



# **Μοντελοποίηση Agent Based**

---

# Agent-based modeling

- Η μοντελοποίηση βασισμένη σε πράκτορες είναι μια σχετικά νέα μέθοδος σε σύγκριση με τα δυναμικά συστήματα και τη μοντελοποίηση διακριτών γεγονότων.
  - Στην πραγματικότητα, η μοντελοποίηση βασισμένη σε πράκτορες ήταν σε μεγάλο βαθμό ένα ακαδημαϊκό θέμα έως ότου οι επαγγελματίες προσομοίωσης άρχισαν να το χρησιμοποιούν πριν από περίπου 15 χρόνια.
-

# Agent-based modeling

- Προκλήθηκε από:
  - Την επιθυμία να αποκτηθούν βαθύτερες γνώσεις στα συστήματα που οι παραδοσιακές προσεγγίσεις μοντελοποίησης δεν τα αποτυπώνουν καλά
  - Πρόοδοι στην τεχνολογία μοντελοποίησης που κατέστη δυνατή από την επιστήμη των υπολογιστών, όπως η αντικειμενοστρεφής μοντελοποίηση, το UML και τα statecharts
  - Η ταχεία ανάπτυξη της ισχύος και της μνήμης της CPU. Τα μοντέλα που βασίζονται σε πράκτορες είναι πιο απαιτητικά από τα δυναμικά συστήματα και τα μοντέλα διακριτών συμβάντων.
-



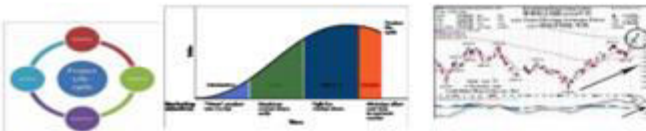
# Agent-based modeling

- Υπάρχουν πολλοί τρόποι προσδιορισμού της συμπεριφοράς ενός πράκτορα. Συχνά ο πράκτορας έχει μια έννοια κράτους και οι πράξεις και οι αντιδράσεις του εξαρτώνται από το κράτος. τότε η συμπεριφορά καθορίζεται καλύτερα με statecharts. Μερικές φορές η συμπεριφορά ορίζεται σε κανόνες που εκτελούνται κατά τη διάρκεια ειδικών γεγονότων.

**People in different roles:**  
consumers, citizens, employees,  
patients, doctors, clients, soldiers, ...



**Non-material things:**  
projects, products, innovations,  
ideas, investments ...



**Equipment, vehicles:**  
trucks, cars, cranes, aircrafts,  
rail cars, machines, ...



**Organizations:**  
companies, political parties, countries, ...



# Δημιουργία Μοντέλου Agent Based

- Θα ξεκινήσουμε δημιουργώντας ένα απλό μοντέλο που απεικονίζει τον τρόπο με τον οποίο η διαφήμιση οδηγεί τους καταναλωτές να αγοράσουν ένα προϊόν.
-



- Θα δημιουργήσουμε ένα μοντέλο που θα βασίζεται σε πράκτορες μιας καταναλωτικής αγοράς
  - Θα μας βοηθήσει να κατανοήσουμε πώς ένα προϊόν εισέρχεται στην αγορά.
  - οι αποφάσεις για τον άνθρωπο περιλαμβάνουν πάντα στοχαστικά μοντέλα,
  - η μοντελοποίηση βασισμένη σε πράκτορες είναι ιδανική για τη μοντελοποίηση των προσομοιώσεων της αγοράς.
-

# Πληθυσμός πρακτόρων

- Το πρώτο πράγμα που πρέπει να αποφασίσουμε είναι ο πληθυσμός των πρακτόρων που θα απευθυνθούμε. Το Target Group που στοχεύουμε.
  - Το μοντέλο περιλαμβάνει 5000 άτομα που δεν χρησιμοποιούν το προϊόν, αλλά ένας συνδυασμός διαφήμισης και από στόμα σε στόμα θα τους οδηγήσει τελικά να το αγοράσουν
-

# Συνθήκες του Μοντέλου

- Οι καταναλωτές του μοντέλου μας δεν θα χρησιμοποιήσουν το προϊόν στην αρχή.
  - Αλλά όλοι δυνητικά ενδιαφέρονται να το χρησιμοποιήσουν.
  - Επίσης, θα μελετήσουμε την επιρροή της διαφήμισης στη ζήτηση των καταναλωτών επιτρέποντας σε ένα συγκεκριμένο ποσοστό να ενδιαφέρονται για την αγορά του προϊόντος κατά τη διάρκεια μιας συγκεκριμένης ημέρας.
  - Για τους σκοπούς μας, η αποτελεσματικότητα διαφήμισης = 0,1 καθορίζει το ποσοστό των πιθανών χρηστών που είναι έτοιμοι να αγοράσουν το προϊόν κατά τη διάρκεια μιας δεδομένης ημέρας.
-

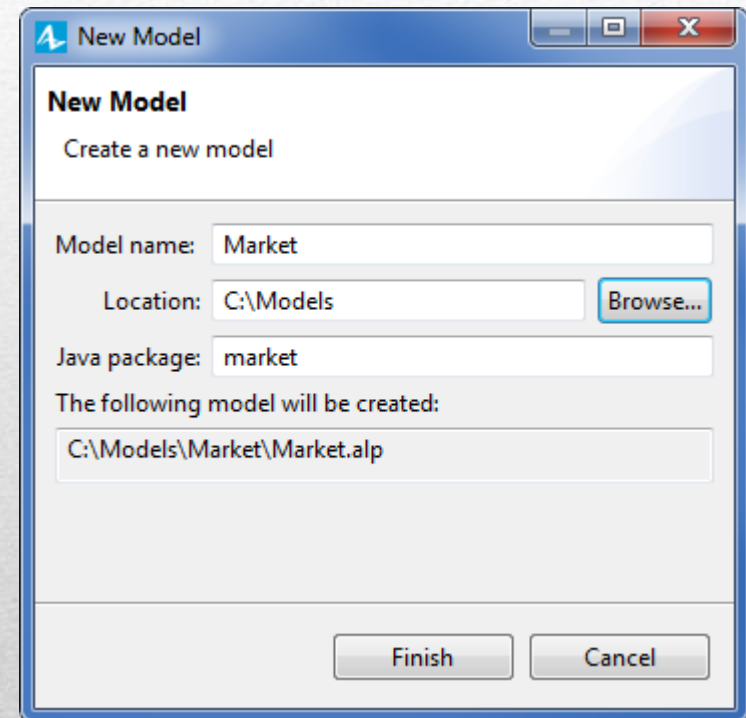


# Δημιουργία Μοντέλου Agent Based

κλείστε τη σελίδα υποδοχής και δημιουργήστε ένα νέο μοντέλο επιλέγοντας File > New > Model από το κύριο μενού της AnyLogic. Θα ανοίξει ο οδηγός του νέου μοντέλου.



- Στο Model name εισάγουμε το όνομα του μοντέλου Market
- Στο Location επιλέξτε το φάκελο στον οποίο θέλετε να δημιουργήσετε το μοντέλο. Μπορείτε να περιηγηθείτε σε ένα φάκελο κάνοντας κλικ στο κουμπί Browse ή πληκτρολογήστε το όνομα του φακέλου που θέλετε να δημιουργήσετε στο πλαίσιο Location.
- Και τέλος πατάμε το κουμπί Finish

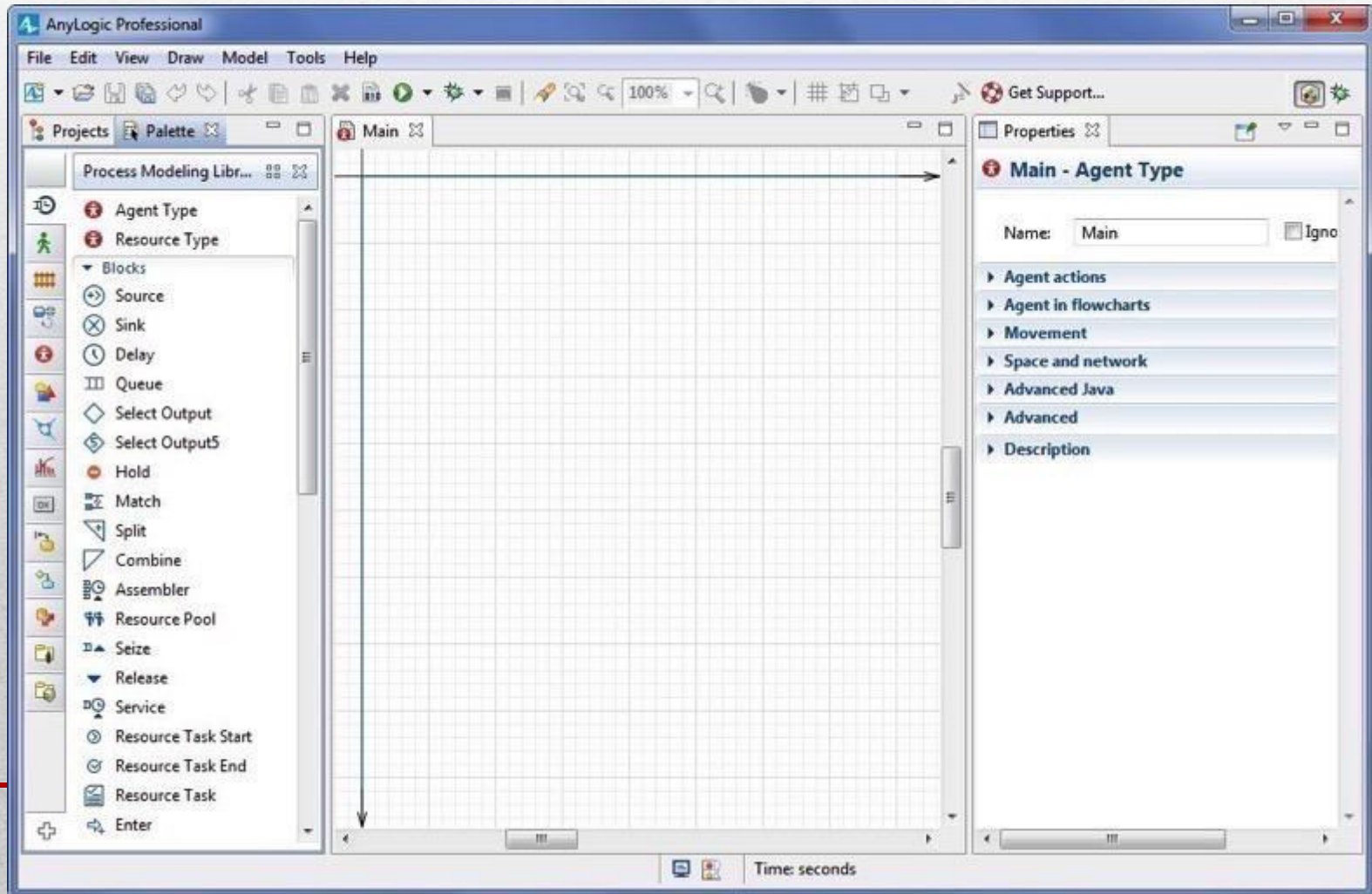


# Περιβάλλον Anylogic

- Τώρα, ας εξετάσουμε σύντομα το περιβάλλον του AnyLogic.
-



# Project Palette, Graphical editor, Properties View



# Graphical Editor

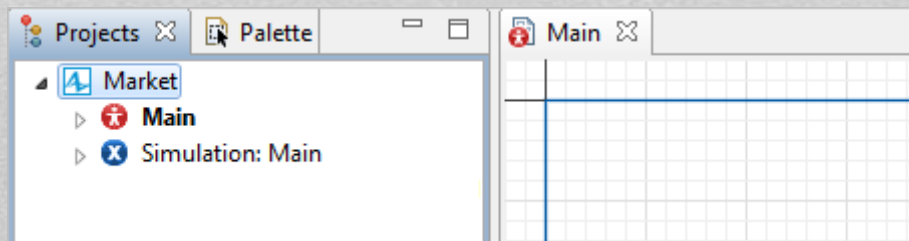
- Ο γραφικός επεξεργαστής σας επιτρέπει να επεξεργαστείτε το διάγραμμα του τύπου πράκτορα και
  - μπορείτε να προσθέσετε στοιχεία μοντέλου μεταφέροντάς τα από την Palette (παλέτα) στο διάγραμμα και τοποθετώντάς τα στον πίνακα του επεξεργαστή.
  - Τα στοιχεία που τοποθετείτε μέσα στο μπλε πλαίσιο θα εμφανιστούν στο παράθυρο του μοντέλου όταν το εκτελείτε.
-

# Project - Palette - Properties

- Το Projects view σας επιτρέπει να έχετε πρόσβαση στα μοντέλα AnyLogic που έχετε ανοίξει στον χώρο εργασίας και το δέντρο χώρου εργασίας σας βοηθά να πλοηγηθείτε εύκολα σε αυτά.
  - Το Palette view (προβολή παλέτας) απαριθμεί τα αντικείμενα που ομαδοποιούνται σε παλέτες. Για να προσθέσετε ένα στοιχείο στο μοντέλο σας, σύρετε το στοιχείο από την παλέτα στο γραφικό πρόγραμμα επεξεργασίας.
  - Η Properties view σας επιτρέπει να προβάλετε και να τροποποιήσετε τις ιδιότητες του επιλεγμένου στοιχείου.
-



- Στην αριστερή ενότητα του χώρου εργασίας και μπορείτε να μεταβείτε από την προβολή Παλέτα στην προβολή Έργα κάνοντας κλικ στην καρτέλα Έργα.
- Ανοίγοντας την προβολή Projects μπορούμε να εξετάσουμε τη δομή του μοντέλου.



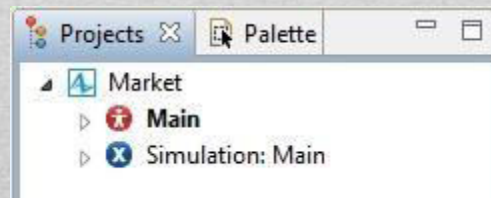
- Η προβολή "Έργα" σας επιτρέπει να έχετε πρόσβαση στα έργα AnyLogic που έχετε ανοίξει στον χώρο εργασίας και μπορείτε να χρησιμοποιήσετε το δέντρο του χώρου εργασίας για γρήγορη και εύκολη πλοήγησή τους.
  - □ Η AnyLogic χρησιμοποιεί μια δομή δέντρου για την εμφάνιση του μοντέλου.
  - Στο κορυφαίο επίπεδο εμφανίζει το μοντέλο, το παρακάτω επίπεδο εμφανίζει τους τύπους πράκτορα και τα πειράματα και στα χαμηλότερου επιπέδου τα στοιχεία που αποτελούν τη δομή του παράγοντα.
-

- Οι πράκτορες είναι τα δομικά στοιχεία ενός μοντέλου και μπορείτε να τα χρησιμοποιήσετε για να μοντελοποιήσετε όλα τα είδη αντικειμένων πραγματικού κόσμου, συμπεριλαμβανομένων οργανισμών, εταιρειών, φορτηγών, σταθμών επεξεργασίας, πόρων, πόλεων, λιανοπωλητών, φυσικών αντικειμένων, ελεγκτών κ.ο.κ.
  - Κάθε πράκτορας τυπικά αντιπροσωπεύει ένα από τα λογικά τμήματα του μοντέλου.
  - Αυτό σας επιτρέπει να αποσυνθέσετε ένα μοντέλο σε πολλά επίπεδα λεπτομέρειας.
-

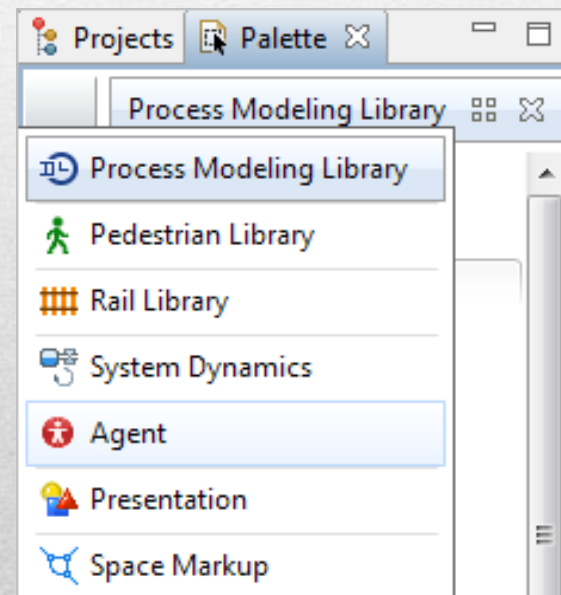
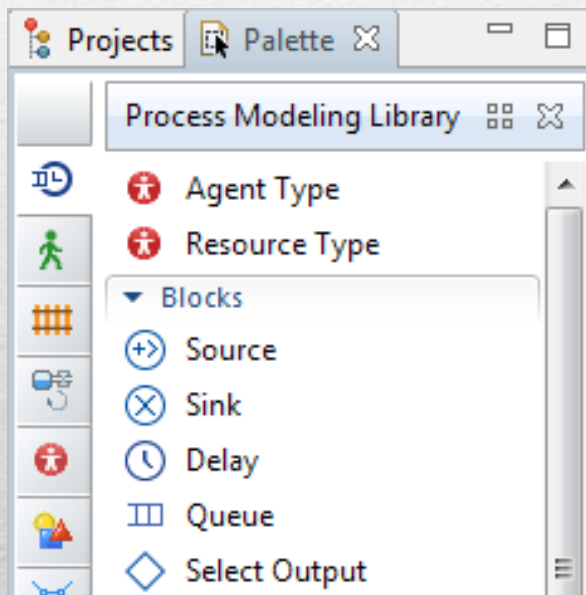


# Πράκτορες

- Θέλουμε να προσθέσουμε ένα νέο στοιχείο μοντέλου, αλλά πρέπει πρώτα να μεταβείτε στην προβολή Palette κάνοντας κλικ στην καρτέλα Παλέτα

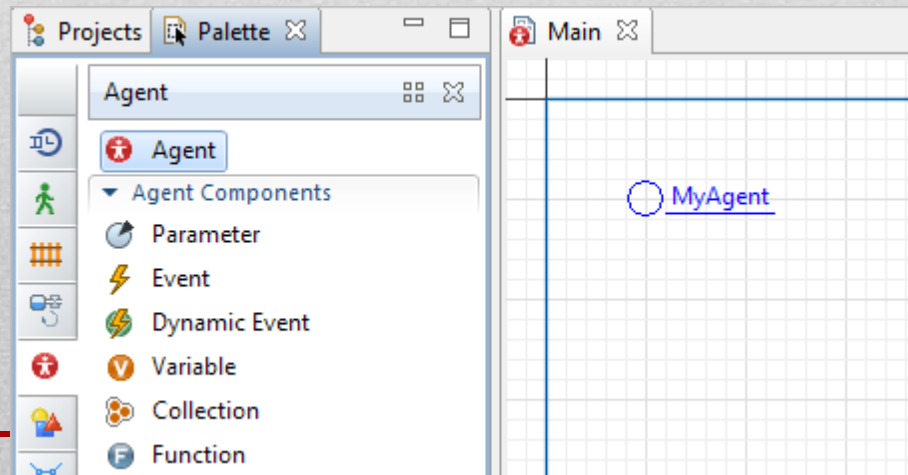


- Άνοιγμα της παλέτας Agent. Για να ανοίξετε μια συγκεκριμένη παλέτα, μεταβείτε στην προβολή Παλέτα και τοποθετήστε το δείκτη του ποντικιού πάνω από τον κάθετο πίνακα πλοήγησης της οθόνης.
  - Θα επεκταθεί για να εμφανίζονται τα ονόματα όλων των παλετών ώστε να μπορείτε να επιλέξετε αυτό που χρειάζεστε. Κάντε κλικ στην παλέτα Agent στη λίστα για να την επιλέξετε.
-






- Σύρετε το Agent από την παλέτα Agent στο κύριο διάγραμμα και θα ανοίξει ο οδηγός New agent



- Στο βήμα 1. Επιλέξτε τη σελίδα που θέλετε να δημιουργήσετε, επιλέξτε την επιλογή που ανταποκρίνεται καλύτερα στις ανάγκες σας.
  - Εφόσον θέλουμε να δημιουργήσουμε πολλαπλούς πράκτορες του ίδιου τύπου, επιλέξτε Πληθυσμός παραγόντων και κάντε κλικ στο κουμπί Επόμενο.
-

New agent

**Step 1. Choose what you want to create**




**Population of agents**

Create a number of agents of the same type living in the same environment in the current agent.

Typical cases:

- People
- Consumers
- Patients
- Trucks
- Projects or products




**A single agent**

Create a single agent that will always exist within the current agent.

Typical cases:

- Supplier, distributor, producer
- Building
- Factory
- Store
- Gas station
- Hospital
- Equipment unit



**Agent type only**

Create an agent type, do not create any agents at this point.

Typical cases:

- Entity type: Patient, Customer, Document, Part, Transaction
- Resource type: Doctor, Worker, ForkliftTruck
- Train or rail car type
- Pedestrian type
- Structural part of the model, such as a subprocess

< Back   Next >   Finish   **Cancel**



- Στο βήμα 2. Δημιουργία νέας σελίδας τύπου πράκτορα, στην ενότητα Το όνομα νέου πλαισίου τύπου, πληκτρολογήστε Consumer. Οι πληροφορίες στο πλαίσιο Όνομα πληθυσμού παραγόντων θα αλλάξουν αυτόματα στους consumers.
-

New agent

### Step 2. Creating new agent type

Do not use template  
I am creating the agent type "from scratch"

Use existing template  
I want to choose one of the predefined agent templates as the basis for my agent type

Person  
Vehicle

The name of new type:

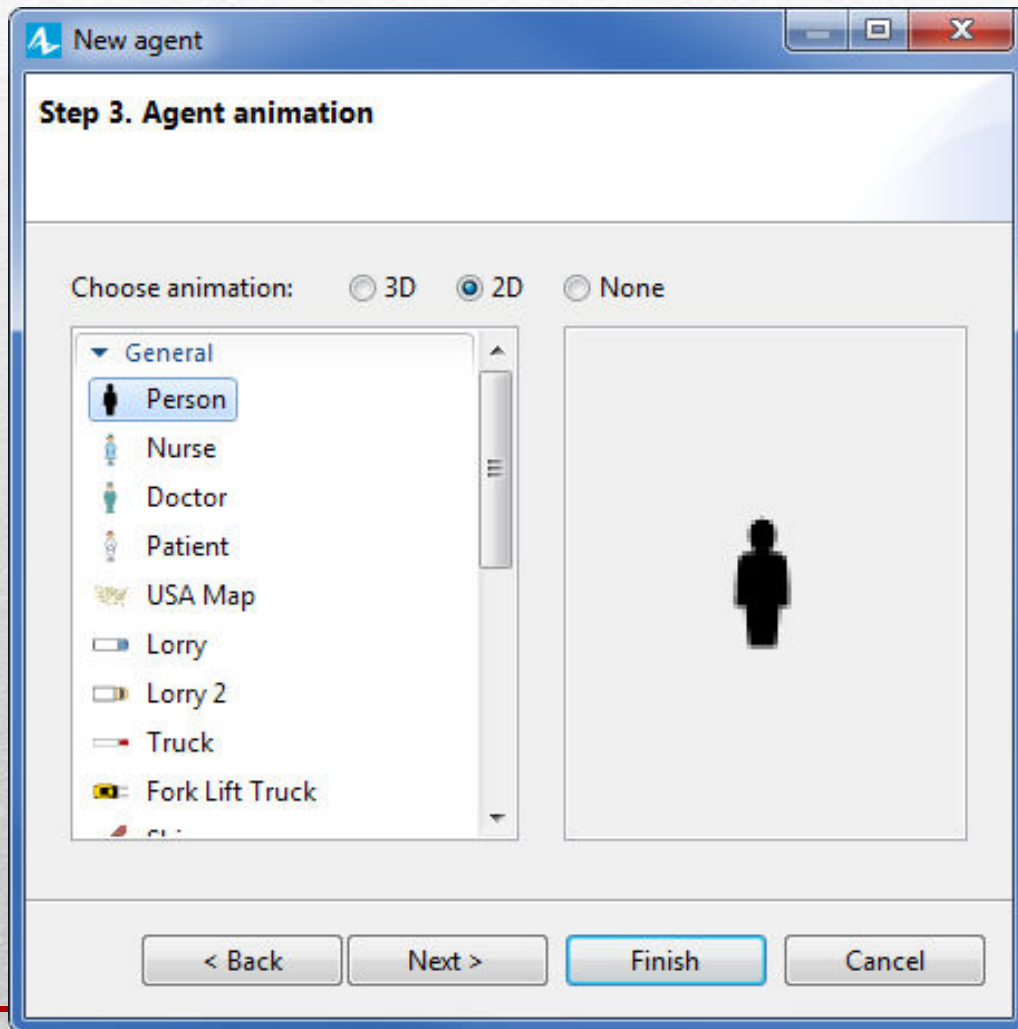
Agent population name:

Agent will be used in flowcharts

< Back   Next >   Finish   Cancel

- Στη σελίδα Agent animation, επιλέξτε το σχήμα κινούμενης επιφάνειας του agent.
  - Αφού δημιουργούμε ένα απλό μοντέλο που χρησιμοποιεί 2D animation, επιλέξτε 2D, επιλέξτε το πρώτο στοιχείο της Γενικής λίστας: Person
  - και κάντε κλικ στο Next
-





- Στη σελίδα Agent Parameter, ορίστε τις παραμέτρους του παράγοντα ή τα Χαρακτηριστικά του.
  - Δεδομένου ότι το μοντέλο μας λαμβάνει υπόψη μόνο τις αγορές προϊόντων που σχετίζονται με τη διαφήμιση, θα προσθέσουμε μια παράμετρο - AdEffectiveness –
  - Με αυτή την παράμετρο μπορούμε να καθορίσουμε το ποσοστό των πιθανών χρηστών που είναι έτοιμοι να αγοράσουν το προϊόν κατά τη διάρκεια μιας δεδομένης ημέρας.
-

- Στην αριστερή ενότητα, στον πίνακα Parameters, κάντε κλικ στο <add new ...> για να δημιουργήσετε μια παράμετρο.
  - Στο πλαίσιο Παράμετροι, αλλάξτε το όνομα της προεπιλεγμένης παραμέτρου σε AdEffectiveness και επιλέξτε double ως τύπος παραμέτρου.
  - Υποθέτουμε ότι κατά μέσο όρο 1% των πιθανών χρηστών του μοντέλου μας θα θέλουν να αγοράσουν το προϊόν κατά τη διάρκεια μιας δεδομένης ημέρας, οπότε καθορίστε το 0,01 ως την τιμή της παραμέτρου.
-



New agent

### Step 5. Agent parameters

Please fix the parameters you want to see in your MyAgent:


Parameters

- AdEffectiveness
- <add new...>

Parameter:

Type:





Specify value or stochastic expression



Follow empirical distribution

Percentage distribution of the population:

Interval start	Interval end	Number of observa...

