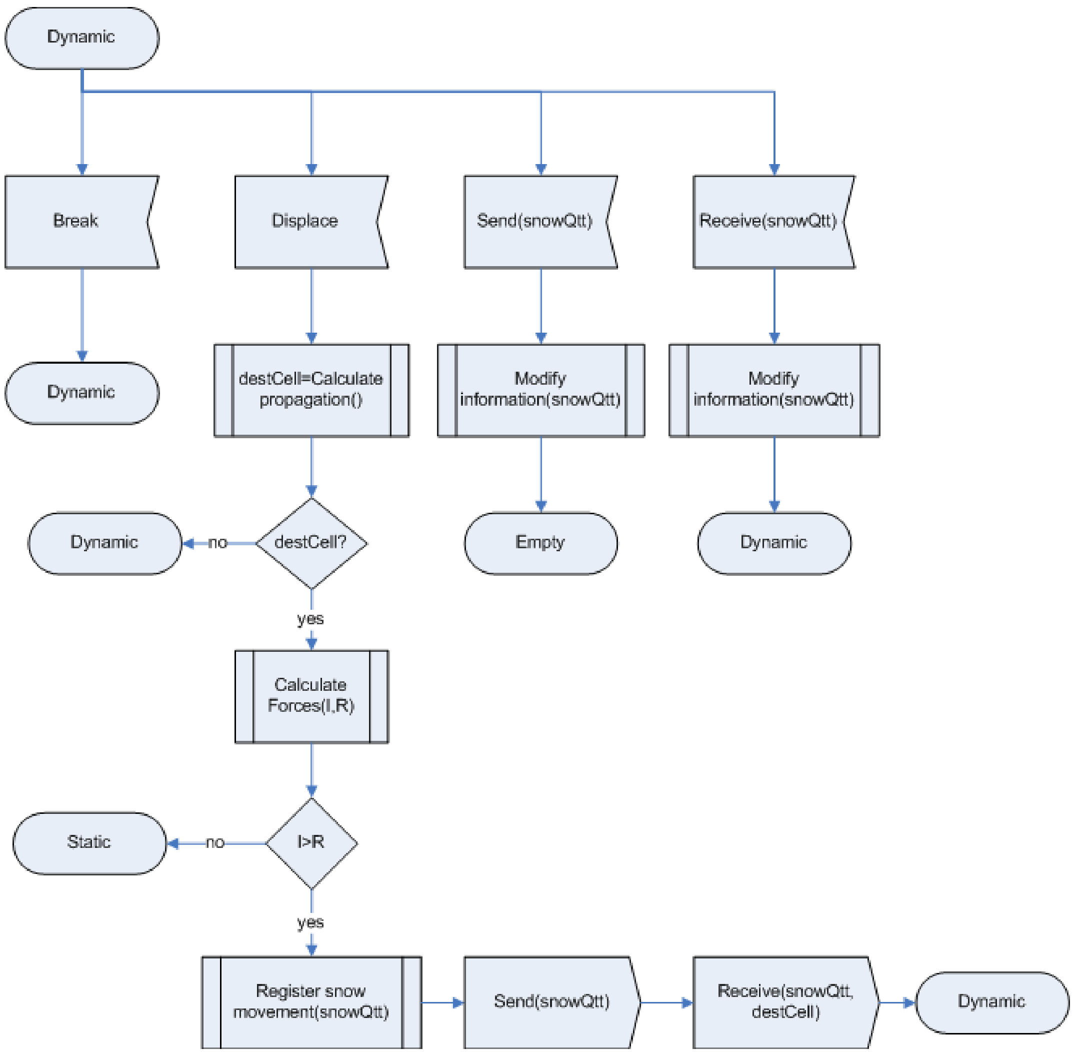


Static process



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Dynamic process



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Evolution function

- The increment in the force is used in the next expression to determine if the snow continues its movement to other cell, or stops its movement, if the force is equal to zero.

