

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ  
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

**Αρχιτεκτονική Πρακτόρων  
για τη Διενέργεια Ψηφοφοριών**

**Γεώργιος Χ. Χαλκιαδάκης**

Μεταπτυχιακή Εργασία

Ηράκλειο, Οκτώβριος 1999

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ  
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ  
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

## Αρχιτεκτονική Πρακτόρων για τη Διενέργεια Ψηφοφοριών

Εργασία που υποβλήθηκε από τον

**Γεώργιο Χ. Χαλκιαδάκη**

ως μερική εκπλήρωση των απαιτήσεων για την απόκτηση  
ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ

Συγγραφέας:

---

Γιώργος Χ. Χαλκιαδάκης  
Τμήμα Επιστήμης Υπολογιστών  
Πανεπιστήμιο Κρήτης

Εισηγητική Επιτροπή:

---

Στέλιος Ορφανουδάκης, Καθηγητής, Επόπτης

---

Πάνος Τραχανιάς, Αναπληρωτής Καθηγητής, Μέλος

---

Βαγγέλης Μαρκάτος, Επίκουρος Καθηγητής, Μέλος

Δεκτή:

---

Πάνος Κωνσταντόπουλος, Καθηγητής  
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

Οκτώβριος 1999

# Αρχιτεκτονική Πρακτόρων για τη Διενέργεια Ψηφοφοριών

Γεώργιος Χ. Χαλκιαδάκης

Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών  
Πανεπιστήμιο Κρήτης

## ΠΕΡΙΛΗΨΗ

Η ψηφοφορία είναι ίσως η πλέον διαδεδομένη διαδικασία επιλογής αποφάσεων από κοινωνίες. Οι σύγχρονες τεχνολογικές εξελίξεις έχουν κάνει δυνατή την τέλεση ψηφοφοριών με ηλεκτρονικό τρόπο, ενώ η έννοια της ψηφοφορίας είναι οικεία και στην Επιστήμη Υπολογιστών, καθώς χρησιμεύει στην επίλυση προβλημάτων - όπως η διασφάλιση της συνέπειας κατανεμημένων συστημάτων αρχείων ή η επιλογή ενεργειών κατά την πλοήγηση ρομπότ - τα οποία παρουσιάζονται σε διάφορους τομείς της.

Έχει αποδειχτεί μαθηματικά πως είναι αδύνατον να βρεθεί μια απλή, "δίκαιη" - με βάση καθορισμένα κριτήρια δικαιοσύνης - και συνεπής διαδικασία για τον προσδιορισμό του αποτελέσματος μιας εκλογής που αφορά περισσότερους από δυο υποψηφίους ("Θεώρημα του Ανέφικτου του Arrow"). Το γεγονός αυτό οδήγησε στην ανάπτυξη της Θεωρίας Ψηφοφοριών, η οποία ασχολείται με την μαθηματική αντιμετώπιση της διαδικασίας με την οποία δημοκρατικές κοινωνίες ή ομάδες μετασχηματίζουν τις πιθανότατα πολλές και αντικρουόμενες απόψεις των μελών τους, όσον αφορά ένα συγκεκριμένο θέμα ή πρόβλημα, σε μια και μόνη επιλογή για όλη την κοινωνία ή ομάδα. Η Επιστήμη Υπολογιστών έχει ως τώρα χρησιμοποιήσει αρκετά απλά σχήματα ψηφοφοριών, παρά την πολυπλοκότητα της

Θεωρίας Ψηφοφοριών και το γεγονός πως ο ορισμός εκλογικού συστήματος και κατάλληλων εκλογικών διαδικασιών είναι βασικοί παράγοντες που επηρεάζουν το αποτέλεσμα μιας ψηφοφορίας.

Η παρούσα μεταπτυχιακή εργασία είχε ως αντικείμενο την ανάλυση, το σχεδιασμό και την υλοποίηση μιας αρκετά γενικής αρχιτεκτονικής, της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών, που να επιτρέπει τη διενέργεια ψηφοφοριών με χρήση οποιουδήποτε εκλογικού συστήματος. Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι μια αρχιτεκτονική πολλών συνεργαζόμενων κατανεμημένων πρακτόρων λογισμικού που αντιπροσωπεύουν πραγματικές οντότητες. Οι οντότητες αυτές εντοπίστηκαν ως σημαντικές για τη διενέργεια ψηφοφοριών μετά από μελέτη της Θεωρίας Ψηφοφοριών και ενός γενικού παραδείγματος ψηφοφορίας, προερχόμενου από τον πραγματικό κόσμο. Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών σχεδιάστηκε με βάση τις αρχές της οντοκεντρικής σχεδίασης συστημάτων, και είναι μια επεκτάσιμη αρχιτεκτονική. Επιτρέπει την τέλεση παράλληλων ψηφοφοριών, και μπορεί να χρησιμοποιηθεί ως υποδομή τέλεσης ψηφοφοριών από αρχιτεκτονικές πρακτόρων, ή από άλλες εφαρμογές στις οποίες μπορούν να χρησιμεύσουν ψηφοφορίες. Η ανάκτηση εικόνων με βάση το περιεχόμενό τους προτείνεται ως μια τέτοια εφαρμογή.

Επόπτης: Στέλιος Ορφανουδάκης

Καθηγητής Επιστήμης Υπολογιστών

Πανεπιστήμιο Κρήτης

# **An Agent-Based Architecture for the Conduction of Voting**

Georgios Ch. Chalkiadakis

Master of Science Thesis

Department of Computer Science

University of Crete

## **ABSTRACT**

*Voting* is perhaps the most commonly used decision selection procedure that communities have. Modern technological developments have permitted voting to be conducted electronically; computer scientists are also familiar with the concept of voting, since it is used to solve various problems - such as assuring the consistency of a distributed file system, or performing action-selection while trying to navigate a robot - that appear in various sections of Computer Science.

It has been proved mathematically that it is impossible to find a simple, "fair" - according to defined Fairness Criteria - and consistent procedure that could specify the result of an election in which more than two candidates participate ("Arrow's Impossibility Theorem"). This fact has led to the development of Voting Theory, which is the mathematical treatment of the process by which democratic societies or groups resolve the many and conflicting opinions of the members of the group into a single choice of the group. So far Computer Science has used relatively simple voting schemes, despite the complexity of Voting Theory and the fact that the definition of a voting system and of appropriate voting procedures imposes major influence on the result of a voting session.

The objective of this master thesis was to design, analyse and implement a quite general architecture, namely the Voting Conduction Architecture, that permits the conduction of voting by use of any appropriate voting system. The Voting Conduction Architecture is an architecture of multiple co-operating distributed software agents that represent real entities. These entities were traced and considered important for the conduction of voting, following the study of Voting Theory and of a generic, real-world voting paradigm. The Voting Conduction Architecture was designed according to the principles of object-oriented systems' design, and is an extensible architecture. It allows voting sessions to be conducted in parallel, and it can be used as an infrastructure for the conduction of voting by agent-based architectures, or other applications that may utilise voting. Image retrieval by content is suggested as such an application.

Supervisor: Stelios Orphanoudakis  
Professor of Computer Science  
University of Crete

## Ευχαριστίες

Καταρχήν θέλω να αναγνωρίσω τη σημαντική συμβολή του Τμήματος Επιστήμης Υπολογιστών του Πανεπιστημίου Κρήτης, καθώς και του Ινστιτούτου Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας, τα οποία μου παρείχαν τους απαραίτητους πόρους και τον απαιτούμενο εξοπλισμό για την ολοκλήρωση της παρούσας εργασίας.

Θα ήθελα επίσης να ευχαριστήσω:

Τον επόπτη και καθηγητή μου, κ. Στέλιο Ορφανουδάκη, που με τις ιδέες και την καθοδήγησή του βοήθησε στο να πάρει μορφή η μεταπτυχιακή αυτή εργασία.

Τα μέλη της Εισηγητικής Επιτροπής, κ. Μαρκάτο και κ. Τραχανιά, για τις υποδείξεις και τις διορθώσεις τους.

Την Ελένη Καλδούδη, που βοήθησε σημαντικά στον προσδιορισμό του θέματος και στα αρχικά στάδια του σχεδιασμού της Αρχιτεκτονικής.

Τον Αντώνη Αργυρό, που αφιέρωσε ένα μέρος από το χρόνο του για να συμβάλει στη διόρθωση των αρχικών εκδόσεων της αναφοράς και να μου δώσει ιδέες για τμήματα του κειμένου.

Ένα ιδιαίτερο ευχαριστώ οφείλω στον Ξενοφώντα Ζαμπούλη, που συνέβαλε αποφασιστικά στην διαμόρφωση της Αρχιτεκτονικής - αλλά και της υλοποίησής της - μέσα από τις πολλές σχετικές συζητήσεις που κάναμε.

Ακόμη ευχαριστώ:

Όλα τα μέλη του Κέντρου Ιατρικής Πληροφορικής και Τηλεματικών Εφαρμογών στην Υγεία, για τη βοήθειά τους όλα αυτά τα χρόνια. Ιδιαίτερα ευχαριστώ τον Δημήτρη Κατεχάκη, που βοήθησε σημαντικά στην αύξηση των γνώσεών μου και στην βελτίωση του τρόπου εργασίας μου. Επίσης, ευχαριστώ το Μάριο Ζήκο, που όποτε του ζήτησα βοήθεια ήταν πρόθυμος να την παράσχει, και που το κείμενο της δικής του μεταπτυχιακής εργασίας με βοήθησε πολύ στον τρόπο συγγραφής του αντίστοιχου δικού μου.

Όλους τους μεταπτυχιακούς του Τμήματος Επιστήμης Υπολογιστών (άλλως CSD), που έτυχε να δουλεύουμε κατά καιρούς πλάι - πλάι, να αστευόμαστε και να στηρίζουμε ο ένας τον άλλο. Τους ευχαριστώ για την αλληλεγγύη τους στη δουλειά και στα ξενύχτια (για δουλειά ή μή). Ιδιαίτερα ευχαριστώ τον "οικιστή του Iivas"

Μανώλη Μαραζάκη, που ένα δύσκολο πρωινό με βοήθησε να ξεπεράσω ένα πρόβλημα του κώδικα. Επίσης το Σπύρο το Λιάπη, τον Ηλία το Γκρίνια, τη Μαρία Τράκα, την Ελευθερία Τζόβα, τον Σταύρο τον Τζουρμπάκη, τον Καλοψικάκη το Δημήτρη και την Άννα Κυρίκογλου, που μαζί μοιραστήκαμε το άγχος των τελευταίων μηνών, δουλεύοντας στο ΙΤΕ ή στο CSD, και που πάντα θα τους θεωρώ αγαπημένους φίλους.

Ακόμη, πολλά ευχαριστώ πάνε και στην υπόλοιπη "παλιοπαρέα", που μου πρόσφερε τη φιλία της και που συνέβαλε αποφασιστικά, καταναλώνοντας κυρίως αλκοολούχα ποτά (μαζί και με πολλούς από τους προαναφερόμενους), στην οικονομική ευρωστία συγκεκριμένων φοιτητικών στεκιών του Ηρακλείου: στον Σέντη τον Πέτρο, στον Κώστα το Χατζηνικόλα, στο Δημήτρη Ντοά, στο Σιδηρόπουλο τον Αντώνη, στο Θανάση το Χρυσό, στον Αγγελέτο το Μάριο, στον Κομοντάκη το Νίκο, στον Παύλο Κεφαλάκη, στον Μανώλη Πετράκη και φυσικά στο Σωτήρη τον Στογιαννόπουλο (αν και ο τελευταίος δεν έπινε και "χάλαγε την πιάτσα").

Πολλά ευχαριστώ και στους υπόλοιπους αγαπημένους φίλους, που η φιλία τους με τίμησε και με στήριξε: στο Θοδωρή τον Πιτσόλη, στον Αντρέα Πάσσαμ και στη Μαριάννα Κατρινάκη, στον Μπάμπη τον Αυγουστάκη, στο Βασίλη τον Τριγωνάκη και στην Εύα Ξενογιαννοπούλου, στην Ελεάνα Βλαχάκη, στο Λευτέρη Σιδηρουργό, στο Νίκο Γιανναδάκη, στον Παναγιώτη Βλάχο, στο Βασιλειάδη τον Σάββα, στην Άντα Χρυσού (που υπήρξε καταπληκτική συμπαραγωγός στον "Αιχμή FM"), στην Αντιγόνη Καπράνα και στη Ρούλα Νικορέλου (αν και το παράκανε με τις φωτογραφίες στην παρουσίαση της εργασίας). Επίσης, ευχαριστώ από την καρδιά μου τη Θέμιδα Γκικοπούλου για την φιλία της. Ακόμη, με ιδιαίτερη συγκίνηση θα θυμάμαι πάντα όλα τα παιδιά που δουλέψαμε κάποτε μαζί για να εκδώσουμε το φοιτητικό περιοδικό "Μίτος της Αριάδνης".

Ευχαριστώ επίσης το Δημήτρη, το Μάνο και τη Δέσποινα, για τη φιλία τους που κρατάει τόσα χρόνια, και για το ότι ήταν πάντα κοντά μου, όσο μακριά κι αν βρεθήκαμε στο χώρο.

Τέλος, ευχαριστώ τους γονείς μου Μπάμπη και Μαρία, που με στήριξαν με την αγάπη τους όλα αυτά τα χρόνια. Σε αυτούς αφιερώνω αυτή την εργασία.

Ηράκλειο, Οκτώβρης 1999

Γιώργος Χαλκιαδάκης



*... Democracy must be structurally modernized, must be mechanically implemented to give it a one-individual-to-another speed and spontaneity of reaction commensurate with the speed and scope of broadcast news now world-wide in seconds.*

**R. Buckminster Fuller, 1940**

## Περιεχόμενα

---

Πίνακας Σχημάτων .....	iv
Πίνακας Εικόνων.....	iv
<b>Κεφάλαιο 1 Εισαγωγή .....</b>	<b>1</b>
1.1 Σκοπός της Εργασίας.....	2
1.2 Το Αρχικό Κίνητρο της Εργασίας.....	3
1.3 Σχετικές Εργασίες.....	4
1.4 Σύνοψη .....	4
1.5 Δομή της Εργασίας .....	5
<b>Κεφάλαιο 2 Θεωρία Ψηφοφοριών.....</b>	<b>7</b>
2.1 Βασική Ορολογία.....	7
2.2 Η Σχετικότητα της Έννοιας της Δικαιοσύνης στις Ψηφοφορίες.....	8
2.3 Κριτήρια Δικαιοσύνης.....	9
2.4 Εκλογικά Συστήματα.....	10
2.5 Συλλογή Ψηφοδελτίων .....	12
2.6 Ηλεκτρονικές Ψηφοφορίες.....	14
2.7 Χρήση Μηχανισμών Ψηφοφοριών στην Επιστήμη Υπολογιστών.....	14
<b>Κεφάλαιο 3 Πράκτορες.....</b>	<b>16</b>
3.1 Χαρακτηριστικά των Πρακτόρων.....	17
3.2 Αρχιτεκτονικές Πρακτόρων .....	18
3.3 Συστήματα Πολλών Πρακτόρων .....	20
3.4 Τεχνητή Νοημοσύνη και Συστήματα Πρακτόρων.....	21
3.5 Γλώσσες Επικοινωνίας Πρακτόρων .....	22
3.6 Χρήσεις των Πρακτόρων .....	24
<b>Κεφάλαιο 4 Αρχιτεκτονική.....</b>	<b>25</b>
4.1 Απαιτήσεις και Επιλογές .....	25
4.2 Ένα Παράδειγμα Ψηφοφορίας στην Καθημερινή Ζωή.....	27
4.3 Πράκτορες, Βασικές Οντότητες και μεταξύ τους Αλληλεπιδράσεις .....	28
4.4 Αρχιτεκτονική ενός Πράκτορα .....	30
4.5 Παρουσίαση των Πρακτόρων και των Βασικών Οντοτήτων.....	30
4.5.1 Γερουσία.....	30
4.5.2 Υποψήφιος.....	32
4.5.3 Εφορευτική Επιτροπή .....	33
4.5.4 Ψηφοφόρος .....	36
4.5.4.1 Διαμόρφωση Άποψης Ψηφοφόρου .....	37
4.5.5 Ψηφοφορία.....	38
4.5.6 Εκλογικό Σύστημα .....	38
4.5.7 Ψηφοδέλτιο .....	39
4.5.8 Εκλογικό Αποτέλεσμα.....	40
4.5.9 Κάλπη.....	40
4.6 Διαχείριση Ομάδας Πρακτόρων και Ψηφοφορίες σε Ομάδες .....	41
4.6.1 Ψηφοφορία σε Ομάδα.....	42
4.7 Ανταλλασσόμενα Μηνύματα.....	43

<b>Κεφάλαιο 5 Ζητήματα Υλοποίησης</b> .....	<b>47</b>
5.1 Επιλογή Γλώσσας Προγραμματισμού .....	47
5.2 Επιλογές Επικοινωνίας .....	48
5.3 Πολυνηματικοί Πράκτορες .....	49
5.4 Εκλογικά Συστήματα που Υλοποιήθηκαν.....	50
5.5 Ιεραρχία Κλάσεων .....	53
<b>Κεφάλαιο 6 Πειράματα</b> .....	<b>56</b>
<b>Κεφάλαιο 7 Συμπεράσματα και Μελλοντική Εργασία</b> .....	<b>61</b>
Παράρτημα I : Γνωρίσματα και Λειτουργίες Βασικών Οντοτήτων .....	66
Παράρτημα II : Γλωσσάρι Όρων Θεωρίας Ψηφοφοριών .....	78
Παράρτημα III : Παραδείγματα Παραβίασης Κριτηρίων Δικαιοσύνης .....	81
Παράρτημα IV : Επισκόπηση Γλωσσών Επικοινωνίας Πρακτόρων και Γλωσσών Αναπαράστασης Γνώσης .....	84
Βιβλιογραφία.....	95
Παραπομπές σε Ηλεκτρονικές Διευθύνσεις.....	98

