

Σχεδιασμός και Ανάπτυξη Πρότυπης Εφαρμογής Διάχυτου Υπολογισμού με Χρήση Διαδραστικών Εγκαταστάσεων



Νικόλαος Βασιλάκης
Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και
Πληροφορικής, Πανεπιστήμιο Πατρών

*“Beware of bugs in the above code;
I have only proved it correct,
not tried it.”*

Donald Knuth

Τίτλος διπλωματικής εργασίας: Σχεδιασμός και Ανάπτυξη Πρότυπης Εφαρμογής Διάχυτου Υπολογισμού με Χρήση Διαδραστικών Εγκαταστάσεων

Συγγραφέας: Νικόλαος Βασιλάκης , Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών, Πολυτεχνική Σχολή, Πανεπιστήμιο Πατρών.

Λέξεις Κλειδιά: *Διάχυτος Υπολογισμός, Περιρρέον Υπολογισμός, Everywhere, FinN, Διασκέδαση, Ψυχαγωγία, Liferay, Processing, Διαδικτυακή Πύλη, OpenGL, Γραφικά, Απόδοση, Διαδραστικές Εγκαταστάσεις, WSN, Αλληλεπίδραση, Sun, Java, SPOT, Πολυχρηστικά, Παιχνίδι, Ασύρματα, Αδόμητα, Δίκτυα Αισθητήρων, Τέχνη, Τεχνολογία, Επαυξημένη Πραγματικότητα, Augmented Reality, Θέμα, Κινητά Τηλέφωνα, Κινητές Συσκευές, Σκίτσο, Κινητά, Chromatizelt!, Tug of War, Hot Potato, Moving Monk.*

Thesis Title: Design and implementation of a novel application of ubiquitous computing using interactive installations.

Author: Nikolaos Vasilakis, Computer Engineering and Informatics Department, Polytechnic School, University of Patras.

Keywords: *Ubiquitous, Computing, Pervasive, Everywhere, FinN, Fun, Liferay, Processing, Portal, OpenGL, Rendering, Interactive, Installations, Sun, Java, SPOT, Multiplayer, Locative, Wireless, Sensor, Networks, Art, Technology, Augmented, Reality, Game, Interaction, Theme, Sketch, Mobile, Chromatizelt!, Tug of War, Hot Potato, Moving Monk, WSN.*

Περίληψη

Ελληνικά

Το σύστημα που παρουσιάζεται στην συγκεκριμένη εργασία επιτρέπει την δημιουργία , την εγκατάσταση και την διαχείριση πολυχρηστικών ψυχαγωγικών εφαρμογών με περιρρέοντα και τοπικά χαρακτηριστικά. Το κύριο συστατικό των εφαρμογών οι οποίες αναπτύσσονται με την χρήση της συγκεκριμένης πλατφόρμας είναι πως οι χρήστες αλληλεπιδρούν τόσο μεταξύ τους όσο και με το περιβάλλον με την χρήση χειρονομιών, κίνησης ή τρεξίματος. Με τις κινήσεις αυτές, οι χρήστες πραγματοποιούν ενέργειες οι οποίες σχετίζονται με το παιχνίδι, με την χρήση συσκευών που φέρουν αισθητήρες. Το σύστημα βασίζεται σε μία αρχιτεκτονική πολλών επιπέδων η οποία από την μία εφοδιάζει τους προγραμματιστές με ένα σύνολο οδηγιών και υπηρεσιών για την ανάπτυξη νέων παιχνιδιών και διαδραστικών εγκαταστάσεων ενώ από την άλλη προσφέρει ένα περιβάλλον το οποίο βασίζεται στον ιστό για την διαχείριση και τον έλεγχο τέτοιων εφαρμογών. Το ίδιο το σύστημα επί της ουσίας χειρίζεται έναν αριθμό πρακτικών ζητημάτων, τα οποία κατά παράδοση παρουσιάζονται στην ανάπτυξη τέτοιων συστημάτων όπως ο συγχρονισμός, η συμμόρφωση του δικτύου, η ανεκτική σε καθυστερήσεις επικοινωνία και η ανακάλυψη γειτόνων.

Αυτό που γίνεται κατανοητό, είναι πως τα αποτελέσματα από την χρήση της πλατφόρμας ξεφεύγουν από την στενή έννοια του παιχνιδιού ή τουλάχιστον της μονόπλευρης διασκέδασης καθώς το πλήθος των εφαρμογών που μπορεί να υλοποιηθεί είναι μεγάλο σε ποικιλία. Μακροσκοπικά, μπορεί να είναι παιχνίδια

σε επίπεδο πόλης, μπορεί να έχουν ψυχαγωγικό ή εκπαιδευτικό χαρακτήρα, εγκαταστάσεις που συνδυάζουν το τερπνόν μετά του ωφελίμου, πανταχού παρόντα συστήματα με χρήση σε μουσεία ή διαδραστικά έργα τέχνης. Αισθητή είναι πάντα η αλληλεπίδραση του παίκτη με τους άλλους χρήστες και τον περιβάλλοντα χώρο με αποτέλεσμα την παρουσία πληθώρας χαρακτηριστικών τα οποία απουσιάζουν από τα παραδοσιακά παιχνίδια – με λίγα λόγια τα παραδοσιακά παιχνίδια μπορούν να ενισχυθούν με την χρήση τέτοιων χαρακτηριστικών. Όσον αφορά τις εγκαταστάσεις, αυτές ποικίλουν, και ξεκινούν από ψυχαγωγικού χαρακτήρα φτάνοντας σε καλλιτεχνικά εδάφη, εφαρμογές για μουσεία και καλλιτεχνικές εκθέσεις.

Όσον αφορά το βάθος των θεμάτων υλοποίησης τα οποία παρουσιάζονται στην παρούσα εργασία, επιχειρείται μία συνολική παρουσίαση όλων των μερών του συστήματος, τόσο με μια σφαιρική, εποπτική προσέγγιση όσο και σε μια πιο αναλυτική και λεπτομερή έπειτα. Ιδιαίτερη έμφαση ωστόσο δίνεται στην υλοποίηση των ανώτατων επιπέδων της αρχιτεκτονικής, τα οποία αφορούν την τελική αλληλεπίδραση με τον χρήστη, είτε αυτή είναι η διαχείριση του συστήματος, είτε η παρακολούθηση του παιχνιδιού, είτε ο σχεδιασμός της τελικής εξόδου του συστήματος. Έτσι, γίνεται παρουσίαση και ανάλυση των τεχνολογιών πίσω από τα επίπεδα αυτά, έκθεση διαφόρων τρόπων υλοποίησης, εξήγηση των λεπτομερειών και των στόχων κάθε επιλογής.



Περίληψη

English

In this work, a framework for developing pervasive applications and interactive installations for entertainment and educational purposes is presented. The key characteristic of these applications is that players interact with each other and their surrounding environment by moving, running and gesturing as a mean to perform game-related actions, using sensor devices. The platform is based on a multilayer architecture, which on the one hand provides developers with a set of templates and services for building and operating new games or interactive installations upon, and on the other hand offers a web-based environment for the administration and management of such applications. The system handles a number of fundamental and practical issues, such as synchronization, network congestion, delay tolerant communication and neighbor discovery.

Regarding the depth of the implementation issues covered, the overall system architecture is firstly described in an abstraction whereas later each layer is presented in detail. Emphasis is layed on the upper tiers of the architecture, which are the ones responsible for the user interaction, the system administration, the game observation or the design of the installation output. Thus, not only extensive presentation and analysis of technologies behind these layers takes place, report of various ways of concretization, but also explanation of the details and the objectives of each choice. Some key implementation issues of the application domains are discussed while some considerations related to the implementation of games using WSN with several conclusions are presented at the end.

IV

What is clear is that the diversity of the applications created with this framework is huge: large-scale urban games, entertainment or educational sets of quests, installations that amuse and train, ubiquitous systems with uses in museums or galleries, interactive works of art, all feature some kind of interaction between users or between users and the surrounding environment. Intense is the presence of features that do not exist in traditional games, resulting in the traditional games being augmented. Regarding installations, these vary heavily spreading from entertainment to artistic regions.

Χρήση

Η συγκεκριμένη διπλωματική εργασία κατατέθηκε στο Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής του Πανεπιστημίου Πατρών.

Η διπλωματική εργασία αυτή έχει αποκλειστικά επιστημονικό και εκπαιδευτικό χαρακτήρα. Όλα τα εμπορικά σήματα που αναφέρονται σε αυτή, ανήκουν στους αντίστοιχους ιδιοκτήτες.

Τα δικαιώματα της συγκεκριμένης διπλωματικής εργασίας είναι δεσμευμένα από τον Νικόλαο Βασιλάκη και την ομάδα ανάπτυξης της πλατφόρμας. Επιτρέπεται ελεύθερα η αναδημοσίευση των περιεχομένων της με την προϋπόθεση αναφοράς της πηγής τους.

Copyright © 2009 Νικόλαος Βασιλάκης.
basilakn@ceid.upatras.gr

Με την επιφύλαξη παντός δικαιώματος.

Βεβαιώνω ότι έχω διαβάσει αυτή την εργασία και ότι κατά την γνώμη μου, είναι πλήρης σε έκταση και ποιότητα.

Παύλος Σπυράκης

Πανεπιστήμιο Πατρών, Πολυτεχνική Σχολή

Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

GR 265 00 Πάτρα, Ελλάδα

<http://www.ceid.upatras.gr>



Περιεχόμενα

VI

ΠΡΟΟΙΜΙΟ

Περίληψη	iii
Άδεια Χρήσης	v
Περιεχόμενα	vi
Ευχαριστίες	vii

ΕΙΣΑΓΩΓΗ

Εισαγωγή	12
----------	----

PROCESSING

Εισαγωγή	22
Ιστορική Αναδρομή	25
Σύγκριση	27
Βιβλιοθήκες	30
Απόδοση	34
Περιβάλλον Ανάπτυξης	39
Σκίτσα	41
Συντεταγμένες	42
Τρόποι Προγραμματισμού	43
Εξαγωγή Applet	45

LIFERAY

Εισαγωγή	50
Ιστορική Αναδρομή	54
Σύγκριση	59
Χαρακτηριστικά	70
Επίπεδο Χρήστη	73
Portlets	80
Θέματα	89
Ανάπτυξη	96

FUN IN NUMBERS

Εισαγωγή	100
Πρότερη Εργασία	102
Στόχοι	108
Αρχιτεκτονική	110
Φρουροί	113
Μηχανές και Σταθμοί	118
Κόσμος	121

ΥΛΟΠΟΙΗΣΗ

Εισαγωγή	130
Υποδομή	131
Εφαρμογές	133
Πύλη	137
Θεματολογία	148
Εγκαταστάσεις	151
Μεταγλώττιση	162

ΕΠΙΛΟΓΟΣ

Αξιολόγηση	166
Μελλοντική Εργασία	174
Κατακλείδα	175

ΠΑΡΑΡΤΗΜΑ

Βιβλιογραφία	180
Περιεχόμενα CD	186

Ευχαριστίες

Πάτρα, Μάιος 2009

Θα ήθελα σε αυτό το σημείο να ευχαριστήσω τον κ. Παύλο Σπυράκη για την εμπιστοσύνη του.

Επίσης, θα ήθελα να ευχαριστήσω τον κ. Ιωάννη Χατζηγιαννάκη για την βοήθεια, την καθοδήγηση και την παρακίνησή του να ασχοληθώ με το θέμα αυτής της εργασίας. Ο χρόνος που μου αφιέρωσε θα με συντροφεύει σε ολόκληρο το επαγγελματικό μου ταξίδι.

Επιπλέον, θα ήθελα να ευχαριστήσω θερμά τους κκ. Γεώργιο Μυλωνά και Παναγιώτη Κόκκινο για την υποστήριξή τους και τις πολλές κατά καιρούς συζητήσεις μας σε θέματα της συγκεκριμένης ερευνητικής περιοχής.

Ακόμα, θα ήθελα να ευχαριστώ τον κ. Ορέστη Ακριβόπουλο και τον κ. Μάριο Λογαρά, συμφοιτητές μου από το Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής, με τους οποίους πέρασα αρκετό χρό-

νο κατά την ανάπτυξη του συστήματος, συζητώντας ιδέες, ανταλλάσσοντας απόψεις και υλοποιώντας μέρη της πλατφόρμας για περισσότερο από ένα έτος.

Τέλος θα ήθελα να ευχαριστήσω τους φίλους μου για την κατανόηση και την συμπαράσταση που μου έδειξαν όλο αυτό τον καιρό, μα κυρίως την οικογένειά μου που μου παρείχε όλα τα απαραίτητα εφόδια, τόσο υλικά όσο και άυλα, για την επιτυχή επίτευξη των ονείρων μου.



Νικόλαος Βασιλάκης

VII

Our societies are increasingly structured around the bipolar opposition of the Net and the Self.

Εισαγωγή

Εισαγωγή

12

Την ώρα που οι πολιτικές ιδεολογίες και οι οικονομικές θεωρίες περνάνε κρίση, οι επιστήμονες και οι καλλιτέχνες εξακολουθούν να τρέφουν ουτοπίες και φαντασιώσεις, να θέτουν ερωτήματα και προβληματισμούς και να παράγουν έργο με κέντρο τον άνθρωπο, το οποίο ποικίλει σε ποιότητα και ποσότητα. Με την ανθρώπινη παρουσία στο επίκεντρο του ενδιαφέροντος της τέχνης και της τεχνολογίας, οι δύο αυτοί πυρήνες της κοινωνίας πολύ συχνά συνεργάζονται ανοίγοντας νέους δρόμους αφήγησης και ανάγνωσης της εικόνας, ψηλάφησης της αλληλεπίδρασης και κατανάλωσης της ψυχαγωγίας. Εξακολουθούν έτσι να αναπτύσσουν θεωρίες και πρακτικές ενώ υποτάσσουν τις υπηρεσίες τους στην εξυπηρέτηση του ανθρώπινου είδους, εκφράζοντας ολοένα και πιο αποδοτικά την ανθρώπινη δημιουργικότητα και φαντασία.

Εφαρμογές που αντλούν σύσταση και ιδιότητες ισάξια από τα δύο αυτά πεδία, όπως για παράδειγμα τα παιχνίδια ή οι διαδραστικές εγκαταστάσεις, καταφέρνουν να εκφράσουν με επιτυχία την αγωνία των χρηστών με κριτική, χιούμορ και θεατρικότητα. Με προσκήνιο ή παρασκήνιο καίρια κοινωνικά, οικονομικά, φιλοσοφικά, υπαρξιακά και ηθικά θέματα της σύγχρονης ζωής –σεναριακό προσκήνιο στα παιχνίδια, πνευματικό παρασκήνιο στις καλλιτεχνικές εγκαταστάσεις – τείνουν να αποτελούν αναπόσπαστο κομμάτι της ίδιας της ανθρώπινης φύσης επαναπροσδιορίζοντάς την διαρκώς. Μεταξύ ουτοπίας και ρεαλισμού, δημι-

ουργούν νοητούς χώρους όπου ανατρέπουν συμβάσεις και αντιλήψεις για την ζωή προτείνοντας μια νέα γλώσσα για την ομορφιά, την μνήμη, τον χώρο και τον χρόνο.

Με τα όρια μεταξύ των δύο αυτών κόσμων να είναι ολοένα και πιο δυσδιάκριτα, καταλήγουν πολλές φορές να βασίζονται το ένα στο άλλο, εργαλεία της μίας περιοχής σε θεωρίες της άλλης και αντίστροφα. Τα παραδείγματα ξεκινούν από την προϊστορία, συνεχίζουν στους αρχαίους χρόνους, ακολουθούν στον μεσαίωνα καταλήγοντας στο σήμερα σε μία επιτυχημένη προσπάθεια να προσδιορίσουν συνεχώς το μέλλον, με τα συγκεκριμένα στιγμιότυπα να αποτελούν μόνο σημεία σε μια συνεχή γραμμή. Ο συνεχής διαπολιτισμικός διάλογος που αναπτύσσεται ανάμεσα στα δύο αυτά άκρα, ποικίλει σε επίπεδα προσέγγισης [1] και αισθητικές αναφορές και προτείνει τον συνεχή επαναπροσδιορισμό των εννοιών, των τεχνικών και των επιτευγμάτων.

Έτσι, η διασκέδαση αλλάζει νόημα [2,3,4], μετουσιώνεται και καταλήγει στην αφετηρία της, επικεντρώνοντας στην ψυχαγωγία με την ετυμολογική της σημασία. Οι δυνατοί συνδυασμοί δεν ήταν ποτέ τόσο πολλοί, με ποικιλίες τρόπων οι οποίοι εκτίνονται από διαδραστικές εγκαταστάσεις σε μουσεία και καλλιτεχνικές γκαλερί, παιχνίδια που συνδυάζουν το τερπνό και το ωφέλιμο, εφαρμογές μέσα από τις οποίες ο χρή-

Εισαγωγή

στης ενημερώνεται, μαθαίνει, πλάθεται, συστήματα όπου πολλοί παίκτες συνεργάζονται για να επιτύχουν στόχους, νέες μορφές τέχνης που ψυχαγωγούν, ξαφνιάζουν και εγείρουν νέα ερωτήματα. Η συνεργασία μεταξύ τέχνης και τεχνολογίας σε αυτά τα πεδία όχι μόνο δίνει νέες λύσεις αλλά δημιουργεί και νέες ανάγκες, νέες προκλήσεις τις οποίες οι άνθρωποι που εργάζονται και παράγουν έργο στους αντίστοιχους χώρους, καλούνται να καλύψουν και να διαχειριστούν. Αυτό γίνεται ακόμη και σε μορφή αντιδανείου, όταν για παράδειγμα η επιστήμη δημιουργεί μία πρόκληση στην οποία απαντά η τέχνη με την βοήθεια πάλι της τεχνολογικής προόδου. Η απάντηση των ερωτημάτων και των προκλήσεων κάθε περιοχής, πιθανότατα, βρίσκεται στην αντίπερα όχθη – η οποία καταλήγει μάλλον να μην είναι απέναντι αλλά δίπλα, κάτι σαν ομαλή συνέχεια, υγιές συμπλήρωμα ή ακόμα και εννοιολογική επικάλυψη.

Αυτή είναι και η έννοια του διάχυτου ή περιρρέοντος υπολογισμού (στην βιβλιογραφία αναφέρεται ως ubiquitous computing, pervasive computing, ambient intelligence, ή, τελευταία, everyware), ένα νέο μοντέλο αλληλεπίδρασης ανθρώπου και υπολογιστή [5]. Πρόκειται για την περίπτωση όπου η επεξεργασία της πληροφορίας έχει περάσει πια στα αντικείμενα και τις δραστηριότητες της καθημερινότητας. Με την χρήση πολλών υπολογιστών, κινητών, αισθητήρων, απτικών εργαλείων και άλλων συσκευών, ο χρήστης διασυνδέεται στο περιβάλλον του για ενέργειες της καθημερινότητας, μερικές φορές αγνοώντας πλήρως ότι κάτι τέτοιο λαμβάνει χώρα. Το μοντέλο αυτό, διαφέρει πλήρως από τις παραδοσιακές μεθόδους αλληλεπίδρασης όπως οι γραμμές εντολών, τα παράθυρα, τα γραφικά περιβάλλοντα ή οι εξομοιωτές. Η ιδέα πάει ακόμα πιο μακριά με την χρήση ρούχων που «αισθά-

νονται», συσκευών γραφικής εξόδου, σύννεφων αισθητήρων (smart dust), ένδυσης η οποία προβάλλει γραφικά και μικροηλεκτρονικών συσκευών τα οποία προβάλλουν στις τρεις διαστάσεις.

Ενδιαφέρον παρουσιάζει τελευταία ο συνδυασμός με στοιχεία κοινωνικής δικτύωσης. Εξαιρετικά περίπλοκες δομές κατασκευασμένες από κόμβους ατόμων και οργανισμών οι οποίοι συνδέονται με έναν ή περισσότερους τύπους αλληλεξάρτησης έχουν κάνει την εμφάνισή τους τα τελευταία χρόνια. Τα ποσά της ανταλλασσόμενης πληροφορίας από καθημερινά νέα μέχρι χωροχρονικές ιδιότητες αντικειμένων και από αρχεία πολυμέσων μέχρι προσωπικές μαρτυρίες είναι τεράστια. Δεν είναι λίγες οι διαφορετικές διαδικτυακές κοινωνίες ανθρώπων πρόθυμων να μοιραστούν τα ενδιαφέροντα και τις δραστηριότητές τους ή να εξερευνήσουν τα ενδιαφέροντα και τις δραστηριότητες άλλων. Ενώ οι περισσότερες υπηρεσίες είναι βασισμένες στον ιστό με τους χρήστες να αλληλεπιδρούν κυρίως μέσω ηλεκτρονικού ταχυδρομείου και στιγμιαίων μηνυμάτων, διάφορες προγραμματιστικές διεπαφές έχουν κάνει την εμφάνισή τους, αντλώντας δεδομένα έμμεσα από τις ζωές των χρηστών, χωρίς να απαιτείται προσπάθεια από τους τελευταίους [7,8].

Δεν είναι λίγα τα θετικά τα οποία αποκομίζονται από μια τέτοια γεφύρωση. Νέα, ανεξερεύνητα μονοπάτια ανοίγονται για την ανάπτυξη εφαρμογών, γέννησης δεδομένων και κατανάλωσης της πληροφορίας. Δεν αποτελεί μικρό επίτευγμα κάποιος να έχει πληροφόρηση σε πραγματικό χρόνο για την υγεία του παιδιού του καθώς αυτό παίζει, ακόμη και αν αυτός βρίσκεται σε απόσταση εκατοντάδων χιλιομέτρων από το παιδί. Νέες μέθοδοι μάθησης, νέοι τρόποι παρουσίασης της πληροφορίας [9,10], παρακολούθησης της υγείας

Εισαγωγή

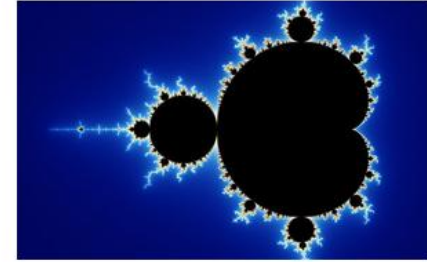
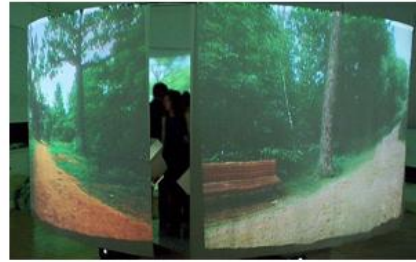
και πρόβλεψης ατυχημάτων αποτελούν πια πραγματικότητα. Οι νέοι τρόποι υπολογισμού στο παρασκήνιο παρουσιάζουν εξαιρετικό ενδιαφέρον, καθώς φέρνουν τις επιστήμες ακόμη πιο κοντά, ενοποιούν τα εργαλεία και καθιστούν τις τέχνες και τα γράμματα εφιαλτήρια για περειαίρω αναζήτηση, ψυχαγωγία, προσδιορισμό και αυτοπροσδιορισμό.

Απαιτείται λοιπόν μία ενοποιημένη πλατφόρμα για την ανάπτυξη διάχυτων εφαρμογών και διαδραστικών εγκαταστάσεων για ψυχαγωγικούς και εκπαιδευτικούς σκοπούς. Με την χρήση κόμβων ασύρματων δικτύων αισθητήρων ως συσκευές αλληλεπίδρασης, η συγκεκριμένη πλατφόρμα θα πρέπει να απλουστεύει την ανάπτυξη εφαρμογών οι οποίες αξιοποιούν δεδομένα εισόδου από ένα πλήθος πηγών όπως αισθητήρες, κάμερες κ.α. προσφέροντας ένα σύνολο υπηρεσιών όπως ανακάλυψη γειτόνων, συγχρονισμός κ.α. Το σύστημα θα καταφέρνει να απαλλάξει τον χρήστη από λεπτομέρειες τέτοιας μορφής, επιλύοντας προβλήματα που αφορούν την αλληλεπίδραση και αποκρύπτοντας αυτή την πολυπλοκότητα στα κατώτερα στάδια της αρχιτεκτονικής και επιτρέποντας στον χρήστη να ασχοληθεί με την υλοποίηση του ονείρου του: Την ανάπτυξη αλληλεπιδραστικών εφαρμογών πολλών χρηστών, είτε αυτές είναι καλλιτεχνικές εγκαταστάσεις είτε παιχνίδια πολλών παικτών.

Το σύστημα που παρουσιάζεται στην συγκεκριμένη εργασία επιτρέπει την δημιουργία, την εγκατάσταση και την διαχείριση πολυχρηστικών ψυχαγωγικών εφαρμογών με περιρρέοντα και τοπικά χαρακτηριστικά. Το κύριο συστατικό των εφαρμογών οι οποίες αναπτύσσονται με την χρήση της συγκεκριμένης πλατφόρμας είναι πως οι χρήστες αλληλεπιδρούν τόσο μεταξύ

τους όσο και με το περιβάλλον με την χρήση χειρονομιών, κίνησης ή τρεξίματος. Με τις κινήσεις αυτές, οι χρήστες πραγματοποιούν ενέργειες οι οποίες σχετίζονται με το παιχνίδι, με την χρήση συσκευών που φέρουν αισθητήρες. Το σύστημα βασίζεται σε μία αρχιτεκτονική πολλών επιπέδων η οποία από την μία εφοδιάζει τους προγραμματιστές με ένα σύνολο οδηγιών και υπηρεσιών για την ανάπτυξη νέων παιχνιδιών και διαδραστικών εγκαταστάσεων ενώ από την άλλη προσφέρει ένα περιβάλλον το οποίο βασίζεται στον ιστό για την διαχείριση και τον έλεγχο τέτοιων εφαρμογών. Το ίδιο το σύστημα επί της ουσίας χειρίζεται έναν αριθμό πρακτικών ζητημάτων, τα οποία κατά παράδοση παρουσιάζονται στην ανάπτυξη τέτοιων εφαρμογών όπως ο συγχρονισμός, η συμμόρφωση του δικτύου, η ανεκτική σε καθυστερήσεις επικοινωνία και η ανακάλυψη γειτόνων.

Σε αυτή την εργασία, στόχο αποτελούν ασχολίες τις οποίες οι άνθρωποι εκτελούν όταν επισκέπτονται μουσεία, αρχαιολογικούς χώρους, φυσικά ή τεχνικά πάρκα κλπ. Σε αυτού του είδους τις ασχολίες, υπάρχουν ήδη προσωπικές και προσωποποιημένες περιηγήσεις, πλατφόρμες ενίσχυσης της πραγματικότητας καθώς και άλλες τεχνολογίες. Σε προηγούμενες ασχολίες οι οποίες γίνονται μεταξύ ανθρώπων, περιλαμβάνονται η διδασκαλία, οι αγορές, τα παιχνίδια κ.α. Η τεχνολογία γίνεται σταδιακά μέρος της ζωής των ανθρώπων με ένα αυξανόμενο πλήθος τρόπων. Πιθανή ανάμιξη της παιδικής ψυχαγωγίας κάνει τον σχεδιασμό και την ανάπτυξη τέτοιων δραστηριοτήτων μια ενδιαφέρουσα πρόκληση. Είναι σημαντικό να είναι πολύ οργανωμένες ώστε να είναι απλές στην αλληλεπίδραση τους με τους χρήστες, να διατηρούν συνεχώς



1. Αυτόματο Εικονικό Περιβάλλον Σπηλιάς (Cave Automatic Virtual Environment - CAVE). Βυθιζόμενο περιβάλλον εικονικής πραγματικότητας, ACM SIGGRAPH από το Εργαστήριο Ηλεκτρονικής Οπτικοποίησης του Πανεπιστημίου του Illinois (Electronic Visualization Laboratory at University of Illinois), 1992.
2. Τοπίο Ένα, (Landscape One). Εγκατάσταση εικονικής πραγματικότητας Ars Electronica Festival, Linz, Αυστρία από τον Luc Courchesne, 1999.
3. Μορφόκλασμα Μάντελμπροτ (Mandelbrot Fractal), σε συνεχώς χρωματισμένο περιβάλλον.
4. Ανώνυμο. Εγκατάσταση ψηφιακής τέχνης, μπιενάλε του Sao Paolo από τον Pascal Dombis, 2008.
- 5,6. Το Τούνελ κάτω από τον Ατλαντικό (The Tunnel under the Atlantic). Διαδραστική εγκατάσταση εικονικής πραγματικότητας, Pompidou Centre, Παρίσι και Museum of Contemporary Art, Montreal από τον Maurice Benayoun, 1995.

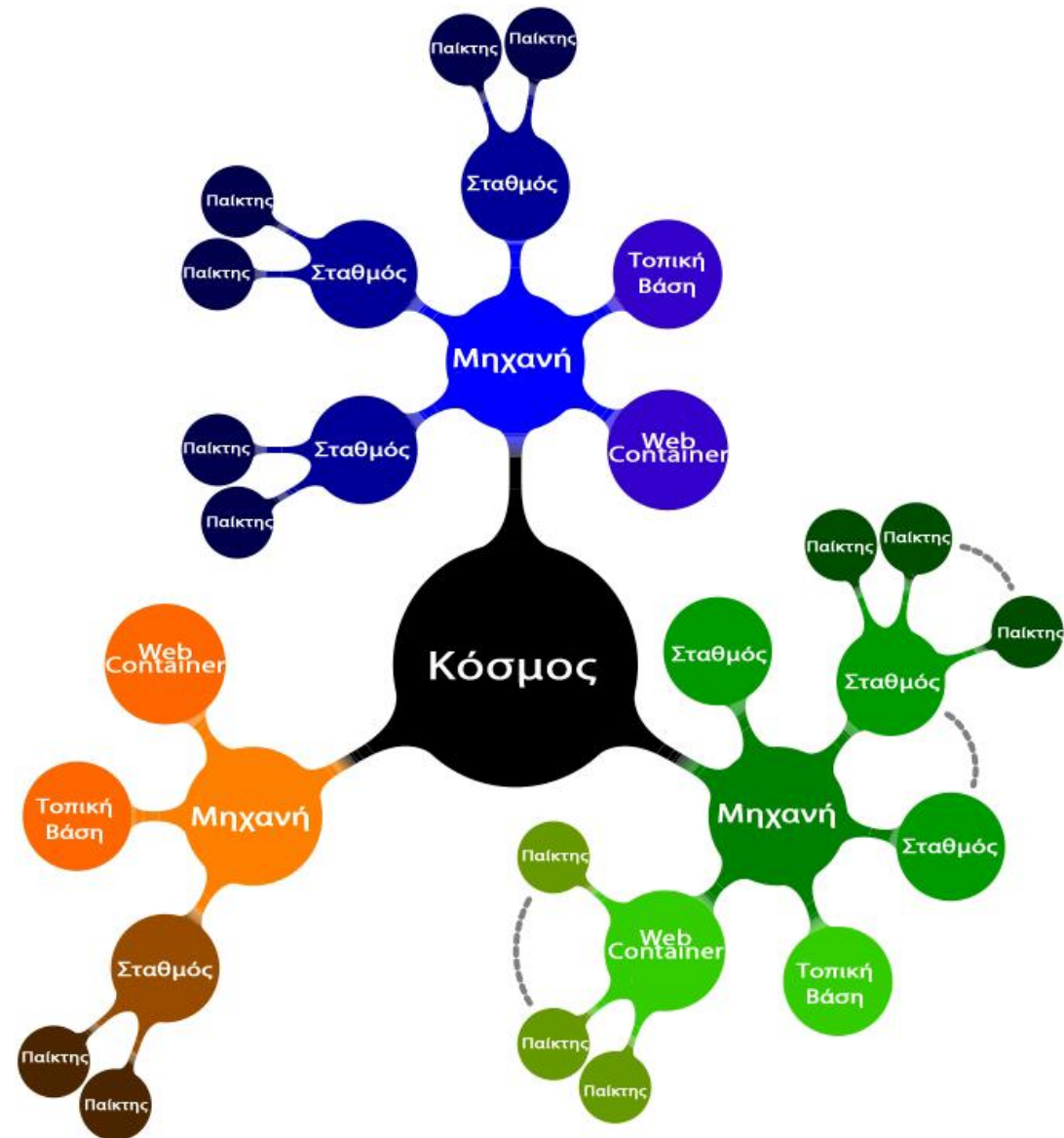
Εισαγωγή

την προσοχή των παικτών, και φυσικά να προσφέρουν στα παιδιά μία ευχαρίστηση ή ψυχαγωγία. Όταν γίνεται αναφορά στην διασκέδαση, δεν εννοείται απαραίτητα η διασκέδαση η οποία προσφέρεται από διάφορες πλατφόρμες παιχνιδιών όπως το Nintendo Wii, το Microsoft Xbox360 κλπ. Ο όρος πιο πολύ αναφέρεται σε ψυχαγωγικές εγκαταστάσεις οι οποίες ωθούν τα παιδιά να ασχοληθούν, να τρέξουν, να σκεφτούν, να καταλάβουν και να αλληλεπιδράσουν. Αυτό είναι ορθά προσανατολισμένο προς την σύγχρονη τάση της αποκόλλησης από τα παραδοσιακά περιβάλλοντα παιχνιδιού, η οποία είναι εμφανής τόσο λόγω των κινητών ψυχαγωγικών συστημάτων όπως το Sony PSP και το Nintendo DS αλλά και των κινητών τηλεφώνων με δυνατότητες παιχνιδιού όπως το iPhone.

Σε αυτή την εργασία προτείνεται επίσης ένας αριθμός τέτοιων εγκαταστάσεων οι οποίες βασίζονται στην χρήση κινητών δικτύων αισθητήρων. Το πεδίο των ασύρματων δικτύων αισθητήρων είναι ένα πεδίο το οποίο έχει ερευνηθεί αρκετά, τόσο στο θεωρητικό όσο και στο πρακτικό του επίπεδο, και αυτή η έρευνα μπορεί να χρησιμοποιηθεί –ή καλύτερα να μετουσιωθεί– σε ένα πλήθος υπηρεσιών και προϊόντων. Ακόμα, οι τελευταίες πρόοδοι όσον αφορά τα ad-hoc δίκτυα, τα κινητά τηλέφωνα και τους αισθητήρες έχουν οδηγήσει στην γέννηση νέων προϊόντων, τα οποία ενσωματώνουν διάφορα ήδη αισθητήρων στις κινητές συσκευές σε λογικό μάλιστα κόστος. Ο συνδυασμός μεταξύ αισθητήρων και κινητών συσκευών με σκοπό την δημιουργία νέων συναρπαστικών εφαρμογών κρύβει πολλά πλεονεκτήματα και παραμένει άγνωστη γη σε πολλά επίπεδα. Από αυτή την άποψη, παρουσιάζεται ένας αριθμός υλοποιημένων εφαρμογών καθώς και προτείνεται ένας αριθμός άλλων.

Η βασική ιδέα εδώ είναι πως στον χρήστη (π.χ. ένα παιδί το οποίο επισκέπτεται ένα μουσείο ή μία έκθεση τέχνης) δίνεται μία κινητή συσκευή, δια μέσου της οποίας ο χρήστης μπορεί να συμμετάσχει σε έναν αριθμό από δικτυακές διαδραστικές εγκαταστάσεις οι οποίες μάλιστα εναρμονίζουν και μία ψυχαγωγική ή εκπαιδευτική διάσταση. Η συμμετοχή των παιδιών σε ένα ψυχαγωγικό γεγονός εγκαινιάζεται απλώς με την προσέγγιση στην εγκατάσταση (π.χ. βιντεοπροβολές, μηχανικές εγκαταστάσεις, διακόπτες κλπ). Με την χρήση κινήσεων, τρεξίματος και χειρονομιών για την εκτέλεση ενεργειών σχετιζόμενων με το παιχνίδι με την βοήθεια των αισθητήρων, ο χρήστης συμμετέχει και αλληλεπιδρά με τις εγκαταστάσεις. Τα πλεονεκτήματα της χρήσης αισθητήρων όπως στην περίπτωση του Nintendo Wii είναι εμφανή και έχουν οδηγήσει σε προϊόντα με μεγάλη εμπορική επιτυχία. Όταν το παιδί θέλει να αφήσει μία εγκατάσταση και να επισκεφτεί κάποια άλλη, μπορεί είτε να ξεκινήσει εκ νέου να αλληλεπιδρά είτε να συνεχίσει μία ήδη υπάρχουσα διάδραση. Επίσης, μία κεντρική οντότητα μαζεύει δεδομένα τα οποία σχετίζονται με την αλληλεπίδραση του χρήστη και ενισχύει τις νέες ενέργειες βασισμένη σε προϋπάρχουσα γνώση ή εμπειρία η οποία έχει εξαχθεί σε προγενέστερη χρήση του συστήματος.

Τέτοιες ψυχαγωγικές εγκαταστάσεις παρουσιάζουν ένα πλήθος κοινών χαρακτηριστικών τα οποία είναι σημαντικό να ταχτοποιηθούν με σκοπό την ανάδειξη των πλεονεκτημάτων μίας τέτοιας προσέγγισης. Οι αλληλεπιδράσεις μεταξύ των παιδιών και των εγκαταστάσεων βασίζονται στην αίσθηση της παρουσίας και της κίνησης (εκτέλεση συναφών ενεργειών). Και οι δύο παρέχονται με την χρήση ασύρματων τεχνολογιών με αισθητήρες. Επίσης, οι εγκαταστάσεις μπο-



Σχηματική αναπαράσταση της πολυεπίπεδης αρχιτεκτονικής του συστήματος.

Εισαγωγή

ρούν να λειτουργήσουν με πληθώρα συμμετεχόντων, με την άμεση προσθήκη πολυχρηστικών (multiplayer) χαρακτηριστικών στις εγκαταστάσεις. Οι χρήστες διαφοροποιούνται εύκολα βάσει της συσκευής την οποία μεταφέρουν, χωρίς αυτό να σημαίνει πως προσωπικά δεδομένα βρίσκονται σε κίνδυνο, καθώς η ταυτότητα των χρηστών είναι συνδεδεμένη μόνο έμμεσα με αυτές. Τα παιδιά μπορούν πολύ εύκολα να ανταλλάξουν συσκευές και να συνεχίσουν να παίζουν – ακόμα φυσικά και στην περίπτωση των δοκιμασιών σε μουσεία και εκθέσεις. Ακόμα, μία κεντρική οντότητα υπάρχει για τον συντονισμό και την διαχείριση των εγκαταστάσεων.

Για την ανάπτυξη, την εγκατάσταση και τον έλεγχο τέτοιων εγκαταστάσεων λυτόν δημιουργήθηκε αυτή η ενοποιημένη, τόσο από άποψη υλικού όσο και από άποψη λογισμικού, πλατφόρμα. Γίνεται χρήση των Sn Sports [11] ως υλικό των κόμβων της πλατφόρμας για την πρότυπη υλοποίηση, τα οποία προσφέρουν βασικές λειτουργίες ασύρματων κόμβων. Η πλατφόρμα βασίζεται σε μία ιεραρχία επιπέδων, για λόγους ευελιξίας και εύκολης προσαρμογής για ένα πλήθος εγκαταστάσεων. Ένα σύνολο υπηρεσιών έχουν υλοποιηθεί, οι οποίες είναι κοινές σε όλες τις εγκαταστάσεις και τα παιχνίδια, επιτρέποντας χωρική συναίσθηση των συσκευών σε εσωτερικούς χώρους, εκτέλεση εργασιών αίσθησης εν κινήσει, συντονισμού καταναμεμένων λειτουργιών.

Στο τεχνικό κομμάτι, γίνεται αρχικά μία περιγραφή της Processing [12], μίας βιβλιοθήκης για γραφικά, ήχο και γενικότερα διάδραση, με την οποία χτίστηκε το ανώτατο επίπεδο της πλατφόρμας. Έτσι, γίνεται μια παρουσίαση του τί είναι η Processing, ποιούς σκοπούς φιλοδοξεί να επιτύχει και ποιά υπάρχοντα κενά προσδοκεί να καλύψει. Προς αυτή την κατεύθυνση προσανατολίζεται τόσο η ιστορική αναδρομή που ακολουθεί, ώστε να

γίνει εμφανές πως προέκυψε μία τέτοια πλατφόρμα, αλλά και η σύγκριση με άλλες γλώσσες, η οποία φανερώνει τις ιδιαιτερότητές της – σε καμία περίπτωση όμως δεν φιλοδοξεί να είναι πληρέστερη από τις πηγές που υπάρχουν διάσπαρτες στο διαδίκτυο. Έπειτα, ακολουθεί η παρουσίαση των ενδογενών βιβλιοθηκών της Processing, βιβλιοθήκες οι οποίες αφορούν γραφικά, ήχο, έλεγχο υλικού, δίκτυο, διανυσματικές εικόνες κ.α. οι οποίες δείχνουν δυνατότητες της γλώσσας αυτής. Προς αυτή την κατεύθυνση στρέφεται και η επόμενη ενότητα, η οποία αφορά την απόδοση των γραφικών από τον υπολογιστή και τις δυνατότητες της Processing στο θέμα αυτό. Έπεται μία περιγραφή του περιβάλλοντος ανάπτυξης και του βιβλίου σχεδίων, δύο σημεία που θα πρέπει να σταθεί κάποιος ο οποίος κάνει τα πρώτα του βήματα στην πλατφόρμα αυτή. Τελειώνοντας, οι τρόποι προγραμματισμού αποτελούν ένα ελαφρά πιο τεχνικό κομμάτι, με την συμπερίληψη κομματιών κώδικα, ενώ η εξαγωγή Applet τελειώνει το κεφάλαιο, επιτρέποντας σε έναν προγραμματιστή να καταλάβει τον τρόπο με τον οποίο γίνονται ορατά τα τελικά αποτελέσματα της δουλειάς του στον τελικό χρήστη.

Στο δεύτερο κεφάλαιο περιγράφεται η τεχνολογία πίσω από την διαδικτυακή πύλη, η οποία αποτελεί το ανώτατο και πιο κεντριοποιημένο μέρος της πλατφόρμας, επιτρέποντας την διαχείριση πολλαπλών παιχνιδιών, φυσικών περιοχών διεξαγωγής παιχνιδιού και χρηστών. Συγκεκριμένα γίνεται μία εισαγωγή στις διαδικτυακές πύλες γενικά καθώς και σε συγκεκριμένα χαρακτηριστικά του Liferay [13]. Επίσης, παρουσιάζεται μία ιστορική αναδρομή της διαχείρισης περιεχομένου στον ιστό, ενώ γίνεται σύγκριση της συγκεκριμένης πύλης με άλλες υπάρχουσες. Έπειτα γίνεται μία παρουσίαση των δυνατών αλληλεπιδράσεων σε επι-

Εισαγωγή

πεδο χρήστη, και επιχειρείται μία εισαγωγή σε ελαφρώς πιο τεχνικά ζητήματα. Έτσι παρουσιάζονται εκτενώς τα portlets και τα θέματα, τα οποία αποτελούν τα δομικά στοιχεία της πύλης, τόσο από την σκοπιά του χρήστη όσο και του προγραμματιστή. Τελειώνοντας, γίνεται περιγραφή του περιβάλλοντος ανάπτυξης.

Ενώ έχουν περιγραφεί εκτενώς οι τεχνολογίες πίσω από το ανώτατο επίπεδο της αρχιτεκτονικής, στο τρίτο κεφάλαιο περιγράφεται το σύνολο της πλατφόρμας. Αρχικά επιχειρείται μία ταυτοποίηση των σημαντικότερων χαρακτηριστικών τα οποία συνθέτουν τις απαιτήσεις ενός τέτοιου εγχειρήματος. Γίνεται επίσης μία διερεύνηση σε προηγούμενες εργασίες και ταυτοποιούνται οι στόχοι του συστήματος. Έπειτα, ακολουθεί η σφαιρική παρουσίαση όλων των επιπέδων της αρχιτεκτονικής του συστήματος ενώ έπονται τρεις ενότητες που περιγράφουν αναλυτικότερα και με μεγαλύτερη λεπτομέρεια το κομμάτι των επιλογών του κάθε επιπέδου της αρχιτεκτονικής ξεχωριστά.

Στο επόμενο κεφάλαιο, στόχος είναι η περιγραφή της υλοποίησης, σε μια πιο τεχνική προσέγγιση. Έτσι, γίνεται μια παρουσίαση των παιχνιδιών και των εγκαταστάσεων οι οποίες έχουν υλοποιηθεί. Παρουσιάζεται η υλική υποδομή κάθε επιπέδου. Επίσης, επιλέγονται συγκεκριμένα σημεία τα οποία αποτελούν χαρακτηριστικά σημεία και παρουσιάζονται εκτενώς. Έτσι γίνεται αιτιολόγηση των εκάστοτε επιλογών, παρουσιάζονται οι λεπτομέρειες της διεπαφής και εμφανίζονται οι λεπτομέρειες πίσω από το περιεχόμενο της διαδικτυακής πύλης. Γίνεται λεπτομερής παρουσίαση των αρχείων που συνθέτουν τα δομικά στοιχεία των ανώτατων επιπέδων της υλοποίησης και το κεφάλαιο τελειώνει με θέματα μεταγλώττισης της

πλατφόρμας. Ο αναγνώστης είναι σε θέση να παίξει και να αξιολογήσει το αποτέλεσμα μόνος του.

Ακολουθεί ο επίλογος στον οποίο συνοψίζεται κριτικά η εργασία καθώς γίνεται ειδική μνεία σε συγκεκριμένα κομμάτια και εξάγονται κάποια ενδιαφέροντα συμπεράσματα. Γίνεται αξιολόγηση των μερών της πλατφόρμας με ένα πλήθος διαφορετικών τρόπων, και διαφαίνονται τα πλεονεκτήματα και τα μειονεκτήματα κάθε επιλογής. Επίσης, γίνεται αναφορά σε μελλοντική εργασία, καθώς και μεταφορά του συστήματος σε άλλες πλατφόρμες. Τελειώνοντας, η τελευταία ενότητα θα μπορούσε να αποτελέσει συμπερασματική περίληψη, αφορμή για προβληματισμό ή θέμα προς συζήτηση καθώς επίσης γίνεται μνεία σε καίριες στιγμές της πλατφόρμας.

Στο παραρτήματα επίσης γίνεται παρουσίαση της βιβλιογραφίας στην οποία θα πρέπει να ανατρέξει κάποιος ο οποίος ενδιαφέρεται περισσότερο για κάποια από τα ζητήματα τα οποία καλύπτονται σε αυτή την εργασία. Τέλος παρουσιάζονται και τα περιεχόμενα του οπτικού δίσκου ο οποίος συνοδεύει την εργασία αυτή.

Vision reaches beyond the thing that is, into the conception of what can be.
Imagination gives you the picture. Vision gives you the impulse to make the picture your own.

Processing

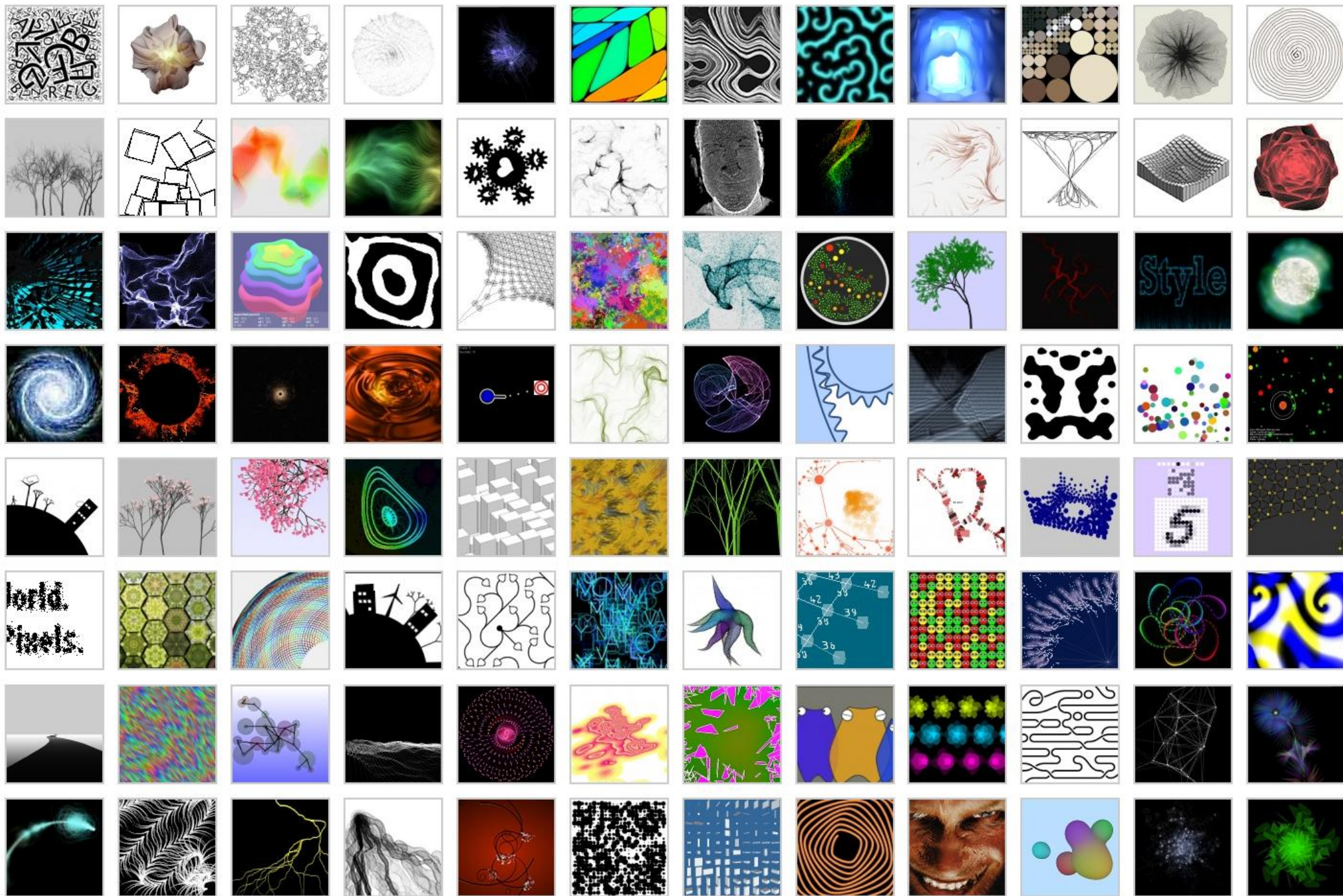
Εισαγωγή

22

Η Processing είναι μία ανοικτού κώδικα γλώσσα προγραμματισμού καθώς και περιβάλλον ανάπτυξης για ανθρώπους οι οποίοι ασχολούνται με τον προγραμματισμό στατικής, κινούμενης εικόνας και γενικότερα πολύπλοκων και περίτεχνων αλληλεπιδράσεων μεταξύ ανθρώπου και μηχανής. Χρησιμοποιείται ευρέως από σπουδαστές -τόσο σε προπτυχιακό όσο και μεταπτυχιακό επίπεδο-, καλλιτέχνες, σχεδιαστές, ερευνητές καθώς και χομπίστες, εισάγοντας πολλαπλά οφέλη σε τομείς όπως η εκμάθηση αρχών προγραμματισμού, η καινοτομία σε πολλές περιοχές οι οποίες άπτονται άμεσα των οπτικοακουστικών επινοήσεων αλλά και η κάθετη αύξηση της παραγωγικότητας. Αρχικά δημιουργήθηκε για την διδασκαλία των αρχών προγραμματισμού υπολογιστών μέσα από ένα οπτικά παραστατικό πρίσμα. Αποτελεί έναν καμβά λογισμικού για την ανάπτυξη προχωρημένων, αφηρημένων και μη, γραφικών εννοιών καθώς και ένα επαγγελματικό εργαλείο για την παραγωγή αυτών. Μερικά ακόμη από τα κύρια πλεονεκτήματα της είναι ότι αποτελεί μία εναλλακτική λύση σε αντίστοιχα πακέτα κλειστού κώδικα καθώς και το ότι είναι διαθέσιμη για GNU/Linux, Mac OS και Windows.

Έκανε την εμφάνισή της την άνοιξη του 2001 και χρησιμοποιήθηκε αρχικά σε ένα εργαστήριο (workshop) στην Ιαπωνία τον Αύγουστο του ίδιου έτους. Αρχικά χτισμένη ως μεμονωμένη (domain-specific) επέκταση στην Java με στόχο να εξυπηρετήσει καλλιτέχνες και σχεδιαστές, έχει εξελιχθεί πια σε ένα πραγματικό εργαλείο σχεδίου και διαμόρφωσης πρωτοτύπων που χρησιμοποιείται για μεγάλης κλίμακας εγκαταστάσεις, κινούμενη εικόνα και πολύπλοκες οπτικοποιήσεις δεδομένων. Αποτελεί ένα απλό περιβάλλον προγραμματισμού που δημιουργήθηκε για να κάνει ευκολότερη την ανάπτυξη εφαρμογών προσανατολισμένες σε γραφικά με έμφαση στην κίνηση και να παρέχει στους χρήστες της άμεση ανατροφοδότηση (feedback) μέσω διαρκούς αλληλεπίδρασης με το τελικό έργο. Καθώς οι δυνατότητες της έχουν επεκταθεί κατά τη διάρκεια των τελευταίων έξι ετών, η Processing έχει χρησιμοποιηθεί σε πιο εξελιγμένα έργα επαγγελματικής παραγωγής εκτός από την εκτεταμένη χρήση της σε σχέδια.

Ένας σημαντικός στόχος για την Processing ήταν να κατασταθεί αυτός ο τύπος προγραμματισμού προσι-



Παραδείγματα υλοποιημένα σε Processing. Πολλά από αυτά περιλαμβάνουν τρισδιάστατα γραφικά, δημιουργία γραμματοσειρών, κινούμενη εικόνα, στερεοσκοπική απεικόνιση (αναγκαία η χρήση γυαλιών 3D για την πλήρη απεικόνιση στα συγκεκριμένα παραδείγματα), επεξεργασία εικόνας, αλληλεπίδραση με τον χρήστη, πειραματική εξομοίωση κ.α. ή συνδυασμός αυτών.

Εισαγωγή

τός στο ευρύτερο κοινό. Για αυτόν τον λόγο, η Processing είναι ανοικτού κώδικα, ελεύθερη να την κατεβάσει ο οποιοσδήποτε και ελεύθερη προς χρήση και τροποποίηση. Επίσης τα προγράμματα που αναπτύχθηκαν και αναπτύσσονται με την χρήση του περιβάλλοντος ανάπτυξης και των ενσωματωμένων βιβλιοθηκών της Processing μπορούν να χρησιμοποιηθούν για οποιοδήποτε σκοπό. Αυτό το πρότυπο είναι ίδιο με το GCC, την GNU Συλλογή Μεταγλωττιστών (GNU Compiler Collection). Το GCC και οι σχετικές βιβλιοθήκες του (π.χ., libc) είναι ανοικτού κώδικα με άδεια χρήσης την GNU (GPL), η οποία ορίζει ότι οποιεσδήποτε αλλαγές στον κώδικα πρέπει να καθίστανται ορατές δημόσια. Εντούτοις, προγράμματα που δημιουργούνται με το GCC (με τα παραδείγματα να μην είναι λίγα) δεν απαιτούνται να είναι τα ίδια ανοικτού κώδικα.

Η Processing αποτελείται από:

- Το περιβάλλον ανάπτυξης της Processing (Processing Development Environment - PDE). Αυτό είναι το λογισμικό που τρέχει όταν κάποιος κάνει διτλό κλικ στο εικονίδιο της Processing. Το PDE είναι ένα Ενσωματωμένο Περιβάλλον Ανάπτυξης με ένα μινιμαλιστικό σύνολο χαρακτηριστικών γνωρισμάτων που έχουν σχεδιαστεί ως απλή εισαγωγή στον προγραμματισμό ή για τη δοκιμή ιδεών "μιας χρήσεως".

- Μια συλλογή των εντολών (επίσης αποκαλούμενων ως λειτουργίες ή μέθοδοι) οι οποίες αποτελούν τον πυρήνα της διεπαφής προγραμματισμού, ή API, καθώς επίσης και διάφορες βιβλιοθήκες που υποστηρίζουν πιο προηγμένα χαρακτηριστικά, όπως η σχεδίαση με

OpenGL, ανάγνωση αρχείων XML, και αποθήκευση σύνθετων καλογολικών στοιχείων σε μορφή PDF.

- Μια γλωσσική σύνταξη, ίδια με την Java αλλά με μερικές τροποποιήσεις - οι διαφορές αναλύονται εκτενέστερα παρακάτω.

- Μια ενεργό κοινότητα, που φιλοξενείται στο <http://Processing.org>.

Λόγω των παραπάνω, οι αναφορές Processing μπορεί να είναι κάπως διφορούμενες. Μιλάμε για το API, το περιβάλλον ανάπτυξης, ή τον ιστοχώρο; Στην δική μας προσέγγιση, αν δεν υπάρχει συγκεκριμένη αναφορά για διαφορετική ερμηνεία αναφερόμαστε στο σύνολο των συναρτήσεων, την ίδια την γλώσσα προγραμματισμού δηλαδή.

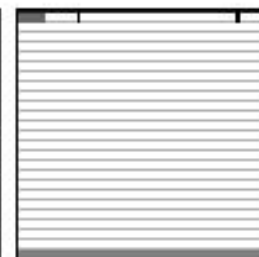
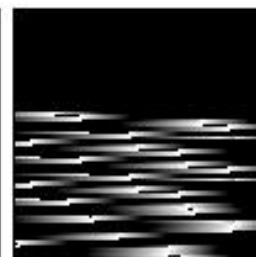
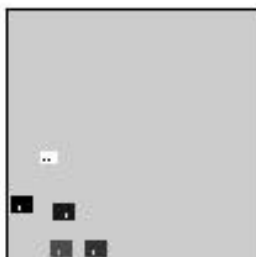
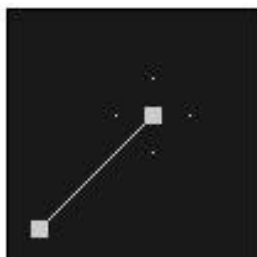
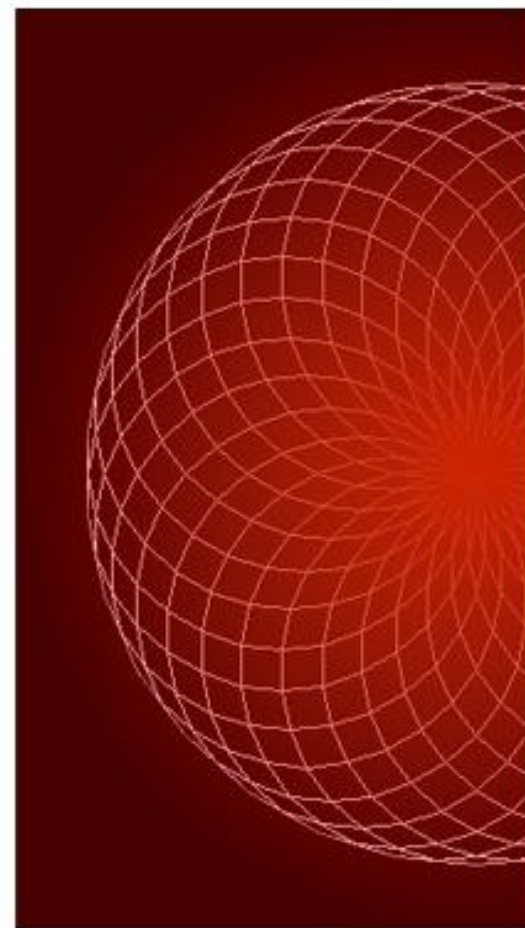
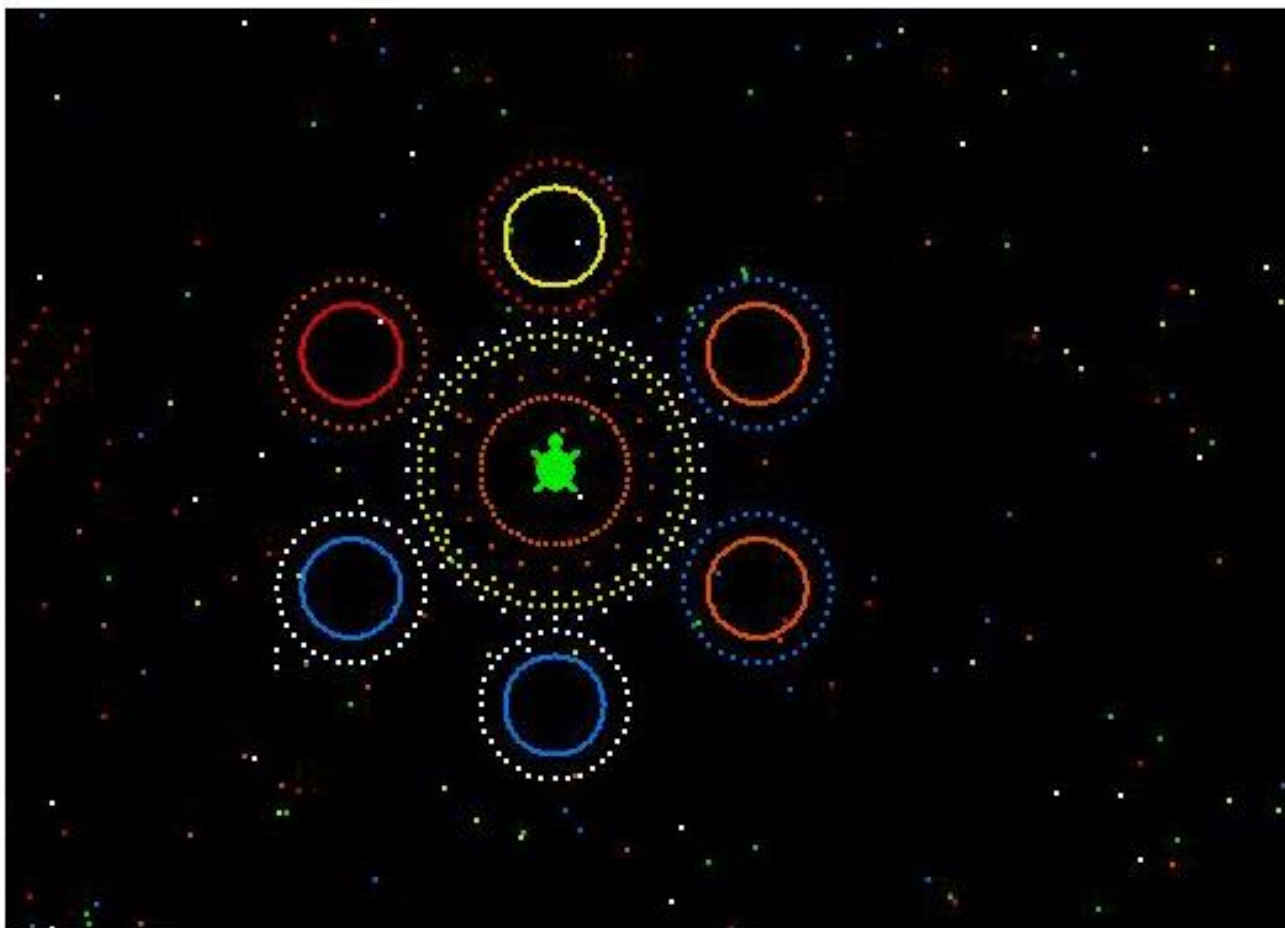
Ιστορική Αναδρομή

Η Processing γεννήθηκε από τους Ben Fry και Casey Reas, μέσα από ιδέες οι οποίες διερευνήθηκαν στην Ομάδα Αισθητικής και Υπολογιστών στο Εργαστήριο Πολυμέσων [14] του Τεχνολογικού Ινστιτούτου της Μασαχουσέτης (Aesthetics and Computation Group, MIT Media Lab). Πολύ γρήγορα το έργο βρήκε υποστήριξη από έναν αριθμό εθελοντών οι οποίοι προσέφεραν κώδικα εθελοντικά στην κοινότητα. Από το σημείο αυτό ξεπήδησαν και άλλες ιδέες οι οποίες σχετίζονται άμεσα με την Processing όπως το Arduino και το Wiring, υλικό το οποίο έχει να κάνει με την δημιουργία διαδραστικών και καλλιτεχνικών έργων. Πρέπει να σημειωθεί ότι η Processing δεν είναι η πρώτη γλώσσα προγραμματισμού ή περιβάλλον προς αυτή την κατεύθυνση.

Γυρνώντας τον χρόνο τρεις δεκαετίες πίσω, κάνει την εμφάνισή της η Logo [16] (ένα αρκτικόλεξο που αντιστοιχεί στο Logic Oriented Graphic Oriented - Προσανατολισμένο στην Λογική Προσανατολισμένο στα Γραφικά), μια γλώσσα προγραμματισμού που χρησιμοποιείται για διαδικαστικό προγραμματισμό. Με την Logo, ένας χρήστης οδηγεί μέσω εντολών μία χελώνα για την δημιουργία σχημάτων και σχεδίων. Είναι μια προσαρμογή -καλύτερα διάλεκτος- της γλώσσας προγραμματισμού Lisp. Μερικοί μάλιστα αποκαλούν την Logo ως Lisp χωρίς τις παρενθέσεις. Σήμερα, είναι γνωστή κυρίως για τη γραφική παράσταση χελώνας (μία μέθοδος που αντιστοιχεί μία χελώνα σε έναν κέρσορα καρτεσιανών συντεταγμένων), αλλά έχει επίσης σημαντικές δυνατότητες για το χειρισμό λιστών, αρχείων, εισόδου/εξόδου και αναδρομής. Η Logo δημιουργήθηκε το 1967 για εκπαιδευτική χρήση, με

κατεύθυνση την κατασκευαστική διδασκαλία, από το Daniel G. Bobrow, τον Wally Feurzeig και τον Seymour Papert. Μπορεί να χρησιμοποιηθεί για να διδάξει τις περισσότερες έννοιες πληροφορικής, όπως κάνει ο ομιλητής Brian Harvey του UC Μπέρκλεϊ στην τριλογία [15] Computer Science Logo Style (Η Επιστήμη των Υπολογιστών σε Στυλ Logo).

Έπειτα, το 1999, παρουσιάστηκε η Design by Numbers [17]. Ο "Σχεδιασμός με Αριθμούς" (ή DBN) ήταν ένα πείραμα με μεγάλες επιδράσεις στην διδασκαλία προγραμματισμού που άρχισε στο MIT Media Lab κατά τη διάρκεια της δεκαετίας του '90. Οδηγημένοι από τον John Maeda, οι σπουδαστές του δημιούργησαν το συγκεκριμένο λογισμικό που στόχευε στο να βοηθήσει σχεδιαστές, καλλιτέχνες και άλλους μη-προγραμματιστές να αρχίσουν να μαθαίνουν προγραμματισμό υπολογιστών. Η DBN, λοιπόν, ήταν μία γλώσσα η οποία έκανε εύκολη την πρόσβαση στον προγραμματισμό σε σχεδιαστές και καλλιτέχνες καθώς με αισθητή ευκολία, βοηθούσε ανθρώπους που ασχολούνταν με την δημιουργία οπτικού υλικού να επιτύχουν σε μεγάλο βαθμό τους στόχους τους, ιδιαίτερα όταν απαιτούνταν η χρήση υπολογιστών και προγραμματισμού. Το ίδιο το λογισμικό μπορούσε να τρέξει σε φυλλομετρητή ιστού και δημοσιεύτηκε μαζί με ένα βιβλίο και επιπλέον διδακτικό υλικό. Η DBN δεν είναι πλέον έργο εν ενεργεία αλλά επηρέασε πολλά άλλα προγράμματα που στοχεύουν στο να καταστήσουν τον προγραμματισμό υπολογιστών πιο προσεγγίσιμο σε ανθρώπους που δεν έχουν την απαραίτητη κατάρτιση.



Παραδείγματα γραφικών τα οποία δημιουργήθηκαν με χρήση Logo και Design By Numbers. Τα πρώτα δύο είναι με χρήση Logo και τα υπόλοιπα με την χρήση της DBN.

Σύγκριση

Java

Η σύγκριση της Processing με άλλες γλώσσες φέρνει στο φως τις δυνατότητές της, τις αδυναμίες της καθώς και τις ιδιαιτερότητές της. Όχι μόνο βοηθά τους χρήστες να επιλέξουν το κατάλληλο εργαλείο που θα χρησιμοποιήσουν σε κάθε περίπτωση και απαντά στο ερώτημα αν η Processing είναι κατάλληλη για τον σκοπό που επιθυμεί κάποιος να την χρησιμοποιήσει, αλλά βοηθάει και ανθρώπους που είναι οικείοι με άλλα πακέτα ή γνωρίζουν παρεμφερείς γλώσσες προγραμματισμού να μεταφέρουν και να χρησιμοποιήσουν την γνώση αυτή αποδοτικά στο περιβάλλον της Processing.

Η Java [18] είναι μια γλώσσα προγραμματισμού που αναπτύχθηκε αρχικά από τον James Gosling στη Sun Microsystems και εκδόθηκε το 1995 ως βασικό τμήμα της πλατφόρμας Java της Sun (Sun Microsystems' Java Platform). Η γλώσσα αντλεί ένα μεγάλο μέρος της σύνταξής της από τη C και C++ αλλά έχει ένα απλούστερο πρότυπο αντικειμένων και λιγότερες χαμηλού επιπέδου δυνατότητες. Οι εφαρμογές σε Java μεταφράζονται (compile) σε bytecode (αρχείο .class) που μπορεί να τρέξει σε οποιαδήποτε εικονική μηχανή της Java (Java Virtual Machine - JVM) ανεξάρτητα από την αρχιτεκτονική του υπολογιστή στον οποίο αυτή τρέχει.

Το περιβάλλον της Processing είναι γραμμένο σε Java. Τα προγράμματα που είναι γραμμένα σε Processing μεταφράζονται επίσης σε Java και έπειτα τρέχουν ως Java προγράμματα. Εν γένει, προγράμματα γραμμένα σε Java και Processing τρέχουν πιο γρήγορα απ' ό,τι προγράμματα βασισμένα σε γλώσσες σεναρίων (scripting languages) όπως η ActionScript ή η Lingo, κάτι το οποίο είναι πολύ σημαντικό για τις περισσότερες εφαρμογές γραφικών. Σημαντικές διαφορές μεταξύ της Processing και της Java είναι καταρχήν η βιβλιοθήκη της Processing για τα γραφικά, και έπειτα ένας πιο απλοποιημένος / αφηρημένος τρόπος προγραμματισμού, ο οποίος δεν απαιτεί από τους χρήστες να αντιλαμβάνονται έννοιες όπως οι κλάσεις (class), τα αντικείμενα (object) ή η κινούμενη εικόνα (animation). Πρέπει να τονιστεί πως τα παραπάνω χαρακτηριστικά δεν απουσιάζουν και είναι προσβάσιμα από τους πιο προχωρημένους χρήστες, ώστε να υπάρχει μεγαλύτερος έλεγχος στο τελικό αποτέλεσμα. Υπό κανονικές συνθήκες, αποτελούν λεπτομέρειες τεχνικού χαρακτήρα οι οποίες πρέπει να προγραμματιστούν σε Java, όμως στην περίπτωσή μας είναι χαρακτηριστικά ενσωματωμένα μέσα στην ίδια την Processing, με αποτέλεσμα να κάνουν τα προγράμματα πιο μικρά και πιο εύκολα στην ανάγνωση και την συντήρηση.

Σύγκριση

ActionScript και Lingo

Η ActionScript (αυτή την στιγμή στην έκδοση 3.0) [19] είναι μια scripting γλώσσα βασισμένη στην ECMAScript [28]. Χρησιμοποιείται κυρίως στην ανάπτυξη ιστοσελίδων και λογισμικού με χρήση της πλατφόρμας Flash της Adobe (υπό μορφή αρχείων SWF που ενσωματώνονται σε ιστοσελίδες). Επίσης, χρησιμοποιείται σε μερικές εφαρμογές βάσεων δεδομένων καθώς και σε απλή ρομποτική. Αρχικά αναπτυγμένη από την Macromedia, την γλώσσα τώρα κατέχει η Adobe (που απόκτησε την Macromedia το 2005). Η ActionScript σχεδιάστηκε αρχικά για τον έλεγχο των απλών διανυσματικών γραφικών δύο διαστάσεων που γίνονταν σε Adobe Flash (στο παρελθόν Macromedia Flash). Οι πιο πρόσφατες εκδόσεις πρόσθεσαν λειτουργίες που επιτρέπουν τη δημιουργία παιχνιδιών βασισμένων στο διαδίκτυο καθώς και εφαρμογών διαδικτύου με ροές δεδομένων (streaming data) όπως το βίντεο και ο ήχος.

Καθώς η ActionScript βασίζεται στην ECMAScript/JavaScript -μάάλιστα, μετά την έκδοση 2.0 είναι αντικειμενοστραφής με εκτεταμένες δυνατότητες- και η Processing σε Java, οι ομοιότητες μεταξύ των γλωσσών είναι πολλές. Οι μεγαλύτερες τους διαφορές συνοψίζονται στα εξής:

Απεικόνιση: Ο κώδικας της ActionScript έχει βελτιστοποιηθεί κατάλληλα και τα σχήματα σχεδιάζονται στην οθόνη γρηγορότερα από ότι στην Processing (εκτός αν χρησιμοποιείται η βιβλιοθήκη της OpenGL). Αυτό επι-

επιτρέπει στις εφαρμογές που είναι γραμμένες σε Flash να είναι μεγαλύτερες όσον αφορά τις διαστάσεις των εικονοστοιχείων και να κάνουν εκτεταμένη χρήση διαφανών (transparent) και εξομαλυμένων (antialiased) γραφικών.

3D: Εσωτερικά η ActionScript δεν αναπαριστά τη χωρική δομή σε τρεις διαστάσεις, ενώ η βιβλιοθήκη απεικόνισης της Processing είναι εγγενώς τρισδιάστατη.

Αντικείμενα: Η ActionScript είναι εγγενώς αντικειμενοστραφής, ενώ η Processing μπορεί να γραφτεί σε διαδικαστικό και αντικειμενοστραφές στυλ, με αποτέλεσμα οι άνθρωποι που μαθαίνουν Processing να μπορούν να αντιληφθούν και να γράψουν σε Processing χωρίς να αντιλαμβάνονται πλήρως τις έννοιες του αντικειμενοστραφούς προγραμματισμού.

Η Lingo [20] είναι μια γλώσσα γραμμένη για το πακέτο λογισμικού Director της Macromedia (τώρα Adobe). Το Director ήταν το κυρίαρχο περιβάλλον σχεδιαστών και καλλιτεχνών για την δημιουργία έργων για cd-rom (π.χ. εισαγωγή, μενού κλπ) αλλά έχασε σε δημοτικότητα στην εποχή του διαδικτύου λόγω της επιτυχίας του Flash. Εξακολουθεί να είναι ένα από τα πιο συχνά χρησιμοποιούμενα περιβάλλοντα και έχει εξαιρετικές βιβλιοθήκες για την επέκταση των λειτουργιών του. Η Lingo είναι ενσωματωμένη σε ένα γραφικό περιβάλλον -μεταφορά ενός θεάτρου- χρησιμοποιώντας όρους όπως "stage" (πάλκο) και "cast" (διανομή ρόλων) και

Σύγκριση

Python και Design By Numbers

χαρακτηρίζεται από την εκτεταμένη αγγλίζουσα σύνταξη. Πρέπει να σημειωθεί πως τα τελευταία χρόνια έχει αναβαθμιστεί ώστε να υποστηρίζει αντικειμενοστραφείς δομές και τρισδιάστατα γραφικά.