- Στη σελίδα Μέγεθος πληθυσμού, πληκτρολογήστε 5000 στο πλαίσιο Δημιουργία πληθυσμού με ... πράκτορες για να δημιουργήσετε 5000 παρουσίες του τύπου Καταναλωτή.
  - Κάθε περίπτωση του πληθυσμού θα μοντελοποιήσει έναν συγκεκριμένο παράγοντα-καταναλωτή.
  - Παρόλο που έχουμε δημιουργήσει τον πληθυσμό του πράκτορα μας, δεν θα βλέπουμε αριθμούς κινουμένων σχεδίων 5.000 ατόμων στο κύριο διάγραμμα.
  - Αντ 'αυτού, η AnyLogic θα χρησιμοποιήσει τους 5000 πράκτορες στον πληθυσμό που ονομάσαμε καταναλωτές για να προσομοιώσουμε την αγορά όταν τρέχουμε το μοντέλο μας.
-

**New agent**

### Step 5. Population size

Create population with  agents

This is the initial population size.  
You will be able to add more agents or delete any agent at runtime.

Create initially empty population, I will add agents at the model runtime

< Back   Next >   **Finish**   Cancel



- Στη σελίδα “Configure new environment”, αποδεχτείτε τις προεπιλεγμένες τιμές για τον τύπο χώρου περιβάλλοντος (συνεχής) και τις δύο τιμές του πλάτους και του ύψους (500). Το AnyLogic θα εμφανίσει τους πράκτορες σε ορθογώνιο 500x500 εικονοστοιχείων.
  - Επιλέξτε το πλαίσιο Apply random layout για την τυχαία διανομή των παραγόντων σε πλάτος και ύψος 500 pixel που έχουμε ορίσει.
  - Δεδομένου ότι δεν θέλουμε να δημιουργήσουμε ένα δίκτυο αντιπροσώπων, θα αποδεχτούμε τον προεπιλεγμένο τύπο δικτύου που δεν έχει καθοριστεί από το δίκτυο / χρήστη.
-

New agent

### Step 6. Configure new environment

This agent will live in the 'Main' agent type.

The following are the environment settings.

You can always change them from the properties of Main agent type (see Environment for other agents section)

Space type:  Continuous  GIS  Discrete

Size:  x

Apply random layout

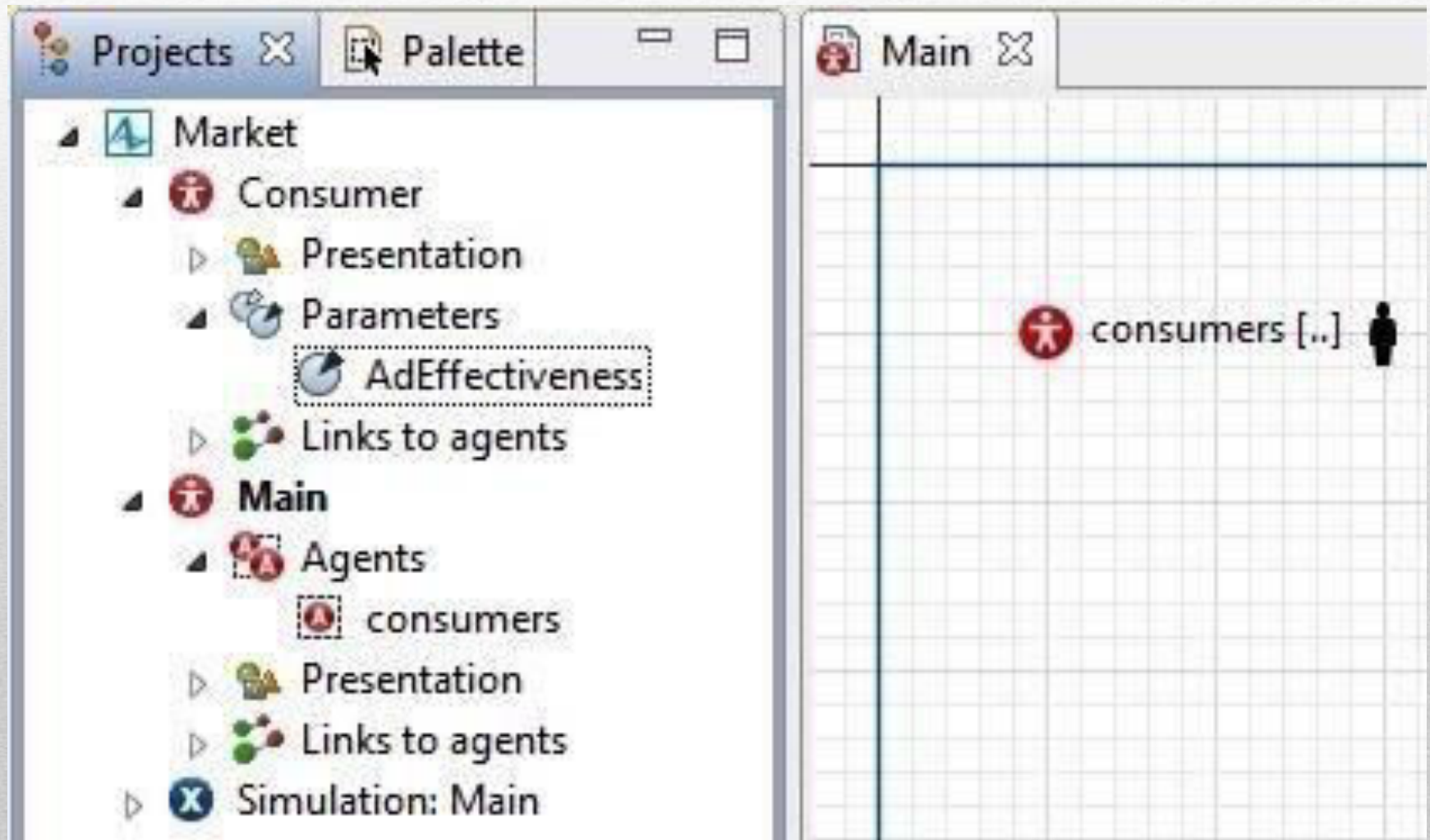
Network type:

< Back Next > Finish Cancel

- Ας χρησιμοποιήσουμε την προβολή Project για να δούμε τα νέα στοιχεία που δημιουργήθηκαν από τον οδηγό. Αναπτύξτε τα κλαδιά του δέντρου για να δούμε τα εσωτερικά στοιχεία του μοντέλου.
-



- Το μοντέλο μας έχει δύο τύπους πρακτόρων : Main and Consumer.
  - Το Consumer agent έχει το animation σχήμα person, και την παράμετρο AdEffectiveness.
  - Το Main agent type συμπεριλαμβάνει το πληθυσμό των πρακτόρων (agent population consumers a set of 5000 agents of type Consumer).
-



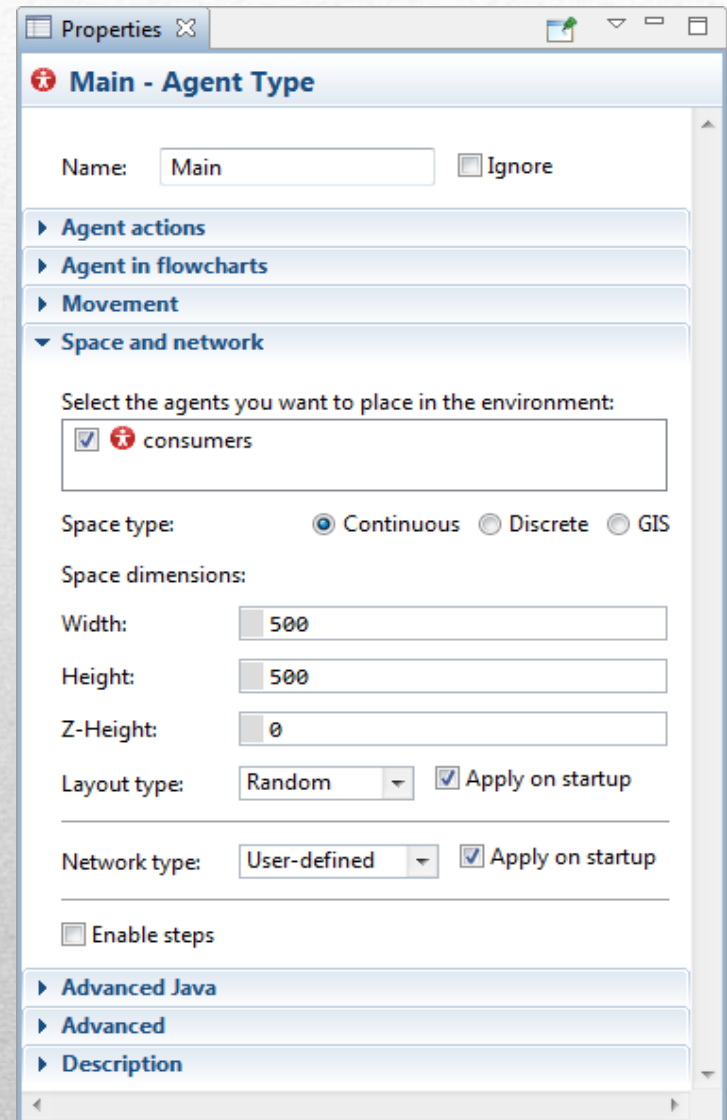
# Περιβάλλον Πρακτόρων


- Το Main Agent ενεργεί ως περιβάλλον για τον πληθυσμό των καταναλωτών.
  - Δεδομένου ότι το περιβάλλον ορίζει το χώρο, τη διάταξη, το δίκτυο και την επικοινωνία που χρησιμοποιούν οι πράκτορές μας, θα χρειαστούμε ένα περιβάλλον για να οργανώσουμε τις διαδράσεις των πρακτόρων μας και να μοντελοποιήσουμε τη διαφήμιση "από στόμα σε στόμα" που συμβαίνει όταν αλληλεπιδρούν οι παράγοντες μας.
-



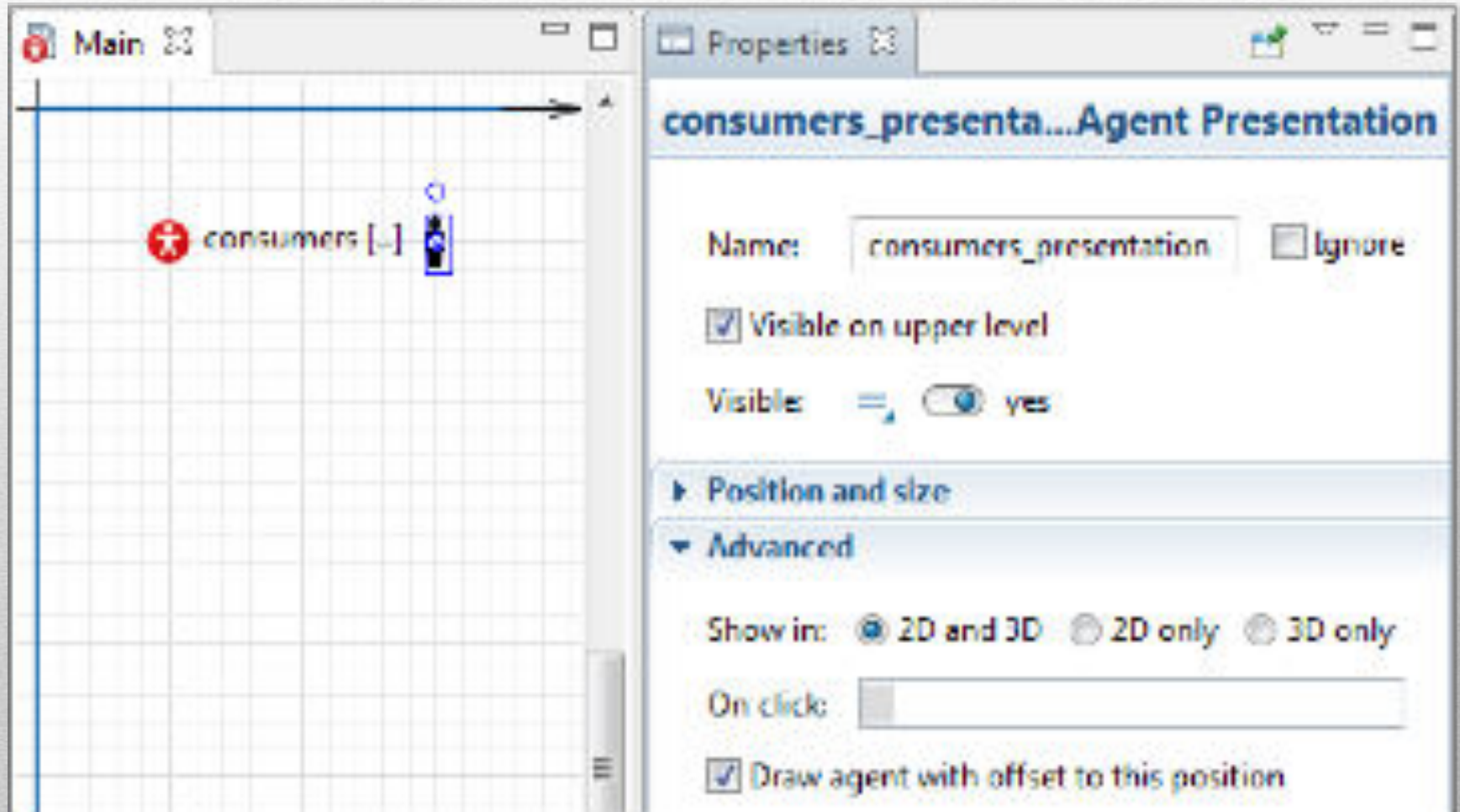
- Κάντε κλικ στο Main στα Project για να ανοίξετε τις ιδιότητές του στην προβολή Ιδιότητες (θα βρείτε τις Ιδιότητες στο δεξιό μισό του παραθύρου του AnyLogic).
  - Στο τμήμα Space and networks των Main properties (κύριων ιδιοτήτων), μπορείτε να προσαρμόσετε τις ρυθμίσεις περιβάλλοντος για τον πληθυσμό των καταναλωτών.
-

- Για να τροποποιήσετε τις ιδιότητες ενός στοιχείου, επιλέξτε το στοιχείο κάνοντας κλικ στο γραφικό επεξεργαστή ή στην προβολή Έργα και, στη συνέχεια, χρησιμοποιήστε την προβολή Ιδιότητες για να τροποποιήσετε τις ιδιότητες
- Η προβολή Ιδιότητες έχει πολλές ενότητες. Για να αναπτύξετε ή να συμπύξετε μια ενότητα, κάντε κλικ στον τίτλο της.
- Το όνομα και ο τύπος του επιλεγμένου στοιχείου εμφανίζονται στην κορυφή της προβολής.

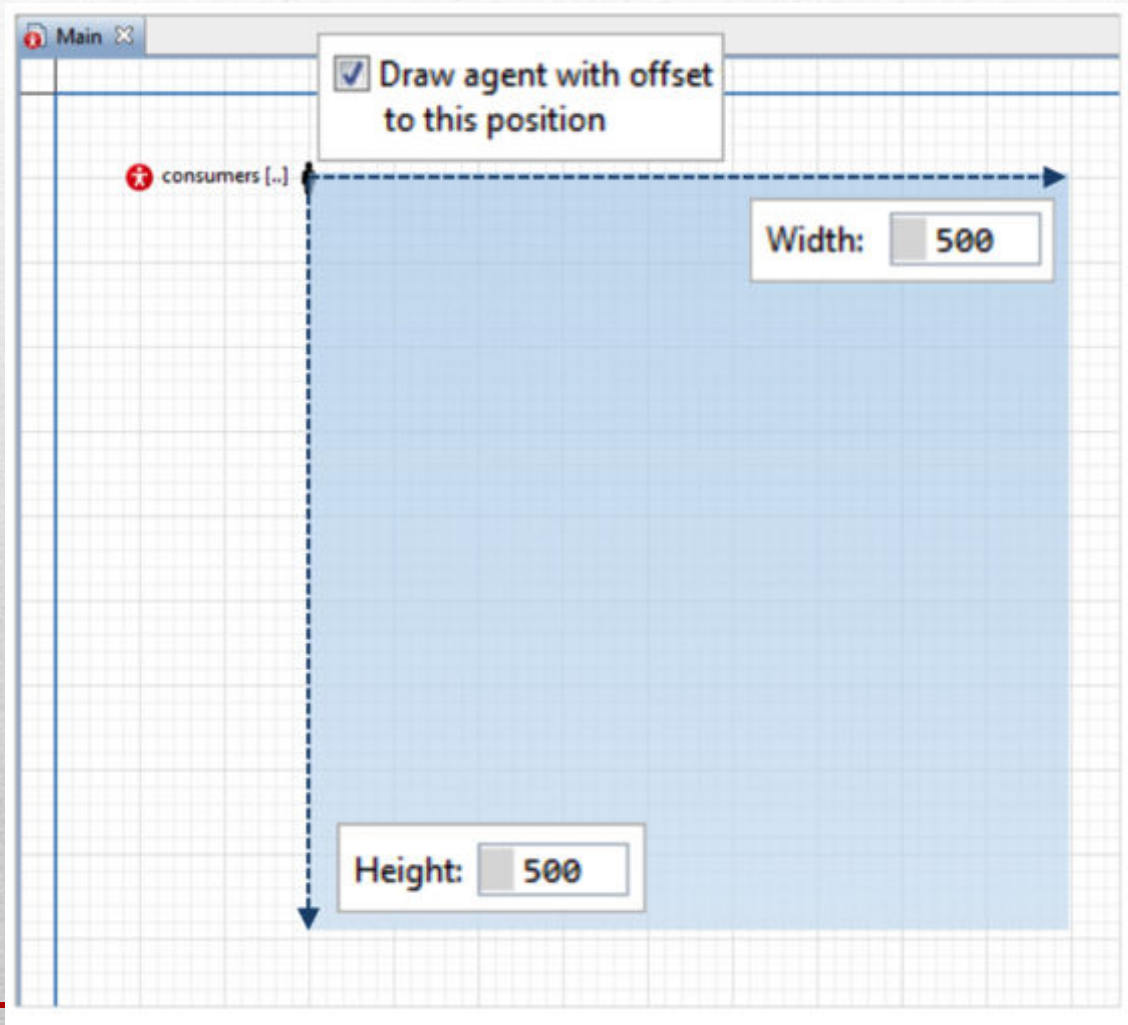


- Στο Main diagram, επιλέξτε τον πληθυσμό των πρακτόρων (agent population's) non-editable embedded
  - animation shape , ανοίξτε το **Advanced** properties, και επιλέξτε το **Draw agent with offset to this position** επιλογή.
-





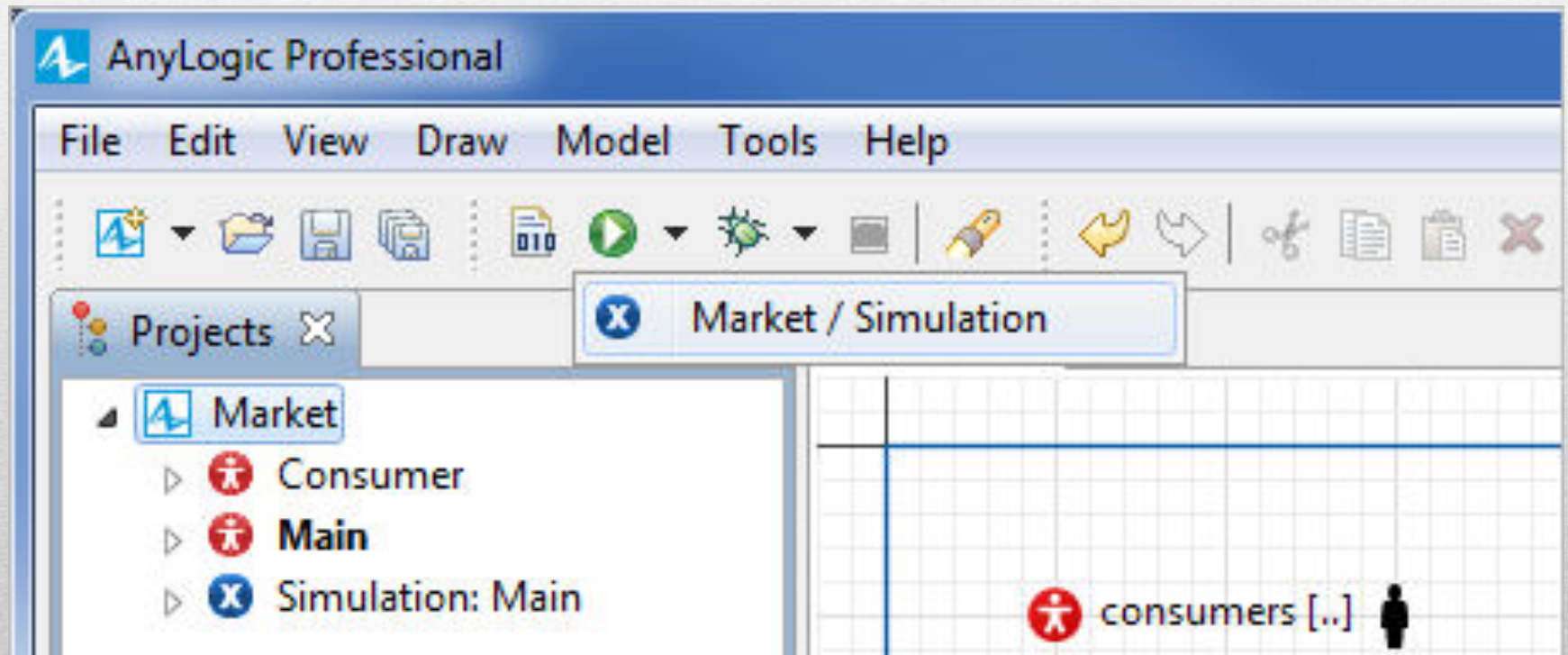


- Όπως βλέπετε στο σχήμα που ακολουθεί, το σχήμα κινούμενης εικόνας ορίζει την επάνω αριστερή γωνία του χώρου των 500x500 εικονοστοιχείων όπου θα βρίσκονται οι επιμέρους πράκτορες όταν τρέχουμε το μοντέλο
-



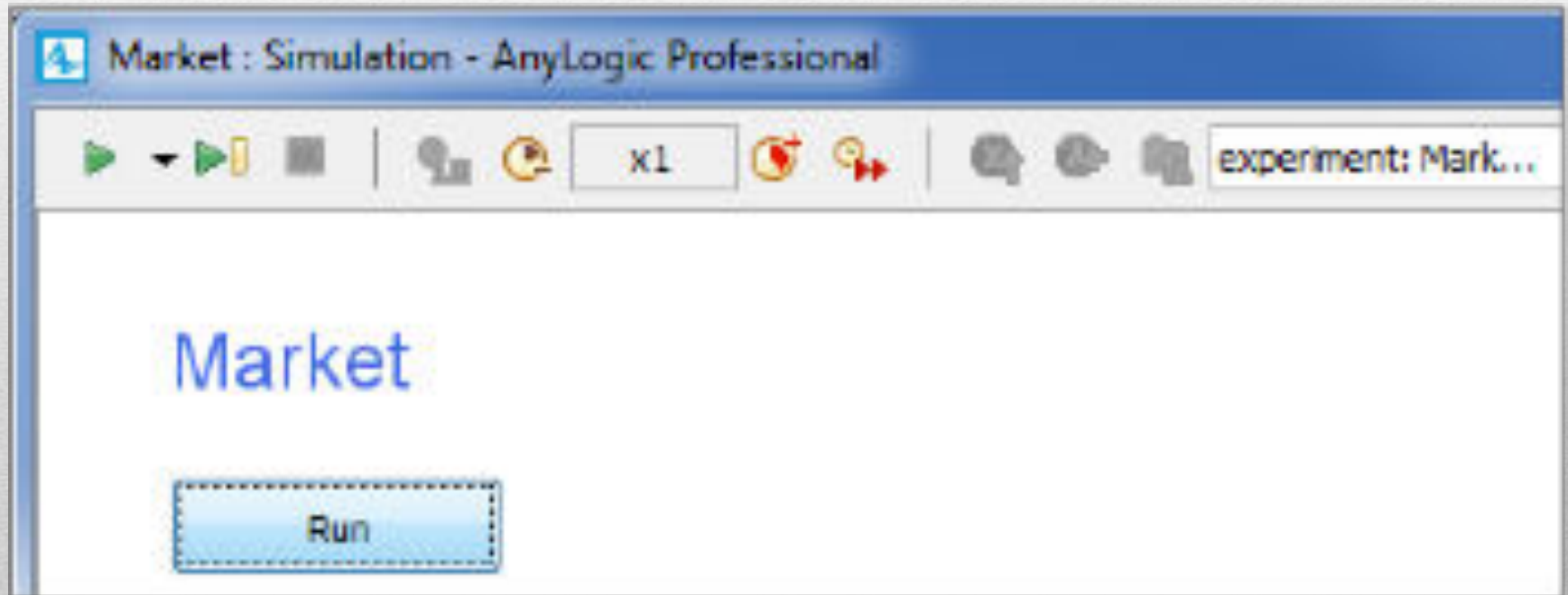


- Στο toolbar, κάνουμε ένα click στο κουμπί **Build**  για να δομηθεί το μοντέλο και να γίνει ο έλεγχος λαθών.
  - Πατάμε το μικρό τρίγωνο  για να τρέξει όποιο μοντέλο έχουμε επιλέξει.
-

