



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ, ΟΙΚΟΝΟΜΙΑΣ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΠΜΣ: «ΕΠΙΣΤΗΜΗ ΥΠΟΛΟΓΙΣΤΩΝ»

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΘΕΜΑ: “Display Advertising - Η Αποτελεσματικότητα της Διαδικτυακής Διαφήμισης στο Καταναλωτικό Κοινό – Δυναμική Μοντελοποίηση των τεχνικών χαρακτηριστικών των διαφημίσεων τύπου Banners”

ΚΟΤΣΩΡΗ ΣΟΦΙΑ
A.M.: 2022201802014

Επιβλέπων Καθηγητής: Dr. Σακάς Δαμιανός
Συνεπιβλέπων Καθηγητής: Dr. Νασιόπουλος Δημήτριος

ΤΡΙΠΟΛΗ, Φεβρουάριος 2020

Ευχαριστίες

Στα πλαίσια εκπόνησης της διπλωματικής μου εργασίας θα ήθελα να ευχαριστήσω τον καθηγητή και μέντορα αυτής της υπερπροσπάθειας κ. Νασσιόπουλο Δημήτριο, ο οποίος ήταν ο επιβλέπων της διπλωματικής μου εργασίας για την πολύτιμη βοήθεια, καθοδήγηση και υποστήριξη κατά τη διάρκεια της έρευνάς μου. Δεν θα μπορούσα φυσικά να μην ευχαριστήσω και τον κ. Σακκα Δαμιανό του Μεταπτυχιακού Προγράμματος «Επιστήμη των Υπολογιστών» για την επιλογή του στο να συμμετάσχω σε αυτό το μεταπτυχιακό, την συνεργασία που είχαμε σε όλη τη διάρκεια του έτους και την ευκαιρία να γνωρίσω θέματα πληροφορικής σε βάθος, καθώς δεν ανήκω στον χώρο αυτό.

Επίσης, θα ήθελα να εκφράσω τις ευχαριστίες μου στην οικογένειά μου, για την αμέριστη και συνεχή στήριξή της, ακόμα και στις καθημερινές δυσκολίες που είχα, τις οποίες αντιμετωπίσαμε από κοινού. Τέλος την συνάδερφο, συνοδοιπόρο και συμφοιτήτρια μου Ιωάννα, την αληθινή φίλη αυτής της δύσκολης για μένα προσπάθειας και συμπαραστάτισσα σε κάθε βήμα του μεταπτυχιακού αυτού.

ΠΕΡΙΛΗΨΗ

Στη διεθνή και ελληνική βιβλιογραφία και αρθρογραφία έχουν γίνει πολλές έρευνες για τη χρησιμότητα της διαδικτυακής διαφήμισης, υπάρχει παρόλα αυτά ερευνητικό κενό για την άποψη που έχουν οι καταναλωτές σχετικά με το display advertising και τις νέες μορφές προβολής διαφήμισης. Το ερευνητικό κενό έρχεται να καλύψει η προβληματική της παρούσας έρευνας, όπου σκοπός της είναι να εντοπίσει τον τύπο και τα τεχνικά χαρακτηριστικά της διαφήμισης banner σε προϊόντα υψηλής τεχνολογίας, που είναι αποτελεσματικότερα κατά την άποψη των καταναλωτών, λαμβάνοντας όμως υπόψη, κάποιες παραμέτρους όπως α) τον τύπο της διαφήμισης banner, β) τον αριθμό banners που θα πρέπει να προβάλλονται σε έναν ιστότοπο, γ) τη θέση που θα πρέπει να καταλαμβάνει ένα banner σε μια ιστοσελίδα, δ) το ιδανικό σχήμα του banner, ώστε να το επισκεφτούν περισσότεροι χρήστες και ε) το μέγεθος του banner που θα «τραβήξει» την προσοχή του καταναλωτή, ώστε να προτιμούνται τόσο απο τις διαφημιστικές εταιρίες, όσο και απο της διαφημιζόμενες. Το θεωρητικό πλαίσιο της παρούσας έρευνας περιλαμβάνει μια σύντομη αναφορά στον ορισμό της διαδικτυακής διαφήμισης και την εξέλιξή της τα τελευταία χρόνια. Επίσης, παρουσιάζονται οι βασικές μορφές της διαδικτυακής διαφήμισης, οι τόποι προβολής της και οι χώροι διάθεσής της, τα βασικά χαρακτηριστικά της on line διαφήμισης, τα πλεονεκτήματα και τα μειονεκτήματά της που έχουν για το καταναλωτικό κοινό, καθώς και τα χαρακτηριστικά, οι προϋποθέσεις επιτυχίας μιας διαδικτυακής διαφήμισης, αλλά και οι σημαντικότερες τεχνικές της αποτελεσματικότητάς της.

Το ερευνητικό πλαίσιο της παρούσας εργασίας περιλαμβάνει μια πρωτογενή έρευνα και συγκεκριμένα επιλέξαμε την ποσοτική μέθοδο με τη συλλογή στοιχείων μέσω της συμπλήρωσης πενήντα (50) ερωτηματολογίων κλειστού τύπου ερωτήσεων, διαφόρων ηλικιών των καταναλωτών. Η στατιστική ανάλυση των ευρημάτων έγινε με το πρόγραμμα του SPSS, ενώ η απεικόνιση του ερευνητικού μας προβλήματος στο fuzzy cognitive map και τέλος η δυναμική μοντελοποίηση των ερευνητικών μας υποθέσεων και προσομοίωση με το πρόγραμμα Any Logic. Τα αποτελέσματα, τα οποία και συμφώνησαν με τα δυναμικά μοντέλα που δημιουργήσαμε στο πρόγραμμα του Any Logic έδειξαν οτι α) η κινούμενη διαφήμιση banner που χρησιμοποιεί πολυμέσα -animations-είναι ένας από τους πιο αγαπημένους και αποτελεσματικότερους τύπους διαφήμισης banner μεταξύ των online καταναλωτών, β) η προβολή απο 1 έως 2 banners θεωρείται ιδανική για μια διαφημιστική και διαφημιζόμενη εταιρία, γ) το κάθετο banner είναι πιο αποτελεσματικό από το οριζόντιο banner, το οποίο τοποθετείται από αριστερά προς τα δεξιά, δ) η τοποθεσία πάνω δεξιά σε μια ιστοσελίδα είναι η ιδανική και ελκυστικότερη στον καταναλωτή, ε) η ελκυστικότερη διαφήμιση είναι η μεγέθους Half Page Ad.

Λέξεις Κλειδιά: προβολή διαφήμισης, αποτελεσματικότητα, μοντελοποίηση.

ABSTRACT

There has been a great deal of research into the usefulness of online advertising in international and Greek literature and articles, but there is still a research gap on how consumers view display advertising and new forms of display advertising. The research gap is to fill the problem of the present research, which aims to identify the type and technical characteristics of banner advertising in high-tech products that are more effective in the consumer's view, but taking into account some parameters such as a) the type of banner ad, b) the number of banners that should be displayed on a website, c) the position that a banner should occupy on a web page, d) the ideal banner shape for more users to visit. and (e) the size of the banner that will attract the attention of the consumer, so that they are preferred by both advertising companies and advertisers. The theoretical framework of the present research includes a brief reference to the definition of online advertising and its evolution in recent years. It also presents the basic forms of online advertising, its locations and venues, the key features of on-line advertising, its advantages and disadvantages for the consumer audience, as well as the features, conditions for success of a online advertising, but also the most important techniques of its effectiveness.

The research framework of this paper includes a primary research and in particular we selected the quantitative method by collecting data by completing fifty (50) closed-ended questionnaires of various ages of consumers. Statistical analysis of the findings was done with the SPSS program, while our research problem was presented on the fuzzy cognitive map and finally the dynamic modeling of our research hypotheses and simulated with the Any Logic program. The results, which agreed with the dynamic models we created in Any Logic's program, showed that a) animated banner advertising using animations-is one of the most popular and effective types of banner advertising among online consumers, b) 1 to 2 banners are considered ideal for an advertising and advertising company, c) the vertical banner is more effective than the horizontal banner, which is placed from left to right, d) the top right banner on a website is the ideal and attractive to the consumer, e) attractive advertising is megethyps Half Page Ad.

Keywords: display advertizing, effectiveness, simulatio3

Κατάλογος περιεχομένων

ΠΕΡΙΛΗΨΗ.....	3
ABSTRACT	4
ΕΙΣΑΓΩΓΗ.....	8
ΚΕΦΑΛΑΙΟ 1: ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ.....	10
1.1 Ορισμός διαδικτυακής διαφήμισης.....	10
1.2 Η Εξέλιξη της διαδικτυακής διαφήμισης.....	10
1.3 Μορφές διαδικτυακής/ψηφιακής διαφήμισης.....	12
1.4 Τόποι προβολής της διαδικτυακής διαφήμισης.....	19
1.5 Βασικά χαρακτηριστικά και ιδιαιτερότητες της ψηφιακής διαφήμισης	21
1.6 Πλεονεκτήματα και μειονεκτήματα της ψηφιακής διαφήμισης.....	24
ΚΕΦΑΛΑΙΟ 2: ΨΗΦΙΑΚΗ ΔΙΑΦΗΜΙΣΗ ΚΑΙ ΚΙΝΗΤΕΣ ΣΥΣΚΕΥΕΣ.....	26
2.1 Κινητές /Φορητές Συσκευές.....	26
2.2 Χαρακτηριστικά ψηφιακής διαφήμισης στην προβολή κινητών συσκευών.....	31
2.3 Προϋποθέσεις επιτυχίας μιας ψηφιακής διαφήμισης.....	32
2.4 Τρόποι αξιοποίησης της ψηφιακής διαφήμισης στο Internet.....	33
2.5 Στατιστικά δημοτικότητας μιας Ιστοσελίδας διαφημίσεων.....	35
2.6 Αποτελεσματικότητα ψηφιακής διαφήμισης.....	38
ΚΕΦΑΛΑΙΟ 3: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ ΓΙΑ ΤΑ ΔΙΑΦΗΜΙΣΤΙΚΑ BANNERS.....	41
3.1 Γενικές έρευνες.....	41
3.2 Τα βασικά στοιχεία της αποτελεσματικότητας της διαφήμισης των banners.....	48
3.3 Σχεδίαση μιας αποτελεσματικής banner διαφήμισης	49
3.4 Σκοπός και στόχοι της έρευνας.....	51
3.5 Ερευνητικές Υποθέσεις.....	52
3.6 Ανάλυση Ερευνητικών Υποθέσεων.....	55
3.6.1 Ερευνητική Υπόθεση 1.....	55
3.6.2 Ερευνητική Υπόθεση 2.....	56
3.6.3 Ερευνητική Υπόθεση 3.....	56
3.6.4 Ερευνητική Υπόθεση 4.....	57
3.6.5 Ερευνητική Υπόθεση 5.....	57
ΚΕΦΑΛΑΙΟ 4: ΜΕΘΟΔΟΛΟΓΙΑ ΕΡΕΥΝΑΣ.....	58
4.1 Εισαγωγή.....	58
4.2 Επιλογή τρόπου έρευνας.....	58
4.3 Εργαλείο μέτρησης/ Σχεδιασμός Ερωτηματολογίου.....	59
4.4 Αξιοπιστία – Εγκυρότητα έρευνας.....	60
4.5 Διανομή ερωτηματολογίου στον πληθυσμό.....	60
4.6 Μέγεθος Δειγματοληψίας και επιλογή Δείγματος	60
ΚΕΦΑΛΑΙΟ 5: ΑΠΟΤΕΣΜΑΤΑ ΕΡΕΥΝΑΣ.....	62
5.1 Ανάλυση Αποτελεσμάτων.....	62
5.2 Συσχέτιση Ερευνητικών Υποθέσεων.....	67
5.3 Συμπεράσματα της Έρευνας.....	69
ΚΕΦΑΛΑΙΟ 6: ΜΟΝΤΕΛΟΠΟΙΗΣΗ – ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΕΡΕΥΝΗΤΙΚΩΝ ΥΠΟΘΕΣΕΩΝ..	71
6.1 Γενικά	71
6.2 Μοντελοποίηση ερευνητικής υπόθεσης Y1.....	72
6.3 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y1.....	75
6.4 Μοντελοποίηση ερευνητικής υπόθεσης Y2.....	77
6.5 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y2.....	78
6.6 Μοντελοποίηση ερευνητικής υπόθεσης Y3.....	79
6.7 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y3.....	81

6.8 Μοντελοποίηση ερευνητικής υπόθεσης Υ4.....	81
6.9 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Υ4.....	83
6.10 Μοντελοποίηση ερευνητικής υπόθεσης Υ5.....	84
6.11 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Υ5.....	86
ΣΥΜΠΕΡΑΣΜΑΤΑ.....	87
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	89
ΠΑΡΑΡΤΗΜΑ Ι.....	98
ΠΑΡΑΡΤΗΜΑ ΙΙ.....	103

Περιεχόμενα Σχημάτων

1) Σχήμα 1 : Διαστάσεις Banner	Σελ. 13
2) Σχήμα 2: Σχηματική παράσταση ερευνητικών υποθέσεων	Σελ. 49
3) Σχήμα 3: Πλαίσιο διεξαγωγής έρευνας	Σελ. 54

Περιεχόμενα Διαγραμμάτων

1) Διάγραμμα 1: Φύλο	Σελ. 58
2) Διάγραμμα 2: Ηλικία	Σελ. 58
3) Διάγραμμα 3: Εκπαιδευτικό Επίπεδο	Σελ. 59
4) Διάγραμμα 4: On line Αγορές	Σελ. 59
5) Διάγραμμα 5: Προϊόντα Υψηλής Τεχνολογίας	Σελ. 60
6) Διάγραμμα 6: Μορφή Banners (1)	Σελ. 60
7) Διάγραμμα 7: Μορφή Banners (2)	Σελ. 61
8) Διάγραμμα 8: Αριθμός Banners	Σελ. 61
9) Διάγραμμα 9: Σχήμα Banners	Σελ. 62
10) Διάγραμμα 10: Τοποθεσία Προβολής Banners	Σελ. 62
11) Διάγραμμα 11: Μέγεθος Banners	Σελ. 63

Περιεχόμενα Εικόνων

1) Εικόνα 5.1: Συσχετίσεις Ερευνητικών Υποθέσεων	Σελ. 64
2) Εικόνα 5.2: FYZZY COGNITIVE MAP Ερευνητικών Υποθέσεων	Σελ. 64
3) Εικόνα 6.1: Δημιουργία Μοντέλου Υ1:Τύπος/Μορφή Banner	Σελ. 67
4) Εικόνα 6.2: Δημιουργία κεντρικού σχήματος	Σελ. 68
5) Εικόνα 6.3: Διάγραμμα συσχετίσεων παραγόντων της Υ1: Μορφή Προβολής Banner	Σελ. 69
6) Εικόνα 6.4: Διάγραμμα Bar Chart	Σελ.70

7) Εικόνα 6.5: Διάγραμμα ροής με την διανομή των πόρων Υ1	Σελ.70
8) Εικόνα 6.6: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ1	Σελ. 71
9) Εικόνα 6.7: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ1 με barcharts	Σελ. 72
10) Εικόνα 6.8: Διάγραμμα συσχετίσεων παραγόντων της Υ2: Αριθμός Προβολής Banner	Σελ. 72
11) Εικόνα 6.9: Διάγραμμα ροής με την διανομή των πόρων Υ2	Σελ. 73
12) Εικόνα 6.10: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ2	Σελ. 74
13) Εικόνα 6.11: Διάγραμμα συσχετίσεων παραγόντων της Υ3: Σχήμα Banner	Σελ. 75
14) Εικόνα 6.12: Διάγραμμα ροής με την διανομή των πόρων Υ3	Σελ. 75
15) Εικόνα 6.13: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ3 με chartbars	Σελ. 76
16) Εικόνα 6.14: Διάγραμμα συσχετίσεων παραγόντων της Υ4: Τοποθεσία Προβολής Banner	Σελ. 77
17) Εικόνα 6.15: Διάγραμμα ροής με την διανομή των πόρων Υ4	Σελ. 78
18) Εικόνα 6.16: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ4	Σελ. 79
19) Εικόνα 6.17: Διάγραμμα συσχετίσεων παραγόντων της Υ5: Μέγεθος Banner	Σελ. 80
20) Εικόνα 6.18: Διάγραμμα ροής με την διανομή των πόρων Υ5	Σελ. 80
21) Εικόνα 6.19: Βελτιστοποίηση Ερευνητικής Υπόθεσης Υ5	Σελ. 81

ΕΙΣΑΓΩΓΗ

Η συνεχόμενα αυξανόμενη χρήση του Internet σε συνδυασμό με την τεχνολογική εξέλιξη, έχουν αλλάξει σημαντικά τη συμπεριφορά των ανθρώπων σε σχέση με τον τρόπο που επικοινωνούν με τον περίγυρο τους, καθώς και το πώς ψυχαγωγούνται και ενημερώνονται. Οι παραπάνω διαφοροποιήσεις έστρεψαν τις διαφημιστικές εταιρίες προς το Διαδίκτυο, το οποίο και αξιοποίησαν στο έπακρο καθώς είναι ένα μέσο καινοτόμο που τους προσφέρει σημαντικό έλεγχο στη διαδικασία της επικοινωνίας σε σχέση με τον διαδικτυακό χρήστη - καταναλωτή. Η ραγδαία εξέλιξη και ανάπτυξη του Internet και η σημαντικά αυξανόμενη χρήση του έχει δημιουργήσει διάφορες μορφές διαφήμισης οι οποίες αξιοποιούνται και λειτουργούν για την προώθηση προϊόντων και την αύξηση των πωλήσεων.

Η αλληλεπίδραση των διαφημίσεων στο Internet είναι ίσως το σημαντικότερο στοιχείο για την αποτελεσματικότητα της και δίνει την δυνατότητα στην δημιουργία νέων μέσων προώθησης προς συγκεκριμένο κοινό, ομάδα χρηστών (target group). Η αποτελεσματικότητα των διαφημίσεων εκτός της αλληλεπίδρασης είναι και συνάρτηση και άλλων παραγόντων. Πρόκληση αποτελεί η πρόσβαση σε μεγάλο αριθμό πληροφοριών για τα διαφημιζόμενα μέσα. Ο θεατής από απλός δέκτης γίνεται ενεργός χρήστης, (Bradley, 2006) δίνοντας με αυτόν τον τρόπο την δυνατότητα στον διαφημιστή να συλλέξει σημαντικές πληροφορίες για την συμπεριφορά του θεατή ως καταναλωτή (Lekakos et. al., 2001).

Στο περιβάλλον του Διαδικτύου διαμορφώθηκαν και ακόμα διαμορφώνονται πολλαπλές μορφές Διαδικτυακής διαφήμισης. Ένας γενικά παραδεκτός διαχωρισμός αναφέρεται σε δύο κατηγορίες: Η πρώτη αφορά στις εταιρικές ιστοσελίδες προβολής και διαφήμισης και η δεύτερη στα υπόλοιπα είδη Διαδικτυακής διαφήμισης που λειτουργούν ως μέσα προώθησης των εταιρικών ιστοσελίδων όπως είναι τα «πλαίσια» (banners), οι χορηγίες (sponsorships), οι «σύνδεσμοι κειμένου» (text links) κ.α.

Οι διαφημιστικές ιστοσελίδες αποτελούν τη βασική διαφημιστική μονάδα κάθε επιχείρησης στο Διαδίκτυο και αποσκοπούν κυρίως στην προβολή και προώθηση των προϊόντων και της ίδιας της επιχείρησης. Πρόκειται για ιστοσελίδες που ενσωματώνουν πολλαπλές πολιτικές και λειτουργίες για να επιτύχουν το βασικό σκοπό της λειτουργίας τους. Λειτουργούν σαφώς ως μέσα παροχής πληροφοριών και δημιουργίας θετικής εικόνας για τα προϊόντα και την επιχείρηση, αλλά ταυτόχρονα και ως διάλογο επικοινωνίας με το κοινό-στόχο, ως μέσα «ανάδρασης», ως μέσα υποστήριξης πελατών και ως μέσα παροχής υπηρεσιών για τους επισκέπτες και πελάτες της εκάστοτε επιχείρησης.

Η μέτρηση αποτελεσματικότητας διαφημιστικών ιστοσελίδων λόγω του πολλαπλού ρόλου

τους και της πολυπλοκότητας και διαφορετικότητας του μέσου, αποτελεί μια πρόκληση για τους θεωρητικούς και πρακτικούς της διαφήμισης. Οι πρώτες προσπάθειες εννοιολογικής προσέγγισης του προβλήματος χαρακτηρίζονται από την υιοθέτηση και χρήση δεικτών, κριτηρίων και αποτελεσμάτων της παραδοσιακής διαφήμισης. Ωστόσο, πολύ σύντομα η ανεπάρκεια της συγκεκριμένης πρακτικής να αξιολογήσει και να αποσαφηνίσει την ανταπόκριση του κοινού σε αυτή τη μορφή διαφήμισης, δημιούργησε την ανάγκη ενός νέου θεωρητικού πλαισίου μέτρησης αποτελεσματικότητας των εταιρικών διαφημιστικών ιστοσελίδων.

Η «μοντελοποίηση» της διαφημιστικής αποτελεσματικότητας αποτελεί μια μέθοδο ερευνητικής προσέγγισης της μέτρησης αποτελεσματικότητας, που χρησιμοποιείται πολύ συχνά από τους ακαδημαϊκούς στο χώρο της διαφήμισης. Συγκεκριμένα, αναφέρεται στην αναγνώριση παραγόντων και στην αναπαράσταση των συσχετισμών τους με συγκεκριμένα αποτελέσματα διαφήμισης, βάσει της ανταπόκρισης του κοινού σε συγκεκριμένα διαφημιστικά «ερεθίσματα». Οι παράγοντες που επηρεάζουν τη στάση του κοινού απέναντι στη διαφήμιση και συντελούν στη διαμόρφωση των επιθυμητών αποτελεσμάτων διαφήμισης, θα αποτελέσουν κριτήρια αξιολόγησης και μέτρησης αποτελεσματικότητας, προκειμένου να ελέγξουν οι επιχειρήσεις την πετυχημένη ή όχι λειτουργία των εταιρικών τους ιστοσελίδων.

Η μέτρηση αποτελεσματικότητας ιστοσελίδων διαφήμισης πρόκειται να απασχολήσει σημαντικά την ακαδημαϊκή κοινότητα στο μέλλον, καθώς η ραγδαία υιοθέτηση του Διαδικτύου από το κοινό και η ανάπτυξη εταιριών στα πλαίσια εφαρμογής «σχεσιακού» και Διαδικτυακού μάρκετινγκ, καθιστούν την αποτελεσματική διαφημιστική παρουσία στο Διαδίκτυο βασική προτεραιότητα για την πλειοψηφία των επιχειρήσεων. Οι μετρήσεις αποτελεσματικότητας ενσωματώνουν θεωρητικές προσεγγίσεις, έρευνες και μελέτες χρηστικότητας, ποσοτικές μετρήσεις με την ανάλυση των log αρχείων. Η συνεισφορά όλων αυτών των μετρήσεων είναι σημαντική για τη βελτιστοποίηση της λειτουργίας των διαφημιστικών ιστοσελίδων. Άλλωστε, είναι κοινός τόπος για τους ανθρώπους της διαφήμισης ότι μια μόνο προσέγγιση δεν επαρκεί για την ολοκληρωμένη κάλυψη του θέματος (Bradley, 2006).

ΚΕΦΑΛΑΙΟ 1: ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ

1.1 Ορισμός διαδικτυακής διαφήμισης

Διαφήμιση μέσω του διαδικτύου ή on-line διαφήμιση (όρος που υιοθετήθηκε τα τελευταία χρόνια από τον ακαδημαϊκό και διαφημιστικό χώρο) είναι το είδος της διαφήμισης που χρησιμοποιεί αποκλειστικά ως μέσω επικοινωνίας και προβολής το διαδίκτυο και πιο συγκεκριμένα το βασικό εργαλείο του διαδικτύου, τον Παγκόσμιο Ιστό πληροφοριών (World Wide Web) (Κοκκίνης, 2017).

Σύμφωνα με τους Δημητριάδη και Μπαλτά (2003), η διαφήμιση στο ίντερνετ είναι κάθε πληρωμένος χώρος σε ιστοσελίδα (site) ή σε ηλεκτρονικό ταχυδρομείο (e-mail) που χρησιμεύει για εμπορική επικοινωνία. Η διαφήμιση στο ίντερνετ μπορεί να αποσκοπεί στην ενίσχυση της μάρκας στην αγορά (brand building) ή να προσπαθεί να προκαλέσει την άμεση αντίδραση του καταναλωτή (direct-response advertising) με τη μεταφορά στη δική μας ιστοσελίδα (site) ή αγορά του προϊόντος μας. Το τελευταίο είδος διαφήμισης αποτελεί μεγάλο πλεονέκτημα του διαδικτύου σε σχέση με άλλα οχήματα διαφήμισης, όπως η τηλεόραση ή το ραδιόφωνο που δεν υποστηρίζουν αμφίδρομη επικοινωνία και συναλλαγές.

1.2 Η Εξέλιξη της διαδικτυακής διαφήμισης

Πριν από μισό αιώνα σε πανεπιστήμια των ΗΠΑ οι ερευνητές πειραματίζονταν με τη διασύνδεση απομακρυσμένων υπολογιστών μεταξύ τους και μελετούσαν την αξιοπιστία της λειτουργίας των δικτύων.

Το 1964 με χρήση απλής τηλεφωνικής γραμμής δημιουργείται σύνδεση δύο Υπολογιστών, ενός στο MIT στη Μασαχουσέτη και ενός σε Εταιρεία Ανάπτυξης Συστημάτων στην Καλιφόρνια.

Στα τέλη της δεκαετίας του 1960 υλοποιήθηκε μια τεχνική μετάδοσης δεδομένων μέσω καλωδιακών συνδέσεων. Ένα δίκτυο γεννιέται το 1968 με επιδότηση από το πρόγραμμα ARPA (Advanced Research Project Agency) του Υπουργείου Άμυνας των ΗΠΑ, με σκοπό να συνδέσει το Υπουργείο με στρατιωτικούς ερευνητικούς οργανισμούς. Το πρώτο, περιορισμένου βεληνεκούς, δίκτυο, ονομάστηκε ARPAnet.

Αρχικός στόχος του προγράμματος ήταν ο πειραματισμός με την τεχνολογία μεταγωγής πακέτων (packet switching), σύμφωνα με την οποία τα προς μετάδοση δεδομένα κόβονται σε πακέτα και πολλοί χρήστες μπορούν να μοιραστούν την ίδια γραμμή.

Η χρήση του πρωτοποριακού λειτουργικού συστήματος UNIX, που αναπτύχθηκε από τους D. Ritchie και K. Thomson, διαδραμάτισε σημαντικό ρόλο στην εξέλιξη αυτού που μετέπειτα

ονομάστηκε Διαδίκτυο.

Σταδιακά συνδέθηκαν με το ARPAnet ιδρύματα από άλλες χώρες, με πρώτα το University College of London στην Αγγλία και το Royal Radar Establishment στη Νορβηγία. Το 1973 ξεκίνησε ένα νέο ερευνητικό πρόγραμμα που ονομάστηκε Πρόγραμμα Διαδικτύωσης (Interneting Project). Από ένα ερευνητικό μέρος του γεννήθηκε το Πρωτόκολλο Διαδικτύωσης (Internet Protocol - I.P.), ενώ ξεπεράστηκε και το πρόβλημα διασύνδεσης ανόμοιων δικτύων και έγινε δυνατή η ομοιόμορφη διακίνηση δεδομένων από το ένα δίκτυο στο άλλο. Από το όνομα του προγράμματος πήρε το 1974 το όνομα του σαν Διαδίκτυο (Internet) στην ορολογία της Πληροφορικής και των Τηλεπικοινωνιών. Έτσι, διαφορετικά δίκτυα, που χρησιμοποιούσαν το κοινό Πρωτόκολλο Διαδικτύωσης, μπορούσαν να αποτελούν ένα διαδίκτυο, με όλους τους υπολογιστές σοδύναμους και με δυνατότητα επικοινωνίας όλων των ενδιαφερόμενων μερών.

Επίσης σχεδιάστηκε για τον έλεγχο της μετάδοσης των δεδομένων το Πρωτόκολλο Ελέγχου Μετάδοσης (Transmission Control Protocol - T.C.P.), καθώς και ο συνδυασμός των T.C.P. και I.P., το πρωτόκολλο T.C.P / I.P.

Στις αρχές της δεκαετίας του 1980, μετά από μελέτες του Πανεπιστημίου Berkeley, έγινε η ανάπτυξη ενός πρωτοκόλλου που έδωσε την οριστική λύση στα προβλήματα συμβατότητας μεταξύ του λειτουργικού συστήματος Berkeley UNIX, το οποίο περιλάμβανε το T.C.P / I.P, συντέλεσε στη γρήγορη εξάπλωση της διαδικτύωσης των υπολογιστών.

Το 1983 το πρωτόκολλο T.C.P / I.P πήρε έγκριση και αναγνωρίστηκε ως πρότυπο από τις αρμόδιες υπηρεσίες, ενώ ορίζονται προδιαγραφές για τη μεταφορά αρχείων (F.T.P) και το Ηλεκτρονικό Ταχυδρομείο (E-mail).

Στη συνέχεια εκατοντάδες Πανεπιστήμια συνδέουν τους υπολογιστές τους στο ARPAnet, το οποίο επιβαρύνεται πολύ και έτσι το 1983 χωρίζεται σε δύο τμήματα, στο MILnet για στρατιωτικές επικοινωνίες και στο νέο ARPAnet για χρήση αποκλειστικά από την πανεπιστημιακή κοινότητα και συνέχιση της έρευνας.

Το 1985 το National Science Foundation (N.S.F), προκειμένου να συνδέσει πέντε κέντρα υπερ-υπολογιστών μεταξύ τους, δημιούργησε ένα δικό του γρήγορο δίκτυο το NSFNET χρησιμοποιώντας το πρωτόκολλο T.C.P / I.P. Στα τέλη της δεκαετίας του 1980 χώρες, όπως η Σουηδία, ο Καναδάς, η Γαλλία, η Αυστραλία, η Γερμανία, η Ιταλία, συνδέθηκαν στο NSFNET, ενώ χιλιάδες Πανεπιστήμια και οργανισμοί δημιουργούν δικά τους δίκτυα που συνδέθηκαν στο παγκόσμιο αυτό δίκτυο.

Όλο και περισσότερες χώρες συνδέθηκαν στο NSFNET, μεταξύ των οποίων και η Ελλάδα το 1990, ενώ το ARPAnet ουσιαστικά το 1990 καταργείται.

Το 1993 το εργαστήριο CERN στην Ελβετία παρουσιάζει την πιο σπουδαία υπηρεσία του

Διαδικτύου, τον Παγκόσμιο Ιστό (World Wide Web - WWW), που αναπτύχθηκε από τον Tim Berners-Lee. Ο Παγκόσμιος Ιστός είναι ένα σύστημα διασύνδεσης πληροφοριών, σε μορφή Πολυμέσων (Multimedia), αποθηκευμένων σε ηλεκτρονικές σελίδες, στις οποίες μπορεί να περιηγηθεί κανείς. Το γραφικό αυτό περιβάλλον έκανε την εξερεύνηση του Διαδικτύου προσιτή στον απλό χρήστη.

Εμφανίσθηκαν στη συνέχεια διάφορα εμπορικά δίκτυα που ανήκουν σε εταιρείες Παροχής Υπηρεσιών Διαδικτύου (Internet Service Providers-I.S.P.), που προσφέρουν πρόσβαση σε αυτό για όλους σε τιμές που διαρκώς μειώνονται, αρκεί κάποιος να διαθέτει τα απαραίτητα για σύνδεση. Ένα αποτέλεσμα αυτού ήταν το φορτίο του NSFNET, που καταργείται πλέον το 1995 επίσης, μεταφέρεται σε εμπορικά δίκτυα.

Η δυνατότητα χρήσης του Παγκόσμιου Ιστού, σε συνδυασμό με την προσφερόμενη δυνατότητα εύκολης πρόσβασης στο Διαδίκτυο, προσέλκυσε ένα μεγάλο αριθμό καινούργιων χρηστών.

Το Ηλεκτρονικό Ταχυδρομείο (e-mail), μέχρι το 2003, ήταν η περισσότερο χρησιμοποιούμενη υπηρεσία του Διαδικτύου παγκοσμίως.

Από το 2004 και μετά παρατηρείται ανάπτυξη και των υπόλοιπων υπηρεσιών, λειτουργιών και εργαλείων του, όπως το Ηλεκτρονικό Εμπόριο, η Τηλεδιάσκεψη, η Τηλεκπαίδευση, η Τηλεργασία κ.λπ.

Τέλος, παρατηρούμε ότι καθημερινά επεκτείνεται και επιδρά στις καθημερινές πρακτικές όλων μας. Περιοδικά και εφημερίδες μας παραπέμπουν στις διευθύνσεις τους στο Διαδίκτυο, επιχειρήσεις και ιδιώτες φτιάχνουν τις δικές τους σελίδες, και είναι προφανές ότι το Διαδίκτυο δεν αποτελεί πλέον ένα δίκτυο των φοιτητών και των ερευνητών, αλλά σχεδόν όλων των κατοίκων του πλανήτη (Κόκοτος & Λιναρδάτος, 2009).

1.3 Μορφές διαδικτυακής/ψηφιακής διαφήμισης

Η διαφήμιση στο διαδίκτυο έχει διάφορες μορφές όπως τις αφίσες (banners), τους συνδέσμους (links), το διαφημιστικό κείμενο (text ads), τις μικρές αγγελίες (classifieds ads), τα pop ups/pop unders, τα rich media, τα interstitials, τις πληρωμένες λέξεις κλειδιά (paid search) και άλλα. Θα εξετάσουμε αναλυτικά μερικά από αυτά.

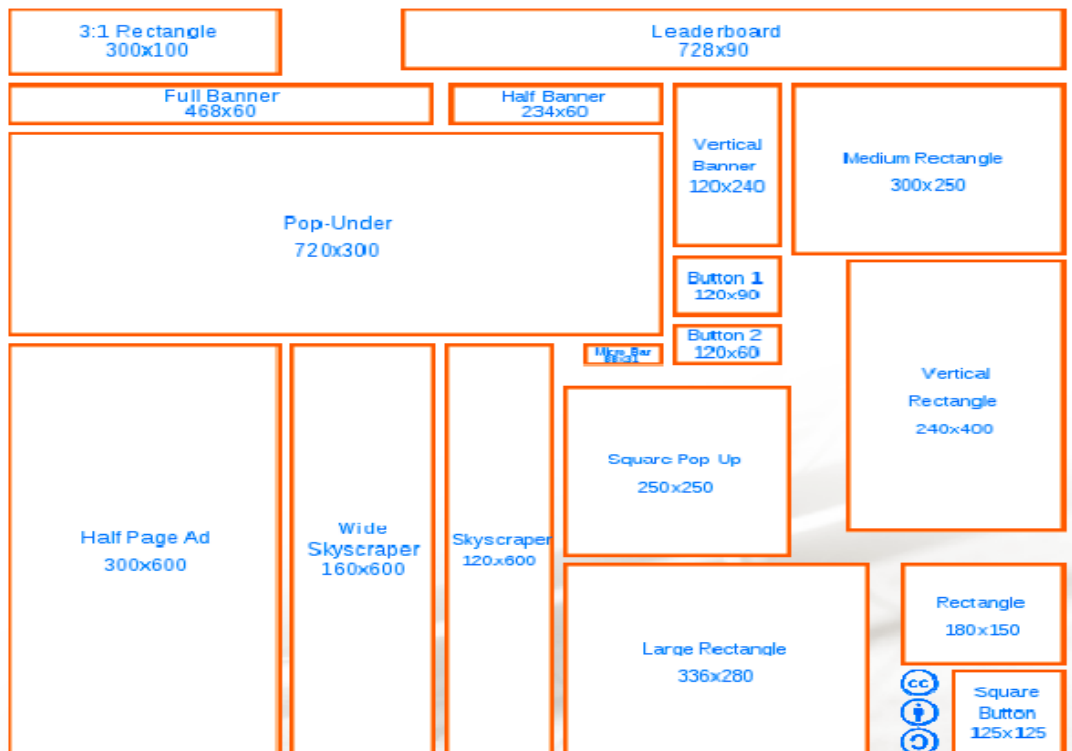
Διαφημιστικά ψηφιακά γραφικά εικονίδια (Advertising Banners)

Τα Banners (ψηφιακά γραφικά) αποτελούν τη πιο γνωστή και διαδεδομένη μορφή διαφημιστικών μηνυμάτων στην διαδικτυακή διαφήμιση. Είναι μικρά, διαφόρων σχημάτων

εικονίδια με γραφικά, τα οποία τοποθετούνται σε εμφανή σημεία ιστοσελίδων για να γίνουν αντιληπτά και συνήθως προωθούν προϊόντα, υπηρεσίες ή άλλες ιστοσελίδες. Τα Banners λειτουργούν ως υπερσύνδεσμοι, που σημαίνει ότι αν ο χρήστης κάνει 'κλικ' πάνω τους τότε εισέρχεται στο διαδικτυακό τόπο που τον σχεδίασε και τον χρηματοδοτεί (Zeff και Aronson, 1999). Η λειτουργία των Banners γίνεται σύμφωνα με το CPC (cost per click), δηλαδή η επιχείρηση κερδίζει όταν κάποιος χρήστης του διαδικτύου κάνει κλικ στη διαφημιζόμενη εικόνα και εισχωρήσει σε μια καινούργια ιστοσελίδα.

Γενικά, οι κατηγορίες banners που υπάρχουν είναι τα static, pop-up, animated, dynamic και rotated (Hussain et al., 2010). Στα πρώτα χρόνια λόγω των τεχνικών αδυναμιών χρησιμοποιήθηκαν static banners τα οποία κατασκευάζονταν σε αρχεία gif. Σήμερα η κατηγορία αυτή δεν χρησιμοποιείται. Αντίθετα, σήμερα χρησιμοποιούνται ευρέως τα κινούμενα ή αλλιώς animated banners», τα οποία με την κίνηση τους προσελκύουν την προσοχή των καταναλωτών αλλά ταυτόχρονα πολλαπλασιάζουν και την επιφάνεια προβολής (Φρίγκας, 2005). Επίσης, υπάρχουν και τα interactive banners, τα οποία ελκύουν ευκολότερα την προσοχή των καταναλωτών και εντυπώνονται καλύτερα στη μνήμη τους. Τα Interactive Banners χωρίζονται στις εξής υποκατηγορίες (Πασχόπουλος, 2009): α) κινούμενα σχέδια τα οποία περιέχουν ενεργές λέξεις ή εικόνες, β) games, τα οποία περιέχουν παιχνίδια ή πάζλ, γ) tickers, που είναι banners σε μορφή ανακοινώσεων και τέλος δ) Audio/Video, δηλαδή banners που περιέχουν ήχο και βίντεο.

Οι παράγοντες που φαίνεται να επηρεάζουν στο αν ένας καταναλωτής δώσει προσοχή σε ένα διαφημιστικό banner είναι το είδος του banner που θα χρησιμοποιηθεί, ο αριθμός των banners σε μία σελίδα, η θέση στην οποία είναι τοποθετημένο, το μέγεθός του, καθώς και το σχήμα του (Hussain et al., 2010). Πιο συγκεκριμένα, θεωρείται ότι, ο μικρότερος αριθμός banners σε μία σελίδα, τα banners που είναι κατακόρυφα και όχι οριζόντια, όσα είναι τοποθετημένα στη δεξιά πλευρά της σελίδας, και όσα είναι μεγαλύτερα σε μέγεθος τείνουν να έχουν περισσότερο θετικά αποτελέσματα (Grigorovici και Constantin, 2004; Hussain et al., 2010). Οι διαστάσεις των banners διακρίνονται στο σχήμα που ακολουθεί:



ΣΧΗΜΑ 1 : Διαστάσεις Banner ΠΗΓΗ: IAB(www.iab.gr).

Επίσης, υπάρχουν κάποιες ενδείξεις ότι τα rich media έδωσαν ώθηση στα banners μέσω της εισαγωγής μουσικής, κίνησης και 3D γραφικών, κάτι το οποίο έφερε μάλλον θετικά αποτελέσματα έναντι των 2D banners (Pharr, 2004; Grigorovici και Constantin, 2004). Πιο συγκεκριμένα, τα rich media θεωρούνται περισσότερο αποτελεσματικά στον τομέα της διαφήμισης, καθώς ενθαρρύνουν τη διαδραστικότητα με το χρήστη και ενισχύουν τη συμμετοχή του, κάτι το οποίο κάνει το χρήστη περισσότερο θετικό σε αυτό το είδος διαφημίσεων (Rosenkrans, 2009). Επίσης, υποστηρίζεται ότι τα banners που λειτουργούν με «κλικ» δεν είναι τόσο αποτελεσματικά όσο τα απλά διαφημιστικά banners χωρίς να υποχρεώνεται ο χρήστης να κάνει «κλικ» (Pharr, 2004). Παρά το γεγονός ότι χρειάζεται περισσότερη έρευνα σχετικά με την αποτελεσματικότητα των banners, υπάρχει η άποψη σύμφωνα με την οποία η αποτελεσματικότητα της διαφήμισης μίας μάρκας εξαρτάται και από τον τύπο του προϊόντος/υπηρεσίας που διαφημίζεται μέσω διαδικτύου, καθώς ορισμένα προϊόντα ενδέχεται να μη διαφημίζονται τόσο αποτελεσματικά στο διαδίκτυο όσο στα παραδοσιακά μέσα (Pharr, 2004).

Rich media ads

Οι διαφημίσεις πλούσιες σε μέσα (rich media ads), για να προβάλλουν το διαφημιστικό μήνυμα συνδυάζουν όπως και τα banners, κείμενο, γραφικά, εικόνες, ήχο, animation αλλά και video. Επιπλέον οι διαφημίσεις αυτές διαθέτουν διάδραση (interaction) με τον θεατή, η οποία

υλοποιείται με την μετακίνηση του ποντικιού πάνω από την περιοχή της διαφήμισης. Τότε η περιοχή που καταλαμβάνει η διαφήμιση συνήθως μεγαλώνει προσφέροντας στο θεατή διάφορες δυνατότητες αλληλεπίδρασης προσελκύοντας το ενδιαφέρον του.

Οι βασικές κατηγορίες πλούσιων σε μέσα διαφημίσεων είναι:

Οι *peel back διαφημίσεις* είναι ένας καινοτομικός τρόπος διαφημιστικής επικοινωνίας η οποία με τη χρήση της οπτικής μεταφοράς της μετάβασης στην επόμενη σελίδα ενός βιβλίου προσφέρει μια πλούσια σε μέσα πλατφόρμα για τη μετάδοση διαφημιστικών μηνυμάτων. Μια *floating διαφήμιση* είναι μια online διαφήμιση, πλούσια σε μέσα, η οποία εμφανίζεται χωρίς να έχει ζητηθεί πάνω από μια σελίδα που έχει ζητήσει ένας θεατής, και εξαφανίζεται ή σταματά να είναι ενοχλητική ύστερα από μια συγκεκριμένη χρονική περίοδο (συνήθως 5-30 δευτερόλεπτα).

Μια *expanding διαφήμιση* είναι μια πλούσια σε μέσα διαφήμιση, η οποία αλλάζει μέγεθος καθώς και το περιεχόμενο της μέσα στη σελίδα την οποία προβάλλεται. Με τον τρόπο αυτό κατεβαίνουν τμηματικά το περιεχόμενο της διαφήμισης, αποφεύγοντας έτσι καθυστερήσεις στο φόρτωμα της σελίδας στην οποία έχει ενσωματωθεί.

Στο διαδίκτυο *interstitials* ονομάζονται οι σελίδες που προβάλλονται πριν ή μετά από την σελίδα με το αναμενόμενο περιεχόμενο. Συχνά οι σελίδες αυτές περιέχουν διαφημίσεις ή άλλες φορές ζητούν την επιβεβαίωση (ή την εισαγωγή), στοιχείων του χρήστη όπως πχ την ηλικία του αν έχει ζητήσει κατάλληλο περιεχόμενο μόνο για ενηλίκους. (Ευαγγελάτος Α.,2010). Οι *interstitials* διαφημίσεις προβάλλονται μετά το αίτημα ενός επισκέπτη ενός διαδικτυακού τόπου να δει συγκεκριμένο περιεχόμενο. Ο επισκέπτης είτε θα αποφύγει τη διαφήμιση και θα οδηγηθεί στην ιστοσελίδα με το περιεχόμενο που ζήτησε ή θα την επιλέξει οπότε θα ανοίξει μια καινούρια σελίδα στον φυλλομετρητή του οδηγώντας τον στην ιστοσελίδα του προϊόντος ή της υπηρεσίας που διαφημίζεται. Οι διαφημίσεις αυτές αφενός πολύ συχνά ενοχλούν τους επισκέπτες ενός διαδικτυακού τόπου και αφετέρου εγείρουν ζητήματα δεοντολογίας σχετικά με το εάν είναι επιτρεπτό να υποχρεώνεις έναν επισκέπτη να δει πληροφορίες που δεν ζητάει.

Οι *pop up και pop under* διαφημίσεις, είναι διαφημίσεις πλούσιες σε μέσα, οι οποίες είναι παράθυρα του φυλλομετρητή που ανοίγουν πάνω με το άνοιγμα της σελίδας που έχει ζητήσει ο επισκέπτης (*pop ups*) ή με το κλείσιμο της (*pop under*). Το συγκεκριμένο παράθυρο το οποίο περιλαμβάνει την διαφήμιση δεν έχει ζητηθεί από τον επισκέπτη με συνέπεια η ενέργεια (η εμφάνιση του παραθύρου) να θεωρείται από την πλειοψηφία των χρηστών ενοχλητική.

Οι *video* διαφημίσεις, είναι πλούσιες σε μέσα διαφημίσεις, οι οποίες κάνοντας χρήση video μεταδίδουν το διαφημιστικό μήνυμα. Η χρήση του video γίνεται με δύο τρόπους α) μέσω της δημιουργίας ενός διαφημιστικού video το οποίο ανεβαίνει στη συνέχεια σε ένα site όπως το you tube όπου μπορούν να το δουν οι θεατές β) Μέσω κειμένου του οποίου γίνεται ένθεση σε διάφορα

video.sites. Η δημιουργία ενός ενδιαφέροντος διαφημιστικού video είναι μια καλή τεχνική viral marketing.

Text ads

Οι διαφημίσεις κειμένου (text ads) είναι ακόμη μια μορφή διαδικτυακής διαφήμισης, είναι κείμενο, το οποίο εκτός από τις πληροφορίες περιεχομένου έχει ενσωματωμένους υπερκείμενους συνδέσμους (hypertext) ώστε πατώντας τους οι επισκέπτες να πηγαίνουν στη διαφημιζόμενη ιστοσελίδα. (Ευαγγελάτος Α., 2012).

Οι διαφημίσεις αυτές συνήθως είναι φθηνότερες έχοντας ένα επιπλέον πλεονέκτημα να μπορούν να καταχωρηθούν σε μηχανές αναζήτησης. Οι βασικές κατηγορίες κειμενικών διαφημίσεων είναι:

Οι διαφημίσεις *υπερσύνδεσμοι (links)* είναι διαφημιστικό κείμενο το οποίο τοποθετείται σε ένα blog ή ένα web site και το οποίο περιλαμβάνει υπερσυνδέσμους (hyperlinks) που οδηγούν τον επισκέπτη της σελίδας που θα κάνει click πάνω τους στην σελίδα που έχει ορίσει για αυτό αυτός που πληρώνει τη διαφήμιση και ονομάζεται διαφημιζόμενος.

Τα *blogs και τα web sites* που έχουν μεγάλη επισκεψιμότητα τοποθετούν διαφημίσεις συνδέσμους στο περιεχόμενο τους με σκοπό τη δημιουργία εσόδων από τους διαφημιζόμενους. Οι εκδότες πληρώνονται από τους διαφημιζόμενους είτε με βάση τον αριθμό των κλικ που γίνονται από τους επισκέπτες προς την σελίδα του διαφημιζόμενου, είτε με κάποια σταθερή χρέωση.

Η *contextual* διαφήμιση είναι μια μορφή διαφήμισης κειμένου η οποία εμφανίζεται κυρίως σε web sites και blogs αλλά έχει αρχίσει να επεκτείνεται και σε άλλα μέσα, όπως τα κινητά τηλέφωνα με πρόσβαση στο διαδίκτυο. Είναι μια μορφή στοχευμένης διαφήμισης όπου το διαφημιστικό κείμενο επιλέγεται και προωθείται για προβολή μέσω αυτοματοποιημένων διαδικασιών σε εκείνα τα μέσα των οποίων το περιεχόμενο είναι συναφές με το περιεχόμενο του διαφημιστικού μηνύματος. Για παράδειγμα ο επισκέπτης ενός αθλητικού site, είναι πολύ πιθανόν να ενδιαφέρεται να δει διαφημίσεις σχετικές με σπορ.

Οι *Search Engine Marketing* διαφημίσεις είναι μια μορφή contextual διαφημίσεων και μάλιστα οι πλέον διαδεδομένη καθώς βασίζεται στον τρόπο λειτουργίας των μηχανών αναζήτησης. Η πιο γνωστή Search Engine Marketing διαφήμιση είναι τα adwords της Google. Όταν ο επισκέπτης πληκτρολογήσει μια λέξη στη μηχανή αναζήτησης πέραν των αποτελεσμάτων, που ονομάζονται οργανικά, παρουσιάζει στο πάνω μέρος καθώς και δεξιά του παραθύρου μια σειρά από διαφημίσεις κειμένου. Τα κείμενα αυτά είναι συναφή με τη λέξη – κλειδί που χρησιμοποίησε ο χρήστης στην αναζήτηση του.

Οι *ηλεκτρονικοί κατάλογοι (online directories)* αποτελούν άλλη μια κατηγορία των

διαφημίσεων κειμένου. Οι ηλεκτρονικοί κατάλογοι είναι διαδικτυακοί τόποι στους οποίους μπορεί να καταχωρήσει κάποιος τα στοιχεία επικοινωνίας της επιχείρησής του, έτσι ώστε κάποιος που την αναζητά να μπορεί εύκολα να την εντοπίσει. Οι ηλεκτρονικοί κατάλογοι συνήθως προσφέρουν θεματική οργάνωση των πληροφοριών καθώς και δική τους μηχανή αναζήτησης. Παραδείγματα τέτοιων ηλεκτρονικών καταλόγων είναι ο Χρυσός Οδηγός, το Vres.gr, το in.gr και άλλα.

Υπερσύνδεσμος κειμένου (Text links)

Τα text links αποτελούν μια μορφή κειμένου, λογότυπου ή μικρών εικόνων, τα οποία ενσωματώνονται σε ιστοσελίδες. Στο χώρο της διαφήμισης αυτό το κείμενο αποτελεί κάποιο κομμάτι ενός διαφημιστικού μηνύματος και ο χρήστης με ένα «κλικ» μεταβαίνει στην σελίδα του διαφημιζόμενου (Βλάχος και Δρόσος, 2004). Το text link είναι μια πολύ απλή μορφή διαδικτυακής διαφήμισης, στην οποία συνήθως παρουσιάζεται ένα μικρό κείμενο μέσα στο οποίο υπάρχει αυτός ο σύνδεσμος που οδηγεί σε άλλη ιστοσελίδα. Συνήθως, διαφέρει σε χρώμα, γραμματοσειρά και μέγεθος από τον υπόλοιπο σκελετό του κειμένου.

Ψηφιακά Κουμπιά (Buttons)

Η κυρίως θέση αυτών των διαφημίσεων είναι στο μέσον της οθόνης ή στην άκρη, σε αντίθεση με τα banners που υπάρχουν στην αρχή ή στο τέλος της ιστοσελίδας, αλλά στην ουσία πρόκειται για διαφημίσεις τύπου banner, παρά το γεγονός ότι διαφέρουν λόγω του μεγέθους τους. Είναι πολύ εύκολο να προστεθούν σε μια ιστοσελίδα και αποτελούν έναν καλό σύνδεσμο με την επιχείρηση του διαφημιζόμενου, καθώς και το μέγεθός τους είναι διακριτικό και είναι φιλικό προς το χρήστη. Ένα button μπορεί να τοποθετηθεί στη σελίδα του διαφημιζόμενου, σε μία διαφορετική σελίδα ή ακόμα να περιέλθει σε ένα διαφημιστικό e-mail (Zeff και Aronson, 1999).

Skyscraper (Κυλιόμενοι ψηφιακοί ουρανοξύστες)

Τα Skyscrapers, που σημαίνουν ουρανοξύστες, είναι διαδικτυακές διαφημίσεις που ονομάστηκαν έτσι είναι γιατί αποτελούν πολύ ψηλά εικονίδια, τα οποία μπορούν να φτάσουν τα 500 με 800 πίξελ (pixel) οθόνης. Βρίσκονται συνήθως στη δεξιά στήλη του site και είναι συνεχώς ορατά από τον χρήστη. Είναι λεπτή και κάθετη, πολύ ψηλή διαφήμιση και δεν εμποδίζει την ανάγνωση του κειμένου γεγονός που την κάνει ιδιαίτερα αγαπητή. Μια ελκυστική skyscraper διαφήμιση με καθαρή μορφή μπορεί να αποδώσει αποτελεσματικά το μήνυμα στον δέκτη (Zeff και Aronson, 1999). Μερικές φορές, οι διαφημίσεις αυτές αποκαλούνται και skyscraper banner, καθώς υπάρχουν ορισμένες που μιμούνται την μορφή ενός banner χρησιμοποιώντας έναν συνδυασμό διαφημιστικών «κουμπιών» και κειμένου. Το πλεονέκτημα αυτής της μορφής διαδικτυακής

διαφήμισης είναι ότι δεν ενοχλεί την ανάγνωση του χρήστη, ενώ παράλληλα καθώς ο χρήστης κατεβαίνει και «κυλάει» στην ιστοσελίδα, η διαφήμιση αυτή συνεχίζει να υπάρχει, λόγω του μεγέθους της. Από την άλλη, ωστόσο, μειονέκτημα είναι ότι δεν είναι κατάλληλη για όλες τις ιστοσελίδες, εξαιτίας του μεγάλου μεγέθους της αλλά και του υψηλού της κόστους (Zeff και Aronson, 1999).

Pop-up ads

Οι διαφημίσεις Pop-up αποτελούνται από μηνύματα τα οποία εμφανίζονται στην οθόνη του χρήστη με ξαφνικό τρόπο, τη στιγμή που βρίσκεται σε μια ιστοσελίδα. Τα μηνύματα αυτά πολλές φορές τα διαγράφει ο χρήστης πριν προλάβουν να εμφανιστούν ολοκληρωμένα στην οθόνη του, καθώς νιώθει ότι τον ενοχλούν. Τα pop-ups εμφανίζονται ως ένα ανεξάρτητο παράθυρο, το οποίο δεν έχει καθορισμένο μέγεθος και έχει μόνο τις εντολές «Κλείσιμο», «Ελαχιστοποίηση» και «Μεγέθυνση». Από την άλλη μεριά υπάρχουν και τα pop – under μηνύματα, τα οποία ανοίγουν πίσω από τη σελίδα στην οποία βρίσκεται ο χρήστης και τα εντοπίζει μόνο αν κλείσει τον ιστότοπο. Αυτά δεν είναι τόσο ενοχλητικά όσο τα pop up και προλαβαίνουν να «ανοίξουν» πριν προλάβει να τα διαγράψει ο δέκτης. Επομένως, ο χρήστης εκτίθεται σε αυτά και αν τον ενδιαφέρει το διαφημιζόμενο προϊόν πιθανώς και να ενδιαφερθεί να ενημερωθεί εκτενέστερα για αυτό (Βλάχος 2004).

Ηλεκτρονικά κουπόνια (Online group buying)

Είναι μια νέα και καινοτόμα ιδέα και κατακτά συνεχώς την προσοχή και το ενδιαφέρον των διαδικτυακών χρηστών. Η πιο συνηθισμένη του μορφή είναι το one deal a day, σύμφωνα με το οποίο προσφέρεται κάποιο προϊόν ή υπηρεσία σε τιμές 50% έως και 90% φθηνότερες από τις αρχικές για ένα εικοσιτετράωρο μέσα από μια ιστοσελίδα η οποία παρουσιάζει αποκλειστικά τέτοιου είδους δικτυακές προσφορές. Εάν κάποιος εγγραφεί στο αντίστοιχο site κατακλύζεται καθημερινά από τις διάφορες προσφορές τις ημέρας. Οι επιχειρήσεις με αυτό τον τρόπο παρέχουν μεγάλες εκπτώσεις και επωφελούνται από το μαζικό ενδιαφέρον των καταναλωτών καθώς αυτοί αναμένουν συνεχώς τις προσφορές που θα προκύψουν. Οι επιχειρήσεις που χρησιμοποιούν αυτόν τον τρόπο πώλησης για να διατηρήσουν τους πελάτες τους, τους ενημερώνουν με διάφορους τρόπους όπως το e-mail και τα RSS (Sigala, 2005).

Άλλες μορφές διαδικτυακής/ψηφιακής διαφήμισης

Άλλες μορφές της online διαφήμισης μπορεί να είναι τα social media, η push τεχνολογία, το e- mail marketing και τα online advertorials. Αυτές οι μορφές διαφήμισης ουσιαστικά κάνουν χρήση τεχνολογιών που αναπτύσσονται στα πλαίσια του διαδικτύου, προκειμένου να επιτύχουν την

παράδοση διαφημιστικών μηνυμάτων. Οι τεχνολογίες αυτές δεν έχουν αναπτυχθεί για να εξυπηρετήσουν αποκλειστικά την διαφημιστική επικοινωνία, όμως η online διαφήμιση βρίσκει σε αυτές σημαντική εφαρμογή.

Οι διαφημίσεις των *social media* περιλαμβάνουν τα blogs, προγράμματα Chat & Instant messaging, και websites κοινωνικής δικτύωσης όπως το Facebook ή το Twitter.

Η διαφήμιση μέσω *e-mail marketing* ουσιαστικά αξιοποιεί την ανυπαρξία κόστους για την μαζική αποστολή χιλιάδων emails σε λίστες αποδεκτών. Διακρίνουμε δύο βασικές κατηγορίες λιστών με αποδέκτες α) τις λίστες τις οποίες οι εγγεγραμμένοι έχουν προηγουμένως ερωτηθεί αν επιθυμούν να λαμβάνουν e-mail και έχουν απαντήσει θετικά. β) τις λίστες τις οποίες οι εγγεγραμμένοι έχουν ερωτηθεί, αλλά έχουν εντοπισθεί από πηγές στις οποίες έχει ελεύθερη πρόσβαση όποιος το επιθυμεί. (Ευαγγελάτος Α., 2010).

1.4 Τόποι προβολής της διαδικτυακής διαφήμισης

Σύμφωνα με τους Πασχόπουλο και Σκαλτσά (2009), υπάρχουν τέσσερις κατηγορίες «τόπων» Διαφήμισης μιας επιχείρησης στο Διαδίκτυο. Αυτές είναι αρχικά οι χώροι μαζικής προέλευσης, οι τόποι διάθεσης περιεχομένου, καθώς και οι εμπορικοί και υβριδικόι τόποι. Πιο αναλυτικά:

- **Χώροι μαζικής προέλευσης:**

1. *Σελίδες παροχέων υπηρεσιών δικτύου:* Αρχικά, ο πάροχος φρόντιζε μέσω του λογισμικού εγκατάστασης που έδινε στους πελάτες του, να βλέπουν πρώτα τις σελίδες του ίδιου. Συνεπάγεται ότι η αρχική σελίδα τραβούσε αρκετά βλέμματα. Σήμερα όμως το σκηνικό έχει αλλάξει και οι σημερινοί χρήστες έχουν περισσότερη εμπειρία και δεν χρειάζονται τον εκάστοτε πάροχο για το ξεκίνημα τους στο Διαδίκτυο. Καθ' αυτό τον τρόπο συνήθως διαφημίζονται τράπεζες, ταξιδιωτικά γραφεία, ασφαλιστικές εταιρίες, κατασκευαστές υπολογιστών κα. (πχ. www.hol.gr)
2. *Μηχανές αναζήτησης και Ηλεκτρονικοί κατάλογοι:* Πρόκειται για τους πιο γνωστούς χώρους μαζικής προβολής. Η διαφορά τους έγκειται στο γεγονός ότι οι Ηλεκτρονικοί κατάλογοι έχουν κατηγοριοποιήσει τις πληροφορίες τους. Επίσης, προσφέρονται για τόσο για μαζική, όσο και για στοχευόμενη προώθηση καθώς δίνουν τη δυνατότητα προβολής banners, βάση λέξεων κλειδιών που χρησιμοποιεί ο χρήστης. (πχ. www.in.gr)
3. *Supersites:* Τα supersites ειδικεύονται σε συγκεκριμένα θέματα ή σε συγκεκριμένες αγορές. Έτσι αν μια εταιρία δραστηριοποιείται σε κάποια αγορά που αντιπροσωπεύεται από ένα ή περισσότερα supersites, η καταχώρηση της σε αυτά κρίνεται απαραίτητη. (πχ. www.agriculture.com)

4. *Ιδεατά εκθεσιακά κέντρα*: Τα ιδεατά εκθεσιακά κέντρα λειτουργούν όπως και τα πραγματικά, με τη διαφορά όμως ότι οι ενδιαφερόμενοι δεν επιβαρύνονται με το κόστος μεταβίβασης. (πχ. www.exponet.gr) (Παπαδοπούλου, 2009)

• **Χώροι Διάθεσης Περιεχομένου**

1. *Εφημερίδες*: Οι εφημερίδες στο Διαδίκτυο λειτουργούν διαφημιστικά, όπως εξάλλου και εκτός Διαδικτύου, πολλές φορές και καλύτερα αφού επιτρέπουν τόσο την παραπομπή στον διαδικτυακό τόπο της εταιρίας, όσο και την αλληλεπίδραση με τον καταναλωτή. Χρησιμοποιούνται για ανακοινώσεις, Διαφήμιση, δελτία τύπου, χορηγίες, ενώ αναλόγως την αρθρογραφία τους προσφέρονται για μαζική ή στοχευόμενη προβολή. (πχ. www.nauftemporiki.gr)

2. *Περιοδικά δικτύου*: Τα περιοδικά Διαδικτύου λειτουργούν διαφημιστικά όπως και οι εφημερίδες. Πολλά από αυτά εκδίδονται μοναχά ηλεκτρονικά και καλύπτουν όσα θέματα καλύπτουν και τα περιοδικά στα περίπτερα. Όμως, με την όλο και περισσότερη χρήση των blogs, έχουν αρχίσει και χάνουν το κοινό τους. (πχ. www.go-ezines.com)

3. *Πύλες (Portals)*: Οι πύλες αποτελούν την εξέλιξη των μηχανών αναζήτησης, καθώς οι χρήστες μπορούν να βρουν πληροφορίες για όποιο θέμα τους ενδιαφέρει. Επίσης, οι πληροφορίες αυτές βρίσκονται είτε στο site της πύλης, είτε σε άλλα sites, στα οποία μεταβαίνει ο επισκέπτης μέσω παραπομπών. (πχ. www.pathfinder.com)

4. *Θεματικά Sites (Vortals)*: Τα sites αυτά μπορούν να χαρακτηριστούν ως μηχανές αναζήτησης για κάποιο συγκεκριμένο θέμα που αναλύεται σε βάθος. Γίνονται όλο και πιο δημοφιλής καθώς προσφέρουν τη μεγαλύτερη στόχευση σε συγκεκριμένο κοινό. (πχ. www.iatronet.gr)

5. *Blogs*: Πρόκειται για σελίδες- ημερολόγια ανοιχτά για το κοινό, όπου ο συγγραφέας μοιράζεται τα ενδιαφέροντα του. Μία τυπική σελίδα ενός blog έχει παραπομπές σε άλλα sites, συνοδευόμενες από λίγες γραμμές που επεξηγούν το θέμα της παραπομπής. Επίσης, χρησιμοποιούνται τελευταία από τις μεγάλες εταιρίες ως μέσο προβολής, καθώς τοποθετούν σε αυτά θέματα που αφορούν τα προϊόντα τους. Παράλληλα, μπορεί τα στελέχη της εταιρίας να δημιουργήσουν ο καθένας το δικό του blog που θα περιλαμβάνει παραπομπές προς το site της εταιρίας. Τέλος, ενδιαφέρον παρουσιάζει η χρήση των blogs από τους spammers. Ως γνωστόν όταν ένα site έχει πολλές παραπομπές, αυτόματα ανεβαίνει στην ιεραρχία των μηχανών αναζήτησης. Οι spammers χρησιμοποιώντας τα δωρεάν προγράμματα κατασκευής blogs από το Google ή το Yahoo, δημιουργούν χιλιάδες blogs με παραπομπές στο διαδικτυακό τους τόπο, ανεβαίνοντας έτσι στα αποτελέσματα αναζητήσεων των χρηστών. (πχ. <http://blogsearch.google.com>)

6. *Παράθυρα Video on demand*: Πρόκειται για ένα όλο και πιο δημοφιλή τόπο διαφήμισεων, καθώς αυξάνονται οι συνδέσεις ADSL. Η Διαφήμιση εδώ είναι πολύ στοχευόμενη ανάλογα με το βίντεο που θέλει να παρακολουθήσει ο χρήστης.
 7. *Media Players*: Η φιλοσοφία εδώ είναι παρόμοια με το video on demand, μόνο που εδώ γίνεται λόγος για το παράθυρο του προγράμματος αναπαραγωγής μουσικής, καθώς ο χρήστης το χρησιμοποιεί για να ακούσει από online ραδιοφωνικούς σταθμούς μουσική. Η συνήθης μορφή Διαφήμισης αφορά κάποιο άλμπουμ που πωλείται σε κάποιο διαδικτυακό κατάστημα.
- **Εμπορικοί Τόποι**: Οι διάφοροι εμπορικοί τόποι όπως είναι η Amazon για παράδειγμα, προσφέρουν ένα μεγάλο πλεονέκτημα στις εταιρίες που θέλουν να διαφημιστούν στις σελίδες τους, καθώς παρέχουν αξιόπιστες πληροφορίες για το ιστορικό των αγορών των καταναλωτών. και επειδή οι περισσότεροι πελάτες δίνουν ένα κωδικό πρόσβασης από τη δεύτερη φορά που θα μπουν στο site, είναι γνωστοί (όνομα, διεύθυνση, τηλέφωνο κα). Έτσι χάρη στην τεχνολογία μπορούν να κατευθύνουν προς αυτούς διαφημίσεις σύμφωνα με τα στοιχεία τους αλλά και τις αγορές που έχουν πραγματοποιήσει. (www.lci.fr)
 - **Υβριδικόί Τόποι**: Οι τόποι αυτοί παρέχουν περιεχόμενο και παράλληλα υποστηρίζουν κάποιο από τα προϊόντα τους. Τα οφέλη από τη Διαφήμιση σε υβριδικούς τόπους είναι παρόμοια με αυτά των τόπων διάθεσης περιεχομένου αλλά και των εμπορικών τόπων. (πχ. www.gfn.com) (Παπαδοπούλου, 2009)

1.5 Βασικά χαρακτηριστικά και ιδιαιτερότητες της ψηφιακής διαφήμισης

Σύμφωνα με θεωρίες, τα βασικά χαρακτηριστικά του περιβάλλοντος στην online διαφήμιση, που τη διαφοροποιούν από την παραδοσιακή, είναι τα εξής.

Κατά αρχήν αυτό που ονομάζεται «*addressability*», δηλαδή η δυνατότητα που δίνεται τα στελέχη μάρκετινγκ να εντοπίσουν τις προτιμήσεις των καταναλωτών πριν ακόμη αυτοί προχωρήσουν σε κάποια αγορά. Η τεχνολογία καθιστά δυνατό οι επισκέπτες μιας ιστοσελίδας να δώσουν πληροφορίες αναφορικά με τις προτιμήσεις τους πριν να πραγματοποιήσουν κάποια αγορά. Πολλοί διαδικτυακοί τόποι ενθαρρύνουν τους επισκέπτες τους να εγγραφούν (log-on) ώστε να αποκτήσουν πρόσβαση σε κάποιες παροχές της ιστοσελίδας. Μάλιστα σε μερικά η εγγραφή αποτελεί απαραίτητη προϋπόθεση για τον επισκέπτη. Οι φόρμες εγγραφής ζητούν κατά κύριο λόγο βασικές πληροφορίες όπως όνομα, επώνυμο ηλικία και κάποια στοιχεία επικοινωνίας έτσι ώστε να είναι εφικτό να δημιουργηθεί το προφίλ των επισκεπτών της ιστοσελίδας και να μπορούν έτσι πιο εύκολα να τους προσεγγίσουν τα στελέχη μάρκετινγκ. Ορισμένοι διαδικτυακοί τόποι μάλιστα

διοργανώνουν και διαγωνισμούς προσφέροντας δώρα προκειμένου να προσελκύσουν περισσότερους επισκέπτες (Möller, K., Halinen, A. 2000).