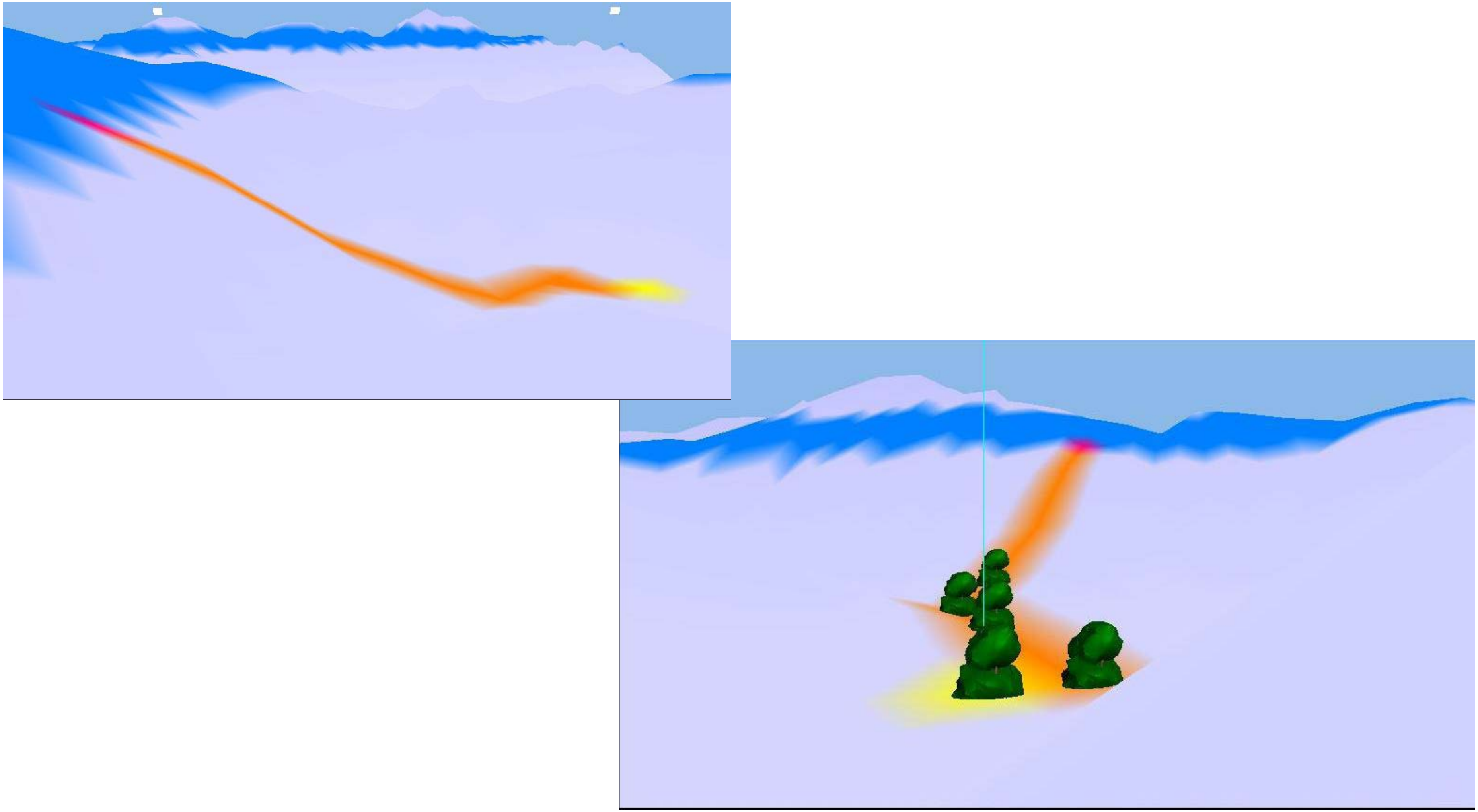
$$F_{i,t} = \max(IF_{i,t} + \Delta F_{i,t}, 0)$$

Evolution function

- $IF_{i,t}$=Impulse force, depends on the quantity and quality of the snow, and the slope.
- $SFF_{i,t}$= Sliding friction force between the avalanche and the underlying snow or ground.
- $IFF_{i,t}$= Internal dynamic shear resistance due to collisions and momentum exchange between particles and blocks of snow, (internal friction force).
- $ASFF_{i,t}$= Turbulent friction within the snow/air suspension, (air suspension friction force).
- $AFF_{i,t}$=Shear between the avalanche and the surrounding air, (air friction force).
- $FFF_{i,t}$= Fluid-dynamic drag at the front of the avalanche (front friction force).
- $OFF_{i,t}$=Obstacle friction force.