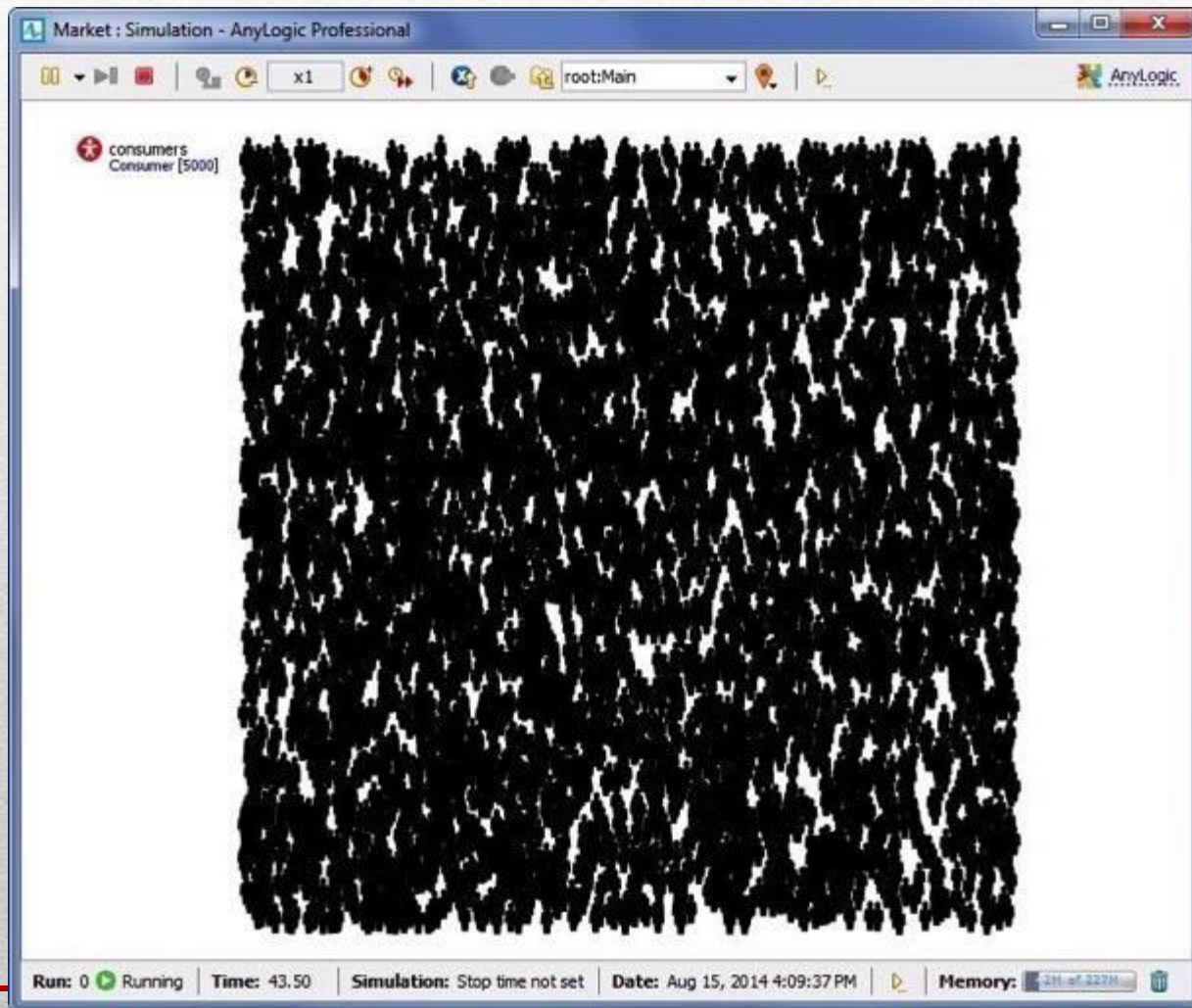


- Επειδή μπορείτε να έχετε πολλά μοντέλα ανοιχτά ταυτόχρονα - και κάθε μοντέλο μπορεί να έχει πολλά πειράματα - πρέπει να επιλέξετε το σωστό πείραμα.
  - Αυτή η ιδιότητα του Anylogic είναι σημαντική διότι μπορούμε να συγκρίνουμε αποτελέσματα διάφορων τρεξιμάτων.
  - Πατώντας το κουμπί run έχουμε αναπτύσσεται ένα καινούργιο παράθυρο
-





- Θα δείτε την παρουσίαση του μοντέλου (την παρουσίαση που δημιουργήσατε για τον κύριο πράκτορα) που δείχνει 5000 κινούμενες εικόνες για τους παράγοντες που αποτελούν τον πληθυσμό των καταναλωτών.
  - Δεδομένου ότι δεν δημιουργήσαμε συμπεριφορά για τους πράκτορές μας, η κινούμενη εικόνα φαίνεται ακίνητη.
-

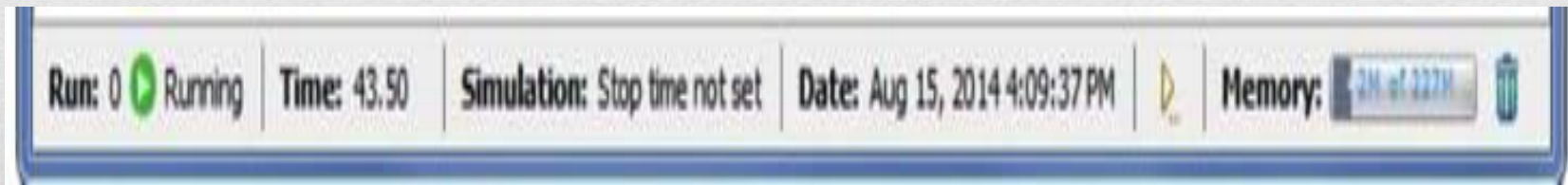




# Περιβάλλον του παραθύρου τρεξίματος του μοντέλου

- Για να διασφαλιστεί ότι το μοντέλο λειτουργεί, ανατρέξτε στη γραμμή κατάστασης στο κάτω μέρος του παραθύρου του μοντέλου:
  - Η γραμμή κατάστασης εμφανίζει την κατάσταση προσομοίωσης του μοντέλου (Running, Paused, Idle ή Finished), την τρέχουσα ώρα μοντέλου, την ημερομηνία μοντέλου κλπ.
  - Μπορείτε να προσαρμόσετε τη γραμμή κατάστασης κάνοντας κλικ στο κίτρινο κουμπί και επιλέγοντας τις απαιτούμενες ενότητες από την αναδυόμενη λίστα.
-

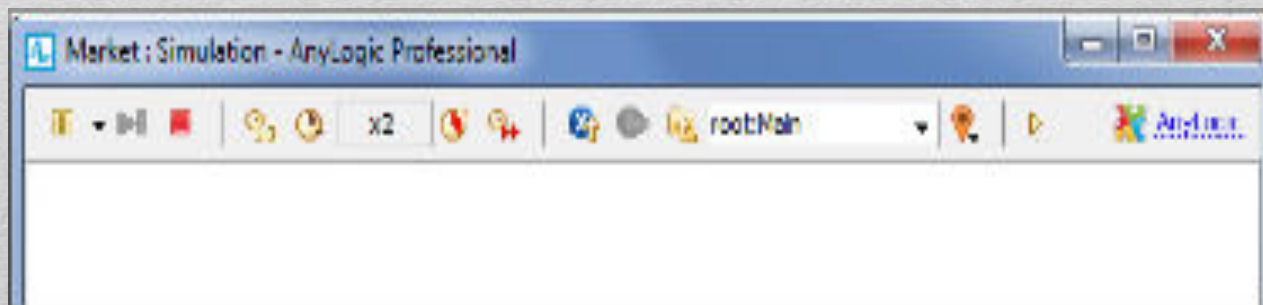
- Προσοχή στην Memory για την PLE άδεια



- Εκτέλεση
- [Ορατό όταν το μοντέλο δεν εκτελείται] Ξεκινά την προσομοίωση ή, αν η προσομοίωση τεθεί σε παύση, το επαναλαμβάνει.
- Βήμα
- Εκτελεί ένα συμβάν μοντέλου και στη συνέχεια διακόπτει την εκτέλεση του μοντέλου.
- Παύση
- [Ορατό όταν εκτελείται το μοντέλο] Παύση της προσομοίωσης. Μπορείτε να συνεχίσετε μια προσομοίωση σε παύση ανά πάσα στιγμή.
- Τερματισμός
- ---

Τερματίζει την τρέχουσα εκτέλεση του μοντέλου.







# Statecharts

---

# Statecharts

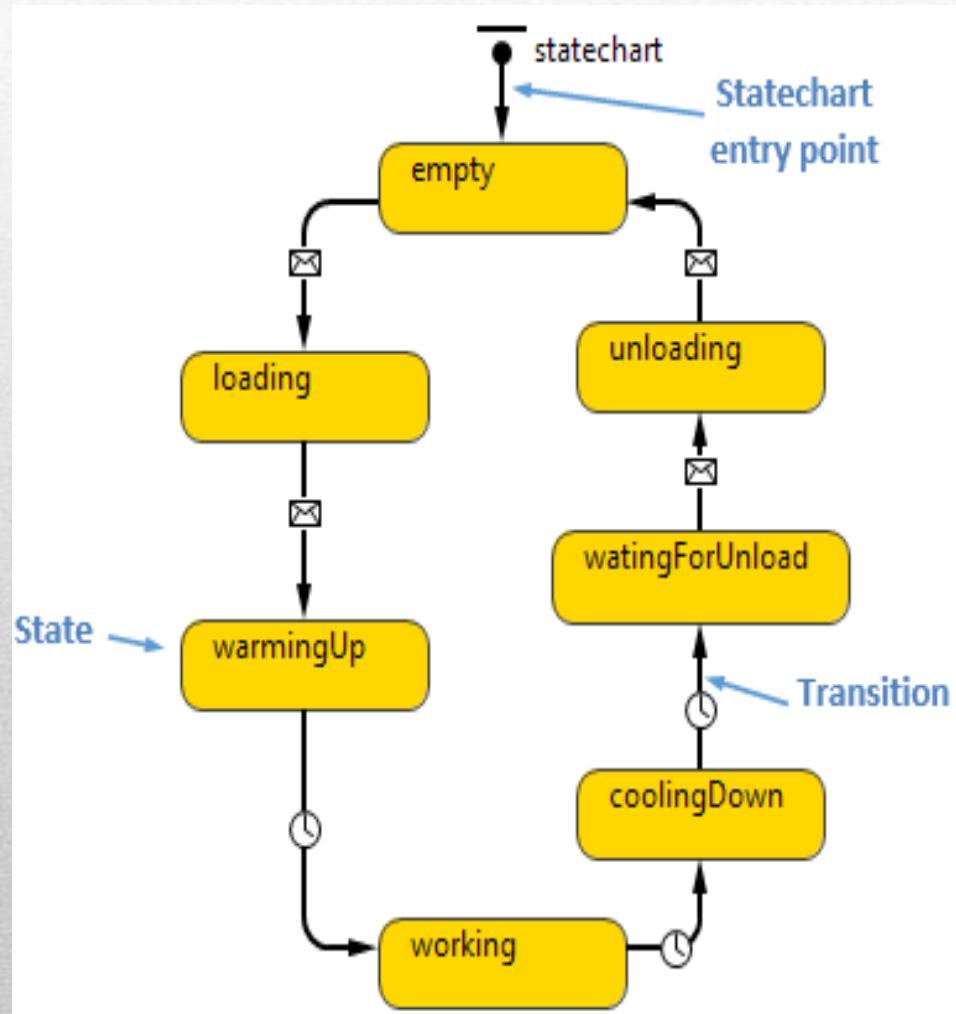
- Θα συνεχίσουμε να αναπτύσσουμε το μοντέλο μας καθορίζοντας τα χαρακτηριστικά και τις συμπεριφορές των καταναλωτών.
  - Ο καλύτερος τρόπος για να ορίσετε μια συμπεριφορά είναι να χρησιμοποιήσετε ένα statechart.
  - □ Τα Statecharts είναι το πιο προηγμένο μοντέλο για την περιγραφή συμπεριφοράς που βασίζεται σε γεγονότα και χρόνο.
  - Για ορισμένα αντικείμενα, αυτή η σειρά εντολών συμβάντων και χρόνου είναι τόσο διαδομένη ώστε μπορείτε να χαρακτηρίσετε καλύτερα τη συμπεριφορά τους χρησιμοποιώντας ένα διάγραμμα μεταβατικής κατάστασης
-



- Τα Statecharts έχουν καταστάσεις και μεταβάσεις.
  - Οι καταστάσεις statechart είναι εναλλακτικές, πράγμα που σημαίνει ότι το αντικείμενο μπορεί να είναι μόνο σε μία κατάσταση κάθε φορά.
  - Μια εκτέλεση μετάβασης μπορεί να οδηγήσει σε μια αλλαγή κατάστασης που κάνει ένα νέο σύνολο μεταβάσεων ενεργό.
  - Τα stateschart μπορεί να είναι ιεραρχικά - μπορεί να περιέχουν ακόμα άλλες καταστάσεις και μεταβάσεις.
  - □ Ένας πράκτορας μπορεί να έχει αρκετά statecharts που περιγράφουν ανεξάρτητα μέρη της συμπεριφοράς του πράκτορα.
-

# Παράδειγμα Statecharts

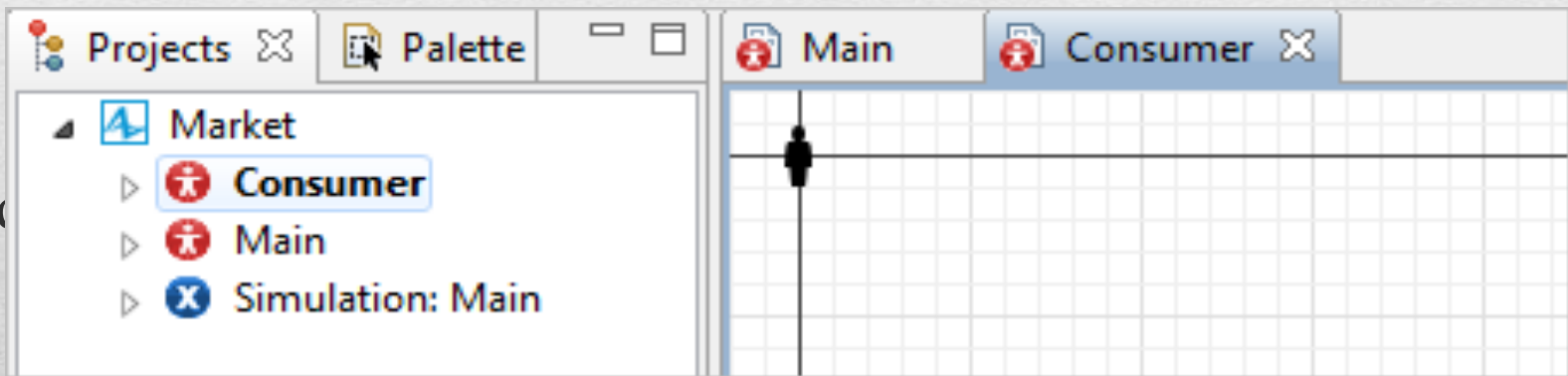
- Entry point
- State
- Transition



# Ανάπτυξη Παραδείγματος

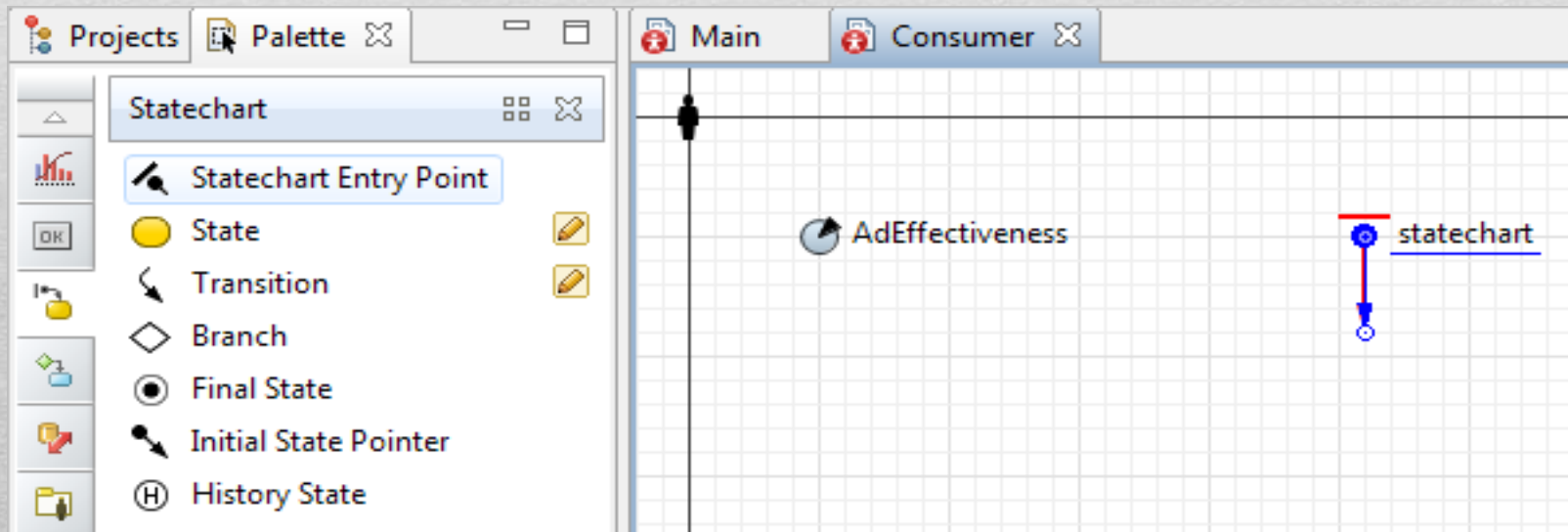
- Δεδομένου ότι το μοντέλο μας έχει δύο τύπους πρακτόρων, πρέπει να επιλεγεί ποιος πράκτορας είναι προς επεξεργασία στον γραφικό επεξεργαστή.
  - Το AnyLogic επιλέγει την καρτέλα του τύπου πράκτορα που έχετε ανοίξει στον γραφικό επεξεργαστή και τονίζει το στοιχείο του στο δέντρο έργων.
  - Μπορείτε να πλοηγηθείτε ανάμεσα σε ανοικτά γραφικά διαγράμματα διαφόρων τύπων πράκτορα κάνοντας κλικ στα ονόματα των καρτελών (για παράδειγμα, Main και Consumer):
-





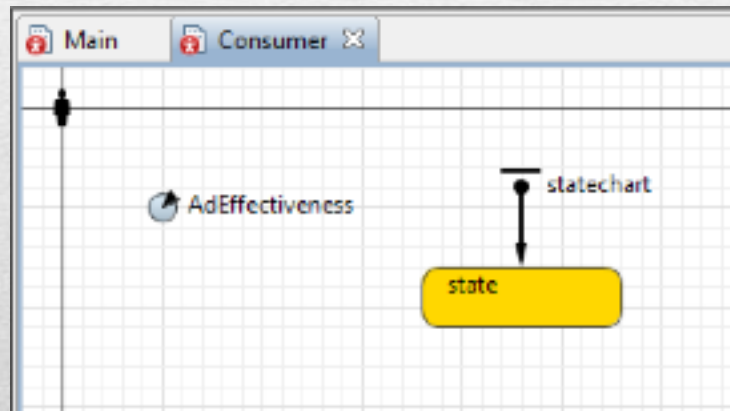
- Η έναρξη σχεδιασμού της κατανομής κατάστασης, σχεδιάζοντας δύο καταστάσεις.
  - Πραγματοποιείται με την παλέτα Statechart.
  - Σύροντας το σημείο εισαγωγής Statechart από την παλέτα Statech στο διάγραμμα Consumer.
  - Με αυτό τον τρόπο γίνεται η έναρξη σχεδιασμού ενός statechart με την προσθήκη ενός σημείου εισαγωγής statechart.
  - Το σημείο εισόδου ορίζει την έναρξη της ροής ελέγχου statechart και το όνομα του statechart.
-

- Είναι ορατό ότι το AnyLogic έχει επισημάνει με κόκκινο χρώμα το σημείο εισαγωγής statechart. Σημαίνει ότι το σημείο εισόδου δεν είναι συνδεδεμένο σε καμία κατάσταση και το τρέχον statechart δεν είναι έγκυρο.



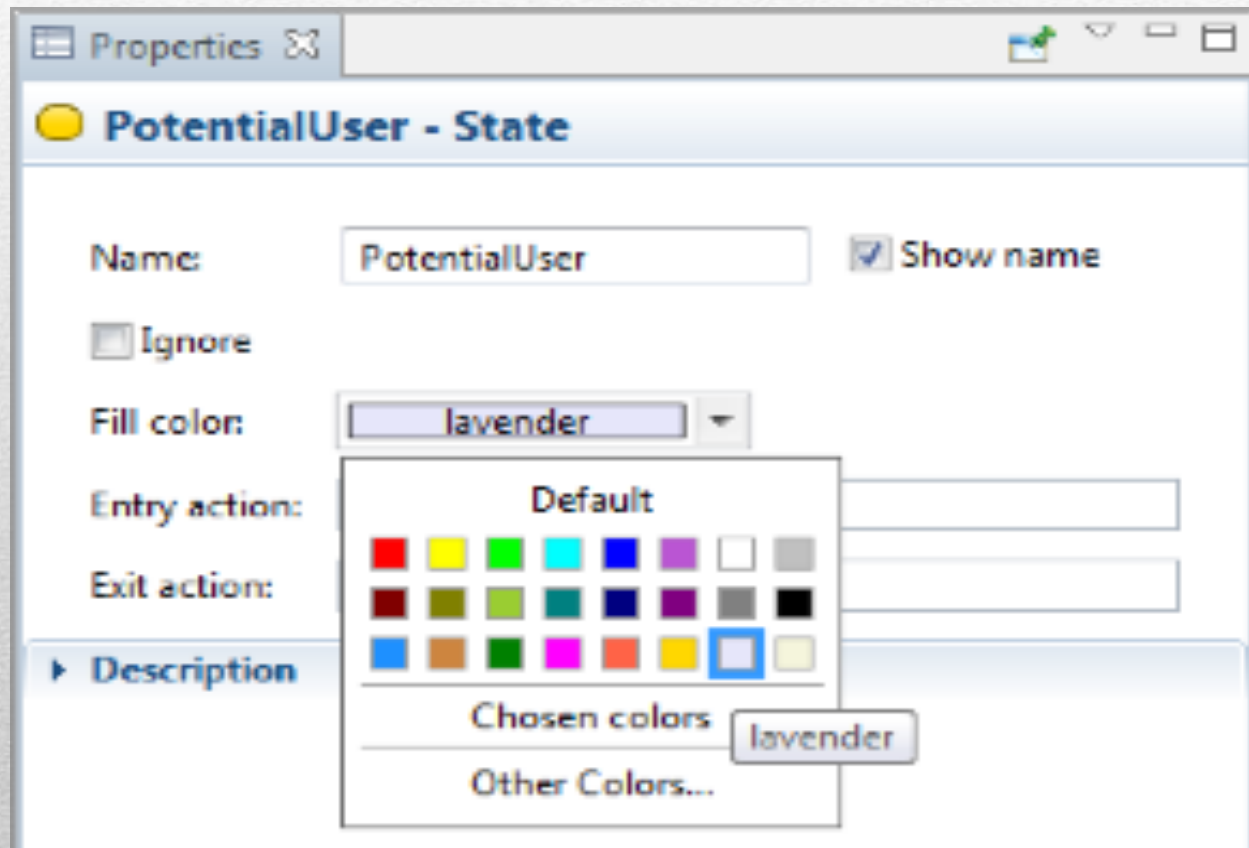


- Για την ανάπτυξη της πρώτης κατάστασης στο statechart του καταναλωτή.
  - Πρέπει να συρθεί το state από την παλέτα Statechart στο γραφικό διάγραμμα και να γίνει σύνδεση με το σημείο εισαγωγής statechart.
  - Για τις ανάγκες του παραδείγματος μας επιλέξτε την state στον επεξεργαστή γραφικών και τροποποιήστε τις ιδιότητές του.
-

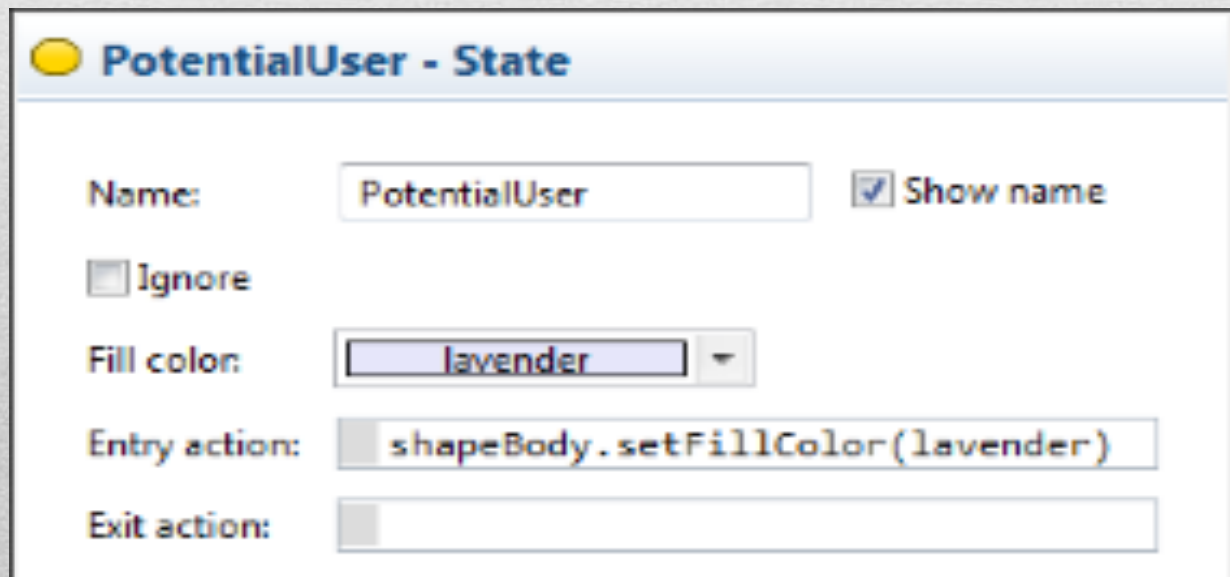


- Ονομάστε την κατάσταση PotentialUser.
  - Χρησιμοποιήστε το κουμπί ελέγχου χρώματος πλήρωσης για να αλλάξετε το χρώμα της κατάστασης σε λεβάντα.
-