## Πίνακας Σχημάτων

---

• Σχήμα 1 Αλληλεπίδραση πράκτορα με το περιβάλλον του.	17
• Σχήμα 2 Πράκτορας με Αρχιτεκτονική Απλής Αντίδρασης.	19
• Σχήμα 3 Πράκτορας με Αρχιτεκτονική Σκοπών.	20
• Σχήμα 4 Πράκτορας με Αρχιτεκτονική Ωφέλειας.	20
• Σχήμα 5 Αλληλεπιδράσεις μεταξύ βασικών οντοτήτων.	29
• Σχήμα 6 Επικοινωνία Γερουσίας - Εφορευτικής Επιτροπής.	34
• Σχήμα 7 Συνοπτική παρουσίαση της συλλογής των ψηφοδελτίων.	35
• Σχήμα 8 Χρήση του Εκλογικού Συστήματος για υπολογισμό του Αποτελέσματος της Ψηφοφορίας.	39
• Σχήμα 9 Δυναμική Δημιουργία Υπο-Ομάδων Ψηφοφοριών.	43
• Σχήμα 10 Η ιεραρχία υλοποιημένων Java κλάσεων.	55
• Σχήμα 11 Πιθανή ιεραρχία επιπέδων ψηφοφοριών για απόφαση περί ομοιότητας εικόνων.	62
• Σχήμα 12 Σύγκριση τμηματοποιημένων εικόνων εντός του χώρου κλιμάκων.	63
• Σχήμα 13 Τα επίπεδα της KQML.	86
• Σχήμα 14 Επικοινωνία πρακτόρων μέσω facilitators.	89
• Σχήμα 15 SKTP Αρχιτεκτονική. Κάθε τμήμα της αντιστοιχεί σε ένα από τα κύρια επίπεδα της KQML.	89
• Σχήμα 16 Πίνακας KQML performatives.	90

## Πίνακας Εικόνων

---

• Εικόνα 1 Διεπιφάνειες χρήσης Γερουσίας και Ψηφοφόρου.	55
• Εικόνα 2 Πείραμα δημοτικών εκλογών: Ανάδειξη του Υποψηφίου Α ως νικητή, με χρήση Μεθόδου Απλής Πλειοψηφίας.	57
• Εικόνα 3 Πείραμα δημοτικών εκλογών: Επιλογή του συστήματος Borda Count για τη διεξαγωγή της ψηφοφορίας.	59
• Εικόνα 4 Πείραμα δημοτικών εκλογών: Ανάδειξη του Β ως νικητή με το εκλογικό σύστημα Borda Count.	59
• Εικόνα 5 Πείραμα Επιλογής Ενέργειας: Η ενέργεια #6 είναι νικήτρια.	60

# Κεφάλαιο 1 Εισαγωγή

Πολλές φορές στην καθημερινή ζωή, προκύπτουν καταστάσεις στις οποίες μια ομάδα καλείται να λάβει μια απόφαση, έχοντας να διαλέξει μεταξύ πολλών επιλογών. Η συνηθισμένη μέθοδος επιλογής είναι τότε η ψηφοφορία μεταξύ των μελών της ομάδας. Δεν υπάρχει όμως ένας συγκεκριμένος τρόπος για τη λήψη της απόφασης με ψηφοφορία, αν οι δυνατές επιλογές είναι άνω των δύο. Μια πληθώρα από εκλογικά συστήματα έχουν προταθεί: κάποια λαμβάνουν υπόψη μόνο την πρώτη προτίμηση του ψηφοφόρου, κάποια άλλα λαμβάνουν υπόψη τη σειρά των προτιμήσεών του, ενώ άλλα επιτρέπουν να δίδεται ξεχωριστό βάρος σε ορισμένες προτιμήσεις. Υπάρχουν πολλές τεχνικές για την συγκέντρωση και την αποτίμηση των ψηφοδελτίων, και συνήθως το εκλογικό αποτέλεσμα είναι συνάρτηση του εκλογικού συστήματος και της διαδικασίας συγκέντρωσης των ψήφων.

Ταυτόχρονα, η πρόοδος της Πληροφορικής και των Τηλεπικοινωνιών, έχει κάνει εφικτή τη χρήση νέων τεχνικών συγκέντρωσης και αποτίμησης των απόψεων μιας ομάδας εκλογέων. Οι ηλεκτρονικές ψηφοφορίες για διάφορα θέματα είναι πολύ συνηθισμένες πια στον Παγκόσμιο Ιστό (WWW), ο οποίος γνωρίζει μεγάλη ανάπτυξη τα τελευταία χρόνια. Δεν είναι μόνο όμως οι ηλεκτρονικές ψηφοφορίες μεταξύ ανθρώπων-χρηστών εκείνες κατά τις οποίες γίνεται χρήση της Πληροφορικής: με δεδομένη την ανάπτυξη των πληροφοριακών συστημάτων, πολλές είναι οι περιπτώσεις που διάφορες υπολογιστικές οντότητες καλούνται να αποφασίσουν για κάποιο θέμα χρησιμοποιώντας ψηφοφορία. Σε άλλες πάλι περιπτώσεις, η ψηφοφορία μεταξύ των τμημάτων μιας υπολογιστικής οντότητας είναι εκείνη που καθορίζει την τελική της συμπεριφορά. Συχνά επίσης μια μέθοδος ψηφοφορίας μεταξύ αντιγράφων αρχείων σε διαφορετικούς κόμβους ενός δικτύου, είναι αυτή που εξασφαλίζει τη συνέπεια του κατανεμημένου συστήματος αρχείων που τα περιλαμβάνει. Επιπλέον, η διενέργεια ψηφοφοριών μεταξύ των μελών μιας ομάδας υπολογιστικών πρακτόρων που έχουν ένα κοινό στόχο, φαίνεται να αποτελεί ένα καλό τρόπο για τη μελέτη της διαμόρφωσης των απόψεων και της συμπεριφοράς των μελών της ομάδας στο πέρασμα του χρόνου. Επίσης, αποτελεί ένα μέσο για την απόκτηση εμπειρίας για τα μέλη μα και την ομάδα συνολικά (μια που, λόγου χάρη, η υιοθέτηση μιας πρότασης που αποδείχτηκε λανθασμένη είναι δυνατόν να μην επαναληφθεί την επόμενη φορά που το εκλογικό σώμα θα κληθεί να ξανααποφασίσει για το ίδιο θέμα), αρκεί να παρέχονται οι κατάλληλοι μηχανισμοί για να επιτυγχάνεται κάτι τέτοιο.

## 1.1 Σκοπός της Εργασίας

Η παρούσα μεταπτυχιακή εργασία, έχει ως αντικείμενο τον σχεδιασμό και την υλοποίηση μιας αρχιτεκτονικής πολλών κατανεμημένων πρακτόρων οι οποίοι συνεργάζονται και δημιουργούν μια υποδομή κατάλληλη για τη διενέργεια ψηφοφοριών. Η ανάπτυξη της Αρχιτεκτονικής έγινε με χρήση οντοκεντρικών μεθόδων ανάλυσης και σχεδιασμού συστημάτων, και σύγχρονων εργαλείων μοντελοποίησης.

Η Αρχιτεκτονική Πρακτόρων για τη Διενέργεια Ψηφοφοριών (εν συντομία: "Αρχιτεκτονική Διενέργειας Ψηφοφοριών") είναι μια αρχιτεκτονική που βασίστηκε σε ένα μοντέλο ψηφοφορίας, το οποίο προέκυψε μετά από μελέτη του πραγματικού κόσμου. Επιδίωξη ήταν το μοντέλο αυτό να είναι όσο το δυνατόν γενικότερο, ώστε να βρίσκει εφαρμογή σε πληθώρα περιπτώσεων. Έτσι, κατά τον σχεδιασμό της Αρχιτεκτονικής έγινε εκτενής μελέτη της μαθηματικής Θεωρίας Ψηφοφοριών, προκειμένου να προσδιοριστούν οντότητες τέτοιες ώστε να ενσωματώνουν στοιχεία της και να ικανοποιούν απαιτήσεις που τίθενται από αυτήν. Η Αρχιτεκτονική είναι ικανή να υποστηρίξει *ιεραρχίες* ψηφοφοριών, καθώς και παράλληλες ψηφοφορίες με όμοια ή όχι θέματα, στις οποίες μπορεί να συμμετέχει ο ίδιος πληθυσμός ψηφοφόρων.

Η αρχιτεκτονική που παρουσιάζεται στην εργασία, επιτρέπει τη χρήση οποιουδήποτε εκλογικού συστήματος για μια ψηφοφορία που διενεργείται στα πλαίσιά της. Μπορεί να χρησιμεύσει έτσι ως πλατφόρμα μελέτης εκλογικών συστημάτων, αλλά και ως πλατφόρμα μελέτης της μεταβολής της συμπεριφοράς εκλογικών πληθυσμών ανάλογα με τις υπάρχουσες, κατά την ψηφοφορία, συνθήκες.

Η Αρχιτεκτονική Πρακτόρων για τη Διενέργεια Ψηφοφοριών είναι μια *επεκτάσιμη* αρχιτεκτονική πρακτόρων, η λειτουργικότητα της οποίας μπορεί εύκολα να επαυξηθεί.

Τέλος, βασικό χαρακτηριστικό της αρχιτεκτονικής είναι το ότι μπορεί να χρησιμεύσει σαν *υποδομή διενέργειας ψηφοφοριών*, που να χρησιμοποιείται από άλλες αρχιτεκτονικές πρακτόρων, ή από διάφορα συστήματα και προγραμματιστικές εφαρμογές. Αυτές, μεταξύ άλλων, μπορεί να είναι αρχιτεκτονικές ή εφαρμογές Μηχανικής Μάθησης (στις οποίες είναι δυνατόν να επιδιώκεται η προαγωγή της συνολικής γνώσης μέσω αξιολόγησης των αποτελεσμάτων συνεχόμενων ψηφοφοριών), εφαρμογές ψηφοφοριών στον Παγκόσμιο Ιστό, συστήματα πλοήγησης ρομπότ (όπου διαφορετικά υποσυστήματα μπορεί να εκφράζουν διαφορετική άποψη για το ποιά είναι η ορθή επόμενη ενέργεια του ρομπότ), ή και συστήματα που να χρησιμοποιούνται για να αντιμετωπίσουν το πρόβλημα της σύγκρισης ή ανάκτησης εικόνων με βάση το περιεχόμενό τους. Η διερεύνηση της δυνατότητας χρήσης ψηφοφοριών για τη λύση του τελευταίου προβλήματος, αποτέλεσε μάλιστα το αρχικό κίνητρο της μεταπτυχιακής αυτής εργασίας.

## 1.2 Το Αρχικό Κίνητρο της Εργασίας

Το πρόβλημα της ανάκτησης εικόνας με βάση το περιεχόμενό της ορίζεται ως το πρόβλημα της ανάκτησης μιας εικόνας με βάση κάποια κύρια χαρακτηριστικά της. Ένα τέτοιο χαρακτηριστικό μπορεί να είναι α) μια ολική ιδιότητα της εικόνας, για παράδειγμα η μέση τιμή της έντασης του γκρι, ή β) μια ειδική ιδιότητα σε κάποιο τμήμα της εικόνας, όπως λόγου χάρη ένας κύκλος, μια γραμμή ή μια επιφάνεια [66]. Για κάποια από τα χαρακτηριστικά της εικόνας υπάρχουν αλγόριθμοι που τα ανιχνεύουν, και η περιγραφή της εικόνας γίνεται με διανύσματα-περιγραφείς των ανιχνευόμενων χαρακτηριστικών.

Έχουν αναπτυχθεί αρκετά συστήματα διαχείρισης και ανάκτησης εικόνων με βάση το περιεχόμενό τους. Πολύ γνωστά τέτοια συστήματα είναι το Photobook [48], το VisualSEEK [60] και το QBIC [29], ενώ στο Ινστιτούτο Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας στην Κρήτη έχουν αναπτυχθεί τα I<sup>2</sup>C [46] και I<sup>2</sup>Cnet [47] για την ανάκτηση ιατρικών εικόνων. Πολλά από τα συστήματα αυτά έχουν ικανοποιητικές επιδόσεις, πάντα όμως μόνο ως ένα βαθμό, καθώς η σχετική έρευνα δεν έχει κατορθώσει να πετύχει την εξαγωγή και την ερμηνεία σημασιολογικής πληροφορίας από τα χαρακτηριστικά των εικόνων με τον τρόπο και την ευκολία που το κάνει ο άνθρωπος. Είναι μάλιστα γεγονός πως ακόμα και τώρα είναι για την Ψυχολογία ανοικτό πρόβλημα το πώς ακριβώς ο άνθρωπος εξηγεί την οπτικά αντιλαμβανόμενη πληροφορία. Πολλές είναι οι ψυχολογικές θεωρίες που έχουν προταθεί για την *εξήγηση της αντίληψης*.

Μια από αυτές τις θεωρίες παρουσιάστηκε από τον Selfridge το 1959, και προτείνει για την εξήγηση της αντίληψης την υιοθέτηση του μοντέλου του "Πανδαιμονίου" [58]. Σύμφωνα με αυτό το μοντέλο, το "γνωστικό σύστημα" του ανθρώπου περιλαμβάνει "μονάδες γνώσης" υπεύθυνες για ειδικά έργα, όπως η αναγνώριση ενός γράμματος. Οι μονάδες αυτές ονομάζονται *"γνωστικοί δαίμονες"*. Στο κατώτερο επίπεδο υπάρχει μια σειρά από *υποδαίμονες*, ο καθένας από τους οποίους αντιδρά προς μια συγκεκριμένη εικόνα ή στοιχείο αισθητήριων δεδομένων (ίσως μια ιδιαίτερη καμπύλη ενός γράμματος). Όταν ένας συγκεκριμένος υποδαίμονας διεγείρεται, "κραυγάζει". Όσο πιο όμοιος είναι ο ερεθισμός προς το χαρακτηριστικό γνώρισμα του υποδαίμονα, τόσο πιο "μεγαλόφωνη" θα είναι η "κραυγή". Στη συνέχεια, οι κραυγές συλλέγονται από τους γνωστικούς δαίμονες του επόμενου επιπέδου συνθετότητας και τους πυροδοτούν. Ωστόσο, μόνο οι γνωστικοί δαίμονες που επικοινωνούν με τους κραυγάζοντες δαίμονες θα πυροδοτηθούν και θα κραυγάσουν με τη σειρά τους (έτσι, αν το επίπεδο συνθετότητάς τους αφορά την αναγνώριση ενός γράμματος, θα κραυγάσουν μόνο εκείνοι οι οποίοι αντιδρούν σε όμοια -ή παρόμοια- γράμματα). Σε αυτό το επίπεδο, η ηχηρότητα των κραυγών των δαιμόνων εξαρτάται από το πόσο δυνατά κραυγάζουν οι υποδαίμονές τους. Οι κραυγές μεταδίδονται κατόπιν σε ένα *δαίμονα αποφάσεων* ο οποίος είναι υπεύθυνος για τον εντοπισμό του

δαίμονα τα χαρακτηριστικά του οποίου συμφωνούν περισσότερο με το αρχικό ερέθισμα, κι ο οποίος είναι κατά πάσα πιθανότητα αυτός που κραυγάζει ηχηρότερα.

Εύκολα το μοντέλο του Πανδαιμονίου μπορεί να συσχετιστεί με μια ιεραρχία ψηφοφοριών μεταξύ αλγορίθμων ανίχνευσης χαρακτηριστικών εικόνων, οι οποίοι βρίσκονται σε διαφορετικά επίπεδα και εκφράζουν την άποψη τους για την ομοιότητα δυο εικόνων με την ψήφο τους. Η θεωρία βέβαια του Πανδαιμονίου, είναι παραδεκτό από τους ψυχολόγους πως δεν εξηγεί πλήρως όλους τους τύπους αντίληψης. Η χρησιμοποίηση όμως ψηφοφοριών μεταξύ των χαρακτηριστικών των εικόνων, μπορεί να οδηγήσει ίσως στην αξιολόγηση του βαθμού σημασίας των διάφορων χαρακτηριστικών, μέσω πειραματισμών με χρήση πολλών διαφορετικών εκλογικών συστημάτων (που επιτρέπουν ή όχι "βάρη" στις ψήφους) και σύγκριση των εκλογικών αποτελεσμάτων.

Η δυνατότητα πειραματισμού με ένα τέτοιο σύστημα, αποτέλεσε το αρχικό ερέθισμα για την ανάπτυξη της Αρχιτεκτονικής Πρακτόρων για τη Διενέργεια Ψηφοφοριών, η οποία παρουσιάζεται σε αυτήν την μεταπτυχιακή εργασία.

### 1.3 Σχετικές Εργασίες

Η ανάλυση, ο σχεδιασμός και η υλοποίηση μιας γενικής αρχιτεκτονικής που να αποσκοπεί στην διενέργεια ψηφοφοριών δε φαίνεται να έχει απασχολήσει το χώρο της Επιστήμης Υπολογιστών ως τώρα, παρόλο που οι ηλεκτρονικές ψηφοφορίες γνωρίζουν άνθηση στον Παγκόσμιο Ιστό, και παρόλο που μηχανισμοί ψηφοφοριών χρησιμοποιούνται σε κάποιους τομείς της Επιστήμης Υπολογιστών.

Σε ειδική ενότητα του δεύτερου κεφαλαίου, το οποίο ασχολείται με τη Θεωρία Ψηφοφοριών, θα γίνει αναφορά σε εργασίες που κάνουν χρήση μηχανισμών ψηφοφοριών.

### 1.4 Σύνοψη

Η παρούσα μεταπτυχιακή εργασία, παρουσιάζει μια αρχιτεκτονική πολλών κατανεμημένων πρακτόρων λογισμικού, κατάλληλη για τη διενέργεια ψηφοφοριών, η οποία

- χρησιμοποιεί στοιχεία της μαθηματικής Θεωρίας Ψηφοφοριών,
- χρησιμοποιεί ένα γενικό μοντέλο ψηφοφορίας, βασιζόμενο σε ένα παράδειγμα του πραγματικού κόσμου,
- μπορεί να υποστηρίξει ιεραρχικά σχήματα ψηφοφοριών,



- μπορεί να υποστηρίξει την παράλληλη διενέργεια σχετιζόμενων ή μη ψηφοφοριών με τη συμμετοχή του ίδιου πληθυσμού ψηφοφόρων,
- επιτρέπει τη διενέργεια ψηφοφοριών με χρήση οποιουδήποτε εκλογικού συστήματος,
- είναι επεκτάσιμη,
- μπορεί εύκολα να διασυνδεθεί με άλλες αρχιτεκτονικές πρακτόρων ή να χρησιμοποιηθεί από εφαρμογές που επιθυμούν να διεξάγουν ψηφοφορίες (για παράδειγμα εφαρμογές σύγκρισης εικόνων, ή εφαρμογές συλλογής πληροφοριών στο διαδίκτυο), και
- σχεδιάστηκε με βάση τις αρχές της οντοκεντρικής ανάλυσης και σχεδιασμού συστημάτων, κάνοντας χρήση σχετικής σύγχρονης τεχνολογίας.