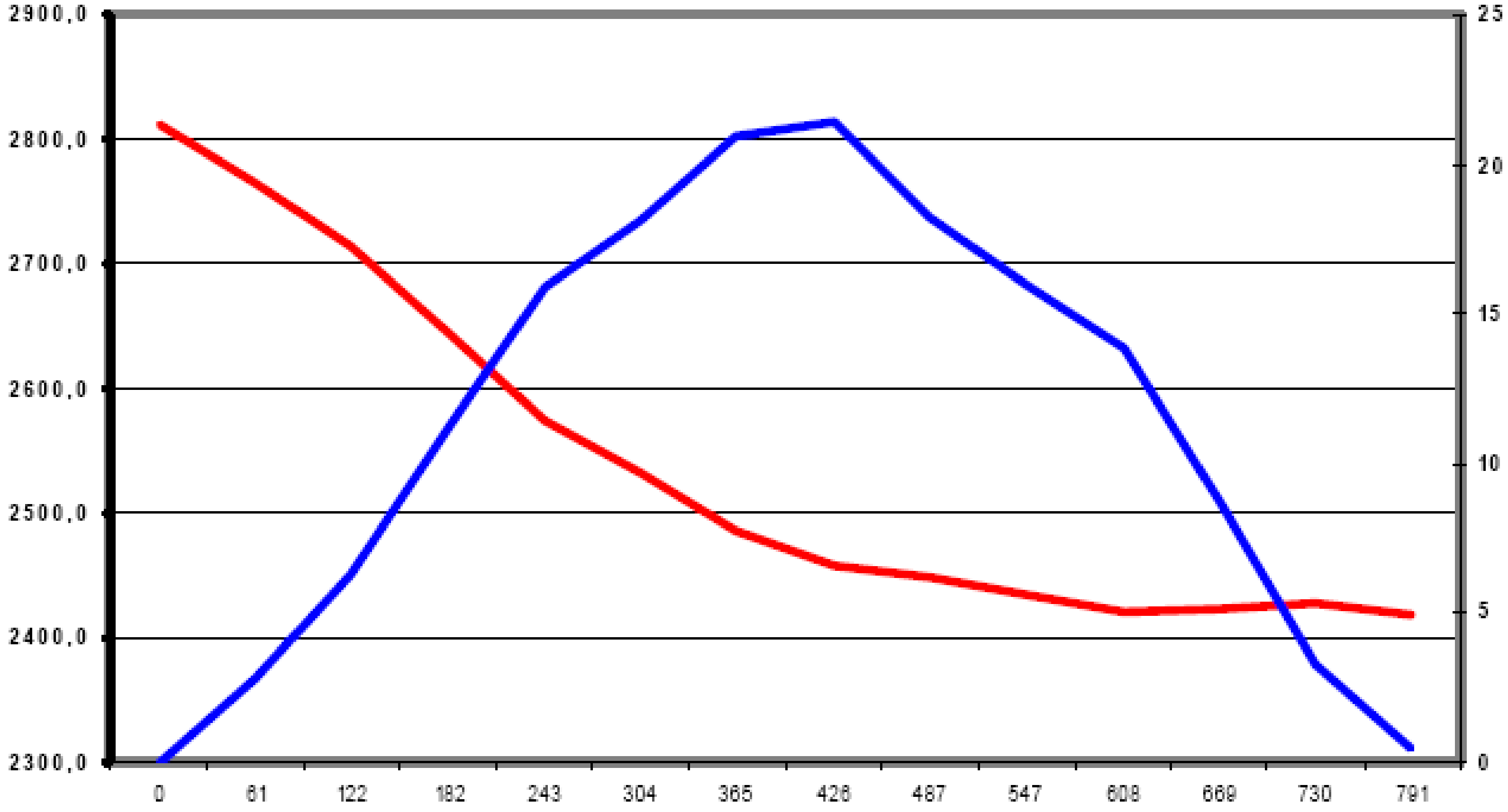
$$\Delta F_{i,t} = IF_{i,t} - (SFF_{i,t} + IFF_{i,t} + ASFF_{i,t} + AFF_{i,t} + FFF_{i,t} + OFF_{i,t})$$