- Η πρώτη εντολή Java
- Πληκτρολογήστε την ακόλουθη εντολή Java στο σημείο **Entry action** : `shapeBody.setFillColor(lavender)`



**PotentialUser - State**

Name:   Show name

Ignore

Fill color:

Entry action:

Exit action:

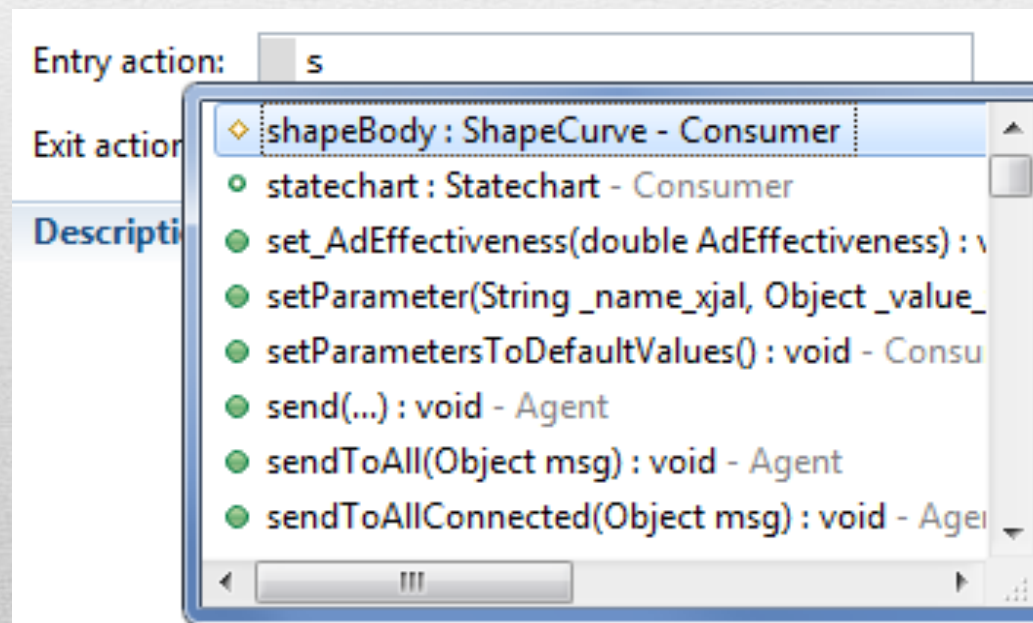
# Βοηθός ολοκλήρωσης κώδικα

- Μπορείτε να χρησιμοποιήσετε τον βοηθό ολοκλήρωσης κώδικα για να αποφύγετε την πληκτρολόγηση των πλήρων ονομάτων στοιχείων και λειτουργιών. Για να ανοίξετε τον βοηθό, κάντε κλικ στην επιθυμητή θέση στο πλαίσιο επεξεργασίας και πατήστε :
  - Ctrl + space για Windows
  - Alt + space στο Mac OS.
-



# Βοηθός ολοκλήρωσης κώδικα

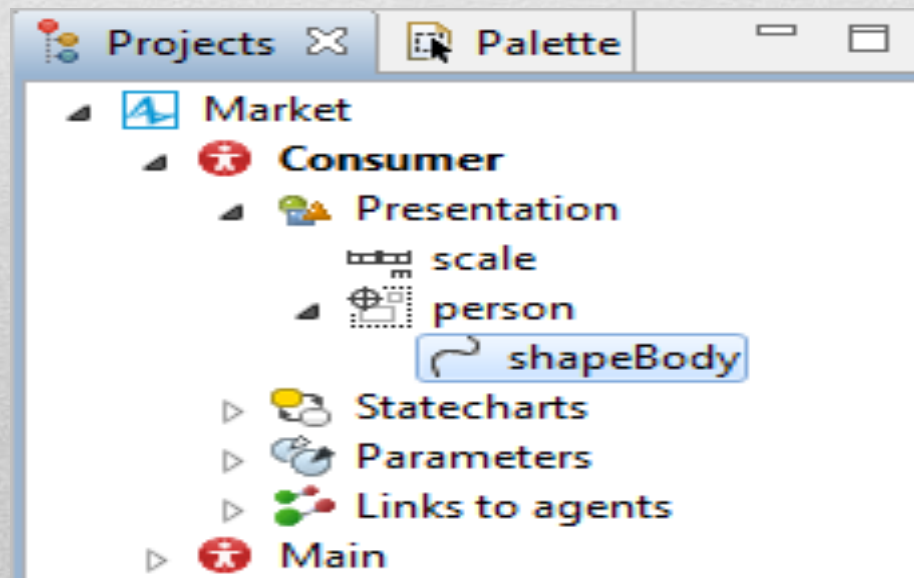
- Το αναδυόμενο παράθυρο παραθέτει τα στοιχεία μοντέλου που είναι διαθέσιμα στο δεδομένο πλαίσιο, όπως μεταβλητές μοντέλου, παραμέτρους ή λειτουργίες.
  - Με την μετάβαση στο όνομα του στοιχείου που είναι επιθυμητό είναι δυνατή η εισαγωγή της εντολής
  - Εναλλακτικά με την πληκτρολόγηση των πρώτων γραμμάτων του στοιχείου μέχρι να εμφανιστεί στη λίστα η εντολή που είναι επιθυμητή.
  - Με το πάτημα του Enter γίνεται εισαγωγή του ονόματος του στοιχείου στο πλαίσιο επεξεργασίας.
-



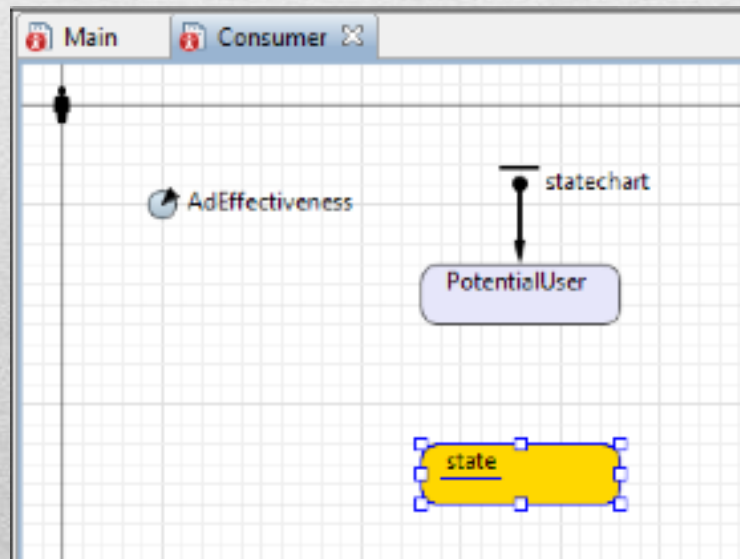
- η ενέργεια εισόδου εκτελείται όταν ο καταναλωτής μεταβαίνει σε άλλη κατάσταση.
  - Αυτός ο κώδικας εμφανίζει την αλλαγή κατάστασης αλλάζοντας το χρώμα του animation του καταναλωτή.
  - Εδώ, το shapeBody είναι το όνομα της μορφής κίνησης του καταναλωτή που δημιούργησε ο νέος πράκτορας πρακτόρων.
  - Εάν επεκτείνετε τον κλάδο παρουσίασης του Consumer στο δέντρο Project, θα δείτε το σχήμα shapeBody μέσα στην ομάδα ατόμων.
-



- Εάν επεκτείνετε τον κλάδο παρουσίασης του Consumer στο δέντρο Project, θα δείτε το σχήμα shapeBody μέσα στην ομάδα ατόμων.



- Πρόθεσή μας είναι να προσθέσουμε άλλη μια κατάσταση δίνοντάς της διαφορετικές ιδιότητες από την προηγούμενη.
  - Με τα εξής στοιχεία
  - **Name: User**
  - **Fill color: yellowGreen**  
**Entry action: shapeBody.setFillColor(yellowGreen);**
-



**Properties**

**User - State**

Name:   Show name  Ignore

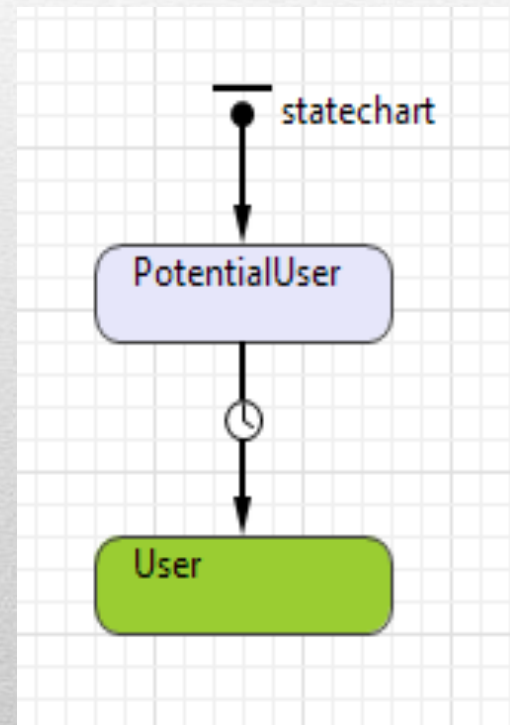
Fill color:

Entry action:



- Για να συνδέσουμε τις δύο καταστάσεις (State)
  - Σχεδιάστε μια μετάβαση (Transition) από το PotentialUser σε κατάσταση User για να οριστεί ο τρόπος με τον οποίο τα άτομα αγοράζουν το προϊόν και γίνονται χρήστες του προϊόντος.
  - Για να το κάνετε αυτό, κάντε διπλό κλικ στο
  - Το στοιχείο μεταβατικής κατάστασης της παλέτας Statechart (το εικονίδιο παλέτας του στοιχείου θα πρέπει να αλλάξει σε), κάντε κλικ στον PotentialUser και κάντε κλικ στην επιλογή User.
-

- Η κατάσταση από την προηγούμενη ενέργεια δείχνει αυτό το σχέδιο.

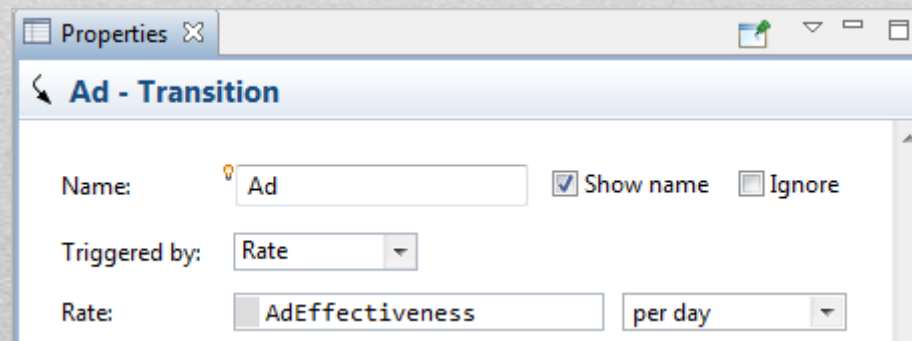




- Όπως την προηγούμενη φορά τροποποιήσαμε τις ιδιότητες του PotentialUser έτσι πρέπει να τροποποιηθούν και οι ιδιότητες του Transition.
  - Αυτό γίνεται διότι και το Transition είναι παραμετροποιήσιμο.
  - Ως πρώτη ιδιότητα πρέπει να του αποδώσουμε όνομα.
-



- Το όνομα που θέλω αποδώσουμε είναι το βραχύσωμο Ad μια σκηνή διαδικασία της διαφήμισης (advertising)
  - Το πλαίσιο Show name το τσεκάρουμε με σκοπό να εμφανίζεται το όνομα του Transition στο γραφικό διάγραμμα.
  - Η μετάβαση από την κατάσταση PotentialUser στο User θα διαμορφώσει τον τρόπο με τον οποίο η διαφήμιση οδηγεί το άτομο να αγοράσει το προϊόν.
  - Στη λίστα Triggered by, κάντε κλικ στην επιλογή Rate.
  - Στο πεδίο Τιμή, πληκτρολογήστε AdEffectiveness και, στη συνέχεια, κάντε κλικ ανά ημέρα (per day).
-

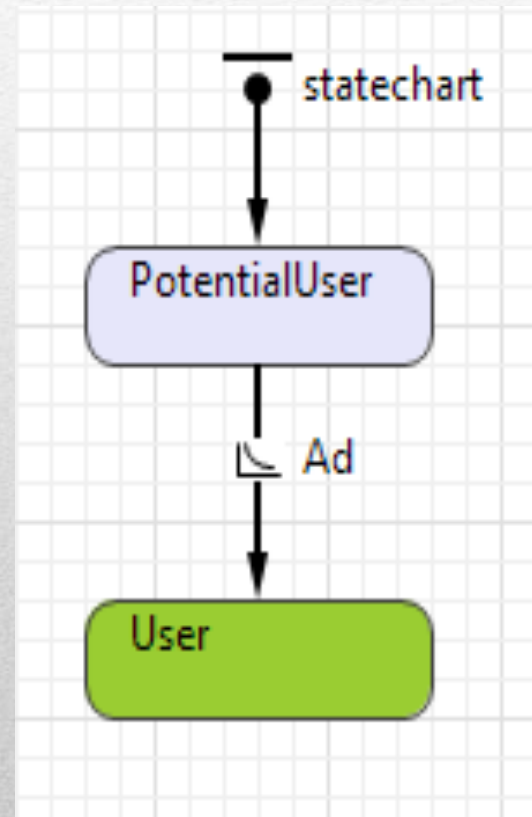
- Η τελική μορφή των ιδιοτήτων του Ad είναι όπως η παρακάτω εικόνα.




- Όπως είναι εύκολα παρατηρήσιμο το εικονίδιο δίπλα από το transition “Ad” έχει αλλάξει από το Ρολόι  σε καμπύλη διαγράμματος  .
  - Λύσεις υπάρχει δυνατότητα μεταφοράς του ονόματος και του εικονιδίου που συνοδεύουν το Transition μετά drag and drop κίνηση του ποντικιού
-




- Η μορφή του μέχρι τώρα σχεδιασμένου μοντέλου είναι η εξής



# Τύποι ενεργοποίησης μετάβασης


Transition trigger	Περιγραφή
<p data-bbox="112 462 285 505"><b>Timeout</b></p> 	<p data-bbox="981 462 1789 739">η μετάβαση λαμβάνει χώρα μετά από ένα συγκεκριμένο χρονικό διάστημα που υπολογίζεται από τη στιγμή που το statechart εισέρχεται στην κατάσταση «πηγής» (source) της μετάβασης.</p> <p data-bbox="981 748 1754 853">Η έκφραση του timeout μπορεί να είναι στοχαστική ή ντετερμινιστική.</p> <p data-bbox="981 862 1296 911">Κύριες χρήσεις:</p> <p data-bbox="981 919 1779 1082">Καθυστέρηση: μένει ο πράκτορας σε μια κατάσταση για ένα δεδομένο χρονικό διάστημα, στη συνέχεια φεύγει.</p> <p data-bbox="981 1090 1715 1310">Timeout: Αλλάζει την κατάσταση του πράκτορα εάν δεν εμφανιστούν άλλα αναμενόμενα συμβάντα εντός του καθορισμένου χρονικού διαστήματος.</p>

# Τύποι ενεργοποίησης μετάβασης


Transition trigger	Περιγραφή
<b>Rate</b> 	<p>Χρησιμοποιείται για την εφαρμογή μιας σποραδικής αλλαγής της κατάστασης με έναν γνωστό μέσο χρόνο.</p> <p>Λειτουργεί με τον ίδιο τρόπο όπως ένα χρονικό όριο που ενεργοποιεί την μετάβαση,</p> <p>Αλλά το χρονικό διάστημα αντλείται από μια εκθετική κατανομή παραμετροποιημένη με ένα δεδομένο ρυθμό.</p> <p>Για παράδειγμα, αν ο ρυθμός είναι 0.2, τα χρονικά όρια θα έχουν μέσες τιμές <math>1 / 0.2 = 5</math> χρονικές μονάδες.</p>




# Τύποι ενεργοποίησης μετάβασης

Transition trigger	Περιγραφή
<b>Condition</b> 	<p>Η μετάβαση παρακολουθεί μια καθορισμένη κατάσταση Boolean και αντιδρά όταν ισχύει.</p> <p>Η συνθήκη είναι μια αυθαίρετη έκφραση boolean και μπορεί να εξαρτάται από τις καταστάσεις οποιωνδήποτε αντικειμένων σε όλο το μοντέλο με συνεχή αλλά και διακριτή δυναμική.</p> <p>Λάβετε υπόψη ότι η κατάσταση ελέγχεται μόνο όταν συμβαίνουν κάποια συμβάντα στο μοντέλο.</p>

# Τύποι ενεργοποίησης μετάβασης

Transition trigger	Περιγραφή
<p>Message</p> 	<p>Αντιδρά σε μηνύματα άλλων παραγόντων. Τα μηνύματα μπορούν να διαμορφώσουν την επικοινωνία μεταξύ ανθρώπων, τις εντολές που δίδονται σε ένα μηχάνημα κλπ.</p> <p>Μπορεί να οριστεί το πρότυπο μηνύματος στις ιδιότητες μετάβασης, αλλά μόνο τα μηνύματα που αντιστοιχούν σε αυτό το πρότυπο θα ενεργοποιήσουν τη μετάβαση.</p>

# Τύποι ενεργοποίησης μετάβασης

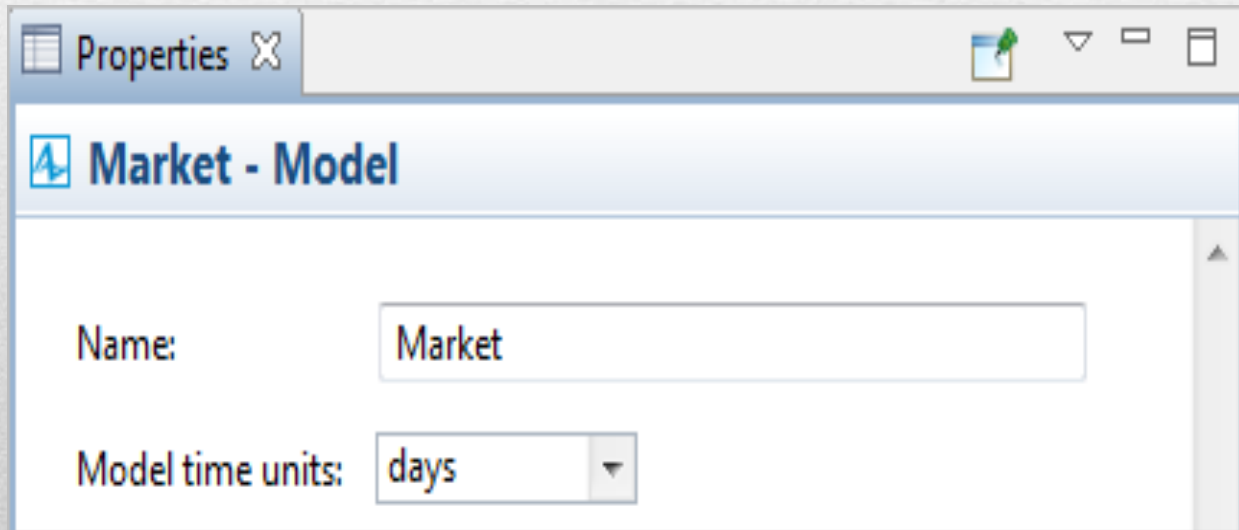
Transition trigger	Περιγραφή
<b>Arrival</b> 	αντιδρά στην άφιξη αυτού του πράκτορα στον προορισμό του. Λάβετε υπόψη ότι η συγκεκριμένη Transition αντιδρά μόνο εάν η κίνηση ξεκίνησε καλώντας τη λειτουργία του agent με την εντολή <code>moveTo()</code> .

---



- Η μετάβαση στο παράδειγμά μας ενεργοποιείται με έναν καθορισμένο ρυθμό.
  - Στην περίπτωσή μας, όταν το statechart εισέλθει στην κατάσταση PotentialUser, γίνεται κλήρωση από την εκθετική κατανομή και το χρονικό όριο που έχει ρυθμιστεί.
  - Ο χρόνος μετάβασης σε διαφορετική κατάσταση κάθε καταναλωτή θα διαφέρει, αν και κατά μέσο όρο το 1% των πιθανών χρηστών θα αγοράσει το προϊόν σε μια δεδομένη ημέρα.
-

- Θα πρέπει να ρυθμιστούν οι μονάδες χρόνου του μοντέλου.
  - Για να οριστεί η ρύθμιση του μοντέλου, πρέπει να υπάξει μετάβαση από την επιλογή Palette στο Project
  - και, στη συνέχεια, κάντε κλικ στο στοιχείο μοντέλου στο δέντρο.
  - Στην προβολή Ιδιότητες, επιλέξτε ημέρες ως Μονάδες χρόνου μοντέλου.
-



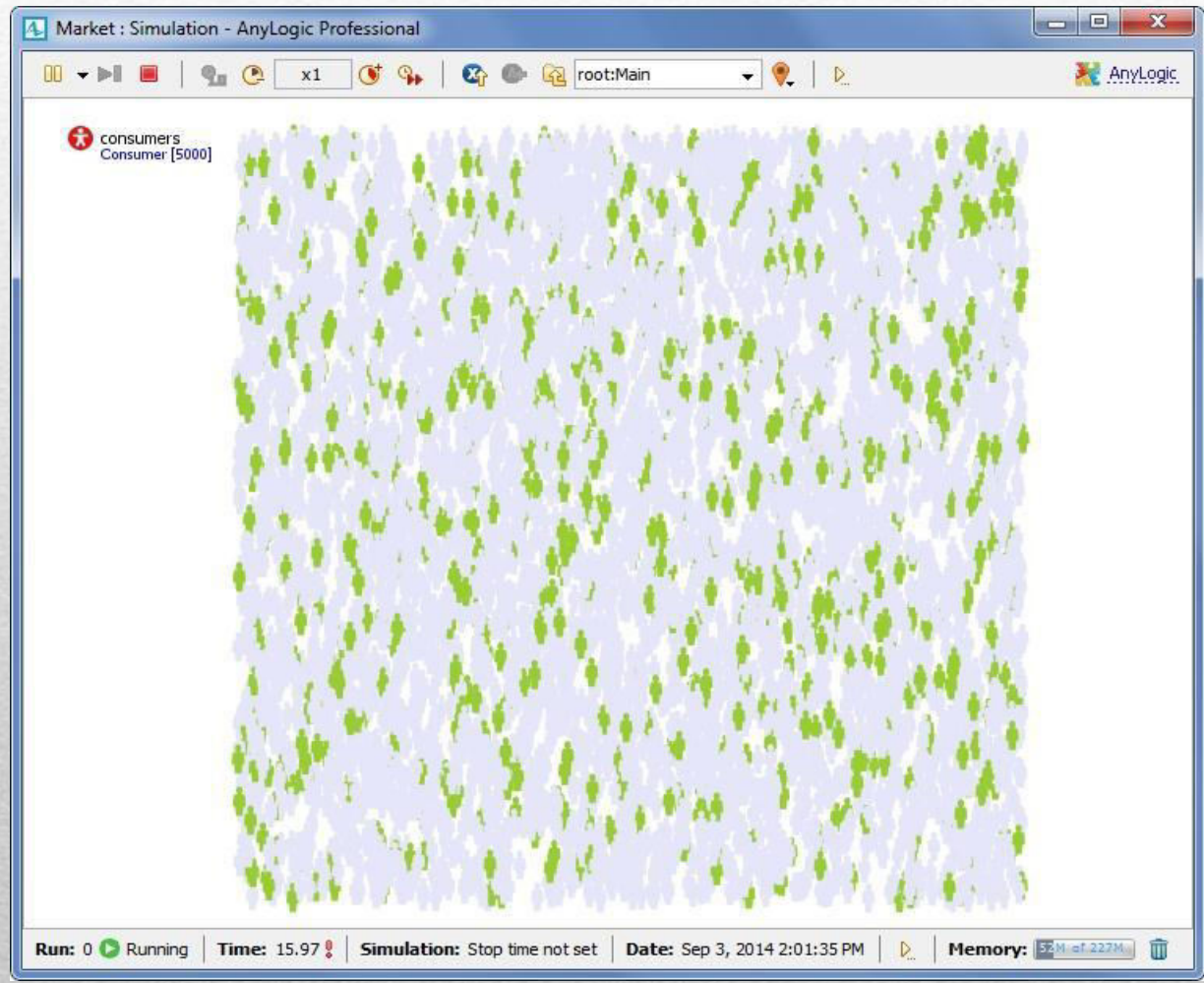


# Χρόνος του Μοντέλου

- Ο χρόνος μοντέλου είναι ο εικονικός (εξομοιωμένος) χρόνος που διατηρεί ο προσομοιωτής του AnyLogic. Ο χρόνος του μοντέλου δεν σχετίζεται με τον πραγματικό χρόνο ή το ρολόι του υπολογιστή, αν και μπορείτε να εκτελέσετε το μοντέλο σε κλίμακα σε πραγματικό χρόνο.
  - Για να ορίσετε τη σχέση μεταξύ του χρόνου μοντέλου και του πραγματικού κόσμου, όπου το μοντέλο του συστήματος ζει, θα πρέπει να επιλέξετε την πιο κατάλληλη μονάδα χρόνου για το μοντέλο σας, κοντά στις τυπικές διάρκειες λειτουργίας του μοντέλου σας.
-