Στα πλαίσια της ανάπτυξης της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών, υλοποιήθηκε ένα σύνολο από εκλογικά συστήματα, μετά από μελέτη των δυνατοτήτων τους. Επίσης, αναπτύχθηκαν γραφικές διεπιφάνειες χρήσης των πρακτόρων, κατάλληλες για την εύκολη επίδειξη της Αρχιτεκτονικής.

## 1.5 Δομή της Εργασίας

Στο επόμενο Κεφάλαιο γίνεται μια επισκόπηση της Θεωρίας Ψηφοφοριών, η οποία αποτελεί, μαζί με την τεχνολογία πρακτόρων, το θεωρητικό υπόβαθρο της εργασίας. Γίνεται επίσης αναφορά στο πώς η σύγχρονη τεχνολογία μπορεί να χρησιμοποιηθεί για την τέλεση ψηφοφοριών και επισκόπηση της χρήσης ψηφοφοριών στην Επιστήμη Υπολογιστών.

Στο Κεφάλαιο 3 γίνεται παρουσίαση της έννοιας του "πράκτορα", και παρουσιάζονται τα βασικά στοιχεία της τεχνολογίας πρακτόρων. Γίνεται αναφορά στα χαρακτηριστικά των πρακτόρων, στις αρχιτεκτονικές τους και στις αρχιτεκτονικές συστημάτων πρακτόρων, στη σχέση της τεχνολογίας πρακτόρων με την Τεχνητή Νοημοσύνη, στις γλώσσες επικοινωνίας πρακτόρων και στις χρήσεις τους.

Στο Κεφάλαιο 4 παρουσιάζεται αναλυτικά η Αρχιτεκτονική Διενέργειας Ψηφοφοριών. Γίνεται αναφορά στον τρόπο σχεδιασμού της, στις απαιτήσεις της και στις επιλογές που έγιναν, στην εσωτερική αρχιτεκτονική των πρακτόρων και στον τρόπο επικοινωνίας τους. Ακόμη, γίνεται αναλυτική παρουσίαση των πρακτόρων και των αλληλεπιδράσεών τους.

Στο Κεφάλαιο 5 αναφέρονται διάφορα ζητήματα και επιλογές που αφορούν την υλοποίηση της Αρχιτεκτονικής. Μεταξύ άλλων, αναφέρονται οι επιλογές γλώσσας προγραμματισμού και τρόπου υλοποίησης της επικοινωνίας, όπως και ο τρόπος υλοποίησης της δομής των

πρακτόρων. Ακόμη, παρουσιάζονται τα εκλογικά συστήματα που υλοποιήθηκαν στα πλαίσια της εργασίας.

Στο Κεφάλαιο 6 αναφέρονται τα πειράματα που έγιναν για να δοκιμαστεί η Αρχιτεκτονική Διενέργειας Ψηφοφοριών. Τέλος, στο Κεφάλαιο 7 συζητούνται τα συμπεράσματα της εργασίας, και αναφέρονται πιθανές μελλοντικές επεκτάσεις και βελτιώσεις της.

## Κεφάλαιο 2 Θεωρία Ψηφοφοριών

Η Θεωρία Ψηφοφοριών είναι η μαθηματική θεωρία που ασχολείται με την αντιμετώπιση της διαδικασίας με την οποία δημοκρατικές κοινωνίες ή ομάδες μετασχηματίζουν τις πιθανότατα πολλές και αντικρουόμενες απόψεις των μελών τους, όσον αφορά ένα συγκεκριμένο θέμα ή πρόβλημα, σε μια και μόνη επιλογή για όλη την κοινωνία ή ομάδα [16][82].

Στο Κεφάλαιο αυτό θα γίνει μια επισκόπηση των βασικών αρχών και εννοιών που σχετίζονται με τη Θεωρία Ψηφοφοριών, ώστε να αναδειχτεί η πολυπλοκότητα και η πληθωρικότητα του θέματος που πραγματεύεται. Επίσης, θα γίνει αναφορά στο πώς η σύγχρονη τεχνολογία μπορεί να χρησιμοποιηθεί για την τέλεση ψηφοφοριών και θα γίνει επισκόπηση της χρήσης ψηφοφοριών στην Επιστήμη Υπολογιστών.

### 2.1 Βασική Ορολογία

Η *ψηφοφορία* είναι η διαδικασία συλλογής και αποτίμησης των απόψεων των μελών ενός αρμόδιου εκλογικού σώματος *ψηφοφόρων /εκλογέων*, προκειμένου να ληφθεί μια απόφαση που αφορά ένα σαφώς καθορισμένο θέμα. Η *ψήφος* αποτελεί την έκφραση της προτίμησης του ψηφοφόρου για το *αποτέλεσμα* της ψηφοφορίας και καταγράφεται στο *ψηφοδέλτιο*, το οποίο αποτελεί το αρχείο που δεικνύει το πώς ψήφισε ο ψηφοφόρος που το χρησιμοποίησε. Πολλές φορές το ψηφοδέλτιο είναι ένα *ψηφοδέλτιο προτιμήσεων*, όπου ο ψηφοφόρος απαριθμεί τους *υποψηφίους* ή τις *εναλλακτικές λύσεις* κατά σειρά προτίμησης. Η έννοια του "υποψηφίου" αντιπροσωπεύει στη Θεωρία Ψηφοφοριών μια πιθανή εναλλακτική λύση που μπορεί να επιλεγεί μέσα από την ψηφοφορία ως το τελικό αποτέλεσμά της, το οποίο προκύπτει με συλλογή των ψηφοδελτίων και εφαρμογή του *εκλογικού συστήματος*, δηλαδή μιας συγκεκριμένης διαδικασίας μαθηματικού συγκερασμού και αποτίμησης των εκφρασμένων απόψεων [16][82].

Βασικές έννοιες στη Θεωρία Ψηφοφοριών είναι κι αυτές της *απόλυτης* και της *απλής πλειοψηφίας*. Απόλυτη πλειοψηφία (majority) συγκεντρώνει ο υποψήφιος που συγκεντρώνει *πάνω από τις μισές ψήφους*. Απλή ή σχετική πλειοψηφία (plurality) έχει ο υποψήφιος που απλά συγκεντρώνει περισσότερες ψήφους από κάθε άλλον υποψήφιο [16][82].

Στο Παράρτημα II παρέχεται ένα σύντομο γλωσσάρι όρων της Θεωρίας Ψηφοφοριών.

## 2.2 Η Σχετικότητα της Έννοιας της Δικαιοσύνης στις Ψηφοφορίες

Η έννοια της ψηφοφορίας όπως μας είναι γνωστή από την καθημερινή ζωή φαντάζει αρκετά απλή για να δικαιολογεί την ύπαρξη μιας μαθηματικής θεωρίας που να ασχολείται με αυτήν. *Είναι άραγε τόσο δύσκολο να βρεθεί μια απλή, δίκαιη και συνεπής διαδικασία για τον προσδιορισμό του αποτελέσματος μιας εκλογής;*

Πράγματι, όταν η εκλογή αφορά μόνο δυο υποψηφίους, τότε η κατάσταση είναι απλή: Οι ψηφοφόροι καταγράφουν στο ψηφοδέλτιο τους το όνομα του ενός εκ των δύο υποψηφίων, τον οποίο προτιμούν, και είναι συνεπές και δίκαιο να επιλεγεί ο υποψήφιος εκείνος που συγκεντρώνει την απόλυτη πλειοψηφία των ψήφων [16][82].

Τα πράγματα όμως περιπλέκονται όταν η εκλογή αφορά τρεις ή περισσότερους υποψηφίους. Το 1951, ο μαθηματικός και οικονομολόγος Kenneth Arrow διατύπωσε και απέδειξε το Γενικό Θεώρημα Πιθανότητας, περισσότερο γνωστό ως **Θεώρημα του Ανέφικτου του Arrow (Arrow's Impossibility Theorem)**, σύμφωνα με το οποίο, *"μέθοδοι για το πέρασμα από ατομικές προτιμήσεις σε κοινωνικές προτιμήσεις, οι οποίες θα είναι ικανοποιητικές<sup>1</sup> και θα οριστούν για ένα μεγάλο εύρος συνόλων ατομικών κατατάξεων προτιμήσεων, δεν μπορεί παρά μόνο να επιβληθούν ή να αποφασιστούν δικτατορικά<sup>2</sup>"* [3], και ως εκ τούτου δεν υπάρχει ικανοποιητική δημοκρατική<sup>3</sup> "συνάρτηση κοινωνικής ευμάρειας"<sup>4</sup>[3]. Το Θεώρημα δείχνει πως αν αξιωματικά οριστούν ορισμένες "προφανείς" δίκαιες συνθήκες - κριτήρια δικαιοσύνης<sup>5</sup> - για την επιλογή μιας λύσης που προάγει το κοινωνικό συμφέρον, *δεν υπάρχει καμμία δημοκρατική μέθοδος για το συγκερασμό ατομικών προτιμήσεων για τρεις ή περισσότερες εναλλακτικές λύσεις, τέτοια ώστε να ικανοποιεί τα κριτήρια δικαιοσύνης και να παράγει πάντα ένα λογικό αποτέλεσμα* [3]. Με άλλα λόγια, δεν υπάρχει καμμία συνεπής δημοκρατική μέθοδος που να επιλέγει δίκαια έναν από τρεις ή περισσότερους υποψηφίους, κάτι που σημαίνει πως καμμία εκλογική διαδικασία και κανένα εκλογικό σύστημα δεν μπορεί πάντοτε να αποφασίζει δίκαια για το αποτέλεσμα μιας εκλογικής μάχης μεταξύ άνω των δύο σε αριθμό υποψηφίων [16][82].

Αναλυτικότερα, ο Arrow διερεύνησε την πιθανότητα να υπάρχει ένα εκλογικό σύστημα που να εμπεριέχει τις ακόλουθες επιθυμητές ιδιότητες:

<sup>1</sup> "Ικανοποιητικές" θεωρούνται οι μέθοδοι αν δεν αντικατοπτρίζουν αρνητικά τις επιθυμίες των ατόμων, και αν οι συνεπαγόμενες κοινωνικές επιλογές αναπαριστώνται με μια κατάσταση που έχει τις συνήθεις ιδιότητες της λογικής οι οποίες αποδίδονται στις ατομικές κατατάξεις προτιμήσεων [3].

<sup>2</sup> Δηλαδή σύμφωνα με τις προτιμήσεις ενός συγκεκριμένου ψηφοφόρου.

<sup>3</sup> Με τη συνηθισμένη έννοια του όρου "δημοκρατική", μη δικτατορική.

<sup>4</sup> Συνάρτηση μετασχηματισμού ενός συνόλου από ατομικές κατατάξεις για διαφορετικές κοινωνικές επιλογές, σε μία αντίστοιχη κοινωνική (συνολική) κατάσταση των διαφορετικών κοινωνικών επιλογών [3].

<sup>5</sup> Δηλαδή ευρύτητα αποδεκτές προϋποθέσεις για να θεωρηθεί σωστός ο μετασχηματισμός του συνόλου των ατομικών προτιμήσεων σε κοινωνική επιλογή. Οι όροι "δίκαιος" και "δικαιοσύνη" χρησιμοποιούνται εδώ με την συνηθισμένη τους έννοια. Ο αναγνώστης μπορεί για περισσότερες πληροφορίες να ανατρέξει σε βιβλιογραφία των πολιτικών και οικονομικών επιστημών.

- Την Παγκοσμιότητα: Το εκλογικό σύστημα θα έπρεπε να παρέχει μία πλήρη κατάταξη όλων των επιλογών των προερχόμενων από οποιοδήποτε σύνολο ψηφοδελτίων ατομικών προτιμήσεων.
- Την Μονοτονία: Αν ένα σύνολο από ψηφοδέλτια προτιμήσεων οδηγεί σε μια συνολική κατάταξη του υποψηφίου  $X$  πάνω από τον υποψήφιο  $Y$ , και αν κάποια από τα ψηφοδέλτια μεταβληθούν ούτως ώστε ο μόνος υποψήφιος με υψηλότερη κατάταξη σε οποιοδήποτε από αυτά είναι ο  $X$ , τότε το εκλογικό σύστημα πρέπει να εξακολουθεί να κατατάσσει τον  $X$  πάνω από τον  $Y$ .
- Την Ανεξαρτησία των Αδιάφορων Εναλλακτικών Επιλογών: Αν ένα σύνολο από ψηφοδέλτια προτιμήσεων οδηγεί σε μια συνολική κατάταξη του υποψηφίου  $X$  πάνω από τον υποψήφιο  $Y$ , και αν κάποια από τα ψηφοδέλτια μεταβληθούν ούτως ώστε να μην αλλάζει σε αυτά η σχετική κατάταξη των  $X$  και  $Y$ , τότε το εκλογικό σύστημα πρέπει να εξακολουθεί να κατατάσσει τον  $X$  πάνω από τον  $Y$ .
- Την Κυριαρχία του Πολίτη: Κάθε πιθανή κατάταξη των υποψηφίων μπορεί να επιτευχθεί από ένα σύνολο ατομικών ψηφοδελτίων προτιμήσεων.
- Την Μή Επιβολή Δικτατορίας: Δεν πρέπει να υπάρχει ένας συγκεκριμένος ψηφοφόρος του οποίου το ατομικό ψηφοδέλτιο επιλογών να υιοθετείται πάντοτε ως αποτέλεσμα της ψηφοφορίας.

Ο Αιτωκ κατέληξε ότι δεν είναι δυνατόν να βρεθεί εκλογικό σύστημα με όλες αυτές τις ιδιότητες, τις οποίες απέδειξε ως αμοιβαία αντιφατικές.

Τα παραπάνω καθιστούν μάλλον σαφές το πόσο σχετική είναι η έννοια της δικαιοσύνης και των όποιων τιθέμενων και υποτίθεται "κοινά αποδεκτών" κριτηρίων δικαιοσύνης όσον αφορά τις ψηφοφορίες, καθώς και το πόσο σημαντικό ρόλο μπορεί να παίξει η επιλογή ενός εκλογικού συστήματος για τη διαμόρφωση του τελικού εκλογικού αποτελέσματος [80]. Τα θέματα αυτά θα αναλυθούν εκτενώς στις επόμενες ενότητες.

## 2.3 Κριτήρια Δικαιοσύνης

Στην πορεία του χρόνου, πολλοί ερευνητές της Θεωρίας Ψηφοφοριών έχουν προτείνει μια σειρά από κριτήρια τα οποία "κατά κοινή παραδοχή" θα έπρεπε να ικανοποιεί μια "δίκαιη" ψηφοφορία. Τα κριτήρια αυτά περιγράφονται με τον όρο "Κριτήρια Δικαιοσύνης" ("Fairness Criteria"). Πολλοί ήταν οι ερευνητές που ως το 1951 που δημοσιεύτηκε το Θεώρημα του Ανέφικτου του Αιτωκ προσπάθησαν να βρουν ένα εκλογικό σύστημα που να ικανοποιεί κάποιο λογικό σύνολο από Κριτήρια Δικαιοσύνης. Το Θεώρημα του Ανέφικτου

κατέστησε προφανές ότι όλες αυτές οι προσπάθειες ήταν μάταιες, καθώς απέδειξε ότι πάντα μπορεί να κατασκευαστεί εκλογικό σύστημα που να μην ικανοποιεί κάποιο από τα κριτήρια ενός οποιουδήποτε συνόλου Κριτηρίων Δικαιοσύνης, όπως και για κάθε εκλογικό σύστημα μπορεί να κατασκευαστεί ένα σύνολο από μη ικανοποιήσιμα από αυτό Κριτήρια Δικαιοσύνης.

Παρ' όλ' αυτά, η Θεωρία Ψηφοφοριών αναγνωρίζει ως αρκετά σημαντικά για βάση στην κατασκευή εκλογικών συστημάτων τα παρακάτω Κριτήρια Δικαιοσύνης [3][16][82]:

- **Το Κριτήριο της Απόλυτης Πλειοψηφίας (The Majority Criterion)** : Αν κάποιος υποψήφιος λάβει την απόλυτη πλειοψηφία των ψήφων, πρέπει να ανακηρυχθεί νικητής.
- **Το Κριτήριο του Condorcet (από τον Marquis de Condorcet, The Condorcet Criterion)** : Αν κάποιος υποψήφιος υπερισχύσει σε όλες τις ανά ζεύγη εκλογικές του αναμετρήσεις με τους συνυποψήφιους του, πρέπει να ανακηρυχθεί νικητής.
- **Το Κριτήριο της Μονοτονίας (The Monotonicity Criterion)** : Αν γίνει μια ψηφοφορία και ανακηρυχθεί νικητής, τότε αυτός πρέπει να παραμείνει νικητής και σε κάθε πιθανή επαναληπτική ψηφοφορία κατά την οποία όλες οι αλλαγές στις προτιμήσεις γίνονται υπέρ αυτού. Με άλλα λόγια, ένα εκλογικό σύστημα είναι μονότονο αν, στην περίπτωση που κάποιος ψηφοφόρος αυξήσει την σειρά προτίμησης του για τον νικητή υποψήφιο, αυτός θα ανακηρυχθεί και πάλι νικητής, όπως επίσης στην περίπτωση που κάποιος ψηφοφόρος μειώσει την σειρά προτίμησης του για κάποιον ηττημένο υποψήφιο, αυτός θα παραμείνει ηττημένος. Όλα τα εκλογικά συστήματα που "ανά γύρο" διαγράφουν υποψηφιότητες δεν εξασφαλίζουν την ικανοποίηση του Κριτηρίου της Μονοτονίας.
- **Το Κριτήριο της Ανεξαρτησίας των Αδιάφορων Εναλλακτικών Επιλογών (The Independence of Irrelevant Alternatives Criterion)** : Αν γίνει μια ψηφοφορία και ανακηρυχθεί νικητής, τότε αυτός πρέπει να παραμείνει νικητής και σε κάθε πιθανή επανακαταμέτρηση των ψήφων, που μπορεί να γίνει λόγω της αποχώρησης ενός ή περισσότερων ηττημένων υποψηφίων.

Στο Παράρτημα III αναφέρονται παραδείγματα παραβίασης των προαναφερθέντων διάσημων Κριτηρίων Δικαιοσύνης από ευρύτατα χρησιμοποιούμενα εκλογικά συστήματα.