Ενώ κάποιες φορές προχωρούν ακόμα και σε συγκεκριμένες έρευνες προκειμένου να μάθουν περισσότερες πληροφορίες για τους πιθανούς καταναλωτές, οι οποίοι επισκέπτονται τον διαδικτυακό τόπο της εταιρίας. Τα στελέχη μάρκετινγκ είχαν και πριν την ύπαρξη του διαδικτύου την δυνατότητα να εντοπίσουν τους καταναλωτές που τους ενδιέφεραν. Πλήρωναν και αποκτούσαν πρόσβαση σε τράπεζες πληροφοριών ή χρησιμοποιούσαν το προσωπικό των πωλήσεων για τη δημιουργία βάσεων δεδομένων. Όμως αυτοί οι τρόποι συλλογής στοιχείων είναι ακριβοί και συνήθως χρησιμοποιούνταν μόνο για ακριβά προϊόντα που δικαιολογούσαν αυτά τα έξοδα. Με την χρήση όμως του διαδικτύου το κόστος διεξαγωγής ερευνών μειώθηκε σημαντικά. Επίσης, έγινε δυνατή η έρευνα καταναλωτών σε μεγάλο γεωγραφικό εύρος (Coviello, N., Milley, R., Marcolin, B. 2001).

Η χρήση του διαδικτύου για την συλλογή πληροφοριών που αφορούν τους καταναλωτές προκαλεί ποικίλες αντιδράσεις. Με την χρήση ειδικού λογισμικού στην περίπτωση που κάποιος επισκέπτης επιλέξει μια ιστοσελίδα, εγκαθίσταται στον υπολογιστή του ένα “*cookie*”, δηλαδή ένα ηλεκτρονικό ίχνος που επιτρέπει να γνωρίζουν πόσο συχνά επισκέπτεται ο χρήστης την ιστοσελίδα, πόσο χρόνο αφιερώνει και άλλες παρόμοιες πληροφορίες. Το όφελος που προκύπτει μέσα από αυτή τη διαδικασία για τους καταναλωτές είναι ότι έχουν την δυνατότητα να εξυπηρετούνται γρηγορότερα καθώς οι υπεύθυνοι προσαρμόζονται ανάλογα με τις ανάγκες των καταναλωτών. Όμως πολλοί καταναλωτές αντιδρούν καθώς αισθάνονται ότι παραβιάζονται προσωπικά τους δεδομένα καθώς νιώθουν ότι παρακολουθούνται συνεχώς οι κινήσεις τους πράγμα που δεν απέχει από την πραγματικότητα. Συνεπώς πολύ σημαντικός είναι ο τρόπος με τον οποίο επιλέγεται να χρησιμοποιηθούν τα στοιχεία που συλλέγονται. Αυτό άλλωστε είναι ένα θέμα που αφορά την ηθική στο μάρκετινγκ και ειδικότερα στην ηλεκτρονική του μορφή. Επίσης, χαρακτηριστική είναι και η αμφίδρομη σχέση που επιτρέπει στους καταναλωτές να εκφράσουν τις ανάγκες τους και τις επιθυμίες τους απευθείας στις επιχειρήσεις (Coviello, N., Milley, R., Marcolin, B. 2001).

Η δυνατότητα αυτή ονομάζεται «*interactivity*», δηλαδή αμφίδρομη επικοινωνία. Πρακτικά η ύπαρξη αυτού σημαίνει ότι οι μπορεί να υπάρξει επικοινωνία σε πραγματικό χρόνο με τους καταναλωτές. Φυσικά αυτή την δυνατότητα πριν την εμφάνιση του διαδικτύου την είχαν μόνο οι πωλητές των επιχειρήσεων, με πολύ υψηλό κόστος βέβαια.

Το παραδοσιακό Μάρκετινγκ λειτουργεί με αξίωμα ότι μια ομάδα φτιάχνει, κάποιιο άλλοι απολαμβάνουν. Η επικοινωνία είναι μονόδρομη και είναι πάντα υπέρ του κατασκευαστή. Στο καθεστώς του ηλεκτρονικού εμπορίου, χρησιμοποιείται η ενσωματωμένη επικοινωνία, η οποία έχει συνοχή και προσαρμόζεται για κάθε άτομο. Παράλληλα η γεωγραφική κάλυψη από τους πωλητές

ήταν περιορισμένη ενώ τώρα παρέχεται η δυνατότητα κάλυψης ακόμα και των πιο απομακρυσμένων περιοχών.

Ένα τρίτο χαρακτηριστικό που αξίζει να αναφέρουμε είναι η λεγόμενη μνήμη, «*memory*». Αναφέρεται δηλαδή στην δυνατότητα πρόσβασης των επιχειρήσεων σε ηλεκτρονικές τράπεζες πληροφοριών, οι οποίες περιέχουν στοιχεία για τους καταναλωτές που ενδιαφέρουν την επιχείρηση («*target group*»), καθώς και ιστορικά των αγορών που έχουν πραγματοποιήσει. Κατόπιν επεξεργάζονται τα στοιχεία αυτά σε πραγματικό χρόνο και δημιουργούν αποτελεσματικά μίγματα μάρκετινγκ για τον κάθε καταναλωτή ξεχωριστά. Τα μίγματα αυτά προϋπήρχαν, η εξέλιξη όμως της τεχνολογίας έδωσε την δυνατότητα για τρομακτική επιτάχυνση της διαδικασίας αλλά και για την άμεση προσαρμογή του μίγματος, ώστε να αγγίξει όσο το δυνατόν πιο αποτελεσματικά τον κάθε καταναλωτή με βάση τις δικές του ανάγκες και επιθυμίες (Coviello, N., Milley, R., Marcolin, B. 2001).

Άλλο χαρακτηριστικό θεωρείται και ο έλεγχος «*control*». Οι επισκέπτες ελέγχουν απόλυτα τις πληροφορίες στις οποίες εκτίθενται καθώς και την συχνότητα και την διάρκεια της έκθεσης τους στις πληροφορίες αυτές. Το διαδίκτυο συχνά χαρακτηρίζεται ως μέσο στο οποίο εφαρμόζεται η μέθοδος «*pull*», καθώς οι επιχειρήσεις που προβάλλονται μέσα από αυτό δεν έχουν την δυνατότητα να ελέγξουν στην πλειοψηφία των περιπτώσεων τις πληροφορίες που τελικά φτάνουν στους καταναλωτές. Σε αντίθεση με άλλα μέσα προβολής, όπως η τηλεόραση και το ραδιόφωνο, στο net οι επισκέπτες διατηρούν τον έλεγχο των πληροφοριών που θα λάβουν καθώς αυτοί αποφασίζουν τους τόπους που θα επισκεφτούν.

Ένα ακόμη χαρακτηριστικό αποτελεί η εύκολη πρόσβαση «*accessibility*». Με τον όρο αυτό κάποιος αναφέρεται στον εξαιρετικά μεγάλο όγκο πληροφοριών που είναι εύκολα προσβάσιμες και προσιτές στους χρήστες του διαδικτύου. Οι καταναλωτές έχουν την δυνατότητα να αναζητούν και να λαμβάνουν πολλές πληροφορίες πριν από την αγορά ενός προϊόντος, με αποτέλεσμα να είναι καλύτερα ενημερωμένοι πριν πραγματοποιήσουν την αγορά. Όμως η εύκολη πρόσβαση για τους καταναλωτές έχει σαν αποτέλεσμα την δραματική αύξηση του ανταγωνισμού όσον αφορά τα προϊόντα και τις υπηρεσίες που θα επιλέξουν μέσα από το διαδίκτυο. Γίνεται ολοένα και πιο δύσκολο μια ιστοσελίδα να διατηρεί το ενδιαφέρον των επισκεπτών με αποτέλεσμα τα στελέχη μάρκετινγκ να πρέπει να γίνονται ολοένα και πιο δημιουργικοί για να καταφέρουν να τους προσεγγίσουν.

Μια ακόμη απόρροια της εύκολης πρόσβασης είναι ότι φαίνεται να ενισχύονται τα γνωστά *Brand Names*. Οι καταναλωτές δυσκολεύονται όλο και περισσότερο να ξεχωρίσουν μεταξύ των αμέτρητων μαρκών που υπάρχουν και έτσι καταφεύγουν στην αγορά ευρέως διαδεδομένων μαρκών οι οποίες προσφέρουν κάποια εγγύηση ποιότητας. Το τελευταίο χαρακτηριστικό που διαφοροποιεί

το e-Μάρκετινγκ είναι η ψηφιοποίηση ή «digitaliazation», δηλαδή η δυνατότητα που προσφέρει το διαδίκτυο για την ψηφιακή παρουσίαση ενός προϊόντος ή και των ωφελειών που προκύπτουν από την χρήση του. Το χαρακτηριστικό αυτό επιτρέπει την διανομή, τη προβολή και τη πώληση των προϊόντων χωρίς να είναι αναγκαία η φυσική τους παρουσία.

1.6 Πλεονεκτήματα και μειονεκτήματα της ψηφιακής διαφήμισης

Η διαφήμιση μέσω του διαδικτύου παρέχει παγκόσμια κάλυψη των διαφημιστικών μηνυμάτων, δίνοντας την ευκαιρία στους διαφημιζόμενους να προβάλλουν τα προϊόντα και τις υπηρεσίες τους σε ένα μεγάλο μέρος καταναλωτών, ανεξάρτητα από τη γεωγραφική τους διασπορά, σε νέες αγορές πολύ πιο εύκολα από ότι με τα παραδοσιακά μέσα διαφήμισης, αυξάνοντας τα μερίδια αγοράς τους (Ling et al., 1999; Vadell, 2005). Ένα ακόμη πλεονέκτημα του διαδικτύου ως μέσο διαφήμισης είναι η δημιουργία κοινοτήτων. Το διαδίκτυο δίνει δυνατότητες δημιουργίας ψηφιακών κοινοτήτων. Οι χρήστες του, μπορούν να ενταχτούν σε κοινότητες ανάλογα με τα ενδιαφέροντα τους. Στόχος των επιχειρήσεων είναι να μπορέσουν να τους οργανώσουν, μέσω του διαδικτύου, σε συλλόγους και κοινότητες που έχουν σχέση με τα προϊόντα τους, έτσι ώστε να ενημερώνονται για τις προσφορές, για τα νέα προϊόντα ή ακόμα και να συζητούν με άλλους πελάτες για τις εμπειρίες σχετικά με το προϊόν. Ειδικότερα για το τουριστικό προϊόν και τις ιδιαιτερότητες που το χαρακτηρίζουν, οι διαδικτυακές κοινότητες αποτελούν τόπο όπου ο εν δυνάμει τουρίστας έχει τη δυνατότητα να ανταλλάξει απόψεις με χρήστες που ίσως έχουν αγοράσει το τουριστικό πακέτο και έμειναν ευχαριστημένοι. Οι πελάτες αυτοί αποτελούν την καλύτερη διαφήμιση μιας επιχείρησης (Φρίγκα, 2005).

Επίσης, οι επιχειρήσεις έχουν τη δυνατότητα πετυχημένου targeting γιατί όπως αναφέρθηκε και πιο πάνω το κοινό του διαδικτύου έχει συγκεκριμένα δημογραφικά χαρακτηριστικά καθώς και τη δυνατότητα να οργανώνεται σε κοινότητες κοινών ενδιαφερόντων. Επιπλέον, η δόμηση της πληροφορίας με υπερσυνδέσμους δίνει τη δυνατότητα στις επιχειρήσεις να προσφέρουν δωρεάν πολλές πληροφορίες σε πραγματικό χρόνο, ενώ στους χρήστες δίνει τη δυνατότητα να επιλέγουν τις πληροφορίες που επιθυμούν. Η εύκολη και ταχύτατη ανάκτηση πληροφοριών είναι ένα ακόμη πλεονέκτημα. Πιο συγκεκριμένα, με τα κατάλληλα προγράμματα μπορεί να ανακτηθεί ένας μεγάλος όγκος πληροφοριών σε πολύ μικρό χρονικό διάστημα, να επεξεργασθεί με τα ανάλογα φίλτρα τα οποία μπορούν να περιορίσουν τα αποτελέσματα με διάφορους τρόπους, όπως για παράδειγμα γεωγραφικούς περιορισμούς που παρέχουν οι μηχανές αναζήτησης, εξειδικεύοντας και οδηγώντας το χρήστη στη ζητούμενη πληροφορία (Βλαχοπούλου, 2003).

Οι επιχειρήσεις καθώς και το κοινό του διαδικτύου δεν λειτουργούν παθητικά αλλά δυναμικά.

Από την μεριά των επιχειρήσεων στον ίδιο χώρο και την ίδια στιγμή μπορεί να γίνει αποστολή προωθητικού υλικού, ενώ από τη μεριά των καταναλωτών μπορεί να γίνει ανάδραση αλλά και ολοκλήρωση της συναλλαγής. Επίσης, στο διαδίκτυο η διαφήμιση δε σταματά ποτέ, αλλά προβάλλεται ακατάπαυστα παρέχοντας στη διαφημιζόμενη επιχείρηση εκτεταμένο χρόνο διαφημιστικής παρουσίας (Vadell, 2005).

Παρόλα αυτά, η διαδικτυακή διαφήμιση έχει και ορισμένα σημαντικά μειονεκτήματα. Το πρώτο μειονέκτημα είναι ότι η ανταπόκριση που θα έχει το διαφημιστικό μήνυμα εξαρτάται από την πρόσβαση των καταναλωτών στον υπολογιστή και το διαδίκτυο. Κατά συνέπεια, η συνδεσιμότητα μιας περιοχής, καθώς και το επίπεδο δομών των τηλεπικοινωνιών της, καθορίζει σε πολύ σημαντικό βαθμό την πρόσβαση των καταναλωτών στο διαδίκτυο και τη διαδικτυακή διαφήμιση (Ling et al., 1999). Επιπρόσθετα, καθώς υπάρχει μεγαλύτερη ευκολία και δυνατότητα εξαπάτησης, το διαδικτυακό κοινό αντιμετωπίζει με δυσπιστία τη διαδικτυακή διαφήμιση και γενικότερα τις online αγορές.

ΚΕΦΑΛΑΙΟ 2: ΨΗΦΙΑΚΗ ΔΙΑΦΗΜΙΣΗ ΚΑΙ ΚΙΝΗΤΕΣ ΣΥΣΚΕΥΕΣ

2.1 Κινητές /Φορητές Συσκευές

Ως κινητή συσκευή ορίζεται κάθε φορητή ηλεκτρονική συσκευή χειρός. Διανύοντας την δεύτερη δεκαετία του 21ου αιώνα, διακρίνουμε πολύ ισχυρές κινητές συσκευές, με χαρακτηριστικά που μέχρι πρόσφατα ήταν διαθέσιμα μόνο σε υπολογιστές. Μιλώντας όμως για κινητές συσκευές, δεν αναφερόμαστε μόνο στα έξυπνα τηλέφωνα, τα smartphones. Συγκεκριμένα, στη κατηγορία των κινητών συσκευών περιλαμβάνονται οι ταμπλέτες, οι φορητές συσκευές αναπαραγωγής πολυμέσων, οι φορητές παιχνιδομηχανές, οι προσωπικές συσκευές πλοήγησης, οι ψηφιακοί προσωπικοί οδηγοί, τα υπολογιστικά ρολόγια, τα έξυπνα ρολόγια, τα head-mounted displays, οι βομβητές, ακόμα και οι ψηφιακές κάμερες και βιντεοκάμερες. (<http://www.gcflearnfree.org/computerbasics>)

Φορητές Συσκευές Αναπαραγωγής Πολυμέσων

Οι φορητές συσκευές αναπαραγωγής μουσικής (Portable Media Players) εμφανίστηκαν το 1954 με το Regency TR-1, το πρώτο τρανζίστορ ραδιόφωνο. Το 1978 στην Ιαπωνία, παρουσιάστηκε το πρώτο «Walkman» από τη Sony. (<http://en.wikipedia.org/wiki/Walkman>)

Την δεκαετία του 1980, το Walkman όρισε την αγορά των φορητών συσκευών αναπαραγωγής κασετών. Το 1984, η Sony πρωτοπορεί για άλλη μια φορά με τη κυκλοφορία του «Discman», τη πρώτη ψηφιακή φορητή συσκευή που έπαιζε μουσική από CD. Τα CD players επικράτησαν μέχρι να πάρουν τη σκυτάλη τα MP3 Players, συσκευές που υποστηρίζουν MP3 (MPEG-2 Audio Layer III - 3ο Επίπεδο Ήχου MPEG-1) και άλλα αντίστοιχα πρότυπα ψηφιακής κωδικοποίησης ήχου. Περνώντας στη ψηφιακή εποχή, εμφανίστηκαν συσκευές με υποστήριξη αναπαραγωγής πολυμέσων, πέρα από τον ήχο. Το 1998, η Panasonic παρουσίασε το πρώτο φορητό DVD Player ενώ ένα χρόνο αργότερα κυκλοφόρησε στο εμπόριο η πρώτη συσκευή με σκληρό δίσκο, το Personal Jukebox. (http://en.wikipedia.org/wiki/Portable_DVD_player.) Με την εμφάνιση συσκευών με σκληρό δίσκο ή μνήμη flash, οι χρήστες απέκτησαν την δυνατότητα επιλογής των αρχείων που θα μετέφεραν στη συσκευή τους, μέσω του υπολογιστή τους. Το 2001 με την εμφάνιση του πρώτου «iPod» της Apple, με σκληρό δίσκο χωρητικότητας 5GB, επαναφορτιζόμενη μπαταρία και πρωτοποριακό για την εποχή σχεδιασμό. Τα χρόνια που ακολούθησαν, οι δυνατότητες των φορητών συσκευών αναπαραγωγής πολυμέσων εμφανίστηκαν σε κινητά και φορητές παιχνιδομηχανές. (<http://www.ebay.com/gds/A-Complete-History-of-Portable-Music-Players-/10000000177628958/g.html>.)

Φτάνοντας στην εποχή των smartphones, συσκευές όπως MP3 Players και φορητές παιχνιδομηχανές είναι διαθέσιμες αλλά έχουν χάσει την μοναδικότητα και τη χρησιμότητα τους,

λόγω της υπεροχής των smartphones.

Φορητές Παιχνιδομηχανές

Οι φορητές παιχνιδομηχανές (Handheld Game Consoles) εμφανίστηκαν το 1977 με τη κυκλοφορία του «Mattel Auto Race». (http://en.wikipedia.org/wiki/Mattel_Auto_Race.) Η Mattel Electronics, κυκλοφόρησε μια κινητή συσκευή η οποία ήταν σχεδιασμένη για το παιχνίδι Auto Race. Ακολούθησαν και άλλες αντίστοιχες κυκλοφορίες και η επιτυχία τους, οδήγησε στην κυκλοφορία παιχνιδομηχανών που υποστήριζαν παραπάνω από ένα παιχνίδι, πιο κοντά στη σημερινή τους μορφή.

Την πρώτη δεκαετία του 21ου αιώνα ακολούθησαν, με μεγάλη επιτυχία, κυκλοφορίες όπως το Game Boy Advance, το PlayStation Portable (PSP), το Nintendo DS κ.α. Την επόμενη δεκαετία, έχοντας φτάσει στην 8η γενιά ηλεκτρονικών παιχνιδιών, ήρθαν τα Nintendo 3DS και PlayStation Vita, ενώ κυκλοφόρησαν τα NVidia Shield και Sony Ericsson Xperia PLAY. (<http://handheld-game-console-review.toptenreviews.com>) Το Xperia Play είναι άλλη μια απόπειρα από εταιρία κινητής τηλεφωνίας να συνδυάσει τηλέφωνο και παιχνιδομηχανή. Συγκεκριμένα, η συσκευή είναι «PlayStation Certified», έχει δηλαδή πιστοποίηση και πρόσβαση στα παιχνίδια για κινητά της κονσόλας PlayStation.

Τα τελευταία χρόνια, με την έλευση των smartphones η μοναδικότητα των φορητών παιχνιδομηχανών έχει χαθεί. Πλέον τα τηλέφωνα με τα εξελιγμένα χαρακτηριστικά τους μπορούν να καλύψουν τις ανάγκες των παικτών με όλες τις λειτουργίες τους. Τα App Store, Google Play και Windows Store έχουν εκατοντάδες χιλιάδες διαθέσιμα παιχνίδια, πολλά από τα οποία έχουν αποκτήσει τεράστια δημοτικότητα. (<http://www.pocketgamer.biz/metrics/app-store/>.) Όπως είναι φυσικό, οι φορητές παιχνιδομηχανές είναι δύσκολο να ανταγωνιστούν τα smartphones. Σαν αποτέλεσμα, την ίδια ώρα που παρατηρείται άνθηση στον χώρο των ηλεκτρονικών παιχνιδιών και συγκεκριμένα στις εφαρμογές ηλεκτρονικών παιχνιδιών, οι φορητές παιχνιδομηχανές έχουν κατακόρυφη πτώση. (<http://wallstcheatsheet.com/technology/did-smartphones-kill-the-handheld-gaming-console.html/>)

Προσωπικές Συσκευές Πλοήγησης

Το Παγκόσμιο Σύστημα Στιγματοθέτησης, ή Θεσιθεσίας (Global Positioning System), γνωστό και ως GPS, αναπτύχθηκε το 1973 για στρατιωτικές εφαρμογές. Τη δεκαετία του 1980, η κυβέρνηση των Ηνωμένων Πολιτειών της Αμερικής το παρέδωσε για εμπορική χρήση.

Το GPS, είναι ένα σύστημα δορυφορικής πλοήγησης αποτελούμενο από ένα δίκτυο είκοσι τεσσάρων δορυφόρων τοποθετημένων σε τροχιά, εφοδιασμένων με ειδικές συσκευές εντοπισμού, οι οποίες ονομάζονται «πομποδέκτες GPS». (<http://www8.garmin.com/aboutGPS/>.) Η χρήση του

GPS γίνεται μέσω ειδικών φορητών συσκευών ή άλλων φορητών συσκευών με ενσωματωμένο δέκτη GPS, όπως τα PDAs, οι ψηφιακές φωτογραφικές μηχανές και τα smartphones. Στις συσκευές πλοήγησης για τα αυτοκίνητα συνήθως περιλαμβάνονται φωνητικές οδηγίες για την διευκόλυνση του οδηγού.

Ψηφιακοί Προσωπικοί Οδηγοί – PDAs

Ο Προσωπικός Ψηφιακός Οδηγός (Personal Digital Assistant) έκανε την εμφάνιση του στα μέσα έως τέλος της δεκαετίας του 1980. Οι πρώτες συσκευές ήταν τα Psion Organiser και Sharp Wizard με πρωταρχικό σκοπό της δημιουργίας τους την αντικατάσταση του ηλεκτρονικού υπολογιστή. Αυτές οι συσκευές περιλάμβαναν βασικά χαρακτηριστικά όπως ξυπνητήρι, ημερολόγιο, αριθμομηχανή, παιχνίδια και υπολογιστικά φύλλα.

Το PDA, συνδυάζει χαρακτηριστικά υπολογιστή, τηλεφώνου/φαξ, και προσωπικού διοργανωτή ημέρας. (<http://www.webopedia.com/TERM/P/PDA.html>.) Μπορεί επίσης να χρησιμοποιηθεί για την διαχείριση προσωπικών δεδομένων, όπως για διάφορες επαφές, συναντήσεις αλλά ακόμα και ως συσκευή εντοπισμού θέσης (GPS). Θεωρείται επέκταση του ηλεκτρονικού υπολογιστή και όχι η πλήρης αντικατάσταση του. Η συσκευή χρησιμοποιείται μέσω ενός ειδικού στυλό, για την γραφή πάνω σε αυτήν και οι χρήσεις της είναι η καταχώρηση και η ανάκτηση διάφορων πληροφοριών. Αύτη την συσκευή την χρησιμοποιούν πολλές μεγάλες εταιρείες για να βλέπουν πιο γρήγορα και ευκολότερα τα αποθέματα της επιχείρησης, καθώς και τις παραγγελίες των πελατών τους. Στην Ελλάδα τέτοιου είδους συσκευές συναντάμε σε διάφορα μαγαζιά εστίασης, σε καφετέριες, σε καταστήματα παιχνιδιών κ.α.

Smartwatches

Το Smartwatch (έξυπνο ρολόι) είναι ένα ηλεκτρονικό ρολόι με λειτουργίες πέρα από τον υπολογισμό της ώρας. Ενώ τα αρχικά μοντέλα εκτελούσαν βασικές εργασίες, όπως υπολογισμούς, μεταφράσεις και παιχνίδια, τα σύγχρονα smartwatches είναι ουσιαστικά φορητοί υπολογιστές. Πολλά από αυτά υποστηρίζουν κινητές εφαρμογές, ενώ ένας μικρότερος αριθμός μοντέλων υποστηρίζει λειτουργικό σύστημα ίδιο με των κινητών τηλεφώνων, αλλά και λειτουργίες όπως αναπαραγωγή πολυμέσων, αναπαραγωγή από FM ραδιόφωνο, ήχο και αρχεία βίντεο μέσω Bluetooth. Μερικά μοντέλα, που καλούνται «watchphones» (ρολόγια-τηλέφωνα), διαθέτουν την δυνατότητα να πραγματοποιούν και να απαντούν κλήσεις.

Τα Smartwatches, ανάλογα την εταιρία, διαθέτουν λειτουργίες όπως κάμερα, θερμόμετρο, πυξίδα, αριθμομηχανή, οθόνη αφής, GPS, χάρτη, SD κάρτες που αναγνωρίζονται ως συσκευή μαζικής αποθήκευσης από έναν υπολογιστή, επαναφορτιζόμενη μπαταρία κ.α. Κάποια έχουν τη

δυνατότητα επικοινωνίας με ασύρματα ακουστικά, κεφαλοτοποθετούμενες συσκευές, αντλία ινσουλίνης, μικρόφωνο, μόντεμ ή άλλες συσκευές. Όπως και οι άλλοι υπολογιστές έτσι και ένα Smartwatch μπορεί να συλλέξει πληροφορίες από εσωτερικές ή εξωτερικούς αισθητήρες. Μπορεί να ελέγχει ή να ανακτά δεδομένα από άλλα μέσα ή υπολογιστές. Μπορεί επίσης να υποστηρίξει ασύρματες τεχνολογίες όπως Bluetooth, Wi-Fi και GPS.

Τέλος, ορισμένες μικρότερες εταιρίες χρησιμοποιούν δημοφιλείς μεθόδους όπως το Kickstarter ή το Indiegogo για να χρηματοδοτήσουν τα σχέδιά τους.

Έξυπνα κινητά (Smartphones)

Σε αυτήν την κατηγορία ανήκουν οι πιο δημοφιλείς σήμερα στους καταναλωτές ηλεκτρονικές συσκευές, οι οποίες παρόλο που έκαναν την πρώτη τους εμφάνιση το 1994, έπρεπε να περάσει μια δεκαετία τουλάχιστον πριν αυτές ξεκινήσουν να γίνονται ευρέως αποδεκτές και (οικονομικά) προσιτές στο καταναλωτικό κοινό (Sager, 2012).

Όλα τα smartphones έχουν εγκατεστημένη ειδική για αυτές έκδοση λογισμικού, (Android, iOS, Windows 8CE, κλπ), διαθέτουν οθόνη αφής που κυμαίνεται συνήθως από 2,8 μέχρι 5 ίντσες, σύνδεση WiFi, συνδεσιμότητα 3G ή 4G, τεχνολογία Bluetooth, κάμερα υψηλής ανάλυσης και πολλαπλούς αισθητήρες. Επίσης μπορούν να διαχειρίζονται μεταξύ άλλων έγγραφα και υπολογιστικά φύλλα, πολλαπλούς λογαριασμούς ηλεκτρονικού ταχυδρομείου και φυλλομετρητές, ενώ μπορούν να χρησιμοποιηθούν παράλληλα και για την ψυχαγωγία του χρήστη (μουσική, βίντεο και παιχνίδια).

Το κυρίαρχο χαρακτηριστικό που κατέστησε τις συσκευές αυτές εξαιρετικά δημοφιλείς, αποτέλεσε η ανάπτυξη από τους προγραμματιστές χιλιάδων εφαρμογών, πολλές εκ των οποίων σχεδιάστηκαν αποκλειστικά για την εξυπηρέτηση των καταναλωτών σε κάθε ένα από τα τρία προαναφερθέντα στάδια του ταξιδιού τους.

Tablet (Υπολογιστής ταμπλέτα)

Οι έξυπνες συσκευές τύπου tablet έκαναν την εμφάνισή τους το 2000, όταν η εταιρεία Microsoft κατασκεύασε το πρώτο Tablet PC κυρίως για επιχειρηματική χρήση, το οποίο όμως είτε λόγω του αυξημένου κόστους, είτε λόγω της περιορισμένης για την εποχή τεχνολογίας που χρησιμοποιούσε, δεν γνώρισε την αναμενόμενη επιτυχία, σε αντίθεση με το πρώτο μοντέλο iPad της εταιρείας Apple, του οποίου οι πωλήσεις ξεπέρασαν κάθε προσδοκία, όταν λανσαρίστηκε στην αγορά το 2010.

Τα tablets, απετέλεσαν στην ουσία τη φυσική μετεξέλιξη των smartphones, διαθέτοντας αντίστοιχα χαρακτηριστικά και τεχνολογικές δυνατότητες, έχοντας όμως ως πλεονέκτημα το

μεγαλύτερο μέγεθος της οθόνης τους, αλλά χωρίς να ενσωματώνουν υποχρεωτικά τις λειτουργίες ενός κινητού τηλεφώνου.

Η αυξημένη αποδοχή που γνώρισαν τα tablets τα τελευταία χρόνια από τους καταναλωτές, έστρεψε γρήγορα το ενδιαφέρον των επαγγελματιών του μάρκετινγκ για τη μέγιστη δυνατή αξιοποίηση τους για ενέργειες mobile marketing. Για παράδειγμα, ερευνητές που σύγκριναν τον αριθμό των κλικ που έκαναν σε διαφημίσεις οι χρήστες smartphones και tablets, διαπίστωσαν πως οι πιθανότητες αποδοχής μιας διαφήμισης είναι πιο αυξημένες όταν αυτή προβάλλεται σε κάποιο tablet, σε σχέση με αυτές που προβάλλονται στα smartphones και κατά συνέπεια παρουσιάζουν συγκριτικά μεγαλύτερο ποσοστό μετατροπής (conversion rate) του απλού ενδιαφέροντος, σε κανονικές αγορές (Hopkins and Turner, 2012).

Phablets

Το κενό που διαπιστώθηκε στην αγορά σε σχέση με τις εξειδικευμένες απαιτήσεις μιας μερίδας καταναλωτών, ήρθε να καλύψει το phablet (από το συνδυασμό των λέξεων phone και tablet), το οποίο ουσιαστικά είναι ένα smartphone με μέγεθος μεταξύ 5 και 6.9 ιντσών, του οποίου τα χαρακτηριστικά το κατατάσσουν σε μια ενδιάμεση κατηγορία από πλευράς μεγέθους μεταξύ των smartphones (<5ίντσες) και των tablets (>7 ίντσες).

Η παγκόσμια πτώση των τιμών, ο ιδανικός συνδυασμός μεγέθους και χαρακτηριστικών, αλλά και η ύπαρξη μεγαλύτερης οθόνης για προβολή βίντεο, οδήγησε τόσο στη σταδιακή αύξηση της αποδοχής από το καταναλωτικό κοινό κάθε ηλικίας (ειδικά ανάμεσα στους επαγγελματίες), όσο και στη γεωμετρική αύξηση της παγκόσμιας παραγωγής των phablets.

Λοιπές κατηγορίες

Οι κατηγορίες που αναφέρθηκαν ανωτέρω, αποτελούν τις πιο συνηθισμένες σήμερα κατηγορίες έξυπνων φορητών συσκευών, χωρίς αυτό να σημαίνει πως είναι και οι μοναδικές.

Σχεδόν οποιαδήποτε ηλεκτρονική συσκευή η οποία αφενός ενσωματώνει κάποια από τις διαθέσιμες σήμερα εκδόσεις mobile λειτουργικών συστημάτων και αφετέρου διαθέτει οθόνη με την οποία ο χρήστης της μπορεί να αλληλεπιδράσει μαζί της, μπορεί να θεωρηθεί ως έξυπνη φορητή συσκευή, ικανή να χρησιμοποιηθεί για τους σκοπούς του τουριστικού μάρκετινγκ.

Τέτοιες συσκευές σήμερα, μπορούν να έχουν τη μορφή ενός ρολογιού χειρός (πχ το Smartwatch της Sony), μιας συσκευής αναπαραγωγής πολυμέσων (πχ το iPod της Apple), ή ακόμα και ψηφιακών φωτογραφικών μηχανών (πχ η Galaxy S4 Zoom της Samsung).

Τα πρόσφατα trends της τεχνολογίας έχουν φτάσει στο σημείο να ενσωματώσουν τις δυνατότητες μιας έξυπνης φορητής συσκευής ακόμα και σε γυαλιά οράσεως (πχ τα smartglasses της

Vuzix), προχωρώντας την έννοια της “φορητότητας” των συσκευών αυτών ένα βήμα παραπέρα και μετασχηματίζοντας την σε “φορετότητα”, ως ένα μικρό δείγμα αφενός των εξελίξεων που έρχονται και αφετέρου στην προσαρμογή της τεχνολογίας στις αγοραστικές ή ταξιδιωτικές συνήθειες των καταναλωτών.

2.2 Χαρακτηριστικά ψηφιακής διαφήμισης στην προβολή κινητών συσκευών

Όπως είναι αναμενόμενο τα χαρακτηριστικά της ψηφιακής διαφήμισης διαφέρουν από αυτά των παραδοσιακών μέσων και αυτό γιατί η διαδικτυακή επικοινωνία με τους καταναλωτές είναι μετρήσιμο στοιχείο και η ανταπόκριση του χρήστη είναι άμεση. Το κόστος, η τμηματοποίηση της αγοράς, οι στοχευμένες εκστρατείες, η παγκόσμια εμβέλεια είναι χαρακτηριστικά που αξίζει να αναλυθούν για την προβολή κινητών συσκευών.

- *Στοχευμένο πελατολόγιο:* η ψηφιακή διαφήμιση δίνει τη δυνατότητα στόχευσης σε συγκεκριμένες ομάδες καταναλωτών ή και ατομικά. Είναι πιο εύκολο να εντοπίσουν οι εταιρείες σε ποιο βαθμό είναι θελκτικό το προϊόν/υπηρεσία τους και να προβούν σε αντίστοιχες ενέργειες. Επίσης η τμηματοποίηση με βάση την ηλικία, την τοποθεσία η συσκευή που χρησιμοποιούν είναι διαθέσιμες επιλογές και έτσι κάθε *brand* μπορεί να φτάσει με ακρίβεια στο στόχο του.
- *Δυνατότητα αμφίδρομης επικοινωνίας:* το διαδίκτυο δίνει τη δυνατότητα ανάμιξης του καταναλωτή, την άμεση ανατροφοδότηση τόσο για πωλητές όσο και για αγοραστές καθώς και την ικανοποίησή τους. Είναι σημαντικό ότι οι διαφημιστές είναι σε θέση να έχουν άμεσα αποτελέσματα για όλα τα παραπάνω σε πραγματικό χρόνο.
- *Έλεγχος προϋπολογισμού:* γενικά το κόστος μιας διαδικτυακής εκστρατείας είναι πιο χαμηλό από αυτό για τα υπόλοιπα μέσα. Επίσης ο καθορισμός ενός μέγιστου ποσού για επένδυση είναι πιο εύκολο να αναπροσαρμοστεί ανάλογα με την ανταπόκριση που έχουν οι καταναλωτές με στόχο το υψηλότερο κέρδος.
- *Έλεγχος απόδοσης:* ποτέ δεν υπήρξε πιο εύκολη η μέτρηση της αποδοτικότητας μιας διαφήμισης. Με τη χρήση εργαλείων όπως το *Google Analytics* ή το *Omniture* οι επιχειρήσεις γνωρίζουν πόσοι άνθρωποι είδαν τη διαφήμιση, τι ώρα είχε μεγαλύτερη απήχηση, πόσα κλικ έγιναν κτλ. Με βάση όλες αυτές τις λεπτομέρειες η καμπάνια προσαρμόζεται ώστε να επιτευχθούν τα καλύτερα αποτελέσματα.
- *Παγκόσμια εμβέλεια:* ο αριθμός των καταναλωτών που χρησιμοποιούν το διαδίκτυο ξεπερνά τους 2.000.000.000 χρήστες και αυξάνεται σε καθημερινή βάση. Συνεπώς, οι διαφημίσεις κάθε επιχείρησης είναι προσιτές σε εκατομμύρια πιθανούς αγοραστές σε όλο τον κόσμο.

(Hill & Dhanda, 2015)

2.3 Προϋποθέσεις επιτυχίας μιας ψηφιακής διαφήμισης

Για να είναι επιτυχημένη μια διαφήμιση οφείλει να πληροί ορισμένες προϋποθέσεις, ενδεικτικά αναφέρονται κάποιες από αυτές ως οι πιο σημαντικές (Bradley, 2006).

- Το περιεχόμενο των διαφημίσεων πρέπει να προσδίδει αξία και χρησιμότητα στους καταναλωτές
- Θα πρέπει να δίνεται έμφαση στο εμπορικό σήμα και την εικόνα της επιχείρησης
- Θα πρέπει να συνδέονται με τη διαδικασία της αγοράς και να τελειώνουν με μια πρόταση
- Θα πρέπει να είναι οπτικά ελκυστικές
- Θα πρέπει να στοχεύουν σε μεμονωμένους καταναλωτές ή σε ομάδες ατόμων, και τέλος
- Κάθε διαφήμιση οφείλει να είναι μέρος μιας ολοκληρωμένης στρατηγικής *marketing*

Σε έρευνα που πραγματοποιήθηκε από τους *Gehrke* και *Turban* εντοπίστηκαν 50 μεταβλητές που μπορούν να μεταβάλλουν την ικανοποίηση του αγοραστή στην περιήγησή του και κατ' επέκταση να διαβάξει και να ανταποκρίνεται στις διαφημίσεις. Μετά από ανάλυση οι μεταβλητές αυτές ομαδοποιήθηκαν σε πέντε κατηγορίες.

1. Ευκολία πλοήγησης:
 - οι δικτυακοί τόποι πρέπει να είναι συμβατοί με προγράμματα πλοήγησης
 - ύπαρξη σωστών και με ακρίβεια διατυπωμένων δεσμών
2. Ασφάλεια και διασφάλιση απορρήτου:
 - δυνατότητα απόρριψης των cookies
 - ασφάλεια και ιδιωτικότητα
3. Ταχύτητα φόρτωσης σελίδας:
 - ύπαρξη γραφημάτων εικονιδίων
 - απλά και περιεκτικά γραφικά και πίνακες ευανάγνωστα από όλες τις οθόνες
4. Επιχειρηματικό περιεχόμενο:
 - όσο είναι δυνατό οι εγγραφές των μελών θα πρέπει να είναι σύντομες και απλές
 - το περιεχόμενο των διαφημίσεων πρέπει να είναι σαφές και περιεκτικό ενώ ο τίτλος και η επικεφαλίδα να εντυπωσιάζουν
5. Εστίαση *marketing* στον πελάτη:
 - θα πρέπει οι όροι να είναι ξεκάθαροι και να δίνουν πληροφορίες για την παράδοση, την επιστροφή εμπορευμάτων, τον τρόπο πληρωμής κτλ.

- χρειάζεται μια σελίδα επιβεβαίωσης της παραγγελίας μετά την αγορά (Gehrke & Turban, 1999)

2.4 Τρόποι αξιοποίησης της ψηφιακής διαφήμισης στο Internet

Το IAB (Interactive Advertising Bureau), διεθνής φορέας υποβοήθησης των επιχειρήσεων για την αύξηση των πωλήσεών τους μέσω αποτελεσματικού online marketing και άλλων μέσων προώθησης, παρέχει ένα κατάλογο με τους τρόπους αξιοποίησης του Διαδικτύου από τους διαφημιζόμενους. Παράλληλα, για καθέναν από αυτούς τους τρόπους προτείνει και τις αντίστοιχες μεθόδους μέτρησης αποτελεσματικότητας (<http://www.iab.gr/gr/research/research-list/>).

Σύμφωνα λοιπόν με την IAB, η online διαφήμιση μεταξύ άλλων:

- Αυξάνει τη διάδοση του εμπορικού ονόματος (brand awareness)

(Αξιολόγηση: Μέτρηση της διάδοσης του εμπορικού ονόματος πριν και μετά τη διαφημιστική εκστρατεία)

- Παροτρύνει στη "δοκιμή" του προβαλλόμενου προϊόντος/υπηρεσίας

(Αξιολόγηση: Μέτρηση της αύξησης του κοινού-στόχος, Μέτρηση των νέων πελατών)

- Παροτρύνει τους πελάτες να αγοράζουν περισσότερα προϊόντα σε κάθε συναλλαγή

(Αξιολόγηση: Μέτρηση της ποσότητας προϊόντων που αγοράστηκαν ανά συναλλαγή)

- Βελτιώνει την εικόνα της επιχείρησης στο κοινό

(Αξιολόγηση: Μέτρηση -σε τακτά χρονικά διαστήματα- της πρόσληψης της εικόνας της επιχείρησης στο αγοραστικό κοινό, συμπεριλαμβανομένης της πρόθεσης αγοράς)

- Αυξάνει τις επαναλαμβανόμενες αγορές

(Αξιολόγηση: Μέτρηση των πελατών που πραγματοποιούν επαναλαμβανόμενες αγορές)

- Κατακτά την εμπιστοσύνη του κοινού στη διαφημιζόμενη επιχείρηση/αυξάνει το βαθμό "εμπλοκής" του κοινού με το προβαλλόμενο προϊόν

(Αξιολόγηση: Μέτρηση επαναλαμβανόμενων αγορών, Μέτρηση της πρόσληψης της εικόνας της επιχείρησης στο αγοραστικό κοινό σε σύγκριση με ανταγωνιστικές εταιρίες, Μέτρηση της πρόθεσης αγοράς προϊόντων της προβαλλόμενης επιχείρησης σε σύγκριση με ανταγωνιστικά προϊόντα)

- Παρέχει πλούσια online πληροφόρηση για το προβαλλόμενο προϊόν

(Αξιολόγηση: Μέτρηση των κλικ, Διάρκεια επίσκεψης στο δικτυακό τόπο της προβαλλόμενης επιχείρησης, Μέτρηση e-mails πελατών που ζητούν περαιτέρω πληροφόρηση, Μέτρηση τηλεφωνικών κλήσεων για περισσότερες λεπτομέρειες σχετικά με το προβαλλόμενο προϊόν)

- Παρέχει/βελτιώνει την εξυπηρέτηση πελατών

(Αξιολόγηση: Μέτρηση των online επισκέψεων έναντι των offline επαφών των πελατών με την επιχείρηση, Μέτρηση -σε τακτά χρονικά διαστήματα- του βαθμού ικανοποίησης των πελατών από την ποιότητα εξυπηρέτησης, Μέτρηση του κόστους της online εξυπηρέτησης πελατών σε σύγκριση με τις offline μεθόδους)

- Μειώνει το κόστος του marketing

(Αξιολόγηση: Σύγκριση του κόστους του online marketing και της αποτελεσματικότητάς του με εκείνα του παραδοσιακού marketing)

- Επιτρέπει τη δοκιμή διαφορετικών διαφημιστικών ιδεών

(Αξιολόγηση: Μέτρηση της πρόθεσης αγοράς και της διάδοσης του εμπορικού ονόματος της προβαλλόμενης επιχείρησης πριν και μετά την online προβολή ενός νέου διαφημιστικού μηνύματος)

- Προωθεί την επιχείρηση μέσω προσφορών για προϊόντα ή υπηρεσίες

(Αξιολόγηση: Μέτρηση της πρόθεσης αγοράς και της πρόσληψης της προσφοράς πριν και μετά την online προβολή της)

- Επιτρέπει την πρόσβαση σε κοινό-στόχο που με άλλα μέσα δεν θα ήταν προσβάσιμο

(Αξιολόγηση: Μέτρηση της διεύρυνσης πρόσβασης σε διαφορετικές καταναλωτικές ομάδες, Αύξηση των πωλήσεων)

- Εντοπίζει τα ενδιαφέροντα και τις τάσεις των καταναλωτών

(Αξιολόγηση: Μέτρηση επιτυχημένων εκστρατειών για νέα προϊόντα ή υπηρεσίες)

- Επιτρέπει τη διατήρηση των σημαντικότερων πελατών

(Αξιολόγηση: Μέτρηση του ποσοστού επαναλαμβανόμενων αγορών, Μέτρηση του μακροπρόθεσμου κέρδους ανά πελάτη, Μέτρηση της ικανοποίησης των καταναλωτών)

- Αυξάνει την κίνηση στο δικτυακό τόπο που προβάλλει τη διαφήμιση

(Αξιολόγηση: Μέτρηση click-throughs, αριθμού visits, διάρκειας visits, μέτρηση page impressions)

- Αυξάνει την αγοραστική κίνηση στα φυσικά καταστήματα της προβαλλόμενης επιχείρησης

(Αξιολόγηση: Μέτρηση της αύξησης αγοραστικής κίνησης στο φυσικό κατάστημα, Μέτρηση των πελατών που πληροφορήθηκαν για το κατάστημα από online διαφήμιση)

- Βελτιστοποιεί την πρόσληψη εργατικού δυναμικού

- (Αξιολόγηση: Μέτρηση της μείωσης του κόστους πρόσληψης προσωπικού, Μέτρηση επιτυχημένων προσλήψεων)

2.5 Στατιστικά δημοτικότητας μιας ιστοσελίδας διαφημίσεων

Τα σημαντικότερα από τα πληροφοριακά στοιχεία τα οποία παρέχονται για τη δημοτικότητα μίας ιστοσελίδας η οποία με βάση τα αποτελέσματα που προκύπτουν μπορεί να κριθεί άξια για να χρησιμοποιηθεί για online διαφήμιση είναι (Hopkins and Turner, 2012):

1. Χτυπήματα (Hits)

Είναι το σύνολο των αρχείων τα οποία ζητήθηκαν από ένα web server. Κάθε σελίδα στο web αποτελείται από ένα αρχείο HTML και από τα αρχεία όλων των γραφικών (φωτογραφίες, εικονίδια, video) που περιέχει. Έτσι, η εμφάνιση μιας σελίδας με 2 φωτογραφίες και 3 γραφικά καταγράφεται ως 6 hits. Τα hits είναι αναξιόπιστα σαν μέτρο υπολογισμού της δημοτικότητας ενός site αλλά δυστυχώς πολλοί εξισώνουν τα hits με σελίδες ή, ακόμη χειρότερα, με αριθμό μοναδικών χρηστών με αποτέλεσμα να δημιουργούνται λάθος εντυπώσεις για τη δημοτικότητα του site τους.

2. Αιτήσεις (Requests ή Page Views)

Τα requests αναφέρονται στον συνολικό αριθμό των αρχείων HTML τα οποία προβλήθηκαν στους χρήστες. Και επειδή κάθε σελίδα αποτελείται από ένα και μόνο αρχείο HTML, 10.000 requests σημαίνει προβολή 10.000 ιστοσελίδων οι οποίες ζητήθηκαν από τους χρήστες. Υπάρχει και η δυνατότητα μέτρησης του πόσες φορές ζητήθηκε η κάθε μια από αυτές τις σελίδες. Αυτό ενδιαφέρει και τους κατόχους των sites αλλά και τους διαφημιστές γιατί οι πρώτοι βλέπουν τι έχει μεγάλη ζήτηση από το site τους ενώ οι δεύτεροι επιλέγουν καλύτερα το σημείο που θα βάλουν τις διαφημίσεις τους σε ένα site.

3. Εμφανίσεις (Impressions)

Ο όρος αυτός χρησιμοποιείται κυρίως για προβολές διαφημίσεων και δείχνει πόσες φορές εμφανίστηκε ένα διαφημιστικό banner.

4. Χρήστες (Users ή Unique IP Addresses)

Όταν ένας χρήστης συνδέεται στο διαδίκτυο παίρνει από τον πάροχο μια δυναμική (μη σταθερή) διεύθυνση IP . Κάθε φορά που συνδέεται στο διαδίκτυο η διεύθυνση αυτή αλλάζει. Υπάρχουν όμως και υπολογιστές που έχουν μια στατική (μόνιμη) IP διεύθυνση. Τέλος, οι περισσότερες εταιρείες διαθέτουν για όλους τους υπαλλήλους τους πρόσβαση στο διαδίκτυο μέσω ενός proxy server ο οποίος έχει μια και μόνη IP διεύθυνση. Αν ο web server που φιλοξενεί το site δεν χρησιμοποιεί cookies (όλο και πιο σπάνιο) τότε ο αριθμός των επισκεπτών ενός site θεωρείται ίσος με τον αριθμό των IP διευθύνσεων από τις οποίες ζητήθηκαν σελίδες. Εδώ όμως θα πρέπει να επισημάνουμε ότι στην πρώτη περίπτωση που εξετάσαμε παραπάνω ο αριθμός των χρηστών υπερεκτιμάται γιατί κάθε φορά που ο ίδιος επισκέπτης μπαίνει στο διαδίκτυο και το ίδιο site καταγράφεται σαν διαφορετικός επισκέπτης. Στην δεύτερη και τρίτη

περίπτωση ο αριθμός των χρηστών υποεκτιμάται. Στην περίπτωση 2 πολλοί επισκέπτες μπαίνουν στο site με την ίδια (στατική) διεύθυνση ενώ στην περίπτωση 3 αν χρησιμοποιούν τον proxy server 10 ή και 15 χρήστες για να «βγουν» στο διαδίκτυο, όλοι αυτοί καταγράφονται σαν ένας επισκέπτης. Μια ακριβέστερη καταμέτρηση επιτυγχάνεται όταν ο web server που φιλοξενεί το site χρησιμοποιεί cookies, διότι τότε δίνει σε κάθε υπολογιστή που επισκέπτεται το site ένα «μπισκότο» (cookie) το οποίο είναι μικρό αρχείο κειμένου που χρησιμοποιείται σαν ταυτότητα του συγκεκριμένου υπολογιστή. Έτσι, οι επισκέπτες αναγνωρίζονται από τα μηχανήματά τους και όχι από τις μεταβαλλόμενες IP διευθύνσεις. Βέβαια αν ο ίδιος χρήστης επισκεφθεί το site από δύο διαφορετικούς υπολογιστές θα υπολογιστεί ως δύο χρήστες. Η περίπτωση που ο χρήστης έχει απενεργοποιήσει τη λήψη cookies (και άρα δεν καταγράφεται σαν επισκέπτης) ανέρχεται στο 2% περίπου και έτσι θεωρείται αμελητέα.

5. Επισκέψεις (Visits ή sessions)

Είναι οι φορές που ο ίδιος χρήστης επισκέφθηκε ένα site μέσα στην ίδια μέρα αλλά σε διαφορετικές χρονικές στιγμές. Μια κοινά αποδεκτή διαφορά χρόνου μεταξύ δύο επισκέψεων θεωρείται η μισή ώρα. Το μέγεθος όμως αυτό είναι αυθαίρετο και μπορεί να αλλάζει από ιστοσελίδα σε ιστοσελίδα. Αιτήσεις από ένα χρήστη για διαφορετικές σελίδες ενός site σε λιγότερο από μισή ώρα δηλώνουν ότι ο χρήστης είναι ενεργός (περιηγείται δηλαδή μέσα στο site χωρίς να έχει φύγει).όσο μεγαλύτερος ο αριθμός των επισκέψεων τόσο πιο πιστούς χρήστες θεωρείται ότι έχει ένα site.

6. Μέσος αριθμός αιτήσεων ανά επίσκεψη (Average requests per visit)

Το νούμερο αυτό δηλώνει πόσες σελίδες είδε (ζήτησε) ο μέσος επισκέπτης σε κάθε επίσκεψή του. Θεωρητικά, όσο μεγαλύτερο είναι αυτό το νούμερο, τόσο πιο χρήσιμο θεωρεί το site ο επισκέπτης και γι' αυτό θέλει να δει περισσότερες σελίδες του.

7. Διάρκεια αίτησης (Request duration)

Είναι ο χρόνος που αφιέρωσε σε κάθε σελίδα ο επισκέπτης μέχρι να ζητήσει μια άλλη του ίδιου site. Αν ο χρόνος είναι μικρός σημαίνει ότι το περιεχόμενο της σελίδας, η δομή της ή η σχεδίαση του site δεν κρίθηκαν αξιόλογα από τον χρήστη.

8. Διάρκεια επίσκεψης (Visit duration)

Είναι το άθροισμα των Request durations κάθε χρήστη και δηλώνει το συνολικό ενδιαφέρον του χρήστη για το site.

9. Entry pages

Είναι οι σελίδες τις οποίες φορτώνουν πρώτες οι επισκέπτες ενός site. Η πιο συνηθισμένη είναι η αρχική σελίδα. Αυτό σημαίνει πως αν υπάρχουν και άλλες τότε για κάποιο λόγο οι χρήστες θεωρούν αυτές τις σελίδες τόσο σημαντικές που "πηγαίνουν" απευθείας. Μπορεί βέβαια αυτό

να δηλώνει μια πετυχημένη εκστρατεία με banners.

10. Exit pages

Οι τελευταίες σελίδες που βλέπουν οι χρήστες πριν φύγουν από το site. Αυτή η μέτρηση μας δίνει δύο - αντίθετα μεταξύ τους – στοιχεία: τις σελίδες που περιείχαν αυτό που ήθελαν οι χρήστες (και έφυγαν αφού το βρήκαν) ή τις σελίδες στις οποίες εγκαταλείπουν συνήθως στην προσπάθεια να βρουν αυτό που θέλουν και φεύγουν.

11. Referrer Data (ή Referrals)

Είναι οι λέξεις-κλειδιά που χρησιμοποιήθηκαν από τους χρήστες μιας μηχανής αναζήτησης και τους οδήγησαν τελικά σε αυτήν την ιστοσελίδα. Οι πιο γνωστές υπηρεσίες ανάλυσης δημοτικότητας είναι η WebTrends, η IBM's SurfAid Analytics και η HitBox Enterprise. Όλες αυτές λειτουργούν με την τοποθέτηση ειδικού κώδικα σε κάθε σελίδα του site. Αλλά απαλλάσσουν τον διαχειριστή του site από οποιαδήποτε άλλη εργασία.

Η πληροφόρηση που δίνουν οι υπηρεσίες αυτές μπορεί να κατηγοριοποιηθεί ως εξής (Hopkins and Turner, 2012):

A. Γενικά στατιστικά

Εδώ υπάρχουν τα βασικότερα στοιχεία και πληροφορίες:

- Πόσες φορές ζητήθηκε η πρώτη σελίδα
- Πόσες σελίδες ζητήθηκαν
- Πόσες επισκέψεις πραγματοποιήθηκαν
- Πόσο διήρκεσε κατά μέσο όρο κάθε επίσκεψη
- Πόσοι ήταν οι επισκέπτες
- Πόσοι επισκέφθηκαν περισσότερο από μια φορά το site

B. Ειδικότερα στατιστικά

- Ποιες είναι οι πιο δημοφιλείς σελίδες του site
- Ποιες είναι οι πρώτες σελίδες που είδαν όσοι επισκέφθηκαν το site
- Από ποιες σελίδες έφυγαν οι επισκέπτες
- Ποιες σελίδες ζήτησαν κάποιοι, τις είδαν και έφυγαν χωρίς να κοιτάξουν κάτι άλλο
- Ποιες ομάδες αρχείων παρουσίασαν τη μεγαλύτερη κίνηση

Είναι σημαντικό να αναλύονται και η η επιτυχία (δημοφιλείς σελίδες) αλλά και η αποτυχία (μη δημοφιλείς) μίας ιστοσελίδας.

Γ. Δημογραφικά επισκεπτών

- Πόσο συχνά οι χρήστες επιστρέφουν στο site
- Από ποιες χώρες προέρχονται οι χρήστες

- Ειδικά το πρώτο στοιχείο δείχνει πια είναι η κρίσιμη μάζα των επισκεπτών στην οποία μπορούμε να απευθυνθούμε για προγράμματα πιστών πελατών και γενικά για περαιτέρω ανάπτυξη των δραστηριοτήτων μας

Δ. Στατιστικά δραστηριότητας

Χρονική κατανομή των επισκέψεων στο site

- Δημοφιλέστερη ώρα της ημέρας
- Δημοφιλέστερη ώρα της εβδομάδας
- Δημοφιλέστερη ημέρα της εβδομάδας

Ε. Sites σύστασης & Λέξεις κλειδιά

Τα στατιστικά αυτά μας δίνουν στοιχεία για τα sites που μας «στέλνουν» επισκέπτες:

- Από ποια sites έρχονται οι επισκέπτες
- Διευθύνσεις ή αναζητήσεις σε μηχανές αναζήτησης που οδηγούν στο site μας (σελίδες δηλαδή άλλων sites που περιέχουν παραπομπές στο δικό μας)
- Επισκέψεις από παραπομπές μηχανών αναζήτησης του Internet και λέξεις κλειδιά που χρησιμοποιήθηκαν και που βάσει αυτών οι χρήστες βρήκαν το site μας

Ζ. Browsers & Platforms

Οι πληροφορίες που παίρνουμε από αυτή την κατηγορία έχουν να κάνουν με:

- Το είδος και την έκδοση των browsers που χρησιμοποιούν οι επισκέπτες μας
- Το λειτουργικό σύστημα των επισκεπτών μας
- Το ποσοστό της κίνησης του site που οφείλεται στις επισκέψεις προγραμμάτων (spiders)

Οι πληροφορίες αυτού του είδους είναι ιδιαίτερα χρήσιμες σε όσους διακινούν προϊόντα που συνδέονται με την τεχνολογία και «παίζουν» σε συγκεκριμένα λειτουργικά και browsers. Ενδιαφέρει επίσης και τους διαφημιστές που δοκιμάζουν νέες τεχνολογίες στη διαφήμιση και δεν θέλουν να «χαραμίσουν» εμφανίσεις διαφημίσεων σε browsers χρηστών που δεν μπορούν να τις δουν.

2.6 Αποτελεσματικότητα ψηφιακής διαφήμισης

Το διαδίκτυο μπαίνει με ένα σημαντικό πλεονέκτημα στο επιχειρηματικό προσκήνιο, επηρεάζοντας το μέχρι τώρα διαφημιστικό πλαίσιο. Το πλεονέκτημα αυτό εκφράζεται με μια λέξη, μετρήσιμο (Bradley, 2006).

Το διαδίκτυο αποτελεί χωρίς καμία αμφιβολία το περισσότερο μετρήσιμο απ' όλα τα διαφημιστικά μέσα και παρέχει μεγάλη πληθώρα δεδομένων πολλά από τα οποία μπορούν να

αποδειχθούν εξαιρετικά πολύτιμα για τη σχεδίαση της διαφημιστικής στρατηγικής μιας επιχείρησης.

Οι διαφημιστές έχουν την δυνατότητα μέσω εργαλείων να καταγράφουν:

1. τον αριθμό εμφανίσεων (impressions) των διαφημίσεων στα sites που χρησιμοποιούν για τη διαφημιστική τους καμπάνια, όχι μόνο για μια αλλά και για περισσότερες διαφημίσεις ταυτόχρονα (2 ή 3 banners, interstitials, superstitials, web overs και άλλες μορφές διαφήμισης) ώστε να καταγράφουν ποια είναι η διαφήμιση που φέρνει τα περισσότερα κλικ στο site του διαφημιζόμενου
2. τον αριθμό των κλικ (click throughs) πάνω σε κάθε διαφήμιση
3. τις ώρες και ημέρες της εβδομάδας με τα περισσότερα κλικ στις διαφημίσεις
4. τις κινήσεις που έκανε ένας συγκεκριμένος χρήστης αφού μπήκε στο site του διαφημιζόμενου (επίσκεψη σελίδων, αγορές ανά προϊόν, συμπλήρωση φόρμας κ.ά.) από κάθε διαφήμιση που «τρέχει» στην καμπάνια.

Οι διαφημιστές έχουν την δυνατότητα να φιλτράρουν:

1. τον αριθμό εμφανίσεων (impressions) μιας διαφήμισης ανά μοναδικό χρήστη. Μπορούν δηλαδή να περιορίσουν τον αριθμό που θα δει ο κάθε μοναδικός χρήστης μια διαφήμιση. Αυτό γίνεται για δύο λόγους. Ο πρώτος είναι ότι έρευνες έχουν δείξει πως αν ένας χρήστης δει μια διαφήμιση πάνω από περίπου 5 φορές και δεν έχει κάνει κλικ επάνω της δεν θα κάνει ποτέ. Ο δεύτερος είναι γιατί η πώληση διαφημιστικού χώρου στην πλειονότητα των sites γίνεται με αγορά αριθμού εμφανίσεων.
2. Τις χώρες (ή και τις πολιτείες για τις ΗΠΑ) που θα εμφανιστεί η διαφήμιση. Μπορούν δηλαδή να κατευθύνουν διαφημίσεις εκτός Ελλάδας μόνο στις χώρες με ελληνόφωνο πληθυσμό.
3. Το λειτουργικό σύστημα και τον τύπο του browser που θα δεχθεί διαφήμιση.

Τέλος, οι διαφημιστές έχουν τη δυνατότητα να μετρούν:

1. Το κόστος ανά 1000 εμφανίσεις ανά site αλλά και για σύνολο των sites που χρησιμοποιήθηκαν για την καμπάνια
2. Το κόστος ανά μοναδικό χρήστη που είδε μια διαφήμιση
3. Το κόστος ανά κλικ
4. Τον αριθμό των κλικ ανά διαφήμιση και ανά site
5. Το ποσοστό των κλικ ανά 100 εμφανίσεις της διαφήμισης ανά διαφήμιση και ανά site
6. Τον αριθμό των μοναδικών χρηστών που είδαν τις διαφημίσεις
7. Τον αριθμό των διαφημίσεων που είδε ο κάθε μοναδικός χρήστης και τις μετέπειτα

ενέργειές του

8. Το κόστος της διαφήμισης για τις πωλήσεις που προήλθαν από τους χρήστες που είδαν τη διαφήμιση

Επίσης, έχουν αρχίσει να γίνονται αρκετά δημοφιλείς υπηρεσίες ανάλυσης δημοτικότητας οι οποίες λειτουργούν με την τοποθέτηση ειδικού κώδικα σε κάθε σελίδα του site. Το κύριο πλεονέκτημά τους είναι ότι όλη η επεξεργασία γίνεται από έναν τρίτο χωρίς την ανάμιξη του ιδιοκτήτη της ιστοσελίδας. Τα μειονεκτήματά τους είναι η τοποθέτηση του ειδικού κώδικα σε όλες τις σελίδες του site, κάτι αρκετά επίπονο και η πρόκληση μιας μικρής καθυστέρησης στο φόρτωμα κάθε σελίδας, καθώς ο ειδικός κώδικας που προστίθεται στη σελίδα πρέπει να ενημερώνει το site της υπηρεσίας ανάλυσης δημοτικότητας κάθε φορά που αυτή ζητείται από ένα χρήστη.

ΚΕΦΑΛΑΙΟ 3: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΕΠΙΣΚΟΠΗΣΗ ΓΙΑ ΤΑ ΔΙΑΦΗΜΙΣΤΙΚΑ BANNERS

3.1 Γενικές έρευνες

Μια σειρά ερευνών και δοκιμών έχουν διεξαχθεί στο παρελθόν για τον τρόπο με τον οποίο παρατηρείται μια διαφήμιση banner. Στις δοκιμές αυτές το σχήμα (banner) παρουσιάστηκε αρκετές φορές, ωστόσο το ποσοστό των ανθρώπων που δήλωσαν ότι είχαν δει τη διαφήμιση ξανά προτού προβληθεί ξανά αυξήθηκε μόνο κατά 8% μεταξύ των δοκιμών. Αυτό το αποτέλεσμα σημαίνει ότι η επίδραση του διαφημιστικού banner μπορεί να χαθεί απλώς και μόνο επειδή το banner δεν παρατηρείται. Η έρευνα συνέκρινε επίσης τις διαφορετικές μορφές νεότερων διαφημίσεων με τις παλαιότερες μορφές banner.

Τα ευρήματα ήταν αρκετά σαφή, τα μεγάλα banners παράγγααν επίπεδα αναγνώρισης διπλάσια από τις κανονικές διαφημίσεις banner. Τα διαφημιστικά σχήματα που περιλάμβαναν κίνηση οδήγησαν σε επίπεδα αναγνώρισης τέσσερις φορές υψηλότερα. Το πιο σημαντικό είναι ότι η μελέτη έδειξε ότι οι νεότερες μορφές διαφήμισης είχαν πιο θετικό αντίκτυπο μεταφέροντας την αναγνωσιμότητα της επωνυμίας. Εν κατακλείδι, η μελέτη έδειξε ότι το μέγεθος και η κίνηση του banner επηρεάζουν σημαντικά την προσοχή των ανθρώπων. Οι νεότερες διαφημίσεις banner επίσης έκαναν καλύτερη δουλειά να μεταφέρουν δεδομένα (du Plessis, Erick and Brown, Millward. , 2005).

Η έρευνα σχετικά με τα διαφημιστικά banners έχει κάνει τις εταιρείες και τους διαφημιστές να συνειδητοποιήσουν τις προκλήσεις γύρω από αυτά. Χρησιμοποιούνται όλο και περισσότερα διαδραστικά banner που με τη σειρά τους μετατρέπονται σε καλύτερα μέσα προβολής αγαθών προς τους καταναλωτές. Η ταχύτερη ευρυζωνική τεχνολογία που είναι διαθέσιμη στο ευρύτερο μέρος του κοινού, επιτρέπει τη δημιουργία καλύτερων και πιο διαδραστικών banner. Ένα banner το οποίο προβάλλεται καθώς ανοίγει μία σελίδα και κρύβει το πίσω μέρος αποτελεί καλό παράδειγμα αυτού του τύπου αλληλεπιδραστικού banner.

Στο βιβλίο του Salmenkivi et al (2007) περιγράφει μια επιτυχημένη χρήση τέτοιου τύπου διαφήμισης. Ένα διαφημιστικό banner μπορεί να προβληθεί για παράδειγμα σε μία δημοφιλή πύλη συζήτησης ή μέσω κοινωνικής δικτύωσης τη στιγμή που συζητείται ένα ευρύτερα γνωστό θέμα ή ένα προϊόν λανσάρεται στην αγορά. Εκτός από το καθαρό μέγεθος του banner που κάνει πιο αισθητό, δίνεται η δυνατότητα της διαδραστικότητας για να τραβήξει την προσοχή. Η διαδραστικότητα από μόνη της αξίζει ελάχιστα, αλλά θα πρέπει να χρησιμοποιηθεί ως συμπλήρωμα του παραδοσιακού μάρκετινγκ και της δημιουργικότητας για την επίτευξη βέλτιστων αποτελεσμάτων (Salmenkivi et Nyman, 2007).

Ο συνδυασμός πληροφόρησης σε πραγματικό χρόνο και διαφήμισης είναι ένα σημαντικό πλεονέκτημα που διαθέτει η διαφήμιση μέσω banner και με αυτό το τρόπο αντιτίθεται στην παραδοσιακή διαφήμιση. Χρησιμοποιώντας αυτό το δεδομένο, το διαφημιστικό μήνυμα μπορεί να βελτιστοποιηθεί για μια δεδομένη κατάσταση.

Έρευνες πελατών έδειξαν για παράδειγμα ότι οι πελάτες επηρεάζονται ανάλογα με τη τοποθεσία από τις καιρικές συνθήκες. Οι αναζητήσεις για εξωτερικές δραστηριότητες κατά τη διάρκεια μιας ηλιόλουστης ημέρας θα είναι πολύ υψηλότερες από ό, τι σε μια βροχερή μέρα για παράδειγμα. Αυτό δίνει το κίνητρο στους πωλητές να δημιουργούν banner που αλλάζουν ανάλογα με τις καιρικές συνθήκες. Σε μια ηλιόλουστη μέρα οι επισκέπτες θα δουν πανό να παρουσιάζουν υπαίθριες δραστηριότητες και σε μια βροχερή μέρα τα πανό θα προωθούσαν για παράδειγμα αδιάβροχα και ομπρέλες. Αυτό δείχνει πώς τα banners μπορούν να χρησιμοποιηθούν με καινοτόμους τρόπους για να αυξήσουν την αποτελεσματικότητά τους (Salmenkivi et Nyman, 2007).

Σύμφωνα με το Salmikivi (2007) είναι σημαντικό όταν οι χρήστες πατάνε το banner, η εφαρμογή να οδηγεί το χρήστη όπου ο χρήστης υποθέτει ότι θα οδηγούσε. Σε νεότερες εκδόσεις μεγάλο μέρος του περιεχομένου μπορεί ήδη να συμπεριληφθεί στο ίδιο το banner και συνεπώς δεν απαιτεί από τους χρήστες να εγκαταλείψουν τη σελίδα. Αυτή η λειτουργία μπορεί να εφαρμοστεί με πολλούς τρόπους, αλλά συχνά δημιουργείται μια μικροσκοπική σελίδα μέσα στο banner (Salmenkivi et Nyman, 2007).

Σε όλο τον κόσμο διάφορα μεγέθη banner έχουν τυποποιηθεί σε κάποιο βαθμό, αλλά εμφανίζονται πολλές διαφοροποιήσεις ανάλογα με την περιοχή. Μια γενική τάση είναι ότι γίνονται όλο και πιο καινοτόμα και περιπλέκονται τόσο σε μορφή όσο και σε λειτουργικότητα.

Τα παραδοσιακά μεγέθη banner, όπως το Skyscraper, που μετρά 200 x 900 pixels με μέσο CTR 0,32% και το Double box με μέση τιμή 468 x 600 pixels με CTR 0,22%, αντικαθίστανται από νεότερα πιο αποτελεσματικά μεγέθη. Τα πρόσφατα μεγέθη banner περιλαμβάνουν μπορεί να είναι π.χ. τύπου leaderboard 980 x 400 pixels που είναι αρκετά μεγάλο και τοποθετείται στην κορυφή του περιεχομένου του ιστότοπου. Ουσιαστικά εκτείνεται σε όλο το μήκος του ιστότοπου. Επίσης, υπάρχουν διάφορες παραλλαγές που κάνουν χρήση του φόντου του ιστότοπου, χρόνου ενεργοποίησης, και άλλα.