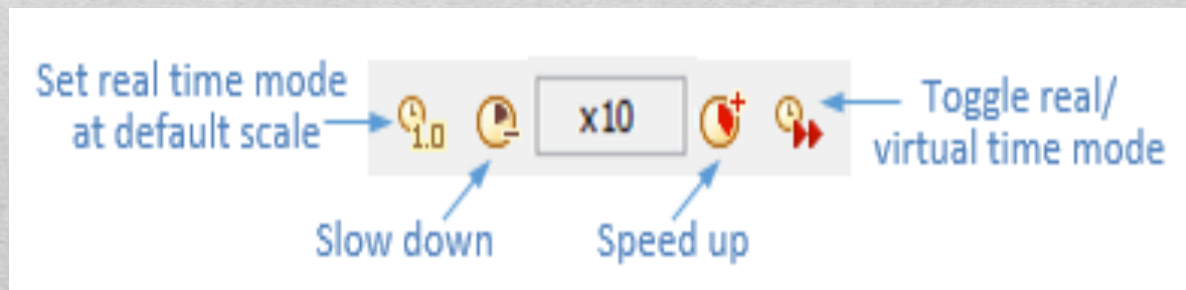
- Τρέχοντας το μοντέλο ο πληθυσμός των πρακτόρων σταδιακά από την κατάσταση PotentialUser θα γίνεται User.
  - Αυτό γραφικά απεικονίζεται από την μετατροπή των πρακτόρων του χρώματος μωβ σε χρώμα πράσινο.
  - Η όλη διαδικασία δείχνει τον επηρεασμό των πιθανών χρηστών από τη διαφήμιση, προκειμένου να μετατραπούν σε χρήστες του προϊόντος.
-







- Για την ρύθμιση της ταχύτητας εκτέλεσης του μοντέλου υπάρχει toolbar μέσω της οποίας μπορούμε να διαχειριστούμε τον εικονικό χρόνο





# **Στατιστικά σε Agent Based**

---

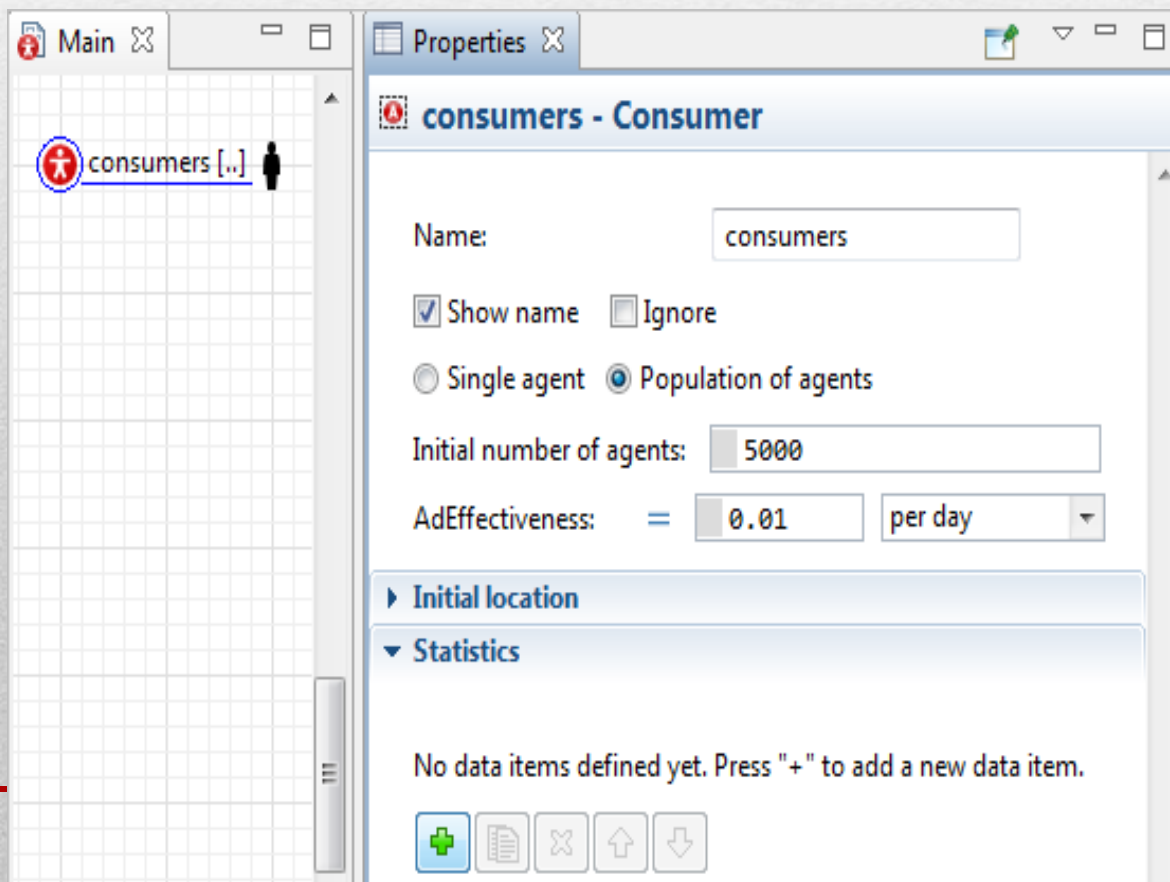
# Δημιουργία Στατιστικών

- Θέλουμε να μάθουμε πόσα άτομα αγόρασαν το προϊόν μας σε μια δεδομένη στιγμή. Με αυτό κατά νου, θα ορίσουμε τις λειτουργίες που μετράνε τους χρήστες και τους πιθανούς χρήστες του προϊόντος μας και στη συνέχεια προσθέτουμε ένα γράφημα για να δείξουμε τη δυναμική.
-



- Κατ 'αρχάς, ορίστε μια συνάρτηση για να μετρήσετε τους δυνητικούς χρήστες
  - Για να προσθέσετε μια νέα λειτουργία που συλλέγει στατιστικά στοιχεία για τους πράκτορες, ανοίξτε το διάγραμμα του τύπου πράκτορα Main, επιλέγουμε τον πληθυσμό των πελατών και στις καρτέλες των ιδιοτήτων ανοίγουμε εκείνη που έχει τα στατιστικά.
-

- Με την διαδικασία αυτή θέλουμε να καθορίσουμε πόσοι πράκτορες υπάρχουν στην ενότητα PotentialUser



- Για το σκοπό αυτό ορίζουμε μία συνάρτηση του τύπου Count με νέο όνομα (Name) Npotential
  - Οι στατιστικές του αριθμού των τύπων επαναλαμβάνονται μέσω ενός δεδομένου πληθυσμού - στην περίπτωσή μας, του αριθμού των πρακτόρων- **για να μετρήσουν εκείνα που πληρούν την επιλεγμένη κατάσταση.**
-



- Εισάγουμε την εντολή `item.inState(Consumer.PotentialUser)` ως συνάρτηση στο πεδίο Condition

▼ Statistics

Name:


Type:  Count  Sum  Average  Min  Max

Condition:


+ [Document Icon] ✕ ↑ ↓


# Εξήγηση της στατιστικής εντολής

- Το `item` αντιπροσωπεύει τον πράκτορα που ελέγχεται επί του παρόντος στην επανάληψη.
  - `inState ()` είναι μια λειτουργία που ελέγχει εάν η καθορισμένη κατάσταση του `statechart` είναι ενεργή.
  - Το `PotentialUser` είναι το όνομα της καθορισμένης από τον πράκτορα κατάστασης και γι 'αυτό χρειάζεται το πρόθεμα τύπου του παράγοντα `Consumer`.
-

- Ορίστε μια δεύτερη λειτουργία στατιστικών για τον υπολογισμό του αριθμού των χρηστών του προϊόντος. Ονομάστε το NUser και αφήστε τον να μετρήσει τον αριθμό των παραγόντων, σύμφωνα με το στοιχείο condition.inState (Consumer.User).
  - Μπορείτε να αντιγράψετε την λειτουργία στατιστικών στοιχείων κάνοντας κλικ στο κουμπί  Διπλότυπο και αλλάζοντας το όνομα και την κατάσταση.
-



Properties 

 **consumers - Consumer**

▼ **Statistics**

Name:






Type:  Count  Sum  Average  Min  Max

Condition:

Name:

Type:  Count  Sum  Average  Min  Max

Condition:



# **Διαγράμματα σε Agent Based**

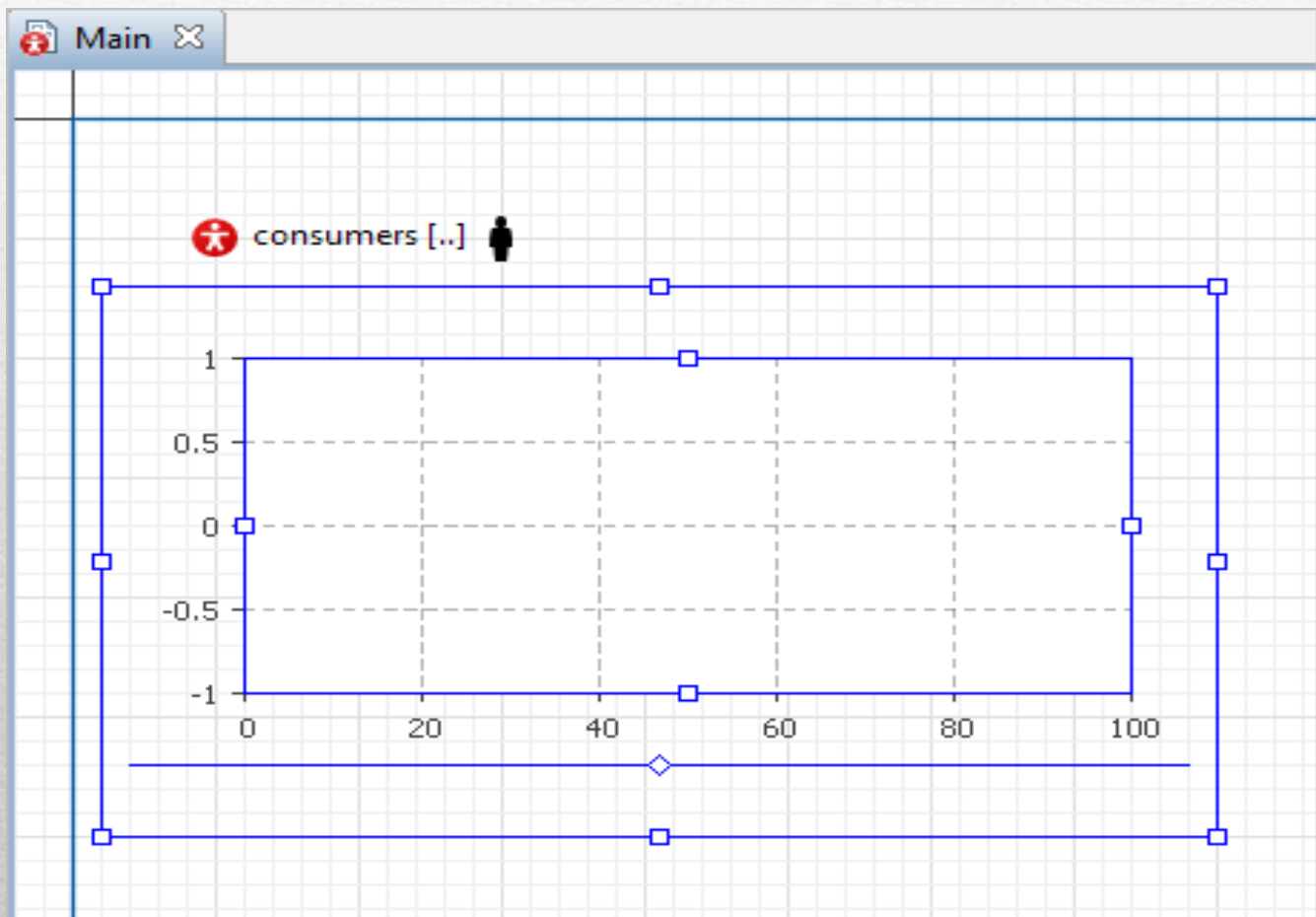
---

# Δημιουργία Διαγράμματος

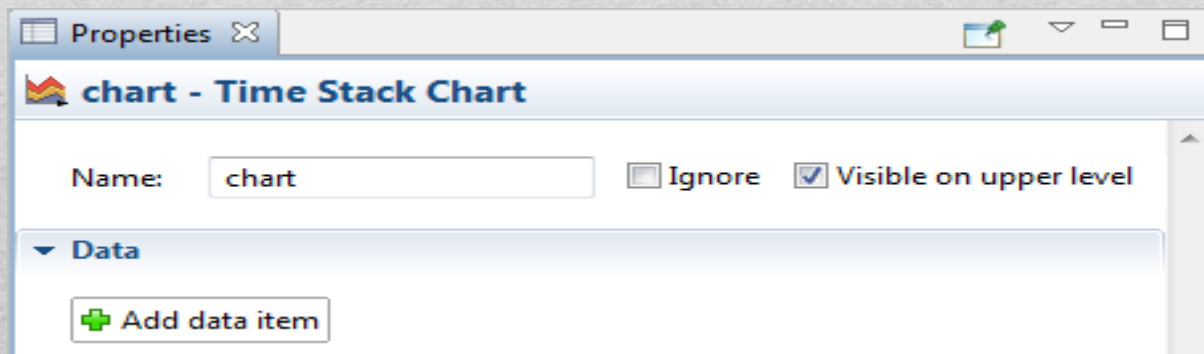
- Μετά την διαδικασία συγκέντρωσης των στατιστικών στοιχείων με τις δύο μεταβλητές, μπορούμε να δημιουργήσουμε ένα γράφημα το οποίο να εμφανίζει την εξέλιξη των στατιστικών στοιχείων σε σχέση με το χρόνο.
-



- Ανοίξτε την παλέτα Analysis και σύρετε το Διάγραμμα χρόνου στοίβας (Time Stack Chart) από την παλέτα Ανάλυσης στο κύριο διάγραμμα για να δημιουργήσετε ένα γράφημα που θα εμφανίζει τη δυναμική των χρηστών και των δυνητικών χρηστών.
  - Επειδή το χρονοδιάγραμμα που θα δημιουργηθεί είναι μικρό σε μέγεθος το μέγιστο πιούμε προκειμένου να μας δώσει την πορεία των δεδομένων μας για μια μεγαλύτερη χρονική περίοδο.
-

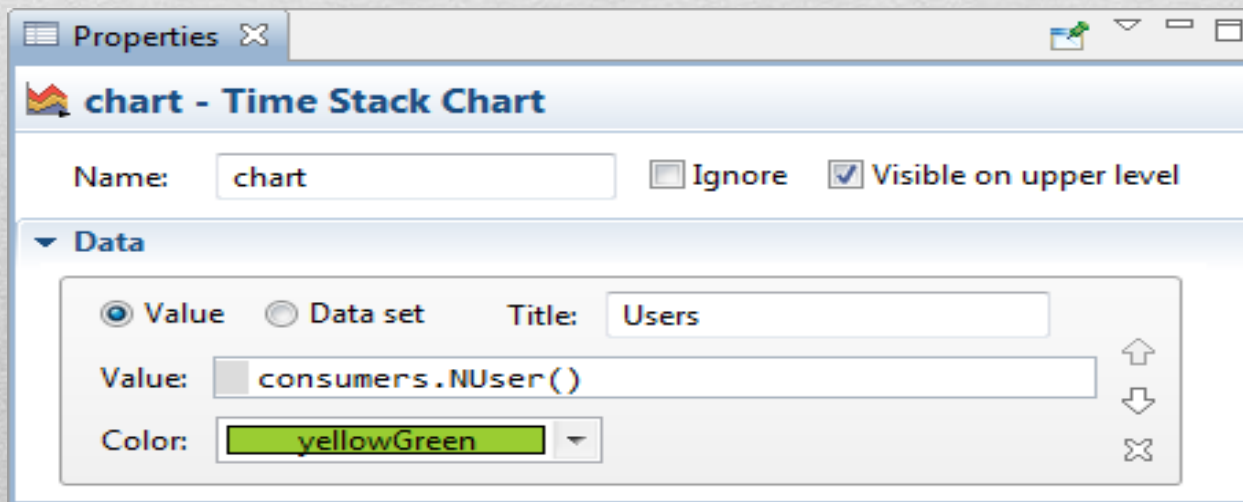


- Στην περίπτωση του παραδείγματος μας θα μπορούσαμε να προσθέσουμε δύο δεδομένα να απεικονίζονται ταυτόχρονα στο διάγραμμα που φτιάξαμε.
- Για το σκοπό αυτό πήγαινε στις ιδιότητες (Properties) του διαγράμματος
- Πατάμε το κουμπί με τον πράσινο σταυρό Add data item

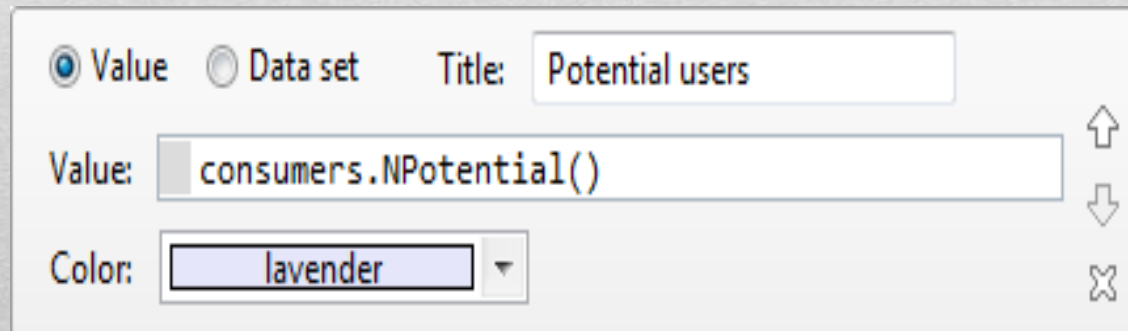




- Και μετατρέπουμε τις ιδιότητες του διαγράμματος με τα εξής δεδομένα
- **Title:** Users – the data item's title.
- **Color:** yellowGreen
- **Value:** consumers.NUser()



- Για να απεικονίζεται στο ίδιο διάγραμμα ο πιθανός χρήστης προσθέτουμε δεδομένα με τον ίδιο τρόπο με τις εξής τιμές
- **Title:** Potential users
- **Color:** lavender
- **Value:** consumers.NPotential()

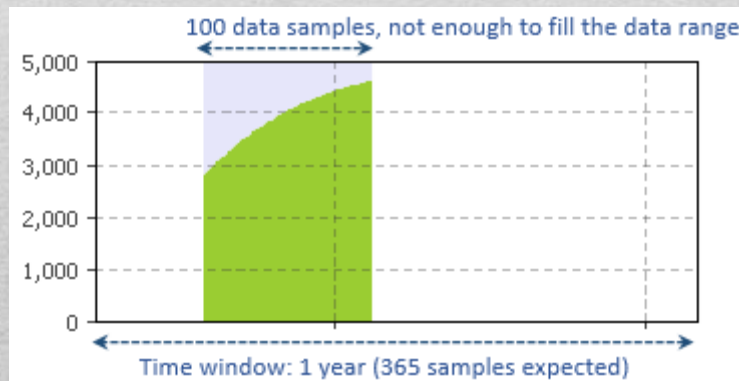


The image shows a configuration dialog box for a chart series. It has three radio buttons at the top: 'Value' (selected), 'Data set', and 'Title'. The 'Title' field contains the text 'Potential users'. Below this, there are three rows of configuration options:

- Value:** A text box containing the expression `consumers.NPotential()`. To the right of this box are two vertical arrows, one pointing up and one pointing down.
- Color:** A dropdown menu showing the color 'lavender'. To the right of this dropdown is a small 'X' icon.

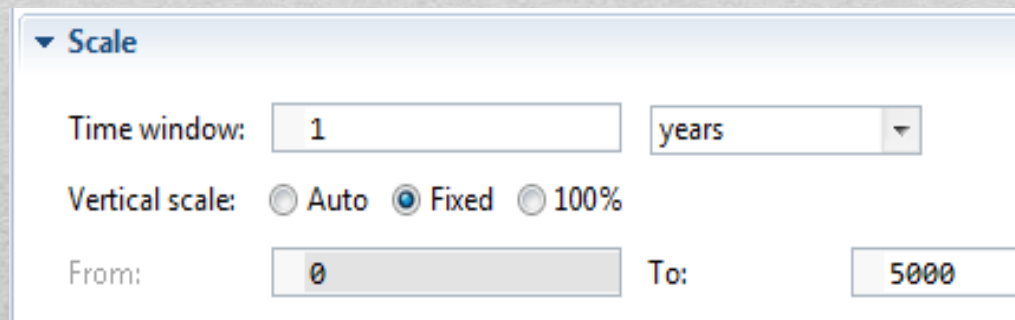
# Ρύθμιση της κλίμακας χρόνου του πίνακα

- Ένα από τα σημαντικά στοιχεία για τη σωστή απεικόνιση των δεδομένων είναι η σωστή ρύθμιση του χρόνου που απεικονίζεται στο διάγραμμα.





- Για να ρυθμίσουμε κατά συνέπεια το διάγραμμα χρονικά πατάμε την επιλογή Scale από τις Properties
- Θέτουμε στο Time window 1 και years
- Στο Check box με το όνομα Vertical scale επιλέγουμε Fixed
- Τέλος εφόσον τα στατιστικά που θέλουμε να κρατηθούν και να απεικονιστούν στο διάγραμμα είναι για το σύνολο του πληθυσμού των πρακτόρων που έχουμε μέσα στο μοντέλο θέτουμε στο πεδίο To: 5000 (όσο και ο πληθυσμός των πρακτόρων που έχω επιλέξει να έχει το μοντέλο)








The screenshot shows a 'Scale' settings panel with the following configuration:

- Time window:** 1 years
- Vertical scale:** Fixed (selected)
- From:** 0
- To:** 5000


- Δεδομένου ότι θα προσθέσουμε ένα δείγμα δεδομένων για κάθε μέρα, τα 365 δείγματα δεδομένων είναι ένα ιδανικό ποσό για ένα χρονικό διάστημα.
- Αφού ορίσαμε το χρονικό παράθυρο, αλλάξτε τον μέγιστο αριθμό δειγμάτων δεδομένων που εμφανίζει το διάγραμμα μεταβαίνοντας στην ενότητα **Data update** και **Display up to 365 latest samples**.

- 

---

Properties     

### chart - Time Stack Chart

Name:    Ignore  Visible on upper level

▶ Data



▼ Data update

Update data automatically  
 Do not update data automatically

---

Use model time  Use calendar dates

First update time (absolute):

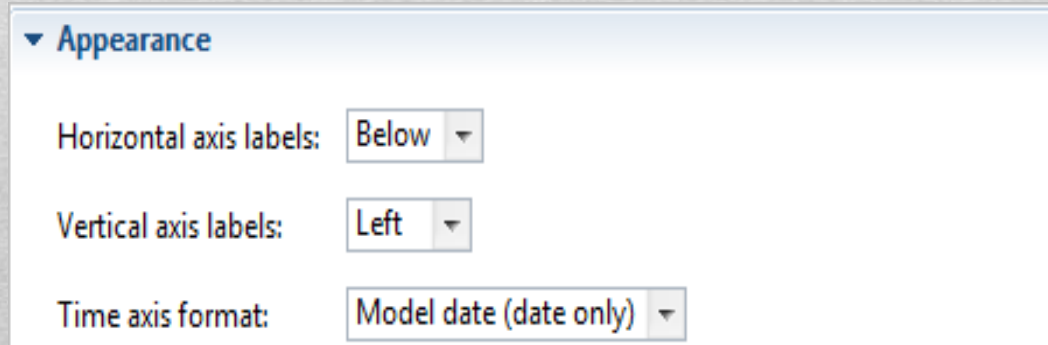
Update date:    

Recurrence time:

Display up to  latest samples  
(applies to "Value" data items only)



- Μεταβείτε στις ιδιότητες εμφάνισης (Appearance properties) της φόρμας χρονικής στοίβας και ρυθμίστε την εμφάνιση της ημερομηνίας μοντέλου (ημερομηνία μόνο) Model date (date only) κοντά στον άξονα του χρόνου.