## 2.4 Εκλογικά Συστήματα

Οι ερευνητές της Θεωρίας Ψηφοφοριών έχουν κατασκευάσει πληθώρα εκλογικών συστημάτων (voting systems /voting aggregation methods) και έχουν προσπαθήσει να καθορίσουν τα πλέον κατάλληλα για διαφορετικές εκλογικές περιπτώσεις. Παρόλο που

υπάρχει κάποια σύγκλιση απόψεων ως προς τα χαρακτηριστικά που είναι επιθυμητά σε ένα εκλογικό σύστημα, υπάρχει έντονη διαφωνία για το ποιά χαρακτηριστικά είναι τα περισσότερο σημαντικά. Επιπλέον, η επιλογή εκλογικού συστήματος στον πραγματικό κόσμο εξαρτάται τις περισσότερες φορές από το ποιές είναι οι πολιτικές συνθήκες παρά από τις συμβουλές των θεωρητικών. Έτσι, η "δημοτικότητα" ενός εκλογικού συστήματος δεν αποτελεί ένδειξη του κατά πόσο είναι "δίκαιο" (στον βαθμό που μπορεί να γίνεται λόγος για "δίκαια εκλογικά συστήματα") [33].

Τα χαρακτηριστικά ενός εκλογικού συστήματος το κατατάσσουν σε μια ή περισσότερες από τις υπάρχουσες κατηγορίες εκλογικών συστημάτων. Στη διεθνή βιβλιογραφία διακρίνονται - με διάφορα κριτήρια - αρκετές τέτοιες κατηγορίες [16][52][44][45][82]:

- Πλειοψηφικά συστήματα (majoritarian systems). Είναι συστήματα που ικανοποιούν το Κριτήριο της Απόλυτης Πλειοψηφίας. Στην κατηγορία αυτή ανήκει το σύστημα απόλυτης πλειοψηφίας, που ανακηρύσσει νικητή μόνο αν αυτός συγκεντρώνει την απόλυτη πλειοψηφία των ψήφων, καθώς και διάφορα συστήματα σχετικής πλειοψηφίας. Επίσης, κατ' επέκταση, στην κατηγορία αυτή εντάσσονται και τα συστήματα που αποσκοπούν στην εκλογή του υποψηφίου που μπορεί να κερδίσει καθένα από τους υπόλοιπους σε "προσωπική" αναμέτρηση (νικητής κατά το Condorcet).
- Συστήματα αναλογικής εκπροσώπησης (proportional representation systems). Στην κατηγορία αυτή ανήκουν πολλά ευρέως χρησιμοποιούμενα συστήματα για εκλογή αντιπροσώπων, τα οποία στηρίζονται στην αναλογική εκπροσώπηση των ψηφοφόρων σε απόλυτο ή σχετικό βαθμό.
- Συστήματα προτιμήσεων ή έκφρασης θέσης (preferential systems /positional systems). Στα συστήματα αυτά χρησιμοποιούνται ψηφοδέλτια προτιμήσεων για την κατάταξη των επιλογών των ψηφοφόρων. Οι πληροφορίες που καταγράφονται στα ψηφοδέλτια προτιμήσεων αξιοποιούνται σε διαφορετικό βαθμό από τα διάφορα εκλογικά συστήματα προτιμήσεων (άλλα τις αξιοποιούν όλες κι άλλα ένα μέρος τους, άλλα τις αξιοποιούν σταδιακά και άλλα όχι).
- Συστήματα διαδοχικών αποκλεισμών υποψηφίων (candidates' elimination systems). Τα συστήματα αυτά ανακηρύσσουν νικητή μετά από διαδοχικούς αποκλεισμούς υποψηφίων, οι οποίοι βασίζονται σε αποτελέσματα διαδοχικών εκλογικών γύρων, ή στις εκφρασμένες σε ψηφοδέλτια προτιμήσεων επιλογές των ψηφοφόρων σε ένα μόνο γύρο.
- Συστήματα εκτίμησης ωφέλειας του υποψηφίου (utilitarian systems). Στα συστήματα αυτά ο ψηφοφόρος δεν κατατάσσει απλά τους υποψηφίους σε σειρά προτίμησης, αλλά επίσης τους χαρακτηρίζει με ένα (αυθαίρετο ή προερχόμενο από προκαθορισμένη κλίμακα)

αριθμητικό βάρος που προσδιορίζει τον βαθμό εκτίμησης προς τον υποψηφίου (άρα και την αναμενόμενη ωφέλεια από αυτόν).

- Συστήματα βεβαρημένης άποψης ψηφοφόρων (weighted systems). Στα συστήματα αυτά η άποψη συγκεκριμένων ψηφοφόρων αξιολογείται με διαφορετικό βάρος από αυτή των υπόλοιπων.

*Η επιλογή του εκλογικού συστήματος μπορεί να επηρεάσει σε πολύ μεγάλο βαθμό το τελικό αποτέλεσμα μιας εκλογικής αναμέτρησης [16]. Μπορεί επίσης να επηρεάσει την ικανότητα των εκλογικών αναλυτών να ερμηνεύσουν τα εκλογικά αποτελέσματα, και συνακόλουθα την ικανότητα των εκλεγμένων αντιπροσώπων να κατανοήσουν τις πραγματικές επιθυμίες των ψηφοφόρων τους. Αυτό οφείλεται στο γεγονός ότι τα διάφορα εκλογικά συστήματα απαιτούν από τους ψηφοφόρους να προμηθεύσουν κυμαινόμενες ποσότητες πληροφοριών για τις προτιμήσεις τους, και στο γεγονός πως υπάρχουν αρκετά ευρέως χρησιμοποιούμενα εκλογικά συστήματα που τείνουν να ενθαρρύνουν τους ψηφοφόρους να μην εκφράσουν τις ειλικρινείς τους επιθυμίες [16].*

## 2.5 Συλλογή Ψηφοδελτίων

Ο τρόπος με τον οποίο τα ψηφοδέλτια συλλέγονται είναι πολλές φορές το ίδιο σημαντικό με το εκλογικό σύστημα που χρησιμοποιείται κατά την ψηφοφορία. Το σύστημα που χρησιμοποιείται για τη συλλογή των ψηφοδελτίων μπορεί να επηρεάσει το συνολικό αποτέλεσμα της ψηφοφορίας, την προσβασιμότητα των ψηφοφόρων στα αποτελέσματα, την πιθανότητα νοθείας, το οικονομικό κόστος της ψηφοφορίας, το ποσό της σκέψης και του χρόνου που δαπανούν οι ψηφοφόροι για να κάνουν τις επιλογές τους και την εμπιστοσύνη που τρέφει το εκλογικό σώμα για την ακρίβεια του εκλογικού αποτελέσματος [16].

Υπάρχουν πολλές ιδιότητες που θεωρούνται επιθυμητές για ένα σύστημα συλλογής ψηφοδελτίων [4][20][43][55][57]:

- **Ακρίβεια (Accuracy).** Ένα σύστημα συλλογής είναι *ακριβές* αν (1) δεν είναι δυνατή η αλλοίωση της ψήφου, (2) δεν είναι δυνατή η μή συμπερίληψη ενός εγκύρου ψηφοδελτίου στον τελικό υπολογισμό του αποτελέσματος και (3) δεν είναι δυνατή η συμπερίληψη ενός άκυρου ψηφοδελτίου στο τελικό αποτέλεσμα. Σε ένα *εντελώς ακριβές* σύστημα, όσες πιθανές ανακρίβειες μπορεί να εμφανιστούν κατά τη συλλογή ψηφοδελτίων, εντοπίζονται και διορθώνονται. Τα *μερικώς ακριβή* συστήματα εντοπίζουν τις ανακρίβειες αλλά δεν τις διορθώνουν. Ο *βαθμός ακρίβειας* του συστήματος μπορεί να μετρηθεί με βάση τον αριθμό των λαθών, την πιθανότητα λάθους ή τον αριθμό των «επίφοβων» σημείων του συστήματος συλλογής στα οποία είναι πιθανή η εμφάνιση λάθους.



- **Ευχρηστία (Convenience).** Ένα σύστημα είναι εύχρηστο αν επιτρέπει στους ψηφοφόρους να ψηφίζουν σύντομα, με μία και μόνη ενέργεια, με ελάχιστο εξοπλισμό και χωρίς ειδικά προσόντα. Η ευχρηστία ενός συστήματος συλλογής είναι κάπως υποκειμενική, και εξαρτάται όχι μόνο από το ίδιο το σύστημα, αλλά και από τον πληθυσμό που το χρησιμοποιεί. Για παράδειγμα, υπάρχουν πληθυσμοί ψηφοφόρων που θεωρούν πιο βολικό να ψηφίζουν σε κεντρικά εκλογικά τμήματα με παραδοσιακούς τρόπους, ενώ άλλοι θεωρούν βολικότερο το να ψηφίζουν μέσω ηλεκτρονικού υπολογιστή από το σπίτι ή το γραφείο τους.
- **Ευελξία (Flexibility).** Ένα σύστημα θεωρείται ευέλικτο αν επιτρέπει την ύπαρξη πληθώρας μορφών ερωτημάτων στα συλλεγόμενα ψηφοδέλτια, συμπεριλαμβανομένων ερωτημάτων «ανοικτής απάντησης», ώστε να είναι δυνατή η «δυναμική» εγγραφή υποψηφίων και η εξαγωγή στατιστικών αποτελεσμάτων.
- **Κινητικότητα (Mobility).** Ένα σύστημα επιδεικνύει κινητικότητα αν δεν υπάρχουν περιορισμοί ως προς το μέρος από το οποίο ένας ψηφοφόρος μπορεί να ψηφίσει. Τα συστήματα που απαιτούν την ψηφοφορία σε προκαθορισμένα εκλογικά κέντρα δεν διαθέτουν κινητικότητα, εκτός κι αν έχει ληφθεί πρόνοια για την έμμεση συλλογή των απόψεων των απόντων ψηφοφόρων.
- **Μυστικότητα (Privacy).** Ένα σύστημα συλλογής είναι «μυστικό» αν (1) ούτε οι εκλογικές αρχές ούτε κανείς άλλος μπορούν να συσχετίσουν ένα ψηφοδέλτιο με τον ψηφοφόρο που το χρησιμοποίησε και (2) κανείς ψηφοφόρος δεν μπορεί να αποδείξει ότι ψήφισε με ένα συγκεκριμένο τρόπο. Έτσι δυσχεραίνονται οι εκλογικοί εκβιασμοί και η εξαγορά ψήφων.
- **Επαληθευσιμότητα (Verifiability).** Ένα σύστημα συλλογής θεωρείται «επαληθεύσιμο» αν ο κάθε εμπλεκόμενος στην ψηφοφορία μπορεί να επαληθεύσει πως όλοι οι ψήφοι έχουν καταμετρηθεί σωστά. Ένας ελαστικότερος ορισμός της επαληθευσιμότητας δέχεται ως επαληθεύσιμο το σύστημα εκείνο που επιτρέπει στους ψηφοφόρους να επαληθεύσουν τις δικές τους ψήφους και ίσως να διορθώσουν πιθανά λάθη χωρίς να διακινδυνεύσουν την ιδιωτικότητα της ψήφου. Τα παραδοσιακά συστήματα συλλογής επιτρέπουν μόνο την μερική επαλήθευση του αποτελέσματος (μέσω της παρακολούθησης της διαδικασίας) από κομματικούς αντιπροσώπους.
- **Μη τρωότητα (Invulnerability).** Ένα σύστημα θεωρείται «άτρωτο» αν (1) επιτρέπει μόνο στους εγγεγραμμένους ψηφοφόρους να ψηφίσουν και (2) εξασφαλίζει ότι κάθε εγγεγραμμένος ψηφοφόρος ψηφίζει μόνο μια φορά.

## 2.6 Ηλεκτρονικές Ψηφοφορίες

Τη συλλογή ψηφοδελτίων με χρήση ηλεκτρονικών μέσων μετάδοσης πληροφορίας την οραματίστηκε ίσως πρώτος το 1940 ο R. Buckminster Fuller, όταν πρότεινε την διεξαγωγή "ηλεκτρισμένων ψηφοφοριών" ("electrified voting") [8]. Οι πρώτες προσπάθειες για διεξαγωγή ηλεκτρονικών ψηφοφοριών έγιναν τη δεκαετία του '70, με χρήση τηλεφωνικών δικτύων. Αργότερα, και ιδίως στην τρέχουσα δεκαετία, οι σχετικές προσπάθειες χρησιμοποιούν πια τα υπάρχοντα δίκτυα ηλεκτρονικών υπολογιστών.

Οι ηλεκτρονικές ψηφοφορίες μπορούν να εγγυηθούν αρκετές από τις επιθυμητές ιδιότητες των συστημάτων συλλογής ψηφοδελτίων, ειδικά αυτές της ευχρηστίας και της ευελιξίας. Όμως η ασφάλεια και η μυστικότητα κατά την ψηφοφορία, όπως και τα προβλήματα της επαληθευσσιμότητας της ψήφου και της επαληθευσσιμότητας της ορθής καταμέτρησης παραμένουν ανοικτά θέματα έρευνας, παρά τις προσπάθειες ανάπτυξης σχετικών κρυπτογραφικών πρωτοκόλλων [4][55][57][10][30][54].

Παρά τα προβλήματα που ενυπάρχουν στις ηλεκτρονικές ψηφοφορίες, η χρήση τους αρχίζει να επεκτείνεται αρκετά. Οι ηλεκτρονικές ψηφοφορίες είναι δημοφιλείς ειδικά στα πανεπιστήμια, όπου τα μέλη της ακαδημαϊκής κοινότητας έχουν εύκολη πρόσβαση σε ηλεκτρονικούς υπολογιστές, και όπου το κόστος διεξαγωγής ηλεκτρονικών ψηφοφοριών είναι μικρό. Επίσης, οι ηλεκτρονικές ψηφοφορίες είναι εξαιρετικά δημοφιλείς στον Παγκόσμιο Ιστό, όπου πληθώρα ερευνών και δημοσκοποήσεων διεξάγονται χωρίς ιδιαίτερο προβληματισμό για θέματα ασφάλειας και μυστικότητας. Οι υποστηρικτές της άμεσης δημοκρατίας βλέπουν στις ηλεκτρονικές ψηφοφορίες τη μελλοντική λύση στα πρακτικά προβλήματα που εμποδίζουν τη συχνή διεξαγωγή δημοψηφισμάτων.

Αν είναι καλά σχεδιασμένα, τα συστήματα ηλεκτρονικών ψηφοφοριών μπορεί να είναι εφαρμόσιμα σε πολλές μορφές εκλογών και δημοσκοπήσεων, ακόμα και σε εθνικές εκλογές. Ήδη πολιτικά κόμματα και οργανισμοί έχουν αρχίσει να διεξάγουν έρευνες και εκλογές ηλεκτρονικά. Βέβαια, αυτό που είναι απαραίτητο για να προχωρήσει η εφαρμογή μεγάλης κλίμακας ηλεκτρονικών ψηφοφοριών με αξιόπιστο τρόπο, είναι η παραπέρα πρόοδος σε θέματα κρυπτογραφίας και η περαιτέρω βελτίωση του λογισμικού που χρησιμοποιείται για ηλεκτρονική αλληλογραφία και για πλοήγηση στον Παγκόσμιο Ιστό [16].

## 2.7 Χρήση Μηχανισμών Ψηφοφοριών στην Επιστήμη Υπολογιστών

Εκτός από τις εργασίες που σχετίζονται με την κρυπτογράφηση και την ασφάλεια των δεδομένων κατά τη διενέργεια ψηφοφοριών, στο χώρο της Επιστήμης Υπολογιστών

υπάρχουν αρκετές περιπτώσεις χρήσης ψηφοφοριών για την αντιμετώπιση προβλημάτων που απαιτούν λήψεις αποφάσεων βασισμένων σε καταναμημένα δεδομένα, κυρίως στους τομείς των Καταναμημένων Συστημάτων και των Βάσεων Δεδομένων, αλλά τελευταία και στην Ρομποτική και τη Μηχανική Όραση.

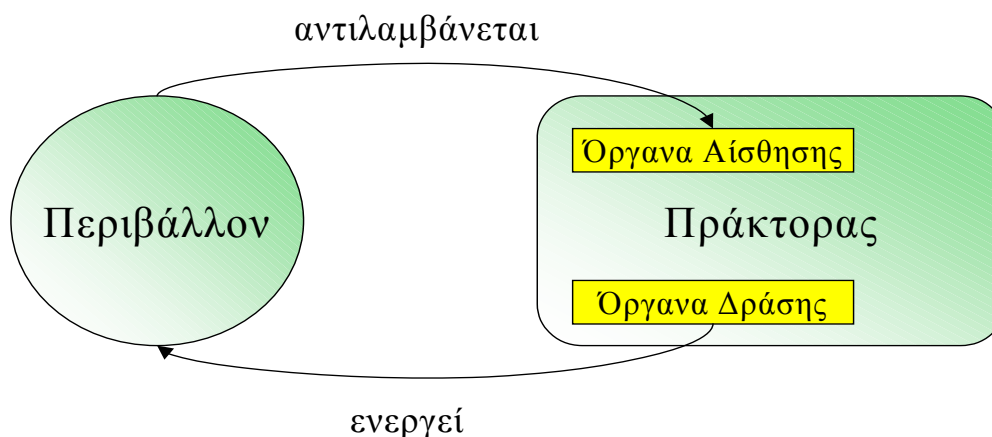
Πρώτος ο Gifford πρότεινε το 1978 ένα πρωτόκολλο βασισμένο σε ψηφοφορία, για να επιλύσει το πρόβλημα της ακεραιότητας κατά την ταυτόχρονη προσπέλαση δεδομένων από διαφορετικούς χρήστες [24]. Το πρωτόκολλο ονομάστηκε "Weighted Voting" ("Σταθμισμένη Ψηφοφορία"), και προβλέπει τη συλλογή ενός πλειοψηφικού ποσοστού "ψήφων" με διαφορετικά βάρη, παρεχόμενων από αντίγραφα αρχείων ανά κόμβο δικτύου, προκειμένου να επιτραπεί η ανάγνωση (δημιουργία "read quorum") ή η εγγραφή (δημιουργία "write quorum") δεδομένων στο αρχείο. Άλλες προσεγγίσεις, όπως το πρωτόκολλο "Δυναμικής Ψηφοφορίας" ("Dynamic Voting") [12], το πρωτόκολλο "Γραμμικής Δυναμικής Ψηφοφορίας" ("Linear Dynamic Voting") [35] ή βελτιώσεις τους [34][31], διαμορφώνουν τα χρησιμοποιούμενα βάρη και ποσοστά πλειοψηφίας δυναμικά, προκειμένου να επιτύχουν μεγαλύτερη διαθεσιμότητα των δεδομένων, ακόμα και σε περιπτώσεις κατάτμησης του δικτύου (network partitioning).