Η αναπαραγωγή βίντεο ή ήχου είναι πρόσθετοι τρόποι για να προσελκύσουν την προσοχή του χρήστη και να αυξήσουν το CTR. Η εικόνα που εμφανίζεται παρακάτω είναι ένα λεγόμενο banner ταπετσαρίας που προάγει μια κινηματογραφική ταινία. Το banner στην πραγματικότητα κάθεται στο φόντο της σελίδας, και κατορθώνει να συγκεντρώνει τη προσοχή του χρήστη λόγω του μεγέθους του. Σε αυτή τη συγκεκριμένη περίπτωση, η λύση του banner επιτρέπει επίσης στον

επισκέπτη να παρακολουθήσει το τρέιλερ ταινιών στη σελίδα χωρίς να τον οδηγήσει σε μια ξεχωριστή σελίδα.

Οι Briggs & Hollis (1997), σε μία από τις πρώτες μελέτες της επίδρασης των διαφημίσεων τύπου banner, διαπίστωσαν ότι ακόμη και χωρίς οι καταναλωτές να κάνουν κλικ πάνω σε αυτές, είχαν θετικό αποτέλεσμα όσον αφορά την αναγνωρισιμότητα του brand, καθώς και σε αλλαγή στάσεων και αντιλήψεων για το brand. Ένα σημείο το οποίο φαίνεται ότι επηρεάζει την αλληλεπίδραση του καταναλωτή με μια διαφήμιση τύπου banner, είναι η κίνηση που αυτή περιέχει, καθώς και η θέση που αυτή καταλαμβάνει σε μια ιστοσελίδα (Josephson, 2005).

Αρκετοί ερευνητές (Zhang 2000; Raudienė, 2011), αναφέρουν ότι τα ψηφιακά banner με κίνηση βοηθούν στην ευκολότερη ανάκληση του περιεχομένου τους από τον καταναλωτή, ενώ ευνοούν και την καλύτερη στάση των καταναλωτών απέναντι τους, οδηγώντας σε μεγαλύτερο αριθμό click-through, σε σχέση με τις στατικές διαφημίσεις. Αντιθέτως, οι Pabedinskaitė & Fiodorovaitė (2011), χρησιμοποιώντας τεχνικές μελέτης παρακολούθησης της κίνησης των ματιών, συμπέραναν ότι κίνηση που περιείχε το banner δεν ήταν σημαντικός παράγοντας για την προσέλκυση της προσοχής του χρήστη του διαδικτύου.

Αναφορικά με τη θέση, τα συμπεράσματα των Josephson (2005) και Ghajarzadeh et. al, (2010), συμφωνούν στο ότι η θέση που διαδραματίζει σημαντικό ρόλο για την αλληλεπίδραση με τις διαφημίσεις τύπου banner, είναι στην κορυφή της ιστοσελίδας. Παράλληλα, άλλες έρευνες (Pabedinskaitė & Fiodorovaitė, 2011) διαπιστώνουν ότι η θέση του ψηφιακού banner δεν επηρεάζει σημαντικά την αλληλεπίδραση των καταναλωτών με αυτό.

Έρευνες έχουν δείξει ότι η αποτελεσματικότητα των διαφημίσεων τύπου banner εξαρτάται από διάφορους παράγοντες, όπως είναι τα χαρακτηριστικά των καταναλωτών (Palanisamy, 2005), ο τρόπος με τον οποίο τα banner παρουσιάζονται στους χρήστες-καταναλωτές (Yaveroglu & Donthu, 2008), καθώς και τα χαρακτηριστικά των banner (Baraggioli & Brasel, 2008), δηλαδή το κείμενο του, οι διαστάσεις του, το χρώμα του, η κίνηση του κ.ά. Ένα από χαρακτηριστικά που είναι κρίσιμα για την επιτυχία των διαφημίσεων τύπου banner, αλλά δεν φαίνεται μελετηθεί αρκετά είναι το περιεχόμενο του διαφημιστικού μηνύματος. Δεδομένου ότι οι διαφημίσεις banner στοχεύουν στο να μεταβάλλουν τις αντιλήψεις των καταναλωτών, το μήνυμα που μεταφέρεται μέσω αυτών, πρέπει να διαδραματίζει καθοριστικό ρόλο στον τρόπο με τον οποίο μεταβάλλονται οι αντιλήψεις των καταναλωτών.

Ο Palanisamy (2005) διαπιστώνει ότι το φύλο επηρεάζει τη στάση και τις προσδοκίες των καταναλωτών απέναντι στη διαφήμιση τύπου banner. Σε μια μελέτη που πραγματοποιήθηκε στην Κίνα και το Ηνωμένο Βασίλειο, η Ju (2013) διαπίστωσε ότι το πολιτιστικό υπόβαθρο διαδραματίζει σημαντικό ρόλο στην αποτελεσματικότητα των διαφημίσεων τύπου banner. γεγονός το οποίο

παρατήρησαν και οι Möller & Elsend (2010), αναφέροντας ότι η πρόθεση των χρηστών του διαδικτύου να κάνουν click σε μια διαφήμιση banner επηρεάζεται από την κουλτούρα του χρήστη. Οι Yaveroglu & Donthu (2008) εστιάζουν περισσότερο στην στρατηγική με τον οποίο εμφανίζονται τα banner. Συγκεκριμένα, η μελέτη τους έδειξε ότι η στρατηγική επανάληψης των διαφημίσεων τύπου banner επηρεάζουν την αποτελεσματικότητα της διαφήμισης, διαπιστώνοντας ότι η επανάληψη των συγκεκριμένων διαφημίσεων οδηγεί σε μεγαλύτερη ανάκληση της μάρκας που διαφημίζεται καθώς και στην πρόθεση του καταναλωτή να κάνει click πάνω σε αυτή. Σε ένα μη ανταγωνιστικό περιβάλλον, λειτουργεί πιο αποτελεσματικά μια στρατηγική διακύμανσης της διαφήμισης καλύτερα, ενώ σε ένα ανταγωνιστικό περιβάλλον, είναι πιο αποτελεσματική μια στρατηγική επανάληψης των διαφημίσεων.

Άλλοι ερευνητές έχουν αναφέρουν ότι το μέγεθος της διαφήμισης (Baltas, 2003), το σημείο εμφάνισης της (Rosenkrans, 2010), καθώς και η διάρκεια της έκθεσης του καταναλωτή σε αυτήν (Wang, Shih & Peracchio, 2013) επηρεάζει θετικά την αποτελεσματικότητα της banner διαφήμισης. Οι Robinson, Wysocka & Hand (2007) μελέτησαν τον αντίκτυπο χαρακτηριστικών των διαφημίσεων banner αναφορικά με την αποτελεσματικότητά τους, όπως η κίνηση, η έντονη παρουσία των λογότυπων των εταιρειών, καθώς και το περιεχόμενο του μηνύματος. Οι Baraggioli & Brasel (2008) στη μελέτη τους χρησιμοποίησαν ένα εργαλείο ανίχνευσης της κίνησης του ματιού του χρήστη, καταλήγοντας στο συμπέρασμα ότι οι μεγαλύτερες κινήσεις σε συνθήκες ευρείας απόστασης μπορούν να οδηγήσουν σε αυξημένη οπτική προσοχή σε περιφερειακά διαφημιστικά banners.

Σε μια παρόμοια έρευνα, οι Lee & Ahn (2012) μελέτησαν την επίδραση της κινούμενης εικόνας στις διαφημίσεις banner αναφορικά με την εστίαση της προσοχής των καταναλωτών και στη μεταγενέστερη επεξεργασία των προβαλλόμενων πληροφοριών. Διαπίστωσαν ότι η κινούμενη εικόνα όχι μόνο προσελκύει λιγότερη προσοχή, αλλά επίσης μειώνει την επίδραση στη μνήμη του καταναλωτή. Ανάλογα είναι τα συμπεράσματα των Thota, Song & Larsen (2010) και Robinson et al. (2007), οι οποίοι διαπίστωσαν ότι η κινούμενη εικόνα δεν κατέχει σημαντικό ρόλο στην αποτελεσματικότητα της διαφημιστικής προβολής με banner, ενώ οι Chtourou & Abida (2010) κατέληξαν η κινούμενη εικόνα στις διαφημίσεις banner επηρεάζει την πρόθεση του χρήστη να κάνει click πάνω σε αυτή.

Οι Chi, Yeh & Chiou (2012) στη μελέτη τους διαπίστωσαν ότι το στυλ παρουσίασης των πληροφοριών σε ένα διαφημιστικό banner επηρεάζει την αποτελεσματικότητά του. Ο Rosenkrans (2010) μελέτησε το ρόλο του σχεδιασμού των διαφημίσεων banner στην αποτελεσματικότητά τους. Συγκεκριμένα, τα κριτήρια σχεδιασμού που μελέτησε ήταν η διαδραστικότητα, η κίνηση και η φύση της επίκλησης του μηνύματος (λογική ή συναισθηματική).

Οι Briggs & Hollis (1997) διαπίστωσαν ότι οι διαφημίσεις banner μπορούν να αλλάξουν τις αντιλήψεις των καταναλωτών για μια μάρκα ακόμη και χωρίς να κάνουν click πάνω της. Η παλαιότερη, αλλά ακόμα η πιο διαδεδομένη μέθοδος μέτρησης της αποτελεσματικότητας μιας διαφήμισης, η οποία προέρχεται από την παραδοσιακή διαφήμιση, βασίζεται στον αριθμό των ατόμων που έχουν εκτεθεί στη διαφήμιση banner. Αυτό το κριτήριο έχει επικριθεί, καθώς είναι δυνατόν ένας καταναλωτής να έχει εκτεθεί σε μια διαφήμιση banner, αλλά να μην έχει δώσει καμία προσοχή σε αυτό (Lee & Ahn, 2012). Οι Drèze & Hussherr (2003) διαπίστωσαν ότι ακόμη και όταν οι καταναλωτές δεν κάνουν click σε μια διαφήμιση banner, είναι αποτελεσματική, αφού μένει στο υποσυνείδητο του καταναλωτή.

Σε αντίθεση με το φυσικό περιβάλλον του εμπορίου, η συμπεριφορά πλοήγησης των καταναλωτών στον ιστό μπορεί να καταγραφεί. Αυτό έχει οδηγήσει στην ανάπτυξη μεγεθών μέτρησης της αλληλεπίδρασης των καταναλωτών με διαφημίσεις banner, όπως είναι τα click (Baltas, 2003; Möller & Eisend, 2010) και τα mouse rollovers (Rosenkrans, 2010). Άλλοι ερευνητές έχουν δείξει ότι τα παραπάνω εργαλεία μέτρησης της αποτελεσματικότητας είναι προβληματικά (Drèze & Hussherr, 2003), αφού μελέτες που έχουν πραγματοποιηθεί με εργαλεία ανίχνευσης της κίνησης του ματιού έχουν δείξει ότι ακόμη και όταν οι καταναλωτές δεν αλληλεπιδρούν με μια διαφήμιση, κάνοντας click πάνω της, μπορεί την προσέξουν και να την επεξεργαστούν διανοητικά (Baraggioli & Brasel, 2008; Lee & Ahn, 2012). Προκειμένου μια διαφήμιση με banner να είναι αποτελεσματική οι καταναλωτές όχι μόνο πρέπει να δώσουν προσοχή σε αυτή, αλλά και να την ανακαλέσουν μετά από γνωστική επεξεργασία. Δύο βασικοί παράγοντες που αυξάνουν την ανάκληση των διαφημίσεων banner είναι τα ενσωματωμένα βίντεο, καθώς και η τιμή του διαφημιζόμενου προϊόντος ή υπηρεσίας (Alijani et al., 2010).

Οι Ghosh & Bhatnagar (2013), μελετώντας μια σειρά από έρευνες σχετικά με την αποτελεσματικότητα των banners, παρατηρούν ότι δεν υπάρχουν συγκεκριμένοι παράγοντες που να την επηρεάζουν, αφού τα αποτελέσματα των διάφορων ερευνών συχνά αντικρούονται. Αξίζει να σημειωθεί ότι αυτό εξαρτάται και από τις συνθήκες βάσει των οποίων διενεργήθηκε η κάθε έρευνα. Παρόλα αυτά σε βάθος χρόνου φαίνεται ότι η κίνηση που περιέχει το banner, καθώς και η θέση στην οποία αυτό εμφανίζεται, επηρεάζουν την αποτελεσματικότητά τους.

Οι έρευνες σχετικά με την αποτελεσματικότητα των banners στον Ελληνικό χώρο είναι σημαντικά μικρότερες σε αριθμό. Οι περισσότερες σχετικές έρευνες αναφέρονται γενικά στη ψηφιακή διαφήμιση και όχι συγκεκριμένα στα ψηφιακά banner. Η Μπραίμη (2017), σε έρευνα με δείγμα καταναλωτών του Πειραιά, μελέτησε τους παράγοντες που βοηθούν μια ψηφιακή διαφήμιση στο να έλξει την προσοχή των καταναλωτών. Συγκεκριμένα, ο παράγοντας με την μεγαλύτερη επίδραση είναι το περιεχόμενο της και στη συνέχεια ακολουθούν παρουσιάζονται τα στοιχεία η

μουσική που ενδεχομένως αυτό περιέχει, καθώς και το κείμενο-μήνυμα, ο γραφικός σχεδιασμός της, τα χρώματα που περιέχει, με το μέγεθος της διαφήμισης να έχει την μικρότερη επίδραση. Στην ίδια έρευνα φαίνεται ότι οι χρήστες του διαδικτύου δεν αλληλεπιδρούν με τις διαφημίσεις τύπου banner, αφού μόλις το 13% κάνουν κλικ σε αυτές τις διαφημίσεις πολύ ή πάρα πολύ, ενώ φαίνεται ότι τα ποσοστά των καταναλωτών των οποίων επηρεάζεται η καταναλωτική συμπεριφορά, είναι σχεδόν μοιρασμένα.

Επιπλέον, πολλές έρευνες δείχνουν ότι οι ηλεκτρονικές διαφημίσεις banner αυξάνουν την αναγνωρισιμότητα των σημάτων των εταιριών που διαφημίζουν και μερικές μάλιστα από αυτές τις διαφημίσεις βοηθούν τους καταναλωτές ώστε να αναπτύξουν ευνοϊκή στάση για τα προϊόντα και τις υπηρεσίες αυτών. Ακόμη και όταν το click-througt rate (είναι ο αριθμός των κλικ που λαμβάνει η διαφήμιση διαιρεμένος με τον αριθμό των φορών που προβάλλεται η διαφήμιση, υπό τη μορφή ποσοστού) είναι χαμηλό και οι καταναλωτές συχνά αποφεύγουν να κοιτάζουν άμεσα μια διαφήμιση banner, η διαφήμιση αυτή επιδρά ευνοϊκά και ενισχύει την αναγνωρισιμότητα του σήματος της εταιρίας. Η καινοτομική κάθετη μορφή των διαφημίσεων banner φαίνεται ακόμη να προσελκύει το ενδιαφέρον των καταναλωτών σε σχέση με τις οριζόντιες. Οι διαφημίσεις banner σε κάθετη μορφή φαίνονται κατά κάποιο τρόπο περισσότερο διασκεδαστικές και εμφανίζουν περισσότερες πληροφορίες από τις οριζόντιες. Πρόσφατα στοιχεία δείχνουν ότι η συμπεριφορά των καταναλωτών ως προς το σχήμα που έχει μια τέτοιου τύπου διαφήμιση συνδέεται με τη συμπεριφορά του ως προς αυτή (Flores et al, 2014).

Από την άλλη πλευρά, ένα βασικό πρόβλημα που αντιμετωπίζουν οι ηλεκτρονικές διαφημίσεις banner, αφορά στην άμυνα που αναπτύσσουν οι χρήστες του διαδικτύου αποφεύγοντας να δουν τις συγκεκριμένες διαφημίσεις, το φαινόμενο αυτό ονομάζεται «banner blindness». Η Barreto (2013) παρουσιάζει στο άρθρο της πολλές έρευνες που έχουν πραγματοποιηθεί προκειμένου να ερευνηθούν το φαινόμενο αυτό, μια εκ των οποίων δείχνει ότι οι εμφανείς, έγχρωμες και με σχέδια διαφημίσεις αποφεύγονται από τους χρήστες εφόσον εκείνοι αναζητούν σε ένα ιστότοπο συγκεκριμένες πληροφορίες αδιαφορώντας για οποιεσδήποτε άλλες. Ταυτόχρονα, το ίδιο μπορεί να συμβεί, με βάση άλλη έρευνα, ακόμη και σε banner κειμένου που δε μοιάζουν με διαφημίσεις. Τα αποτελέσματα άλλων ερευνών έδειξαν ακόμη, ότι οι χρήστες βλέποντας μια ιστοσελίδα γνωρίζουν άμεσα που τοποθετούνται οι διαφημίσεις banner και τις αποφεύγουν.

Στον αντίποδα αυτών των ερευνών υπάρχουν άλλες, τα αποτελέσματα των οποίων έδειξαν είτε υψηλά επίπεδα αναγνώρισης και ανάκλησης διαφημιστικών μηνυμάτων, είτε ότι οι πιο ελκυστικές banner διαφημίσεις προσελκύουν την προσοχή, είτε ακόμη και ότι πάνω από τους μισούς συμμετέχοντες είδαν διαφημίσεις banner. Σε γενικές γραμμές, το φαινόμενο μπορεί να περιγραφεί ως ένας μηχανισμός άμυνας που βασίζεται στην περιφερειακή όραση των χρηστών και

στην εμπειρία του παρελθόντος. Εκφράζει ακόμα, την προκατάληψη που υπάρχει ότι οι πληροφορίες που εμπεριέχονται σε ένα πλαίσιο δεν είναι ενδιαφέρουσες ή σχετικές με αυτό που οι χρήστες αναζητούν, ανεξάρτητα από το αν έχουν τη μορφή διαφήμισης ή αν μοιάζουν με εμπορικές πληροφορίες. Ένας σημαντικός παράγοντας που υποστηρίζει τη θεωρία αυτή είναι το αναγνωρισμένο φαινόμενο «visual-spatial attention», με βάση το οποίο δίνεται προσοχή σε πολύ συγκεκριμένες πτυχές του περιβάλλοντος οδηγώντας τα άτομα σε καλύτερη επεξεργασία των εξωτερικών ερεθισμάτων που δέχονται. Η «βελτιωμένη» αυτή επεξεργασία ωστόσο επιβαρύνει την επεξεργασία των υπόλοιπων πληροφοριών που παρουσιάζονται ταυτόχρονα. Ένας άλλος παράγοντας ακόμη, είναι το λεγόμενο «cocktail party effect» που περιγράφεται ως η ικανότητα να εστιάσει κανείς την προσοχή του σε ένα μόνο «ομιλητή» ανάμεσα σε μια какоφωνία συνομιλιών και θορύβων του περιβάλλοντος. Και οι δυο αυτοί παράγοντες απεικονίζουν τους επιλεκτικούς μηχανισμούς προσοχής που χρησιμοποιούν οι χρήστες του διαδικτύου και επηρεάζουν την παρακολούθηση των διαφημίσεων banner.

Για μια εταιρία είναι μια σημαντική απόφαση να συμπεριλάβει διαφημίσεις banner στην ιστοσελίδα της μάρκας της, δεδομένου ότι υπάρχει μια σειρά από συναφή κόστη και οφέλη. Πέρα όμως από τα προφανή οφέλη, τα κόστη που καλείται να αντιμετωπίσει αφορούν στην «αταξία» που θα φέρει μια τέτοια διαφήμιση στην ιστοσελίδα, στις αρνητικές επιδράσεις που θα έχει για την εικόνα της μάρκας μια δυσμενής σύνδεση με το εμπορικό σήμα της διαφήμισης, στη δυσαρμονία μεταξύ του περιεχομένου της διαφήμισης και της ιστοσελίδας που τη φιλοξενεί και τέλος η απώλεια χώρου στην ιστοσελίδα. Τα διοικητικά στελέχη μιας τέτοιας εταιρίας συχνά λαμβάνουν την απόφαση τοποθέτησης διαφημίσεων banner με μόνο κριτήριο τα οικονομικά οφέλη που θα έχουν από αυτή ώστε να διατηρήσουν την ιστοσελίδα τους. Από τις αρνητικές επιπτώσεις όμως που αναφέραμε παραπάνω γίνεται κατανοητό ότι μια τέτοια απόφαση είναι πολύ πιο περίπλοκη και επηρεάζεται από μια σωρεία άλλων παραγόντων (Newman et al, 2004).

Τέλος η ανασκόπηση της βιβλιογραφίας αποκάλυψε τέσσερις δομικούς παράγοντες, οι οποίοι επηρεάζουν τη συμπεριφορά του χρήστη για αποτελεσματικότερη διαφήμιση, όπως α) τα κινούμενα σχέδια, β) η φωτεινότητα του χρώματος, γ) η προεξέχουσα θέση και δ) το μέγεθος.

Σύμφωνα με την έρευνα της Mattke (2019) τα αποτελέσματα δείχνουν ότι οι διαφημίσεις με κινούμενα σχέδια αυξάνουν την πιθανότητα ότι ένας χρήστης θα κάνει κλικ (Bruce et al., 2017, Chandon et al., 2003, Lohtia et al., 2003, Rosenkrans 2009, Yoo et al., 2004). Το animation των διαφημίσεων ορίζεται ως μια "προγραμματισμένη εμφάνιση διαδοχικών εικόνων, δημιουργώντας την ψευδαίσθηση ότι αντικείμενα στην εικόνα αιωρούνται" (IAB 2019). Η κίνηση των διαφημίσεων βρέθηκε να σχετίζεται θετικά με τη συμπεριφορά των κλικ του χρήστη.

Η έρευνα έδειξε επίσης ότι οι διαφημίσεις online με λαμπρότερο και φωτεινότερο χρώμα

είναι πιο πιθανό να κεντρίσουν το ενδιαφέρον του καταναλωτή (Lohtia et al., 2003, Sokolik et al., 2014).

Επιπλέον, οι διαφημίσεις που εμφανίζονται στο πάνω μέρος της οθόνης είναι πιο πιθανό να είναι αποτελεσματικότερες και στη συνέχεια οι διαφημίσεις στο μεσαίο ή κατώτερο τμήμα μιας ιστοσελίδας (Abhishek et al., 2015, Agarwal et al., 2011, Agarwal and Mukhopadhyay 2016, Animesh κ.ά., 2011, Chan and Young-Hoon 2015, Gong κ.ά., 2018, Narayanan and Kalyanam 2015, Rutz κ.ά., 2012). Έτσι, η προβολή της θέσης σχετίζεται θετικά με τη συμπεριφορά του καταναλωτή.

Τέλος, η έρευνα εξέτασε την επίδραση του μεγέθους της διαφήμισης. Η έρευνα αποκαλύπτει ότι οι χρήστες είναι πιο πιθανό να κάνουν κλικ σε μεγαλύτερες διαφημίσεις στο διαδίκτυο (Bruce et al., 2017, Chandon et al., 2003, Rosenkrans 2009, Sokolik et al., 2014), απο ότι σε μικρότερες. Κατά συνέπεια, ένα μεγαλύτερο μέγεθος μιας διαφήμισης σχετίζεται με την υψηλότερη πιθανότητα της θετικής συμπεριφοράς σε κλικ του χρήστη.

3.2 Τα βασικά στοιχεία της αποτελεσματικότητας της διαφήμισης των banners

Το μεγάλο μέγεθος της διαφήμισης banner, τα μηνύματα-πληροφορίες που εμπεριέχει ως προς το πλήθος τους και η ανυπαρξία άμεσων κινήτρων προώθησης είναι σημαντικοί παράγοντες που επηρεάζουν θετικά την αποτελεσματικότητα αυτής, με βάση την έρευνα των Robinson et al. (2007). Κολοσσοί του διαδικτύου όπως η Google και η Yahoo, μέσα από έρευνες και πληροφορίες που παρατίθενται σε διαδικτυακά σεμινάρια, έχουν εντοπίσει και παρουσιάσει τα στοιχεία που μπορούν να κάνουν μια διαφήμιση banner αποτελεσματική.

Πιο αναλυτικά, η Yahoo υποστηρίζει ότι η τοποθέτηση της διαφήμισης banner παίζει καθοριστικό ρόλο στην αποτελεσματικότητα που θα έχει. Συγκεκριμένα, στην πρώτη θέση, ως προς την αποτελεσματικότητα της διαφήμισης, έρχεται η τοποθέτηση στην κορυφή της ιστοσελίδας που τη φιλοξενεί ενώ στη δεύτερη η τοποθέτηση εντός του περιεχομένου αυτής. Όσον αφορά το μέγεθος της διαφήμισης, στα διαδικτυακά σεμινάρια της Google αναφέρεται ότι οι μεγάλες ορθογώνιες διαφημίσεις banner τείνουν να είναι περισσότερο αποτελεσματικές, ωστόσο προειδοποιεί ότι μπορεί να χρειάζεται κάποια προσαρμογή καθώς άλλες μορφές διαφήμισης πιθανώς είναι καταλληλότερες για διαφορετικούς τύπους περιεχομένου (Cho, 1999).

Πολύ σημαντικό όμως παράγοντα για την αποτελεσματικότητα μιας τέτοιας διαφήμισης, με βάση το πλήθος των δημοσιεύσεων που υποστηρίζουν αυτή την άποψη, διαδραματίζει η στενή σύνδεση του περιεχομένου-αντικειμένου της διαφήμισης με το αντίστοιχο της ιστοσελίδας στην οποία φιλοξενείται (Hofacker, C. and Murphy J. 1998; Becker-Olsen, 2003; Newman et al, 2004;

Burke et al, 2005). Οι σχεδιαστές και οι ιδιοκτήτες ιστοσελίδων πρέπει να δημοσιεύουν διαφημίσεις που σχετίζονται στενά με το περιεχόμενο της σελίδας τους ελπίζοντας με αυτό τον τρόπο να προσελκύσουν μεγαλύτερο αριθμό χρηστών να τις επιλέξουν. Οι συμμετέχοντες στην έρευνα των Burke et al. (2005) είχαν σαν πρωταρχική σκέψη να μη δουν καμία διαφήμιση banner, ανεξάρτητα από το πόσο ελκυστική ήταν και το πιο αξιοπρόσεκτο από όλα είναι ότι καμία κίνηση, σχέδιο ή εικόνα δεν έκανε «εμφανή» μια άσχετη με το περιεχόμενο της ιστοσελίδας διαφήμιση. Η σύνδεση των στόχων συνεπώς των χρηστών με μια διαφήμιση μπορεί να κάνει την τελευταία περισσότερο επιτυχημένη. Η υψηλή συσχέτιση μεταξύ διαφήμισης και ιστοσελίδας μπορεί ακόμη να βελτιώσει σημαντικά τη στάση του καταναλωτή- χρήστη ως προς την εταιρία, την ανταπόκριση του σε αυτή αλλά και την αντιλαμβανόμενη ποιότητα των προϊόντων της (Becker-Olsen, 2003).

Οι Hofacker και Murphy (1998) από την πλευρά τους επισημαίνουν την υψηλή συσχέτιση της ιστοσελίδας που φιλοξενεί τη διαφήμιση banner με το περιεχόμενο της τελευταίας ενώ παράλληλα, υπογραμμίζουν τη σημασία της μη επαναλαμβανόμενης έκθεσης αυτής καθώς το ποσοστό ανταπόκρισης μειώνεται δραματικά μετά την πρώτη προβολή.

3.3 Σχεδίαση μιας αποτελεσματικής banner διαφήμισης

Η ουσία λοιπόν των banners είναι να περάσει μέσω γραφικών το μήνυμα της επιχείρησης, σε αντίθεση με το κείμενο που μπορεί να είναι πληκτικό ή κουραστικό για τον αναγνώστη. Τα χρώματα και οι εικόνες προσδίδουν ζωνρότητα στη διαφήμιση, προσελκύοντας το ενδιαφέρον του καταναλωτή να μάθει ακόμη περισσότερα για τα προϊόντα και τις υπηρεσίες της επιχείρησης. (Καράμπελα Χ., 2012).

Μια επιχείρηση που στηρίζει τις πωλήσεις της στη διαδικτυακή διαφήμιση μέσω των banners θα πρέπει να λάβει σοβαρά υπόψη της την αύξηση της επισκεψιμότητας της ιστοσελίδας, ακολουθώντας τους παρακάτω κανόνες:

- Τα banners θα πρέπει να καταχωρούνται σε ιστότοπους σχετικού περιεχομένου με αυτό του διαφημιζόμενου.
- Ένα προϊόν θα πρέπει να διαφημίζεται σε ένα banner, για να αποφεύγεται η τυχόν σύγχυση με την διαφήμιση περισσότερων του ενός προϊόντων.
- Ο σύνδεσμος του banner θα ήταν προτιμότερο να οδηγεί τον επισκέπτη απευθείας στο διαφημιζόμενο προϊόν και όχι στην αρχική σελίδα της εταιρείας.
- Το banner θα ήταν καλύτερο να τοποθετηθεί στην αρχή και όχι στο τέλος της ιστοσελίδας.
- Το animation θα πρέπει να κυριαρχεί σε σχέση με το κείμενο.
- Η σωστή σχεδίαση του banner είναι σημαντική για την θετική εντύπωση που μπορεί να

αποκομίσει ο χρήστης για την εταιρεία.

Τα οφέλη του animation στη διαφήμιση είναι πολύ σημαντικά καθώς επιδεικνύει τη χρήση και τα μέρη του προϊόντος, χωρίς να χρειάζεται ζωντανή παρουσίαση, δημιουργώντας μια αριστοκρατική αίσθηση του προϊόντος ή της υπηρεσίας. Το προϊόν φαίνεται πιο ελκυστικό, με πιο δυναμική εμφάνιση, «σπάζοντας» την μονοτονία με τη χρήση γραφικών και κινουμένων σχεδίων. Παρουσιάζει το προϊόν με τεχνολογία αιχμής και αναδεικνύει το ρόλο της, δημιουργώντας κάτι νέο που δεν μπορεί να επιτευχθεί μέσω ζωντανών λήψεων. (Καράμπελα Χ., 2012).

Το μήνυμα και το δημιουργικό

Σε όλες τις μορφές δημιουργικού, προτείνεται το μέγεθος του αρχικού αρχείου να μην ξεπερνά τα 30Κ, ενώ αν περιέχει video, αυτό να μην ξεπερνά τα 15 δευτερόλεπτα σε διάρκεια.

- Προτείνεται η διακριτική χρήση του label «Διαφήμιση» ή «Advertisement».
- Να μην χρησιμοποιούνται στο μήνυμα εικόνες ή λέξεις που παραπέμπουν σε οικίες λειτουργίες του λειτουργικού συστήματος ή δημοφιλών προγραμμάτων, με σκοπό την παραπλάνηση του χρήστη.
- Μηνύματα που αφορούν σε κάποιο βραβείο θα πρέπει να συμμορφώνονται με τους ισχύοντες κανόνες διαφήμισης.
- Η χρήση εναλλασσόμενων χρωμάτων υψηλής αντίθεσης δεν προκαλεί εντυπωσιασμό αλλά ενόχληση.

Ήχος

Ο ήχος θα πρέπει πάντα να είναι user-initiated, με μοναδική εξαίρεση αν αποτελεί κομμάτι άλλου stream που έχει ήδη αποδεχθεί ο χρήστης.

Κουμπιά Ελέγχου

Όλα τα διαφημιστικά formats που χρησιμοποιούν ήχο ή/και video θα πρέπει να περιλαμβάνουν κουμπιά ελέγχου (ήχος on/off, play, pause, stop). Τα κουμπιά αυτά θα πρέπει:

- Να εμφανίζονται σε κάθε frame του video
- Να φαίνονται καθαρά (με προσοχή στη χρωματική παλέτα της σελίδας όπου φιλοξενείται η διαφήμιση)
- Να βρίσκονται σε θέση που φαίνεται καθαρά.

Κουμπί κλεισίματος (close)

Το κουμπί κλεισίματος θα πρέπει

- Να εμφανίζεται σε κάθε frame του video
- Να εμφανίζεται σε σταθερή θέση (την πάνω δεξιά γωνία του δημιουργικού). Σε περιπτώσεις expandable, προτείνεται να μην βρίσκεται στην expanded περιοχή

- Να φαίνεται καθαρά (με προσοχή στη χρωματική παλέτα της σελίδας όπου φιλοξενείται η διαφήμιση)
- Να περιέχει είτε ένα X, είτε τη λέξη ‘CLOSE’ σε font size 16, ή και τα δύο

Σε περιπτώσεις expandable δημιουργικών, το δημιουργικό πρέπει να κλείνει αυτόματα μόλις το ποντίκι απομακρυνθεί από την περιοχή του expanded δημιουργικού. Σε περιπτώσεις between-the-page (transitional) διαφημίσεων, η συνήθης πρακτική είναι η χρήση της έκφρασης “Skip Advertisement” στο κάτω μέρος του δημιουργικού. Τέλος, σε περιπτώσεις pop-ups ή pop-unders, ο browser παρέχει τον συγκεκριμένο έλεγχο.

Frequency Capping

Θα πρέπει να υπάρχει μια ισορροπία μεταξύ του διαφημιστικού όγκου στον οποίο εκτίθεται ο χρήστης κατά την επίσκεψή του σε ένα website, και του όγκου του content της σελίδας. Μεγάλη επαναληπτικότητα σε μορφές rich media εκνευρίζει τους χρήστες και αποδυναμώνει το μήνυμα. Ως εμπειρικό κανόνα, οι χρήστες δεν θα πρέπει να εκτίθενται σε περισσότερα από τρία ‘intrusive’ μηνύματα (rich media, pop-ups ή pop-unders) σε μία τυπική επίσκεψή τους σε ένα website.

3.4 Σκοπός και στόχοι της έρευνας

Η ανάπτυξη του διαδικτύου έχει λάβει σημαντική προσοχή από τους καταναλωτές, για το λόγο αυτό πολλοί ερευνητές προσπάθησαν να διερευνήσουν τον αντίκτυπο της διαφήμισης banner για τους καταναλωτές, τόσο από ψυχολογική όσο και απο συμπεριφορική αντίδραση (Cho, Lee, & Tharp, 2001, Razzouk & Seitz, 2003, Diao & Sundar, 2004). Ωστόσο, κανένας τύπος της διαφήμισης banner δεν έχει ερευνηθεί ακόμη που να είναι αποτελεσματικός, και η επιλογή των διαφημίσεων banner καθώς τα διαφημιστικά ερεθίσματα έγιναν με βάση την προσωπική κρίση της κάθε εταιρίας που θέλει να διαφημιστεί, αντί της επιστημονικής έρευνας.

Για το λόγο αυτό, καθώς υπάρχει επιστημονικό κενό απο την σύγχρονη βιβλιογραφία, σκοπός της παρούσας εργασίας είναι να εντοπιστεί ο τύπος και τα χαρακτηριστικά της διαφήμισης banner που είναι αποτελεσματικότερα κατά την άποψη των καταναλωτών, λαμβάνοντας υπόψη, κάποιες παραμέτρους που θα αποτελέσουν και τις ερευνητικές υποθέσεις της παρούσας.

Συγκεκριμένα, θα λάβουμε υπόψη α) τον τύπο της διαφήμισης banner, β) τον αριθμό banners που θα πρέπει να προβάλλονται σε έναν ιστότοπο, γ) τη θέση που θα πρέπει να καταλαμβάνει ένα banner σε μια ιστοσελίδα, δ) το ιδανικό σχήμα του banner, ώστε να το επισκεφτούν περισσότεροι χρήστες και ε) το μέγεθος του banner που θα «τραβήξει» την προσοχή του καταναλωτή. Όλα τα ανωτέρω στοιχεία μπορούν να διαδραματίζουν κρίσιμο ρόλο στην καθιέρωση ενός συστήματος μέτρησης της αποτελεσματικότητας της διαδικτυακής διαφήμισης της μορφής banner.

Με βάση την ανασκόπηση της βιβλιογραφίας και του ερευνητικού σκοπού της παρούσας εργασίας, διαμορφώνονται οι ακόλουθοι επιμέρους ερευνητικοί στόχοι:

- Να προσδιοριστεί η αποτελεσματικότερη μορφή/ τύπος της διαφήμισης banner.
- Να προσδιοριστεί ο αριθμός των διαφημιστικών banner, που είναι ιδανικά να υπάρχουν σε μια ιστοσελίδα.
- Να προσδιοριστεί ποιο σχήμα διαφήμισης banner, είναι πιο ελκυστικό.
- Να προσδιοριστεί η ιδανική τοποθεσία προβολής της διαφήμισης banner.
- Να προσδιοριστεί το σωστό μέγεθος της διαφήμισης banner.

3.5 Ερευνητικές Υποθέσεις

Οι ερευνητικές μας υποθέσεις θα βασιστούν στα τεχνικά χαρακτηριστικά που θα πρέπει να έχει μια διαφήμιση banner, όπως είναι ο τύπος, ο αριθμός, το σχήμα, η τοποθεσία και το μέγεθος της. Αναλυτικότερα:

Τύπος

Υ1: Σε ένα banner θα πρέπει να κυριαρχούν τα πολυμέσα (animations) σε σχέση με το κείμενο.

Γενικά, οι τύποι διαφήμισης banner, όπως έχουμε ήδη αναφέρει και στις προηγούμενες ενότητες, είναι στατικές, αναδύμενες, κινούμενες, δυναμικές, και περιστρεφόμενες (E.Zlistmailer.com, n.d.). Μια στατική διαφήμιση banner ούτε μετακινεί ούτε αλλάζει το περιεχόμενό της σε κάθε σελίδα φόρτωσης και περιλαμβάνει μόνο ένα αρχείο εικόνας GIF ή JPEG. Η στατική διαφήμιση banner θεωρείται παθητική, επειδή δεν διακόπτει τη δραστηριότητα των επισκεπτών (Cho & Leckenby, 2000, Chatterjee, Hoffman & Novak, 2003). Αντίθετα, μια αναδύμενη διαφήμιση εμφανίζεται σε ξεχωριστό μικρό παράθυρο στην κορυφή της οθόνης του επισκέπτη κατά την πλοήγηση από μια ιστοσελίδα σε μια άλλη και θεωρείται παρεμβατική επειδή διακόπτει την πλοήγηση στον ιστό ενός επισκέπτη (Edwards, Li, & Lee, 2002) ιδιαίτερα όταν οι πληροφορίες που εμφανίζονται δεν ταιριάζουν με το ενδιαφέρον ενός επισκέπτη.

Η κινούμενη διαφήμιση banner ή διαφήμιση που χρησιμοποιεί πολυμέσα -animations-είναι σύμφωνα με πολλούς ερευνητές ένας από τους πιο αγαπημένους και αποτελεσματικότερους τύπους διαφήμισης banner μεταξύ των online καταναλωτών και οι διαφημιζόμενοι τη χρησιμοποίησαν με επιτυχία γιατί βασίστηκαν στην προσοχή και στη συνειδητοποίηση των χρηστών (Li & Bukovac, 1999, Dreze & Hussherr, 2003, Diao & Sundar, 2004, Bruce et al., 2017, Chandon et al., 2003, Lohtia et al., 2003, Rosenkrans 2009, Yoo et al. 2004). Η δυναμική αυτή του τύπου διαφήμιση banner αποτελείται από ήχο, βίντεο, εικόνες, φωτεινά και λαμπερά χρώματα, και μοιάζει περισσότερο σαν τις τηλεοπτικές διαφημίσεις (Koegel, 2003). Έχει σχεδιαστεί ώστε να επηρεάσει

τους καταναλωτές με τον ίδιο τρόπο όπως κάνει και το τηλεοπτικό κανάλι με μια πρόσθετη διάσταση της διαδραστικότητας.

Αριθμός Προβολής Διαφήμισης Banner

Y2: Η προβολή ενός banner σε ένα ιστότοπο και όχι περισσότερων, ώστε να αποφεύγεται η σύγχυση με τα υπόλοιπα διαφημιστικά banners

Αφορά το συνολικό αριθμό διαφημίσεων που μπορούν να προβάλλονται σε έναν ιστοχώρο. Ο υπερβολικός αριθμός διαφημίσεων που μπορεί να εμφανίζονται σε μια ιστοσελίδα επιλογής του χρήστη, θεωρείται ως μια σημαντική μεταβλητή που επηρεάζει την στάση των καταναλωτών απέναντι στη διαφήμιση (Ray & Webb, 1986, Kent, 1993, Ha, 1996, Ha & Litmann, 1997, Elliott & Speck, 1998). Πράγματι, η παρουσία πολλών ανταγωνιστικών διαφημίσεων οδηγεί σε έλλειψη ανάκλησης του μηνύματος της διαφήμισης (Keller, 1987, Burke & Srull, 1988), και υπονομεύει τη στάση ενός ατόμου απέναντι στη διαφήμιση (Keller, 1991). Έτσι, πολλοί διαφημιστές διστάζουν να τοποθετήσουν πολλές διαφημίσεις banner σε ιστότοπους. Τα παραπάνω μας οδηγούν και στην κάτωθι ερευνητική υπόθεση της εργασίας μας:

Σχήμα

Y3: Τα κάθετα banners είναι πιο ελκυστικά από τα υπόλοιπα σχήματα banners

Παρά τον μεγάλο αριθμό ερευνητικών μελετών που πραγματοποιήθηκαν σχετικά με τη διαφήμιση μέσω διαδικτύου, ελάχιστη προσοχή έχει δοθεί στην επίδραση του σχήματος της διαφήμισης banner. Μερικές έρευνες έχουν δείξει ότι το κάθετο banner είναι πιο αποτελεσματικό από το οριζόντιο banner, το οποίο τοποθετείται από αριστερά προς τα δεξιά (Dreze & Hussherr, 2003). Έτσι η επόμενη ερευνητική υπόθεση που θα κάνουμε είναι:

Τοποθεσία

Y4: Το banner θα πρέπει να τοποθετηθεί στην κορυφή μιας ιστοσελίδας και όχι στο τέλος της

Η θέση των διαφημίσεων έχει σημαντικό αντίκτυπο στην αποτελεσματικότητα της διαφήμισης στο διαδίκτυο (Hofacker & Murphy, 1998). Ο Janiszewski (1990) σε έρευνά του κατέληξε στο συμπέρασμα ότι δύο διαφορετικές πλευρές του εγκεφάλου εκτελούν μια διαφορετική επεξεργασία πληροφοριών. Η αριστερή πλευρά είναι η λογική επεξεργασία πληροφοριών ενώ η δεξιά πλευρά εκτελεί την οπτική επεξεργασία. Στο διαδίκτυο, έρευνες έχουν δείξει ότι οι διαφημίσεις banner που τοποθετούνται στην κορυφή μιας ιστοσελίδας έχουν μεγαλύτερη πιθανότητα να τραβήξουν την προσοχή του χρήστη από όταν είναι τοποθετημένες στο κάτω μέρος ή στη μέση μιας ιστοσελίδας (Abhishek et al., 2015, Agarwal et al., 2011, Agarwal and

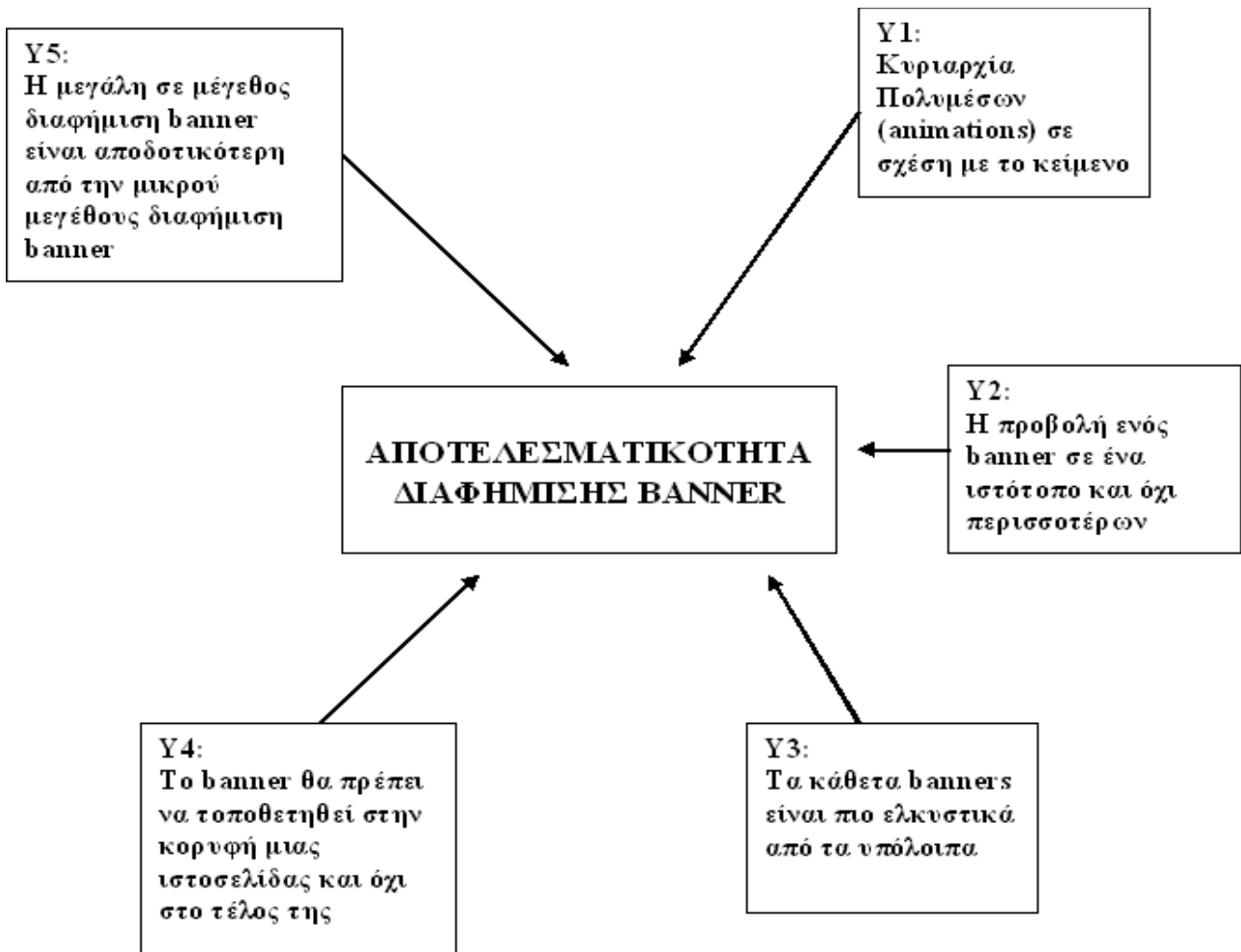
Mukhopadhyay 2016, Animesh κ.ά., 2011, Chan and Young-Hoon 2015, Gong κ.ά., 2018, Narayanan and Kalyanam 2015, Rutz κ.ά., 2012). Επομένως μια άλλη ερευνητική υπόθεση που μπορούμε να κάνουμε είναι η εξής:

Μέγεθος

Y5: Η μεγάλη σε μέγεθος διαφήμιση banner είναι αποδοτικότερη από την μικρού μεγέθους διαφήμιση banner

Παραδοσιακά, το μέγεθος μιας διαφήμισης επηρεάζει την προσοχή (Hendon, 1973), την μνήμη, την ανάκληση του μηνύματος και την αναγνώριση της επωνυμίας ενός προϊόντος (Homer, 1995, Kirmani, 1990). Στην πράξη, μια μεγαλύτερη διαφήμιση είναι πιο αποτελεσματική από μια μικρότερη (Bruce et al., 2017, Chandon et al., 2003, Rosenkrans 2009, Sokolik et al., 2014). Στο διαδίκτυο, ο αντίκτυπος της διαφήμισης banner για το μέγεθος των απαντήσεων των καταναλωτών είναι ανάμεικτο. Li και Bukovac (1999) σε έρευνά τους διαπίστωσαν ότι οι μεγαλύτερες διαφημίσεις banner είναι καλύτερα κατανοητές από τις μικρότερες διαφημίσεις, αλλά δεν δημιουργούν απαραίτητα περισσότερη διαδραστικότητα (Euijin & Edwards, 2002). Ομοίως, οι Kim και Leckenby (2002) δεν βρήκαν οποιαδήποτε σημαντική σχέση μεταξύ του μεγέθους διαφήμισης banner και της αναλογίας των κλικ προς τον αριθμό εμφανίσεων. Μία πιθανή εξήγηση για τα αποτελέσματα αυτά θα μπορούσε να οφείλεται στο μικρό μέγεθος της οθόνης και τη διαθεσιμότητα περιορισμένου χώρου για τον χειρισμό του μεγέθους διαφημίσεων banner. Επομένως η τελευταία ερευνητική υπόθεση που θα γίνει είναι:

Οι ανωτέρω ερευνητικές υποθέσεις παρουσιάζονται καλύτερα και παρακάτω σχηματικά:



Σχήμα 2: Σχηματική παράσταση ερευνητικών υποθέσεων

3.6 Ανάλυση Ερευνητικών Υποθέσεων

Στη συνέχεια θα αποτυπώσουμε τις ανωτέρω ερευνητικές υποθέσεις με βάση το ερωτηματολόγιο που σχεδιάσαμε για την παρούσα έρευνα και αναλύεται ως εξής:

3.6.1 Ερευνητική Υπόθεση 1

Η ερευνητική υπόθεση Y1 απαντάται από τις ερωτήσεις 6 και 7 του ερωτηματολογίου και συγκεκριμένα:

Κατά πόσο τα παρακάτω στοιχεία τραβούν την προσοχή σας σε μια διαφήμιση banner?

	1	2	3	4	5
Το σταθερό/ στατικό κείμενο					
Το κινούμενο μήνυμα ή κείμενο					
Οι πολλές εικόνες					
Τα έντονα/ζωηρά χρώματα					
Το υπόβαθρο του μηνύματος					
Η μουσική ή ο τυχόν ήχος					

Η ευκολία πλοήγησης στο προϊόν που διαφημίζεται					
Η ταχύτητα φόρτωσης της ιστοσελίδας του διαφημιζόμενου προϊόντος					

Ποια μορφή banner διαφήμισης παρατηρείτε περισσότερο στο διαδίκτυο?

	1	2	3	4	5
Τα στατικά (static) banners					
Τα κινούμενα (animated) banners					
Τα διαδραστικά (interactive) banners					
Τα αναδυόμενα (Pop-up) banners					
Τα banners που περιέχουν ενεργές λέξεις ή εικόνες					
Τα banners που περιέχουν παιχνίδια ή παζλ					
Τα banners που περιέχουν ήχο και video					
Τα banners σε μορφή ανακοινώσεων					
Τα banners με φωτεινά χρώματα που αναβοσβήνουν					

3.6.2 Ερευνητική Υπόθεση 2

Η ερευνητική υπόθεση Υ2 απαντάται από την ερώτηση 8 του ερωτηματολογίου και αφορά ποιος πρέπει να είναι ο ιδανικός αριθμός Προβολής διαφημίσεων banner σε μια ιστοσελίδα, ώστε να μην είναι ενοχλητικά και, ήτοι:

Πόσα σε αριθμό banners θεωρείτε ότι πρέπει να προβάλλονται σε μια ιστοσελίδα που επισκέπτεστε, έτσι ώστε να μην είναι ενοχλητικά και να τραβήξουν την προσοχή σας?

	1	2	3	4	5
Από 1 έως 2					
Από 2 έως 3					
Από 3 έως 4					
Από 4 έως 5					
Από 6 και πάνω					

3.6.3 Ερευνητική Υπόθεση 3

Η ερευνητική υπόθεση Υ3 απαντάται από την ερώτηση 9 του ερωτηματολογίου και αφορά τα Σχήματα διαφημίσεων banner, ήτοι:

Ποιο από τα παρακάτω σε σχήμα banner θα κάνατε κλικ?

	1	2	3	4	5
Οριζόντιο banner					
Κάθετο banner					
Τετράγωνο banner					
Στρογγυλό banner					

3.6.4 Ερευνητική Υπόθεση 4

Η ερευνητική υπόθεση Υ4 απαντάται από την ερώτηση 10 του ερωτηματολογίου και αφορά την Τοποθεσία προβολής των διαφημίσεων banner, ήτοι:

Σε ποια από τις παρακάτω τοποθεσίες μιας ιστοσελίδας που προβάλλονται διαφημίσεις banner θα κάνατε κλικ?

1	2	3
4	5	6
7	8	9

	1	2	3	4	5
Η τοποθεσία 1, πάνω αριστερά					
Η τοποθεσία 2, πάνω κεντρικά					
Η τοποθεσία 3, πάνω δεξιά					
Η τοποθεσία 4, μέση αριστερά					
Η τοποθεσία 5, κεντρικά					
Η τοποθεσία 6, μέση δεξιά					
Η τοποθεσία 7, κάτω αριστερά					
Η τοποθεσία 8, κάτω κεντρικά					
Η τοποθεσία 9, κάτω δεξιά					

3.6.5 Ερευνητική Υπόθεση 5

Η ερευνητική υπόθεση Υ5 απαντάται από την ερώτηση 11 του ερωτηματολογίου και αφορά τα διάφορα μεγέθη (διαστάσεις) των διαφημίσεων banner, όπου χρησιμοποιήσαμε τα πιο σημαντικά και χωρίζονται σε:

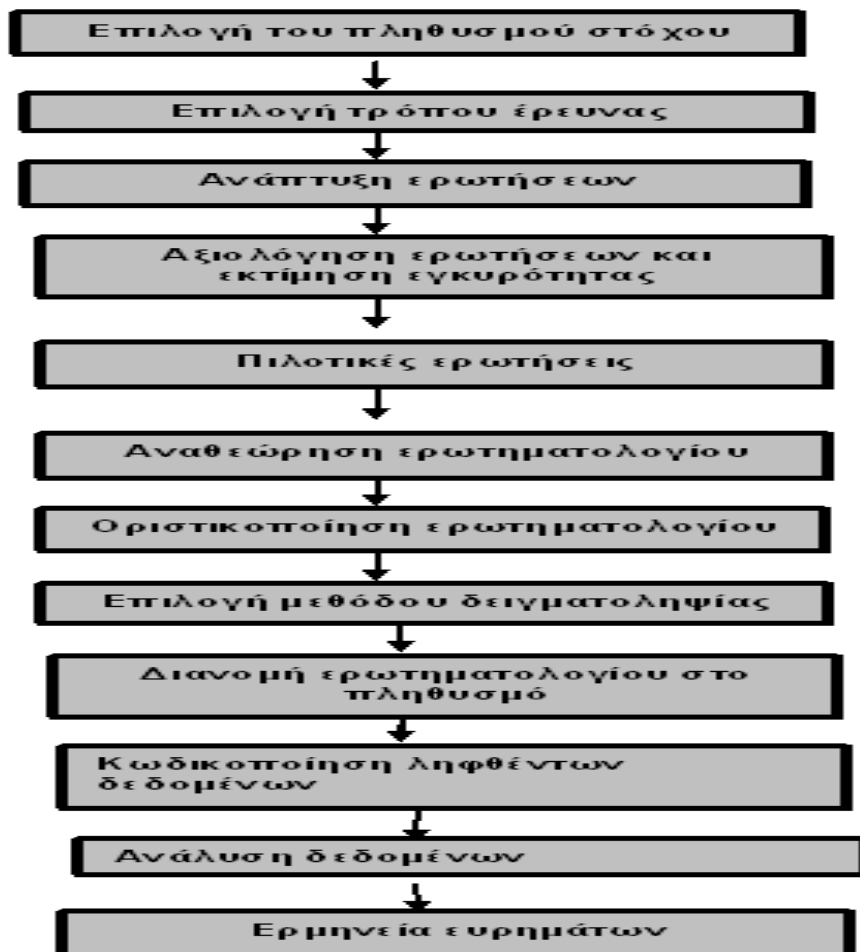
- Τετράγωνο Μεγάλο, Square Pop Up, 250*250
- Τετράγωνο Μικρό, Square Button, 125*125
- Οριζόντιο Κανονικό, Full Banner, 468*60
- Οριζόντιο Μικρό, Half Banner, 234*60
- Οριζόντιο Μεγάλο, Leaderboard, 728*90
- Κάθετο Μικρό, Wide Skyscraper, 160*600 και
- Κάθετο Μεγάλο, Half Page Ad, 300*600

ΚΕΦΑΛΑΙΟ 4: ΜΕΘΟΔΟΛΟΓΙΑ ΕΡΕΥΝΑΣ

4.1 Εισαγωγή

Στα πλαίσια εκπόνησης της παρούσας εργασίας, επιλέχθηκε η διεξαγωγή πρωτογενούς έρευνας, κάνοντας χρήση αντίστοιχων ποσοτικών μεθόδων, ενώ σχεδιάστηκε με τέτοιο τρόπο, ώστε να συλλέξει και να αξιοποιήσει κατάλληλα, πρωτογενή δεδομένα, που αφορούν την αποτελεσματικότητα της διαδικτυακής διαφήμισης στο καταναλωτικό κοινό.

Παράλληλα, σε όλες τις φάσεις της έρευνας λήφθηκε η κατάλληλη μέριμνα ώστε να εξασφαλιστεί στο μέγιστο δυνατό βαθμό η αξιοπιστία και η εγκυρότητα του περιεχομένου, τόσο του οργάνου μέτρησης της έρευνας, όσο και των ληφθέντων δεδομένων. Για το λόγο αυτό, ακολουθήθηκε το προτεινόμενο από τους Bryman & Bell (2011) πλαίσιο αναφοράς κατά τη διεξαγωγή κοινωνικής έρευνας, το οποίο περιλαμβάνει τα παρακάτω βήματα:



Σχήμα 3: Πλαίσιο διεξαγωγής έρευνας

4.2 Επιλογή τρόπου έρευνας

Στην παρούσα ενότητα αναπτύσσεται η μεθοδολογία της έρευνας που πραγματοποιήθηκε στα

πλαίσια της διπλωματικής. Ο Creswell (2009) στο βιβλίο του για το σχεδιασμό των ερευνών αναφέρει τρεις ερευνητικές προσεγγίσεις, τις ποιοτικές, τις ποσοτικές και τέλος τις μεικτές.

Για τους σκοπούς της παρούσας έρευνας, θα ακολουθήσουμε την ποσοτική μέθοδο, σχεδιάζοντας ένα ερωτηματολόγιο δεκαεπτά ερωτήσεων με πολλαπλές απαντήσεις. Οι συγκεκριμένες ερωτήσεις αποβλέπουν στην καλύτερη κατανόηση της συμπεριφοράς των χρηστών-καταναλωτών από το σύνολο της ηλεκτρονικής τους εμπειρίας εστιάζοντας στα συναισθήματα και τις αντιλήψεις τους για τις ηλεκτρονικές διαφημίσεις και στο κατά πόσο όλα τα παραπάνω συμβάλουν ή όχι στην αγορά των διαφημιζόμενων προϊόντων υψηλής τεχνολογίας.

Η ποσοτική έρευνα είναι μια προσέγγιση που βοηθάει στον έλεγχο της θεωρίας εξετάζοντας τη σχέση μεταξύ των μεταβλητών. Αυτές οι μεταβλητές, με τη σειρά τους, μπορούν να μετρηθούν, συνήθως με τη χρήση διαφόρων εργαλείων, έτσι ώστε τα δεδομένα σε αριθμητική μορφή να είναι επεξεργάσιμα με τη χρήση ορισμένων διαδικασιών στατιστικής ανάλυσης. Η τελική γραπτή αναφορά έχει συγκεκριμένη δομή, που αποτελείται από την εισαγωγή, τη βιβλιογραφία, τη θεωρία, τις μεθόδους, και τα αποτελέσματα. Όπως και στην περίπτωση της ποιοτικής έρευνας έτσι και εδώ αντίστοιχα, όσοι συμμετέχουν στις έρευνες αυτές υποστηρίζουν ένα τρόπο θεώρησης της έρευνας που βασίζεται στον απαγωγικό λογισμό, πραγματοποιώντας ελέγχους για την ύπαρξη εναλλακτικών ερμηνειών, καθιστώντας με τον τρόπο αυτό ευκολότερη τη γενίκευση και την αναπαραγωγή των ευρημάτων της έρευνας.

4.3 Εργαλείο μέτρησης/ Σχεδιασμός Ερωτηματολογίου

Ως ερευνητικό εργαλείο για τη συλλογή των δεδομένων της έρευνας, χρησιμοποιήθηκε το ερωτηματολόγιο, το οποίο συμπληρώθηκε έντυπα από τους ίδιους τους συμμετέχοντες στην έρευνα (self-administered), περιλαμβάνοντας σύντομες σε μέγεθος και διατυπωμένες με σαφήνεια κλειστού τύπου (ως επί το πλείστον) ερωτήσεις, καθώς αυτού του είδους οι ερωτήσεις αφενός θεωρούνται ως οι πιο δόκιμες για στατιστική ανάλυση και κωδικοποίηση ποσοτικών δεδομένων, και αφετέρου γιατί οι απαντήσεις από τους συμμετέχοντες θεωρούνται πιο εύκολες και χρονικά πιο γρήγορες (Bryman & Bell, 2011).

Τα είδη των ερωτήσεων που χρησιμοποιήθηκαν και οι οποίες ήταν χωρισμένες εννοιολογικά σε τρία μέρη: Α) «Δημογραφικά Στοιχεία» με 3 τύπου ερωτήσεις, Β) «Προϊόντα Υψηλής Τεχνολογίας» με δύο (2) τύπου ερωτήσεις και Γ) «Τεχνικά Στοιχεία της διαφήμισης Banner» με 6 εξειδικευμένες ερωτήσεις- που αφορούν τις ερευνητικές υποθέσεις που κάναμε στην ανωτέρω ενότητα και περιελάμβαναν ερωτήσεις κλειστού τύπου της πενταβάθμιας κλίμακας likert (Likert scale), οι οποίες κυμαίνονταν το ένα (1) ως το πέντε (5), όπου το 1 θα υποδηλώνει «Καθόλου», το 2

«Λίγο», το 3 «Μέτρια», το 4 «Πολύ» και το 5 «Πάρα Πολύ». Ειδικά στοχευμένες ερωτήσεις για την τρέχουσα αποτύπωση ορισμένων καταναλωτικών συνηθειών στην παρακολούθηση των διαφημίσεων banner και στην εξαγωγή χρήσιμων συμπερασμάτων.

4.4 Αξιοπιστία – Εγκυρότητα έρευνας

Για την εξασφάλιση ακόμη περισσότερο της αξιοπιστίας και της εγκυρότητας του ερωτηματολογίου, τέθηκε με την ίδια σειρά σε όλους τους συμμετέχοντες, ο ίδιος αριθμός από κυρίως υποχρεωτικές, αλλά χωρίς χρονικό όριο στην απάντησή τους ερωτήσεις.

Επιπλέον, και προκειμένου να εξασφαλιστεί ακόμη περισσότερο η εγκυρότητα του περιεχομένου του ερωτηματολογίου, ώστε κατ' αντιστοιχία να εξασφαλιστεί ότι ως εργαλείο μέτρησης επιτυγχάνει την εννοιολογική κάλυψη των υπό μέτρηση μεταβλητών, ακολουθήθηκε η πραγματοποίηση πιλοτικής δοκιμής του ερωτηματολογίου, διανέμοντάς το ηλεκτρονικά σε δέκα εθελοντές, με ποίκιλα δημογραφικά χαρακτηριστικά αλλά και επίπεδο εκπαίδευσης, από τους οποίους ζητήθηκε όχι μόνο η έντυπη συμπλήρωσή του αλλά και ο σχολιασμός των ερωτήσεων σε σχέση με την σαφήνιά του, τη λειτουργικότητά του αλλά και τον απαιτούμενο χρόνο για τη συμπλήρωσή του. Και πάλι η ανατροφοδότηση που λήφθηκε είχε ως αποτέλεσμα την αναδιατύπωση ορισμένων ερωτήσεων, ώστε να εξαλειφθεί οποιαδήποτε δυσνόητη ή αμφίσημη έννοια από τους συμμετέχοντες στην πιλοτική δοκιμή.

4.5 Διανομή ερωτηματολογίου στον πληθυσμό

Για την παρούσα έρευνα, το ερωτηματολόγιο σχεδιάστηκε τον Νοέμβριο του 2019. Στη συνέχεια, η διανομή του στους ερωτώμενους πραγματοποιήθηκε το χρονικό διάστημα Νοεμβρίου-Δεκεμβρίου 2019. Το ερωτηματολόγιο μοιράστηκε τόσο σε χώρους παροχής υπηρεσιών όσο και σε δημόσιους χώρους και όχι ηλεκτρονικά ώστε οι ερωτώμενοι να είναι σε θέση να απαντήσουν ανακαλώντας στη μνήμη τους τα συναισθήματα που τους προκλήθηκαν κατά την αλληλεπίδραση τους με τις ηλεκτρονικές διαφημίσεις. Επιπλέον, ήταν σημαντικό να κατανοήσουμε εάν θυμούνται το περιεχόμενο των ηλεκτρονικών διαφημίσεων εφόσον πραγματοποίησαν κάποια ενέργεια που σχετίζεται με αυτές και ιδιαίτερα εάν αγόρασαν κάποιο προϊόν υψηλής τεχνολογίας παρακινούμενοι από τέτοιου τύπου διαφημίσεις.

4.6 Μέγεθος Δειγματοληψίας και επιλογή Δείγματος

Ως μέθοδος δειγματοληψίας (sampling), δηλαδή ο τρόπος με τον οποίο επιλέχθηκαν τα

στοιχεία του πληθυσμού τα οποία θα αποτελούσαν το δείγμα της παρούσας έρευνας, χρησιμοποιήθηκε η «Συστηματική Τυχαία Δειγματοληψία» (Systematic Random Sampling), η οποία σε σχέση με την απλή τυχαία δειγματοληψία (Simple Random Sampling) υιοθετεί μια πιο συγκεκριμένη και με συστηματικό τρόπο διαδικασία επιλογής του δείγματος.

Το ερωτηματολόγιο απάντησαν συνολικά 50 άτομα εκ των οποίων 28 ήταν άνδρες και 22 γυναίκες, συνεπώς το δείγμα μας ήταν κατά κάποιο τρόπο ισόποσα κατανεμημένο ως προς το φύλο των συμμετεχόντων στην έρευνα. Οι ηλικίες των συμμετεχόντων ήταν μεταξύ 18 και 74 ετών και κατανεμήθηκαν σε ομάδες εντός του ερωτηματολογίου. Το εκπαιδευτικό υπόβαθρο των ερωτηθέντων ποικίλλει από τη βασική/υποχρεωτική μόρφωση ως και την ανώτατη.

Το δείγμα μας αν και είναι ισόποσα κατανεμημένο ως προς τη μεταβλητή φύλο, εντούτοις υπάρχουν διαφοροποιήσεις τόσο ως προς τον αριθμό των ατόμων που αποτελούν τις ηλικιακές ομάδες όσο και ως προς το εκπαιδευτικό τους υπόβαθρο. Οι διαφορές αυτές οφείλονται στο γεγονός ότι η έρευνα αποτελεί μέρος διπλωματικής εργασίας με περιορισμό στο χρόνο και στους πόρους που θα απαιτούσε μια έρευνα με μεγαλύτερο δείγμα.

ΚΕΦΑΛΑΙΟ 5: ΑΠΟΤΕΣΜΑΤΑ ΕΡΕΥΝΑΣ

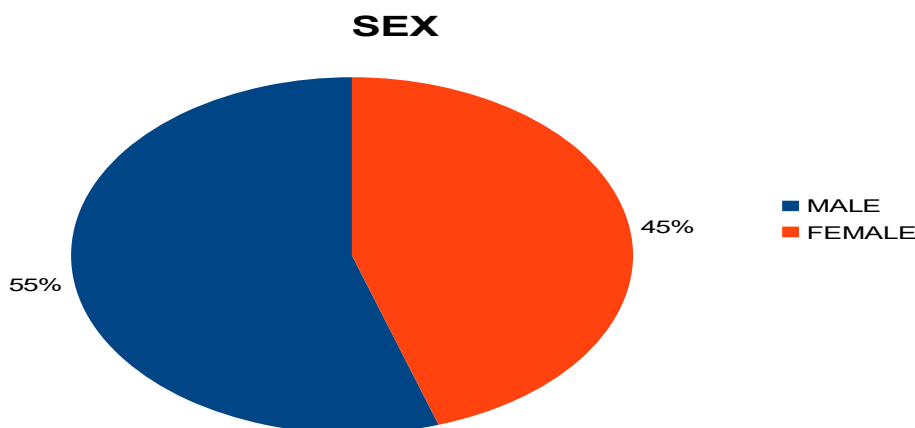
Τα δεδομένα περάστηκαν στο στατιστικό πρόγραμμα SPSS 17 και ακολουθεί η ανάλυση αυτών με τα σχετικά διαγράμματα που βγήκαν απο το ανωτέρω πρόγραμμα.

