

---

Pietro Terna

Department of Economics and Public Finance, University of Torino, Italy  
terna@econ.unito.it  
web.econ.unito.it/terna

# Applying Agent Based Simulation to a Huge Actual Problem: the Interbank Payment System

ESSA Summer School, 18 - 22 July, 2011, University of Surrey

---

---

## How to start a huge simulation project

---

# How to start a huge simulation project

**vs.**

your boss, your supervisor,

yourself

Build a (tiny) proof of concept

For the the actual problem of delays and lacking liquidity in interbank payments, the starting point has been a “visual” description and implementation:

<http://jasss.soc.surrey.ac.uk/12/1/2.html>



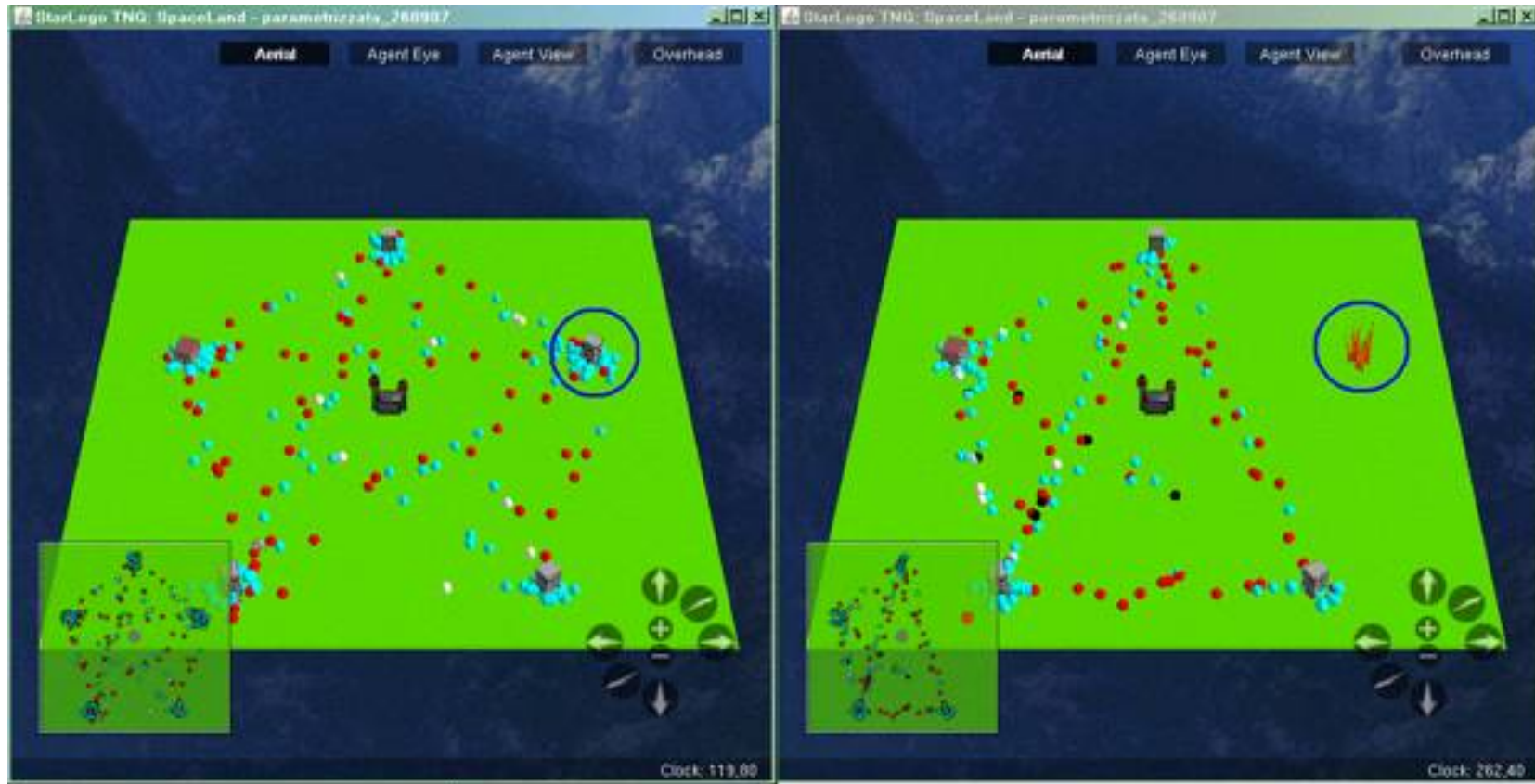
Luca Arciero, Claudia Biancotti, Leandro D'Aurizio and Claudio Impenna (2009)

## **Exploring Agent-Based Methods for the Analysis of Payment Systems: A Crisis Model for StarLogo TNG**

The paper presents an exploratory agent-based model of a real time gross settlement (RTGS) payment system.

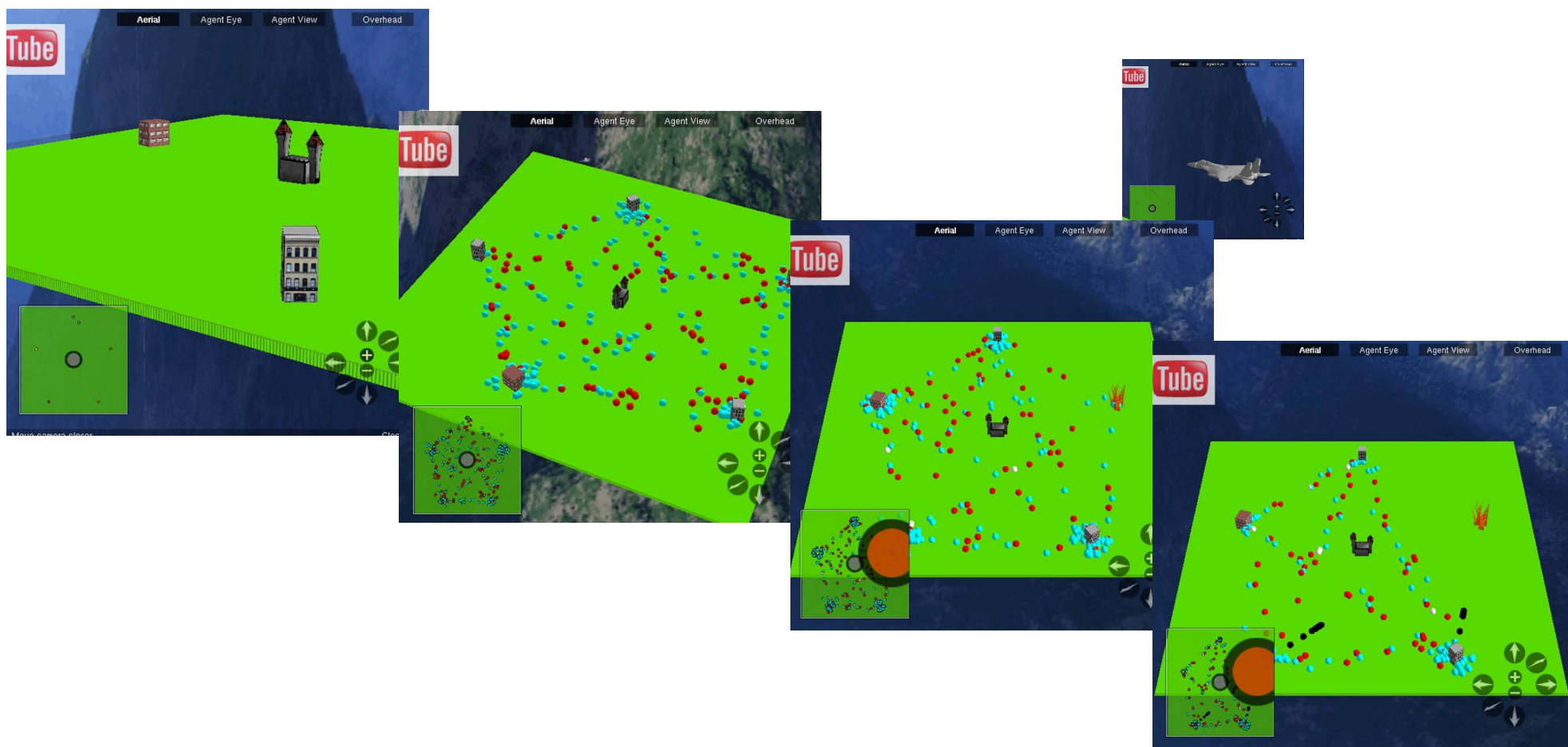
Banks are represented as agents who exchange payment requests, which are then settled according to a set of simple rules. The model features the main elements of a real-life system, including a central bank acting as liquidity provider, and a simplified money market.

A simulation exercise using synthetic data of the Italian RTGS predicts the macroscopic impact of a disruptive event on the flow of interbank payments.



The simulated world is depicted one second before (left) and 140 seconds after (right) a critical event hitting the circled agent.