Στους τομείς της Ρομποτικής και της Μηχανικής Όρασης, πρόσφατες εργασίες από τον Paolo Pirjanian αφορούν την επιλογή και τον έλεγχο καταναμημένων συμπεριφορών με χρήση σχημάτων ψηφοφορίας - αλλά και τη δυναμική δημιουργία αξιόπιστων συμπεριφορών. Οι ψηφοφορίες οδηγούν στην επιλογή συμπεριφορών και ενεργειών που αποδεικνύεται ότι έχουν υψηλή πιθανότητα να είναι οι ορθές. Σχετικά πειράματα αφορούν την πλοήγηση ρομπότ και την ομαλή παρακολούθηση κίνησης αντικειμένων με κάμερα [49] [50]. Τέλος, ο Carsten Brautigam παρουσιάζει στη διδακτορική του διατριβή το πώς μπορεί να χρησιμοποιηθεί ένα σχήμα ψηφοφορίας προκειμένου να επιτευχθεί η ανίχνευση επίπεδων επιφανειών σε εικόνες, με βάση κάποιες αρχικές εκτιμήσεις (οι οποίες και "συμψηφίζονται") [6].

## Κεφάλαιο 3 Πράκτορες

Η έννοια του "πράκτορα" είναι γνωστή από την καθημερινή ζωή: περιγράφει κάποιον "ενδιάμεσο", ένα άτομο που ενεργεί εκ μέρους ενός άλλου ατόμου, έναν αντιπρόσωπο που προσπαθεί να εκπληρώσει ένα σύνολο στόχων, ενδεχόμενα μέσα σε δυναμικά μεταβαλλόμενες συνθήκες. Στην προσπάθειά του αυτή ένας πράκτορας αναγκάζεται να αντιδρά στις μεταβολές του περιβάλλοντος μέσα στο οποίο ενεργεί (re-activeness of the agent), ή ακόμα να προνοεί ώστε ο ίδιος να ενεργήσει προκειμένου να προσαρμόσει τις συνθήκες του περιβάλλοντος ώστε να εξυπηρετηθούν οι σκοποί του (pro-activeness of the agent). Στην πορεία υλοποίησης των στόχων του, ένας πράκτορας πιθανότατα αξιοποιεί τις όποιες δυνατότητες μάθησης και συνεργασίας διαθέτει.

Στο χώρο της Επιστήμης Υπολογιστών, η έννοια του "πράκτορα" χρησιμοποιείται ευρύτατα για να περιγράψει μια υπολογιστική οντότητα ή ένα ρομπότ, που προσπαθεί να ικανοποιήσει ένα σύνολο στόχων μέσα σε ένα πολύπλοκο και δυναμικό περιβάλλον [53]. Ο όρος προέρχεται από το χώρο της Τεχνητής Νοημοσύνης, και την προσπάθεια κατασκευής τεχνητών όντων που να μιμούνται ανθρώπινες ικανότητες [53][7]. Κατ' αναλογία με τα ανθρώπινα όργανα αίσθησης και δράσης, ο πράκτορας διαθέτει αισθητήρες (sensors) για την αντίληψη του περιβάλλοντος, και όργανα δράσης (actuators) για να ενεργεί μέσα σε αυτό (Σχήμα 1)[53]. Οι στόχοι του πράκτορα μπορεί να είναι από εξαιρετικά απλοί ως εξαιρετικά πολύπλοκοι, να είναι στόχοι ενδιάμεσοι που να εξυπηρετούν ένα τελικό στόχο, ή να είναι εσωτερικές ανάγκες ή κίνητρα που πρέπει να κρατηθούν εντός συγκεκριμένων ορίων [67][39]. Για να επιτύχει τους στόχους του, μεταβάλλοντας πιθανόν τις συνθήκες του περιβάλλοντός του, ο πράκτορας πρέπει να εκτελέσει συμπερασματικές διαδικασίες (*reasoning*) και να επιδείξει συμπεριφορές που με τη σειρά τους μπορεί να είναι από στερεότερες ως εξαιρετικά πολύπλοκες [25][26].



• Σχήμα 1 Αλληλεπίδραση πράκτορα με το περιβάλλον του.

### 3.1 Χαρακτηριστικά των Πρακτόρων

Η έννοια του πράκτορα, όπως παρουσιάστηκε παραπάνω, είναι αρκετά γενική και σχετικά ασαφής. Πράγματι, δεν υπάρχει ως τώρα στη διεθνή βιβλιογραφία της Επιστήμης Υπολογιστών ένας αυστηρός ορισμός της έννοιας "πράκτορας". Οι διάφοροι συγγραφείς χρησιμοποιούν τον όρο δίνοντας βάρος σε διαφορετικές πτυχές της έννοιας ανάλογα με το πρόβλημα που προσπαθούν να επιλύσουν, ομιλώντας για "έξυπνους πράκτορες", για "αυτόνομους πράκτορες", γενικότερα για "υπολογιστικούς πράκτορες" ή για συνδυασμούς των όρων αυτών. Άλλοι πράκτορες σχεδιάζονται να δουλεύουν μόνοι τους, άλλοι να συνεργάζονται μέσα σε ομάδες ή κοινωνίες, άλλοι έχουν τη δυνατότητα να μαθαίνουν, άλλοι είναι κινητοί και άλλοι όχι. Παρ' όλ' αυτά, είναι δυνατόν να παρατεθεί ένα σύνολο από κύρια χαρακτηριστικά που επιτρέπουν τον προσδιορισμό μιας οντότητας ως πράκτορα [39][25][27][62][32]:

- **Αυτονομία (autonomy)**, δηλαδή το να έχει η ίδια οντότητα έλεγχο επάνω στις ενέργειες της και στην εσωτερική της κατάσταση.
- **Προνοητικότητα ή προενεργητικότητα (pro-activeness)**, η ικανότητα δηλαδή της οντότητας να λαμβάνει πρωτοβουλίες για ενέργειες τέτοιες που να βοηθούν στην πραγμάτωση των σκοπών της.
- **Αποκρισιμότητα (responsiveness) ή αντίδραση (re-activeness)**, δηλαδή η ικανότητα γρήγορης και «σωστής» απόκρισης στις αλλαγές του περιβάλλοντός της, όπως το αντιλαμβάνεται η ίδια η οντότητα («αντίληψη περιβάλλοντος»).

- **Κοινωνική ικανότητα (social ability)**, δηλαδή η ικανότητα κοινωνικής συμπεριφοράς κατά την αλληλεπίδραση με άλλους πράκτορες και ανθρώπους. Η επικοινωνία μεταξύ των πρακτόρων μπορεί να γίνεται με χρήση ειδικής γλώσσας επικοινωνίας πρακτόρων.

Πέρα από αυτά τα κύρια χαρακτηριστικά, δευτερεύουσες ιδιότητες που συχνά χαρακτηρίζουν τους πράκτορες είναι η *ειλικρίνεια*, δηλαδή η μή "συνειδητή" παραπληροφόρηση άλλων, η *καλή προαίρεση*, δηλαδή η μή ύπαρξη αλληλοσυγκρουόμενων στόχων, και η *λογική*, δηλαδή η εκτέλεση μόνο ενεργειών που φαίνεται ότι θα συμβάλλουν στην επίτευξη των στόχων (ή τουλάχιστον δε φαίνεται να την αποτρέπουν). Επίσης, άλλη ιδιότητα που συχνά χαρακτηρίζει πράκτορες είναι η *κινητικότητα*, δηλαδή η δυνατότητα να μεταφέρονται μεταξύ συστημάτων (να "μετοικούν"), ώστε να εκμεταλλεύονται απομακρυσμένους πόρους ή να συνεργάζονται με άλλους πράκτορες.<sup>6</sup>

### 3.2 Αρχιτεκτονικές Πρακτόρων

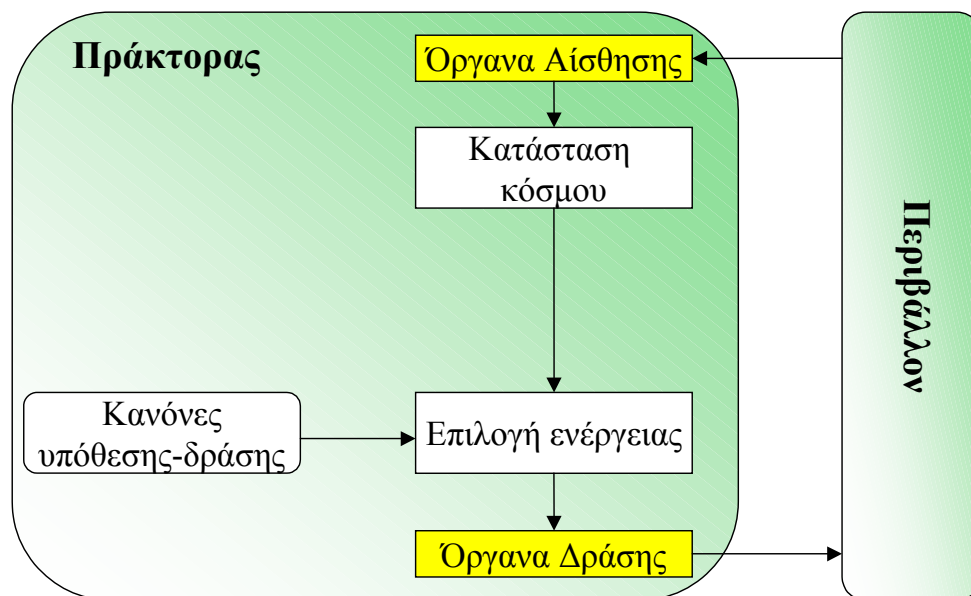
Η αρχιτεκτονική ενός πράκτορα, καθορίζει τα συστατικά του στοιχεία και τις αλληλεπιδράσεις μεταξύ τους. Με τη σειρά τους, τα συστατικά στοιχεία του πράκτορα και οι αλληλεπιδράσεις τους καθορίζουν τις ενέργειες του πράκτορα και τη μελλοντική του κατάσταση, σε σχέση με το ποιά είναι τα δεδομένα που προσλαμβάνονται στα αισθητήρια όργανα του πράκτορα και το ποιά είναι η τωρινή του εσωτερική κατάσταση.

Η συνηθισμένη γενική μοντελοποίηση της συμπεριφοράς ενός πράκτορα ως μια συνεχή ακολουθία λειτουργιών αντίληψης, νοητικής διεργασίας και δράσης, αποτελεί βασικό χαρακτηριστικό σχεδιασμού όσον αφορά την αρχιτεκτονική του. Έτσι, ένα γενικό και ευρέως χρησιμοποιούμενο αρχιτεκτονικό μοντέλο ενός πράκτορα υπαγορεύει οι λειτουργίες αυτές να πραγματοποιούνται από αντίστοιχα τμήματα λογισμικού, τα οποία μπορούν να χαρακτηριστούν ως αισθητήρες, νοητικό ή γνωσιακό τμήμα, και όργανα δράσης (perception – cognition – action) [25][26]. Το γνωσιακό τμήμα είναι αυτό το οποίο υλοποιεί τις συμπερασματικές διαδικασίες, δηλαδή αυτό που ονομάζουμε «νόηση» της οντότητας, επηρεάζοντας την επιλογή των προς εκτέλεση ενεργειών, διαμορφώνοντας "σχέδια δράσης", αντιλαμβανόμενο τη σημασιολογία των γεγονότων. Τέλος, είναι επίσης το τμήμα που είναι υπεύθυνο για τις - όποιες - δυνατότητες μάθησης του πράκτορα.

Με βάση αυτό το γενικό αρχιτεκτονικό μοντέλο, και ανάλογα με τη δομή και την πολυπλοκότητα του γνωσιακού τμήματος, μπορούν να διακριθούν οι παρακάτω κατηγορίες αρχιτεκτονικών πρακτόρων [53][67]:

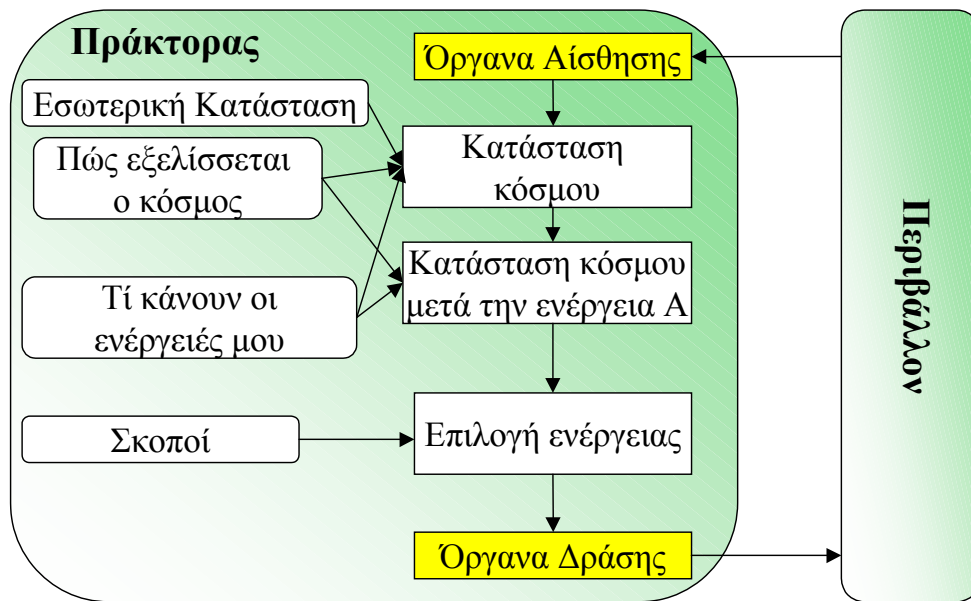
<sup>6</sup> Είναι ευρύτατα διαδεδομένη η απόδοση του όρου "πράκτορας" σε κινητά ή ενεργά μετοικούντα αντικείμενα. Είναι προφανές από τα αναφερόμενα σε αυτό το κεφάλαιο ότι η άποψη αυτή σε καμμία περίπτωση δεν συνιστά πλήρη επιστημονικό ορισμό, και εσφαλμένα περιορίζει την έννοια "πράκτορας".

- **Αρχιτεκτονικές απλής αντίδρασης (reactive / simple reflexive)** : Στις αρχιτεκτονικές αυτές, δεν υπάρχει κάποιο εσωτερικό συμβολικό μοντέλο του κόσμου, και οι αποφάσεις δράσης λαμβάνονται με απλούς κανόνες υπόθεσης-δράσης (condition-action rules). Ο πράκτορας ουσιαστικά αντιδρά απλώς στα δεδομένα που προσλαμβάνονται στους αισθητήρες (Σχήμα 2).

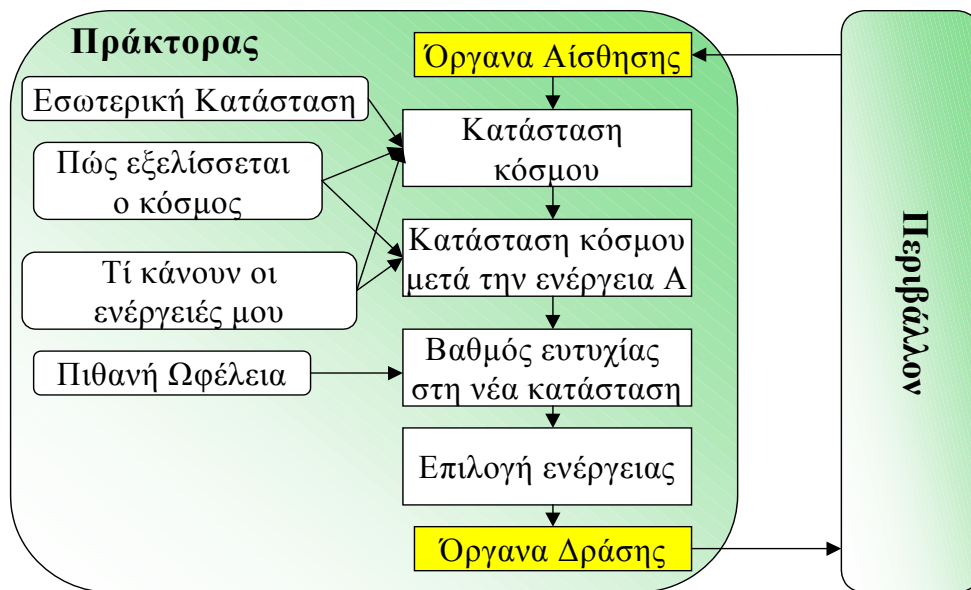


• Σχήμα 2 Πράκτορας με Αρχιτεκτονική Απλής Αντίδρασης.

- **Αρχιτεκτονικές "προκαθορισμού" (deliberative)** : Στις αρχιτεκτονικές αυτές ο πράκτορας διαθέτει ένα απόλυτα καθορισμένο συμβολικό μοντέλο του κόσμου, και οι αποφάσεις δράσης λαμβάνονται μετά από συλλογισμούς που βασίζονται στο συνδυασμό υπάρχοντων προτύπων, των δεδομένων των αισθητήρων και το μετασχηματισμό συμβόλων. Οι αρχιτεκτονικές αυτές συνήθως διακρίνονται σε "αρχιτεκτονικές σκοπών" ("goal-based", Σχήμα 3) όπου καθοριστικό ρόλο στην επιλογή ενέργειας παίζουν οι υπάρχοντες σκοποί του πράκτορα, και "αρχιτεκτονικές ωφέλειας" ("utility-based", Σχήμα 4), όπου καθοριστικό ρόλο στην επιλογή ενέργειας παίζει η "ωφέλεια" που ελπίζεται ότι θα προκύψει από την κάθε ενέργεια προς επιλογή (ο βαθμός "ευτυχίας" ή "ικανοποίησης" του πράκτορα στην νέα κατάσταση όπου θα βρεθεί).
- **Υβριδικές αρχιτεκτονικές** : Αρχιτεκτονικές με στοιχεία και από τις δύο προαναφερθείσες κατηγορίες.



• Σχήμα 3 Πράκτορας με Αρχιτεκτονική Σκοπών.



• Σχήμα 4 Πράκτορας με Αρχιτεκτονική Ωφέλειας.