5.1 Ανάλυση Αποτελεσμάτων

Α) Δημογραφικά Στοιχεία

Πληθυσμός

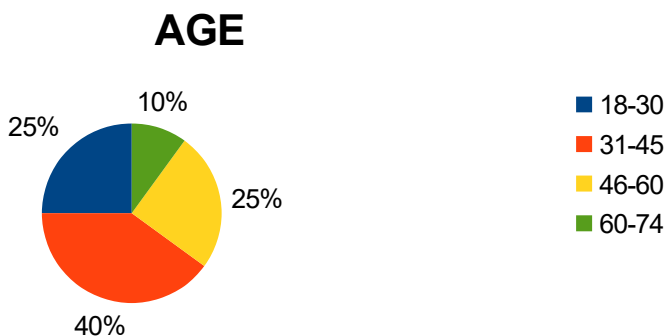
Στο σύνολο απάντησαν 50 άτομα εκ των οποίων 28 ήταν άνδρες και 22 γυναίκες, ήτοι 56% και 44% αντίστοιχα, διαγραμματικά:



Διάγραμμα 1: Φύλο

Ηλικία

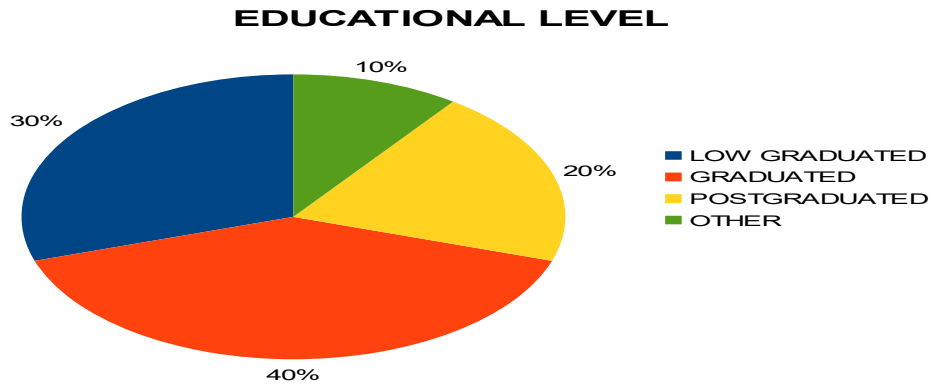
Τα ποσοστά φαίνονται στο παρακάτω διάγραμμα, και παρατηρούμε οτι η μεγαλύτερη μερίδα ανθρώπων που απάντησαν ήταν ηλικίας απο 31 έως 45 ετών



Διάγραμμα 2: Ηλικία

Εκπαιδευτικό Επίπεδο

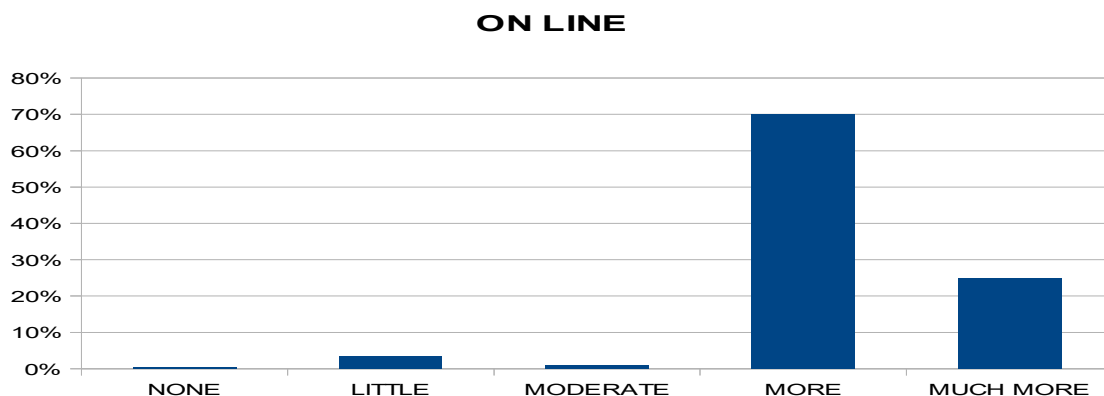
Το εκπαιδευτικό επίπεδο των περισσότερων ατόμων με ποσοστό 40% είναι κάτοχοι ΑΕΙ/ΤΕΙ, ήτοι Τριτοβάθμιας εκπαίδευσης, που σημαίνει ότι έχουν κάποια σχετική γνώση για τις νέες τεχνολογίες



Διάγραμμα 3: Εκπαιδευτικό Επίπεδο

B) Προϊόντα Υψηλής Τεχνολογίας

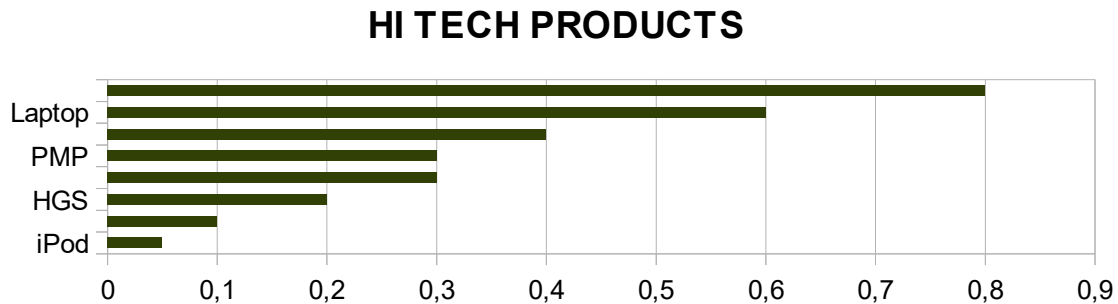
Το Β μέρος του ερωτηματολογίου έχει δυο ερωτήσεις σχετικές με τα προϊόντα υψηλής τεχνολογίας, καθώς η παρούσα ερευνητική διπλωματική εργασία αφορά τα συγκεκριμένα προϊόντα, τα οποία αναλύθηκαν και στο Κεφάλαιο 2. Η πρώτη ερώτηση είναι γενική και αφορά, εάν οι συγκεκριμένοι καταναλωτές κάνουν on line αγορές. Οι απαντήσεις φαίνονται στο παρακάτω διάγραμμα, από το οποίο παρατηρούμε ότι περίπου το 85% του δείγματος πραγματοποιεί διαδικτυακές αγορές



Διάγραμμα 4: On line Αγορές

Προϊόντα Υψηλής Τεχνολογίας

Ενώ απο τα πιο σημαντικά προϊόντα υψηλής τεχνολογίας που αγοράζουν διαδικτυακά το συγκεκριμένο δείγμα καταναλωτών είναι πρώτα τα Smartphones, μετά τα Laptop και μετά τα Tablets και ακολουθούν τα υπόλοιπα όπως φαίνονται και στο παρακάτω διάγραμμα.



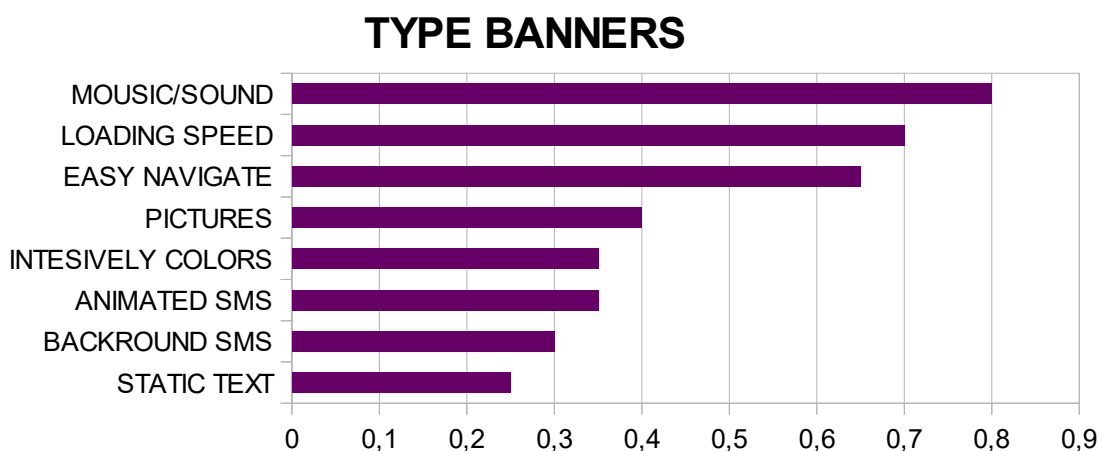
Διάγραμμα 5: Προϊόντα Υψηλής Τεχνολογίας

Γ) Στοιχεία Διαφήμισης Τύπου Banner

Σ' αυτό το μέρος του ερωτηματολογίου, διατυπώνονται τα πέντε (5) ερευνητικά ερωτήματα της παρούσας διπλωματικής, που αφορούν τα τεχνικά χαρακτηριστικά της διαφήμισης τύπου Banner

Υ1: Η Μορφή Banner

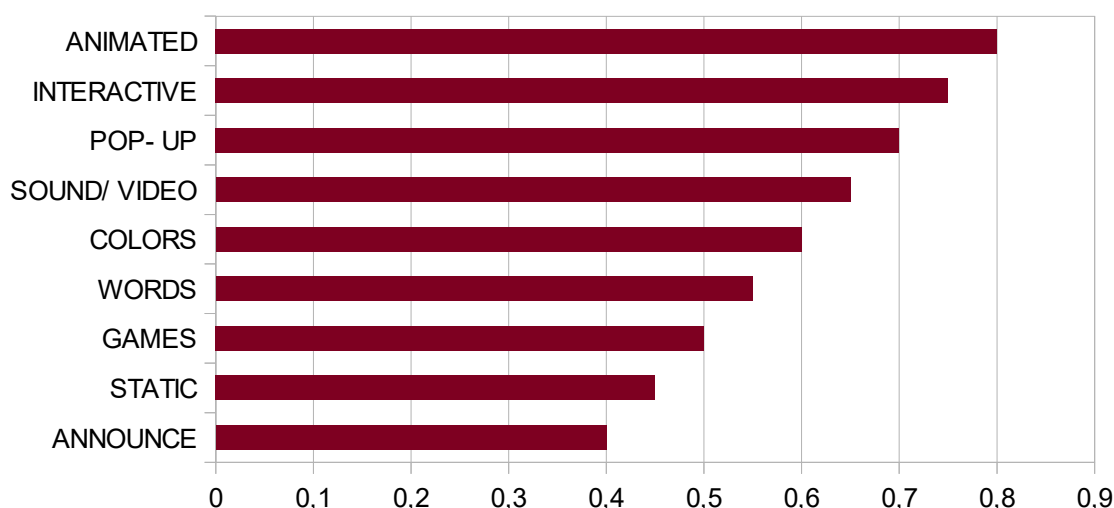
Απο την ερώτηση 6, διαπιστώνεται οτι οι καταναλωτές του δείγματος προτιμούν καλύτερα τα διαφημιστικά banner που είναι σχεδιασμένα με μουσική και ήχο, ύστερα τα banner που μπορούν να φορτώνονται με γρήγορη ταχύτητα, ώστε να μην είναι κουραστικά, μετά αυτά που η πλοήγησή τους είναι εύκολη, ενώ ακολουθούν οι εικόνες, τα έντονα χρώματα και το κινούμενο κείμενο.



Διάγραμμα 6: Μορφή Banners

Στην ερώτηση 7, που είναι πιο συγκεκριμένη για την μορφή/ τύπο του διαφημιστικού banner, το καταναλωτικό κοινό απάντησε ότι προτιμότερα τελικά είναι τα animated banner, μετά τα interactive, τα pop-up και τελευταία τα static και τα banner της μορφής ανακοινώσεων.

TYPE BANNERS

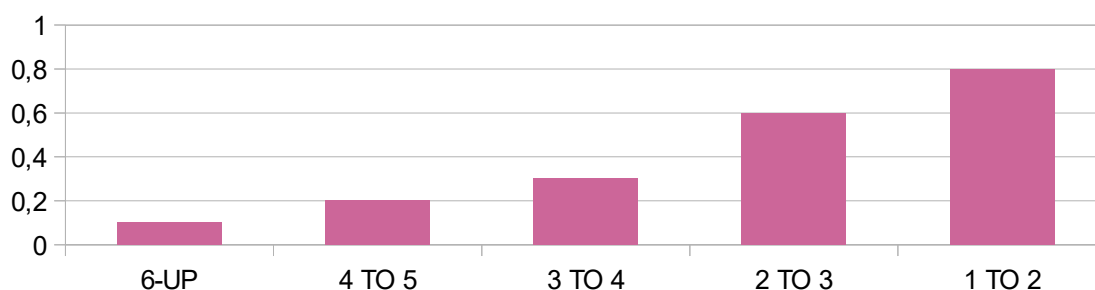


Διάγραμμα 7: Μορφή Banners

Υ2: Αριθμός Προβολής Banner

Ο αριθμός των διαφημιστικών banner που θα πρέπει να προβάλλεται σε μια ιστοσελίδα πλοήγησης, ώστε να μην είναι κουραστικά και αποτελεσματικά είναι με βάση τις απαντήσεις του συγκεκριμένου δείγματος από 1 έως 2. Διαγραμματικά φαίνεται και κάτωθι:

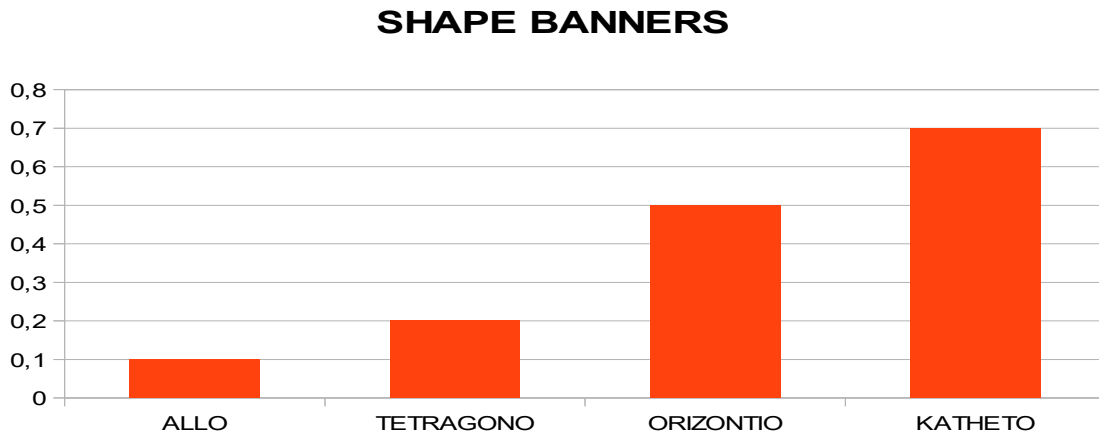
NUMBER BANNERS



Διάγραμμα 8: Αριθμός Banners

Υ3: Σχήμα Banners

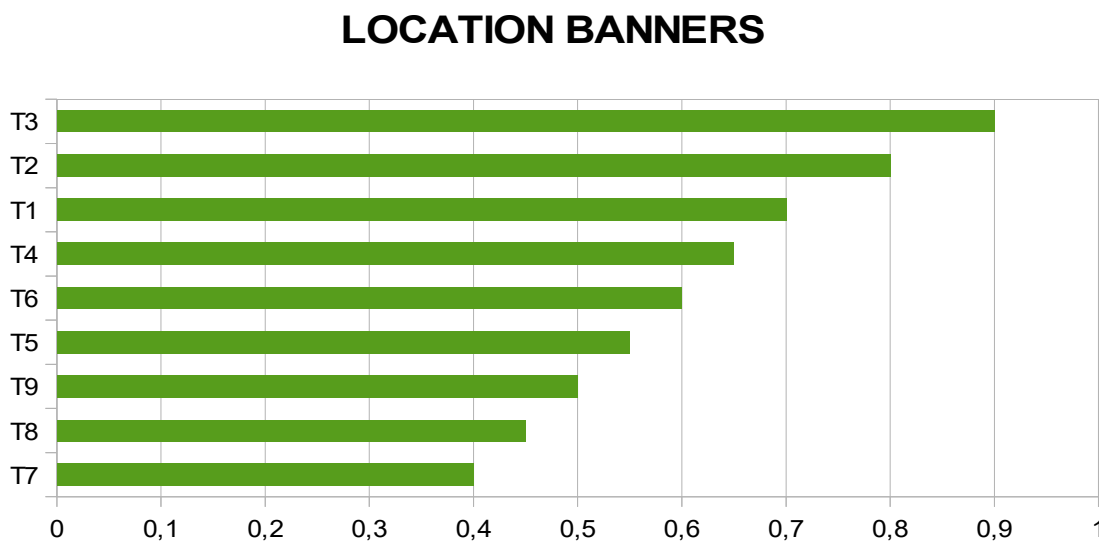
Απο τις απαντήσεις του δείγματος, αποδεικνύεται οτι τι δημοφιλεστερο σχήμα banner είναι το κάθετο και μετά το οριζόντιο. Συγκεκριμένα:



Διάγραμμα 9: Σχήμα Banners

Υ4: Τοποθεσία Banners

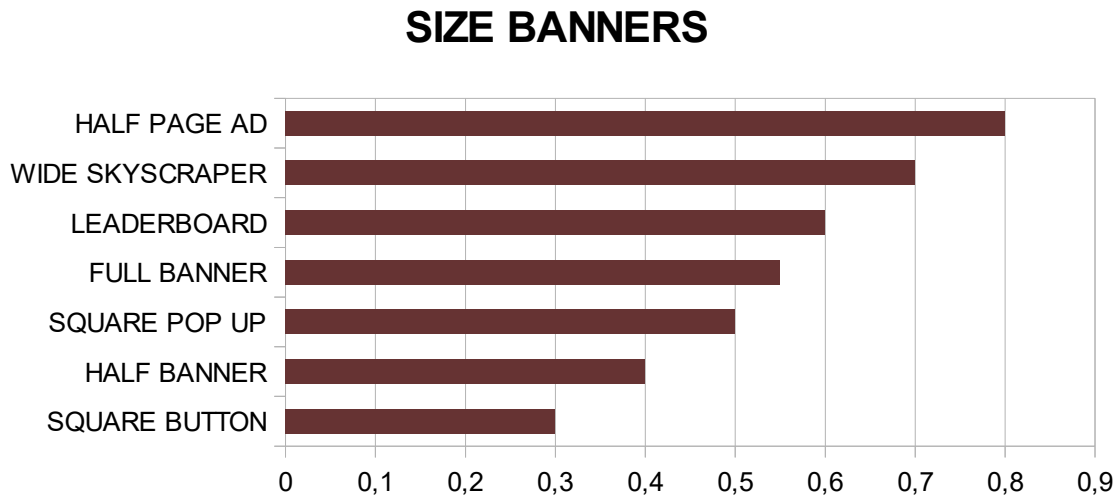
Η Τοποθεσία στην ιστοσελίδα κάποιου διαδικτυακού τόπου που βρίσκεται ένα banner έχει μεγάλη σημασία, προκειμένου να προσελκύσει το ενδιαφέρον του καταναλωτή. Οι απαντήσεις που δόθηκαν δείχνουν οτι η Τοποθεσία 1, πάνω δεξιά είναι η δημοφιλέστερη και ύστερα η Τοποθεσία 2, πάνω κεντρικά, ενώ η λιγότερη δημοφιλής τοποθεσία είναι η Τοποθεσία 7, κάτω αριστερά. Με άλλα λόγια, παρατηρούμε οτι οι τοποθεσίες που κάνουν ένα banner ελκυστικό είναι απο το πάνω μέρος και αριστερά της ιστοσελίδας και καταλήγει στο κάτω μέρος και δεξιά.



Διάγραμμα 10: Τοποθεσία Προβολής Banners

Y5: Μέγεθος Banners

Το μέγεθος, οι διαστάσεις αλλιώς, των διαφημιστικών banners, παίζουν καθοριστικό ρόλο στην προσέλκυση του καταναλωτικού κοινού. Η συγκεκριμένη έρευνα απέδειξε ότι το Half Page Ad, είναι το δημοφιλέστερο και τελευταίο το Square Button.



Διάγραμμα 11: Μέγεθος Banners

5.2 Συσχέτιση Ερευνητικών Υποθέσεων

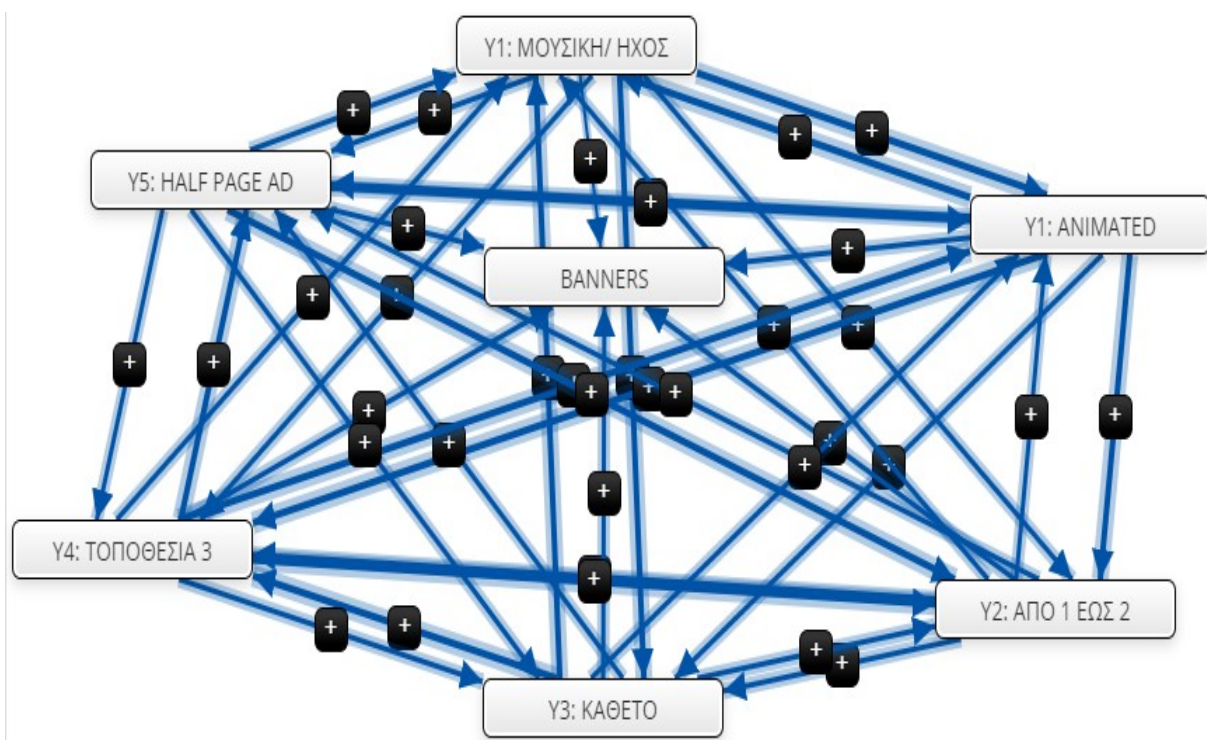
Απο τις απαντήσεις του συγκεκριμένου δείγματος και τα αποτελέσματα που προέκυψαν από το πρόγραμμα του SPSS, καταλήγουμε στο συμπέρασμα ότι οι και οι πέντε (5) ερευνητικές υποθέσεις σχετίζονται θετικά μεταξύ τους. Συγκεκριμένα, η Y1 (Μορφή) έχει θετική συσχέτιση με την Y2 (Αριθμός Προβολής), Y3 (Σχήμα), Y4 (Τοποθεσία) και Y5 (Μέγεθος) ίσο με 0,80 περίπου, πολύ κοντά στο 1. Ομοίως και η Y2, Y3, Y4 και Y5 μεταξύ τους. Αυτό αποτυπώνεται και στο παρακάτω πίνακα:

Correlations							
		ΜΟΥΣΙΚΗ_ΗΧΟΣ	ANIMATED	ΑΠΟ_1_2	ΚΑΘΕΤΟ	T3	HALF_PAGE_AD
ΜΟΥΣΙΚΗ_ΗΧΟΣ	Pearson Correlation	1	,857**	,857**	,863**	,812**	,880**
	Sig. (2-tailed)		,000	,000	,000	,000	,000
	N	50	50	50	50	50	50
ANIMATED	Pearson Correlation	,857**	1	1,000**	,864**	,903**	,831**
	Sig. (2-tailed)	,000		,000	,000	,000	,000
	N	50	50	50	50	50	50
ΑΠΟ_1_2	Pearson Correlation	,857**	1,000**	1	,864**	,903**	,831**
	Sig. (2-tailed)	,000	,000		,000	,000	,000
	N	50	50	50	50	50	50
ΚΑΘΕΤΟ	Pearson Correlation	,863**	,864**	,864**	1	,860**	,907**
	Sig. (2-tailed)	,000	,000	,000		,000	,000
	N	50	50	50	50	50	50
T3	Pearson Correlation	,812**	,903**	,903**	,860**	1	,842**
	Sig. (2-tailed)	,000	,000	,000	,000		,000
	N	50	50	50	50	50	50
HALF_PAGE_AD	Pearson Correlation	,880**	,831**	,831**	,907**	,842**	1
	Sig. (2-tailed)	,000	,000	,000	,000	,000	
	N	50	50	50	50	50	50

** . Correlation is significant at the 0.01 level (2-tailed).

Εικόνα 5.1: Συσχετίσεις Ερευνητικών Υποθέσεων

Όλα τα ανωτέρω μπορούν να αποτυπωθούν και στο πρόγραμμα του FZZZY COGNITIVE MAP του MENTAL MODELER, όπου φαίνονται οι θετικές συσχετίσεις των ερευνητικών υποθέσεων μεταξύ τους, οι ενώσεις τους και ότι υπάρχει αμφίδρομη θετική σχέση.



Εικόνα 5.2: FZZZY COGNITIVE MAP Ερευνητικών Υποθέσεων

5.3 Συμπεράσματα της Έρευνας

Τα Συμπεράσματα απο την παρούσα διπλωματική έρευνα είναι τα εξής:

- Οι περισσότεροι καταναλωτές του δείγματος πραγματοποιούν on line αγορές.
- Τα δημοφιλέστερα προϊόντα υψηλής τεχνολογίας που αγοράζουν απο το internet οι καταναλωτές του συγκεκριμένου δείγματος είναι τα Smartphones.
- Η μορφή banners που επιλέγουν οι καταναλωτές είναι αυτά που έχουν Μουσική/Ήχο και τα Animated, με άλλα λόγια επικρατεί το animation σε σχέση με το κείμενο.
- Ο αριθμός των banners που θα πρέπει να υπάρχει σε μια ιστοσελίδα, ώστε να μην είναι ενοχλητική, κυμαίνεται απο 1 έως 2.
- Το σχήμα των Banners, σύμφωνα με τα αποτελέσματα, που κάνει μια διαφήμιση αποτελεσματικότερη είναι το κάθετο.
- Η τοποθεσία των banners, που θεωρείται οτι πρέπει να τοποθετούνται ώστε να είναι περισσότερο ελκυστικά είναι στο πάνω μέρος και δεξιά της ιστοσελίδας
- Το μέγεθος banner και οι διαστάσεις του είναι αυτό του Half Page Ad 300*600, δηλαδή προτιμούνται οι μεγάλες σε μέγεθος διαφημίσεις.

Παρατηρούμε οτι οι ερευνητικές μας υποθέσεις επιβεβαιώνονται και απο την υπάρχουσα βιβλιογραφία, όπως αυτή αναφέρθηκε και στο Κεφάλαιο 3. Συγκεκριμένα:

Τύπος

Y1: Σε ένα banner θα πρέπει να κυριαρχούν τα πολυμέσα (animations) σε σχέση με το κείμενο.

Επιβεβαιώνεται, καθόσον, η κινούμενη διαφήμιση banner ή διαφήμιση που χρησιμοποιεί πολυμέσα -animations-είναι ένας από τους πιο αγαπημένους και αποτελεσματικότερους τύπους διαφήμισης banner μεταξύ των online καταναλωτών (Li & Bukovac, 1999, Dreze & Hussherr, 2003, Diao & Sundar, 2004, Bruce et al., 2017, Chandon et al., 2003, Lohtia et al., 2003, Rosenkrans 2009, Yoo et al. 2004). Η δυναμική αυτή του τύπου διαφήμιση banner αποτελείται από ήχο, βίντεο, εικόνες, φωτεινά και λαμπερά χρώματα, και μοιάζει περισσότερο σαν τις τηλεοπτικές διαφημίσεις (Koegel, 2003). Έχει σχεδιαστεί ώστε να επηρεάσει τους καταναλωτές με τον ίδιο τρόπο όπως κάνει και το τηλεοπτικό κανάλι με μια πρόσθετη διάσταση της διαδραστικότητας.

Αριθμός Προβολής Διαφήμισης Banner

Y2: Η προβολή ενός banner σε ένα ιστότοπο και όχι περισσότερων, ώστε να αποφεύγεται η σύγχυση με τα υπόλοιπα διαφημιστικά banners

Επιβεβαιώνεται, καθόσον ο υπερβολικός αριθμός διαφημίσεων που μπορεί να εμφανίζονται

σε μια ιστοσελίδα επιλογής του χρήστη, θεωρείται ως μια σημαντική μεταβλητή που επηρεάζει την στάση των καταναλωτών απέναντι στη διαφήμιση (Ray & Webb, 1986, Kent, 1993, Ha, 1996, Ha & Litmann, 1997, Elliott & Speck, 1998).

Σχήμα

Υ3: Τα κάθετα banners είναι πιο ελκυστικά από τα υπόλοιπα σχήματα banners

Επιβεβαιώνεται, καθόσον το κάθετο banner είναι πιο αποτελεσματικό από το οριζόντιο banner, το οποίο τοποθετείται από αριστερά προς τα δεξιά (Dreze & Hussherr, 2003).

Τοποθεσία

Υ4: Το banner θα πρέπει να τοποθετηθεί στην κορυφή μιας ιστοσελίδας και όχι στο τέλος της

Επιβεβαιώνεται, καθόσον, οι διαφημίσεις banner που τοποθετούνται στην κορυφή μιας ιστοσελίδας έχουν μεγαλύτερη πιθανότητα να τραβήξουν την προσοχή του χρήστη απο όταν είναι τοποθετημένες στο πάνω μέρος ή στη μέση μιας ιστοσελίδα (Abhishek et al., 2015, Agarwal et al., 2011, Agarwal and Mukhopadhyay 2016, Animesh κ.ά., 2011, Chan and Young-Hoon 2015, Gong κ.ά., 2018, Narayanan and Kalyanam 2015, Rutz κ.ά., 2012).

Μέγεθος

Υ5: Η μεγάλη σε μέγεθος διαφήμιση banner είναι αποδοτικότερη από την μικρού μεγέθους διαφήμιση banner

Επιβεβαιώνεται, καθόσον, στην πράξη, μια μεγαλύτερη διαφήμιση είναι πιο αποτελεσματική από μια μικρότερη (Bruce et al., 2017, Chandon et al., 2003, Rosenkrans 2009, Sokolik et al., 2014).

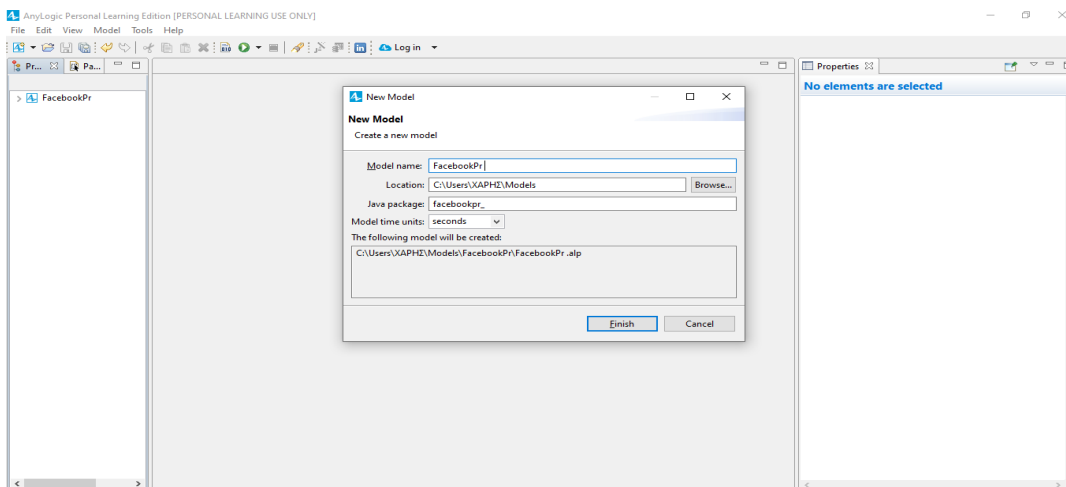
ΚΕΦΑΛΑΙΟ 6: ΜΟΝΤΕΛΟΠΟΙΗΣΗ – ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΕΡΕΥΝΗΤΙΚΩΝ ΥΠΟΘΕΣΕΩΝ

6.1 Γενικά

Η δημιουργία του μοντέλου είναι πολύ εύκολη και η διαχείριση του προγράμματος AnyLogic εφικτή σε όποιον έχει ευχέρεια με την χρήση προγραμμάτων αλλά ακόμα και αν δεν ισχύει κάτι τέτοιο είναι εύκολο να το μάθει κάποιος και να ξετυλίξει την πληθώρα των δυνατοτήτων του. Σημαντικό είναι πριν ξεκινήσει η δημιουργία του μοντέλου να υπάρχει ένα πρόχειρο προσχέδιο σε χαρτί όπου να αποφαινίζεται ο βαθμός επηρεασμού και η σχέση που υπάρχει ανάμεσα στους παράγοντες και το μοντέλο που δημιουργείται. Εμείς θα δημιουργήσουμε δυναμικά μοντέλα για τις παραπάνω ερευνητικές υποθέσεις με τα αποτελέσματα που βγήκαν απο την έρευνά μας.

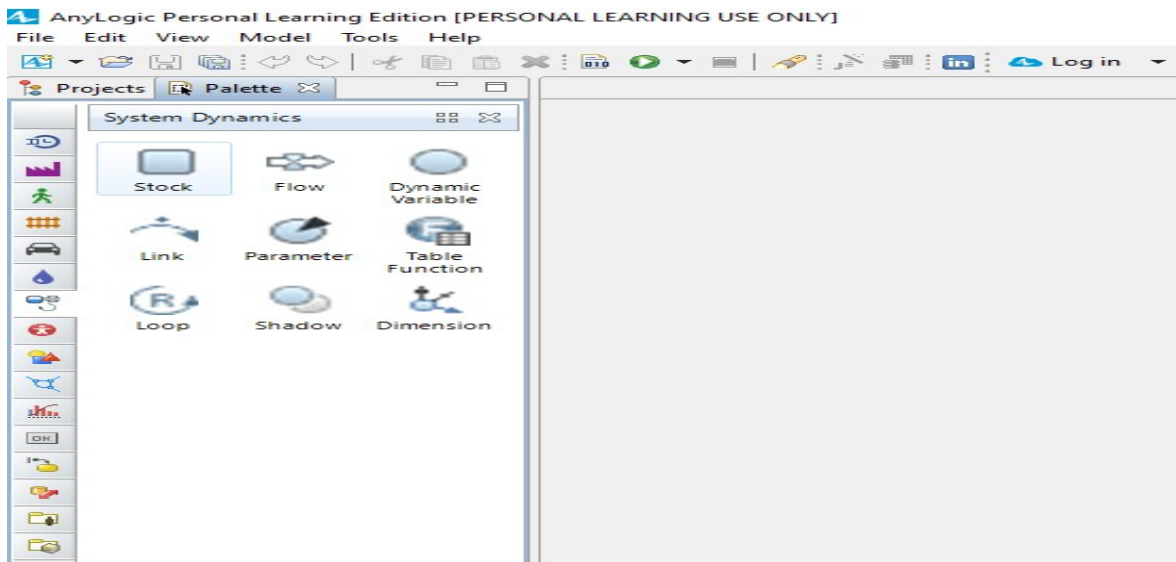
Εφόσον εγκαταστήσω το πρόγραμμα AnyLogic που θα μας εξυπηρετήσει στην διαδικασία της μοντελοποίησης δημιουργώ το Μοντέλο μου στο περιβάλλον - interface του AnyLogic φτιάχνοντας τους παράγοντες ρυθμίζοντας τα στοιχεία που έχω συλλέξει ώστε να λειτουργήσει σωστά η μοντελοποίηση από την πρότερη έρευνα.

Βήμα 1: File/New/Model και ονομάζω το Μοντέλο, στο Model time units δίνω την επιλογή days έπειτα επιλέγω Finish.



Εικόνα 6.1: Δημιουργία Μοντέλου Υ1:Τύπος/Μορφή Banner.

Βήμα 2: Palette/System Dynamic



Εικόνα 6.2: Δημιουργία κεντρικού σχήματος.

Με χρήση των εικονιδίων που δίνονται στην επιλογή που έχουμε επιλέξει System Dynamics θα δημιουργήσουμε το διάγραμμα που περιέχει τα ερευνητικά μας πεδία και τις συσχετίσεις τους. Το button stock - απόθεμα αντιπροσωπεύει την συγκέντρωση ενός φυσικού ή μη φυσικού μεγέθους (συγκέντρωση πόρων) είναι αυτό που θα χρησιμοποιηθεί για τα βασικά σημεία του μοντέλου, τα ερευνητικά ερωτήματα και την κατανομή των πόρων. Το button flow είναι μια ροή που αντιπροσωπεύει μια δραστηριότητα, η οποία γεμίζει ή εξαντλεί ένα απόθεμα. Το βέλος υποδεικνύει την κατεύθυνση της θετικής ροής μέσα ή έξω από ένα απόθεμα που εφόσον τοποθετηθεί σωστά θα απεικονίσει την ροή των πόρων, ποιοι απορροφώνται και ποιοι χάνονται.

Οι πόροι που απορροφώνται είναι αυτοί που έχουν απόδοση και αυτοί που χάνονται είναι εκείνοι που δαπανήθηκαν αλλά δεν απέδωσαν για τον τύπο/μορφή του διαφημιστικού banner, αυτοί οι πόροι δηλαδή που δεν βρίσκουν ανταπόκριση στους δέκτες της προωθητικής ενέργεια που χρησιμοποιήθηκαν.

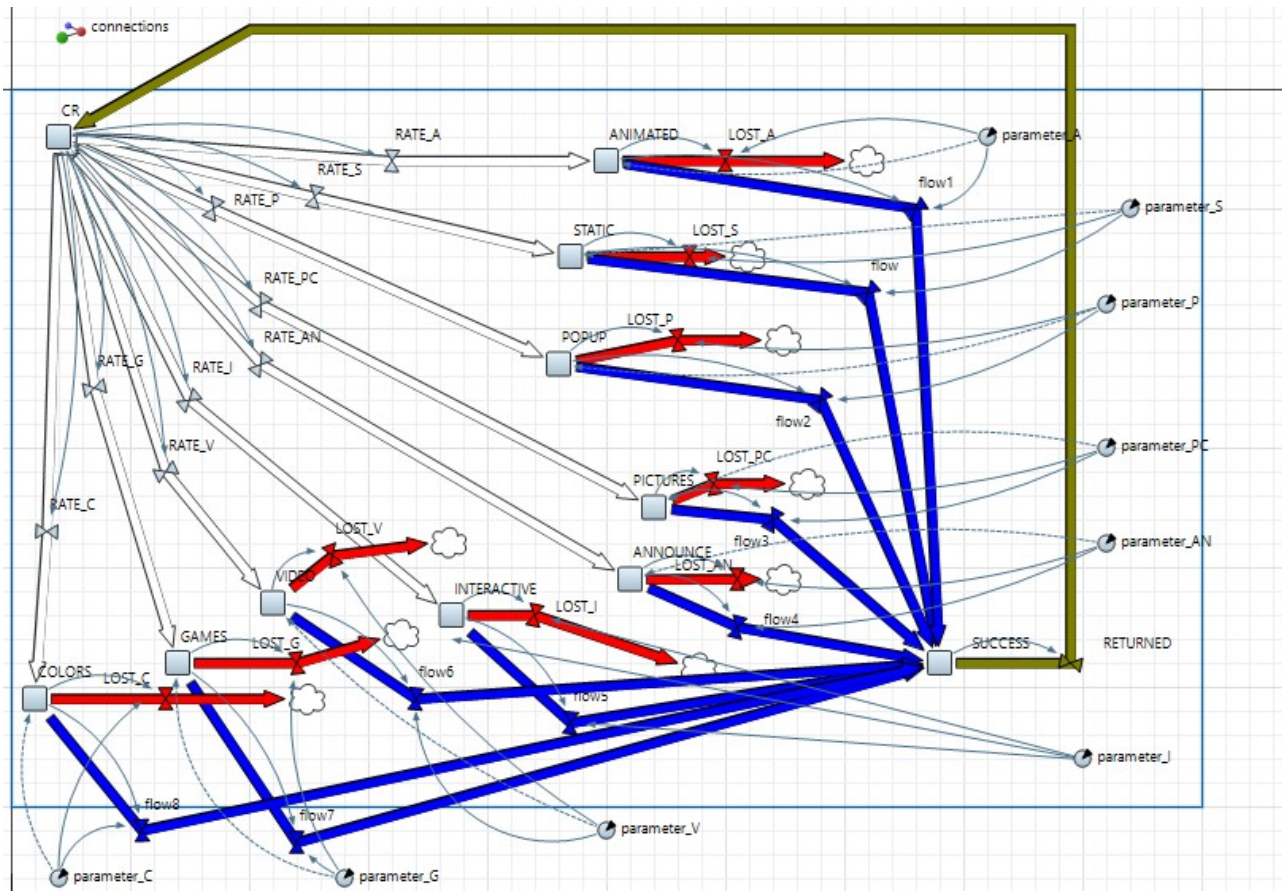
Το button parameter το χρησιμοποιούμε για να δηλώσουμε την σχέση της παραμέτρου με το σχηματισμό του μοντέλου. Το button link ή αλλιώς υποδοχές (connector) παρέχουν τις συνδέσεις μεταξύ των στοιχείων του μοντέλου. Υπάρχουν δύο ειδών connectors: α) οι συμπαγής σύνδεση, όπου αντιπροσωπεύει ένα σύνδεσμο δράσης και β) διακεκομμένη σύνδεση όπου αντιπροσωπεύει μια σύνδεση πληροφορίας. χρησιμοποιείται για να συσχετιστούν οι παράγοντες μεταξύ τους. Τέλος το button loop δίνει την δυνατότητα βρόχου του μοντέλου.

6.2 Μοντελοποίηση ερευνητικής υπόθεσης Y1

Χρησιμοποιήσαμε και τους εννέα (9) παράγοντες που επηρεάζουν την μορφή του banner και που απαντήθηκαν και στην έρευνα μας με το δομημένο ερωτηματολόγιο. Για απλοποίηση του

μοντέλου οι βασικοί παράγοντες ήταν: static, animated, interactive, pop-up, pictures, games, video, announce & colours. Τα αρχικά ποσοστά που βάλουμε στην αρχική απεικόνιση δυναμικής μοντελοποίησης, ήταν περίπου 11,1%

Το διάγραμμα που προκύπτει είναι το παρακάτω:



Εικόνα 6.3: Διάγραμμα συσχετίσεων παραγόντων της Y1: Μορφή Προβολής Banner

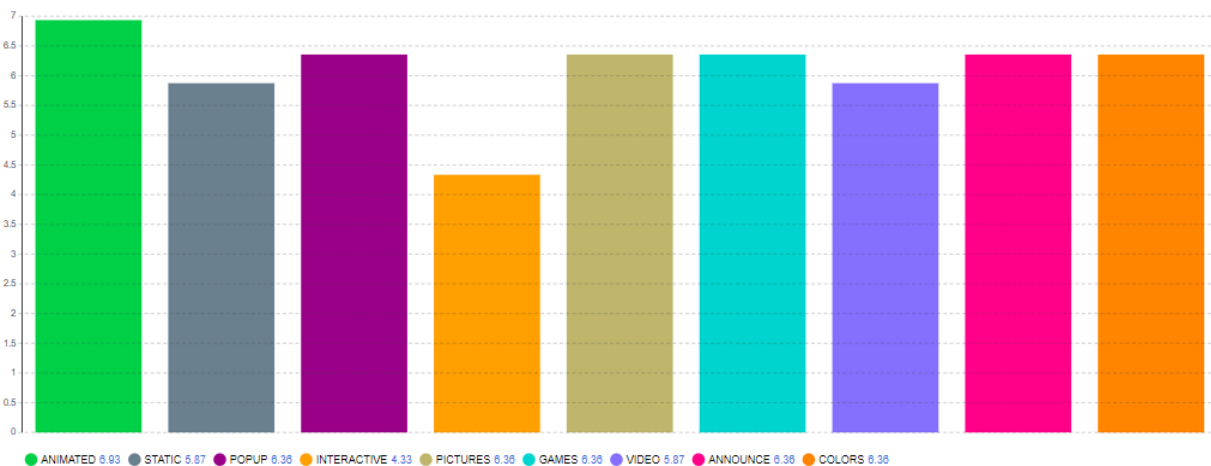
Θα συνεχίσουμε να αναπτύσσουμε το μοντέλο μας καθορίζοντας τα χαρακτηριστικά και τις συμπεριφορές δυναμικών πελατών μας. Ο καλύτερος τρόπος για να ορίσετε μια συμπεριφορά είναι να χρησιμοποιήσετε ένα διάγραμμα (chart). Τα charts είναι το πιο προηγμένο μοντέλο για την περιγραφή συμπεριφοράς που βασίζεται 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. Τα charts μπορεί να είναι ιεραρχικά - μπορεί να περιέχουν ακόμα άλλες καταστάσεις και μεταβάσεις (μπάρες, πίτες κλπ).

Μπορούμε να παρατηρήσουμε πώς οι μεταβλητές αλλάζουν κατά τη διάρκεια της προσομοίωσης και να εξετάσουμε τη δυναμικότητά τους. Για παράδειγμα έστω ότι θέλουμε να

δημιουργήσουμε ένα Time Plot από τη σελίδα Analysis page του πεδίου Palette. Με την επιλογή του μπορούμε να το προσθέσουμε στη στην επιφάνεια του AnyLogic.

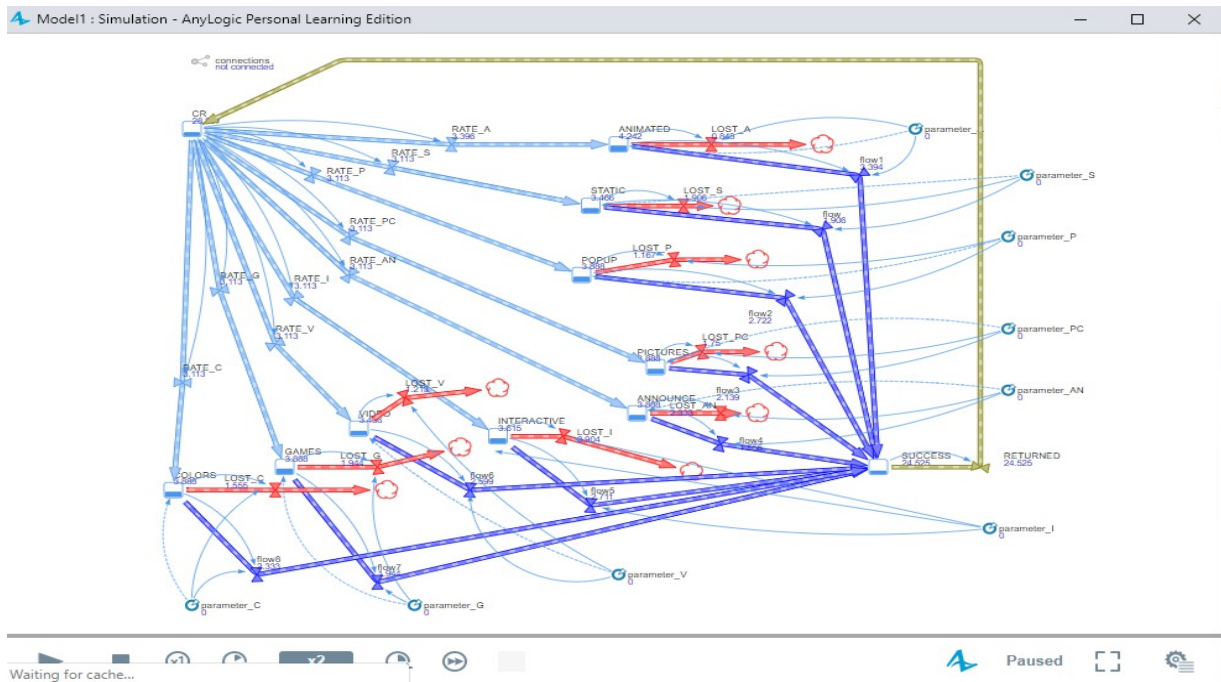
1. Στη συνέχεια, πηγαίνουμε στη σελίδα General, της καρτέλας Properties του Time Plot.
2. Στο πεδίο Time Window μπορούμε να επιλέξουμε το χρονικό εύρος που θα ελέγξουμε.
3. Ορίζουμε τα data sets, τις μεταβλητές που θα εμφανίζονται στο Chart πατώντας το κουμπί Add Data Set και ορίζουμε τις παραμέτρους από το πεδίο property.
4. Επιλέγουμε Update automatically.

Στην συνέχεια λαμβάνουμε το διάγραμμα όπως φαίνεται παρακάτω:



Εικόνα 6.4: Διάγραμμα Bar Chart

Μετά από την δημιουργία των διαγραμμάτων τρέχει το μοντέλο με την επιλογή Run στην άνω μπάρα επιλογών και διαπιστώνεται η ροή του Μοντέλου πως διαμοιράζει τους πόρους και πως αλληλεπιδρούν τα ερευνητικά πεδία.

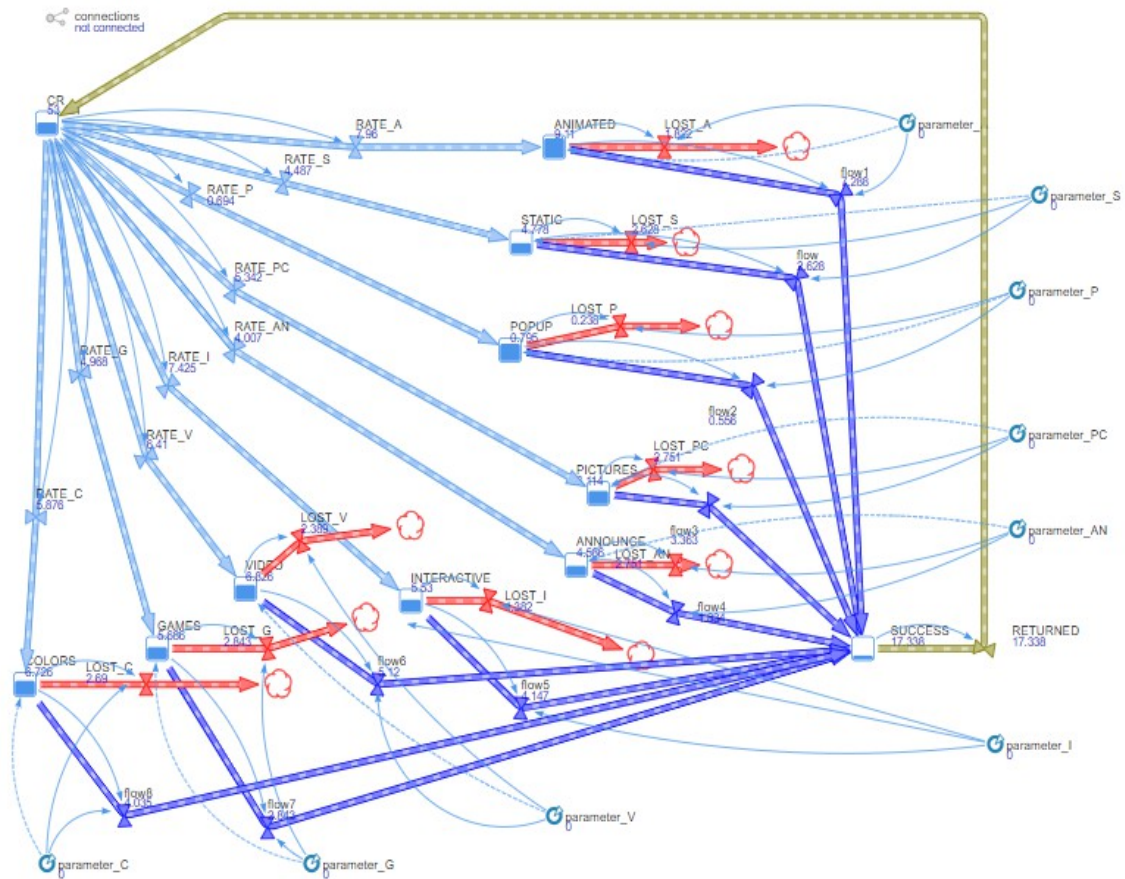


Εικόνα 6.5: Διάγραμμα ροής με την διανομή των πόρων.

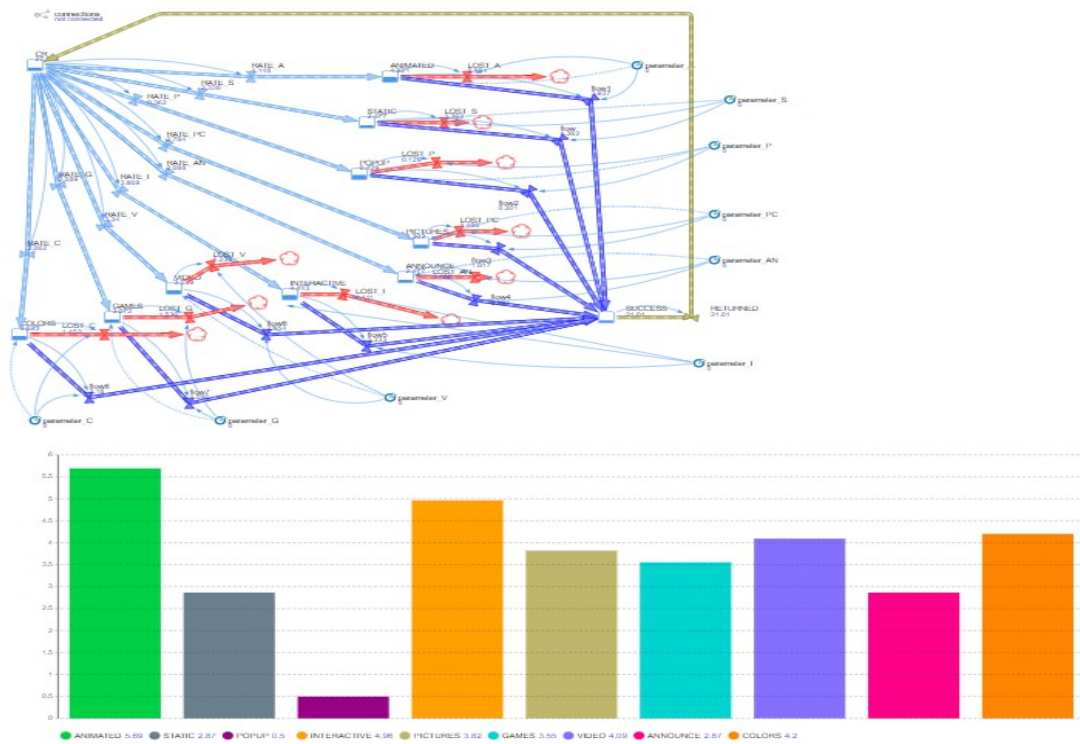
Όπως φαίνεται στο παραπάνω σχήμα και σε όλη την πορεία της μοντελοποίησης η ροή πόρων ξεκινάει από το CompanyResources και με ένα flow κατευθύνεται προς τους βασικούς παράγοντες: static, animated, interactive, pop-up, pictures, games, video, announce & colours. Από τον άνω διαμορισμό προκύπτουν αποθέματα που χάνονται και άλλα που αποδίδουν την επένδυση τους, αυτά που αποδίδουν συγκεντρώνονται στο resources και επαναδρομολογούνται προς το CompanyResources για να επαναυπολογιστούν και να δρομολογηθούν ξανά στους παράγοντες του ερευνητικού μας ερωτήματος. Το total_resources ελέγχει και επανατοποθετεί πόρους στο σύστημα. Οι παράμετροι που συναντώνται ελέγχουν την ροή των πόρων.

6.3 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y1

Για να βελτιστοποιήσουμε το παραπάνω μοντέλο της ερευνητικής υπόθεσης Y1, πήραμε τα ποσοστά που βγήκαν από την πρωτογενή έρευνά μας και από το πρόγραμμα του SPSS, δίνοντας το μεγαλύτερο ποσοστό στα animated banners, καθώς αυτά ήταν και τα πιο δημοφιλή σύμφωνα με τις απαντήσεις των ερωτώμενων μας.



Εικόνα 6.6: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y1



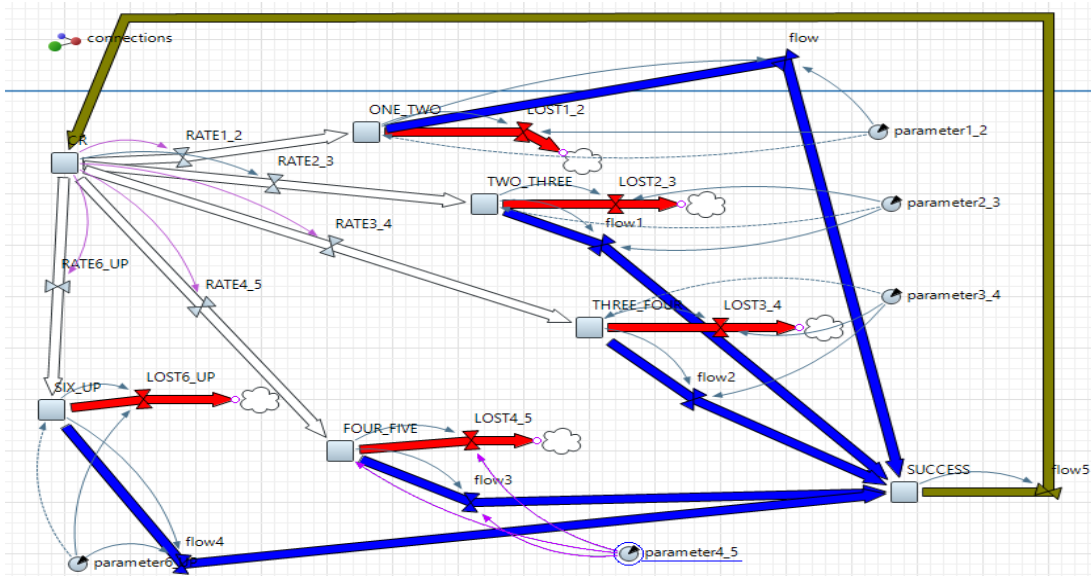
Εικόνα 6.7: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y1 με barcharts

Απο το οποίο προκύπτει και στην βελτιστοποίηση τα animated banners, είναι τα καλύτερα απο τα υπόλοιπα.

6.4 Μοντελοποίηση ερευνητικής υπόθεσης Y2

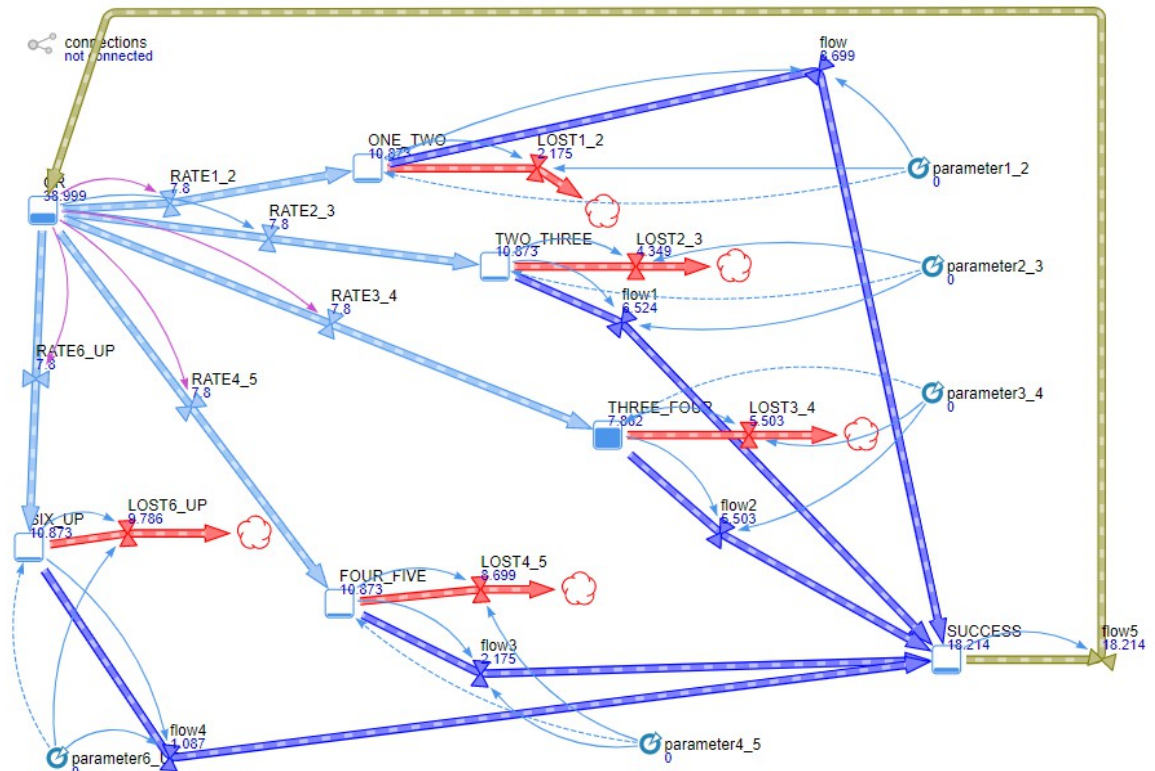
Με τον ίδιο τρόπο όπως στην ερευνητική υπόθεση Y1, ακολουθούμε και για την ερευνητική υπόθεση Y2: Αριθμός Προβολής Banners σε μια ιστοσελίδα, οι παράγοντες είναι πέντε (5) και συγκεκριμένα απο ONE_TWO, TWO_THREE, THREE_FOUR, FOUR_FIVE & SIX_UP. Τα αρχικά ποσοστά που βάλαμε στην αρχική απεικόνιση δυναμικής μοντελοποίησης, ήταν 20%

Το διάγραμμα που προκύπτει είναι το παρακάτω:



Εικόνα 6.8: Διάγραμμα συσχετίσεων παραγόντων της Y2: Αριθμός Προβολής Banner

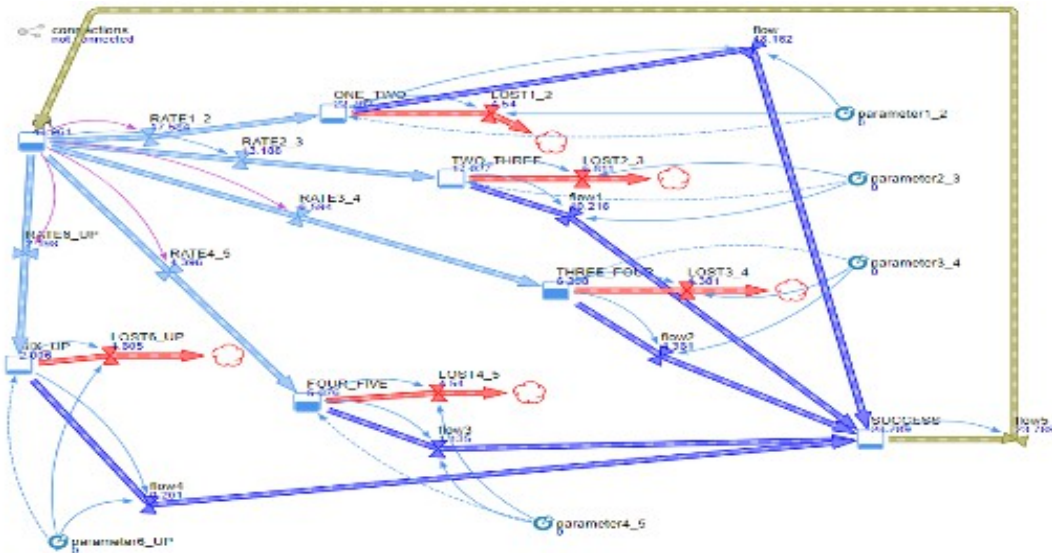
Μετά από την δημιουργία του ανωτέρω διαγράμματος και τις συσχετίσεις μεταξύ των τρέχει το μοντέλο με την επιλογή Run στην άνω μπάρα επιλογών και διαπιστώνεται η ροή του Μοντέλου πως διαμοιράζει τους πόρους και πως αλληλεπιδρούν τα ερευνητικά πεδία.



Εικόνα 6.9: Διάγραμμα ροής με την διανομή των πόρων Y2

6.5 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y2

Για να βελτιστοποιήσουμε το παραπάνω μοντέλο της ερευνητικής υπόθεσης Y2, πήραμε τα ποσοστά που βγήκαν από την πρωτογενή έρευνά μας και από το πρόγραμμα του SPSS, δίνοντας το μεγαλύτερο ποσοστό στον ONE_TWO (ήτοι από 1-2) banners, καθώς αυτό ήταν και τα πιο δημοφιλή σύμφωνα με τις απαντήσεις των ερωτώμενων μας.



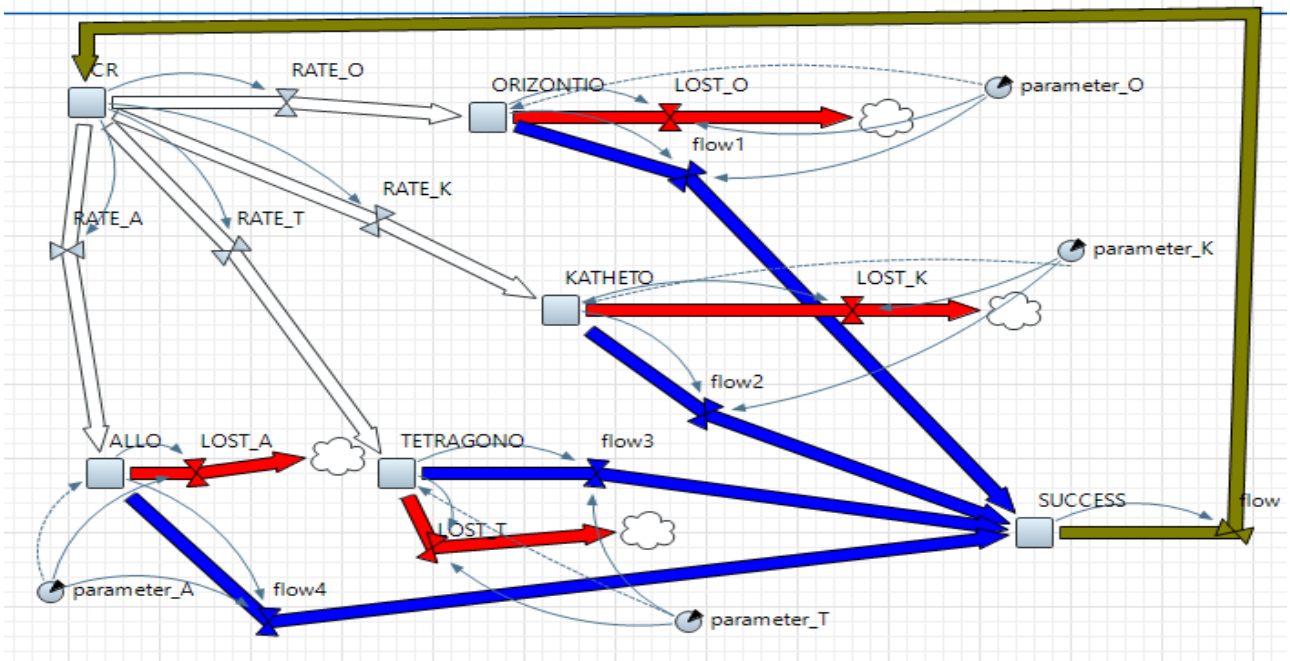
Εικόνα 6.10: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y2

Απο το οποίο προκύπτει και στην βελτιστοποίηση τα ONE_TWO (ήτοι απο 1-2) banners, είναι τα καλύτερα απο τα υπόλοιπα.