Η Python [21,22] είναι μια γενικής χρήσης υψηλού επιπέδου γλώσσα προγραμματισμού. Ολόκληρη η φιλοσοφία σχεδιασμού της γλώσσας αυτής δίνει έμφαση στην αναγνωσιμότητα κώδικα. Η Python προβάλλει ως κύριο χαρακτηριστικό της την ισχύ της σε συνδυασμό με την καθαρή σύνταξή της, με την ενσωματωμένη βιβλιοθήκη της να είναι μεγάλη και περιεκτική. Ένα ασυνήθιστο, μεταξύ των πιο δημοφιλών γλωσσών προγραμματισμού, χαρακτηριστικό της είναι η διάκριση μεταξύ των μπλοκ κώδικα με την χρήση των εσοχών. Η Python υποστηρίζει πολλαπλά είδη προγραμματισμού (πρωτίστως αντικειμενοστρέφεια, προστακτικό προγραμματισμό και συναρτησιακό) και χαρακτηρίζεται από ένα πλήρως δυναμικό σύστημα τύπων και αυτόματη διαχείριση μνήμης, παρόμοια με αυτή της Perl, της Ruby, της Scheme και της Tcl. Όπως και άλλες δυναμικές γλώσσες, η Python χρησιμοποιείται συχνά ως γλώσσα σεναρίων (scripting language). Η γλώσσα αυτή έχει ένα ανοικτό μοντέλο ανάπτυξης κώδικα που υποστηρίζεται από μία ενεργή κοινότητα η οποία διευθύνεται από το μη κερδοσκοπικό ίδρυμα λογισμικού Python [23] (Python Software Foundation), το οποίο διατηρεί και το de facto τυποποιημένο καθορισμό της γλώσσας σε CPython, η οποία επί της ουσίας αποτελεί την υλοποίηση των τεχνικών προδιαγραφών. Η Python θεωρείται εξαιρετική γλώσσα για διδασκαλία, και αυτό γιατί έχει καθαρή σύντα-

ξη και δομή. Τυπικά, χρησιμοποιείται σε εφαρμογές που δεν έχουν να κάνουν με γραφικά. Δεν ενσωματώνει κάποια βιβλιοθήκη για γραφικά όπως η Processing, όμως η δημιουργία εφαρμογών γραφικών είναι δυνατή με την χρήση γραφικών πακέτων (toolkits), μερικά από τα οποία είναι ανεξάρτητα πλατφόρμας (platform independent).

Η Design By Numbers (DBN) [17] αναπτύχθηκε για την διδασκαλία γενικών εννοιών προγραμματισμού σε καλλιτέχνες και σχεδιαστές με καμία προηγούμενη εμπειρία στον προγραμματισμό. Η DBN είναι μία εξαιρετικά μινιμαλιστική γλώσσα και περιβάλλον, με αποτέλεσμα να είναι εύκολη στην εκμάθηση αλλά με περιορισμούς στις δυνατότητες της για την δημιουργία προχωρημένων εφαρμογών. Πρέπει εδώ να σημειωθεί πως η DBN δημιουργήθηκε από τον John Maeda, διευθυντή της Ομάδας Αισθητικής και Υπολογιστών στο Εργαστήριο Πολυμέσων του Τεχνολογικού Ινστιτούτου της Μασαχουσέτης (Aesthetics and Computation Group, MIT Media Lab). Η συγγένεια της Processing βασίζεται σε ακριβώς αυτό το γεγονός, ότι γεννήθηκε και αυτή στο ίδιο τμήμα, μετά από πολλά χρόνια ανάπτυξης και διδασκαλίας με DBN, κάτι το οποίο επηρέασε την πρόοδο της Processing. Τέλος, αξιοσημείωτο είναι το γεγονός ότι η DBN είναι δωρεάν και έχει την δυνατότητα να τρέξει μέσα από φυλλομετρητή ιστού (web browser).

Βιβλιοθήκες

Video - Βιβλιοθήκη Εγγραφής και Προβολής Ταινιών

Όποτε καλούμε μια συνάρτηση της Processing όπως για παράδειγμα η `line()`, `background()`, `stroke()` κλπ., αυτή αντιστοιχεί σε κάποια διαθέσιμη υλοποίηση στον πυρήνα της Processing. Πρακτικά, στην επιστήμη των υπολογιστών, ο όρος βιβλιοθήκη αναφέρεται σε μία συλλογή από "βοηθητικό" κώδικα. Μία βιβλιοθήκη μπορεί να αποτελείται από συναρτήσεις, μεταβλητές και αντικείμενα. Αν και η βιβλιοθήκη του πυρήνα της Processing καλύπτει όλες τις βασικές λειτουργίες - και πολύ περισσότερες από αυτές -, για πιο προχωρημένες λειτουργίες όπως λ.χ. για επεξεργασία γραφικών, βίντεο, δικτυακών λειτουργιών κλπ., πρέπει να χρησιμοποιηθούν "εξωτερικές" (external) βιβλιοθήκες. Υπάρχουν εξωτερικές βιβλιοθήκες ενσωματωμένες στον πυρήνα (built-in) καθώς και βιβλιοθήκες τρίτων (third party, contributed). Οι ενσωματωμένες βιβλιοθήκες συνοψίζονται στις εξής:

- **Video** — Για την σύλληψη εικόνων από κάμερα, αναπαραγωγή και εγγραφή αρχείων κινούμενης εικόνας κλπ.
- **Serial** — Για την αποστολή δεδομένων μεταξύ της Processing και εξωτερικής συσκευής με την χρήση σειριακής σύνδεσης.
- **OpenGL** — Για την απόδοση με την χρήση επιτάχυν

σης κάρτας γραφικών.

- **Network** — Για την δημιουργία σχεδίων πελάτη-εξυπηρετητή τα οποία επικοινωνούν μέσω διαδικτύου.
- **PDF** — Για την δημιουργία υψηλής ανάλυσης PDF από εικόνες και γραφικά τα οποία δημιουργήθηκαν από την Processing.
- **XML** — Για την εισαγωγή δεδομένων από αρχεία XML.

Το σύνολο των βιβλιοθηκών τρίτων είναι ένα τοπίο το οποίο ποικίλει και συνεχώς αλλάζει με πάνω από 50 βιβλιοθήκες των οποίων οι δυνατότητες εκτείνονται από επεξεργασία ήχου σε εξομίωση φυσικής και από οδήγηση υλικού σε υπολογιστική όραση. Για να εγκαταστήσει κάποιος μία βιβλιοθήκη τρίτου, πρέπει να εγκαταστήσει τα αντίστοιχα αρχεία στον φάκελο της Processing, στον υποφάκελο `Libraries`.

Η βιβλιοθήκη βίντεο επιτρέπει στην Processing να προβάλλει αρχεία QuickTime, να συλλαμβάνει στοιχεία βίντεο από φωτογραφική μηχανή ή κάμερα, και να δημιουργεί άμεσα αρχεία βίντεο σε μορφή QuickTime από το εκάστοτε χρησιμοποιούμενο πρό-

Βιβλιοθήκες

Network - Βιβλιοθήκη Δικτύου

Serial - Βιβλιοθήκη Σειριακής Προσπέλασης

γραμμα. Το βίντεο μπορεί να συλληφθεί από φωτογραφικές μηχανές ή κάμερες οι οποίες έχουν συνδεθεί στον υπολογιστή με USB ή IEEE 1394 (Firewire), ή από κάρτες γραφικών με συνδέσεις Composite ή S-Video που συνδέονται με κάποια πηγή. Το βίντεο μπορεί να φορτωθεί από τα αρχεία QuickTime που βρίσκονται στον υπολογιστή ή οπουδήποτε στο διαδίκτυο. Για να λειτουργήσει αυτή η βιβλιοθήκη είναι απαραίτητο να εγκατασταθεί το QuickTime (ή τα iTunes, τα οποία περιλαμβάνουν το QuickTime). Άλλες λύσεις όπως λ.χ. εναλλακτικές λύσεις ανοιχτού κώδικα για το QuickTime ή το QT Lite δεν θα λειτουργήσουν και πρέπει να αποφευχθούν. Επειδή από την Apple το QuickTime βγαίνει μόνο για τα λειτουργικά συστήματα Mac OS και Windows, άλλες πλατφόρμες (όπως Linux) δεν μπορούν δυστυχώς να χρησιμοποιήσουν την βιβλιοθήκη αυτή. Επίσης, είναι απαραίτητοι οι οδηγοί της συσκευής που αποτελεί πηγή (device drivers) για τη μετατροπή από τον τύπο της τηλεοπτικής πηγής - ενδεχομένως κλειστού κώδικα - στις συνήθως χρησιμοποιούμενες μορφές βίντεο. Τέλος, αν γίνεται χρήση του λειτουργικού συστήματος Windows, τότε είναι απαραίτητη επίσης η χρήση οδηγών για την μετατροπή από την μορφή που έρχεται το βίντεο στον υπολογιστή στην μορφή που χρησιμοποιεί το QuickTime. Το λογισμικό των δύο τελευταίων περιπτώσεων μπορεί είτε να δοθεί από τον κατασκευαστή της εκάστοτε συσκευής της οποίας η έξοδος δίνεται ως είσοδος

στον υπολογιστή, είτε μπορεί να μεταφορτωθεί από το διαδίκτυο.

Η βιβλιοθήκη δικτύου καθιστά εύκολη την ανάγνωση και εγγραφή δεδομένων σε μηχανήματα που βρίσκονται στο διαδίκτυο. Επιτρέπει την δημιουργία πελατών (clients) και εξυπηρετητών (servers). Ένας κεντρικός υπολογιστής συνδέεται με έναν κατάλογο πελατών για την ανάγνωση και εγγραφή δεδομένων. Ένας πελάτης είναι σε θέση να διαβάσει και να γράψει τα στοιχεία σε έναν κεντρικό υπολογιστή.

Η βιβλιοθήκη σειριακής προσπέλασης της Processing επιτρέπει την εύκολη ανάγνωση και εγγραφή δεδομένων από και σε εξωτερικές συσκευές. Επιτρέπει σε δύο υπολογιστές να στείλουν και να λάβουν δεδομένα και προσφέρει αξιοσημείωτη ευελιξία στην επικοινωνία με συσκευές μικροελεγκτών ειδικής κατασκευής, οι οποίοι αξιοποιούνται ως είσοδοι ή έξοδοι των προγραμμάτων Processing. Η σειριακή θύρα είναι μία θύρα εισόδου/εξόδου εννέα ακίδων που υπάρχει σε πληθώρα υπολογιστών και μπορεί να προσομοιωθεί μέσω USB με την προσαρμοστική θύρα Keyspan USB και άλλες συμβατές συσκευές.

Βιβλιοθήκες

OpenGL - Βιβλιοθήκη Απόδοσης Γραφικών

....

Το OpenGL (Open Graphics Library - Ανοικτή Βιβλιοθήκη Γραφικών) [32,33] είναι μια διεπαφή γραφικών ανεξαρτήτου πλατφόρμας για δισδιάστατη και τρισδιάστατη αναπαράσταση. Η βιβλιοθήκη αυτή επιτρέπει στα προγράμματα να αξιοποιήσουν την ταχύτητα ενός επιταχυντή OpenGL στην κάρτα γραφικών. Κάτι τέτοιο επεκτείνει τις δυνατότητες για το σχεδιασμό περισσότερων γραφικών στην οθόνη, τη δημιουργία μεγαλύτερων σχημάτων και πιο "βαριών" αναπαραστάσεων [31]. Η Processing διασυνδέεται με την OpenGL μέσω JOGL, μια πρωτοβουλία της Τεχνολογικής Ομάδας Παιχνιδιών (Game Technology Group) της Sun. Για την πλήρη αξιοποίηση της βιβλιοθήκης αυτής, ο χρήστης πρέπει να εγκαταστήσει μια κάρτα γραφικών με επιτάχυνση OpenGL στον υπολογιστή του. Περισσότερες πληροφορίες περιγράφονται παρακάτω, στους τρόπους απόδοσης (σελ. 25). Εδώ απλά να σημειωθεί πως δεν προσφέρει κάποιες περαιτέρω κλάσεις ή συναρτήσεις στην ίδια την Processing. Απλά δίνει την δυνατότητα για διαφορετική απόδοση της γεωμετρίας. Επίσης, η Processing δεν επιχειρεί να είναι ένα περιβάλλον ανάπτυξης ειδικά για τη χρήση της προγραμματιστικής διεπαφής του JOGL. Τα προγράμματα πρέπει να εισάγουν τη βιβλιοθήκη περιέχοντας την γραμμή `processing.opengl.*` και η επιλογή απόδοσης OpenGL πρέπει να διευκρινιστεί ως τρίτη παράμετρος στη συνάρτηση `size()`. Τέλος, τονίζεται πως το λογισμικό που χρησιμοποιεί αυτήν την βιβλιοθήκη μπορεί να

εμφανιστεί διαφορετικά στις ποικίλες κάρτες γραφικών -- μερικές κάρτες υποστηρίζουν το OpenGL καλύτερα από άλλες.

Το Minim είναι μια βιβλιοθήκη ήχου που χρησιμοποιεί το JavaSound API, ένα κομμάτι από το Tritonus, και το MP3SPI Javazoom για να παρέχει μια εύχρηστη ακουστική βιβλιοθήκη για τους ανθρώπους που αναπτύσσουν στο περιβάλλον της Processing. Η φιλοσοφία πίσω από την προγραμματιστική διεπαφή (API) είναι να καταστήσει δυνατή την ενσωμάτωση ήχου με όσο το δυνατόν απλούστερο τρόπο παρέχοντας παράλληλα αρκετή ευελιξία για τους πιο προηγμένους χρήστες. Προσφέρει ένα επίπεδο αφαιρετικότητας, επιτρέποντας στον χρήστη να χειριστεί εύκολα και γρήγορα χρονοσειρές ήχου. Μερικά από τα χαρακτηριστικά που προσφέρει είναι:

- **AudioPlayer:** Μονοφωνική και στερεοφωνική αναπαραγωγή αρχείων WAV, AIFF, AU, SND και MP3.
- **AudioRecorder:** Μονοφωνική και στερεοφωνική εγγραφή είτε στην μνήμη είτε στον σκληρό δίσκο.
- **AudioOutput:** Μονοφωνική και στερεοφωνική σύνθεση ήχου.

Βιβλιοθήκες

PDF - Βιβλιοθήκη Απόδοσης Διανυσματικών Εικόνων.

- AudioEffect: Μία απλή διεπαφή που δίνει την δυνατότητα στον χρήστη να γράψει τα δικά του εφέ.
- Προσφέρει μία κλάση ταχέως μετασχηματισμού Fourier για την ανάλυση συχνотήτων.
- Προσφέρει μία κλάση BeatDetect η οποία αναγνωρίζει το τύμπανο χαμηλών συχνотήτων – και κατ' επέκταση τον ρυθμό.

Reader, ή αντίστοιχα προγράμματα και επεκτάσεις (plugins) ανοιχτού ή κλειστού κώδικα.

Η βιβλιοθήκη απόδοσης διανυσματικών γραφικών καθιστά δυνατή την εγγραφή αρχείων PDF άμεσα από την Processing. Τα αρχεία αυτά διανυσματικών γραφικών μπορούν να κλιμακωθούν σε οποιοδήποτε μέγεθος και να αναπαραχθούν σε πολύ υψηλές αναλύσεις. Εδώ να σημειωθεί πως υπάρχει επίσης μια βιβλιοθήκη για την εξαγωγή εικόνων σε μορφή SVG (Scalable Vector Graphics -Κλιμακούμενα Διανυσματικά Γραφικά) – περισσότερες πληροφορίες μπορεί κάποιος να αναζητήσει στο [24]. Η βιβλιοθήκη PDF μπορεί να προβάλει τα τρισδιάστατα στοιχεία σε ένα διανυσματικό αρχείο δύο διαστάσεων, αλλά για την εξαγωγή στις τρεις διαστάσεις, απαιτείται η βιβλιοθήκη DXF. Η βιβλιοθήκη PDF χρησιμοποιείται συχνά με την συνάρτηση `size()` της Processing, σε συνδυασμό με την `beginRecord()` και `endRecord()` ή με την `beginRaw()` και `endRaw()`. Απαραίτητο για την ανάγνωση του συγκεκριμένου τύπου δεδομένων είναι το πρόγραμμα Adobe

Στην επόμενη σελίδα, η εικόνα έχει αποδοθεί ηλεκτρονικά με την χρήση του POV-Ray 3.6 επιδεικνύοντας εκπληκτικό φωτορεαλισμό. Αποτελεί ένα δείγμα του που βρίσκεται αυτή την στιγμή η απόδοση γραφικών στους ηλεκτρονικούς υπολογιστές.

Απόδοση

Απόδοση (Rendering) είναι η διαδικασία κατά την οποία μια εικόνα παράγεται από ένα πρότυπο, με τη βοήθεια υπολογιστικών εργαλείων. Το πρότυπο είναι μια περιγραφή των τρισδιάστατων αντικειμένων σε μια αυστηρά καθορισμένη γλώσσα ή δομή δεδομένων. Η περιγραφή περιέχει πληροφορίες γεωμετρίας, άποψης στον χώρο (προοπτική), σύστασης, φωτισμού, και σκίασης. Η εικόνα είναι μια ψηφιακή εικόνα ή εικόνα ράστερ γραφικών (raster graphics image). Για να γίνει πιο κατανοητός ο όρος απόδοση, κάποιος μπορεί να τον αντιληφθεί κατ' αναλογία με την απόδοση μιας σκηνής από έναν καλλιτέχνη. Επίσης, ο όρος rendering χρησιμοποιείται για να περιγράψει τη διαδικασία υπολογισμού των επιδράσεων των εφέ, σε ένα αρχείο στατικής ή κινούμενης εικόνας, ώστε να παραχθεί το τελικό αποτέλεσμα.

Η απόδοση είναι ένα από τα σημαντικότερα θέματα των τρισδιάστατων γραφικών στους υπολογιστές, και στην πράξη είναι πάντα συνδεδεμένο με τα υπόλοιπα. Σε όλα τα στάδια που διεκπεραιώνουν την διαδικασία γραφικής αναπαράστασης, είναι το τελευταίο σημαντικό βήμα που δίνει την τελική εμφάνιση στα μοντέλα και την κίνηση. Με την ολοένα και αυξανόμενη πρόοδο των γραφικών από τη δεκαετία του 1970 και έπειτα, έχει γίνει ένα αυτόνομο και αυτοτελές θέμα. Η απόδοση απαντάται στην αρχιτεκτονική, τα παιχνίδια υπολογιστών, στους εξομοιωτές, τα ειδικά εφέ στον κινηματογράφο ή την τηλεόραση και γενικότερα σε ότι έχει να κάνει με την απεικόνιση σχεδίου, με την κάθε περίπτωση να υιοθετεί μια διαφορετική ισορροπία μεταξύ χαρακτηριστικών, γνωρισμάτων και τεχνικών. Στην αγορά υπάρχει μια πληθώρα από μηχανές που κάνουν απόδοση γραφικών (renderers). Μερικές είναι

ενσωματωμένες σε μεγαλύτερα πακέτα μοντελοποίησης ή γραφικών, μερικές είναι αυτόνομες και μερικές είναι ανοικτού κώδικα.

Στο εσωτερικό, μία μηχανή που κάνει απόδοση είναι ένα πρόγραμμα πολύ προσεκτικά κατασκευασμένο, βασισμένο σε συγκεκριμένες αρχές που σχετίζονται με την φυσική του φωτός, την οπτική αντίληψη, τα μαθηματικά, και την ανάπτυξη λογισμικού. Στην περίπτωση της τρισδιάστατης γραφικής αναπαράστασης, η απόδοση μπορεί να γίνει αργά (όπως λ.χ. στην προ-απόδοση), ή σε πραγματικό χρόνο. Η προ-απόδοση (pre-rendering) είναι μια υπολογιστικά έντονη διαδικασία που χρησιμοποιείται εκτενώς στην δημιουργία ταινιών, ενώ η απόδοση σε πραγματικό χρόνο γίνεται συχνά για τα παιχνίδια υπολογιστών που στηρίζονται στη χρήση των καρτών γραφικών με τρισδιάστατους επιταχυντές υλικού (3D hardware accelerators).



Απόδοση

Java2D και P3D

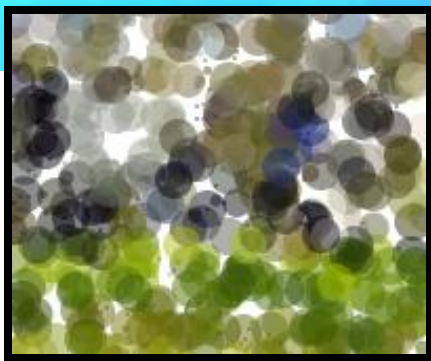
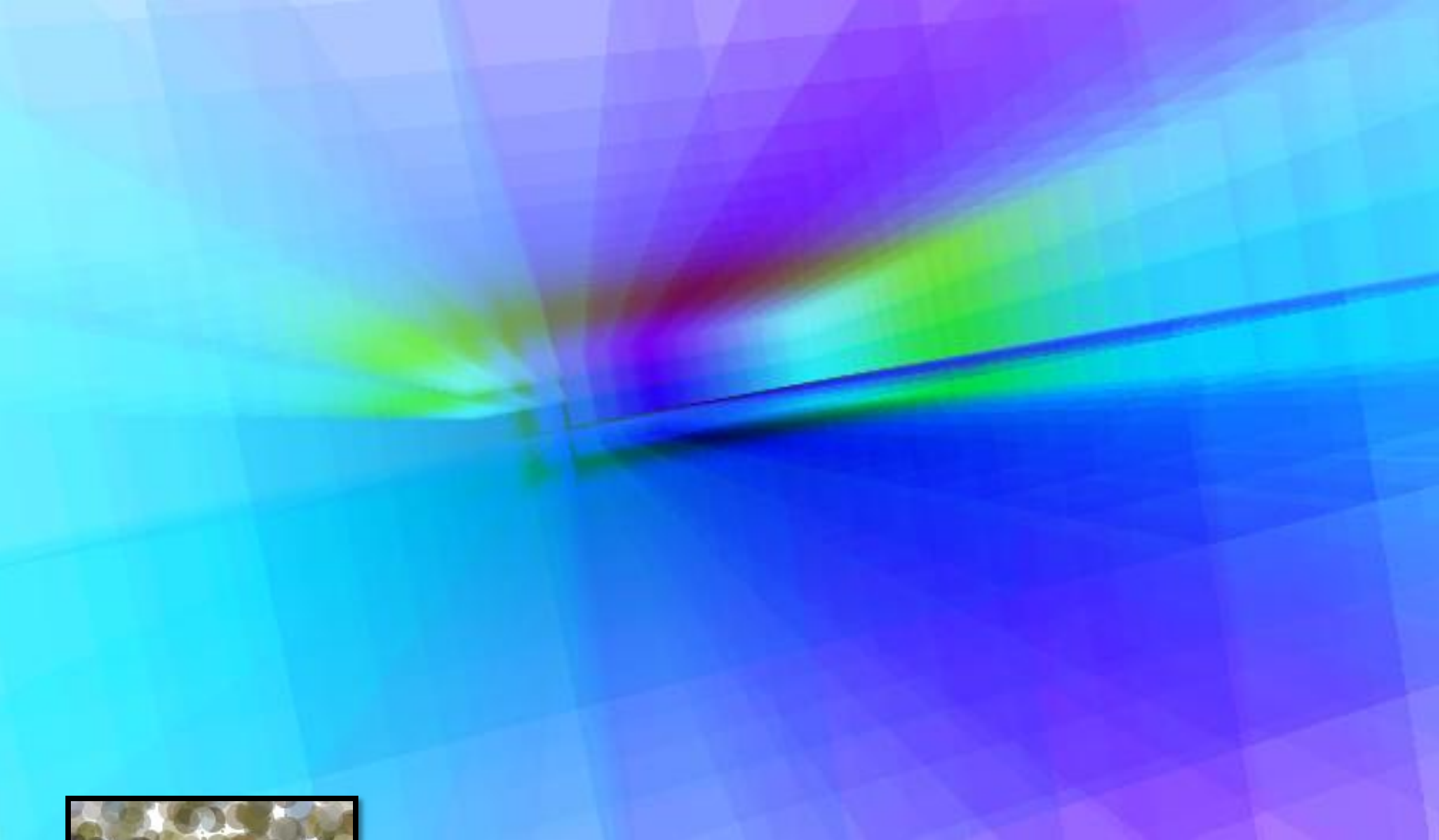
Η Processing έχει τρεις τρόπους να αποδώσει τα γραφικά: JAVA2D, P3D και OpenGL. Η επιλογή μεταξύ των τρόπων γίνεται στην κλήση της `size` (π.χ. `size(800,600,OPENGL)`) και ελέγχει τον τρόπο με τον οποίο γίνεται η απόδοση στα οπτικά δεδομένα ή η μετατροπή σε εικονοστοιχεία (Picture Elements - pixels) στην οθόνη.

Η JAVA2D χρησιμοποιεί την δυοδιάστατη βιβλιοθήκη γραφικών της ίδιας της Java για να κάνει απόδοση στις δύο διαστάσεις. Σύμφωνα με την Sun η βιβλιοθήκη είναι ένα σύνολο κλάσεων για προχωρημένα γραφικά δύο διαστάσεων και δημιουργία εικόνων. Η βιβλιοθήκη περιέχει πάνω από 100 κλάσεις οι οποίες είναι οργανωμένες σε πακέτα σύμφωνα με την λειτουργικότητά τους (ανάλογα με το αν σχετίζονται με στατικές εικόνες, σχεδιασμό, ή χρώματα). Η JAVA2D είναι ο προεπιλεγμένος τρόπος για απόδοση στην Processing. Για τον λόγο αυτό, όταν η κλήση της `size()` γίνεται χωρίς τρίτο όρισμα, χρησιμοποιείται αυτόματα η επιλογή JAVA2D.

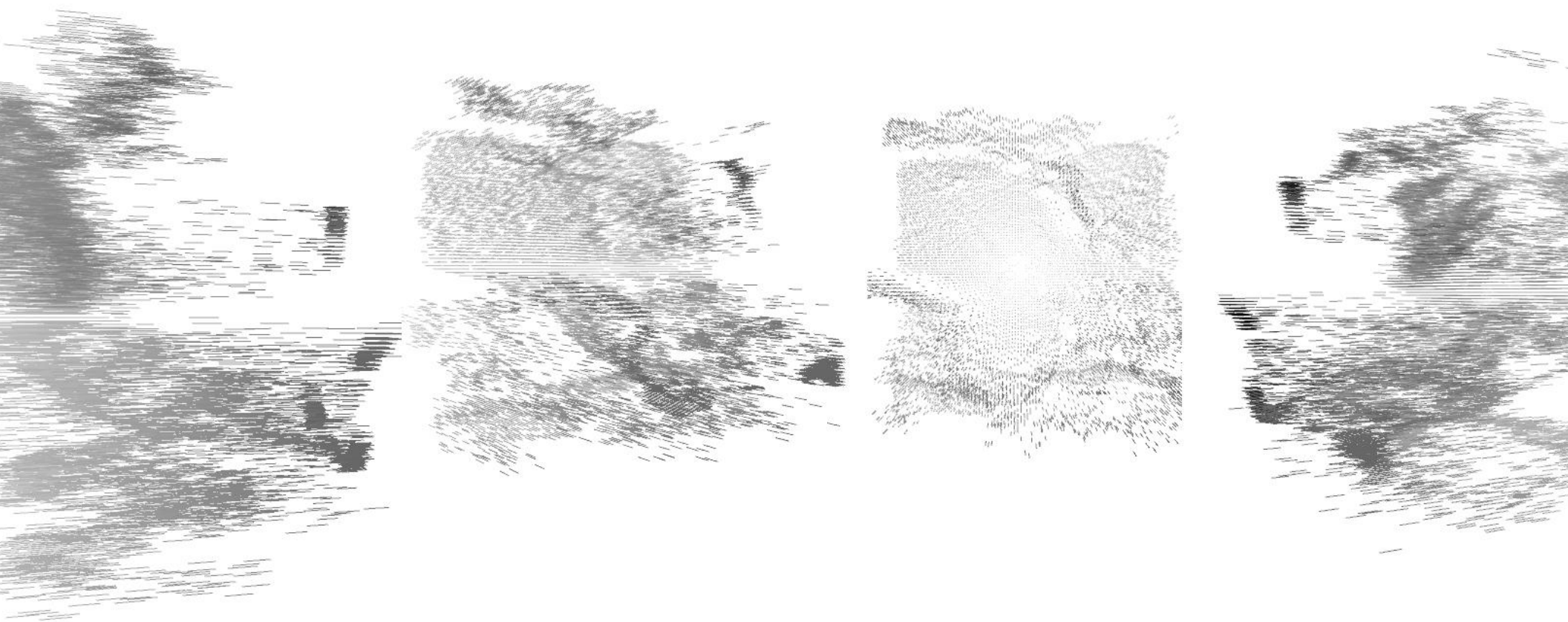
Η επιλογή P3D δίνει πρόσβαση σε μια προσαρμοσμένη μηχανή τρισδιάστατης απεικόνισης στην Processing. Μία μηχανή 3D (3D engine) είναι στην ουσία λογισμικό το οποίο μας επιτρέπει να "χειριστούμε" τον άξονα z με την ίδια ευκολία που χειριζόμαστε τους κλασσικούς άξονες x και y. Στο 3D, η γεωμετρία υπολογίζεται στις τρεις διαστάσεις, κάτι το οποίο, για τον υπολογιστή, είναι απλώς ένα μαθηματικό πρόβλημα. Παρ' όλ' αυτά, η οθόνη του υπολογιστή ή της τηλεόρασης ή ακόμα και η επιφάνεια πάνω στην οποία αντικατοπτρίζεται η έξοδος ενός προτζέκτορα, είναι μόνο στις δύο διαστάσεις, γι' αυτό και η μηχανή πρέ-

πει να μεταφράσει τρόπον τινά δεδομένα τριών διαστάσεων πάνω σε μία επιφάνεια δύο διαστάσεων. Για παράδειγμα, στον πραγματικό κόσμο, καθώς απομακρυνόμαστε από ένα αντικείμενο, οι αισθήσεις μας αντιλαμβάνονται το αντικείμενο αυτό να μικραίνει σε μέγεθος. Μπορούμε να απεικονίσουμε αυτή την αίσθηση -δημιουργώντας μία ψευδαίσθηση -δίνοντας βάθος- όπως ακριβώς κάνει ένας ζωγράφος σε έναν παραδοσιακό πίνακα. Το σύστημα βάθους λειτουργεί παρόμοια και στο P3D -χωρίς όμως να πρέπει να χρησιμοποιήσουμε χάρακα! Οι προβολές από τις τρεις διαστάσεις στις δύο σε έναν υπολογιστή ακολουθούν τον τρόπο παρουσίασης με τον οποίο θα λειτουργούσε μία εικονική κάμερα, η οποία έχει την δυνατότητα για λήψεις πολύ ευρείας γωνίας (στις οποίες τα αντικείμενα παρουσιάζονται μεγάλα, πολύ κοντά και σε πολλές περιπτώσεις αλλοιωμένα -για παράδειγμα με την χρήση φακών «όρασης-ψαριού») και πολύ μεγάλης απόστασης. Πέραν των δυνατοτήτων εικονικής κάμερας, το P3D προσφέρει εικονικό φωτισμό και απλή αντιστοίχιση στοιχείων (image mapping), επιτρέποντας σε μία εικόνα να προσκολληθεί στα σημεία (επιφάνειες) μίας τρισδιάστατης γεωμετρίας. Το P3D έχει χτιστεί με στόχο την ταχύτητα, στο κόστος της ελάχιστης μειωμένης ποιότητας απόδοσης. Παρ' όλ' αυτά, είναι αρκετά απλό, για αρχικούς πειραματισμούς στις τρεις διαστάσεις, με λίγο κώδικα και μικρή πολυπλοκότητα.

Το P3D αξιοποιεί μία μηχανή 3D σε επίπεδο λογισμικού, με την έννοια ότι όλοι οι υπολογισμοί τριών διαστάσεων γίνονται από την Java ακριβώς όπως οι κλασσικοί υπολογισμοί δύο διαστάσεων. Τα μαθημα-



- 1.(κάτω αριστερά)Απόδοση με το P2D. Η συγκεκριμένη εφαρμογή δημιουργεί ένα εφφέ πουαντλισμού, χρησιμοποιώντας ελλείψεις οι οποίες έχουν χρωματιστεί σύμφωνα με τα αντίστοιχα εικονοστοιχεία μίας εικόνας. Με την χρήση του ποντικιού, αλλάζει το μέγεθός τους.
2. Απόδοση με το P3D. Το σύνολο των ημιδιαφανών κύβων περιστρέφεται στον χώρο. Γίνεται χρήση των συναρτήσεων `pushMatrix()` and `popMatrix()`.
- 3.Απόδοση με το OpenGL: Μετατρέπει μία επίπεδη εικόνα σε χωρικά σημεία δεδομένων τα οποία περιστρέφονται γύρω από το κέντρο. Καθώς η μάζα περιστρέφεται γύρω από έναν κεντρικό άξονα και έχουν τραβηχτεί τέσσερα στιγμιότυπα. Η κίνηση είναι γρήγορη και μπορεί χωρίς δυσκολία να αποδοθεί σε ιδιαίτερα υψηλές αναλύσεις χωρίς να υπάρχει καθυστέρηση με μία σχετικά μέτρια κάρτα γραφικών η οποία υποστηρίζει OpenGL.



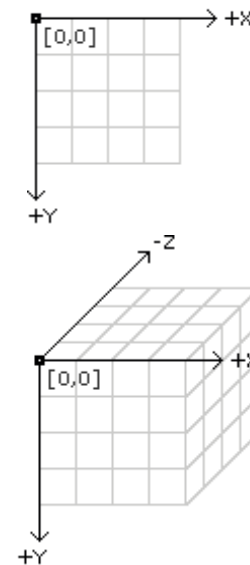
Απόδοση

OpenGL

τικά τριών διαστάσεων κάμπτονται και μετατρέπονται στις δύο διαστάσεις πριν να αποσταλούν στο υλικό που είναι υπεύθυνο για τα γραφικά και το σχεδιασμό της εξόδου στο επίπεδο. Το υλικό δεν γνωρίζει καν πως τα δεδομένα ξεκίνησαν την ζωή τους στις τρεις διαστάσεις. Παρ' όλ' αυτά, το υλικό έχει την δυνατότητα να κάμψει τους αριθμούς, και μάλιστα μπορεί να το κάνει εξαιρετικά πιο γρήγορα από την ίδια την Java. Το κόλπο βέβαια, έγκειται στην άμεση επικοινωνία με το υλικό που είναι υπεύθυνο για τα γραφικά ώστε να επιταχυνθεί η διαδικασία.

Το OpenGL είναι μία ανεξαρτήτου πλατφόρμας (platform-independent) βιβλιοθήκη η οποία λειτουργεί ως μία διεπαφή μεταξύ του κώδικα και του υλικού που είναι υπεύθυνο για τα γραφικά. Αναπτύχθηκε το 1992 από την Silicon Graphics, αλλά τώρα επιβλέπεται από μία αρκετά μεγάλη ομάδα από οργανισμούς, υπό την αιγίδα του OpenGL Architecture Review Board. Σχεδιασμένη ως ένα σύνολο από προδιαγραφές η OpenGL είναι ανεξάρτητη του υλικού και μπορεί να αξιοποιηθεί, στην ουσία, από όλα τα συστήματα τα οποία έχουν OpenGL επιτάχυνση στο υλικό. Η Processing και η Java συνδέονται με την OpenGL μέσω του JOGL. Η JOGL, η οποία αναπτύχθηκε από την Ομάδα Τεχνολογίας Παιχνιδιών (Game Technology Group) στην Sun, τεχνικά μπορεί να θεωρηθεί ως μία Java διασύνδεση για την OpenGL, και δίνει την δυνατότητα στα σχέδια της Processing τα οποία χρησιμοποιούν την επιλογή της OpenGL για απόδοση την δυνατότητα να αναπαριστούν περισσότερα αντικείμενα, πιο γρή-

γορα και σε μεγαλύτερα μεγέθη. Για να υπάρχουν όμως πραγματικά οφέλη από την επιλογή της OpenGL είναι απαραίτητη μία κάρτα γραφικών με επιτάχυνση OpenGL.



Περιβάλλον Ανάπτυξης

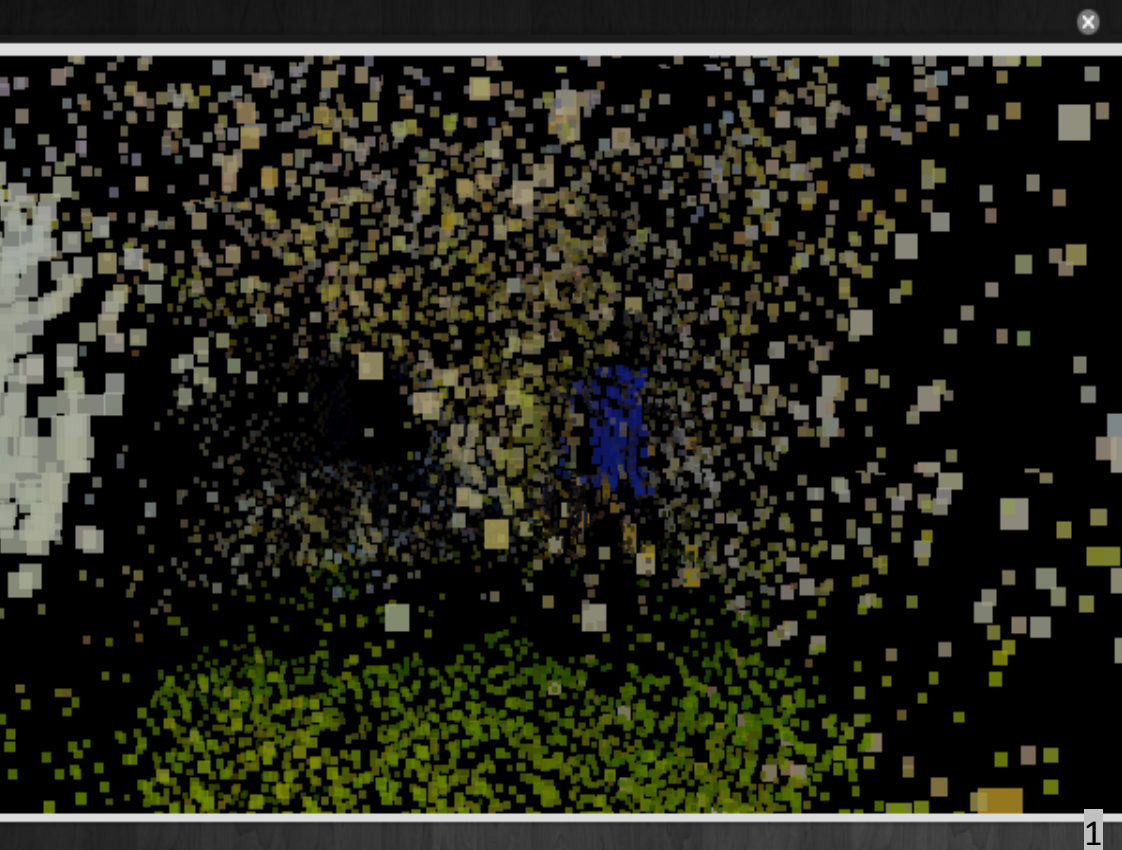
Το περιβάλλον ανάπτυξης της Processing (PDE) αποτελείται από έναν απλό κειμενογράφο για το γράψιμο του κώδικα, μια περιοχή μηνυμάτων, μια κονσόλα κειμένου, τις ετικέτες (tabs) για τη διαχείριση αρχείων, μια μπάρα εργαλείων με τα κουμπιά για τις πιο κοινές ενέργειες και μια σειρά επιλογών. Όταν τα προγράμματα τρέχουν, ανοίγουν σε ένα νέο παράθυρο το οποίο αποκαλείται παράθυρο επίδειξης.

Το λογισμικό που δημιουργείται χρησιμοποιώντας την Processing καλείται σκίτσο (sketch). Αυτά τα σκίτσα γράφονται στον κειμενογράφο. Έχει δυνατότητες για αποκοπή/επικόλληση και για εύρεση/αντικατάσταση κειμένου. Η περιοχή μηνυμάτων αποκρίνεται όταν αποθηκεύεται ή εξάγεται ένα αρχείο καθώς και εμφανίζει τα σφάλματα. Η κονσόλα απεικονίζει κείμενο το οποίο αποτελεί έξοδο από τα προγράμματα της Processing περιλαμβάνοντας πλήρη μηνύματα λάθους καθώς και κείμενο το οποίο θα πρέπει να εμφανιστεί με την χρήση των συναρτήσεων `print()` και `println()`. Τα κουμπιά της μπάρας εργαλείων επιτρέπουν στον χρήστη να ξεκινά και να σταματά το τρέξιμο προγραμμάτων, να δημιουργεί, να ανοίγει, να σώζει και να εξάγει σχέδια της Processing

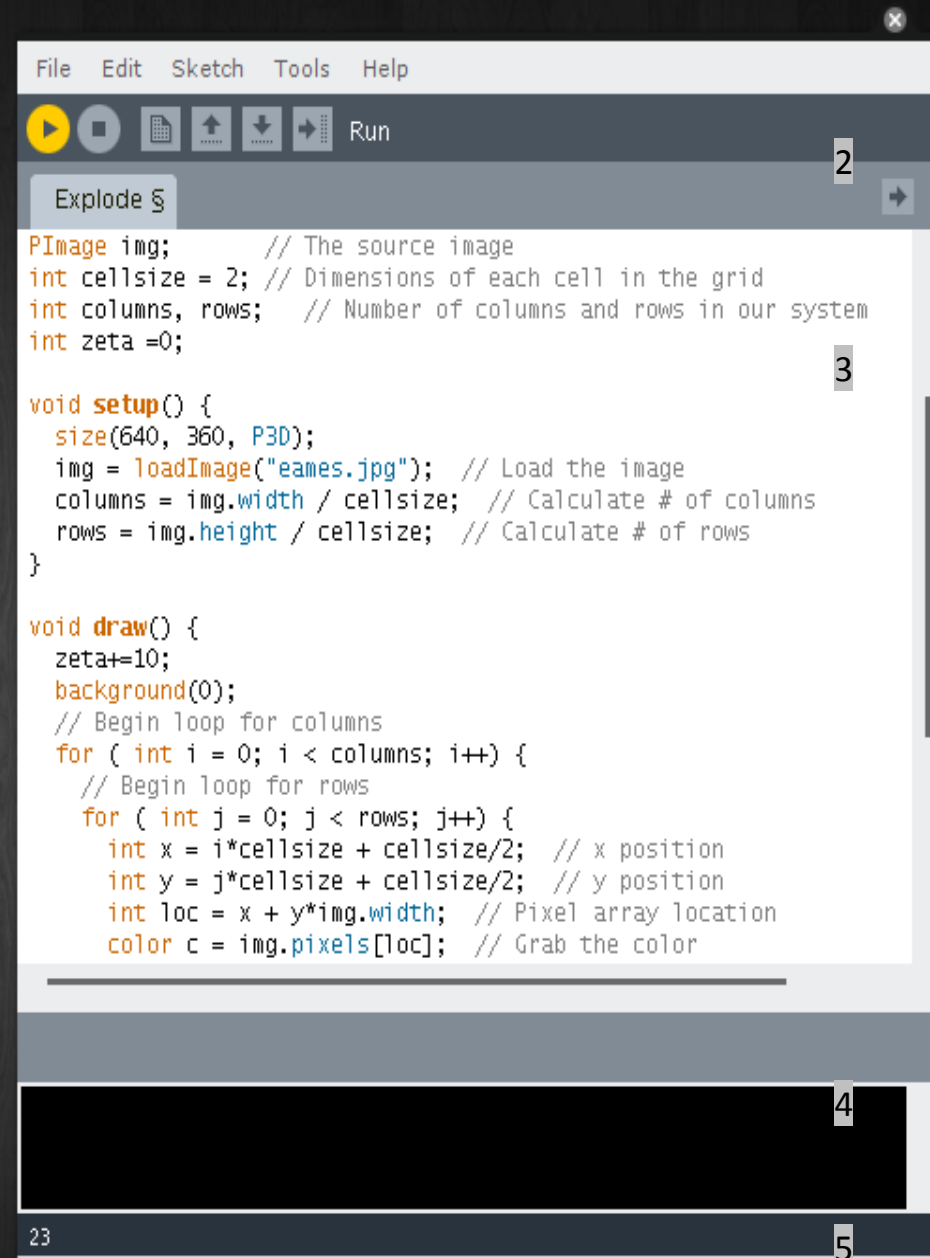
- Τρέξιμο - Run: Μεταφράζει τον κώδικα, ανοίγει ένα παράθυρο εμφάνισης και τρέχει το πρόγραμμα μέσα σε αυτό. Πατώντας το shift, αντί για τρέξιμο γίνεται παρουσίαση του προγράμματος.
- Τερματισμός - Stop: Τερματίζει το πρόγραμμα το οποίο τρέχει.
- Νέο - New: Δημιουργεί ένα νέο σκίτσο (έργο) στο τρέχον παράθυρο. Για την δημιουργία ενός σκίτσου σε νέο παράθυρο, ο χρήστης πρέπει να πάει στο μενού "Φάκελος" (file) και να επιλέξει από εκεί "Νέο" (new).

- Άνοιγμα - Open: Παρέχει ένα μενού επιλογών για το άνοιγμα αρχείων από το βιβλίο σκίτσων, άνοιγμα ενός παραδείγματος ή άνοιγμα ενός σκίτσου από οπουδήποτε στον υπολογιστή. Εδώ πρέπει να σημειωθεί πως το περιβάλλον ανάπτυξης περιλαμβάνει μία εξαιρετική ποικιλία και αριθμό σκίτσων, τα οποία εκτείνονται από πολύ απλά σκίτσα μέχρι πολύ περίπλοκα με εντυπωσιακή έξοδο και μπορούν να χρησιμοποιηθούν για την εκμάθηση της γλώσσας. Όμοια με την χρήση της επιλογής "Νέο" αν χρησιμοποιηθεί το κουμπί από την γραμμή εργαλείων, τότε το σκίτσο που ανοίγεται θα αντικαταστήσει το υπάρχον, ενώ αν γίνει χρήση του μενού "Φάκελος" -> "Άνοιγμα" (File->Open) τότε το σκίτσο θα ανοίξει σε νέο παράθυρο.
- Αποθήκευση - Save: Αποθηκεύει το τρέχον σκίτσο στην τρέχουσα τοποθεσία. Στην περίπτωση που ο χρήστης επιθυμεί να δώσει ένα νέο όνομα στο σκίτσο, πρέπει να επιλέξει "Αποθήκευση" από το μενού "Φάκελος".
- Εξαγωγή - Export: Εξάγει το συγκεκριμένο σκίτσο ως ένα Java Applet "ενσωματωμένο" σε ένα αρχείο HTML. Ο φάκελος που περιέχει τα αρχεία αυτά ανοίγεται αυτόματα. Για να φορτώσει κάποιος το λογισμικό, κάνει κλικ στο index.html και αυτό φορτώνεται στον προεπιλεγμένο φυλλομετρητή. Πατώντας το shift, το έργο εξάγεται σε μορφή εφαρμογής (application) αντί applet. Αξίζει να σημειωθεί πως κάθε νέα εξαγωγή του σκίτσου, διαγράφει τα προηγούμενα αποτελέσματα.

Περισσότερες επιλογές υπάρχουν διαθέσιμες μέσα στα μενού Φάκελος (File), Επεξεργασία (Edit), Σκίτσο (Sketch), Εργαλεία (Tools), Βοήθεια (Help).



1



2

3

4

5

- 1.Εικόνα του παράθυρου προβολής (display window). 2.Ετικέτες εργαλείων 3.Κειμενογράφος 4.Περιοχή μηνυμάτων 5.Περιοχή κειμένου.
6.(σελ 29) Σύστημα συντεταγμένων στις δύο διαστάσεις 7. Σύστημα συντεταγμένων στις τρεις διαστάσεις.

Σκίτσα

Όλα τα έργα στην Processing ονομάζονται σκίτσα (Sketches). Κάθε σκίτσο έχει τον δικό του φάκελο. Το κύριο αρχείο προγράμματος για κάθε σκίτσο έχει το ίδιο όνομα με τον φάκελο και βρίσκεται μέσα σε αυτόν. Για παράδειγμα, αν το σκίτσο ονομάζεται "Upatras" τότε ο φάκελος στον οποίο βρίσκεται ονομάζεται "Upatras" και το κύριο αρχείο ονομάζεται "Upatras.pde". Η επέκταση .pde του αρχείου δείχνει ότι πρόκειται για αρχείο σκίτσου της Processing και είναι ένα ακρωνύμιο που σημαίνει Processing Development Environment δηλαδή Περιβάλλον Ανάπτυξης της Processing.

Ένας φάκελος σχεδίου καμιά φορά περιέχει και άλλους φακέλους για αρχεία πολυμέσων -εικόνες, ήχους, γραμματοσειρές κλπ- καθώς και βιβλιοθήκες κώδικα. Όταν μία γραμματοσειρά ή μία εικόνα προστίθεται στο σκίτσο επιλέγοντας "Προσθήκη Αρχείου.." ("Add File..") από το μενού Σκίτσο ("Sketch"), ένας φάκελος με το όνομα "Data" (Δεδομένα) δημιουργείται αυτόματα. Αρχεία μπορούν επίσης να προστεθούν στο έργο με επιλογή και σύρσιμο στον κειμενογράφο - διαδικασία γνωστή ως "drag 'n' drop"- με αποτέλεσμα να δημιουργείται αυτόματα ένα αντίγραφο τους στον υποφάκελο δεδομένων. Όλες οι εικόνες, ήχοι, γραμματοσειρές και λοιπά αρχεία που έχουν φορτωθεί στο σκίτσο πρέπει να βρίσκονται στον υποφάκελο δεδομένων του σκίτσου. Τα έργα γενικά αποθηκεύονται στον φάκελο Processing, κάτι το οποίο διαφέρει σαν τοποθεσία από σύστημα σε σύστημα, καθώς εξαρτάται από το λειτουργικό σύστημα και τις ρυθμίσεις που έχουν γίνει. Για να βρει κάποιος γρήγορα τον φάκελο αυτόν, επιλέγει τις "Ρυθμίσεις" ("Preferences") από το μενού "Φάκελος" ("Folder") και κοιτά στην "Τοποθεσία του Βιβλίου Σχεδίων" ("Sketchbook Location").

Είναι δυνατόν να υπάρχουν πολλαπλά αρχεία κώδικα σε ένα σκίτσο. Αυτά μπορεί να είναι αρχεία Processing με την επέκταση .pde ή αρχεία Java με την επέκταση .java. Για την δημιουργία αρχείου, ο χρήστης μπορεί να πατήσει στο κουμπί με το βελάκι δεξιά από τις ετικέτες (tab) των φακέλων. Εκεί, υπάρχει πρόσβαση στην δημιουργία, διαγραφή και μετονομασία των αρχείων που αποτελούν το συγκεκριμένο σκίτσο. Το να δουλεύει κάποιος με πολλαπλά αρχεία, βοηθάει στην επαναχρησιμοποίηση του κώδικα και στην διάσπαση των προγραμμάτων σε υποπρογράμματα, με όλα τα πλεονεκτήματα που αυτό συνεπάγεται.

Η Processing χρησιμοποιεί ένα Καρτεσιανό σύστημα συντεταγμένων με αρχή των αξόνων την αριστερά πάνω γωνία. Αν το πρόγραμμα είναι 1920 εικονοστοιχεία σε πλάτος και 1200 σε ύψος, τότε το σημείο (0,0) είναι το πάνω αριστερό εικονοστοιχείο και το (1920,1200) είναι το κάτω δεξιά. Πρέπει να τονιστεί ότι το τελευταίο ορατό εικονοστοιχείο στην κάτω δεξιά γωνία του παραδείγματος μας είναι στο σημείο (1919,1199) επειδή τα εικονοστοιχεία σχεδιάζονται προς τα δεξιά και κάτω από την συντεταγμένη. Η Processing μπορεί επίσης να εξομοιώσει σχεδίαση σε τρεις διαστάσεις. Στην επιφάνεια της εικόνας η τιμή του άξονα z είναι μηδέν, με τις αρνητικές τιμές να υπάρχουν καθώς κινούμαστε προς τα πίσω στον χώρο. Σημειώνεται πως όταν γίνεται σχεδίαση στις τρεις διαστάσεις, η κάμερα τοποθετείται στο κέντρο της οθόνης.

Προγραμματισμός

Βασικός Τρόπος

Η Processing επιτρέπει στον χρήστη να προγραμματίσει σε τρία επίπεδα πολυπλοκότητας: Βασικός τρόπος, Συνεχής τρόπος και Java τρόπος. Άνθρωποι νέοι στον προγραμματισμό ενδείκνυται να ξεκινήσουν με τον Βασικό τρόπο για να μάθουν περισσότερα για τις συντεταγμένες, τις μεταβλητές και τους βρόγχους πριν περάσουν στον Συνεχή ή τον Java τρόπο.

Ο βασικός τρόπος χρησιμοποιείται για την σχεδίαση στατικών εικόνων και την εκμάθηση των βασικών χαρακτηριστικών του προγραμματισμού. Απλές γραμμές κώδικα έχουν άμεση αναπαράσταση στην οθόνη. Λόγου χάρη, το παρακάτω παράδειγμα σχεδιάζει ένα κίτρινο ορθογώνιο στην οθόνη. Η έξοδος του προγράμματος φαίνεται στο διπλανό σχήμα.

Κώδικας:

```
size(200, 200);  
background(255);  
noStroke();  
fill(255, 204, 0);  
rect(30, 20, 50, 50);
```

Έξοδος



Προγραμματισμός

Συνεχής τρόπος

Αυτός ο τρόπος παρέχει μία δομή `setup()` η οποία τρέχει μία φορά όταν το πρόγραμμα ξεκινάει και μία `draw()` της οποίας η προεπιλεγμένη συμπεριφορά είναι να επαναλαμβάνει αέναα τον κώδικα που περιέχεται. Αυτή η δομή επιτρέπει το γράψιμο συναρτήσεων και κλάσεων ειδικής κατασκευής καθώς και την χρήση γεγονότων που προέρχονται από το πληκτρολόγιο ή το ποντίκι. Λόγου χάρη, το παρακάτω παράδειγμα σχεδιάζει τέσσερις κύκλους στην οθόνη και αξιοποιεί μία ειδικά γραμμένη συνάρτηση η οποία ονομάζεται `circles()`. Η συνάρτηση αυτή, δεν είναι μέρος της Processing, αλλά γράφτηκε ως παράδειγμα. Πρέπει να τονιστεί πως ο κώδικας μέσα στην `draw()` τρέχει μόνο μία φορά και αυτό διότι έχει γίνει κλήση της `noLoop()` μέσα στην `setup()`.

Κώδικας:

```
void setup() {
  size(200, 200);
  noStroke();
  background(255);
  fill(0, 102, 153, 204);
  smooth();
  noLoop();
}
void draw() {
  circles(40, 80);
  circles(90, 70);
}
void circles(int x, int y) {
  ellipse(x, y, 50, 50);
  ellipse(x+20, y+20, 60, 60);
}
```

Το παρακάτω παράδειγμα σχεδιάζει ορθογώνια τα οποία ακολουθούν την θέση του ποντικιού (η οποία είναι αποθηκευμένη κάθε στιγμή στην θέση `mouseX` και `mouseY`). Ο κώδικας της `draw()` τρέχει αέναα, μέχρι το πρόγραμμα να σταματήσει, δημιουργώντας έτσι προοπτική για κίνηση και αλληλεπίδραση.

Η έξοδος των δύο αυτών προγραμμάτων παρουσιάζεται στην πρώτη και δεύτερη εικόνα της επόμενης σελίδας.

Κώδικας:

```
void setup() {
  size(200, 200);
  rectMode(CENTER);
  noStroke();
  fill(0, 102, 153, 204);
}
void draw() {
  background(255);
  rect(width-mouseX, height-mouseY, 50, 50);
  rect(mouseX, mouseY, 50, 50);
}
```

Προγραμματισμός

Java Τρόπος

Αυτός ο τρόπος είναι ο πιο ευέλικτος, επιτρέποντας ολόκληρα προγράμματα σε Java να γράφονται μέσα στο περιβάλλον της Processing -- εφόσον εξακολουθούν να είναι υποκλάσεις του PApplet. Αυτός ο τρόπος είναι για πιο προχωρημένους χρήστες και δεν ενδείκνυται γενικά. Η χρήση αυτού του τρόπου σημαίνει πως επιπρόσθετες ετικέτες (tab) δεν θα συνεχίσουν να είναι εσωτερικές, που σημαίνει ότι κάποιος θα πρέπει να κάνει επιπρόσθετη δουλειά για να τις κάνει να δουλεύουν σωστά με το PApplet οικοδεσπότη (host). Δεν είναι απαραίτητο να χρησιμοποιηθεί αυτός ο τρόπος απλά και μόνο για να υπάρχει πρόσβαση σε χαρακτηριστικά της γλώσσας Java. Το παρακάτω παράδειγμα δείχνει τα παραπάνω:

Κώδικας:

```
public class Upatras extends PApplet
{
  void setup() {
    size(200, 200);
    rectMode(CENTER);
    noStroke();
    fill(0, 102, 153, 204);
  }
  void draw() {
    background(255);
    rect(width-mouseX, height-mouseY, 50, 50);
    rect(mouseX, mouseY, 50, 50);
  }
}
```

Η έξοδος του προγράμματος είναι ακριβώς η ίδια με το δεύτερο πρόγραμμα από την προηγούμενη σελίδα. Η θέση των δύο τετραγώνων είναι συνάρτηση της θέσης του ποντικιού στην οθόνη (κάτι το οποίο δεν μπορεί να φανεί στην εικόνα).

Έξοδος:



Εξαγωγή Applet

Η επιλογή της εξαγωγής πακετάρει το σκίτσο με τρόπο που να μπορεί να παίξει σε έναν φυλλομετρητή ιστού. Όταν ο κώδικας εξάγεται από την Processing μετατρέπεται σε κώδικα Java και έπειτα μεταφράζεται ως Java Applet. Όταν ολόκληρο το έργο εξάγεται, ένα σύνολο αρχείων γράφεται σε έναν φάκελο που ονομάζεται "applet" και ο οποίος δημιουργείται μέσα στον φάκελο του σκίτσου. Τα αρχεία από τον φάκελο του σκίτσου πακετάρονται σε ένα συμπιεσμένο αρχείο τύπου jar με το ίδιο όνομα. Για παράδειγμα αν το σκίτσο ονομάζεται "UPatras" τότε το συμπιεσμένο αρχείο ονομάζεται "UPatras.jar". Το applet περιέχει τα παρακάτω:

- index.html: Αρχείο html με το applet να είναι ενσωματωμένο σε αυτό και με ένα σύνδεσμο στον πηγαίο κώδικα και την ιστοσελίδα της Processing. Πατώντας διπλό κλικ, το αρχείο αυτό ανοίγει το έργο στον προεπιλεγμένο φυλλομετρητή ιστού.
- UPatras.jar: Συμπιεσμένο αρχείο Java το οποίο περιέχει όλα τα απαραίτητα αρχεία για να τρέξει το σκίτσο. Περιέχει τις κλάσεις του πυρήνα της Processing, αυτές που έχουν γραφτεί για το συγκεκριμένο σκίτσο καθώς και όλα τα αρχεία πολυμέσων τα οποία βρίσκονταν στον φάκελο δεδομένων του σκίτσου όπως εικόνες, γραμματοσειρές και ήχοι.
- UPatras.java: Το αρχείο Java το οποίο δημιουργείται από το προεπεξεργαστή προερχόμενο από το αρχείο του περιβάλλοντος ανάπτυξης της Processing. Αυτό είναι το καθ' εαυτώ αρχείο το οποίο μεταφράζεται σε Applet

από τον μεταφραστή Java ο οποίος χρησιμοποιείται από την Processing.

- loading.gif: Μία εικόνα η οποία εμφανίζεται όταν το πρόγραμμα φορτώνεται στον φυλλομετρητή ιστού.

Κάθε φορά που το σκίτσο εξάγεται, τα περιεχόμενα του φακέλου "applet" διαγράφονται και τα αρχεία γράφονται πάλι από την αρχή. Οποιοσδήποτε αλλαγές στο αρχείο index.html χάνονται. Για την προσαρμογή του κώδικα html ο οποίος δημιουργείται κατά την εξαγωγή του σκίτσου, πρέπει να γίνει αντιγραφή του applet.html από το Processing-> lib->export στην ρίζα της ιεραρχίας φακέλων του σκίτσου.

Τα αρχεία πολυμέσων τα οποία δεν απαιτούνται από το Applet πρέπει να διαγραφούν από τον φάκελο δεδομένων πριν την εξαγωγή για να διατηρηθεί το μέγεθος του τελικού αρχείου μικρό. Για παράδειγμα, αν υπάρχουν μη χρησιμοποιημένες εικόνες στον φάκελο δεδομένων, αυτές θα καταλήξουν στο τελικό jar αυξάνοντας σημαντικά το μέγεθός του.

Όταν γίνεται εξαγωγή του Applet, μία περιγραφή μπορεί να εμπεριέχεται στην προεπιλεγμένη html σελίδα η οποία δημιουργείται. Η Processing θα περιλάβει οτιδήποτε υπάρχει μεταξύ της πρώτης ε-

Κώδικας:

```
/**
 * Μη μου τους κύκλους τάραττε.
 */
```

Εξαγωγή Applet

Περιορισμοί Ασφαλείας των Applet

εμφάνισης ενός `"/**"` (χωρίς τα εισαγωγικά) μέχρι το επόμενο `"*/"` ως ένα σχόλιο Javadoc. Έτσι αν κάποιος γράψει στην αρχή του παραπάνω παραδείγματος με τους κύκλους το παραπάνω κομμάτι κώδικα τότε αυτό θα καταλήξει να εμπεριέχεται στην τελική εξαχθείσα ιστοσελίδα. Καθώς πρόκειται για σχόλια σε μορφή Javadoc, το κείμενο μπορεί και θα πρέπει να περιέχει οποιαδήποτε έγκυρη έκφραση html καθώς αυτή θα περιληφθεί αυτούσια στην ιστοσελίδα.

Τα Applet τα οποία τρέχουν μέσα σε έναν φυλλομετρητή ιστού έχουν διάφορους περιορισμούς ασφαλείας για να προστατέψουν τους χρήστες του διαδικτύου. Για παράδειγμα, ένα applet δεν μπορεί να διαβάσει αρχεία από τον τοπικό σκληρό δίσκο, ή να συνδεθεί σε εξυπηρετητές εκτός από αυτόν από τον οποίο τρέχει. Για να αρθούν αυτοί οι περιορισμοί, είναι δυνατό να "υπογραφεί" το applet. Σε αυτή την περίπτωση, ένα παράθυρο διαλόγου θα εμφανιστεί και θα ζητήσει από τον τελικό χρήστη αν εμπιστεύεται ή όχι το applet, και αν επιτρέπει από την εφαρμογή να εκτελέσει ενέργειες οι οποίες μπορεί να αποτελούν ρίσκο για την ασφάλειά του. Περισσότερα για τα Applets μπορεί κάποιος να αναζητήσει στα [25,26] και για την ασφάλεια των applets στο [27]

Nobody got anywhere in the world by simply being content.

Louis L'Amour (US Novelist of westerns, 1908 – 1988)

Liferay

Εισαγωγή

50

Μία πύλη (portal), γνωστή και ως εταιρική πύλη ή επιχειρηματική πύλη πληροφοριών (corporate portal και enterprise information portal αντίστοιχα), αποτελεί ένα πλαίσιο για την ενσωμάτωση πληροφοριών, ανθρώπων και διαδικασιών ανάμεσα στα σύνορα οργανισμών, οργανώσεων και φορέων. Παρέχει ένα ενοποιημένο σημείο πρόσβασης [39], συχνά με την μορφή μίας διεπαφής χρήστη βασισμένης στον ιστό (web-based user interface, WUI) και είναι σχεδιασμένο έτσι ώστε να συλλέγει και να εξατομικεύει την πληροφορία μέσω μεμονωμένων (ανάλογα με την εφαρμογή, application-specific) portlet, μικρές εφαρμογές ιστού οι οποίες τρέχουν σε ένα μέρος μίας σελίδας. Ένα χαρακτηριστικό γνώρισμα των πυλών είναι η κατανομημένη συνεισφορά και διαχείριση περιεχομένου, κάτι το οποίο διατηρεί τις πληροφορίες διαρκώς ενημερωμένες.

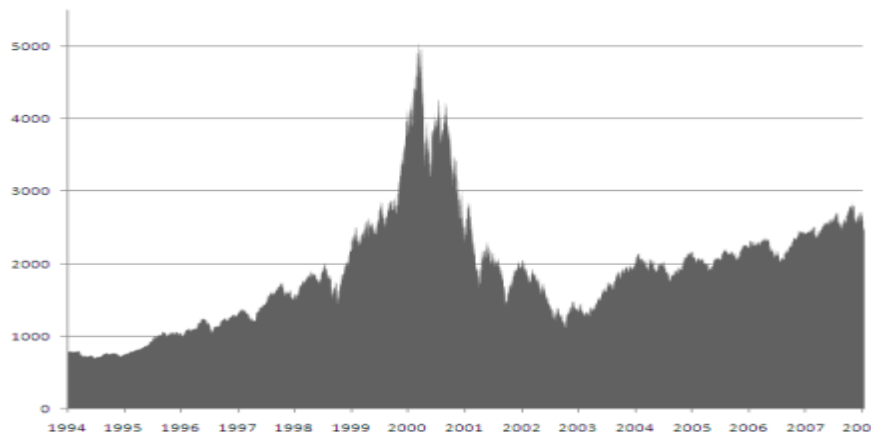
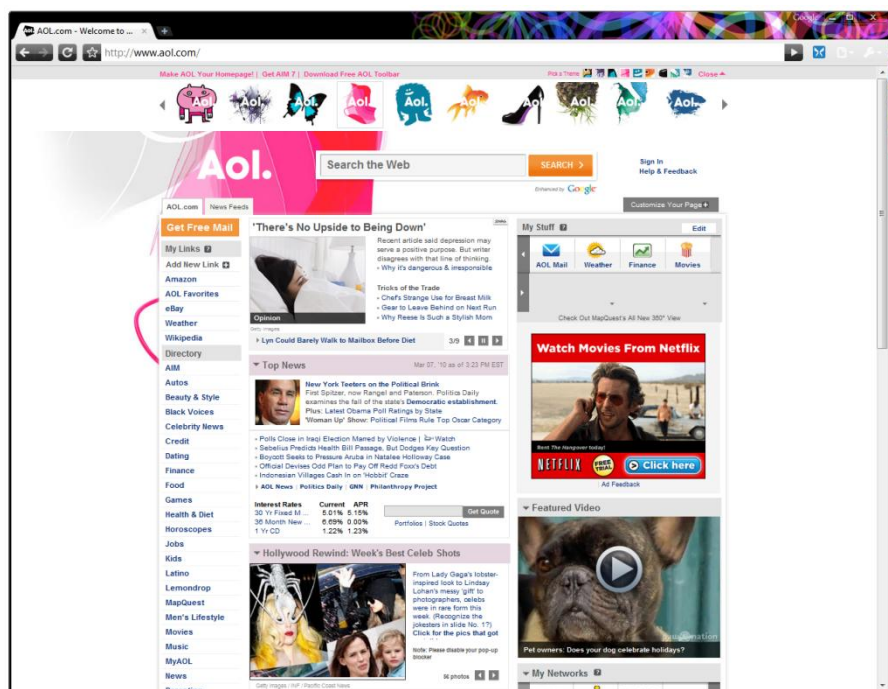
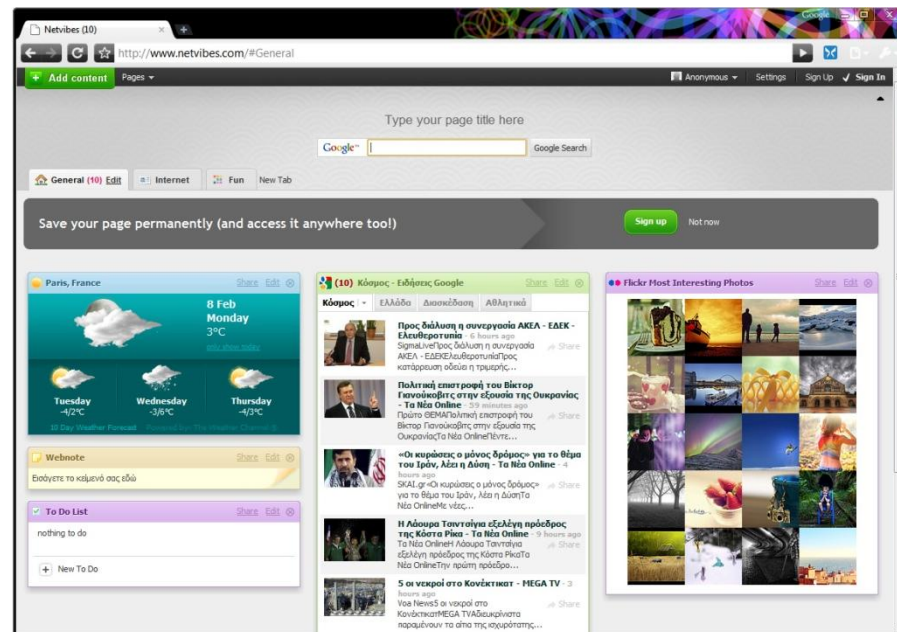
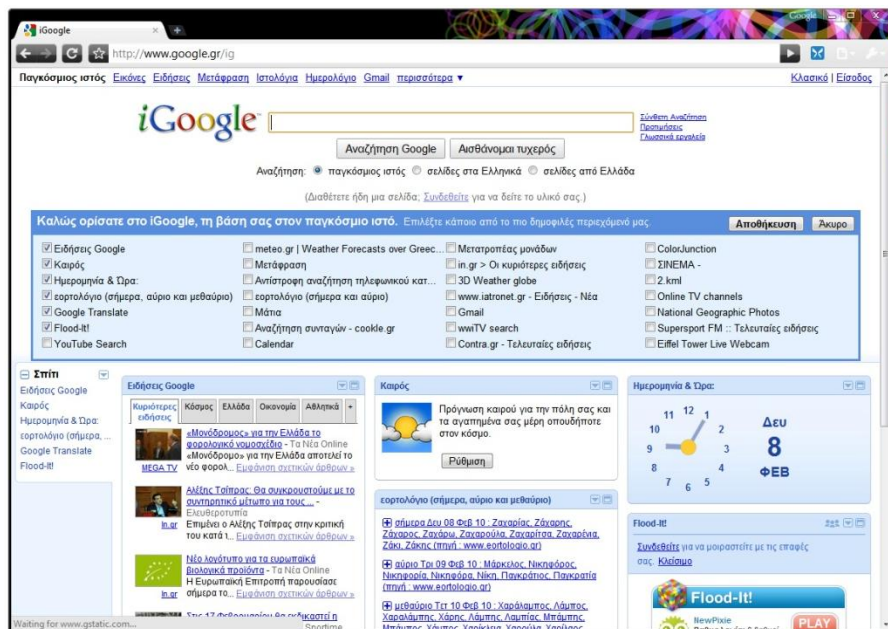
Μια διαδικτυακή πύλη (web portal) είναι ένα ιδιαίτερο είδος πύλης το οποίο παρουσιάζει πληροφορίες από διάφορες πηγές σε ένα ενιαίο τρόπο και είναι προσβάσιμα από φυλλομετρητή ιστού. Εκτός από το κλασσικό πια χαρακτηριστικό της μηχανής αναζήτησης, οι δικτυακές πύλες προσφέρουν και άλλες υπηρεσίες όπως ηλεκτρονικό ταχυδρομείο, τιμές μετοχών, ειδήσεις, πληροφορίες, βάσεις δεδομένων και ψυχαγωγία. Οι διαδικτυακές πύλες προσφέρουν στις επιχειρήσεις έναν τρόπο να είναι συνεπείς όσον αφορά την εμφάνιση και παρουσίασή τους στον ιστό με έλεγ-

χο πρόσβασης, ομάδες χρηστών και διάφορες μεθόδους για εφαρμογές και βάσεις δεδομένων, οι οποίες σε διαφορετική περίπτωση θα ήταν εντελώς ξεχωριστές οντότητες. Παραδείγματα δημοσίων πυλών του παγκόσμιου ιστού είναι το MSNBC [35], η Yahoo![36], η AOL[37], το iGoogle[38] και τα Netvibes[50].

Όπως ορίζεται στις προδιαγραφές του JSR-168 [40] "ένα portal είναι μια δικτυακή εφαρμογή που συνήθως παρέχει προσωποποίηση, απλή σύνδεση, συνάθροιση περιεχομένου από διαφορετικές πηγές και φιλοξενεί το στρώμα παρουσίασης συστημάτων πληροφοριών. Η συνάθροιση είναι η πράξη της ενσωμάτωσης του περιεχομένου από διαφορετικές πηγές μέσα σε μια ιστοσελίδα."

Οι διαδικτυακές πύλες μπορούν να κατηγοριοποιηθούν σύμφωνα με ένα πλήθος κριτηρίων, ανάλογα με την περιοχή κάλυψης, το εύρος των χρηστών, τον τρόπο χρήσης κ.α. Αν και οι παρακάτω κατηγοριοποιήσεις δεν είναι τυπικές, αποτελούν έναν τρόπο διαφοροποίησής του περιεχομένου και του τρόπου λειτουργίας τους.

Οριζόντια και Κάθετα: Δύο ευρείες κατηγοριοποιήσεις των διαδικτυακών πυλών είναι οι οριζόντιες πύλες, οι οποίες καλύπτουν πολλές περιοχές και οι κάθετες πύλες οι οποίες εστιάζουν σε μία συγκεκριμένη περιοχή. Ένας άλλος ορισμός της οριζόντιας πύλης είναι ότι



(από κάτω αριστερά, και αντίστροφα από την φορά του ρολογιού). Η κεντρική εικόνα της διαδικτυακής πύλης του AOL (εμφανής είναι η αλλαγή δυνατού θέματος), η κεντρική σελίδα του iGoogle, με δυνατότητες συρσίματος με το ποντίκι (drag 'n' drop) κλπ, μία άποψη της πύλης Netvibes και μία εικόνα που απεικονίζει την φύσκα του dot com. Ο ιδιαίτερα βαρύς τεχνολογικά δείκτης Nasdaq Composite κορυφώθηκε σε 5.048 μονάδες τον Μάρτιο του 2000, ένα γεγονός που αντικατοπτρίζει την «ακμή» της φύσκας

Εισαγωγή

χρησιμοποιείται ως πλατφόρμα για μία πληθώρα εταιριών στον ίδιο οικονομικό τομέα ή από παρεμφερή τύπο κατασκευαστών, εμπορικών αντιπροσώπων ή διανομέων. Κατ' επέκταση, μία κάθετη πύλη αποτελεί ένα ειδικευμένο σημείο πρόσβασης σε μία συγκεκριμένη αγορά ή βιομηχανική εσοχή, ερευνητική περιοχή ή περιοχή ενδιαφέροντος, και καλείται επίσης vortal.