▼ Appearance

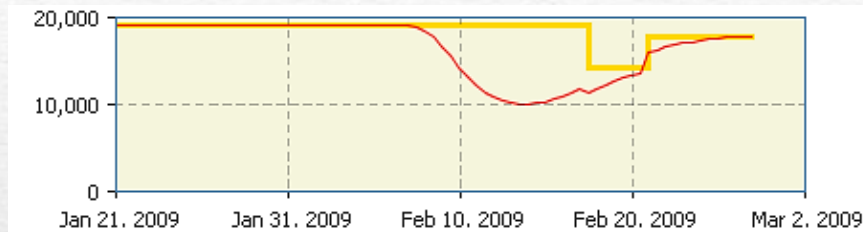
Horizontal axis labels: Below ▼

Vertical axis labels: Left ▼

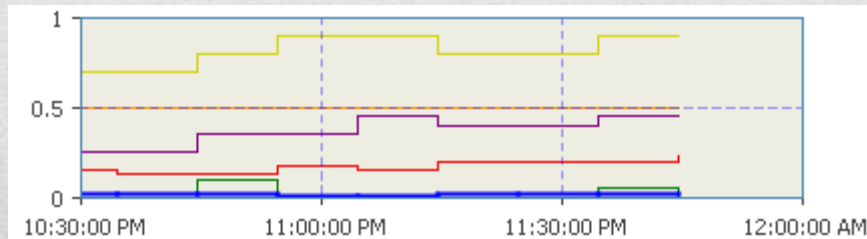
Time axis format: Model date (date only) ▼

- Η τελευταία ρύθμιση των διαγραμμάτων είναι αναφορικά με το τι θέλουμε να απεικονίζει ο άξονας των  $X$  από άποψη εύρους χρόνου.
-

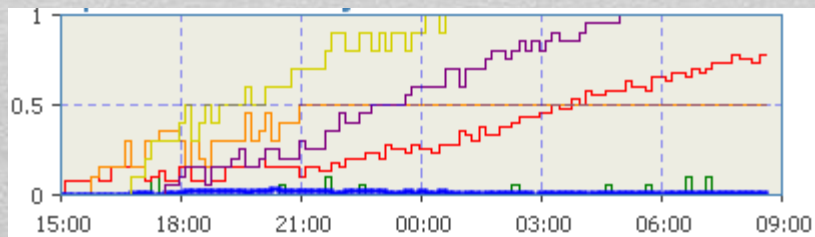
- **Model date (date only)**



## Model date (time only)

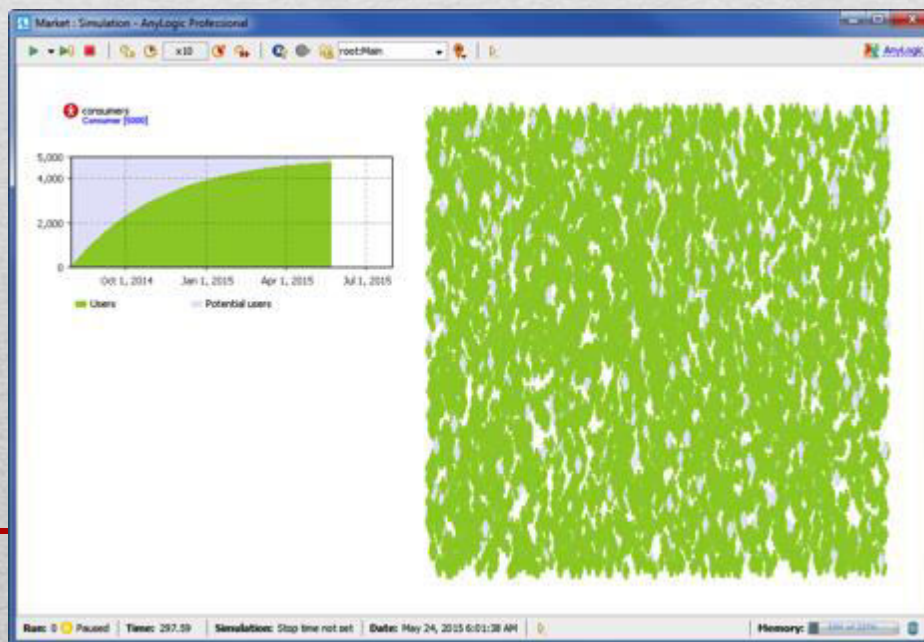
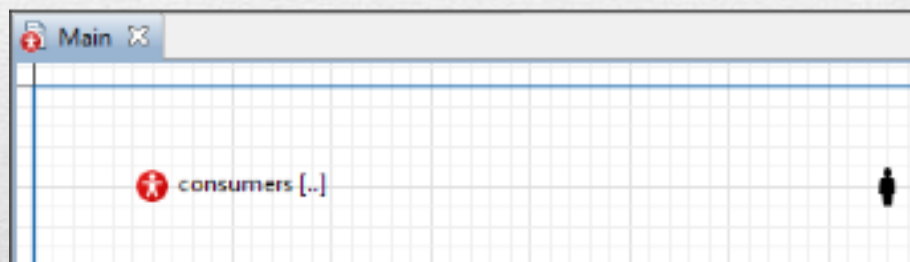


- **Custom (HH:mm) - Only hours and minutes are displayed**



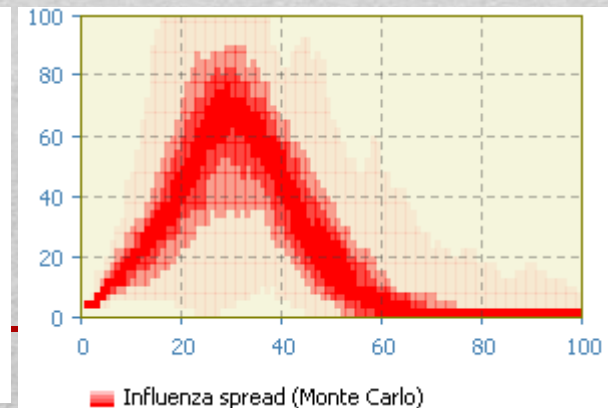
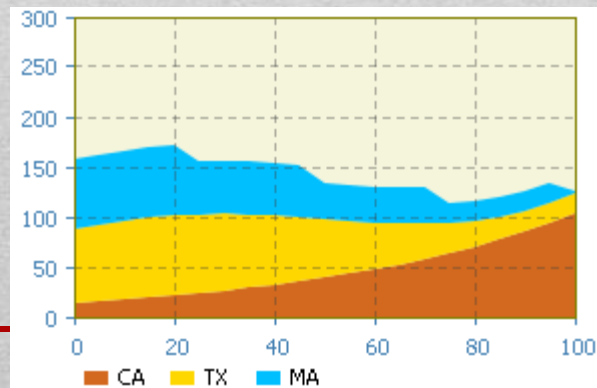
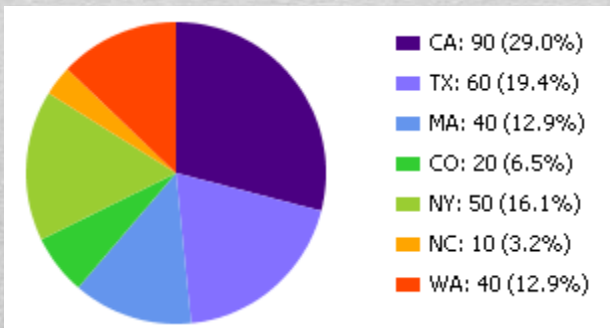
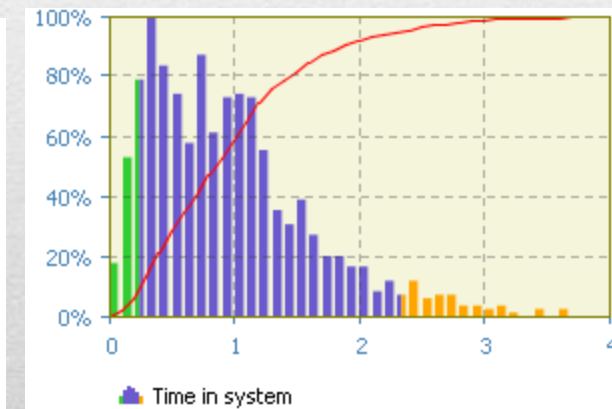
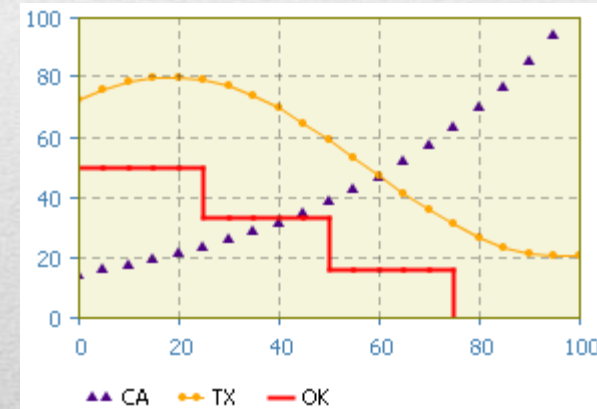
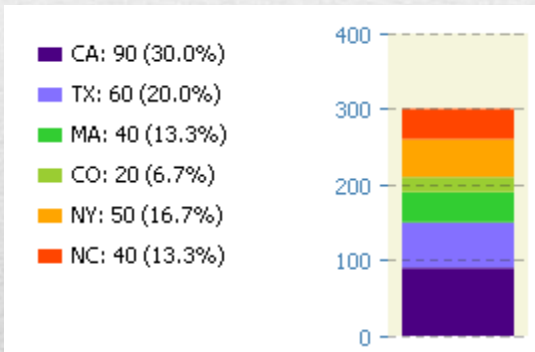
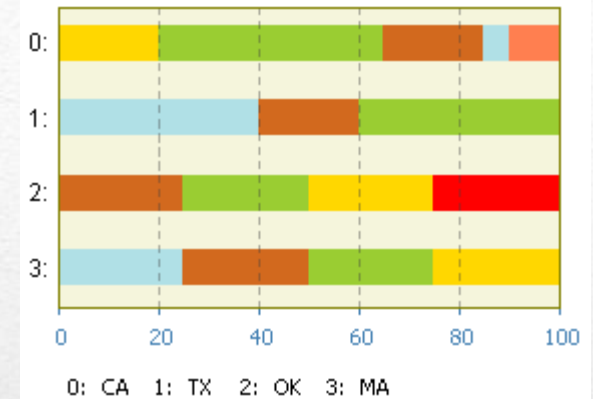
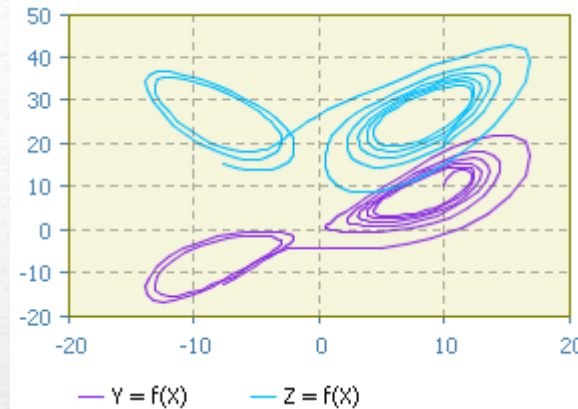
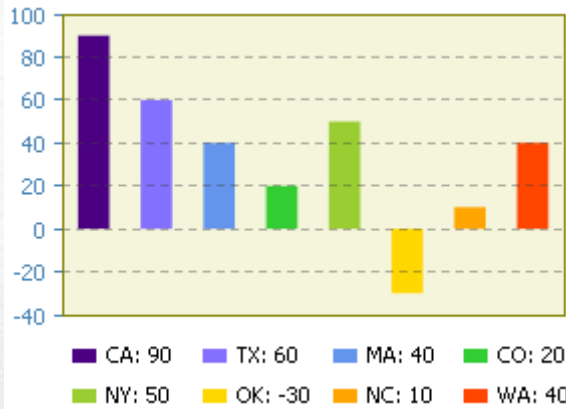


- Για την σωστή απεικόνιση του διαγράμματος κατά τη διάρκεια του τρεξίματος του μοντέλου θα πρέπει να κινήσουμε το μαύρο ανθρωπάκι (πράκτορα) στα δεξιά προκειμένου να δώσουμε τον χώρο που θα τρέχει το διάγραμμα.



# Τύποι Διαγραμμάτων

- Προκειμένου να επιλέξουμε τον τύπο του διαγράμματος που θα απεικονίζει τα δεδομένα μας, πρέπει να έχουμε μια γνώση σχετικά με τους τύπους των διαγραμμάτων που υπάρχουν στο Anylogic.
-





# **Διάδραση πρακτόρων σε Agent Based**

---

# Διάδραση Πρακτόρων

- Το AnyLogic υποστηρίζει ένα μηχανισμό επικοινωνίας μοναδικό για τη μοντελοποίηση που βασίζεται στο πέρασμα μηνυμάτων από πράκτορα σε πράκτορα.
  - Ο πράκτορας μπορεί να στείλει ένα μήνυμα σε έναν μεμονωμένο πράκτορα ή σε μια ομάδα πρακτόρων.
  - Ένα μήνυμα μπορεί να είναι ένα αντικείμενο οποιουδήποτε τύπου ή πολυπλοκότητας, συμπεριλαμβανομένης μιας συμβολοσειράς κειμένου, ακέραιο αριθμός, αναφορά σε αντικείμενο ή δομή με πολλαπλά πεδία.
-

- Για να στείλετε το μήνυμα σε έναν άλλο πράκτορα, χρησιμοποιούμε συγκεκριμένες εντολές για πράκτορες
  - Οι παρακάτω είναι οι πιο συχνά χρησιμοποιούμενες λειτουργίες για την αποστολή μηνυμάτων από έναν πράκτορα σε άλλο:
  - `sendToAll (msg)` - στέλνει το μήνυμα σε όλους τους παράγοντες του ίδιου πληθυσμού.
  - `sendToRandom (msg)` - στέλνει το μήνυμα σε έναν τυχαία επιλεγμένο πράκτορα από τον ίδιο πληθυσμό.
  - `send (msg, agent)` - στέλνει το μήνυμα στον συγκεκριμένο πράκτορα.
-



- Σε αυτή τη φάση, θα μοντελοποιήσουμε αυτό που λέγεται συχνά διαφήμιση από στόμα σε στόμα.
  - Πρόκειται για τον τρόπο που οι άνθρωποι πείθουν τους άλλους να αγοράσουν το προϊόν μας.
  - Στο μοντέλο μας, ένας καταναλωτής έρχεται σε επαφή κατά μέσο όρο με ένα άλλο άτομο κάθε μέρα.
-

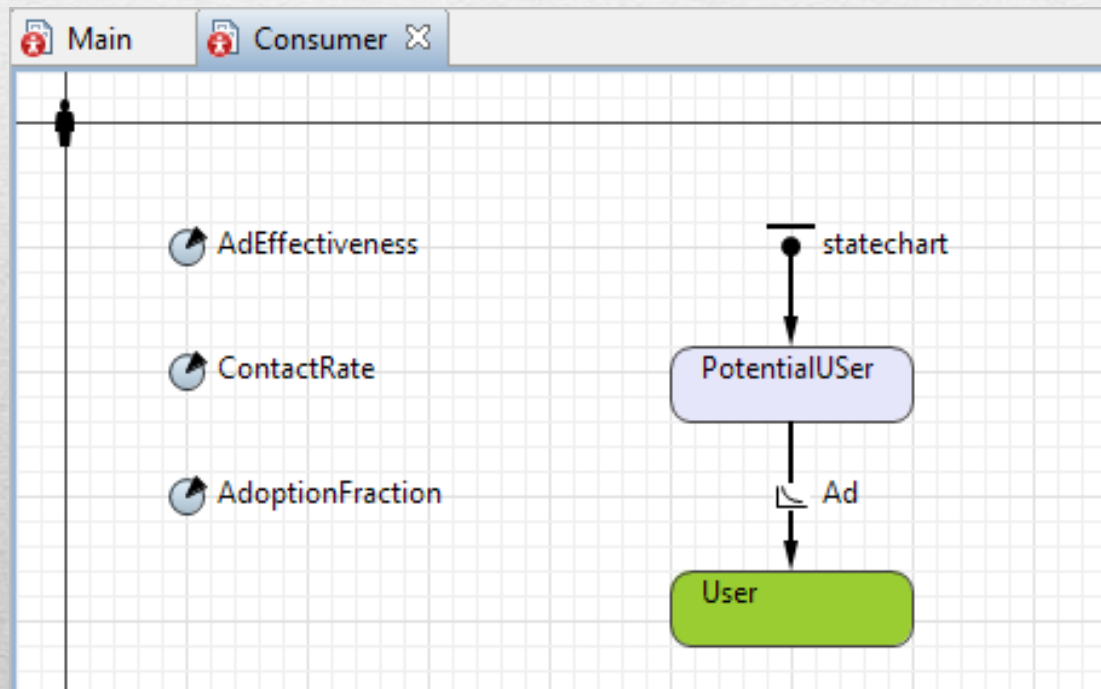
- Οι τρέχοντες χρήστες του προϊόντος μας ενδέχεται να επηρεάσουν τους πιθανούς χρήστες κατά τις συναντήσεις αυτές. Θα καθορίσουμε την πιθανότητα ενός δυνητικού χρήστη να αγοράσει το προϊόν ως  $\text{AdoptionFraction} = 0,01$ .
  - Ας αναπτύξουμε τη λογική του μοντέλου προσθέτοντας δύο παραμέτρους του καταναλωτή:  $\text{ContactRate}$  και  $\text{AdoptionFraction}$ .
-

- 1. Στο δέντρο Projects, ανοίξτε το διάγραμμα Consumer κάνοντας διπλό κλικ στην επιλογή Consumer.
  - Προσθέστε μια παράμετρο (parameter) για να ορίσετε τις μέσες ημερήσιες επαφές ενός καταναλωτή.
  - Σύρετε το Parameter από την Agent palette στο διάγραμμα.
  - Ορίστε την παράμετρο ContactRate.
  - Ο ρυθμός επαφής είναι 1 ανά ημέρα, οπότε πληκτρολογήστε 1 ως προεπιλεγμένη τιμή της παραμέτρου.
-



- Προσθέστε άλλη μια παράμετρο - AdoptionFraction - για να ορίσετε την επιρροή ενός ατόμου σε άλλους, έναν αριθμό που θα εκφράσουμε ως το ποσοστό των ανθρώπων που θα χρησιμοποιήσουν το προϊόν αφού έρχονται σε επαφή με τον καταναλωτή.
  - Αφήστε τον τύπο προεπιλεγμένης παραμέτρου: double και ορίστε την προεπιλεγμένη τιμή (Default value): 0,01
-

- Το νέο διάγραμμα του πράκτορα θα έχει την εξής μορφή:



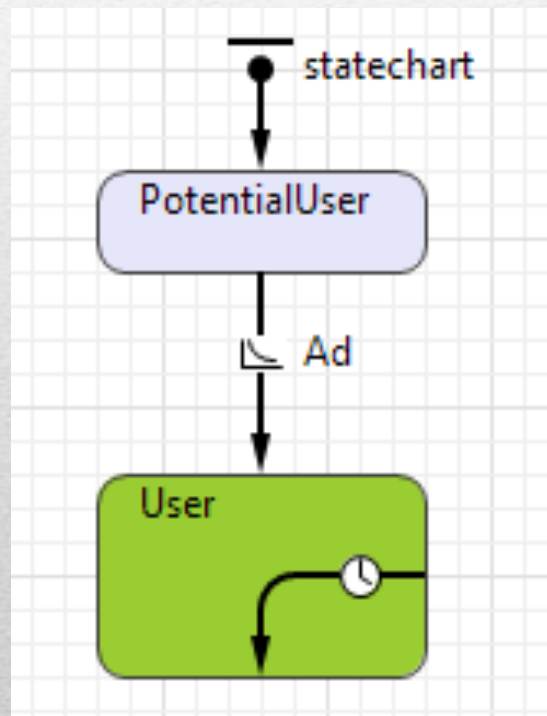
# **Μοντελοποίηση Εσωτερικών Μεταβάσεων**

---

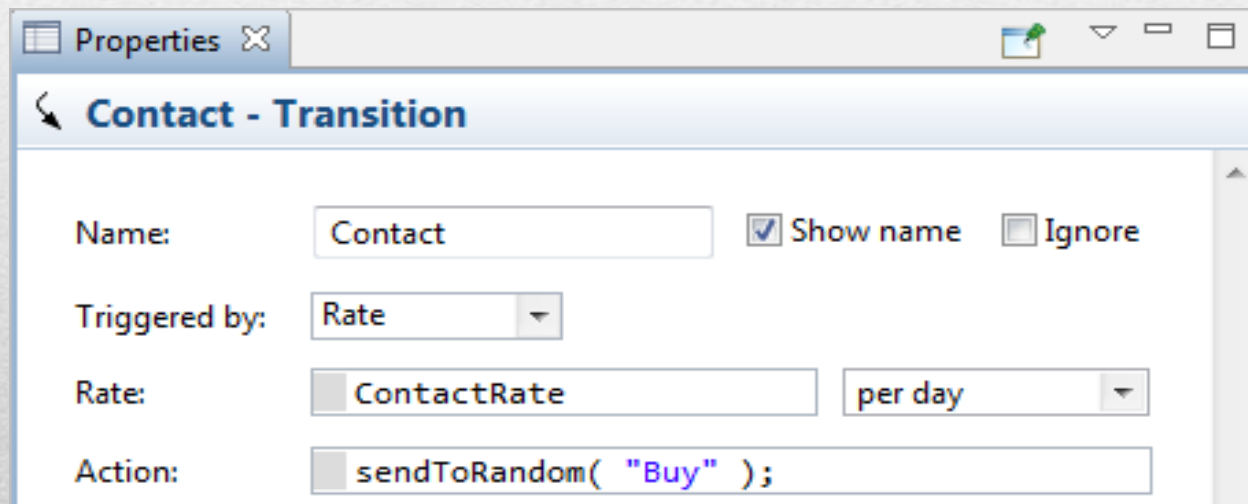


# Εσωτερικές Μεταβάσεις

- Ανοίξτε το διάγραμμα Consumer και μεγαλώστε την κατάσταση User για να σχεδιαστεί μια εσωτερική μετάβαση
  - Σχεδιάστε μια εσωτερική μετάβαση μέσα στην κατάσταση User.
  - Για να σχεδιάσετε μια μετάβαση όπως αυτή που παρουσιάζεται στο επόμενο σχήμα, σύρετε το Transition από την παλέτα Statechart εντός της κατάστασης έτσι ώστε το σημείο έναρξης της μετάβασης να βρίσκεται στα σύνορα.
  - Στη συνέχεια, μπορείτε να μετακινήσετε το τελικό σημείο μετάβασης σε άλλο σημείο στα σύνορα.
-



- Τροποποιήστε τις ιδιότητες μετάβασης.
- Αυτή η μετάβαση θα πραγματοποιηθεί με την καθορισμένη τιμή Rate ContactRate.



The screenshot shows a 'Properties' window for a 'Contact - Transition'. The window contains the following fields and options:

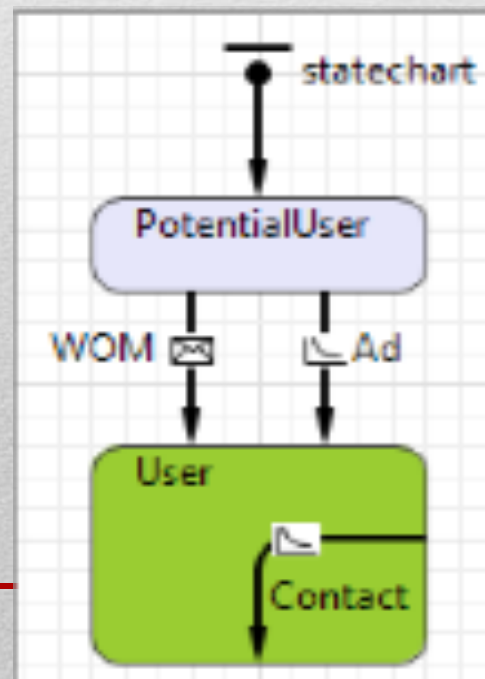
- Name:** A text box containing 'Contact'. To its right are two checkboxes:  Show name and  Ignore.
- Triggered by:** A dropdown menu showing 'Rate'.
- Rate:** A text box containing 'ContactRate' and a dropdown menu showing 'per day'.
- Action:** A text box containing the code `sendToRandom( "Buy" );`.

- Καθορίστε τη Δράση (Action) που θα εκτελεστεί κατά την ενεργοποίηση αυτής της μετάβασης `sendToRandom( "Buy" );`
-



# Εξωτερικές Διαδράσεις

- Δημιουργούμε μια ακόμη transition που συνδέει το PotentialUser με το User.
- Την ονομάζουμε WOM. Αυτή η μετάβαση μοντελοποιεί την transition που σπρώχνει τον πιθανό καταναλωτή να μεταβεί στην κατάσταση του χρήστη με την διαδικασία του word of mouth.