6.6 Μοντελοποίηση ερευνητικής υπόθεσης Y3

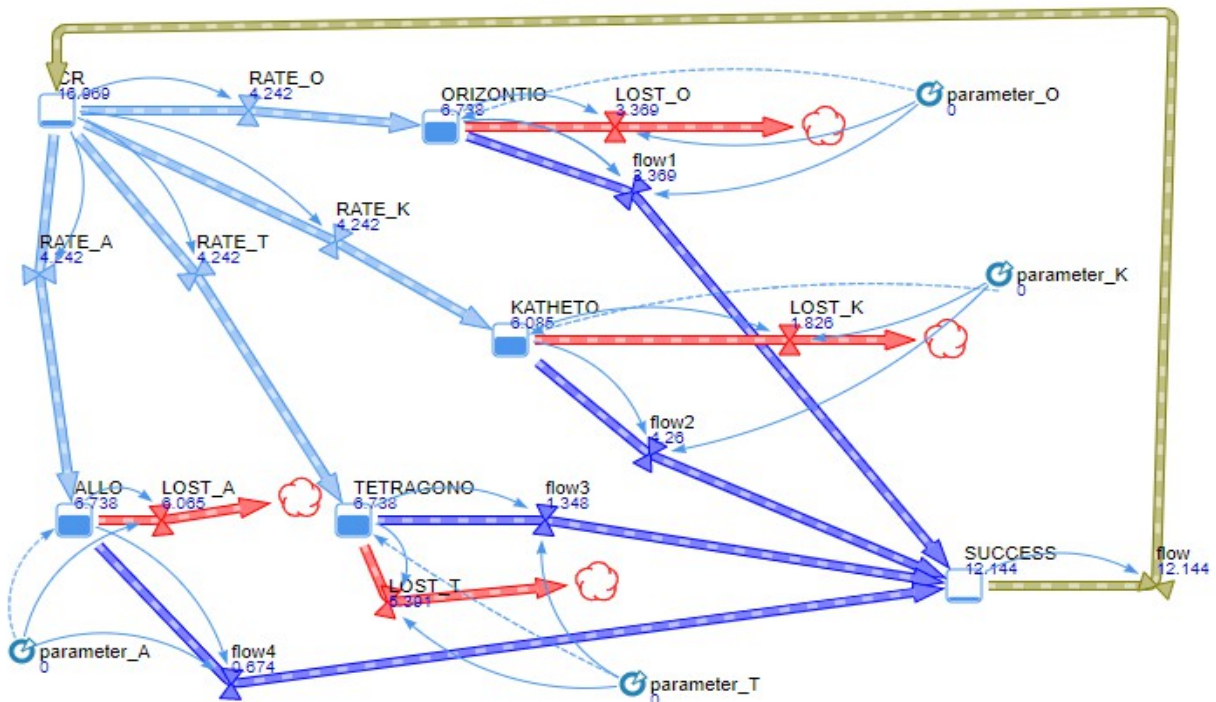
Με τον ίδιο τρόπο όπως και στις προηγούμενες ερευνητικές υποθέσεις, ακολουθούμε και για την ερευνητική υπόθεση Y3: Σχήμα Banners σε μια ιστοσελίδα, οι παράγοντες είναι τέσσερις (4) και συγκεκριμένα ORIZONTIO, KATHETO, TETRAGONO, ALLO. Τα αρχικά ποσοστά που βάλαμε στην αρχική απεικόνιση δυναμικής μοντελοποίησης, ήταν 25%

Το διάγραμμα που προκύπτει είναι το παρακάτω:



Εικόνα 6.11 Διάγραμμα συσχετίσεων παραγόντων της Y3: Σχήμα Banner

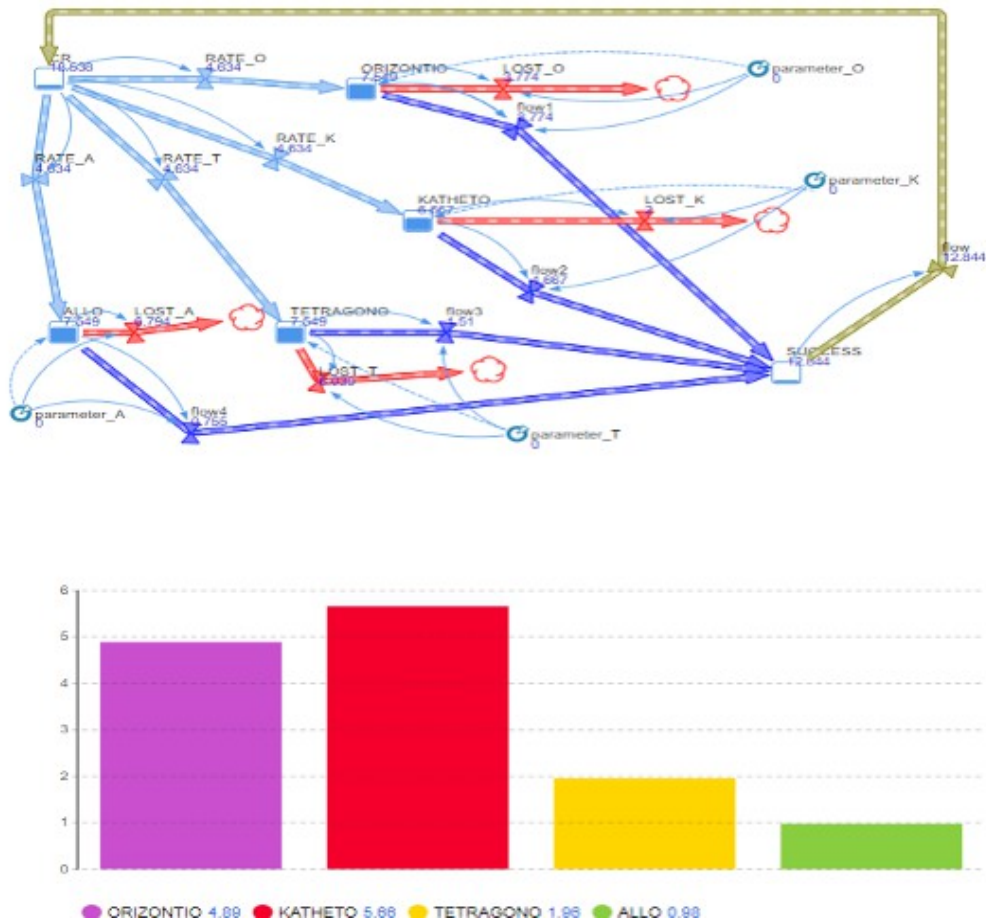
Μετά από την δημιουργία του ανωτέρω διαγράμματος και τις συσχετίσεις μεταξύ των τρέχει το μοντέλο με την επιλογή Run στην άνω μπάρα επιλογών και διαπιστώνεται η ροή του Μοντέλου πως διαμοιράζει τους πόρους και πως αλληλεπιδρούν τα ερευνητικά πεδία.



Εικόνα 6.12: Διάγραμμα ροής με την διανομή των πόρων Y3

6.7 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y3

Για να βελτιστοποιήσουμε το παραπάνω μοντέλο της ερευνητικής υπόθεσης Y3, πήραμε τα ποσοστά που βγήκαν απο την πρωτογενή έρευνά μας και απο το πρόγραμμα του SPSS, δίνοντας το μεγαλύτερο ποσοστό στο KATHETO (ήτοι Κάθετο) banner, καθώςον αυτό ήταν και τα πιο δημοφιλή σύμφωνα με τις απαντήσεις των ερωτώμενών μας.



Εικόνα 6.13: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y3 με chartbars

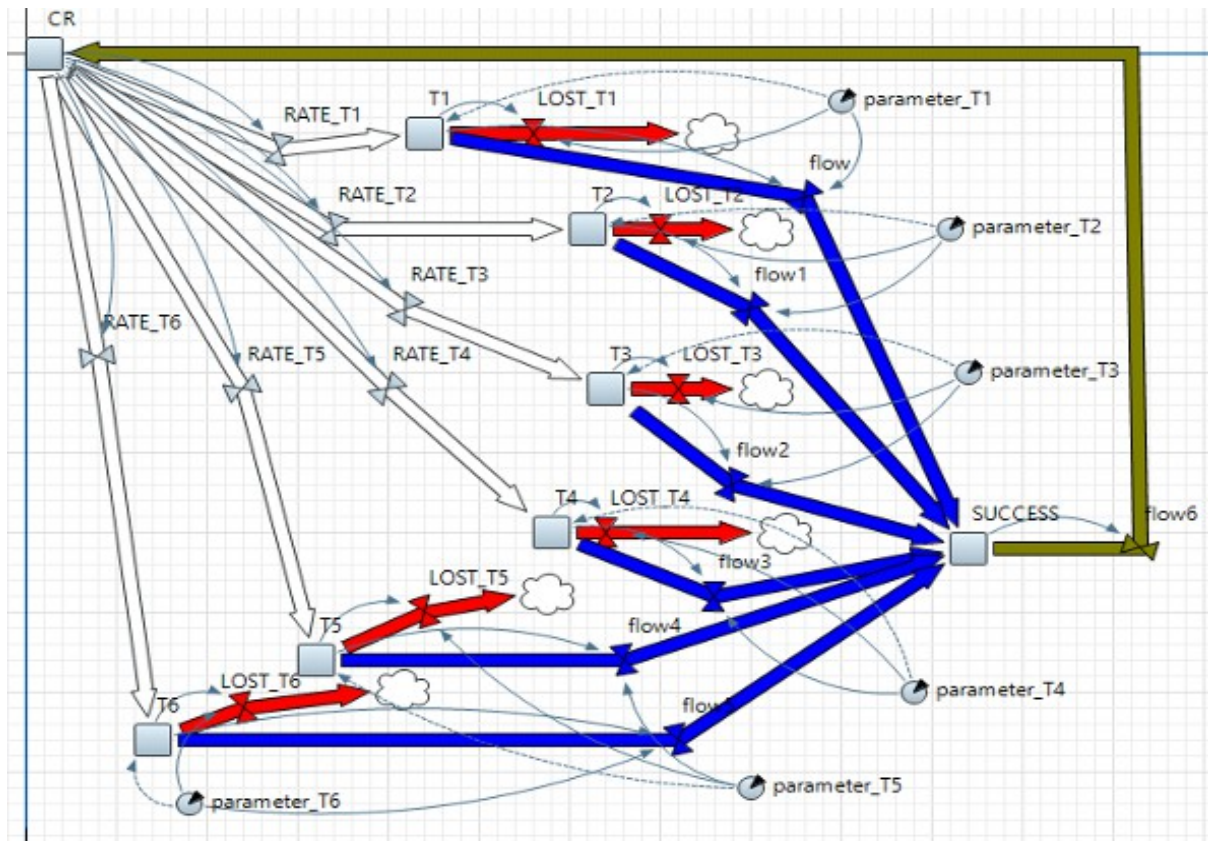
Απο το οποίο προκύπτει και στην βελτιστοποίηση το KATHETO (ήτοι Κάθετο) banner, είναι το καλύτερο απο τα υπόλοιπα.

6.8 Μοντελοποίηση ερευνητικής υπόθεσης Y4

Με τον ίδιο τρόπο όπως και στις προηγούμενες ερευνητικές υποθέσεις, ακολουθούμε και για την ερευνητική υπόθεση Y4: Τοποθεσία Προβολής Banners σε μια ιστοσελίδα, οι παράγοντες είναι εννέα (9), αλλά για λόγους ευκολίας και για να μην περιπλακούν οι παράγοντες πάρα πολύ, εμείς πήραμε τους έξι (6) πρώτους, που είναι και οι πιο δημοφιλή και συγκεκριμένα T1, T2, T3, T4, T5

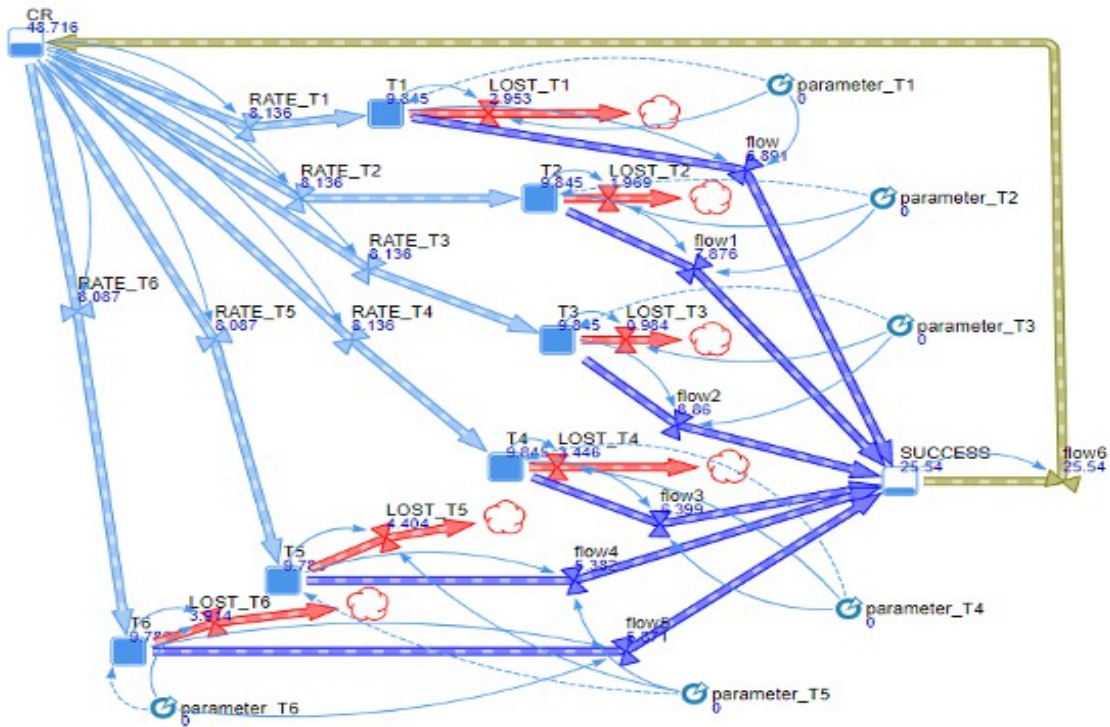
και T6. Τα αρχικά ποσοστά που βάλουμε στην αρχική απεικόνιση δυναμικής μοντελοποίησης, ήταν περίπου 16,66%

Το διάγραμμα που προκύπτει είναι το παρακάτω:



Εικόνα 6.14: Διάγραμμα συσχετίσεων παραγόντων της Y4: Τοποθεσία Προβολής Banner

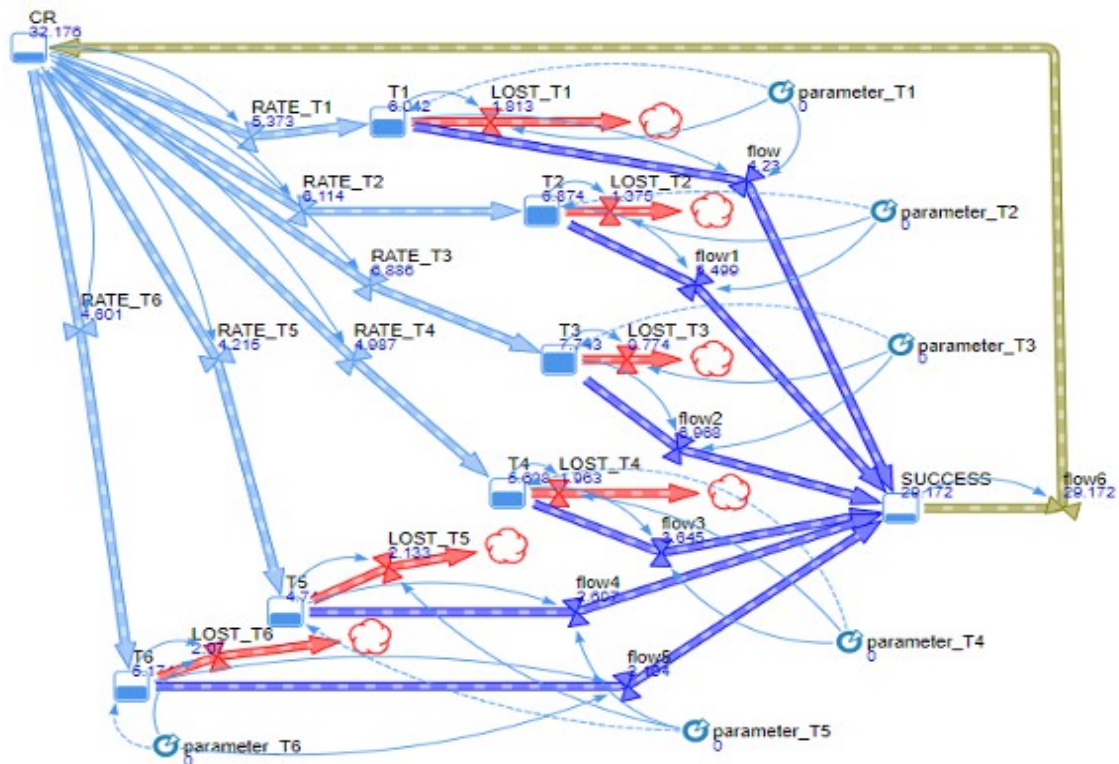
Μετά από την δημιουργία του ανωτέρω διαγράμματος και τις συσχετίσεις μεταξύ των τρέχει το μοντέλο με την επιλογή Run στην άνω μπάρα επιλογών και διαπιστώνεται η ροή του Μοντέλου πως διαμοιράζει τους πόρους και πως αλληλεπιδρούν τα ερευνητικά πεδία.



Εικόνα 6.15: Διάγραμμα ροής με την διανομή των πόρων Y4

6.9 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y4

Για να βελτιστοποιήσουμε το παραπάνω μοντέλο της ερευνητικής υπόθεσης Y4, πήραμε τα ποσοστά που βγήκαν απο την πρωτογενή έρευνά μας και απο το πρόγραμμα του SPSS, δίνοντας το μεγαλύτερο ποσοστό στο T3 (ήτοι στην Τοποθεσία 3, πάνω δεξιά), καθώςον αυτό ήταν και τα πιο δημοφιλή σύμφωνα με τις απαντήσεις των ερωτώμενων μας.



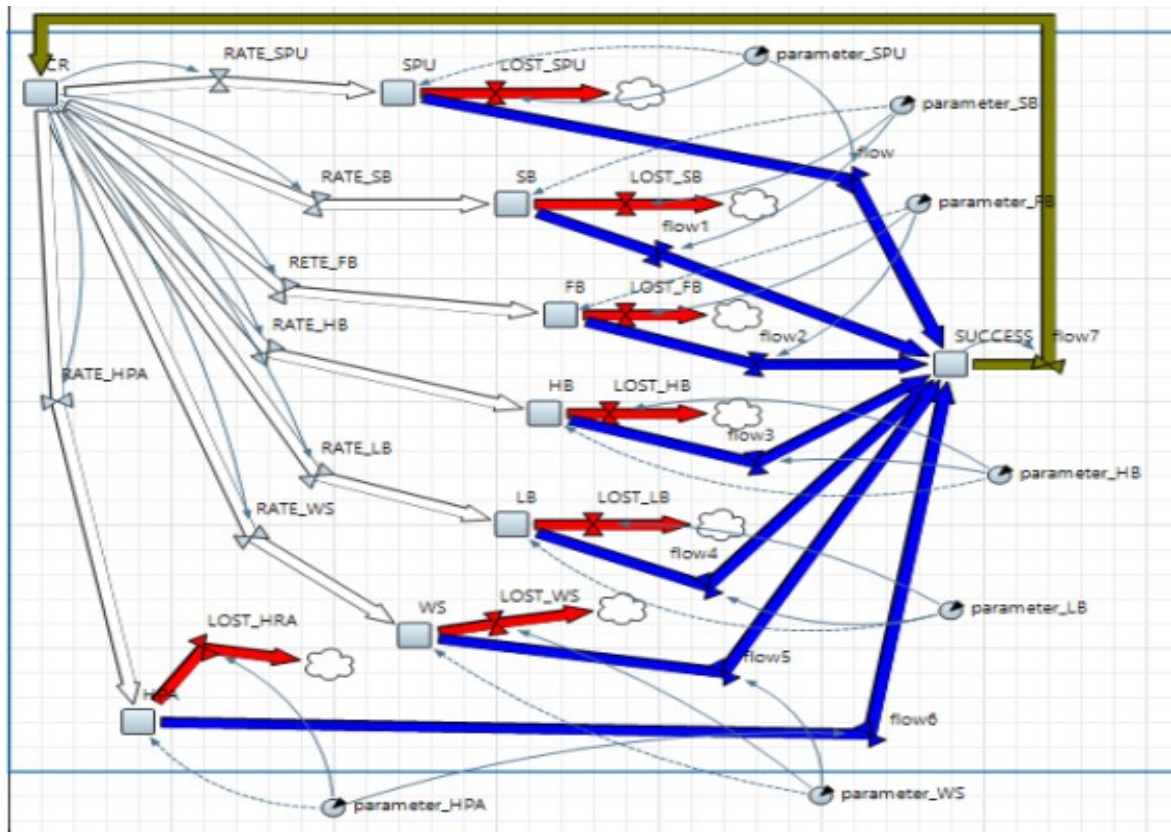
Εικόνα 6.16: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y4

Απο το οποίο προκύπτει και στην βελτιστοποίηση το T3 (ήτοι η Τοποθεσία 3, πάνω δεξιά), είναι η καλύτερη απο τα υπόλοιπα.

6.10 Μοντελοποίηση ερευνητικής υπόθεσης Y5

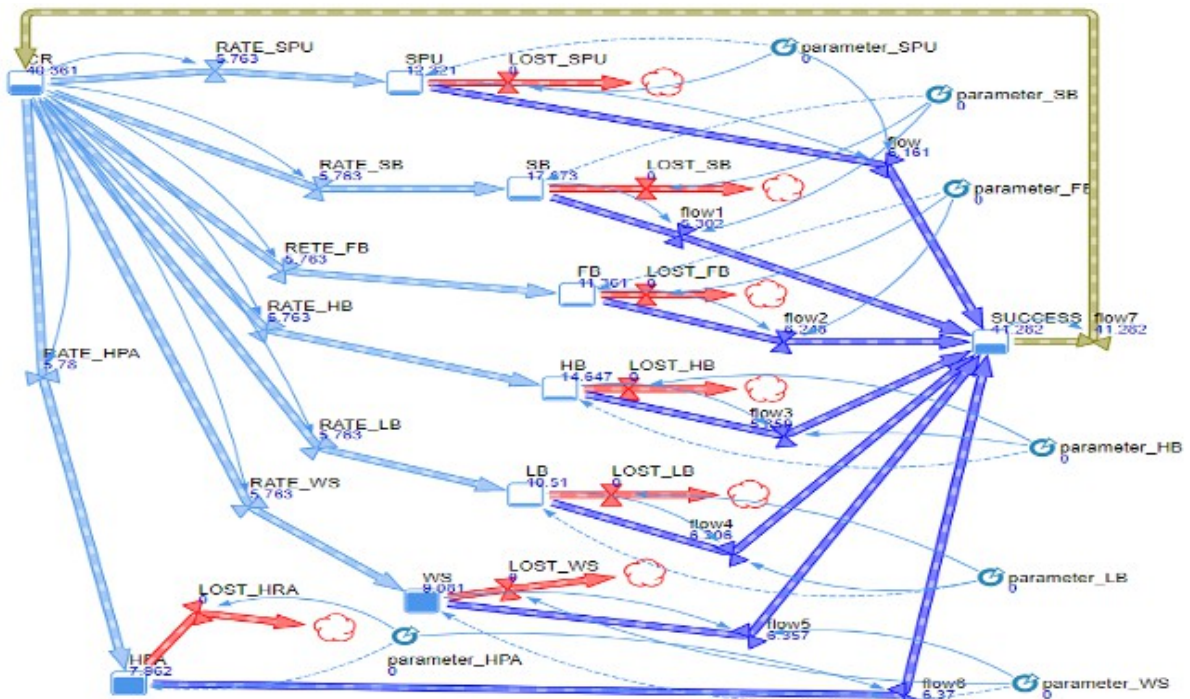
Με τον ίδιο τρόπο όπως και στις προηγούμενες ερευνητικές υποθέσεις, ακολουθούμε και για την ερευνητική υπόθεση Y5: Μέγεθος Banners σε μια ιστοσελίδα, οι παράγοντες είναι επτά (7) και συγκεκριμένα SPU, SB, FB, HB, LB, WS και HRA. Τα αρχικά ποσοστά που βάλαμε στην αρχική απεικόνιση δυναμικής μοντελοποίησης, ήταν περίπου 14,23%

Το διάγραμμα που προκύπτει είναι το παρακάτω:



Εικόνα 6.17: Διάγραμμα συσχετίσεων παραγόντων της Υ5: Μέγεθος Banner

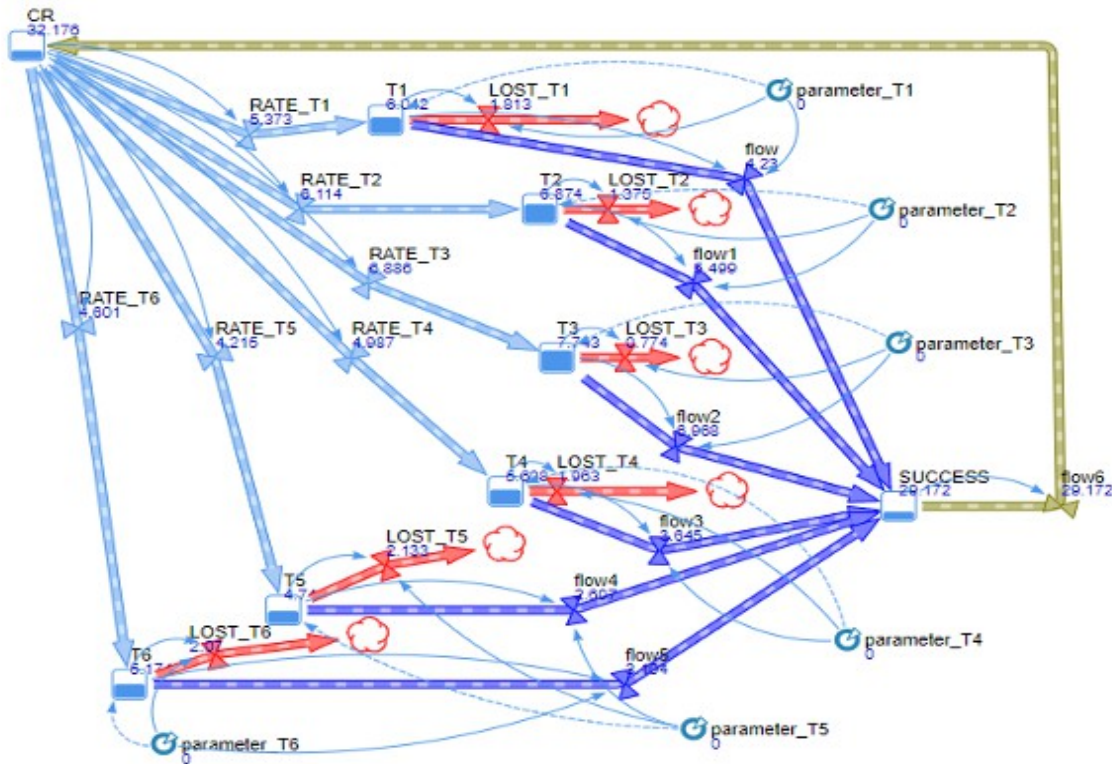
Μετά από την δημιουργία του ανωτέρω διαγράμματος και τις συσχετίσεις μεταξύ των τρέχει το μοντέλο με την επιλογή Run στην άνω μπάρα επιλογών και διαπιστώνεται η ροή του Μοντέλου πως διαμοιράζει τους πόρους και πως αλληλεπιδρούν τα ερευνητικά πεδία.



Εικόνα 6.18: Διάγραμμα ροής με την διανομή των πόρων Υ5

6.11 Βελτιστοποίηση Μοντέλου Ερευνητικής Υπόθεσης Y5

Για να βελτιστοποιήσουμε το παραπάνω μοντέλο της ερευνητικής υπόθεσης Y5, πήραμε τα ποσοστά που βγήκαν απο την πρωτογενή έρευνά μας και απο το πρόγραμμα του SPSS, δίνοντας το μεγαλύτερο ποσοστό στο HRA (ήτοι στο Half Page Ad), καθώςον αυτό ήταν και τα πιο δημοφιλή σύμφωνα με τις απαντήσεις των ερωτώμενών μας.



Εικόνα 6.19: Βελτιστοποίηση Ερευνητικής Υπόθεσης Y5

Απο το οποίο προκύπτει και στην βελτιστοποίηση για το HRA (ήτοι το Half Page Ad), είναι το καλύτερο απο τα υπόλοιπα.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Τα Συμπεράσματα απο την παρούσα πρωτογενή έρευνα είναι τα εξής:

- Οι περισσότεροι καταναλωτές του δείγματος πραγματοποιούν on line αγορές.
- Τα δημοφιλέστερα προϊόντα υψηλής τεχνολογίας που αγοράζουν απο το internet οι καταναλωτές του συγκεκριμένου δείγματος είναι τα Smartphones.
- Η μορφή banners που επιλέγουν οι καταναλωτές είναι αυτά που έχουν Μουσική/Ήχο και τα Animated, με άλλα λόγια επικρατεί το animation σε σχέση με το κείμενο.
- Ο αριθμός των banners που θα πρέπει να υπάρχει σε μια ιστοσελίδα, ώστε να μην είναι ενοχλητική, κυμαίνεται απο 1 έως 2.
- Το σχήμα των Banners, σύμφωνα με τα αποτελέσματα, που κάνει μια διαφήμιση αποτελεσματικότερη είναι το κάθετο.
- Η τοποθεσία των banners, που θεωρείται οτι πρέπει να τοποθετούνται ώστε να είναι περισσότερο ελκυστικά είναι στο πάνω μέρος και δεξιά της ιστοσελίδας
- Το μέγεθος banner και οι διαστάσεις του είναι αυτό του Half Page Ad 300*600, δηλαδή προτιμούνται οι μεγάλες σε μέγεθος διαφημίσεις.

Τα οποία και συμφώνησαν με τα δυναμικά μοντέλα που δημιουργήσαμε στο πρόγραμμα του Any Logic. Συγκεκριμένα:

Τύπος

Y1: Σε ένα banner θα πρέπει να κυριαρχούν τα πολυμέσα (animations) σε σχέση με το κείμενο.

Η κινούμενη διαφήμιση banner ή διαφήμιση που χρησιμοποιεί πολυμέσα -animations-είναι ένας από τους πιο αγαπημένους και αποτελεσματικότερους τύπους διαφήμισης banner μεταξύ των online καταναλωτών (Li & Bukovac, 1999, Dreze & Hussherr, 2003, Diao & Sundar, 2004, Bruce et al., 2017, Chandon et al., 2003, Lohtia et al., 2003, Rosenkrans 2009, Yoo et al. 2004). Η δυναμική αυτή του τύπου διαφήμιση banner αποτελείται από ήχο, βίντεο, εικόνες, φωτεινά και λαμπερά χρώματα, και μοιάζει περισσότερο σαν τις τηλεοπτικές διαφημίσεις (Koegel, 2003). Έχει σχεδιαστεί ώστε να επηρεάσει τους καταναλωτές με τον ίδιο τρόπο όπως κάνει και το τηλεοπτικό κανάλι με μια πρόσθετη διάσταση της διαδραστικότητας. Επομένως, μια διαφημιστική εταιρία θα επιλέξει να χρησιμοποιήσει animated banners, έναντι άλλων ειδών και μορφών banners.

Αριθμός Προβολής Διαφήμισης Banner

Y2: Η προβολή ενός banner σε ένα ιστότοπο και όχι περισσότερων, ώστε να αποφεύγεται η σύγχυση με τα υπόλοιπα διαφημιστικά banners

Ο υπερβολικός αριθμός διαφημίσεων που μπορεί να εμφανίζονται σε μια ιστοσελίδα επιλογής του χρήστη, θεωρείται ως μια σημαντική μεταβλητή που επηρεάζει την στάση των καταναλωτών απέναντι στη διαφήμιση (Ray & Webb, 1986, Kent, 1993, Ha, 1996, Ha & Litmann, 1997, Elliott & Speck, 1998). Για το λόγο αυτό η προβολή απο 1 έως 2 banners θεωρείται ιδανική για μια διαφημιστική εταιρία

Σχήμα

Y3: Τα κάθετα banners είναι πιο ελκυστικά από τα υπόλοιπα σχήματα banners

Το κάθετο banner είναι πιο αποτελεσματικό από το οριζόντιο banner, το οποίο τοποθετείται από αριστερά προς τα δεξιά (Dreze & Hussherr, 2003). Επομένως, μια διαφημιστική εταιρία θα επιλέξει και θα προτείνει τα κάθετα διαφημιστικά banners

Τοποθεσία

Y4: Το banner θα πρέπει να τοποθετηθεί στην κορυφή μιας ιστοσελίδας και όχι στο τέλος της

Οι διαφημίσεις banner που τοποθετούνται στην κορυφή μιας ιστοσελίδας έχουν μεγαλύτερη πιθανότητα να τραβήξουν την προσοχή του χρήστη απο όταν είναι τοποθετημένες στο πάνω μέρος ή στη μέση μιας ιστοσελίδα (Abhishek et al., 2015, Agarwal et al., 2011, Agarwal and Mukhopadhyay 2016, Animesh κ.ά., 2011, Chan and Young-Hoon 2015, Gong κ.ά., 2018, Narayanan and Kalyanam 2015, Rutz κ.ά., 2012). Όπως αποδείξαμε η τοποθεσία πάνω δεξιά είναι η ιδανική και ελκυστικότερη στον καταναλωτή και επομένως μια διαφημιστική εταιρία θα προτείνει την θέση αυτή για προβολή της διαφήμισης.

Μέγεθος

Y5: Η μεγάλη σε μέγεθος διαφήμιση banner είναι αποδοτικότερη από την μικρού μεγέθους διαφήμιση banner

Μια μεγαλύτερη διαφήμιση είναι πιο αποτελεσματική από μια μικρότερη (Bruce et al., 2017, Chandon et al., 2003, Rosenkrans 2009, Sokolik et al., 2014). Στην συγκεκριμένη έρευνα η ελκυστικότερη διαφήμιση ήταν Half Page Ad, την οποία θα την προτείνει μια διαφημιστική εταιρία στους πελάτες της.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Ξενογλώσση

- Abhishek, V., Hosanagar, K., and Fader, P. S. (2015). "Aggregation Bias in Sponsored Search Data: The Curse and the Cure," *Marketing Science*, (34:1), 59–77.
- Agarwal, A., Hosanagar, K., and Smith, M. D. (2011). "Location, Location, Location: An Analysis of Profitability of Position in Online Advertising Markets," *Journal of Marketing Research (JMR)*, (48:6), 1057–1073.
- Agarwal, A., and Mukhopadhyay, T. 2016. "The Impact of Competing Ads on Click Performance in Sponsored Search," *Information Systems Research* (27:3), pp. 538–557.
- Aguirre, E., Mahr, D., Grewal, D., Ruyter, K. de, and Wetzels, M. (2015). "Unraveling the Personalization Paradox: The Effect of Information Collection and Trust-Building Strategies on Online Advertisement Effectiveness," *Journal of Retailing* (91:1), 34–49.
- Alijani, G.S., Mancuso, L.C., Kwun, O. and Omar, A. (2010). Effectiveness of online advertisement factors in recalling a product. *Academy of Marketing Studies Journal*, Vol. 14, No. 1, 1-10.
- Animesh, A., Viswanathan, S., and Agarwal, R. (2011). "Competing "Creatively" in Sponsored Search Markets: The Effect of Rank, Differentiation Strategy, and Competition on Performance," *Information Systems Research* (22:1), 153–169.
- Baltas, G. (2003). Determinants of internet advertising effectiveness: an empirical study. *International Journal of Market Research*, Vol. 45, No. 4, 505-513.
- Baraggioli, F. & Brasel, S.A. (2008). Visual velocity: content font effects and incidental online ad exposure. *Advances in Consumer Research*, Vol. 35, 600-606.
- Barreto, A. (2013) "Do users look at banner ads on Facebook?," *Journal of Research in Interactive Marketing*, Vol. 7, Iss 2, 119-139.
- Beal V., "PDA - personal digital assistant," [Online]. Available: <http://www.webopedia.com/TERM/P/PDA.html>.
- Becker-Olsen, K. (2003) "And now, a word from our sponsor - a look at the effects of sponsored content and banner advertising", *Journal of Advertising*, Vol. 32, Iss. 2, 17-32.
- Bradley G. W., Blashki K., (2006). A new look into iTV & interactivity, Universitaet Leuven, Leuven, Dutch.
- Breuer, R. and Brettel, M. (2012) "Short- and Long-term Effects of Online Advertising: Differences

- between New and Existing Customers”, *Journal of Interactive Marketing*, Vol. 26, Iss. 3, 155-166.
- Briggs, R. & Hollis, N. (1997). Advertising on the web: is there response before clickthrough? *Journal of Advertising Research*, Vol. 37, No. 2, pp. 33-46.
- Bruce, N. I., Murthi, B.P.S., and Rao, R. C. (2017). “A Dynamic Model for Digital Advertising: The Effects of Creative Format, Message Content, and Targeting on Engagement,” *Journal of Marketing Research* (54:2), 202–218.
- Burke, M., Hornof, A., Nilsen E and Gorman, N. (2005) “High-cost banner blindness: ads increase perceived workload, hinder visual search, and are forgotten”, *Journal ACM Transactions on Computer-Human Interaction*, Vol. 12, Iss. 4, 423-445.
- Burke, R. R., & Srull, K. T. (1988). Competitive interference and consumer memory for advertising. *Journal of Consumer Research*, 15(1), 55–67.
- Chan, T. Y., and Young-Hoon, P. (2015). “Consumer Search Activities and the Value of Ad Positions in Sponsored Search Advertising,” *Marketing Science* (34:4), 606–623.
- Chandon, J. L., Chtourou, M. S., and Fortin, D. R. (2003). “Effects of configuration and exposure levels on responses to web advertisements,” *Journal of Advertising Research* (43:2), 217–229.
- Chatterjee, P., Hoffman, D. L., & Novak, T. P. (2003). Modeling the clickstream: Implications for web-based advertising efforts. *Marketing Science*, 22(4), 520– 541.
- Chi, H.K., Yeh, H.R. and Chiou, C.Y. (2012). The mediation effect of information presentation style on the relationship between banner advertisements and advertising effectiveness. *International Journal of Business and Management*, Vol. 7, No. 14, 46-52.
- Cho, C. (1999) “How Advertising Works on the WWW: Modified Elaboration Likelihood Model”, *Journal of Current Issues & Research in Advertising*, Vol. 21, Iss. 1, 34-50.
- Cho, C., Lee, J., & Tharp, M. (2001). Different forced-exposure levels to banner advertisements. *Journal of Advertising Research*, 41(4), 45–56.
- Cho, C., & Leckenby, J. D. (2000). *The impact of banner exposure and clicking on attitude change*. Paper presented at the American Academy of Advertising, Newport, Rhode Island.
- Chtourou, M.S. & Abida, F.C. (2010). What makes one animation more effective than another? an exploratory study of the characteristics and effects of animation in internet advertising.

- Coviello N., Milley R., Marcolin B., (2001). Understanding IT-enabled interactivity.
- Diao, F., & Sundar, S. S. (2004). Orienting response and memory for web advertisements: Exploring effects of pop-up window and animation. *Communication Research*, 31(5), 537–567.
- Diao, F., & Sundar, S. S. (2004). Orienting response and memory for web advertisements: Exploring effects of pop-up window and animation. *Communication Research*, 31(5), 537–567.
- Dreze, X., & Hussherr, F. X. (2003). Internet advertising: Is anybody watching? *Journal of Interactive Marketing*, 17(4), 8–23.
- Dreze, X., & Hussherr, F. X. (2003). Internet advertising: Is anybody watching? *Journal of Interactive Marketing*, 17(4), 8–23.
- Drèze, X. & Hussherr, F.X. (2003). Internet advertising: is anybody watching? *Journal of Interactive Marketing*, Vol. 17, No. 4, 8-23.
- du Plessis, Erick and Brown, Millward, (2005). The advertised mind: groundbreaking insights into how our brains respond to advertising. Millward Brown and Kogan Page Limited, London. , ISBN 0 7494 4366 9.
- Edwards, S. M., Li, H., & Lee, J. (2002). Forced exposure and psychological reactance: Antecedents and consequences of the perceived intrusiveness of pop-up ads. *Journal of Advertising*, 33(3), 83–95.
- Elliott, M. T., & Speck, P. S. (1998). Consumer perceptions of advertising clutter and its impact across various media. *Journal of Advertising Research*, 38(1), 29–41.
- Euijin, A., & Edwards, S. M. (2002). *Does size really matter? Brand attitude versus click-through in response to banner ads*. Paper presented at the American Academy of Advertising annual conference, Jacksonville, Florida.
- EZlistmailer.com (2009). Advertising for e-marketing professionals. White paper. Bowie, MD: Author. Retrieved August 11, 2009, from <http://www.ezlistmailer.com/images/downloads/EZL-Advertising for E-marketing Professionals.swf>.
- Flores, W., Victor Chen, J. and Ross W. (2014) “The effect of variations in banner ad, type of product, website context, and language of advertising on Internet user’s attitudes”, *Computers in Human Behavior*, Vol. 31, 37-47.

- Gehrke D., & Turban E. (1999). Determinants of Successful Website Design: Relative Importance and Recommendations for Effectiveness. International Conference on System Sciences. Hawaii.
- Ghajarzadeh, A., Sahebjamnia, N., Sahaleh, K., Chavosh, A.; Halimi, A. B. (2010), A new model of online advertising effectiveness on customer responsiveness: a case of laptop companies in Malaysia. *Journal of Business and Policy Research* 5(2), 237 – 261.
- Ghosh, H. & Bhatnagar, A. (2013). On Measuring and Increasing the Effectiveness of Banner Advertising. *MIS Review* Vol. 19, No. 1, September (2013), 25-44.
- Gibert J.E. (1999). E-Commerce & Web Branding: A Use For Persuasive Technologies. *BDPA Journal Online*, 1(2). Διαθέσιμο στην ιστοσελίδα <http://www.bdpa.org/>
- Gong, J., Abhisek, V., and Li, B. (2018). “Examining the Impact of Keyword Ambiguity on Search Advertising Performance: A Topic Model Approach,” *MIS Quarterly* (42:3), 805–829.
- Grigorovici D.M., Constantin C.D. (2004). Experiencing Interactive Advertising beyond Rich Media. *Journal of Interactive Advertising*, 5(1), σελ. 22-36.
- Ha, L. (1996). Advertising clutter in consumer magazine: dimensions and effects. *Journal of Advertising Research*, 36(4), 76–84.
- Ha, L., & James, E. L. (1998). Interactivity re-examined: A baseline analysis of early business web sites. *Journal of Broadcasting and Electronic Media*, 43(4), 457–474.
- Ha, L., & Litmann, B. R. (1997). Does advertising clutter have diminishing and negative returns? *Journal of Advertising*, 26(1), 31–42.
- Hendon, D. W. (1973). How mechanical factors affect ad perception. *Journal of Marketing Research*, 13(4), 39–45.
- Hill R., & Dhanda K., (2015). Advertising, technology, and the digital divide: A global perspective. *International Advertising Research*, 175-193.
- Hofacker, C. and Murphy J. (1998) “World Wide Web banner advertisement copy testing”, *European Journal of Marketing*, Vol. 32, No. 7/8, 703-712.
- Hofacker, C. and Murphy J. (1998) “World Wide Web banner advertisement copy testing”, *European Journal of Marketing*, Vol. 32, No. 7/8, 703-712.
- Hofacker, C. F., & Murphy, J. (1998). World Wide Web banner advertisement copy testing. *European Journal of Marketing*, 32(7/8), 703–712.

- Hopkins J., & Turner J. (2012), *Go Mobile*. New York: John Wiley & Sons Inc.
- Hussain R., Sweeny A., Mort G.S. (2010). Typologies of Banner Advertisements' Attributes: A Content Analysis. *Journal of Promotion Management*, 16(1-2), 96-113.
- Janiszewski, C. (1990). The influence of nonattended material on the processing of advertising claims. *Journal of Marketing Research*, 27(3), 263–278.
- Josephson, S. (2005). Eye tracking methodology and the internet, in *Handbook of visual communication: Theory, methods, and media*, New Jersey: Lawrence Erlbaum, 63-81.
- Ju, B. (2013). A proposed cross-cultural examination of online advertising effectiveness in china and the UK. *International Journal of Business and Management*, Vol. 8, No. 6, 34-39.
- Keller, K. L. (1987). Memory factors in advertising: The effect of advertising retrieval cues on brand evaluations. *Journal of Consumer Research*, 14(3), 316– 333.
- Keller, K. L. (1991). Memory and evaluation effects in competitive advertising environments. *Journal of Consumer Research*, 17(4), 463–476.
- Kent, R. (1993). Competitive versus noncompetitive clutter in television advertising. *Journal of Advertising Research*, 33(2), 40–46.
- Kim, H. G., & Leckenby, J. D. (2002). *Creative factors in interactive advertising*. Paper presented at the American Academy of Advertising annual conference, Jacksonville, Florida.
- Kirmani, A. (1990). The effects of perceived advertising cost on brand perception. *Journal of Consumer Research*, 17(2), 160–171.
- Koegel, K. (2003, July). *Richmedia: What? Where? Why? A Double Click white paper*. New York: DoubleClick.
- Lee, J.W. & Ahn, J.H. (2012). Attention to banner ads and their effectiveness: an eyetracking approach. *International Journal of Electronic Commerce*, Vol. 17, No. 1, 119-138.
- Lekkakos G., Chorianopoulos K., Spinellis D., (2001). In *Proceedings of the 9th European Conference on Information Systems*, Bled, Slovenia, 319-329.
- Li, H., & Bukovac, J. L. (1999). Cognitive impact of banner ad characteristics: An experimental study. *Journalism and Mass Communication Quarterly*, 76(2), 341–353.
- Li, H., & Bukovac, J. L. (1999). Cognitive impact of banner ad characteristics: An experimental study. *Journalism and Mass Communication Quarterly*, 76(2), 341–353.

- Ling M., Lawle, K., Mcbain N., Moscardini A. (1999). Economics of advertising: Emerging functions of Internet advertising. *Netnomics*, 1, 127 – 136.
- Lohtia, R., Donthu, N., and Hershberger, E. K. (2003). “The impact of content and design elements on banner advertising click-through rates,” *Journal of Advertising Research* (43:4), 410–418.
- Mattke, J., Müller, L., and Maier, C. (2019). “Paid, Owned and Earned Media: A Qualitative Comparative Analysis revealing Attributes Influencing Consumer’s Brand Attitude in Social Media,” Hawaii International Conference on System Sciences 2019 (HICSS-52).
- Möller K., Halinen A. (2000), Relationship marketing theory: its roots and direction, *Journal of Marketing Management*, Vol. 16, 29-54.
- Möller, J. & Eisend, M. (2010). A global investigation into the cultural and individual antecedents of banner advertising effectiveness. *Journal of International Marketing*, Vol. 18, No. 2, 80-98.
- Narayanan, S., and Kalyanam, K. (2015). “Position Effects in Search Advertising and their Moderators: A Regression Discontinuity Approach,” *Marketing Science* (34:3), 388–407.
- Pabedinskaitė, A., Fiodorovaitė, D. (2011). E-marketing for higher education institution, in 10th International Conference, Liberec Economic Forum 2011“. Liberec, Czech Republic 19-20 September 2011, 382-391.
- Palanisamy, R. (2004). Impact of gender differences on online consumer characteristics on web-based banner advertising effectiveness. *Journal of Services Research*, Vol. 4, No. 2, 45-55.
- Pharr J.M. (2004). Research Agenda for Brand-Building on the Internet with Banner Advertising. *Services Marketing Quarterly*, 26(2), σελ. 39-47.
- Raudienė I. (2011). Effective online advertising model. Master thesis. Supervisor: Davidavičienė V. Vilnius: VGTU.
- Ray, M. L., & Webb, P. H. (1986). Three prescriptions for clutter. *Journal of Advertising Research*, 26(1), 69–77.
- Razzouk, N., & Seitz, V. A. (2003). Banner advertising and consumer recall: An empirical study. *Journal of Promotion Management*, 9(1/2), 71–80.
- Robinson, H., Wysocka, A. & Hand, C. (2007). Internet advertising effectiveness: the effect of design on click-through rates for banner ads. *International Journal of Advertising*, Vol. 26, No. 4, 527-541.

- Rosenkrans G. (2009). The Creativeness and Effectiveness of Online Interactive Rich Media Advertising. *Journal of Interactive Advertising*, 9(2), 8-31.
- Rosenkrans, G. (2010). Maximizing user interactivity through banner ad design. *Journal of Promotion Management*, Vol. 16, No. 3, 265.
- Rosenkrans, G. (2009). "The creativeness and effectiveness of online interactive rich media advertising," *Journal of interactive advertising* (9:2), 18–31.
- Rutz, O. J., Bucklin, R. E., and Sonnier, G. P. (2012). "A Latent Instrumental Variables Approach to Modeling Keyword Conversion in Paid Search Advertising," *Journal of Marketing Research (JMR)* (49:3), 306–319.
- Sager I., (2012), Before iPhone and Android Came Simon, the First Smartphone. Bloomberg Businessweek.
- Salmenkivi et Nyman. (2007). Yhteisöllinen media ja muuttuva markkinointi 2.0. Talentum Media Oy, Helsinki. ISBN: 978-952-14-1239-4. 311 -315.
- Scanlon J., "Did Smartphones Kill the Handheld Gaming Console?,". [Online]. Available: <http://wallstcheatsheet.com/technology/did-smartphones-kill-the-handheld-gaming-console.html/>.
- Sigala M. (2005). Customer Relationship Management (CRM) Evaluation: Diffusing CRM benefits into business processes. Διαθέσιμο στην ιστοσελίδα <http://is2.lse.ac.uk/asp/aspecis/20040157.pdf>.
- Sokolik, K., Magee, R. G., and Ivory, J. D. (2014). "Red-Hot and Ice-Cold Web Ads: The Influence of Web Ads' Warm and Cool Colors on Click-Through Rates," *Journal of interactive advertising* (14:1), 31–37.
- Thota, S.C., Song, J.H. & Larsen, V. (2010). Do animated banner ads hurt websites? The moderating roles of website loyalty and need for cognition. *Academy of Marketing Studies Journal*, Vol. 14, No. 1, 91-116.
- Vadell J.B.G. (2005). Internet Use in the Lodging Industry: Attitudes, Opinions and Perceptions Towards its Implementation. *Anatolia: An International Journal of Tourism and Hospitality Research*, 16(2), 162-175.
- Wang, K.Y., Shih, E. & Peracchio, L.A. (2013). How banner ads can be effective: investigating the influences of exposure duration and banner ad complexity. *International Journal of*

Advertising, Vol. 32, No. 1, 121-141.

Yaveroglu, I. & Donthu, N. (2008). Advertising repetition and placement issues in on-line environments. *Journal of Advertising*, Vol. 37, No. 2, pp. 31-43.

Yoo, C. Y., Kim, K., and Stout, P. A. (2004). "Assessing the Effects of Animation in Online Banner Advertising: Hierarchy of effects model," *Journal of interactive advertising*, (4:2), 49–60.

Zeff R., Aronson B. (1999). Advertising on the Internet. New York: Wiley Computer Publishing.

Zhang, P. (2000). The effects of animation on information seeking performance on the world wide web: securing attention or interfering with primary tasks? *Journal of Association Information Systems*, 1(1), 1- 28.

Ελληνική

Βλαχοπούλου Μ. (2003). e Marketing, Διαδικτυακό Μάρκετινγκ. Αθήνα: Rosili.

Βλάχος Π. & Δρόσος Δ. (2004). Νέες τεχνολογίες και διαφήμιση. Αθήνα: E-Business Forum.

Βουχάρα Σ. (2009). e-Tourism: Χρήση του Διαδικτύου & των τεχνολογιών πληροφορίας & επικοινωνιών στον τομέα του τουρισμού. Μεταπτυχιακή Διπλωματική Εργασία, Πάντειο Πανεπιστήμιο.

Δημητριάδης Σ. & Μπάλτας, Γ. (2003), Ηλεκτρονικό Εμπόριο και Μάρκετινγκ, 1η έκδοση. Εκδόσεις Rosili: Αθήνα.

Ευαγγελάτος Α., (2010). Διαφήμιση στο Ίντερνετ, Πανεπιστημιακές σημειώσεις, Εργαστήριο Εταιρικής Επικοινωνίας ΟΠΑ.

Καράμπελα Χ., (2012), Μελέτη χρήσης και παιχνιδιού Flash για διαφημιστικούς σκοπούς, Διπλωματική εργασία, ΑΤΕΙ ΚΡΗΤΗΣ.

Κοκκίνης Γ., (2017). Εισαγωγή στην Διαφήμιση. Β τάξη ΕΠΑΛ. Αθήνα: Εκδόσεις Διόφαντος.

Κόκοτος Χ. Δ. & Λιναρδάτος, Σ. Δ. (2009), Πρόβλεψη ενεργειών και υποστήριξη λήψης αποφάσεων στο Ηλεκτρονικό Επιχειρείν. Εκδόσεις Σταμούλη Α.Ε.: Αθήνα.

Μπαλτάς Γ. & Παπασταθοπούλου, Π., (2013). Συμπεριφορά Καταναλωτή. Β έκδοση. Αθήνα: Εκδόσεις Rosili.

Μπραϊμή, Α. (2017). Διαδικτυακή Διαφήμιση: Η στάση των καταναλωτών στον Πειραιά. Πτυχιακή εργασία: ΤΕΙ Κρήτης.

Παπαδοπούλου Σ., (2009). «ΔΙΑΔΙΚΤΥΑΚΗ ΔΙΑΦΗΜΙΣΗ: Η ΔΙΑΦΗΜΙΣΤΙΚΗ ΙΣΤΟΣΕΛΙΔΑ Η

ΠΡΟΒΟΛΗ ΚΑΙ Η ΠΡΟΩΘΗΣΗ ΤΗΣ ΜΕΣΑ ΑΠΟ ΤΙΣ ΜΗΧΑΝΕΣ ΑΝΑΖΗΤΗΣΗΣ, Η ΠΕΡΙΠΤΩΣΗ ΤΗΣ GOOGLE ΚΑΙ ΜΕΛΕΤΕΣ ΠΕΡΙΠΤΩΣΕΩΝ», Πτυχιακή Εργασία στο Τμήμα Εμπορίας και Διαφήμισης του Αλεξάνδρειου Τεχνολογικού Εκπαιδευτικού Ιδρύματος Θεσσαλονίκης, Διαθέσιμη στο:

<http://eureka.lib.teithe.gr:8080/bitstream/handle/10184/2110/Papadopoulou%20S.%20%5BMain%5D.pdf?sequence=1>.

Πασχόπουλος Α. & Σκαλτσάς, Π. (2009). Ηλεκτρονικό Εμπόριο. Αθήνα: Κλειδάριθμος.

Φρίγκας Γ., (2005). Διαφήμιση και Μάρκετινγκ στο Διαδίκτυο. Αθήνα: Κριτική.

Ιστοσελίδες

"Computer Basics - Mobile Devices," [Online]. Available:

<http://www.gcflearnfree.org/computerbasics/9>.

"A Complete History of Portable Music Players," 09 06 2014 . [Online]. Available:

<http://www.ebay.com/gds/A-Complete-History-of-Portable-Music-Players-/10000000177628958/g.html>.

Handheld Game Console Review," [Online]. Available: <http://handheld-game-console-review.toptenreviews.com/>.

"App Store Metrics," [Online]. Available: <http://www.pocketgamer.biz/metrics/app-store/>.

"What is GPS?," [Online]. Available: <http://www8.garmin.com/aboutGPS/>.

IAB – Interactive Advertising Bureau, Οργανισμός Διαδραστικής Επικοινωνίας, διαθέσιμο στο: <http://www.iab.gr/gr/organization/>

IAB – Interactive Advertising Bureau, Τρόποι αξιοποίησης του διαδικτύου από τους διαφημιζόμενους, διαθέσιμο στο: <http://www.iab.gr/gr/research/research-list/>.

ΠΑΡΑΡΤΗΜΑ Ι

ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ

Το παρόν ερωτηματολόγιο συντάχθηκε στα πλαίσια διπλωματικής εργασίας που σκοπό έχει να μελετήσει την αποτελεσματικότητα της διαφήμισης τύπου banner. Τα στοιχεία θα παραμείνουν κρυφά και δεν θα διακινηθούν, πωληθούν, επανατοποθετηθούν σε άλλες λίστες ή θα χρησιμοποιηθούν για προώθηση προϊόντων και υπηρεσιών.

Η συμπλήρωση του ερωτηματολογίου απαιτεί 5 με 6 λεπτά από τον χρόνο σας.

Παρακαλούμε απαντήστε στις ερωτήσεις που ακολουθούν με σαφήνεια. Συμπληρώστε την απάντησή σας στο κενό δίπλα σε κάθε ερώτηση.

A) Δημογραφικά Στοιχεία

1. Φύλο:

- Άνδρας
 Γυναίκα

2. Ηλικία:

- 18-30
 31-45
 46-60
 60-74

3. Εκπαιδευτικό Επίπεδο:

- Απόφοιτος Λυκείου
 ΑΕΙ/ ΤΕΙ
 Μεταπτυχιακό
 Άλλο

B) Προϊόντα Υψηλής Τεχνολογίας

Επιλέξτε κατά πόσο συμφωνείτε ή διαφωνείτε για κάθε μία από τις παρακάτω προτάσεις σε μια κλίμακα από το ένα (1) ως το πέντε (5), όπου το 1 θα υποδηλώνει «Καθόλου», το 2 «Λίγο», το 3 «Μέτρια», το 4 «Πολύ» και το 5 «Πάρα Πολύ».

4. Πραγματοποιείτε online αγορές σε προϊόντα υψηλής τεχνολογίας?

	1	2	3	4	5

5. Εφόσον πραγματοποιείτε online αγορές, ποιο/ ποια από τα παρακάτω προϊόντα υψηλής τεχνολογίας αγοράζετε συνήθως?

(μπορείτε να επιλέξετε περισσότερες από μια απαντήσεις)

	1	2	3	4	5
Έξυπνα τηλέφωνα (Smartphone's)					
Φορητός υπολογιστής (Laptop)					
Έξυπνα ρολόγια (Smart watches)					
Υπολογιστής ταμπλέτα (Tablets)					
Φορητές συσκευές Αναπαραγωγής πολυμέσων (Portable Media Players)					
Συσκευή αναπαραγωγής πολυμέσων (iPod)					
Φορητές παιχνιδιομηχανές (Handheld Game Consoles)					
Συσκευές πλοήγησης (Global					

Positioning System)					
---------------------	--	--	--	--	--

Γ) Στοιχεία Διαφήμισης Τύπου Banner

(Banner: ορθογώνια εικονίδια διαφόρων μεγεθών τα οποία λειτουργούν ως σύνδεσμοι στις διάφορες ιστοσελίδες)

Υ1: Η μορφή προβολής της banner διαφήμισης

6. Κατά πόσο τα παρακάτω στοιχεία τραβούν την προσοχή σας σε μια διαφήμιση banner?

	1	2	3	4	5
Το σταθερό/ στατικό κείμενο					
Το κινούμενο μήνυμα ή κείμενο					
Οι πολλές εικόνες					
Τα έντονα/ζωηρά χρώματα					
Το υπόβαθρο του μηνύματος					
Η μουσική ή ο τυχόν ήχος					
Η ευκολία πλοήγησης στο προϊόν που διαφημίζεται					
Η ταχύτητα φόρτωσης της ιστοσελίδας του διαφημιζόμενου προϊόντος					

7. Ποια μορφή banner διαφήμισης παρατηρείτε περισσότερο στο διαδίκτυο?

	1	2	3	4	5
Τα στατικά (static) banners					
Τα κινούμενα (animated) banners					
Τα διαδραστικά (interactive) banners					
Τα αναδυόμενα (Pop-up) banners					
Τα banners που περιέχουν ενεργές λέξεις ή εικόνες					
Τα banners που περιέχουν παιχνίδια ή παζλ					
Τα banners που περιέχουν ήχο και video					
Τα banners σε μορφή ανακοινώσεων					
Τα banners με φωτεινά χρώματα που αναβοσβήνουν					

Υ2: Αριθμός Προβολής banner σε μια Ιστοσελίδα

8. Πόσα σε αριθμό banners θεωρείτε ότι πρέπει να προβάλλονται σε μια ιστοσελίδα που επισκέπτεστε, έτσι ώστε να μην είναι ενοχλητικά και να τραβήξουν την προσοχή σας?

	1	2	3	4	5
Από 1 έως 2					
Από 2 έως 3					
Από 3 έως 4					
Από 4 έως 5					
Από 6 και πάνω					

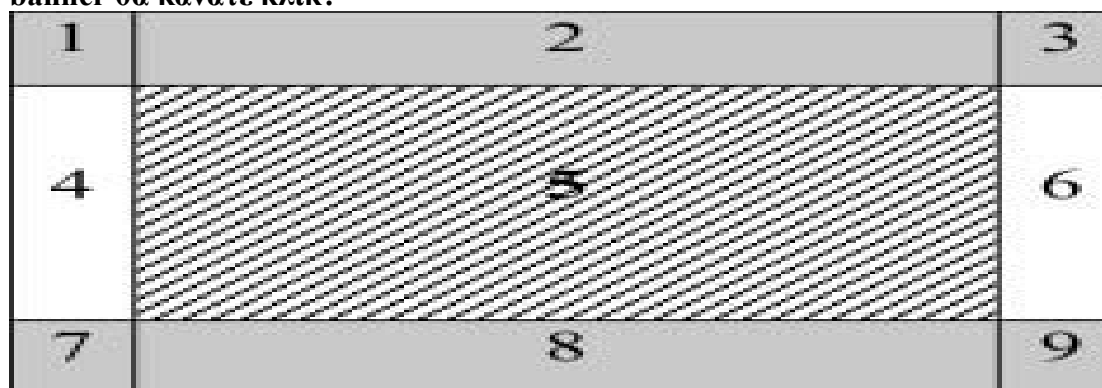
Υ3: Το Σχήμα της banner διαφήμισης

9. Ποιο από τα παρακάτω σε σχήμα banner θα κάνετε κλικ?

	1	2	3	4	5
Οριζόντιο banner					
Κάθετο banner					
Τετράγωνο banner					
Άλλο banner					

Υ4: Η Τοποθεσία προβολής της banner διαφήμισης




10. Σε ποια από τις παρακάτω τοποθεσίες μιας ιστοσελίδας που προβάλλονται διαφημίσεις banner θα κάνετε κλικ?



	1	2	3	4	5
Η τοποθεσία 1, πάνω αριστερά					
Η τοποθεσία 2, πάνω κεντρικά					
Η τοποθεσία 3, πάνω δεξιά					
Η τοποθεσία 4, μέση αριστερά					
Η τοποθεσία 5, κεντρικά					
Η τοποθεσία 6, μέση δεξιά					
Η τοποθεσία 7, κάτω αριστερά					
Η τοποθεσία 8, κάτω κεντρικά					
Η τοποθεσία 9, κάτω δεξιά					

Υ5: Το Μέγεθος της banner διαφήμισης

11. Ποιο από τα παρακάτω σε μέγεθος (διαστάσεις) banner είναι πιο ελκυστικά:

	1	2	3	4	5
 Τετράγωνο Μεγάλο 250*250					
 Τετράγωνο Μικρό 125*125					
 Οριζόντιο Κανονικό 468*60					

234x60	Οριζόντιο Μικρό 234*60					
728x90	Οριζόντιο Μεγάλο 728*90					
160x600	Κάθετο Μικρό 160*600					
300x600	Κάθετο Μεγάλο 300*600					

ΠΑΡΑΡΤΗΜΑ ΙΙ

```
package model1;

import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayDeque;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.Collection;
import java.util.Collections;
import java.util.Comparator;
import java.util.Currency;
import java.util.Date;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Hashtable;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import java.util.Map;
import java.util.PriorityQueue;
import java.util.Random;
import java.util.Set;
import java.util.SortedMap;
import java.util.SortedSet;
import java.util.Stack;
import java.util.Timer;
import java.util.TreeMap;
import java.util.TreeSet;
import java.util.Vector;
import java.awt.Color;
import java.awt.Font;
import com.anylogic.engine.connectivity.ResultSet;
import com.anylogic.engine.connectivity.Statement;
import com.anylogic.engine.elements.*;
import com.anylogic.engine.markup.Network;
import com.anylogic.engine.Position;
import com.anylogic.engine.markup.PedFlowStatistics;
import com.anylogic.engine.markup.DensityMap;

import static java.lang.Math.*;
import static com.anylogic.engine.UtilitiesArray.*;
import static com.anylogic.engine.UtilitiesCollection.*;
import static com.anylogic.engine.presentation.UtilitiesColor.*;
import static com.anylogic.engine.HyperArray.*;

import com.anylogic.engine.*;
import com.anylogic.engine.analysis.*;
import com.anylogic.engine.connectivity.*;
import com.anylogic.engine.database.*;
import com.anylogic.engine.gis.*;
import com.anylogic.engine.markup.*;
import com.anylogic.engine.routing.*;
```

```

import com.anylogic.engine.presentation.*;
import com.anylogic.engine.gui.*;

import java.awt.geom.Arc2D;

public class Main extends Agent
{
    // Parameters

    public
double   parameter_S;

    /**
     * Returns default value for parameter <code>parameter_S</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_S_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_S( double parameter_S ) {
        if (parameter_S == this.parameter_S) {
            return;
        }
        double _oldValue_xjal = this.parameter_S;
        this.parameter_S = parameter_S;
        onChange_parameter_S_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_S.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
     by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_S()</code> method instead.
     */
    protected void onChange_parameter_S() {
        onChange_parameter_S_xjal( parameter_S );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_S_xjal( double oldValue ) {
    }

    public
double   parameter_P;

    /**
     * Returns default value for parameter <code>parameter_P</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_P_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_P( double parameter_P ) {

```



```

    if (parameter_P == this.parameter_P) {
        return;
    }
    double _oldValue_xjal = this.parameter_P;
    this.parameter_P = parameter_P;
    onChange_parameter_P_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_P.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_P()</code> method instead.
 */
protected void onChange_parameter_P() {
    onChange_parameter_P_xjal( parameter_P );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_P_xjal( double oldValue ) {
}

public
double parameter_PC;

/**
 * Returns default value for parameter <code>parameter_PC</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_PC_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_PC( double parameter_PC ) {
    if (parameter_PC == this.parameter_PC) {
        return;
    }
    double _oldValue_xjal = this.parameter_PC;
    this.parameter_PC = parameter_PC;
    onChange_parameter_PC_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_PC.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_PC()</code> method instead.
 */
protected void onChange_parameter_PC() {
    onChange_parameter_PC_xjal( parameter_PC );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_PC_xjal( double oldValue ) {
}

```

```

public
double parameter_AN;

/**
 * Returns default value for parameter parameter_AN.
 * This method should not be called by user
 */
@AnyLogicInternalCodegenAPI
public double _parameter_AN_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_AN( double parameter_AN ) {
    if (parameter_AN == this.parameter_AN) {
        return;
    }
    double _oldValue_xjal = this.parameter_AN;
    this.parameter_AN = parameter_AN;
    onChange_parameter_AN_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_AN.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call set_parameter_AN() method instead.
 */
protected void onChange_parameter_AN() {
    onChange_parameter_AN_xjal( parameter_AN );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_AN_xjal( double oldValue ) {
}

public
double parameter_I;

/**
 * Returns default value for parameter parameter_I.
 * This method should not be called by user
 */
@AnyLogicInternalCodegenAPI
public double _parameter_I_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_I( double parameter_I ) {
    if (parameter_I == this.parameter_I) {
        return;
    }
    double _oldValue_xjal = this.parameter_I;
    this.parameter_I = parameter_I;
    onChange_parameter_I_xjal( _oldValue_xjal );
    onChange();
}

```

```

}

/**
 * Calls "On change" action for parameter parameter_I.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_I()</code> method instead.
 */
protected void onChange_parameter_I() {
    onChange_parameter_I_xjal( parameter_I );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_I_xjal( double oldValue ) {
}

public
double parameter_V;

/**
 * Returns default value for parameter <code>parameter_V</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_V_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_V( double parameter_V ) {
    if (parameter_V == this.parameter_V) {
        return;
    }
    double _oldValue_xjal = this.parameter_V;
    this.parameter_V = parameter_V;
    onChange_parameter_V_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_V.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_V()</code> method instead.
 */
protected void onChange_parameter_V() {
    onChange_parameter_V_xjal( parameter_V );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_V_xjal( double oldValue ) {
}

public
double parameter_G;

/**
 * Returns default value for parameter <code>parameter_G</code>.