Προσωπικές Πύλες: Μία προσωπική πύλη είναι ένας χώρος στον παγκόσμιο ιστό ο οποίος τυπικά παρέχει δυνατότητες προσωποποίησης στους επισκέπτες της, παρέχοντας ένα μονοπάτι σε περεταίρω περιεχόμενο. Είναι σχεδιασμένο ώστε να χρησιμοποιεί καταγεγραμμένες εφαρμογές, διάφορους αριθμούς και τύπους υλικολογισμικού (middleware) και υλικού (hardware) ώστε να παρέχει υπηρεσίες από ένα πλήθος διαφορετικών πηγών. Επίσης, οι διαδικτυακές πύλες επιχειρήσεων είναι σχεδιασμένες ώστε να κατανέμουν την συνεργασία σε διάφορους χώρους εργασίας. Μία επιπλέον απαίτηση που δημιουργείται από τις επιχειρήσεις είναι η δυνατότητα των διαδικτυακών πυλών να παρουσιάζουν περιεχόμενο σε μία πληθώρα από πλατφόρμες όπως προσωπικοί υπολογιστές (Personal Computer – PC), προσωπικοί ψηφιακοί οδηγοί (Personal Digital Assistants, PDAs), κινητά τηλέφωνα κλπ. Πληροφορίες, νέα και ενημερώσεις είναι μερικά μόνο παραδείγματα περιεχομένου το οποίο θα μπορούσε να παραδίδεται διαμέσου μιας τέτοιας πύλης. Οι προσωπικές πύλες θα μπορούσαν σχετίζονται με οποιοδήποτε συγκεκριμένο πεδίο όπως για παράδειγμα να παρέχουν πληροφορίες για φίλους σε ένα κοινωνικό δίκτυο ή συνδέσμους σε εξωτερικό περιεχόμενο το οποίο θα μπορούσε να βοηθήσει περεταίρω τους χρήστες. Οι πύλες δεν περιορίζονται φυσικά στις λειτουργίες μίας απλής ιστοσελίδας, καθώς μπορούν

να συνδυάσουν πληροφορία από ένα πλήθος πηγών, πολλαπλά επίπεδα χρηστών, αναζητήσιμο περιεχόμενο καθώς και πληθώρα τρόπων αποδοτικής διαχείρισης περιεχομένου.

Περιφερειακές Πύλες: Παράλληλα με την ανάπτυξη και την επιτυχία των διεθνών προσωπικών πυλών όπως το Yahoo!, παρουσιάστηκαν επίσης περιφερειακές παραλλαγές αυτών. Κάποιες περιφερειακές πύλες παρέχουν πληροφορίες τοπικού χαρακτήρα, όπως πρόγνωση καιρού, χάρτες των δρόμων και δεδομένα τύπου χρυσού οδηγού. Ένα εμφανές αποτέλεσμα είναι η δημιουργία νέων ειδών αγορών και υπηρεσιών οι οποίες στο παρελθόν ήταν αδιανόητες (e-malls etc). Διαδικτυακές πύλες του τύπου "Τοπικό Περιεχόμενο – Παγκόσμια προσέγγιση" ("Local content - global reach") έχουν κάνει την εμφάνισή τους σε χώρες όπως η Ελλάδα, η Κορέα, η Ινδία, η Κίνα, η Ιταλία κλπ. Παράδειγμα στην χώρα μας αποτελεί το www.in.gr.

Κυβερνητικές Πύλες: Καθώς η φούσκα του dot-com έφτανε στο τέλος της στις αρχές της δεκαετίας του 2000, πολλές κυβερνήσεις είχαν ήδη δεσμευτεί για την δημιουργία διαδικτυακών πυλών για την εξυπηρέτηση των πολιτών τους. Για παράδειγμα, στις ΗΠΑ η κύρια πύλη είναι το usa.gov και το gobiernousa.gov στα αγγλικά και τα ισπανικά αντίστοιχα σε συνδυασμό με άλλες πύλες οι οποίες δημιουργήθηκαν για συγκεκριμένα ακροατήρια όπως το disability.gov για άτομα με ειδικές ανάγκες, ενώ στην Αγγλία είναι το direct.gov για την εξυπηρέτηση των πολιτών και το businesslink.gov.uk για την εξυπηρέτηση των εταιρειών. Ένα άλλο παράδειγμα είναι η επίσημη πύλη για την Ευρωπαϊκή Ένωση, η οποία βρίσκεται στην διεύθυνση Europa.eu.

Εισαγωγή

Εταιρικές Πύλες: Τα εταιρικά διαδίκτυα (intranet) άρχισαν σταδιακά να απαντώνται συχνά μέσα στα μέσα της δεκαετίας του 1990. Καθώς τα ενδοδίκτυα αυτά άρχισαν να πολλαπλασιάζονται σε μέγεθος και πολυπλοκότητα, οι υπεύθυνοι ιστοσελίδων βρέθηκαν αντιμέτωποι με τις αυξανόμενες προκλήσεις περιεχομένου και διαχείρισης χρηστών. Μια ενοποιημένη εικόνα των στοιχείων της εταιρείας κρίθηκε ανεπαρκής. Οι χρήστες είχαν ανάγκες εξατομίκευσης και προσαρμογής. Σε σπάνιες περιπτώσεις, οι υπεύθυνοι διαχείρισης των ιστοσελίδων, αν είχαν τις κατάλληλες δεξιότητες, κατάφερναν να προσφέρουν κάποιες δυνατότητες, όμως οι περισσότερες περιπτώσεις κατέληγαν στο να απωθήσουν τους χρήστες από το να χρησιμοποιήσουν το ενδοδίκτυο. Πολλές εταιρείες άρχισαν να προσφέρουν εργαλεία ώστε να βοηθήσουν τους υπεύθυνους των ιστοχώρων να διαχειριστούν τα δεδομένα, τις εφαρμογές και γενικότερα την πληροφορία ευκολότερα και μέσω εξατομικευμένων όψεων. Οι λύσεις των διαδικτυακών πυλών περιλαμβάνουν διαχείριση ροών εργασιών (workflow management), συνεργασία μεταξύ των διαφορετικών ομάδων εργασίας (work group collaboration) καθώς και εφαρμογές διαχείρισης περιεχομένου πολιτικής (policy management content application). Οι περισσότερες επιτρέπουν εσωτερική και εξωτερική πρόσβαση σε συγκεκριμένες εταιρικές πληροφορίες χρησιμοποιώντας ασφαλή ταυτοποίηση χρήστη ή καθολική σύνδεση ενός σημείου (Single Sign On – SSO). Με την αύξηση του περιεχομένου το οποίο δημιουργείται από χρήστες, τις ανομοιογενείς αποθήκες δεδομένων και τους νέους τύπους αρχείων, καθίσταται απαραίτητη η συμβολή πληροφοριακών αρχιτεκτονών και ταξινομητών οι οποίοι θα επιτρέψουν στους χρήστες να χαρακτηρίσουν (κατατάξουν) τα δεδομένα.

Φιλοξενούμενες Πύλες: Καθώς οι εταιρικές πύλες κέρδιζαν σε δημοτικότητα, ένα πλήθος εταιρειών άρχισε να τις προσφέρει ως μία φιλοξενούμενη υπηρεσία. Η αγορά των φιλοξενούμενων πυλών άλλαξε ριζικά την δομή και την σύνθεση των πυλών. Σε πολλές περιπτώσεις προσέφεραν υπηρεσίες δημοσίευσης της πληροφορίας σε αντίθεση με πιο υψηλούς στόχους ενσωμάτωσης παλιών εφαρμογών ή παρουσίασης συγκεντρωμένων πληροφοριών από διάφορες πηγές ή βάσεις δεδομένων. Οι πρώτες εταιρείες φιλοξενούμενων πυλών -παραδείγματα των οποίων αποτελούν η Hyperoffice και η IntranetPortal επικεντρώθηκαν σε εφαρμογές συνεργασίας και χρονοπρογραμματισμού σε συνδυασμό με τον διαμοιρασμό εταιρικών δεδομένων. Καθώς οι φιλοξενούμενες πύλες κέρδιζαν σε δημοτικότητα, το σύνολο των χαρακτηριστικών που προσέφεραν άρχισε να μεγαλώνει, συμπεριλαμβάνοντας την φιλοξενία βάσεων δεδομένων, διαχείριση εγγράφων, ηλεκτρονικό ταχυδρομείο, περιοχές συζητήσεων κ.α. Αυτό το είδος πυλών προσφέρει αυτοματοποιημένη εξατομίκευση περιεχομένου το οποίο δημιουργείται από τα δομικά στοιχεία του συστήματος έτσι ώστε να προσφέρουν μία πλήρως προσωποποιημένη εμπειρία στους χρήστες τους. Από αυτή την άποψη, παραμένουν πιστοί στους αρχικούς στόχους των προγενέστερων εταιρικών διαδικτυακών πυλών.

Μεμονωμένες Πύλες (Domain Specific): Πλήθος πυλών έχει καταλήξει να είναι προσανατολισμένη προς μία συγκεκριμένη περιοχή, προσφέροντας πρόσβαση σε σχετικές ή σχετιζόμενες εταιρείες και υπηρεσίες. Ένα καλό παράδειγμα του συγκεκριμένου τύπου διαδικτυακών πυλών είναι οι πύλες που σχετίζονται με ιδιοκτησίες και παρέχουν πρόσβαση σε υπηρεσίες.

Ιστορική Αναδρομή

54

Θα μπορούσε κάποιος να υποστηρίξει πως η ιστορία των Συστημάτων Διαχείρισης Περιεχομένου είναι τόσο παλιά όσο και η ιστορία του ίδιου του περιεχομένου η οποία μάλλον συμπίπτει χρονικά με την ιστορία της ανθρώπινης γνώσης και την διαχείριση αυτής, συμπεριλαμβανομένης και της ιστορίας. Στην προσέγγιση που επιχειρείται στην συγκεκριμένη εργασία, δίνεται έμφαση σε πρακτικές οι οποίες άλλαξαν δραστικά τον τρόπο με τον οποίο οι άνθρωποι αντιλαμβάνονται την διαχείριση περιεχομένου και σε εργαλεία τα οποία βελτίωσαν ή προσέθεσαν κάτι στην μετάβαση αυτή. Αποτελεί κοινό τόπο πως στην σύγχρονη εποχή, η επεξεργασία και διαχείριση της των δεδομένων, της πληροφορίας και της γνώσης είναι εξαιρετικά πιο αποδοτική σε σύγκριση με αυτή των προγόνων μας.

Ταξιδεύοντας γρήγορα μέσα από την εφεύρεση της γραφής, την δημιουργία μεταφερόμενων τύπων εγγράφων τα οποία βασίζονται στο χαρτί -οι πάπυροι-, την σύνταξη των σελίδων σε βιβλία, την συλλογή βιβλίων σε βιβλιοθήκες, την μαζική παραγωγή βιβλίων με την ανακάλυψη της τυπογραφίας, την ταξινόμηση των βιβλίων σε καταλόγους και κατηγορίες διαφαίνεται η προσπάθεια καταγραφής, αναζήτησης και πλοήγησης μέσα στον ολόένα και αυξανόμενο όγκο πληροφοριών, περιεχομένου και γνώσης. Οι αλλαγές που συντελέστηκαν σε αυτούς τους αιώνες δεν διαφέρουν πολύ με τις προόδους που πραγματοποιήθηκαν στις δεκαετίες του δεύτερου μισού του εικοστού αιώνα στον τομέα της σύγχρονης ψηφιακής διαχείρισης του περιεχομένου.

Αρκετές σημαντικές πρόοδοι στο υλικό και το λογισμικό των υπολογιστών μπορούν να χρησιμοποιηθούν

για να σηματοδοτήσουν ιστορικά ορόσημα στην ιστορία της ανάπτυξης των συστημάτων διαχείρισης περιεχομένου τα τελευταία πενήντα χρόνια. Όσον αφορά το υλικό, ορόσημο αποτελεί ο ίδιος ο υπολογιστής, η αποθήκευση δεδομένων από διάτρητες κάρτες και μαγνητικές ταινίες σε σκληρούς δίσκους και δίσκους στερεάς κατάστασης, κοινόχρηστη πρόσβαση σε κεντρικούς υπολογιστές από τερματικά, ο προσωπικός υπολογιστής, το μοντέλο Πελάτη-Εξυπηρετητή, και φυσικά ο Παγκόσμιος Ιστός από διασυνδεδεμένους υπολογιστές στο διαδίκτυο. Οι εφευρέσεις που αφορούν το λογισμικό συνοψίζονται στα λειτουργικά συστήματα, τα συστήματα αρχείων, τις βάσεις δεδομένων, τον κατακευκτισμένο υπολογισμό, τις περιγραφικές γλώσσες (HTML και XML), όλο το σύνολο των πρωτοκόλλων που αφορούν το διαδίκτυο καθώς και όλες οι εφαρμογές, σουίτες γραφείου, λεξικολογικής ανάλυσης και προγραμματισμού.

Έτσι, στα μέσα της δεκαετίας του 1970 έχουμε διαχείριση περιεχομένου σε κεντρικούς υπολογιστές ή, καλύτερα, την ηλεκτρονική δημοσίευση (Mainframe CM or Electronic Publishing), στα μέσα της δεκαετίας του 1980 έχουμε την διαχείριση περιεχομένου στον προσωπικό υπολογιστή ή την δημοσίευση επιφάνειας εργασίας (Personal Computer CM or Desktop Publishing), στις αρχές του 1990 την διαχείριση περιεχομένου σε επίπεδο πελάτη-εξυπηρετητή και λίγο αργότερα την διαχείριση περιεχομένου στον Ιστό ή δημοσίευση στον Ιστό (Web Content Management or Web Publishing). Ενδεικτικοί σταθμοί σε αυτό το ταξίδι είναι η αυγή του προσωπικού υπολογιστή με μία διεπαφή κειμένου το 1977, το γραφικό περιβάλλον με τα συστήματα Xerox PARC και Apple Lisa το 1982,

Ιστορική Αναδρομή

το 1984 με την δημιουργία του κειμενογράφου της Apple (Mac Write, Mac Publisher), το 1992 με την Σουίτα Lotus Notes και το 1993 με τον Περιηγητή Ιστού Mosaic

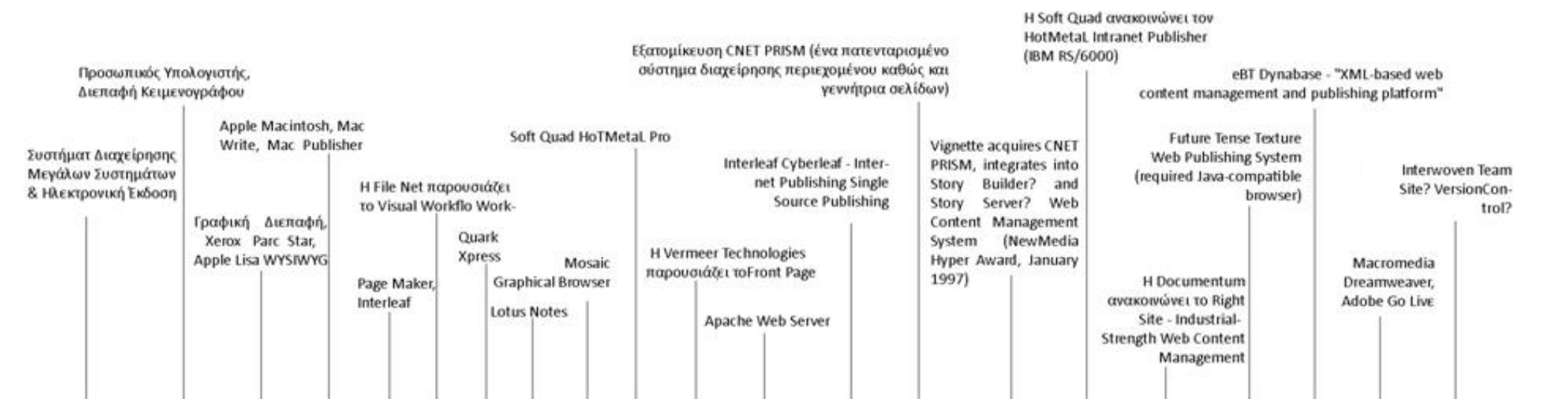
Στα μέσα της δεκαετίας του 1990 ένα πλήθος δημοσίων διαδικτυακών πυλών έκανε την εμφάνισή του όπως το Altavista, το AOL, το Excite και το Yahoo!. Αυτοί οι ιστοχώροι παρείχαν ένα καίριο σύνολο χαρακτηριστικών (όπως νέα, ηλεκτρονικό ταχυδρομείο, πρόβλεψη καιρού, παρακολούθηση μετοχών και αναζήτηση) τα οποία συχνά παρουσιάζονταν μέσα στα όρια συγκεκριμένων κουτιών, αυτό που σήμερα ονομάζεται portlet. Εταιρίες όλων των μεγεθών άρχισαν να βλέπουν την ανάγκη για μία παρεμφερή αρχική σελίδα για το πλήθος των εσωτερικών τους αποθετηρίων και εφαρμογών, και πολλές από αυτές άρχισαν να μετακινούνται προς τέτοιες τεχνολογίες βασισμένες στον ιστό [34].

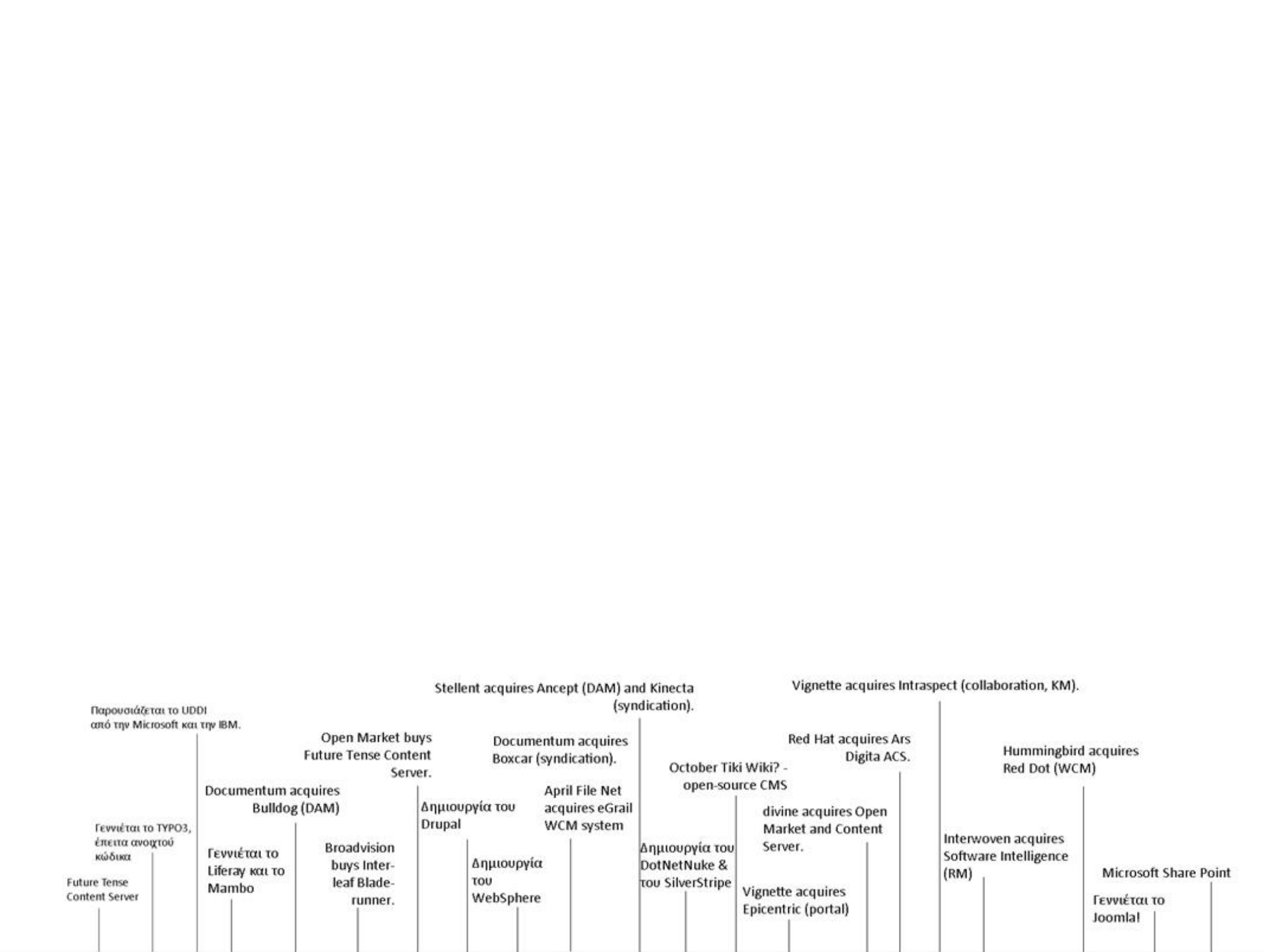
Μέχρι τα τέλη της δεκαετίας του 1990, οι εταιρείες ανάπτυξης και διακίνησης λογισμικού άρχισαν να αναπτύσσουν προπακεταρισμένες εταιρικές πύλες. Αυτά τα πακέτα λογισμικού αποτελούσαν εργαλεία για τις εταιρείες με τα οποία θα μπορούσαν να αναπτύξουν και να εγκαταστήσουν γρήγορα τις δικές τους εξατομικευμένες εκδοχές των εταιρικών πυλών. Αξίζει να σημειωθεί πως η πρώτη εταιρεία ανάπτυξης εμπορικού λογισμικού δικτυακής πύλης έκανε την εμφάνισή της το 1998. Πρωτεργάτες στην αγορά αποτελούσαν εταιρείες με μοναδική σύσταση στο διαδίκτυο όπως η Epicentric, η Plumtree Software και η Viador. Παρ' όλ' αυτά, δεν άργησαν να μπουν στο παιχνίδι πωλητές διακομιστών εφαρμογών (application server vendors) οι οποίοι είδαν τις πύλες ως μία ευκαιρία

Να μετατρέψουν την εκάστοτε τεχνολογία διακομιστή εφαρμογών σε εμπορεύσιμο προϊόν και υπηρεσία καθώς και οι πωλητές επιχειρηματικής νοημοσύνης (Business Intelligence Vendors) με ανταγωνιστικές προσφορές προϊόντων.

Στα τέλη της δεκαετίας του 1990, η διαδικτυακή πύλη αποτελούσε ένα δημοφιλές αγαθό. Με την διάδοση των φυλλομετρητών ιστού στα τέλη του 1990, πολλές εταιρείες προσπάθησαν να χτίσουν ή να αποκτήσουν μία διαδικτυακή πύλη, με σκοπό να κατέχουν εν τέλει μία θέση στην πίτα του διαδικτύου. Οι διαδικτυακές πύλες κέρδισαν την προσοχή καθώς για πολλούς χρήστες αποτελούσαν την αρχική σελίδα του φυλλομετρητή τους. Έτσι, η Netscape (βλ. Ιστορία του Internet) έγινε μέρος της America Online, η Walt Disney έθεσε σε λειτουργία το go.com, ενώ το Excite και @Home έγιναν μέρος της AT&T στα τέλη του 1990. Η γνωστή πύλη αναζήτησης Lycos θεωρήθηκε πιθανός στόχος άλλων μέσων μαζικής επικοινωνίας όπως το αμερικάνικο CBS. Η μόδα των πυλών, με τις παραδοσιακές εταιρείες μαζικής ενημέρωσης να ανταγωνίζονται μεταξύ τους στον αγώνα για διαδικτυακές ιδιοκτησίες, εξασθένησε με την φούσκα των dot-com (dot-com bubble) το 2000 και 2001. Η Disney έκλεισε το δικό της go.com ενώ το Excite χρεοκόπησε και ότι απέμεινε από αυτό πουλήθηκε στο iWon.com. Παρ' όλ' αυτά κάποιες διαδικτυακές πύλες όπως το Yahoo! Κατάφεραν να βγουν ζωντανές από την φούσκα και συνέχισαν να είναι επιτυχημένες. Ο χρόνος έδωσε την ευκαιρία σε νέες υπηρεσίες και κατ' επέκταση νέες διαδικτυακές πύλες (π.χ. Το worldpress κλπ) να κάνουν την εμφάνισή τους.

Οι επιχειρήσεις πια μπορούν να επιλέξουν να αναπτύ-





Ιστορική Αναδρομή

ξουν πολλαπλές εταιρικές πύλες βασισμένες σε επιχειρηματική δομή και στρατηγική εστίαση ενώ παράλληλα επαναχρησιμοποιούν αρχιτεκτονικά πλαίσια (architectural frameworks), βιβλιοθήκες συστατικών στοιχείων (component libraries) ή τυποποιημένες μεθοδολογίες έργου (standardized project methodologies). Το 2003, πάροχοι εταιρικών διαδικτυακών πυλών βασισμένων σε Java δημιούργησαν ένα νέο πρότυπο γνωστό ως JSR-168 [40]. Επί της ουσίας αφορούσε μία προγραμματιστική διεπαφή για την διαλειτουργικότητα μεταξύ εταιρικών πυλών και portlets. Οι πάροχοι λογισμικού άρχισαν να παράγουν portlet τα οποία πειθαρχούν στο πρότυπο JSR-168 και τα οποία μπορούν να εγκατασταθούν σε οποιαδήποτε διαδικτυακή πύλη η οποία υπακούει σε αυτό το πρότυπο. Η δεύτερη έκδοση αυτού του προτύπου, το JSR-286, εκδόθηκε στις 12 Ιουνίου του 2008 [30,31].

Σύγκριση

Η συντριπτική πλειονότητα των διαδικτυακών πυλών και συστημάτων διαχείρισης περιεχομένου ανοιχτού κώδικα (open source) είναι γραμμένα σε PHP. Ένα μικρό ποσοστό είναι γραμμένο σε JAVA, ένα μικρότερο σε .NET, ενώ η συνεισφορά των υπόλοιπων γλωσσών είναι σχεδόν αμελητέα (περίπου 1%). Αυτό δεν αποτελεί έκπληξη, καθώς η PHP σχεδιάστηκε για να λύσει το “πρόβλημα του διαδικτύου”, με μία “ανοιχτή” προσέγγιση. Όντας μία γλώσσα προγραμματισμού πολύ εύκολη στην εκμάθηση καθώς και ισχυρότατη όσον αφορά τις δυνατότητές της, η PHP χρησιμοποιείται για την δημιουργία ισχυρών πυρήνων στα συστήματα στο διαδίκτυο, με κομψή λογική (business logic). Όσον αφορά την βάση δεδομένων, στα περισσότερα συστήματα αυτή είναι MySQL, καθώς δένει τέλεια και χωρίς προβλήματα με την PHP και είναι ευρύτατα χρησιμοποιούμενη. Επίσης τα περισσότερα πακέτα hosting έχουν εγκατεστημένο τον Apache και την MySQL. Έτσι, στην περίπτωση που ένας χρήστης επιθυμεί να στήσει ένα σύστημα διαχείρισης περιεχομένου εύκολα, φθηνά και γρήγορα, τότε το δίδυμο PHP-MYSQL αποτελεί μονόδρομο.

Ένα από τα βασικά πλεονεκτήματα της Java έναντι των περισσότερων άλλων γλωσσών είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουν ακριβώς το ίδιο σε Windows, Linux, Unix και Macintosh (σύντομα θα τρέχουν και σε Playstation καθώς και σε άλλες κονσόλες παιχνιδιών) χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Το Liferay βασίζεται στην Java και τα θετικά του απορρέουν από τις δυνατότητες ενότητας και ευελιξίας που έχει, καθώς επίσης και οι απεριόριστες

δυνατότητες του όσον αφορά την επιλογή χρήσης των τελικών τεχνολογιών. Παράδειγμα αποτελεί η ανεξαρτησία βάσης δεδομένων καθώς πρακτικά μπορεί να γίνει επιλογή χρήσης ανάμεσα σε περισσότερες από 10 επιλογές. Όμοια γίνεται και όσον αφορά τον διακομιστή εγκατάστασης (περισσότερες από 15 επιλογές) με αποτέλεσμα οι επιτρεπτοί συνδυασμοί εγκατάστασης να ξεπερνούν τους 200. Η συνοπτική παρουσίαση άλλων πυλών φανερώνει τις ιδιαιτερότητες του Liferay, είτε αυτές είναι οι δυνατότητές του είτε η αδυναμίες του. Έγκειται στην ευχέρεια του τελικού χρήστη να επιλέξει το κατάλληλο σύστημα για τον σκοπό που το χρειάζεται καθώς και την συνιστώμενη παραμετροποίηση του συστήματός του. Εδώ συνοψίζονται μερικά στατιστικά στοιχεία όσον αφορά τις τεχνολογίες υλοποίησης συστημάτων διαχείρισης περιεχομένου.

Εμπορικά Συστήματα:

JAVA	40.1%
ASP.NET	24.1%
PHP	21.8%
.NET	14.1%

Συστήματα Ανοιχτού Κώδικα:

PHP	68%
JAVA	13%
PERL	7%
PYTHON	4%
ASP.NET	3%
ROR	2%
Άλλα	3%

Σύγκριση

Drupal

Το Drupal αποτελεί ένα από τα παλιότερα ανοικτού κώδικα συστήματα διαχείρισης περιεχομένου με εκπληκτική ευρύτητα χρήσης σήμερα – ένα παράδειγμα αποτελεί η ιστοσελίδα της AOL. Το σύστημα είναι γραμμένο σε PHP, με πολύ προσεγμένη αρχιτεκτονική, εξαιρετική ποιότητα κώδικα και σημαντική υποστήριξη από την κοινότητά του (ενεργή). Ενδεχομένως απαιτείται κάποιος χρόνος από τον χρήστη για να συνηθίσει την ορολογία του Drupal, όμως μπορεί να λειτουργήσει σε περιβάλλοντα με εξαιρετικές απαιτήσεις. Θεωρείται ένα από τα πιο ελαφριά συστήματα, το οποίο όμως δεν έχει και το πιο φιλικό γραφικό περιβάλλον για τον τελικό χρήστη. Όσον αφορά τον κώδικα του Drupal, αυτός είναι πολύ εύκολος στην κατανόηση και την επέκταση – μέσω του μηχανισμού hooks. Η ομάδα του Drupal κάνει ότι απαιτείται για να παράγει ποιοτικό κώδικα, κάτι το οποίο βοηθάει τον τελικό προγραμματιστή ο οποίος αναπτύσσει εφαρμογές για αυτό το σύστημα.

Αρχικά γραμμένο από τον Dries Buytaert ως σύστημα πίνακα ανακοινώσεων (*BBS, bulletin board system*), το Drupal μετατράπηκε σε εγχείρημα ανοικτού κώδικα το 2001. *Drupal* είναι η διατύπωση στην Αγγλική γλώσσα της Ολλανδικής λέξης "druppel", που σημαίνει "σταγόνα". Το όνομα πάρθηκε από τον ξεπερασμένο πλέον ιστοχώρο Drop.org, του οποίου ο κώδικας εξελίχθηκε στο Drupal. Ο Buytaert ήθελε να ονομάσει τον ιστότοπο "dorp" (στα Ολλανδικά σημαίνει "χωριό", αναφερόμενος στη διάσταση της κοινότητας), αλλά έκανε

ένα ορθογραφικό λάθος κατά τη διαδικασία ελέγχου του ονόματος χώρου (domain name) και τελικά σκέφτηκε ότι ακούγεται καλύτερα. Από το Μάιο του 2006 ως τον Απρίλιο του 2007, χρήστες κατέβασαν το Drupal από τον επίσημο ιστοχώρο περισσότερες από 600.000 φορές [42]. Μια μεγάλη κοινότητα χρηστών λαμβάνει πλέον μέρος στη συνεχή εξέλιξη του Drupal.

Το Drupal έχει λάβει επαινούς από τους διαχειριστές ιστοσελίδων, σχεδιαστές και προγραμματιστές για τον αρθρωτό σχεδιασμό του, με το βασικό του στρώμα, ή "πυρήνα", να παρέχει τα βασικά χαρακτηριστικά του Drupal στην προεπιλεγμένη εγκατάστασή του και πρόσθετα χαρακτηριστικά λειτουργικότητας και παρουσίασης μπορούν να επεκτείνουν τον πυρήνα με την προσθήκη προσαρτώμενων μονάδων και θεματικών παραλλαγών. Οι μονάδες του Drupal χρησιμοποιούνται για να "υπερβούν" τα ενσωματωμένα χαρακτηριστικά του πυρήνα, επεκτείνοντας έτσι ή και αντικαθιστώντας την εξ' ορισμού συμπεριφορά του Drupal, χωρίς την επέμβαση στον αυτούσιο κώδικα των αρχείων του πυρήνα του Drupal. Η δυνατότητα αυτή της τροποποίησης της λειτουργικότητας του πυρήνα έχει επίπτωση στην προσαρμοστικότητα του Drupal καθώς και στην ασφάλειά του, ειδικότερα σε θέματα ασφαλείας, όπως η έγχυση εντολών SQL (SQL injection). Προσαρμοσμένες θεματικές παραλλαγές, που μπορούν να προστεθούν χωρίς να επηρεάζουν τον πυρήνα του Drupal, χρησιμοποιούν προτυποποιημένες μορφές που μπορούν να δημιουργηθούν

Σύγκριση

Drupal

από μηχανές σχεδίασης θεματικών παραλλαγών τρίτων.

Ορισμένοι χρήστες [41] θεωρούν το Drupal δυσκολότερο στην εκμάθηση και στην εγκατάσταση συγκρινόμενο με άλλες δημοφιλείς εφαρμογές διαχείρισης περιεχομένου. Συγκεκριμένα, η πολυπλοκότητα των επιλογών των ρυθμίσεων και η σπαρτιάτικη εμφάνιση ενός ιστοτόπου του οποίου μόλις ολοκληρώθηκε η εγκατάσταση συχνά έρχονται σε αντιπαράθεση με την απλότητα εργαλείων όπως το WordPress και της πρόσφατα ανανεωμένης εμπειρίας χρήστη του Joomla. Το Drupal 5.0, που ανακοινώθηκε στις 15 Ιανουαρίου 2007, έρχεται πακέτο με ένα εγκαταστάτη διεπαφής ιστοσελίδας, με μια νέα θεματική παραλλαγή και με ένα επανασχεδιασμένο πίνακα διαχείρισης για να απαντήσει εν μέρει σε αυτές τις κριτικές. Το Drupal υποστηρίζει τη χρήση προφίλ εγκαταστάσεων με προκαθορισμένο περιεχόμενο για τον ιστοχώρο, αλλά λίγα έχουν εκδοθεί έως τώρα. Ορισμένοι προγραμματιστές ασκούν κριτική στο Drupal για την αποκλειστική χρήση διαδικαστικής PHP αντί για αντικειμενοστραφή PHP. Το άρθρο Προγραμματισμός του Drupal από μια οπτική αντικειμενοστρέφειας [44] εξηγεί πως αρχές του αντικειμενοστραφούς προγραμματισμού και της έννοιας του προσανατολισμένου προς τα χαρακτηριστικά προγραμματισμού (Aspect-Oriented Programming - AOP) εφαρμόζονται στο Drupal, παρόλο που δεν χρησιμοποιούνται τα αντικειμενοστραφή χαρακτηριστικά της PHP. Η βελτιωμένη υποστήριξη για τον αντικειμενοστραφή προγραμματισμό στην PHP5 υπήρξε ένας από τους λόγους [43] για τα σχέδια του έργου του Drupal να αφαιρέσει την

υποστήριξη για την PHP4 σε μελλοντικές εκδόσεις του λογισμικού. Τέλος, κριτική έχει ασκηθεί για διάφορα χαρακτηριστικά σε επίπεδο τελικού χρήστη τα οποία απουσιάζουν όπως για παράδειγμα ένας συντάκτης WYSIWYG (What You See Is What You Get, WYSIWYG Editor) ένα σύστημα στο οποίο το περιεχόμενο κατά την επεξεργασία παρουσιάζεται ακριβώς ή πολύ κοντά στην τελική έξοδο, στο πως δηλαδή θα φαίνεται το περιεχόμενο τελικά στην ιστοσελίδα.

Σύγκριση

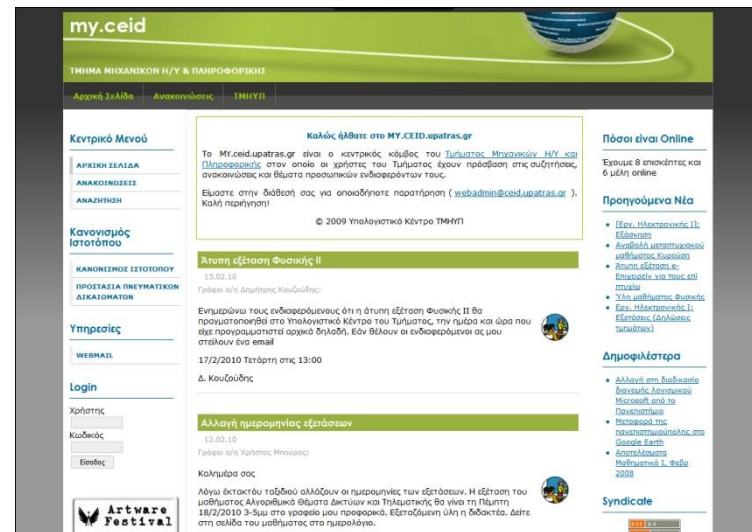
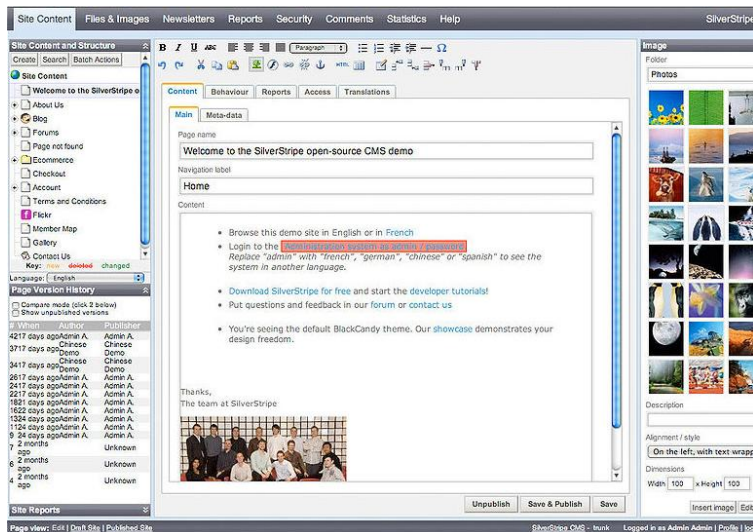
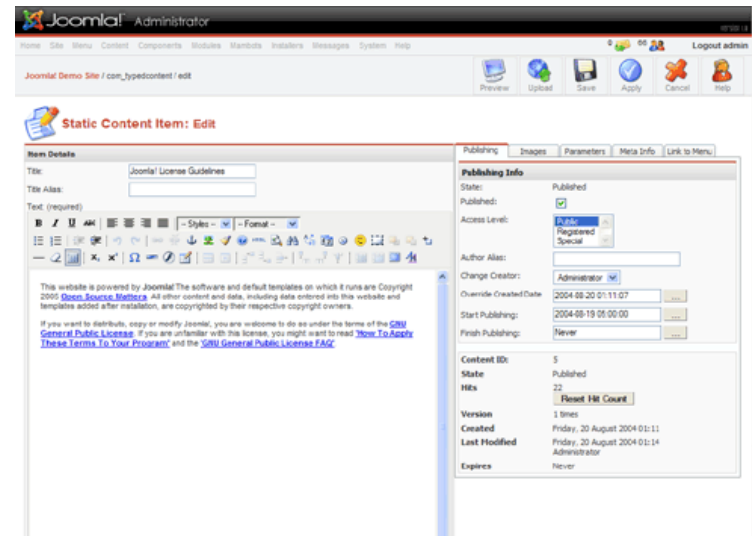
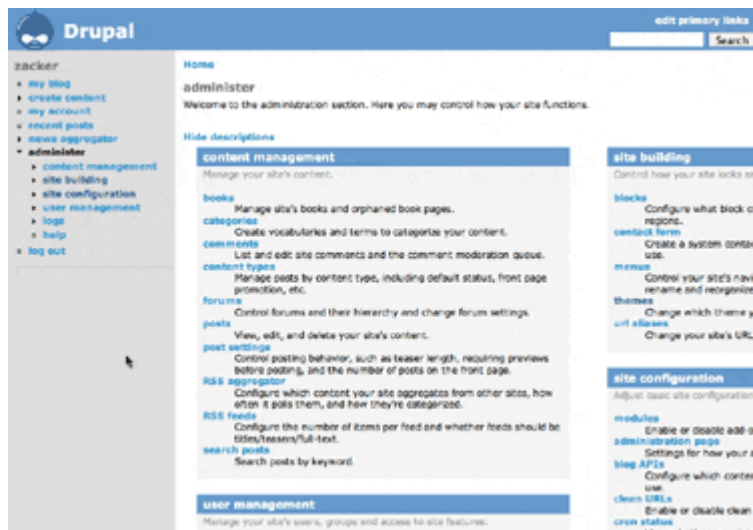
Joomla

Η ποιότητα του κώδικα, αν και δεν είναι εφάμιλλη του Drupal, είναι σε πολύ καλό επίπεδο. Το σύστημα απαιτεί κάποιο χρόνο για την εκμάθησή του καθώς το σύστημα διαχείρισης του πίσω μέρους της αρχιτεκτονικής (back-end) θα μπορούσε να είναι απλούστερο. Η εγκατάστασή του είναι αρκετά απλή καθώς μπορεί είτε να εγκατασταθεί χειροκίνητα από τον πηγαίο κώδικα στον Apache, είτε μέσω κάποιου συστήματος διαχείρισης πακέτων (package management system), είτε μέσω ενός περιβάλλοντος αυτοματοποιημένης εγκατάστασης κατά το οποίο ο χρήστης επιλέγει κάποιες ρυθμίσεις όπως θέματα που αφορούν την βάση δεδομένων κλπ από ένα πάνελ διαχείρισης.

Έχουν περάσει λιγότερα από πέντε χρόνια από τον Αύγουστο του 2005 όποτε γεννήθηκε το Joomla ως μία νέα εκδοχή του συστήματος Mambo. Εκείνη την εποχή τα Mambo άνηκαν στην Miro Ltd η οποία αποτελούσε έναν μη κερδοσκοπικό οργανισμό ο οποίος ανέφερε τους σκοπούς του έργου και είχε ως καθήκον να το προστατεύει νομικά. Η ομάδα ανάπτυξης άρχισε σταδιακά να υποστηρίζει πως το έργο είχε αρχίσει να ξεφεύγει από τον αρχικό του στόχο καθώς και να καταπατά κάποιες αξίες του ανοιχτού λογισμικού. Έτσι, δημιουργήθηκε η διαδικτυακή ιστοσελίδα opensourcematters.com (στα ελληνικά σημαίνει ότι έχει σημασία ο ανοιχτός κώδικας) και παρείχε πληροφορίες στους χρήστες, τους προγραμματιστές, τους σχεδιαστές και γενικότερα την κοινότητα. Μετά από κάποιες δημοσιεύσεις στην ιστοσελίδα του Mambo, τα πράγματα άρχισαν να παίρνουν έκταση και το θέμα

επεκτάθηκε στο πως ορίζεται το ανοιχτό λογισμικό, με πλήθος χρηστών του διαδικτύου να εκφέρουν θετικές ή αρνητικές απόψεις επί του θέματος. Τον Αύγουστο του 2005, η ομάδα ανάπτυξης ζήτησε από την κοινότητα να προτείνει ένα όνομα για το σύστημα, και τον Σεπτέμβριο του ίδιου χρόνου το Joomla ξεκίνησε την πορεία του. Σήμερα, το Joomla έχει μία ενεργή κοινότητα με περισσότερους από 200.000 ανθρώπους που χρησιμοποιούν και αναπτύσσουν το σύστημα, περισσότερες από 4.000 μονάδες επέκτασης και εκπληκτική πληθώρα θεμάτων.

Μερικά σημαντικά θέματα με το Joomla είναι πως, λόγω της ανοιχτής φιλοσοφίας του -η οποία αποτελεί και ακρογωνιαίο λίθο της καμπάνιας του- δεν είναι δυνατή η χρήση εμπορικών επεκτάσεων – από κερδοσκοπικούς δηλαδή οργανισμούς. Επίσης, το σύστημα δεν υποστηρίζει την ύπαρξη διαφορετικού περιεχομένου για διαφορετικά domain names ούτε την χρήση διαφορετικών θεμάτων μέσα στην ίδια εγκατάσταση. Ακόμη, δεν είναι δυνατή η δημιουργία περισσότερων επιπέδων πρόσβασης από τα είδη υπάρχοντα του συστήματος χωρίς την χρήση επεκτάσεων – ένα χαρακτηριστικό το οποίο υπάρχει ενδογενώς σε άλλα συστήματα. Τέλος, ένα σημαντικό χαρακτηριστικό το οποίο απουσιάζει ενδογενώς και έχει δεχθεί αυστηρή κριτική είναι η βελτιστοποίηση για τις μηχανές αναζήτησης (Search Engine Optimization - SEO).



Σύγκριση

Silverstripe

Το Silverstripe είναι ένα ανερχόμενο σύστημα διαχείρισης περιεχομένου το οποίο αξιοποιεί τα πλεονεκτήματα της αντικειμενοστρέφειας της τελευταίας έκδοσης της PHP, της PHP5. Τεχνικά, βασίζεται σε ένα σχεδιαστικό πρότυπο "Μοντέλο-Προβολή - Ελεγκτή" (Model-View-Controller Design pattern, MVC) και χρησιμοποιεί ένα σύστημα αντιστοίχισης ενός αντικειμενοστραφούς μοντέλου (object oriented domain model) σε μια παραδοσιακή συσχετιστική βάση δεδομένων (relational database). Στην ίδια λογική με το Ruby on Rails, το Silverstripe παράγει την περιγραφή (markup) μέσω μιας διαμορφωμένης γλώσσας πατρών (template) με την χρήση του Sapphire.

Όσον αφορά το επίπεδο χρήση, η επεξεργασία περιεχομένου είναι εξαιρετικά απλή καθώς ο χρήστης ξεκινά με την οθόνη διαχείρισης περιεχομένου, και δεν είναι απαιτείται μία πολύπλοκη διαδικασία για να φτάσει κάποιος εκεί. Έχει ένα πολύ εύκολο σύστημα θεμάτων -το οποίο δυστυχώς σε πληθώρα δεν φτάνει τους αριθμούς του Joomla- και εκτενή τεκμηρίωση τόσο σε επίπεδο χρήστη όσο και σε επίπεδο προγραμματιστή. Όλο το περιεχόμενο είναι διαθέσιμο στα αριστερά, καθώς η επιλογή του εκάστοτε αντικειμένου δεν φορτώνει ολόκληρη την σελίδα αλλά ένα μέρος της στα δεξιά το οποίο δείχνει και το περιεχόμενο. Σημαντικό χαρακτηριστικό είναι ότι παρέχει ενσωματωμένα εργαλεία διαχείρισης εικόνων, που επιτρέπει στον χρήστη για παράδειγμα να κόψει εικόνες ή να αλλάξει το μέγεθός τους. Τέλος, προσφέρει αναλυτικές αναφορές και στατιστικά στοιχεία σχετικά με την επισκεψιμότητα της κάθε ιστοσελίδας.

Στο προηγούμενο δισέλιδο παρουσιάζεται αρχικά η εικόνα διαχείρισης του Drupal, του Joomla και του Silverstripe, καθώς και μία τυπική εικόνα χρήστη του Joomla, καθώς χρησιμοποιείται και για την διαδικτυακή πύλη της σχολής μας από τις αρχές του 2008. (<http://my.ceid.upatras.gr>) - προηγούμενα, το σύστημα ήταν χτισμένο στο PHPNuke. Στην επόμενη σελίδα, παρουσιάζεται μία τυπική εικόνα διαχείρισης περιεχομένου από το Plone, και δεξιά επάνω είναι η σελίδα επίδειξης των χαρακτηριστικών του DotNetNuke και από κάτω η αντίστοιχη του DotCMS.

Σύγκριση

DotNetNuke

Το DotNetNuke είναι η πιο συχνά χρησιμοποιούμενη πλατφόρμα διαχείρισης περιεχομένου ανοιχτού κώδικα γραμμένη σε τεχνολογίες .NET της Microsoft. Είναι γραμμένη σε VB.NET, παρέχεται τόσο σε ελεύθερη όσο και σε εμπορική άδεια, απαρτίζεται από ένα ολοένα και αυξανόμενο οικοσύστημα προγραμματιστών τόσο από την κοινότητα όσο και από ξένους προς την κοινότητα φορείς και είναι παραμετροποιήσιμο μέσω θεμάτων και επεκτάσεων. Αυτό έχει ως αποτέλεσμα την ύπαρξη τόσο ελεύθερων όσο και εμπορικών επεκτάσεων, με αρκετές εταιρείες να απασχολούνται με την ανάπτυξη επαγγελματικών θεμάτων για το DotNetNuke. Η αρχιτεκτονική του πυρήνα βασίζεται σε ένα μοντέλο τριών επιπέδων. Απαιτεί τον IIS 6, ASP.NET 2.0 μέχρι 3.5 και από βάσεις δεδομένων υποστηρίζει μόνο τον SQL Server 2005 και 2008, κάτι το οποίο μπορεί να αποτελεί περιοριστικό παράγοντα σε συγκεκριμένα περιβάλλοντα.

Η ιστορία του DotNetNuke ξεκινάει λίγο μετά την αλλαγή της χιλιετίας, όταν μία άλλη εφαρμογή, το iBuySpy Workshop, είχε δημιουργηθεί ως μία βελτίωση στην διαδικτυακή πύλη iBuySpy η οποία ξεκίνησε ως δείγμα εφαρμογής σε .NET. Οι αρχικές εκδόσεις του DotNetNuke εκδόθηκαν από τον δημιουργό της συγκεκριμένης βελτίωσης και αργότερα άρχισε να λαμβάνει ενεργό μέρος η κοινότητα ανοιχτού κώδικα. Όπως είναι προφανές, το όνομα δημιουργήθηκε ως μία ένωση των λέξεων .NET και nuke, ένα γνωστό επίθεμα σε ήδη υπάρχοντα συστήματα όπως το PHP-Nuke και το PostNuke. Το 2006 δημιουργήθηκε μία

εταιρεία για την υποστήριξη του DotNetNuke και λίγο αργότερα έκαναν την εμφάνισή τους οι άλλες δύο εκδόσεις του συστήματος (professional και elite).

Ένα κοινό παράπονο είναι πως οι οδηγίες για την δημιουργία σωστά πακεταρισμένων μονάδων επέκτασης είναι βραδυκίνητες και απαιτούν προσκόλληση σε μία συγκεκριμένη ταξινόμια. Επίσης, όσον αφορά την τεκμηρίωση της προγραμματιστικής διεπαφής, αυτή είναι γραμμένη σε μορφή προσανατολισμένη προς το έργο. Ένα σημαντικό αρνητικό είναι πως οι καινούριες εκδόσεις του DotNetNuke δεν είναι πλήρως προς τα πίσω συμβατές με συγκεκριμένες επεκτάσεις, με αποτέλεσμα πολλές μονάδες να μην λειτουργούν όπως θα έπρεπε. Επίσης, καθώς η πλατφόρμα δέχεται συνεχείς ενημερώσεις, αντίθετα με την επαγγελματική έκδοση, η έκδοση της κοινότητας δεν έχει δοκιμαστεί εκτενώς ούτε έχει πιστοποιηθεί από την εταιρεία. Τέλος, η πλατφόρμα δεν υποστηρίζει ενδογενώς μεταφράσεις, ένα πολύ σημαντικό πρόβλημα για μία ιστοσελίδα που διαθέτει περιεχόμενο σε άλλη γλώσσα από τα αγγλικά. Παρ' όλ' αυτά μέσα από επεκτάσεις τρίτων είναι δυνατή η προσθήκη γλωσσών ως ένα βαθμό.

Σύγκριση

Plone και DotCMS

Το Plone είναι ένα πολύ δυνατό σύστημα διαχείρισης περιεχομένου χτισμένο σε Python, και πατάει πάνω στον εξυπηρετητή εφαρμογών Zope. Είναι σημαντικό το γεγονός ότι εκδίδεται με άδεια χρήσης την Γενική Άδεια Χρήσης GNU. Όπως αναφέρεται και στην σελίδα του έργου, 73% είναι σε Python, 15% σε XML, 6% σε Javascript, και 6% σε άλλες γλώσσες, συμπεριλαμβανομένων των Perl, PHP, AWK κλπ. Τα δεδομένα αποθηκεύονται σε μία εσωτερική βάση δεδομένων του εξυπηρετητή εφαρμογών, την ZODB. Το σύστημα υποστηρίζει περισσότερες από 35 γλώσσες, μία πληθώρα θεμάτων και επεκτάσεων καθώς και την χρήση από άτομα με ειδικές ανάγκες.

Όσον αφορά τα χαρακτηριστικά του, το Plone υποστηρίζει τα πρότυπα περισσότερο από κάθε άλλο σύστημα διαχείρισης περιεχομένου. Είναι ανεξάρτητο της πλατφόρμας εγκατάστασης (platform in-dependent), καθώς βασίζεται σε τεχνολογίες αυτής της μορφής (cross-platform). Σημαντικό είναι το γεγονός πως υποστηρίζει άτομα με ειδικές ανάγκες. Ακόμη κατέχει το ρεκόρ όσον αφορά την ασφάλεια σε σχέση με άλλα συστήματα διαχείρισης περιεχομένου. Ένα σημαντικό αρνητικό είναι πως είναι δεσμευτικό όσον αφορά την βάση δεδομένων, ένα χαρακτηριστικό στο οποίο το Plone είναι γνωστό ευρέως πως υστερεί. Επίσης, λόγω του ότι βασίζεται σε Python, ο χρόνος που χρειάζεται κάποιος για να αναπτύξει πακέτα στο Plone είναι σημαντικός. Αν και η Python αποτελεί εύκολη γλώσσα στην εκμάθησή, ο συνδυασμός Plone και Zope μπορεί να απαιτήσει ακόμη και μερικές εβδομάδες πριν να είναι κάποιος έτοιμος να αναπτύξει εφαρμογές, ακόμα

και αν έχει προηγούμενη εμπειρία στον προγραμματισμό.

Τέλος, το DotCMS αρχικά αναπτύχθηκε ως μία Java εναλλακτική επιλογή από την μία στα ήδη υπάρχοντα PHP συστήματα στην αγορά και από την άλλη στις ήδη υπάρχουσες υψηλού κόστους εταιρικές λύσεις. Το σύστημα γεννήθηκε το 2005 από τον πηγαίο κώδικα του Liferay, και αυτό για να είναι συμβατό με το πρότυπο JSR-168 για τις διαδικτυακές πύλες. Μέσα στο 2009 παρουσίασε μία νέα αρχιτεκτονική επεκτάσεων, καθώς και ένα νέο πρόγραμμα εγκατάστασης για Windows. Οι τεχνολογίες του αποτελούν ένα γνήσιο υποσύνολο των τεχνολογιών που υποστηρίζει το Liferay. Πρόσφατα ανακοινώθηκε επίσημα ότι στην νέα έκδοση θα υποστηρίζονται υπηρεσίες διαλειτουργικότητας διαχείρισης περιεχομένου (Content Management Interoperability Services, CMIS), ένα νέο πρότυπο το οποίο υλοποιείται πρώτη φορά από σύστημα διαχείριση περιεχομένου.

Στις επόμενες σελίδες παρουσιάζεται ένα λεπτομερές συγκριτικό μεταξύ των παραπάνω συστημάτων διαχείρισης περιεχομένου και διαδικτυακών πυλών, συμπεριλαμβανομένων και άλλων κορυφαίων συστημάτων τέτοιου είδους μεταξύ αυτών και του Liferay. Τα αρχικά δεδομένα οφείλονται στο [45], αν και έχει γίνει αρκετή δουλειά για την μετατροπή τους και την τελική παρουσίαση.

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI 7.7
System Requirement									
Application Server	J2EE	IIS/.Net	Apache	CGI	J2EE	Zope	Apache	Apache	Apache
Database	Other	MSSQL	MySQL	MySQL	DB2	Other	MySQL	MySQL	MySQL
License	Open Source	Open Source	Open Source	Open Source	Closed Source	Open Source	Closed Source	Open Source	Open Source
Operating System	Platform Independent	Windows Only	Platform Independent	Platform Independent	Platform Independent	Platform Independent	*nix Only	Platform Independent	Platform Independent
Programming Language	Java	VB	PHP	PHP	Java	Python	PHP	PHP	Perl
Root Access	✗	✗	✗	✗	✓	✗	✓	✗	✓
Shell Access	✓	✗	✗	✗	✓	✓	✓	✗	✓
Web Server	Other	IIS	Apache	Apache	Other	Apache	Apache	Apache	Apache
Security									
Audit Trail	✓	—	✓	✗	✓	✓	—	✓	✓
Captcha	✓	✓	✓	✓	✓	✓	✗	✓	✓
Content Approval	✓	—	✓	✓	✓	✓	✓	✓	✓
Email Verification	✓	✓	✓	✓	✓	✓	✓	✓	✓
Granular Privileges	✓	✓	✓	✗	✓	✓	—	✓	✓
Kerberos Authentication	+	✗	✗	✗	✓	✓	✗	✓	✗
LDAP Authentication	✓	✗	✓	✓	✓	✓	✗	✓	✓
Login History	✓	✓	✓	✓	✓	✓	✓	✓	✓
NIS Authentication	+	✗	✗	✗	✗	✓	✗	✓	✗
NTLM Authentication	+	✓	✓	✗	✓	✓	✗	✓	✗
Pluggable Authentication	✓	✓	✓	✗	✗	✓	✗	✓	✓
Problem Notification	✓	✓	✗	✗	✗	✓	✓	✓	✓
Sandbox	✓	✓	✗	✗	✓	✓	✓	✓	✓
Session Management	✓	✓	✓	✓	✓	✓	✗	✓	✓
SMB Authentication	✓	✗	✗	✗	✗	✓	✗	✓	✗
SSL Authentication	✓	✓	✓	✓	✓	✓	✓	✓	✓
SSL Logins	✓	✓	✗	✓	✓	✓	✓	✓	✓
SSL Pages	✓	✓	✗	✓	✓	✗	✓	✓	✓
Versioning	✓	—	✓	✓	✓	✓	—	✓	✓
Support									
Certification Program	✓	✓	✗	✗	✓	✗	✗	✓	✗
Commercial Manuals	✓	✓	✓	✓	✓	✓	✓	✓	✓
Commercial Support	✓	✓	✓	✓	✓	✓	✓	✓	✓
Commercial Training	✓	✓	✓	✓	✓	✓	✓	✓	✓
Developer Community	✓	✓	✓	✓	✓	✓	✗	✓	✓
Online Help	✓	✓	✓	✓	✓	✓	✓	✓	✓
Pluggable API	✓	✓	✓	✓	✓	✓	—	✓	✓
Professional Hosting	✓	✓	✓	✓	✓	✓	✓	✓	✓
Professional Services	✓	✓	✓	✓	✓	✓	✓	✓	✓
Public Forum	✓	✓	✓	✓	✓	✓	✗	✓	✓
Public Mailing List	✓	✓	✓	✗	✓	✓	✗	✓	✓
Third-Party Developer	✓	✓	✓	✓	✓	✓	✗	✓	✓
Users Conference	✓	✓	✓	✓	✓	✓	✗	✓	✓

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI 7.7
Ease of Use									
Drag-N-Drop Content	✓	✓	✓	✗	✓	✓	✓	✓	✓
Email To Discussion	+	+	✓	✓	✓	✓	✗	✓	✓
Friendly URLs	✓	✓	✓	✓	✓	✓	✓	✓	✓
Image Resizing	✓	✓	✓	✓	✓	✓	✓	✓	✓
Macro Language	✓	✓	✓	✓	✓	✓	✓	✓	✓
Mass Upload	✓	✓	✓	✓	✓	✓	✓	✓	✓
Prototyping	✓	✓	—	✓	✓	✓	✓	✓	✓
Server Page Language	✓	✓	✓	✓	✓	✓	✓	✓	✓
Spell Checker	✓	✓	✓	✗	✓	✓	✓	✓	✓
Subscriptions	✓	✓	✓	+	✓	✓	+	✓	✓
Template Language	✓	✓	—	✓	✓	✓	✓	✓	✓
UI Levels	✓	✓	✗	✓	✓	✓	✓	✓	✓
Undo	✓	—	—	✗	—	✓	✓	✓	✓
WYSIWYG Editor	✓	✓	✓	✓	✓	✓	✓	✓	✓
Performance									
Advanced Caching	✓	✓	✓	✓	✓	✓	✓	✓	✓
Database Replication	✓	✗	—	✗	✓	✓	✗	✓	✓
Load Balancing	✓	✓	✓	✓	✓	✓	✗	✓	✓
Page Caching	✓	✓	✓	✓	✓	✓	✗	✓	✓
Static Content Export	✓	✓	✗	✗	✓	✓	✗	✓	✓
Management									
Advertising Management	✓	✓	✓	✓	✓	✓	✓	✓	✓
Asset Management	✓	✓	✓	✓	✓	✓	✓	✓	✓
Clipboard	✓	✓	✗	✗	✗	✓	✓	✓	✓
Content Scheduling	✓	✓	✓	✗	✓	✓	✓	✓	✓
Content Staging	✓	—	✓	✗	✓	✓	✓	✓	—
Inline Administration	✓	✓	✓	✓	✓	✓	✗	✓	✓
Online Administration	✓	✓	✓	✓	✓	✓	✓	✓	✓
Package Deployment	✓	✓	✗	✗	✓	✓	—	✓	✓
Sub-sites / Roots	✓	✓	✓	✓	✓	✓	—	✓	✓
Themes / Skins	✓	✓	✓	✓	✓	✓	—	✓	✓
Trash	✓	✓	✗	✓	✗	✓	✓	✓	✓
Web Statistics	✓	✓	✓	✓	—	✓	✓	✓	✓
Web-based Templ Management	✓	✓	✓	✓	✓	✓	+	✓	✓
Web-based Translation	✓	✓	✓	✓	✓	✓	+	✓	✓
Workflow Engine	✓	—	—	✗	—	✓	+	—	✓

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI 7.7
Flexibility									
CGI-mode Support	✗	✗	✓	✓	✗	✓	—	✓	✗
Content Reuse	✓	✓	—	✓	✓	✓	✓	✓	✓
Extensible User Profiles	✓	✓	✓	✓	—	✓	✗	✓	✓
Interface Localization	✓	✓	✓	✓	✓	✓	✗	✓	✓
Metadata	✓	✓	✓	✓	✓	✓	✓	✓	✓
Multi-lingual Content	✓	🔑	✓	🔑	✓	✓	+	✓	✓
Multi-lingual Content Integration	✓	🔑	🔑	🔑	✓	✓	+	✓	✗
Multi-Site Deployment	✓	✓	✓	✓	✓	✓	✗	✓	✓
URL Rewriting	✓	✓	✓	✓	✓	✓	✓	✓	✓

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI 7.7
Built-in Applications									
Blog	✓	✓	✓	✓	✓	✓	+	🔑	✓
Chat	+	✓	🔑	🔑	✓	🔑	✗	🔑	🔑
Classifieds	✓	✓	🔑	🔑	✗	🔑	✗	🔑	✓
Contact Management	✓	✓	🔑	🔑	✓	🔑	✗	🔑	✓
Data Entry	✓	✓	🔑	🔑	✗	✓	✓	🔑	✓
Database Reports	✓	✓	✗	🔑	✗	✓	✓	🔑	✓
Discussion / Forum	+	✓	✓	🔑	✓	🔑	+	🔑	✓
Document Management	✓	✓	—	🔑	✓	✓	+	🔑	—
Events Calendar	✓	✓	🔑	🔑	✓	✓	+	🔑	✓
Expense Reports	+	+	✗	🔑	—	🔑	✗	🔑	✗
FAQ Management	✓	✓	✓	✓	—	🔑	✓	🔑	✓
File Distribution	✓	✓	🔑	🔑	✓	✓	+	🔑	✓
Graphs and Charts	✓	✓	✗	🔑	🔑	✓	—	🔑	✓
Groupware	✓	+	🔑	🔑	✓	🔑	✗	🔑	✗
Guest Book	✓	✓	🔑	🔑	✓	🔑	+	🔑	✓
Help Desk / Bug Reporting	+	+	🔑	🔑	+	🔑	✓	🔑	✓
HTTP Proxy	✓	✓	✗	✗	✓	🔑	✗	🔑	✓
In/Out Board	+	+	✗	✗	✗	🔑	✓	✗	✓
Job Postings	✓	+	🔑	🔑	✗	✓	✓	🔑	✓
Link Management	✓	✓	🔑	✓	✓	✓	—	🔑	✓
Mail Form	✓	✓	🔑	✓	✓	✓	✗	✓	✓
My Page / Dashboard	✓	✓	🔑	✗	✓	✓	✗	🔑	✓
Newsletter	✓	✓	🔑	🔑	—	✓	✓	🔑	✓
Photo Gallery	✓	✓	🔑	🔑	✓	✓	—	🔑	✓
Polls	✓	✓	✓	✓	✓	✓	✗	🔑	✓
Product Management	✓	✓	🔑	✓	—	✓	+	🔑	✓
Project Tracking	+	+	🔑	🔑	—	✓	✓	🔑	✓
Search Engine	✓	✓	✓	✓	—	✓	✓	✓	✓
Site Map	✓	✓	🔑	🔑	✓	✓	✓	✓	✓
Surveys	🔑	—	🔑	🔑	✓	✓	+	🔑	✓
Syndicated Content (RSS)	✓	✓	✓	✓	✓	✓	—	✓	✓
Tests / Quizzes	+	🔑	🔑	🔑	✗	🔑	—	🔑	✓
Time Tracking	+	+	🔑	✗	+	🔑	✗	🔑	✓
User Contributions	✓	✓	✓	✓	✓	✓	+	✓	✓
Web Services Front End	✓	🔑	—	✓	✓	✗	✗	🔑	✓
Wiki	—	✓	🔑	🔑	✓	🔑	✗	🔑	✓

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI
Interoperability									
Content Syndication (RSS)	✓	✓	✓	✓	✓	✓	—	✓	✓
FTP Support	+	✓	—	✓	—	✓	✓	✓	✗
UTF-8 Support	✓	✓	✓	✓	✓	✓	—	✓	✓
WAI Compliant	+	—	—	✗	—	✓	✓	🔑	✓
WebDAV Support	✓	+	✗	✗	✓	✓	✓	🔑	✗
XHTML Compliant	✓	✓	✓	✗	✓	✓	✓	✓	✓

	dotCMS 1.7a	DotNetNuke 5.0.0	Drupal 6.10	Joomla! 1.5.10	Liferay Portal 5.2	Plone 3.0	SilverStripe 1.3	TYPO3 4.2.9	WebGUI
Commerce									
Affiliate Tracking	+	✓	🔑	🔑	✗	✗	✗	🔑	✗
Inventory Management	✓	+	🔑	🔑	✓	🔑	✗	🔑	✓
Pluggable Payments	✓	✓	🔑	🔑	—	🔑	—	🔑	✓
Pluggable Shipping	✓	+	🔑	🔑	—	🔑	✗	🔑	✓
Pluggable Tax	✓	+	🔑	🔑	—	🔑	+	🔑	✓
Point of Sale	+	+	✗	🔑	✓	🔑	✗	✗	✓
Shopping Cart	✓	✓	🔑	🔑	✓	🔑	+	🔑	✓
Subscriptions	+	✓	🔑	🔑	✗	🔑	✗	✗	✓
Wish Lists	+	✓	🔑	🔑	✗	✗	✓	🔑	✗

Χαρακτηριστικά

70

Το Liferay είναι ένα από τα πιο δημοφιλή προϊόντα στην αγορά των διαδικτυακών πυλών ανοικτού κώδικα σήμερα. Έχει λάβει πληθώρα βραβείων από κορυφαία έντυπα της βιομηχανίας, και έχει ένα εντυπωσιακό ποσοστό λήψεων - πάνω από 60.000 λήψεις το μήνα και συνολικά πάνω από ένα εκατομμύριο. Επί της ουσίας έχει έτοιμες όλες τις λειτουργίες που κάποιος μπορεί να χρειάζεται για να δημιουργήσει έναν επιτυχημένο ιστοχώρο, ιδιαίτερα αν πρόκειται για μεγάλες ιστοσελίδες, ιστοσελίδες οργανισμών, κυβερνητικών φορέων, βιομηχανιών, ενδοδικτύων (intranet) ή οποιονδήποτε συνδυασμό αυτών.

Χτισμένο πάνω στις τελευταίες Java και Web 2.0 τεχνολογίες, μερικά από τα χαρακτηριστικά του είναι ότι:

- Τρέχει πάνω σε όλους τους δημοφιλείς εξυπηρετητές (application servers), βάσεις δεδομένων και λειτουργικά συστήματα με περισσότερους από 700 συνδυασμούς εγκατάστασης.
- Είναι συμβατό με το πρότυπο JSR-286.
- Είναι άμεσα έτοιμο προς χρήση και προπακεταρισμένο με περισσότερα από εξήντα portlets.
- Εμπεριέχει Σύστημα Διαχείρισης Περιεχομένου (Content Management Systems - CMS) και Σουίτα Συνεργασίας (Collaboration Suite) μεταξύ των χρηστών.
- Έχει εξατομικευμένες σελίδες για όλους τους χρήστες.

- Έχει αναγνωριστεί με την χρήση μετροπρογραμμάτων ως μία από τις πιο ασφαλείς πλατφόρμες.

Έχοντας χτιστεί για την επιχείρηση, το Liferay παρέχει μηχανισμούς όπου οι χρήστες μπορούν να δουλεύουν κεντριοποιημένα, κατανεμημένα και συνεργατικά. Με κύριο στόχο τον τελικό χρήστη, η διεπαφή του έχει γίνει αρκετά εύκολη, έτσι ώστε ακόμη και τεχνικά μη καταρτισμένοι χρήστες να μπορούν να το χρησιμοποιήσουν. Με ένα συνεχώς αναπτυσσόμενο σύνολο χαρακτηριστικών, παραμένει ανάμεσα στις πιο δημοφιλείς επιλογές για την ενίσχυση της απόδοσης μιας επιχείρησης διατηρώντας τον απαιτούμενο χρόνο και προσπάθεια στο ελάχιστο.

Παρακάτω συνοψίζονται μερικά από τα βασικά χαρακτηριστικά του Liferay.

Εξατομίκευση και εύκολη προσαρμογή: Με μεγάλη ευκολία μπορεί κάποιος να δώσει στους σωστούς ανθρώπους τη σωστή πρόσβαση στις σωστές εφαρμογές και τα έγγραφα. Ένα ιδιαίτερα κλιμακούμενο σύστημα αδειών (permission system) επιτρέπει στον διαχειριστή να προσαρμόσει την εμπειρία χρηστών σε πολλά επίπεδα.

Προσαρμοστικότητα ανάλογα με την ροή: Η τεχνολογία στην οποία είναι χτισμένο το Liferay προσαρμόζεται γρήγορα στις επιχειρηματικές και τις οργανωτικές αλλαγές, κάτι το οποίο εξασφαλίζει μηδανινό χρόνο

Χαρακτηριστικά

διακοπής στη σημερινή, ταχύτατα εξελισσόμενη αγο-
ρά.

Φιλικότητα προς τις μάρκες: Με την χρήση αποδοτι-
κών συστημάτων κώδικα, συγκεκριμένων παραμετρο-
ποιήσεων και ευκολιών στην αλλαγή θεμάτων και
λογοτύπων, το Liferay προσαρμόζεται πολύ εύκολα
στην εμπορική γραμμή που χαράσσουν προϊόντα,
εταιρίες και λογότυπα.

Βραβευμένη διεπαφή χρήστη: Το Liferay διέπεται από
δυναμικά, αισθητικά και φιλικά προς τον χρήστη χα-
ρακτηριστικά τα οποία ενθαρρύνουν τους χρήστες να
υιοθετήσουν το σύστημα για τις ανάγκες τους και
εντείνουν την αλληλεπίδραση μεταξύ ανθρώπου και
υπολογιστή. Συγκεκριμένα, με καινοτομίες όπως η
εγκατάσταση μονάδων με πιάσιμο και σύρσιμο με το
ποντίκι (drag 'n' drop) ακόμη και χρήστες οι οποίοι δεν
έχουν τεχνική κατάρτιση μπορούν να το χρησιμοποιή-
σουν αποδοτικά. Αλλαγές σε ολόκληρη την εμφάνιση
γίνονται μόνο με ένα κλικ, χωρίς να χρειαστεί κάποιος
να βουτήξει στον κώδικα. Η διαχείριση αρχείων γίνε-
ται ακριβώς όπως στην επιφάνεια εργασίας με αντί-

στοιχο σύστημα φακέλων, με αλληλεπικαλύψεις και
σύρσιμο των παραθύρων (drag 'n' drop), των μενού
και άλλων μονάδων και την τελική αίσθηση να είναι
αντίστοιχη ενός λειτουργικού συστήματος –με την
διαφορά ότι όλο αυτό εκτελείται σε φυλλομετρητή
ιστού(Web OS).

Στους παραπάνω πίνακες παρουσιάζονται συνοπτικά
μερικά από τα σημαντικότερα χαρακτηριστικά του
Liferay. Από κάτω δεξιά και με την φορά του ρολογιού,
στον Πίνακα 1 (Deployment Compatibility) παρου-
σιάζονται όλοι οι πιθανοί τρόποι εγκατάστασης του
Liferay (Λειτουργικά Συστήματα, Διακομιστές Εφαρ-
μογών, Βάσεις Δεδομένων κλπ), στον Πίνακα 2 (Portlet
Containers) παρουσιάζονται μονολεκτικά όλες οι τεχ-

νικές λεπτομέρειες και προδιαγραφές της Πύλης (Τεχ-
νολογίες Χρήσης, Πρότυπα, Αρχιτεκτονική κλπ), στον
Πίνακα 3 παρουσιάζονται πληροφορίες διαχείρισης
περιεχομένου (Βιβλιοθήκη Αρχείων, Θέματα, Γλώσ-
σες κλπ) ενώ στον Πίνακα 4 παρουσιάζεται το μεγα-
λύτερο μέρος των εφαρμογών (Πίνακες Μηνυμάτων,
Άμεσα Μηνύματα, Ημερολόγιο, Ηλεκτρονικό Ταχυδ-
ρομείο κλπ)

Content Management

Document Library

- JSR-170 compliant Java Content Repository
- Check in/check out
- Integration with Microsoft Office®
- Versioning
- Workflow
- Fine grained permissioning
- Multi-file uploads
- File format conversion among common formats like Microsoft Office, PDF, TXT and HTML (import and export)
- WebDAV enabled
- Image gallery

Themes & Layout

- User group page templates
- Automatic page structure inheritance
- jQuery standardized
- Semantic & nonobtrusive markup
- LAR import/export of themes
- Hot-deployable

Language Support

- I18N support for any language
- Ships with default translations for 22 languages.