### 3.3 Συστήματα Πολλών Πρακτόρων

Σε αντίθεση με τα συστήματα που χρησιμοποιούν ένα πράκτορα (Single Agent Systems - SAS) ο οποίος μπορεί να επικοινωνεί με ένα χρήστη ή με τοπικούς και απομακρυσμένους πόρους αλλά όχι με άλλους πράκτορες, τα συστήματα πολλών πρακτόρων (Multi-Agent Systems - MAS) περιλαμβάνουν πολλούς πράκτορες που επικοινωνούν μεταξύ τους. Η χρήση MAS μπορεί να είναι προτιμητέα για μια σειρά από λόγους [61]:



- τη βελτίωση ταχύτητας του συστήματος, λόγω παράλληλης εκτέλεσης τμημάτων μιας εργασίας από διαφορετικούς πράκτορες
- τη βελτίωση της σταθερότητας και της ασφάλειας του συστήματος, λόγω διαμοιρασμού εργασιών και μή ύπαρξης μοναδικού σημείου κατάρρευσης του συστήματος
- τη δυνατότητα επεκτασιμότητας και αναβάθμισης του συστήματος, απλά με την προσθήκη νέων πρακτόρων

Οι πράκτορες που συμμετέχουν σε MAS μπορεί να εξυπηρετούν κοινούς στόχους του συστήματος, ή να μην έχουν κοινούς στόχους, αλλά σε κάθε περίπτωση συνεργάζονται μέσα στην ομάδα ώστε να τους επιτύχουν. Στην περίπτωση μή ύπαρξης κοινών στόχων, γίνεται συνήθως αναφορά σε συνασπισμούς (coalitions) πρακτόρων, και σε πράκτορες που πολλές φορές είναι ανταγωνιστικοί, δηλαδή προσπαθούν να μεγιστοποιήσουν τα δικά τους οφέλη από τη συμμετοχή στο συνασπισμό. Υπάρχουν περιπτώσεις αρχιτεκτονικών MAS ή αρχιτεκτονικών πρακτόρων που μετέχουν σε MAS, οι οποίες επιτρέπουν την προσωρινή ή οριστική εγκατάλειψη επιμέρους στόχων των πρακτόρων, προκειμένου να επιτευχθεί ένας κοινός στόχος ή οι πλέον μακροπρόθεσμοι επιμέρους στόχοι. Οι πράκτορες που συμμετέχουν σε τέτοιου είδους MAS συνήθως ονομάζονται *συνεργατικοί/ συνεργαζόμενοι* πράκτορες [61][56][40].

Σημαντικό ρόλο στην επιτυχή λειτουργία ενός MAS παίζει ο *διαμοιρασμός της γνώσης* των πρακτόρων σε όλα τα μέλη της ομάδας. Προκειμένου να επιτυγχάνεται ο διαμοιρασμός της γνώσης, είναι απαραίτητη η ύπαρξη συνεπών πρωτοκόλλων συνεννόησης και η σαφής αναπαράσταση των συνολικών και επιμέρους στόχων, προθέσεων και σχεδίων δράσης [11][63]. Επίσης, είναι σημαντική η ύπαρξη *πρωτοκόλλων επίλυσης συγκρούσεων* (οι οποίες πιθανόν προκύπτουν λόγω των επιμέρους στόχων των πρακτόρων). Η επίλυση συγκρούσεων πολλές φορές γίνεται μέσα από διαπραγματεύσεις, ή ακόμα κι από δημοπρασίες. Τέλος, η επικοινωνία των πρακτόρων γίνεται είτε απευθείας, είτε μέσω διαμεσολαβητή [23].

### 3.4 Τεχνητή Νοημοσύνη και Συστήματα Πρακτόρων

Ο όρος "πράκτορας" προήλθε, όπως προαναφέρθηκε, από το πεδίο της Τεχνητής Νοημοσύνης. Στην πορεία του χρόνου όμως, η ανάλυση, ο σχεδιασμός και η υλοποίηση συστημάτων πρακτόρων διαφοροποιήθηκε απόλυτα από την "παραδοσιακή" προσέγγιση ανάπτυξης συστημάτων Τεχνητής Νοημοσύνης, με την επιρροή και των εξελίξεων στους τομείς των Πληροφοριακών Συστημάτων, των Κατανεμημένων Συστημάτων και της Μηχανικής Μάθησης [41][53]. Η ανάπτυξη συστημάτων κατανεμημένων αυτόνομων

πρακτόρων, που σε μικρό βαθμό απαιτούν την ύπαρξη κεντρικών αναπαραστάσεων γνώσης, είναι αυτή που σήμερα προσπαθεί να ξεπεράσει κυρίως τα προβλήματα επεκτασιμότητας σε μεγάλη κλίμακα που αντιμετώπισε η "παραδοσιακή" Τεχνητή Νοημοσύνη - παρόλο που ο βαθμός επιτυχίας και της τεχνολογίας πρακτόρων είναι από πολλούς αμφισβητήσιμος [1]. Στην ενότητα αυτή θα επιχειρηθεί μια αναφορά των βασικών διαφορών μεταξύ της τεχνολογίας πρακτόρων και της Τεχνητής Νοημοσύνης [69].

Κατ' αρχήν, η Τεχνητή Νοημοσύνη έχει στο παρελθόν κυρίως δώσει έμφαση στην δημιουργία κεντροποιημένων συστημάτων που προσπαθούν να ειδικευτούν σε βάθος σε ένα προχωρημένο πρόβλημα, χρησιμοποιώντας πολύπλοκες γνωσιακές δομές (τα συστήματα αυτά είναι γνωστά με τον όρο expert systems). Αντίθετα, σε ένα σύστημα πολλών πρακτόρων η γνώση και οι ικανότητες διαμοιράζονται, και κάθε πράκτορας είναι επιφορτισμένος με την επίτευξη συγκεκριμένων απλούστερων στόχων και την τέλεση απλών ενεργειών. Η συνεργασία (ή ο ανταγωνισμός) των πρακτόρων είναι που αυξάνει τη "συνολική ευφυΐα" του συστήματος.

Είναι γνωστό ακόμα πως τα παραδοσιακά συστήματα Τεχνητής Νοημοσύνης συνήθως αποτελούν "κλειστές" εφαρμογές, χωρίς άμεση αλληλεπίδραση με το περιβάλλον. Οι αυτόνομοι πράκτορες όμως βρίσκονται σε άμεση και συνεχή αλληλεπίδραση με το περιβάλλον, μέσω των οργάνων αίσθησης και δράσης που διαθέτουν.

Επιπλέον, τα συστήματα Τεχνητής Νοημοσύνης συνήθως απασχολούνται χωρίς χρονικούς περιορισμούς με την επίλυση ενός προβλήματος τη φορά. Αντίθετα, ένα σύστημα πρακτόρων μπορεί να χρειάζεται να αντιμετωπίσει σε πραγματικό χρόνο πάνω από ένα προβλήματα, καλούμενο πιθανότατα να συμβιβάσει επιμέρους στόχους και σχέδια δράσης.

Τέλος, τα παραδοσιακά expert systems συνήθως διαθέτουν στατικές γνωσιακές δομές, ενώ τα συστήματα πρακτόρων διαθέτουν προσαρμόσιμες γνωσιακές δομές, μεταβαλλόμενες λόγω των προσλαμβανόμενων γνώσεων και εμπειριών.

### 3.5 Γλώσσες Επικοινωνίας Πρακτόρων

Το δομικό στοιχείο της αλληλεπίδρασης μεταξύ "έξυπνων" οντοτήτων είναι ο διαμοιρασμός γνώσεων, ο οποίος περιλαμβάνει τόσο την αμοιβαία κατανόηση της γνώσης όσο και την μετάδοση αυτής της γνώσης. Η αλληλεπίδραση μεταξύ πρακτόρων λογισμικού και η διαλειτουργικότητα μεταξύ τους απαιτεί α) μια κοινή γλώσσα β) μια κοινή κατανόηση της ανταλλασόμενης γνώσης και γ) την ικανότητα να ανταλλάσσεται οτιδήποτε εμπεριέχεται στα α) και β) [19].

Μια γλώσσα επικοινωνίας πρακτόρων (Agent Communication Language - ACL) έχει ακριβώς το ρόλο της επίτευξης επικοινωνίας μεταξύ κατανεμημένων και ετερογενών πρακτόρων μέσω αυτής.

Μια συνηθισμένη θεώρηση των πρακτόρων τους βλέπει σαν οντότητες "υψηλού επιπέδου", που χρησιμοποιούν συμβολική αναπαράσταση, έχουν πεποιθήσεις, γνώσεις, επιδεικνύουν εξυπνάδα και επιλέγουν ενέργειες/ διαμορφώνουν σχέδια δράσης μετά από συλλογισμούς. Τέτοιοι πράκτορες, μπορεί να θεωρηθεί πως βρίσκονται στο "γνωσιακό επίπεδο" [42], και έτσι δεν εξυπηρετούνται από συνήθεις γλώσσες και πρωτόκολλα που χρησιμοποιούνται στον κατανεμημένο προγραμματισμό. Ως αποτέλεσμα, μια ACL *πρέπει να είναι αρκετά ισχυρή ώστε να υποστηρίζει επικοινωνία μεταξύ προγραμμάτων που ευρίσκονται σε αυτό το υψηλότερο επίπεδο* - ειδάλλως οι πράκτορες θα είναι υποχρεωμένοι να μεταβιβάζουν πληροφορίες και να μεταφράζουν εκφράσεις συνεχώς μεταξύ του "κατώτερου" επιπέδου/ γλώσσας επικοινωνίας και του ανώτερου δικού τους επιπέδου [19].

Ο ρόλος μιας ACL είναι τόσο σημαντικός που κάποιοι συγγραφείς φτάνουν να περιλαμβάνουν στον ορισμό της έννοιας "πράκτορας" την απαίτηση αυτός να επικοινωνεί με άλλους μέσω μιας ACL [19] [23] [81].

Πρέπει να γίνει σαφές πως μια ACL **δεν αποτελεί ένα πρωτόκολλο**. Μπορεί όμως να **χρησιμοποιεί (ή να χρησιμοποιείται από)** πρωτόκολλα: Μπορεί να χρησιμοποιεί πρωτόκολλα για μεταφορά πληροφοριών (HTTP, FTP κλπ.), ενώ μπορεί να χρησιμοποιείται από πρωτόκολλα αλληλεπίδρασης (δηλαδή "πλαίσια εργασίας" για υψηλού επιπέδου αλληλεπίδραση, π.χ. διαπραγματευτικά πρωτόκολλα ή πρωτόκολλα θεωρίας παιγνίων) ως τρόπος υλοποίησής τους. Ακόμα, μπορεί στην περιγραφή της να περιλαμβάνει *πρωτόκολλα περιορισμών εγκυρότητας ανταλλάξιμης πληροφορίας*. Σε καμμία περίπτωση πάντως δεν αποτελεί η ίδια ένα πρωτόκολλο [19].

Επίσης, πρέπει να γίνει σαφής η διαφορά μεταξύ μιας ACL και μιας "γλώσσας αναπαράστασης γνώσης" (ή "γλώσσας περιεχομένου"). Μια ACL ασχολείται με την επίτευξη επικοινωνίας μεταξύ των πρακτόρων (παρέχοντας μεταπληροφορίες για τους ίδιους και για το τί είδους πληροφορίες/ γνώσεις ανταλλάσσουν), ενώ μια γλώσσα αναπαράστασης γνώσης κωδικοποιεί το περιεχόμενο της γνώσης. Μια ACL μπορεί να χρησιμοποιήσει μια οποιαδήποτε γλώσσα περιεχομένου για κωδικοποίηση της ανταλλασσόμενης γνώσης.

Στο Παράρτημα IV γίνεται μια αρκετά εκτενής επισκόπηση των κυριότερων ACLs και γλωσσών αναπαράστασης γνώσης.

### 3.6 Χρήσεις των Πρακτόρων

Η χρήση πρακτόρων λογισμικού είναι πλέον συνήθης σε μια σειρά από περιοχές και εφαρμογές. Ως τέτοιες, ενδεικτικά και συνοπτικά μπορούν να αναφερθούν οι ακόλουθες [39][37][38][15][32]:

- Το ηλεκτρονικό εμπόριο (αναφορά μπορεί να γίνει σε ένα ευρύ φάσμα εφαρμογών, ξεκινώντας από την απλή πρόταση για αγορά ενός βιβλίου με βάση τις προτιμήσεις του ανθρώπου χρήστη μέχρι την συμμετοχή πρακτόρων σε δημοπρασίες ή /και τη διενέργεια τους από αυτούς) [36][68]
- Η διαχείριση χρόνου του ανθρώπου-χρήστη (λ.χ. προγραμματισμός συναντήσεων)
- Η διαχείριση διεπιφάνειας χρήσης
- Η διαχείριση προσωπικής ηλεκτρονικής αλληλογραφίας
- Η επιλογή ηλεκτρονικών ειδήσεων και γενικότερα άντληση πληροφοριών από τον Παγκόσμιο Ιστό με βάση τις προτιμήσεις του ανθρώπου-χρήστη
- Τα ρομποτικά συστήματα (λ.χ. πλοήγηση ρομπότ ή δράση πρακτόρων μέσα σε ομάδες για προσομοιώσεις ποδοσφαιρικών αγώνων και στρατιωτικών επιχειρήσεων) [59][64][28][40]
- Η ψυχαγωγία και τα ηλεκτρονικά παιχνίδια [38]
- Η επεξεργασία κατανεμημένων δεδομένων
- Η διαχείριση δικτύου και υπολογιστικών πόρων [9]
- Η διενέργεια επιστημονικών υπολογισμών [13]
- Η διαχείριση ροών εργασίας

Στην εργασία αυτή, σχεδιάστηκαν και αναπτύχθηκαν πράκτορες λογισμικού που χρησιμοποιούνται για την διενέργεια ψηφοφοριών. Στο κεφάλαιο της Αρχιτεκτονικής θα γίνει σαφές το πώς οι πράκτορες αυτοί αντιστοιχούν σε οντότητες του πραγματικού κόσμου, οι οποίες συμμετέχουν με διακριτούς ρόλους σε ένα γενικό και επεκτάσιμο μοντέλο ψηφοφοριών οι οποίες μπορεί να διενεργούνται ιεραρχικά.

## Κεφάλαιο 4 Αρχιτεκτονική

Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι μια αρχιτεκτονική πολλών συνεργαζόμενων πρακτόρων. Για την ανάπτυξή της χρησιμοποιήθηκαν οι αρχές της αντικειμενοστρεφούς ανάλυσης και σχεδιασμού [5][21]. Εγινε ο διαχωρισμός των εργασιών του συνολικού συστήματος και η κατανομή τους σε ανάλογες οντότητες, κάποιες από τις οποίες αποτελούν πράκτορες που επικοινωνούν με ανταλλαγή μηνυμάτων, και οι οποίοι μπορούν να διαχειρίζονται άλλες, μή ενεργές, οντότητες. Η πολυπλοκότητα του προβλήματος έκανε απαραίτητη τη χρήση κατάλληλου συμβολισμού και σύγχρονων εργαλείων για την ανάλυση και το σχεδιασμό της Αρχιτεκτονικής. Έτσι, επιλέχθηκε η χρησιμοποίηση της Ενοποιημένης Γλώσσας Μοντελοποίησης (Unified Modeling Language), η οποία τείνει να καθιερωθεί ως πρότυπο γλώσσας μοντελοποίησης, συνδυάζοντας τους συμβολισμούς Booch, OMT και Objectory, για τα διάφορα στάδια της ανάλυσης και του σχεδιασμού [51][77]. Το εργαλείο που χρησιμοποιήθηκε για την μοντελοποίηση σε UML ήταν το ευρέως χρησιμοποιούμενο Rational Rose [51] [79].

Στο Κεφάλαιο αυτό παρουσιάζεται η Αρχιτεκτονική Διενέργειας Ψηφοφοριών, όπως αυτή αναπτύχθηκε με χρήση τεχνολογίας πρακτόρων και με θεωρητικό υπόβαθρο βασικό σε στοιχεία της Θεωρίας Ψηφοφοριών. Αναλύονται οι απαιτήσεις και οι επιλογές που έγιναν κατά το σχεδιασμό της Αρχιτεκτονικής, παρουσιάζεται ένα γενικό παράδειγμα ψηφοφορίας από τον πραγματικό κόσμο, το οποίο χρησιμοποιήθηκε ως μοντέλο για την Αρχιτεκτονική, αναλύεται η εσωτερική αρχιτεκτονική ενός πράκτορα της Αρχιτεκτονικής, και γίνεται αναλυτική παρουσίαση όλων των πρακτόρων και των βασικών οντοτήτων, όπως επίσης των αλληλεπιδράσεών τους και των μηνυμάτων που ανταλλάσσουν για την επικοινωνία τους.

### 4.1 Απαιτήσεις και Επιλογές

Βασική απαίτηση κατά το σχεδιασμό και την υλοποίηση μιας οποιασδήποτε αρχιτεκτονικής συστήματος είναι η *επεκτασιμότητα*, δηλαδή η δυνατότητα αύξησης της λειτουργικότητας του συστήματος με την εύκολη προσθήκη λειτουργικών οντοτήτων με ελάχιστες αλλαγές στα ήδη υπάρχοντα μέρη του συστήματος. Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών υλοποιήθηκε ως μια αρχιτεκτονική πολλών *κατανεμημένων συνεργαζόμενων πρακτόρων*, κάτι που, όπως αναλύθηκε στο Κεφάλαιο "Πράκτορες" εξασφαλίζει την πλήρωση της απαίτησης της επεκτασιμότητας, καθώς η προσθήκη λειτουργικότητας στην συνολική

αρχιτεκτονική μπορεί να ταυτιστεί με την προσθήκη ενός νέου πράκτορα σε αυτήν, ο οποίος θα φέρει την επιπλέον λειτουργικότητα.

Άλλη απαίτηση για μια αρχιτεκτονική είναι αυτή της *δυνατότητας διασύνδεσής της με άλλες υπάρχουσες αρχιτεκτονικές*, προκειμένου να επιτυγχάνεται η συνεργασία της με αυτές χωρίς να πρέπει να εκτεθούν οι εσωτερικές λεπτομέρειες της υλοποίησης. Η Αρχιτεκτονική που παρουσιάζεται πληρεί αυτή την προϋπόθεση, καθώς μπορεί να χρησιμοποιηθεί ως τμήμα μιας οποιαδήποτε αρχιτεκτονικής της οποίας τα τμήματα επιθυμούν να εμπλακούν σε ψηφοφορίες. Αυτό μπορεί να επιτευχθεί με την απλή ενσωμάτωση των πρακτόρων της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών στα τμήματα της ευρύτερης αρχιτεκτονικής, τα οποία έτσι μπορούν να αποκτήσουν την λειτουργικότητα που απαιτείται για τη διεξαγωγή ψηφοφοριών.