```

```

    * <i>This method should not be called by user</i>
    */
@AnyLogicInternalCodegenAPI
public double _parameter_G_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_G( double parameter_G ) {
    if (parameter_G == this.parameter_G) {
        return;
    }
    double _oldValue_xjal = this.parameter_G;
    this.parameter_G = parameter_G;
    onChange_parameter_G_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_G.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_G()</code> method instead.
 */
protected void onChange_parameter_G() {
    onChange_parameter_G_xjal( parameter_G );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_G_xjal( double oldValue ) {
}

public
double parameter_C;

/**
 * Returns default value for parameter <code>parameter_C</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_C_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_C( double parameter_C ) {
    if (parameter_C == this.parameter_C) {
        return;
    }
    double _oldValue_xjal = this.parameter_C;
    this.parameter_C = parameter_C;
    onChange_parameter_C_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_C.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>

```

```

    * Please call <code>set_parameter_C()</code> method instead.
    */
    protected void onChange_parameter_C() {
        onChange_parameter_C_xjal( parameter_C );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_C_xjal( double oldValue ) {
    }

    public
    double parameter_A;

    /**
     * Returns default value for parameter <code>parameter_A</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_A_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_A( double parameter_A ) {
        if (parameter_A == this.parameter_A) {
            return;
        }
        double _oldValue_xjal = this.parameter_A;
        this.parameter_A = parameter_A;
        onChange_parameter_A_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_A.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
    by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_A()</code> method instead.
     */
    protected void onChange_parameter_A() {
        onChange_parameter_A_xjal( parameter_A );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_A_xjal( double oldValue ) {
    }

    @Override
    public void setParametersToDefaultValues() {
        super.setParametersToDefaultValues();
        parameter_S = _parameter_S_DefaultValue_xjal();
        parameter_P = _parameter_P_DefaultValue_xjal();
        parameter_PC = _parameter_PC_DefaultValue_xjal();
        parameter_AN = _parameter_AN_DefaultValue_xjal();
        parameter_I = _parameter_I_DefaultValue_xjal();
        parameter_V = _parameter_V_DefaultValue_xjal();
        parameter_G = _parameter_G_DefaultValue_xjal();
        parameter_C = _parameter_C_DefaultValue_xjal();
        parameter_A = _parameter_A_DefaultValue_xjal();
    }

```

```

}

@Override
public boolean setParameter(String _name_xjal, Object _value_xjal, boolean
_callOnChange_xjal) {
    switch ( _name_xjal ) {
        case "parameter_S":
            if ( _callOnChange_xjal ) {
                set_parameter_S( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_S = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_P":
            if ( _callOnChange_xjal ) {
                set_parameter_P( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_P = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_PC":
            if ( _callOnChange_xjal ) {
                set_parameter_PC( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_PC = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_AN":
            if ( _callOnChange_xjal ) {
                set_parameter_AN( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_AN = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_I":
            if ( _callOnChange_xjal ) {
                set_parameter_I( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_I = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_V":
            if ( _callOnChange_xjal ) {
                set_parameter_V( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_V = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_G":
            if ( _callOnChange_xjal ) {
                set_parameter_G( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_G = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_C":
            if ( _callOnChange_xjal ) {
                set_parameter_C( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_C = ((Number) _value_xjal).doubleValue();
            }
            return true;
    }
}

```

```

    case "parameter_A":
        if ( _callOnChange_xjal ) {
            set_parameter_A( ((Number) _value_xjal).doubleValue() );
        } else {
            parameter_A = ((Number) _value_xjal).doubleValue();
        }
        return true;
    default:
        return super.setParameter( _name_xjal, _value_xjal, _callOnChange_xjal );
}
}

@Override
public <T> T getParameter(String _name_xjal) {
    Object _result_xjal;
    switch ( _name_xjal ) {
        case "parameter_S": _result_xjal = parameter_S; break;
        case "parameter_P": _result_xjal = parameter_P; break;
        case "parameter_PC": _result_xjal = parameter_PC; break;
        case "parameter_AN": _result_xjal = parameter_AN; break;
        case "parameter_I": _result_xjal = parameter_I; break;
        case "parameter_V": _result_xjal = parameter_V; break;
        case "parameter_G": _result_xjal = parameter_G; break;
        case "parameter_C": _result_xjal = parameter_C; break;
        case "parameter_A": _result_xjal = parameter_A; break;
        default: _result_xjal = super.getParameter( _name_xjal ); break;
    }
    return (T) _result_xjal;
}

@AnyLogicInternalCodegenAPI
private static String[] _parameterNames_xjal;

@Override
public String[] getParameterNames() {
    String[] result = _parameterNames_xjal;
    if (result == null) {
        List<String> list = new ArrayList<>( Arrays.asList(
super.getParameterNames() ) );
        list.add( "parameter_S" );
        list.add( "parameter_P" );
        list.add( "parameter_PC" );
        list.add( "parameter_AN" );
        list.add( "parameter_I" );
        list.add( "parameter_V" );
        list.add( "parameter_G" );
        list.add( "parameter_C" );
        list.add( "parameter_A" );
        result = list.toArray( new String[ list.size() ] );
        _parameterNames_xjal = result;
    }
    return result;
}

// Dynamic (Flow/Auxiliary/Stock) Variables

public double RATE_S;

public double RATE_I;

public double RATE_V;

```

```
public double RATE_C;
public double RATE_G;
public double RATE_P;
public double RATE_PC;
public double RATE_AN;
public double LOST_S;
public double flow;
public double LOST_P;
public double flow2;
public double LOST_PC;
public double flow3;
public double LOST_AN;
public double flow4;
public double LOST_I;
public double flow5;
public double LOST_V;
public double flow6;
public double LOST_G;
public double flow7;
public double LOST_C;
public double flow8;
public double RETURNED;
public double RATE_A;
public double LOST_A;
public double flow1;
public double CR;
public double STATIC;
public double INTERACTIVE;
public double POPUP;
public double PICTURES;
public double GAMES;
```



```

public double VIDEO;

public double COLORS;

public double ANNOUNCE;

public double SUCCESS;

public double ANIMATED;
/**
 * Writes model variables into given arrays
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    _d_xjal[ _d_idx_xjal + 0 ] = CR;
    _d_xjal[ _d_idx_xjal + 1 ] = STATIC;
    _d_xjal[ _d_idx_xjal + 2 ] = INTERACTIVE;
    _d_xjal[ _d_idx_xjal + 3 ] = POPUP;
    _d_xjal[ _d_idx_xjal + 4 ] = PICTURES;
    _d_xjal[ _d_idx_xjal + 5 ] = GAMES;
    _d_xjal[ _d_idx_xjal + 6 ] = VIDEO;
    _d_xjal[ _d_idx_xjal + 7 ] = COLORS;
    _d_xjal[ _d_idx_xjal + 8 ] = ANNOUNCE;
    _d_xjal[ _d_idx_xjal + 9 ] = SUCCESS;
    _d_xjal[ _d_idx_xjal + 10 ] = ANIMATED;
    _d_idx_xjal += 11;
    super.getPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * Writes given arrays to model variables
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    CR = _d_xjal[ _d_idx_xjal + 0 ];
    STATIC = _d_xjal[ _d_idx_xjal + 1 ];
    INTERACTIVE = _d_xjal[ _d_idx_xjal + 2 ];
    POPUP = _d_xjal[ _d_idx_xjal + 3 ];
    PICTURES = _d_xjal[ _d_idx_xjal + 4 ];
    GAMES = _d_xjal[ _d_idx_xjal + 5 ];
    VIDEO = _d_xjal[ _d_idx_xjal + 6 ];
    COLORS = _d_xjal[ _d_idx_xjal + 7 ];
    ANNOUNCE = _d_xjal[ _d_idx_xjal + 8 ];
    SUCCESS = _d_xjal[ _d_idx_xjal + 9 ];
    ANIMATED = _d_xjal[ _d_idx_xjal + 10 ];
    _d_idx_xjal += 11;
    super.putPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * This function gets phase vector from initial conditions (is used while solving
algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    _a_xjal[ _a_idx_xjal + 0 ] = LOST_S;
    _a_xjal[ _a_idx_xjal + 1 ] = flow;
    _a_xjal[ _a_idx_xjal + 2 ] = LOST_P;
    _a_xjal[ _a_idx_xjal + 3 ] = flow2;
    _a_xjal[ _a_idx_xjal + 4 ] = LOST_PC;
}

```

```

    _a_xjal[ _a_idx_xjal + 5 ] = flow3;
    _a_xjal[ _a_idx_xjal + 6 ] = LOST_AN;
    _a_xjal[ _a_idx_xjal + 7 ] = flow4;
    _a_xjal[ _a_idx_xjal + 8 ] = LOST_V;
    _a_xjal[ _a_idx_xjal + 9 ] = flow6;
    _a_xjal[ _a_idx_xjal + 10 ] = LOST_G;
    _a_xjal[ _a_idx_xjal + 11 ] = flow7;
    _a_xjal[ _a_idx_xjal + 12 ] = LOST_C;
    _a_xjal[ _a_idx_xjal + 13 ] = flow8;
    _a_xjal[ _a_idx_xjal + 14 ] = LOST_A;
    _a_xjal[ _a_idx_xjal + 15 ] = flow1;
    _a_idx_xjal += 16;
    super.getPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

/**
 * This function places phase vector into initial conditions (is used while solving
 algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    LOST_S = _a_xjal[ _a_idx_xjal + 0 ];
    flow = _a_xjal[ _a_idx_xjal + 1 ];
    LOST_P = _a_xjal[ _a_idx_xjal + 2 ];
    flow2 = _a_xjal[ _a_idx_xjal + 3 ];
    LOST_PC = _a_xjal[ _a_idx_xjal + 4 ];
    flow3 = _a_xjal[ _a_idx_xjal + 5 ];
    LOST_AN = _a_xjal[ _a_idx_xjal + 6 ];
    flow4 = _a_xjal[ _a_idx_xjal + 7 ];
    LOST_V = _a_xjal[ _a_idx_xjal + 8 ];
    flow6 = _a_xjal[ _a_idx_xjal + 9 ];
    LOST_G = _a_xjal[ _a_idx_xjal + 10 ];
    flow7 = _a_xjal[ _a_idx_xjal + 11 ];
    LOST_C = _a_xjal[ _a_idx_xjal + 12 ];
    flow8 = _a_xjal[ _a_idx_xjal + 13 ];
    LOST_A = _a_xjal[ _a_idx_xjal + 14 ];
    flow1 = _a_xjal[ _a_idx_xjal + 15 ];
    _a_idx_xjal += 16;
    super.putPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

@AnyLogicInternalCodegenAPI
public void assignInitialConditions_xjal() {
    super.assignInitialConditions_xjal();
    _assign_CR_Formula_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_INTERACTIVE_Formula_xjal();
    _assign_SUCCESS_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_STATIC_Formula_xjal();
    _assign_POPUP_Formula_xjal();
}

```

```

    _assign_PICTURES_Formula_xjal();
    _assign_GAMES_Formula_xjal();
    _assign_VIDEO_Formula_xjal();
    _assign_COLORS_Formula_xjal();
    _assign_ANNOUNCE_Formula_xjal();
    _assign_ANIMATED_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void setupInitialConditions_xjal(Class<?> callerClass) {
    if (callerClass != Main.class) {
        return;
    }
    if (getInitialAlgebraicFlatEquationsCount_xjal() > 0) {
        SDIntegrationManager integrationManagerForInitialConditions = new
SDIntegrationManager( 0, getInitialAlgebraicFlatEquationsCount_xjal(),
getInitialFormulaFlatEquationsCount_xjal() );
        integrationManagerForInitialConditions.doStep( this, 0, 0.1, true );
    } else {
        assignInitialConditions_xjal();
    }
}

@AnyLogicInternalCodegenAPI
public void _assign_CR_Formula_xjal() {
    CR =
100
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_S_Formula_xjal() {
    RATE_S =
CR * 0.084
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_V_Formula_xjal() {
    RATE_V =
CR * 0.12
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_C_Formula_xjal() {
    RATE_C =
CR * 0.11
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_G_Formula_xjal() {
    RATE_G =
CR * 0.093
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_P_Formula_xjal() {
    RATE_P =
CR * 0.013
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_PC_Formula_xjal() {
    RATE_PC =

```

```

CR * 0.10
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_AN_Formula_xjal() {
    RATE_AN =
CR * 0.075
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_A_Formula_xjal() {
    RATE_A =
CR * 0.149
;
}
@AnyLogicInternalCodegenAPI
public void _assign_INTERACTIVE_Formula_xjal() {
    INTERACTIVE =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_SUCCESS_Formula_xjal() {
    SUCCESS =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_I_Formula_xjal() {
    RATE_I =
CR * 0.139
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_I_Formula_xjal() {
    LOST_I =
INTERACTIVE * 0.25 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow5_Formula_xjal() {
    flow5 =
INTERACTIVE * 0.75 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RETURNED_Formula_xjal() {
    RETURNED =
SUCCESS * 1
;
}
@AnyLogicInternalCodegenAPI
public void _assign_STATIC_Formula_xjal() {
    STATIC =
RATE_S - LOST_S - flow + 0 + parameter_S
;
}
@AnyLogicInternalCodegenAPI
public void _assign_POPUP_Formula_xjal() {
    POPUP =
RATE_P - flow2 - LOST_P + 0 + parameter_P
;
}

```

```

    }
    @AnyLogicInternalCodegenAPI
    public void _assign_PICTURES_Formula_xjal() {
        PICTURES =
RATE_PC - flow3 - LOST_PC + 0 + parameter_PC
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_GAMES_Formula_xjal() {
        GAMES =
RATE_G - flow7 - LOST_G + 0 + parameter_G
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_VIDEO_Formula_xjal() {
        VIDEO =
RATE_V - flow6 - LOST_V + 0 + parameter_V
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_COLORS_Formula_xjal() {
        COLORS =
RATE_C - LOST_C - flow8 + 0 + parameter_C
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANNOUNCE_Formula_xjal() {
        ANNOUNCE =
RATE_AN - flow4 - LOST_AN + 0 + parameter_AN
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANIMATED_Formula_xjal() {
        ANIMATED =
RATE_A - flow1 - LOST_A + 0 + parameter_A
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_S_Formula_xjal() {
        LOST_S =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow_Formula_xjal() {
        flow =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_P_Formula_xjal() {
        LOST_P =
POPUP * 0.3 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow2_Formula_xjal() {
        flow2 =
POPUP * 0.7 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI

```

```

    public void _assign_LOST_PC_Formula_xjal() {
        LOST_PC =
PICTURES * 0.45 + parameter_PC
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow3_Formula_xjal() {
        flow3 =
PICTURES * 0.55 + parameter_PC
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_AN_Formula_xjal() {
        LOST_AN =
ANNOUNCE * 0.6 + parameter_AN
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow4_Formula_xjal() {
        flow4 =
ANNOUNCE * 0.40 + parameter_AN
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_V_Formula_xjal() {
        LOST_V =
VIDEO * 0.35 + parameter_V
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow6_Formula_xjal() {
        flow6 =
VIDEO * 0.75
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_G_Formula_xjal() {
        LOST_G =
GAMES * 0.5 + parameter_G
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow7_Formula_xjal() {
        flow7 =
GAMES * 0.5 + parameter_G
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_C_Formula_xjal() {
        LOST_C =
COLORS * 0.4
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow8_Formula_xjal() {
        flow8 =
COLORS * 0.6 + parameter_C
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_A_Formula_xjal() {
        LOST_A =

```

```

ANIMATED * 0.20 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow1_Formula_xjal() {
    flow1 =
ANIMATED * 0.8 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void formulasExecute_xjal() {
    super.formulasExecute_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_LOST_S_Formula_xjal();
    _assign_flow_Formula_xjal();
    _assign_LOST_P_Formula_xjal();
    _assign_flow2_Formula_xjal();
    _assign_LOST_PC_Formula_xjal();
    _assign_flow3_Formula_xjal();
    _assign_LOST_AN_Formula_xjal();
    _assign_flow4_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_LOST_V_Formula_xjal();
    _assign_flow6_Formula_xjal();
    _assign_LOST_G_Formula_xjal();
    _assign_flow7_Formula_xjal();
    _assign_LOST_C_Formula_xjal();
    _assign_flow8_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_LOST_A_Formula_xjal();
    _assign_flow1_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void getRightPart_xjal( double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal ) {
    _get_CR_Derivative_xjal( _d_xjal, _d_idx_xjal + 0 );
    _get_STATIC_Derivative_xjal( _d_xjal, _d_idx_xjal + 1 );
    _get_INTERACTIVE_Derivative_xjal( _d_xjal, _d_idx_xjal + 2 );
    _get_POPUP_Derivative_xjal( _d_xjal, _d_idx_xjal + 3 );
    _get_PICTURES_Derivative_xjal( _d_xjal, _d_idx_xjal + 4 );
    _get_GAMES_Derivative_xjal( _d_xjal, _d_idx_xjal + 5 );
    _get_VIDEO_Derivative_xjal( _d_xjal, _d_idx_xjal + 6 );
    _get_COLORS_Derivative_xjal( _d_xjal, _d_idx_xjal + 7 );
    _get_ANNOUNCE_Derivative_xjal( _d_xjal, _d_idx_xjal + 8 );
    _get_SUCCESS_Derivative_xjal( _d_xjal, _d_idx_xjal + 9 );
    _get_ANIMATED_Derivative_xjal( _d_xjal, _d_idx_xjal + 10 );
    _d_idx_xjal += 11;
    super.getRightPart_xjal( _d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void getRightPartForInitialConditions_xjal( double[] _a_xjal, int

```

```

_a_idx_xjal ) {
    _get_LOST_S_Equation_xjal( _a_xjal, _a_idx_xjal + 0 );
    _get_flow_Equation_xjal( _a_xjal, _a_idx_xjal + 1 );
    _get_LOST_P_Equation_xjal( _a_xjal, _a_idx_xjal + 2 );
    _get_flow2_Equation_xjal( _a_xjal, _a_idx_xjal + 3 );
    _get_LOST_PC_Equation_xjal( _a_xjal, _a_idx_xjal + 4 );
    _get_flow3_Equation_xjal( _a_xjal, _a_idx_xjal + 5 );
    _get_LOST_AN_Equation_xjal( _a_xjal, _a_idx_xjal + 6 );
    _get_flow4_Equation_xjal( _a_xjal, _a_idx_xjal + 7 );
    _get_LOST_V_Equation_xjal( _a_xjal, _a_idx_xjal + 8 );
    _get_flow6_Equation_xjal( _a_xjal, _a_idx_xjal + 9 );
    _get_LOST_G_Equation_xjal( _a_xjal, _a_idx_xjal + 10 );
    _get_flow7_Equation_xjal( _a_xjal, _a_idx_xjal + 11 );
    _get_LOST_C_Equation_xjal( _a_xjal, _a_idx_xjal + 12 );
    _get_flow8_Equation_xjal( _a_xjal, _a_idx_xjal + 13 );
    _get_LOST_A_Equation_xjal( _a_xjal, _a_idx_xjal + 14 );
    _get_flow1_Equation_xjal( _a_xjal, _a_idx_xjal + 15 );
    _a_idx_xjal += 16;
    super.getRightPartForInitialConditions_xjal( _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void _get_CR_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RETURNED - RATE_S - RATE_I- RATE_V- RATE_C- RATE_G- RATE_P- RATE_PC- RATE_AN- RATE_A
;
}
@AnyLogicInternalCodegenAPI
public void _get_STATIC_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_S - LOST_S - flow
;
}
@AnyLogicInternalCodegenAPI
public void _get_INTERACTIVE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_I - LOST_I - flow5
;
}
@AnyLogicInternalCodegenAPI
public void _get_POPUP_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_P - LOST_P - flow2
;
}
@AnyLogicInternalCodegenAPI
public void _get_PICTURES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_PC - LOST_PC - flow3
;
}
@AnyLogicInternalCodegenAPI
public void _get_GAMES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_G - LOST_G - flow7
;
}
@AnyLogicInternalCodegenAPI
public void _get_VIDEO_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_V - LOST_V - flow6
;
}

```



```

}
@AnyLogicInternalCodegenAPI
public void _get_COLORS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_C - LOST_C - flow8
;
}
@AnyLogicInternalCodegenAPI
public void _get_ANNOUNCE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_AN - LOST_AN - flow4
;
}
@AnyLogicInternalCodegenAPI
public void _get_SUCCESS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+flow + flow2 + flow3+ flow4+ flow5+ flow6+ flow7+ flow8+ flow1 - RETURNED
;
}
@AnyLogicInternalCodegenAPI
public void _get_ANIMATED_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_A - LOST_A - flow1
;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_S_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- LOST_S;
}
@AnyLogicInternalCodegenAPI
public void _get_flow_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- flow;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_P_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
POPUP * 0.3 + parameter_P
- LOST_P;
}
@AnyLogicInternalCodegenAPI
public void _get_flow2_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
POPUP * 0.7 + parameter_P
- flow2;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_PC_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.45 + parameter_PC
- LOST_PC;
}
@AnyLogicInternalCodegenAPI
public void _get_flow3_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.55 + parameter_PC
- flow3;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _get_LOST_AN_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.6 + parameter_AN
- LOST_AN;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow4_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.40 + parameter_AN
- flow4;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_V_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.35 + parameter_V
- LOST_V;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow6_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.75
- flow6;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_G_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- LOST_G;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow7_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- flow7;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_C_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
COLORS * 0.4
- LOST_C;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow8_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
COLORS * 0.6 + parameter_C
- flow8;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_A_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.20 + parameter_A
- LOST_A;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow1_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.8 + parameter_A
- flow1;
    }
    @AnyLogicInternalCodegenAPI
    protected SDIntegrationManager integrationManager_xjal = null;

```

```

@AnyLogicInternalCodegenAPI
public SDIntegrationManager getIntegrationManager_xjal() {
    if (integrationManager_xjal == null) {
        integrationManager_xjal = new
SDIntegrationManager( getDifferentialFlatEquationsCount_xjal(),
getRuntimeAlgebraicFlatEquationsCount_xjal(),
getRuntimeFormulaFlatEquationsCount_xjal() );
    }
    return integrationManager_xjal;
}

@Override
@AnyLogicInternalCodegenAPI
public int getDifferentialFlatEquationsCount_xjal() {
    return super.getDifferentialFlatEquationsCount_xjal() + 11;
}

@Override
@AnyLogicInternalCodegenAPI
public int getRuntimeFormulaFlatEquationsCount_xjal() {
    return super.getRuntimeFormulaFlatEquationsCount_xjal() + 28;
}

@Override
@AnyLogicInternalCodegenAPI
public int getInitialAlgebraicFlatEquationsCount_xjal() {
    return super.getInitialAlgebraicFlatEquationsCount_xjal() + 16;
}

@Override
@AnyLogicInternalCodegenAPI
public int getInitialFormulaFlatEquationsCount_xjal() {
    return super.getInitialFormulaFlatEquationsCount_xjal() + 23;
}

@AnyLogicInternalCodegenAPI
private static Map<String, IElementDescriptor> elementDescriptors_xjal =
createElementDescriptors( Main.class );

@AnyLogicInternalCodegenAPI
@Override
public Map<String, IElementDescriptor> getElementDescriptors() {
    return elementDescriptors_xjal;
}
@AnyLogicCustomProposalPriority(type =
AnyLogicCustomProposalPriority.Type.STATIC_ELEMENT)
public static final Scale scale = new Scale( 10.0 );

@Override
public Scale getScale() {
    return scale;
}

// Events

@AnyLogicInternalCodegenAPI
public EventTimeout _chart_autoUpdateEvent_xjal = new EventTimeout(this);
@AnyLogicInternalCodegenAPI
public EventTimeout _autoCreatedDS_xjal = new EventTimeout(this);

@Override

```

```

@AnyLogicInternalCodegenAPI
public String getNameOf( EventTimeout _e ) {
    if( _e == _chart_autoUpdateEvent_xjal ) return "chart auto update event";
    if( _e == _autoCreatedDS_xjal ) return "Auto-created DataSets auto update event";
    return super.getNameOf( _e );
}

@Override
@AnyLogicInternalCodegenAPI
public EventTimeout.Mode getModeOf( EventTimeout _e ) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
    if ( _e == _autoCreatedDS_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
    return super.getModeOf( _e );
}

@Override
@AnyLogicInternalCodegenAPI
public double getFirstOccurrenceTime( EventTimeout _e ) {
    double _t;
    if ( _e == _chart_autoUpdateEvent_xjal ) {
0
;
        _t = toModelTime( _t, SECOND );
        return _t;
    }
    if ( _e == _autoCreatedDS_xjal ) {
0
;
        _t = toModelTime( _t, SECOND );
        return _t;
    }
    return super.getFirstOccurrenceTime( _e );
}

@Override
@AnyLogicInternalCodegenAPI
public double evaluateTimeoutOf( EventTimeout _e ) {
    double _t;
    if( _e == _chart_autoUpdateEvent_xjal) {
1
;
        _t = toModelTime( _t, SECOND );
        return _t;
    }
    if( _e == _autoCreatedDS_xjal) {
1
;
        _t = toModelTime( _t, SECOND );
        return _t;
    }
    return super.evaluateTimeoutOf( _e );
}

@Override
@AnyLogicInternalCodegenAPI
public void executeActionOf( EventTimeout _e ) {
    if ( _e == _chart_autoUpdateEvent_xjal ) {
        chart.updateData();
    }
}

```

```

    return;
}
if ( _e == _autoCreatedDS_xjal ) {
    _ds_RATE_S.update();
    _ds_RATE_I.update();
    _ds_RATE_V.update();
    _ds_RATE_C.update();
    _ds_RATE_G.update();
    _ds_RATE_P.update();
    _ds_RATE_PC.update();
    _ds_RATE_AN.update();
    _ds_LOST_S.update();
    _ds_flow.update();
    _ds_LOST_P.update();
    _ds_flow2.update();
    _ds_LOST_PC.update();
    _ds_flow3.update();
    _ds_LOST_AN.update();
    _ds_flow4.update();
    _ds_LOST_I.update();
    _ds_flow5.update();
    _ds_LOST_V.update();
    _ds_flow6.update();
    _ds_LOST_G.update();
    _ds_flow7.update();
    _ds_LOST_C.update();
    _ds_flow8.update();
    _ds_RETURNED.update();
    _ds_RATE_A.update();
    _ds_LOST_A.update();
    _ds_flow1.update();
    _ds_CR.update();
    _ds_STATIC.update();
    _ds_INTERACTIVE.update();
    _ds_POPUP.update();
    _ds_PICTURES.update();
    _ds_GAMES.update();
    _ds_VIDEO.update();
    _ds_COLORS.update();
    _ds_ANNOUNCE.update();
    _ds_SUCCESS.update();
    _ds_ANIMATED.update();
    return;
}
super.executeActionOf( _e );
}

```

```

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _STATECHART_COUNT_xjal = 0;

```

```

/**
 * Auto-created data set(s) for RATE_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {

```

```

        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_P
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_P = new DataSet( 100, new DataUpdater_xjal() {

```

```

    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_AN );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_P

```

```

*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow2
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow2 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow2 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow3
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow3 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow3 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_AN );
        _lastUpdateTime = time();
    }
}

```



```

} );
/**
 * Auto-created data set(s) for flow4
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow4 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow4 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow5
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow5 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow5 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow6
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow6 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }

```

```

        _d.add( time(), Main.this.flow6 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow7
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow7 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow7 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow8
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow8 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow8 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RETURNED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RETURNED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;

```

```

@Override
public void update( DataSet _d ) {
    if ( time() == _lastUpdateTime ) { return; }
    _d.add( time(), Main.this.RETURNED );
    _lastUpdateTime = time();
}
} );
/**
 * Auto-created data set(s) for RATE_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow1
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow1 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow1 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for CR
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_CR = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.CR );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for STATIC
 */

```

```

@AnyLogicInternalCodegenAPI
public DataSet _ds_STATIC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.STATIC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for INTERACTIVE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_INTERACTIVE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.INTERACTIVE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for POPUP
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_POPUP = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.POPUP );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for PICTURES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_PICTURES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.PICTURES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for GAMES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_GAMES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.GAMES );
        _lastUpdateTime = time();
    }
} );

```

```

/**
 * Auto-created data set(s) for VIDEO
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_VIDEO = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.VIDEO );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for COLORS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_COLORS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.COLORS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANNOUNCE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANNOUNCE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANNOUNCE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for SUCCESS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_SUCCESS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.SUCCESS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANIMATED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANIMATED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANIMATED );
    }
} );

```

```

        _lastUpdateTime = time();
    }
} );
// View areas
public ViewArea _origin_VA = new ViewArea( this, "[Origin]", 0, 0, 1000.0, 600.0 );
@Override
@AnyLogicInternalCodegenAPI
public int getViewAreas(Map<String, ViewArea> _output) {
    if ( _output != null ) {
        _output.put( "_origin_VA", this._origin_VA );
    }
    return 1 + super.getViewAreas( _output );
}
@AnyLogicInternalCodegenAPI
protected static final int _chart = 1;

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _SHAPE_NEXT_ID_xjal = 2;

@AnyLogicInternalCodegenAPI
public boolean isPublicPresentationDefined() {
    return true;
}

@AnyLogicInternalCodegenAPI
public boolean isEmbeddedAgentPresentationVisible( Agent _a ) {
    return super.isEmbeddedAgentPresentationVisible( _a );
}

/**
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
private double _chart_DataItem0Value() {
    return
ANIMATED
;
}

/**
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
private double _chart_DataItem1Value() {
    return
STATIC
;
}

/**
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
private double _chart_DataItem2Value() {
    return
POPUP
;
}

/**

```

```

    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem3Value() {
        return
INTERACTIVE
;
    }

    /**
    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem4Value() {
        return
PICTURES
;
    }

    /**
    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem5Value() {
        return
GAMES
;
    }

    /**
    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem6Value() {
        return
VIDEO
;
    }

    /**
    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem7Value() {
        return
ANNOUNCE
;
    }

    /**
    * <i>This method should not be called by user</i>
    */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem8Value() {
        return
COLORS
;
    }
    }
    protected BarChart chart;
    @AnyLogicInternalCodegenAPI
    private void _createPersistentElementsBP0_xjal() {
    }

```

```

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsAP0_xjal() {
    {
        List<DataItem> _items = new ArrayList<DataItem>( 9 );
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem0Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem1Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem2Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem3Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem4Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem5Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem6Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem7Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem8Value() );
            }
        });
        List<String> _titles = new ArrayList<String>( 9 );
        _titles.add( "ANIMATED" );
        _titles.add( "STATIC" );
    }
}

```



```

_titles.add( "POPOP" );
_titles.add( "INTERACTIVE" );
_titles.add( "PICTURES" );
_titles.add( "GAMES" );
_titles.add( "VIDEO" );
_titles.add( "ANNOUNCE" );
_titles.add( "COLORS" );
List<Color> _colors = new ArrayList<Color>( 9 );
_colors.add( limeGreen );
_colors.add( slateGray );
_colors.add( darkMagenta );
_colors.add( orange );
_colors.add( darkKhaki );
_colors.add( mediumTurquoise );
_colors.add( lightSlateBlue );
_colors.add( deepPink );
_colors.add( darkOrange );
chart = new BarChart(
    Main.this, true, 20.0, 700.0,
    1420.0, 580.0,
    null, null,
    50.0, 20.0,
    1330.0, 460.0, white, black, black,
    80.0, Chart.SOUTH,
    Chart.NORTH, Chart.SCALE_AUTO,
    0, 0, 0.8,
    Chart.GRID_DEFAULT,
    darkGray, darkGray, _items, _titles, _colors );
}
}

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsBS0_xjal() {
}

// Static initialization of persistent elements
{
_createPersistentElementsBP0_xjal();
}
protected ShapeTopLevelPresentationGroup presentation;
protected ShapeModelElementsGroup icon;

@Override
@AnyLogicInternalCodegenAPI
public ShapeTopLevelPresentationGroup getPresentationShape() {
return presentation;
}

@Override
@AnyLogicInternalCodegenAPI
public ShapeModelElementsGroup getModelElementsShape() {
return icon;
}

/**
 * Constructor
 */
public Main( Engine engine, Agent owner, AgentList<? extends Main> ownerPopulation )
{
super( engine, owner, ownerPopulation );
}

```

```

    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
public void onOwnerChanged_xjal() {
    super.onOwnerChanged_xjal();
    setupReferences_xjal();
}

@AnyLogicInternalCodegenAPI
public void instantiateBaseStructure_xjal() {
    super.instantiateBaseStructure_xjal();
    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
private void instantiateBaseStructureThis_xjal() {
    setupReferences_xjal();
    // Registering in Engine continuous part
    getEngine().registerAgentWithEquations( this );
}

@AnyLogicInternalCodegenAPI
private void setupReferences_xjal() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main( double parameter_S, double parameter_P, double parameter_PC, double
parameter_AN, double parameter_I, double parameter_V, double parameter_G, double
parameter_C, double parameter_A ) {
    markParametersAreSet();
    this.parameter_S = parameter_S;
    this.parameter_P = parameter_P;
    this.parameter_PC = parameter_PC;
    this.parameter_AN = parameter_AN;
    this.parameter_I = parameter_I;
    this.parameter_V = parameter_V;
    this.parameter_G = parameter_G;
    this.parameter_C = parameter_C;
    this.parameter_A = parameter_A;
}

@Override
@AnyLogicInternalCodegenAPI
public void doCreate() {
    super.doCreate();
    // Assigning initial values for plain variables
    setupPlainVariables_Main_xjal();
    // Dynamic initialization of persistent elements
    _createPersistentElementsAP0_xjal();
    presentation = new ShapeTopLevelPresentationGroup( Main.this, true, 0, 0, 0, 0 ,
chart );
}

```

```

    icon = new ShapeModelElementsGroup( Main.this, getElementProperty(
"model1.Main.icon", IElementDescriptor.MODEL_ELEMENT_DESCRIPTOR ) );
    icon.setIconOffsets( 0.0, 0.0 );
        // Port connectors with non-replicated objects
        // Creating replicated embedded objects
        setupInitialConditions_xjal( Main.class );
        // Dynamic initialization of persistent elements
        _createPersistentElementsBS0_xjal();
    }

@Override
@AnyLogicInternalCodegenAPI
public void doStart() {
    super.doStart();
    _chart_autoUpdateEvent_xjal.start();
    _autoCreatedDS_xjal.start();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
public void setupPlainVariables_xjal() {
    setupPlainVariables_Main_xjal();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
private void setupPlainVariables_Main_xjal() {

// User API -----
@AnyLogicInternalCodegenAPI
public static LinkToAgentAnimationSettings _connections_commonAnimationSettings_xjal
= new LinkToAgentAnimationSettingsImpl( false, black, 1.0, LINE_STYLE_SOLID,
ARROW_NONE, 0.0 );

public LinkToAgentCollection<Agent, Agent> connections = new
LinkToAgentStandardImpl<Agent, Agent>(this, _connections_commonAnimationSettings_xjal);
@Override
public LinkToAgentCollection<? extends Agent, ? extends Agent>
getLinkToAgentStandard_xjal() {
    return connections;
}

@AnyLogicInternalCodegenAPI
public void drawLinksToAgents(boolean _underAgents_xjal, LinkToAgentAnimator
_ animator_xjal) {
    super.drawLinksToAgents(_underAgents_xjal, _animator_xjal);
    if ( _underAgents_xjal ) {
        _animator_xjal.drawLink( this, connections, true, true );
    }
}
}

```

```

public AgentList<? extends Main> getPopulation() {
    return (AgentList<? extends Main>) super.getPopulation();
}

public List<? extends Main> agentsInRange( double distance ) {
    return (List<? extends Main>) super.agentsInRange( distance );
}

@Override
@AnyLogicInternalCodegenAPI
public boolean isLoggingToDB(EventOriginator _e) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return false;
    if ( _e == _autoCreatedDS_xjal ) return false;
    return super.isLoggingToDB( _e );
}

@AnyLogicInternalCodegenAPI
public void onDestroy() {
    _chart_autoUpdateEvent_xjal.onDestroy();
    _autoCreatedDS_xjal.onDestroy();
    // Unregistering in Engine continuous part
    getEngine().unregisterAgentWithEquations( this );
    _ds_RATE_S.destroyUpdater_xjal();
    _ds_RATE_I.destroyUpdater_xjal();
    _ds_RATE_V.destroyUpdater_xjal();
    _ds_RATE_C.destroyUpdater_xjal();
    _ds_RATE_G.destroyUpdater_xjal();
    _ds_RATE_P.destroyUpdater_xjal();
    _ds_RATE_PC.destroyUpdater_xjal();
    _ds_RATE_AN.destroyUpdater_xjal();
    _ds_LOST_S.destroyUpdater_xjal();
    _ds_flow.destroyUpdater_xjal();
    _ds_LOST_P.destroyUpdater_xjal();
    _ds_flow2.destroyUpdater_xjal();
    _ds_LOST_PC.destroyUpdater_xjal();
    _ds_flow3.destroyUpdater_xjal();
    _ds_LOST_AN.destroyUpdater_xjal();
    _ds_flow4.destroyUpdater_xjal();
    _ds_LOST_I.destroyUpdater_xjal();
    _ds_flow5.destroyUpdater_xjal();
    _ds_LOST_V.destroyUpdater_xjal();
    _ds_flow6.destroyUpdater_xjal();
    _ds_LOST_G.destroyUpdater_xjal();
    _ds_flow7.destroyUpdater_xjal();
    _ds_LOST_C.destroyUpdater_xjal();
    _ds_flow8.destroyUpdater_xjal();
    _ds_RETURNED.destroyUpdater_xjal();
    _ds_RATE_A.destroyUpdater_xjal();
    _ds_LOST_A.destroyUpdater_xjal();
    _ds_flow1.destroyUpdater_xjal();
    _ds_CR.destroyUpdater_xjal();
    _ds_STATIC.destroyUpdater_xjal();
    _ds_INTERACTIVE.destroyUpdater_xjal();
    _ds_POPUP.destroyUpdater_xjal();
    _ds_PICTURES.destroyUpdater_xjal();
    _ds_GAMES.destroyUpdater_xjal();
    _ds_VIDEO.destroyUpdater_xjal();
    _ds_COLORS.destroyUpdater_xjal();
    _ds_ANNOUNCE.destroyUpdater_xjal();
    _ds_SUCCESS.destroyUpdater_xjal();
    _ds_ANIMATED.destroyUpdater_xjal();
    super.onDestroy();
}

```

```

package model1;

import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayDeque;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.Collection;
import java.util.Collections;
import java.util.Comparator;
import java.util.Currency;
import java.util.Date;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Hashtable;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import java.util.Map;
import java.util.PriorityQueue;
import java.util.Random;
import java.util.Set;
import java.util.SortedMap;
import java.util.SortedSet;
import java.util.Stack;
import java.util.Timer;
import java.util.TreeMap;
import java.util.TreeSet;
import java.util.Vector;
import java.awt.Color;
import java.awt.Font;
import com.anylogic.engine.connectivity.ResultSet;
import com.anylogic.engine.connectivity.Statement;
import com.anylogic.engine.elements.*;
import com.anylogic.engine.markup.Network;
import com.anylogic.engine.Position;
import com.anylogic.engine.markup.PedFlowStatistics;
import com.anylogic.engine.markup.DensityMap;

import static java.lang.Math.*;
import static com.anylogic.engine.UtilitiesArray.*;
import static com.anylogic.engine.UtilitiesCollection.*;
import static com.anylogic.engine.presentation.UtilitiesColor.*;
import static com.anylogic.engine.HyperArray.*;

import com.anylogic.engine.*;
import com.anylogic.engine.analysis.*;
import com.anylogic.engine.connectivity.*;
import com.anylogic.engine.database.*;
import com.anylogic.engine.gis.*;
import com.anylogic.engine.markup.*;
import com.anylogic.engine.routing.*;
import com.anylogic.engine.presentation.*;
import com.anylogic.engine.gui.*;

```

```

import java.awt.geom.Arc2D;

public class Main extends Agent
{
    // Parameters

    public
double   parameter_S;

    /**
     * Returns default value for parameter <code>parameter_S</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_S_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_S( double parameter_S ) {
        if (parameter_S == this.parameter_S) {
            return;
        }
        double _oldValue_xjal = this.parameter_S;
        this.parameter_S = parameter_S;
        onChange_parameter_S_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_S.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_S()</code> method instead.
     */
    protected void onChange_parameter_S() {
        onChange_parameter_S_xjal( parameter_S );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_S_xjal( double oldValue ) {

    }

    public
double   parameter_P;

    /**
     * Returns default value for parameter <code>parameter_P</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_P_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_P( double parameter_P ) {
        if (parameter_P == this.parameter_P) {
            return;
        }
    }
}

```

```

    }
    double _oldValue_xjal = this.parameter_P;
    this.parameter_P = parameter_P;
    onChange_parameter_P_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_P.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_P()</code> method instead.
 */
protected void onChange_parameter_P() {
    onChange_parameter_P_xjal( parameter_P );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_P_xjal( double oldValue ) {
}

public
double parameter_PC;

/**
 * Returns default value for parameter <code>parameter_PC</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_PC_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_PC( double parameter_PC ) {
    if (parameter_PC == this.parameter_PC) {
        return;
    }
    double _oldValue_xjal = this.parameter_PC;
    this.parameter_PC = parameter_PC;
    onChange_parameter_PC_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_PC.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_PC()</code> method instead.
 */
protected void onChange_parameter_PC() {
    onChange_parameter_PC_xjal( parameter_PC );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_PC_xjal( double oldValue ) {
}

```

```

public
double parameter_AN;

/**
 * Returns default value for parameter <code>parameter_AN</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_AN_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_AN( double parameter_AN ) {
    if (parameter_AN == this.parameter_AN) {
        return;
    }
    double _oldValue_xjal = this.parameter_AN;
    this.parameter_AN = parameter_AN;
    onChange_parameter_AN_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_AN.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_AN(</code> method instead.
 */
protected void onChange_parameter_AN() {
    onChange_parameter_AN_xjal( parameter_AN );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_AN_xjal( double oldValue ) {
}

public
double parameter_I;

/**
 * Returns default value for parameter <code>parameter_I</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_I_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_I( double parameter_I ) {
    if (parameter_I == this.parameter_I) {
        return;
    }
    double _oldValue_xjal = this.parameter_I;
    this.parameter_I = parameter_I;
    onChange_parameter_I_xjal( _oldValue_xjal );
    onChange();
}

```



```

/**
 * Calls "On change" action for parameter parameter_I.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_I()</code> method instead.
 */
protected void onChange_parameter_I() {
    onChange_parameter_I_xjal( parameter_I );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_I_xjal( double oldValue ) {
}

public
double parameter_V;

/**
 * Returns default value for parameter <code>parameter_V</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_V_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_V( double parameter_V ) {
    if (parameter_V == this.parameter_V) {
        return;
    }
    double _oldValue_xjal = this.parameter_V;
    this.parameter_V = parameter_V;
    onChange_parameter_V_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_V.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_V()</code> method instead.
 */
protected void onChange_parameter_V() {
    onChange_parameter_V_xjal( parameter_V );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_V_xjal( double oldValue ) {
}

public
double parameter_G;

/**
 * Returns default value for parameter <code>parameter_G</code>.
 * <i>This method should not be called by user</i>
 */

```

```

@AnyLogicInternalCodegenAPI
public double _parameter_G_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_G( double parameter_G ) {
    if (parameter_G == this.parameter_G) {
        return;
    }
    double _oldValue_xjal = this.parameter_G;
    this.parameter_G = parameter_G;
    onChange_parameter_G_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_G.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_G()</code> method instead.
 */
protected void onChange_parameter_G() {
    onChange_parameter_G_xjal( parameter_G );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_G_xjal( double oldValue ) {
}

public
double parameter_C;

/**
 * Returns default value for parameter <code>parameter_C</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_C_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_C( double parameter_C ) {
    if (parameter_C == this.parameter_C) {
        return;
    }
    double _oldValue_xjal = this.parameter_C;
    this.parameter_C = parameter_C;
    onChange_parameter_C_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_C.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_C()</code> method instead.
 */

```

```

protected void onChange_parameter_C() {
    onChange_parameter_C_xjal( parameter_C );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_C_xjal( double oldValue ) {
}

public
double parameter_A;

/**
 * Returns default value for parameter <code>parameter_A</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_A_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_A( double parameter_A ) {
    if (parameter_A == this.parameter_A) {
        return;
    }
    double _oldValue_xjal = this.parameter_A;
    this.parameter_A = parameter_A;
    onChange_parameter_A_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_A.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_A()</code> method instead.
 */
protected void onChange_parameter_A() {
    onChange_parameter_A_xjal( parameter_A );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_A_xjal( double oldValue ) {
}

@Override
public void setParametersToDefaultValues() {
    super.setParametersToDefaultValues();
    parameter_S = _parameter_S_DefaultValue_xjal();
    parameter_P = _parameter_P_DefaultValue_xjal();
    parameter_PC = _parameter_PC_DefaultValue_xjal();
    parameter_AN = _parameter_AN_DefaultValue_xjal();
    parameter_I = _parameter_I_DefaultValue_xjal();
    parameter_V = _parameter_V_DefaultValue_xjal();
    parameter_G = _parameter_G_DefaultValue_xjal();
    parameter_C = _parameter_C_DefaultValue_xjal();
    parameter_A = _parameter_A_DefaultValue_xjal();
}

```

```

@Override
public boolean setParameter(String _name_xjal, Object _value_xjal, boolean
_callOnChange_xjal) {
    switch ( _name_xjal ) {
        case "parameter_S":
            if ( _callOnChange_xjal ) {
                set_parameter_S( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_S = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_P":
            if ( _callOnChange_xjal ) {
                set_parameter_P( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_P = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_PC":
            if ( _callOnChange_xjal ) {
                set_parameter_PC( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_PC = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_AN":
            if ( _callOnChange_xjal ) {
                set_parameter_AN( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_AN = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_I":
            if ( _callOnChange_xjal ) {
                set_parameter_I( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_I = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_V":
            if ( _callOnChange_xjal ) {
                set_parameter_V( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_V = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_G":
            if ( _callOnChange_xjal ) {
                set_parameter_G( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_G = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_C":
            if ( _callOnChange_xjal ) {
                set_parameter_C( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_C = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_A":
            if ( _callOnChange_xjal ) {

```

```

        set_parameter_A( ((Number) _value_xjal).doubleValue() );
    } else {
        parameter_A = ((Number) _value_xjal).doubleValue();
    }
    return true;
default:
    return super.setParameter( _name_xjal, _value_xjal, _callOnChange_xjal );
}
}

@Override
public <T> T getParameter(String _name_xjal) {
    Object _result_xjal;
    switch ( _name_xjal ) {
    case "parameter_S": _result_xjal = parameter_S; break;
    case "parameter_P": _result_xjal = parameter_P; break;
    case "parameter_PC": _result_xjal = parameter_PC; break;
    case "parameter_AN": _result_xjal = parameter_AN; break;
    case "parameter_I": _result_xjal = parameter_I; break;
    case "parameter_V": _result_xjal = parameter_V; break;
    case "parameter_G": _result_xjal = parameter_G; break;
    case "parameter_C": _result_xjal = parameter_C; break;
    case "parameter_A": _result_xjal = parameter_A; break;
    default: _result_xjal = super.getParameter( _name_xjal ); break;
    }
    return (T) _result_xjal;
}

@AnyLogicInternalCodegenAPI
private static String[] _parameterNames_xjal;

@Override
public String[] getParameterNames() {
    String[] result = _parameterNames_xjal;
    if (result == null) {
        List<String> list = new ArrayList<>( Arrays.asList(
super.getParameterNames() ) );
        list.add( "parameter_S" );
        list.add( "parameter_P" );
        list.add( "parameter_PC" );
        list.add( "parameter_AN" );
        list.add( "parameter_I" );
        list.add( "parameter_V" );
        list.add( "parameter_G" );
        list.add( "parameter_C" );
        list.add( "parameter_A" );
        result = list.toArray( new String[ list.size() ] );
        _parameterNames_xjal = result;
    }
    return result;
}

// Dynamic (Flow/Auxiliary/Stock) Variables

public double RATE_S;

public double RATE_I;

public double RATE_V;

public double RATE_C;

```

```
public double RATE_G;
public double RATE_P;
public double RATE_PC;
public double RATE_AN;
public double LOST_S;
public double flow;
public double LOST_P;
public double flow2;
public double LOST_PC;
public double flow3;
public double LOST_AN;
public double flow4;
public double LOST_I;
public double flow5;
public double LOST_V;
public double flow6;
public double LOST_G;
public double flow7;
public double LOST_C;
public double flow8;
public double RETURNED;
public double RATE_A;
public double LOST_A;
public double flow1;
public double CR;
public double STATIC;
public double INTERACTIVE;
public double POPUP;
public double PICTURES;
public double GAMES;
public double VIDEO;
```

```

public double COLORS;

public double ANNOUNCE;

public double SUCCESS;

public double ANIMATED;
/**
 * Writes model variables into given arrays
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    _d_xjal[ _d_idx_xjal + 0 ] = CR;
    _d_xjal[ _d_idx_xjal + 1 ] = STATIC;
    _d_xjal[ _d_idx_xjal + 2 ] = INTERACTIVE;
    _d_xjal[ _d_idx_xjal + 3 ] = POPUP;
    _d_xjal[ _d_idx_xjal + 4 ] = PICTURES;
    _d_xjal[ _d_idx_xjal + 5 ] = GAMES;
    _d_xjal[ _d_idx_xjal + 6 ] = VIDEO;
    _d_xjal[ _d_idx_xjal + 7 ] = COLORS;
    _d_xjal[ _d_idx_xjal + 8 ] = ANNOUNCE;
    _d_xjal[ _d_idx_xjal + 9 ] = SUCCESS;
    _d_xjal[ _d_idx_xjal + 10 ] = ANIMATED;
    _d_idx_xjal += 11;
    super.getPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * Writes given arrays to model variables
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    CR = _d_xjal[ _d_idx_xjal + 0 ];
    STATIC = _d_xjal[ _d_idx_xjal + 1 ];
    INTERACTIVE = _d_xjal[ _d_idx_xjal + 2 ];
    POPUP = _d_xjal[ _d_idx_xjal + 3 ];
    PICTURES = _d_xjal[ _d_idx_xjal + 4 ];
    GAMES = _d_xjal[ _d_idx_xjal + 5 ];
    VIDEO = _d_xjal[ _d_idx_xjal + 6 ];
    COLORS = _d_xjal[ _d_idx_xjal + 7 ];
    ANNOUNCE = _d_xjal[ _d_idx_xjal + 8 ];
    SUCCESS = _d_xjal[ _d_idx_xjal + 9 ];
    ANIMATED = _d_xjal[ _d_idx_xjal + 10 ];
    _d_idx_xjal += 11;
    super.putPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * This function gets phase vector from initial conditions (is used while solving
algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    _a_xjal[ _a_idx_xjal + 0 ] = LOST_S;
    _a_xjal[ _a_idx_xjal + 1 ] = flow;
    _a_xjal[ _a_idx_xjal + 2 ] = LOST_P;
    _a_xjal[ _a_idx_xjal + 3 ] = flow2;
    _a_xjal[ _a_idx_xjal + 4 ] = LOST_PC;
    _a_xjal[ _a_idx_xjal + 5 ] = flow3;
    _a_xjal[ _a_idx_xjal + 6 ] = LOST_AN;
}

```

```

    _a_xjal[ _a_idx_xjal + 7 ] = flow4;
    _a_xjal[ _a_idx_xjal + 8 ] = LOST_V;
    _a_xjal[ _a_idx_xjal + 9 ] = flow6;
    _a_xjal[ _a_idx_xjal + 10 ] = LOST_G;
    _a_xjal[ _a_idx_xjal + 11 ] = flow7;
    _a_xjal[ _a_idx_xjal + 12 ] = LOST_C;
    _a_xjal[ _a_idx_xjal + 13 ] = flow8;
    _a_xjal[ _a_idx_xjal + 14 ] = LOST_A;
    _a_xjal[ _a_idx_xjal + 15 ] = flow1;
    _a_idx_xjal += 16;
    super.getPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

/**
 * This function places phase vector into initial conditions (is used while solving
 algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    LOST_S = _a_xjal[ _a_idx_xjal + 0 ];
    flow = _a_xjal[ _a_idx_xjal + 1 ];
    LOST_P = _a_xjal[ _a_idx_xjal + 2 ];
    flow2 = _a_xjal[ _a_idx_xjal + 3 ];
    LOST_PC = _a_xjal[ _a_idx_xjal + 4 ];
    flow3 = _a_xjal[ _a_idx_xjal + 5 ];
    LOST_AN = _a_xjal[ _a_idx_xjal + 6 ];
    flow4 = _a_xjal[ _a_idx_xjal + 7 ];
    LOST_V = _a_xjal[ _a_idx_xjal + 8 ];
    flow6 = _a_xjal[ _a_idx_xjal + 9 ];
    LOST_G = _a_xjal[ _a_idx_xjal + 10 ];
    flow7 = _a_xjal[ _a_idx_xjal + 11 ];
    LOST_C = _a_xjal[ _a_idx_xjal + 12 ];
    flow8 = _a_xjal[ _a_idx_xjal + 13 ];
    LOST_A = _a_xjal[ _a_idx_xjal + 14 ];
    flow1 = _a_xjal[ _a_idx_xjal + 15 ];
    _a_idx_xjal += 16;
    super.putPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

@AnyLogicInternalCodegenAPI
public void assignInitialConditions_xjal() {
    super.assignInitialConditions_xjal();
    _assign_CR_Formula_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_INTERACTIVE_Formula_xjal();
    _assign_SUCCESS_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_STATIC_Formula_xjal();
    _assign_POPUP_Formula_xjal();
    _assign_PICTURES_Formula_xjal();
    _assign_GAMES_Formula_xjal();
}

```



```

    _assign_VIDEO_Formula_xjal();
    _assign_COLORS_Formula_xjal();
    _assign_ANNOUNCE_Formula_xjal();
    _assign_ANIMATED_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void setupInitialConditions_xjal(Class<?> callerClass) {
    if (callerClass != Main.class) {
        return;
    }
    if (getInitialAlgebraicFlatEquationsCount_xjal() > 0) {
        SDIntegrationManager integrationManagerForInitialConditions = new
SDIntegrationManager( 0, getInitialAlgebraicFlatEquationsCount_xjal(),
getInitialFormulaFlatEquationsCount_xjal() );
        integrationManagerForInitialConditions.doStep( this, 0, 0.1, true );
    } else {
        assignInitialConditions_xjal();
    }
}

@AnyLogicInternalCodegenAPI
public void _assign_CR_Formula_xjal() {
    CR =
100
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_S_Formula_xjal() {
    RATE_S =
CR * 0.084
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_V_Formula_xjal() {
    RATE_V =
CR * 0.12
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_C_Formula_xjal() {
    RATE_C =
CR * 0.11
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_G_Formula_xjal() {
    RATE_G =
CR * 0.093
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_P_Formula_xjal() {
    RATE_P =
CR * 0.013
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_PC_Formula_xjal() {
    RATE_PC =
CR * 0.10
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_AN_Formula_xjal() {
    RATE_AN =
CR * 0.075
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_A_Formula_xjal() {
    RATE_A =
CR * 0.149
;
}
@AnyLogicInternalCodegenAPI
public void _assign_INTERACTIVE_Formula_xjal() {
    INTERACTIVE =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_SUCCESS_Formula_xjal() {
    SUCCESS =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_I_Formula_xjal() {
    RATE_I =
CR * 0.139
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_I_Formula_xjal() {
    LOST_I =
INTERACTIVE * 0.25 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow5_Formula_xjal() {
    flow5 =
INTERACTIVE * 0.75 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RETURNED_Formula_xjal() {
    RETURNED =
SUCCESS * 1
;
}
@AnyLogicInternalCodegenAPI
public void _assign_STATIC_Formula_xjal() {
    STATIC =
RATE_S - LOST_S - flow + 0 + parameter_S
;
}
@AnyLogicInternalCodegenAPI
public void _assign_POPUP_Formula_xjal() {
    POPUP =
RATE_P - flow2 - LOST_P + 0 + parameter_P
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _assign_PICTURES_Formula_xjal() {
        PICTURES =
RATE_PC - flow3 - LOST_PC + 0 + parameter_PC
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_GAMES_Formula_xjal() {
        GAMES =
RATE_G - flow7 - LOST_G + 0 + parameter_G
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_VIDEO_Formula_xjal() {
        VIDEO =
RATE_V - flow6 - LOST_V + 0 + parameter_V
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_COLORS_Formula_xjal() {
        COLORS =
RATE_C - LOST_C - flow8 + 0 + parameter_C
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANNOUNCE_Formula_xjal() {
        ANNOUNCE =
RATE_AN - flow4 - LOST_AN + 0 + parameter_AN
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANIMATED_Formula_xjal() {
        ANIMATED =
RATE_A - flow1 - LOST_A + 0 + parameter_A
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_S_Formula_xjal() {
        LOST_S =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow_Formula_xjal() {
        flow =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_P_Formula_xjal() {
        LOST_P =
POPUP * 0.3 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow2_Formula_xjal() {
        flow2 =
POPUP * 0.7 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_PC_Formula_xjal() {
        LOST_PC =

```

```

PICTURES * 0.45 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow3_Formula_xjal() {
    flow3 =
PICTURES * 0.55 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_AN_Formula_xjal() {
    LOST_AN =
ANNOUNCE * 0.6 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow4_Formula_xjal() {
    flow4 =
ANNOUNCE * 0.40 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_V_Formula_xjal() {
    LOST_V =
VIDEO * 0.35 + parameter_V
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow6_Formula_xjal() {
    flow6 =
VIDEO * 0.75
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_G_Formula_xjal() {
    LOST_G =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow7_Formula_xjal() {
    flow7 =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_C_Formula_xjal() {
    LOST_C =
COLORS * 0.4
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow8_Formula_xjal() {
    flow8 =
COLORS * 0.6 + parameter_C
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_A_Formula_xjal() {
    LOST_A =
ANIMATED * 0.20 + parameter_A
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_flow1_Formula_xjal() {
    flow1 =
ANIMATED * 0.8 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void formulasExecute_xjal() {
    super.formulasExecute_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_LOST_S_Formula_xjal();
    _assign_flow_Formula_xjal();
    _assign_LOST_P_Formula_xjal();
    _assign_flow2_Formula_xjal();
    _assign_LOST_PC_Formula_xjal();
    _assign_flow3_Formula_xjal();
    _assign_LOST_AN_Formula_xjal();
    _assign_flow4_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_LOST_V_Formula_xjal();
    _assign_flow6_Formula_xjal();
    _assign_LOST_G_Formula_xjal();
    _assign_flow7_Formula_xjal();
    _assign_LOST_C_Formula_xjal();
    _assign_flow8_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_LOST_A_Formula_xjal();
    _assign_flow1_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void getRightPart_xjal( double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal ) {
    _get_CR_Derivative_xjal( _d_xjal, _d_idx_xjal + 0 );
    _get_STATIC_Derivative_xjal( _d_xjal, _d_idx_xjal + 1 );
    _get_INTERACTIVE_Derivative_xjal( _d_xjal, _d_idx_xjal + 2 );
    _get_POPUP_Derivative_xjal( _d_xjal, _d_idx_xjal + 3 );
    _get_PICTURES_Derivative_xjal( _d_xjal, _d_idx_xjal + 4 );
    _get_GAMES_Derivative_xjal( _d_xjal, _d_idx_xjal + 5 );
    _get_VIDEO_Derivative_xjal( _d_xjal, _d_idx_xjal + 6 );
    _get_COLORS_Derivative_xjal( _d_xjal, _d_idx_xjal + 7 );
    _get_ANNOUNCE_Derivative_xjal( _d_xjal, _d_idx_xjal + 8 );
    _get_SUCCESS_Derivative_xjal( _d_xjal, _d_idx_xjal + 9 );
    _get_ANIMATED_Derivative_xjal( _d_xjal, _d_idx_xjal + 10 );
    _d_idx_xjal += 11;
    super.getRightPart_xjal( _d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void getRightPartForInitialConditions_xjal( double[] _a_xjal, int
_a_idx_xjal ) {
    _get_LOST_S_Equation_xjal( _a_xjal, _a_idx_xjal + 0 );

```

```

    _get_flow_Equation_xjal( _a_xjal, _a_idx_xjal + 1 );
    _get_LOST_P_Equation_xjal( _a_xjal, _a_idx_xjal + 2 );
    _get_flow2_Equation_xjal( _a_xjal, _a_idx_xjal + 3 );
    _get_LOST_PC_Equation_xjal( _a_xjal, _a_idx_xjal + 4 );
    _get_flow3_Equation_xjal( _a_xjal, _a_idx_xjal + 5 );
    _get_LOST_AN_Equation_xjal( _a_xjal, _a_idx_xjal + 6 );
    _get_flow4_Equation_xjal( _a_xjal, _a_idx_xjal + 7 );
    _get_LOST_V_Equation_xjal( _a_xjal, _a_idx_xjal + 8 );
    _get_flow6_Equation_xjal( _a_xjal, _a_idx_xjal + 9 );
    _get_LOST_G_Equation_xjal( _a_xjal, _a_idx_xjal + 10 );
    _get_flow7_Equation_xjal( _a_xjal, _a_idx_xjal + 11 );
    _get_LOST_C_Equation_xjal( _a_xjal, _a_idx_xjal + 12 );
    _get_flow8_Equation_xjal( _a_xjal, _a_idx_xjal + 13 );
    _get_LOST_A_Equation_xjal( _a_xjal, _a_idx_xjal + 14 );
    _get_flow1_Equation_xjal( _a_xjal, _a_idx_xjal + 15 );
    _a_idx_xjal += 16;
    super.getRightPartForInitialConditions_xjal( _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void _get_CR_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RETURNED - RATE_S - RATE_I- RATE_V- RATE_C- RATE_G- RATE_P- RATE_PC- RATE_AN- RATE_A
;
}
@AnyLogicInternalCodegenAPI
public void _get_STATIC_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_S - LOST_S - flow
;
}
@AnyLogicInternalCodegenAPI
public void _get_INTERACTIVE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_I - LOST_I - flow5
;
}
@AnyLogicInternalCodegenAPI
public void _get_POPUP_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_P - LOST_P - flow2
;
}
@AnyLogicInternalCodegenAPI
public void _get_PICTURES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_PC - LOST_PC - flow3
;
}
@AnyLogicInternalCodegenAPI
public void _get_GAMES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_G - LOST_G - flow7
;
}
@AnyLogicInternalCodegenAPI
public void _get_VIDEO_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_V - LOST_V - flow6
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _get_COLORS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_C - LOST_C - flow8
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANNOUNCE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_AN - LOST_AN - flow4
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_SUCCESS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+flow + flow2 + flow3+ flow4+ flow5+ flow6+ flow7+ flow8+ flow1 - RETURNED
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANIMATED_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_A - LOST_A - flow1
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_S_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- LOST_S;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- flow;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_P_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.3 + parameter_P
- LOST_P;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow2_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.7 + parameter_P
- flow2;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_PC_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.45 + parameter_PC
- LOST_PC;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow3_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.55 + parameter_PC
- flow3;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_AN_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =

```

```

ANNOUNCE * 0.6 + parameter_AN
- LOST_AN;
}
@AnyLogicInternalCodegenAPI
public void _get_flow4_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.40 + parameter_AN
- flow4;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_V_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.35 + parameter_V
- LOST_V;
}
@AnyLogicInternalCodegenAPI
public void _get_flow6_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.75
- flow6;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_G_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- LOST_G;
}
@AnyLogicInternalCodegenAPI
public void _get_flow7_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- flow7;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_C_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.4
- LOST_C;
}
@AnyLogicInternalCodegenAPI
public void _get_flow8_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.6 + parameter_C
- flow8;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_A_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.20 + parameter_A
- LOST_A;
}
@AnyLogicInternalCodegenAPI
public void _get_flow1_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.8 + parameter_A
- flow1;
}
@AnyLogicInternalCodegenAPI
protected SDIntegrationManager integrationManager_xjal = null;

@AnyLogicInternalCodegenAPI
public SDIntegrationManager getIntegrationManager_xjal() {

```



```

        if (integrationManager_xjal == null) {
            integrationManager_xjal = new
SDIntegrationManager( getDifferentialFlatEquationsCount_xjal(),
getRuntimeAlgebraicFlatEquationsCount_xjal(),
getRuntimeFormulaFlatEquationsCount_xjal() );
        }
        return integrationManager_xjal;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getDifferentialFlatEquationsCount_xjal() {
        return super.getDifferentialFlatEquationsCount_xjal() + 11;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getRuntimeFormulaFlatEquationsCount_xjal() {
        return super.getRuntimeFormulaFlatEquationsCount_xjal() + 28;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialAlgebraicFlatEquationsCount_xjal() {
        return super.getInitialAlgebraicFlatEquationsCount_xjal() + 16;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialFormulaFlatEquationsCount_xjal() {
        return super.getInitialFormulaFlatEquationsCount_xjal() + 23;
    }

    @AnyLogicInternalCodegenAPI
    private static Map<String, IElementDescriptor> elementDescriptors_xjal =
createElementDescriptors( Main.class );

    @AnyLogicInternalCodegenAPI
    @Override
    public Map<String, IElementDescriptor> getElementDescriptors() {
        return elementDescriptors_xjal;
    }
    @AnyLogicCustomProposalPriority(type =
AnyLogicCustomProposalPriority.Type.STATIC_ELEMENT)
    public static final Scale scale = new Scale( 10.0 );

    @Override
    public Scale getScale() {
        return scale;
    }

    // Events

    @AnyLogicInternalCodegenAPI
    public EventTimeout _chart_autoUpdateEvent_xjal = new EventTimeout(this);
    @AnyLogicInternalCodegenAPI
    public EventTimeout _autoCreatedDS_xjal = new EventTimeout(this);

    @Override
    @AnyLogicInternalCodegenAPI
    public String getNameOf( EventTimeout _e ) {

```

```

        if( _e == _chart_autoUpdateEvent_xjal ) return "chart auto update event";
        if( _e == _autoCreatedDS_xjal ) return "Auto-created DataSets auto update event";
        return super.getNameOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public EventTimeout.Mode getModeOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        if ( _e == _autoCreatedDS_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        return super.getModeOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double getFirstOccurrenceTime( EventTimeout _e ) {
        double _t;
        if ( _e == _chart_autoUpdateEvent_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if ( _e == _autoCreatedDS_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.getFirstOccurrenceTime( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double evaluateTimeoutOf( EventTimeout _e ) {
        double _t;
        if( _e == _chart_autoUpdateEvent_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if( _e == _autoCreatedDS_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.evaluateTimeoutOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public void executeActionOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) {
            chart.updateData();
            return;
        }
    }

```

```

if ( _e == _autoCreatedDS_xjal ) {
    _ds_RATE_S.update();
    _ds_RATE_I.update();
    _ds_RATE_V.update();
    _ds_RATE_C.update();
    _ds_RATE_G.update();
    _ds_RATE_P.update();
    _ds_RATE_PC.update();
    _ds_RATE_AN.update();
    _ds_LOST_S.update();
    _ds_flow.update();
    _ds_LOST_P.update();
    _ds_flow2.update();
    _ds_LOST_PC.update();
    _ds_flow3.update();
    _ds_LOST_AN.update();
    _ds_flow4.update();
    _ds_LOST_I.update();
    _ds_flow5.update();
    _ds_LOST_V.update();
    _ds_flow6.update();
    _ds_LOST_G.update();
    _ds_flow7.update();
    _ds_LOST_C.update();
    _ds_flow8.update();
    _ds_RETURNED.update();
    _ds_RATE_A.update();
    _ds_LOST_A.update();
    _ds_flow1.update();
    _ds_CR.update();
    _ds_STATIC.update();
    _ds_INTERACTIVE.update();
    _ds_POPUP.update();
    _ds_PICTURES.update();
    _ds_GAMES.update();
    _ds_VIDEO.update();
    _ds_COLORS.update();
    _ds_ANNOUNCE.update();
    _ds_SUCCESS.update();
    _ds_ANIMATED.update();
    return;
}
super.executeActionOf( _e );
}

```

```

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _STATECHART_COUNT_xjal = 0;

```

```

/**
 * Auto-created data set(s) for RATE_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_S );
    }
}

```

```

        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_P
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override

```

```

    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_AN );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_P
 */
@AnyLogicInternalCodegenAPI

```

```

public DataSet _ds_LOST_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow2
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow2 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow2 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow3
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow3 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow3 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_AN );
        _lastUpdateTime = time();
    }
} );
/**

```

```

* Auto-created data set(s) for flow4
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow4 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow4 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_I
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_I );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow5
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow5 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow5 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_V
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_V );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow6
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow6 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow6 );
        _lastUpdateTime = time();
    }
} );

```

```

    }
} );
/**
 * Auto-created data set(s) for LOST_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow7
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow7 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow7 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow8
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow8 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow8 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RETURNED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RETURNED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {

```



```

        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RETURNED );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow1
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow1 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow1 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for CR
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_CR = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.CR );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for STATIC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_STATIC = new DataSet( 100, new DataUpdater_xjal() {

```

```

    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.STATIC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for INTERACTIVE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_INTERACTIVE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.INTERACTIVE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for POPUP
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_POPUP = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.POPUP );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for PICTURES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_PICTURES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.PICTURES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for GAMES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_GAMES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.GAMES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for VIDEO

```

```

*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_VIDEO = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.VIDEO );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for COLORS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_COLORS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.COLORS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANNOUNCE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANNOUNCE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANNOUNCE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for SUCCESS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_SUCCESS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.SUCCESS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANIMATED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANIMATED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANIMATED );
        _lastUpdateTime = time();
    }
}

```

```

    } );
    // View areas
    public ViewArea _origin_VA = new ViewArea( this, "[Origin]", 0, 0, 1000.0, 600.0 );
    @Override
    @AnyLogicInternalCodegenAPI
    public int getViewAreas(Map<String, ViewArea> _output) {
        if ( _output != null ) {
            _output.put( "_origin_VA", this._origin_VA );
        }
        return 1 + super.getViewAreas( _output );
    }
    @AnyLogicInternalCodegenAPI
    protected static final int _chart = 1;

    /** Internal constant, shouldn't be accessed by user */
    @AnyLogicInternalCodegenAPI
    protected static final int _SHAPE_NEXT_ID_xjal = 2;

    @AnyLogicInternalCodegenAPI
    public boolean isPublicPresentationDefined() {
        return true;
    }

    @AnyLogicInternalCodegenAPI
    public boolean isEmbeddedAgentPresentationVisible( Agent _a ) {
        return super.isEmbeddedAgentPresentationVisible( _a );
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem0Value() {
        return
ANIMATED
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem1Value() {
        return
STATIC
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem2Value() {
        return
POPUP
;
    }

    /**
     * <i>This method should not be called by user</i>
     */

```

```

    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem3Value() {
        return
INTERACTIVE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem4Value() {
        return
PICTURES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem5Value() {
        return
GAMES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem6Value() {
        return
VIDEO
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem7Value() {
        return
ANNOUNCE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem8Value() {
        return
COLORS
;
    }
    protected BarChart chart;
    @AnyLogicInternalCodegenAPI
    private void _createPersistentElementsBP0_xjal() {
    }

    @AnyLogicInternalCodegenAPI

```

```

private void _createPersistentElementsAP0_xjal() {
    {
        List<DataItem> _items = new ArrayList<DataItem>( 9 );
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem0Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem1Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem2Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem3Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem4Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem5Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem6Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem7Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem8Value() );
            }
        });
        List<String> _titles = new ArrayList<String>( 9 );
        _titles.add( "ANIMATED" );
        _titles.add( "STATIC" );
        _titles.add( "POPUP" );
        _titles.add( "INTERACTIVE" );
    }
}