Web & Workspace Publishing

- Rich (WYSIWYG) text editors
- Separation of content from layout
- Reusable content
- CSS
- Logical content templates using either Velocity or XSL
- Bookmarks
- Dynamically generated site taxonomy
- Drag and droppable site map
- Sitemaps protocol support
- Search Engine Optimization (SEO)
- OpenSearch
- Live page editing
- Staging and scheduling to remote servers
- Friendly URLs
- Meta tagging

Knowledge Management

- Bootstrap knowledge base via Docbook file import
- Taxonomy (categorization) framework
- Framework to add taxonomies to custom portlets
- Journal, Wiki, Knowledge Base Taxonomy-enabled
- Category-based navigation

Portal Details

Technologies Used

- AJAX
- Apache ServiceMix
- Docbook
- ehcache
- Groovy
- Hibernate
- ICEfaces
- Java J2EE/JEE
- JBPM
- JGroups
- jQuery JavaScript Framework
- Lucene
- MuleSource ESB
- PHP
- Ruby
- Seam
- Spring & AOP
- Struts & Tiles
- Tapestry
- Velocity

Standards

- AJAX
- iCalendar & Microformat
- JSR-168
- JSR-127
- JSR-170
- JSR-286 (Portlet 2.0)*
- JSF-314 (JSF 2.0)
- OpenSearch

Web Services

- JSON
- Hessian
- Burlap
- REST
- RMI
- Spring HTTP
- WSRP (full support for 1.0 and 2.0)
- WebDAV

Architecture

- Hierarchical and extensible system of communities and organizations
- Mule or ServiceMix ESB
- Single Sign On (Yale CAS and Siteminder out of the box) Also supports JAAS, LDAP, Netegrity, Microsoft Exchange
- Security (DES, MD5, RSA)

Performance & Scalability

- Clustering at any combination of tiers (presentation, service, business logic, and database)
- Terracotta DSO integration
- Advanced Caching
- Page Caching
- Load Balancing
- Session Replication
- Distributed Cache
- Static Content Export

Security

- Liferay Portal uses industry standard, government-grade encryption technologies including advanced algorithms such as DES, MD5 and RSA. Liferay was benchmarked as among the most secure portal platforms using LogicLibrary's Logiscan suite.
- Pluggable Authentication
 - Email Verification
 - Granular Permissioning
 - LDAP Authentication
 - Session Management

Collaboration & Social

Wikis

- Rich text editor (WYSIWYG)
- Versioning & reversion
- Creole syntax support
- File attachments
- Threaded comments
- Recent changes tracking
- LDAP integration
- Tag-based navigation
- RSS

Blogs

- Rich Text Editor (WYSIWYG)
- Social bookmarking
- Threaded comments
- Tags and Labels
- Social Bookmarking
- Rating system
- Trackback URLs
- RSS

Message Boards

- Rich text editor (WYSIWYG)
- Role-based permissioning
- User data display
- Threaded comments and categories management
- Versioning
- Statistics
- Recent posts
- Email based subscriptions
- RSS
- Presence

Presence

- AJAX chat client
- Dynamic friend list
- Activity wall
- Activity tracker with Activity Tracker API
- Mashups & Integration

Mashups & Integration

- User location map
- Facebook
- OpenSocial
- iGoogle / Google Gadget

Instant Messaging

- AJAX-based IM client
- User profile pictures
- User controlled status updates
- Jabber protocol
- Chat sessions persist across all pages

Calendar

- AJAX-based interface
- iCal and Microformats support
- Task lists for event creation, management, and search
- Events sharing across communities
- Event reminders via email, IM, or SMS

Webmail Client

- Rich text editor (WYSIWYG)
- AJAX-based interface
- Multiple email accounts
- Supports IMAP and SMTP
- Custom preconfigured accounts

Alerts & Announcements

- AJAX-based
- Targeted group delivery
- Remote delivery (email, SMS)

Deployment Compatibility

Operating Systems

- Linux (CentOS, RHES, SUSE, Ubuntu, and others)
- Unix (AIX, HP-UX, Mac OS X, Solaris, and others)
- Windows

Servlet Containers

- Jetty
- Resin
- Tomcat

Application Servers

- Apache Geronimo
- Sun GlassFish 2 UR1
- JBoss
- JOnAS
- OracleAS
- SUN JSAS
- WebLogic
- WebSphere

Databases

- Apache Derby
- IBM DB2
- Firebird
- Hypersonic
- InterBase
- JDataStore
- MySQL
- Oracle
- PostgreSQL
- SAP MaxDB
- SQL Server
- Sybase

Επίπεδο Χρήστη

Σε προηγούμενη ενότητα παρουσιάστηκαν πολύ συνοπτικά κάποια χαρακτηριστικά του Liferay. Στην συγκεκριμένη ενότητα αιτιολογείται το γιατί έγινε η επιλογή του συγκεκριμένου συστήματος καθώς και γιατί θεωρείται ένα τόσο πλήρες σύστημα σε επίπεδο χρήστη.

Έτοιμα Εργαλεία: Το Liferay παρέχει με την εγκατάσταση περισσότερες εφαρμογές από κάθε άλλη διαδικτυακή πύλη στην αγορά. Με περισσότερα από εξήντα portlet, ο χρήστης μπορεί να παραμετροποιήσει σε πολύ μεγάλο βαθμό το περιβάλλον εργασίας του. Περιλαμβάνονται το σύστημα διαχείρισης περιεχομένου και η πλατφόρμα συνεργασίας προσφέροντας δημοσίευση πληροφορίας στον ιστό, διαχείριση περιεχομένου, συ-νεργασία και κοινωνική δικτύωση.

Αρχιτεκτονική Προσανατολισμένη προς τις Υπηρεσίες: Η συγκεκριμένη διαδικτυακή πύλη έχει σχεδιαστεί με την χρήση μίας ανοιχτής αρχιτεκτονικής προσανατολισμένης προς τις υπηρεσίες (Service-Oriented Architecture, SOA) η οποία την κάνει ιδανική επιλογή για τις εταιρείες παγκοσμίως όσον αφορά την ενσωμάτωση με εταιρικές εφαρμογές. Έτσι, είναι δυνατή η ενσωμάτωση συστημάτων ανθρώπινων πόρων, λογιστικών συστημάτων ή συστημάτων πωλήσεων καθώς και διάφορων τύπων δεδομένων.

Ασφαλής Ενιαία Πρόσβαση (Single Sign On, SSO): Το Liferay προσφέρει ενοποιημένη παρουσίαση και πρόσβαση στην πληροφορία, φέρνοντας όλα τα διαφορετικά συστήματα κοντά και κάνοντάς τα διαθέσιμα με τον χρήστη να είναι απαραίτητο να συνδεθεί μόνο μία φορά δια μέσου ασφαλούς ενιαίας πρόσβασης.

Παραμετροποιήσιμη Εξουσιοδότηση: Για να είναι σίγουρο πως οι κατάλληλοι άνθρωποι έχουν πλήρη έλεγχο στην κατάλληλη πληροφορία, οι διαχειριστές της πύλης μπορούν να αναθέσουν σε συγκεκριμένες ομάδες ή μοναδικούς χρήστες διαφορετικούς ρόλους, οι οποίοι παρέχουν κυμαινόμενα προνόμια πρόσβασης και επεξεργασίας σε συγκεκριμένες κοινότητες, αρχεία, εφαρμογές και εργαλεία. Για παράδειγμα, ένας διευθυντής πωλήσεων μπορεί να δει και να επεξεργαστεί όλα τα έγγραφα που αφορούν τις πωλήσεις, ενώ ένας βοηθός πωλήσεων μπορεί μόνο να τα δει.

Κοινότητες και Οργανισμοί: Οι χρήστες του Liferay μπορούν να εννοιολογικά να ομαδοποιηθούν σε μία ιεραρχία οργανισμών ή διαρθρωτικές κοινότητες, παρέχοντας προσαρμοστικότητα και ευκολία στην διαχείριση. Για παράδειγμα, μέλη διαφορετικών γεωγραφικών περιοχών μπορούν να ομαδοποιηθούν σε οργανισμούς, ενώ οι κλάδοι όπως οι πωλήσεις, η χημεία και η πληροφορική μπορούν να ομαδοποιηθούν ως κοινότητες. Κάθε τέτοια ομάδα θα μπορούσε να έχει μια δικιά της ξεχωριστή μικροπύλη, με το δικό της σύνολο σελίδων, σύστημα διαχείρισης περιεχομένου, διαμοιρασμένο ημερολόγιο και σύστημα πρόσβασης. Ένας χρήστης όχι μόνο μπορεί να ανήκει σε διαφορετικές ομάδες αλλά και να έχει διαφορετικό ρόλο σε κάθε ομάδα και παρ' όλ' αυτά να πλοηγείται απροβλημάτιστα μεταξύ τους.

Εύκολη Παραμετροποίηση: Η διεπαφή του Liferay είναι γρήγορη και “απαντά” άμεσα με αποτέλεσμα η παραμετροποίηση να είναι εύκολη και διασκεδαστική για κάθε χρήστη. Παραδοσιακά αργές διαδικασίες όπως η αλλαγή της διάταξης μίας σελίδας, η προσθή-

Επίπεδο Χρήστη

κη νέων εφαρμογών και περιεχομένου καθώς και η αλλαγή της εικόνας και της αίσθησης μίας σελίδας μπορεί να πραγματοποιηθεί με ένα κλικ, και χωρίς καν να χρειαστεί να φορτωθεί ξανά η σελίδα.

Δυναμικό Σύρσιμο (Drag 'n' Drop): Όντας η πρώτη διαδικτυακή πύλη με χαρακτηριστικά συρσίματος με το ποντίκι, επιτρέπει στους χρήστες να μεταφέρουν διάφορα στοιχεία σε ολόκληρη την πύλη με την ευκολία της απλής επιλογής και μετακίνησης με το ποντίκι.

Επιφάνεια Εργασίας: Είναι εξαιρετικά απλό να εργαστεί κάποιος από τον υπολογιστή του, στο τοπικό σύστημα αρχείων της επιφάνειας εργασίας του. Απλή μεταφορά των φακέλων με το ποντίκι στους WebDAV φακέλους επιτρέπει στα δεδομένα να ανεβαίνουν αυτόματα στην διαδικτυακή πύλη.

Αναζήτηση και Τιτλοφόρηση: Είναι δυνατή η προσθήκη ετικετών στο περιεχόμενο ιστού, στα έγγραφα, στα νήματα μηνυμάτων κλπ με αποτέλεσμα δυναμικό διαμοιρασμό σημαντικού ή ενδιαφέροντος περιεχομένου με άλλους χρήστες της πύλης. Οι χρήστες μπορούν να αναζητήσουν παρεμφερή πληροφορία μέσα σε συγκεκριμένες εφαρμογές, κοινότητες, ολόκληρη την πύλη ή ακόμα και εξωτερικές ενσωματωμένες εφαρμογές.

Προσωπικές Σελίδες Χρηστών: Όλοι οι χρήστες έχουν τον δικό τους χώρο ο οποίος μπορεί είτε να δημοσιευθεί στο κοινό (με διάφορες παραμέτρους όσον αφορά συγκεκριμένο κοινό, όσον αφορά τα μέρη τα οποία θα κοινοποιηθούν και όσον αφορά τις διευθύνσεις οι οποίες θα δείχνουν στο περιεχόμενο αυτό) είτε να κρατηθούν προσωπικές. Ο χρήστης έχει στην διάθεσή

του μία πληθώρα επιλογών όσον αφορά τον τρόπο με τον οποίον θα παρουσιάζεται ο χώρος, τι εργαλεία και εφαρμογές θα παρέχονται, τι πάει στην βιβλιοθήκη εγγράφων καθώς και ποιός μπορεί να έχει πρόσβαση σε τι καθώς και τι είδους πρόσβαση.

Υποστήριξη για Γλώσσες: Διεθνείς ή πολύγλωσσοι οργανισμοί έχουν στην διάθεσή τους υποστήριξη για 22 γλώσσες. Η αλλαγή μπορεί να γίνει από τον κάθε χρήστη με το πάτημα ενός κουμπιού. Είναι επίσης δυνατή η προσθήκη και άλλων γλωσσών και αυτό με εξαιρετική ευκολία.

Βιβλιοθήκη Εγγράφων και Γκαλερί Εικόνων: Το Liferay παρέχει ένα κεντρικό σημείο όπου συγκεντρώνεται και διαχειρίζεται όλο το περιεχόμενο. Κάθε κοινότητα έχει τη δικιά της ξεχωριστή βιβλιοθήκη εγγράφων και γκαλερί εικόνων. Και οι δύο είναι εξοπλισμένοι με παραμετροποιήσιμους φακέλους και λειτουργούν ως ένα βασισμένο στον ιστό, διαμοιραζόμενο σκληρό δίσκο για όλα τα μέλη της εκάστοτε ομάδας, ανεξάρτητα από το που βρίσκονται τα μέλη αυτά χωρικά. Λόγω του ότι το περιεχόμενο είναι προσβάσιμο μόνο από εκείνους οι οποίοι επιτρέπονται από τον διαχειριστή, κάθε αρχείο είναι όσο ανοιχτό ή ασφαλές πρέπει να είναι.

Ενσωμάτωση με την Σουίτα Γραφείου της Microsoft: Η συγκεκριμένη διαδικτυακή πύλη ενσωματώνει πλήρως το Office της Microsoft, κάτι το οποίο πρακτικά σημαίνει όταν κάποιος ενημερώνει αρχεία και έγγραφα από τον τοπικό σκληρό δίσκο, τότε αυτά μπορούν να ενημερώνουν αυτόματα την διαδικτυακή βιβλιοθήκη. Το Liferay χρησιμοποιεί τα διαθέσιμα στο κοινό πρωτόκολλα SharePoint με αποτέλεσμα να επιτρέπει στους

Blog

Your request processed successfully.

Search

Search Entries

Add Blog Entry

New Blog on new portal!

By Mike Life, On 4/22/08 3:58 PM

Tags: portal

Edit

Permissions

Delete

This is my first blog entry. Everybody who is registered on the portal can have a blog. It is easy: just add the Blogs portfolio to a page in your personal community, and start adding entries! Because we have the Recent Bloggers portfolio configured on the Community page, you will be able to see at a glance all the recent bloggers, and your blog entries will appear there! Isn't this cool?

BlogStar

del.icio.us

Digg

Reddit

StumbleUpon

Technorati

Your Rating

Average (0 Votes)

★★★★★

★★★★★

0 Comments

Subscribe to this blog

Showing 1 result

RSS

Liferay News

News and announcements from Liferay, Inc.

Is the portal the application?

4/9/08 5:14 PM

Alice Cheng

"Here in the US, Liferay is seeing more requests for integrated collaboration capabilities, like profiles, wikis, blogs and discussion forums, that are delivered to end users in the portal itself. The company is even toying with out how best to refer to its product in this new world..."

InfoQ Reviews Liferay 4.4

Liferay Portal Standardizes on jQuery

The Server Side Reviews Liferay Portal 4.4

jQuery: Standardization & Services Support!

Liferay and Sygnity Form Services Partnership

British Airways sets up tech innovation unit

Liferay Powers SpringSource Website

FreeAnalysis Embedded into Liferay

Communities

Return to Full Page

Edit Permissions for Page: Admin

Users

Organizations

User Groups

Regular Roles

Community Roles

Community

Guest

« Back

Current

Available

Search

Search Roles

Update Permissions

Showing 3 results.

	Name	Type	Permissions
☐	Community Administrator	Community	
☐	Community Member	Community	
☐	Community Owner	Community	

Message Boards

Return to Full Page

Categories

My Posts

My Subscriptions

Recent Posts

Statistics

Banned Users

Categories » General Discussion » Welcome to the message boards!

Threads [Previous | Next]

Post New Thread

Subscribe

Move Thread

Welcome to the message boards!

Reply

Reply with Quote

Top

4/22/08 7:30 PM

These are our message boards! Here you can discuss any number of topics. To get more message categories added to the boards, please send an email to lifeman@mycompany.com.

Recent Posts

Edit

Permissions

Delete

Επίπεδο Χρήστη

χρήστες να ανοίγουν, να σώζουν, να κλειδώνουν και να μοιράζονται έγγραφα απευθείας μεταξύ του Liferay και του Microsoft Office. Κάτι τέτοιο δημιουργεί νέους τρόπους ενσωμάτωσης της εργασίας στην επιφάνεια του υπολογιστή με την συνεργασία και την αποθήκευση εγγράφων που βασίζεται στον ιστό. Για παράδειγμα μία ομάδα μάρκετινγκ η οποία δουλεύει πάνω σε ένα νέο προϊόν μπορεί να αποθηκεύσει αρχεία διαφανειών, φυλλάδια και γραφικά σε μία βιβλιοθήκη εγγράφων και ανταλλάσει ιδέες με την χρήση συνεργατικών εργαλείων όπως υπηρεσίες άμεσων μηνυμάτων, περιοχές συζητήσεων κλπ

Ανέβασμα Πολλαπλών Αρχείων: Αντί να προσθέτουν ένα-ένα τα αρχεία, οι χρήστες μπορούν να ανεβάσουν πολλά αρχεία τη φορά στην βιβλιοθήκη εγγράφων, την γκαλερί εικόνων και οποιοδήποτε από τα εργαλεία επιτρέπουν επισύναψη αρχείων. Επίσης στην βιβλιοθήκη εγγράφων είναι δυνατή η μετατροπή του τύπου του αρχείου κατά το ανέβασμα (π.χ. Η μετατροπή ενός αρχείου κειμένου σε μορφή αρχείου σύνθετων καλογικών στοιχείων – pdf).

Έκδοση στον Ιστό: Πέρα από την διαχείριση υπάρχοντος περιεχομένου, οι χρήστες μπορούν εύκολα να δημιουργήσουν και να διαχειριστούν τα πάντα από απλά άρθρα κειμένου και εικόνων σε πλήρως λειτουργικούς διαδικτυακούς τόπους. Τα εργαλεία δημοσίευσης στον ιστό δουλεύουν παράλληλα με άλλες εφαρμογές του συστήματος, κάτι το οποίο σημαίνει ότι κάποιος μπορεί να προσθέσει λειτουργίες καλαθιού αγορών, δημοσκοπήσεις επισκεπτών, φόρμες στον ιστό, εργαλεία συνεργασίας στις κοινότητες κ.α. Όλα γίνονται με την συλλογή εύχρηστων υπηρεσιών στις

οποίες ο χρήστης είναι εξοικειωμένος όπως εμπλουτισμένοι κειμενογράφοι και φιλικές διεπαφές.

Βελτιστοποίηση για τις Μηχανές Αναζήτησης: Το Liferay αυτόματα ενημερώνει τον χάρτη της ιστοσελίδας με την νέα πληροφορία και κάνει τις νέες ιστοσελίδες άμεσα αναζητήσιμες από τις μηχανές αναζήτησης.

Εύκολοι και Πλούσιοι Κειμενογράφοι: Η δημιουργία και η δημοσίευση περιεχομένου στην πλατφόρμα είναι εξαιρετικά εύκολη. Ένας πλούσιος κειμενογράφος με ορθογραφικό έλεγχο και μορφοποίηση κειμένου επιτρέπει σε όλους να δημιουργούν περιεχόμενο χωρίς να γνωρίζουν πως να προγραμματίζουν σαν να επεξεργάζονταν ένα αρχείο κειμενογράφου. Η δυνατότητα της επεξεργασίας του πηγαίου κώδικα δεν χάνεται, καθώς οι χρήστες που γνωρίζουν πως να προγραμματίζουν μπορούν να έχουν περεταίρω έλεγχο στο αποτέλεσμα όταν ενεργοποιήσουν την λειτουργία πηγαίου κώδικα.

Δημοσίευση στους Χώρους Εργασίας: Οργανισμοί παγκοσμίως χρησιμοποιούν την συγκεκριμένη διαδικτυακή πύλη για να δημοσιεύσουν στα ενδοδίκτυα, στις πλατφόρμες συνεργασίας και σε άλλα περιβάλλοντα για την κάλυψη των λειτουργικών αναγκών τους. Αξιοποιώντας το σύστημα κοινότητων και οργανισμών, κάθε τμήμα ή ομάδα μπορεί να έχει το δικό της ασφαλή χώρο, εξοπλισμένο με ένα σύνολο χαρακτηριστικών, εμφάνιση και συνολική αίσθηση της επιλογής του. Κάθε χώρος έχει το δικό του σύστημα διαχείρισης περιεχομένου και βιβλιοθήκη εγγράφων τα οποία μπορούν να συμπληρωθούν από

Επίπεδο Χρήστη

οποιοδήποτε άλλο συνεργατικό ή διαχειριστικό εργαλείο και εφαρμογή. Μπορούν ακόμη να μοιραστούν έγγραφα και πληροφορία μεταξύ τομέων ή κοινοτήτων.

Υπηρεσίες Ζωντανής Επεξεργασίας Σελίδων και Χρονοπρογραμματισμού: Αν παρουσιαστεί η ανάγκη αλλαγής κάποιων σελίδων οι οποίες έχουν δημιουργηθεί και δημοσιευθεί, αυτό μπορεί να γίνει ζωντανά. Το σύστημα επιτρέπει την επεξεργασία και προεπισκόπηση των ενημερώσεων σε πραγματικό χρόνο, χωρίς να επηρεάζεται αυτό που το κοινό βλέπει μέχρι το περιεχόμενο να είναι απόλυτα έτοιμο προς δημοσίευση. Αυτό μπορεί να γίνει σε μεμονωμένες σελίδες, ολόκληρες κοινότητες ή ακόμα και σε πλήρεις διαδικτυακές πύλες. Στην περίπτωση που δεν είναι επιθυμητή η άμεση δημοσίευση, είναι στην ευχέρεια του χρήστη να ενημερώσει το σύστημα για το πότε να δημοσιεύσει, τι να δημοσιεύσει και πού να δημοσιεύσει. Είναι δυνατή ακόμη και η επιστροφή σε παλαιότερες εκδόσεις του περιεχομένου.

Δυναμική Εικονική Φιλοξενία: Μετά την δημιουργία ενός διαδικτυακού τόπου, ενός ενδοδικτύου ή οποιασδήποτε άλλης εγκατάστασης, δυναμική εικονική φιλοξενία επιτρέπει στον χρήστη να χρησιμοποιήσει την ίδια εγκατάσταση του Liferay για να αναγεννήσει έναν άπειρο αριθμό από διαδικτυακές πύλες, κάθε μία με το δικό της URL, την δική της εμφάνιση και την δική της λειτουργικότητα. Για παράδειγμα, ένας μουσικός διαδικτυακός τόπος, μπορεί να τρέχει έναν άπειρο αριθμό από υποσελίδες για διάφορους μουσικούς. Κάθε μουσικός θα είχε στα χέρια του μία δικιά του ιστοσελίδα – επί της ουσίας μία μικροπύλη – με ειδική πρόσβαση για τους θαυμαστές του, μοναδική διεύθυνση URL και μοναδικό θέμα.

Wikis: Οι χρήστες μπορούν να χτίσουν και να τεκμηριώσουν σημαντικές ή ενδιαφέρουσες πληροφορίες ως κοινότητα με το Wiki, ένα χαρακτηριστικό το οποίο είναι εφάμιλλο των αντίστοιχων αυτόνομων προϊόντων. Κάθε κοινότητα αποκτά το δικό της wiki, με το δικό του σύνολο εξουσιοδοτήσεων. Καθένας με τα κατάλληλα δικαιώματα μπορεί να συνεισφέρει πληροφορία σε αυτού του είδους τα διαδικτυακά έγγραφα.

Πίνακες Μηνυμάτων: Οι πίνακες μηνυμάτων αποτελούν πολύ αποδοτικές λύσεις για την διευκόλυνση συζητήσεων γύρω από κοινές ιδέες μέσα σε ένα τμήμα ή μία ομάδα έργου. Μπορούν ακόμη να χρησιμοποιηθούν για την σύλληψη και αποθήκευση της υπονοούμενης γνώσης της ομάδας εργασίας. Το Liferay προσφέρει εκθέσεις των στατιστικών στοιχείων δραστηριότητας των πινάκων μηνυμάτων και οι χρήστες μπορούν να εγγράφονται στα θέματα μέσω RSS και να απαντούν μέσω email. Όπως και οι υπόλοιπες εφαρμογές, έτσι και οι πίνακες μηνυμάτων είναι ασφαλείς μέσω του κοκκώδους συστήματος πρόσβασης, δικαιωμάτων και ρόλων, το οποίο προσφέρει διαφορετικά επίπεδα ελέγχου σε διαφορετικούς χρήστες.

Ιστολόγια: Τα ιστολόγια του συστήματος παρέχουν τα καλύτερα χαρακτηριστικά των σύγχρονων εργαλείων δημοσίευσης στον ιστό, σε συνδυασμό με τα οφέλη της κοινοτητοκεντρικής φύσης του Liferay. Επιτρέπουν στους χρήστες να μεταφέρουν πληροφορίες και διευκολύνουν τις συζητήσεις γύρω από τα οποιοδήποτε θέμα, ωθώντας το άμεσα σε ιστοσελίδες, ενδοδίκτυα ή σελίδες κοινωνικής δικτύωσης. Τα χαρακτηριστικά περιλαμβάνουν ένα φιλικό προς τον χρήστη πλούσιο επεξεργαστή κειμένου, συνδέσμους προς διαδικτυα-

Επίπεδο Χρήστη

κές υπηρεσίες σελιδοδεικτών, ειδοποιήσεις ηλεκτρονικού ταχυδρομείου, και ένα σύστημα αξιολόγησης. Όλα τα ιστολόγια είναι διαθέσιμα προς παρακολούθηση μέσω RSS και οι χρήστες μπορούν να προγραμματίσουν τις ανακοινώσεις τους να δημοσιευθούν σε συγκεκριμένες ημερομηνίες και ώρες.

Παρακολούθηση της Δραστηριότητας: Προσφέρονται δυνατότητες εντοπισμού της πιο πρόσφατης δραστηριότητας στα ιστολόγια, τις περιοχές συζητήσεων, τους πίνακες μηνυμάτων και άλλα εργαλεία. Αυτή η πληροφορία παρουσιάζεται στις εφαρμογές “πρόσφατης δραστηριότητας” καθώς και σε έναν τοίχο δραστηριότητας που θυμίζει πολύ το facebook [46].

Άμεσα Μηνύματα: Μία λίστα φίλων εμφανίζει αυτόματα τα ονόματα όλων των συνδεδεμένων στο σύστημα χρηστών. Αν κάποιος επιθυμεί να επικοινωνήσει, η διαδικασία είναι εξαιρετικά απλή, και δεν διαφέρει από τις αντίστοιχες εφαρμογές στην επιφάνεια εργασίας καθώς υπάρχουν φωτογραφίες προφίλ και προσαρμοσμένα μηνύματα όταν ο χρήστης είναι απασχολημένος. Για ευκολία, τα παράθυρα των συζητήσεων παραμένουν στο κάτω μέρος της οθόνης και ακολουθούν τον χρήστη καθώς πλοηγείται μέσα στην διαδικτυακή πύλη. Οι συζητήσεις παραμένουν ασφαλείς καθώς η κουβέντα λαμβάνει χώρα πίσω από το τείχος προστασίας του εκάστοτε οργανισμού.

Ηλεκτρονικό Ταχυδρομείο: Το Liferay δίνει την δυνατότητα στους χρήστες να στέλνουν μηνύματα ηλεκτρονικού ταχυδρομείου μέσα από το σύστημα. Ένα πλήρες υποσύστημα ηλεκτρονικού ταχυδρομείου το οποίο βασίζεται σε τεχνολογίες ασύγχρονων κλήσεων (AJAX) με ένα φιλικό προς τον χρήστη περιβάλλον

μπορεί να παραμετροποιηθεί έτσι ώστε να συνδέεται με όλους τους δημοφιλείς διακομιστές emails IMAP.

RSS: Αποτελεί έναν τρόπο να ενημερώνεται ο χρήστης για γεγονότα και νέα από άλλους χρήστες ή και κανάλια πληροφορίας από τους πίνακες ανακοινώσεων, τις περιοχές συζητήσεων και τα ιστολόγια μέσα στην διαδικτυακή πύλη.

Κοινά Ημερολόγια: Ημερολόγια με έδρα την κοινότητα και λίστες εργασιών επιτρέπουν στους χρήστες να δημιουργούν, να διαχειρίζονται και να αναζητούν γεγονότα. Τα γεγονότα μπορούν να διαμοιράζονται δια μέσου των κοινοτήτων καθώς και υπενθυμίσεις γεγονότων μπορούν να ειδοποιούν τους χρήστες για επερχόμενα γεγονότα μέσω ηλεκτρονικού ταχυδρομείου, άμεσα μηνύματα, ή μηνύματα στο κινητό τηλέφωνο. Υποστήριξη διαφόρων τύπων επιτρέπει στον χρήστη την μεταφορά των πληροφοριών ημερολογίου και χρηστών μέσω των προτύπων του Ιστού 2.0 (Web 2.0)

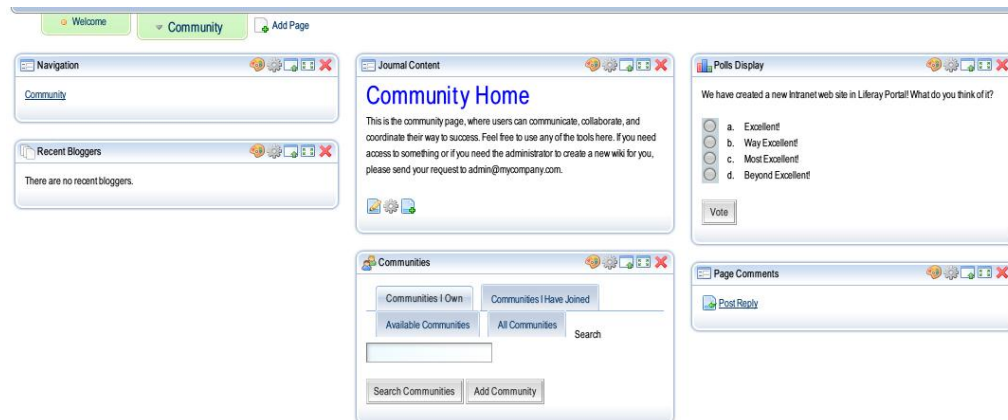
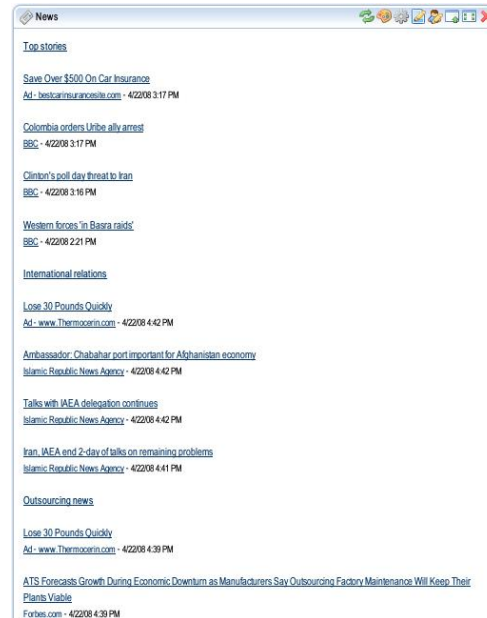
Ανακοινώσεις και Ειδοποιήσεις: Οι χρήστες έχουν την δυνατότητα να εκπέμψουν μηνύματα σε διαφορετικές ομάδες χρηστών, ανεξάρτητα από το αν είναι απλές ανακοινώσεις ή ενημερωτικά δελτία. Κάθε χρήστης μπορεί να ελέγχει πως θα λαμβάνει ειδοποιήσεις: μέσω ειδοποιήσεων στην διαδικτυακή πύλη, μέσω μηνυμάτων στο κινητό τηλέφωνο, μέσω ηλεκτρονικού ταχυδρομείου ή οποιονδήποτε άλλο μηχανισμό υλοποίησης ο οποίος έχει ρυθμιστεί από τον διαχειριστή της διαδικτυακής πύλης.

Δημοσκοπήσεις: Δημοσκοπήσεις πολλαπλής επιλογής μπορούν να δημιουργηθούν, ενώ παράλληλα γίνεται

Επίπεδο Χρήστη

αποθήκευση των ψήφων. Πολλά ξεχωριστά δημοψηφίσματα μπορούν να διαχειρίζονται και μία ξεχωριστή εφαρμογή μπορεί να ρυθμιστεί ώστε να εμφανίζει τα αποτελέσματα μίας συγκεκριμένης δημοσκόπησης.

Αυτά είναι μερικά μόνο από τα χαρακτηριστικά στοιχεία του Liferay. Το σύστημα παρέχει ένα επαναστατικό σύνολο λειτουργιών για τον χρήστη, εξαιρετικούς συνδυασμούς εγκατάστασης για τον διαχειριστή και ένα σημαντικό περιβάλλον ανάπτυξης καθώς και έναν ανοιχτού κώδικα πυρήνα για τον προγραμματιστή εφαρμογών. Τα χαρακτηριστικά αυτά κάνουν το Liferay ιδανικό ως επιλογή στην ανάπτυξη του και την χρήση δικού μας συστήματος. Όχι μόνον τα στοιχεία αυτά δεν θα ήταν διαθέσιμα διαφορετικά, αλλά και οι δυνατές επιλογές εγκατάστασης προσφέρουν πολλαπλά πλεονεκτήματα έναντι των ανταγωνιστικών πακέτων.



Portlets

80

Οι εφαρμογές των διαδικτυακών πυλών έρχονται σε δύο βασικά είδη: (1) Portlets μπορούν να δημιουργηθούν για να παρέχουν μικρές ποσότητες λειτουργικότητας και έπειτα να συνδυάζονται από τον εξυπηρετητή σε μία μεγάλη εφαρμογή και (2) Ολόκληρες εφαρμογές μπορούν να γραφτούν οι οποίες θα βασίζονται σε ένα ή μερικά portlets. Η επιλογή βρίσκεται σε αυτόν που σχεδιάζει την εφαρμογή. Ο προγραμματιστής έχει μόνο να ασχοληθεί με το τι θα γίνεται μέσα στο ίδιο το portlet. Ο εξυπηρετητής αναλαμβάνει να χτίσει την ιστοσελίδα και να την παρουσιάσει στον χρήστη.

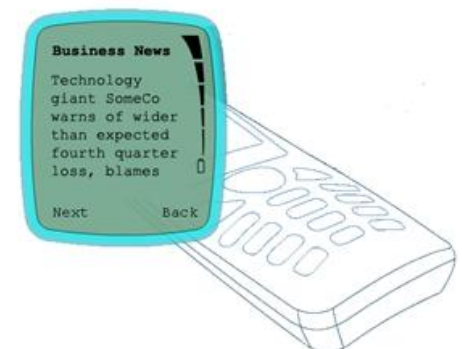
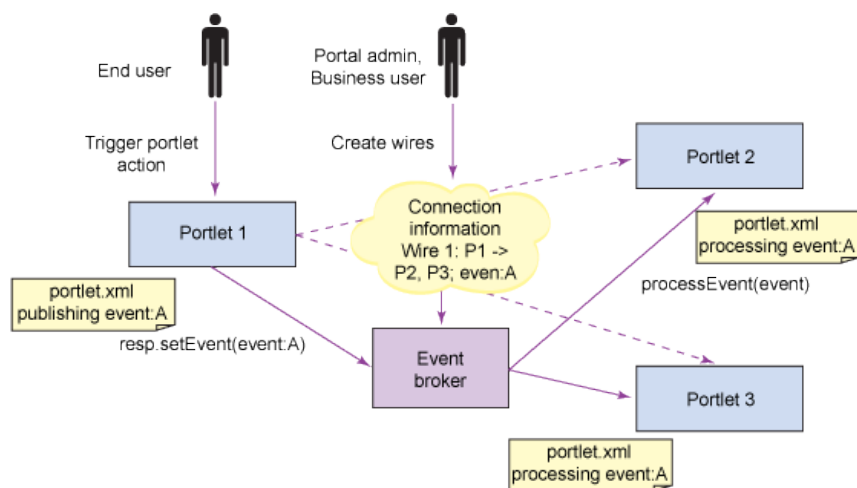
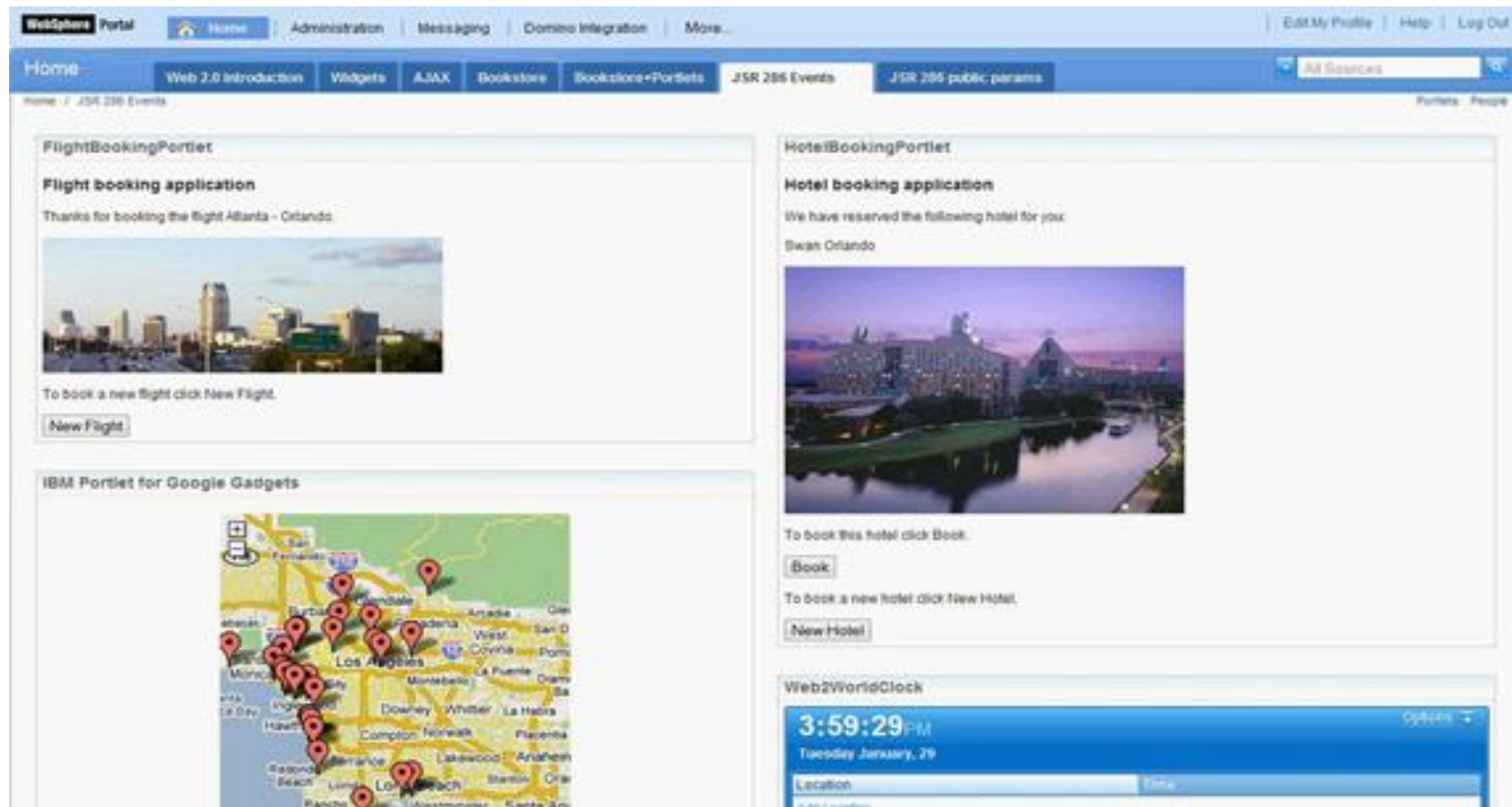
Ένα portlet χειρίζεται αιτήσεις και δημιουργεί δυναμικό περιεχόμενο, το οποίο ονομάζεται τεμάχιο (fragment). Τα τεμάχια είναι σε κάποια γλώσσα ετικετών (markup language), όπως HTML, XHTML ή WML. Τα τεμάχια από τα διάφορα portlets συνενώνονται σε ένα ολοκληρωμένο κείμενο. Το πρόγραμμα – πελάτης του χρήστη αλληλεπιδρά με τα portlets στέλνοντας αιτήσεις και λαμβάνοντας κάποια απάντηση. Αυτή η αλληλεπίδραση μπορεί να είναι κάποιος σύνδεσμος που θα ακολουθήσει ο χρήστης ή η αποστολή δεδομένων χρησιμοποιώντας μια φόρμα. Οι ενέργειες του χρήστη προωθούνται από τον portlet container στο κατάλληλο portlet το οποίο αναλαμβάνει να εκτελέσει τις κατάλληλες ενέργειες. Όπως αναφέρθηκε παραπάνω, τα portlets μοιράζονται πολλές ιδέες με τα servlets. Όπως γίνεται και με το servlet container και τα servlets, έτσι και τον κύκλο ζωής των portlet τον διαχειρίζεται ο portlet container:

- Init. Αρχικοποίηση του portlet.
- Διαχείριση αιτήσεων. Επεξεργασία των ενεργειών του χρήστη.

- Καταστροφή. Λήξη του κύκλου ζωής του portlet.

Μία επιπλέον δυνατότητα των portlets είναι ότι έχουν καταστάσεις λειτουργίας οι οποίες δείχνουν τη λειτουργία που εκτελεί το portlet μία δεδομένη χρονική στιγμή. Οι καταστάσεις λειτουργίας μπορούν να χωριστούν σε τρεις κύριες κατηγορίες:

- Υποχρεωτικές καταστάσεις. Αυτές είναι οι καταστάσεις Επεξεργασίας (Edit), Βοήθειας (Help) και Όψης (View). Ένα portlet πρέπει να υποστηρίζει την κατάσταση Όψης προκειμένου να μπορεί να δημιουργήσει τις κατάλληλες ετικέτες για μία σελίδα. Η κατάσταση Επεξεργασίας επιτρέπει στο χρήστη να μεταβάλλει τα δεδομένα που χρησιμοποιεί το portlet, ενώ η κατάσταση Βοήθειας εμφανίζει μια οθόνη με βοήθεια για το χρήστη.
- Προαιρετικές ειδικές καταστάσεις. Τέτοιες καταστάσεις περιλαμβάνουν την κατάσταση Σχετικά, (About) όπου εμφανίζονται πληροφορίες για το portlet, η κατάσταση Ρυθμίσεων (Config) που επιτρέπει στους διαχειριστές να αλλάξουν τις ρυθμίσεις ενός portlet, Επεξεργασία_Αρχικών_Ρυθμίσεων (Edit_defaults) που επιτρέπει στους διαχειριστές να ορίσουν εκ των προτέρων κάποιες τιμές για την κατάσταση Επεξεργασίας, και η κατάσταση Εκτύπωσης (Print) για την εμφάνιση του περιεχομένου ενός portlet σε μορφή που μπορεί να εκτυπωθεί εύκολα.
- Καταστάσεις που εξαρτώνται από τον κατασκευαστή της δικτυακής πύλης. Οι καταστάσεις αυτές δεν ορίζονται στις προδιαγραφές.



Portlets

Οι περισσότεροι προγραμματιστές σήμερα προτιμούν να χρησιμοποιούν συγκεκριμένες πλατφόρμες (framework) για να αναπτύξουν τις εφαρμογές τους, και αυτό διότι αυτές οι πλατφόρμες παρέχουν λειτουργικότητα και δομή σε ένα έργο. Για παράδειγμα, η τεχνολογία Struts ενισχύει το σχεδιαστικό πρότυπο "Μοντέλο-Προβολή-Ελεγκτής" (Model-View-Controller design pattern, MVC) και παρέχει πολλή λειτουργικότητα, όπως προσαρμοσμένες ετικέτες (tags) και λειτουργίες ελέγχων (validations), τα οποία βοηθούν τον προγραμματιστή να υλοποιήσει συγκεκριμένα χαρακτηριστικά. Με το Liferay οι χρήστες είναι ελεύθεροι να χρησιμοποιήσουν όλες τις γνωστές πλατφόρμες στην περιοχή της JavaEE, όπως Struts, Spring και JSF (Java Server Faces). Αυτό επιτρέπει στους προγραμματιστές που γνωρίζουν ήδη αυτές τις πλατφόρμες να υλοποιήσουν portlet με μεγαλύτερη ευκολία, και επίσης βοηθά στην εύκολη και γρήγορη μεταφορά μιας εφαρμογής που υπάρχει έχει υλοποιηθεί σε αυτές τις πλατφόρμες να περάσει σε υλοποίηση portlet.

που έχουν αναπτυχθεί από χρήστες, καθώς και σαν παραδείγματα στο πακέτο ανάπτυξης λογισμικού επεκτάσεων του Liferay.

82

Επιπλέον, το Liferay επιτρέπει την χρήση εφαρμογών (portlets) γραμμένων σε PHP και Ruby να απορροφθούν χωρίς δυσκολία ως portlets, με αποτέλεσμα να μην χρειάζεται κάποιος να είναι προγραμματιστής Java για να αξιοποιήσει τις δυνατότητες του Liferay (όπως διαχείριση χρηστών, κοινότητες, δημιουργία σελίδων, διαχείριση περιεχομένου κλπ). Κάποιος μπορεί να χρησιμοποιήσει το SDK των επεκτάσεων (Plugins SDK), όπως περιγράφεται σε άλλη ενότητα, για να εγκαταστήσει μία εφαρμογή γραμμένη σε PHP ή Ruby ως portlet, και αυτή θα τρέξει απροβλημάτιστα στο Liferay. Μάλιστα υπάρχει πληθώρα παραδειγμάτων από τον κατάλογο των portlets που υπάρχουν στην βιβλιοθήκη του Liferay, από τον κατάλογο portlets

Portlets

Δημιουργία Portlet

Η δημιουργία portlet με το SDK είναι μία πολύ απλή διαδικασία. Όπως έχει αναφερθεί νωρίτερα, υπάρχει ένας φάκελος portlets μέσα στον φάκελο του SDK. Σε αυτό τον φάκελο πρόκειται να εδρεύσει το έργο. Για την δημιουργία, αρχικά πρέπει να τεθεί ένα όνομα, τόσο το όνομα του έργου, όσο και το όνομα που θα εμφανίζεται (π.χ. μπορεί να υπάρχουν τα έργα "seeMe1.1", "seeMe1.2" και "seeMe2.6" όμως σε όλα το όνομα εμφάνισης να είναι "See Me!"). Απλά δίνεται η παρακάτω εντολή.

Κώδικας:

```
ant -Dportlet.name=<project name> \
-Dportlet.display.name=<portletTitle>
"\ create
```

και το έργο δημιουργείται μέσα στον φάκελο portlets του SDK. Δηλαδή για το τρίτο έργο του παραπάνω παραδείγματος είναι ο παρακάτω:

Κώδικας:

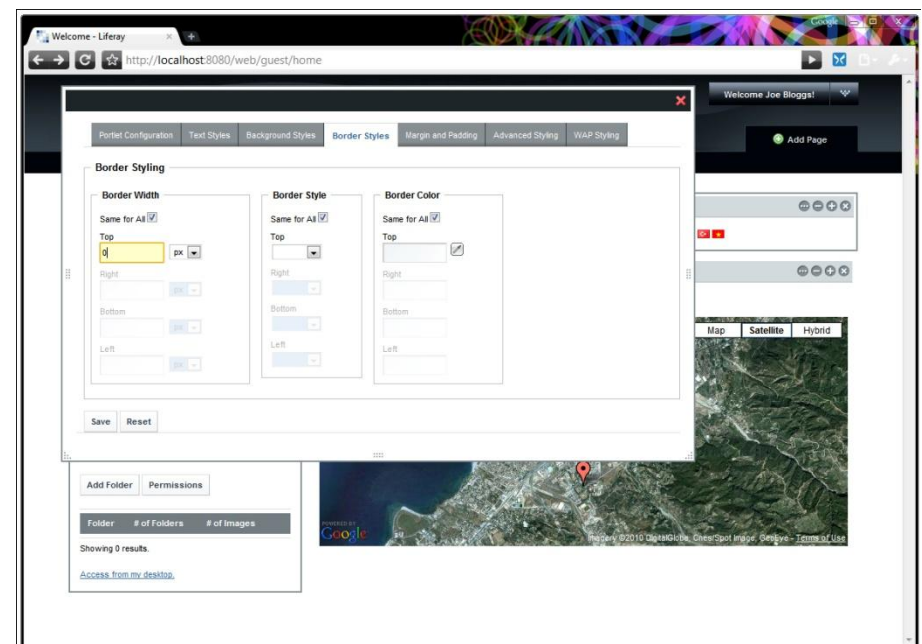
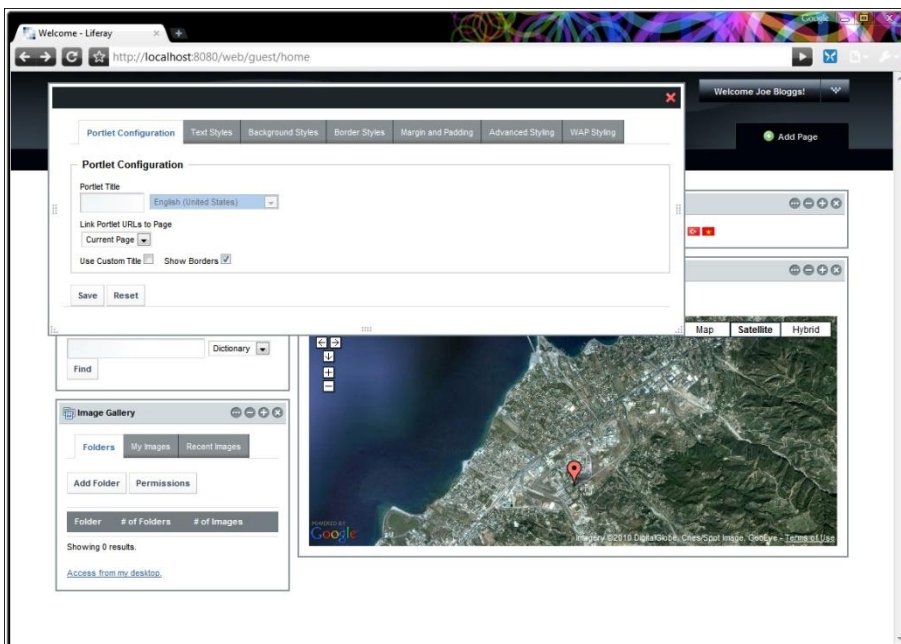
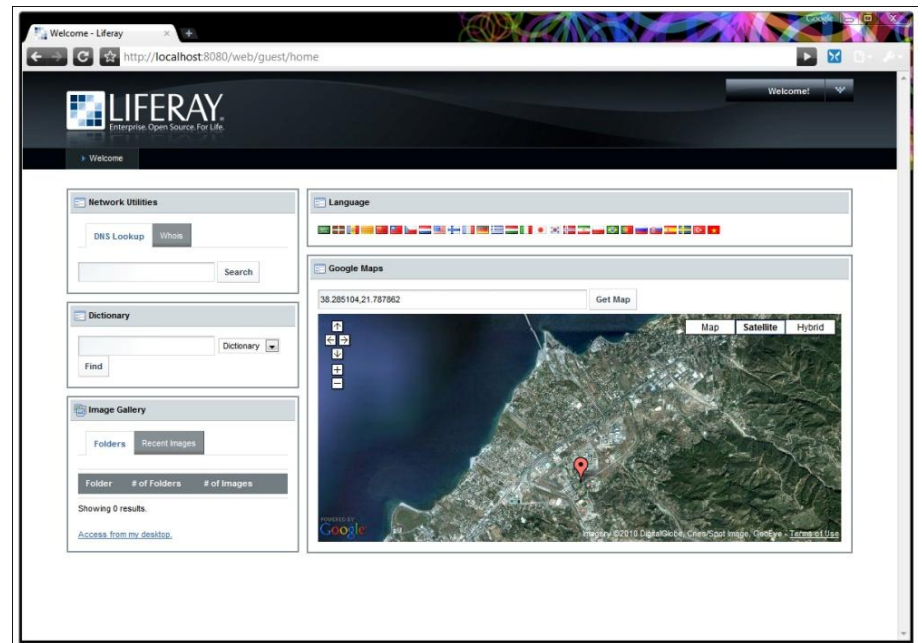
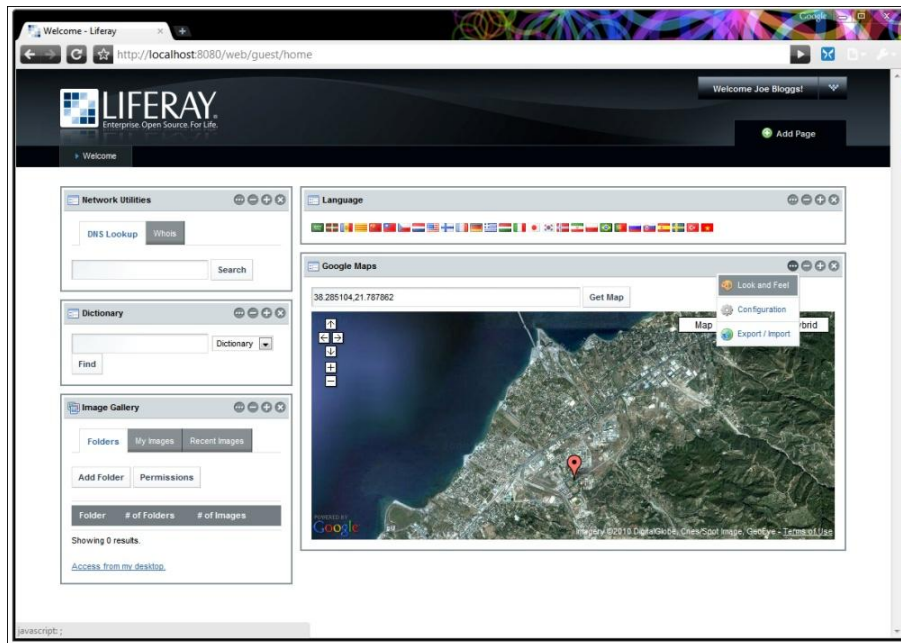
```
ant -Dtheme.name=seeMe1.3\
-Dtheme.display.name="See Me!" create
```

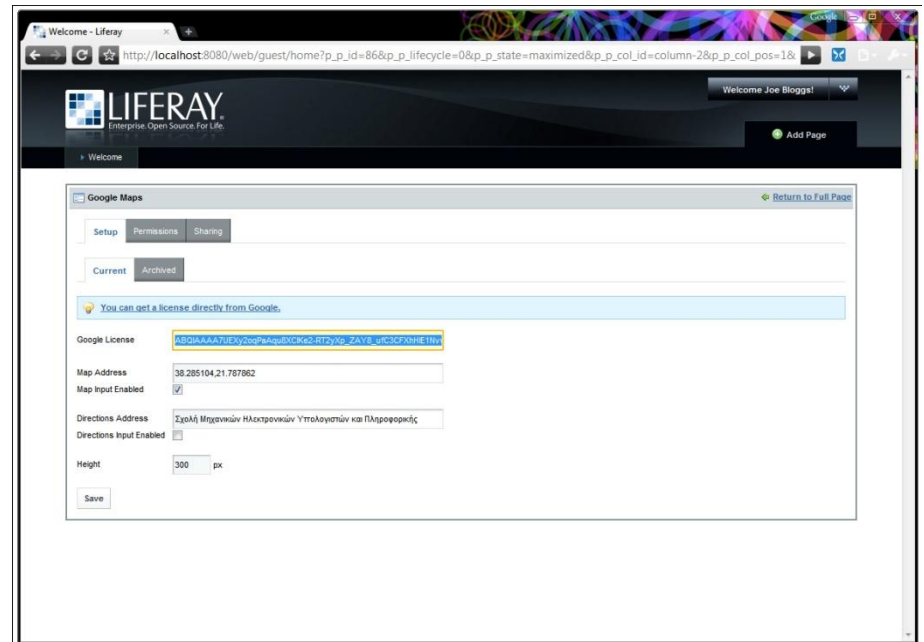
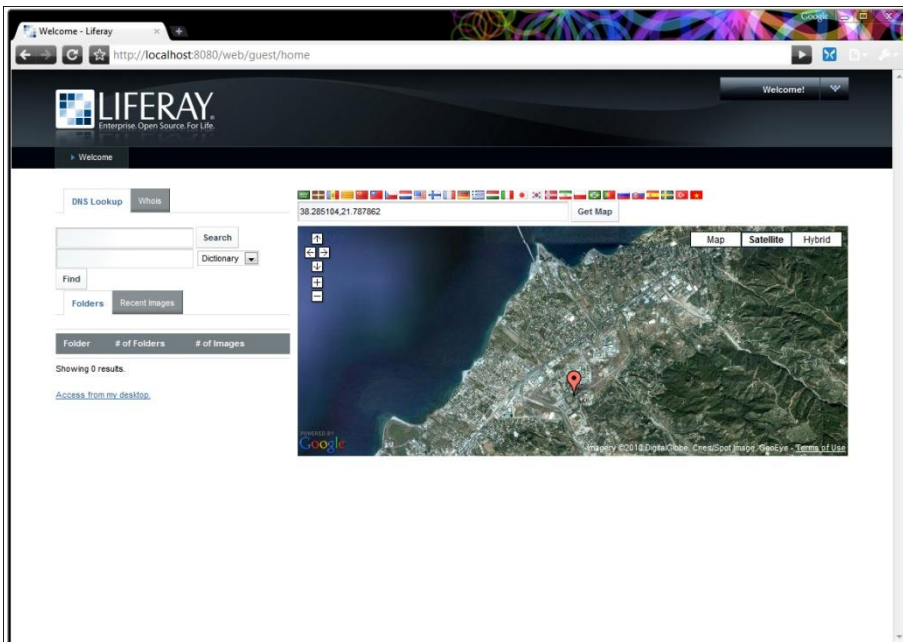
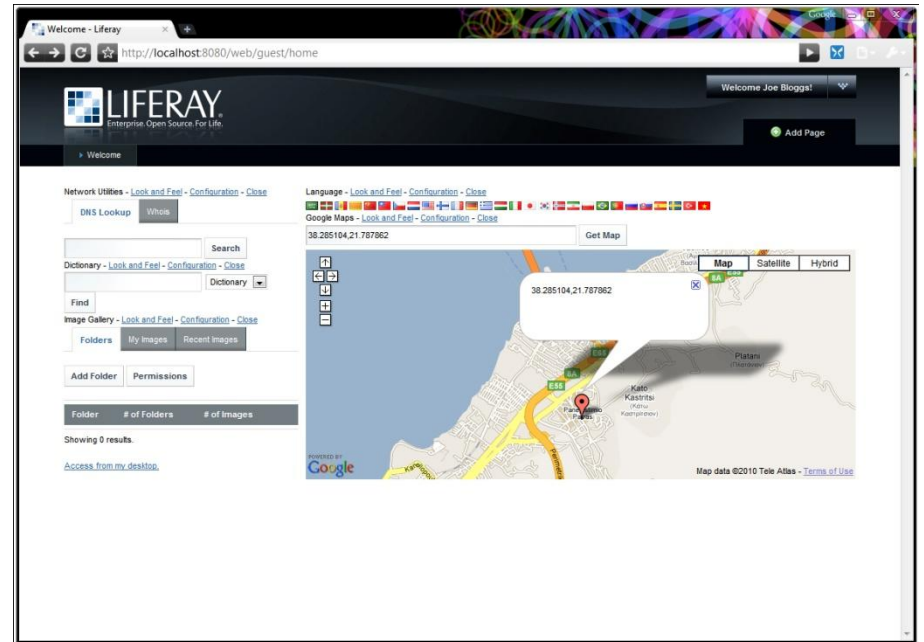
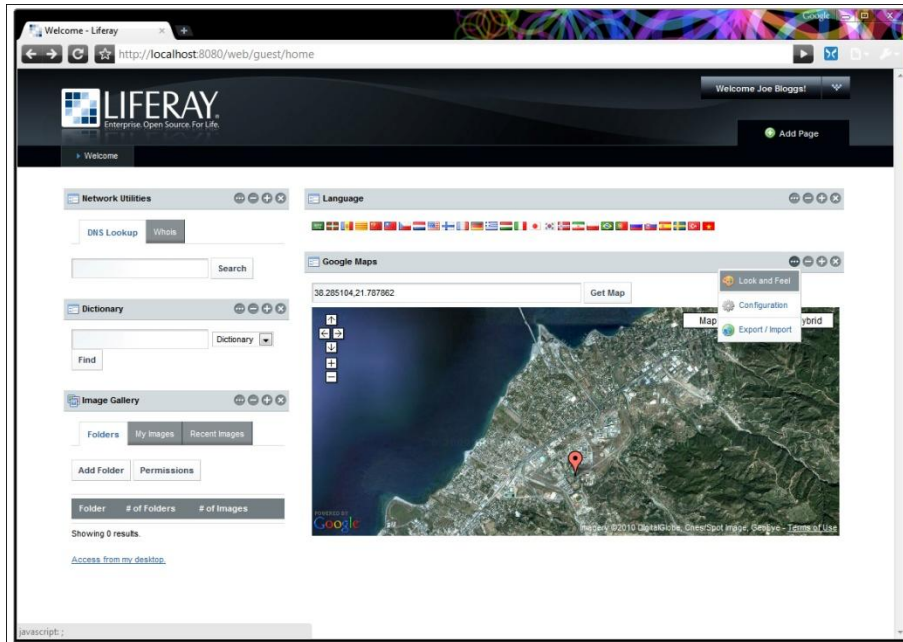
αφού εμφανιστεί ένα μήνυμα "BUILD SUCCESSFUL" από το Ant. Εδώ θα υλοποιηθούν οι λειτουργίες του portlet.

Εναλλακτικά, αν κάποιος δεν χρησιμοποιεί τον προτεινόμενο τρόπο, μπορεί να αντιγράψει το έργο που μόλις δημιουργήθηκε στο περιβάλλον ανάπτυξης της

επιλογής του και να δουλέψει εκεί. Αν γίνει αυτό, πρέπει να είναι σίγουρο πως από το περιβάλλον υπάρχουν οι κατάλληλες αναφορές στα .jar αρχεία της βιβλιοθήκης του Liferay, αλλιώς το πιθανότερο είναι να προκύψουν σφάλματα. Καθώς τα ant scripts το κάνουν αυτόματα, δεν δημιουργούνται τέτοια σφάλματα όταν κάποιος εργάζεται με το Plugins SDK. Για να επιλυθούν τέτοιου είδους εξαρτήσεις, μία λύση βρίσκεται στις εγγραφές του build-common.xml. Συγκεκριμένα, από τις εγγραφές plugin.classpath και portal.classpath καθορίζεται ποιά .jar πακέτα είναι απαραίτητα για το νέο portlet project.

Στις παρακάτω εικόνες παρουσιάζεται η αλλαγή στην εμφάνιση ενός portlet και πώς αυτό επηρεάζει την τελική εμφάνιση ολόκληρης της ιστοσελίδας. Έτσι, αρχικά αφού απενεργοποιηθούν τα όρια του portlet, απενεργοποιούνται και τα εικονίδια στην γραμμή εργασιών κάθε μονάδας. Το αποτέλεσμα είναι οι δομικές αυτές μονάδες να μην μοιάζουν ξεχωριστές, αλλά στον χρήστη να δίνεται η εικόνα μιας ενοποιημένης σελίδας. Προφανώς οι επιλογές στην εμφάνιση δεν περιορίζονται εδώ, αλλά είναι πάρα πολλές, και διαφέρουν ανάλογα με τον ρόλο που έχει ο χρήστης στο σύστημα. Στην τελευταία εικόνα, φαίνεται η οθόνη ρυθμίσεων (edit) του συγκεκριμένου portlet, όπως περιγράφηκε σε προηγούμενη υποενότητα





Portlets

Ανατομία ενός portlet

Ένα portlet αποτελείται απλό τα τρία παρακάτω στοιχεία:

- Τον Java κώδικα.
- Τα αρχεία διαμόρφωσης(Configuration files).
- Τα αρχεία του πελάτη (Client-side files) .

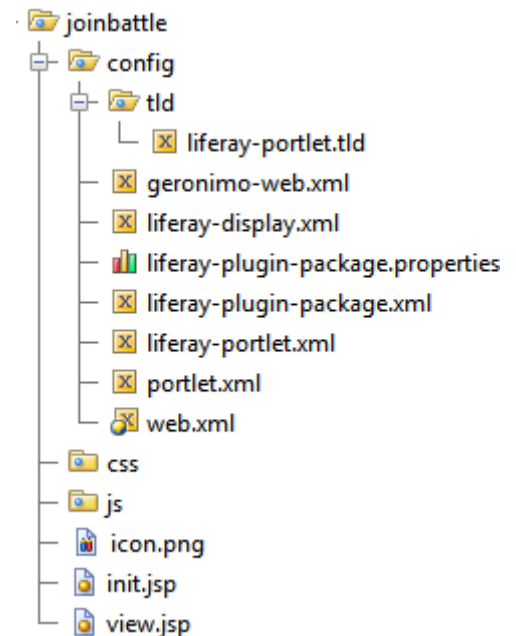
Τα αρχεία αυτά αποθηκεύονται σε μία συγκεκριμένη ιεραρχία, όπως ορίζει το πρότυπο. Το προεπιλεγμένο portlet το οποίο δημιουργείται με την εκτέλεση των εντολών, έχει διαμορφωθεί ως ένα JSR-168 portlet, με διαχωρισμένα JSP για τις τρεις καταστάσεις που περιγράφηκαν παραπάνω (view, edit, help). Μόνο το view.jsp είναι υλοποιημένο στο παράδειγμα, και ο κώδικας θα πρέπει να προσαρμοστεί για να ενεργοποιηθούν και οι υπόλοιπες καταστάσεις.

Ο πηγαίος κώδικας της Java είναι τοποθετημένος στον φάκελο docroot/WEB-INF/src. Κάποιος, μπορεί να πάει εκεί και να προσαρμόσει (και να μετονομάσει) την κλάση ή και να προσθέσει ότι κλάσεις είναι απαραίτητες για την υλοποίηση της απαιτούμενης λειτουργικότητας.

Τα αρχεία διαμόρφωσης είναι τοποθετημένα στον φάκελο docroot/WEB-INF . Υπάρχουν δύο καθιερωμένα αρχεία διαμόρφωσης, τα web.xml και portlet.xml, τα οποία ορίζονται από το πρότυπο JSR-168. Ακόμη υπάρχουν τρία άλλα αρχεία διαμόρφωσης, τα οποία ορίζουν ιδιότητες που αντιστοιχούν στο Liferay. Αυτά δεν είναι απαραίτητα εν γένει, είναι όμως σημαντικά

αν τα portlet τα οποία αναπτύσσονται θα χρησιμοποιηθούν στο Liferay. Τα προαιρετικά αρχεία είναι τα:

- liferay-display.xml: Περιγράφει σε ποια κατηγορία του παραθύρου "Προσθήκη Εφαρμογής" (Add Application) της διαδικτυακής πύλης θα εμφανίζεται το portlet.



Portlets

Ανατομία ενός portlet

- Η `liferay-portlet.xml`: Περιγράφει κάποια προαιρετικά στοιχεία που σχετίζονται με το Liferay και επεκτείνουν τις προδιαγραφές του JSR-168. Για παράδειγμα, κάποιος μπορεί να ορίσει αν το portlet θα αποτελεί στιγμιότυπο (instanceable), αν δηλαδή μπορούν να τοποθετηθούν περισσότερα από ένα αντίγραφα του ίδιου στην ίδια σελίδα διατηρώντας ξεχωριστά δεδομένα ή αν θα έχουν κοινά. Για περισσότερες πληροφορίες, κάποιος θα πρέπει να αναφερθεί στο DTD που περιγράφει το συγκεκριμένο αρχείο, καθώς τα στοιχεία που μπορούν να προσδιοριστούν εδώ είναι πάρα πολλά.
- `liferay-plugin-package.properties`: Περιγράφει την συγκεκριμένη επέκταση στον οδηγό που είναι υπεύθυνος για την εγκατάσταση από την πλευρά του εξυπηρετητή (server's hot deployer). Ένα από τα στοιχεία που μπορούν να περιγραφούν εδώ είναι οι εξαρτήσεις από άλλα αρχεία `.jar`. Αν μία επέκταση για παράδειγμα έχει εξαρτήσεις αρχείων τα οποία βρίσκονται μέσα στο Liferay, τότε αυτά μπορούν να προσδιοριστούν εδώ και ο οδηγός εγκατάστασης (deployer) θα αναλάβει να τροποποιήσει κατάλληλα το τελικό `.war` αρχείο, ώστε να προστεθούν τα κατάλληλα `.jar` αρχεία.

Τα αρχεία του πελάτη είναι `.jsp`, `.css` και Javascript αρχεία τα οποία γράφονται για να υλοποιήσουν την διεπαφή χρήστη του ίδιου του portlet. Αυτά τα αρχεία μπορούν να μπουν οπουδήποτε μέσα στην ιεραρχία του αρχικού φακέλου - είτε δηλαδή στην ρίζα, είτε σε

κάποια δομή φακέλων η οποία θα μπει στην ρίζα. Πρέπει να τονιστεί ξανά πως τα portlet ασχολούνται μόνο με μέρη της σελίδας που θα επιστραφεί στον φυλλομετρητή. Έτσι, δεν θα πρέπει σε καμία περίπτωση να περιλαμβάνονται γενικές ετικέτες (όπως `<html>` ή `<head>` κλπ).

Στην επόμενη ενότητα θα επιχειρηθεί μία βαθύτερη ανάλυση επιλεγμένων συστατικών στοιχείων που αποτελούν ένα portlet, με πραγματικά, υλοποιημένα παραδείγματα από την πλατφόρμα. Αρχικά γίνεται μία παρουσίαση δύο αρχείων XML τα οποία αντιστοιχούν ιδιότητες του συστήματος και έπειτα δίνεται ο κώδικας των αρχείων αυτών. Περισσότερα στοιχεία μπορεί κάποιος να βρει στο κεφάλαιο “Υλοποίηση” της παρούσας διπλωματικής εργασίας.

Portlets

Ανάλυση και Κώδικας

88

docroot/WEB-INF/portlet.xml

- Το στοιχείο `portlet-name` περιέχει το κανονικό όνομα του portlet. Κάθε όνομα portlet πρέπει να είναι μοναδικό, τουλάχιστον μέσα στην εφαρμογή. Έτσι, μέσα σε ένα στιγμιότυπο Liferay, μόνο ένα portlet μπορεί να έχει το συγκεκριμένο όνομα.
- Το στοιχείο `display-name` περιέχει μία εκδοχή μικρού μήκους για το όνομα, η οποία και θα φαίνεται από τα διάφορα εργαλεία. Δεν είναι απαραίτητο να είναι μοναδικό.
- Το στοιχείο `portlet-class` περιέχει το πλήρες όνομα της κλάσης ενός portlet.
- Το στοιχείο `init-param` περιέχει ένα ζευγάρι όνομα-τιμή, ως μεταβλητή αρχικοποίησης του portlet.
- Το στοιχείο `expiration-cache` ορίζει πρακτικά τον χρόνο σε δευτερόλεπτα από τον οποίο και έπειτα λήγει η έξοδος του portlet. Η τιμή `-1` σημαίνει πως η έξοδος δεν λήγει ποτέ.
- Το στοιχείο `supports` περιέχει τον υποστηριζόμενο τύπο mime. Επίσης, καταδεικνύει τις καταστάσεις τις οποίες υποστηρίζει ένα portlet για ένα συγκεκριμένο είδος περιεχομένου. Όλα πρέπει να υποστηρίζουν την κατάσταση view.

- Το στοιχείο `portlet-info` ορίζει πληροφορίες που αφορούν το portlet.
- Το στοιχείο `security-role-ref` ορίζει τους ρόλους ασφαλείας στον κώδικα της εφαρμογής ιστού. Ειδικά για το Liferay, οι τα ονόματα-ρόλοι αναφέρονται στο ποι οι ρόλοι μπορούν να προσπελάσουν το portlet (ένας υπερχρήστης μπορεί να παραμετροποιήσει την διαδικτυακή πύλη, ενώ ένας χρήστης δεν μπορεί).

docroot/WEB-INF/liferay-portlet.xml

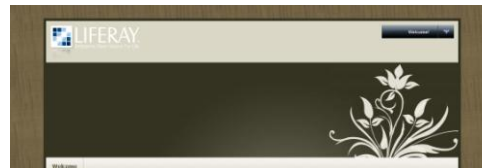
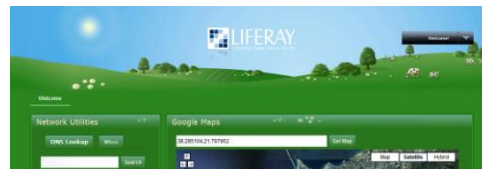
- Το στοιχείο `portlet-name` περιέχει το κανονικό όνομα ενός portlet. Αυτό πρέπει να είναι το ίδιο με το όνομα που δίνεται στο `portlet.xml`
- Το στοιχείο `icon` περιέχει το μονοπάτι για το εικονίδιο του.
- Το στοιχείο `instanceable` καταδεικνύει αν πολλά στιγμιότυπα του συγκεκριμένου portlet μπορούν να ανήκουν στην ίδια σελίδα.
- Το στοιχείο `struts-path` λέει στο portal ότι όλες οι αιτήσεις με το μονοπάτι `ext/library/*` θεωρούνται μέρος της σκοπιάς του portlet. Για παράδειγμα ένα μονοπάτι `struts` του τύπου `ext/library` θα δώσει πρόσβαση στο `/ext/library/view` στο `struts-config.xml` αλλά όχι στο `/ext/reports/view`.

Θέματα

Τα τροποποιημένα θέματα (themes) βασίζονται στις διαφορές μεταξύ του τροποποιημένου κώδικα και των προεγκατεστημένων θεμάτων του Liferay, που αλλιώς ονομάζονται και "κλασσικά". Κάθε ενδιαφερόμενος για την δημιουργία θεμάτων παρατηρεί πώς υπάρχει ένας φάκελος `_diffs` στον φάκελο τροποποιημένων (Custom) θεμάτων. Σε αυτό το σημείο τοποθετείται ο κώδικας του θέματος. Τροποποίηση χρειάζονται μόνο εκείνα τα μέρη του θέματος που θέλουμε να διαφέρουν από τα ήδη υπάρχοντα κλασσικά θέματα. Για να επιτευχθεί αυτό, δημιουργείται ένα ακριβές αντίγραφο της δομής των φακέλων των κλασσικών θεμάτων μέσα στον φάκελο `_diffs`, και τοποθετείται σε αυτό μόνο το σύνολο των φακέλων και των αρχείων που είναι απαραίτητα για την τροποποίηση.

Για παράδειγμα, για να τροποποιήσουμε το Dock (ένα απαραίτητο στοιχείο όλων των θεμάτων), θα αντιγράψαμε μόνο το αρχείο `dock.vtm` από το Liferay (Το κλασσικό θέμα βρίσκεται στο `<Tomcat_Home>/webapps/ROOT/html/themes/classic`) μέσα στον φάκελο των θεμάτων `_diffs/templates`. Έπειτα ανοίγουμε αυτό το αρχείο και το τροποποιούμε όπως επιθυμούμε. Για παράδειγμα, ίσως θελήσουμε να αλλάξουμε το μήνυμα καλωσορίσματος σε κάτι σαν, "Σύνδεσμοι." Για την τροποποίηση των Διαδοχικών Φύλλων Στυλ (Cascading Style Sheets – CSS), προτείνεται η δημιουργία ενός φακέλου `css` και η τοποθέτηση ενός μοναδικού αρχείου μέσα σε αυτόν με το όνομα `custom.css`. Σε αυτόν τον φάκελο θα τοποθετούνται όλα τα νέα styles καθώς και όλα τα τροποποιημένα, με βάση τα προεγκατεστημένα στυλ του Liferay. Είναι καλύτερο να ακολουθηθεί ο συγκεκριμένος τρόπος εξαιτίας του τρόπου που τα αρχεία `.css` φορτώνονται. Το `Custom.css` φορτώνεται τελευταίο, έτσι οτιδήποτε περιέχεται μέσα σε αυτό το

αρχείο να είναι σίγουρο ότι θα επικαλύψει κάθε άλλο στυλ που υπάρχει σε οποιοδήποτε από τα υπόλοιπα Φύλλα Στυλ.



Θέματα

Ανατομία ενός Θέματος

Οι φάκελοι στα θέματα σχεδιάζονται ώστε να είναι εύκολοι στην πλοήγηση και την κατανόηση. Η τρέχουσα δομή του νέου φακέλου είναι δίδεται παρακάτω. Όπως φαίνεται, η δομή είναι σε μορφή Αρχείου Εφαρμογής Διαδικτύου (Web Application Archive - WAR). Οποιοδήποτε από αυτά τα αρχεία μπορεί να αντιγραφεί από το προεγκατεστημένο θέμα στον φάκελο _diffs έτσι ώστε να υποστεί τροποποίηση.

```
/THEME_NAME/  
  /css/  
    base.css, custom.css, main.css, navigation.css,  
    forms.css,  
    portlet.css, deprecated.css, tabs.css, layout.css  
  /images/  
    (many directories)  
  /javascript/  
    javascript.js  
  /templates/  
    dock.vm  
    navigation.vm  
    portal_normal.vm  
    portal_popup.vm  
    portlet.vm  
  /WEB-INF  
  /META-INF
```

Παρακάτω δίνεται μία συνοπτική παρουσίαση των ξεχωριστών κομματιών που αφορούν την δημιουργία θεμάτων στο Liferay. Στον οπτικό δίσκο ο οποίος συνοδεύει την διπλωματική εργασία παρατίθεται ο κώδικας που υλοποιεί το θέμα της εργασίας.