# Have a look to a movie reproducing the simulation

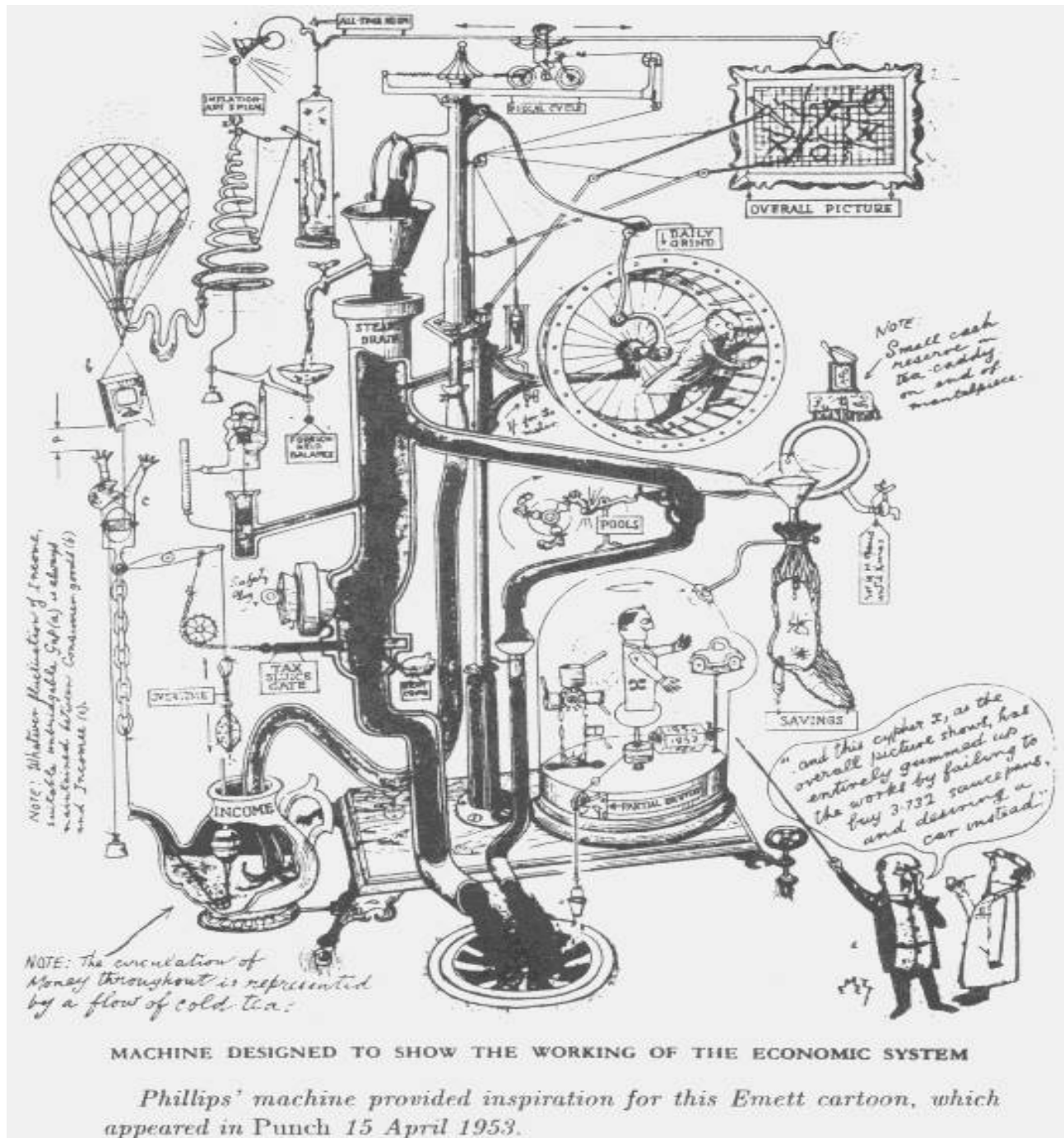


---

## Moving to a realistic scale model

---

Our goal:  
 understanding  
 and controlling  
 a tiny part of  
 the economic  
 complexity in  
 action



MACHINE DESIGNED TO SHOW THE WORKING OF THE ECONOMIC SYSTEM

Phillips' machine provided inspiration for this Emmett cartoon, which appeared in Punch 15 April 1953.

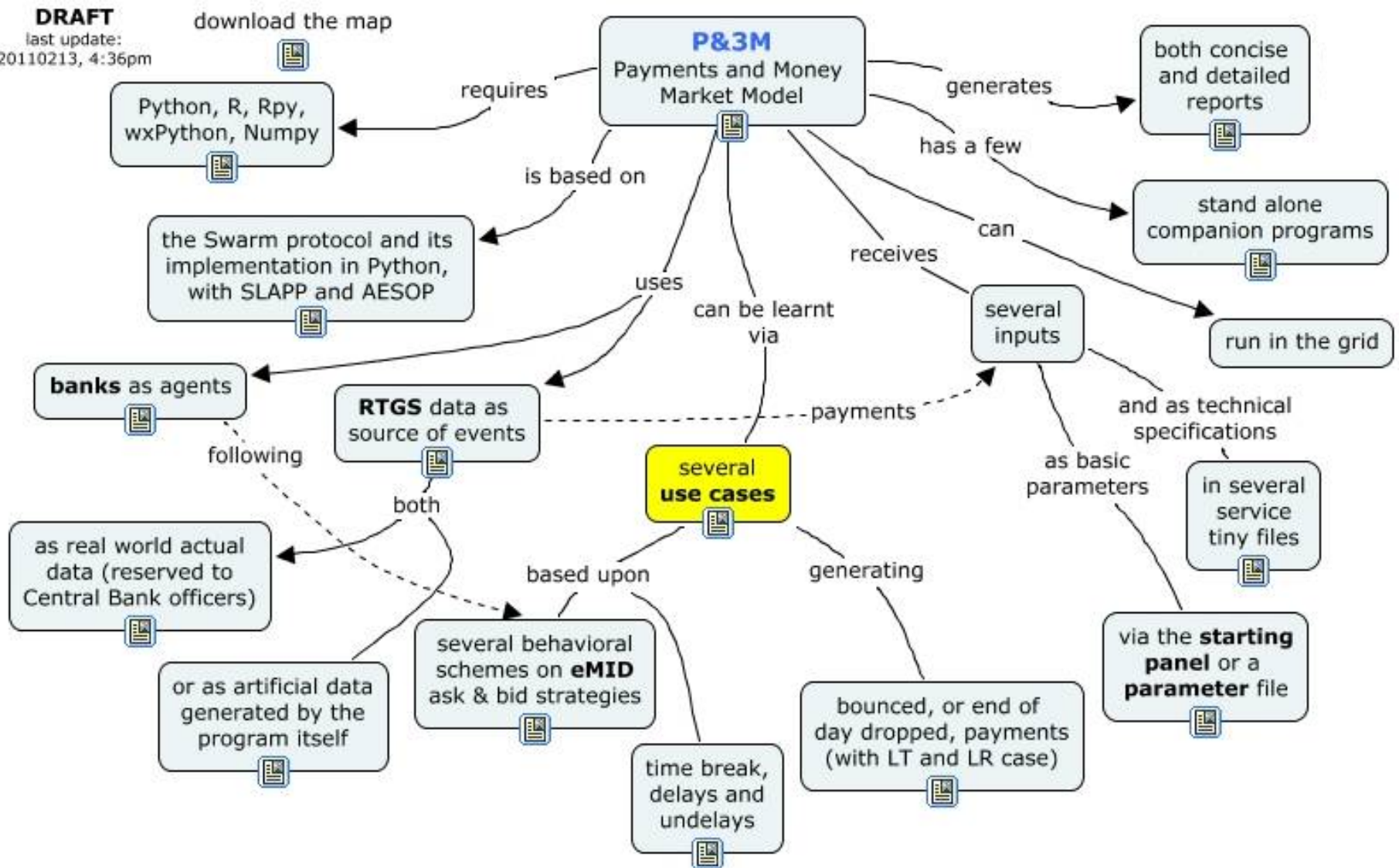
Moving to an actual scale model of the same problem, with a conceptual map showing the structure of the model and of its “making of”, at

<http://eco83.econ.unito.it/terna/P&3M/P&3M.html>

with a basic formal description (with Bank of Italy colleagues: Luca Arciero, Claudia Biancotti, Leandro D'Aurizio, Giuseppe Ilardi, Claudio Impenna, Cristina Picillo, at