Κατά το σχεδιασμό του μοντέλου της Αρχιτεκτονικής καταβλήθηκε προσπάθεια αυτό να είναι όσον το δυνατόν πιο *γενικό*, ώστε να μπορεί η Αρχιτεκτονική να χρησιμοποιείται σε πληθώρα περιπτώσεων. Επιπλέον, το γεγονός πως η Αρχιτεκτονική σχεδιάστηκε με βάση τις αρχές του αντικειμενοστρεφούς προγραμματισμού [5][21], επιτρέπει την εύκολη ενσωμάτωση σε αυτή πολλών διαφορετικών εκλογικών συστημάτων. Έτσι, επιτρέπεται η διεξαγωγή πολλών διαφορετικών πειραμάτων ψηφοφοριών, μια που η Θεωρία Ψηφοφοριών βεβαιώνει για τη μη ύπαρξη ενός *"δίκαιου" εκλογικού συστήματος*, αλλά και προκειμένου να είναι δυνατή η χρήση της Αρχιτεκτονικής από διάφορες εφαρμογές που πιθανότατα απαιτούν διενέργεια ψηφοφοριών με διαφορετικά εκλογικά συστήματα.

Λήφθηκε επίσης πρόνοια για την δυνατότητα *αξιολόγησης των επιδόσεων* των πρακτόρων που *απαρτίζουν* την Αρχιτεκτονική. Αυτό γίνεται με την καταγραφή των συμπεριφορών των πρακτόρων και τη δυνατότητα *επεξεργασίας* των καταγεγραμμένων δεδομένων από τους πράκτορες. Έτσι επιτρέπεται η *χρησιμοποίηση* της Αρχιτεκτονικής από εφαρμογές που επιθυμούν να μελετήσουν την συμπεριφορά των ψηφοφόρων ή και να την μεταβάλλουν δυναμικά.

Ακόμη, προβλέφθηκε η δυνατότητα *διαμόρφωσης άποψης* των συμμετέχοντων πρακτόρων μετά από μεταξύ τους διαβουλεύσεις, όπως *άλλωστε συμβαίνει* κατά τη διενέργεια ψηφοφοριών στην πραγματική ζωή. Η Αρχιτεκτονική παρέχει την κατάλληλη υποδομή για να γίνεται κάτι τέτοιο.

Επιπρόσθετη επιλογή κατά το σχεδιασμό της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών ήταν αυτή να έχει τη δυνατότητα υποστήριξης ψηφοφοριών σε *πολλαπλά επίπεδα ιεραρχίας*, κάτι που είναι επιθυμητό σε πολλές εφαρμογές ψηφοφοριών. Προκειμένου να επιτευχθεί αυτός ο στόχος, η Αρχιτεκτονική επιτρέπει τη δυναμική μετατροπή εκλογικών αποτελεσμάτων σε νέες υποψηφιότητες, ώστε αυτές να συμμετέχουν σε ψηφοφορίες που βρίσκονται υψηλότερα σε μια ιεραρχία ψηφοφοριών.

Τέλος, η Αρχιτεκτονική που παρουσιάζεται, ως αρχιτεκτονική για διενέργεια ψηφοφοριών, ικανοποιεί στο μεγαλύτερο δυνατό βαθμό τις επιθυμητές ιδιότητες ενός συστήματος *συλλογής ψηφοδελτίων*.

Στις ενότητες που ακολουθούν θα γίνει σαφέστερο το πώς ικανοποιούνται και υλοποιούνται οι προαναφερθείσες απαιτήσεις και επιλογές.

## 4.2 Ένα Παράδειγμα Ψηφοφορίας στην Καθημερινή Ζωή

Όπως προαναφέρθηκε, η ψηφοφορία είναι μια διαδικασία συλλογής και αποτίμησης των απόψεων των μελών ενός αρμόδιου εκλογικού σώματος *ψηφοφόρων /εκλογέων*, προκειμένου να ληφθεί μια απόφαση που αφορά ένα σαφώς καθορισμένο θέμα. Στην παράγραφο αυτή παρουσιάζεται ένα γενικό παράδειγμα διεξαγωγής μιας ψηφοφορίας στον πραγματικό κόσμο, το οποίο χρησιμοποιήθηκε ως μοντέλο για την ανάπτυξη της παρουσιαζόμενης Αρχιτεκτονικής.

Στον πραγματικό κόσμο, όταν υπάρχει η ανάγκη για τη διεξαγωγή μιας ψηφοφορίας, μια *δημόσια αρχή* (φερ' ειπείν η Βουλή ή η Κυβέρνηση) καθορίζει το ότι η ψηφοφορία θα διεξαχθεί, καθώς και τους όρους κάτω από τους οποίους αυτή θα διεξαχθεί. Τέτοιοι όροι διεξαγωγής είναι:

- Ο καθορισμός του σώματος των *ψηφοφόρων*, οι οποίοι έχουν το δικαίωμα να εκφράσουν την άποψή τους με την ψήφο τους.
- Ο καθορισμός των *υποψηφίων* (προσώπων ή εναλλακτικών λύσεων) ανάμεσα στους οποίους θα γίνει η επιλογή.
- Το *εκλογικό σύστημα* που θα χρησιμοποιηθεί, και το οποίο καθορίζει το πώς θα αποτιμηθεί η γνώμη του εκλογικού σώματος *σε εκλογικό αποτέλεσμα*.
- Η κατάρτιση *ψηφοδελτίων* όπου εμφανίζονται οι υποψήφιοι. Η ακριβής μορφή των ψηφοδελτίων εξαρτάται από το τί καθορίζει το εκλογικό σύστημα (λ.χ. μπορεί ο εκλογέας να επιλέξει πάνω από έναν υποψήφιο, μπορεί να χαρακτηρίσει μια επιλογή υποψηφίου με ένα αριθμητικό βάρος κλπ.).
- Η *διάρκεια* της εκλογικής διαδικασίας, μετά το πέρας της οποίας οι εκλογείς δεν δικαιούνται πλέον να ψηφίσουν.
- Ο ορισμός *εφορευτικής επιτροπής* που θα είναι υπεύθυνη για την ομαλή διεξαγωγή της ψηφοφορίας, την ορθή καταμέτρηση των ψήφων που έχουν κατατεθεί στην *κάλπη* και τη συνακόλουθη *εξαγωγή αποτελέσματος*, με βάση το τί ορίζει το εκλογικό σύστημα.

Κάποιοι από τους όρους διεξαγωγής της ψηφοφορίας ίσως να είναι ήδη καθορισμένοι όταν εμφανιστεί η ανάγκη για αυτήν. Μπορεί όμως και να μην είναι καθορισμένοι και να χρειαστεί να καθοριστούν (όπως π.χ. η εφορευτική επιτροπή). Η δημόσια αρχή που προκηρύσσει την ψηφοφορία, ή κάποια άλλη δημόσια αρχή, είναι σε θέση να καθορίσει κάποιους από τους όρους της ψηφοφορίας σε ανύποπτο χρονικό σημείο, πριν να προκύψει η ανάγκη για αυτήν (για παράδειγμα η Βουλή έχει τη δυνατότητα αλλαγής εκλογικού νόμου που θα ισχύσει στις κοινοβουλευτικές εκλογές, πολύ πριν αυτές διεξαχθούν).

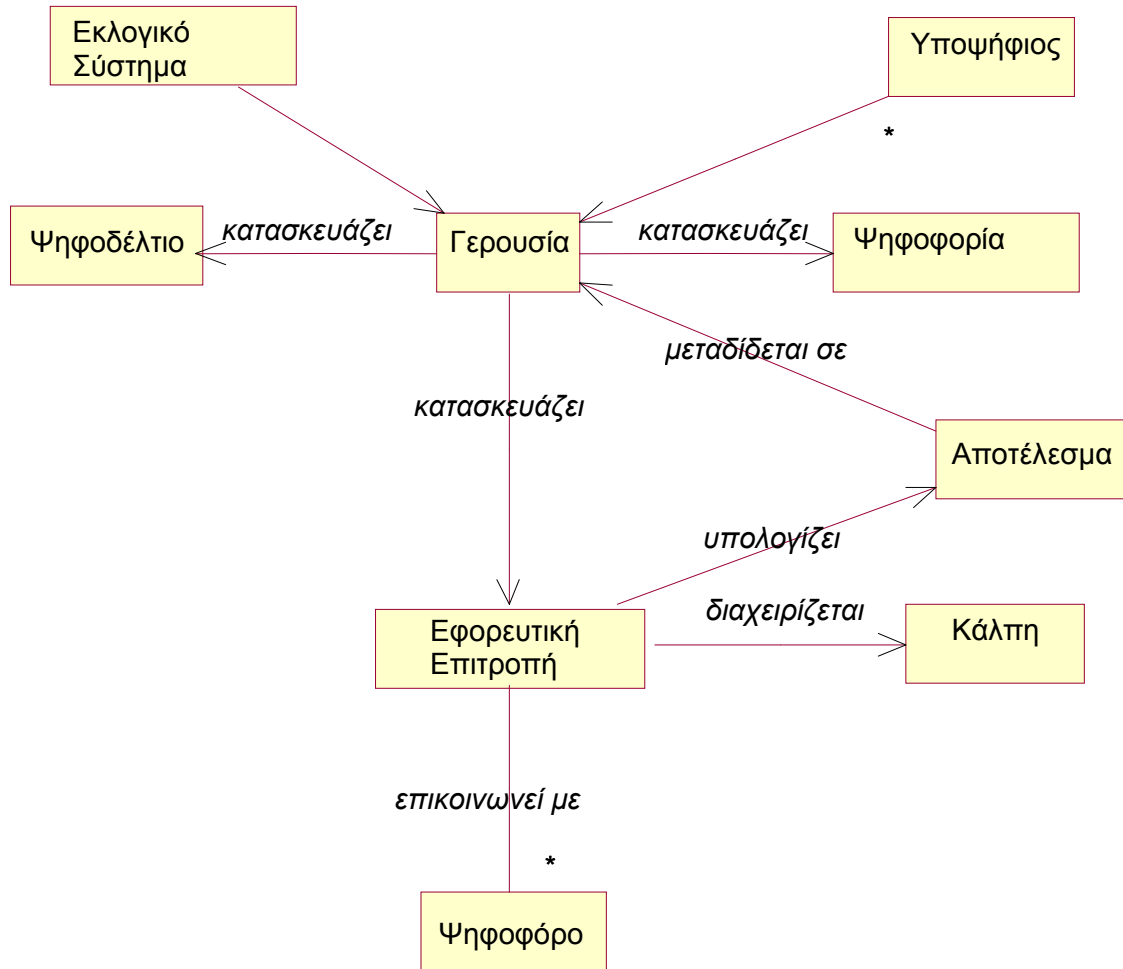
### 4.3 Πράκτορες, Βασικές Οντότητες και μεταξύ τους Αλληλεπιδράσεις

Με δεδομένο το παράδειγμα του πραγματικού κόσμου, οι βασικές οντότητες που εντοπίστηκαν στην Αρχιτεκτονική Διενέργειας Ψηφοφοριών, είναι η Γερουσία (Senate), ο Ψηφοφόρος (Voter), ο Υποψήφιος (Candidate), το Εκλογικό Σύστημα (Voting System), το Ψηφοδέλτιο (Ballot), η Κάλπη (Ballot Box), το Εκλογικό Αποτέλεσμα (Result), η Εφορευτική Επιτροπή (Voting Committee) και η ίδια η Ψηφοφορία (Voting Session). Οι ενεργές οντότητες που αποτελούν τους πράκτορες που συμμετέχουν στην Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι η Γερουσία, ο Ψηφοφόρος, ο Υποψήφιος και η Εφορευτική Επιτροπή.

Το Σχήμα 5 αποδίδει με συνοπτικό τρόπο τις αλληλεπιδράσεις μεταξύ των βασικών οντοτήτων που συμμετέχουν στην αρχιτεκτονική διενέργειας ψηφοφοριών.

Η Γερουσία είναι εκείνη που προκηρύσσει μια νέα ψηφοφορία (εκλογές) για κάποιο θέμα, αφού έχει δεχτεί ένα εξωτερικό ερέθισμα (μήνυμα) που της το ζητεί. Το μήνυμα αυτό προέρχεται από το σύστημα που χρησιμοποιεί την αρχιτεκτονική διεξαγωγής ψηφοφοριών (πιθανά να πρόκειται για αίτηση χρήστη). Το μήνυμα προσδιορίζει το εκλογικό σύστημα, πιθανά τους υποψηφίους, και τις υπόλοιπες εκλογικές παραμέτρους (όπως τους χρονικούς περιορισμούς για τη διάρκεια της προεκλογικής περιόδου και του διαστήματος ψηφοφορίας). Όταν η Γερουσία ορίσει μια νέα Ψηφοφορία, ενημερώνει όλους τους πράκτορες για το ότι αυτή προγραμματίστηκε, ώστε να αποφασίσουν αν θα συμμετάσχουν είτε ως Ψηφοφόροι, είτε ως Υποψήφιοι (οπότε και πρέπει να υποβάλλουν υποψηφιότητα εντός καθορισμένης περιόδου υποβολής υποψηφιοτήτων).





• Σχήμα 5 Αλληλεπιδράσεις μεταξύ βασικών οντοτήτων.

Στο σχήμα μπορεί να παρατηρηθεί ακόμα πως η Γερουσία καταρτίζει το ψηφοδέλτιο, και "ορίζει" μια εφορευτική επιτροπή για να διενεργήσει την ψηφοφορία, εφοδιάζοντάς την με το ψηφοδέλτιο, το εκλογικό σύστημα και τις υπόλοιπες απαιτούμενες εκλογικές παραμέτρους.

Η Εφορευτική Επιτροπή και οι ψηφοφόροι επικοινωνούν, είτε γιατί πρέπει να ενημερωθούν οι δεύτεροι για την έναρξη της ψηφοφορίας, είτε γιατί οι ψηφοφόροι αποστέλλουν το συμπληρωμένο ψηφοδέλτιο στην πρώτη.

Στο σχήμα φαίνεται το ότι η Εφορευτική Επιτροπή διαχειρίζεται την κάλπη (εισάγοντας τα έγκυρα ψηφοδέλτια και αργότερα καταμετρώντας τα περιεχόμενά της), και εξάγει το εκλογικό αποτέλεσμα, το οποίο αποστέλλει στην Γερουσία.

## 4.4 Αρχιτεκτονική ενός Πράκτορα

Η εσωτερική αρχιτεκτονική ενός πράκτορα της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών ακολουθεί το μοντέλο perception-cognition-action, το οποίο παρουσιάστηκε στο σχετικό με την τεχνολογία πρακτόρων κεφάλαιο. Ο κάθε πράκτορας μπορεί δηλαδή να θεωρηθεί ως αποτελούμενος από τρία τμήματα: το τμήμα αίσθησης, το γνωσιακό τμήμα και το τμήμα δράσης.

Το τμήμα αίσθησης επιτελεί τη λειτουργία της συλλογής μηνυμάτων που απευθύνονται προς τον πράκτορα, ενώ το τμήμα δράσης, ασχολείται με την εκτέλεση των ενεργειών του πράκτορα, την αποστολή μηνυμάτων και την καταγραφή συμπεριφορών.

Το γνωσιακό τμήμα του πράκτορα ασχολείται με την επιτέλεση των "νοητικών" λειτουργιών του. Οι νοητικές αυτές λειτουργίες ποικίλλουν ανάλογα με τη φύση του πράκτορα (με το αν είναι Ψηφοφόρος, Υποψήφιος, Γερουσία ή Εφορευτική Επιτροπή). Για παράδειγμα, το γνωσιακό τμήμα των πρακτόρων-Ψηφοφόρων, όπως έχει υλοποιηθεί, έχει μεταξύ άλλων τη δυνατότητα διασύνδεσης με "εξωτερικό" της Αρχιτεκτονικής υποσύστημα διεξαγωγής συμπερασματικών λειτουργιών, που καθορίζει την ψήφο του, ενώ το γνωσιακό τμήμα της Γερουσίας έχει τη δυνατότητα διασύνδεσης με τμήμα στατιστικής επεξεργασίας εκλογικών αποτελεσμάτων (κάτι που χρησιμεύει και στην αξιολόγηση των Ψηφοφόρων).

Οι νοητικές λειτουργίες του γνωσιακού τμήματος ενός πράκτορα μπορούν εύκολα να επαυξηθούν ή να αλλάξουν ριζικά, απλά με την αντικατάστασή του. Κάτι τέτοιο είναι απλό να γίνει, καθώς ο οντοκεντρικός σχεδιασμός της αρχιτεκτονικής επιτρέπει την θεώρηση του κάθε τμήματος του πράκτορα ως ένα "μαύρο κουτί".

## 4.5 Παρουσίαση των Πρακτόρων και των Βασικών Οντοτήτων

Οι πράκτορες, οι οντότητες και οι μεταξύ τους αλληλεπιδράσεις είναι τα στοιχεία που συναπαρτίζουν την κατανεμημένη Αρχιτεκτονική πολλών πρακτόρων που παρουσιάζεται. Η ενότητα που ακολουθεί έχει ως σκοπό να περιγράψει με αναλυτικό τρόπο όλους τους πράκτορες, τις βασικές οντότητες, καθώς και τις αλληλεπιδράσεις τους.

### 4.5.1 Γερουσία

Πρόκειται για τον πράκτορα που αποτελεί το ανάλογο μιας δημόσιας αρχής/ νομοθετικού σώματος. Είναι ο πράκτορας που μπορεί να εκκινεί - μετά από ερέθισμα προερχόμενο από εξωτερική οντότητα που χρησιμοποιεί την Αρχιτεκτονική Διενέργειας Ψηφοφοριών, πιθανά τον χρήστη - πολλές διαφορετικές ψηφοφορίες (voting sessions), ορίζοντας εφορευτική

επιτροπή για κάθε μια από αυτές, την οποία εφοδιάζει με όλες τις απαραίτητες εκλογικές παραμέτρους (σκοπό ψηφοφορίας, εκλογικό σύστημα, χρονικούς περιορισμούς ψηφοφορίας, υποψήφιες εναλλακτικές προτάσεις, ψηφοδέλτιο, αριθμό "νικηφόρων" υποψηφίων) αφού προσδιορίσει όσες από αυτές είναι αρχικά απροσδιόριστες.

Αναλυτικότερα, η Γερουσία δέχεται από εξωτερική οντότητα μήνυμα/ εξωτερικό ερέθισμα με προσδιορισμό του σκοπού ψηφοφορίας. Στο μήνυμα αυτό μπορεί ή όχι να περιέχεται ως παράμετρος το εκλογικό σύστημα - αν όχι, τότε το αποφασίζει η ίδια. Η Γερουσία δέχεται επίσης τους χρονικούς περιορισμούς της ψηφοφορίας σαν ποιοτική παράμετρο, και αποφασίζει τον πραγματικό χρόνο διάρκειας ψηφοφορίας στον οποίο αντιστοιχούν οι ποιοτικοί προσδιορισμοί, καθώς και τη διάρκεια της προεκλογικής περιόδου. Η σχετική απόφαση λαμβάνεται μετά από αποτίμηση του υπολογιστικού φόρτου.