# Τροποποίηση των ιδιοτήτων του Transition

Για την τροποποίηση των ιδιοτήτων του καινούργιου transition ακολουθούμε τα εξής:

- Στην **Triggered by** λίστας, κάνουμε κλίκ στο **Message**.
  - Στην περιοχή **Fire transition**, επιλέγουμε **On particular message**.
  - Στο πεδίο **Message**, γράφουμε "Buy"
-

- δεδομένου ότι γνωρίζουμε ότι όλες οι επαφές που πραγματοποιούνται μεταξύ πρακτόρων δεν είναι επιτυχείς - με άλλα λόγια, μια επαφή μπορεί να μην πείσει τον πιθανό χρήστη να αγοράσει το προϊόν μας - θα χρησιμοποιήσουμε το `AdoptionFraction` για να κάνουμε τις επιτυχείς επαφές λιγότερο συνηθισμένες.
  - Καθορίζουμε την μετάβαση Guard: `randomTrue` (`AdoptionFraction`)
-



Properties ✕ 📌 ▾ ☰ □

### WOM - Transition

Name:   Show name  Ignore

Triggered by:  ▾

Message type:  ▾

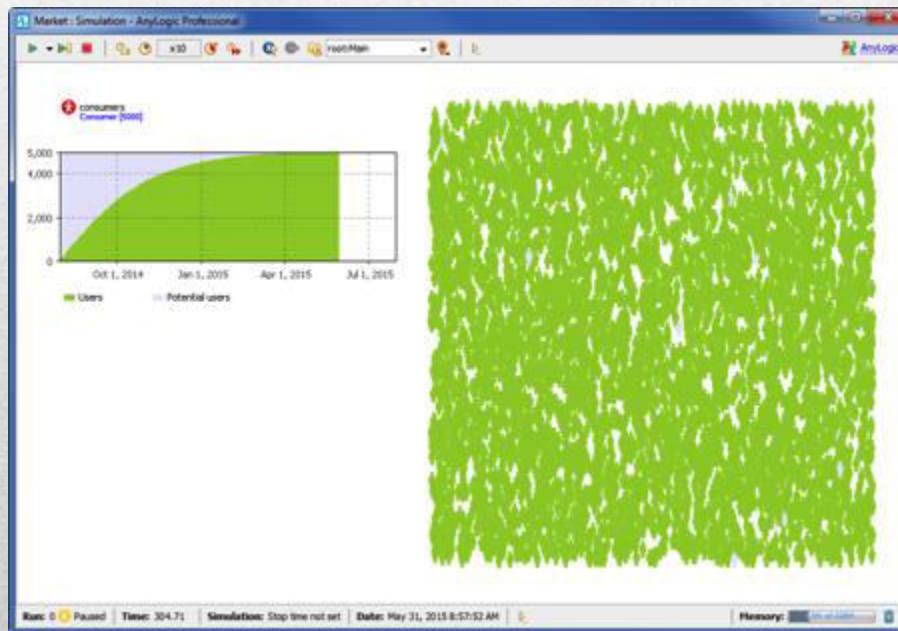
Fire transition:  Unconditionally  
 On particular message  
 If expression is true

Message:

Action:

Guard:

- Αφού αποθηκεύσουμε τρέχουμε το μοντέλο



- Παρατηρήστε ότι η αύξηση των χρηστών είναι γρηγορότερη μιας και μαζί με την διαφήμιση Ad αυξάνονται και με το WoM



# **Μοντελοποίηση κύκλου ζωής**

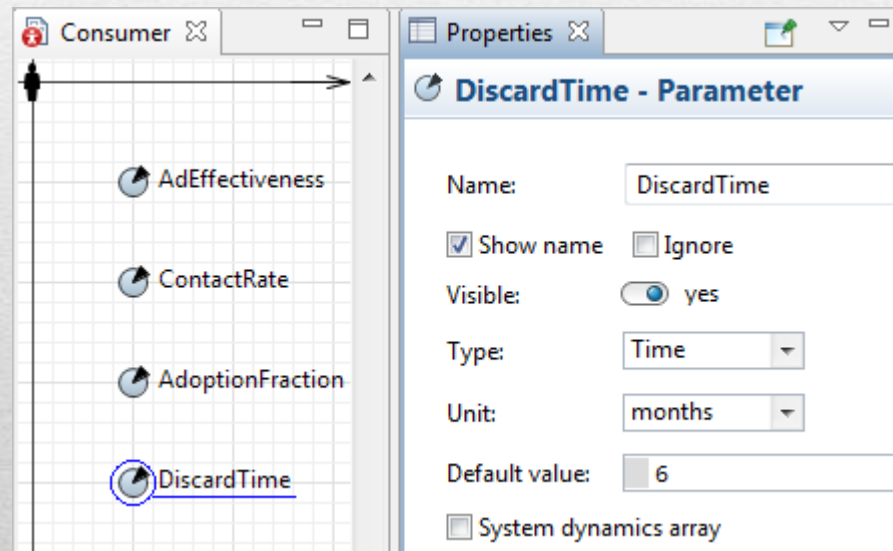
---



- Σε αυτή τη φάση, θα μοντελοποιήσουμε την απόρριψη των προϊόντων.
  - Ας υποθέσουμε ότι η μέση διάρκεια της ενεργής χρήσης του προϊόντος μας είναι έξι μήνες.
  - Μόλις ο χρήστης απορρίψει ή καταναλώσει το προϊόν, θα χρειαστεί αντικατάσταση.
  - Σκοπός είναι να διαμορφώσουμε τη συμπεριφορά μιας επαναλαμβανόμενης αγοράς,
  - Ως υπόθεση του μοντέλου έχουμε ότι οι τελικοί χρήστες θα γίνουν δυνητικοί χρήστες όταν απορρίψουν ή καταναλώσουν το προϊόν (με άλλα λόγια, όταν ο User επανέλθει στην κατάσταση του PotentialUser).
-

- Ανοίγουμε το διάγραμμα Consumer και προσθέτουμε την παράμετρο DiscartTime
-

- Ανοίγουμε το διάγραμμα Consumer και προσθέτουμε την παράμετρο DiscardTime



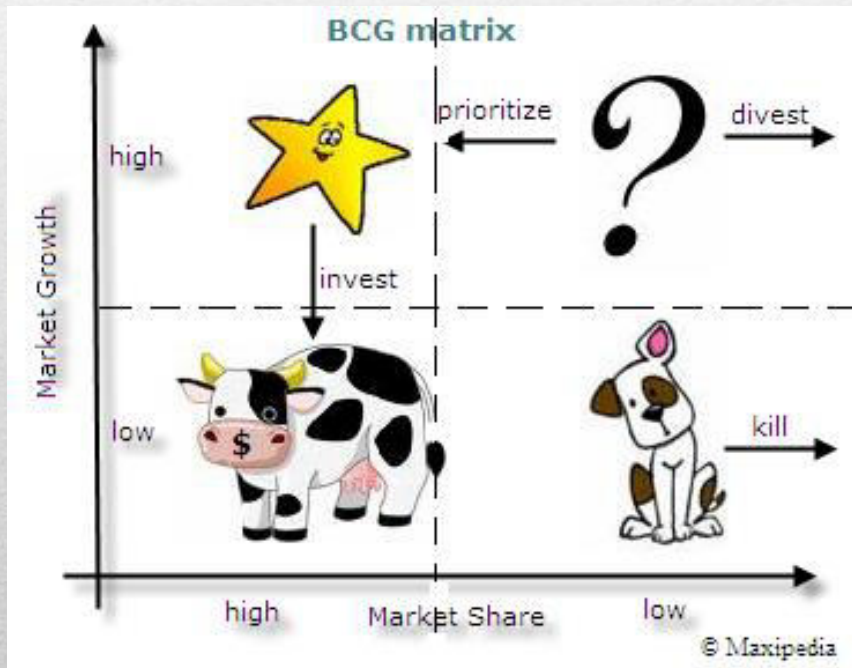
- Αυτή η παράμετρος θα καθορίσει τη διάρκεια ζωής του προϊόντος μας.
-



- Τα στοιχεία της παραμέτρου DiscardTime
  - Type : Time
  - Unit : months
  - Default Value : 6
  - Το Unit και το Default Value ποικιλομορφεί ανάλογα το προϊόν ή την υπηρεσία πληροφορικής που μοντελοποιείτε
-

# Product Lifespan

- Boston Consulting Group Matrix



- Αγελάδα
- Το προϊόν είναι mainstream και πολλοί άνθρωποι το αγοράζουν, «αρμέγουμε την αγελάδα» και υπάρχει μεγάλο οικονομικό κέρδος
- Αστέρι
- Τώρα οι άνθρωποι γνωρίζουν το προϊόν και θα ήθελαν να έχουν ένα - ίσως είναι λίγο ακριβό.

- Τον άξονα Y, που αντιπροσωπεύει την ανάπτυξη της αγοράς (market growth). Πρόκειται για το ρυθμό ανάπτυξης από ένα έτος στο επόμενο εκφρασμένο ως ποσοστό.

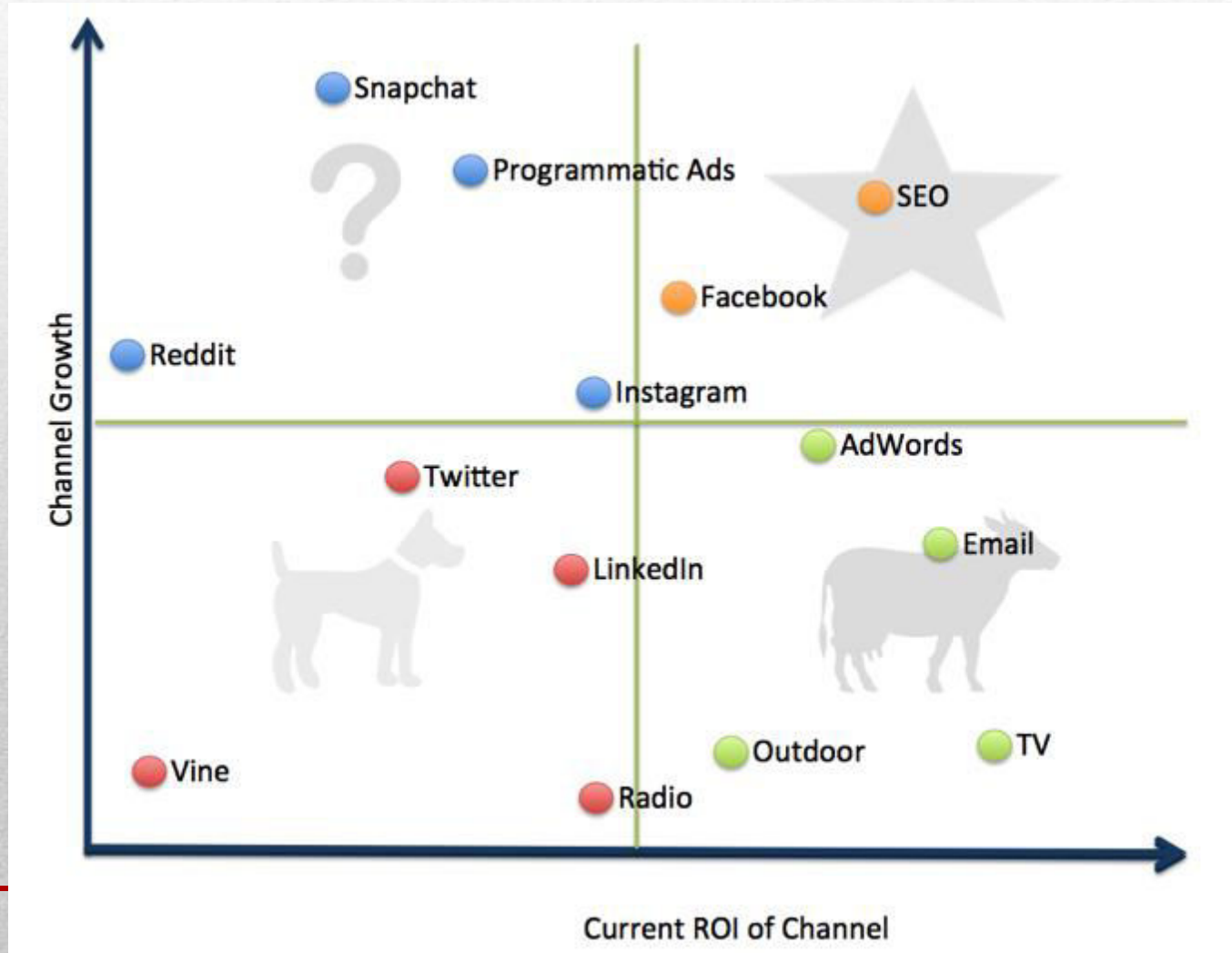
$$r\% = \left\{ \frac{\text{ολική αγορά έτους } X}{\text{ολική αγορά έτους } X-1} \right\}$$

- Τον άξονα X που αντιπροσωπεύει το σχετικό μερίδιο αγοράς (relative market share), δηλαδή τον λόγο του μεριδίου αγοράς του υπό εξέταση προϊόντος προς το μερίδιο της αγοράς του ισχυρότερου ανταγωνιστή. (Το μερίδιο αγοράς μετριέται σε ποσότητα και όχι σε αξία).

$$s = \left\{ \frac{\text{μερίδιο αγοράς προϊόντος}}{\text{μερίδιο αγοράς προϊόντος ηγέτη}} \right\}$$



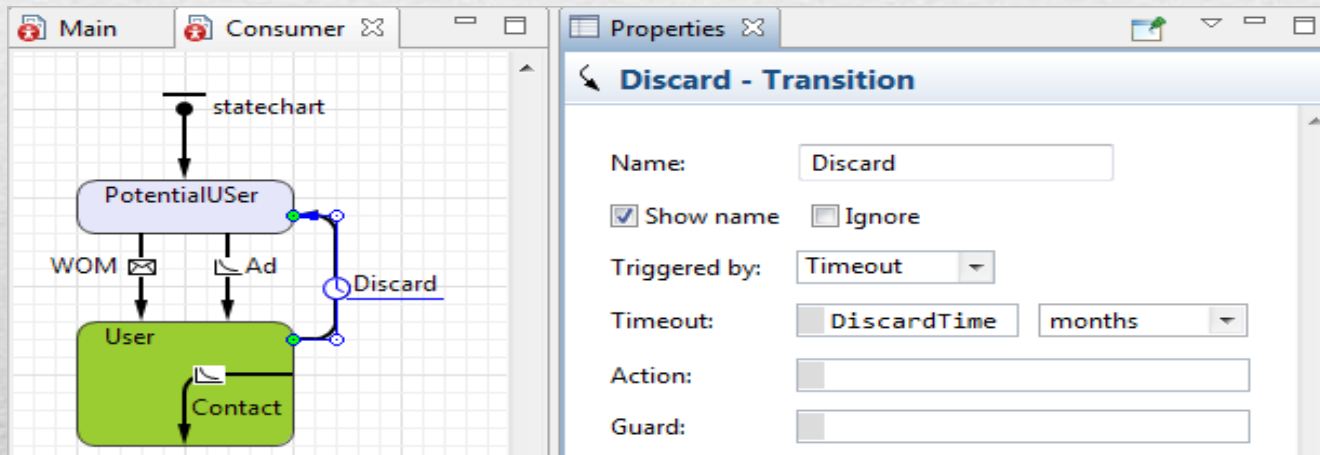
# Informatics Product Lifespan



# Παραδείγματα κύκλου ζωής

- Laptops 5.5
  - Tablets 5.1
  - Smartphone 4.6
  - Game Console 5.7
-

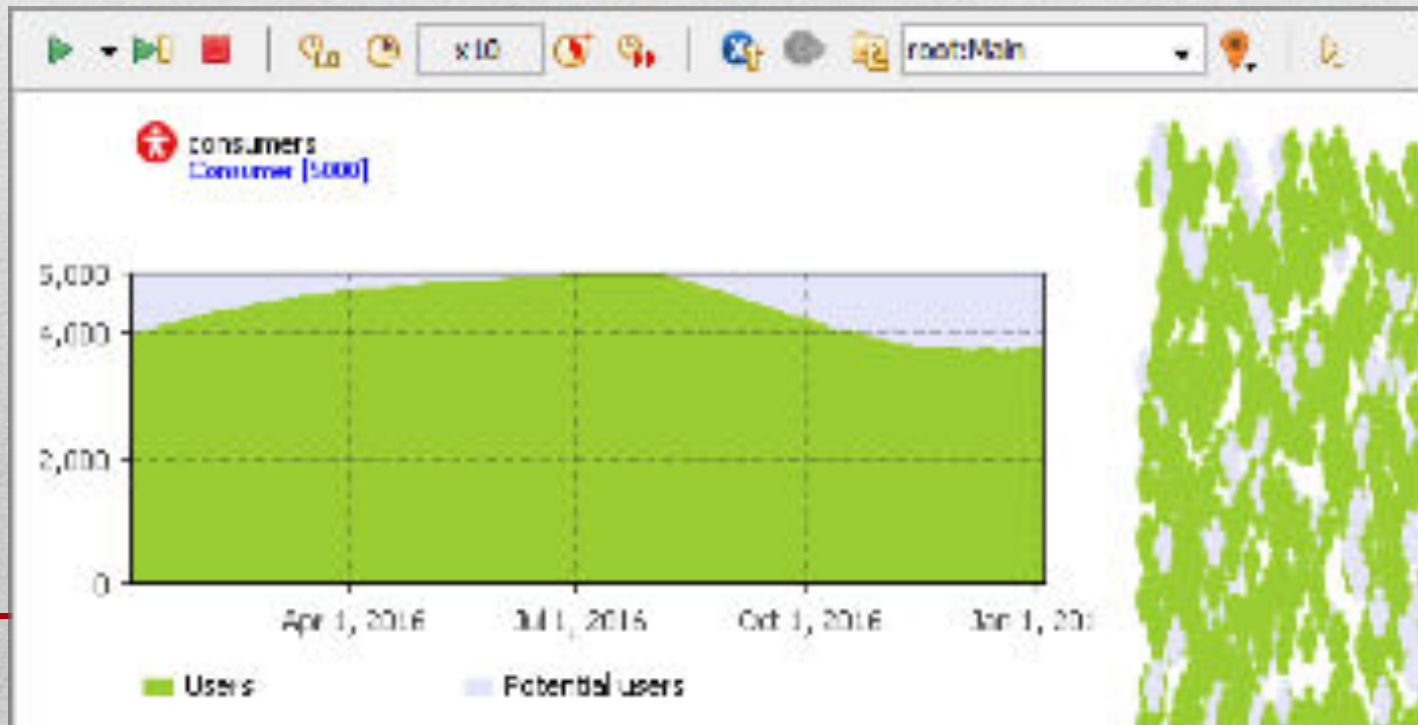
- Σχεδιασμός μιας διαδικασίας Transition στο Statechart από τον User στον PotentialUser. Προσοχή στην φορά του βέλους της Transition.



- Στοιχεία της Transition
- Name : Discard
- Triggered by: Timeout
- Timeout : DiscardTime και months



- Τρέξτε το μοντέλο και δείτε πώς οι απορρίψεις επηρεάζουν τη δυναμική υιοθέτησης του προϊόντος.
- Ακόμη και όταν το προϊόν μας πλημμυρίσει την αγορά, θα παρατηρήσετε περιστασιακές απορρίψεις προϊόντων.



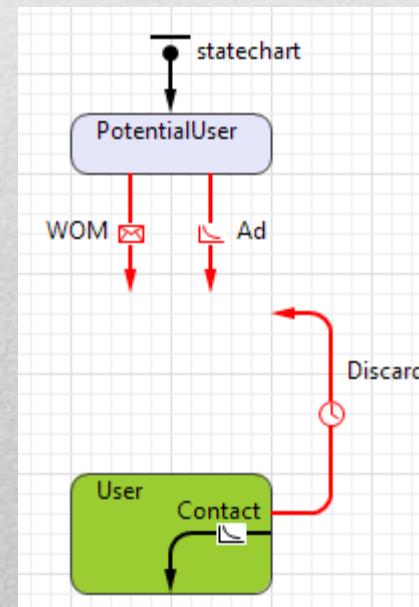
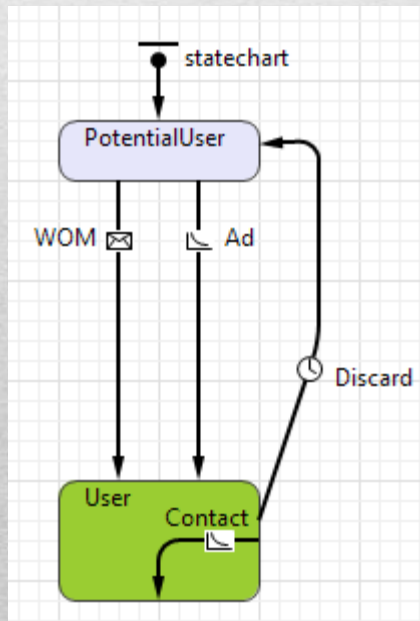
**Δημιουργία  
Ενδιάμεσης  
κατάστασης στα  
Agent Based**

---

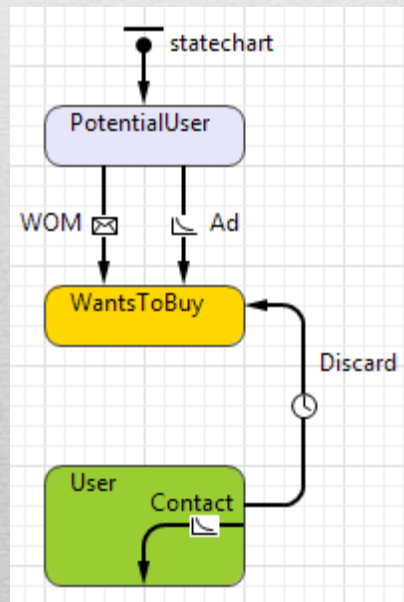
- Το μοντέλο μας υποθέτει ότι το προϊόν είναι πάντα διαθέσιμο και ότι η μετάβαση από το PotentialUser στο User είναι άνευ όρων και άμεση.
  - Στο σύνολο των μοντέλων όμως δεν είναι αυτή η πραγματικότητα.
  - Τώρα, θα βελτιώσουμε το μοντέλο προσθέτοντας μια κατάσταση στο statechart που αντικατοπτρίζει το χρονικό διάστημα μεταξύ της απόφασης ενός πράκτορα για την αγορά του προϊόντος και του χρόνου που το λαμβάνει.
-



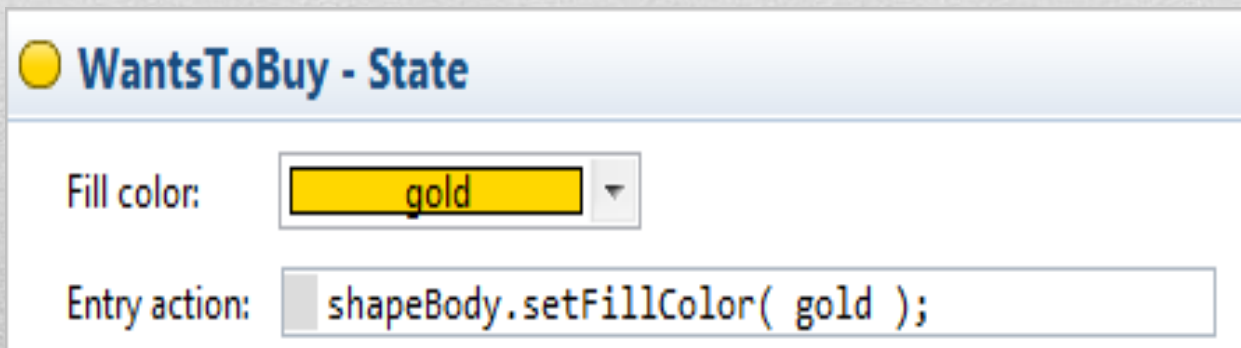
- Πρώτο βήμα είναι η αποσύνδεση όλων των transitions.



- Προσθέτουμε μια ακόμη κατάσταση (State) από την παλέτα Statechart την οποία ονομάζουμε WantsToBuy.
- Συνδέουμε όλα τα transition



- αποδίδουμε τις πρώτες ιδιότητες στο State : WantsToBuy για το χρώμα αυτής της κατάστασης.
- Fill color: gold
- Entry action: `shapeBody.setFillColor(gold)`

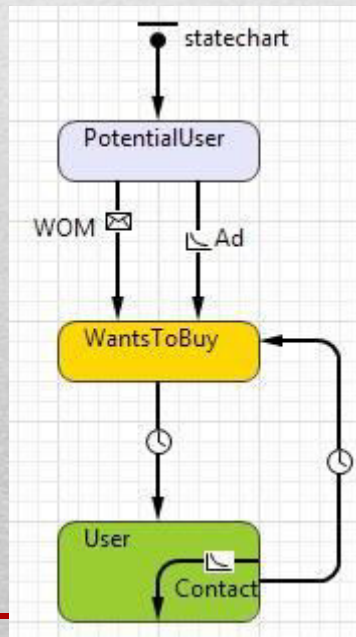


The image shows a configuration window for a state named "WantsToBuy - State". The window has a title bar with a yellow circle icon and the text "WantsToBuy - State". Below the title bar, there are two fields:

- A "Fill color:" field with a dropdown menu showing "gold" and a yellow background.
- An "Entry action:" field with a text input containing the code `shapeBody.setFillColor( gold );`

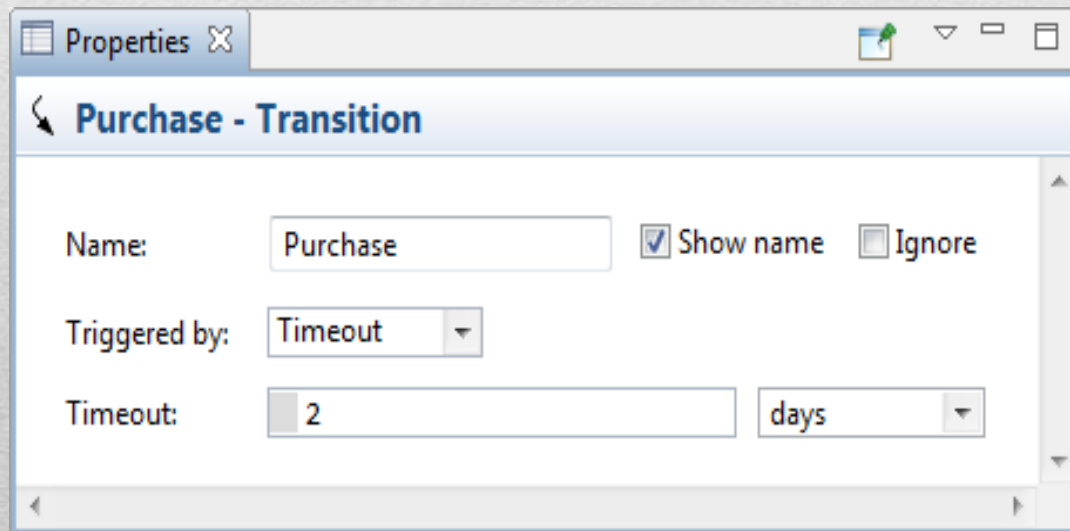


- Προσθέτουμε μια transition από το WantsToBuy στο User state προσπαθώντας να απεικονίσουμε τον χρόνο που κάνει να παραδωθεί το προϊόν.
- Το ονομάζουμε Purchase.



- Ας υποθέσουμε ότι ένας τυπικός χρόνος παράδοσης του προϊόντος είναι δύο μέρες.
  - Αυτό σημαίνει ότι ο πελάτης μπαίνει σε κατάσταση WantsToBuy και θα περάσει στην κατάσταση User με δύο μέρες καθυστέρηση.
-

- Με το δεδομένο αυτό κατά νου θέτουμε στις ιδιότητες της transition Purchase τα εξής δεδομένα 2 days στο timeout και στο Triggered by: επιλέγουμε Timeout.