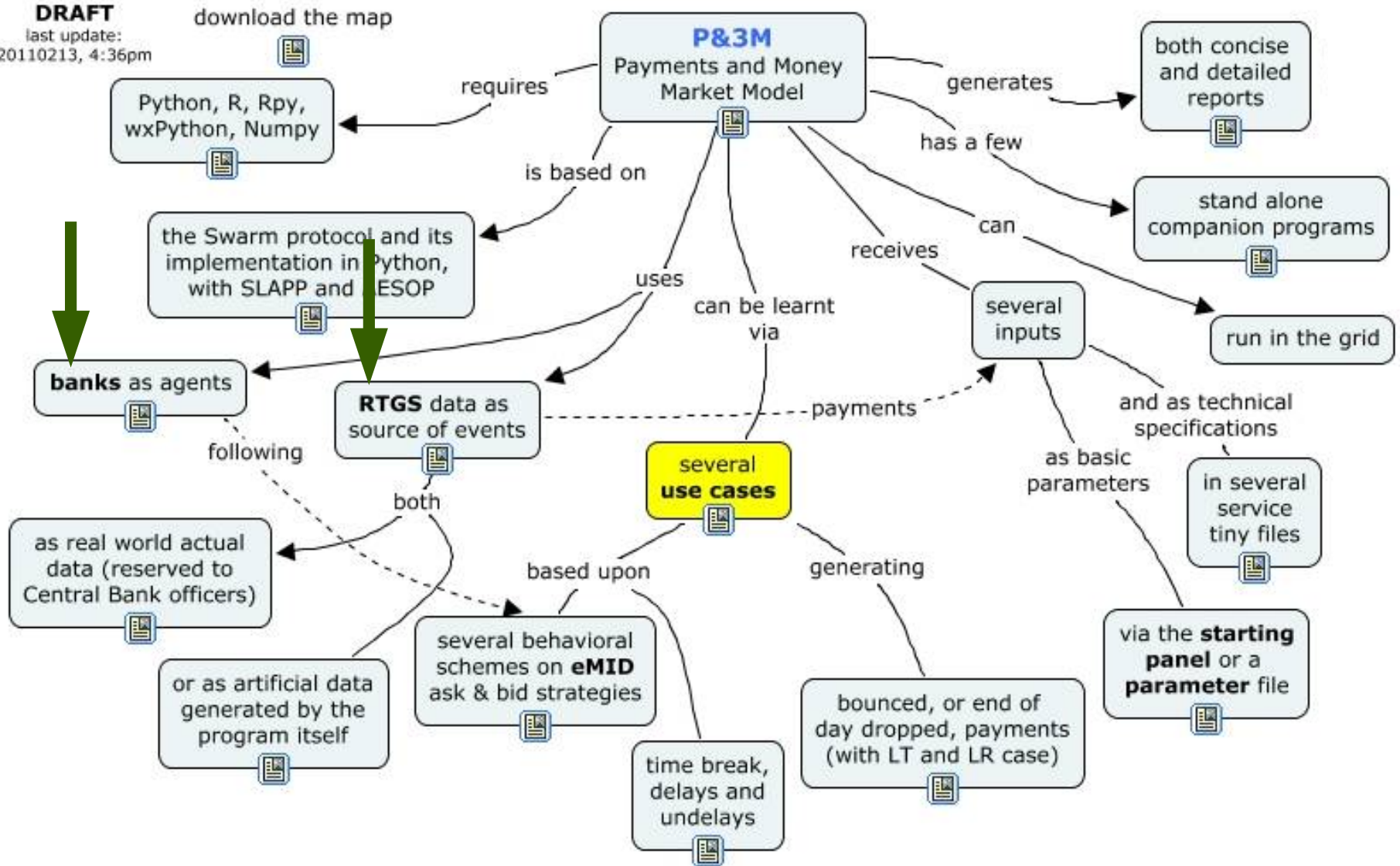
[http://eco83.econ.unito.it/terna/abm-baf09/abstracts\\_papers/Ilardi%28paper%29\\_ABM-BaF09.pdf](http://eco83.econ.unito.it/terna/abm-baf09/abstracts_papers/Ilardi%28paper%29_ABM-BaF09.pdf)

**DRAFT**  
last update:  
20110213, 4:36pm

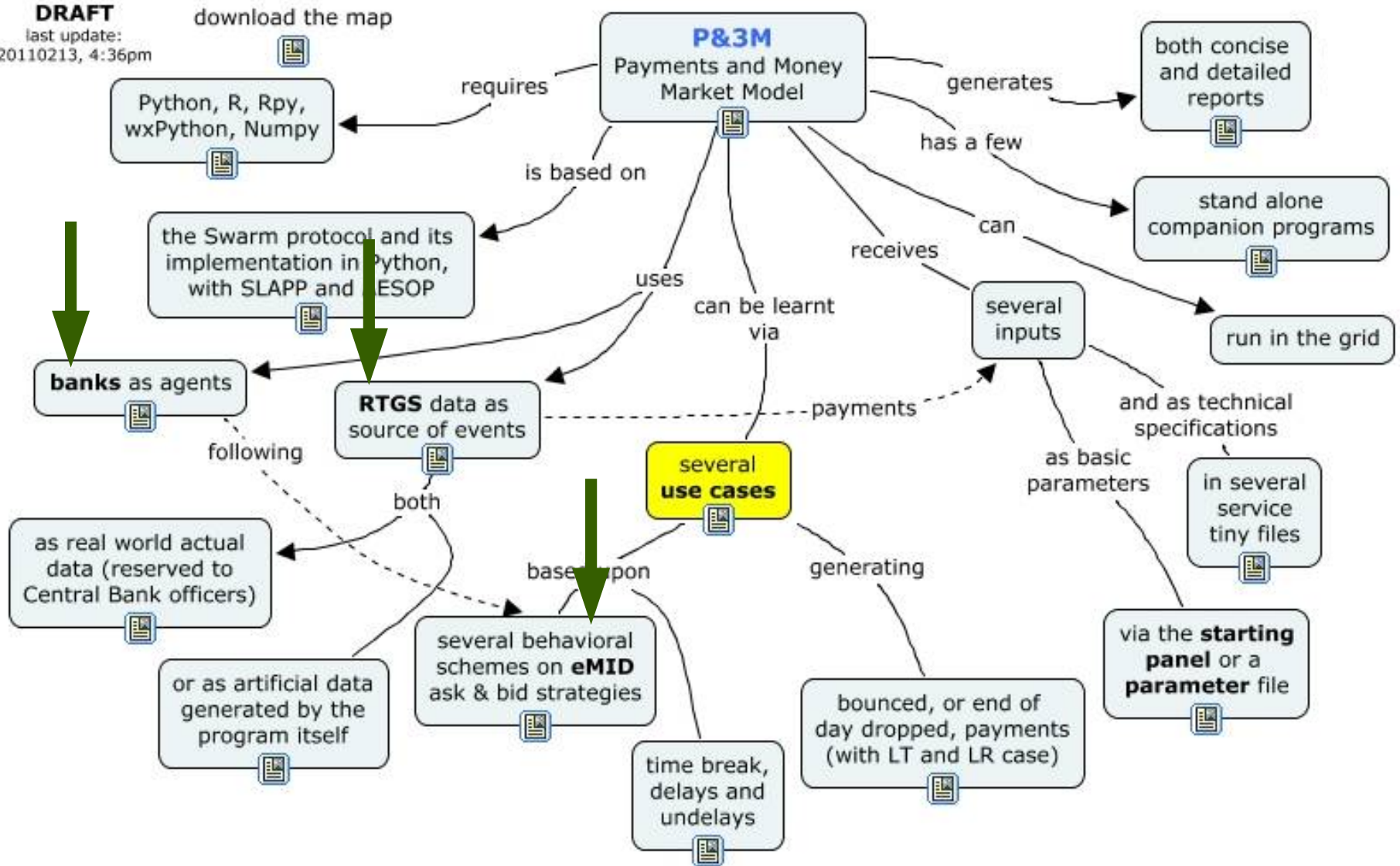




**DRAFT**  
last update:  
20110213, 4:36pm



**DRAFT**  
last update:  
20110213, 4:36pm

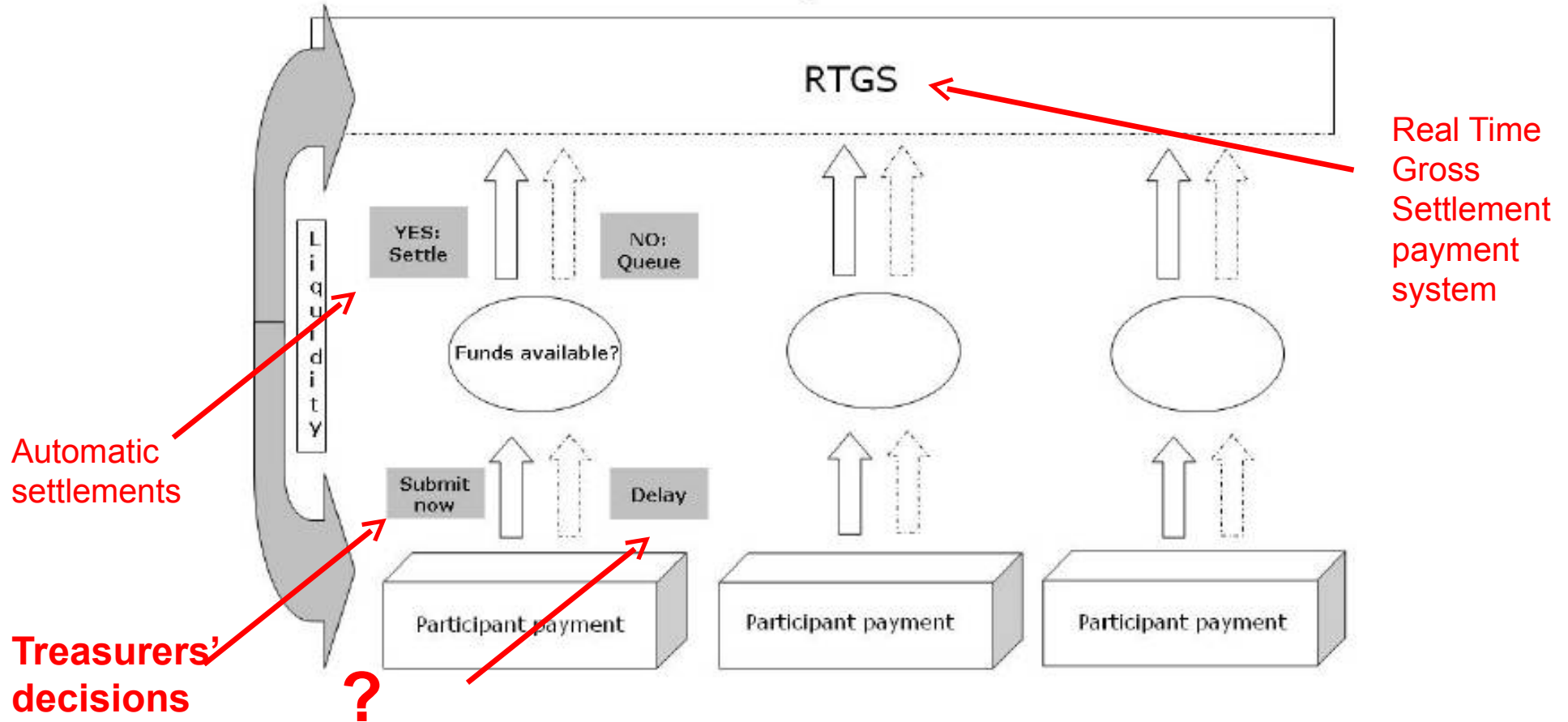


---

## The Italian Central Bank model of the internal interbank payment system

---

### Basic functioning of an RTGS environment



This figure, related to a StarLogo TNG implementation of the model, comes from: Luca Arciero+, Claudia Biancotti\*, Leandro D’Aurizio\*, Claudio Impenna+ (2009), as in slide 4.