Results



UNIVERSITAT POLITÈCNICA DE CATALUNYA

Results


2900,0
2800,0
2700,0
2600,0
2500,0
2400,0
2300,0
25
20
15
10
5
0
0
61
122
182
243
304
365
426
487
547
608
669
730
791

Results (5)

Desencadenant:

Localització: 8 cel·les, de (108, 33) fins (115, 33)

Gruix de neu de placa: 50cm (per totes les cel·les fracturades)

Terreny subjacent: Neu dura

Obstacles: No

Característiques de l'allau:

Terreny subjacent del camí: Neu dura

Màxima distancia recorreguda: 1101,14m

Desnivell superat: 520,40m

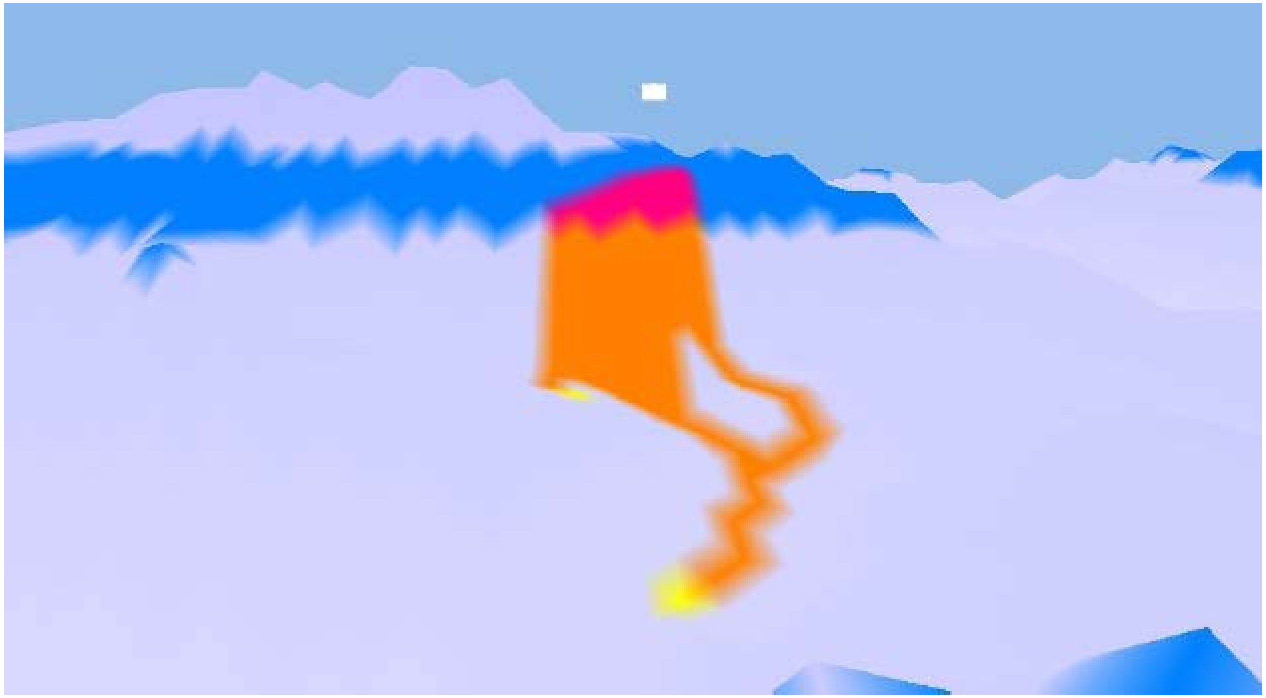
Massa transportada: 10625kg

Massa de neu en dipòsit: 9957,50kg

Massa de neu perduda pel camí: 667,5kg

Velocitat màxima: 67,23m/s

Results (5)



Results (5)

Results (1 vs 3)