```

```

_titles.add( "PICTURES" );
_titles.add( "GAMES" );
_titles.add( "VIDEO" );
_titles.add( "ANNOUNCE" );
_titles.add( "COLORS" );
List<Color> _colors = new ArrayList<Color>( 9 );
_colors.add( limeGreen );
_colors.add( slateGray );
_colors.add( darkMagenta );
_colors.add( orange );
_colors.add( darkKhaki );
_colors.add( mediumTurquoise );
_colors.add( lightSlateBlue );
_colors.add( deepPink );
_colors.add( darkOrange );
chart = new BarChart(
    Main.this, true, 20.0, 700.0,
    1420.0, 580.0,
    null, null,
    50.0, 20.0,
    1330.0, 460.0, white, black, black,
    80.0, Chart.SOUTH,
    Chart.NORTH, Chart.SCALE_AUTO,
    0, 0, 0.8,
    Chart.GRID_DEFAULT,
    darkGray, darkGray, _items, _titles, _colors );
}
}

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsBS0_xjal() {
}

// Static initialization of persistent elements
{
_createPersistentElementsBP0_xjal();
}
protected ShapeTopLevelPresentationGroup presentation;
protected ShapeModelElementsGroup icon;

@Override
@AnyLogicInternalCodegenAPI
public ShapeTopLevelPresentationGroup getPresentationShape() {
return presentation;
}

@Override
@AnyLogicInternalCodegenAPI
public ShapeModelElementsGroup getModelElementsShape() {
return icon;
}

/**
 * Constructor
 */
public Main( Engine engine, Agent owner, AgentList<? extends Main> ownerPopulation )
{
super( engine, owner, ownerPopulation );
instantiateBaseStructureThis_xjal();
}
}

```

```

@AnyLogicInternalCodegenAPI
public void onOwnerChanged_xjal() {
    super.onOwnerChanged_xjal();
    setupReferences_xjal();
}

@AnyLogicInternalCodegenAPI
public void instantiateBaseStructure_xjal() {
    super.instantiateBaseStructure_xjal();
    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
private void instantiateBaseStructureThis_xjal() {
    setupReferences_xjal();
    // Registering in Engine continuous part
    getEngine().registerAgentWithEquations( this );
}

@AnyLogicInternalCodegenAPI
private void setupReferences_xjal() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main( double parameter_S, double parameter_P, double parameter_PC, double
parameter_AN, double parameter_I, double parameter_V, double parameter_G, double
parameter_C, double parameter_A ) {
    markParametersAreSet();
    this.parameter_S = parameter_S;
    this.parameter_P = parameter_P;
    this.parameter_PC = parameter_PC;
    this.parameter_AN = parameter_AN;
    this.parameter_I = parameter_I;
    this.parameter_V = parameter_V;
    this.parameter_G = parameter_G;
    this.parameter_C = parameter_C;
    this.parameter_A = parameter_A;
}

@Override
@AnyLogicInternalCodegenAPI
public void doCreate() {
    super.doCreate();
    // Assigning initial values for plain variables
    setupPlainVariables_Main_xjal();
    // Dynamic initialization of persistent elements
    _createPersistentElementsAP0_xjal();
    presentation = new ShapeTopLevelPresentationGroup( Main.this, true, 0, 0, 0, 0 ,
chart );
    icon = new ShapeModelElementsGroup( Main.this, getElementProperty(
"modell1.Main.icon", IElementDescriptor.MODEL_ELEMENT_DESCRIPTOR ) );
}

```



```

    icon.setIconOffsets( 0.0, 0.0 );
    // Port connectors with non-replicated objects
    // Creating replicated embedded objects
    setupInitialConditions_xjal( Main.class );
    // Dynamic initialization of persistent elements
    _createPersistentElementsBS0_xjal();
}

@Override
@AnyLogicInternalCodegenAPI
public void doStart() {
    super.doStart();
    _chart_autoUpdateEvent_xjal.start();
    _autoCreatedDS_xjal.start();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
public void setupPlainVariables_xjal() {
    setupPlainVariables_Main_xjal();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
private void setupPlainVariables_Main_xjal() {

// User API -----
@AnyLogicInternalCodegenAPI
public static LinkToAgentAnimationSettings _connections_commonAnimationSettings_xjal
= new LinkToAgentAnimationSettingsImpl( false, black, 1.0, LINE_STYLE_SOLID,
ARROW_NONE, 0.0 );

public LinkToAgentCollection<Agent, Agent> connections = new
LinkToAgentStandardImpl<Agent, Agent>(this, _connections_commonAnimationSettings_xjal);
@Override
public LinkToAgentCollection<? extends Agent, ? extends Agent>
getLinkToAgentStandard_xjal() {
    return connections;
}

@AnyLogicInternalCodegenAPI
public void drawLinksToAgents(boolean _underAgents_xjal, LinkToAgentAnimator
_animator_xjal) {
    super.drawLinksToAgents(_underAgents_xjal, _animator_xjal);
    if ( _underAgents_xjal ) {
        _animator_xjal.drawLink( this, connections, true, true );
    }
}

public AgentList<? extends Main> getPopulation() {
    return (AgentList<? extends Main>) super.getPopulation();
}

```

```

}

public List<? extends Main> agentsInRange( double distance ) {
    return (List<? extends Main>) super.agentsInRange( distance );
}

@Override
@AnyLogicInternalCodegenAPI
public boolean isLoggingToDB(EventOriginator _e) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return false;
    if ( _e == _autoCreatedDS_xjal ) return false;
    return super.isLoggingToDB( _e );
}

@AnyLogicInternalCodegenAPI
public void onDestroy() {
    _chart_autoUpdateEvent_xjal.onDestroy();
    _autoCreatedDS_xjal.onDestroy();
    // Unregistering in Engine continuous part
    getEngine().unregisterAgentWithEquations( this );
    _ds_RATE_S.destroyUpdater_xjal();
    _ds_RATE_I.destroyUpdater_xjal();
    _ds_RATE_V.destroyUpdater_xjal();
    _ds_RATE_C.destroyUpdater_xjal();
    _ds_RATE_G.destroyUpdater_xjal();
    _ds_RATE_P.destroyUpdater_xjal();
    _ds_RATE_PC.destroyUpdater_xjal();
    _ds_RATE_AN.destroyUpdater_xjal();
    _ds_LOST_S.destroyUpdater_xjal();
    _ds_flow.destroyUpdater_xjal();
    _ds_LOST_P.destroyUpdater_xjal();
    _ds_flow2.destroyUpdater_xjal();
    _ds_LOST_PC.destroyUpdater_xjal();
    _ds_flow3.destroyUpdater_xjal();
    _ds_LOST_AN.destroyUpdater_xjal();
    _ds_flow4.destroyUpdater_xjal();
    _ds_LOST_I.destroyUpdater_xjal();
    _ds_flow5.destroyUpdater_xjal();
    _ds_LOST_V.destroyUpdater_xjal();
    _ds_flow6.destroyUpdater_xjal();
    _ds_LOST_G.destroyUpdater_xjal();
    _ds_flow7.destroyUpdater_xjal();
    _ds_LOST_C.destroyUpdater_xjal();
    _ds_flow8.destroyUpdater_xjal();
    _ds_RETURNED.destroyUpdater_xjal();
    _ds_RATE_A.destroyUpdater_xjal();
    _ds_LOST_A.destroyUpdater_xjal();
    _ds_flow1.destroyUpdater_xjal();
    _ds_CR.destroyUpdater_xjal();
    _ds_STATIC.destroyUpdater_xjal();
    _ds_INTERACTIVE.destroyUpdater_xjal();
    _ds_POPUP.destroyUpdater_xjal();
    _ds_PICTURES.destroyUpdater_xjal();
    _ds_GAMES.destroyUpdater_xjal();
    _ds_VIDEO.destroyUpdater_xjal();
    _ds_COLORS.destroyUpdater_xjal();
    _ds_ANNOUNCE.destroyUpdater_xjal();
    _ds_SUCCESS.destroyUpdater_xjal();
    _ds_ANIMATED.destroyUpdater_xjal();
    super.onDestroy();
}

```

```

package model1;

import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayDeque;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.Collection;
import java.util.Collections;
import java.util.Comparator;
import java.util.Currency;
import java.util.Date;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Hashtable;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import java.util.Map;
import java.util.PriorityQueue;
import java.util.Random;
import java.util.Set;
import java.util.SortedMap;
import java.util.SortedSet;
import java.util.Stack;
import java.util.Timer;
import java.util.TreeMap;
import java.util.TreeSet;
import java.util.Vector;
import java.awt.Color;
import java.awt.Font;
import com.anylogic.engine.connectivity.ResultSet;
import com.anylogic.engine.connectivity.Statement;
import com.anylogic.engine.elements.*;
import com.anylogic.engine.markup.Network;
import com.anylogic.engine.Position;
import com.anylogic.engine.markup.PedFlowStatistics;
import com.anylogic.engine.markup.DensityMap;

import static java.lang.Math.*;
import static com.anylogic.engine.UtilitiesArray.*;
import static com.anylogic.engine.UtilitiesCollection.*;
import static com.anylogic.engine.presentation.UtilitiesColor.*;
import static com.anylogic.engine.HyperArray.*;

import com.anylogic.engine.*;
import com.anylogic.engine.analysis.*;
import com.anylogic.engine.connectivity.*;
import com.anylogic.engine.database.*;
import com.anylogic.engine.gis.*;
import com.anylogic.engine.markup.*;
import com.anylogic.engine.routing.*;
import com.anylogic.engine.presentation.*;
import com.anylogic.engine.gui.*;

```

```

import java.awt.geom.Arc2D;

public class Main extends Agent
{
    // Parameters

    public
double    parameter_S;

    /**
     * Returns default value for parameter <code>parameter_S</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_S_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_S( double parameter_S ) {
        if (parameter_S == this.parameter_S) {
            return;
        }
        double _oldValue_xjal = this.parameter_S;
        this.parameter_S = parameter_S;
        onChange_parameter_S_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_S.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_S()</code> method instead.
     */
    protected void onChange_parameter_S() {
        onChange_parameter_S_xjal( parameter_S );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_S_xjal( double oldValue ) {

    }

    public
double    parameter_P;

    /**
     * Returns default value for parameter <code>parameter_P</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_P_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_P( double parameter_P ) {
        if (parameter_P == this.parameter_P) {
            return;
        }
    }
}

```

```

    }
    double _oldValue_xjal = this.parameter_P;
    this.parameter_P = parameter_P;
    onChange_parameter_P_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_P.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_P()</code> method instead.
 */
protected void onChange_parameter_P() {
    onChange_parameter_P_xjal( parameter_P );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_P_xjal( double oldValue ) {

public
double parameter_PC;

/**
 * Returns default value for parameter <code>parameter_PC</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_PC_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_PC( double parameter_PC ) {
    if (parameter_PC == this.parameter_PC) {
        return;
    }
    double _oldValue_xjal = this.parameter_PC;
    this.parameter_PC = parameter_PC;
    onChange_parameter_PC_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_PC.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_PC()</code> method instead.
 */
protected void onChange_parameter_PC() {
    onChange_parameter_PC_xjal( parameter_PC );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_PC_xjal( double oldValue ) {

```

```

public
double parameter_AN;

/**
 * Returns default value for parameter <code>parameter_AN</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_AN_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_AN( double parameter_AN ) {
    if (parameter_AN == this.parameter_AN) {
        return;
    }
    double _oldValue_xjal = this.parameter_AN;
    this.parameter_AN = parameter_AN;
    onChange_parameter_AN_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_AN.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_AN</code> method instead.
 */
protected void onChange_parameter_AN() {
    onChange_parameter_AN_xjal( parameter_AN );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_AN_xjal( double oldValue ) {
}

public
double parameter_I;

/**
 * Returns default value for parameter <code>parameter_I</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_I_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_I( double parameter_I ) {
    if (parameter_I == this.parameter_I) {
        return;
    }
    double _oldValue_xjal = this.parameter_I;
    this.parameter_I = parameter_I;
    onChange_parameter_I_xjal( _oldValue_xjal );
    onChange();
}

```

```

/**
 * Calls "On change" action for parameter parameter_I.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_I()</code> method instead.
 */
protected void onChange_parameter_I() {
    onChange_parameter_I_xjal( parameter_I );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_I_xjal( double oldValue ) {
}

public
double parameter_V;

/**
 * Returns default value for parameter <code>parameter_V</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_V_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_V( double parameter_V ) {
    if (parameter_V == this.parameter_V) {
        return;
    }
    double _oldValue_xjal = this.parameter_V;
    this.parameter_V = parameter_V;
    onChange_parameter_V_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_V.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_V()</code> method instead.
 */
protected void onChange_parameter_V() {
    onChange_parameter_V_xjal( parameter_V );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_V_xjal( double oldValue ) {
}

public
double parameter_G;

/**
 * Returns default value for parameter <code>parameter_G</code>.
 * <i>This method should not be called by user</i>
 */

```

```

@AnyLogicInternalCodegenAPI
public double _parameter_G_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_G( double parameter_G ) {
    if (parameter_G == this.parameter_G) {
        return;
    }
    double _oldValue_xjal = this.parameter_G;
    this.parameter_G = parameter_G;
    onChange_parameter_G_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_G.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_G()</code> method instead.
 */
protected void onChange_parameter_G() {
    onChange_parameter_G_xjal( parameter_G );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_G_xjal( double oldValue ) {
}

public
double parameter_C;

/**
 * Returns default value for parameter <code>parameter_C</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_C_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_C( double parameter_C ) {
    if (parameter_C == this.parameter_C) {
        return;
    }
    double _oldValue_xjal = this.parameter_C;
    this.parameter_C = parameter_C;
    onChange_parameter_C_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_C.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_C()</code> method instead.
 */

```



```

protected void onChange_parameter_C() {
    onChange_parameter_C_xjal( parameter_C );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_C_xjal( double oldValue ) {
}

public
double parameter_A;

/**
 * Returns default value for parameter <code>parameter_A</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_A_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_A( double parameter_A ) {
    if (parameter_A == this.parameter_A) {
        return;
    }
    double _oldValue_xjal = this.parameter_A;
    this.parameter_A = parameter_A;
    onChange_parameter_A_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_A.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_A()</code> method instead.
 */
protected void onChange_parameter_A() {
    onChange_parameter_A_xjal( parameter_A );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_A_xjal( double oldValue ) {
}

@Override
public void setParametersToDefaultValues() {
    super.setParametersToDefaultValues();
    parameter_S = _parameter_S_DefaultValue_xjal();
    parameter_P = _parameter_P_DefaultValue_xjal();
    parameter_PC = _parameter_PC_DefaultValue_xjal();
    parameter_AN = _parameter_AN_DefaultValue_xjal();
    parameter_I = _parameter_I_DefaultValue_xjal();
    parameter_V = _parameter_V_DefaultValue_xjal();
    parameter_G = _parameter_G_DefaultValue_xjal();
    parameter_C = _parameter_C_DefaultValue_xjal();
    parameter_A = _parameter_A_DefaultValue_xjal();
}

```

```

@Override
public boolean setParameter(String _name_xjal, Object _value_xjal, boolean
_callOnChange_xjal) {
    switch ( _name_xjal ) {
        case "parameter_S":
            if ( _callOnChange_xjal ) {
                set_parameter_S( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_S = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_P":
            if ( _callOnChange_xjal ) {
                set_parameter_P( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_P = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_PC":
            if ( _callOnChange_xjal ) {
                set_parameter_PC( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_PC = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_AN":
            if ( _callOnChange_xjal ) {
                set_parameter_AN( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_AN = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_I":
            if ( _callOnChange_xjal ) {
                set_parameter_I( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_I = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_V":
            if ( _callOnChange_xjal ) {
                set_parameter_V( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_V = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_G":
            if ( _callOnChange_xjal ) {
                set_parameter_G( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_G = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_C":
            if ( _callOnChange_xjal ) {
                set_parameter_C( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_C = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_A":
            if ( _callOnChange_xjal ) {

```

```

        set_parameter_A( ((Number) _value_xjal).doubleValue() );
    } else {
        parameter_A = ((Number) _value_xjal).doubleValue();
    }
    return true;
default:
    return super.setParameter( _name_xjal, _value_xjal, _callOnChange_xjal );
}
}

@Override
public <T> T getParameter(String _name_xjal) {
    Object _result_xjal;
    switch ( _name_xjal ) {
    case "parameter_S": _result_xjal = parameter_S; break;
    case "parameter_P": _result_xjal = parameter_P; break;
    case "parameter_PC": _result_xjal = parameter_PC; break;
    case "parameter_AN": _result_xjal = parameter_AN; break;
    case "parameter_I": _result_xjal = parameter_I; break;
    case "parameter_V": _result_xjal = parameter_V; break;
    case "parameter_G": _result_xjal = parameter_G; break;
    case "parameter_C": _result_xjal = parameter_C; break;
    case "parameter_A": _result_xjal = parameter_A; break;
    default: _result_xjal = super.getParameter( _name_xjal ); break;
    }
    return (T) _result_xjal;
}

@AnyLogicInternalCodegenAPI
private static String[] _parameterNames_xjal;

@Override
public String[] getParameterNames() {
    String[] result = _parameterNames_xjal;
    if (result == null) {
        List<String> list = new ArrayList<>( Arrays.asList(
super.getParameterNames() ) );
        list.add( "parameter_S" );
        list.add( "parameter_P" );
        list.add( "parameter_PC" );
        list.add( "parameter_AN" );
        list.add( "parameter_I" );
        list.add( "parameter_V" );
        list.add( "parameter_G" );
        list.add( "parameter_C" );
        list.add( "parameter_A" );
        result = list.toArray( new String[ list.size() ] );
        _parameterNames_xjal = result;
    }
    return result;
}

// Dynamic (Flow/Auxiliary/Stock) Variables

public double RATE_S;

public double RATE_I;

public double RATE_V;

public double RATE_C;

```

```
public double RATE_G;
public double RATE_P;
public double RATE_PC;
public double RATE_AN;
public double LOST_S;
public double flow;
public double LOST_P;
public double flow2;
public double LOST_PC;
public double flow3;
public double LOST_AN;
public double flow4;
public double LOST_I;
public double flow5;
public double LOST_V;
public double flow6;
public double LOST_G;
public double flow7;
public double LOST_C;
public double flow8;
public double RETURNED;
public double RATE_A;
public double LOST_A;
public double flow1;
public double CR;
public double STATIC;
public double INTERACTIVE;
public double POPUP;
public double PICTURES;
public double GAMES;
public double VIDEO;
```

```

public double COLORS;

public double ANNOUNCE;

public double SUCCESS;

public double ANIMATED;
/**
 * Writes model variables into given arrays
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    _d_xjal[ _d_idx_xjal + 0 ] = CR;
    _d_xjal[ _d_idx_xjal + 1 ] = STATIC;
    _d_xjal[ _d_idx_xjal + 2 ] = INTERACTIVE;
    _d_xjal[ _d_idx_xjal + 3 ] = POPUP;
    _d_xjal[ _d_idx_xjal + 4 ] = PICTURES;
    _d_xjal[ _d_idx_xjal + 5 ] = GAMES;
    _d_xjal[ _d_idx_xjal + 6 ] = VIDEO;
    _d_xjal[ _d_idx_xjal + 7 ] = COLORS;
    _d_xjal[ _d_idx_xjal + 8 ] = ANNOUNCE;
    _d_xjal[ _d_idx_xjal + 9 ] = SUCCESS;
    _d_xjal[ _d_idx_xjal + 10 ] = ANIMATED;
    _d_idx_xjal += 11;
    super.getPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * Writes given arrays to model variables
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    CR = _d_xjal[ _d_idx_xjal + 0 ];
    STATIC = _d_xjal[ _d_idx_xjal + 1 ];
    INTERACTIVE = _d_xjal[ _d_idx_xjal + 2 ];
    POPUP = _d_xjal[ _d_idx_xjal + 3 ];
    PICTURES = _d_xjal[ _d_idx_xjal + 4 ];
    GAMES = _d_xjal[ _d_idx_xjal + 5 ];
    VIDEO = _d_xjal[ _d_idx_xjal + 6 ];
    COLORS = _d_xjal[ _d_idx_xjal + 7 ];
    ANNOUNCE = _d_xjal[ _d_idx_xjal + 8 ];
    SUCCESS = _d_xjal[ _d_idx_xjal + 9 ];
    ANIMATED = _d_xjal[ _d_idx_xjal + 10 ];
    _d_idx_xjal += 11;
    super.putPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * This function gets phase vector from initial conditions (is used while solving
algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    _a_xjal[ _a_idx_xjal + 0 ] = LOST_S;
    _a_xjal[ _a_idx_xjal + 1 ] = flow;
    _a_xjal[ _a_idx_xjal + 2 ] = LOST_P;
    _a_xjal[ _a_idx_xjal + 3 ] = flow2;
    _a_xjal[ _a_idx_xjal + 4 ] = LOST_PC;
    _a_xjal[ _a_idx_xjal + 5 ] = flow3;
    _a_xjal[ _a_idx_xjal + 6 ] = LOST_AN;
}

```

```

    _a_xjal[ _a_idx_xjal + 7 ] = flow4;
    _a_xjal[ _a_idx_xjal + 8 ] = LOST_V;
    _a_xjal[ _a_idx_xjal + 9 ] = flow6;
    _a_xjal[ _a_idx_xjal + 10 ] = LOST_G;
    _a_xjal[ _a_idx_xjal + 11 ] = flow7;
    _a_xjal[ _a_idx_xjal + 12 ] = LOST_C;
    _a_xjal[ _a_idx_xjal + 13 ] = flow8;
    _a_xjal[ _a_idx_xjal + 14 ] = LOST_A;
    _a_xjal[ _a_idx_xjal + 15 ] = flow1;
    _a_idx_xjal += 16;
    super.getPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

/**
 * This function places phase vector into initial conditions (is used while solving
 algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    LOST_S = _a_xjal[ _a_idx_xjal + 0 ];
    flow = _a_xjal[ _a_idx_xjal + 1 ];
    LOST_P = _a_xjal[ _a_idx_xjal + 2 ];
    flow2 = _a_xjal[ _a_idx_xjal + 3 ];
    LOST_PC = _a_xjal[ _a_idx_xjal + 4 ];
    flow3 = _a_xjal[ _a_idx_xjal + 5 ];
    LOST_AN = _a_xjal[ _a_idx_xjal + 6 ];
    flow4 = _a_xjal[ _a_idx_xjal + 7 ];
    LOST_V = _a_xjal[ _a_idx_xjal + 8 ];
    flow6 = _a_xjal[ _a_idx_xjal + 9 ];
    LOST_G = _a_xjal[ _a_idx_xjal + 10 ];
    flow7 = _a_xjal[ _a_idx_xjal + 11 ];
    LOST_C = _a_xjal[ _a_idx_xjal + 12 ];
    flow8 = _a_xjal[ _a_idx_xjal + 13 ];
    LOST_A = _a_xjal[ _a_idx_xjal + 14 ];
    flow1 = _a_xjal[ _a_idx_xjal + 15 ];
    _a_idx_xjal += 16;
    super.putPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

@AnyLogicInternalCodegenAPI
public void assignInitialConditions_xjal() {
    super.assignInitialConditions_xjal();
    _assign_CR_Formula_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_INTERACTIVE_Formula_xjal();
    _assign_SUCCESS_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_STATIC_Formula_xjal();
    _assign_POPUP_Formula_xjal();
    _assign_PICTURES_Formula_xjal();
    _assign_GAMES_Formula_xjal();
}

```

```

    _assign_VIDEO_Formula_xjal();
    _assign_COLORS_Formula_xjal();
    _assign_ANNOUNCE_Formula_xjal();
    _assign_ANIMATED_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void setupInitialConditions_xjal(Class<?> callerClass) {
    if (callerClass != Main.class) {
        return;
    }
    if (getInitialAlgebraicFlatEquationsCount_xjal() > 0) {
        SDIntegrationManager integrationManagerForInitialConditions = new
SDIntegrationManager( 0, getInitialAlgebraicFlatEquationsCount_xjal(),
getInitialFormulaFlatEquationsCount_xjal() );
        integrationManagerForInitialConditions.doStep( this, 0, 0.1, true );
    } else {
        assignInitialConditions_xjal();
    }
}

@AnyLogicInternalCodegenAPI
public void _assign_CR_Formula_xjal() {
    CR =
100
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_S_Formula_xjal() {
    RATE_S =
CR * 0.084
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_V_Formula_xjal() {
    RATE_V =
CR * 0.12
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_C_Formula_xjal() {
    RATE_C =
CR * 0.11
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_G_Formula_xjal() {
    RATE_G =
CR * 0.093
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_P_Formula_xjal() {
    RATE_P =
CR * 0.013
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_PC_Formula_xjal() {
    RATE_PC =
CR * 0.10
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_AN_Formula_xjal() {
    RATE_AN =
CR * 0.075
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_A_Formula_xjal() {
    RATE_A =
CR * 0.149
;
}
@AnyLogicInternalCodegenAPI
public void _assign_INTERACTIVE_Formula_xjal() {
    INTERACTIVE =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_SUCCESS_Formula_xjal() {
    SUCCESS =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_I_Formula_xjal() {
    RATE_I =
CR * 0.139
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_I_Formula_xjal() {
    LOST_I =
INTERACTIVE * 0.25 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow5_Formula_xjal() {
    flow5 =
INTERACTIVE * 0.75 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RETURNED_Formula_xjal() {
    RETURNED =
SUCCESS * 1
;
}
@AnyLogicInternalCodegenAPI
public void _assign_STATIC_Formula_xjal() {
    STATIC =
RATE_S - LOST_S - flow + 0 + parameter_S
;
}
@AnyLogicInternalCodegenAPI
public void _assign_POPUP_Formula_xjal() {
    POPUP =
RATE_P - flow2 - LOST_P + 0 + parameter_P
;
}
@AnyLogicInternalCodegenAPI

```



```

    public void _assign_PICTURES_Formula_xjal() {
        PICTURES =
RATE_PC - flow3 - LOST_PC + 0 + parameter_PC
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_GAMES_Formula_xjal() {
        GAMES =
RATE_G - flow7 - LOST_G + 0 + parameter_G
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_VIDEO_Formula_xjal() {
        VIDEO =
RATE_V - flow6 - LOST_V + 0 + parameter_V
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_COLORS_Formula_xjal() {
        COLORS =
RATE_C - LOST_C - flow8 + 0 + parameter_C
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANNOUNCE_Formula_xjal() {
        ANNOUNCE =
RATE_AN - flow4 - LOST_AN + 0 + parameter_AN
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANIMATED_Formula_xjal() {
        ANIMATED =
RATE_A - flow1 - LOST_A + 0 + parameter_A
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_S_Formula_xjal() {
        LOST_S =
STATIC * 0.55 + parameter_S
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow_Formula_xjal() {
        flow =
STATIC * 0.55 + parameter_S
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_P_Formula_xjal() {
        LOST_P =
POPUP * 0.3 + parameter_P
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow2_Formula_xjal() {
        flow2 =
POPUP * 0.7 + parameter_P
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_PC_Formula_xjal() {
        LOST_PC =

```

```

PICTURES * 0.45 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow3_Formula_xjal() {
    flow3 =
PICTURES * 0.55 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_AN_Formula_xjal() {
    LOST_AN =
ANNOUNCE * 0.6 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow4_Formula_xjal() {
    flow4 =
ANNOUNCE * 0.40 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_V_Formula_xjal() {
    LOST_V =
VIDEO * 0.35 + parameter_V
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow6_Formula_xjal() {
    flow6 =
VIDEO * 0.75
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_G_Formula_xjal() {
    LOST_G =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow7_Formula_xjal() {
    flow7 =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_C_Formula_xjal() {
    LOST_C =
COLORS * 0.4
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow8_Formula_xjal() {
    flow8 =
COLORS * 0.6 + parameter_C
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_A_Formula_xjal() {
    LOST_A =
ANIMATED * 0.20 + parameter_A
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_flow1_Formula_xjal() {
    flow1 =
ANIMATED * 0.8 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void formulasExecute_xjal() {
    super.formulasExecute_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_LOST_S_Formula_xjal();
    _assign_flow_Formula_xjal();
    _assign_LOST_P_Formula_xjal();
    _assign_flow2_Formula_xjal();
    _assign_LOST_PC_Formula_xjal();
    _assign_flow3_Formula_xjal();
    _assign_LOST_AN_Formula_xjal();
    _assign_flow4_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_LOST_V_Formula_xjal();
    _assign_flow6_Formula_xjal();
    _assign_LOST_G_Formula_xjal();
    _assign_flow7_Formula_xjal();
    _assign_LOST_C_Formula_xjal();
    _assign_flow8_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_LOST_A_Formula_xjal();
    _assign_flow1_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void getRightPart_xjal( double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal ) {
    _get_CR_Derivative_xjal( _d_xjal, _d_idx_xjal + 0 );
    _get_STATIC_Derivative_xjal( _d_xjal, _d_idx_xjal + 1 );
    _get_INTERACTIVE_Derivative_xjal( _d_xjal, _d_idx_xjal + 2 );
    _get_POPUP_Derivative_xjal( _d_xjal, _d_idx_xjal + 3 );
    _get_PICTURES_Derivative_xjal( _d_xjal, _d_idx_xjal + 4 );
    _get_GAMES_Derivative_xjal( _d_xjal, _d_idx_xjal + 5 );
    _get_VIDEO_Derivative_xjal( _d_xjal, _d_idx_xjal + 6 );
    _get_COLORS_Derivative_xjal( _d_xjal, _d_idx_xjal + 7 );
    _get_ANNOUNCE_Derivative_xjal( _d_xjal, _d_idx_xjal + 8 );
    _get_SUCCESS_Derivative_xjal( _d_xjal, _d_idx_xjal + 9 );
    _get_ANIMATED_Derivative_xjal( _d_xjal, _d_idx_xjal + 10 );
    _d_idx_xjal += 11;
    super.getRightPart_xjal( _d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void getRightPartForInitialConditions_xjal( double[] _a_xjal, int
_a_idx_xjal ) {
    _get_LOST_S_Equation_xjal( _a_xjal, _a_idx_xjal + 0 );

```

```

    _get_flow_Equation_xjal( _a_xjal, _a_idx_xjal + 1 );
    _get_LOST_P_Equation_xjal( _a_xjal, _a_idx_xjal + 2 );
    _get_flow2_Equation_xjal( _a_xjal, _a_idx_xjal + 3 );
    _get_LOST_PC_Equation_xjal( _a_xjal, _a_idx_xjal + 4 );
    _get_flow3_Equation_xjal( _a_xjal, _a_idx_xjal + 5 );
    _get_LOST_AN_Equation_xjal( _a_xjal, _a_idx_xjal + 6 );
    _get_flow4_Equation_xjal( _a_xjal, _a_idx_xjal + 7 );
    _get_LOST_V_Equation_xjal( _a_xjal, _a_idx_xjal + 8 );
    _get_flow6_Equation_xjal( _a_xjal, _a_idx_xjal + 9 );
    _get_LOST_G_Equation_xjal( _a_xjal, _a_idx_xjal + 10 );
    _get_flow7_Equation_xjal( _a_xjal, _a_idx_xjal + 11 );
    _get_LOST_C_Equation_xjal( _a_xjal, _a_idx_xjal + 12 );
    _get_flow8_Equation_xjal( _a_xjal, _a_idx_xjal + 13 );
    _get_LOST_A_Equation_xjal( _a_xjal, _a_idx_xjal + 14 );
    _get_flow1_Equation_xjal( _a_xjal, _a_idx_xjal + 15 );
    _a_idx_xjal += 16;
    super.getRightPartForInitialConditions_xjal( _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void _get_CR_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RETURNED - RATE_S - RATE_I- RATE_V- RATE_C- RATE_G- RATE_P- RATE_PC- RATE_AN- RATE_A
;
}
@AnyLogicInternalCodegenAPI
public void _get_STATIC_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_S - LOST_S - flow
;
}
@AnyLogicInternalCodegenAPI
public void _get_INTERACTIVE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_I - LOST_I - flow5
;
}
@AnyLogicInternalCodegenAPI
public void _get_POPUP_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_P - LOST_P - flow2
;
}
@AnyLogicInternalCodegenAPI
public void _get_PICTURES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_PC - LOST_PC - flow3
;
}
@AnyLogicInternalCodegenAPI
public void _get_GAMES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_G - LOST_G - flow7
;
}
@AnyLogicInternalCodegenAPI
public void _get_VIDEO_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_V - LOST_V - flow6
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _get_COLORS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_C - LOST_C - flow8
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANNOUNCE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_AN - LOST_AN - flow4
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_SUCCESS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+flow + flow2 + flow3+ flow4+ flow5+ flow6+ flow7+ flow8+ flow1 - RETURNED
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANIMATED_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_A - LOST_A - flow1
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_S_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- LOST_S;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- flow;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_P_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.3 + parameter_P
- LOST_P;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow2_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.7 + parameter_P
- flow2;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_PC_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.45 + parameter_PC
- LOST_PC;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow3_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.55 + parameter_PC
- flow3;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_AN_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =

```

```

ANNOUNCE * 0.6 + parameter_AN
- LOST_AN;
}
@AnyLogicInternalCodegenAPI
public void _get_flow4_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.40 + parameter_AN
- flow4;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_V_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.35 + parameter_V
- LOST_V;
}
@AnyLogicInternalCodegenAPI
public void _get_flow6_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.75
- flow6;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_G_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- LOST_G;
}
@AnyLogicInternalCodegenAPI
public void _get_flow7_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- flow7;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_C_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.4
- LOST_C;
}
@AnyLogicInternalCodegenAPI
public void _get_flow8_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.6 + parameter_C
- flow8;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_A_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.20 + parameter_A
- LOST_A;
}
@AnyLogicInternalCodegenAPI
public void _get_flow1_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.8 + parameter_A
- flow1;
}
@AnyLogicInternalCodegenAPI
protected SDIntegrationManager integrationManager_xjal = null;

@AnyLogicInternalCodegenAPI
public SDIntegrationManager getIntegrationManager_xjal() {

```

```

        if (integrationManager_xjal == null) {
            integrationManager_xjal = new
SDIntegrationManager( getDifferentialFlatEquationsCount_xjal(),
getRuntimeAlgebraicFlatEquationsCount_xjal(),
getRuntimeFormulaFlatEquationsCount_xjal() );
        }
        return integrationManager_xjal;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getDifferentialFlatEquationsCount_xjal() {
        return super.getDifferentialFlatEquationsCount_xjal() + 11;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getRuntimeFormulaFlatEquationsCount_xjal() {
        return super.getRuntimeFormulaFlatEquationsCount_xjal() + 28;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialAlgebraicFlatEquationsCount_xjal() {
        return super.getInitialAlgebraicFlatEquationsCount_xjal() + 16;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialFormulaFlatEquationsCount_xjal() {
        return super.getInitialFormulaFlatEquationsCount_xjal() + 23;
    }

    @AnyLogicInternalCodegenAPI
    private static Map<String, IElementDescriptor> elementDescriptors_xjal =
createElementDescriptors( Main.class );

    @AnyLogicInternalCodegenAPI
    @Override
    public Map<String, IElementDescriptor> getElementDescriptors() {
        return elementDescriptors_xjal;
    }
    @AnyLogicCustomProposalPriority(type =
AnyLogicCustomProposalPriority.Type.STATIC_ELEMENT)
    public static final Scale scale = new Scale( 10.0 );

    @Override
    public Scale getScale() {
        return scale;
    }

    // Events

    @AnyLogicInternalCodegenAPI
    public EventTimeout _chart_autoUpdateEvent_xjal = new EventTimeout(this);
    @AnyLogicInternalCodegenAPI
    public EventTimeout _autoCreatedDS_xjal = new EventTimeout(this);

    @Override
    @AnyLogicInternalCodegenAPI
    public String getNameOf( EventTimeout _e ) {

```

```

        if( _e == _chart_autoUpdateEvent_xjal ) return "chart auto update event";
        if( _e == _autoCreatedDS_xjal ) return "Auto-created DataSets auto update event";
        return super.getNameOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public EventTimeout.Mode getModeOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        if ( _e == _autoCreatedDS_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        return super.getModeOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double getFirstOccurrenceTime( EventTimeout _e ) {
        double _t;
        if ( _e == _chart_autoUpdateEvent_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if ( _e == _autoCreatedDS_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.getFirstOccurrenceTime( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double evaluateTimeoutOf( EventTimeout _e ) {
        double _t;
        if( _e == _chart_autoUpdateEvent_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if( _e == _autoCreatedDS_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.evaluateTimeoutOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public void executeActionOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) {
            chart.updateData();
            return;
        }
    }

```



```

if ( _e == _autoCreatedDS_xjal ) {
    _ds_RATE_S.update();
    _ds_RATE_I.update();
    _ds_RATE_V.update();
    _ds_RATE_C.update();
    _ds_RATE_G.update();
    _ds_RATE_P.update();
    _ds_RATE_PC.update();
    _ds_RATE_AN.update();
    _ds_LOST_S.update();
    _ds_flow.update();
    _ds_LOST_P.update();
    _ds_flow2.update();
    _ds_LOST_PC.update();
    _ds_flow3.update();
    _ds_LOST_AN.update();
    _ds_flow4.update();
    _ds_LOST_I.update();
    _ds_flow5.update();
    _ds_LOST_V.update();
    _ds_flow6.update();
    _ds_LOST_G.update();
    _ds_flow7.update();
    _ds_LOST_C.update();
    _ds_flow8.update();
    _ds_RETURNED.update();
    _ds_RATE_A.update();
    _ds_LOST_A.update();
    _ds_flow1.update();
    _ds_CR.update();
    _ds_STATIC.update();
    _ds_INTERACTIVE.update();
    _ds_POPUP.update();
    _ds_PICTURES.update();
    _ds_GAMES.update();
    _ds_VIDEO.update();
    _ds_COLORS.update();
    _ds_ANNOUNCE.update();
    _ds_SUCCESS.update();
    _ds_ANIMATED.update();
    return;
}
super.executeActionOf( _e );
}

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _STATECHART_COUNT_xjal = 0;

/**
 * Auto-created data set(s) for RATE_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_S );
    }
}

```

```

        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_P
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override

```

```

        public void update( DataSet _d ) {
            if ( time() == _lastUpdateTime ) { return; }
            _d.add( time(), Main.this.RATE_P );
            _lastUpdateTime = time();
        }
    } );
/**
 * Auto-created data set(s) for RATE_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_AN );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_P
 */
@AnyLogicInternalCodegenAPI

```

```

public DataSet _ds_LOST_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow2
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow2 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow2 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow3
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow3 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow3 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_AN );
        _lastUpdateTime = time();
    }
} );
/**

```

```

* Auto-created data set(s) for flow4
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow4 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow4 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_I
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_I );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow5
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow5 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow5 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_V
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_V );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow6
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow6 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow6 );
        _lastUpdateTime = time();
    }
} );

```

```

    }
} );
/**
 * Auto-created data set(s) for LOST_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow7
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow7 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow7 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow8
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow8 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow8 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RETURNED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RETURNED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {

```

```

        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RETURNED );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow1
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow1 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow1 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for CR
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_CR = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.CR );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for STATIC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_STATIC = new DataSet( 100, new DataUpdater_xjal() {

```

```

    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.STATIC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for INTERACTIVE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_INTERACTIVE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.INTERACTIVE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for POPUP
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_POPUP = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.POPUP );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for PICTURES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_PICTURES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.PICTURES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for GAMES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_GAMES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.GAMES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for VIDEO

```



```

*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_VIDEO = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.VIDEO );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for COLORS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_COLORS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.COLORS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANNOUNCE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANNOUNCE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANNOUNCE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for SUCCESS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_SUCCESS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.SUCCESS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANIMATED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANIMATED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANIMATED );
        _lastUpdateTime = time();
    }
}

```

```

    } );
    // View areas
    public ViewArea _origin_VA = new ViewArea( this, "[Origin]", 0, 0, 1000.0, 600.0 );
    @Override
    @AnyLogicInternalCodegenAPI
    public int getViewAreas(Map<String, ViewArea> _output) {
        if ( _output != null ) {
            _output.put( "_origin_VA", this._origin_VA );
        }
        return 1 + super.getViewAreas( _output );
    }
    @AnyLogicInternalCodegenAPI
    protected static final int _chart = 1;

    /** Internal constant, shouldn't be accessed by user */
    @AnyLogicInternalCodegenAPI
    protected static final int _SHAPE_NEXT_ID_xjal = 2;

    @AnyLogicInternalCodegenAPI
    public boolean isPublicPresentationDefined() {
        return true;
    }

    @AnyLogicInternalCodegenAPI
    public boolean isEmbeddedAgentPresentationVisible( Agent _a ) {
        return super.isEmbeddedAgentPresentationVisible( _a );
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem0Value() {
        return
ANIMATED
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem1Value() {
        return
STATIC
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem2Value() {
        return
POPUP
;
    }

    /**
     * <i>This method should not be called by user</i>
     */

```

```

    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem3Value() {
        return
INTERACTIVE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem4Value() {
        return
PICTURES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem5Value() {
        return
GAMES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem6Value() {
        return
VIDEO
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem7Value() {
        return
ANNOUNCE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem8Value() {
        return
COLORS
;
    }
    protected BarChart chart;
    @AnyLogicInternalCodegenAPI
    private void _createPersistentElementsBP0_xjal() {
    }

    @AnyLogicInternalCodegenAPI

```

```

private void _createPersistentElementsAP0_xjal() {
    {
        List<DataItem> _items = new ArrayList<DataItem>( 9 );
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem0Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem1Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem2Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem3Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem4Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem5Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem6Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem7Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem8Value() );
            }
        });
        List<String> _titles = new ArrayList<String>( 9 );
        _titles.add( "ANIMATED" );
        _titles.add( "STATIC" );
        _titles.add( "POPOP" );
        _titles.add( "INTERACTIVE" );
    }
}

```

```

_titles.add( "PICTURES" );
_titles.add( "GAMES" );
_titles.add( "VIDEO" );
_titles.add( "ANNOUNCE" );
_titles.add( "COLORS" );
List<Color> _colors = new ArrayList<Color>( 9 );
_colors.add( limeGreen );
_colors.add( slateGray );
_colors.add( darkMagenta );
_colors.add( orange );
_colors.add( darkKhaki );
_colors.add( mediumTurquoise );
_colors.add( lightSlateBlue );
_colors.add( deepPink );
_colors.add( darkOrange );
chart = new BarChart(
    Main.this, true, 20.0, 700.0,
    1420.0, 580.0,
    null, null,
    50.0, 20.0,
    1330.0, 460.0, white, black, black,
    80.0, Chart.SOUTH,
    Chart.NORTH, Chart.SCALE_AUTO,
    0, 0, 0.8,
    Chart.GRID_DEFAULT,
    darkGray, darkGray, _items, _titles, _colors );
}
}

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsBS0_xjal() {
}

// Static initialization of persistent elements
{
_createPersistentElementsBP0_xjal();
}
protected ShapeTopLevelPresentationGroup presentation;
protected ShapeModelElementsGroup icon;

@Override
@AnyLogicInternalCodegenAPI
public ShapeTopLevelPresentationGroup getPresentationShape() {
return presentation;
}

@Override
@AnyLogicInternalCodegenAPI
public ShapeModelElementsGroup getModelElementsShape() {
return icon;
}

/**
 * Constructor
 */
public Main( Engine engine, Agent owner, AgentList<? extends Main> ownerPopulation )
{
super( engine, owner, ownerPopulation );
instantiateBaseStructureThis_xjal();
}

```

```

@AnyLogicInternalCodegenAPI
public void onOwnerChanged_xjal() {
    super.onOwnerChanged_xjal();
    setupReferences_xjal();
}

@AnyLogicInternalCodegenAPI
public void instantiateBaseStructure_xjal() {
    super.instantiateBaseStructure_xjal();
    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
private void instantiateBaseStructureThis_xjal() {
    setupReferences_xjal();
    // Registering in Engine continuous part
    getEngine().registerAgentWithEquations( this );
}

@AnyLogicInternalCodegenAPI
private void setupReferences_xjal() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main( double parameter_S, double parameter_P, double parameter_PC, double
parameter_AN, double parameter_I, double parameter_V, double parameter_G, double
parameter_C, double parameter_A ) {
    markParametersAreSet();
    this.parameter_S = parameter_S;
    this.parameter_P = parameter_P;
    this.parameter_PC = parameter_PC;
    this.parameter_AN = parameter_AN;
    this.parameter_I = parameter_I;
    this.parameter_V = parameter_V;
    this.parameter_G = parameter_G;
    this.parameter_C = parameter_C;
    this.parameter_A = parameter_A;
}

@Override
@AnyLogicInternalCodegenAPI
public void doCreate() {
    super.doCreate();
    // Assigning initial values for plain variables
    setupPlainVariables_Main_xjal();
    // Dynamic initialization of persistent elements
    _createPersistentElementsAP0_xjal();
    presentation = new ShapeTopLevelPresentationGroup( Main.this, true, 0, 0, 0, 0 ,
chart );
    icon = new ShapeModelElementsGroup( Main.this, getElementProperty(
"modell1.Main.icon", IElementDescriptor.MODEL_ELEMENT_DESCRIPTOR ) );
}

```

```

    icon.setIconOffsets( 0.0, 0.0 );
    // Port connectors with non-replicated objects
    // Creating replicated embedded objects
    setupInitialConditions_xjal( Main.class );
    // Dynamic initialization of persistent elements
    _createPersistentElementsBS0_xjal();
}

@Override
@AnyLogicInternalCodegenAPI
public void doStart() {
    super.doStart();
    _chart_autoUpdateEvent_xjal.start();
    _autoCreatedDS_xjal.start();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
public void setupPlainVariables_xjal() {
    setupPlainVariables_Main_xjal();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
private void setupPlainVariables_Main_xjal() {
}

// User API -----
@AnyLogicInternalCodegenAPI
public static LinkToAgentAnimationSettings _connections_commonAnimationSettings_xjal
= new LinkToAgentAnimationSettingsImpl( false, black, 1.0, LINE_STYLE_SOLID,
ARROW_NONE, 0.0 );

public LinkToAgentCollection<Agent, Agent> connections = new
LinkToAgentStandardImpl<Agent, Agent>(this, _connections_commonAnimationSettings_xjal);
@Override
public LinkToAgentCollection<? extends Agent, ? extends Agent>
getLinkToAgentStandard_xjal() {
    return connections;
}

@AnyLogicInternalCodegenAPI
public void drawLinksToAgents(boolean _underAgents_xjal, LinkToAgentAnimator
_animator_xjal) {
    super.drawLinksToAgents(_underAgents_xjal, _animator_xjal);
    if ( _underAgents_xjal ) {
        _animator_xjal.drawLink( this, connections, true, true );
    }
}

public AgentList<? extends Main> getPopulation() {
    return (AgentList<? extends Main>) super.getPopulation();
}

```

```

}

public List<? extends Main> agentsInRange( double distance ) {
    return (List<? extends Main>) super.agentsInRange( distance );
}

@Override
@AnyLogicInternalCodegenAPI
public boolean isLoggingToDB(EventOriginator _e) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return false;
    if ( _e == _autoCreatedDS_xjal ) return false;
    return super.isLoggingToDB( _e );
}

@AnyLogicInternalCodegenAPI
public void onDestroy() {
    _chart_autoUpdateEvent_xjal.onDestroy();
    _autoCreatedDS_xjal.onDestroy();
    // Unregistering in Engine continuous part
    getEngine().unregisterAgentWithEquations( this );
    _ds_RATE_S.destroyUpdater_xjal();
    _ds_RATE_I.destroyUpdater_xjal();
    _ds_RATE_V.destroyUpdater_xjal();
    _ds_RATE_C.destroyUpdater_xjal();
    _ds_RATE_G.destroyUpdater_xjal();
    _ds_RATE_P.destroyUpdater_xjal();
    _ds_RATE_PC.destroyUpdater_xjal();
    _ds_RATE_AN.destroyUpdater_xjal();
    _ds_LOST_S.destroyUpdater_xjal();
    _ds_flow.destroyUpdater_xjal();
    _ds_LOST_P.destroyUpdater_xjal();
    _ds_flow2.destroyUpdater_xjal();
    _ds_LOST_PC.destroyUpdater_xjal();
    _ds_flow3.destroyUpdater_xjal();
    _ds_LOST_AN.destroyUpdater_xjal();
    _ds_flow4.destroyUpdater_xjal();
    _ds_LOST_I.destroyUpdater_xjal();
    _ds_flow5.destroyUpdater_xjal();
    _ds_LOST_V.destroyUpdater_xjal();
    _ds_flow6.destroyUpdater_xjal();
    _ds_LOST_G.destroyUpdater_xjal();
    _ds_flow7.destroyUpdater_xjal();
    _ds_LOST_C.destroyUpdater_xjal();
    _ds_flow8.destroyUpdater_xjal();
    _ds_RETURNED.destroyUpdater_xjal();
    _ds_RATE_A.destroyUpdater_xjal();
    _ds_LOST_A.destroyUpdater_xjal();
    _ds_flow1.destroyUpdater_xjal();
    _ds_CR.destroyUpdater_xjal();
    _ds_STATIC.destroyUpdater_xjal();
    _ds_INTERACTIVE.destroyUpdater_xjal();
    _ds_POPUP.destroyUpdater_xjal();
    _ds_PICTURES.destroyUpdater_xjal();
    _ds_GAMES.destroyUpdater_xjal();
    _ds_VIDEO.destroyUpdater_xjal();
    _ds_COLORS.destroyUpdater_xjal();
    _ds_ANNOUNCE.destroyUpdater_xjal();
    _ds_SUCCESS.destroyUpdater_xjal();
    _ds_ANIMATED.destroyUpdater_xjal();
    super.onDestroy();
}

```



```

package model1;

import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayDeque;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.Collection;
import java.util.Collections;
import java.util.Comparator;
import java.util.Currency;
import java.util.Date;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Hashtable;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import java.util.Map;
import java.util.PriorityQueue;
import java.util.Random;
import java.util.Set;
import java.util.SortedMap;
import java.util.SortedSet;
import java.util.Stack;
import java.util.Timer;
import java.util.TreeMap;
import java.util.TreeSet;
import java.util.Vector;
import java.awt.Color;
import java.awt.Font;
import com.anylogic.engine.connectivity.ResultSet;
import com.anylogic.engine.connectivity.Statement;
import com.anylogic.engine.elements.*;
import com.anylogic.engine.markup.Network;
import com.anylogic.engine.Position;
import com.anylogic.engine.markup.PedFlowStatistics;
import com.anylogic.engine.markup.DensityMap;

import static java.lang.Math.*;
import static com.anylogic.engine.UtilitiesArray.*;
import static com.anylogic.engine.UtilitiesCollection.*;
import static com.anylogic.engine.presentation.UtilitiesColor.*;
import static com.anylogic.engine.HyperArray.*;

import com.anylogic.engine.*;
import com.anylogic.engine.analysis.*;
import com.anylogic.engine.connectivity.*;
import com.anylogic.engine.database.*;
import com.anylogic.engine.gis.*;
import com.anylogic.engine.markup.*;
import com.anylogic.engine.routing.*;
import com.anylogic.engine.presentation.*;
import com.anylogic.engine.gui.*;

```

```

import java.awt.geom.Arc2D;

public class Main extends Agent
{
    // Parameters

    public
double    parameter_S;

    /**
     * Returns default value for parameter <code>parameter_S</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_S_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_S( double parameter_S ) {
        if (parameter_S == this.parameter_S) {
            return;
        }
        double _oldValue_xjal = this.parameter_S;
        this.parameter_S = parameter_S;
        onChange_parameter_S_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_S.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_S()</code> method instead.
     */
    protected void onChange_parameter_S() {
        onChange_parameter_S_xjal( parameter_S );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_S_xjal( double oldValue ) {

    }

    public
double    parameter_P;

    /**
     * Returns default value for parameter <code>parameter_P</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_P_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_P( double parameter_P ) {
        if (parameter_P == this.parameter_P) {
            return;
        }
    }
}

```

```

    }
    double _oldValue_xjal = this.parameter_P;
    this.parameter_P = parameter_P;
    onChange_parameter_P_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_P.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_P()</code> method instead.
 */
protected void onChange_parameter_P() {
    onChange_parameter_P_xjal( parameter_P );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_P_xjal( double oldValue ) {

public
double parameter_PC;

/**
 * Returns default value for parameter <code>parameter_PC</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_PC_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_PC( double parameter_PC ) {
    if (parameter_PC == this.parameter_PC) {
        return;
    }
    double _oldValue_xjal = this.parameter_PC;
    this.parameter_PC = parameter_PC;
    onChange_parameter_PC_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_PC.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_PC()</code> method instead.
 */
protected void onChange_parameter_PC() {
    onChange_parameter_PC_xjal( parameter_PC );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_PC_xjal( double oldValue ) {

```

```

public
double parameter_AN;

/**
 * Returns default value for parameter parameter_AN.
 * This method should not be called by user
 */
@AnyLogicInternalCodegenAPI
public double _parameter_AN_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_AN( double parameter_AN ) {
    if (parameter_AN == this.parameter_AN) {
        return;
    }
    double _oldValue_xjal = this.parameter_AN;
    this.parameter_AN = parameter_AN;
    onChange_parameter_AN_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_AN.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call set_parameter_AN() method instead.
 */
protected void onChange_parameter_AN() {
    onChange_parameter_AN_xjal( parameter_AN );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_AN_xjal( double oldValue ) {
}

public
double parameter_I;

/**
 * Returns default value for parameter parameter_I.
 * This method should not be called by user
 */
@AnyLogicInternalCodegenAPI
public double _parameter_I_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_I( double parameter_I ) {
    if (parameter_I == this.parameter_I) {
        return;
    }
    double _oldValue_xjal = this.parameter_I;
    this.parameter_I = parameter_I;
    onChange_parameter_I_xjal( _oldValue_xjal );
    onChange();
}

```

```

/**
 * Calls "On change" action for parameter parameter_I.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_I()</code> method instead.
 */
protected void onChange_parameter_I() {
    onChange_parameter_I_xjal( parameter_I );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_I_xjal( double oldValue ) {
}

public
double parameter_V;

/**
 * Returns default value for parameter <code>parameter_V</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_V_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_V( double parameter_V ) {
    if (parameter_V == this.parameter_V) {
        return;
    }
    double _oldValue_xjal = this.parameter_V;
    this.parameter_V = parameter_V;
    onChange_parameter_V_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_V.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_V()</code> method instead.
 */
protected void onChange_parameter_V() {
    onChange_parameter_V_xjal( parameter_V );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_V_xjal( double oldValue ) {
}

public
double parameter_G;

/**
 * Returns default value for parameter <code>parameter_G</code>.
 * <i>This method should not be called by user</i>
 */

```

```

@AnyLogicInternalCodegenAPI
public double _parameter_G_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_G( double parameter_G ) {
    if (parameter_G == this.parameter_G) {
        return;
    }
    double _oldValue_xjal = this.parameter_G;
    this.parameter_G = parameter_G;
    onChange_parameter_G_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_G.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_G()</code> method instead.
 */
protected void onChange_parameter_G() {
    onChange_parameter_G_xjal( parameter_G );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_G_xjal( double oldValue ) {
}

public
double parameter_C;

/**
 * Returns default value for parameter <code>parameter_C</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_C_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_C( double parameter_C ) {
    if (parameter_C == this.parameter_C) {
        return;
    }
    double _oldValue_xjal = this.parameter_C;
    this.parameter_C = parameter_C;
    onChange_parameter_C_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_C.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_C()</code> method instead.
 */

```

```

protected void onChange_parameter_C() {
    onChange_parameter_C_xjal( parameter_C );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_C_xjal( double oldValue ) {
}

public
double parameter_A;

/**
 * Returns default value for parameter <code>parameter_A</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_A_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_A( double parameter_A ) {
    if (parameter_A == this.parameter_A) {
        return;
    }
    double _oldValue_xjal = this.parameter_A;
    this.parameter_A = parameter_A;
    onChange_parameter_A_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_A.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_A()</code> method instead.
 */
protected void onChange_parameter_A() {
    onChange_parameter_A_xjal( parameter_A );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_A_xjal( double oldValue ) {
}

@Override
public void setParametersToDefaultValues() {
    super.setParametersToDefaultValues();
    parameter_S = _parameter_S_DefaultValue_xjal();
    parameter_P = _parameter_P_DefaultValue_xjal();
    parameter_PC = _parameter_PC_DefaultValue_xjal();
    parameter_AN = _parameter_AN_DefaultValue_xjal();
    parameter_I = _parameter_I_DefaultValue_xjal();
    parameter_V = _parameter_V_DefaultValue_xjal();
    parameter_G = _parameter_G_DefaultValue_xjal();
    parameter_C = _parameter_C_DefaultValue_xjal();
    parameter_A = _parameter_A_DefaultValue_xjal();
}

```

```

@Override
public boolean setParameter(String _name_xjal, Object _value_xjal, boolean
_callOnChange_xjal) {
    switch ( _name_xjal ) {
        case "parameter_S":
            if ( _callOnChange_xjal ) {
                set_parameter_S( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_S = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_P":
            if ( _callOnChange_xjal ) {
                set_parameter_P( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_P = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_PC":
            if ( _callOnChange_xjal ) {
                set_parameter_PC( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_PC = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_AN":
            if ( _callOnChange_xjal ) {
                set_parameter_AN( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_AN = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_I":
            if ( _callOnChange_xjal ) {
                set_parameter_I( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_I = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_V":
            if ( _callOnChange_xjal ) {
                set_parameter_V( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_V = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_G":
            if ( _callOnChange_xjal ) {
                set_parameter_G( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_G = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_C":
            if ( _callOnChange_xjal ) {
                set_parameter_C( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_C = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_A":
            if ( _callOnChange_xjal ) {

```



```

        set_parameter_A( ((Number) _value_xjal).doubleValue() );
    } else {
        parameter_A = ((Number) _value_xjal).doubleValue();
    }
    return true;
default:
    return super.setParameter( _name_xjal, _value_xjal, _callOnChange_xjal );
}
}

@Override
public <T> T getParameter(String _name_xjal) {
    Object _result_xjal;
    switch ( _name_xjal ) {
    case "parameter_S": _result_xjal = parameter_S; break;
    case "parameter_P": _result_xjal = parameter_P; break;
    case "parameter_PC": _result_xjal = parameter_PC; break;
    case "parameter_AN": _result_xjal = parameter_AN; break;
    case "parameter_I": _result_xjal = parameter_I; break;
    case "parameter_V": _result_xjal = parameter_V; break;
    case "parameter_G": _result_xjal = parameter_G; break;
    case "parameter_C": _result_xjal = parameter_C; break;
    case "parameter_A": _result_xjal = parameter_A; break;
    default: _result_xjal = super.getParameter( _name_xjal ); break;
    }
    return (T) _result_xjal;
}

@AnyLogicInternalCodegenAPI
private static String[] _parameterNames_xjal;

@Override
public String[] getParameterNames() {
    String[] result = _parameterNames_xjal;
    if (result == null) {
        List<String> list = new ArrayList<>( Arrays.asList(
super.getParameterNames() ) );
        list.add( "parameter_S" );
        list.add( "parameter_P" );
        list.add( "parameter_PC" );
        list.add( "parameter_AN" );
        list.add( "parameter_I" );
        list.add( "parameter_V" );
        list.add( "parameter_G" );
        list.add( "parameter_C" );
        list.add( "parameter_A" );
        result = list.toArray( new String[ list.size() ] );
        _parameterNames_xjal = result;
    }
    return result;
}

// Dynamic (Flow/Auxiliary/Stock) Variables

public double RATE_S;

public double RATE_I;

public double RATE_V;

public double RATE_C;

```

```
public double RATE_G;
public double RATE_P;
public double RATE_PC;
public double RATE_AN;
public double LOST_S;
public double flow;
public double LOST_P;
public double flow2;
public double LOST_PC;
public double flow3;
public double LOST_AN;
public double flow4;
public double LOST_I;
public double flow5;
public double LOST_V;
public double flow6;
public double LOST_G;
public double flow7;
public double LOST_C;
public double flow8;
public double RETURNED;
public double RATE_A;
public double LOST_A;
public double flow1;
public double CR;
public double STATIC;
public double INTERACTIVE;
public double POPUP;
public double PICTURES;
public double GAMES;
public double VIDEO;
```

```

public double COLORS;

public double ANNOUNCE;

public double SUCCESS;

public double ANIMATED;
/**
 * Writes model variables into given arrays
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    _d_xjal[ _d_idx_xjal + 0 ] = CR;
    _d_xjal[ _d_idx_xjal + 1 ] = STATIC;
    _d_xjal[ _d_idx_xjal + 2 ] = INTERACTIVE;
    _d_xjal[ _d_idx_xjal + 3 ] = POPUP;
    _d_xjal[ _d_idx_xjal + 4 ] = PICTURES;
    _d_xjal[ _d_idx_xjal + 5 ] = GAMES;
    _d_xjal[ _d_idx_xjal + 6 ] = VIDEO;
    _d_xjal[ _d_idx_xjal + 7 ] = COLORS;
    _d_xjal[ _d_idx_xjal + 8 ] = ANNOUNCE;
    _d_xjal[ _d_idx_xjal + 9 ] = SUCCESS;
    _d_xjal[ _d_idx_xjal + 10 ] = ANIMATED;
    _d_idx_xjal += 11;
    super.getPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * Writes given arrays to model variables
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    CR = _d_xjal[ _d_idx_xjal + 0 ];
    STATIC = _d_xjal[ _d_idx_xjal + 1 ];
    INTERACTIVE = _d_xjal[ _d_idx_xjal + 2 ];
    POPUP = _d_xjal[ _d_idx_xjal + 3 ];
    PICTURES = _d_xjal[ _d_idx_xjal + 4 ];
    GAMES = _d_xjal[ _d_idx_xjal + 5 ];
    VIDEO = _d_xjal[ _d_idx_xjal + 6 ];
    COLORS = _d_xjal[ _d_idx_xjal + 7 ];
    ANNOUNCE = _d_xjal[ _d_idx_xjal + 8 ];
    SUCCESS = _d_xjal[ _d_idx_xjal + 9 ];
    ANIMATED = _d_xjal[ _d_idx_xjal + 10 ];
    _d_idx_xjal += 11;
    super.putPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * This function gets phase vector from initial conditions (is used while solving
algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    _a_xjal[ _a_idx_xjal + 0 ] = LOST_S;
    _a_xjal[ _a_idx_xjal + 1 ] = flow;
    _a_xjal[ _a_idx_xjal + 2 ] = LOST_P;
    _a_xjal[ _a_idx_xjal + 3 ] = flow2;
    _a_xjal[ _a_idx_xjal + 4 ] = LOST_PC;
    _a_xjal[ _a_idx_xjal + 5 ] = flow3;
    _a_xjal[ _a_idx_xjal + 6 ] = LOST_AN;
}

```

```

    _a_xjal[ _a_idx_xjal + 7 ] = flow4;
    _a_xjal[ _a_idx_xjal + 8 ] = LOST_V;
    _a_xjal[ _a_idx_xjal + 9 ] = flow6;
    _a_xjal[ _a_idx_xjal + 10 ] = LOST_G;
    _a_xjal[ _a_idx_xjal + 11 ] = flow7;
    _a_xjal[ _a_idx_xjal + 12 ] = LOST_C;
    _a_xjal[ _a_idx_xjal + 13 ] = flow8;
    _a_xjal[ _a_idx_xjal + 14 ] = LOST_A;
    _a_xjal[ _a_idx_xjal + 15 ] = flow1;
    _a_idx_xjal += 16;
    super.getPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

/**
 * This function places phase vector into initial conditions (is used while solving
 algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    LOST_S = _a_xjal[ _a_idx_xjal + 0 ];
    flow = _a_xjal[ _a_idx_xjal + 1 ];
    LOST_P = _a_xjal[ _a_idx_xjal + 2 ];
    flow2 = _a_xjal[ _a_idx_xjal + 3 ];
    LOST_PC = _a_xjal[ _a_idx_xjal + 4 ];
    flow3 = _a_xjal[ _a_idx_xjal + 5 ];
    LOST_AN = _a_xjal[ _a_idx_xjal + 6 ];
    flow4 = _a_xjal[ _a_idx_xjal + 7 ];
    LOST_V = _a_xjal[ _a_idx_xjal + 8 ];
    flow6 = _a_xjal[ _a_idx_xjal + 9 ];
    LOST_G = _a_xjal[ _a_idx_xjal + 10 ];
    flow7 = _a_xjal[ _a_idx_xjal + 11 ];
    LOST_C = _a_xjal[ _a_idx_xjal + 12 ];
    flow8 = _a_xjal[ _a_idx_xjal + 13 ];
    LOST_A = _a_xjal[ _a_idx_xjal + 14 ];
    flow1 = _a_xjal[ _a_idx_xjal + 15 ];
    _a_idx_xjal += 16;
    super.putPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

@AnyLogicInternalCodegenAPI
public void assignInitialConditions_xjal() {
    super.assignInitialConditions_xjal();
    _assign_CR_Formula_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_INTERACTIVE_Formula_xjal();
    _assign_SUCCESS_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_STATIC_Formula_xjal();
    _assign_POPUP_Formula_xjal();
    _assign_PICTURES_Formula_xjal();
    _assign_GAMES_Formula_xjal();
}

```

```

    _assign_VIDEO_Formula_xjal();
    _assign_COLORS_Formula_xjal();
    _assign_ANNOUNCE_Formula_xjal();
    _assign_ANIMATED_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void setupInitialConditions_xjal(Class<?> callerClass) {
    if (callerClass != Main.class) {
        return;
    }
    if (getInitialAlgebraicFlatEquationsCount_xjal() > 0) {
        SDIntegrationManager integrationManagerForInitialConditions = new
SDIntegrationManager( 0, getInitialAlgebraicFlatEquationsCount_xjal(),
getInitialFormulaFlatEquationsCount_xjal() );
        integrationManagerForInitialConditions.doStep( this, 0, 0.1, true );
    } else {
        assignInitialConditions_xjal();
    }
}

@AnyLogicInternalCodegenAPI
public void _assign_CR_Formula_xjal() {
    CR =
100
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_S_Formula_xjal() {
    RATE_S =
CR * 0.084
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_V_Formula_xjal() {
    RATE_V =
CR * 0.12
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_C_Formula_xjal() {
    RATE_C =
CR * 0.11
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_G_Formula_xjal() {
    RATE_G =
CR * 0.093
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_P_Formula_xjal() {
    RATE_P =
CR * 0.013
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_PC_Formula_xjal() {
    RATE_PC =
CR * 0.10
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_AN_Formula_xjal() {
    RATE_AN =
CR * 0.075
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_A_Formula_xjal() {
    RATE_A =
CR * 0.149
;
}
@AnyLogicInternalCodegenAPI
public void _assign_INTERACTIVE_Formula_xjal() {
    INTERACTIVE =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_SUCCESS_Formula_xjal() {
    SUCCESS =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_I_Formula_xjal() {
    RATE_I =
CR * 0.139
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_I_Formula_xjal() {
    LOST_I =
INTERACTIVE * 0.25 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow5_Formula_xjal() {
    flow5 =
INTERACTIVE * 0.75 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RETURNED_Formula_xjal() {
    RETURNED =
SUCCESS * 1
;
}
@AnyLogicInternalCodegenAPI
public void _assign_STATIC_Formula_xjal() {
    STATIC =
RATE_S - LOST_S - flow + 0 + parameter_S
;
}
@AnyLogicInternalCodegenAPI
public void _assign_POPUP_Formula_xjal() {
    POPUP =
RATE_P - flow2 - LOST_P + 0 + parameter_P
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _assign_PICTURES_Formula_xjal() {
        PICTURES =
RATE_PC - flow3 - LOST_PC + 0 + parameter_PC
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_GAMES_Formula_xjal() {
        GAMES =
RATE_G - flow7 - LOST_G + 0 + parameter_G
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_VIDEO_Formula_xjal() {
        VIDEO =
RATE_V - flow6 - LOST_V + 0 + parameter_V
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_COLORS_Formula_xjal() {
        COLORS =
RATE_C - LOST_C - flow8 + 0 + parameter_C
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANNOUNCE_Formula_xjal() {
        ANNOUNCE =
RATE_AN - flow4 - LOST_AN + 0 + parameter_AN
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANIMATED_Formula_xjal() {
        ANIMATED =
RATE_A - flow1 - LOST_A + 0 + parameter_A
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_S_Formula_xjal() {
        LOST_S =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow_Formula_xjal() {
        flow =
STATIC * 0.55 + parameter_S
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_P_Formula_xjal() {
        LOST_P =
POPUP * 0.3 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow2_Formula_xjal() {
        flow2 =
POPUP * 0.7 + parameter_P
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_PC_Formula_xjal() {
        LOST_PC =

```

```

PICTURES * 0.45 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow3_Formula_xjal() {
    flow3 =
PICTURES * 0.55 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_AN_Formula_xjal() {
    LOST_AN =
ANNOUNCE * 0.6 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow4_Formula_xjal() {
    flow4 =
ANNOUNCE * 0.40 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_V_Formula_xjal() {
    LOST_V =
VIDEO * 0.35 + parameter_V
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow6_Formula_xjal() {
    flow6 =
VIDEO * 0.75
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_G_Formula_xjal() {
    LOST_G =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow7_Formula_xjal() {
    flow7 =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_C_Formula_xjal() {
    LOST_C =
COLORS * 0.4
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow8_Formula_xjal() {
    flow8 =
COLORS * 0.6 + parameter_C
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_A_Formula_xjal() {
    LOST_A =
ANIMATED * 0.20 + parameter_A
;