+ Bank of Italy, Payment System Oversight Office; \* Bank of Italy, Economic and Financial Statistics Department.

- Delays\* in payments ...
- ... liquidity shortages ...
- ... in presence of unexpected negative operational or financial shocks ...
- ... financial crisis (generated or amplified by \* ), with domino effects

Two parallel highly connected institutions:

- RTGS (Real Time Gross Settlement payment system)
- eMID (electronic Market of Interbank Deposit)

Starting from actual data, we simulate delays, looking at the emergent interest rate dynamics into the eMID



**Agent based simulation as a magnifying glass to understand reality**

---

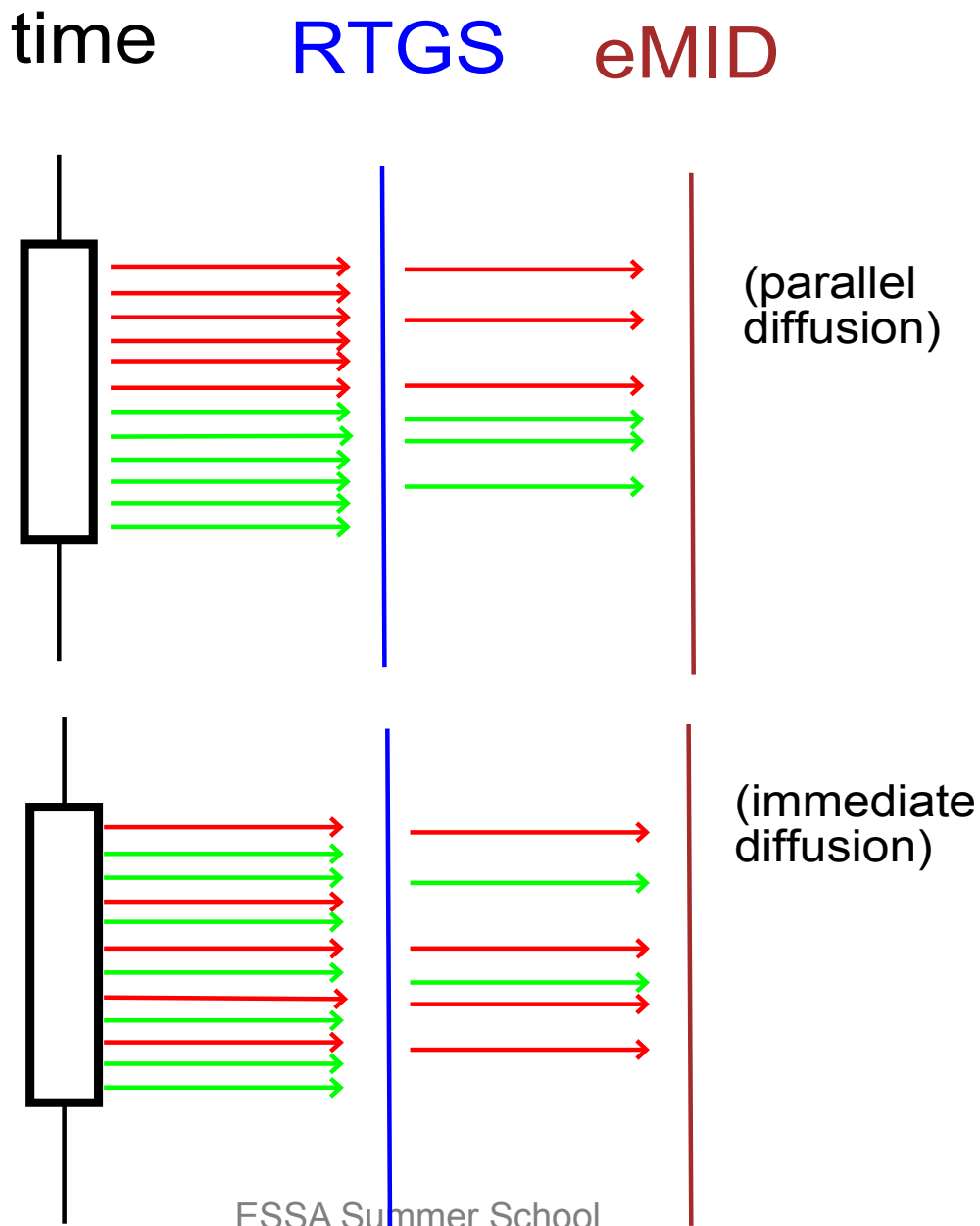
The Italian Central Bank model:  
a few complicated microstructures

---

- A treasurer making a payment: she bids a price to obtain money with  $P = p$
- A treasurers receiving a payment: she asks a price to employ money with  $P = p$

A “quasi UML”  
representation

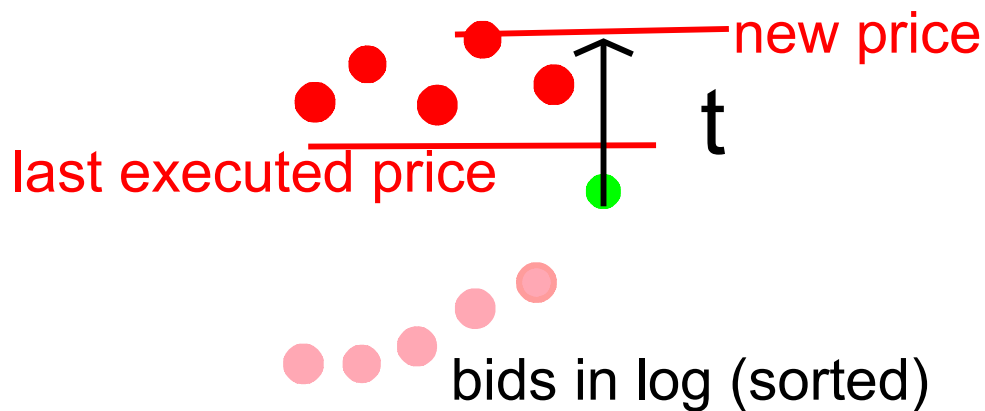
**NB prices are  
bid (or  
offered) by  
buyers and  
asked by  
sellers**



● ask

● bid

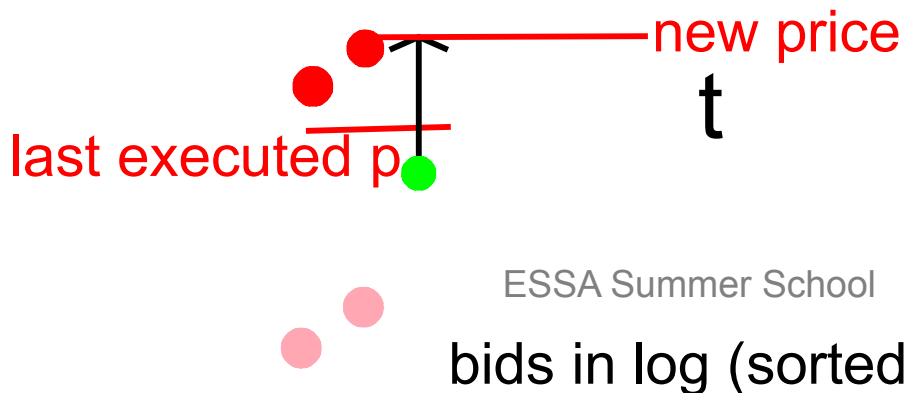
★ parallel payment diffusion, looking at the last executed price



● ask

● bid

★ immediate payment diffusion, looking at the last executed price



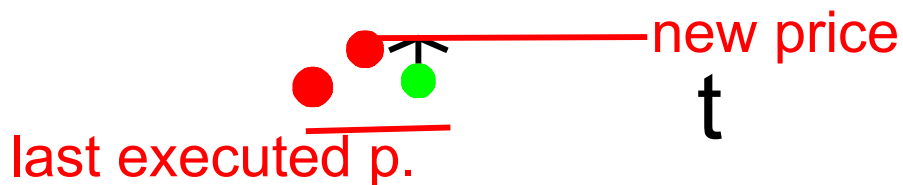
● ask ● bid

★ parallel payment diffusion, looking at the ★ best price in opposite log



● ask ● bid

★ immediate payment diffusion, looking at the ★ best price in opposite log



---

## A technical digression

---

**SLAPP**, or Swarm-Like Agent Protocol in Python, is a simplified implementation of the original Swarm protocol, choosing Python as a simultaneously simple and complete object-oriented framework.