- Προκειμένου να απεικονιστούν τα προαναφερόμενα ορίζουμε μια ακόμη στατιστική συνάρτηση
- Αυτή η στατιστική συνάρτηση μετράει το product's market-driven demand.
- Στον editor του Main, κάνουμε click στο consumers, και ανοίγουμε το Statistics properties section, και προσθέτουμε το στατιστικό αντικείμενο : NWantToBuy με συνθήκη `item.inState(Consumer.WantsToBuy)`

▼ Statistics

Name:

Type:  Count  Sum  Average  Min  Max

Condition:

Name:

Type:  Count  Sum  Average  Min  Max

Condition:

Name:

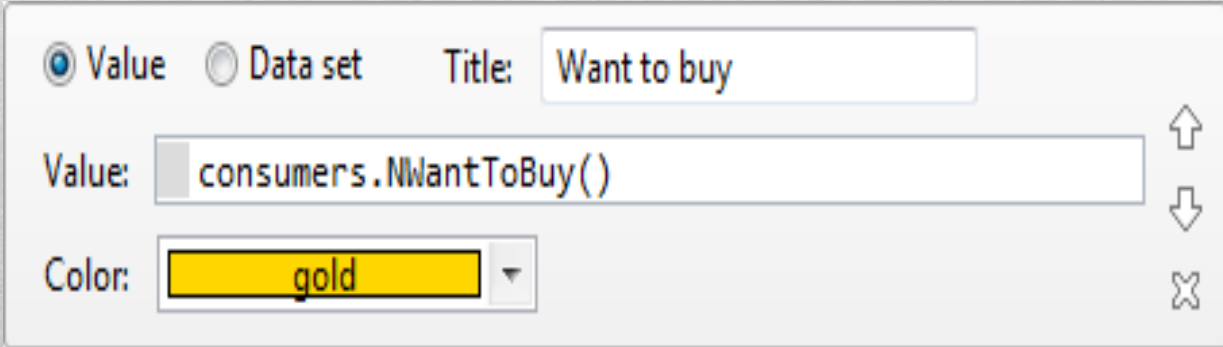
Type:  Count  Sum  Average  Min  Max

Condition:


+ [ ] [X] [↑] [↓]

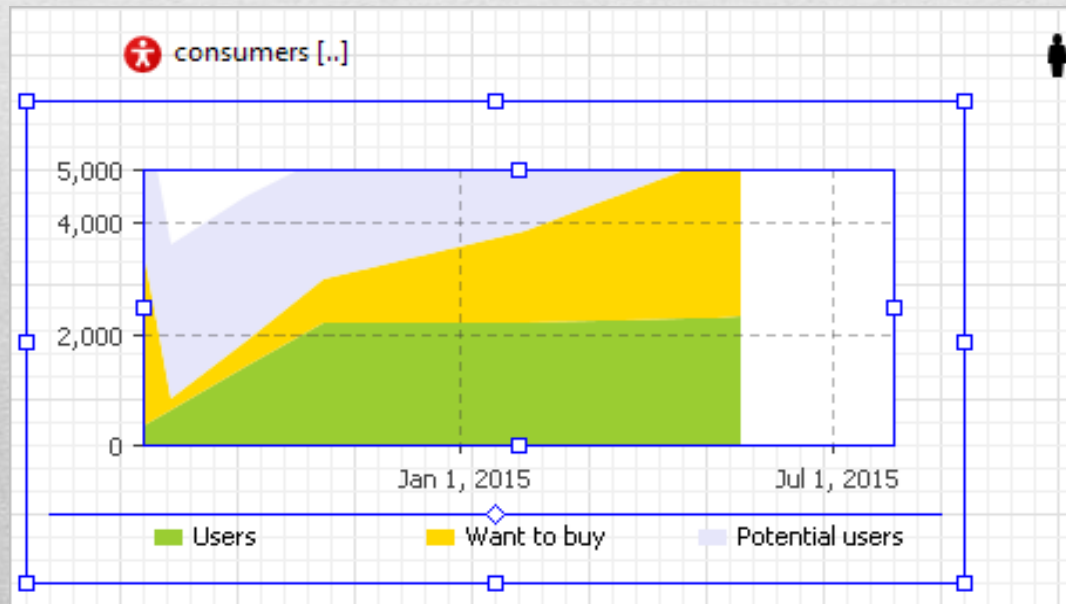
# Απεικόνιση των δεδομένων στο διάγραμμα

- Στο Main, επιλέγουμε το διάγραμμα που έχουμε εισάγει (time stack chart), και προσθέτουμε ένα ακόμη δεδομένο που θέλουμε να απεικονίζεται
- Με δεδομένα
- Title : Want to Buy
- Value : consumers.NWantToBuy()
- color : gold



The image shows a configuration dialog for a chart series. It has three radio buttons: 'Value' (selected), 'Data set', and 'Title'. The 'Title' field contains 'Want to buy'. The 'Value' field contains the expression 'consumers.NWantToBuy()'. The 'Color' field is a dropdown menu with 'gold' selected. On the right side, there are three vertical icons: an upward arrow, a downward arrow, and a close button (X).

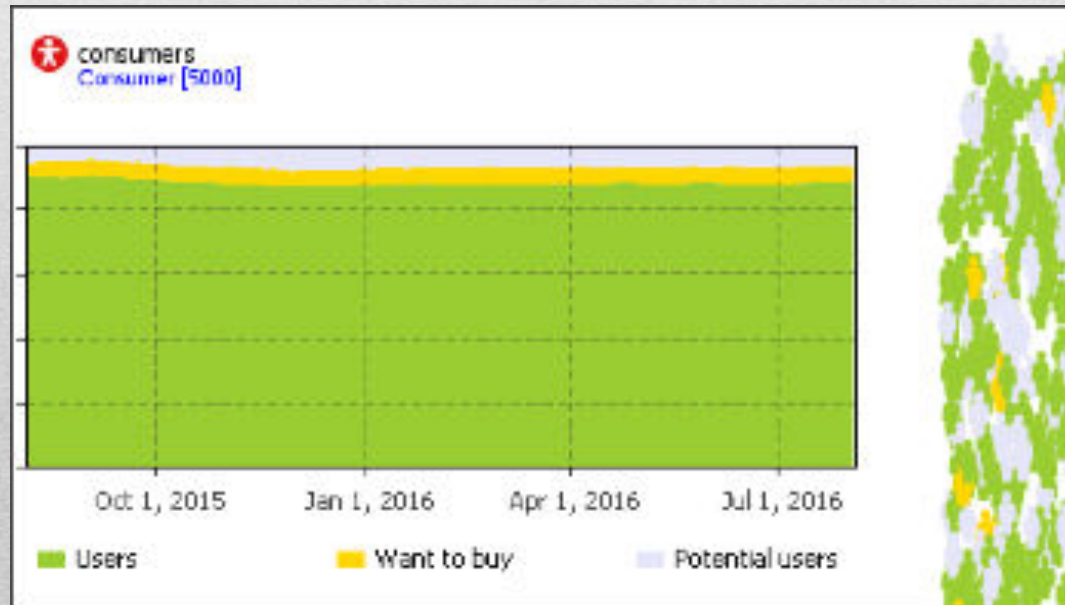
- Προκειμένου το διάγραμμα να αποκτήσει την ίδια σειρά με την μορφή που έχει το διάγραμμα ροής των καταστάσεων ανεβάζουμε την κατάσταση Want to buy στην δεύτερη θέση με το κουμπί 





# Τρέξιμο του Μοντέλου

- Εκτελέστε το μοντέλο και θα παρατηρήσετε ότι το AnyLogic εμφανίζει τον αριθμό των καταναλωτών που περιμένουν το προϊόν με κίτρινο χρώμα.



# **Μοντελοποίηση και κατανομές Πιθανοτήτων**

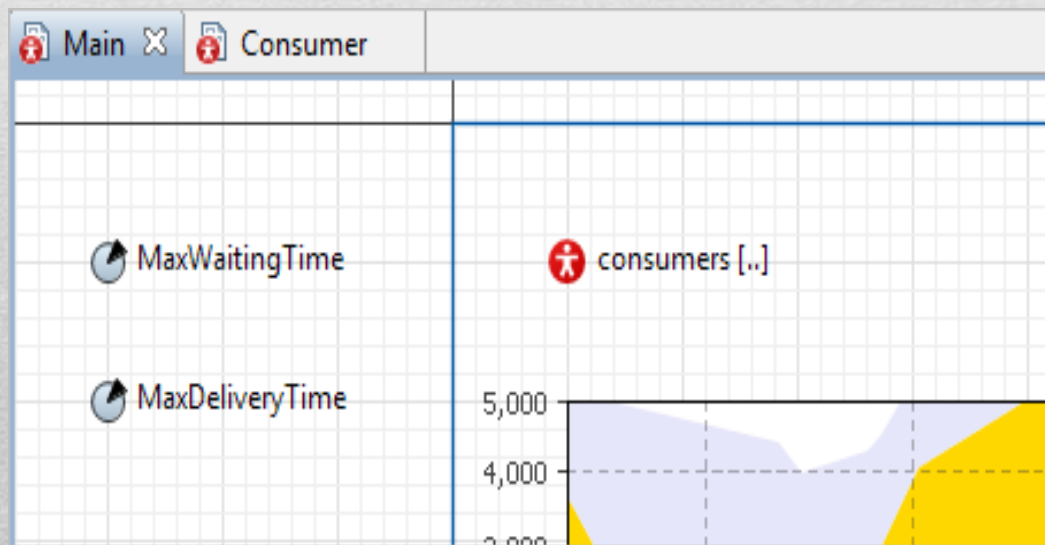
---

- Το μοντέλο μας πρέπει να αντιμετωπίσει την ποικιλομορφία του χρονικού διαστήματος που οι καταναλωτές είναι πρόθυμοι να περιμένουν την παράδοση του προϊόντος τους.
  - Εάν ο χρόνος παράδοσης υπερβαίνει τον χρόνο που ο καταναλωτής είναι πρόθυμος να περιμένει, ο καταναλωτής θα επανεξετάσει την απόφασή του και θα επιστρέψει σε δυνητικό χρήστη παρά σε κάποιον που θέλει να αγοράσει.
-





- Ξεκινάμε καθορίζοντας δύο καινούργιες παραμέτρους parameters στο Main:
  - Την μέγιστη τιμή που μπορεί να παραδοθεί το προϊόν
    - maximum product delivery time (25 days)
  - Και το μέγιστο χρονικό διάστημα που μπορεί να περιμένει ο καταναλωτής
  - maximum consumer's waiting time (7 days).
-

- Επειδή δεν θέλουμε να φαίνονται μέσα στο παράθυρο μπορούμε να σύρουμε τις δύο αυτές παραμέτρους έξω από το παράθυρο του μοντέλου.



- Στο επόμενο βήμα καθορίζουμε τις ιδιότητες των παραμέτρων.

Properties 


 **MaxWaitingTime - Parameter**


Name:

Visible:  yes

Type:  ▾

Default value:

Properties 

 **MaxDeliveryTime - Parameter**

Name:

Visible:  yes

Type:  ▾

Default value:



# Συναρτήσεις κατανομής πιθανοτήτων

- Σε περίπτωση που δεν θέλουμε η παράμετρος μας να έχει μια σταθερή τιμή.
  - Υποθέτουμε ότι διαρκεί η παράδοση του προϊόντος από μία έως 25 ημέρες - με μέσο όρο δύο ημερών
  - Έχοντας αυτό υπόψη, ας αλλάξουμε τον χρόνο παράδοσης από μια σταθερή περίοδο παράδοσης δύο ημερών στην στοχαστική έκφραση που περιγράφει αυτό το μοτίβο.
-

- Ανοίξτε το διάγραμμα Consumer και επιλέξτε την transition Purchase.
  - Θέλουμε να αλλάξουμε την έκφραση της χρονικής υπέρβασης της μετάβασης
  - και θα το κάνουμε χρησιμοποιώντας έναν οδηγό για να επιλέξετε τη συνάρτηση κατανομής
  - και να εισαγάγετε το όνομα της λειτουργίας στις ιδιότητες.
  - Για να αντικαταστήσετε την υπάρχουσα τιμή, χρησιμοποιήστε το ποντίκι για να επιλέξετε την υπάρχουσα έκφραση Timeout.
-

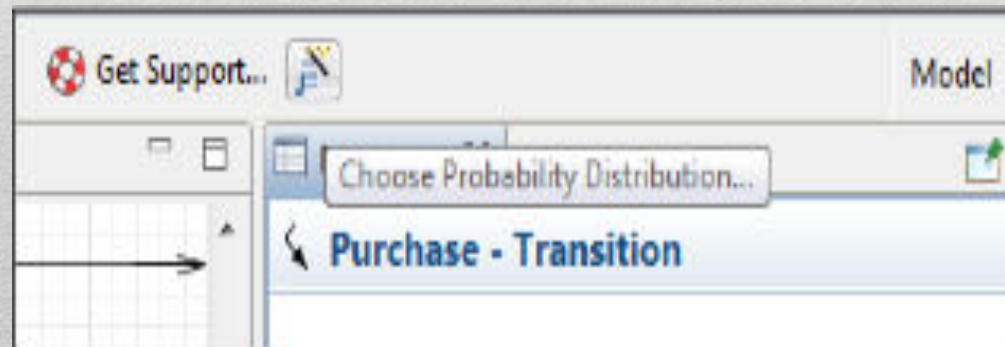
**Purchase - Transition**

Name:

Show name  Ignore

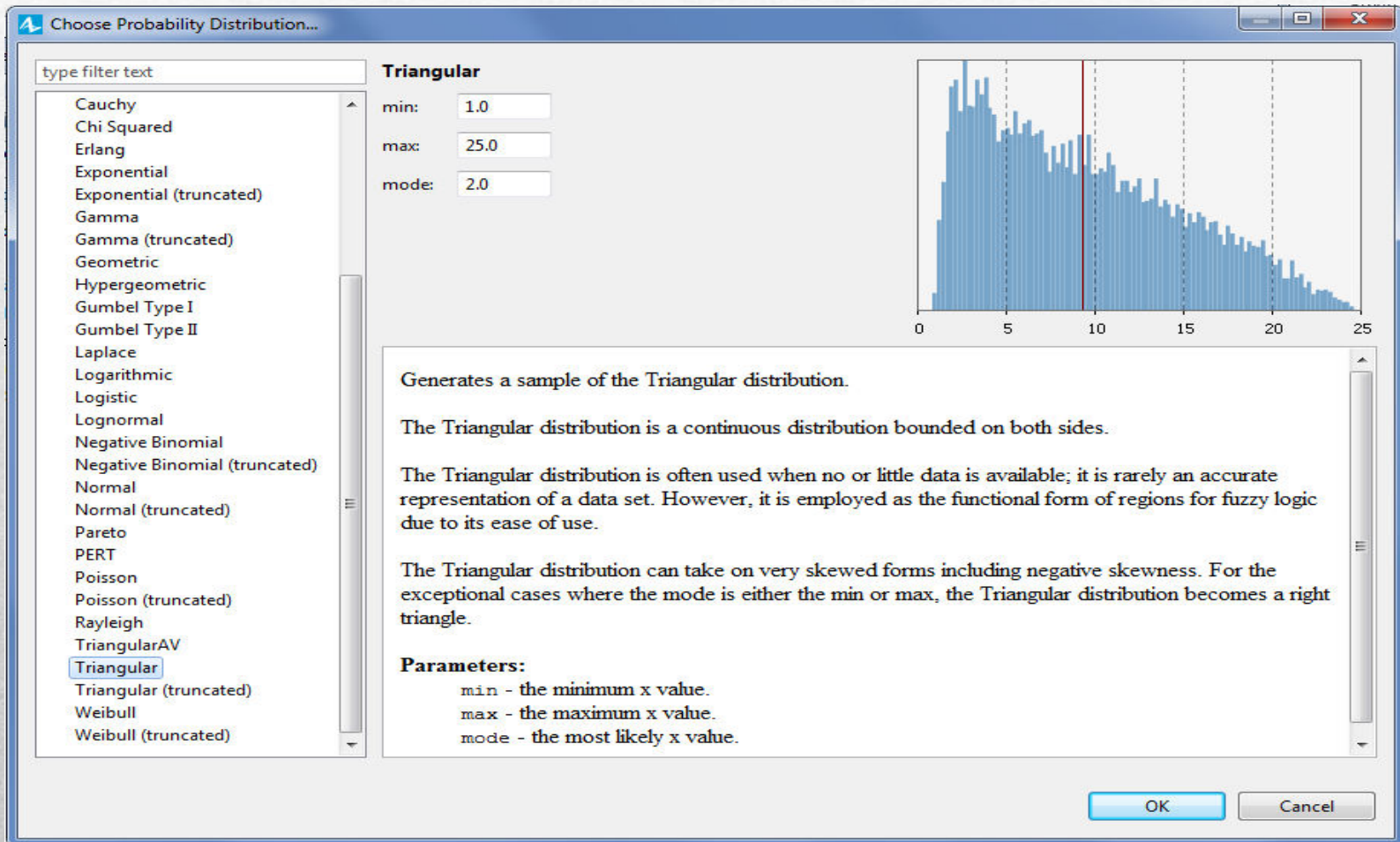
Triggered by:  ▼

Timeout:   ▼



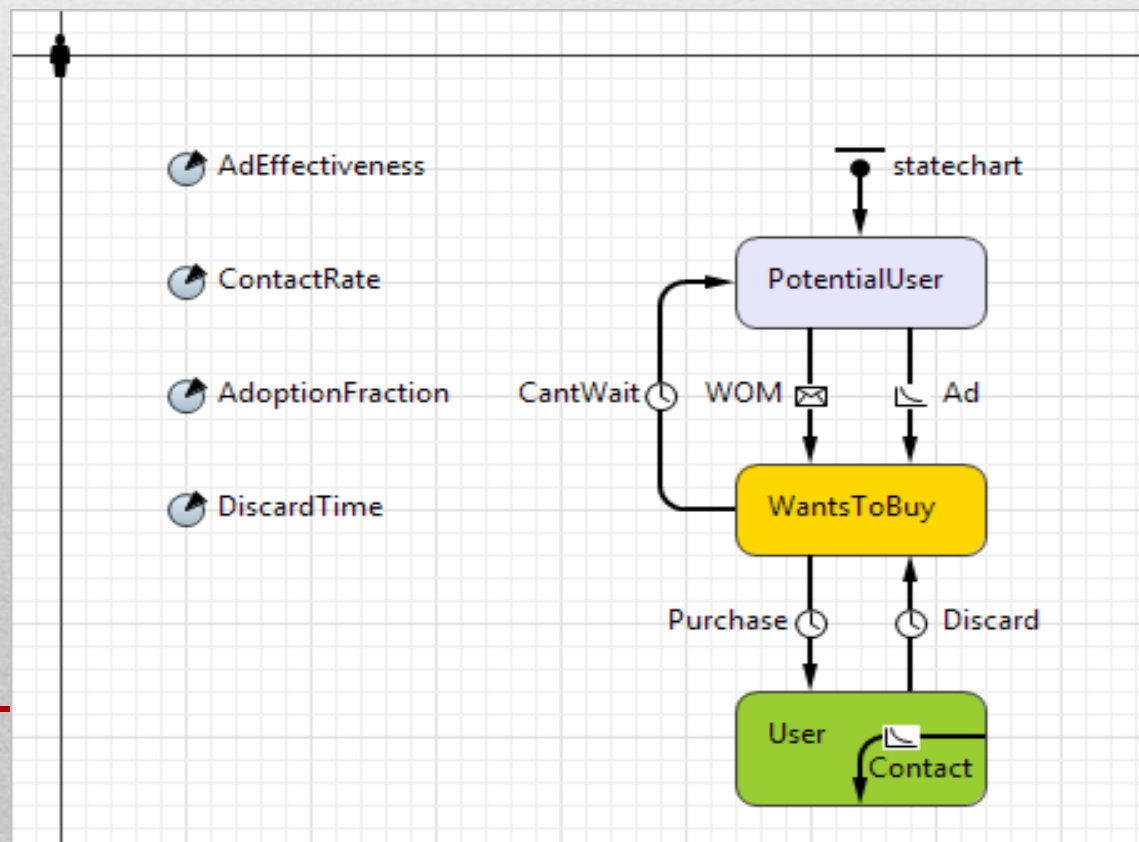


- Στο παράθυρο Choose Probability Description μας επιτρέπεται να δούμε και να επιλέξουμε από μια μεγάλη γκάμα συναρτήσεων πιθανοτήτων.



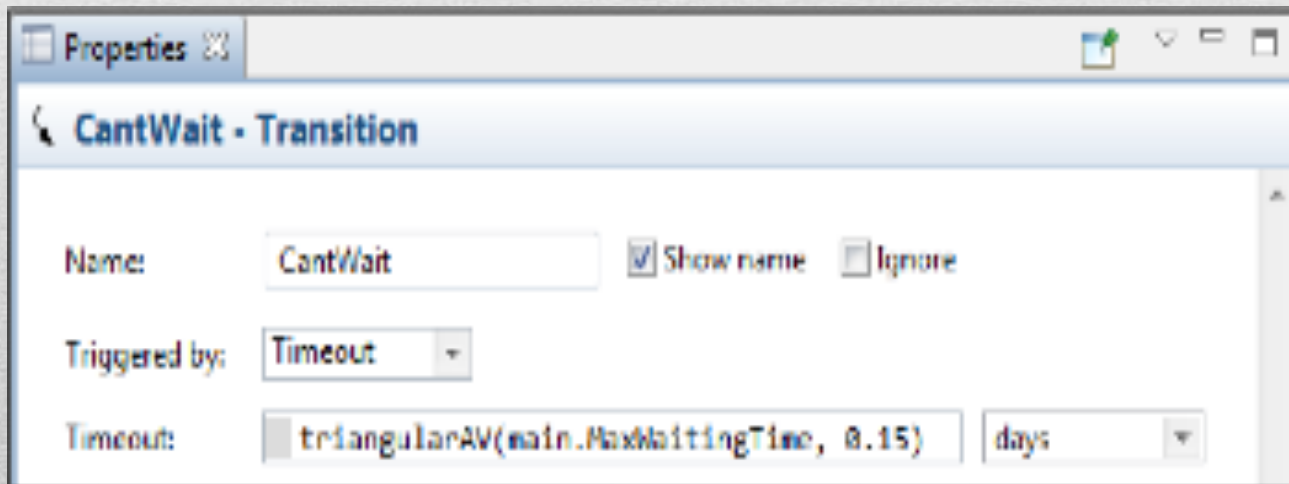
- Για την δική μας περίπτωση επιλέγουμε την τριγωνομετρική συνάρτηση με τιμές
  - `triangular(1, 25, 2)`
  
  - Την μετατρέπουμε δίνοντας την τιμή που έχουμε προκαθορίσει
  - `triangular(1, main.MaxDeliveryTime, 2)`
-

- Σχεδιάστε την τελευταία μετάβαση (transition) με όνομα CantWait που πηγαίνει από το WantsToBuy στην κατάσταση PotentialUser.
- Αυτή η μετάβαση θα διαμορφώσει τον τρόπο με τον οποίο η ανυπομονησία του καταναλωτή τους αναγκάζει να αλλάξουν την απόφασή τους για την αγορά του συγκεκριμένου προϊόντος.





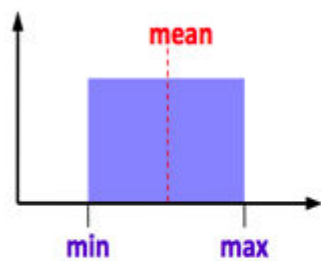
- Και της αποδίδω της εξής ιδιότητες
- Timeout η οποία ισούται με `triangularAV(main.MaxWaitingTime, 0.15) days`



## Probability distribution

## Primary use

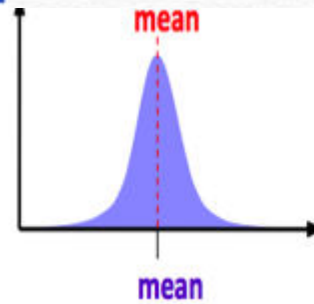
## Uniform



`uniform( min, max )`

You know the minimum and the maximum values but lack any knowledge about how the remaining values are distributed between them. In other words, you don't know if any values are more frequent than others and assume any location between min and max has the same chance of receiving a value.

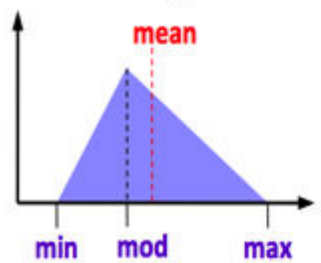
## mean



`normal( sigma, mean )`

Note that the normal distribution is unbounded on both sides, so if you wish to impose limits (e.g. to avoid negative values) you have to use its truncated form or use other distributions such as Lognormal, Weibull, Gamma, or Beta.

## Triangular

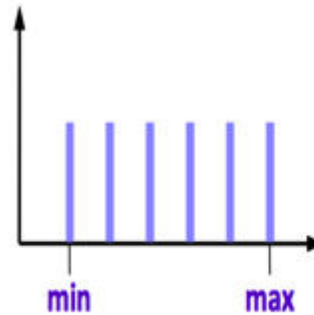


`triangular( min, mode, max )`

You know the minimum and the maximum, and you have a guess about the most likely (modal) value.

A triangular distribution is often used for service times or the duration of operations where you don't have enough samples to build a meaningful distribution shape.

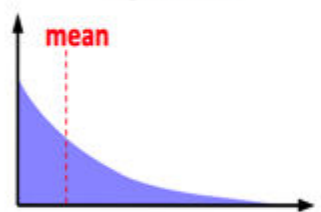
## Discrete uniform



Used to model a finite number of outcomes that are equally probable, or when you have no knowledge about which outcomes are more likely to occur.

Note that both the minimum and maximum values are included in the set of possible results, so a call of `uniform_discr( 3, 7 )` may return 3, 4, 5, 6, or 7. (Borshchev, 2013)

## Exponential



`exponential( lambda, min )`

Describes the times between events in a Poisson process, i.e. when events occur independently at a constant average rate.

Used as the inter-arrival time for input streams of customers, parts, calls, orders, transactions or failures in process models.

In agent based models, an exponential distribution is used as timeout for rate transitions that model independent events in agents that are known to occur at a certain global average rate.



# Πολυπλοκότητα Agent Based

- Το σύνολο των προηγούμενων διαφανειών αναλύουν τις δυνατότητες μοντελοποίησης των Agent Based συστημάτων
- Η πολυπλοκότητα δημιουργίας του μοντέλου έγκειται στην πολυπλοκότητα του προβλήματος και κατά συνέπεια στην μοντελοποίηση του συνόλου των παραγόντων που λαμβάνουν μέρος.