JavaScript

Το Liferay πλέον περιέχει την βιβλιοθήκη jQuery JavaScript, και οι προγραμματιστές θεμάτων μπορούν να συμπεριλάβουν οποιαδήποτε πρόσθετα υποστηρίζει η jQuery. Όμως η \$ μεταβλητή, δεν υποστηρίζεται (για την επίτευξη μεγαλύτερης συμβατότητας με διαφορετικά portlets). Μέσα στο αρχείο javascript.js, υπάρχουν τρεις διαφορετικές κλήσεις συναρτήσεων, σαν τις παρακάτω:

Κώδικας:

```
jQuery(document).ready(  
    function() {  
        //Custom javascript goes here  
    }  
);  
Liferay.Portlet.ready(  
    function(portletId, jqueryObj) {  
        //Custom javascript goes here  
    }  
);
```

Θέματα

Ανατομία ενός Θέματος

```
jQuery(document).last(
function() {
    //Custom javascript goes here
}
);
```

jQuery(document).ready(fn): Όταν αυτή πάρει ως όρισμα μία συνάρτηση (μπορεί να είναι μια ορισμένη συνάρτηση, ή μία ανώνυμη όπως αυτή παραπάνω), η συνάρτηση εκτελείται αφού η HTML σελίδα έχει τελειώσει την διαδικασία της φόρτωσης (εκτός κι αν τα portlets φορτώνονται με την χρήση ajax).

Liferay.Portlet.ready(fn): Όταν αυτή πάρει ως όρισμα μία συνάρτηση (μπορεί να είναι μια ορισμένη συνάρτηση, ή μία ανώνυμη όπως αυτή παραπάνω), η συνάρτηση εκτελείται μετά την φόρτωση κάθε portlet. Η εκτελούμενη συνάρτηση λαμβάνει δύο μεταβλητές, την portletId και την jQueryObj. Η portletId είναι το id του τρέχοντος portlet που μόλις έχει φορτωθεί, και η jQueryObj είναι το jQuery object του τρέχοντος αντικειμένου του portlet.

jQuery(document).last(fn): Όταν αυτή πάρει ως όρισμα μία συνάρτηση (μπορεί να είναι μια ορισμένη συνάρτηση, ή μία ανώνυμη όπως αυτή παραπάνω), η συνάρτηση εκτελείται αφού όλα τα portlets — συμπεριλαμβανομένων και αυτών που έχουν υλοποιηθεί με χρήση AJAX — έχουν φορτωθεί στην σελίδα.

Έκτος από την JavaScript που αφορά το ολόκληρο το θέμα (theme-wide) υπάρχει υποστήριξη και για

JavaScript που αφορά συγκεκριμένες σελίδες (page specific). Η φόρμα ρυθμίσεων της σελίδας παρέχει τρία ξεχωριστά κομμάτια JavaScript που μπορούν να εισαχθούν οπουδήποτε στο εκάστοτε θέμα. Το παρακάτω χρησιμοποιείται ώστε να συμπεριληφθεί ο κώδικας αυτών των ρυθμίσεων:

Κώδικας:

```
$layout.getTypeSettingsProperties().get
Property("javascript-1")
$layout.getTypeSettingsProperties().get
Property("javascript-2")
$layout.getTypeSettingsProperties().get
Property("javascript-3")
```

Το περιεχόμενο των πεδίων των ρυθμίσεων για την JavaScript είναι αποθηκευμένο στην βάση δεδομένων ως "Java Properties". Αυτό σημαίνει ότι κάθε πεδίο μπορεί να περιέχει μόνο μια γραμμή κειμένου. Για κείμενα πολλών γραμμών, κάθε νέα γραμμή θα ξεκινάει με τον χαρακτήρα \, όπως σε κάθε συνηθισμένο αρχείο .properties.

Ρυθμίσεις

Κάθε θέμα μπορεί να ορίσει ένα σύνολο από ρυθμίσεις που να το καθιστούν τροποποιήσιμο. Οι ρυθμίσεις καθορίζονται στο liferay-look-and-feel.xml

91

Θέματα

Ανατομία ενός Θέματος

χρησιμοποιώντας την ακόλουθη σύνταξη:

Κώδικας:

```
<settings>
  <setting key="my-setting"
    value="my-value" />
...
</settings>
```

Αυτές οι ρυθμίσεις μπορούν να προσπελαστούν στα πρότυπα θεμάτων(templates) χρησιμοποιώντας τον ακόλουθο κώδικα:

Κώδικας:

```
$theme.getSetting("my-setting")
```

Για παράδειγμα, ας πούμε ότι θέλουμε να δημιουργήσουμε δύο θέματα που να είναι ακριβώς ίδια με εξαίρεση κάποιες αλλαγές στην επικεφαλίδα(header). Ένα από τα θέματα έχει περισσότερες λεπτομέρειες ενώ το άλλο είναι μικρότερο (και καταλαμβάνει μικρότερο χώρο στην οθόνη). Αντί να δημιουργήσουμε δύο διαφορετικά θέματα, θα δημιουργήσουμε μόνο ένα και θα χρησιμοποιήσουμε μια ρύθμιση για να διαλέξουμε την επικεφαλίδα(header) που θέλουμε. Ενώ προγραμματίζουμε το θέμα φτάνουμε στην επικεφαλίδα(header). Στο portal_normal.vm πρότυπο(template) γράφουμε:

Κώδικας:

```
if ($theme.getSetting("header-type")
== "detailed") {
#parse("$full_templates_path/header_detailed.vm")
} else {
#parse("$full_templates_path/header_brief.vm")
}
```

Έπειτα γράφουμε το liferay-look-and-feel.xml, δίνουμε δύο διαφορετικές εισόδους που να αναφέρονται στο ίδιο θέμα αλλά να έχουν διαφορετική τιμή για την ρύθμιση που αφορά την επικεφαλίδα(header-type):

Κώδικας:

```
<theme id="beauty1" name="Beauty 1">
  <root-path>/html/themes/beauty</root-path>
  <templates-path>${root-path}/templates</templates-path>
  <images-path>${root-path}/images</images-path>
  <template-extension>vm</template-extension>
  <settings>
    <setting key="header-type" value="detailed"/>
  </settings>
  <color-scheme id="01" name="Blue">
    <css-class>blue</css-class>
    <color-scheme-images-path>
```

Θέματα

Ανατομία ενός Θέματος

```
    ${images-
path}/color_schemes/${css-class}
    </color-scheme-images-path>
    Creating Liferay Themes 15
    Hot Deploy Development
    </color-scheme>
    ...
</theme>
<theme id="beauty2" name="Beauty 2">
<root-path>/html/themes/beauty</root-path>
<templates-path>${root-
path}/templates</templates-path>
<images-path>${root-path}/images</images-
path>
<template-extension>vm</template-
extension>
<settings>
<setting key="header-type" val-
ue="brief"/>
</settings>
<color-scheme id="01" name="Blue">
<css-class>blue</css-class>
<color-scheme-images-path>
    ${images-
path}/color_schemes/${css-class}
    </color-scheme-images-path>
</color-scheme>
    ...
</theme>
```

Συνδυασμός Χρωμάτων

Οι συνδυασμοί χρωμάτων ορίζονται από το όνομα μιας CSS κλάσης, με την οποία όχι μόνο μπορεί κάποιος να αλλάξει χρώματα, αλλά και να επιλέξει διαφορετικές εικόνες ως φόντο, διαφορετικά χρώματα για τα περιγράμματα κ.ο.κ. Στο liferay-look-and-feel.xml (που βρίσκεται στον WEB-INF), ένα όνομα κλάσης μπορεί να οριστεί ως εξής:

Κώδικας:

```
<theme id="my_theme" name="My Theme">
  <root-path>/my_theme</root-path>
  <templates-path>${root-
path}/templates</templates-path>
  <images-path>${root-
path}/images</images-path>
  <template-extension>vm</template-
extension>
  <color-scheme id="01" name="Blue">
    <css-class>blue</css-class>
    <color-scheme-images-path>
      ${images-
path}/color_schemes/${cssclass}
    </color-scheme-images-path>
    </color-scheme>
  <color-scheme id="02" name="Green">
    <css-class>green</css-class>
    </color-scheme>
  </theme>
```

Μέσα στον φάκελο `css`, δημιουργούμε έναν φάκελο που να ονομάζεται `color_schemes`. Μέσα σε αυτόν τον φάκελο, τοποθετούμε ένα αρχείο `.css` για κάθε συνδυασμό χρωμάτων. Στην παραπάνω περίπτωση, θα μπορούσαμε είτε να έχουμε ένα με το όνομα `green.css` και να αφήσουμε το αρχικό (προ εγκατεστημένο) `style` να διαχειρίζεται τον πρώτο συνδυασμό χρωμάτων, ή θα μπορούσαμε να έχουμε και το `blue.css` και το `green.css`. Έπειτα, μέσα στο `custom.css`, τοποθετούμε τις ακόλουθες γραμμές:

Κώδικας:

```
@import url(color_schemes/blue.css);
@import url(color_schemes/green.css);
```

Θέματα

Ανάλυση και Κώδικας

Για να ξεχωρίζουμε τα στυλ μεταξύ των CSS χρησιμοποιούμε προθέματα. Στο blue.css βάζουμε μπροστά από όλα τα css styles το πρόθεμα blue ως εξής:

Κώδικας:

```
.blue a {color: #06C;}  
.blue h1 (border-bottom: 1px solid  
#06C}
```

Και στο green.css βάζουμε μπροστά από όλα τα css styles το πρόθεμα green ως εξής:

Κώδικας:

```
.green a {color: #06C;}  
.green h1 (border-bottom: 1px solid  
#06C}
```

Το στοιχείο struts-path λέει στο portal ότι όλες οι αιτήσεις με το μονοπάτι ext/library/* θεωρούνται μέρος της σκοπιάς του portlet. Για παράδειγμα ένα μονοπάτι struts του τύπου ext/library θα δώσει πρόσβαση στο /ext/library/view στο struts-config.xml αλλά όχι στο /ext/reports/view

Προκαθορισμένες Ρυθμίσεις:

Το portal καθορίζει κάποιες ρυθμίσεις οι οποίες επιτρέπουν στο θέμα να έχει συγκεκριμένες συμπεριφορές. Μέχρι στιγμής υπάρχουν μόνο δύο προκαθορισ-

μένες ρυθμίσεις αλλά ο αριθμός τους είναι πιθανό να αυξηθεί στο μέλλον.

portlet-setup-show-borders-default: Εάν είναι ορισμένη ως ψευδής, τα portlets του portal δεν θα έχουν πλαίσια(borders). Η προκαθορισμένη τιμή αυτής της ρύθμισης είναι αληθής.

Παράδειγμα:

```
<settings>  
<setting key="portlet-setup-show-  
borders-default" value="false" />  
</settings>
```

Αυτή η προκαθορισμένη συμπεριφορά μπορεί να επικαλυφθεί για συγκεκριμένα portlets χρησιμοποιώντας τα:

- liferay-portlet.xml
- Portlet CSS popup setting

bullet-style-options: Η τιμή πρέπει να είναι μία λίστα, από διαχωρισμένα με κενό, έγκυρα styles κουκκίδων(bullet). Η προκαθορισμένη τιμή είναι κενή λίστα(empty list). Το portlet πλοήγησης δε θα δείξει καμία επιλογή στην οθόνη διαμόρφωσης του portlet.

Παράδειγμα:

```
<settings>  
<setting key="bullet-style-  
options"
```

Θέματα

Ανατομία ενός Θέματος

```
value="classic,cool,tablemenu" />
</settings>
```

Τα styles κουκίδων(bullet) που αναφέρονται στην ρύθμιση, καθορίζονται σε κάθε αρχείο CSS του θέματος ακολουθώντας το ακόλουθο υπόδειγμα:

Κώδικας:

```
.nav-menu-style-{BULLET_STYLE_OPTION}
{
... CSS selectors ...
}
```

Παρακάτω παρουσιάζουμε ένα παράδειγμα του HTML κώδικα που χρειάζεται για να δημιουργήσουμε κάποιο style με την χρήση CSS κώδικα. Σε αυτήν την περίπτωση το style κουκίδας (bullet):

Κώδικας:

```
<div class="nav-menu nav-menu-style-cool">
  <h3>
    <a
href="/web/guest/community">Community</a>
  </h3>
  <ul class="layouts">
    <li class="">
      <a class=""
href="/web/guest/community/documentation">
Documentation
      </a>
    </li>
    <li class="">
```

```
      <a class=""
href="http://wiki.liferay.com" tar-
get="_blank">
        wiki
      </a>
    </li>
    <li class="">
      <a class=""
href="/web/guest/community/forums">Forums<
/a>
    </li>
  </ul>
</div>
```

Ανάπτυξη

Αυτό το κομμάτι αποτελεί έναν περιληπτικό οδηγό για την δημιουργία επεκτάσεων πάνω στον πυρήνα του συστήματος. Όταν μιλάμε για επεκτάσεις, εννοούμε είτε μονάδες (portlets) είτε θέματα (themes). Οι επεκτάσεις είναι ο προτεινόμενος τρόπος για να προσθέσει κάποιος λειτουργίες στο Liferay, καθώς έχουν πολλά πλεονεκτήματα:

- Μπορούν να συντεθούν από πολλά μικρότερα portlet ή θέματα. Κάτι τέτοιο μειώνει την πολυπλοκότητα των ατομικών έργων, επιτρέποντας στους προγραμματιστές να διαχωρίσουν ευκολότερα την λειτουργικότητα του έργου.
- Είναι εντελώς ανεξάρτητες από τον πυρήνα του Liferay. Portlets τα οποία πειθαρχούν στο πρότυπο JSR-168 μπορούν να εγκατασταθούν σε οποιοδήποτε portlet container.
- Μπορούν να εγκατασταθούν σε πραγματικό χρόνο (hot deploy), δηλαδή ενώ ο διακομιστής τρέχει, και είναι προσβάσιμα άμεσα. Αυτό είναι πολύ σημαντικό πλεονέκτημα, καθώς για να προστεθούν νέες λειτουργίες δεν απαιτείται να "πέσει" ο δικτυακός τόπος για κάποια περίοδο την οποία μπορεί να τον προσπελαύνουν χρήστες.

Υπάρχουν πολλοί τρόποι για να δημιουργήσει κάποιος μονάδες επέκτασης για το Liferay. Πολλά Ενσωματωμένα Περιβάλλοντα Ανάπτυξης (Integrated Development Environment - IDE) υποστηρίζουν εγγενώς την ανάπτυξη portlets, και η δημιουργία θεμάτων δεν είναι κάτι άλλο από κλασικές μονάδες ιστού

(web modules) με την προσθήκη διαδοχικών φύλλων στυλ (cascading style sheets - css), εικόνων, και πιθανότατα Javascript κώδικα μέσα τους. Για τον λόγο αυτό, υπάρχει πληθώρα εργαλείων για την δημιουργία επεκτάσεων, από απλούς κειμενογράφους έως υπερπλήρη ενσωματωμένα περιβάλλοντα ανάπτυξης. Κάθε χρήστης λοιπόν ενδείκνυται να χρησιμοποιεί το εργαλείο στο οποίο είναι συνηθισμένος. Επειδή η κοινότητα του Liferay καταβάλλει προσπάθεια για να διατηρήσει το εργαλείο αγνωστικιστικό, παρέχεται ένα Πακέτο Ανάπτυξης Λογισμικού Επεκτάσεων (Plugins Software Development Kit) το οποίο μπορεί να χρησιμοποιήσει κανείς για να δημιουργήσει τόσο portlets όσο και θέματα. Αυτό το πακέτο μπορεί να χρησιμοποιηθεί με οποιονδήποτε κειμενογράφο ή IDE για την δημιουργία επεκτάσεων για το Liferay. Έτσι, ενώ δεν είναι απαραίτητη η χρήση του, είναι καθόλα προτεινόμενη.

Να σημειωθεί πως είναι απαραίτητο να έχει εγκατασταθεί το Apache Ant 1.7.0 (ή νεότερο). Για την εγκατάσταση κάποιος μπορεί να κατεβάσει το ant από το <http://ant.apache.org>, να το αποσυμπίσει και να το τοποθετήσει σε κάποιον φάκελο της επιλογής του. Έπειτα, θέτει μία μεταβλητή συστήματος ANT_HOME που δείχνει στον συγκεκριμένο φάκελο, ανάλογα με το λειτουργικό σύστημα. Επίσης είναι απαραίτητο να υπάρχει εγκατεστημένο το Liferay στο σύστημα του χρήστη για να είναι δυνατή η εγκατάσταση και το αξιολόγηση των επεκτάσεων. Συνίσταται η χρήση του πακέτου Liferay-Tomcat, καθώς ο Tomcat είναι μικρός, γρήγορος και καταναλώνει λιγότερους πόρους από τις υπόλοιπες επιλογές. Η εγκατάσταση γίνεται με παρόμοιο τρόπο: κατέβασμα του πακέτου, αποσυμπίεση

Ανάπτυξη

Εγκατάσταση

και τοποθέτηση στον επιθυμητό φάκελο. Έπειτα, η εκκίνηση γίνεται από την αντίστοιχη εντολή για το κάθε λειτουργικό στον bin φάκελο του αρχικού φακέλου του Tomcat. (δηλαδή startup.bat για Windows ή ./startup.sh για Unix, Linux ή Mac).

Το στήσιμο του SDK για τις μονάδες επέκτασης είναι πολύ απλό. Αρχικά, κάποιος κατεβάζει το πακέτο από την ιστοσελίδα του Liferay (συγκεκριμένα από το <http://www.liferay.com/web/guest/downloads/additional>). Έπειτα, αποσυμπίεζει το πακέτο εκεί που επιθυμεί. Οι δύο φάκελοι που θα χρησιμοποιηθούν περισσότερο είναι ο φάκελος portlet και themes. Επίσης, στον φάκελο υπάρχει ένα αρχείο ονόματι build.properties, το οποίο περιέχει τις ρυθμίσεις για το που είναι εγκατεστημένο το Liferay στο σύστημα του χρήστη, που θα είναι ο φάκελος εγκατάστασης των επεκτάσεων καθώς και άλλες ρυθμίσεις. Επειδή δεν πρέπει να γίνει επεξεργασία του συγκεκριμένου αρχείου, στον ίδιο φάκελο ο χρήστης δημιουργεί ένα αρχείο build.\${user.name}.properties, όπου \${user.name} είναι η ταυτότητα του συγκεκριμένου χρήστη στο σύστημα. Για παράδειγμα αν το όνομα του χρήστη είναι nvasilakis τότε το αρχείο θα έπρεπε να λέγεται build.nvasilakis.properties. Οι κύριες ιδιότητες οι οποίες θα πρέπει να παραμετροποιηθούν είναι η app.server.dir (η οποία αντιστοιχεί στον φάκελο όπου έχει εγκατασταθεί ο διακομιστής εφαρμογών - application server- στην περίπτωση μας ο Tomcat) και η java.compiler (η οποία αντιστοιχεί στον προεπιλεγμένο μεταγλωττιστή Java του συστήματος).

Το SDK επεκτάσεων είτε μπορεί να χρησιμοποιηθεί για να φιλοξενήσει όλα τα έργα που αφορούν την δημιουργία portlet και θεμάτων, είτε μπορεί να χρησιμοποιηθεί ξεχωριστό SDK για κάθε έργο (project). Για παράδειγμα, αν υπάρχει ένα ιδιωτικό ενδοδίκτυο (intranet) το οποίο χρησιμοποιεί Liferay και υπάρχουν κάποια προσαρμοσμένα portlets για ιδιωτική χρήση, τότε καλό είναι αυτά να έχουν το δικό τους SDK το καθένα. Αν υπάρχει και ένα εξωτερικό στιγμιότυπο του Liferay το οποίο χρησιμοποιείται για να φιλοξενεί μία δημόσια ιστοσελίδα, μπορεί να χρησιμοποιηθεί ένα SDK για όλα τα themes και τα portlets σε αυτό. Ή μπορεί να χρησιμοποιηθεί κάτι σαν την προηγούμενη διάταξη. Επίσης, κάποιος μπορεί να χρησιμοποιήσει το SDK ως μία απλή γεννήτρια έργων ανεξάρτητη της πλατφόρμας: Μπορεί να γίνει χρήση των ant σεναρίων του SDK για την δημιουργία του έργου, και έπειτα η αντιγραφή των αρχείων του έργου από τον φάκελο portlet ή themes στον φάκελο του IDE που θα χρησιμοποιηθεί. Ενδεχομένως να χρειαστούν μικρές αλλαγές στα σενάρια ant για κάτι τέτοιο, αλλά αυτό το σενάριο αποτελεί πανάκεια όταν γίνεται χρήση αυστηρών προτύπων από εταιρίες ή οργανισμούς όσον αφορά την διαδικασία παραγωγής λογισμικού.

The most exciting phrase to hear in science, the one that heralds new discoveries,
is not 'Eureka!' but 'That's funny ...'.

Isaac Asimov (author and professor of biochemistry, 1920 – 1992)

Fun in Numbers

Εισαγωγή

100

Τα τελευταία χρόνια υπάρχει μια γοργά αναπτυσσόμενη τάση για την διασύνδεση μεταξύ ψηφιακής και φυσικής πραγματικότητας. Αυτό καταδεικνύεται από την σημαντική δραστηριότητα που επιτελείται στον ερευνητικό τομέα των ασύρματων δικτύων αισθητήρων (Wireless Sensor Networks - WSN) καθώς και την συνεχή ενσωμάτωση συσκευών αισθητήρων σε πολλαπλά είδη εφαρμογών. Παρότι χρησιμοποιούμε τους αισθητήρες με ολοένα και περισσότερους τρόπους, στην πραγματικότητα ίσα που έχουμε αγγίζει την κορυφή του παγόβουνου όσον αφορά τις χρήσεις των αισθητήρων στους τομείς της διασκέδασης και συγγενικών με την ψυχαγωγία τομέων. Το παιχνίδι είναι μια περιοχή εφαρμογών η οποία δεν έχει λάβει την απαραίτητη προσοχή μέχρι σήμερα, τουλάχιστον από την σκοπιά των ασύρματων δικτύων αισθητήρων. Παρότι έχουν προταθεί διάφοροι τρόποι χρήσης των WSN, λίγοι έχουν να κάνουν με κινητά, διαδραστικά παιχνίδια πολλών παικτών (multiplayer) όπου οι χρήστες μπορούν, μεταφέροντας συσκευές που διαθέτουν αισθητήρες, να αλληλεπιδράσουν μεταξύ τους καθώς και με το περιβάλλον τους.

Τα παιχνίδια, τις τελευταίες δεκαετίες αποτελούν αναμφίβολα ένα από τα μεγαλύτερα κομμάτια της βιομηχανίας των υπολογιστών και σίγουρα οδηγούν την τεχνολογία ως ένα βαθμό ανεβάζοντας τον πήχη τόσο στον τομέα του λογισμικού όσο και σε αυτόν του υλικού. Πρόσφατα στην κινητή τηλεφωνία εισήχθησαν

νέα προϊόντα τα οποία ενσωματώνουν αισθητήρες στο επίπεδο των συσκευών. Το 2008 το σύνολο των εγγεγραμμένων χρηστών κινητών τηλεφώνων ξεπέρασε τα 3 δισεκατομμύρια, με αποτέλεσμα να υπάρχει ένας τεράστιος αριθμός χρηστών οι οποίοι πιθανότατα να χρησιμοποιούν τις συσκευές αυτές για να παίζουν. Σίγουρα υπάρχουν πολλά οφέλη για την παραγωγή νέων συναρπαστικών εφαρμογών από τον συνδυασμό αισθητήρων με τις κινητές συσκευές. Η χρήση των αισθητήρων σε κονσόλες παιχνιδιών όπως στο Nintendo Wii έχει ήδη αποδειχτεί μεγάλη επιτυχία. Επίσης υπάρχει μια επιπλέον τάση για απελευθέρωση από τις παραδοσιακές μεθόδους παιχνιδιών, η οποία φανερώνεται και από την μεγάλη επιτυχία που παρουσιάζουν τα Sony PSP και Nintendo DS καθώς και συσκευές κινητών τηλεφώνων που παρέχουν σύνδεση στο διαδίκτυο, όπως είναι το iPhone. Το γεγονός είναι ένα: Οι άνθρωποι θέλουν να αλληλεπιδρούν μεταξύ τους οποιαδήποτε ώρα και σε οποιοδήποτε μέρος.

Πρόσφατα, τα πανταχού παρόντα παιχνίδια (pervasive games) έκαναν την εμφάνισή τους ως ένα καυτό θέμα συζήτησης, παρότι βέβαια ο όρος συχνά χρησιμοποιείται για εντυπωσιασμό, καταλήγει να μην έχει την ίδια σημασία για όλους τους ανθρώπους. Έτσι καθίσταται δύσκολο να αποδοθεί η έννοια και ιδιαίτερα το ύφος με το οποίο χρησιμοποιείται όσον αφορά την ερευνητική δραστηριότητα. Εδώ θα επικεντρωθούμε στο πως μπορούμε να συνδυάσουμε τα παιχνίδια

Εισαγωγή

αυτά με τους αισθητήρες και πως μπορούμε να παρέχουμε αποδοτικές λύσεις για κάποια από αυτά. Μερικά από τα πιο σημαντικά θέματα είναι:

- Οι δυσκολίες σύνδεσης σε συσκευές επικοινωνίας όπως είναι οι συχνές αποσυνδέσεις, η συμφόρηση, η μεταβλητή εκπομπή και άλλα.
- Εφόσον ο αριθμός των συμμετεχόντων - παικτών θα είναι αρκετά μεγάλος πρέπει να ληφθεί υπόψη η μετάβασή τους σε πολύ μεγαλύτερη κλίμακα καθώς και η διαχείριση αυτής.
- Η κατανομημένη φύση των παιχνιδιών απαιτεί τον *συγχρονισμό* των κόμβων, ο οποίος θα να γίνει με αποτελεσματικό και διαφανή τρόπο.
- Ένα επίπεδο συντονισμού θα πρέπει να υπάρχει για να διασφαλίσει την σωστή διαχείριση του παιχνιδιού και την τήρηση των κανόνων οι οποίοι θα είναι γνωστοί στους παίκτες.
- Ένα είδος υποδομής “σπονδυλικής στήλης” θα πρέπει να εισαχθεί για να υποστηρίξει την λειτουργία του παιχνιδιού και να αποδίδει περισσότερα χαρακτηριστικά (συναίσθηση χώρου και συμφραζομένων – location and context awareness)

Απαιτείται ένα πλαίσιο για την διαχείριση αυτών των θεμάτων με ένα τρόπο ομοιόμορφο που δεν θα αποπροσανατολίζει τον προγραμματιστή από τον σκοπό του, ο οποίος είναι ο σχεδιασμός, η ανάπτυξη και ο έλεγχος συναρπαστικών και διασκεδαστικών παιχνιδιών. Η γνώση η οποία προέρχεται από τις πρόσφατες έρευνες σχετικές με τα ασύρματα δίκτυα

αισθητήρων, μπορεί να εφαρμοστεί σε νέα πλαίσια παιχνιδιών για την δημιουργία αποδοτικών συστημάτων. Πολλά από αυτά τα θέματα επιλύονται ή απαντώνται έως ένα βαθμό με την χρήση της συγκεκριμένης πλατφόρμας η οποία αποτελεί ένα πλαίσιο για την δημιουργία, ανάπτυξη και διαχείριση παιχνιδιών με την χρήση ασύρματων κόμβων αισθητήρων ως συσκευές παιχνιδιού. Το κύριο χαρακτηριστικό αυτών των παιχνιδιών είναι ότι οι παίκτες αλληλεπιδρούν μεταξύ τους καθώς και με το περιβάλλον, είτε κινούμενοι, είτε τρέχοντας είτε με την χρήση κινήσεων για να κάνουν ενέργειες σχετικές με το παιχνίδι. Η πλατφόρμα αυτή τη στιγμή χρησιμοποιεί συσκευές Sun SPOT για την πρότυπη υλοποίηση και παρέχει στους προγραμματιστές ουσιαστικά ένα σύνολο από πρότυπα και υπηρεσίες για την κατασκευή και διαχείριση νέων παιχνιδιών. Αυτά μπορούν να προσαρμοστούν με σημαντική ευκολία, έτσι ώστε να μειώσουν την συνολική προσπάθεια της υλοποίησης ενός νέου παιχνιδιού στο ελάχιστο. Επίσης, προσφέρει ένα ξεχωριστό, φιλικό προς τον χρήστη και ευέλικτο περιβάλλον ιστού για την δημιουργία, εγκατάσταση, παρακολούθηση και διαχείριση παρελθοντικών, παροντικών και μελλοντικών παιχνιδιών.

101

Πρότερη Εργασία

Αρκετή δουλειά έχει γίνει στον τομέα των πανταχού παρόντων παιχνιδιών. Ο στόχος του έργου IperG [52] το οποίο επιχορηγήθηκε από την Ευρωπαϊκή Ένωση, ήταν η διερεύνηση της εμπειρίας των πανταχού παρόντων παιχνιδιών και η υλοποίηση ενός συνόλου παιχνιδιών επίδειξης. Μέσα από αυτό το έργο δημιουργήθηκαν αρκετά παιχνίδια στις κατηγορίες του κρυμμένου θησαυρού, παιχνίδια σε μέγεθος επίπεδου πόλης, παιχνίδια εναλλακτικής πραγματικότητας (Alternate Reality Games, ARG), μαζικά παιχνίδια πολλών παικτών με χρήση κινητών τηλεφώνων καθώς και άλλα είδη λιγότερο η περισσότερο γνωστά. Πλήθος άλλων εργασιών παρουσιάζει διάφορες υλοποιήσεις τέτοιων παιχνιδιών, για παράδειγμα στις εργασίες [53] και [54]. Μερικά παραδείγματα γνωστών παιχνιδιών είναι το CatchBob!, Uncle Roy All Around You, SupraFly[55], Human Pacman[56]. Στο [57], αρκετά ενδιαφέροντα ζητήματα παρουσιάζονται, τα οποία σχετίζονται με την θεωρία γύρω από τα πανταχού παρόντα παιχνίδια. Στο [55], οι συγγραφείς αξιολόγησαν πώς ο κόσμος αντιλαμβάνεται, παρατηρεί και παίζει αυτού του είδους τα παιχνίδια, σε κανονικές, καθημερινές συνθήκες. Σε γενικές γραμμές, οι περισσότερες εργασίες εστιάζουν σε θέματα σχεδιασμού τα οποία εγείρονται από συγκεκριμένα παιχνίδια. Κάποιες από αυτές τις εργασίες κάνουν μία προσπάθεια γενίκευσης τέτοιων θεμάτων, τα οποία αφορούν τον σχεδιασμό αυτού του είδους των παιχνιδιών, ή παρέχουν στατιστικά πάνω σε προϋπάρχουσες εργασίες. Τέτοια παραδείγματα είναι το [58] και το [59]. Στο [60] οι συγγραφείς παρουσιάζουν σενάρια τα οποία δείχνουν τα προσδοκώμενα χαρακτηριστικά τέτοιων παιχνιδιών καθώς και προτείνουν υπηρεσίες για την ανάπτυξη και την εγκατάσταση διαψυχαγωγικών παιχνιδιών

(cross-media games), δηλαδή παιχνιδιών τα οποία παίζονται σε μία πληθώρα πλατφόρμων με ποικίλα χαρακτηριστικά.

Τα τελευταία χρόνια έχουν αρχίσει να χρησιμοποιούνται συσκευές οι οποίες αξιοποιούν την φυσική κίνηση του σώματος σε εμπορική κλίμακα, όπως για παράδειγμα το Nintendo Wii [47]. Η τελευταία κονσόλα παιχνιδιών που δημιούργησε η εταιρεία Nintendo. Η αρχική ονομασία που είχε διαλέξει η εταιρεία για το Wii, ήταν Revolution, που σημαίνει επανάσταση. Διατήρησε αυτή την ονομασία όσο βρισκόταν στο στάδιο κατασκευής, δηλαδή μέχρι τις 27 Απριλίου του 2006, όπου και μετονομάστηκε σε Wii. Σύμφωνα με τα λεγόμενα της Nintendo την ονόμασαν έτσι, καθώς το Wii παραπέμπει στην αγγλική λέξη "We" (εμείς), καθώς προσκαλεί τους παίκτες να παίξουν μαζί. Μάλιστα τα δύο "ii" αναφέρονται στα χειριστήρια της κονσόλας. Το στοιχείο που διαφοροποιεί το Wii από τις υπόλοιπες κονσόλες νέας γενιάς είναι ο χειρισμός του. Ενώ μέχρι στιγμής έχουμε συνηθίσει στα κοινά χειριστήρια παιχνιδιών, το Wii κάνει τη διαφορά. Ο χειρισμός γίνεται μέσω δύο τμημάτων: με το Remote, το οποίο μοιάζει με τηλεχειριστήριο και το Nunchuk, το οποίο είναι ένα κλασικό joystick σε μικρότερο μέγεθος. Αυτά τα δύο τμήματα συνδέονται με ένα καλώδιο μεταξύ τους. Το Remote ενσωματώνει αισθητήρες κίνησης, οι οποίοι λαμβάνουν τις κινήσεις που κάνει ο παίκτης με τα χέρια του και μεταφέρονται στο παιχνίδι. Για παράδειγμα σε ένα παιχνίδι τένις κάποιος μπορεί να χρησιμοποιήσει το Remote σαν ρακέτα, σε ένα παιχνίδι ξιφομαχιών σαν σπαθί κτλ

Πρότερη Εργασία

Το Wii διαθέτει πολύ κομψή εμφάνιση καθώς είναι ιδιαίτερα μικρό. Το μέγεθός του είναι όσο τρεις θήκες DVD τοποθετημένες οριζόντια και μπορεί να τοποθετηθεί κάθετα ή οριζόντια. Η Nintendo θέλοντας να απλοποιήσει όσο το δυνατόν περισσότερο το Wii, δημιούργησε ένα εισαγωγικό μενού με μορφή καναλιών. Για παράδειγμα, υπάρχει ένα κανάλι για να παίζει κάποιος παιχνίδια, ένα για να πλοηγείται στο διαδίκτυο αλλά και κανάλια πολλών άλλων ειδών, όπως Κανάλι για τον καιρό στον κόσμο κλπ. *Virtual Console* είναι η ονομασία της υπηρεσίας μέσω της οποίας παρέχεται η δυνατότητα λήψης παιχνιδιών για παλιές κονσόλες της Nintendo.

Το Wii προσφέρει επαναστατικό χειρισμό, χρησιμοποιώντας φυσικές κινήσεις του χειριστηρίου, το οποίο μοιάζει με ένα τυπικό τηλεχειριστήριο τηλεόρασης. Μέχρι και τέσσερα χειριστήρια μπορούν να συνδεθούν ταυτόχρονα στο Wii, χρησιμοποιώντας την τεχνολογία Bluetooth. Το σήμα του Bluetooth αγγίζει τα 10 μέτρα. Τόσο το Wii remote, όσο και το Nunchuk, περιλαμβάνουν αισθητήρες κινήσεως στους 3 άξονες του χώρου. Το Wii Remote επιπλέον, περιλαμβάνει ένα ηχείο, δυνατότητες δόνησης καθώς και μια θύρα επέκτασης, ενώ μπορεί να χρησιμοποιηθεί σαν δείκτης πάνω στην τηλεόραση μέσα σε απόσταση πέντε μέτρων. Το Wii Remote έχει ένα διακόπτη on/off της κονσόλας, σταυρό κατεύθυνσης, κουμπιά A, B, -, + και Home, ενώ το Nunchuk έχει αναλογικό μοχλό και κουμπιά C και Z. Ακόμη, το Wii μπορεί να μένει συνδεδεμένο ασύρματα στο Internet, ακόμη και όταν είναι κλειστό. Αυτή η υπηρεσία με την ονομασία "WiiConnect24", μπορεί να φέρνει στους παίκτες νέες εκπλήξεις ή αναβαθμίσεις στα παιχνίδια τους, ακόμη και όταν αυτοί δεν παίζουν. Οι χρήστες μπορούν να

συνδεθούν ασύρματα χρησιμοποιώντας Wi-Fi IEEE802.11b/g, ή έναν USB 2.0 LAN adaptor. Το Wii τέλος, μπορεί να επικοινωνήσει ασύρματα και με το DS. Το self-loading drive του Wii δέχεται τόσο single όσο double-layered οπτικούς δίσκους 12 εκατοστών και επιπλέον όλα τα παιχνίδια του GameCube που είναι αποθηκευμένα σε δίσκους 8 εκατοστών. Σε σύγκριση με τους ανταγωνιστές της, το Playstation 3 και το Xbox 360, υστερεί από άποψη γραφικών και άλλων τεχνικών χαρακτηριστικών. Όπως έχει τονίσει η ίδια η εταιρεία, το δυνατό σημείο της κονσόλας είναι ο χειρισμός, ο οποίος γίνεται με έναν αρκετά επαναστατικό τρόπο από αυτούς έχουμε συνηθίσει μέχρι στιγμής. Μέχρι στιγμής το Wii έχει τις μεγαλύτερες πωλήσεις κονσόλας νέας γενιάς, καθώς έχει πουλήσει πάνω από 44 εκατομμύρια συστήματα.

Η μεγάλη εμπορική επιτυχία αυτού του προϊόντος οδήγησε και άλλες εταιρείες όπως η Microsoft και η Sony να δημιουργήσουν τις δικές τους εκδοχές τέτοιων συστημάτων τα οποία δέχονται ως είσοδο φυσικά δεδομένα από αισθητήρες. Υπάρχουν τέτοια συστήματα, και τα οποία αξιοποιούν από επιταχυνσιόμετρα και γυροσκοπία (όπως το Nintendo που προαναφέρθηκε) μέχρι τον συνδυασμό τέτοιων αισθητήρων με κάμερες (π.χ. Sony) ή μόνο κάμερες (π.χ. Microsoft). Παρότι αυτά τα προϊόντα επιτρέπουν την ανίχνευση της φυσικής κίνησης των παικτών, διαφέρουν με ποικίλους τρόπους από την παρούσα προσέγγιση. Το Playstation 3 (ή PS3 σε συντομογραφία) είναι μια κονσόλα παιχνιδιών της Sony και αποτελεί την 3η κονσόλα της σειράς Playstation.

Η κονσόλα αυτή επιτρέπει στον χρήστη να παίζει παιχνίδια, να παρακολουθήσει ταινίες υψηλής ευκρίνειας

Πρότερη Εργασία

ας (High Definition) σε Blu-ray Disc, να πλοηγηθεί εύκολα στην αγαπημένη του μουσική, τις φωτογραφίες και το διαδίκτυο. Πρέπει να σημειωθεί πως πρόκειται για την μοναδική κονσόλα η οποία επιτρέπει την χρήση δίσκων υψηλής ευκρίνειας. Επίσης, διαθέτει ενσωματωμένη κάμερα και μικρόφωνο για ενδοεπικοινωνία στο παιχνίδι, ενορχηστρώνοντας αποδοτικά τις επιλογές παιχνιδιών πολλών παικτών (multiplayer games). Ακόμα, συνεργάζεται άρτια με την τηλεόραση, επιτρέποντας την καταγραφή των προγραμμάτων στον σκληρό, ή την προβολή DVD μέσω αυτής. Αντίστοιχα με τις ανταγωνιστικές εταιρείες, η Sony το έχει εφοδιάσει με πρόσβαση στο Playstation Network.

Το Xbox 360 είναι η δεύτερη κονσόλα παιχνιδιών βίντεο που παράγεται από τη Microsoft, και ο διάδοχος του Xbox. Το Xbox 360 ανταγωνίζεται το Playstation 3 της Sony και Wii της Nintendo, στο πλαίσιο της έβδομης γενιάς από κονσόλες βιντεοπαιχνιδιών. Μερικά κύρια χαρακτηριστικά του Xbox 360 είναι ολοκληρωμένη υπηρεσία του Xbox Live που επιτρέπει στους παίκτες να διαγωνιστούν σε απευθείας σύνδεση, να κατεβάσουν παιχνίδια, demos, τηλεοπτικές εκπομπές, ταινίες και μουσική με τις δυνατότητες του Windows Media Center. Επίσης, προσφέρει πρόσβαση σε υπηρεσίες ροής, όπως για παράδειγμα ψηφιακή τηλεόραση και ενημέρωση. Αξίζει να σημειωθεί

104



Πρότερη Εργασία

πως τον Ιανουάριο του 2010, υπάρχουν πάνω από 39 εκατομμύρια κονσόλες Xbox 360 παγκοσμίως. Το Xbox 360 είναι τώρα διαθέσιμο σε δύο εκδόσεις (Arcade και Elite) η κάθε μία με τη δική της επιλογή αξεσουάρ.

Στην συγκεκριμένη εργασία γίνεται διερεύνηση προς διαφορετικές κατευθύνσεις όσον αφορά τις εφαρμογές και τις εγκαταστάσεις. Έτσι, εδώ δεν αξιοποιείται μόνον η αναγνώριση κίνησης αλλά απαιτείται είσοδος από διαφορετικές πηγές όπως το φως, η υγρασία, η παρουσία, η θερμοκρασία κ.λ.π.. Επίσης, οι εφαρμογές αυτές παίζονται από πλήθος παικτών σε μη ελεγχόμενα περιβάλλοντα, με αποτέλεσμα η χρήση καμερών να καθίσταται πιθανότατα αδύνατη, καθώς οι προβολές των παικτών μπροστά στις κάμερες μπορεί να αλληλεπικαλύπτονται. Ακόμη, είναι πιθανός ο συντονισμός μεταξύ των χρηστών οι οποίοι συμμετέχουν (όσον αφορά τον χρόνο και τις κινήσεις), κάτι το οποίο θα μπορούσε να αποβεί πολύ δύσκολο στην υλοποίηση αλλιώς. Πρέπει να σημειωθεί επίσης πως το πεδίο εφαρμογών στο οποίο στοχεύει η συγκεκριμένη πλατφόρμα είναι εντελώς διαφορετικό. Τέλος, η εργασία αυτή βασίζεται σε εργαλεία και πλατφόρμες ανοιχτού κώδικα, ενώ τα προαναφερθέντα είναι εμπορικά προϊόντα κλειστού κώδικα, τόσο όσον αφορά το υλικό όσο και το λογισμικό.

Αρκετή προεργασία επίσης έχει γίνει στην χρήση περιφερώντων μεθόδων και εργαλείων σε διαδραστικές εγκαταστάσεις με έναν προσανατολισμό σε μουσεία και εκθέσεις όπως για παράδειγμα στα [61],[62][63]. Αυτές οι εργασίες διαφέρουν από την συγκεκριμένη καθώς δεν χρησιμοποιούν τέτοια ποικιλία εισόδων από αισθητήρες με σκοπό να παρέχουν επιπρόσθετη αλληλεπίδραση με τους χρήστες, ακολουθούν μία κεντροποιημένη αρχιτεκτονική καθώς επίσης δεν

προσφέρουν πολλά όσον αφορά τον συγχρονισμό και την αντίληψη καταστάσεων. Μία σχετικά εκτενής συζήτηση πάνω σε αυτά τα ζητήματα, όσον αφορά την ενσωμάτωση δικτύων αισθητήρων σε εφαρμογές προσανατολισμένες προς την εκπαίδευση περιλαμβάνεται στο [64].

Ενώ έχουν γίνει κάποιες προσπάθειες για την ανάπτυξη παιχνιδιών πολλαπλών παικτών τα οποία βασίζονται σε συσκευές οι οποίες μπορούν να αισθανθούν ιδιότητες του πραγματικού κόσμου, αυτές οι εργασίες είναι σχετικά περιορισμένες τόσο στον αριθμό όσο και στο πεδίο δράσης τους, και είναι ακόμα λιγότερες στο πλαίσιο των παιχνιδιών περιρρέοντος υπολογισμού πολλών παικτών. Παραδείγματα παιχνιδιών τα οποία βασίζονται σε ασύρματα δίκτυα αισθητήρων αποτελούν τα [65], [66]. Πιο συγκεκριμένα, στο [66] περιγράφεται μία ιδέα, αρκετά κοντά στην παρούσα, η οποία υλοποιήθηκε με την χρήση υλικολογισμικού (middleware) κινητών πρακτόρων, αλλά με πιο περιορισμένο πεδίο εφαρμογής και χωρίς την καινοτομική διεπαφή χρήση η οποία έχει υλοποιηθεί στην εργασία αυτή. Επίσης, στο [67] παρουσιάζεται ένα τουριστικό παιχνίδι με τοπικά χαρακτηριστικά το οποίο χρησιμοποιεί συγκεκριμένες απλές κινήσεις ώστε να αναγνωρίσει την είσοδο από τον χρήστη, πολύ κοντά στην ιδέα που χρησιμοποιείται στην παρούσα εργασία, με την βοήθεια επιταχυνσιόμετρων τα οποία είναι ενσωματωμένα σε κινητά τηλέφωνα, ενώ εδώ γίνεται χρήση κόμβων ασύρματων δικτύων αισθητήρων. Μία πλατφόρμα με διαφορετικό σκοπό από την παρούσα, αλλά στο ίδιο πνεύμα όσον αφορά το πραγματικό υλικό και την διεπαφή χρήστη την οποία παρέχει περιγράφεται στο [68]. Όσον αφορά την τεχνολογία δικτύωσης που χρησιμοποιείται σε παρόμοια παιχνί-

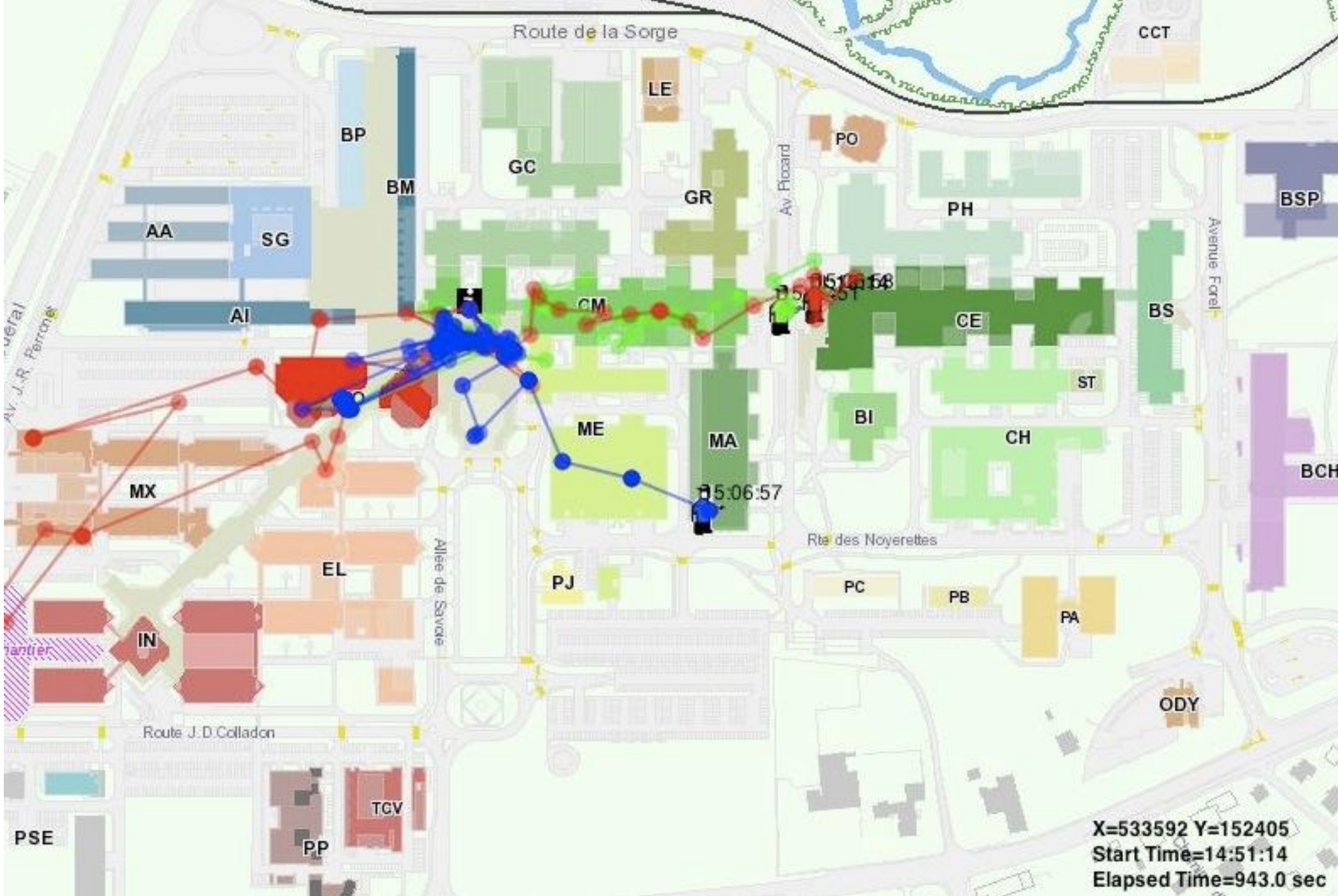
Πρότερη Εργασία

δια, στο [69] περιγράφονται οι εμπειρίες από την χρήση τεχνολογίας Bluetooth για την ανάπτυξη ενός παιχνιδιού πολλών παικτών. Η παρούσα εργασία εστιάζει στην χρήση συσκευών συμβατών με το IEEE 802.15.4. Πρέπει να σημειωθεί πως η συγκεκριμένη εργασία είναι η πρώτη, μέσα στο συγκεκριμένο πλαίσιο η οποία χρησιμοποιεί αυτό τον τύπο συσκευών. Υπάρχει επίσης ένα αξιολογούμενο μέγεθος εργασιών όσον αφορά την αναγνώριση κίνησης και χειρονομιών σε συσκευές περιορισμένων πόρων -όπως αυτή η οποία χρησιμοποιείται στην συγκεκριμένη εργασία- όπως για παράδειγμα το [70].

Όλες οι παραπάνω εργασίες διαφέρουν από την παρούσα, τουλάχιστον όσον αφορά τους στόχους τους. Είναι σημαντικό το γεγονός ότι οι εφαρμογές που δημιουργούνται με την χρήση της συγκεκριμένης πλατφόρμας περιλαμβάνουν πολλούς παίκτες, ταχεία φυσική δραστηριότητα καθώς και χρήση χειρονομιών, ενώ η συντριπτική πλειοψηφία των ήδη υπαρχόντων προσεγγίσεων δεν απαιτεί έντονη φυσική δράση. Επίσης, η χρήση συσκευών μικρών στο μέγεθος οι οποίες είναι εύκολες στην μεταφορά και την χρήση, έρχεται σε αντιδιαστολή με την εφαρμογή ακριβών κινητών τηλεφώνων και προσωπικών ψηφιακών οδηγών (pda) που παρουσιάζεται στις προαναφερθείσες εργασίες και η οποία περιορίζει τους χρήστες, τόσο λόγω του μεγέθους όσο και λόγω της αξίας των συσκευών, οι οποίες μπορούν να καταστραφούν κατά την διάρκεια του παιχνιδιού. Επίσης, στα συγκεκριμένα παιχνίδια, εμπεριέχεται και το στοιχείο του σεναρίου, κάτι το οποίο είναι εμφανές τόσο κατά την δημιουργία του παιχνιδιού όσο και κατά το παίξιμο του. Στην συγκεκριμένη πλατφόρμα απουσιάζει αυτό το χαρα-

κτηριστικό, καθώς η κατεύθυνση είναι λιγότερο σεναριακή, κάτι το οποίο κάνει τα παιχνίδια λιγότερο περλίτλοκα στην εγκατάσταση και το παίξιμό τους. Ένα τέτοιο χαρακτηριστικό είναι πολύ σημαντικό όσον αφορά την αποδοχή τέτοιων συστημάτων στο κοντινό μέλλον.





Στόχοι

Παρακάτω παρουσιάζονται οι βασικές απαιτήσεις ενός τέτοιου συστήματος και οι κύριοι σχεδιαστικοί στόχοι της πλατφόρμας που αφορά την ανάπτυξη εφαρμογών και αλληλεπιδραστικών εγκαταστάσεων για εκπαιδευτικούς και ψυχαγωγικούς σκοπούς. Γίνεται μία προσπάθεια ταυτοποίησης των διαφορετικών παραγόντων της συγκεκριμένης ιδέας από προηγούμενες, ήδη υπάρχουσες, προσεγγίσεις και κάποιων απαιτήσεων των εγκαταστάσεων. Οι συγκεκριμένοι παράγοντες-κλειδιά υπάρχουν στις εκπαιδευτικές και ψυχαγωγικές εφαρμογές των αδόμητων και κινητών δικτύων αισθητήρων.

Ταυτόχρονη Συμμετοχή Πολλών Παικτών

Ο στόχος είναι τα παιχνίδια και οι εγκαταστάσεις στα οποία συμμετέχουν ομάδες παικτών, πιθανώς μεγάλες σε αριθμό. Οι παίκτες θα βρίσκονται ο ένας κοντά στον άλλο, πιθανότατα σε εσωτερικούς χώρους, και θα χρησιμοποιούν αυτές τις εφαρμογές είτε αλληλεπιδρώντας μεταξύ τους είτε αλληλεπιδρώντας με μια οντότητα που θα παρέχεται από μια οργανωτική αρχή. Ανάλογα με την φύση της εφαρμογής, οι παίκτες μπορεί να χρειάζεται να συνεργάζονται ή να ανταγωνίζονται, π.χ. για να καταφέρουν να πετύχουν τους στόχους ενός ομαδικού παιχνιδιού μέσα σε ένα μουσείο, και αυτό πρέπει να γίνεται σε πραγματικό χρόνο. Όσον αφορά την υλοποίηση, αυτή προϋποθέτει την ύπαρξη ενός μηχανισμού ανακάλυψης γειτόνων, εφοδιασμένη με την ικανότητα αναγνώρισης απόστασης καθώς και με μηχανισμούς που να μεταλαμπαδεύουν την γνώση

της τοποθεσίας και του περιεχομένου στους παίκτες και το λογισμικό. Αυτοί οι μηχανισμοί είναι αναγκαίο να προσφέρονται και σε μεγάλο αριθμό παικτών και σε διαφορετικά μεγέθη περιοχών.

Μη Συμβατές Μέθοδοι Διεπαφής

Οι συμμετέχοντες πρέπει να είναι σε θέση να αποκορυπτογραφούν το προσωπικό ή/και το συλλογικό σκορ/κατάσταση όταν συμμετέχουν στα προτεινόμενα σχήματα αλληλεπίδρασης, όπως επίσης και οι διεπαφές του συστήματος πρέπει να αντικατοπτρίζουν την θέση και την γνώση του περιεχομένου και της πληροφορίας που υπάρχει έμφυτο σε αυτές τις περιπτώσεις. Η χρήση διακοπών όπως το άναμμα/σβήσιμο του φωτός, το άνοιγμα/κλείσιμο της πόρτας, απτικές διεπαφές (haptics) κ.α. θα επιτρέψουν μια όσο το δυνατόν πιο καθηλωτική εμπειρία.

Πολλαπλοί Τύποι Εισόδου

Απαιτείται χρήση πληθώρας δεδομένων εισόδου, τα πιο γενικά από τα οποία είναι η παρουσία, η κίνηση και άλλοι τύποι αισθητήρων. Αυτά τα δεδομένα εισόδου τις περισσότερες φορές θα παρέχονται από τις κινητές συσκευές που έχουν πάνω τους οι παίκτες. Αυτό σημαίνει ότι π.χ. στην περίπτωση του μουσείου, οι μαθητές ή οι επισκέπτες του μουσείου θα κουβαλούν κινητές συσκευές που είναι σε θέση να

Στόχοι

καταλάβουν τη θέση τους (απόλυτα ή σχετικά με τους υπόλοιπους παίκτες ή με προκαθορισμένα στοιχεία στο χώρο), τις κινήσεις τους (τόσο όσον αφορά την ανίχνευση κίνησης όσο και την αναγνώριση χειρονομιών) και άλλα φυσικά δεδομένα (π.χ. η συσκευή θα μπορούσε να ανιχνεύσει αν ο παίκτης είναι σε ζεστό/κρύο ή φωτεινό/σκοτεινό περιβάλλον). Έτσι, μοιάζει απαραίτητη η χρήση μιας επεκτάσιμης αρχιτεκτονικής για να καλύψει τα διαφορετικά είδη αισθητήρων που μπορούν να χρησιμοποιηθούν σε μία μόνο συσκευή και να μπορούν αυτές οι πληροφορίες να διαβιβαστούν στα ανώτερα στρώματα του συστήματος μαζί με μηχανισμούς για αξιόπιστη ανίχνευση κίνησης και χειρονομιών. Στην περίπτωση που χρησιμοποιηθούν και κάμερες μέσα στο σύστημα, πρέπει να εφαρμοσθούν οι αντίστοιχοι μηχανισμοί για τις ίδιες ενέργειες.

Λειτουργία Κατανεμημένου Δικτύου

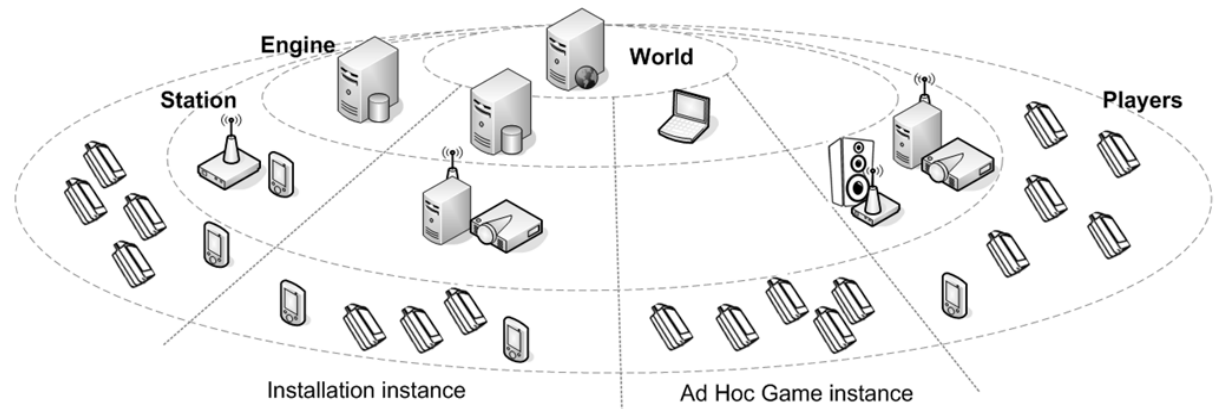
Οι ενσωματωμένοι αισθητήρες και τα αδόμητα δίκτυα προϋποθέτουν ότι το λογισμικό που εκτελείται από τις κινητές συσκευές που φέρουν οι παίκτες είναι “ελαφρύ”. Τα σύνθετα κομμάτια της λογικής του συστήματος πρέπει να υλοποιούνται στην σταθερή υποδομή που ορίζει τον κορμό. Επιπλέον, ανάλογα με την τελική εφαρμογή μπορεί να χρειάζονται επιπλέον λειτουργίες οι οποίες να βασίζονται στην επικοινωνία πραγματικού χρόνου ανάμεσα στους χρήστες και την ακριβή γνώση της θέσης τους ή να εκτελούνται σε ένα κομμάτι του δικτύου το οποίο είναι αποσυνδεδεμένο. Χρειάζεται λοιπόν η αρχιτεκτονική να είναι κατανεμημένη, ετερογενής και να συνθέτεται από πολλά κομμάτια. Μηχανισμοί ανεκτικοί σε καθυστερήσεις

μπορούν να ενεργοποιηθούν για να διασφαλίσουν την σωστή λειτουργία του συστήματος. Επίσης, αξιόπιστοι μηχανισμοί επικοινωνίας πολλών βημάτων (multihop) και πολλαπλής μετάδοσης (multicast) μπορεί να απαιτούνται για να διασφαλίσουν όλες τις πιθανές επικοινωνίες μεταξύ των παικτών και των υλοποιήσεων.

Ανάγκη για Συγχρονισμό και Συντονισμό

Στα περισσότερα παιχνίδια οι παίκτες ανταγωνίζονται ή συνεργάζονται για να πετύχουν τους στόχους που θέτει η συγκεκριμένη εφαρμογή. Οι παίκτες πρέπει να αλληλεπιδρούν απευθείας ο ένας με τον άλλο και το όλο σύστημα με ένα συγχρονισμένο τρόπο. Αυτά τα σχήματα συγχρονισμού πρέπει να καλύπτουν ενημερώσεις της κατάστασης των παικτών και του συστήματος, και πιθανώς να συντονίζουν τους τρόπους που οι χρήστες κινούνται και δρουν μέσα στο πεδίο του παιχνιδιού. Μηχανισμοί για αμοιβαίο αποκλεισμό, συμφωνία και εκλογή αρχηγού μπορούν να χρησιμοποιηθούν για να διασφαλίσουν την ορθή λειτουργία του συστήματος.

Αρχιτεκτονική



110

Το σύστημα σχεδιάστηκε και υλοποιήθηκε με στόχο σενάρια παιχνιδιών όπου ένας μεγάλος αριθμός παιχτών συμμετέχει σε πολλά στιγμιότυπα (game instance) και τύπους παιχνιδιών με τη χρήση ασύρματων συσκευών (hand-held devices) με αισθητήρες. Τα παιχνίδια αυτά μπορούν να λαμβάνουν χώρα στον ίδιο ή διαφορετικό χώρο και χρόνο. Η λειτουργία των παιχνιδιών υποστηρίζεται από μία υποδομή κορμού η οποία προσφέρει έναν αριθμό από υπηρεσίες (π.χ. εντοπισμός και συναίσθηση συμφραζομένων - local

ization, context awareness) διαμέσου των οποίων μία κεντρική οντότητα συντονίζει και καταγράφει την πρόοδο τους. Η αρχιτεκτονική βασίστηκε πάνω σε αυτές τις αρχές και υλοποιήθηκε από μία ιεραρχία επιπέδων. Σε κάθε επίπεδο έχει ανατεθεί ένας συγκεκριμένος ρόλος στην εφαρμογή.

Αρχιτεκτονική

Επίπεδο Φρουρών και Σταθμών Παιχνιδιού

Το επίπεδο φρουρών αποτελείται από τις συσκευές που χρησιμοποιούν οι παίκτες κατά την διάρκεια των παιχνιδιών. Ο *Φρουρός* είναι το πακέτο λογισμικού το οποίο τρέχει στην ασύρματη συσκευή με αισθητήρες κάθε παίκτη και αξιοποιεί τις δυνατότητες της συσκευής όσον αφορά την διεπαφή χρήστη, την επικοινωνία κ.τ.λ. Πρωτόκολλα για την ανακάλυψη και την επικοινωνία με την υποδομή κορμού (backbone) και άλλους Φρουρούς παρέχονται (π.χ. η υπηρεσία πρωτοκόλλου ηχούς - echo protocol). Όταν ένας άλλος Φρουρός ανιχνεύεται, ο παίκτης παροτρύνεται για περαιτέρω ενέργειες χρησιμοποιώντας τους αισθητήρες και τα κουμπιά της συσκευής. Για την εποπτεία της εξέλιξης του παιχνιδιού, κάθε ενέργεια που σχετίζεται με το παιχνίδι αναπαρίσταται με ένα *Γεγονός*. Επίσης οι Φρουροί υλοποιούν υπηρεσίες οι οποίες τους επιτρέπουν να αλληλεπιδρούν ακόμη και όταν βρίσκονται αποσυνδεδεμένοι από την υποδομή κορμού για μεγάλες χρονικές περιόδους. Συγκεκριμένα όταν ένα Γεγονός συμβαίνει, ο Φρουρός το αποθηκεύει στην μνήμη της συσκευής και όταν η επικοινωνία με την υποδομή καθίσταται εφικτή, τότε όλα τα συλλεχθέντα γεγονότα προωθούνται κατάλληλα (υπηρεσία επικοινωνίας με ανοχή στην καθυστέρηση - delay tolerant communication service). Επίσης οι Φρουροί παρέχουν ένα υποσύστημα το οποίο επεξεργάζεται τα δείγματα του επιταχυνσιόμετρου και αναγνωρίζει κινήσεις οι οποίες αντιστοιχίζονται σε ενέργειες που σχετίζονται με το παιχνίδι.

Αυτό το επίπεδο υλοποιεί την υποδομή κορμού η οποία είναι σημαντική αλλά όχι απαραίτητη για όλα τα παιχνίδια τα οποία αναπτύσσονται. Προσφέρει

υπηρεσίες εντοπισμού και συναίσθησης συμφραζόμενων. Μέσω αυτής της υποδομής τα δεδομένα των παιχτών μεταφέρονται από και προς τα υψηλότερα επίπεδα της αρχιτεκτονικής για λόγους συντονισμού και αποθήκευσης. Αυτός ο ασύρματος κορμός εδράινεται από *Σταθμούς*, με τον κάθε Σταθμό να ελέγχει μια συγκεκριμένη φυσική περιοχή. Κατά την διάρκεια της αρχικοποίησης του κάθε παιχνιδιού ένας Σταθμός αναλαμβάνει επίσης το ρόλο της *Μηχανής* Παιχνιδιού με αποτέλεσμα να είναι υπεύθυνος για τον συντονισμό όλης της υποδομής και του ίδιου του παιχνιδιού. Οι Σταθμοί επικοινωνούν με τις συσκευές των παιχτών είτε μέσω τοπικών δικτύων είτε μέσω προσωπικής περιοχής δικτύων χωρίς IP και δρουν ως πύλες δικτύου επιτρέποντας την επικοινωνία μεταξύ των συσκευών των παιχτών και της Μηχανής του Παιχνιδιού. Πολλαπλοί Σταθμοί μπορούν να προσκολληθούν σε μία Μηχανή με στόχο την μεγιστοποίηση της περιοχής κάλυψης ή των σημείων ενδιαφέροντος. Κατά την διάρκεια αρχικοποίησης ενός παιχνιδιού οι Σταθμοί επικοινωνούν με την Μηχανή και ανακτούν δεδομένα, όπως το σύνολο των παιχτών, τα οποία έχουν καταχωρηθεί για το συγκεκριμένο στιγμιότυπο, τις συσχετίσεις μεταξύ των χαρακτηριστικών των συσκευών των παιχτών και των σημείων ενδιαφέροντος. Οι Σταθμοί είναι επίσης υπεύθυνοι για την αρχικοποίηση των Φρουρών και για την προώθηση όλων των δεδομένων τα οποία δημιουργούνται κατά την διάρκεια του παιχνιδιού στην Μηχανή.

Επίσης υπάρχει επιλογή για την χρήση *κινητών* Σταθμών κατά την διάρκεια ενός παιχνιδιού. Σε αυτή την περίπτωση οι συγκεκριμένοι Σταθμοί λειτουργούν με

111

Αρχιτεκτονική

Επίπεδο Μηχανής Παιχνιδιού και Επίπεδο Κόσμου

έναν ελαφρώς διαφορετικό τρόπο από ότι όταν είναι ακίνητοι - ο ρόλος τους είναι κυρίως να προσφέρουν υπηρεσίες εντοπισμού θέσης κατά την διάρκεια των παιχνιδιών ενώ η επικοινωνία με τα ανώτερα επίπεδα είτε έχει κατασταλεί (με την παράλληλη ενημέρωση των κατώτερων επιπέδων ώστε να χρησιμοποιήσουν άλλους τρόπους για την μετάδοση γεγονότων σχετιζόμενων με το παιχνίδι στα ανώτερα επίπεδα) είτε μετατρέπεται σε λειτουργία ανοχής της καθυστέρησης (delay tolerant mode).

Κάθε στιγμιότυπο παιχνιδιού ανατίθεται και συντονίζεται από μια συγκεκριμένη *Μηχανή* Παιχνιδιού η οποία αποτελεί και τη τοπική αρχή για κάθε φυσική περιοχή παιχνιδιού. Η Μηχανή ανακτά δεδομένα από τα υψηλότερα επίπεδα και τα αποθηκεύει τοπικά για όλη τη διάρκεια του συγκεκριμένου παιχνιδιού. Σε μια προσπάθεια να αποφευχθεί επιπλέον κόστος υπολογισμού και επικοινωνίας, τα δεδομένα μεταξύ των υψηλότερων επιπέδων και της Μηχανής συγχρονίζονται περιοδικά. Έτσι η επεξεργασία και αποθήκευση των γεγονότων που δημιουργήθηκαν κατά τη διάρκεια ενός παιχνιδιού πραγματοποιείται τοπικά. Η Μηχανή είναι επίσης ένας μηχανισμός ελέγχου ο οποίος προσφέρει υπηρεσίες συγκεκριμένες για κάθε παιχνίδι και υλοποιεί διάφορα σενάρια παιχνιδιού. Η επικοινωνία μεταξύ της Μηχανής και του Σταθμού γίνεται

μεταξύ ενσύρματων ή / και ασύρματων δικτύων βασισμένα στην IP. Τέλος η Μηχανή παρέχει μία ενσωματωμένη διαδικτυακή διεπαφή (επί της ουσίας έναν web container) για να παρέχει στους παίκτες περαιτέρω πληροφορίες σχετιζόμενες με το παιχνίδι.

Το Επίπεδο *Κόσμου* είναι το υψηλότερο επίπεδο της ιεραρχίας επιτρέποντας την διαχείριση πολλαπλών παιχνιδιών, φυσικών περιοχών διεξαγωγής παιχνιδιού και χρηστών. Αυτό το επίπεδο περιλαμβάνει τη διαδικτυακή πύλη του Κόσμου (World Portal) η οποία είναι το κεντρικό σημείο διαχείρισης του συστήματος, παρέχοντας αλληλεπίδραση με όλα τα διαφορετικά στιγμιότυπα παιχνιδιών που υπάρχουν στον πραγματικό κόσμο. Είναι επίσης το κεντρικό σημείο αποθήκευσης για όλα τα δεδομένα που σχετίζονται με το παιχνίδι, όπως στατιστικά που σχετίζονται με τους παίκτες και ιστορικά των παιχνιδιών. Ακόμη προσφέρει δυνατότητες προσωποποίησης και πιθανής αλληλεπίδρασης με εξωτερικές ιστοσελίδες κοινωνικής δικτύωσης.

Φρουροί

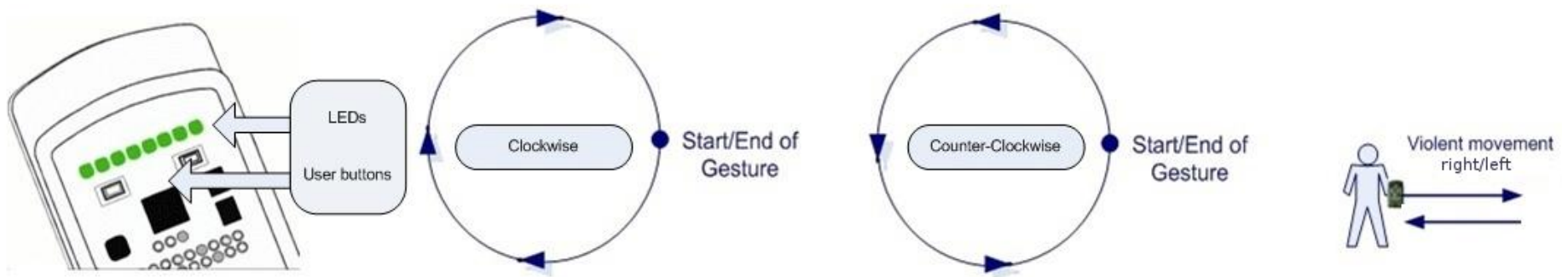
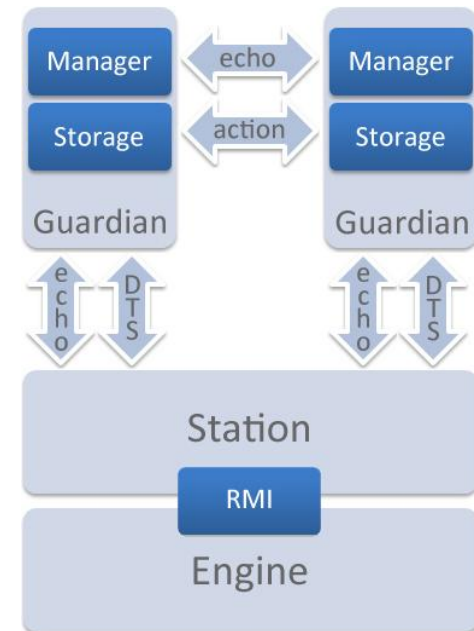
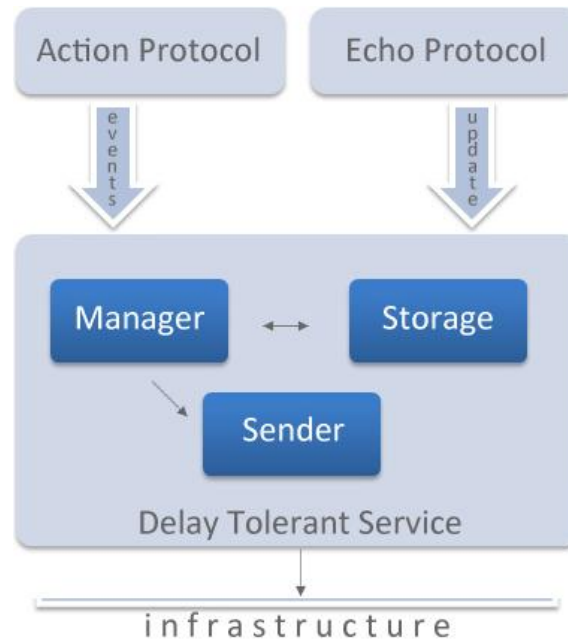
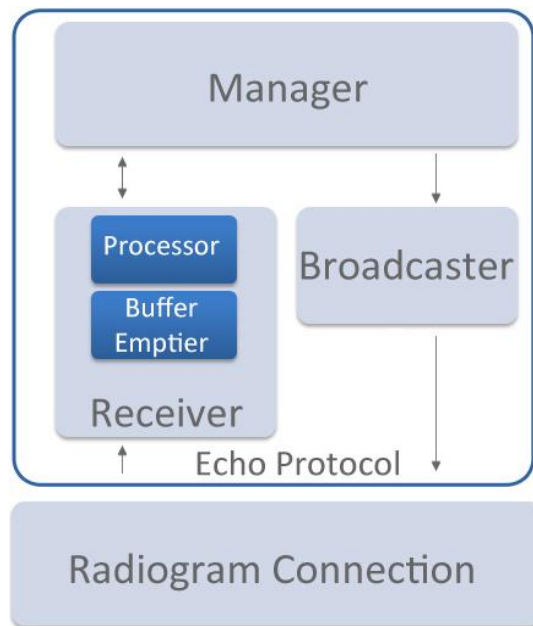
Πρωτόκολλο Ανακάλυψης Γειτόνων

Εδώ παρουσιάζεται ένα ελάχιστο σύνολο από υπηρεσίες και πρωτόκολλα τα οποία είναι διαθέσιμα στις συσκευές των παιχτών και τα οποία είναι αναγκαία για την πραγματοποίηση των παιχνιδιών που περιγράφθηκαν παραπάνω. Το πρωτόκολλο ανακάλυψης γειτόνων (echo protocol), το πρωτόκολλο ενεργειών (action protocol) και η υπηρεσία ανοχής της καθυστέρησης (delay tolerant service) έχουν υλοποιηθεί σε Java Micro Edition (J2ME) και τρέχουν στις συσκευές Spot της SUN. Η υλοποίηση του J2ME στα SUN Spot προσφέρει δύο διαφορετικά πρωτόκολλα επικοινωνίας, το πρωτόκολλο Ραδιοροής (RadioStream) και Ραδιοτηλεγραφήματος (Radio-Gram). Έγινε επιλογή χρήσης του ραδιοτηλεγραφήματος επειδή προσφέρει αξιόπιστη επικοινωνία με την χρήση ενσωματωμένων μεθόδων επιβεβαίωσης με τη χρήση datagram.