SLAPP is a **demonstration that we can easily implement the Swarm protocol** [Minar, N., R. Burkhart, C. Langton, and M. Askenazi (1996), *The Swarm simulation system: A toolkit for building multi-agent simulations*. Working Paper 96-06-042, Santa Fe Institute, Santa Fe (\*)] **in Python**

(\*) <http://www.swarm.org/images/b/bb/MinarEtAl96.pdf>

Key points (quoting from that paper):

- *Swarm defines a structure for simulations, a framework within which models are built.*
- *The core commitment is to a discrete-event simulation of multiple agents using an object-oriented representation.*
- *To these basic choices Swarm adds the concept of the "swarm," a **collection of agents with a schedule of activity.***

---

**The container:**  
the Swarm protocol

---

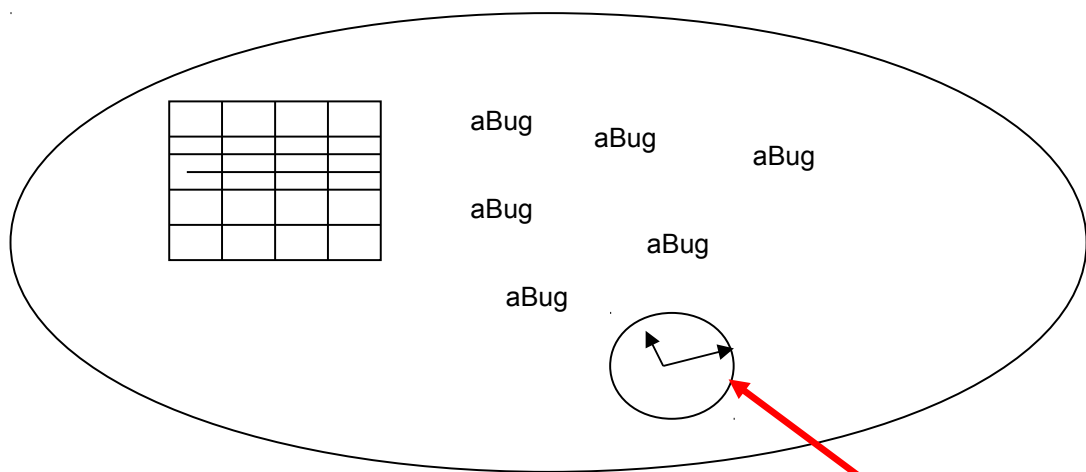
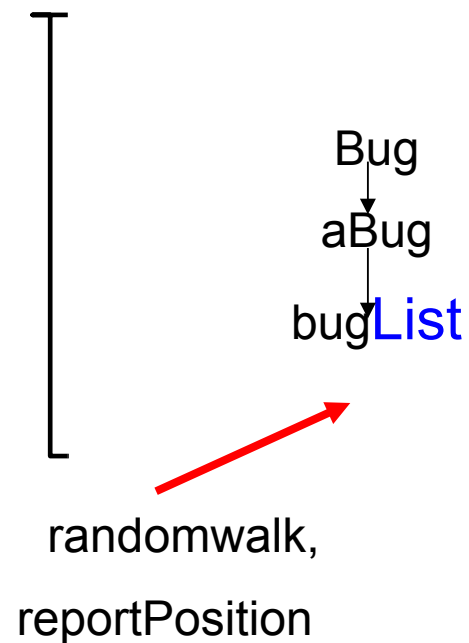
# Swarm = a library of functions and a **protocol**