```



```

}
@AnyLogicInternalCodegenAPI
public void _assign_flow1_Formula_xjal() {
    flow1 =
ANIMATED * 0.8 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void formulasExecute_xjal() {
    super.formulasExecute_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_LOST_S_Formula_xjal();
    _assign_flow_Formula_xjal();
    _assign_LOST_P_Formula_xjal();
    _assign_flow2_Formula_xjal();
    _assign_LOST_PC_Formula_xjal();
    _assign_flow3_Formula_xjal();
    _assign_LOST_AN_Formula_xjal();
    _assign_flow4_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_LOST_V_Formula_xjal();
    _assign_flow6_Formula_xjal();
    _assign_LOST_G_Formula_xjal();
    _assign_flow7_Formula_xjal();
    _assign_LOST_C_Formula_xjal();
    _assign_flow8_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_LOST_A_Formula_xjal();
    _assign_flow1_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void getRightPart_xjal( double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal ) {
    _get_CR_Derivative_xjal( _d_xjal, _d_idx_xjal + 0 );
    _get_STATIC_Derivative_xjal( _d_xjal, _d_idx_xjal + 1 );
    _get_INTERACTIVE_Derivative_xjal( _d_xjal, _d_idx_xjal + 2 );
    _get_POPUP_Derivative_xjal( _d_xjal, _d_idx_xjal + 3 );
    _get_PICTURES_Derivative_xjal( _d_xjal, _d_idx_xjal + 4 );
    _get_GAMES_Derivative_xjal( _d_xjal, _d_idx_xjal + 5 );
    _get_VIDEO_Derivative_xjal( _d_xjal, _d_idx_xjal + 6 );
    _get_COLORS_Derivative_xjal( _d_xjal, _d_idx_xjal + 7 );
    _get_ANNOUNCE_Derivative_xjal( _d_xjal, _d_idx_xjal + 8 );
    _get_SUCCESS_Derivative_xjal( _d_xjal, _d_idx_xjal + 9 );
    _get_ANIMATED_Derivative_xjal( _d_xjal, _d_idx_xjal + 10 );
    _d_idx_xjal += 11;
    super.getRightPart_xjal( _d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void getRightPartForInitialConditions_xjal( double[] _a_xjal, int
_a_idx_xjal ) {
    _get_LOST_S_Equation_xjal( _a_xjal, _a_idx_xjal + 0 );

```

```

    _get_flow_Equation_xjal( _a_xjal, _a_idx_xjal + 1 );
    _get_LOST_P_Equation_xjal( _a_xjal, _a_idx_xjal + 2 );
    _get_flow2_Equation_xjal( _a_xjal, _a_idx_xjal + 3 );
    _get_LOST_PC_Equation_xjal( _a_xjal, _a_idx_xjal + 4 );
    _get_flow3_Equation_xjal( _a_xjal, _a_idx_xjal + 5 );
    _get_LOST_AN_Equation_xjal( _a_xjal, _a_idx_xjal + 6 );
    _get_flow4_Equation_xjal( _a_xjal, _a_idx_xjal + 7 );
    _get_LOST_V_Equation_xjal( _a_xjal, _a_idx_xjal + 8 );
    _get_flow6_Equation_xjal( _a_xjal, _a_idx_xjal + 9 );
    _get_LOST_G_Equation_xjal( _a_xjal, _a_idx_xjal + 10 );
    _get_flow7_Equation_xjal( _a_xjal, _a_idx_xjal + 11 );
    _get_LOST_C_Equation_xjal( _a_xjal, _a_idx_xjal + 12 );
    _get_flow8_Equation_xjal( _a_xjal, _a_idx_xjal + 13 );
    _get_LOST_A_Equation_xjal( _a_xjal, _a_idx_xjal + 14 );
    _get_flow1_Equation_xjal( _a_xjal, _a_idx_xjal + 15 );
    _a_idx_xjal += 16;
    super.getRightPartForInitialConditions_xjal( _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void _get_CR_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RETURNED - RATE_S - RATE_I- RATE_V- RATE_C- RATE_G- RATE_P- RATE_PC- RATE_AN- RATE_A
;
}
@AnyLogicInternalCodegenAPI
public void _get_STATIC_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_S - LOST_S - flow
;
}
@AnyLogicInternalCodegenAPI
public void _get_INTERACTIVE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_I - LOST_I - flow5
;
}
@AnyLogicInternalCodegenAPI
public void _get_POPUP_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_P - LOST_P - flow2
;
}
@AnyLogicInternalCodegenAPI
public void _get_PICTURES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_PC - LOST_PC - flow3
;
}
@AnyLogicInternalCodegenAPI
public void _get_GAMES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_G - LOST_G - flow7
;
}
@AnyLogicInternalCodegenAPI
public void _get_VIDEO_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_V - LOST_V - flow6
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _get_COLORS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_C - LOST_C - flow8
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANNOUNCE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_AN - LOST_AN - flow4
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_SUCCESS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+flow + flow2 + flow3+ flow4+ flow5+ flow6+ flow7+ flow8+ flow1 - RETURNED
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANIMATED_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_A - LOST_A - flow1
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_S_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- LOST_S;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- flow;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_P_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.3 + parameter_P
- LOST_P;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow2_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.7 + parameter_P
- flow2;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_PC_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.45 + parameter_PC
- LOST_PC;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow3_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.55 + parameter_PC
- flow3;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_AN_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =

```

```

ANNOUNCE * 0.6 + parameter_AN
- LOST_AN;
}
@AnyLogicInternalCodegenAPI
public void _get_flow4_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.40 + parameter_AN
- flow4;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_V_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.35 + parameter_V
- LOST_V;
}
@AnyLogicInternalCodegenAPI
public void _get_flow6_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.75
- flow6;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_G_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- LOST_G;
}
@AnyLogicInternalCodegenAPI
public void _get_flow7_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- flow7;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_C_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.4
- LOST_C;
}
@AnyLogicInternalCodegenAPI
public void _get_flow8_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.6 + parameter_C
- flow8;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_A_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.20 + parameter_A
- LOST_A;
}
@AnyLogicInternalCodegenAPI
public void _get_flow1_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.8 + parameter_A
- flow1;
}
@AnyLogicInternalCodegenAPI
protected SDIntegrationManager integrationManager_xjal = null;

@AnyLogicInternalCodegenAPI
public SDIntegrationManager getIntegrationManager_xjal() {

```

```

        if (integrationManager_xjal == null) {
            integrationManager_xjal = new
SDIntegrationManager( getDifferentialFlatEquationsCount_xjal(),
getRuntimeAlgebraicFlatEquationsCount_xjal(),
getRuntimeFormulaFlatEquationsCount_xjal() );
        }
        return integrationManager_xjal;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getDifferentialFlatEquationsCount_xjal() {
        return super.getDifferentialFlatEquationsCount_xjal() + 11;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getRuntimeFormulaFlatEquationsCount_xjal() {
        return super.getRuntimeFormulaFlatEquationsCount_xjal() + 28;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialAlgebraicFlatEquationsCount_xjal() {
        return super.getInitialAlgebraicFlatEquationsCount_xjal() + 16;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialFormulaFlatEquationsCount_xjal() {
        return super.getInitialFormulaFlatEquationsCount_xjal() + 23;
    }

    @AnyLogicInternalCodegenAPI
    private static Map<String, IElementDescriptor> elementDescriptors_xjal =
createElementDescriptors( Main.class );

    @AnyLogicInternalCodegenAPI
    @Override
    public Map<String, IElementDescriptor> getElementDescriptors() {
        return elementDescriptors_xjal;
    }
    @AnyLogicCustomProposalPriority(type =
AnyLogicCustomProposalPriority.Type.STATIC_ELEMENT)
    public static final Scale scale = new Scale( 10.0 );

    @Override
    public Scale getScale() {
        return scale;
    }

    // Events

    @AnyLogicInternalCodegenAPI
    public EventTimeout _chart_autoUpdateEvent_xjal = new EventTimeout(this);
    @AnyLogicInternalCodegenAPI
    public EventTimeout _autoCreatedDS_xjal = new EventTimeout(this);

    @Override
    @AnyLogicInternalCodegenAPI
    public String getNameOf( EventTimeout _e ) {

```

```

        if( _e == _chart_autoUpdateEvent_xjal ) return "chart auto update event";
        if( _e == _autoCreatedDS_xjal ) return "Auto-created DataSets auto update event";
        return super.getNameOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public EventTimeout.Mode getModeOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        if ( _e == _autoCreatedDS_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        return super.getModeOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double getFirstOccurrenceTime( EventTimeout _e ) {
        double _t;
        if ( _e == _chart_autoUpdateEvent_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if ( _e == _autoCreatedDS_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.getFirstOccurrenceTime( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double evaluateTimeoutOf( EventTimeout _e ) {
        double _t;
        if( _e == _chart_autoUpdateEvent_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if( _e == _autoCreatedDS_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.evaluateTimeoutOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public void executeActionOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) {
            chart.updateData();
            return;
        }
    }

```

```

if ( _e == _autoCreatedDS_xjal ) {
    _ds_RATE_S.update();
    _ds_RATE_I.update();
    _ds_RATE_V.update();
    _ds_RATE_C.update();
    _ds_RATE_G.update();
    _ds_RATE_P.update();
    _ds_RATE_PC.update();
    _ds_RATE_AN.update();
    _ds_LOST_S.update();
    _ds_flow.update();
    _ds_LOST_P.update();
    _ds_flow2.update();
    _ds_LOST_PC.update();
    _ds_flow3.update();
    _ds_LOST_AN.update();
    _ds_flow4.update();
    _ds_LOST_I.update();
    _ds_flow5.update();
    _ds_LOST_V.update();
    _ds_flow6.update();
    _ds_LOST_G.update();
    _ds_flow7.update();
    _ds_LOST_C.update();
    _ds_flow8.update();
    _ds_RETURNED.update();
    _ds_RATE_A.update();
    _ds_LOST_A.update();
    _ds_flow1.update();
    _ds_CR.update();
    _ds_STATIC.update();
    _ds_INTERACTIVE.update();
    _ds_POPUP.update();
    _ds_PICTURES.update();
    _ds_GAMES.update();
    _ds_VIDEO.update();
    _ds_COLORS.update();
    _ds_ANNOUNCE.update();
    _ds_SUCCESS.update();
    _ds_ANIMATED.update();
    return;
}
super.executeActionOf( _e );
}

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _STATECHART_COUNT_xjal = 0;

/**
 * Auto-created data set(s) for RATE_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_S );
    }
}

```

```

        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_P
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override

```



```

    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_AN );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_P
 */
@AnyLogicInternalCodegenAPI

```

```

public DataSet _ds_LOST_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow2
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow2 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow2 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow3
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow3 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow3 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_AN );
        _lastUpdateTime = time();
    }
} );
/**

```

```

* Auto-created data set(s) for flow4
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow4 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow4 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_I
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_I );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow5
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow5 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow5 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_V
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_V );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow6
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow6 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow6 );
        _lastUpdateTime = time();
    }
} );

```

```

    }
} );
/**
 * Auto-created data set(s) for LOST_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow7
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow7 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow7 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow8
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow8 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow8 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RETURNED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RETURNED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {

```

```

        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RETURNED );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow1
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow1 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow1 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for CR
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_CR = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.CR );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for STATIC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_STATIC = new DataSet( 100, new DataUpdater_xjal() {

```

```

    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.STATIC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for INTERACTIVE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_INTERACTIVE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.INTERACTIVE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for POPUP
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_POPUP = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.POPUP );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for PICTURES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_PICTURES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.PICTURES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for GAMES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_GAMES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.GAMES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for VIDEO

```

```

*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_VIDEO = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.VIDEO );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for COLORS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_COLORS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.COLORS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANNOUNCE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANNOUNCE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANNOUNCE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for SUCCESS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_SUCCESS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.SUCCESS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANIMATED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANIMATED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANIMATED );
        _lastUpdateTime = time();
    }
}

```

```

    } );
    // View areas
    public ViewArea _origin_VA = new ViewArea( this, "[Origin]", 0, 0, 1000.0, 600.0 );
    @Override
    @AnyLogicInternalCodegenAPI
    public int getViewAreas(Map<String, ViewArea> _output) {
        if ( _output != null ) {
            _output.put( "_origin_VA", this._origin_VA );
        }
        return 1 + super.getViewAreas( _output );
    }
    @AnyLogicInternalCodegenAPI
    protected static final int _chart = 1;

    /** Internal constant, shouldn't be accessed by user */
    @AnyLogicInternalCodegenAPI
    protected static final int _SHAPE_NEXT_ID_xjal = 2;

    @AnyLogicInternalCodegenAPI
    public boolean isPublicPresentationDefined() {
        return true;
    }

    @AnyLogicInternalCodegenAPI
    public boolean isEmbeddedAgentPresentationVisible( Agent _a ) {
        return super.isEmbeddedAgentPresentationVisible( _a );
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem0Value() {
        return
ANIMATED
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem1Value() {
        return
STATIC
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem2Value() {
        return
POPUP
;
    }

    /**
     * <i>This method should not be called by user</i>
     */

```



```

    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem3Value() {
        return
INTERACTIVE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem4Value() {
        return
PICTURES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem5Value() {
        return
GAMES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem6Value() {
        return
VIDEO
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem7Value() {
        return
ANNOUNCE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem8Value() {
        return
COLORS
;
    }
    protected BarChart chart;
    @AnyLogicInternalCodegenAPI
    private void _createPersistentElementsBP0_xjal() {
    }

    @AnyLogicInternalCodegenAPI

```

```

private void _createPersistentElementsAP0_xjal() {
    {
        List<DataItem> _items = new ArrayList<DataItem>( 9 );
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem0Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem1Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem2Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem3Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem4Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem5Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem6Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem7Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem8Value() );
            }
        });
        List<String> _titles = new ArrayList<String>( 9 );
        _titles.add( "ANIMATED" );
        _titles.add( "STATIC" );
        _titles.add( "POPOP" );
        _titles.add( "INTERACTIVE" );
    }
}

```

```

_titles.add( "PICTURES" );
_titles.add( "GAMES" );
_titles.add( "VIDEO" );
_titles.add( "ANNOUNCE" );
_titles.add( "COLORS" );
List<Color> _colors = new ArrayList<Color>( 9 );
_colors.add( limeGreen );
_colors.add( slateGray );
_colors.add( darkMagenta );
_colors.add( orange );
_colors.add( darkKhaki );
_colors.add( mediumTurquoise );
_colors.add( lightSlateBlue );
_colors.add( deepPink );
_colors.add( darkOrange );
chart = new BarChart(
    Main.this, true, 20.0, 700.0,
    1420.0, 580.0,
    null, null,
    50.0, 20.0,
    1330.0, 460.0, white, black, black,
    80.0, Chart.SOUTH,
    Chart.NORTH, Chart.SCALE_AUTO,
    0, 0, 0.8,
    Chart.GRID_DEFAULT,
    darkGray, darkGray, _items, _titles, _colors );
}
}

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsBS0_xjal() {
}

// Static initialization of persistent elements
{
_createPersistentElementsBP0_xjal();
}
protected ShapeTopLevelPresentationGroup presentation;
protected ShapeModelElementsGroup icon;

@Override
@AnyLogicInternalCodegenAPI
public ShapeTopLevelPresentationGroup getPresentationShape() {
return presentation;
}

@Override
@AnyLogicInternalCodegenAPI
public ShapeModelElementsGroup getModelElementsShape() {
return icon;
}

/**
 * Constructor
 */
public Main( Engine engine, Agent owner, AgentList<? extends Main> ownerPopulation )
{
super( engine, owner, ownerPopulation );
instantiateBaseStructureThis_xjal();
}
}

```

```

@AnyLogicInternalCodegenAPI
public void onOwnerChanged_xjal() {
    super.onOwnerChanged_xjal();
    setupReferences_xjal();
}

@AnyLogicInternalCodegenAPI
public void instantiateBaseStructure_xjal() {
    super.instantiateBaseStructure_xjal();
    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
private void instantiateBaseStructureThis_xjal() {
    setupReferences_xjal();
    // Registering in Engine continuous part
    getEngine().registerAgentWithEquations( this );
}

@AnyLogicInternalCodegenAPI
private void setupReferences_xjal() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main( double parameter_S, double parameter_P, double parameter_PC, double
parameter_AN, double parameter_I, double parameter_V, double parameter_G, double
parameter_C, double parameter_A ) {
    markParametersAreSet();
    this.parameter_S = parameter_S;
    this.parameter_P = parameter_P;
    this.parameter_PC = parameter_PC;
    this.parameter_AN = parameter_AN;
    this.parameter_I = parameter_I;
    this.parameter_V = parameter_V;
    this.parameter_G = parameter_G;
    this.parameter_C = parameter_C;
    this.parameter_A = parameter_A;
}

@Override
@AnyLogicInternalCodegenAPI
public void doCreate() {
    super.doCreate();
    // Assigning initial values for plain variables
    setupPlainVariables_Main_xjal();
    // Dynamic initialization of persistent elements
    _createPersistentElementsAP0_xjal();
    presentation = new ShapeTopLevelPresentationGroup( Main.this, true, 0, 0, 0, 0 ,
chart );
    icon = new ShapeModelElementsGroup( Main.this, getElementProperty(
"modell1.Main.icon", IElementDescriptor.MODEL_ELEMENT_DESCRIPTOR ) );
}

```

```

    icon.setIconOffsets( 0.0, 0.0 );
    // Port connectors with non-replicated objects
    // Creating replicated embedded objects
    setupInitialConditions_xjal( Main.class );
    // Dynamic initialization of persistent elements
    _createPersistentElementsBS0_xjal();
}

@Override
@AnyLogicInternalCodegenAPI
public void doStart() {
    super.doStart();
    _chart_autoUpdateEvent_xjal.start();
    _autoCreatedDS_xjal.start();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
public void setupPlainVariables_xjal() {
    setupPlainVariables_Main_xjal();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
private void setupPlainVariables_Main_xjal() {
}

// User API -----
@AnyLogicInternalCodegenAPI
public static LinkToAgentAnimationSettings _connections_commonAnimationSettings_xjal
= new LinkToAgentAnimationSettingsImpl( false, black, 1.0, LINE_STYLE_SOLID,
ARROW_NONE, 0.0 );

public LinkToAgentCollection<Agent, Agent> connections = new
LinkToAgentStandardImpl<Agent, Agent>(this, _connections_commonAnimationSettings_xjal);
@Override
public LinkToAgentCollection<? extends Agent, ? extends Agent>
getLinkToAgentStandard_xjal() {
    return connections;
}

@AnyLogicInternalCodegenAPI
public void drawLinksToAgents(boolean _underAgents_xjal, LinkToAgentAnimator
_animator_xjal) {
    super.drawLinksToAgents(_underAgents_xjal, _animator_xjal);
    if ( _underAgents_xjal ) {
        _animator_xjal.drawLink( this, connections, true, true );
    }
}

public AgentList<? extends Main> getPopulation() {
    return (AgentList<? extends Main>) super.getPopulation();
}

```

```

}

public List<? extends Main> agentsInRange( double distance ) {
    return (List<? extends Main>) super.agentsInRange( distance );
}

@Override
@AnyLogicInternalCodegenAPI
public boolean isLoggingToDB(EventOriginator _e) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return false;
    if ( _e == _autoCreatedDS_xjal ) return false;
    return super.isLoggingToDB( _e );
}

@AnyLogicInternalCodegenAPI
public void onDestroy() {
    _chart_autoUpdateEvent_xjal.onDestroy();
    _autoCreatedDS_xjal.onDestroy();
    // Unregistering in Engine continuous part
    getEngine().unregisterAgentWithEquations( this );
    _ds_RATE_S.destroyUpdater_xjal();
    _ds_RATE_I.destroyUpdater_xjal();
    _ds_RATE_V.destroyUpdater_xjal();
    _ds_RATE_C.destroyUpdater_xjal();
    _ds_RATE_G.destroyUpdater_xjal();
    _ds_RATE_P.destroyUpdater_xjal();
    _ds_RATE_PC.destroyUpdater_xjal();
    _ds_RATE_AN.destroyUpdater_xjal();
    _ds_LOST_S.destroyUpdater_xjal();
    _ds_flow.destroyUpdater_xjal();
    _ds_LOST_P.destroyUpdater_xjal();
    _ds_flow2.destroyUpdater_xjal();
    _ds_LOST_PC.destroyUpdater_xjal();
    _ds_flow3.destroyUpdater_xjal();
    _ds_LOST_AN.destroyUpdater_xjal();
    _ds_flow4.destroyUpdater_xjal();
    _ds_LOST_I.destroyUpdater_xjal();
    _ds_flow5.destroyUpdater_xjal();
    _ds_LOST_V.destroyUpdater_xjal();
    _ds_flow6.destroyUpdater_xjal();
    _ds_LOST_G.destroyUpdater_xjal();
    _ds_flow7.destroyUpdater_xjal();
    _ds_LOST_C.destroyUpdater_xjal();
    _ds_flow8.destroyUpdater_xjal();
    _ds_RETURNED.destroyUpdater_xjal();
    _ds_RATE_A.destroyUpdater_xjal();
    _ds_LOST_A.destroyUpdater_xjal();
    _ds_flow1.destroyUpdater_xjal();
    _ds_CR.destroyUpdater_xjal();
    _ds_STATIC.destroyUpdater_xjal();
    _ds_INTERACTIVE.destroyUpdater_xjal();
    _ds_POPUP.destroyUpdater_xjal();
    _ds_PICTURES.destroyUpdater_xjal();
    _ds_GAMES.destroyUpdater_xjal();
    _ds_VIDEO.destroyUpdater_xjal();
    _ds_COLORS.destroyUpdater_xjal();
    _ds_ANNOUNCE.destroyUpdater_xjal();
    _ds_SUCCESS.destroyUpdater_xjal();
    _ds_ANIMATED.destroyUpdater_xjal();
    super.onDestroy();
}

```

```

package model1;

import java.io.Serializable;
import java.sql.Connection;
import java.sql.SQLException;
import java.util.ArrayDeque;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Calendar;
import java.util.Collection;
import java.util.Collections;
import java.util.Comparator;
import java.util.Currency;
import java.util.Date;
import java.util.Enumeration;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Hashtable;
import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.LinkedHashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import java.util.Map;
import java.util.PriorityQueue;
import java.util.Random;
import java.util.Set;
import java.util.SortedMap;
import java.util.SortedSet;
import java.util.Stack;
import java.util.Timer;
import java.util.TreeMap;
import java.util.TreeSet;
import java.util.Vector;
import java.awt.Color;
import java.awt.Font;
import com.anylogic.engine.connectivity.ResultSet;
import com.anylogic.engine.connectivity.Statement;
import com.anylogic.engine.elements.*;
import com.anylogic.engine.markup.Network;
import com.anylogic.engine.Position;
import com.anylogic.engine.markup.PedFlowStatistics;
import com.anylogic.engine.markup.DensityMap;

import static java.lang.Math.*;
import static com.anylogic.engine.UtilitiesArray.*;
import static com.anylogic.engine.UtilitiesCollection.*;
import static com.anylogic.engine.presentation.UtilitiesColor.*;
import static com.anylogic.engine.HyperArray.*;

import com.anylogic.engine.*;
import com.anylogic.engine.analysis.*;
import com.anylogic.engine.connectivity.*;
import com.anylogic.engine.database.*;
import com.anylogic.engine.gis.*;
import com.anylogic.engine.markup.*;
import com.anylogic.engine.routing.*;
import com.anylogic.engine.presentation.*;
import com.anylogic.engine.gui.*;

```

```

import java.awt.geom.Arc2D;

public class Main extends Agent
{
    // Parameters

    public
double   parameter_S;

    /**
     * Returns default value for parameter <code>parameter_S</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_S_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_S( double parameter_S ) {
        if (parameter_S == this.parameter_S) {
            return;
        }
        double _oldValue_xjal = this.parameter_S;
        this.parameter_S = parameter_S;
        onChange_parameter_S_xjal( _oldValue_xjal );
        onChange();
    }

    /**
     * Calls "On change" action for parameter parameter_S.<br>
     * Note that 'oldValue' in that action will be unavailable if this method is called
by user
     * (current parameter value will be passed as 'oldValue').<br>
     * Please call <code>set_parameter_S()</code> method instead.
     */
    protected void onChange_parameter_S() {
        onChange_parameter_S_xjal( parameter_S );
    }

    @AnyLogicInternalCodegenAPI
    protected void onChange_parameter_S_xjal( double oldValue ) {
    }

    public
double   parameter_P;

    /**
     * Returns default value for parameter <code>parameter_P</code>.
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    public double _parameter_P_DefaultValue_xjal() {
        final Main self = this;
        return 0.0;
    }

    public void set_parameter_P( double parameter_P ) {
        if (parameter_P == this.parameter_P) {
            return;
        }
    }
}

```



```

    }
    double _oldValue_xjal = this.parameter_P;
    this.parameter_P = parameter_P;
    onChange_parameter_P_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_P.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_P()</code> method instead.
 */
protected void onChange_parameter_P() {
    onChange_parameter_P_xjal( parameter_P );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_P_xjal( double oldValue ) {

public
double parameter_PC;

/**
 * Returns default value for parameter <code>parameter_PC</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_PC_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_PC( double parameter_PC ) {
    if (parameter_PC == this.parameter_PC) {
        return;
    }
    double _oldValue_xjal = this.parameter_PC;
    this.parameter_PC = parameter_PC;
    onChange_parameter_PC_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_PC.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_PC()</code> method instead.
 */
protected void onChange_parameter_PC() {
    onChange_parameter_PC_xjal( parameter_PC );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_PC_xjal( double oldValue ) {

```

```

public
double parameter_AN;

/**
 * Returns default value for parameter parameter_AN.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_AN_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_AN( double parameter_AN ) {
    if (parameter_AN == this.parameter_AN) {
        return;
    }
    double _oldValue_xjal = this.parameter_AN;
    this.parameter_AN = parameter_AN;
    onChange_parameter_AN_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_AN.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call set_parameter_AN() method instead.
 */
protected void onChange_parameter_AN() {
    onChange_parameter_AN_xjal( parameter_AN );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_AN_xjal( double oldValue ) {
}

public
double parameter_I;

/**
 * Returns default value for parameter parameter_I.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_I_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_I( double parameter_I ) {
    if (parameter_I == this.parameter_I) {
        return;
    }
    double _oldValue_xjal = this.parameter_I;
    this.parameter_I = parameter_I;
    onChange_parameter_I_xjal( _oldValue_xjal );
    onChange();
}

```

```

/**
 * Calls "On change" action for parameter parameter_I.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_I()</code> method instead.
 */
protected void onChange_parameter_I() {
    onChange_parameter_I_xjal( parameter_I );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_I_xjal( double oldValue ) {
}

public
double parameter_V;

/**
 * Returns default value for parameter <code>parameter_V</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_V_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_V( double parameter_V ) {
    if (parameter_V == this.parameter_V) {
        return;
    }
    double _oldValue_xjal = this.parameter_V;
    this.parameter_V = parameter_V;
    onChange_parameter_V_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_V.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_V()</code> method instead.
 */
protected void onChange_parameter_V() {
    onChange_parameter_V_xjal( parameter_V );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_V_xjal( double oldValue ) {
}

public
double parameter_G;

/**
 * Returns default value for parameter <code>parameter_G</code>.
 * <i>This method should not be called by user</i>
 */

```

```

@AnyLogicInternalCodegenAPI
public double _parameter_G_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_G( double parameter_G ) {
    if (parameter_G == this.parameter_G) {
        return;
    }
    double _oldValue_xjal = this.parameter_G;
    this.parameter_G = parameter_G;
    onChange_parameter_G_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_G.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_G()</code> method instead.
 */
protected void onChange_parameter_G() {
    onChange_parameter_G_xjal( parameter_G );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_G_xjal( double oldValue ) {
}

public
double parameter_C;

/**
 * Returns default value for parameter <code>parameter_C</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_C_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_C( double parameter_C ) {
    if (parameter_C == this.parameter_C) {
        return;
    }
    double _oldValue_xjal = this.parameter_C;
    this.parameter_C = parameter_C;
    onChange_parameter_C_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_C.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_C()</code> method instead.
 */

```

```

protected void onChange_parameter_C() {
    onChange_parameter_C_xjal( parameter_C );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_C_xjal( double oldValue ) {
}

public
double parameter_A;

/**
 * Returns default value for parameter <code>parameter_A</code>.
 * <i>This method should not be called by user</i>
 */
@AnyLogicInternalCodegenAPI
public double _parameter_A_DefaultValue_xjal() {
    final Main self = this;
    return 0.0;
}

public void set_parameter_A( double parameter_A ) {
    if (parameter_A == this.parameter_A) {
        return;
    }
    double _oldValue_xjal = this.parameter_A;
    this.parameter_A = parameter_A;
    onChange_parameter_A_xjal( _oldValue_xjal );
    onChange();
}

/**
 * Calls "On change" action for parameter parameter_A.<br>
 * Note that 'oldValue' in that action will be unavailable if this method is called
by user
 * (current parameter value will be passed as 'oldValue').<br>
 * Please call <code>set_parameter_A()</code> method instead.
 */
protected void onChange_parameter_A() {
    onChange_parameter_A_xjal( parameter_A );
}

@AnyLogicInternalCodegenAPI
protected void onChange_parameter_A_xjal( double oldValue ) {
}

@Override
public void setParametersToDefaultValues() {
    super.setParametersToDefaultValues();
    parameter_S = _parameter_S_DefaultValue_xjal();
    parameter_P = _parameter_P_DefaultValue_xjal();
    parameter_PC = _parameter_PC_DefaultValue_xjal();
    parameter_AN = _parameter_AN_DefaultValue_xjal();
    parameter_I = _parameter_I_DefaultValue_xjal();
    parameter_V = _parameter_V_DefaultValue_xjal();
    parameter_G = _parameter_G_DefaultValue_xjal();
    parameter_C = _parameter_C_DefaultValue_xjal();
    parameter_A = _parameter_A_DefaultValue_xjal();
}

```

```

@Override
public boolean setParameter(String _name_xjal, Object _value_xjal, boolean
_callOnChange_xjal) {
    switch ( _name_xjal ) {
        case "parameter_S":
            if ( _callOnChange_xjal ) {
                set_parameter_S( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_S = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_P":
            if ( _callOnChange_xjal ) {
                set_parameter_P( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_P = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_PC":
            if ( _callOnChange_xjal ) {
                set_parameter_PC( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_PC = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_AN":
            if ( _callOnChange_xjal ) {
                set_parameter_AN( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_AN = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_I":
            if ( _callOnChange_xjal ) {
                set_parameter_I( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_I = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_V":
            if ( _callOnChange_xjal ) {
                set_parameter_V( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_V = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_G":
            if ( _callOnChange_xjal ) {
                set_parameter_G( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_G = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_C":
            if ( _callOnChange_xjal ) {
                set_parameter_C( ((Number) _value_xjal).doubleValue() );
            } else {
                parameter_C = ((Number) _value_xjal).doubleValue();
            }
            return true;
        case "parameter_A":
            if ( _callOnChange_xjal ) {

```

```

        set_parameter_A( ((Number) _value_xjal).doubleValue() );
    } else {
        parameter_A = ((Number) _value_xjal).doubleValue();
    }
    return true;
default:
    return super.setParameter( _name_xjal, _value_xjal, _callOnChange_xjal );
}
}

@Override
public <T> T getParameter(String _name_xjal) {
    Object _result_xjal;
    switch ( _name_xjal ) {
    case "parameter_S": _result_xjal = parameter_S; break;
    case "parameter_P": _result_xjal = parameter_P; break;
    case "parameter_PC": _result_xjal = parameter_PC; break;
    case "parameter_AN": _result_xjal = parameter_AN; break;
    case "parameter_I": _result_xjal = parameter_I; break;
    case "parameter_V": _result_xjal = parameter_V; break;
    case "parameter_G": _result_xjal = parameter_G; break;
    case "parameter_C": _result_xjal = parameter_C; break;
    case "parameter_A": _result_xjal = parameter_A; break;
    default: _result_xjal = super.getParameter( _name_xjal ); break;
    }
    return (T) _result_xjal;
}

@AnyLogicInternalCodegenAPI
private static String[] _parameterNames_xjal;

@Override
public String[] getParameterNames() {
    String[] result = _parameterNames_xjal;
    if (result == null) {
        List<String> list = new ArrayList<>( Arrays.asList(
super.getParameterNames() ) );
        list.add( "parameter_S" );
        list.add( "parameter_P" );
        list.add( "parameter_PC" );
        list.add( "parameter_AN" );
        list.add( "parameter_I" );
        list.add( "parameter_V" );
        list.add( "parameter_G" );
        list.add( "parameter_C" );
        list.add( "parameter_A" );
        result = list.toArray( new String[ list.size() ] );
        _parameterNames_xjal = result;
    }
    return result;
}

// Dynamic (Flow/Auxiliary/Stock) Variables

public double RATE_S;

public double RATE_I;

public double RATE_V;

public double RATE_C;

```

```
public double RATE_G;
public double RATE_P;
public double RATE_PC;
public double RATE_AN;
public double LOST_S;
public double flow;
public double LOST_P;
public double flow2;
public double LOST_PC;
public double flow3;
public double LOST_AN;
public double flow4;
public double LOST_I;
public double flow5;
public double LOST_V;
public double flow6;
public double LOST_G;
public double flow7;
public double LOST_C;
public double flow8;
public double RETURNED;
public double RATE_A;
public double LOST_A;
public double flow1;
public double CR;
public double STATIC;
public double INTERACTIVE;
public double POPUP;
public double PICTURES;
public double GAMES;
public double VIDEO;
```



```

public double COLORS;

public double ANNOUNCE;

public double SUCCESS;

public double ANIMATED;
/**
 * Writes model variables into given arrays
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    _d_xjal[ _d_idx_xjal + 0 ] = CR;
    _d_xjal[ _d_idx_xjal + 1 ] = STATIC;
    _d_xjal[ _d_idx_xjal + 2 ] = INTERACTIVE;
    _d_xjal[ _d_idx_xjal + 3 ] = POPUP;
    _d_xjal[ _d_idx_xjal + 4 ] = PICTURES;
    _d_xjal[ _d_idx_xjal + 5 ] = GAMES;
    _d_xjal[ _d_idx_xjal + 6 ] = VIDEO;
    _d_xjal[ _d_idx_xjal + 7 ] = COLORS;
    _d_xjal[ _d_idx_xjal + 8 ] = ANNOUNCE;
    _d_xjal[ _d_idx_xjal + 9 ] = SUCCESS;
    _d_xjal[ _d_idx_xjal + 10 ] = ANIMATED;
    _d_idx_xjal += 11;
    super.getPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * Writes given arrays to model variables
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVector_xjal(double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal) {
    CR = _d_xjal[ _d_idx_xjal + 0 ];
    STATIC = _d_xjal[ _d_idx_xjal + 1 ];
    INTERACTIVE = _d_xjal[ _d_idx_xjal + 2 ];
    POPUP = _d_xjal[ _d_idx_xjal + 3 ];
    PICTURES = _d_xjal[ _d_idx_xjal + 4 ];
    GAMES = _d_xjal[ _d_idx_xjal + 5 ];
    VIDEO = _d_xjal[ _d_idx_xjal + 6 ];
    COLORS = _d_xjal[ _d_idx_xjal + 7 ];
    ANNOUNCE = _d_xjal[ _d_idx_xjal + 8 ];
    SUCCESS = _d_xjal[ _d_idx_xjal + 9 ];
    ANIMATED = _d_xjal[ _d_idx_xjal + 10 ];
    _d_idx_xjal += 11;
    super.putPhaseVector_xjal(_d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal);
}
/**
 * This function gets phase vector from initial conditions (is used while solving
algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void getPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    _a_xjal[ _a_idx_xjal + 0 ] = LOST_S;
    _a_xjal[ _a_idx_xjal + 1 ] = flow;
    _a_xjal[ _a_idx_xjal + 2 ] = LOST_P;
    _a_xjal[ _a_idx_xjal + 3 ] = flow2;
    _a_xjal[ _a_idx_xjal + 4 ] = LOST_PC;
    _a_xjal[ _a_idx_xjal + 5 ] = flow3;
    _a_xjal[ _a_idx_xjal + 6 ] = LOST_AN;
}

```

```

    _a_xjal[ _a_idx_xjal + 7 ] = flow4;
    _a_xjal[ _a_idx_xjal + 8 ] = LOST_V;
    _a_xjal[ _a_idx_xjal + 9 ] = flow6;
    _a_xjal[ _a_idx_xjal + 10 ] = LOST_G;
    _a_xjal[ _a_idx_xjal + 11 ] = flow7;
    _a_xjal[ _a_idx_xjal + 12 ] = LOST_C;
    _a_xjal[ _a_idx_xjal + 13 ] = flow8;
    _a_xjal[ _a_idx_xjal + 14 ] = LOST_A;
    _a_xjal[ _a_idx_xjal + 15 ] = flow1;
    _a_idx_xjal += 16;
    super.getPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

/**
 * This function places phase vector into initial conditions (is used while solving
 algebraic loops in initial conditions)
 */
@AnyLogicInternalCodegenAPI
public void putPhaseVectorForInitialConditions_xjal(double[] _a_xjal, int
_a_idx_xjal) {
    LOST_S = _a_xjal[ _a_idx_xjal + 0 ];
    flow = _a_xjal[ _a_idx_xjal + 1 ];
    LOST_P = _a_xjal[ _a_idx_xjal + 2 ];
    flow2 = _a_xjal[ _a_idx_xjal + 3 ];
    LOST_PC = _a_xjal[ _a_idx_xjal + 4 ];
    flow3 = _a_xjal[ _a_idx_xjal + 5 ];
    LOST_AN = _a_xjal[ _a_idx_xjal + 6 ];
    flow4 = _a_xjal[ _a_idx_xjal + 7 ];
    LOST_V = _a_xjal[ _a_idx_xjal + 8 ];
    flow6 = _a_xjal[ _a_idx_xjal + 9 ];
    LOST_G = _a_xjal[ _a_idx_xjal + 10 ];
    flow7 = _a_xjal[ _a_idx_xjal + 11 ];
    LOST_C = _a_xjal[ _a_idx_xjal + 12 ];
    flow8 = _a_xjal[ _a_idx_xjal + 13 ];
    LOST_A = _a_xjal[ _a_idx_xjal + 14 ];
    flow1 = _a_xjal[ _a_idx_xjal + 15 ];
    _a_idx_xjal += 16;
    super.putPhaseVectorForInitialConditions_xjal(_a_xjal, _a_idx_xjal);
}

@AnyLogicInternalCodegenAPI
public void assignInitialConditions_xjal() {
    super.assignInitialConditions_xjal();
    _assign_CR_Formula_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_INTERACTIVE_Formula_xjal();
    _assign_SUCCESS_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_STATIC_Formula_xjal();
    _assign_POPUP_Formula_xjal();
    _assign_PICTURES_Formula_xjal();
    _assign_GAMES_Formula_xjal();
}

```

```

    _assign_VIDEO_Formula_xjal();
    _assign_COLORS_Formula_xjal();
    _assign_ANNOUNCE_Formula_xjal();
    _assign_ANIMATED_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void setupInitialConditions_xjal(Class<?> callerClass) {
    if (callerClass != Main.class) {
        return;
    }
    if (getInitialAlgebraicFlatEquationsCount_xjal() > 0) {
        SDIntegrationManager integrationManagerForInitialConditions = new
SDIntegrationManager( 0, getInitialAlgebraicFlatEquationsCount_xjal(),
getInitialFormulaFlatEquationsCount_xjal() );
        integrationManagerForInitialConditions.doStep( this, 0, 0.1, true );
    } else {
        assignInitialConditions_xjal();
    }
}

@AnyLogicInternalCodegenAPI
public void _assign_CR_Formula_xjal() {
    CR =
100
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_S_Formula_xjal() {
    RATE_S =
CR * 0.084
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_V_Formula_xjal() {
    RATE_V =
CR * 0.12
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_C_Formula_xjal() {
    RATE_C =
CR * 0.11
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_G_Formula_xjal() {
    RATE_G =
CR * 0.093
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_P_Formula_xjal() {
    RATE_P =
CR * 0.013
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_PC_Formula_xjal() {
    RATE_PC =
CR * 0.10
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_AN_Formula_xjal() {
    RATE_AN =
CR * 0.075
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_A_Formula_xjal() {
    RATE_A =
CR * 0.149
;
}
@AnyLogicInternalCodegenAPI
public void _assign_INTERACTIVE_Formula_xjal() {
    INTERACTIVE =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_SUCCESS_Formula_xjal() {
    SUCCESS =
0
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RATE_I_Formula_xjal() {
    RATE_I =
CR * 0.139
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_I_Formula_xjal() {
    LOST_I =
INTERACTIVE * 0.25 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow5_Formula_xjal() {
    flow5 =
INTERACTIVE * 0.75 + parameter_I
;
}
@AnyLogicInternalCodegenAPI
public void _assign_RETURNED_Formula_xjal() {
    RETURNED =
SUCCESS * 1
;
}
@AnyLogicInternalCodegenAPI
public void _assign_STATIC_Formula_xjal() {
    STATIC =
RATE_S - LOST_S - flow + 0 + parameter_S
;
}
@AnyLogicInternalCodegenAPI
public void _assign_POPUP_Formula_xjal() {
    POPUP =
RATE_P - flow2 - LOST_P + 0 + parameter_P
;
}
@AnyLogicInternalCodegenAPI

```

```

    public void _assign_PICTURES_Formula_xjal() {
        PICTURES =
RATE_PC - flow3 - LOST_PC + 0 + parameter_PC
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_GAMES_Formula_xjal() {
        GAMES =
RATE_G - flow7 - LOST_G + 0 + parameter_G
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_VIDEO_Formula_xjal() {
        VIDEO =
RATE_V - flow6 - LOST_V + 0 + parameter_V
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_COLORS_Formula_xjal() {
        COLORS =
RATE_C - LOST_C - flow8 + 0 + parameter_C
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANNOUNCE_Formula_xjal() {
        ANNOUNCE =
RATE_AN - flow4 - LOST_AN + 0 + parameter_AN
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_ANIMATED_Formula_xjal() {
        ANIMATED =
RATE_A - flow1 - LOST_A + 0 + parameter_A
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_S_Formula_xjal() {
        LOST_S =
STATIC * 0.55 + parameter_S
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow_Formula_xjal() {
        flow =
STATIC * 0.55 + parameter_S
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_P_Formula_xjal() {
        LOST_P =
POPUP * 0.3 + parameter_P
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_flow2_Formula_xjal() {
        flow2 =
POPUP * 0.7 + parameter_P
;
    }
    @AnyLogicInternalCodegenAPI
    public void _assign_LOST_PC_Formula_xjal() {
        LOST_PC =

```

```

PICTURES * 0.45 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow3_Formula_xjal() {
    flow3 =
PICTURES * 0.55 + parameter_PC
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_AN_Formula_xjal() {
    LOST_AN =
ANNOUNCE * 0.6 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow4_Formula_xjal() {
    flow4 =
ANNOUNCE * 0.40 + parameter_AN
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_V_Formula_xjal() {
    LOST_V =
VIDEO * 0.35 + parameter_V
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow6_Formula_xjal() {
    flow6 =
VIDEO * 0.75
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_G_Formula_xjal() {
    LOST_G =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow7_Formula_xjal() {
    flow7 =
GAMES * 0.5 + parameter_G
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_C_Formula_xjal() {
    LOST_C =
COLORS * 0.4
;
}
@AnyLogicInternalCodegenAPI
public void _assign_flow8_Formula_xjal() {
    flow8 =
COLORS * 0.6 + parameter_C
;
}
@AnyLogicInternalCodegenAPI
public void _assign_LOST_A_Formula_xjal() {
    LOST_A =
ANIMATED * 0.20 + parameter_A
;
}

```

```

}
@AnyLogicInternalCodegenAPI
public void _assign_flow1_Formula_xjal() {
    flow1 =
ANIMATED * 0.8 + parameter_A
;
}
@AnyLogicInternalCodegenAPI
public void formulasExecute_xjal() {
    super.formulasExecute_xjal();
    _assign_RATE_S_Formula_xjal();
    _assign_RATE_I_Formula_xjal();
    _assign_RATE_V_Formula_xjal();
    _assign_RATE_C_Formula_xjal();
    _assign_RATE_G_Formula_xjal();
    _assign_RATE_P_Formula_xjal();
    _assign_RATE_PC_Formula_xjal();
    _assign_RATE_AN_Formula_xjal();
    _assign_LOST_S_Formula_xjal();
    _assign_flow_Formula_xjal();
    _assign_LOST_P_Formula_xjal();
    _assign_flow2_Formula_xjal();
    _assign_LOST_PC_Formula_xjal();
    _assign_flow3_Formula_xjal();
    _assign_LOST_AN_Formula_xjal();
    _assign_flow4_Formula_xjal();
    _assign_LOST_I_Formula_xjal();
    _assign_flow5_Formula_xjal();
    _assign_LOST_V_Formula_xjal();
    _assign_flow6_Formula_xjal();
    _assign_LOST_G_Formula_xjal();
    _assign_flow7_Formula_xjal();
    _assign_LOST_C_Formula_xjal();
    _assign_flow8_Formula_xjal();
    _assign_RETURNED_Formula_xjal();
    _assign_RATE_A_Formula_xjal();
    _assign_LOST_A_Formula_xjal();
    _assign_flow1_Formula_xjal();
}

@AnyLogicInternalCodegenAPI
public void getRightPart_xjal( double[] _d_xjal, int _d_idx_xjal, double[] _a_xjal,
int _a_idx_xjal ) {
    _get_CR_Derivative_xjal( _d_xjal, _d_idx_xjal + 0 );
    _get_STATIC_Derivative_xjal( _d_xjal, _d_idx_xjal + 1 );
    _get_INTERACTIVE_Derivative_xjal( _d_xjal, _d_idx_xjal + 2 );
    _get_POPUP_Derivative_xjal( _d_xjal, _d_idx_xjal + 3 );
    _get_PICTURES_Derivative_xjal( _d_xjal, _d_idx_xjal + 4 );
    _get_GAMES_Derivative_xjal( _d_xjal, _d_idx_xjal + 5 );
    _get_VIDEO_Derivative_xjal( _d_xjal, _d_idx_xjal + 6 );
    _get_COLORS_Derivative_xjal( _d_xjal, _d_idx_xjal + 7 );
    _get_ANNOUNCE_Derivative_xjal( _d_xjal, _d_idx_xjal + 8 );
    _get_SUCCESS_Derivative_xjal( _d_xjal, _d_idx_xjal + 9 );
    _get_ANIMATED_Derivative_xjal( _d_xjal, _d_idx_xjal + 10 );
    _d_idx_xjal += 11;
    super.getRightPart_xjal( _d_xjal, _d_idx_xjal, _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void getRightPartForInitialConditions_xjal( double[] _a_xjal, int
_a_idx_xjal ) {
    _get_LOST_S_Equation_xjal( _a_xjal, _a_idx_xjal + 0 );

```

```

    _get_flow_Equation_xjal( _a_xjal, _a_idx_xjal + 1 );
    _get_LOST_P_Equation_xjal( _a_xjal, _a_idx_xjal + 2 );
    _get_flow2_Equation_xjal( _a_xjal, _a_idx_xjal + 3 );
    _get_LOST_PC_Equation_xjal( _a_xjal, _a_idx_xjal + 4 );
    _get_flow3_Equation_xjal( _a_xjal, _a_idx_xjal + 5 );
    _get_LOST_AN_Equation_xjal( _a_xjal, _a_idx_xjal + 6 );
    _get_flow4_Equation_xjal( _a_xjal, _a_idx_xjal + 7 );
    _get_LOST_V_Equation_xjal( _a_xjal, _a_idx_xjal + 8 );
    _get_flow6_Equation_xjal( _a_xjal, _a_idx_xjal + 9 );
    _get_LOST_G_Equation_xjal( _a_xjal, _a_idx_xjal + 10 );
    _get_flow7_Equation_xjal( _a_xjal, _a_idx_xjal + 11 );
    _get_LOST_C_Equation_xjal( _a_xjal, _a_idx_xjal + 12 );
    _get_flow8_Equation_xjal( _a_xjal, _a_idx_xjal + 13 );
    _get_LOST_A_Equation_xjal( _a_xjal, _a_idx_xjal + 14 );
    _get_flow1_Equation_xjal( _a_xjal, _a_idx_xjal + 15 );
    _a_idx_xjal += 16;
    super.getRightPartForInitialConditions_xjal( _a_xjal, _a_idx_xjal );
}

@AnyLogicInternalCodegenAPI
public void _get_CR_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RETURNED - RATE_S - RATE_I- RATE_V- RATE_C- RATE_G- RATE_P- RATE_PC- RATE_AN- RATE_A
;
}
@AnyLogicInternalCodegenAPI
public void _get_STATIC_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_S - LOST_S - flow
;
}
@AnyLogicInternalCodegenAPI
public void _get_INTERACTIVE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_I - LOST_I - flow5
;
}
@AnyLogicInternalCodegenAPI
public void _get_POPUP_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_P - LOST_P - flow2
;
}
@AnyLogicInternalCodegenAPI
public void _get_PICTURES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_PC - LOST_PC - flow3
;
}
@AnyLogicInternalCodegenAPI
public void _get_GAMES_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_G - LOST_G - flow7
;
}
@AnyLogicInternalCodegenAPI
public void _get_VIDEO_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
    _d_xjal[ _d_idx_xjal ] =
+RATE_V - LOST_V - flow6
;
}
@AnyLogicInternalCodegenAPI

```



```

    public void _get_COLORS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_C - LOST_C - flow8
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANNOUNCE_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_AN - LOST_AN - flow4
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_SUCCESS_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+flow + flow2 + flow3+ flow4+ flow5+ flow6+ flow7+ flow8+ flow1 - RETURNED
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_ANIMATED_Derivative_xjal( double[] _d_xjal, int _d_idx_xjal ) {
        _d_xjal[ _d_idx_xjal ] =
+RATE_A - LOST_A - flow1
    ;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_S_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- LOST_S;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
STATIC * 0.55 + parameter_S
- flow;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_P_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.3 + parameter_P
- LOST_P;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow2_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
POPUP * 0.7 + parameter_P
- flow2;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_PC_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.45 + parameter_PC
- LOST_PC;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_flow3_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =
PICTURES * 0.55 + parameter_PC
- flow3;
    }
    @AnyLogicInternalCodegenAPI
    public void _get_LOST_AN_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
        _a_xjal[ _a_idx_xjal ] =

```

```

ANNOUNCE * 0.6 + parameter_AN
- LOST_AN;
}
@AnyLogicInternalCodegenAPI
public void _get_flow4_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANNOUNCE * 0.40 + parameter_AN
- flow4;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_V_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.35 + parameter_V
- LOST_V;
}
@AnyLogicInternalCodegenAPI
public void _get_flow6_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
VIDEO * 0.75
- flow6;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_G_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- LOST_G;
}
@AnyLogicInternalCodegenAPI
public void _get_flow7_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
GAMES * 0.5 + parameter_G
- flow7;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_C_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.4
- LOST_C;
}
@AnyLogicInternalCodegenAPI
public void _get_flow8_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
COLORS * 0.6 + parameter_C
- flow8;
}
@AnyLogicInternalCodegenAPI
public void _get_LOST_A_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.20 + parameter_A
- LOST_A;
}
@AnyLogicInternalCodegenAPI
public void _get_flow1_Equation_xjal( double[] _a_xjal, int _a_idx_xjal ) {
    _a_xjal[ _a_idx_xjal ] =
ANIMATED * 0.8 + parameter_A
- flow1;
}
@AnyLogicInternalCodegenAPI
protected SDIntegrationManager integrationManager_xjal = null;

@AnyLogicInternalCodegenAPI
public SDIntegrationManager getIntegrationManager_xjal() {

```

```

        if (integrationManager_xjal == null) {
            integrationManager_xjal = new
SDIntegrationManager( getDifferentialFlatEquationsCount_xjal(),
getRuntimeAlgebraicFlatEquationsCount_xjal(),
getRuntimeFormulaFlatEquationsCount_xjal() );
        }
        return integrationManager_xjal;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getDifferentialFlatEquationsCount_xjal() {
        return super.getDifferentialFlatEquationsCount_xjal() + 11;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getRuntimeFormulaFlatEquationsCount_xjal() {
        return super.getRuntimeFormulaFlatEquationsCount_xjal() + 28;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialAlgebraicFlatEquationsCount_xjal() {
        return super.getInitialAlgebraicFlatEquationsCount_xjal() + 16;
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public int getInitialFormulaFlatEquationsCount_xjal() {
        return super.getInitialFormulaFlatEquationsCount_xjal() + 23;
    }

    @AnyLogicInternalCodegenAPI
    private static Map<String, IElementDescriptor> elementDescriptors_xjal =
createElementDescriptors( Main.class );

    @AnyLogicInternalCodegenAPI
    @Override
    public Map<String, IElementDescriptor> getElementDescriptors() {
        return elementDescriptors_xjal;
    }
    @AnyLogicCustomProposalPriority(type =
AnyLogicCustomProposalPriority.Type.STATIC_ELEMENT)
    public static final Scale scale = new Scale( 10.0 );

    @Override
    public Scale getScale() {
        return scale;
    }

    // Events

    @AnyLogicInternalCodegenAPI
    public EventTimeout _chart_autoUpdateEvent_xjal = new EventTimeout(this);
    @AnyLogicInternalCodegenAPI
    public EventTimeout _autoCreatedDS_xjal = new EventTimeout(this);

    @Override
    @AnyLogicInternalCodegenAPI
    public String getNameOf( EventTimeout _e ) {

```

```

        if( _e == _chart_autoUpdateEvent_xjal ) return "chart auto update event";
        if( _e == _autoCreatedDS_xjal ) return "Auto-created DataSets auto update event";
        return super.getNameOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public EventTimeout.Mode getModeOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        if ( _e == _autoCreatedDS_xjal ) return EVENT_TIMEOUT_MODE_CYCLIC;
        return super.getModeOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double getFirstOccurrenceTime( EventTimeout _e ) {
        double _t;
        if ( _e == _chart_autoUpdateEvent_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if ( _e == _autoCreatedDS_xjal ) {
0
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.getFirstOccurrenceTime( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public double evaluateTimeoutOf( EventTimeout _e ) {
        double _t;
        if( _e == _chart_autoUpdateEvent_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        if( _e == _autoCreatedDS_xjal) {
1
;
            _t = toModelTime( _t, SECOND );
            return _t;
        }
        return super.evaluateTimeoutOf( _e );
    }

    @Override
    @AnyLogicInternalCodegenAPI
    public void executeActionOf( EventTimeout _e ) {
        if ( _e == _chart_autoUpdateEvent_xjal ) {
            chart.updateData();
            return;
        }
    }

```

```

if ( _e == _autoCreatedDS_xjal ) {
    _ds_RATE_S.update();
    _ds_RATE_I.update();
    _ds_RATE_V.update();
    _ds_RATE_C.update();
    _ds_RATE_G.update();
    _ds_RATE_P.update();
    _ds_RATE_PC.update();
    _ds_RATE_AN.update();
    _ds_LOST_S.update();
    _ds_flow.update();
    _ds_LOST_P.update();
    _ds_flow2.update();
    _ds_LOST_PC.update();
    _ds_flow3.update();
    _ds_LOST_AN.update();
    _ds_flow4.update();
    _ds_LOST_I.update();
    _ds_flow5.update();
    _ds_LOST_V.update();
    _ds_flow6.update();
    _ds_LOST_G.update();
    _ds_flow7.update();
    _ds_LOST_C.update();
    _ds_flow8.update();
    _ds_RETURNED.update();
    _ds_RATE_A.update();
    _ds_LOST_A.update();
    _ds_flow1.update();
    _ds_CR.update();
    _ds_STATIC.update();
    _ds_INTERACTIVE.update();
    _ds_POPUP.update();
    _ds_PICTURES.update();
    _ds_GAMES.update();
    _ds_VIDEO.update();
    _ds_COLORS.update();
    _ds_ANNOUNCE.update();
    _ds_SUCCESS.update();
    _ds_ANIMATED.update();
    return;
}
super.executeActionOf( _e );
}

```

```

/** Internal constant, shouldn't be accessed by user */
@AnyLogicInternalCodegenAPI
protected static final int _STATECHART_COUNT_xjal = 0;

```

```

/**
 * Auto-created data set(s) for RATE_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_S );
    }
}

```

```

        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_I
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_I );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_V
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_V );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_P
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override

```

```

    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_AN );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_S
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_S = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_S );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_P
 */
@AnyLogicInternalCodegenAPI

```

```

public DataSet _ds_LOST_P = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_P );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow2
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow2 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow2 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_PC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_PC = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_PC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow3
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow3 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow3 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_AN
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_AN = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_AN );
        _lastUpdateTime = time();
    }
} );
/**

```



```

* Auto-created data set(s) for flow4
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow4 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow4 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_I
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_I = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_I );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow5
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow5 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow5 );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for LOST_V
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_V = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_V );
        _lastUpdateTime = time();
    }
} );
/**
* Auto-created data set(s) for flow6
*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow6 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow6 );
        _lastUpdateTime = time();
    }
} );

```

```

    }
} );
/**
 * Auto-created data set(s) for LOST_G
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_G = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_G );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow7
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow7 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow7 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_C
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_C = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_C );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow8
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow8 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow8 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RETURNED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RETURNED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {

```

```

        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RETURNED );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for RATE_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_RATE_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.RATE_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for LOST_A
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_LOST_A = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.LOST_A );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for flow1
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_flow1 = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.flow1 );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for CR
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_CR = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.CR );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for STATIC
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_STATIC = new DataSet( 100, new DataUpdater_xjal() {

```

```

    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.STATIC );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for INTERACTIVE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_INTERACTIVE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.INTERACTIVE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for POPUP
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_POPUP = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.POPUP );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for PICTURES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_PICTURES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.PICTURES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for GAMES
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_GAMES = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.GAMES );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for VIDEO

```

```

*/
@AnyLogicInternalCodegenAPI
public DataSet _ds_VIDEO = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.VIDEO );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for COLORS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_COLORS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.COLORS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANNOUNCE
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANNOUNCE = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANNOUNCE );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for SUCCESS
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_SUCCESS = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.SUCCESS );
        _lastUpdateTime = time();
    }
} );
/**
 * Auto-created data set(s) for ANIMATED
 */
@AnyLogicInternalCodegenAPI
public DataSet _ds_ANIMATED = new DataSet( 100, new DataUpdater_xjal() {
    double _lastUpdateTime = Double.NaN;
    @Override
    public void update( DataSet _d ) {
        if ( time() == _lastUpdateTime ) { return; }
        _d.add( time(), Main.this.ANIMATED );
        _lastUpdateTime = time();
    }
}

```

```

    } );
    // View areas
    public ViewArea _origin_VA = new ViewArea( this, "[Origin]", 0, 0, 1000.0, 600.0 );
    @Override
    @AnyLogicInternalCodegenAPI
    public int getViewAreas(Map<String, ViewArea> _output) {
        if ( _output != null ) {
            _output.put( "_origin_VA", this._origin_VA );
        }
        return 1 + super.getViewAreas( _output );
    }
    @AnyLogicInternalCodegenAPI
    protected static final int _chart = 1;

    /** Internal constant, shouldn't be accessed by user */
    @AnyLogicInternalCodegenAPI
    protected static final int _SHAPE_NEXT_ID_xjal = 2;

    @AnyLogicInternalCodegenAPI
    public boolean isPublicPresentationDefined() {
        return true;
    }

    @AnyLogicInternalCodegenAPI
    public boolean isEmbeddedAgentPresentationVisible( Agent _a ) {
        return super.isEmbeddedAgentPresentationVisible( _a );
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem0Value() {
        return
ANIMATED
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem1Value() {
        return
STATIC
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem2Value() {
        return
POPUP
;
    }

    /**
     * <i>This method should not be called by user</i>
     */

```

```

    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem3Value() {
        return
INTERACTIVE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem4Value() {
        return
PICTURES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem5Value() {
        return
GAMES
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem6Value() {
        return
VIDEO
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem7Value() {
        return
ANNOUNCE
;
    }

    /**
     * <i>This method should not be called by user</i>
     */
    @AnyLogicInternalCodegenAPI
    private double _chart_DataItem8Value() {
        return
COLORS
;
    }
    protected BarChart chart;
    @AnyLogicInternalCodegenAPI
    private void _createPersistentElementsBP0_xjal() {
    }

    @AnyLogicInternalCodegenAPI

```

```

private void _createPersistentElementsAP0_xjal() {
    {
        List<DataItem> _items = new ArrayList<DataItem>( 9 );
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem0Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem1Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem2Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem3Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem4Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem5Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem6Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem7Value() );
            }
        });
        _items.add( new DataItem() {
            @Override
            public void update() {
                setValue( _chart_DataItem8Value() );
            }
        });
        List<String> _titles = new ArrayList<String>( 9 );
        _titles.add( "ANIMATED" );
        _titles.add( "STATIC" );
        _titles.add( "POPOP" );
        _titles.add( "INTERACTIVE" );
    }
}

```



```

_titles.add( "PICTURES" );
_titles.add( "GAMES" );
_titles.add( "VIDEO" );
_titles.add( "ANNOUNCE" );
_titles.add( "COLORS" );
List<Color> _colors = new ArrayList<Color>( 9 );
_colors.add( limeGreen );
_colors.add( slateGray );
_colors.add( darkMagenta );
_colors.add( orange );
_colors.add( darkKhaki );
_colors.add( mediumTurquoise );
_colors.add( lightSlateBlue );
_colors.add( deepPink );
_colors.add( darkOrange );
chart = new BarChart(
    Main.this, true, 20.0, 700.0,
    1420.0, 580.0,
    null, null,
    50.0, 20.0,
    1330.0, 460.0, white, black, black,
    80.0, Chart.SOUTH,
    Chart.NORTH, Chart.SCALE_AUTO,
    0, 0, 0.8,
    Chart.GRID_DEFAULT,
    darkGray, darkGray, _items, _titles, _colors );
}
}

@AnyLogicInternalCodegenAPI
private void _createPersistentElementsBS0_xjal() {
}

// Static initialization of persistent elements
{
_createPersistentElementsBP0_xjal();
}
protected ShapeTopLevelPresentationGroup presentation;
protected ShapeModelElementsGroup icon;

@Override
@AnyLogicInternalCodegenAPI
public ShapeTopLevelPresentationGroup getPresentationShape() {
return presentation;
}

@Override
@AnyLogicInternalCodegenAPI
public ShapeModelElementsGroup getModelElementsShape() {
return icon;
}

/**
 * Constructor
 */
public Main( Engine engine, Agent owner, AgentList<? extends Main> ownerPopulation )
{
super( engine, owner, ownerPopulation );
instantiateBaseStructureThis_xjal();
}
}

```

```

@AnyLogicInternalCodegenAPI
public void onOwnerChanged_xjal() {
    super.onOwnerChanged_xjal();
    setupReferences_xjal();
}

@AnyLogicInternalCodegenAPI
public void instantiateBaseStructure_xjal() {
    super.instantiateBaseStructure_xjal();
    instantiateBaseStructureThis_xjal();
}

@AnyLogicInternalCodegenAPI
private void instantiateBaseStructureThis_xjal() {
    setupReferences_xjal();
    // Registering in Engine continuous part
    getEngine().registerAgentWithEquations( this );
}

@AnyLogicInternalCodegenAPI
private void setupReferences_xjal() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main() {
}

/**
 * Simple constructor. Please add created agent to some population by calling
goToPopulation() function
 */
public Main( double parameter_S, double parameter_P, double parameter_PC, double
parameter_AN, double parameter_I, double parameter_V, double parameter_G, double
parameter_C, double parameter_A ) {
    markParametersAreSet();
    this.parameter_S = parameter_S;
    this.parameter_P = parameter_P;
    this.parameter_PC = parameter_PC;
    this.parameter_AN = parameter_AN;
    this.parameter_I = parameter_I;
    this.parameter_V = parameter_V;
    this.parameter_G = parameter_G;
    this.parameter_C = parameter_C;
    this.parameter_A = parameter_A;
}

@Override
@AnyLogicInternalCodegenAPI
public void doCreate() {
    super.doCreate();
    // Assigning initial values for plain variables
    setupPlainVariables_Main_xjal();
    // Dynamic initialization of persistent elements
    _createPersistentElementsAP0_xjal();
    presentation = new ShapeTopLevelPresentationGroup( Main.this, true, 0, 0, 0, 0 ,
chart );
    icon = new ShapeModelElementsGroup( Main.this, getElementProperty(
"modell1.Main.icon", IElementDescriptor.MODEL_ELEMENT_DESCRIPTOR ) );
}

```

```

    icon.setIconOffsets( 0.0, 0.0 );
    // Port connectors with non-replicated objects
    // Creating replicated embedded objects
    setupInitialConditions_xjal( Main.class );
    // Dynamic initialization of persistent elements
    _createPersistentElementsBS0_xjal();
}

@Override
@AnyLogicInternalCodegenAPI
public void doStart() {
    super.doStart();
    _chart_autoUpdateEvent_xjal.start();
    _autoCreatedDS_xjal.start();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
public void setupPlainVariables_xjal() {
    setupPlainVariables_Main_xjal();
}

/**
 * Assigning initial values for plain variables<br>
 * <em>This method isn't designed to be called by user and may be removed in future
releases.</em>
 */
@AnyLogicInternalCodegenAPI
private void setupPlainVariables_Main_xjal() {

// User API -----
@AnyLogicInternalCodegenAPI
public static LinkToAgentAnimationSettings _connections_commonAnimationSettings_xjal
= new LinkToAgentAnimationSettingsImpl( false, black, 1.0, LINE_STYLE_SOLID,
ARROW_NONE, 0.0 );

public LinkToAgentCollection<Agent, Agent> connections = new
LinkToAgentStandardImpl<Agent, Agent>(this, _connections_commonAnimationSettings_xjal);
@Override
public LinkToAgentCollection<? extends Agent, ? extends Agent>
getLinkToAgentStandard_xjal() {
    return connections;
}

@AnyLogicInternalCodegenAPI
public void drawLinksToAgents(boolean _underAgents_xjal, LinkToAgentAnimator
_animator_xjal) {
    super.drawLinksToAgents(_underAgents_xjal, _animator_xjal);
    if ( _underAgents_xjal ) {
        _animator_xjal.drawLink( this, connections, true, true );
    }
}

public AgentList<? extends Main> getPopulation() {
    return (AgentList<? extends Main>) super.getPopulation();
}

```

```

}

public List<? extends Main> agentsInRange( double distance ) {
    return (List<? extends Main>) super.agentsInRange( distance );
}

@Override
@AnyLogicInternalCodegenAPI
public boolean isLoggingToDB(EventOriginator _e) {
    if ( _e == _chart_autoUpdateEvent_xjal ) return false;
    if ( _e == _autoCreatedDS_xjal ) return false;
    return super.isLoggingToDB( _e );
}

@AnyLogicInternalCodegenAPI
public void onDestroy() {
    _chart_autoUpdateEvent_xjal.onDestroy();
    _autoCreatedDS_xjal.onDestroy();
    // Unregistering in Engine continuous part
    getEngine().unregisterAgentWithEquations( this );
    _ds_RATE_S.destroyUpdater_xjal();
    _ds_RATE_I.destroyUpdater_xjal();
    _ds_RATE_V.destroyUpdater_xjal();
    _ds_RATE_C.destroyUpdater_xjal();
    _ds_RATE_G.destroyUpdater_xjal();
    _ds_RATE_P.destroyUpdater_xjal();
    _ds_RATE_PC.destroyUpdater_xjal();
    _ds_RATE_AN.destroyUpdater_xjal();
    _ds_LOST_S.destroyUpdater_xjal();
    _ds_flow.destroyUpdater_xjal();
    _ds_LOST_P.destroyUpdater_xjal();
    _ds_flow2.destroyUpdater_xjal();
    _ds_LOST_PC.destroyUpdater_xjal();
    _ds_flow3.destroyUpdater_xjal();
    _ds_LOST_AN.destroyUpdater_xjal();
    _ds_flow4.destroyUpdater_xjal();
    _ds_LOST_I.destroyUpdater_xjal();
    _ds_flow5.destroyUpdater_xjal();
    _ds_LOST_V.destroyUpdater_xjal();
    _ds_flow6.destroyUpdater_xjal();
    _ds_LOST_G.destroyUpdater_xjal();
    _ds_flow7.destroyUpdater_xjal();
    _ds_LOST_C.destroyUpdater_xjal();
    _ds_flow8.destroyUpdater_xjal();
    _ds_RETURNED.destroyUpdater_xjal();
    _ds_RATE_A.destroyUpdater_xjal();
    _ds_LOST_A.destroyUpdater_xjal();
    _ds_flow1.destroyUpdater_xjal();
    _ds_CR.destroyUpdater_xjal();
    _ds_STATIC.destroyUpdater_xjal();
    _ds_INTERACTIVE.destroyUpdater_xjal();
    _ds_POPUP.destroyUpdater_xjal();
    _ds_PICTURES.destroyUpdater_xjal();
    _ds_GAMES.destroyUpdater_xjal();
    _ds_VIDEO.destroyUpdater_xjal();
    _ds_COLORS.destroyUpdater_xjal();
    _ds_ANNOUNCE.destroyUpdater_xjal();
    _ds_SUCCESS.destroyUpdater_xjal();
    _ds_ANIMATED.destroyUpdater_xjal();
    super.onDestroy();
}

```