Το πρωτόκολλο ανακάλυψης γειτόνων είναι μία από τις βασικές υπηρεσίες τις πλατφόρμας η οποία υλοποιεί ανακάλυψη γειτόνων και επιτρέπει την συνδεσιμότητα. Διαμέσου του πρωτοκόλλου αυτού μια συσκευή ενός παίχτη ανακαλύπτει γειτονικούς Φρουρούς ή Σταθμούς και παρακολουθεί την κατάσταση της σύνδεσης τους. Η σπουδαιότητα του πρωτοκόλλου αυτού δικαιολογείται από την φύση των παιχνιδιών τα οποία αναπτύσσονται και στα οποία οι συσκευές δημιουργούν ένα δίκτυο που χαρακτηρίζεται από μεγάλη κινητικότητα και μεταβλητές ακτίνες μετάδοσης. Η τοπολογία ενός τέτοιου δικτύου αλλάζει δυναμικά και απρόβλεπτα καθώς οι παίχτες μετακινούνται ακαθόριστα χωρίς να ακολουθούν μια προκαθορισμένη λογική.

Το συγκεκριμένο πρωτόκολλο έχει σχεδιαστεί έτσι ώστε να τρέχει σε συσκευές με περιορισμένες δυνατότητες κυρίως όσον αφορά την επεξεργαστική τους ισχύ. Είναι εύρωστο με την δυνατότητα να προσαρμόζεται γρήγορα σε συχνές και σημαντικές αλλαγές τις τοπολογίας του δικτύου, καθώς και είναι ικανό να διακρίνει τους διαφορετικούς ρόλους των ανακαλυφθέντων γειτονικών κόμβων (δηλ. Φρουρών, Σταθμών και κινητών Σταθμών). Επιπλέον επιτρέπει την προσαρμογή των μηνυμάτων που αποστέλλονται και το κυριότερο έχει την δυνατότητα να αναγνωρίζει αν η επικοινωνία με τις περιβάλλουσες συσκευές είναι διπλής κατεύθυνσης ή όχι.

Πρακτικά, το πρωτόκολλο αποτελείται από τρία βασικά στοιχεία: τον αποστολέα (broadcaster), τον παραλήπτη (receiver) και τον διαχειριστή του πρωτοκόλλου ηχούς (echo protocol manager). Ο αποστολέας είναι ένα νήμα της Java υπεύθυνο για την δημιουργία και τη μετάδοση των μηνυμάτων με τη χρήση μίας σύνδεσης datagram εκπομπής σε μία συγκεκριμένη θύρα. Τα απαραίτητα πεδία προστίθενται ξεχωριστά στο datagram χρησιμοποιώντας διαφορετικούς πρωταρχικούς τύπους δεδομένων καθώς δεν υπάρχει υποστήριξη για σειριοποιησιμότητα στην J2ME. Ο παραλήπτης είναι επίσης ένα Java νήμα που ακούει για μεταδιδόμενα μηνύματα χρησιμοποιώντας μια σύνδεση εξυπηρετητή σε μία συγκεκριμένη θύρα. Μόλις ένα datagram λαμβάνεται, ένα αντικείμενο Φρουρού ή Σταθμού δημιουργείται ανάλογα με το ρόλο του αποστολέα. Τα δεδομένα τα οποία εξάγονται από το datagram χρησιμοποιούνται ώστε να αποδοθούν



Στην πρώτη εικόνα παρουσιάζεται σχηματικά το πρωτόκολλο ανακάλυψης γειτόνων και τα στοιχεία που αποτελούν την λειτουργία του. Στην δεύτερη εικόνα παρουσιάζεται σχηματικά η υπηρεσία ανοχής της καθυστέρησης, με τα στοιχεία που το αποτελούν να εμφανίζονται ως οντότητες. Στην τρίτη εικόνα παρουσιάζεται η επικοινωνία μεταξύ των επιπέδων: Η επικοινωνία μεταξύ Φρουρών και μεταξύ Φρουρών και Σταθμού γίνεται με την χρήση του πρωτοκόλλου ανακάλυψης γειτόνων και του πρωτοκόλλου ενεργειών, και μεταξύ Σταθμών και Μηχανής παιχνιδιού με την χρήση της RMI. Στην τέταρτη εικόνα παρουσιάζονται τα βασικά στοιχεία της διεπαφής του χρήστη: τα LED, τα κουμπιά, η κυκλική κίνηση με την φορά του ρολογιού, η κυκλική κίνηση με φορά αντίθετη από αυτή του ρολογιού και η βίαιη δεξιά/αριστερή κίνηση.

Φρουροί

Πρωτόκολλο Ενεργειών

τις παραμέτρους κατάλληλα και απαντά ότι είναι έτοιμος να λάβει την ενέργεια με ταυτότητα id το συγκεκριμένο ActionID. Όταν το A συλλέγει όλες τις αναμενόμενες επιβεβαιώσεις, προχωρά στην επόμενη φάση εκτελώντας το μέρος A και ζητώντας από όλους τους στόχους να εκτελέσουν το μέρος B. Στο τέλος η κατάσταση της Ενέργειας προσδιορίζεται χρησιμοποιώντας δυο λογικές μεταβλητές (Πίνακας 1): η Ενέργεια μπορεί να έχει μείνει ανολοκλήρωτη, μπορεί να έχει εκτελεσθεί επιτυχώς ή ανεπιτυχώς.

Εξαιτίας πολλών λόγων (π.χ. ακαθόριστη κίνηση των παικτών, διάφορες στρατηγικές παιχνιδιού) η επικοινωνία με την υποδομή Κορμού (τους Σταθμούς) μπορεί να μην είναι πάντα δυνατή. Κατά τη διάρκεια αυτής της περιόδου η εξέλιξη του παιχνιδιού δε θα έπρεπε να επηρεάζεται καθώς οι παίκτες αλληλεπιδρούν μεταξύ τους και δημιουργούν γεγονότα. Για το σκοπό αυτό έχει δημιουργηθεί η υπηρεσία ανοχής καθυστέρησης (Delay Tolerant Service - DTS) η οποία επιτρέπει τη λειτουργία τόσο σε κατάσταση σύνδεσης όσο και σε αποσύνδεσης. Όταν η επικοινωνία με την υποδομή καθίσταται εφικτή, τα δεδομένα μεταφέρονται (είτε άμεσα είτε μέσω άλλων κινητών συσκευών οι οποίες έχουν συνδεθεί) ενώ όταν αυτό δεν είναι δυνατό τα δεδομένα αποθηκεύονται και αποστέλλονται αργότερα – όταν η επικοινωνία αποκαθίσταται. Έτσι οι παίκτες μπορούν να μπαίνουν και να βγαίνουν από την εμβέλεια της υποδομής απολαμβάνοντας τα παιχνίδια χωρίς κανένα περιορισμό.

Όταν μία συσκευή παράγει ένα γεγονός, επί της ουσίας

ας αυτό προωθείται διαφανώς στο DTS. Έπειτα το DTS ελέγχει την κατάσταση της σύνδεσης η οποία και παρέχεται από το πρωτόκολλο ηχούς. Αν υφίσταται μία σύνδεση με κάποιο σταθμό, τότε το γεγονός στέλνεται κατευθείαν σε αυτόν. Αν δεν υπάρχει κάποια τέτοια σύνδεση, τότε το γεγονός αποθηκεύεται στη μνήμη του SUN Spot. Με την ανακάλυψη ενός σταθμού από το πρωτόκολλο ηχούς, το DTS ενημερώνεται και ο διαχειριστής του DTS προσπαθεί να αποστείλει όλα τα αποθηκευμένα γεγονότα. Το DTS ακολουθεί το observable/observer design pattern παρατηρώντας το πρωτόκολλο ηχούς για την αποφυγή συνεχών επερωτήσεων όσον αφορά τις αλλαγές στην κατάσταση της σύνδεσης ενός σταθμού. Για το λόγο αυτό έπειτα από μία μεγάλη περίοδο λειτουργίας σε κατάσταση αποσύνδεσης, πολλαπλά γεγονότα θα πρέπει να είναι αποθηκευμένα στη μνήμη της συσκευής. Σε μια τέτοια περίπτωση, η πιθανή ενημέρωση που θα έρθει από το πρωτόκολλο ανακάλυψης γειτόνων, θα καταλήξει στην χρονοβόρα διαδικασία της αποστολής πολλαπλών γεγονότων. Για την αποφυγή αυτού του φαινομένου, το νήμα αποστολής γεγονότων (event sender thread), χρησιμοποιείται για την αποστολή των γεγονότων σώζοντας το πρωτόκολλο ηχούς από την περιττή καθυστέρηση. Ένας κόμβος θεωρείται ότι βρίσκεται άμεσα συνδεδεμένος όταν βρίσκεται στην εμβέλεια ενός Σταθμού. Η απόσταση σε αυτή την περίπτωση είναι ένα βήμα (one hop away). Παρ' όλ' αυτά, ένας κόμβος μπορεί να βρίσκεται έμμεσα συνδεδεμένος με τις παραμέτρους κατάλληλα και απαντά ότι είναι έτοιμος να λάβει την ενέργεια με ταυτότητα id το συγκεκριμένο ActionID. Όταν το A συλλέγει όλες τις α-

Φρουροί

Υπηρεσία Ανοχής Καθυστέρησης

ναμενόμενες επιβεβαιώσεις, προχωρά στην επόμενη φάση εκτελώντας το μέρος Α και ζητώντας από όλους τους στόχους να εκτελέσουν το μέρος Β. Στο τέλος η κατάσταση της Ενέργειας προσδιορίζεται χρησιμοποιώντας δυο λογικές μεταβλητές (Πίνακας 1): η Ενέργεια μπορεί να έχει μείνει ανολοκλήρωτη, μπορεί να έχει εκτελεσθεί επιτυχώς ή ανεπιτυχώς.

Εξαιτίας πολλών λόγων (π.χ. ακαθόριστη κίνηση των παικτών, διάφορες στρατηγικές παιχνιδιού) η επικοινωνία με την υποδομή Κορμού (τους Σταθμούς) μπορεί να μην είναι πάντα δυνατή. Κατά τη διάρκεια αυτής της περιόδου η εξέλιξη του παιχνιδιού δε θα έπρεπε να επηρεάζεται καθώς οι παίκτες αλληλεπιδρούν μεταξύ τους και δημιουργούν γεγονότα. Για το σκοπό αυτό έχει δημιουργηθεί η υπηρεσία ανοχής καθυστέρησης (Delay Tolerant Service - DTS) η οποία επιτρέπει τη λειτουργία τόσο σε κατάσταση σύνδεσης όσο και σε αποσύνδεσης. Όταν η επικοινωνία με την υποδομή καθίσταται εφικτή, τα δεδομένα μεταφέρονται (είτε άμεσα είτε μέσω άλλων κινητών συσκευών οι οποίες έχουν συνδεθεί) ενώ όταν αυτό δεν είναι δυνατό τα δεδομένα αποθηκεύονται και αποστέλλονται αργότερα – όταν η επικοινωνία αποκαθίσταται. Έτσι οι παίκτες μπορούν να μπαίνουν και να βγαίνουν από την εμβέλεια της υποδομής απολαμβάνοντας τα παιχνίδια χωρίς κανένα περιορισμό.

Όταν μία συσκευή παράγει ένα γεγονός, επί της ουσίας αυτό προωθείται διαφανώς στο DTS. Έπειτα το DTS ελέγχει την κατάσταση της σύνδεσης η οποία και παρέχεται από το πρωτόκολλο ηχούς. Αν υφίσταται

μία σύνδεση με κάποιο σταθμό, τότε το γεγονός στέλνεται κατευθείαν σε αυτόν. Αν δεν υπάρχει κάποια τέτοια σύνδεση, τότε το γεγονός αποθηκεύεται στη μνήμη του SUN Spot. Με την ανακάλυψη ενός σταθμού από το πρωτόκολλο ηχούς, το DTS ενημερώνεται και ο διαχειριστής του DTS προσπαθεί να αποστείλει όλα τα αποθηκευμένα γεγονότα. Το DTS ακολουθεί το observable/observer design pattern παρατηρώντας το πρωτόκολλο ηχούς για την αποφυγή συνεχών επερωτήσεων όσον αφορά τις αλλαγές στην κατάσταση της σύνδεσης ενός σταθμού. Για το λόγο αυτό έπειτα από μία μεγάλη περίοδο λειτουργίας σε κατάσταση αποσύνδεσης, πολλαπλά γεγονότα θα πρέπει να είναι αποθηκευμένα στη μνήμη της συσκευής. Σε μια τέτοια περίπτωση, η πιθανή ενημέρωση που θα έρθει από το πρωτόκολλο ανακάλυψης γειτόνων, θα καταλήξει στην χρονοβόρα διαδικασία της αποστολής πολλαπλών γεγονότων. Για την αποφυγή αυτού του φαινομένου, το νήμα αποστολής γεγονότων (event sender thread), χρησιμοποιείται για την αποστολή των γεγονότων σώζοντας το πρωτόκολλο ηχούς από την περιττή καθυστέρηση. Ένας κόμβος θεωρείται ότι βρίσκεται άμεσα συνδεδεμένος όταν βρίσκεται στην εμβέλεια ενός Σταθμού. Η απόσταση σε αυτή την περίπτωση είναι ένα βήμα (one hop away). Παρ' όλ' αυτά, ένας κόμβος μπορεί να βρίσκεται έμμεσα συνδεδεμένος με ένα σταθμό, μέσω ενός μονοπατιού πολλών βημάτων. Για να είναι εφικτό κάτι τέτοιο, πραγματοποιείται περιορισμένη πλημμύρα των μηνυμάτων των Σταθμών στο δίκτυο.

Φρουροί

Λοιπές Υπηρεσίες

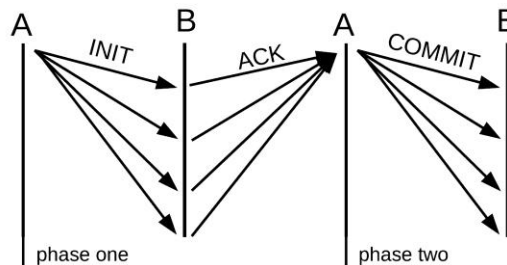
Οι συσκευές των παιχτών υλοποιούν επιτρόσθετες υπηρεσίες υποστήριξης. Η ενότητα της αποθήκευσης (storage module) είναι υπεύθυνη για να παρέχει δυνατότητες μόνιμης αποθήκευσης. Με τη χρήση αυτής της ενότητας, η κατάσταση των Φρουρών και διάφορων γεγονότων του παιχνιδιού τα οποία και δημιουργούνται αποθηκεύονται και ανακτώνται ακόμη και όταν η συσκευή επανεκκινείται (είτε από λάθος είτε ηθελημένα) ή σβήνει λόγω εξάντλησης της μπαταρίας. Στην J2ME η λειτουργικότητα της αποθήκευσης παρέχεται με την χρήση αποθήκευσης εγγραφών (recordstores). Η ενότητα αποθήκευσης που έχει υλοποιηθεί παρέχει ένα σύνολο από απλές συναρτήσεις για την αποθήκευση και ανάκτηση δεδομένων, οι οποίες αξιοποιούν την αποθήκευση εγγραφών της J2ME.

Η ενότητα της διεπαφής χρήστη είναι υπεύθυνη για την αλληλεπίδραση με τον παίκτη. Όπως έχει αναφερθεί και προηγουμένως, γίνεται χρήση των συσκευών spot της SUN, ως συσκευές παιχνιδιού. Η παρεχόμενη διεπαφή χρήστη, αξιοποιεί τα 8 LED των spot, τα δύο κουμπιά χρήστη και το επιταχυνσιόμετρο το οποίο βρίσκεται στις συσκευές. Κάθε LED μπορεί να πάρει ένα από 256 διαφορετικά χρώματα, παρόλο που στην πράξη ο αριθμός των διακριτών από το ανθρώπινο μάτι χρωμάτων, είναι αρκετά μικρότερος. Εδώ αξίζει να σημειωθεί πώς η εμπειρία των χρηστών θα είχε βελτιωθεί σε πολύ μεγάλο βαθμό αν τόσο τα LED όσο και τα κουμπιά στις συσκευές ήταν μεγαλύτερα.

Η ενότητα αναγνώρισης κίνησης, αξιοποιεί το επιταχυνσιόμετρο στις συσκευές για να αναγνωρίσει τρείς

βασικές κινήσεις: i) μια με τη φορά του ρολογιού ii) μια με αντίθετη από την φορά του ρολογιού και iii) την βίαιη προς τα δεξιά κίνηση. Όταν μια συγκεκριμένη κίνηση αναγνωρίζεται, το πρωτόκολλο ενεργειών εκκινείται. Τέλος, η ενότητα του διαχειριστή Φρουρών είναι υπεύθυνη για την αρχικοποίηση της συσκευής και τον υψηλού επιπέδου έλεγχο όλων των υπηρεσιών και πρωτοκόλλων. Για παράδειγμα διατηρεί την κατάσταση των φρουρών χρησιμοποιώντας την ενότητα Αποθήκευσης, χειρίζεται την είσοδο (π.χ. το πάτημα κουμπιών ή την πραγματοποίηση κινήσεων) και παράγει την έξοδο. (π.χ. ανάβει ένα LED ή δημιουργεί και στέλνει ένα Γεγονός). Επίσης, ο διαχειριστής Φρουρών είναι υπεύθυνος για την παρακολούθηση της τήρησης των κανόνων κάθε παιχνιδιού, από την πλευρά του Φρουρού.

Στην παρακάτω εικόνα παρουσιάζεται το χρονικό του πρωτοκόλλου ενεργειών (Action Protocol).



117

Μηχανές και Σταθμοί

118

Οι σταθμοί επικοινωνούν με τους Φρουρούς με σκοπό να τους αρχικοποιήσουν, να λάβουν γεγονότα, να προσφέρουν υπηρεσίες εντοπισμού και να συγκεντρώσουν στατιστικά. Από την άλλη πλευρά, η Μηχανή ενός παιχνιδιού είναι υπεύθυνη για τον κεντροκοιμημένο συντονισμό των Σταθμών και γενικότερα του παιχνιδιού (δηλαδή τον έλεγχο για την τήρηση των κανόνων και την εγκυρότητα των Γεγονότων). Η επικοινωνία μεταξύ της Μηχανής και των Σταθμών πραγματοποιείται χρησιμοποιώντας την RMI (Remote Method Invocation - Απομακρυσμένη Κλήση Συναρτήσεων). Με λίγα λόγια, η RMI επιτρέπει σε ένα αντικείμενο που τρέχει σε μία Εικονική Μηχανή της Java (δηλαδή Σταθμό ή Μηχανή) να καλεί μεθόδους από ένα αντικείμενο το οποίο τρέχει σε μια άλλη Μηχανή Java (ομοίως). Παρ' όλ' αυτά, με την κλήση συναρτήσεων μέσω της RMI, ήταν δυνατό για αυτόν που καλεί την συνάρτηση να καθυστερεί μέχρι να τελειώσει η εκτέλεση της συνάρτησης στην άλλη άκρη. Για να αποφευχθεί κάτι τέτοιο, υλοποιήθηκε μία επικοινωνία δεδομένων με χρήση ουρών FIFO (First In-First Out Queues). Ένας σταθμός στέλνει ένα σύνολο από Γεγονότα στην Μηχανή χωρίς να περιμένει από την Μηχανή να ολοκληρώσει την προώθηση αυτών των δεδομένων στα ανώτερα επίπεδα της ιεραρχίας. Με αυτόν τον τρόπο, τόσο η Μηχανή όσο και ο Σταθμός μπορούν να συνεχίσουν απρόσκοπτα την λειτουργία τους, ασύγχρονα και με χρονική αλληλοεπικάλυψη. Να σημειωθεί πως η Μηχανή και ο Σταθμός Παιχνιδιού υλοποιήθηκαν με την χρήση του J2SE 1.6.

Η Μηχανή επικοινωνεί έμμεσα με τα υψηλότερα επίπεδα της αρχιτεκτονικής (επίπεδο του Κόσμου) μέσω της βάσης δεδομένων. Αυτή η ενότητα είναι ένα από

τα κεντρικότερα κομμάτια της πλατφόρμας, καθώς όλη η πληροφορία που σχετίζεται με το παιχνίδι αποθηκεύεται οριστικά στην βάση. Η αρχιτεκτονική της Μηχανής ακολουθεί ένα μοντέλο τριών επιπέδων, με κάθε επίπεδο να αναπτύσσεται και να συντηρείται ως μία ανεξάρτητη υποενότητα. Το υψηλότερο υποεπίπεδο αυτής της ιεραρχίας αποτελείται από τους διαχειριστές οντοτήτων, οι οποίοι και παρέχουν τις βασικές συναρτήσεις για μόνιμη αποθήκευση. (δημιουργία, ανάγνωση, ενημέρωση και διαγραφή - create, read, update and delete, CRUD). Συγκεκριμένα, έχει υλοποιηθεί ένας αφηρημένος διαχειριστής οντοτήτων, ο οποίος και παρέχει αυτές τις συναρτήσεις. Κάθε κλάση της Java η οποία και χρειάζεται πρόσβαση στην βάση δεδομένων, επεκτείνει τον αφηρημένο αυτόν διαχειριστή και παρέχει, αν είναι απαραίτητο, ακόμη πιο υψηλού επιπέδου συναρτήσεις. Οι διαχειριστές οντοτήτων δέχονται ως είσοδο αντικείμενα της Java και τα προωθούν στο μεσαίο υποεπίπεδο, το επίπεδο διατήρησης (Persistence Layer). Σε αυτό το υποεπίπεδο, οι οντότητες μετατρέπονται σε εγγραφές στην βάση, και οι συναρτήσεις της Java σε επερωτήσεις στην βάση και αντίστροφα. Το τελευταίο επίπεδο είναι η ίδια η βάση (μπορεί να είναι MySQL, Oracle, HSQL κλπ) όπου τα δεδομένα αποθηκεύονται και ανακτώνται.

Το μεσαίο επίπεδο έχει υλοποιηθεί με την χρήση της Hibernate, [82] μία βιβλιοθήκη της Java η οποία προσφέρει ένα πλαίσιο για την αντιστοίχιση ενός αντικειμενοστραφούς μοντέλου (object oriented domain model) σε μια παραδοσιακή συσχετιστική βάση δεδομένων (relational database). Η Hibernate προσφέρει ανεξαρτησία τελικής βάσης δεδομένων, επιτρέποντας με αυτό τον τρόπο την εύκολη μεταφο-

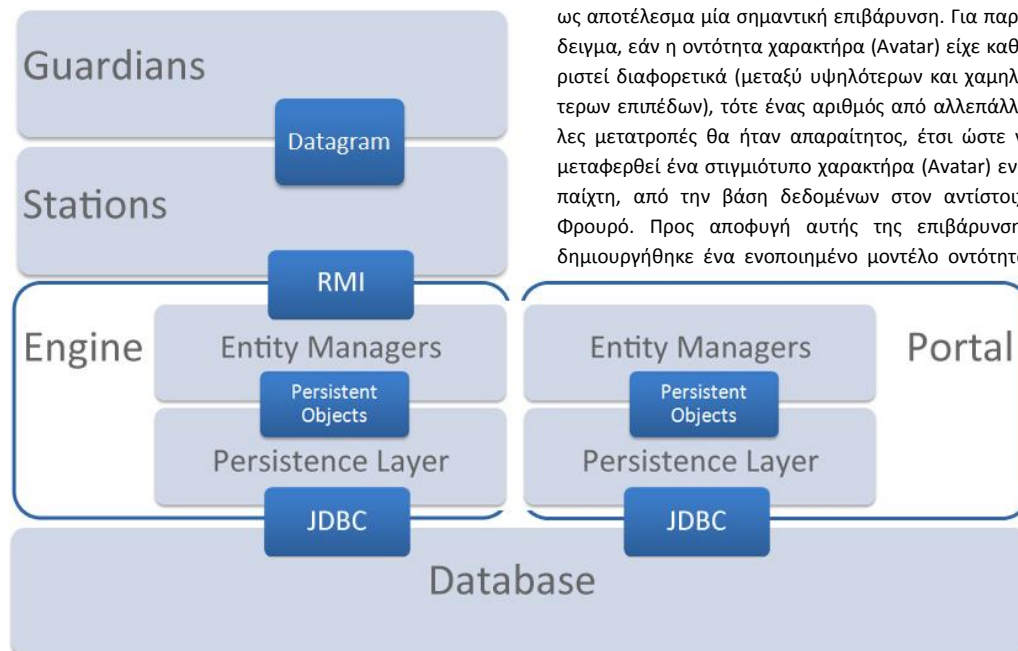
Μηχανές και Σταθμοί

Διατμηματική Διατήρηση Οντοτήτων

ρά των οντοτήτων των παιχνιδιών σε οποιοδήποτε σύστημα διαχείρισης βάσεων δεδομένων (relational database management system, *RDBMS*). Για κάθε οντότητα έχουν υλοποιηθεί κάποια αρχεία XML τα οποία περιέχουν τις αντιστοιχίες μεταξύ των χαρακτηριστικών των Java κλάσεων και των αντίστοιχων στηλών των πινάκων, μαζί φυσικά και με τις συσχετίσεις μεταξύ των ίδιων των πινάκων.

Ένα από τα βασικότερα χαρακτηριστικά του συστήματος είναι η Διατμηματική Επικοινωνία. Η επικοινωνία

της βάσης δεδομένων και της Μηχανής επιτυγχάνεται μέσω της Τεχνολογίας Hibernate, μεταξύ Μηχανής και Σταθμού μέσω RMI, μεταξύ Σταθμού και Φρουρού μέσω Radiograms. Η επιλογή διαφορετικών τεχνολογιών για την επίτευξη επικοινωνίας μεταξύ επιπέδων, δημιουργεί διάφορα θέματα συμβατότητας. Μία απλή λύση είναι για κάθε επίπεδο, οι οντότητες να αναπαριστώνται βασισμένες στα τεχνικά θέματα τις κάθε τεχνολογίας ξεχωριστά. Έπειτα, η διατμηματική επικοινωνία θα απαιτούσε συνεχή μετατροπή δεδομένων μεταξύ διαφορετικών μοντέλων οντοτήτων, έχοντας ως αποτέλεσμα μία σημαντική επιβάρυνση. Για παράδειγμα, εάν η οντότητα χαρακτήρα (Avatar) είχε καθοριστεί διαφορετικά (μεταξύ υψηλότερων και χαμηλότερων επιπέδων), τότε ένας αριθμός από αλληλλάλλες μετατροπές θα ήταν απαραίτητος, έτσι ώστε να μεταφερθεί ένα στιγμιότυπο χαρακτήρα (Avatar) ενός παίχτη, από την βάση δεδομένων στον αντίστοιχο Φρουρό. Προς αποφυγή αυτής της επιβάρυνσης, δημιουργήθηκε ένα ενοποιημένο μοντέλο οντότητας

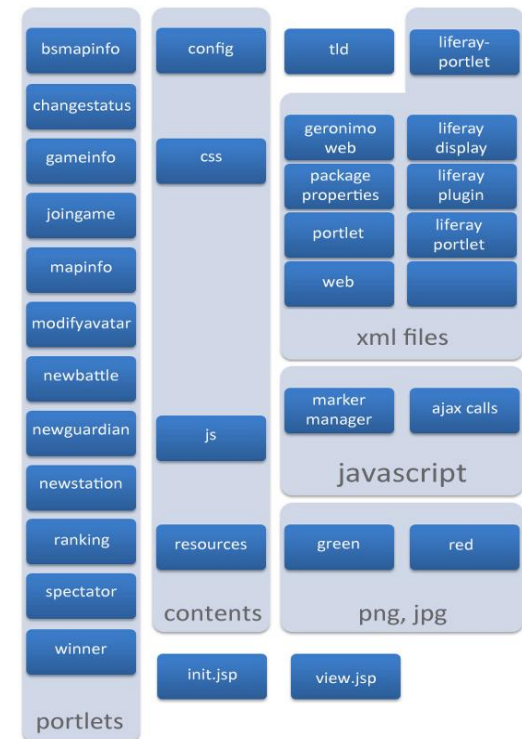


Μηχανές και Σταθμοί

Διατμηματική Διατήρηση Οντοτήτων

για όλα τα επίπεδα, το οποίο ελαχιστοποιεί τις μετατροπές και επιταχύνει την διατμηματική αλληλεπίδραση. Ένα βασικό πρόβλημα με το ενοποιημένο μοντέλο είναι ότι τα επίπεδα έχουν υλοποιηθεί σε διαφορετικές εκδόσεις της Java. Οι εκδόσεις που χρησιμοποιούνται έχουν ως εξής, Java Standard Edition (J2SE) για την Hibernate, τη Μηχανή και τον Σταθμό ενώ Java Micro Edition (J2ME) χρησιμοποιείται για τους Φρουρούς, η οποία τρέχει σε Sun SPOT συσκευές. Συγκεκριμένα, το κύριο πρόβλημα εμφανίζεται στην αλληλεπίδραση μεταξύ Φρουρού και Βάσης Δεδομένων/Hibernate, ειδικά όσον αφορά την αλληλεπίδραση πολλών-με-πολλές συσχετίσεις όπου η Hibernate απαιτεί τις `java.util.Set` και `java.util.HashSet` κλάσεις, οι οποίες δεν είναι ενσωματωμένες στην J2ME. Επιπλέον, η κλάση `HashSet` υλοποιεί την σειριοποιησιμότητα όταν αυτή δεν υποστηρίζεται πλήρως από την J2ME. Για να ξεπεραστούν αυτά τα προβλήματα και να επιτευχθεί η υψηλότερη δυνατή ενσωμάτωση, διατηρώντας την δυνατότητα επαναχρησιμοποίησης του κώδικα μεταξύ των επιπέδων, δημιουργήθηκαν ειδικά scripts μεταγλώττισης με τη βοήθεια του Apache Ant έτσι ώστε κάθε μη-συμβατή Java οντότητα να μπορεί να αντικατασταθεί από συμβατές κατά την μετακίνηση από υψηλότερα σε χαμηλότερα επίπεδα και αντίστροφα. Έτσι, το `java.util.Set` αντικαθίσταται με το `java.Object`, έχοντας ως σκοπό, ταυτόχρονα, να ελαττώσει την κατανάλωση μνήμης, εφόσον η κλάση `Object` είναι η πρωταρχική κλάση στην ιεραρχία των κλάσεων. Επιπλέον, κάθε Μέθοδος που αναφέρεται στην συγκεκριμένη κλάση έχει σχολιαστεί. Αυτή η

λύση δεν προκαλεί προβλήματα στα χαμηλότερα επίπεδα (π.χ. στο επίπεδο Φρουρού) καθώς η αίσθηση του Φρουρού για τον Κόσμο είναι περιορισμένη και δεν απαιτεί γνώση συσχετίσεων μεταξύ των αντικειμένων.



120

Κόσμος

Ο Κόσμος προσφέρει ένα σύνολο από υπηρεσίες διαθέσιμες στους παίκτες οι οποίες δεν είναι όλες αναγκαίες ταυτόχρονα για την πραγματοποίηση κάθε εφαρμογής αλλά χρησιμοποιούνται παράλληλα ανάλογα με τον τύπο της. Το επίπεδο αυτό είναι το υψηλότερο επίπεδο της αρχιτεκτονικής επιτρέποντας την εποπτική διαχείριση πολλαπλών εφαρμογών, την κεντρική διεπαφή σε πλήθος εγκαταστάσεων, την παρακολούθηση διεξαγωγής παιχνιδιών, καθώς και την διαχείριση των χρηστών. Η επικοινωνία με τα υπόλοιπα επίπεδα της αρχιτεκτονικής μπορεί να γίνει είτε από την βάση, (για παράδειγμα σε εφαρμογές στις οποίες είναι απαραίτητο το ιστορικό του παιχνιδιού) είτε άμεσα από τις Μηχανές (για παράδειγμα, σε καλλιτεχνικές διαδραστικές εγκαταστάσεις).

Ένα μέρος του επιπέδου αυτού περιλαμβάνει τη διαδικτυακή πύλη του Κόσμου (World Portal) η οποία είναι το κεντρικό σημείο διαχείρισης του συστήματος, παρέχοντας αλληλεπίδραση με όλα τα διαφορετικά στιγμιότυπα παιχνιδιών που υπάρχουν στον πραγματικό κόσμο. Εδώ περιλαμβάνονται επίσης τα υποσυστήματα της παρακολούθησης των παιχνιδιών καθώς και της διαδικασίας συμμετοχής στις εφαρμογές της πλατφόρμας οι οποίες απαιτούν την ταυτοποίηση του χρήστη. Πρακτικά, στο επίπεδο του χρήστη αυτό σημαίνει πως εδώ δηλώνει συμμετοχή και διαχειρίζεται τα χαρακτηριστικά του λογαριασμού του, ανεξάρτητα από το πώς αυτό μεταφράζεται στο περιεχόμενο κάθε παιχνιδιού. Προφανώς είναι δυνατή η συμμετοχή και διαχείριση πολλαπλών παιχνιδιών από έναν και μοναδικό λογαριασμό χρήστη. Ακόμη, η διαδικτυακή πύλη είναι το κεντρικό σημείο αποθήκευσης για όλα τα δεδομένα που σχετίζονται με το παιχνίδι, όπως στατιστικά που σχετίζονται με τους παίκτες και ιστορικά των παιχνιδιών. Τέλος, προσφέρει δυνατότητες προ-

σωποποίησης και πιθανής αλληλεπίδρασης με εξωτερικές ιστοσελίδες κοινωνικής δικτύωσης όπως το facebook, το twitter και το YouTube. Έτσι, υπάρχει η δυνατότητα από χρήστες να παρακολουθούν διαφορετικές εξελίξεις των εφαρμογών ή να βλέπουν το σκορ των παιχνιδιών ακόμα και από διαδικτυακούς τόπους τρίτων.

Ένα άλλο μέρος του επιπέδου αυτού περιλαμβάνει την αλληλεπίδραση (interaction) με την διεπαφή η οποία χρησιμοποιείται στις διαδραστικές εγκαταστάσεις ψυχαγωγίας και τέχνης, είτε αυτή είναι προβολείς, είτε οθόνες, και ιδιαίτερα το λογισμικό το οποίο δημιουργεί την διεπαφή. Η επεξεργασία και η οπτικοποίηση των δεδομένων της εφαρμογής με τρόπο τέτοιο όπως έχει οριστεί από τους δημιουργούς των σεναρίων είναι επίσης αρμοδιότητα του συγκεκριμένου επιπέδου. Έτσι, στην περίπτωση που κάποιος δημιουργήσει μία εφαρμογή στην οποία για παράδειγμα ο χρήστης πρέπει να "πιάσει" διάφορους ήχους και να τους διαχωρίσει στα αντίστοιχα κουτάκια ανάλογα με το συχνотικό τους περιεχόμενο, τότε, τόσο η δημιουργία των ήχων στην εγκατάσταση όσο και οι εικόνες της επιτυχούς ολοκλήρωσης κάθε στόχου στον χώρο (και όχι στις συσκευές) είναι ευθύνη του συγκεκριμένου επιπέδου. Έτσι, με διάφορες βιβλιοθήκες επεξεργασίας της εισόδου από τους αισθητήρες παράγεται ένα οπτικοακουστικό σκηνικό, το οποίο αυξάνει την συγκίνηση του παιχνιδιού ενώ είναι ενδεχομένως απαραίτητο στην περίπτωση της καλλιτεχνικής εγκατάστασης.

Όταν γίνεται αναφορά στον "Κόσμο", χωρίς περεταίρω επεξήγηση, τότε ο προσανατολισμός είναι η διαδικτυακή πύλη, και τεχνικά το πακέτο λογισμικού το οποίο υλοποιεί το επίπεδο αυτό. Αυτό διότι χρονικά, πρώτα σχεδιάστηκε και υλοποιήθηκε η πύλη, ένα κομ-

121

Κόσμος

μάτι απαραίτητο για την έναρξη και τον τερματισμό πλήθους εφαρμογών, και έπειτα η βιβλιοθήκη με τα πακέτα "επεξεργασίας" τα οποία υποβοηθούν τις διαδραστικές εγκαταστάσεις.

Το αρχιτεκτονικό διάγραμμα ολόκληρης της βιβλιοθήκης του "Κόσμου" ως πύλη, φαίνεται στο σχήμα της σελίδας 17. Εκεί παρουσιάζεται ένα υποσύνολο των οντοτήτων (portlets), και από αυτά αναλύεται μόνο το πρώτο στα συστατικά του στοιχείο. Η βιβλιοθήκη δομείται με κέντρο το παιχνίδι. Όλες οι ενότητες λογισμικού του "Κόσμου" χρησιμοποιούν τις τελευταίες εκδόσεις όσον αφορά τις Java και Web 2.0 τεχνολογίες, έτσι ώστε να συνδέονται με μία ή περισσότερες βάσεις δεδομένων στο πίσω μέρος (back-end) και να προσφέρουν την φιλικότερη δυνατή εμπειρία στον τελικό χρήστη (front-end). Ενεργώντας ως βάση για μία ποικιλία από πιο περίπλοκες υλοποιήσεις, αυτές οι οντότητες (portlets) αποσκοπούν επίσης στο να παρουσιάσουν, στο μέτρο του δυνατού, τις δυνατότητες συγκεκριμένης προσέγγισης. Σε απλές εφαρμογές αυτά τα portlets εξοικονομούν χρόνο και κόπο κατά την φάση της ανάπτυξης, καθώς συνήθως μόνο ένα γνήσιο υποσύνολο αυτών είναι απαραίτητο, ενώ σε πιο προχωρημένες περιπτώσεις ποικιλόμορφοι συνδυασμοί μπορούν να χρησιμοποιηθούν για την δημιουργία πιο σύνθετων οντοτήτων. Αποτελώντας βάση για περαιτέρω υλοποιήσεις, με πολύ μικρές αλλαγές ακριβώς τα ίδια portlets μπορούν να χρησιμοποιηθούν για την εισαγωγή ή ανάκτηση άλλων τύπων πληροφοριών με σκοπό την αναβάθμιση διαφορετικών τύπων ψυχαγωγικών σεναρίων.

Με την χρήση της βιβλιοθήκης αυτής, η διαδικασία καθίσταται εξαιρετικά απλή. Όπως έχει δειχθεί νωρί-

τερα, η χρήση προτύπων έχει ως αποτέλεσμα τα κομμάτια της βιβλιοθήκης να μπορούν να επαναχρησιμοποιηθούν, ανεξάρτητα από το τι έχει ο χρήστης κατά νου, ενώ ταυτόχρονα διατηρούν την υλοποίηση όσο πιο ευέλικτη γίνεται. Η ευελιξία αυτή επιτρέπει την εγκατάσταση σε διαφορετικές αρχιτεκτονικά πλατφόρμες -π.χ. χρήση διαφορετικής διαδικτυακής πύλης με άλλα χαρακτηριστικά που να ταιριάζουν περισσότερο στις ανάγκες του χρήστη (βλ. DotCMS) ενώ αποτελεί εύκαμπτη βάση για κλιμάκωση χωρίς να περιορίζει τις επιλογές του τελικού σχεδιαστή. Από την άλλη, η ανεξαρτησία της τελικής εφαρμογής, δίνει την δυνατότητα για πλήρη επαναχρησιμοποίηση αυτών των μονάδων, σε διαφορετικές εφαρμογές, παιχνίδια ή εγκαταστάσεις, ακόμα και αν το υπόλοιπο σύστημα διαφέρει δραστικά (κάτω από το επίπεδο της βάσης δεδομένων).

Για παράδειγμα, υπάρχει ένα portlet το οποίο υλοποιεί έναν οδηγό δημιουργίας νέου παιχνιδιού ο οποίος είναι προσανατολισμένος στο παιχνίδι της καυτής πατάτας. Με μηδαμινή προσπάθεια, ο χρήστης έχει την δυνατότητα να δημιουργήσει ένα αρκετά τροποποιημένο portlet "νέας μάχης" για παιχνίδια με εντελώς διαφορετικά χαρακτηριστικά. Αυτή η προσέγγιση επίσης ενισχύει την ευελιξία. Οι χρήστες έχουν την δυνατότητα να ορίσουν οντότητες βασισμένες στους δικούς τους κανόνες, και αυτή η ευελιξία διατηρείται από την σχεδίαση μέχρι την υλοποίηση (ξεχωριστή λογική σχεδιασμού, ανεξαρτησία βάσης δεδομένων κλπ). Αρκετά από τα portlets επίσης παρέχουν ιδέες οι οποίες μπορούν να επαναχρησιμοποιηθούν από τους χρήστες πάνω σε εντελώς νέες ιδέες, παρουσιάζουν δυνατότητες ενσωμάτωσης σε ακόμη περισσότερες υ-



A platform
developing mol
locative and collabora
distributed ga

Home

Games

Documentation

Community

FAQ

About

Games » Hot Potato » Games

HP Engine Map Info



Observe all game instances that are about to start, in progress, or completed via Google Maps.

Click on each sphere to get more information on the specific battle.

New Game!

Do you want to create your
game? [Click here!](#)



HP Game Info

Brief information on all game instances.

Click on a particular game instance to get analytical information on the players involved and all the related data. Examine how games progress.

Name	Location	Max Players
Saturday's Potato	CTI	15
Saturday Potato 2	CTI	8
Library Game	UPatras, Library	20
Black Engine	CTI, O.II.7	1
Rio Bay Area Potato	Rio Bay Area	1
CEID Test	CTI, PREFAB	5
TPI Friday (Game 1)	CEID, Upatras	15
TPI Friday (Game 2)	CEID, Upatras	15

Κόσμος

124

πηρεσίες στο διαδίκτυο όπως το Twitter και το YouTube, αλληλεπιδρούν με ιστοχώρους κοινωνικής δικτύωσης όπως το Facebook και το MySpace και κάνουν εκτενή χρήση πλήθους προγραμματιστικών διεπαφών -Google Maps API, ροές RSS για κάθε παιχνίδι κλπ. Με την χρήση δυναμικών, επεκτάσιμων και ευέλικτων χαρακτηριστικών, μία προσέγγιση αυτής της μορφής κλιμακώνεται, εξοικονομεί πόρους και προσαρμόζεται εύκολα, κάνοντας τα πράγματα ευκολότερα για τον τελικό χρήστη του συστήματος - είτε είναι προγραμματιστής, είτε σχεδιαστής παιχνιδιών, είτε παίκτης.

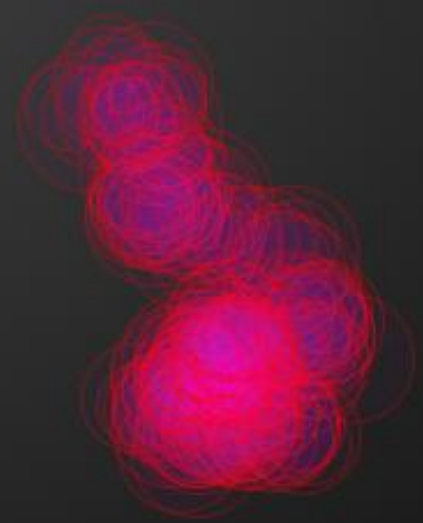
Μία τυπική υλοποίηση αποτελείται από (1) τον πηγαίο κώδικα της Java (Java source code), ο οποίος είναι απαραίτητος για την υλοποίηση της απαραίτητης λειτουργικότητας, (2) τα αρχεία ρυθμίσεων (Mapping files), τα οποία είναι διάφορα αρχεία σε μορφή XML που αντιστοιχούν διάφορες λειτουργίες και (3) τα αρχεία του "πελάτη" (Client-side files) τα οποία είναι διάφορα JSP, CSS και Javascript αρχεία που υλοποιούν την διεπαφή με τον χρήστη. Η βιβλιοθήκη προσφέρει πολύ περισσότερα από ένα εκτενές εφαλτήριο, μία στερεή βάση και για τα τρία είδη. Αυτή η προσέγγιση βοηθάει στην διατήρηση τόσο της λειτουργικότητας όσο και της δομής των μονάδων που υλοποιούνται για ένα νέο παιχνίδι στο επίπεδο του "Κόσμου". Επίσης, η χρήση σεναρίων Ant (Ant scripts) [81] κάνει ακόμα πιο εύκολη την δημιουργία νέων μονάδων χρησιμοποιώντας τις ήδη υπάρχουσες, σε μία προσπάθεια να αυτοματοποιηθεί η επαναχρησιμοποίηση ιδεών ενώ η προσαρμογή των portlet σε υψηλό επίπεδο να χρειάζεται ελάχιστη δουλειά για κάθε παιχνίδι. Αξίζει να σημειωθεί πως μία αντίστοιχη προσέγγιση χρησιμοποιείται και από το Περιβάλλον Επέκτασης του Liferay (Liferay Extension Environment) [13].

Μία τυπική άποψη χρήσης της διαδικτυακής πύλης φαίνεται στην εικόνα της προηγούμενης σελίδας. Ο χρήστης παρακολουθεί οπτικά μία περίπτωση (instance) παιχνιδιού καυτής πατάτας, ενώ παράλληλα δέχεται πληροφορίες για άλλες περιπτώσεις του ίδιου παιχνιδιού. Αξιοσημείωτη είναι η ευκολία συρσίματος με το ποντίκι (drag 'n' drop) με την οποία ο χρήστης αναδιαμορφώνει τις μονάδες της πλατφόρμας σε πραγματικό χρόνο με σκοπό την παρακολούθηση, την αλληλεπίδραση ή την διαχείριση των παιχνιδιών. Εμφανής είναι επίσης η επιλογή του οπτικού θέματος της διαδικτυακής πύλης, ένα θέμα το οποίο έχει χρησιμοποιηθεί εκτενώς και σε άλλα εργαλεία της πλατφόρμας. Καθώς κανένα από ήδη υπάρχοντα θέματα της βιβλιοθήκης θεμάτων του Liferay δεν ικανοποιούσε τις ανάγκες, έγινε πλήρης σχεδιασμός ενός νέου θέματος από την αρχή, ώστε αυτό να ταιριάζει με την όλη φιλοσοφία της πλατφόρμας. Μικρές λεπτομέρειες οι οποίες είναι δύσκολο να παρουσιαστούν μέσα από στατικές εικόνες, το κάνουν ιδιαίτερα φιλικό προς τον χρήστη και τον προ(σ)καλούν να αλληλεπιδράσει με τις μονάδες του συστήματος.

Η σουίτα επίδειξης χρησιμοποιεί Liferay 5.2 μαζί με Tomcat 6.0 για την συσκευασία των portlets της βιβλιοθήκης. Στο πίσω μέρος, το portal κάνει χρήση τόσο μίας MySQL βάσης δεδομένων όσο και μίας HSQL. Μέσω της Hibernate, στην MySQL αποθηκεύονται δεδομένα που σχετίζονται με τα παιχνίδια -όπως γεγονότα, στοιχεία χαρακτήρα- και τα portlets που δημιουργούν για τα παιχνίδια οι χρήστες της πλατφόρμας. Στην HSQL, από την άλλη πλευρά, αποθηκεύονται δεδομένα προσωποποίησης (personalization data) καθώς και πληροφορίες που σχετίζονται με τους χρήστες, όπως οι δημοσιεύσεις των χρηστών



>> Stage: 0%



Κόσμος

στις περιοχές συζητήσεων (forum), οι δημοσιεύσεις στις περιοχές ενημέρωσης (blogs), τα δεδομένα από τις ενσωματωμένες βιβλιοθήκες (wikis) κλπ και προσφέρει σε θέματα που αφορούν την κοινότητα και την ταυτοποίηση των χρηστών. Το κύριο πλεονέκτημα από μια τέτοια προσέγγιση είναι πως οι μονάδες της βιβλιοθήκης (portlets) τα οποία και σχετίζονται με τα παιχνίδια παραμένουν ανεξάρτητες από τα ανώτερα επίπεδα της πραγματικής υλοποίησης (κοινότητες, χρήστες κλπ) και τις τελικές επιλογές αγοράς και χρήσης διαδικτυακής πύλης και portlet container. Πέρα από το χαρακτηριστικό της ευελιξίας, η προσαρμογή ενός πακέτου ανοικτού λογισμικού δημιουργίας διαδικτυακών πυλών -το οποίο μάλιστα βρίσκεται σε επίπεδο επιχείρησης- στις απαιτήσεις του συστήματος έχει πολλά θετικά. Όχι μόνο παρέχονται ενσωματωμένες λειτουργίες για την διαχείριση λογαριασμών, forum, blog, περιεχομένου και κοινοτήτων αλλά και ένα προϋπάρχον σύνολο από portlets, τα οποία παρέχουν χαρακτηριστικά που ξεκινούν από διαδικτυακές γκαλερί εικόνων (image galleries) μέχρι περιοχές συζητήσεων πραγματικού χρόνου, και από σύνθετα συστήματα πρόσβασης μέχρι εργαλεία ανάκτησης κωδικών.

Το άλλο μέρος του επιπέδου αυτού, το οποίο περιλαμβάνει την αλληλεπίδραση (interaction) με την διεπαφή, είναι χτισμένο με προσανατολισμό την εκάστοτε εφαρμογή. Έτσι, ανάλογα με την εγκατάσταση η οποία υλοποιείται κάθε φορά, γίνεται ανάλογη υλοποίηση στο επίπεδο αυτό, καθώς οι απαιτήσεις κάθε φορά μπορεί να διαφέρουν εξαιρετικά. Υπάρχουν επαναχρησιμοποιούμενα κομμάτια, καθώς το επίπεδο αυτό βασίζεται σε βιβλιοθήκες (βιβλιοθήκη γραφικών, ήχου, φυσικών νόμων κλπ) με αποτέλεσμα

οι διαφορές να έγκεινται στις αυστηρές καθ' εαυτώ προδιαγραφές της εκάστοτε εγκατάστασης. Η ανεξαρτησία όμως αυτή, δίνει την δυνατότητα για εξ' ολοκλήρου διαφορετικές υλοποιήσεις, και είναι αυτό το χαρακτηριστικό το οποίο επιτρέπει στην πλατφόρμα να ακολουθεί άλλοτε καλλιτεχνικά, άλλοτε εκπαιδευτικά και άλλοτε ψυχαγωγικά μονοπάτια, με τα χαρακτηριστικά αυτά να εναρμονίζονται εξάίσια κάτω από την ομπρέλα του επιπέδου αυτού.

Αρχιτεκτονικά, το επίπεδο αυτό αποτελείται από τα πακέτα των υλοποιήσεων τα οποία εξαρτώνται από τον τρόπο μετάδοσης της πληροφορίας. Έτσι, αν πρόκειται για μία εγκατάσταση όπου θα υπάρχει κινούμενη εικόνα ενός ακαθόριστου υγρού στις τέσσερις έδρες ενός κύβου, αυτή είναι διαφορετική υλοποίηση από την περίπτωση που αναμένονται δείγματα ήχου και η επεξεργασία τους με αισθητήρες. Το μεγαλύτερο μέρος της πλατφόρμας επίδειξης έχει δημιουργηθεί με βιβλιοθήκες της Processing και συγκεκριμένα με την έκδοση 1.09. Οι κλάσεις άλλοτε κάνουν χρήση σύνθετων γραφικών, άλλοτε παράγουν κίνηση και άλλοτε συνθέτουν στις τρεις διαστάσεις. Το πιο ενδιαφέρον είναι πως σε συνδυασμό με την αξιοποίηση των πληροφοριών από τα υπόλοιπα επίπεδα της πλατφόρμας, συνθέτουν μία πρωτόγνωρη εμπειρία, η οποία προκαλεί, αλλάζει δυναμικά και διασκεδάζει.

Great minds discuss ideas, average minds discuss events, small minds discuss people.

Eleanor Roosevelt (author, speaker, politician, and activist, 1884 – 1962)

Υλοποίηση

Εισαγωγή



Με την χρήση της πλατφόρμας έχει υλοποιηθεί ένα πλήθος εφαρμογών. Τα παιχνίδια μπορεί να έχουν ψυχαγωγικό ή εκπαιδευτικό χαρακτήρα, αλλά αισθητή είναι η αλληλεπίδραση του παίκτη με τους άλλους χρήστες και τον περιβάλλοντα χώρο. Αυτό έχει ως αποτέλεσμα την παρουσία πληθώρας χαρακτηριστικών τα οποία απουσιάζουν από τα παραδοσιακά παιχνίδια – με λίγα λόγια τα παραδοσιακά παιχνίδια μπορούν να ενισχυθούν με την χρήση τέτοιων χαρακτηριστικών. Όσον αφορά τις εγκαταστάσεις, αυτές ποικίλουν και ξεκινούν από ψυχαγωγικού χαρακτήρα φτάνοντας σε καλλιτεχνικά εδάφη, εφαρμογές για μουσεία και καλλιτεχνικές εγκαταστάσεις.

Στα πλαίσια της πλατφόρμας αναπτύχθηκαν τέσσερα πιλοτικά παιχνίδια σε μια προσπάθεια να αναγνωριστούν οι απαιτήσεις και οι δυνατότητές της. Το κύριο χαρακτηριστικό αυτών των παιχνιδιών είναι ότι οι παίκτες αλληλεπιδρούν τόσο μεταξύ τους όσο και με το περιβάλλον. Αυτό επιτυγχάνεται με τη κίνηση του παίκτη στο χώρο, με κινήσεις των χεριών και άλλους τρόπους οι οποίοι αποτελούν μέσα για να πραγματο-

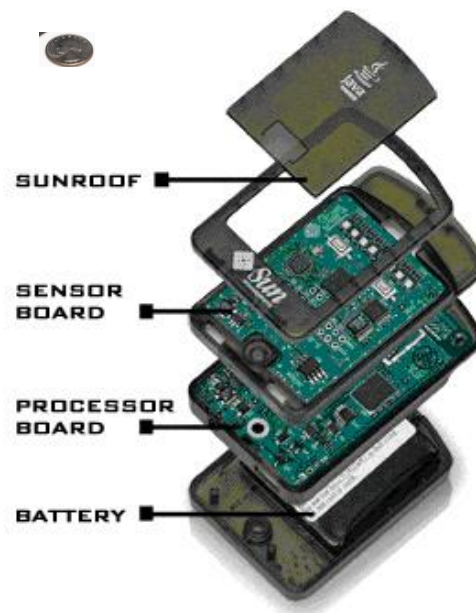
ποιηθούν συγκεκριμένες ενέργειες που σχετίζονται με το παιχνίδι. Ο παίκτης, ως μια φυσική οντότητα, είναι το κέντρο του παιχνιδιού. Η είσοδος που απαιτείται να δώσει ο χρήστης στο παιχνίδι είναι η ελάχιστη δυνατή (π.χ. με τη χρήση κινήσεων) και ταυτόχρονα έμμεση-παθητική (π.χ. βασιζόμενη στη τοποθεσία του). Κατά τον ίδιο τρόπο, η ανταπόκριση από το παιχνίδι στον παίκτη κρατιέται επίσης στο ελάχιστο (π.χ. νίκη ή ήττα) και είναι ως επί το πλείστον σποραδική (π.χ. ένδειξη ότι παίκτης έφτασε σε μία συγκεκριμένη τοποθεσία ή είναι κοντά σε έναν αντίπαλο). Υπάρχει μικρή ανάγκη για συνεχή οπτική ανταπόκριση σε σύγκριση με τα περισσότερα ηλεκτρονικά παιχνίδια του σήμερα (βλέπε τη χρήση οθονών κ.τ.λ.). Τα παιχνίδια που έχουν υλοποιηθεί μπορούν να παιχτούν οπουδήποτε και οποτεδήποτε χωρίς να είναι απαραίτητη μια προϋπάρχουσα υποδομή (backbone). Αφού το εκάστοτε παιχνίδι τελειώσει, οι παίκτες μπορούν να ανεβάσουν τα δεδομένα τα οποία οι συσκευές συνέλλεξαν σε μία διαδικτυακή πύλη (web portal) με στοιχεία κοινωνικής δικτύωσης (social networks) όπως παρουσιάστηκε προηγουμένως.

Υποδομή

Κινητές Συσκευές

Εδώ παρουσιάζεται η υποδομή των διαδραστικών εγκαταστάσεων, τόσο ψυχαγωγικών όσο και καλλιτεχνικών. Ως υλικό της πλατφόρμας για τις συσκευές των συμμετεχόντων γίνεται χρήση των SPOT της Sun (επίπεδο Φρουρών παιχνιδιού) [51]. Είναι μία μικρή συσκευή η οποία λειτουργεί με χρήση μπαταρίας εφοδιασμένη με την Εικονική Μηχανή Squawk της Java, η οποία ενεργεί τόσο ως ένα λειτουργικό σύστημα όσο και ως πλατφόρμα εφαρμογών λογισμικού, επιτρέποντας τον προγραμματισμό των συσκευών σε Java Micro Edition (JME). Η επιλογή των συσκευών αυτών έρχεται σε αντίθεση με την χρήση ακριβών κινητών τηλεφώνων και PDA τα οποία δεν θα επέτρεπαν την απλόχερη ανάπτυξη και δοκιμή των διαφόρων επιπέδων της αρχιτεκτονικής. Τα Μικρά αυτά Προγραμματιζόμενα Αντικείμενα της Sun(Small Programmable Objects Technologies – SPOTs) χρησιμοποιούν έναν ARM9 επεξεργαστή, με χρονισμό στα 180 MHz, έχουν 512 KB μνήμης RAM και 4 MB μνήμης ROM. Για τις ραδιοεπικοινωνίες, τα SPOT χρησιμοποιούν τον πομποδέκτη CC2420 Chipcon, ο οποίος είναι συμβατός με το IEEE 802.15.4. Επίσης, είναι εμπλουτισμένα με μία πολύ απλή διεπαφή χρήστη (user interface), δηλαδή δύο κουμπάκια και 8 LED, η οποία βρίσκεται στο ανώτατο επίπεδο του υλικού (πάνω από το επίπεδο του επεξεργαστή, μνήμης κλπ). Στο ίδιο επίπεδο βρίσκονται όλοι οι αισθητήρες (επιταχυνσιόμετρο, αισθητήρας γωνίας, αισθητήρας θερμοκρασίας, αισθητήρας φωτός κλπ). Πρέπει να σημειωθεί πως είναι δυνατή η εγκατάσταση και άλλων αισθητήρων καθώς τα SPOT

δίνουν την δυνατότητα αναλογικών και ψηφιακών εισόδων και εξόδων.



131

Υποδομή

Υπολογιστές

Κατά την διάρκεια της ανάπτυξης και της εκτίμησης του συστήματος, έγινε χρήση σταθερών και φορητών προσωπικών υπολογιστών σε συνδυασμό με έναν συγκεκριμένο αριθμό από Alix [73], με σκοπό το χτίσιμο της υποδομής κορμού (backbone infrastructure). Τα Alix είναι εφοδιασμένα με έναν AMD Geode επεξεργαστή με χρονισμό στα 500Mhz, 256 MB μνήμης RAM, 4GB μνήμης Compact Flash, καθώς και 2 ή 4 USB. Έρχονται με προεγκατεστημένη σύνδεση Ethernet καθώς και προαιρετική συνδεσιμότητα Wi-Fi και καταναλώνουν 4W συνολικού ρεύματος. Το μικρό τους μέγεθος τα καθιστά ιδανικά για την χρήση τους ως Σταθμούς και κυρίως Μηχανές Παιχνιδιού. Έγινε χρήση της ελαφριάς διανομής Linux Xubuntu, μία παραλλαγή του Ubuntu που χρησιμοποιεί το περιβάλλον εργασίας και διαχειριστή παραθύρων XFCE αντί του καθιερωμένου αλλά αρκετά βαρύτερου Gnome. Η επικοινωνία μεταξύ των Σταθμών παιχνιδιού έγινε μέσω του 802.11.b, ενώ η επικοινωνία της Μηχανής με την βάση (MySQL v5.0.51a) έγινε μέσω 100Mbps LAN Ethernet.

Όσον αφορά τις διαδραστικές εγκαταστάσεις, σε αυτές ήταν απαραίτητη υψηλότερη υπολογιστική ισχύς, και αυτό για την απόδοση (rendering) πολύπλοκων γραφικών σε πολύ υψηλές αναλύσεις. Έτσι, έγινε χρήση σταθερών υπολογιστών με επεξεργαστή Intel Core 2 Quad 9650 και χρονισμό στα 3.0 GHz, μνήμη στα 4GB και κάρτες γραφικών NVidia SLI 8800 με μνήμη 512 MB για να αποδώσουν χωρίς δυσκολία γραφικά με χρήση της βιβλιοθήκης OpenGL της Processing.

Το λειτουργικό σύστημα ήταν Linux και συγκεκριμένα διανομή Ubuntu 8.10. Τέλος, για το στήσιμο της δικτυακής πύλης (<http://finn.cti.gr>), χρησιμοποιήθηκε διακομιστής ο οποίος στεγάζεται στο ΕΑ.ΙΤΥ της Πάτρας.



Εφαρμογές

Πιλοτικά Παιχνίδια

Με προσοχή επιλέχθηκαν τα παρακάτω 4 παιχνίδια για να αποκαλύψουν όσο είναι δυνατόν την ποικιλία και την ευχαρίστηση που προκαλείται από εφαρμογές οι οποίες έχουν υλοποιηθεί με τη χρήση της πλατφόρμας. Πιο συγκεκριμένα ο Κινούμενος Μοναχός (Moving Monk) και το Καζανόβα (Casanova) αναδεικνύουν υπηρεσίες τοπικής ή χωρικής συναίσθησης (location awareness). Το Καζανόβα και η Καυτή Πατάτα (Hot Potato) παρουσιάζουν αλληλεπίδραση μεταξύ παιχτών σε πραγματικό χρόνο. Το Ανώνυμος (Anonymous) προσφέρει υπηρεσίες συναίσθησης των συμφραζομένων. Η Καυτή Πατάτα, το Καζανόβα, και ο Ανώνυμος υποστηρίζουν υπηρεσίες διαδικτύωσης με ανοχή καθυστέρησης. Ο Καζανόβα απαιτεί είσοδο που σχετίζεται με την κίνηση των παικτών και την μεταξύ τους απόσταση. Αυτά τα παιχνίδια έχουν υλοποιηθεί, αξιολογηθεί και παίζεται σε πληθώρα περιστάσεων σε μία προσπάθεια να αξιολογηθεί η πλατφόρμα.

Κινούμενος Μοναχός (Moving Monk): Κάθε παίκτης, ο οποίος αποκαλείται «Μοναχός», κινείται ασταμάτητα μεταξύ ενός προκαθορισμένου συνόλου «Ναών». Ο στόχος κάθε παίκτη είναι να επισκεφτεί όλους τους ναούς στον καλύτερο δυνατό χρόνο και να επιβεβαιώσει την επίσκεψη κάνοντας συγκεκριμένες «Προσευχές» σε κάθε τοποθεσία (με τις Προσευχές να ορίζονται από ένα σύνολο κινήσεων). Ο «Ναός» είναι η ακτίνα κάλυψης μίας συσκευής της υπάρχουσας υποδομής. Διάφορα στοιχεία μπορούν να δοθούν στον μοναχό, ως βοήθεια, στην προσπάθεια του να βρει τους Ναούς (κυνήγι του θησαυρού κ.τ.λ.), αλλά στην γενική περίπτωση οι παίκτες έχουν πλήρη άγνοια για τις θέσεις των ναών. Ο νικητής του παιχνιδιού είναι ο

πρώτος Μοναχός που θα καταφέρει να επισκεφτεί όλους τους Ναούς.

Καυτή Πατάτα (Hot Potato): σε αυτό το παιχνίδι κάθε συσκευή γεννάει τυχαία μια «Καυτή Πατάτα» η οποία «ανατινάζεται» μετά από κάποιο χρόνο με αποτέλεσμα να βγαίνει από το παιχνίδι ο παίκτης που κρατούσε την συγκεκριμένη συσκευή. Κάθε παίκτης μπορεί να πετάξει την πατάτα σε κάποιον από τους κοντινούς (γείτονες) παίκτες κάνοντας μια συγκεκριμένη κίνηση κρατώντας την συσκευή. Έτσι κάθε παίκτης προσπαθεί να περάσει την πατάτα από την συσκευή του σε κάποιον από τους γείτονες, ώστε να αποφύγει να αποκλειστεί με την έκρηξη της πατάτας. Αν ο παίκτης προσπαθήσει να αποφύγει να πάρει την πατάτα με το να βρίσκεται έξω από την ακτίνα των υπόλοιπων παιχτών, μια νέα πατάτα, έχει πολύ υψηλή πιθανότητα, να δημιουργηθεί στη συσκευή του. Αυτό έχει ως αποτέλεσμα να υπάρχουν περισσότερες από μια πατάτες ενεργές ταυτόχρονα στο παιχνίδι. Όταν ένας παίκτης ο οποίος ήδη κουβαλάει μια πατάτα δεχθεί ακόμη μία από ένα γείτονα του, οι δύο πατάτες ενώνονται – με την πιο πρόσφατη να είναι αυτή η οποία υπερέχει. Νικητής του παιχνιδιού είναι ο παίκτης που θα μείνει τελευταίος ζωντανός ενώ όλοι οι αντίπαλοί του θα έχουν εξουδετερωθεί.

Καζανόβα (Casanova): Αυτό είναι ένα παιχνίδι δύο παιχτών. Με τυχαίο τρόπο ο ένας από τους δύο επιλέγεται να είναι ο «Casanova» ενώ ο άλλος είναι η «Bianca». Ο στόχος του «Casanova» είναι να τρέξει μακριά από την «Bianca» ενώ της «Bianca» να μην

133

Εφαρμογές

Πιλοτικά Παιχνίδια

χάσει τον «Casanova» από τα μάτια της, τρέχοντας όποτε αυτός τρέχει και μένοντας ακίνητη όταν αυτός μένει ακίνητος. Για να ξεκινήσει το παιχνίδι ο ένας παίκτης πρέπει να βρίσκεται στην εμβέλεια του άλλου. Όταν αυτό συμβεί, υπάρχει μία φάση αρχικοποίησης κατά την οποία δίνονται ταυτότητες στους παίκτες (μια διαδικασία εκλογής αρχηγού – leader election process). Οι δύο παίκτες ενημερώνονται από την συσκευή για το ποιος είναι ποιος και το παιχνίδι ξεκινά. Η «Bianca» χάνει ένα βαθμό όταν 1) δεν είναι κοντά στον «Casanova» (3-4 μέτρα) 2) ο «Casanova» είναι ακίνητος ενώ αυτή κινείται. Ως αποτέλεσμα ο «Casanova» προσπαθεί να κερδίσει τρέχοντας μακριά από την «Bianca» ή μένοντας απότομα ακίνητος.

νός, και περισσότεροι από ένας αν είναι μαθητευόμενοι, είναι ο νικητής.

Ανώνυμος (Anonymous): Σε αυτό το παιχνίδι υπάρχει ο «Ανώνυμος» και οι «Μαθητευόμενοι». Ο «Ανώνυμος» είναι πολύ ισχυρός ενώ οι «Μαθητευόμενοι» βρίσκονται σε μία συνεχή αναζήτηση της ταυτότητας του «Ανώνυμου». Ο στόχος του «Ανώνυμου» είναι να κρύβει τον ρόλο του, να βρίσκει τους μαθητευόμενους και να τους εξοντώνει σταδιακά χρησιμοποιώντας συγκεκριμένες κινήσεις. Από την άλλη ο στόχος των «Μαθητευόμενων» είναι να διατηρηθούν στο παιχνίδι όσο το δυνατόν περισσότερο αποφεύγοντας τον «Ανώνυμο». Αν οι «Μαθητευόμενοι» αποκαλύψουν την ταυτότητα του «Ανώνυμου» μπορούν να τον εξοντώσουν συνδυάζοντας τις δυνάμεις τους, κάνοντας δηλαδή συγκεκριμένες συγχρονισμένες κινήσεις. Ο τελευταίος παίχτης ο οποίος καταφέρει να μείνει ζωντα-



Εφαρμογές

Πιλοτικές Διαδραστικές Εγκαταστάσεις

Παρακάτω παρουσιάζονται πέντε πιλοτικές διαδραστικές εγκαταστάσεις οι οποίες έχουν υλοποιηθεί χρησιμοποιώντας την πλατφόρμα. Οι εγκαταστάσεις χαρακτηρίζονται από μια σχετική απλότητα, είναι αυτοεξηγούμενες και αποτελούν πρόκληση για παίκτες όλων των ηλικιών, αλλά κυρίως μικρής ηλικίας. Όπως και πριν, βασιζόμαστε σε δύο θεμελιώδεις δυνατότητες αίσθησης από την πλευρά των συσκευών. Αυτή της παρουσίας (κοντά σε ένα σημείο ενδιαφέροντος ή κοντά σε έναν άλλο άνθρωπο) και αυτή της ανίχνευσης και αναγνώρισης κίνησης. Ένα αξιοσημείωτο πλεονέκτημα της χρήσης κινητών δικτύων αισθητήρων ad-hoc και στην περίπτωση μας των SUN Spot αντί καμερών είναι ότι κάθε συσκευή παρέχει μία μοναδική ταυτοποίηση για κάθε παίκτη, κάτι το οποίο δουλεύει παρόμοια με τα RFID (Radio Frequency Identification), και μπορεί να χρησιμοποιηθεί για τον εντοπισμό του ιστορικού ενός χρήστη ή για την βελτίωση ολόκληρης της ψυχαγωγικής εμπειρίας.

Συμπληγάδες Πέτρες (Clashing Rocks): Η κατάσταση αυτή αναπαριστά ένα αρχαίο πλοίο (τριήρης) όπου οι παίκτες έχουν τον ρόλο του «κωπηλάτη». Κάθε παίκτης κρατάει ένα κουπί στο οποίο είναι δεμένο ένα SUN Spot, το οποίο συλλαμβάνει και αναγνωρίζει την κίνηση του κουπιού. Κωπηλατώντας γρήγορα οι παίκτες προσπαθούν να αποφύγουν να συνθλιφθούν. Με το να κωπηλατούν γρήγορα και συγχρονισμένα το πλοίο σταδιακά επιταχύνει φτάνοντας κάποια μέγιστη ταχύτητα. Ένας κελευστής σημαίνει ακουστικά τον κωπηλατικό ρυθμό ο οποίος αυξάνεται καθώς οι παίκτες διατηρούν τον συγχρονισμό τους. Η εμπειρία ολοκληρώνεται με την γραφική αναπαράσταση της

προόδου του παιχνιδιού. Σε αυτό το παιχνίδι τα βασικά χαρακτηριστικά του συγχρονισμού της αναγνώρισης κίνησης και της χρήσης γραφικών συνδυάζονται.

Χρωμάτισε το! (Chromatize It!): Η συγκεκριμένη εγκατάσταση είναι ταυτόχρονα εκπαιδευτική και ψυχαγωγική και βασίζεται στη μίξη των βασικών χρωμάτων. Τα βασικά χαρακτηριστικά τα οποία παρουσιάζονται είναι η εγγύτητα μεταξύ των συσκευών, η είσοδος από τους παίκτες καθώς και η χρήση γραφικών για την αναπαράσταση της εξόδου. Μία λευκή μάζα, ένα ζητούμενο χρώμα και κάποιες στατιστικές πληροφορίες εμφανίζονται καθώς ο παίκτης πλησιάζει στην οθόνη. Επιλέγοντας μεταξύ των βασικών χρωμάτων που είναι διαθέσιμα στη συσκευή, ο παίκτης χρωματίζει την μάζα συνδυάζοντας τα σε μία προσπάθεια να πετύχει το ζητούμενο χρώμα κάθε επιπέδου. Ο επιτυχημένος συνδυασμός των χρωμάτων οδηγεί σε νέο επίπεδο μεγαλύτερης δυσκολίας, όσον αφορά τη χρωματική πολυπλοκότητα του ζητούμενου χρώματος. Περισσότεροι από έναν παίκτες μπορούν να συμμετάσχουν ταυτόχρονα.

Διεγκυστίδα - (Tug of War): Σε αυτό το ανταγωνιστικό παιχνίδι πολλών παιχτών, οι παίκτες εισέρχονται σε μια κατασκευή σε σχήμα κύβου (περίπου 2x2x2) στην κάθε πλευρά της οποίας επιπλέουν χρώματα. Η περιοχή κάθε παίχτη ορίζεται από ένα χρώμα. Ο στόχος κάθε παίχτη είναι να επεκτείνει την περιοχή του το μέγιστο δυνατό. Αυτό επιτυγχάνεται με τη σωστή και γρήγορη εκτέλεση συγκεκριμένων κινήσεων. Η γραφική αναπαράσταση καθώς και αναγνώριση κίνησης είναι τα βασικά χαρακτηριστικά της συγκεκριμένης ε-

135

Εφαρμογές

Πιλοτικές Διαδραστικές Εγκαταστάσεις

γκατάστασης. Κάθε παίκτης κατέχει μια χρωματική περιοχή την οποία στόχος του είναι να επεκτείνει ενάντια στους αντιπάλους του. Εκτελώντας τις απαιτούμενες κινήσεις γρηγορότερα απ' ό,τι οι αντίπαλοι του καταφέρνει να επικρατήσει. Δεν υπάρχει όριο όσον αφορά το πλήθος των παιχτών.

Ομαδική Δουλειά (Teamwork): Σε αυτή την εγκατάσταση αξιοποιούνται οι δυνατότητες ανίχνευσης κίνησης καθώς και οι αισθητήρες των sunspots. Επίσης χρησιμοποιούνται δύο οθόνες και δύο ομάδες παιχτών, με τους παίκτες που βρίσκονται κοντά στη μία οθόνη να ανήκουν στην αντίστοιχη ομάδα. Οι δύο ομάδες μπορούν να ανταγωνίζονται σε διάφορα "σχήματα" π.χ. πλοία που πλέουν στη θάλασσα, αυτοκίνητα που οδηγούνται στο δρόμο κ.τ.λ. Οι παίκτες εκτελούν κινήσεις με στόχο να κινήσουν τα οχήματα. Όσο περισσότεροι είναι οι παίκτες σε μία ομάδα και όσο πιο βίαιες είναι οι κινήσεις τις οποίες οι παίκτες εκτελούν, το αποτέλεσμα πολλαπλασιάζεται αντίστοιχα(π.χ. ένα αυτοκίνητο κινείται πιο γρήγορα).

Ο προφήτης (The oracle): Ένα πλήθος οθονών τοποθετούνται μέσα σε ένα μουσείο και παράλληλα ένα ερωτηματολόγιο πολλαπλής επιλογής, σχετικό με το θέμα της έκθεσης, μοιράζεται στους συμμετέχοντες. Η βασική ιδέα είναι ότι οι ερωτήσεις μπορούν να απαντηθούν από (1) έναν άνθρωπο ο οποίος θα κάνει μια κίνηση η οποία αντιστοιχεί στην σωστή απάντηση (π.χ. ένας κύκλος σημαίνει α) (2) ομάδες ανθρώπων μέσα από μια διαδικασία ψηφοφορίας. Οι σωστές απαντήσεις οδηγούν τους παίκτες στην απόκτηση συγκεκρι-

μένων στοιχείων τα οποία είναι σημαντικά για την μετάβαση στο επόμενο επίπεδο της έκθεσης. Αυτά μπορούν να είναι είτε i) ένα μοναδικό αντικείμενο π.χ. ένα κλειδί το οποίο επιτρέπει την πρόσβαση σε μία συγκεκριμένη περιοχή του μουσείου ή ii) ένας συνδυασμός στοιχείων από διαφορετικά σημεία π.χ. πάρει ένα κλειδί από το δωμάτιο α, έπειτα ένα νόμισμα από το δωμάτιο β κ.τ.λ.