modelSwarm

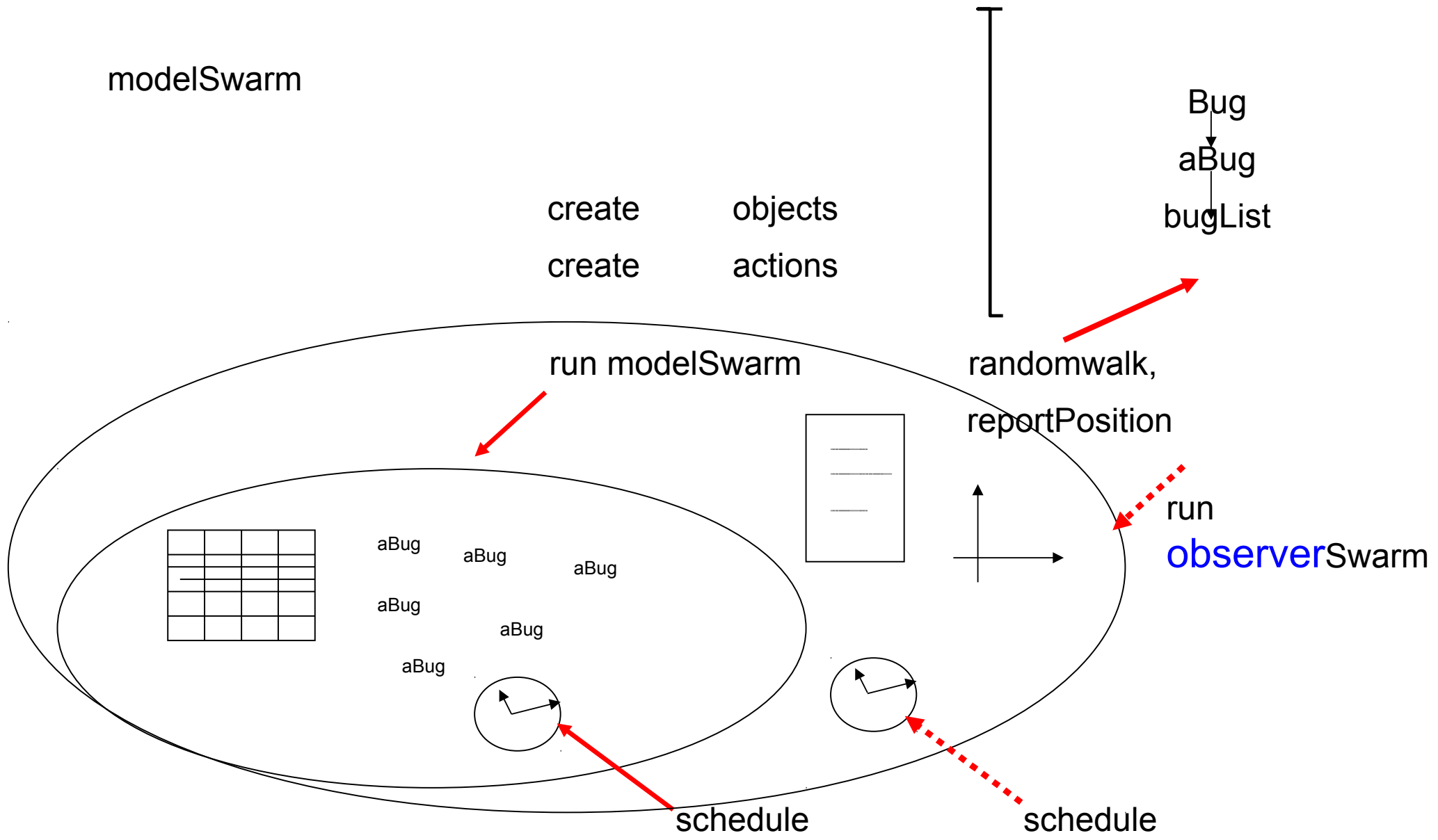
create agents

create actions

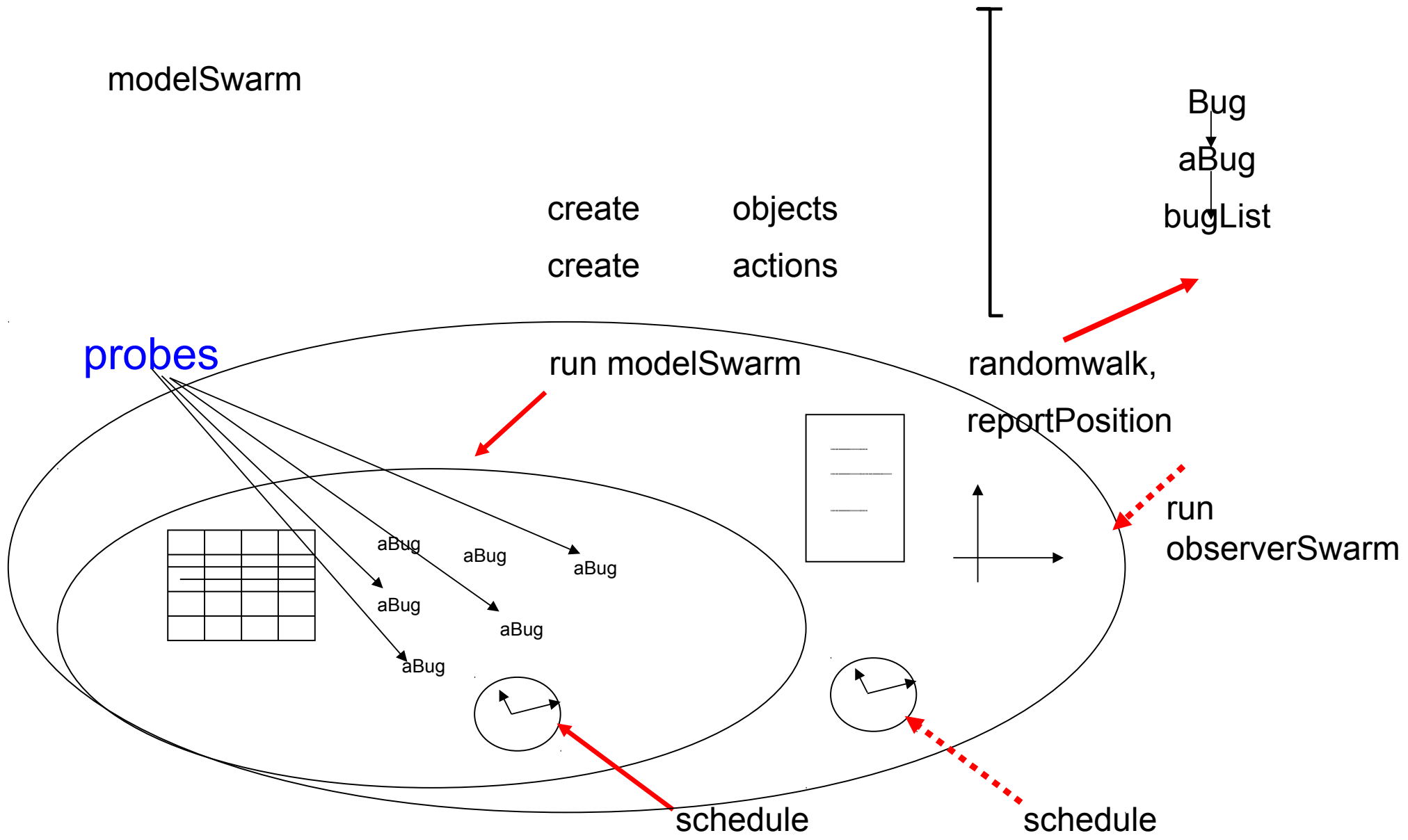
run modelSwarm



# Swarm = a library of functions and a **protocol**



# Swarm = a library of functions and a **protocol**



---

**The content:**

the ERA scheme (Environment, Agents and Rules)

---

# ERA

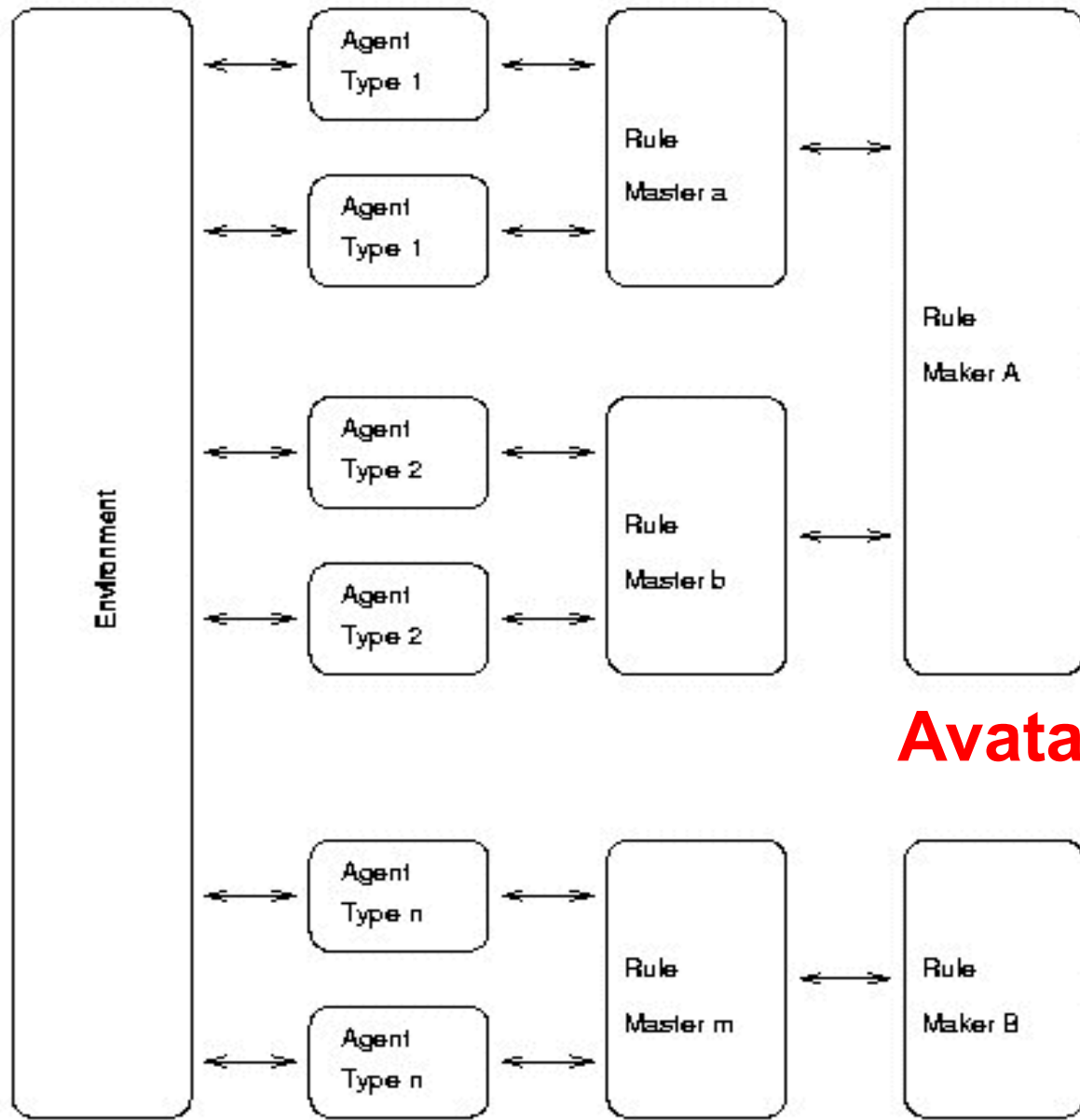
N.Gilbert, P.Terna (2000), How to build and use agent-based models in social science, *Mind & Society*, no. 1, pp.57-72

or

N.Gilbert, P.Terna (2010), How to build and use agent-based models in social science. In Nigel Gilbert (ed.) *Computational Social Science*, Sage Publications London, Vol. IV, pp.229-246

the scheme at

<http://web.econ.unito.it/terna/ct-era/ct-era.html>



**Fixed rules**

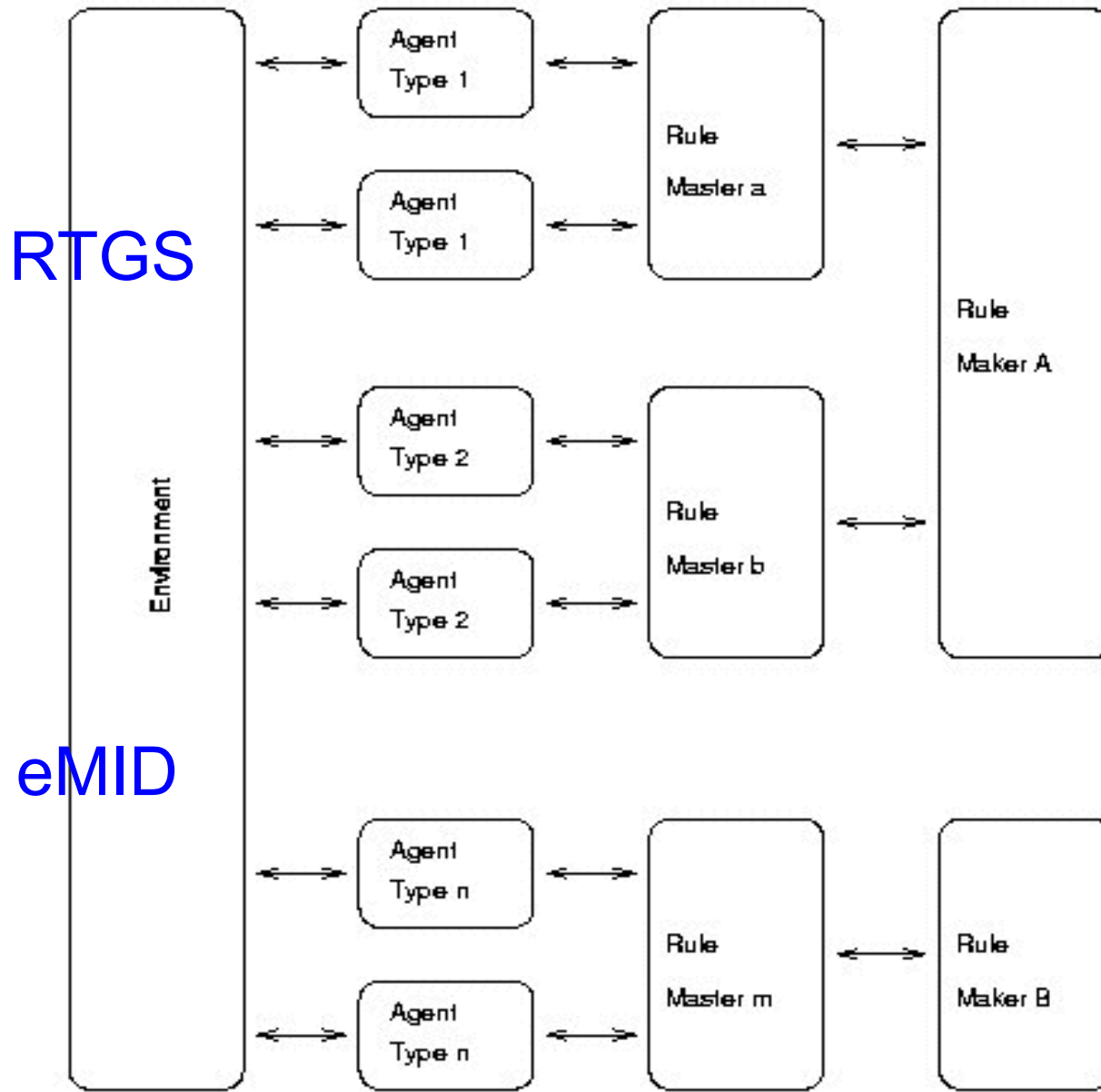
**NN**

**CS**

**GA**

**Reinforcement learning**

**Avatar**



---

The search for an equilibrium between the “simple” equation based description of the of the banks and the “complicated” definition of the micro-details in their actions