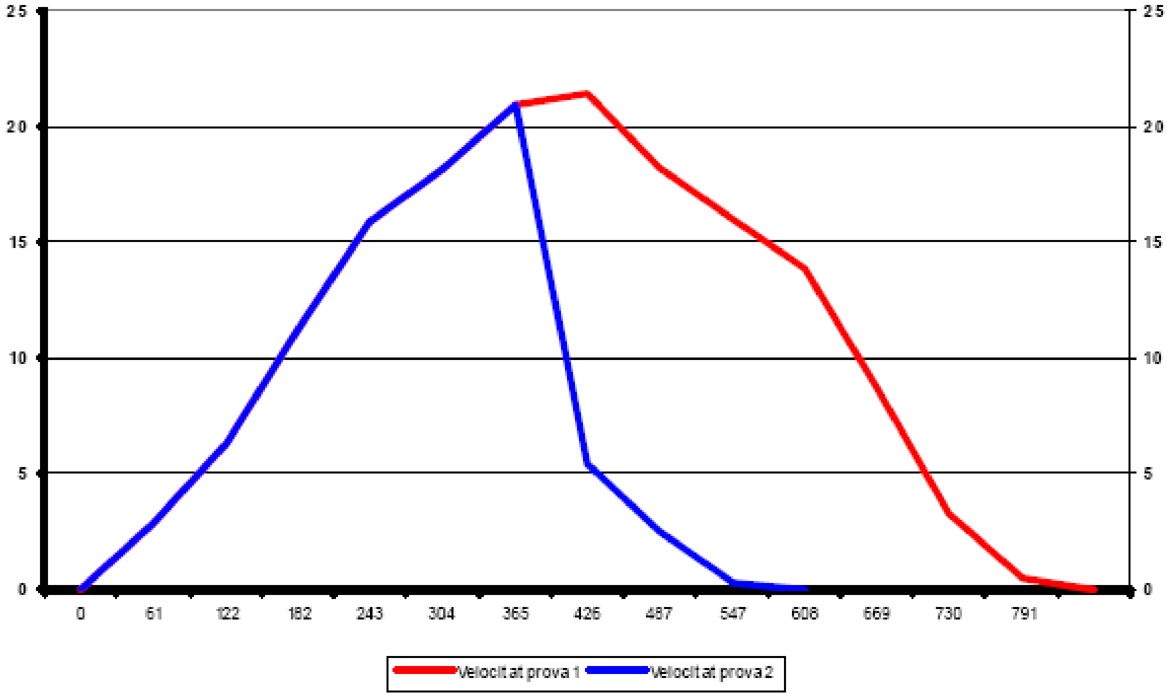


Figura 7.9 – Velocitat Sim.1 VS Velocitat Sim.3

Some references

- Wuensche, Andrew; Lesser, Mike. 1992, *The gobal dynamics of cellular automata*, Santa Fe institute, studies in the sciences of complexity, reference volume I, Addison Wesley, 1992
- Teran Villegas, Oswaldo Ramon. 2001, *Emergent Tendencies in Multi-Agent-based Simulations using Constraint-based Methods to Effect Practical Proofs over Finite Subsets of Simulation Outcomes*, PhD Manchester Metropolitan University
- Fonseca P.; Casanovas J. 2005. ESS205, *Simplifying GIS data use inside discrete event simulation model through m:n-AC cellular automaton*; Proceedings ESS 2005.

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Some references

- Doran, Jim. E. 2000, *HARD PROBLEMS IN THE USE OF AGENT-BASED MODELLING,* Proceedings of the Fifth International Conference on Logic and Methodology, Cologne, October 3-6
- Roher, G. 2000, *Can Simulation models Support Social Research? A Critical Discussion.*
- Itzhak Benenson, Itzhak Omer. 2001, *Agent-Based modeling of residential Distribution*. Departament of Geography and human Environment University Tel Aviv. Israel.
- Epstein, Joshua M. Axtell, Robert. 1996 *Growing Artificial Societies: Social Science from the Bottom Up*. MIT Press, Cambridge, MA, USA.

Some references

- Torrens, Paul M. Benenson, Itzhak. 2004, Geosimulation, John Wiley & Sons Ltd., ISBN: 0-470-84349-7
- Batty, Michael, 2995, *Cities and complexity*.The MIT Press.

Thanks!

Pau Fonseca i Casas

pau@fib.upc.edu

Technical University of Catalonia

Barcelona School of Informatics
Computing laboratory

Barcelona

Jordi Girona 1-3

(+34)93401773

UPC UNIVERSITAT POLITÈCNICA DE CATALUNYA