Πύλη

Βιβλιοθήκη

Η βιβλιοθήκη των portlets του κόσμου περιλαμβάνει περί τα 20 portlet, τα οποία εκτελούν μία πληθώρα λειτουργιών, τόσο σε προσωπικό επίπεδο όσο και σε ομαδικό. Κοινός τους στόχος είναι η οπτικοποίηση των δεδομένων των εφαρμογών οι οποίες εκτελούνται με την βοήθεια της υποδομής κορμού, αλλά και χωρίς αυτήν, καθώς μεταφέρουν τα δεδομένα μόλις η σύνδεση με το σύστημα είναι εκ νέου διαθέσιμη. Αρχικά γίνεται μία γρήγορη και περιληπτική παρουσίαση της βιβλιοθήκης, ενώ έπειτα μία τυπική υλοποίηση δίνεται με μεγαλύτερη λεπτομέρεια. Όπως έχει αναφερθεί και αλλού, όλες οι οντότητες πειθαρχούν στο πρότυπο JSR-286 μεγιστοποιώντας την μεταφερισιμότητα τους - δεδομένου βέβαια και ότι ο container είναι συμβατός με το πρότυπο JSR-286.

- BSMarInfo: Αυτό το portlet εμφανίζει διάφορες πληροφορίες για τον Σταθμό Παιχνιδιού, όπως τις συντεταγμένες, την ταυτότητα, το όνομα κλπ, οπτικοποιημένες με την χρήση της προγραμματιστικής διεπαφής Google Maps.
- ChangeStatus: Ένα portlet για την αλλαγή της κατάστασης μίας μάχης, δηλαδή την έναρξη ή τον τερματισμό μιας ήδη καταχωρημένης μάχης από χρήστη που έχει τα κατάλληλα δικαιώματα.
- GamelInfo: Αυτό το portlet παρέχει συνοπτικές και δομημένες πληροφορίες για όλα τα στιγμιότυπα παιχνιδιού όπως όνομα, περιοχή, μέγιστο πλήθος παικτών κλπ.
- JoinGame: Το συγκεκριμένο portlet παρέχει ένα απλό τρόπο για την προσθήκη ήδη καταχωρημένων χαρακτήρων σε μία μάχη.
- ListAvatars: Ένα portlet που εμφανίζει πληροφορίες για όλους τους χαρακτήρες οι οποίοι εμπλέκονται σε ένα συγκεκριμένο στιγμιότυπο παιχνιδιού.
- MarInfo: Εμφανίζει πληροφορίες που αφορούν την Μηχανή παιχνιδιού, όπως αριθμός των παικτών σε ολόκληρο το παιχνίδι, κατάσταση του παιχνιδιού κλπ οπτικοποιημένα με χρήση του Google Maps API.
- ModifyAvatar: Αυτό το portlet βοηθά τον χρήστη στο να κάνει μετατροπές στα στοιχεία του χαρακτήρα του.
- NewAvatar: Μία διεπαφή για την δημιουργία νέων χαρακτήρων, με τον ορισμό διαφόρων στοιχείων όπως όνομα, φύλο κλπ.

Πύλη

Βιβλιοθήκη

- NewBattle: Αυτό το portlet προσφέρει στους χρήστες έναν οδηγό ο οποίος τους βοηθά στην δημιουργία νέων Μηχανών παιχνιδιού, ορίζοντας ένα πλήθος γνωρισμάτων και προσθέτοντας σταθμούς παιχνιδιού.
- NewGuardian: Ένα portlet για την καταχώρηση συσκευής - να σημειωθεί πως ένας χρήστης μπορεί να έχει πολλαπλές συσκευές.
- NewStation: Εδώ ο χρήστης καταχωρεί Σταθμό παιχνιδιού. Ο Σταθμός μπορεί επίσης να προστεθεί ταυτόχρονα και σε μία υπάρχουσα μηχανή παιχνιδιού (on-the-fly).
- NewTeam: Αυτό το portlet προσφέρει υποστήριξη για ομάδες σε συγκεκριμένους τύπους παιχνιδιών.
- Ranking: Το συγκεκριμένο portlet εμφανίζει έναν πίνακα κατάταξης, τόσο για τους παίκτες που έχουν τερματίσει όσο και για αυτούς που δεν έχουν τερματίσει. Η υλοποίηση αξιοποιεί ασύγχρονες κλήσεις στην βάση (AJAX) με αποτέλεσμα να υπάρχει διαρκής ενημέρωση όταν νέα γεγονότα καταφθάνουν.
- Spectator: Παράθυρο γεγονότων, όπου διαφορετικά γεγονότα από το παιχνίδι ομα-

δοποιούνται κατάλληλα. Η ροή των γεγονότων στο παράθυρο ενημερώνεται ασύγχρονα από την βάση με την χρήση AJAX.

- Tutorial: Ένα βοηθητικό portlet το οποίο καλωσορίζει τον χρήστη στις δυνατότητες του συστήματος.
- Winner: Σε αντιστοιχία με το portlet κατάταξης (ranking), το συγκεκριμένο εμφανίζει τον μοναδικό νικητή σε συγκεκριμένες κατηγορίες παιχνιδιού όπου δεν υπάρχει κατάταξη.

Παρακάτω παρουσιάζεται λεπτομερώς η υλοποίηση ενός από τα παραπάνω portlets. Τυχαία επιλέγεται το BSMarInfo, με τα υπόλοιπα να ακολουθούν τις ίδιες αρχές ανάπτυξης. Η συγκεκριμένη μονάδα αξιοποιεί το Google Maps API, για την οπτικοποίηση γεωγραφικών πληροφοριών. Για παράδειγμα, όταν ένα παιχνίδι διεξάγεται σε μία περιοχή, αφού οι χρήστες έχουν ορίσει σωστά την θέση μέσα από το portlet δημιουργίας παιχνιδιού, τότε τα στατιστικά εμφανίζονται σωστά οπτικοποιημένα με την χρήση χαρτών σε παγκόσμια κλίμακα, μέσα από την διαδοδική πύλη

Η χρήση υπηρεσιών όπως το Google Maps έχει αλλάξει το είδος των χρηστών που χρησιμοποιούν την γεωγραφική πληροφορία στο διαδίκτυο, κάνοντάς την.

Πύλη

Δείγμα Υλοποίησης Μονάδας

προσιτή σε όλους. Από την άλλη δίνει τη δυνατότητα στους προγραμματιστές να δημιουργούν εφαρμογές οι οποίες να βασίζονται πάνω στις υπηρεσίες που παρέχει το Google Maps αλλά να επεκτείνουν τις δυνατότητες του παράγοντας εφαρμογές με αυξημένη προστιθέμενη αξία. Η παρουσίαση και η διαχείριση γεωγραφικής πληροφορίας μέσω του διαδικτύου έχει γνωρίσει αλματώδη ανάπτυξη τα τελευταία χρόνια, φτάνοντας μέχρι το σημείο της χρήσης του φυλλομετρητή ιστού ως αντικαταστάτη της κλασσικής συσκευής γεωγραφικού συστήματος πληροφοριών για ακόμη περισσότερες εργασίες. Η εισαγωγή αυτής της υπηρεσίας της Google (και των συναφών εφαρμογών και συστημάτων που ακολούθησαν, όπως τα Yahoo! Maps της Yahoo και τα Live Maps της Microsoft) άλλαξε τον τρόπο που η γεωγραφική πληροφορία παρουσιάζεται και χρησιμοποιείται και κυρίως άλλαξε την διείσδυση της γεωγραφικής πληροφορίας σε μη αμιγώς γεωγραφικές εφαρμογές και ιστοτόπους στο διαδίκτυο. Η αλλαγή αυτή επιτελέστηκε με τρεις κυρίως τρόπους: (α) με την δυνατότητα χρήσης υψηλής ποιότητας ψηφιακών χαρτών (τόσο σε μορφή διανυσματική όσο και δορυφορικών φωτογραφιών) για σχεδόν όλον τον κόσμο, (β) με την χρήση εξελιγμένου λογισμικού και τεχνικών διαδικτυακού προγραμματισμού που επέτρεψαν εξαιρετικά υψηλές ταχύτητες πρόσβασης σε αυτά τα δεδομένα και (γ) με την δυνατότητα που παρέχεται σε άλλους προγραμματιστές να χρησιμοποιούν τις παρεχόμενες υπηρεσίες, μέσω μιας ανοικτής προγραμματιστικής διεπαφής, σε εφαρμογές διαδικτύου χωρίς να χρειάζεται η παρέμβαση του παροχέα των δεδομένων.

Η Google δίνει την δυνατότητα προγραμματιστικής πρόσβασης στους χάρτες που κάνει διαθέσιμους σε όλους μέσω του Google Maps API. Έτσι, ένα portlet εκμεταλλεύεται τις προγραμματιστικές διεπαφές που έχουν δημοσιοποιηθεί και με τη χρήση υπηρεσιών διαδικτύου (web services) μπορεί να επικοινωνεί με την εφαρμογή των χαρτών της Google (Google Maps). Η εφαρμογή προσφέρει την επιπλέον πληροφορία που ενδιαφέρεται να οπτικοποιήσει σε μορφή κατάλληλη για επεξεργασία από τις υπηρεσίες διαδικτύου μέσω μίας καθορισμένου προγραμματιστικής διεπαφής (API). Αυτό επιτρέπει την ενσωμάτωση των χαρτών της Google στις σελίδες χρηστών. Όπως θα δούμε παρακάτω, η συγκεκριμένη υλοποίηση διαχειρίζεται τα δεδομένα και τα μετατρέπει σε κατάλληλη μορφή ώστε να οπτικοποιούνται αποδοτικά στην τελική ιστοσελίδα. Έτσι, είναι δυνατή η απόδοση επάνω στο χάρτη γεωγραφικών δεδομένων από την βάση, αλλά και προσθαφαίρεση αντικειμένων στο χάρτη οποιουδήποτε από τους τρεις τύπους υποστηριζόμενων γεωμετριών: *POINT*, *POLYLINE* και *POLYGON*. Οι βιβλιοθήκες για τη διαχείριση της γεωγραφικής πληροφορίας των Google Maps είναι γραμμένες σε JavaScript, η οποία είναι από τις πιο διαδεδομένες γλώσσες προγραμματισμού που είναι ενσωματωμένες σε φυλλομετρητές ιστού.

Η δομή του πηγαίου κώδικα του Portlet έχει παρουσιαστεί και σε προηγούμενο κεφάλαιο, και το BSMarInfo portlet ακολουθεί ακριβώς αυτή την αρχιτεκτονική. Χωρίζεται στον φάκελο config, ο οποίος περιλαμβάνει όλα τα XML τα οποία περιγράφουν ιδιότητες του port

Πύλη

Δείγμα Υλοποίησης Μονάδας

let, στον φάκελο resources, ο οποίος περιλαμβάνει αρχεία πολυμέσων, στον φάκελο JS ο οποίος περιλαμβάνει αρχεία Javascript, ενώ υπάρχουν και κάποια αρχεία JSP στην ρίζα της ιεραρχίας του πηγαίου κώδικα του Portlet. Παρακάτω δίνονται αναλυτικές πληροφορίες για κάθε αρχείο που υλοποιεί το portlet.

web.xml

Το web.xml είναι ένα τυποποιημένο αρχείο περιγραφής διαδικτυακής εφαρμογής, το οποίο απαιτείται από οποιονδήποτε Java 2 Platform Enterprise Edition (J2EE) servlet container όπως ο Tomcat. Στην περίπτωση μας έχουμε ένα αρχείο .war το οποίο εγκαθίσταται στον Tomcat, και αυτό το web.xml περιγράφει αυτήν ακριβώς την εφαρμογή portlet.

<web-app> Αυτό το στοιχείο είναι η ρίζα του περιγραφής των ρυθμίσεων ανάπτυξης για την διαδικτυακή εφαρμογή.

<display-name> Το συγκεκριμένο στοιχείο περιέχει ένα σύντομο όνομα το οποίο πρόκειται να εμφανιστεί από τα εργαλεία. Το εμφανιζόμενο όνομα δεν είναι απαραίτητα μοναδικό. Στην συγκεκριμένη περίπτωση έχει τοποθετηθεί ένα πρόθεμα πριν από το ίδιο το όνομα, το οποίο αντικαθίσταται από το ant κατά την μεταγλώττιση, ανάλογα με τον στόχο για τον οποίο γίνεται η μεταγλώττιση (π.χ. Παιχνίδι καυτής πατάτας).

<listener> Το στοιχείο αυτό προσδιορίζει τις ιδιότητες εγκατάστασης ενός listener bean της διαδικτυακής εφαρμογής.

<listener-class> Το στοιχείο αυτό προσδιορίζει πως μία κλάση από την εφαρμογή πρέπει να καταχωρηθεί ως listener bean της διαδικτυακής εφαρμογής. Η τιμή του πρέπει να είναι ένα έγκυρο όνομα της κλάσης listener.

<servlet> Το συγκεκριμένο στοιχείο περιέχει τα δεδομένα δήλωσης ενός servlet. Αν ένα jsp αρχείο καθορίζεται και το αρχείο εκκίνησης είναι παρόν, τότε η jsp θα πρέπει να είναι προ-μεταφρασμένη (precompiled) και φορτωμένη (loaded).

<servlet-class> Το παρόν στοιχείο περιέχει το πλήρως έγκυρο όνομα της κλάσης του servlet.

<init-param> Το στοιχείο αυτό περιέχει ένα ζευγάρι όνομα-τιμή, ως μία παράμετρος αρχικοποίησης του servlet.

<param-name> Το συγκεκριμένο στοιχείο περιέχει το όνομα μίας παραμέτρου. Πρέπει να σημειωθεί ότι το όνομα της παραμέτρου πρέπει να είναι μοναδικό μέσα στην διαδικτυακή εφαρμογή.

<param-value> Περιέχει την τιμή της παραμέτρου αρχικοποίησης.

Πύλη

Δείγμα Υλοποίησης Μονάδας

`<load-on-startup>` Το στοιχείο αυτό καταδεικνύει ότι το συγκεκριμένο servlet θα πρέπει να είναι φορτωμένο (δηλαδή να γίνεται στιγμιότυπο και να καλείται η `init()` του) κατά την έναρξη της διαδικτυακής εφαρμογής. Το προαιρετικό περιεχόμενο αυτού του στοιχείου πρέπει να είναι ένας ακέραιος αριθμός που να δείχνει τη σειρά με την οποία το servlet πρέπει να φορτωθεί. Εάν η τιμή είναι ένας αρνητικός ακέραιος, ή το στοιχείο δεν είναι παρόν, ο container είναι ελεύθερος να φορτώσει το servlet την στιγμή που θα επιλέξει. Εάν η τιμή είναι θετικός ακέραιος ή 0, το δοχείο πρέπει να φορτώσει και να προετοιμάσει το servlet κατά την εγκατάσταση της εφαρμογής. Ο container πρέπει να εξασφαλίζει ότι τα servlets που σημειώνονται με χαμηλότερο ακέραιο έχουν φορτωθεί πριν από τα servlets τα οποία σημειώνονται με υψηλότερο ακέραιο. Ο container μπορεί να επιλέξει τη σειρά φόρτωσης των servlets με την ίδια τιμή στο πεδίο `load-on-startup`.

`<servlet-mapping>` Το συγκεκριμένο στοιχείο ορίζει μία αντιστοίχιση μεταξύ μίας διεύθυνσης URL και ενός servlet.

`<servlet-name>` Αυτό το στοιχείο περιέχει το κανονικό όνομα του servlet. Κάθε τέτοιο όνομα είναι μοναδικό μέσα στην εφαρμογή.

`<url-pattern>` Το στοιχείο αυτό περιέχει το πρότυπο της διεύθυνσης URL της αντιστοίχισης. Πρέπει να σημειωθεί ότι το συγκεκριμένο πρότυπο πρέπει να

ακολουθεί τους κανόνες οι οποίοι ορίζονται στην ενότητα 11.2 από τις προδιαγραφές του Servlet API.

`<taglib>` Αυτό το στοιχείο χρησιμοποιείται για να περιγράψει μία JSP βιβλιοθήκη ετικετών (tag library)

`<taglib-uri>` Το συγκεκριμένο στοιχείο περιγράφει μία διεύθυνση URI, σχετική με την τοποθεσία του `web.xml` η οποία ορίζει μία βιβλιοθήκη ετικετών (tag library) η οποία χρησιμοποιείται στην διαδικτυακή εφαρμογή.

`<taglib-location>` Το συγκεκριμένο στοιχείο περιέχει την διεύθυνση (ως σχετική με την ρίζα της διαδικτυακής εφαρμογής) για το που βρίσκεται το αρχείο περιγραφής της βιβλιοθήκης ετικετών (Tag Library Description).

`portlet.xml`

`<portlet-name>` Το στοιχείο αυτό περιέχει το κανονικό όνομα του portlet. Κάθε όνομα portlet είναι μοναδικό μέσα στην εφαρμογή.

`<display-name>` Το συγκεκριμένο στοιχείο περιέχει ένα μικρό όνομα το οποίο απεικονίζεται από τα εργαλεία. Έχει ένα προαιρετικό στοιχείο γλώσσας το οποίο είναι το `xml:lang`.

`<init-param>` Το στοιχείο αυτό περιέχει ένα ζευγάρι όνομα-τιμή, ως μία παράμετρος αρχικοποίησης του portlet.

141

Πύλη

Δείγμα Υλοποίησης Μονάδας

<portlet-class> Το στοιχείο αυτό περιέχει το πλήρως έγκυρο όνομα του portlet.

<param-name> Το συγκεκριμένο στοιχείο περιέχει το όνομα μίας παραμέτρου. Πρέπει να σημειωθεί ότι το όνομα της παραμέτρου πρέπει να είναι μοναδικό μέσα στην διαδικτυακή εφαρμογή.

<param-value> Περιέχει την τιμή της παραμέτρου αρχικοποίησης.

<expiration-cache> Περιέχει την βασισμένη στην λήξη προσωρινή αποθήκευση (expiration based caching) του portlet. Η παράμετρος δείχνει τον χρόνο σε δευτερόλεπτα μετά την οποία η έξοδος του portlet λήγει. Η τιμή -1 δείχνει ότι δεν λήγει ποτέ.

<supports> Το στοιχείο αυτό περιέχει το υποστηριζόμενο mime-type. Ο τύπος αυτός πρέπει να είναι μοναδικός μέσα στο portlet.

<mime-type> Το στοιχείο αυτό περιέχει το όνομα του mime-type, για παράδειγμα "text/html". Μπορεί να περιέχει και χαρακτήρες αναπλήρωσης (wildcard characters)

<portlet-info> Δέσμη στοιχείων η οποία περιέχει πληροφορίες για το portlet.

<title> Στατικός τίτλος του portlet.

<short-title> Μικρό αντίστοιχο του τίτλου του portlet.

<keywords> Το στοιχείο αυτό περιγράφει λέξεις κλειδιά οι οποίες σχετίζονται με το συγκεκριμένο portlet.

<security-role-ref> Το στοιχείο αυτό περιέχει τον ορισμό μίας αναφοράς στον κώδικα της διαδικτυακής εφαρμογής ενός ρόλου ασφαλείας (security role). Ο ορισμός περιλαμβάνει μία προαιρετική περιγραφή, το όνομα του ρόλου ασφαλείας ο οποίος χρησιμοποιείται στον κώδικα, καθώς και ένα προαιρετικό σύνδεσμο στον ρόλο. Αν ο ρόλος δεν ορίζεται, ο εγκαταστάτης (deployer) πρέπει να επιλέγει τον κατάλληλο ρόλο ασφαλείας.

<role-name> Το στοιχείο αυτό, όπως περιγράφηκε νωρίτερα, ορίζει το όνομα του ρόλου ασφαλείας.

liferay-portlet.xml

<liferay-portlet-app> Αυτό το στοιχείο είναι η ρίζα του περιγραφέα των ρυθμίσεων εγκατάστασης του portlet του Liferay.

<portlet> Το συγκεκριμένο στοιχείο περιέχει δεδομένα δήλωσης ενός portlet.

<instanciable>. Στοιχείο το οποίο δέχεται τις τιμές true και false. Αν είναι στην τιμή true, τότε το portlet μπορεί να εμφανίζεται πολλαπλές φορές σε μία σελίδα. Αν έχει τεθεί στο false, τότε μπορεί να εμφανίζεται μόνον μία φορά σε μία σελίδα. Η προεπιλεγμένη τιμή είναι false.

Πύλη

Δείγμα Υλοποίησης Μονάδας

`<portlet-name>` Το στοιχείο αυτό περιέχει το όνομα του portlet το οποίο πρέπει να είναι μοναδικό, και να είναι ταυτόσημο με το στοιχείο `portlet-name` το οποίο έχει οριστεί στο αρχείο `portlet.xml`

`<icon>` Το στοιχείο αυτό προσδιορίζει το εικονίδιο το οποίο θα αναπαριστά το συγκεκριμένο portlet. Για παράδειγμα, μία ολόκληρη ομάδα portlets που θα είχε να κάνει με θέματα μουσικής θα μπορούσε να έχει ένα εικονίδιο κασέτας ή δίσκου.

`<header-portlet-css>` Αυτό το στοιχείο προσδιορίζει το μονοπάτι του CSS το οποίο θα αναφέρεται στην κεφαλίδα της σελίδας σε σχέση με το μονοπάτι των συμφραζομένων του portlet. Στην συγκεκριμένη περίπτωση, δείχνει στον φάκελο CSS του portlet, στο `test.css`.

`<header-portlet-javascript>` Το στοιχείο αυτό, προσδιορίζει το μονοπάτι της Javascript η οποία θα αναφέρεται στην κεφαλίδα της σελίδας, και αυτό σε σχέση με το μονοπάτι των συμφραζομένων του portlet. Στην περίπτωση μας, δείχνει στον φάκελο `js` του portlet, και συγκεκριμένα στο αρχείο `test.js`

`<role-mapper>` Ο `role-mapper` επί της ουσίας προσδιορίζει μία αντιστοίχιση ρόλων (μπορούν να υπάρχουν πολλαπλές αντιστοιχίες μέσα σε ένα αρχείο ρυθμίσεων τύπου `liferay-portlet.xml`) και περιέχει δύο ονόματα τα οποία ορίζονται από το `role-name` και `role-link`. Η τιμή του `role-name` πρέπει να είναι ένας ρόλος ο οποίος να έχει οριστεί στο `portlet.xml`. Η τιμή του `role-link` πρέπει να είναι το όνομα ενός ρόλου του Liferay ο οποίος να υπάρχει στην βάση δεδομένων.

Επί της ουσίας, το στοιχείο `role-mapper` ενώνει αυτές τις δύο τιμές για να αντιστοιχίσει ρόλους από το `portlet.xml` στην βάση δεδομένων. Αυτό είναι απαραίτητο καθώς οι ρόλοι του Liferay μπορούν να περιέχουν κενά, ενώ οι ρόλοι που ορίζονται στο `portlet.xml` δεν μπορούν να περιέχουν. Αυτό επίσης προσθέτει επιπλέον ευελιξία καθώς ο πωλητής του portlet δεν χρειάζεται να γνωρίζει τίποτα για τους ρόλους του Liferay.

`<role-name>` Η τιμή του `role-name` πρέπει να είναι ένας ρόλος ο οποίος να έχει οριστεί στο `portlet.xml`.

`<role-link>` Η τιμή του `role-link` πρέπει να είναι το όνομα ενός ρόλου του Liferay ο οποίος να υπάρχει στην βάση δεδομένων.

`liferay-plugin-package.xml`

`<plugin-package>` Αυτό το στοιχείο είναι η ρίζα του περιγραφέα των ρυθμίσεων που αφορούν τις επιπρόσθετες μονάδες (plugins) του Liferay. Οι πληροφορίες αυτές είναι ορατές κατά την εγκατάσταση της μονάδας καθώς και όταν γίνεται απαρίθμηση των διαθέσιμων μονάδων από ένα αποθετήριο.

`<name>` Το στοιχείο αυτό ορίζει το όνομα του πακέτου το οποίο θα εμφανίζεται στους χρήστες.

`<module-id>` Το στοιχείο αυτό περιέχει ολόκληρο το αναγνωριστικό του πακέτου, χρησιμοποιώντας την σύνταξη του Maven [74], δηλαδή είναι της μορφής `groupId/artifactId/version/file-type`.

143

Πύλη

Δείγμα Υλοποίησης Μονάδας

<types> Το στοιχείο αυτό περιέχει την λίστα των τύπων επιτρόσθετων μονάδων τα οποία περιέχονται σε αυτό το πακέτο.

<type> Το συγκεκριμένο στοιχείο ορίζει τον τύπο της μονάδας. Έγκυρες τιμές είναι: portlet, layout-template, theme.

<tags> Το στοιχείο αυτό περιέχει μία λίστα τιμών οι οποίες κατηγοριοποιούν το portlet.

<tag> Αυτό το στοιχείο ορίζει μία τιμή η οποία κατηγοριοποιεί το portlet.

<short-description> Το συγκεκριμένο στοιχείο περιέχει μία σύντομη περιγραφή της μονάδας.

<long-description> Το στοιχείο αυτό περιέχει μία λεπτομερή περιγραφή της μονάδας. Συνίσταται να δίνονται πληροφορίες εγκατάστασης ή ενημέρωσης, αν ο διαχειριστής της διαδικτυακής πύλης πρέπει να κάνει περεταίρω βήματα για να το χρησιμοποιήσει, αφού εγκατασταθεί.

<change-log> Το στοιχείο αυτό περιέχει μία περιγραφή των αλλαγών που έχουν γίνει στην τελευταία έκδοση της μονάδας.

<page-url> Το συγκεκριμένο στοιχείο περιλαμβάνει την αρχική σελίδα της συγκεκριμένης μονάδας.

<author> Το στοιχείο αυτό περιέχει το ονοματεπώνυμο του συγγραφέα.

<licenses> Αυτό το στοιχείο περιλαμβάνει την λίστα των αδειών υπό τις οποίες η συγκεκριμένη μονάδα παρέχεται.

<license> Το στοιχείο αυτό ορίζει μία άδεια υπό την οποία παρέχεται η συγκεκριμένη μονάδα προς χρήση.

<liferay-versions> Το στοιχείο αυτό περιέχει μία λίστα των εκδόσεων της διαδικτυακής πύλης οι οποίες υποστηρίζονται από την συγκεκριμένη μονάδα

<liferay-version> Αυτό το στοιχείο ορίζει μία έκδοση της διαδικτυακής πύλης η οποία υποστηρίζεται από την μονάδα.

liferay-display.xml

<display> Το στοιχείο αυτό είναι η ρίζα του περιγράφεα εγκατάστασης ο οποίος περιγράφει τον τρόπο με τον οποίο τα συστατικά στοιχεία της διαδικτυακής πύλης κατηγοριοποιούνται και εμφανίζονται στους χρήστες ώστε να επιλέξουν πως θα διαμορφώσουν μία σελίδα του Liferay.

<category> Το στοιχείο αυτό οργανώνει και ομαδοποιεί ένα σύνολο από portlets. Ένα portlet

Πύλη

Δείγμα Υλοποίησης Μονάδας

μπορεί να υπάρχει σε περισσότερες από μία κατηγορίες. Το χαρακτηριστικό name πάνω σε αυτό το στοιχείο, αντιστοιχεί το όνομα στις ιδιότητες γλώσσας του Liferay. Για παράδειγμα, αν το όνομα κατηγορίας είναι “FinN” τότε το κλειδί στο πακέτο των πόρων της δικτυακής πύλης θα είναι “category.FinN”

<portlet> Το στοιχείο αυτό αναπαριστά ένα portlet. Το χαρακτηριστικό id πάνω σε αυτό το στοιχείο, πρέπει να αντιστοιχεί πλήρως στο μοναδικό όνομα του portlet, δηλαδή το στοιχείο portlet-name του portlet.xml.

liferay-portlet.tld

Το αρχείο liferay-portlet.tld χρησιμοποιείται για να περιγράψει μία JSP βιβλιοθήκη ετικετών (tag library), καθώς η κατάληξη αποτελείται από τα αρχικά των λέξεων “Tag Library Description”. Περισσότερες πληροφορίες μπορεί κάποιος να βρει στο [99]

liferay-plugin-package.properties

Το αρχείο liferay-plugin-package.properties περιέχει πληροφορίες σε μορφή ζευγαριών μεταβλητής-τιμής. Οι πληροφορίες αυτές είναι ορατές κατά την εγκατάσταση της μονάδας καθώς και όταν γίνεται απαρίθμηση των διαθέσιμων μονάδων από ένα αποθετήριο μονάδων διαδικτυακής πύλης. Επί της ουσίας είναι οι ίδιες πληροφορίες που υπάρχουν στο Liferay-plugin-package.xml, και δημιουργούνται με την χρήση του Πακέτου Ανάπτυξης Λογισμικού Επεκτάσεων (Plugins Software Development Kit) το οποίο μπορεί να χρησιμοποιήσει κανείς για να δημιουργήσει τόσο portlets

όσο και θέματα. Επίσης μπορεί κάποιος να το χρησιμοποιήσει για να προσθέσει εξαρτήσεις βιβλιοθηκών για το portlet του.

Javascript

Στο js βρίσκονται όλα τα αρχεία javascript τα οποία είναι απαραίτητα για το συγκεκριμένο portlet. Στο portlet το οποίο έχει επιλεγεί προς παρουσίαση, τα αρχεία javascript συνοψίζονται στο markermanager.js της Google.

Ο διαχειριστής σημείων σήμανσης είναι μία διασύνδεση μεταξύ του χάρτη και του χρήστη, σχεδιασμένη έτσι ώστε να διαχειρίζεται την προσθήκη και την αφαίρεση πολλών σημείων όταν αλλάζει η οπτική. Ο αλγόριθμος τοποθετεί τους δείκτες σε ένα πλέγμα, παρόμοιο με τους τίτλους του χάρτη. Όταν ο χρήστης αλλάζει την οπτική του, ο διαχειριστής σημείων σήμανσης υπολογίζει ποιά κελιά έχουν προστεθεί ή αφαιρεθεί από την θέα του χρήστη, και εμφανίζει ή υποκρύπτει τους αντίστοιχους δείκτες. Σε πρακτικές συνθήκες, αυτό επιτρέπει 10.000 δείκτες να καταναμηθούν σε μία μεγάλη περιοχή, και εφόσον μόνο 100 με 200 είναι ορατοί κάθε στιγμή από κάποια οπτική, ο χρήστης θα συναντήσει καλή απόδοση που αντιστοιχεί σε 100 σήματα και όχι κακή που αντιστοιχεί σε 10.000 σήματα. Πρέπει να σημειωθεί ότι ορισμένες γραμμές κώδικα είναι βελτιστοποιημένες με γνώμονα την ταχύτητα και όχι τον χώρο, με στόχο να μπορούν να επεξεργαστούν τάξεις χιλιάδων δεικτών.

145

Πύλη

Δείγμα Υλοποίησης Μονάδας

Μέσα στον φάκελο `resources` υπάρχουν όλοι οι πόροι πολυμέσων οι οποίοι αξιοποιούνται από το portlet. Μπορεί να είναι εικόνες, μπορεί να είναι video ή μπορεί να είναι εφαρμογές σε flash. Στην συγκεκριμένη περίπτωση πρόκειται για εικόνες σημείων σήμανσης, οι οποίες χρησιμοποιούνται στην απεικόνιση σημείων διαφορετικής σημασίας από τους χάρτες (Google Maps). Έτσι, η κυανή κουκίδα σημαίνει παιχνίδια τα οποία δεν έχουν ήδη ξεκινήσει, η κόκκινη παιχνίδια τα οποία έχουν τελειώσει κλπ. Φυσικά, πατώντας πάνω στην εκάστοτε κουκίδα ο χρήστης έχει την δυνατότητα να λάβει περισσότερες πληροφορίες για το συγκεκριμένο σημείο (π.χ. Αριθμός παικτών, όνομα μάχης, ταυτότητες συσκευών κ.α.).

Αρχεία Προβολής

Στο συγκεκριμένο portlet το αρχείο προβολής είναι το `view.jsp` αλλά εδώ μπορεί να υπάρχουν και άλλα αρχεία τα οποία υλοποιούν διαφορετικές λειτουργίες άμεσης προβολής με τον χρήστη. Έτσι, όταν το αρχείο μεταφράζεται από τον διακομιστή, εμφανίζεται ο τελικός κώδικας της εφαρμογής, με τα τελικά δεδομένα από την βάση. Καθώς ο κώδικας αυτός αντιστοιχεί σε ένα κομμάτι της τελικής σελίδας που θα αποκριθεί ο διακομιστής στον φυλλομετρητή του χρήστη, περιλαμβάνει κώδικα γραμμένο σε Java, κώδικα σε JavaScript, κώδικα HTML, κώδικα CSS καθώς και JSTL.

Αρχικά γίνονται κάποια imports των βιβλιοθηκών που θα χρησιμοποιηθούν. Εκτός από τα κλασσικές Java βιβλιοθήκες για τον χειρισμό δομών (π.χ. Λίστες κλπ) υπάρχουν και κάποια που αφορούν το επίπεδο της διατμηματικής διατήρησης οντοτήτων, δηλαδή της hibernate. Έτσι, γίνεται εισαγωγή των απαραίτητων managers αυτού του επιπέδου, καθώς και των κλάσεων του μοντέλου της βάσης που θα χρησιμοποιηθούν στο συγκεκριμένο portlet (εδώ είναι τα BattleEngine και Station). Τέλος υπάρχουν και κάποια imports που αφορούν την ίδια την hibernate και την διαχείριση των συναλλαγών με την βάση.

Έπειτα, αφού δίνονται κάποιες πληροφορίες στον χρήστη μέσα από HTML, δημιουργούνται κάποια αντικείμενα που χρειάζονται για τις συναλλαγές με την βάση και την διάσχιση του δέντρου οντοτήτων της. Έπειτα, καθώς αυτό το portlet έχει ως στόχο να χρησιμοποιηθεί ως μονάδα λεπτομερειών στην διαδικτυακή πύλη, δηλαδή θα απαντάται όταν ο χρήστης έχει επιλέξει ένα συγκεκριμένο παιχνίδι, γίνεται ανάκτηση της ταυτότητας του συγκεκριμένου παιχνιδιού. Αφού συλλέγεται το συγκεκριμένο αναγνωριστικό και μετατρέπεται σε αριθμό, χρησιμοποιείται ως όρισμα για να γίνει επερώτημα στην βάση μέσω των κλάσεων της hibernate. Αφού επιστρέφονται όλοι οι σταθμοί οι οποίοι σχετίζονται με το συγκεκριμένο παιχνίδι από την βάση, γίνεται οπτικοποίηση των ιδιοτήτων και των πληροφοριών του κάθε σταθμού. Τελειώνοντας με τον κώδικα του backend, πρέπει να σημειωθεί η εντολή διάπραξης της συναλλαγής, η οποία σφραγίζει επί της

Πύλη

Δείγμα Υλοποίησης Μονάδας

ουσίας την συναλλαγή, φέρνοντας τα δεδομένα μπροστά (σε άλλες περιπτώσεις, γράφοντας στην βάση).

Όσον αφορά τον κώδικα της javascript, αυτός στο συγκεκριμένο Portlet αφορά τους χάρτες της Google. Έτσι, για την φάση αρχικοποίησης, αφού δημιουργηθούν οι πίνακες με τις εικόνες και τα εικονίδια, δημιουργείται κάθε εικονίδιο ξεχωριστά, με την χρήση εικόνων από τα resources, σκιών, παραθύρων πληροφοριών και άλλων χαρακτηριστικών και τοποθετείται στον πίνακα με τα εικονίδια. Έπειτα με την κλήση της `getWeatherMarkers()` δημιουργούνται τα απαραίτητα αντικείμενα με τα σημεία σήμανσης και τα αντίστοιχα tool-tips με τις πληροφορίες που αντιστοιχούν σε κάθε σταθμό παιχνιδιού. Αυτά τοποθετούνται στον πίνακα batch ο οποίος επιστρέφεται στην μέθοδο `setUpWeatherMarkers()` η οποία στήνει τα σημάρδια στον συγκεκριμένο χάρτη με την χρήση του διαχειριστή σημείων `markermanager.js` που υπάρχει στον φάκελο `js` του συγκεκριμένου portlet. Έτσι, αν ο φυλλομετρητής του χρήστη είναι συμβατός, δημιουργείται ο χάρτης, στο συγκεκριμένο `div`, και τίθενται μερικές ιδιότητές του στις επιθυμητές τιμές, ενώ στο τέλος καλείται και η συνάρτηση η οποία εγκαθιστά τα σημεία σήμανσης πάνω σε αυτόν.

Με αυτήν ακριβώς την λογική έχουν φτιαχτεί και τα υπόλοιπα portlets. Μία διαφορά μπορεί να είναι πως δεν γίνεται χρήση των ίδιων αρχείων και μεθόδων javascript, με αποτέλεσμα εντελώς διαφορετική λειτουργικότητα. Μία άλλη διαφορά μπορεί να είναι πως γίνεται ανάγνωση πληροφοριών από άλλους πίνακες

της βάσης, με εντελώς διαφορετικές ιδιότητες τα οποία πρέπει να συνδυαστούν καταλλήλως για να παράγουν διαφορετικά αποτελέσματα. Μία ειδοποιός διαφορά είναι πως μπορεί να γίνεται εγγραφή στην βάση, και αυτό γίνεται από την αντίστοιχη κλάση Java του φακέλου `portlet` (π.χ. `NewAvatarPortlet`). Εκεί, αφού γίνεται προβολή με την μέθοδο `donview`, την λειτουργικότητα αναλαμβάνει η `processAction()`, η οποία με την ίδια ακριβώς λογική, δημιουργεί ένα αντικείμενο, θέτει τις κατάλληλες ιδιότητες και το εγγράφει στην βάση με την βοήθεια της `Hibernate`.

Ανάλογα λοιπόν με την επιθυμητή λειτουργία του portlet, αυτό πρέπει να προγραμματιστεί ανάλογα, πρέπει να τοποθετηθούν οι κατάλληλοι πόροι στον φάκελο `resources`, να γίνει σωστά η αντιστοίχιση στα `xml` διαμόρφωσης, να γραφτεί ο κατάλληλος Java κώδικας ο οποίος και θα υλοποιεί την καθ' εαυτώ λειτουργικότητα της διαδικτυακής εφαρμογής, να γραφτεί ο κατάλληλος Javascript κώδικας για την χρηστική διεπαφή, να τοποθετηθούν οι αντίστοιχες βιβλιοθήκες στους φακέλους `Js` και `CSS`, και φυσικά να γραφτούν τα στατικά δεδομένα σε μορφή HTML για την ανάγνωση από τον φυλλομετρητή του χρήστη. Με την ίδια ακριβώς λογική έχει δημιουργηθεί ολόκληρη η βιβλιοθήκη των portlets της διαδικτυακής πύλης.

147

Θεματολογία

Οργάνωση Πληροφορίας

Οι πληροφορίες που παρέχονται από την διαδικτυακή πύλη είναι δομημένες με συγκεκριμένο τρόπο και παρουσιάζονται στην αγγλική γλώσσα. Στην αρχική σελίδα παρουσιάζονται τα νέα της πλατφόρμας, καθώς υπάρχει συνεχής δημιουργία νέων παιχνιδιών, παρουσίαση δημοσιεύσεων, εξήγηση κανόνων και παρουσίαση εγκαταστάσεων. Επίσης δίνονται συνοπτικά τα σημαντικότερα σημεία της πλατφόρμας. Στο δεξί μέρος, υπάρχει μία στήλη στην οποία εμφανίζονται πληροφορίες σε μορφή βίντεο. Είναι ένας εύσχημος τρόπος ώστε οι χρήστες να καταλάβουν τις δυνατότητες της πλατφόρμας, να δουν σε λεπτομέρεια σκηνές δράσης και να καταλάβουν τι ακριβώς σημαίνει διάχυτα παιχνίδια και διαδραστικές εγκαταστάσεις. Στην σελίδα των παιχνιδιών εμφανίζονται αρχικά συνοπτικά μερικά δείγματα παιχνιδιών (π.χ. Ο Κινούμενος Μοναχός, η Καυτή Πατάτα, ο Καζανόβα κλπ) καθώς και μερικές εκπαιδευτικές εγκαταστάσεις (π.χ. Διεγκυστίνδα, ΧρωμάτισέΤο!) . Πατώντας στον σύνδεσμο “περισσότερες πληροφορίες”, ο χρήστης μεταφέρεται στην σελίδα κάθε εφαρμογής, όπου παρουσιάζονται λεπτομερώς οι κανόνες, υπάρχει εκτενές βίντεο που εξηγεί την εφαρμογή, παρουσιάζεται η διεπαφή χρήστη καθώς επίσης παρέχονται εκτενείς πληροφορίες οι οποίες σχετίζονται με την εφαρμογή. Έτσι κάθε παιχνίδι έχει την δικιά του σελίδα, και παιχνίδια τα οποία βρίσκονται υπό ανάπτυξη, όπου πολλοί χρήστες, δημιουργοί σεναρίων, εκπαιδευτικοί και προγραμματιστές μπορούν να συνεργαστούν, να ανταλλάξουν ιδέες και να δημιουργήσουν ανταγωνιστικές και διασκεδαστικές εφαρμογές.

Όσον αφορά τις υπόλοιπες πληροφορίες της πύλης, υπάρχουν κάποιες εικόνες, βίντεο και παρουσιάσεις στην υποσελίδα της γκαλερί, τα οποία είναι τοποθετημένα και οργανωμένα ανά γεγονός (δημόσιο event, συνέδριο κλπ). Στην σελίδα της τεκμηρίωσης, υπάρχουν γενικές πληροφορίες σε μορφή διαδικτυακών εγγράφων, παρουσιάζεται η καθολική αρχιτεκτονική του συστήματος, παρατίθενται σύνδεσμοι για τα περιβάλλοντα ανάπτυξης, το διαδικτυακό περιβάλλον διαχείρισης του κώδικα, του έργου (στόχοι, φάσεις, εργασίες κλπ) καθώς και στο περιβάλλον που είναι υπεύθυνο για την μεταγλώττιση του κώδικα (nightly builds) και την παροχή στατιστικών πάνω σε αυτόν. Τέλος παρέχονται κάποιες παρουσιάσεις σε μορφή εγγράφων και το σύνολο των δημοσιεύσεων από την ομάδα ανάπτυξης. Ακόμη, έχει δημιουργηθεί μία ιστοσελίδα συχνών ερωτήσεων (Frequently Asked Questions – FAQ) οι οποίες, ιδιαίτερα στις αρχές που τα έγγραφα τεκμηρίωσης δεν ήταν τόσο καλά δομημένα, στόχευαν στο να παρουσιάσουν διάφορα χαρακτηριστικά της πλατφόρμας στον μέσο χρήστη, απλά και εύληπτα. Στην τελευταία σελίδα δίνονται πληροφορίες για τους προγραμματιστές του συστήματος, τα χρηματοδοτούμενα έργα καθώς και την σχολή των Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής.

Έτσι, ήταν δυνατή η δημιουργία ξεχωριστών μικροσελίδων (game micro-sites) μέσα στην δικτυακή πύλη της πλατφόρμας. Υπάρχει ένα οικοσύστημα από ιστο-

Θεματολογία

Δομή Διαδικτυακής Πύλης

σελίδες παιχνιδιών, ιστοσελίδες κοινοτήτων και ιστοσελίδες παικτών το οποίο προσφέρει ανεξαρτησία διαχείρισης. Κάθε μικρο-σελίδα έχει τα δικά της μονάδες, την δικιά της κοινότητα συντήρησης και διαχείρισης (η οποία μπορεί να είναι ακόμη και ένα άτομο) και τους δικούς της κανόνες πρόσβασης. Αυτό έχει ως αποτέλεσμα ένας χρήστης, για παράδειγμα, να μπορεί να φιλοξενήσει portlets δημόσιας πρόσβασης τα οποία δείχνουν τις ικανότητές του ή τα επιτεύγματά του σε κάποιο παιχνίδι (ή περισσότερα από ένα) καθώς και ιδιωτικής πρόσβασης τα οποία του επιτρέπουν να αλλάζει πανοπλία, να προγραμματίζει τις επόμενες συγκρούσεις κλπ. Από την άλλη πλευρά, στο ίδιο παράδειγμα, μία ιστοσελίδα παιχνιδιού μπορεί να φιλοξενεί δημόσια portlets τα οποία εμφανίζουν διάφορες κατατάξεις ενώ και ιδιωτικά τα οποία μπορεί να είναι portlets-πανδοχεία όπου οι παίκτες μπορούν να αγοράσουν αντικείμενα, να συζητήσουν με άλλους παίκτες και να ανταλλάξουν ικανότητες. Υπάρχει ένα απειροσύνολο δυνατοτήτων, καθώς μικρο-σελίδες συζητήσεων, blog, πληροφόρησης, σεναρίων, επεξεργασίας εγγράφων και οτιδήποτε άλλο μπορούν να συνδυαστούν με τις μονάδες της βιβλιοθήκης. Ακόμη, πρέπει να σημειωθεί πως αυτά τα χαρακτηριστικά συνδυάζονται -ή, καλύτερα, εναρμονίζονται- με μία κεντροποιημένη διαχείριση της πλατφόρμας, δηλαδή "παιχνίδια υπό ανάπτυξη", "νέες ιδέες" κλπ.

Έπειτα, με την προσέγγιση αυτή, είναι δυνατός ο διαχωρισμός του επιπέδου των ιδεών -κανόνες παιχνιδιού, σενάρια παιχνιδιών, νέα παιχνίδια- από το επίπεδο της ανάπτυξης. Για παράδειγμα, ένα σύνολο από παιδαγωγούς ξεκινά μία μικρό-σελίδα και εκδίδει

ένα εξαιρετικό σενάριο για παιχνίδι προσθέτοντας άθλους (quests) σε μουσεία και βιβλιοθήκες τα οποία εναρμονίζονται πλήρως με συγκεκριμένες παιδαγωγικές δραστηριότητες. Έπειτα, μία ξεχωριστή και εντελώς ανεξάρτητη ομάδα μπορεί να δουλέψει στο να προσθέσει τις οντότητες που χρειάζονται από την ήδη υπάρχουσα βιβλιοθήκη, να προσαρμόσει κάποιες και ενδεχομένως να δημιουργήσει κάποιες άλλες. Με τον συνδυασμό όλων αυτών, ολοκληρώνεται και ένα νέο παιχνίδι με ολοκληρωμένα χαρακτηριστικά. Έτσι, συναντήσεις με σκοπό την παραγωγή ιδεών λαμβάνουν χώρα στην δικτυακή πύλη (όπου και φιλοξενούνται), όπου διάφοροι άνθρωποι οι οποίοι δημοσιεύουν ιδέες πάνω σε νέα σενάρια και κανόνες παιχνιδιών δεν είναι απαραίτητα και οι άνθρωποι οι οποίοι τα υλοποιούν - οι άνθρωποι άλλωστε κάνουν καλύτερα αυτό που τους αρέσει να κάνουν.

Ένα τελευταίο χαρακτηριστικό το οποίο αξίζει να σημειωθεί είναι ο συνδυασμός παιχνιδιών. Οι χρήστες μπορούν να δημιουργήσουν παιχνίδια τα οποία αποτελούνται από διάφορα υποπαιχνίδια, δηλαδή άθλους ή αποστολές οι οποίες αναπτύσσονται από ανεξάρτητους ανθρώπους ή διαφορετικές κοινότητες. Αυτά τα παιχνίδια μπορεί να περιέχουν εντελώς διαφορετικά ή ακόμη και ασύμβατα είδη παιχνιδιών με αποτέλεσμα να δημιουργούν εντυπωσιακούς συνδυασμούς. Κάτι τέτοιο φαίνεται επίσης να αποτελεί ανακούφιση στους χρήστες οι οποίοι δεν νιώθουν αρκετά σίγουροι για την πλήρη υλοποίηση ενός νέου παιχνιδιού αλλά προτιμούν να προσθέσουν απλώς ένα χαρακτηριστικό ή που δεν έχουν σκεφτεί ολόκληρο σενάριο αλλά συμμετέχουν δίνοντας μία συγκεκριμένη αποστολή.



Welcome!

A platform for developing mobile, locative and collaborative distributed games

- Home
- Games
- Gallery
- Documentation
- FAQ
- About

Games

// games

Moving Monk

"Past. Present. Future."



Each player is a monk moving continuously amongst a pre-made set of chambers/temples.

The goal of each player is to visit the temples as fast as possible, perform specific prayers in each location and be the first to pray on them all. A temple is defined by the coverage range of a base station and the prayers performed are specific gestures.

To help the monk find the temples there is a chance of clues given regarding the exact location of a temple.

Find out [more](#).

Pssst!

Dont' forget: You can also define your own rules, even create your very own game and finally play it!. And it also takes less than 10 minutes! Simple and Efficient. Check out the [documentation](#) of the platform.

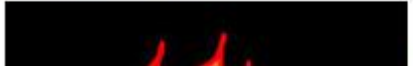
We use Sun SPOT

Sun Microsystems developed the SPOT platform, a small, battery operated, device running the Squawk Java Virtual Machine (VM) without an underlying OS, built upon the IEEE 802.15.4 standard.



Hot Potato

"Loads and loads of fun!"



A self-generated hot potato has a per-second decreasing counter. Each player can randomly pass the potato to one of the neighboring players,

Εγκαταστάσεις

Chromatizelt!

Όσον αφορά την εγκατάσταση “Χρωμάτισέ Το!” υπάρχουν δώδεκα επίπεδα από τα οποία τα πρώτα έξι είναι η “απλή” μίξη των βασικών χρωμάτων (όλοι οι συνδυασμοί με ίση αναλογία μεταξύ τους ή μηδέν). Αυτό κάνει και εύκολη την πρόοδο στο επόμενο επίπεδο - είτε από τον ίδιο παίκτη είτε αναλάβει άλλος στην συνέχεια την ίδια συσκευή- και εύκολη την υλοποίηση από τα κατώτερα επίπεδα της ιεραρχίας καθώς από τις συσκευές ο παίκτης θα δίνει “εύκολες” αναλογίες (δηλαδή της τάξης 1-0). Αυτό σημαίνει πρακτικά πως δεν θα χρειαστεί δηλαδή να 133 φορές κόκκινο 22 πράσινο και 88 μπλε. Θα δίνει 1 κόκκινο και 1 πράσινο και αυτό είναι κίτρινο. Στα επόμενα επίπεδα, τα πιο απαιτητικά, η μόνη διαφορά είναι ότι η αναλογία είναι δύο φορές από το ένα χρώμα και ένα από κάποιο άλλο – ή ενδεχομένως κανένα. Σε επίπεδο χρήστη υπάρχει ένα εφέ που δείχνει μία λάμψη η οποία σταδιακά εξασθενεί και χάνεται (γνωστό ως fade out) και δείχνει τι χρώμα προσέθεσε ή αφαίρεσε ο χρήστης από τα βασικά ώστε να μπορεί να το διακρίνει και στην οθόνη εκτός από την συσκευή αλλά και για να μπορούν να το διακρίνουν και όλοι οι παρευρισκόμενοι. Η μάζα αλλάζει χρώμα σιγά-σιγά, για να καταλήξει στο ζητούμενο χρώμα. Υπάρχουν επίσης κάποια στατιστικά στο πλάι της οθόνης, όπως χρόνος, ένταση του προστιθέμενου ή αφαιρούμενου χρώματος κλπ, τα οποία χρησιμεύουν ώστε να βοηθήσουν τον χρήστη να αντιληφθεί τις έννοιες της εγκατάστασης, και εντείνουν την αγωνία.

Όσον αφορά την υλοποίηση, αυτή αποτελείται από τρεις δομικές κλάσεις και μία κλάση η οποία αφορά τα εφέ της αλλαγής επιπέδου. Η κλάση MorphomeBody

δημιουργεί ένα αντικείμενο το οποίο θα μπορούσαμε να χαρακτηρίσουμε ως δομικό στοιχείο του “σώματος” του μορφώματος το οποίο απεικονίζεται τελικά στην οθόνη. Έτσι, το δομικό αυτό στοιχείο περιλαμβάνει μεταβλητές όπως οι συντεταγμένες του στην οθόνη, η διάμετρός του, η συχνότητα με την οποία ταλαντεύεται κλπ. Οι μέθοδοι της κλάσης αυτής επί της ουσίας αποτελούν βασικές λειτουργίες αυτού του δομικού στοιχείου. Έτσι, υπάρχει η μέθοδος `moveHead(int x, int y)` η οποία είναι υπεύθυνη για την περίπτωση που η συγκεκριμένη δομική μονάδα είναι η κεφαλή του σώματος, η μέθοδος `moveBody(int x, int y)`, η οποία είναι η υπεύθυνη για την αναπαράσταση της μετακίνησης κάθε δομικού στοιχείου, η μέθοδος `display()`, η οποία αποδίδει τα τελικά γραφικά στην οθόνη και η μέθοδος `speedUp()` η οποία αλλάζει δυναμικά τις συχνότητες ταλάντωσης κάθε μέρους του τελικού μορφώματος. Αξίζει να σημειωθεί πως οι δύο πρώτες μεταβλητές υλοποιούν μία συνάρτηση της μορφής

- $x(t) = x(t-1) + \sin(\text{radians}(\gamma\omega\nu\acute{\iota}\alpha)) * \text{radiiX}$
- $y(t) = y(t-1) + \cos(\text{radians}(\gamma\omega\nu\acute{\iota}\alpha)) * \text{radiiY}$

για τους δύο άξονες αντίστοιχα. Στην περίπτωση βέβαια που δεν πρόκειται για την κεφαλή, γίνεται έλεγχος για τυχόν συγκρούσεις στους άξονες, και αυτές επιλύονται με την βοήθεια γεννήτριας τυχαίων αριθμών.

Η κλάση Morphome είναι ένα επίπεδο πιο πάνω, και υλοποιεί επί της ουσίας το μόρφωμα ως μία οντότητα, με πολλαπλά στιγμιότυπα της προηγούμενης κλάσης

151

Εγκαταστάσεις

Chromatizelt!

MorphomeBody. Έτσι, περιέχει μεταβλητές όπως η πολυπλοκότητα (με αυξημένη πολυπλοκότητα να ανεβάζει ανάλογα και τις απαιτήσεις σε επεξεργαστική ισχύ), την ταχύτητα σε κάθε άξονα κλπ. Η συνάρτηση δημιουργός κατασκευάζει ένα σύνολο από δομικές μονάδες καλώντας την MorphomeBody με ψευδοτυχαίες ιδιότητες (παίρνουν τιμές μέσα σε ένα διάστημα), ώστε να δοθεί μία φυσικότητα στο σύνολο των μονάδων. Επίσης υπάρχουν και δύο μέθοδοι που υλοποιούν την κίνηση στην οθόνη, η halt() και η move(), με την πρώτη να εμφανίζει “παγωμένη” την μάζα, ενώ η δεύτερη την μετακινεί σύμφωνα με έναν αλγόριθμο, και αξιοποιεί τις μεθόδους της MorphomeBody για τις δομικές λειτουργίες.

Όσον αφορά την κλάση titles, αυτή αποτελεί μία προγραμματιστική διεπαφή (Abstract Programming Interface – API) με διάφορες μεθόδους οι οποίες υλοποιούν εφέ πραγματικού χρόνου με δυναμικές παραμέτρους. Έτσι, είναι δυνατή η παρουσίαση πλήθους κειμένων και μικρών φράσεων (π.χ. “You Win!”) με διάφορους τρόπους. Υλοποιήθηκε με τέτοιο τρόπο ώστε να χρησιμεύει και σε άλλες εγκαταστάσεις, και εδώ έχει χρησιμοποιηθεί για την εμφάνιση διασκεδαστικών μηνυμάτων κατά την αλλαγή του επιπέδου. Έτσι, δημιουργεί ένα αντικείμενο τίτλων, το οποίο αποτελείται από μοναδικούς τίτλους, και υλοποιεί συναρτήσεις σταδιακής εμφάνισης ή εξασθένησης (fade in και fade out), φωτίζει την περίμετρο των γραμμάτων, αλλάζει τα ποσοστά διαφάνειας, χαμηλώνει ή αυξάνει τον φωτισμό στο παρασκήνιο (σβήσι-

μο) κ.α. Η ίδια κλάση χρησιμοποιείται και από τις άλλες εγκαταστάσεις για τους αντίστοιχους τίτλους.

Τέλος, όσον αφορά την κλάση chromatizelt(), αυτή αποτελεί την κεντρική κλάση της συγκεκριμένης εφαρμογής. Είναι μία τυπική κλάση της Processing, περιέχοντας τις συναρτήσεις setup(), draw() κλπ όπως είδαμε σε προηγούμενο κεφάλαιο. Περιέχει ένα πλήθος μεταβλητών, όπως είναι οι εικόνες που χτίζουν κάθε επίπεδο, οι εικόνες του μενού, οι γραμματοσειρές, οι πολυπλοκότητα του μορφώματος, μεταβλητές επιπέδων κλπ. Αρχικά, στην setup() αρχικοποιούνται οι μεταβλητές και στήνεται το περιβάλλον (π.χ. Η smooth() ομαλοποιεί την απόδοση των σχημάτων, η size(1024,768) ορίζει το μέγεθος της επιφάνειας, η setFrameRate(30) ορίζει τον αριθμό των καρτών ανά δευτερόλεπτο κλπ) ενώ επίσης τρέχουν οι συναρτήσεις οι οποίες αρχικοποιούν τα γραφικά. Η draw(), η οποία καλείται επαναληπτικά, είναι η βασική συνάρτηση της εφαρμογής. Επί της ουσίας κάνει συνεχή αίτηση για δημιουργία ενός επιπέδου (με συγκεκριμένες ιδιότητες), και αν αυτή ολοκληρωθεί (δηλαδή βγει νικητής) τότε δημιουργεί κάποια εφέ και ξανατρέχει την setup για το επόμενο επίπεδο. Η public void createUI(int offset) δημιουργεί όλα τα στατικά γραφικά του κάθε επιπέδου, δηλαδή το παρασκήνιο, τα μενού, τα χρονόμετρα κλπ. Επίσης, είναι υπεύθυνη για την εμφάνιση του μορφώματος, καλώντας τις απαραίτητες συναρτήσεις και τροφοδοτώντας τις με τις αντίστοιχες παραμέτρους. Η συνάρτηση αυτή είναι υπεύθυνη για το γεγονός ότι ο κώδικας της setup() και

152

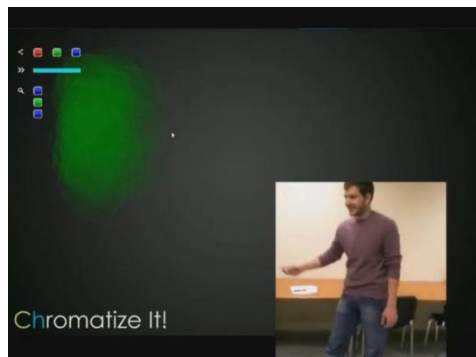
Εγκαταστάσεις

ChromatizeIt!

`draw()` έχει κρατηθεί σε τόσο μικρό μέγεθος. Η `createStage(int stage, int offset)` δημιουργεί κάθε επίπεδο με τις ορισμένες ιδιότητές του. Εδώ υπάρχει ο στόχος κάθε επιπέδου, ο οποίος αν επιταχυνθεί από τον χρήστη, εγκρίνεται η αίτηση της `draw()` για νέο επίπεδο. Προφανώς η `createStage(int stage, int offset)` είναι αυτή που καλεί και την `createUI` για να μπορέσει να προβάλει τα γραφικά κάθε επιπέδου. Ακόμη, υπάρχει η `void checkStroke()`, υπεύθυνη για τον φωτισμό της περιμέτρου κάθε μορφώματος όταν πετιέται ένα χρώμα, η `void initFader(int _barWidth)` υπεύθυνη για το ουράνιο τόξο που εμφανίζεται κατά την αλλαγή επιπέδου και η `public String showtime()` η οποία εμφανίζει ένα χρονόμετρο από την αρχή της προσπάθειας κάθε παίκτη, μέχρι το επίπεδο στο οποίο φτάνει. Τέλος, η `public void keyPressed()` εξομοιώνει τις κινήσεις των SPOT με πλήκτρα του πληκτρολογίου, ενώ η `public void update(Observable obs, Object arg)` είναι υπεύθυνη για την πραγματική επικοινωνία με τα υπόλοιπα επίπεδα.

Αξίζει να σημειωθεί πως η συγκεκριμένη εγκατάσταση όπως έχουν οριστεί οι παράμετροι της άμορφης μάζας, σε φορητό υπολογιστή με μνήμη 2048MB 667MHz DDR2 SDRAM (2x1024), επεξεργαστή Intel® Core™ 2 Duo Processor T9300 (2.50 GHz, 800 MHz FSB, 6 MB L2 cache) και κάρτα γραφικών 128 MB NVIDIA® GeForce® 8400M GS, τα καρέ παραμένουν στα 30 fps σε Windows 7 και λίγο λιγότερα σε Linux. Στους υπολογιστές που εν τέλει χρησιμοποιήθηκαν για τις εγκαταστάσεις, τα καρέ παρέμεναν στην κορυφή, καθώς είναι καθαρά θέμα παραμετροποίησης. Προφανώς πρόκειται για ένα παζάρι εντυπωσιασμού, όπου αν

ανέβει υψηλά η πολυπλοκότητα, το αποτέλεσμα είναι πιο φανταχτερό, αλλά “κολλάει” σε υπολογιστές μέτριας ισχύος.



153

Εγκαταστάσεις

Tug of War

Η διελκυστίνδα (Tug Of War) βασίζεται στην ίδια λογική, με την διαφορά ότι η απόδοση των γραφικών (rendering) γίνεται με OpenGL. Τα αποτελέσματα είναι εντυπωσιακά, και, τηρουμένων των αναλογιών, είναι πιο φιλικό όσον αφορά την κατανάλωση υπολογιστικών πόρων.

Για να είναι δυνατή η χρήση επιτάχυνσης OpenGL από την Java, απαραίτητη είναι η χρήση κάποιων εγγενών βιβλιοθηκών ανάλογα με το λειτουργικό σύστημα που χρησιμοποιείται καθώς επίσης και την αρχιτεκτονική εντολών του υπολογιστή. Η τελευταία έκδοση των αρχείων αυτών βρίσκεται και στον οπτικό δίσκο που συνοδεύει την δυτλωματική εργασία, ενώ οι οδηγίες για την εγκατάσταση διαφέρουν ανάλογα με την εγκατάσταση συστήματος η οποία χρησιμοποιείται. Έτσι, αν πρόκειται για Windows, το μόνο που χρειάζεται είναι να τοποθετηθούν τα JOGL native .dll's στον φάκελο `JDK_HOME/bin/` (`JDK_HOME` είναι ο αρχικός φάκελος του kit ανάπτυξης της Java). Όμοια, αν πρόκειται για Linux, απαιτείται τοποθέτηση των `gluegen-rt` και `jogl jars` καθώς και των native .so's στο `JDK_HOME/jre/lib/ext/`.

Η λειτουργικότητα επιτρέπει εξομίωση τεσσάρων πλευρών ενός κύβου σε μία έξοδο. Ακόμα και σε μία οθόνη, όταν το πλάτος ισοδυναμεί με τέσσερις φορές το ύψος, νοητά κάθε ένα από τα τέσσερα διακριτά ζευγάρια της μορφής (ύψος, ύψος) είναι η πλευρά ενός κύβου. Με λίγα λόγια αυτό που βλέπει ο χρήστης στην οθόνη διπλώνει γύρω-γύρω νοητά και το τέλος του είναι η αρχή του (βλ. σχήμα). Οπότε όταν γράφε-

ται κάτι στην οθόνη, γράφεται στην αντίστοιχη πλευρά στην οποία το φαίνεται να είναι νοητά τοποθετημένο. Άλλωστε, αν αντί για μία έξοδο στον υπολογιστή τοποθετηθούν τέσσερις έξοδοι, και η εικόνα μοιραστεί ομοιόμορφα, τότε έχουμε τις τέσσερις προβολές στις πλευρές του κύβου, με κάθε πλευρά να αντιδρά ανεξάρτητα και να προβάλλει σε κάθετη επιφάνεια σε σχέση με τις δύο γειτονικές της.

Καθώς πρόκειται για εγκατάσταση, το σενάριο επαναλαμβάνεται τακτικά όταν κάποιος παίκτης βγει νικητής. Χωρίς διακοπή, το σύστημα ανακοινώνει τον νικητή, και μετά από κάποια εφφέ είναι έτοιμο πάλι να υποδεχθεί τους νέους παίκτες. Λόγω συγκεκριμένων μεταβλητών κατά την αρχικοποίηση κάθε επανάληψης του σεναρίου, κάθε φορά το παιχνίδι είναι διαφορετικό. Ακόμη και οι πλευρές οι οποίες έχουν κάποια ιδιότητα διαφέρουν, ο τρόπος που ανακοινώνεται ο νικητής ή η σειρά των πλευρών που σκοτεινιάζουν για να υποδεχθούν το νέο παιχνίδι (dim lights). Κάθε γύρος δεν κρατάει πολύ. Οι μεταβάσεις δεν καθυστερούν πολύ αν σκεφτεί κάποιος ότι μέσα σε αυτό το διάστημα πρέπει να αλλάξει η ομάδα των χρηστών που είναι μέσα στον κύβο, ενώ ταυτόχρονα γίνεται εξήγηση των κανόνων.

Σεναριακά, η ακολουθία γεγονότων στο παιχνίδι είναι η ακόλουθη. Αρχικά παρουσιάζονται κάποια αστεία μηνύματα που υποδηλώνουν ότι φορτώνεται το περιβάλλον της εγκατάστασης. Τα μηνύματα αυτά είναι τυχαία κάθε φορά, εμφανίζονται σε διαφορετική πλευρά του κύβου και καθώς το παρασκήνιο είναι

154

Εγκαταστάσεις

Tug of War

λευκό, δείχνουν να χάνονται – εξασθενούν με την εμφάνιση ενός νέου μηνύματος. Έπειτα, και οι τέσσερις πλευρές σκοτεινιάζουν, και τέσσερις σκοτεινές μάζες διαφορετικού χρώματος κάνουν την εμφάνισή τους. Οι μάζες αυτές θυμίζουν χαρακτηριστικά μποκέ, θολά φώτα όταν το μάτι δεν έχει εστιάσει πάνω τους, και σε αυτή την περίοδο τα πλήκτρα είναι απενεργοποιημένα. Οποιαδήποτε κίνηση και αν επιχειρήσουν οι χρήστες των συσκευών, αυτή δεν περνάει στις μάζες. Παραμένοντας έτσι για 3 δευτερόλεπτα μέχρι να προετοιμαστούν όλοι οι παίκτες, ξαφνικά οι άμορφες μάζες ακτινοβολούν, θυμίζοντας βυθό διαφορετικού χρώματος, υγρό του οποίου τα δομικά στοιχεία αναδιατάσσονται αέναα. Τώρα οι παίκτες έχουν την δυνατότητα να παίζουν, και με κατάλληλες κινήσεις, πρέπει να επεκτείνουν την υγρή μάζα που τους αντιστοιχεί έναντι των υπολοίπων. Ένας παίκτης χάνει αν και μόνον αν η μάζα του έχει εξαφανιστεί πλήρως. Όταν οι τρεις από τις τέσσερις μάζες εξαφανιστούν, ο παίκτης που έχει καταλάβει και όλες τις πλευρές του κύβου αναδεικνύεται αυτόματα νικητής. Τότε η μάζα του συρρικνώνεται στην πλευρά από την οποία ξεκίνησε, σχηματίζοντας μία μπάλα φωτός η οποία εκρήγνυται γεμίζοντας χρώματα τον κύβο. Καθώς αυτά τα χρώματα χάνονται γύρω, σταδιακά οι πλευρές γίνονται πάλι λευκές, η μία μετά την άλλη με ένα εντυπωσιακό εφέ, και ένας νέος γύρος ξεκινά από την αρχή.

Όσον αφορά την υλοποίηση, αυτή αποτελείται από τρεις δομικές κλάσεις και μία κλάση η οποία αφορά τα εφέ της αλλαγής επιπέδου, την Titles, η οποία είναι ταυτίσιμη με την Titles των άλλων εγκαταστάσεων. Η κλάση του κατώτατου επιπέδου, των βασικών δομι-

κών στοιχείων κάθε μάζας είναι η Circle, η οποία δεν είναι κάτι διαφορετικό από έναν κύκλο με επαυξημένες ιδιότητες. Η κλάση αυτή, κρατάει πληροφορίες όπως η θέση του (x,y) η ταχύτητά του σε κάθε άξονα, η γωνία του, το μέγιστο και ελάχιστο της περιοχής στην οποία μπορεί να βρίσκεται μέσα σε μία οθόνη, το χρώμα του, την πλευρά στην οποία αντιστοιχεί κλπ. Επίσης, περιέχει κάποιες μεθόδους, οι οποίες έχουν να κάνουν με την θέση, την κίνηση και το χρώμα κάθε κύκλου, έτσι ώστε να μπορεί να δημιουργήσει τα βασικά εφέ κίνησης κάθε δομικού στοιχείου και να τα απεικονίσει ανάλογα. Έτσι, στον δημιουργό αρχικοποιούνται οι μεταβλητές και αντιστοιχίζεται κάθε κύκλος που δημιουργείται σε ποιά πλευρά του κύβου ανήκει. Η συνάρτηση draw() είναι εξαιρετικά απλή και ζωγραφίζει τον κύκλο, δεδομένων της θέσης και της ακτίνας. Πρέπει να σημειωθεί πως μόνον αυτές οι πληροφορίες είναι απαραίτητες για την απεικόνιση του κύκλου, ενώ οι υπόλοιπες απαιτούνται για τον υπολογισμό της θέσης κάθε στιγμή, ανάλογα με τον αλγόριθμο ο οποίος δημιουργεί την ψευδαίσθηση της άμορφης μάζας.

Οι τέσσερις συναρτήσεις συστολής και διαστολής (void shrinkLeft(int _side, int ratio), expandLeft(int _side,int ratio), shrinkRight(int _side, int ratio), expandRight(int _side, int ratio)) , μικραίνουν ή μεγάλωνουν την περιοχή μέσα στην οποία μπορεί να κινηθεί ένας κύκλος, με τα σύνορα της μίας ομάδας κύκλων που αποτελούν την μάζα ενός παίκτη να είναι το όριο για μία άλλη ομάδα κύκλων ενός άλλου παίκτη. Η συνάρτηση move() αναλαμβάνει την κίνηση

155

Εγκαταστάσεις

Tug of War

των κύκλων μέσα στην συγκεκριμένη περιοχή, και μόλις ένας κύκλος ακουμπήσει τα τοιχώματα της περιοχής μέσα στην οποία πρέπει να είναι, τότε αλλάζει κατεύθυνση ανάλογα με την προσπίπτουσα. Τέλος η μέθοδος `blow()` χρησιμεύει για την δημιουργία του εφφέ της σφαίρας ενέργειας λευκού χρώματος, όταν ένας παίκτης κερδίζει.

Η κλάση `Interactome` αντιπροσωπεύει ένα στιγμιότυπο άμορφης μάζας, και το πώς αυτή συμπεριφέρεται, συγκεντρώνοντας έναν αριθμό από αντικείμενα της κλάσης `Circle`. Έτσι περιλαμβάνει την αναλογία χρωμάτων που είναι απαραίτητη για την δημιουργία του συγκεκριμένου αριθμού στιγμιότυπων, και ανάλογα με την ζητούμενη πολυπλοκότητα, δημιουργεί και τους αντίστοιχους κύκλους. Η μέθοδος `display()` εμφανίζει την κάθε μάζα στην αντίστοιχη πλευρά του κύβου, ενώ αν πρόκειται για νικητή, πυροδοτεί το `bow()` εφφέ. Η `serene()`, μετά το `blow()`, εξασθενεί την ενέργεια των χρωματιστών κύκλων, για να έρθει η επανεκκίνηση του παιχνιδιού. Η `expandTerritory()` μεγαλώνει την περιοχή της συγκεκριμένης μάζας αριστερά και δεξιά, ενώ παράλληλα μειώνει λίγο την αριστερή από τα δεξιά της, και την δεξιά από τα αριστερά της. Το αποτέλεσμα είναι ότι η συγκεκριμένη εμφανίζεται μεγαλύτερη έναντι των γειτονικών της. Τέλος, οι `int getRight(int me)` και `int getLeft(int me)`, διατρέχουν τις μάζες για να βρουν ποιός είναι ο δεξιός και ο αριστερός γείτονας αντίστοιχα, καθώς αυτό που ήταν στην προηγούμενη προσπάθεια μπορεί να έχει αντικατασταθεί από έναν άλλον ο οποίος τον εξαφάνισε.