---

the search for an equilibrium between:

the “*simple*” *equation based description* of the behavior of the banks’ treasurers

and the complexity arising from a “*complicated*” *definition of the micro-details in their actions*

## *From abstraction to realism*

(i)

The standard economic approach tries to compress the complexity of the reality into the mathematical structure of models based on ex ante hypotheses arising from the standard paradigms, as the rationality in behavior, so ignoring interaction and sociality among the real world agents.

The use of those models, for individual and collective choices, determines the **ineffectiveness of corrective actions in policies.**

(ii)

The first step in a pro-social direction is to bring the equations of the models

at the level of individual agents,  
possibly with some degree of heterogeneity,  
closer both to realism and social dimension;

especially in the presence of heterogeneity, **it is a step ahead the pure numerical simulation, but still too tied to the standard paradigm.**

(iii)

If the stylized agents, with their equations, operate (calculate) in a system with **institutions** (for example, a trading system), the model is taking another step forward realism and usefulness.

(iv)

Finally, it is only with a full implementation of the ABM paradigm that we can respond to the needs expressed by **policy maker calling**, as Trichet (2010 [<http://www.ecb.int/press/key/date/2010/html/sp101118.en.html>]), **for a radical change in the models.**

## *A whole line of work*

A huge work awaits us: the construction of models that can compare in a parallel way

(i) the standard setting of the purely mathematical model, with equations that reproduce the behavior in a decentralized way,

in case (ii) also with heterogeneity, and

in case (iii) through interaction with institutions and rules, until

the use (iv) of models completely based on agents, both with simple behavioral skills or with arbitrarily complicated way of behaving.

Those skills are related to the structures of relationships, mutual influence, information, ...

---

## A technical appendix

---

The current package for "Agent based simulation model of RTGS system and money market", or Payments and Money market model (in short: **P&3M**), is a special implementation of **SLAPP**.

In perspective, it will evolve to a (special) **AESOP** application.

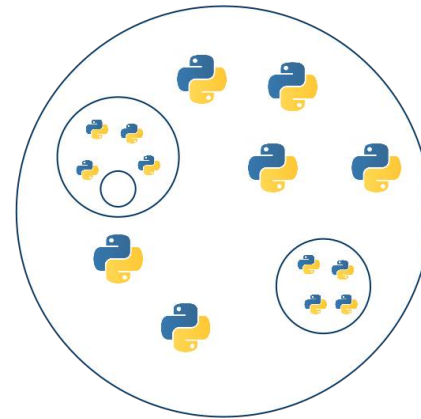
SLAPP, [Swarm-Like Agent Protocol in Python](#), as a simplified implementation of the original Swarm protocol, choosing Python as a simultaneously simple and complete object-oriented framework.

The SLAPP project has also the goal of offering to scholars interested in agent-based models a set of programming examples that could be easily understood in all their details and adapted to other applications.



AESOP, [Agents and Emergencies for Simulating Organizations in Python](#)), written upon SLAPP and intended to be a simplified way to describe and generate interaction within artificial agents.

- bland agents (simple, unspecific, basic, insipid, ...) used to populate our simulated world with agents doing basic actions;
- tasty agents (specialized, with given skills, acting in a discretionary way, ...), used to specify important roles into the simulation scenario.



A huge **surprise**: thanks to Steve Rogers, we will have a parallel SLAPP implementation, running within **OMQ**,

<http://zeromq.org/>

[http://www.youtube.com/watch?v=\\_JCBphyciAs](http://www.youtube.com/watch?v=_JCBphyciAs)

So ... **OSLAPP** is born  
(see <http://code.google.com/p/slapp-tools/> ).

---

A quick look to **SLAPP**  
Swarm-Like Agent Based Protocol in Python

---

# Why a new tool?

- For didactical reasons, applying a such rigorous and simple object oriented language as Python is
- To build models upon transparent code: Python does not have hidden parts or features coming from *magic*, it has no obscure libraries
- **To use the openness of Python**
- **To have the possibility of using the key feature of the Swarm protocol in an easy way**

About Python and ABMs have also a look to  
the 2011 Alan Isaac paper at

<http://jasss.soc.surrey.ac.uk/14/2/5.html>



---

[Alan G. Isaac](#) (2011)

**The ABM Template Models: A Reformulation with Reference Implementations**

*Journal of Artificial Societies and Social Simulation* **14** (2) 5  
<<http://jasss.soc.surrey.ac.uk/14/2/5.html>>















- ... going from Python to R  
(R is at <http://www.r-project.org/> ;  
rpy library is at <http://rpy.sourceforge.net/>)
- ... going from OpenOffice (Calc, Writer, ...) to Python and viceversa (via  
the Python-UNO bridge, incorporated in OOo and LibreOffice)
- ... doing symbolic calculations in Python (via  
<http://code.google.com/p/sympy/> )
- ... doing declarative programming with PyLog, a Prolog implementation  
in Python ( <http://www.cdsoft.fr/pylog/index.html> )
- ... using Social Network Analysis from Python; examples:
- Igraph <http://cneurocv.s.rmki.kfki.hu/igraph/>
- NetworkX <http://networkx.lanl.gov/>

---

The SLAPP package at  
<http://eco83.econ.unito.it/terna/slapp>

---

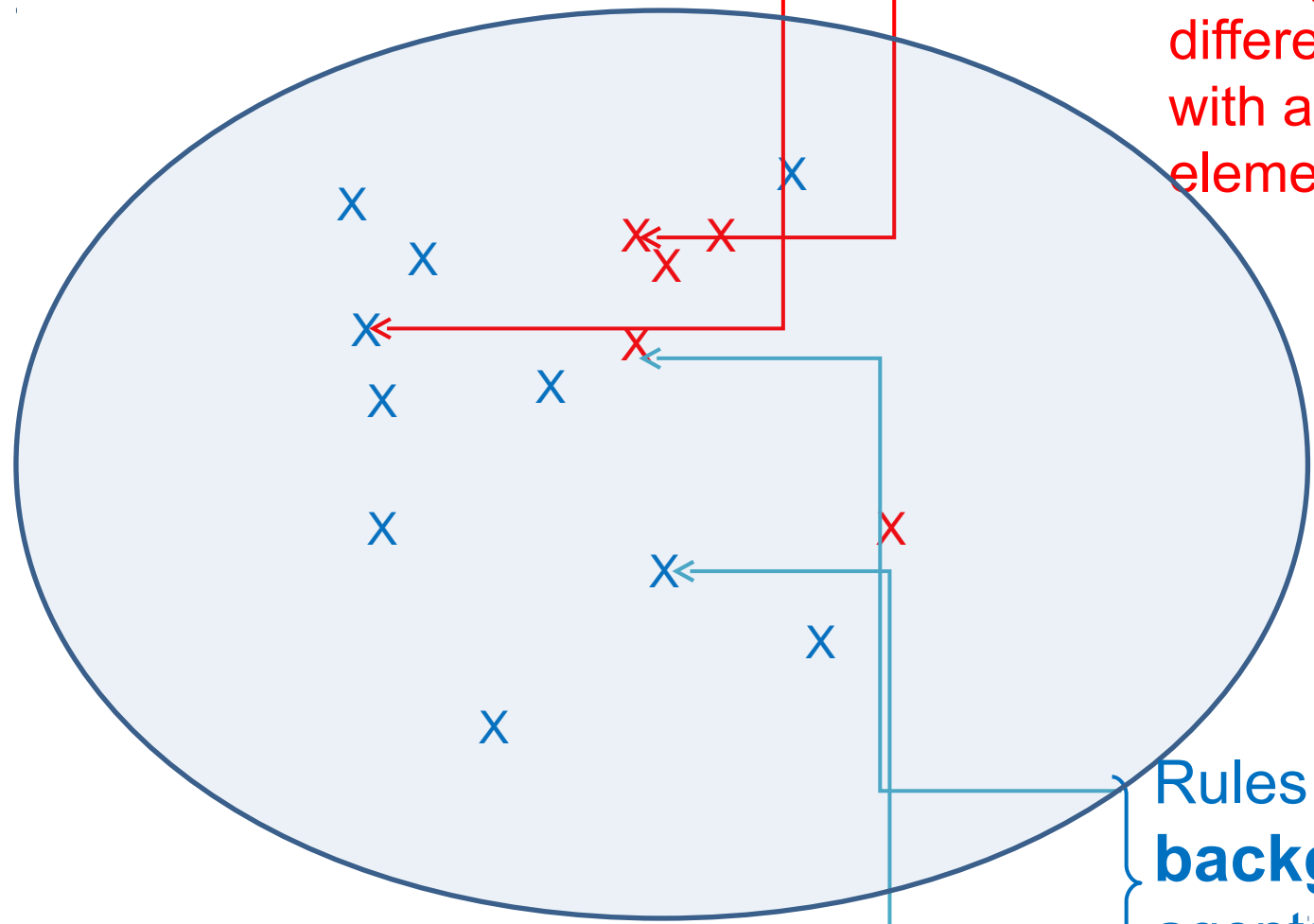
- ▶ 1 plainProgrammingBug
- ▶ 2 basicObjectProgrammingBug
- ▶ 3 basicObjectProgrammingManyBugs
- ▶ 4 basicObjectProgram...s\_bugExternal+\_shuffle
- ▶ 5 objectSwarmModelBugs
- ▶ 6 objectSwarmObserverAgents\_AESOP\_turtleLib ←
- ▶ 7 toBeDeveloped objectSwarmObserverTkBugs
- ▶ 7b toBeDeveloped Tk test
- ▶ 8 toBeDeveloped simpleExpertBug
- ▶ readme.txt
- ▶ SLAPP 0 tutorial.txt
- ▶ Swarm\_original 0 tutorial.txt