Όσον αφορά την κλάση `tows`, αυτή είναι η κεντρική κλάση της εφαρμογής η οποία πυροδοτεί και τα αποτελέσματα των άλλων τριών. Αρχικά περιλαμβάνει κάποιες γενικές μεταβλητές της εγκατάστασης, όπως για παράδειγμα ο αριθμός των πλευρών – μπορεί να γίνεται προβολή σε πεντάγωνο ή εξαγώνο κλπ- η πολυπλοκότητα, η μέγιστη και η ελάχιστη διάμετρος των δομικών στοιχείων της κάθε μάζας κλπ. Η `setup()` δημιουργεί το περιβάλλον (η `size` έχει ως τρίτο όρισμα το `OPENGL`), αρχικοποιεί τις μεταβλητές και δημιουργεί τις τέσσερις μάζες. Ξεκινώντας αντίστροφα, η `void initializeTitles()` δημιουργεί την δεξαμενή των τίτλων, από όπου και “τραβιούνται” τυχαία κάποιοι όταν εμφανίζονται τα μηνύματα που φορτώνουν την εφαρμογή. Η μέθοδος `void createRandomness()` είναι αυτή που επιλέγει το ποιοί εν τέλει θα είναι αυτοί οι τίτλοι που θα εμφανιστούν. Έπειτα, η μέθοδος `boolean createTransition(int frames)` είναι υπεύθυνη για ένα εφφέ που θυμίζει βροχή από ημιδιαφανείς τελείες, με αποτέλεσμα, κατά την αρχή κάθε επιτέδου, μετά τους τίτλους, να σκοτεινιάζει η οθόνη για να αποτελέσει καμβά για τις τέσσερις μάζες. Η `public void keyPressed()` εξομοιώνει τις τιμές των αισθητήρων στο πληκτρολόγιο, ενώ η `void eraser(int _color, int _alpha)` σβήνει ότι υπάρχει στην οθόνη επαναληπτικά, αφαιρώντας το ίχνος ενός χρώματος σε κυμαίνουσα ένταση. Τέλος, η `void glow()` είναι υπεύθυνη για την απόδοση του εφφέ των αυτόφωτων υγρών μαζών, με την χρήση της `OPENGL`. Όσο για την `draw()`, αυτή χωρίζεται σε φάσεις ανάλογα με την καθ’ εαυτώ φάση του παιχνιδιού. Στην πρώτη φάση εναλλάσσονται οι τίτλοι,

156

Εγκαταστάσεις

Tug of War

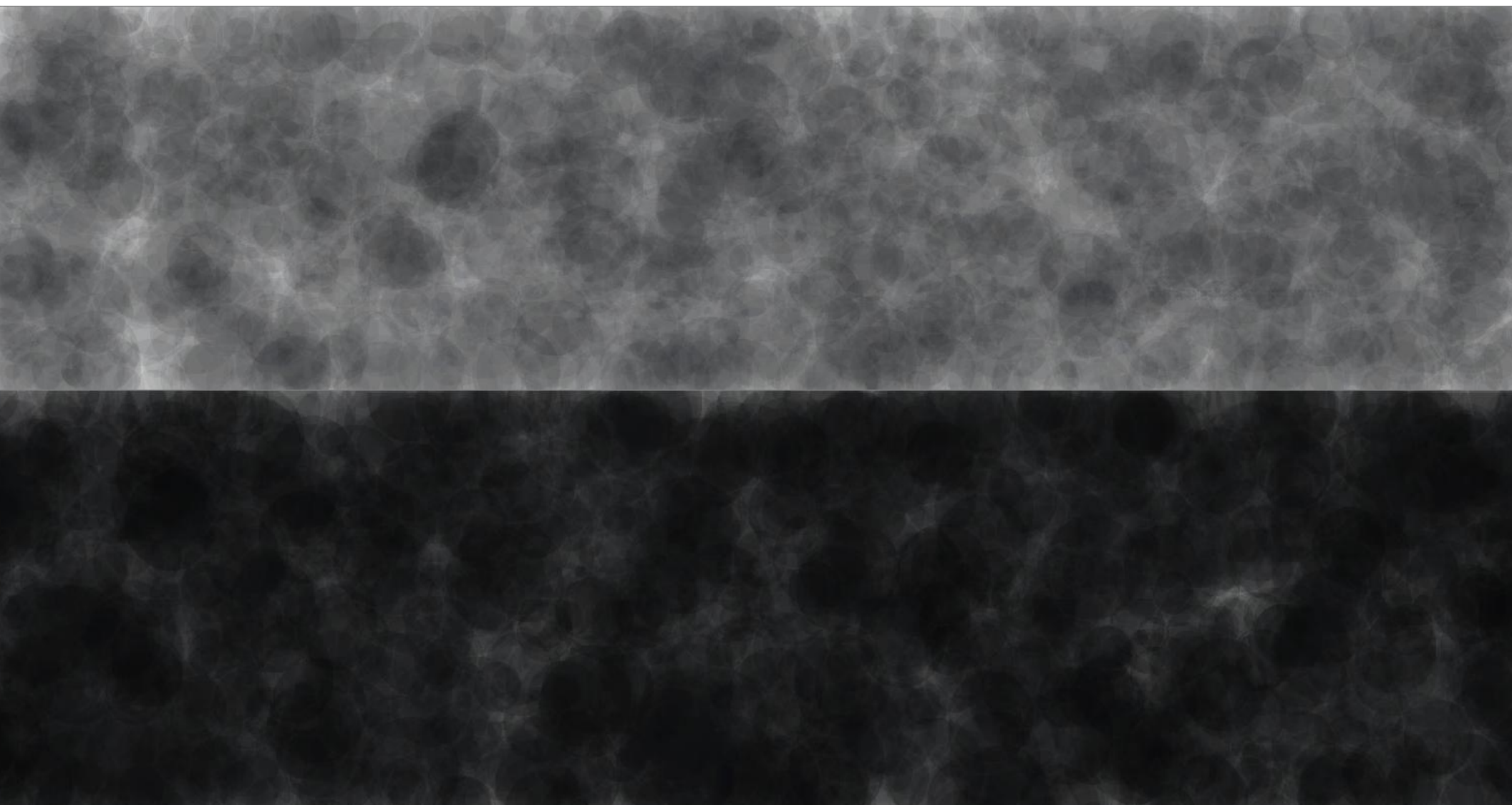
στην δεύτερη σκοτεινιάζει η οθόνη για να υποδεχθεί τις μάζες, στην τρίτη δημιουργείται το τελικό περιβάλλον με τις μάζες αρχικά να είναι κλειδωμένες, έπειτα να ξεκλειδώνονται οι μάζες με τον χώρο να βρίθεται χρωματικών αποχρώσεων, και τελικά να συρρικνώνονται στην φωτεινή σφαίρα, στην πέμπτη αφού η σφαίρα εκραγεί όλα γυρνάνε στο λευκό χρώμα, στην έκτη δημιουργούνται οι τίτλοι του τέλους και παρουσιάζεται ο νικητής, στην έβδομη σβήνουν η μία μετά την

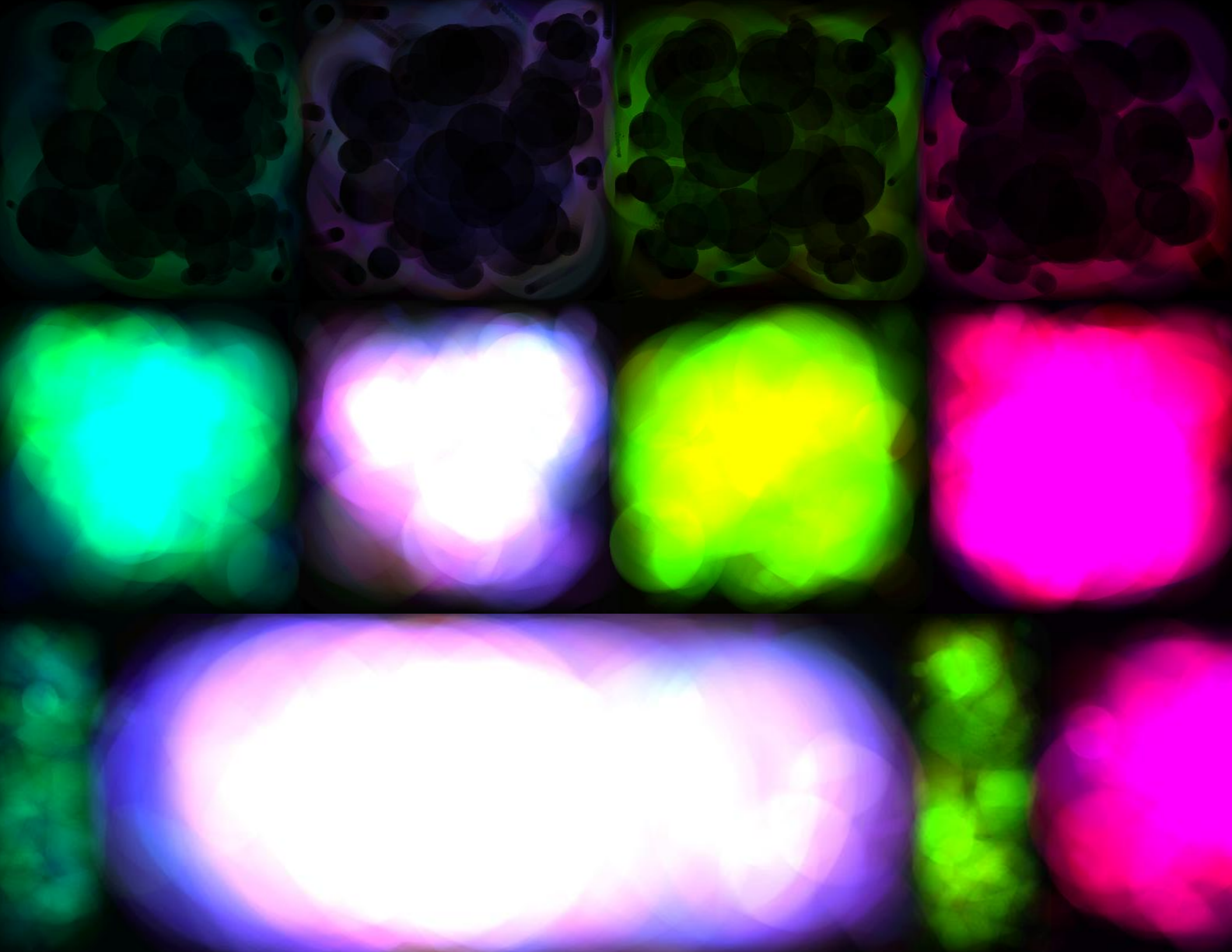
άλλη όλες οι πλευρές του κύβου, και στην όγδοη ξαναγεννιέται το λευκό φώς ταυτόχρονα για να υποδεχθεί το νέο παιχνίδι.

Στα επόμενα σχήματα παρουσιάζεται ένα ολόκληρο στιγμιότυπο παιχνιδιού, όπως περιγράφηκε σεναριακά στις προηγούμενες σελίδες. Σε κάθε “καρέ” εμφανίζονται οι τέσσερις πλευρές του κύβου (4 παίκτες) ταυτόχρονα. Ο μπλε παίκτης κερδίζει τον γύρο.



Mixing Ingredients...







<http://finn.cti.gr>

<http://finn.cti.gr>

<http://finn.cti.gr>

<http://finn.cti.gr>

p://finn.cti.gr

http://finn.cti.gr

http://finn.cti.gr

http://finn.cti.gr

Μεταγλώττιση

Για την μεταγλώττιση των δομικών μονάδων του έργου, γίνεται χρήση του εργαλείου Ant. Το Apache Ant είναι ένα εργαλείο λογισμικού για την αυτοματοποίηση των διαδικασιών μεταγλώττισης λογισμικού. Είναι παρόμοιο με το Make, αλλά έχει υλοποιηθεί με την χρήση της γλώσσας Java, απαιτεί την πλατφόρμα αυτή για να τρέξει και είναι το πλέον κατάλληλο εργαλείο για την μεταγλώττιση έργων γραμμένων σε Java. Η πιο άμεσα αισθητή διαφορά μεταξύ του Ant και του Make, είναι ότι το Ant χρησιμοποιεί XML για να περιγράψει την διαδικασία μεταγλώττισης και τις εξαρτήσεις του, ενώ το Make χρησιμοποιεί τα Makefiles. Εξ' ορισμού, το αρχείο αυτό ονομάζεται build.xml. Το Ant είναι ένα έργο του οργανισμού Apache, και είναι λογισμικό ανοιχτού κώδικα, με άδεια χρήσης την Apache Software License.

Ιστορικά, το Ant (Another Neat Tool - Άλλο ένα Πετυχημένο Εργαλείο) επινοήθηκε από τον James Duncan Davidson καθώς μετέτρεπε ένα προϊόν της Sun σε ανοιχτού κώδικα. Αυτό το προϊόν, μετέπειτα έγινε ο Apache Tomcat. Μία εμπορική έκδοση του εργαλείου make χρησιμοποιούνταν για την μεταγλώττιση του στο λειτουργικό σύστημα Solaris, αλλά στον κόσμο του ανοιχτού λογισμικού δεν υπήρχε τρόπος κάποιος να ελέγξει σε ποιά πλατφόρμα θα γινόταν η μετάφραση του Tomcat. Έτσι, δημιουργήθηκε το Ant ως ένα απλό εργαλείο για την μεταγλώττιση του Tomcat από οδηγίες σε ένα αρχείο μετάφρασης (build file) σε μορφή XML. Από την αρχή, το εργαλείο αυτό έμελε να γίνει πιο διαδεδομένο από τον ίδιο τον Tomcat για τον οποίο δημιουργήθηκε. Το Ant (έκδοση 1.1) εκδόθηκε επίσημα ως αυτόνομο προϊόν τον Ιούλιο του 2000. Πολλά εργαλεία έχουν προταθεί ως η έκδοση 2 του Ant, συμπεριλαμβανομένου του AntEater του ίδιου

δημιουργού, του Myrmidon και του Mutant, κανένα από τα οποία δεν κατάφερε να βρει μεγάλη αποδοχή από την κοινότητα προγραμματιστών. Σήμερα το Ant χρησιμοποιείται από τα περισσότερα έργα που βασίζονται σε Java. Για παράδειγμα, οι περισσότεροι προγραμματιστές ανοιχτού λογισμικού περιλαμβάνουν αρχεία build.xml με τον πηγαίο κώδικά τους. Το εργαλείο αυτό, μπορεί να χρησιμεύσει και στην ενσωμάτωση εργαλείων δοκιμών, εγκατάστασης εφαρμογών κ.α. Εργαλεία της ίδιας κατηγορίας αποτελούν το Maven και το JavaMake.

Για να μεταγλωττίσει κάποιος τον κώδικα και να παίξει κάποιο παιχνίδι, πρέπει να ακολουθήσει μερικά απλά βήματα. Με την χρήση του Ant η διαδικασία μεταγλώττισης έχει αυτοματοποιηθεί, και έτσι είναι πολύ εύκολο για κάποιον ο οποίος δεν έχει σχέση με τον κώδικα, να παίξει με τις ήδη υπάρχουσες εφαρμογές ή ακόμη και να γράψει μία δικιά του. Καταρχήν, πρέπει να υπάρχει πρόσβαση στον πηγαίο κώδικα του συστήματος, είτε από τον οπτικό δίσκο, είτε από την ιστοσελίδα του έργου για την τελευταία έκδοση (γίνεται χρήση του εργαλείου Subversion [80] για την διαχείριση του κώδικα, οπότε είναι απαραίτητη η αντίστοιχη εφαρμογή πελάτη). Γίνεται μεταγλώττιση του συστήματος με την χρήση του αρχείου ANT με όνομα build.xml και στόχο "jar". Για παράδειγμα, σε ένα περιβάλλον UNIX, ο χρήστης πρέπει να γράψει "ant.jar". Έπειτα, απαραίτητο είναι το ανέβασμα του κώδικα στις συσκευές, με την χρήση του αρχείου build-XX.xml, ανάλογα με την εφαρμογή (για παράδειγμα για το παιχνίδι της καυτής πατάτας είναι όπου XX το HP, ενώ για την περίπτωση του κινούμενου μοναχού είναι το MM). Αφού γίνει σύνδεση των συσκευών με τον υπολογιστή μέσω μίας mini

162

Μεταγλώττιση

USB σύνδεσης, πρέπει να γίνει χρήση του στόχου “guardian” με την παροχή του αριθμού της θύρας στην οποία είναι συνδεδεμένη η συσκευή. Για παράδειγμα, σε ένα περιβάλλον Unix, αν η συσκευή είναι συνδεδεμένη στο /dev/ttyACM0, η αντίστοιχη εντολή θα είναι:

Κώδικας:

```
ant -f build-MM.xml -Dport=/dev/ttyACM0 guardian.
```

Έπειτα, απαιτείται μία σύνδεση στην διαδικτυακή πύλη (finn.cti.gr) και αφού δημιουργηθεί νέος λογαριασμός για τον νέο χρήστη, με την μετάβαση στην υποσελίδα του συγκεκριμένου παιχνιδιού -όπως παρουσιάστηκε νωρίτερα- ακολουθούμενη από τον σύνδεσμο αλληλεπίδρασε (interact), ο χρήστης μπορεί να δημιουργήσει ένα νέο παιχνίδι, με την χρήση του portlet “Νέο Παιχνίδι” (New Game). Απαιτείται ακόμα η δημιουργία ενός σταθμού παιχνιδιού, με την χρήση του ίδιου αρχείου Ant με πριν, για παράδειγμα σε ένα περιβάλλον UNIX:

Κώδικας:

```
ant -f build-MM.xml -DengineIP=<localIP> engine.
```

Τέλος, από το ίδιο αρχείο απαιτείται η δημιουργία ενός σταθμού παιχνιδιού, για παράδειγμα αν ο σταθμός έχει συνδεθεί στο /dev/ttyACM0:

Κώδικας:

```
ant -f build-MM.xml -DengineIP=<engine's IP> -  
DstationIP=<localIP> -Dport=/dev/ttyACM0 station.
```

Αν οι συσκευές είναι φορτισμένες, το παιχνίδι μπορεί να ξεκινήσει!

163

The definition of insanity is doing the same thing over and over again and expecting different results.

Albert Einstein (Theoretical Physicist, Philosopher and Author, 1879 –1955)

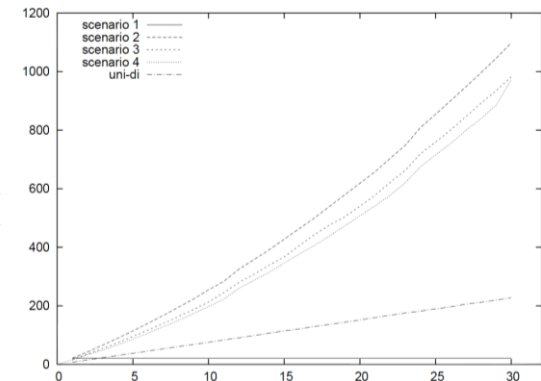
Επίλογος

Αξιολόγηση

Εδώ παρουσιάζεται η εμπειρία η οποία αποκομίζεται από την υλοποίηση και λειτουργία της συγκεκριμένης πλατφόρμας και των εφαρμογών που περιγράφηκαν σε προηγούμενα κεφάλαια. Εκτός από τα πειράματα στο εργαστήριο, έγινε πλήρης χρήση του συστήματος σε πραγματικές συνθήκες, καθώς η αξιολόγηση παιχνιδιών θα πρέπει να γίνει με ανθρώπους οι οποίοι πραγματικά παίζουν το κάθε παιχνίδι και ζουν την εμπειρία σε πρώτο πρόσωπο. Είναι σημαντικό να αναφερθεί πως δεν ήταν δυνατή η συγκέντρωση στατιστικών στοιχείων από καλλιτεχνικές και ψυχαγωγικές εγκαταστάσεις, οι οποίες στήθηκαν χωρίς δυνατότητες καταγραφής στατιστικών. Έγινε επιλογή διεξαγωγής πειραμάτων τόσο σε εσωτερικούς χώρους όσο και σε εξωτερικούς, καθώς ο στόχος αυτών των εφαρμογών είναι να λειτουργήσουν σε όλα τα δυνατά περιβάλλοντα. Πιο συγκεκριμένα, η αξιολόγηση σε εξωτερικό χώρο έγινε στο πίσω μέρος της σχολής μας (κτήριο Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής, τυπικά γνωστό ως Β κτήριο) καθώς και στο πίσω μέρος του Ερευνητικού Ακαδημαϊκού Ινστιτούτου Τεχνολογίας Υπολογιστών στην Πάτρα, μία ανοιχτή περιοχή καλυμμένη με γρασίδι χωρίς εμπόδια (δέντρα, θάμνους κλπ). Τα πειράματα εσωτερικού χώρου έγιναν i. σε ένα δωμάτιο με διαστάσεις 10x15x3 μέτρα ii. σε ένα στεγασμένο αίθριο με διαστάσεις 15x15x8 μέτρα, με τέσσερις κολώνες διαμέτρου ενός μέτρου σχετικά κοντά μεταξύ τους. Και οι δύο εσωτερικοί χώροι περιβάλλονταν από τσιμεντένιους τοίχους.

Όσον αφορά τα πειράματα, διάφορες συνεδρίες παιχνιδιών οργανώθηκαν σε ομάδες από 5..14 ανθρώπους. Ο συνολικός αριθμός των ατόμων οι οποίοι έλαβαν μέρος στο πείραμα ήταν 23. Όλοι οι παίκτες οι οποίοι

έλαβαν μέρος ήταν μεταξύ 23 και 37 ετών, με το 75% αυτών να έχουν ένα τεχνικό υπόβαθρο ενώ το υπόλοιπο 25% να προέρχεται από διάφορες επιστήμες. Κάθε ομάδα παικτών συμμετείχε σε πολλαπλές συνεδρίες, με μερικούς χρήστες να συμμετέχουν σε διαφορετικές αναλογίες ομάδων, όσον αφορά τον αριθμό των παικτών. Όσον αφορά τις ποσοτικές μετρήσεις των πειραμάτων, έγινε χρήση περεταίρω μονάδων λογισμικού ώστε να είναι δυνατή η μέτρηση μίας ποικιλίας μεταβλητών.



Σύγκριση των χρόνων επεξεργασίας, σε σχέση με τον αριθμό των κόμβων στα πέντε σενάρια που παρουσιάστηκαν προηγουμένως. Στον οριζόντιο άξονα βρίσκεται ο αριθμός των κόμβων ενώ στον κάθετο άξονα παρουσιάζεται ο χρόνος επεξεργασίας σε ms.

Αξιολόγηση

Γενική Συζήτηση

Τα πειράματα τα οποία διεξήχθησαν με τον συγκεκριμένο τύπο υλικού δείχνουν πως ενώ οι συσκευές ασύρματων δικτύων αισθητήρων μπορούν να χρησιμοποιηθούν σε περιρρέοντα παιχνίδια πολλών παικτών και διαδραστικές εγκαταστάσεις, υπάρχει ένας αριθμός πρακτικών περιορισμών στις υλοποιήσεις τέτοιων εφαρμογών. Συγκεκριμένα, έγινε διερεύνηση της απόδοσης του συστήματος όσον αφορά τις ανάγκες εμπλοκής ενός μεγάλου αριθμού χρηστών στις εφαρμογές αυτές. Πέντε διαφορετικοί τύποι σεναρίων εξετάστηκαν, όπου υπήρχε ένα σύνολο κόμβων, με ένα σύνολο ήδη γνωστών γειτόνων (scenario 1), σε μία τοπολογία πλήρως συνδεδεμένη (scenario 2), έπειτα με γείτονες οι οποίοι δεν ήταν γνωστοί (scenario 3), έναν άγνωστο γείτονα μίας κατεύθυνσης (scenario 4) καθώς και η περίπτωση συνδέσμων μίας κατεύθυνσης (uni-di). Η εικόνα 3 παρουσιάζει τον υπολογιστικό χρόνο της ανακάλυψης γειτόνων καθώς και υπηρεσιών παρακολούθησης και καταγραφής έτσι ώστε να παραχθεί η σωστή ενημέρωση της κατάστασης του δικτύου. Γίνεται αντιληπτό ότι οι εφαρμογές οι οποίες απαιτούν επικοινωνία διπλής κατεύθυνσης πραγματικού χρόνου (π.χ. το παιχνίδι της Καυτής Πατάτας ή η εγκατάσταση της Διεγκυστίνδας) δεν μπορούν να εξυπηρετήσουν έναν μεγάλο αριθμό παικτών (δηλαδή, αν οι ενημερώσεις πρέπει να είναι ορατές μέσα σε ένα πλαίσιο της τάξης των 500 ms, τότε οι συσκευές δεν μπορούν να μεταχειριστούν περισσότερους από 15 παίκτες ταυτόχρονα).

Από την άλλη, σε παιχνίδια στα οποία η επικοινωνία μπορεί να είναι μονής κατεύθυνσης (π.χ. το παιχνίδι του Καζανόβα, ή η εγκατάσταση του Προφήτη) οι

συσκευές αυτές, με όλους τους περιορισμούς ενέργειας και επεξεργαστικής ισχύος στους οποίους εμπίπτουν, συμπεριφέρονται κατά πολύ καλύτερα. Εν τέλει, ανάλογα με τις απαιτήσεις των εφαρμογών οι οποίες υλοποιούνται σε εκάστοτε περίπτωση, οι σχεδιαστές εφαρμογών πρέπει να είναι ενήμεροι τέτοιων πρακτικών περιορισμών και να έχουν την δυνατότητα και την γνώση να επιλέξουν μεταξύ των διαφορετικών επιλογών.

Φαινόμενα ίδιου τύπου παρουσιάζονται όταν πολλοί παίκτες συγκεντρώνονται γύρω από έναν Σταθμό Παιχνιδιού. Στην εικόνα 4, είναι εμφανές ότι οι διεπαφές 802.15.4 των Σταθμών τείνουν να γίνουν αιτίες καθυστέρησης όταν πολλοί παίκτες προσπαθούν να μεταδώσουν δεδομένα ταυτόχρονα. Όταν περισσότεροι από 6 παίκτες συγκεντρώνονται σε μία περιοχή, προβλήματα συμφόρησης παρουσιάζονται. Έτσι, στις περιπτώσεις ψυχαγωγικών ή εκπαιδευτικών εγκαταστάσεων οι οποίες απαιτούν απόκριση σε πραγματικό χρόνο στις ενέργειες των παικτών, είναι απαραίτητη η ύπαρξη μίας κατάλληλης υποδομής κορμού ώστε να υποστηριχθούν πλήρως αυτά τα χαρακτηριστικά. Όταν τα δεδομένα αγγίζουν τους σταθμούς, τότε η περαιτέρω καθυστέρηση την οποία δέχονται σε όλα τα υπόλοιπα επίπεδα, είναι ελάχιστη σε σχέση με την καθυστέρηση μέχρι το συγκεκριμένο επίπεδο. Από την άλλη, εφαρμογές σαν την Καυτή Πατάτα, όπου η εφαρμογή τρέχει σε αποσυνδεδεμένο mode, και όλα τα δεδομένα ανεβαίνουν στα ανώτερα επίπεδα στο τέλος του παιχνιδιού, το ζήτημα αυτό δεν υπάρχει. Όλα έχουν να κάνουν άμεσα με τα ίδια τα παιχνίδια τα οποία υλοποιούνται κάθε φορά.

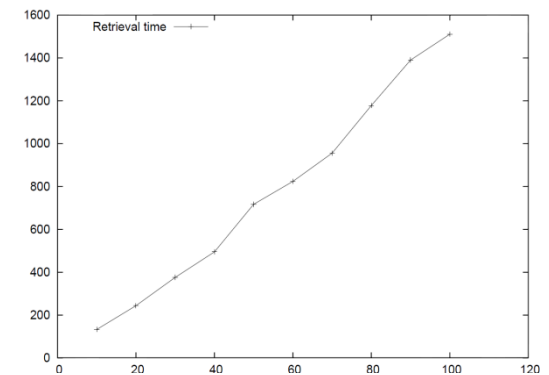
167

Αξιολόγηση

Θέματα Υλικού

Είναι σίγουρο πως τα SPOT είναι μία πολλά υποσχόμενη πλατφόρμα για πειραματισμό και προτυποποίηση, με την δυνατότητα της χρήσης της Java ως γλώσσας προγραμματισμού τους, ένα πολύ θετικό χαρακτηριστικό. Παρ' όλ' αυτά, υπάρχει ακόμα ένας αριθμός θεμάτων τα οποία περιορίζουν την χρήση τους σε μια πλήρη εφαρμογή, τόσο από άποψη υλικού όσο και από άποψη λογισμικού. Καταρχήν, υπάρχουν κάποια θέματα συμβατότητας όσον αφορά την χρήση της εικονικής μηχανής Squawk της Java στα SPOT καθώς και θέματα διατήρησης στο υπόλοιπο σύστημα λόγω της χρήσης διαφορετικών εκδόσεων Java. Επιπλέον, υπάρχουν κρυμμένες παγίδες όπως στην χρήση multithor broadcast mode, όπου δεν είναι καθαρό στον προγραμματιστή, ότι η εκπομπή μπορεί να προκαλέσει συμφόρηση στο δίκτυο, και αυτό διότι από προεπιλογή η συσκευή έχει οριστεί να εκπέμπει σε απόσταση δύο κόμβων (παρότι αυτή η παράμετρος μπορεί να οριστεί ακόμα και σε απόσταση ενός κόμβου). Υπάρχει επίσης το θέμα της αργής μνήμης flash η οποία είναι εγκατεστημένη στις συσκευές. Τα πειράματα στην διπλανή εικόνα δείχνουν ότι περίπου το 1/3 του συνολικού χρόνου για την αποστολή ενός καταγεγραμμένου πακέτου από τις συσκευές των χρηστών στα ανώτατα επίπεδα οφείλεται στον χρόνο που απαιτείται για την ανάκτησή του από την μονάδα μνήμης flash της συσκευής. Επίσης, υπάρχει και το θέμα της μη επαρκούς επεξεργαστικής ισχύος έτσι ώστε να γίνει αναγνώριση πολύπλοκων χειρονομιών με τις συσκευές σε πραγματικό χρόνο. Μία ακόμα

φάση εκπαίδευσης πρέπει να χρησιμοποιηθεί, ένα μάλλον ανεπιθύμητο χαρακτηριστικό.



Χρόνος ανάκτησης από την μνήμη flash της συσκευής. Συγκεκριμένα στον οριζόντιο άξονα φαίνεται ο εκάστοτε αριθμός των γεγονότων, ενώ στον κάθετο ο αντίστοιχος χρόνος ανάκτησης (σε ms).

Αξιολόγηση

Ερωτηματολόγια

Κατά το πέρας των πειραμάτων, δόθηκαν ερωτηματολόγια προς συμπλήρωση σε όλους τους συμμετέχοντες (προφανώς η ομάδα όλων των ανθρώπων της ανάπτυξης δεν συμμετείχε στα παιχνίδια). Τα ερωτήματα που δόθηκαν στους παίκτες, κυμαίνονταν από απόψεις πάνω στην συνολική ικανοποίηση/διασκέδαση την οποία είχαν από την συμμετοχή τους στις συνεδρίες, μέχρι και ερωτήσεις σχετικά με το πώς θα αντιδρούσαν σε βελτιώσεις όσον αφορά το υλικό και την διεπαφή χρήστη της πλατφόρμας.

Από τα αποτελέσματα, εμφανή συμπεράσματα είναι τα παρακάτω:

- Όλοι οι συμμετέχοντες ήταν θετικοί για τον παράγοντα διασκέδασης καθώς και την ευκολία χρήσης της συσκευής. 24% ήταν πολύ θετικοί και 71% θετικοί για την εμπειρία των εφαρμογών.
- Η πλειονότητα των παικτών ήταν θετική όσον αφορά την φυσική δραστηριότητα την οποία εμπειριέχε το παιχνίδι. (82% θετικοί ή πολύ θετικοί και 18% ουδέτεροι), με την συντριπτική πλειοψηφία (16 στους 17) επίσης να αναφέρουν πως η χρήση χειρονομιών και κινήσεων είναι πολύ πιο ευχάριστη και διασκεδαστική από την χρήση κουμπιών.
- Οι απαντήσεις όσον αφορά την τωρινή υλοποίηση αναγνώρισης χειρονομιών ήταν

αρκετά πιο ουδέτερη, με το 24% των συμμετεχόντων να είναι θετικοί, το 42% να είναι ουδέτεροι και το 35% να είναι αρνητικοί ή πολύ αρνητικοί, μάλλον λόγω της επιλογής ανεξάρτητης από τον χρήστη στρατηγικής αναγνώρισης κινήσεων.

- Τα παιχνίδια ήταν πιο διασκεδαστικά όταν περισσότεροι παίκτες ήταν παρόντες, με το 76% των συμμετεχόντων να είναι πολύ θετικοί όσον αφορά τις μεγάλες ομάδες παικτών.
- Οι παίκτες ανέφεραν ότι διάφορες προσθήκες στην χρηστική διεπαφή, όπως οθόνη ή δόνηση θα ήταν πολύ ευπρόσδεκτη, ακόμα και όταν δεν προέρχονταν από επιστήμες πληροφορικής και υπολογιστών. Η αντίδραση στην προσθήκη ηχητικών εφέ ήταν πολύ πιο ουδέτερη (36% ήταν ουδέτεροι ή αρνητικοί)
- Η συνολική λειτουργία του συστήματος ήταν πολύ ικανοποιητική κατά την γνώμη των συμμετεχόντων, τόσο στους εσωτερικούς όσο και στους εξωτερικούς χώρους, με την αντίδραση στους εσωτερικούς χώρους να είναι ελάχιστα πιο θετική.

169

Αξιολόγηση

Πειράματα

170

Σύμφωνα με την γενική αρχιτεκτονική της πλατφόρμας η συμπεριφορά του συστήματος αξιολογείται με ένα σύνολο σεναρίων, τα οποία έχουν ενισχυθεί με την χρήση μονάδων λογισμικού καταγραφής των γεγονότων. Έτσι, μόλις δημιουργείται ένα γεγονός, διατηρούνται συγκεκριμένα χαρακτηριστικά στην μνήμη της συσκευής, ενώ δεν γίνεται εγγραφή στην μόνιμη μνήμη, καθώς επιβαρύνει χρονικά τις μετρήσεις. Οι μετρούμενες ποσότητες έχουν κρατηθεί στο ελάχιστο, και μόλις τελειώνει κάθε στιγμιότυπο παιχνιδιού, οι μετρήσεις επιστρέφονται ασύρματα σε έναν σταθμό παιχνιδιού. Για να εξαχθούν κάποια συμπεράσματα, έγινε μέτρηση του χρόνου διάδοσης των μηνυμάτων σε κάθε επίπεδο του συστήματος, των γειτόνων κάθε συσκευής ανά μονάδα χρόνου, τον αριθμό των χειρονομιών οι οποίες αναγνωρίστηκαν επιτυχώς, τον αριθμό των ενεργειών που απέτυχαν, την χρήση του επεξεργαστή καθώς και την χρήση της μνήμης κάθε χρονική στιγμή. Πρέπει να σημειωθεί πως δεν παρουσιάζεται καμία μέτρηση που να αφορά την κατανάλωση ενέργειας, καθώς η διάρκεια των παιχνιδιών είναι πολύ μικρή για κάτι τέτοιο. Οι ενδείξεις παρ' όλ' αυτά καταδεικνύουν πως η κατανάλωση ενέργειας δεν είναι μεγάλη.

Αρχικά αξιολογείται το σύνολο του χρόνου που απαιτείται για ένα γεγονός να φτάσει μέχρι το επίπεδο του Κόσμου. Όλες οι συσκευές σε αυτή την περίπτωση βρίσκονται συνδεδεμένες με την υποδομή. Έτσι, για το επίπεδο των φρουρών έχουμε Τφ, τον χρόνο που απαιτείται για τον έλεγχο της σύνδεσης με τους σταθ-

μούς, το άνοιγμα μίας σύνδεσης ραδιοτηλεγραφήματος (radiogram) με τον σταθμό και την αποστολή της ραδιοροής (datagram) μέσω 802.15.4. Για το επίπεδο Σταθμών, έχουμε Τσ τον χρόνο που απαιτείται για την ώθηση της ληφθείσας ραδιοροής (datagram) στην ουρά, για την παραλαβή από την ουρά, την μετατροπή σε γεγονός και την μεταφορά του γεγονότος στην Μηχανή μέσω 802.11b. Για το επίπεδο Μηχανής έχουμε Τμ τον χρόνο που απαιτείται για την ώθηση του γεγονότος στην ουρά, την παραλαβή του από την ουρά και την αποστολή του στην βάση μέσω μίας 100Mbps σύνδεσης Ethernet. Με την αποστολή 100 γεγονότων, ο συνολικός χρόνος κατά μέσο όρο ήταν $T_{\phi} + T_{\sigma} + T_{\mu} = 38 + 2.5 + 5.5 \text{ ms} = 46 \text{ ms} = T_{\text{ολ}}$. Το 83% του χρόνου σχετίζεται με το επίπεδο φρουρών, και με περεταίρω ανάλυση παρατηρήθηκε πως οφείλεται ως επί το πλείστον στην εικονική μηχανή. Συγκεκριμένα, το SPOT χρειάζεται 13ms για να ανοίξει την σύνδεση, 6.5ms για να την κλείσει, και 12 ms για να μετατρέψει τα δεδομένα σε πακέτο του 802.15.4

Σε ένα δεύτερο σενάριο θεωρείται πως το παιχνίδι εξελίσσεται έξω από την κάλυψη της υποδομής κορμού, και οι χρήστες συγκεντρώνονται γύρω από έναν σταθμό για να μεταφέρουν τα γεγονότα στα ανώτερα επίπεδα. Σε αυτή την περίπτωση, έχει ενδιαφέρον η μέτρηση του απαιτούμενου χρόνου για να φτάσουν τα δεδομένα στην βάση, σε σχέση με τον αριθμό των φρουρών οι οποίοι συμμετέχουν στην διαδικασία. Κρατώντας λοιπόν σταθερό τον αριθμό των γεγονότων ανά παίκτη, αυξάνεται ο αριθμός των συσκευών οι ο-

Αξιολόγηση

Πειράματα

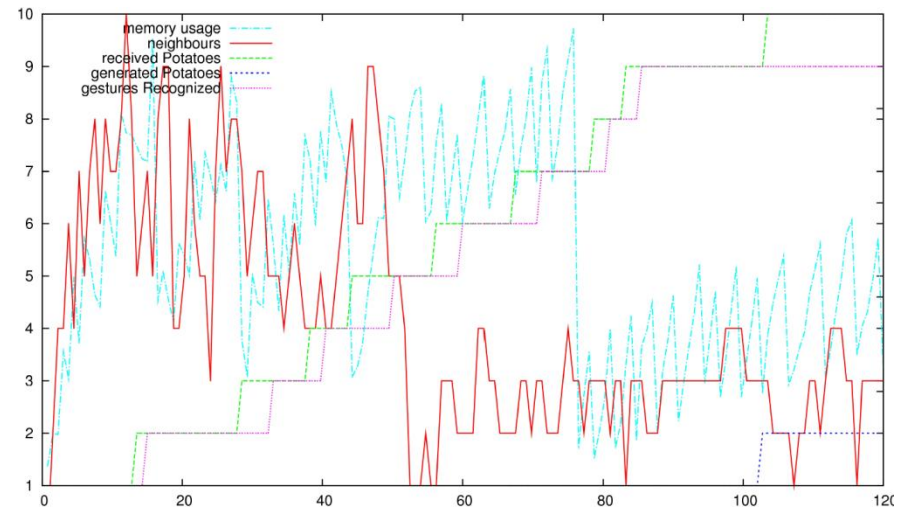
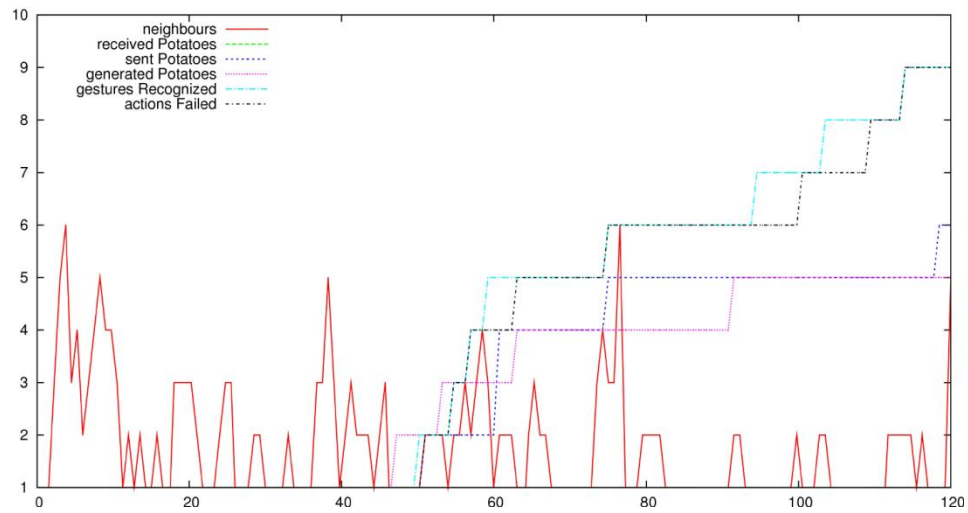
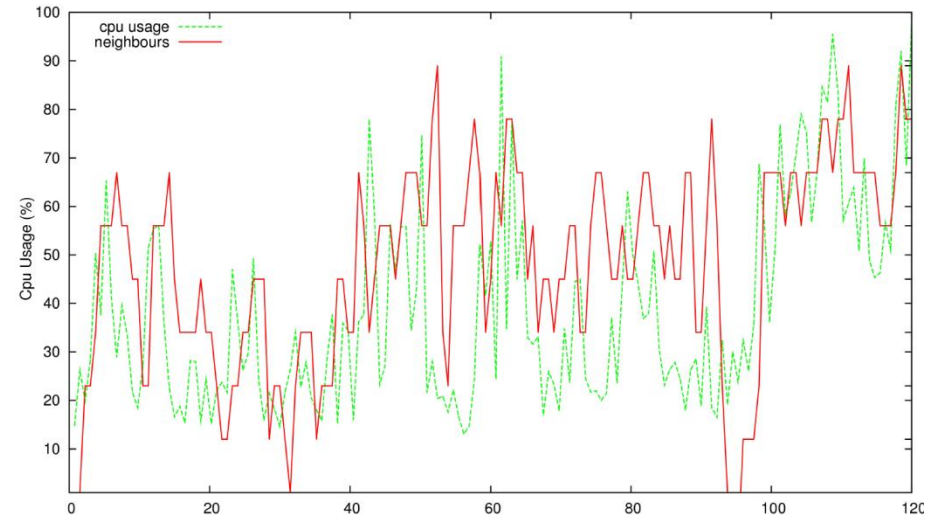
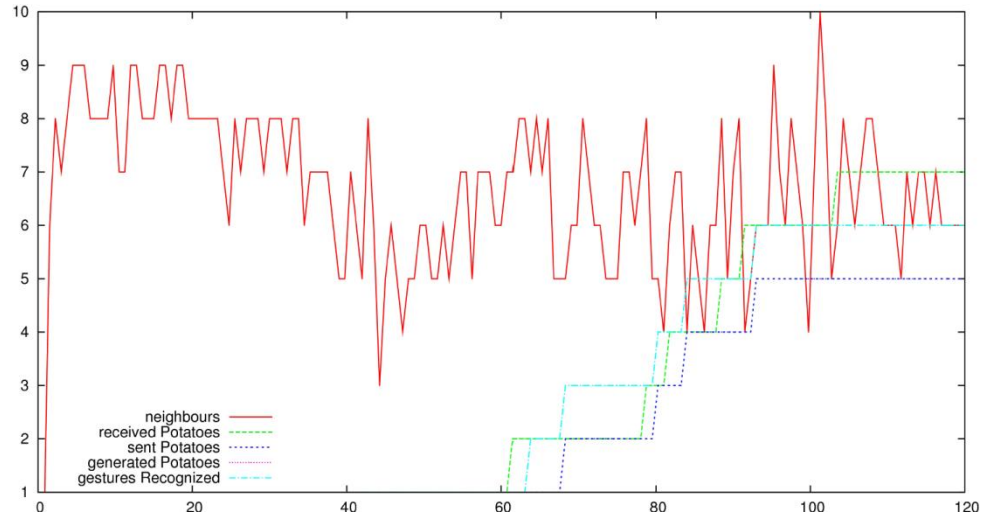
ποίες στέλνουν γεγονότα. Για κάθε επίπεδο i , το $total_i$ είναι το $Ti1-Ti2$ όπου $Ti1$ είναι η στιγμή που έφθασε το πρώτο γεγονός σε αυτό το επίπεδο, και $Ti2$ είναι η στιγμή που προωθήθηκε το τελευταίο γεγονός στο ανώτερο επίπεδο $i+1$. Ο χρόνος επεξεργασίας $proci$ είναι το σύνολο των Ti για το σύνολο των γεγονότων g . Ο ρυθμός παραλαβής π είναι $\pi = Ti/g$ ενώ ο ρυθμός επεξεργασίας ϵ είναι $\epsilon = proci/g$.

Έτσι, με ένα φρουρό είναι $\pi = 37.6 \text{ ms/event}$ και $\epsilon = 2.4 \text{ ms/event}$ για το επίπεδο Σταθμών και $\epsilon = 5.3 \text{ ms/event}$ για το επίπεδο Μηχανής. Έτσι, τα επίπεδα αυτά μένουν αδρανή για 35.2 και 32.3 ms αντίστοιχα. Ο τρόπος που έχουν υλοποιηθεί τα επίπεδα επιτρέπει ασύγχρονη μετάδοση δεδομένων, με αποτέλεσμα να υπάρχουν χρονικές αλληλεπικαλύψεις στην διαδικασία μετάδοσης - ο μόνος περιορισμός αυτή την στιγμή βρίσκεται στις συσκευές των χρηστών να αποστέλλουν δεδομένα με μεγαλύτερο ρυθμό. Αυξάνοντας τον αριθμό των συσκευών στην περιοχή, και επαναλαμβάνοντας την ίδια ακριβώς διαδικασία, παρατηρείται αύξηση του ρυθμού παραλαβής ο οποίος ξεπερνά τον ρυθμό επεξεργασίας των γεγονότων, ωθώντας έναν σημαντικό αριθμό γεγονότων στην ουρά. Στην εικόνα 16 φαίνεται πως όταν ο αριθμός των συσκευών αυξάνεται, ο ρυθμός παραλαβής γεγονότων μειώνεται. Παρ' όλ' αυτά, η συμπεριφορά αυτή αλλάζει κατά την εισαγωγή της έκτης συσκευής στο πείραμα. Φαίνεται πως η αυξημένη πυκνότητα του δικτύου (6 φρουροί και ένας σταθμός) καθώς και το σύνολο των δεδομένων (6x100) απαιτούν ένα σημαντικό ποσοστό από τους ήδη περιορισμένους πόρους των συσκευών. Ακόμα, συγκρούσεις στο 802.15.4 οδηγούν σε αναμε-

ταδόσεις καταλήγοντας σε ακόμα μεγαλύτερη κίνηση. Το δίκτυο καθίσταται ασταθές με την αύξηση των φρουρών σε περισσότερους από έξι

Οι εικόνες 3 και 4 δείχνουν την εκπληκτική δυναμικότητα του δικτύου σε ένα παιχνίδι που κράτησε 2 λεπτά. Σε ορισμένα σημεία, η γειτονιά του πρώτου χρήστη άλλαξε από εννέα γείτονες σε τρεις σε λιγότερο από τρία δευτερόλεπτα. Το πείραμα καταδεικνύει πως η επιλογή του IEEE 802.15.4 είναι πολύ καλή, καθώς είναι ικανό να αναγνωρίσει την εγγύτητα των συσκευών πολύ γρήγορα υποστηρίζοντας αποτελεσματικά τους εκάστοτε κανόνες. Στις δύο εικόνες εμφανίζονται δύο διαφορετικές στρατηγικές παικτών, καθώς και τα αποτελέσματά τους, σε ένα στιγμιότυπο παιχνιδιού καυτής πατάτας. Ο πρώτος παίκτης, βρίσκεται μέσα στην περιοχή των γειτόνων του, για να μπορέσει να διώξει οποιαδήποτε πατάτα από την συσκευή του. Αντιδρά γρήγορα, οι ενέργειές του αναγνωρίζονται, όμως στο τέλος, μετά από τέσσερις εναλλαγές πατάτας, αυτή εκρήγνυται στην συσκευή του. Ο δεύτερος επιλέγει να μείνει εκτός της περιφέρειας των ανιχνεύσιμων του, αλλά σύμφωνα με τους κανόνες του παιχνιδιού, δεν αρνεί να δημιουργήσει μία πατάτα, την οποία δεν καταφέρνει να περάσει γρήγορα σε κάποιον γείτονα, ενώ οι κινήσεις του αναγνωρίζονται με επιτυχία. Ακόμη και με ένα τόσο απλό παιχνίδι, όσον αφορά την σύλληψη των κανόνων του, είναι εμφανής η φυσική δραστηριότητα των παικτών, οι δυνατές στρατηγικές, η συμμετοχή πολλών παικτών και κινήσεων. Από την άλλη, σε τεχνικό επίπεδο, τα αποτελέσματα δείχνουν με λεπτομέρεια την λειτουργία των πρωτοκόλλων και την επιτυχία ή αποτυχία αυτών.

171



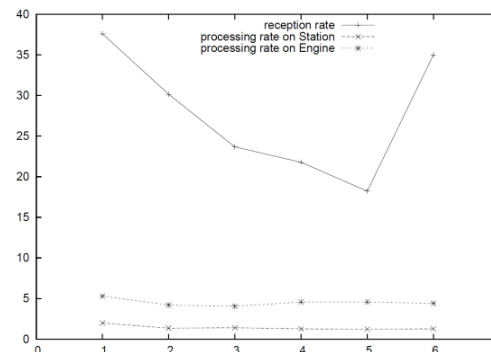
Αξιολόγηση

Πειράματα

Η εικόνα της διπλανής σελίδας (πάνω δεξιά) δείχνει την χρήση του επεξεργαστή και την δυναμική του δικτύου για μία περίοδο 2 λεπτών. Παρουσιάζει την υπολογιστική πολυπλοκότητα του πρωτοκόλλου ανακάλυψης των γειτόνων της συσκευής. Προφανώς, η αναγνώριση των χειρονομιών, η λειτουργία των LED, ο έλεγχος των κουμπιών καθώς και η συλλογή απορριμμάτων απαιτούν επεξεργαστική ισχύ. Παρ' όλ' αυτά η συντριπτική πλειοψηφία των πόρων του επεξεργαστή αξιοποιούνται από το πρωτόκολλο ανακάλυψης γειτόνων. Το πείραμα παρουσιάζει ότι ο επεξεργαστής ARM9 των συσκευών αυτών, μπορεί να χρησιμοποιηθεί για ταυτόχρονο παίξιμο -δηλαδή στην ίδια περιοχή- τουλάχιστον δεκατεσσάρων παικτών χωρίς να εμποδίζεται η εμπειρία των παικτών.

Τέλος, όσον αφορά την μνήμη, η τελευταία εικόνα της διπλανής σελίδας παρουσιάζει πειραματικές μετρήσεις που αφορούν την χρήση της στο παιχνίδι. Διαφαίνεται πως τα 512 Kb της μνήμης μπορούν με επιτυχία να υποστηρίξουν το μέγεθος της εικονικής μηχανής Squawk, τον κώδικα του παιχνιδιού καθώς και τις οντότητες οι οποίες δημιουργούνται κατά την διάρκεια του παιχνιδιού. Είναι στην φύση της γλώσσας να παράγει έναν μεγάλο αριθμό στιγμιότυπων. Το πρωτόκολλο ανακάλυψης γειτόνων παράγει ένα νέο αντικείμενο κάθε περίπου 500 ms, το πρωτόκολλο ενεργειών παράγει τρία νέα αντικείμενα για κάθε ενέργεια κλπ. Δύο φορές μέσα στην διάρκεια του παιχνιδιού η χρήση της μνήμης άγγιξε το 100% και με αποτέλεσμα την κλήση της συλλογή απορριμμάτων. Παρ' όλ' αυτά, καθ' όλη την διάρκεια του παιχνιδιού καλούνταν η συλλογή απορριμμάτων, για να ελευθερώσει μνήμη -

κάτι το οποίο βέβαια χρησιμοποιούσε επεξεργαστικούς πόρους. Από αυτό το χαρακτηριστικό και μόνο, είναι εμφανές πως η υποστήριξη τέτοιων παιχνιδιών από το υλικό, και εν τέλει η συνολική ανάπτυξη αυτού του συστήματος θα ήταν πολύ δύσκολη, αν όχι αδύνατη, αν είχε γίνει επιλογή κάποιας άλλης πλατφόρμας υλικού.



173

Μελλοντική Εργασία

Μελλοντική δουλειά θα μπορούσε να αποτελέσει η μεταφορά του συστήματος σε πιο δυνατές πλατφόρμες, με προσανατολισμό στις συσκευές κινητών τηλεφώνων όπως για παράδειγμα το OpenMoko, το iPhone ή συσκευές με Android. Αυτό όχι μόνο έχει το θετικό της μεγάλης προϋπάρχουσας βάσης χρηστών, αλλά προσφέρει μεγαλύτερες δυνατότητες αλληλεπίδρασης του χρήστη με την εφαρμογή. Οι συσκευές αυτές ενσωματώνουν επιταχυνσιόμετρα και άλλους αισθητήρες ενώ επίσης παρέχουν περισσότερα χαρακτηριστικά όπως voice-over-IP για ενδοεπικοινωνία μέσα στο παιχνίδι. Επίσης, το γεγονός ότι στις περισσότερες από αυτές τις συσκευές η J2ME υπάρχει, σημαίνει πως η μεταφορά θα διευκολύνεται σημαντικά. Στα πλαίσια μάλιστα της συγκεκριμένης εργασίας, έγινε μερική ανάπτυξη στο SDK του Android και του Symbian, του λειτουργικού συστήματος της Nokia. Πρέπει να σημειωθεί, πως από προηγούμενες εργασίες [8] είναι γνωστό το γεγονός ότι για να υπάρχει πλήρης πρόσβαση στους αισθητήρες της συσκευής είναι απαραίτητη η συγγραφή κάποιου μέρους του κώδικα στην εγγενή γλώσσα του λειτουργικού συστήματος. Έτσι, στην περίπτωση του Nokia N95, για να είναι διαθέσιμες οι μετρήσεις των αισθητήρων στις συσκευές, είναι

απαραίτητη η συγγραφή μέρους του κώδικα σε Symbian C++. Το καλό στην περίπτωση του Android είναι πως οι εφαρμογές του λειτουργικού γράφονται σε J2SE (Java Standard Edition) και όχι J2ME (Java Micro Edition) με την πληθώρα πλεονεκτημάτων που αυτό επιφέρει. Είναι σημαντικό το γεγονός ότι η πρόσβαση στις τιμές των αισθητήρων γίνεται άμεσα από την Java, ενώ η μεταφορά θα είναι εξαιρετικά απλή. Οι βιβλιοθήκες στο Android είναι γραμμένες σε C και C++ για λόγους ταχύτητας, ενώ το λειτουργικό βασίζεται σε έναν βαριά τροποποιημένο πυρήνα Linux. Τέλος, όσον αφορά το iPhone, η μεταφορά πρέπει να γίνει σε objective C, μία γλώσσα η οποία συνδυάζει τα περάσματα μηνυμάτων της Smalltalk με την C, δημιουργώντας μια αντικειμενοστραφή έκδοση. Η μεταφορά σε iPhone αποτελεί μάλλον το πιο δύσκολο από τα τρία εγχειρήματα, με την βάση χρηστών όμως να είναι η μεγαλύτερη με συντριπτική διαφορά (με τους χρήστες που χρησιμοποιούν Android να μειώνει το κενό πολύ γρήγορα). Για παράδειγμα, οι υπάρχουσες εφαρμογές αυτή την στιγμή για το Android είναι γύρω στις 10.000 ενώ το Android ξεπερνά κατά πολύ τις 100.000 εφαρμογές.

174



Κατακλείδα

Στην εργασία αυτή παρουσιάστηκε η χρήση αδόμητων δικτύων αισθητήρων στις περιοχές του διάχυτου υπολογισμού και των διαδραστικών εγκαταστάσεων. Παρουσιάστηκε μία πλατφόρμα για την δημιουργία, την εγκατάσταση και την διαχείριση παιχνιδιών πολλών παικτών καθώς και ψυχαγωγικών, εκπαιδευτικών ή καλλιτεχνικών εγκαταστάσεων. Έγινε παρουσίαση τόσο της αρχιτεκτονικής όσο και της υλοποίησης, ενώ δόθηκε ιδιαίτερη βαρύτητα στα ανώτερα επίπεδα της αρχιτεκτονικής. Επίσης, έγινε παρουσίαση επιτυχιών υλοποιήσεων, όπως ένας αριθμός παιχνιδιών και εγκαταστάσεων τα οποία υλοποιήθηκαν, χρησιμοποιήθηκαν και εν τέλει αξιολογήθηκαν. Μέσα από αυτή την εργασία καθίσταται εμφανές πως οι διαθέσιμες σήμερα ασύρματες συσκευές με αισθητήρες μπορούν να υποστηρίξουν παιχνίδια μεγάλης κλίμακας καθώς και έχουν να προσφέρουν πολλά σε τέτοιου είδους εγκαταστάσεις. Προβλήματα μπορεί να προκύψουν όσον αφορά τους επεξεργαστικούς και επικοινωνιακούς πόρους στις περιπτώσεις που ο αριθμός των παικτών σε μία περιοχή είναι εξαιρετικά μεγάλος. Η εμπειρία από αυτήν την εργασία είναι πως η έρευνα που σχετίζεται με τα ασύρματα δίκτυα αισθητήρων μπορεί να εφαρμοστεί σε αυτές τις περιοχές, ώστε να δημιουργήσει νέες εμπειρίες, ενδιαφέρουσες προτάσεις καθώς ακόμη και πρωτότυπα προϊόντα.

Έτσι, έγινε παρουσίαση όλων των μερών της πλατφόρμας με αρκετή λεπτομέρεια, ενώ παρουσιάστηκαν επίσης οι τεχνολογίες οι οποίες υλοποιούν την χρηστική διεπαφή του ανώτατου επιπέδου. Από την μία είναι η χρήση της Processing, μίας νέας γλώσσας προγραμματισμού για την επεξεργασία και οπτικοποίηση δεδομένων, και από την άλλη η χρήση του

Liferaγ, ενός από τα πιο σύνθετα εργαλεία δημιουργίας διαδικτυακών πυλών με στοιχεία διαχείρισης περιεχομένου, συνεργασίας και διεύθυνσης των επί μέρους λειτουργιών της πλατφόρμας.

Με τις δυνατότητες αυτού του ανώτατου αυτού επιπέδου, είναι εμφανείς οι ανεξάντλητες δυνατότητες του συγκεκριμένου συστήματος. Όπως έγινε κατανοητό, είναι δυνατή η χρήση του σε εκπαιδευτικές εγκαταστάσεις, με μία νέα, πιο ευχάριστη και ταυτόχρονα πιο αποδοτική προσέγγιση. Από την άλλη είναι δυνατή η χρήση της για ψυχαγωγικά παιχνίδια και εγκαταστάσεις, κάτι το οποίο αποτελεί μία ανερχόμενη κουλούρα στην εποχή μας. Εκτός του ότι θεωρείται ένας τομέας όπου δαπανούνται εξαιρετικά υψηλά ποσά, με πολλαπλάσια κέρδη, οδηγεί ως ένα βαθμό τις προόδους της τεχνολογίας, συγκεντρώνοντας δεδομένα από πολλές επιστήμες και παρουσιάζοντας κάτι ολοένα και πιο κοντά στις ανάγκες του χρήστη (βλ. Καινούριες χρηστικές διεπαφές, τρόποι επικοινωνίας των συσκευών ψυχαγωγίας, αντίληψης των εμπειριών από τους χρήστες). Τέλος, στον τομέα των καλλιτεχνικών εγκαταστάσεων, οι δυνατότητες είναι άπειρες, καθώς είναι μάλλον από τις πρώτες φορές που παρουσιάζεται μία πλατφόρμα με τέτοιες δυνατότητες προς αυτή την κατεύθυνση. Οι πιθανοί συνδυασμοί εντυπωσιάζουν, με το αποτέλεσμα να αφήνει άφωνους ανθρώπους τόσο από μία όχθη τόσο από την άλλη όσο και ανθρώπους οι οποίοι δεν σχετίζονται με αυτά κανέναν από τους δύο τομείς. Το πάντρεμα τέχνης και τεχνολογίας συνεχίζει να προάγει την πρόοδο, να εξυπηρετεί τον άνθρωπο και να αλληλοβοηθείται εξιστορώντας τα επιτεύγματα του ανθρώπινου είδους με μοναδικό τρόπο.

175

Κατακλείδα

Κατά καιρούς, η ιδέα της κατασκευής μίας ειδικής και εύκολα παραμετροποιήσιμης συσκευής η οποία θα συνοδεύει το λογισμικό της πλατφόρμας έχει αποτελέσει θέμα συζήτησης. Έτσι οι χρήστες της πλατφόρμας, εφοδιασμένοι με αυτή την συσκευή θα μπορούν να συμμετέχουν σε ακόμη περισσότερα περιβάλλοντα και υπό ακόμη μεγαλύτερες ποικιλίες συνθηκών. Ένα άλλο θέμα το οποίο χρίζει έρευνας είναι η το ζήτημα της ενέργειας των κινητών συσκευών και αποδοτικής αξιοποίησής της - να σημειωθεί πως τα SUN Sports δεν χαρακτηρίζονται ως φειδωλά στην κατανάλωση ενέργειας.

Τελειώνοντας, πρέπει να σημειωθεί το αντίκτυπο της συγκεκριμένης πλατφόρμας σε διεθνές επίπεδο. Η αρχιτεκτονική του συστήματος παρουσιάστηκε στο [76] το οποίο και έλαβε χώρα στο Ίδρυμα Μείζονος Ελληνισμού τον Οκτώβρη του 2009. Πολύ θετικά σχόλια λάβαμε για το [77] και το [78] με το πρώτο γεγονός να λαμβάνει χώρα στην Βόρεια Καρολίνα των Ηνωμένων Πολιτειών, ενώ το δεύτερο στην Πανεπιστημιούπολη Αθηνών. Τα παιχνίδια παρουσιάστηκαν στο [79] το οποίο έλαβε χώρα στην Ατλάντα των Ηνωμένων Πολιτειών και απέσπασε το βραβείο της καλύτερης επίδειξης συστήματος του συνεδρίου (Best Demo Award). Τέλος, όσον αφορά τις ψυχαγωγικές εγκαταστάσεις, αυτές παρουσιάστηκαν στο [75], στον πολυχώρο του IME, και απέσπασαν το χρυσό βραβείο δημιουργικής επίδειξης (Golden Showcase Award).

Great things are not done by impulse, but by a series of small things brought together.

Vincent van Gogh (Painter, 1853 - 1890)

Παράρτημα

Βιβλιογραφία

180

[1] Ira Greenberg: Processing: Creative Coding and Computational Art , 2008.

[2] Έκθεση Ηλεκτρονικής Ψυχαγωγίας (Electronic Entertainment Expo), <http://www.e3expo.com>.

[3] The Gaming Industry Trends of E3 2009, <http://www.sidequesting.com/2009/06/the-gaming-industry-trends-of-e3-2009>.

[4] E3 Trends: Social Networking, Controller-Free Devices Lead Buzz, http://adage.com/digitalnext/article?article_id=137166

[5] Adam Greenfield - Everyware: the dawning age of Ubiquitous Computing, 2006.

[6]Περισσότερες πληροφορίες μπορεί κάποιος να βρεί στο http://en.wikipedia.org/wiki/Ubiquitous_computing

[7] E.Miluzzo, N.D. Lane, K.Fodor, R.Peterson, H.Lu, M.Musolesi, S.B. Eisenman, X.Zheng, A.T. Campbell. Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application στο Proceedings of the 6th ACM conference on Embedded network sensor systems, σελίδες 337-350, Raleigh, NC, USA.

[8] S.Gaonkar, J.Li, R.R.Choudhury, L.Cox, A.Schmidt. Micro-Blog: Sharing and Querying Content Through Mobile Phones and Social Participation στο Proceeding of the 6th international conference on Mobile systems, applications, and services, σελίδες 174-186, Breckenridge, CO, USA

[9] Zum Kuckuck - Telekom Realtime Information Graphics,<http://projects.zumkuckuck.com/realtime/>

[10] Burak Arıkan - MYPOCKET, <http://turbulence.org/Works/mypocket/>

[11] Η ιστοσελίδα που αφορά τα sun spots βρίσκεται στην διεύθυνση <http://www.sunspotworld.com/>.

[12] Processing is an open source "programming language and integrated development environment στην ιστοσελίδα <http://processing.org/>

[13] Liferay, an open source content management system, στην ιστοσελίδα <http://www.liferay.com>

[14] Το MIT Media Lab βρίσκεται στην ιστοσελίδα <http://www.media.mit.edu/>

[15] Computer Science Logo Style, Brian Harvey, MIT Press (3 τόμοι) ISBN 0-262-58148-5, ISBN 0-262-58149-3, ISBN 0-262-58150-7.

[16] P. Boytchev (2009): <http://elica.net/download/papers/LogoTreeProject.pdf>

[17] Maeda, John (October 1, 2001). Design by Numbers. MIT Press. pp. 256.

[18] James Gosling, Bill Joy, Guy Steele, and Gilad Bracha, The Java language specification, third edition. Addison-Wesley, 2005. ISBN 0-321-24678-0

[19] ActionScript, a scripting language based on ECMAScript <http://en.wikipedia.org/wiki/ActionScript>

Βιβλιογραφία

[20] Περισσότερες πληροφορίες για την γλώσσα Lingo, [http://en.wikipedia.org/wiki/Lingo_\(programming_language\)](http://en.wikipedia.org/wiki/Lingo_(programming_language))

[21] "What is Python Good For?". General Python FAQ. Python Foundation, <http://www.python.org/doc/faq/general/#what-is-python-good-for>

[22] "What is Python? Executive Summary". Python documentation. Python Foundation, <http://www.python.org/doc/essays/blurb/>

[23] Python Software Foundation, <http://www.python.org/psf/>

[24] <http://www.processing.org/reference/libs/>

[25] <http://java.sun.com/applets/>

[26] http://en.wikipedia.org/wiki/Java_applet

[27] Java Security FAQ, <http://www.cs.princeton.edu/sip/faq/java-faq.php3>

[28] RFC 4329 (limit compatible with ECMAScript), <http://tools.ietf.org/html/rfc4329>

[29] SGI - OpenGL Overview, <http://www.sgi.com/products/software/opengl/overview.html>

[30] JSR 286: Portlet Specification 2.0, <http://www.jcp.org/en/jsr/detail?id=286>

[31] Hepper, Stefan (18 March 2008). "What's new in the Java Portlet Specification V2.0 (JSR 286)?" IBM.

[32] The OpenGL Graphics System Specification. Version 2.1, <http://www.opengl.org/registry/doc/glspec21.20061201.pdf>

[33] Richard S. Wright, Benjamin Lipchak, Nicholas Haemel: OpenGL SuperBible: Comprehensive Tutorial and Reference, 4th Edition, Addison-Wesley, June 18, 2007, ISBN 0-321-49882-8

[34] Knorr, Eric (2004-01-09). "The new enterprise portal". InfoWorld.

[35] MSNBC, ένα Αμερικάνικο καλωδιακό κανάλι ειδήσεων, μεταξύ Microsoft και NBC, <http://www.msnbc.msn.com/id/3096434/>

[36] <http://m.www.yahoo.com/>

[37] <http://www.aol.com/>

[38] iGoogle, a service of Google, is a customizable AJAX-based startpage or personal web portal <http://www.google.com/ig>

[39] Boye, Janus (2005-01-18). "Portal Software: Passing Fad or Real Value?". CMS Watch.

[40] JSR 168: Portlet Specification, <http://www.jcp.org/en/jsr/detail?id=168>

[41] Alister Lewis-Bowen et al., "Using open source software to design, develop, and deploy a collaborative Web site," IBM, 11 Ιουλίου, 2006, <http://www-128.ibm.com/developerworks/ibm/library/i-ospace1/index.html>

Βιβλιογραφία

182

[42] Στατιστικά λήψης του Drupal, <http://buytaert.net/tag/statistics>

[43] Άρθρο σχετικά με το Drupal 7 και την PHP 5.2, <http://drupal.org/gophp5>

[44] Πηγές σχετικά με τον αντικειμενοστραφή προγραμματισμό στο Drupal, στο <http://api.drupal.org/api/file/developer/topics/oop.html/6>

[45] CMSMatrix, μία ιστοσελίδα με δυναμικά συγκριτικά για συστήματα διαχείρισης περιεχομένου, στο <http://www.cmsmatrix.org/>

[46] Facebook, ιστοσελίδα κοινωνικής δικτύωσης -η μεγαλύτερη του είδους της- στο <http://www.facebook.com/>

[47] Nintendo Game Console, <http://www.nintendo.com/wii>

[48] Sony Playstation Game Console, περισσότερες πληροφορίες στο <http://us.playstation.com/>

[49] Microsoft Xbox 360 Game Console, περισσότερες πληροφορίες στο <http://www.xbox.com>

[50] Netvibes Web Portal, www.netvibes.com/

[52] Integrated Project on Pervasive Gaming website, <http://www.pervasive-gaming.org/>.

[53] S. Björk, J. Falk, R. Hansson, and P. Ljungstr, "Pirates! using the physical world as a game board," in Proceedings of Interact 2001, 2001, pp. 9–16.

[54] I. Lindt and W. Broll, "Netattack - first steps towards pervasive gaming," ERCIM News, no. 57, pp. 49–50, April 2004.

[55] K. Jegers and M. Wiberg, "Pervasive gaming in the everyday world," Pervasive Computing, IEEE, vol. 5, no. 1, pp. 78–85, Jan.-March 2006.

[56] A. D. Cheok, S. W. Fong, K. H. Goh, X. Yang, W. Liu, and F. Farzbiz, "Human pacman: a sensing-based mobile entertainment system with ubiquitous computing and tangible interaction," in NetGames '03: Proceedings of the 2nd workshop on Network and system support for games. New York, NY, USA: ACM, 2003, pp. 106–117.

[57] B. K. Walther, "Atomic actions – molecular experience: theory of pervasive gaming," Comput. Entertain., vol. 3, no. 3, pp. 4–4, 2005.

[58] S. Benford, C. Magerkurth, and P. Ljungstrand, "Bridging the physical and digital in pervasive gaming," Commun. ACM, vol. 48, no. 3, pp. 54–57, 2005.

[59] H. Korhonen, H. Saarenpää, and J. Paavilainen, "Pervasive mobile games — a new mindset for players and developers," in Proceedings of the 2nd International Conference on Fun and Games. Berlin, Heidelberg: Springer-Verlag, 2008, pp.21–32.

[60] F. Trinta, D. Pedrosa, C. Ferraz, and G. Ramalho, "Evaluating a middleware for crossmedia games," Comput. Entertain., vol. 6, no. 3, pp. 1–19, 2008.

Βιβλιογραφία

[61] J. Schroyen, K. Gabriëls, K. Luyten, D. Teunkens, K. Robert, K. Coninx, E. Flerackers, and E. Manshoven, "Training social learning skills by collaborative mobile gaming in museums," in ACE '08: Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology. New York, NY, USA: ACM, 2008, pp. 46–49.

[62] H. Tarumi, K. Yokoo, S. Nishimoto, K. Matsubara, Y. Harada, F. Kusunoki, S. Kim, and Y. Mizukubo, "Open experiments of mobile sightseeing support systems with shared virtual worlds," in ACE '06: Proc. of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology. New York, USA: ACM, 2006, p. 13.

[63] F. Barrenho, T. Romao, T. Martins, and N. Correia, "Inauthoring environment: interfaces for creating spatial stories and gaming activities," in ACE '06: Proc. of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology. New York, USA: ACM, 2006, p. 9.

[64] J. Burke, J. Friedman, E. Mendelowitz, H. Park, and M. B. Srivastava, "Embedding expression: Pervasive computing architecture for art and entertainment," Pervasive and Mobile Computing, vol. 2, no. 1, pp. 1–36, 2006.

[65] S. N. I. Mount, E. I. Gaura, and R. M. Newman, "Sensorium games: usability considerations for pervasive gaming," in SIGDOC '05: Proceedings of the 23rd annual international conference on Design of communication. New York, NY, USA: ACM, 2005, pp. 146–150.

[66] L. Mottola, A. L. Murphy, and G. P. Picco, "Pervasive games in a mote-enabled virtual world using tuple space middleware," in NetGames '06: Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games. New York, NY, USA: ACM, 2006, pp. 29–36.

[67] R. Ballagas, A. Kuntze, and S. P. Walz, "Gaming tourism: Lessons from evaluating reexplorer, a pervasive game for tourists," in 6th International Conference, Pervasive 2008 Sydney, Australia, 2008, pp. 244–262.

[68] D. Merrill, J. Kalanithi, and P. Maes, "Siftables: towards sensor network user interfaces," in TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction. New York, NY, USA: ACM, 2007, pp. 75–78.

[69] M. Klonowski, M. Handy, and F. Golatowski, "Lessons learned from developing a bluetooth multiplayer-game," in Workshop on Gaming Applications in Pervasive Computing Environments, 2004.

[70] J. Liu, Z. Wang, L. Zhong, J. Wickramasuriya, and V. Vasudevan, "uwave: Accelerometer-based personalized gesture recognition and its applications," Pervasive Computing and Communications, IEEE International Conference on, vol. 0, pp. 1–9, 2009.

[71] J. Kela, P. Korpimäki, J. Mantyjärvi, S. Kallio, G. Savino, L. Jozzo, and D. Marca, "Accelerometer-based gesture control for a design environment," Personal Ubiquitous Comput., vol. 10, no. 5, pp. 285–299, 2006.

Βιβλιογραφία

184

[72] S. Hay and R. Harle, “Bluetooth tracking without discoverability,” in 4th International Symposium of Location and Context Awareness (LoCA 2009), ser. Lecture Notes in Computer Science, vol. 5561. Springer, 2009, pp. 120–137.

[73] Alix Gateway, περισσότερες πληροφορίες στο <http://www.pcengines.ch/alix1d.htm>.

[74] Maven, a software project management and comprehension tool, περισσότερες πληροφορίες στο <http://maven.apache.org/>.

[75] Akribopoulos Orestis, Logaras Marios, Vasilakis Nikos, Kokkinos Panagiotis, Mylonas Georgios, Chatzigiannakis Ioannis and Spirakis Paul, Multiplayer Pervasive Games and Networked Interactive Installations using Ad hoc Mobile Sensor Networks, 5th International Conference on Advances in Computer Entertainment Technology (ACE09) Creative Showcases, Athens, Greece, 2009. *Received the Gold Creative Showcase Award.*

[76] Akribopoulos Orestis, Logaras Marios, Vasilakis Nikos, Kokkinos Panagiotis, Mylonas Georgios, Chatzigiannakis Ioannis and Spirakis Paul, Developing Multiplayer Pervasive Games and Networked Interactive Installations using Ad hoc Mobile Sensor Nets, 5th ACM International Conference on Advances in Computer Entertainment Technology (ACE09), Athens, Greece, 2009.

[77] Akribopoulos Orestis, Logaras Marios, Vasilakis Nikos, Kokkinos Panagiotis, Mylonas Georgios and

Chatzigiannakis Ioannis, Using Wireless Sensor Networks to Develop Multi-player Games, 6th ACM Conference on Embedded Networked Sensor Systems (SENSYS08), Raleigh, NC, 2008.

[78] Akribopoulos Orestis, Logaras Marios, Vasilakis Nikos, Kokkinos Panagiotis, Mylonas Georgios and Chatzigiannakis Ioannis, Fun in Numbers: A Platform for Developing and Playing Multiplayer Pervasive Games, 1st Student Workshop on Wireless Sensor Networks (SWWSN08), Athens, Greece, 2008.

[79] Orestis Akribopoulos, Dimitrios Bousis, Dionysios Efsthathiou, Haris Koutsouridis, Marios Logaras, Andreas Loukas, Alexandros Nafas, Georgios Oikonomou, Irini Thireou, Nikos Vasilakis, Panagiotis Kokkinos, Georgios Mylonas and Ioannis Chatzigiannakis, A Software Platform for Developing Multiplayer Pervasive Games Using Small Programmable Object Technologies, 5th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS08), Atlanta, Georgia, 2008. *Received the Best Demo Award.*

[80] Subversion Concurrent Versions System, Σύστημα διαχείρισης εκδόσεων πηγαίου κώδικα, στην ιστοσελίδα <http://subversion.apache.org/>

[81] Ant, a Java library and command-line tool, <http://ant.apache.org/>

[82] Hibernate, a powerful, high performance object/relational persistence and query service. <https://www.hibernate.org/>

Περιεχόμενα CD

Ο οπτικός δίσκος (CD) ο οποίος συνοδεύει την διπλωματική εργασία περιέχει μία έκδοση του κώδικα (συγκεκριμένα έκδοση 1418) όπως αυτή βρίσκεται στο αποθετήριο (repository) πηγαίου κώδικα. Η τελευταία έκδοση είναι κάθε φορά διαθέσιμη από το <http://ru1.cti.gr/svn/finn> μέσω κάποιου προγράμματος πελάτη Subversion [80].

Πιο αναλυτικά, η δομή της ιεραρχίας φακέλων περιγράφεται ως ακολούθως:

Ακόμη, στον δίσκο περιέχεται μία έκδοση του Liferay όπου κάποιος μπορεί να εγκαταστήσει και να δοκιμάσει και ο κώδικας της βάσης της διαδικτυακής πύλης, ανάλογα με τον τύπο βάσης που θα χρησιμοποιηθεί (περιέχει δεδομένα επίδειξης, για κάποιον που επιθυμεί να χρησιμοποιήσει την διαδικτυακή πύλη εύκολα και γρήγορα). Επίσης, περιέχεται η τελευταία έκδοση –κατά την συγγραφή αυτού του κειμένου– της Processing (συγκεκριμένα είναι η έκδοση 1.09) που περιλαμβάνει τις βιβλιοθήκες και το περιβάλλον ανάπτυξης. Τέλος, στην κορυφή της ιεραρχίας φακέλων βρίσκεται και η παρούσα διπλωματική σε ψηφιακή μορφή.

186

- *FinN*
 - db
 - lib
 - resources
 - src
- *Liferay*
 - data
 - deploy
 - license
- *SQL*
 - create
 - create-minimal
 - indexes
 - portal
 - portal-minimal
 - sequences
- *Multimedia*
 - Pictures
 - Videos
- *Processing*
 - Linux
 - Mac OS X
 - Windows
- *Thesis.pdf*