-  a\_note\_on\_AESOP.txt
-  ActionGroup.py
- ▶  basic
-  convert\_xls\_txt.py
-  ModelSwarm.py
-  ObserverSwarm.py
-  Pen.py
- ▶  school
-  SLAPP 6 objectSwarmObserverAgents.txt
-  start 6 objectSwarmObserverAgentsAESOP.py
-  Swarm\_original 7 simpleObserverBug.txt
-  Swarm\_original 8 simpleObserverBug2.txt
-  Tools.py
-  WARNING.txt

---

## Agents and schedule

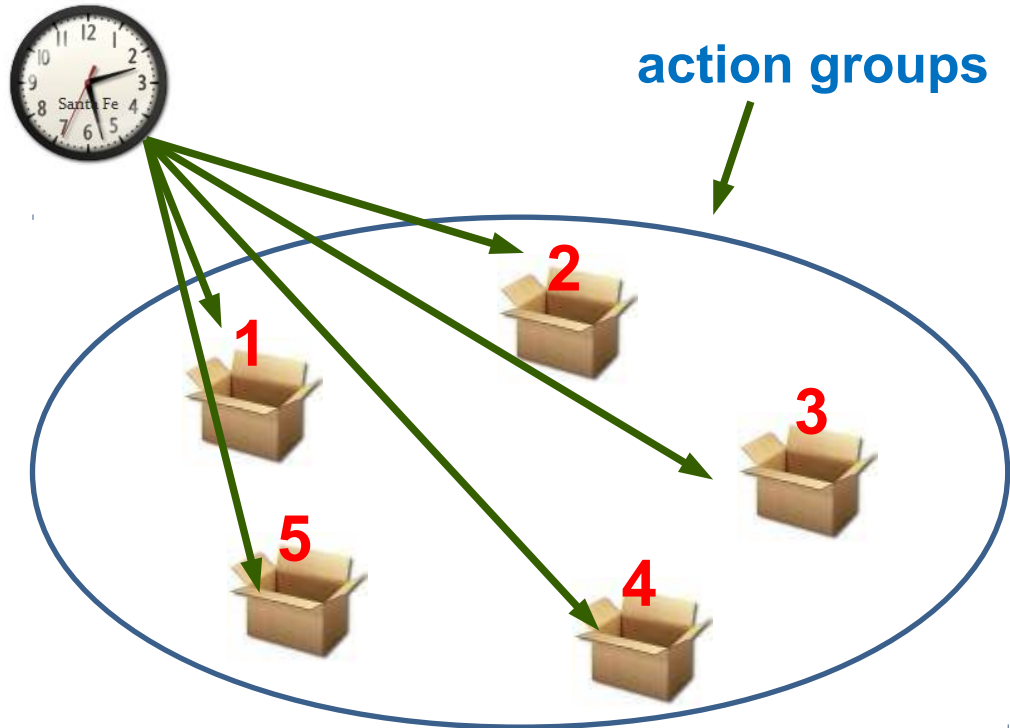
---



Rules operating “in the **foreground**”, explicitly managed via a script (with different sets of agents, with a different number of elements)

Rules operating “in the **background**” for all the agents, or only for the blue ones or for a specific set of tasty agents

*\*Bland = simple, unspecific, basic, insipid, ...*  
*#Tasty = specialized, with given skills, discretionary, ...*



**What in each box?**

**Tasks to be executed (with  $p=1$  or with  $p<1$ )**

**Tasks are included into the code in a static way, or can be added/activated dynamically by other tasks, also via agents' actions**

**Tasks can be read – via a 'read' task schedule element – from an external source (file, web interaction, ...)**

**A special type of task to be read from an external source is that of the **recipes****

## tasks read from an external archive

a\_n – a specific agent (instance of class A)

a\_X – a randomly chosen agent (instance of class A)

a\_%all – a quota of all the agents (instances) of the class A

a\_all – all the agents (instances) of the class A



[agent method]



methods specific of each agent or inherited from the basic types 'Agent' or 'Turtle'

# recipes, as macro, read from an external archive

[ [agent method] [agent method] [agent method] [agent method] [agent method] ... ]

→ to be executed in a sequential way by a given thread of agents (statically or dynamically)

[ [agent method] **[ [agent method] [agent method] [agent method] ]** [agent method] ... ]

with segments to be executed in a parallel way

[ [agent method] [agent method] **N**[agent method] [agent method] [agent method] ... ]

[ ..... : [agent method] ..... ]  
 [ ..... : ..... ]  
 [ ..... : [agent method] ..... ]

with components belonging to different recipes to be executed as a whole

---

An example, studying pupils behavior  
in primary school classes (with Sandro  
Brignone, Aldo Milano and Tiziana Pasta)

---

schedule.xls - LibreOffice Calc

Calibri 11

E86 = bisognerebbe evidenziare il fatto che i b. eseguono la consegna con velocità diverse

	A	B	C	D
1				th
2	macro	repeatedAction		VI
3	#		1	
4	macro	transitionPhase		
5	#		2	
6	macro	checkToObtainAttention		
7	#		3	
8	scPupil	askWell		IN
9	scPupil	talkTeacherWell		
10	macro	individualFocus		
11	#		4	
12	macro	assignmentEasy		
13	zCen		0,2 checkFastWork	
14	scPupil	checkWork		
15	#		5	
16	macro	revision		
17	macro	easyQuestion		
18	all		0,095 bePraised	do
19	all		0,7 fidget	au
20	all		0,286 doWork	Di

schedule transitionPhase checkToObtainAttention individualFocus assignmentEasy revision easyQuestion

Foglio 1 / 8 PageStyle\_schedule STD Somma=0 115%

schedule.xls - LibreOffice Calc

Calibri 11

E86 = bisognerebbe evidenziare il fatto che i b. eseguono la consegna con velocità diverse

	A	B	C	D
21	f3dx		0,6 checkTeacherTalkClose	
22	rossoPupil		0,16 tidy	
23	verdePupil		0,1 tidy	
24	#		6	
25	macro	revision		
26	macro	easyQuestion		
27	all		0,5 fidget	at
28	verdePupil		-1 doWork	D
29	verdePupil		0,1 tidy	tic
30	#		7	D
31	macro	revision		
32	verdePupil		0,1 attractTeacherAttentionWell	fu
33	verdePupil		0,4 beQuiteBored	fu
34	scPupil	talkTeacherBad		
35	scPupil	beScolded		
36	#		8	
37	macro	revision		
38	macro	easyQuestion		
39	rossoPupil		0,16 attractTeacherAttentionNotWell	
40	saPupil	tease		fu

Foglio 1 / 8 PageStyle\_schedule STD Somma=0 115%

schedule.xls - LibreOffice Calc

Calibri 11

E15

	A	B	C	D	E	F	G	H
1	<u>gialloPupil</u>	sitDownNotWell						
2	<u>verdePupil</u>	-1 sitDownNotWell						
3	all	0,3 fidget						
4	<u>saPupil</u>	0,8 shake						
5	<u>verdePupil</u>	-1 shake						
6	<u>treV</u>	talkClose						
7	<u>f4dx</u>	talkClose						
8	<u>saPupil</u>	0,5 answerBad						
9	all	0,15 answerWell						
10	<u>verdePupil</u>	0,1 tidy			<u>questi possono anche essere messi fuori dal macrom</u>			
11	<u>verdePupil</u>	0,1 untidyTidy			"			
12	<u>verdePupil</u>	0,1 wellness			"			
13	<u>rossoPupil</u>	0,16 wellness			"			
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								

schedule transitionPhase checkToObtainAttention individualFocus assignmentEasy revision easyQuestion

Foglio 3 / 8 PageStyle\_checkToObtainAttention STD Somma=0 100%

---

# Thanks

Pietro Terna  
[terna@econ.unito.it](mailto:terna@econ.unito.it)  
[web.econ.unito.it/terna](http://web.econ.unito.it/terna)

---