Η Γερουσία *λαμβάνει ή δε λαμβάνει* πληροφορία για υποψήφιες εναλλακτικές προτάσεις από το εξωτερικό ερέθισμα.

*Αν δεν λαμβάνει*, τότε κάποιες ενεργές οντότητες θα θέσουν κάποιες υποψήφιες εναλλακτικές προτάσεις μέσα σε ένα χρονικό διάστημα "υποβολής υποψηφιοτήτων" προς τη Γερουσία, το οποίο καθορίζεται με παρόμοιο τρόπο με τους χρονικούς περιορισμούς της ψηφοφορίας.

*Αν λαμβάνει*, τότε η πληροφορία αυτή είναι:

- υποψήφιες εναλλακτικές προτάσεις
- ενεργές οντότητες (πράκτορες) που θα καταθέσουν υποψήφιες εναλλακτικές προτάσεις<sup>7</sup>

Δεδομένου του σκοπού, του εκλογικού συστήματος και των υποψηφίων εναλλακτικών προτάσεων, η Γερουσία καταρτίζει το ψηφοδέλτιο. Δημιουργεί μια νέα διαδικασία ψηφοφορίας (νέα voting session), καθώς και μια εφορευτική επιτροπή που να αντιστοιχεί σε αυτήν, την οποία και εφοδιάζει με όλες τις εκλογικές παραμέτρους. Επίσης, φροντίζει να ενημερώσει τους πράκτορες-ψηφοφόρους και τους πράκτορες-υποψηφίους για το ότι προγραμματίζεται μια νέα ψηφοφορία, αποστέλλοντάς τους και τις παραμέτρους, ώστε αυτοί να αποφασίσουν αν θα συμμετάσχουν σε αυτήν είτε ως ψηφοφόροι είτε ως υποψήφιοι (οπότε και πρέπει να υποβάλλουν υποψηφιότητα).

Η Γερουσία διαθέτει λειτουργίες που μετατρέπουν το αποτέλεσμα μιας ψηφοφορίας σε πράκτορα - υποψήφιο. Έτσι, το αποτέλεσμα μιας ψηφοφορίας μπορεί να μεταφερθεί ως πιθανή υποψήφια πρόταση σε μια άλλη ψηφοφορία, και ως εκ τούτου είναι δυνατή η διενέργεια μιας ιεραρχικής ακολουθίας από ψηφοφορίες.

<sup>7</sup> Αυτό γίνεται μέσα στην περίοδο υποβολής υποψηφιοτήτων

Ρόλος της Γερουσίας είναι και το να επεξεργάζεται στατιστικά συγκεντρωμένες πληροφορίες από το σύστημα (για ενέργειες των οντοτήτων και τα αποτελέσματα των ψηφοφοριών), καθώς και το να αξιολογεί τους ψηφοφόρους βασισμένη στα παραπάνω επεξεργασμένα στοιχεία. Ο ακριβής καθορισμός του βέλτιστου τρόπου εκπλήρωσης αυτών των ρόλων, αποτελεί αντικείμενο μελλοντικής εργασίας. Αντικείμενο μελλοντικής εργασίας αποτελεί και ο προσδιορισμός της διαδικασίας επιλογής (αν δεν προσδιορίζεται από το μήνυμα) ή και παραγωγής ενός εκλογικού συστήματος από τη Γερουσία.

Η Γερουσία έχει διάρκεια ζωής όση και το σύστημα.

#### 4.5.2 Υποψήφιος

Πρόκειται για τον πράκτορα που αντιπροσωπεύει τον υποψήφιο. Κύρια γνωρίσματα της οντότητας είναι το χαρακτηριστικό ταυτότητας του Υποψηφίου (που περιλαμβάνει και το όνομά του), και η εναλλακτική πρόταση που αυτός καταθέτει προς ψήφιση. Ο Υποψήφιος έχει τη δυνατότητα να υποβάλλει την υποψηφιότητά του για μια ψηφοφορία στην Γερουσία εντός συγκεκριμένης προθεσμίας υποβολής υποψηφιοτήτων. Ο Υποψήφιος λαμβάνει, χρησιμοποιώντας το γνωσιακό του τμήμα, την απόφαση για συμμετοχή ή μή σε μια ψηφοφορία, αφού ενημερωθεί, από τη Γερουσία, για το θέμα της. Το γνωσιακό τμήμα ελέγχει το αν η εναλλακτική πρόταση του Υποψηφίου είναι συμβατή με το θέμα, κι ακόμα μπορεί να ελέγχει το αν η πρότασή του αυτή έχει υπερψηφιστεί στο παρελθόν σε σχετικές ψηφοφορίες.

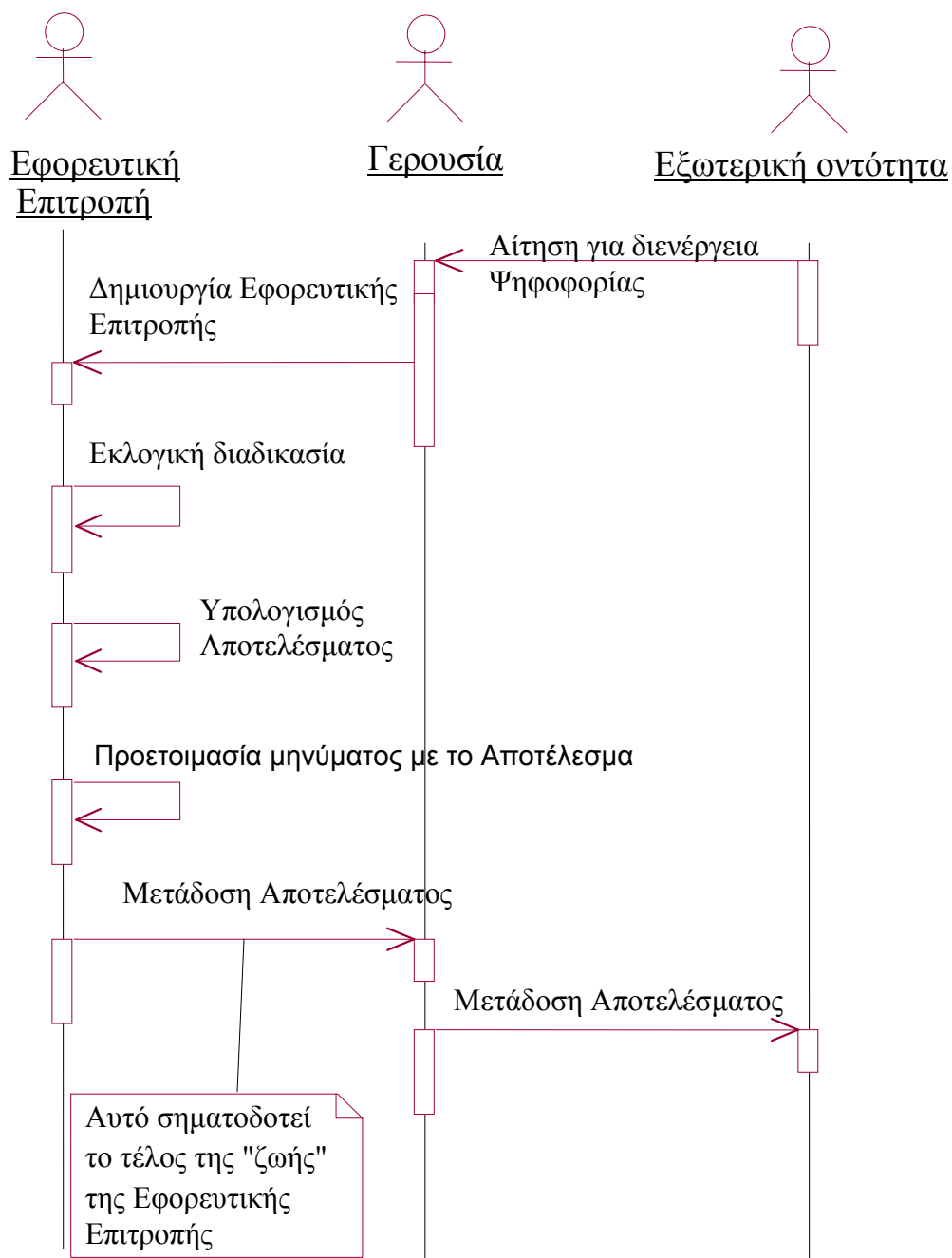
Μέρος μελλοντικής εργασίας αποτελεί το να δοθούν στον Υποψήφιο περισσότερες δυνατότητες "ενεργητικής δράσης", όπως η δυνατότητα να καθορίζει και να αποστέλλει "προεκλογικό υλικό" κατά τη διάρκεια της προεκλογικής περιόδου. Το προεκλογικό υλικό μπορεί να είναι απλή περιγραφή των δυνατοτήτων του Υποψηφίου, αλλά θα μπορούσε να είναι και "παροχή οφελημάτων" προς τους ψηφοφόρους (**εξαγορά ψήφων**). Σε μια τέτοια περίπτωση, που απαιτεί βέβαια τα χρησιμοποιούμενα γνωσιακά τμήματα των πρακτόρων να αποκτήσουν την ανάλογη πολυπλοκότητα, η Αρχιτεκτονική Ψηφοφοριών θα μπορούσε να συσχετιστεί με αρχιτεκτονικές δημοπρασιών.

### 4.5.3 Εφορευτική Επιτροπή

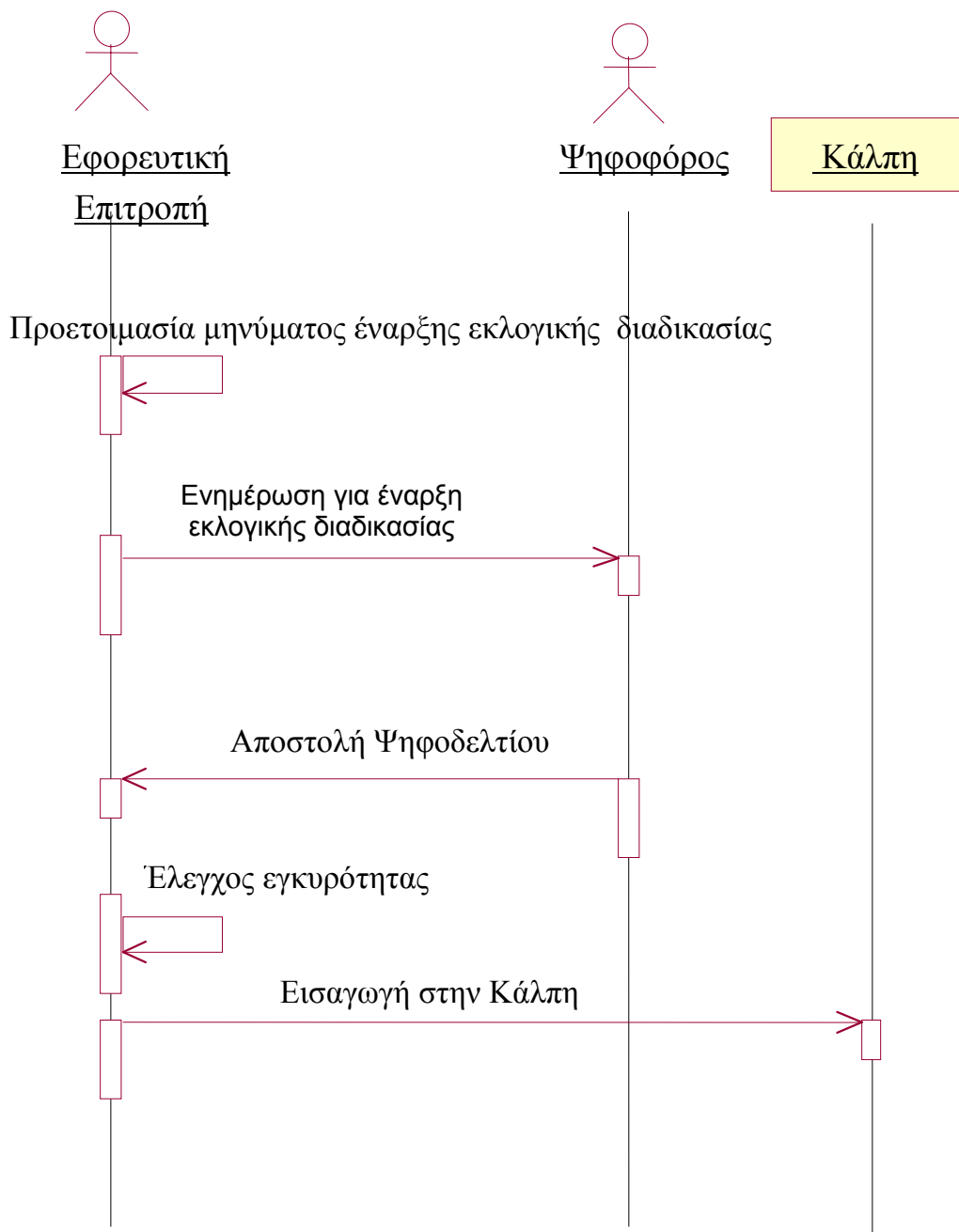
Πρόκειται για τον πράκτορα που αντιστοιχεί στην εφορευτική επιτροπή που συναντάται σε μια πραγματική εκλογική διαδικασία. Δημιουργείται ως οντότητα (από την Γερουσία) δυναμικά για κάθε ξεχωριστή ψηφοφορία, και "ζει" όσο διαρκεί αυτή (Σχήμα 6). Κατά τη δημιουργία της λαμβάνει το εκλογικό σύστημα, το ψηφοδέλτιο, τις υποψήφιες εναλλακτικές προτάσεις, και τις λοιπές παραμέτρους της ψηφοφορίας (όπως τους χρονικούς περιορισμούς).

Ο ρόλος της περιλαμβάνει την έναρξη και την επίβλεψη της ομαλής διεξαγωγής της εκλογικής διαδικασίας (με τήρηση των - χρονικών - περιορισμών της ψηφοφορίας), τον έλεγχο της εγκυρότητας των ψηφοδελτίων που αποστέλλονται σε αυτήν από τους ψηφοφόρους, την εισαγωγή των εγκύρων ψηφοδελτίων στην κάλπη, την καταμέτρηση των ψήφων (με βάση το εκλογικό σύστημα) και την εξαγωγή του αποτελέσματος της ψηφοφορίας. Επίσης, τελική ενέργεια της Εφορευτικής Επιτροπής είναι η αποστολή του αποτελέσματος στην Γερουσία.

Η έναρξη της ψηφοφορίας γίνεται με την αποστολή σχετικού μηνύματος προς τους πράκτορες-Ψηφοφόρους. Ο έλεγχος της εγκυρότητας των ψηφοδελτίων γίνεται με βάση το ποιοί είναι οι υποψήφιοι και το εκλογικό σύστημα. Η Εφορευτική Επιτροπή αποφασίζει πότε έρχεται το τέλος της ψηφοφορίας (ελέγχοντας αν επήλθε η λήξη του προσδιορισμένου από τους περιορισμούς ψηφοφορίας χρόνου). Η διαδικασία συλλογής ψηφοδελτίων παρουσιάζεται συνοπτικά στο Σχήμα 7.



• Σχήμα 6 Επικοινωνία Γερουσίας - Εφορευτικής Επιτροπής. Το Αποτέλεσμα της Ψηφοφορίας μεταδίδεται στην οντότητα που ζήτησε τη διενέργεια της Ψηφοφορίας.



• Σχήμα 7 Συνοπτική παρουσίαση της συλλογής των ψηφοδελτίων.

Η συλλογή των ψηφοδελτίων από την Εφορευτική Επιτροπή γίνεται κατά τρόπο **ακριβή** και **μή τρωτό**, όπως είναι επιθυμητό να συμβαίνει κατά οποιαδήποτε διαδικασία *συλλογής ψηφοδελτίων*. Η **ακρίβεια** κατά την συλλογή εξασφαλίζεται με τους ελέγχους εγκυρότητας που κάνει η Εφορευτική Επιτροπή όταν παραλαμβάνει το ψηφοδέλτιο που ο ψηφοφόρος έχει αποστείλει. Επιπλέον, η χρονική εγκυρότητα εξασφαλίζεται από το ότι ο ψηφοφόρος είναι δυνατόν να αποστείλει την ψήφο του μόνο εντός των χρονικών ορίων της ψηφοφορίας, κι όχι εκπρόθεσμα, καθώς σε μια τέτοια περίπτωση η Εφορευτική Επιτροπή δε θα κάνει

δεκτή την ψήφο - πιθανότατα επειδή δε θα υφίσταται πλέον ως οντότητα. Η **μή τρωότητα** κατά τη συλλογή των ψηφοδελτίων εξασφαλίζεται από το ότι ένας πράκτορας-ψηφοφόρος έχει γίνει αποδεκτός κατ' αρχήν στην κοινωνία των πρακτόρων (με τρόπο που αναφέρεται σε επόμενη ενότητα), πράγμα που του παρέχει το δικαίωμα να ψηφίσει, καθώς και από το γεγονός ότι ο ίδιος (λόγω "προγραμματιστικής επιταγής") αποστέλλει την ψήφο του μία φορά. Η Εφορευτική Επιτροπή θα μπορούσε να εξασφαλίσει άλλωστε την αποφυγή "πολυψηφίας" με έλεγχο του "προσδιοριστικού ταυτότητας" του ψηφοφόρου (το οποίο μπορεί να αναγράφεται στο ψηφοδέλτιο) προτού "τοποθετήσει" το ψηφοδέλτιο στην κάλπη. Κάτι τέτοιο όμως δε γίνεται, καθώς δεν είναι υποχρεωτικό για τον ψηφοφόρο να φανερώνει την ταυτότητά του για λόγους **μυστικότητας** της ψηφοφορίας (παρ' όλο που το να φανερώνει την ταυτότητά του είναι σε πολλές εφαρμογές επιθυμητό, για λόγους δυνατότητας καταγραφής της συμπεριφοράς του και συνακόλουθης "αξιολόγησής" του).

Με όσα αναφέρθηκαν, η Εφορευτική Επιτροπή φαίνεται να μη διαθέτει κάποιο "ευφές" ή αρκετά δυναμικό γνωσιακό τμήμα. Αυτό όμως είναι επόμενο λόγω της φύσης του ρόλου της (που τη θέλει να υποτάσσεται σε υπερκείμενους κανόνες, αυτούς που υπαγορεύει το εκλογικό σύστημα), μπορεί όμως εύκολα να αποκτήσει μια διαφορετική συμπεριφορά - ενδεχομένως περισσότερο δυναμική και ευέλικτη - με κατάλληλη αντικατάσταση του γνωσιακού της τμήματος (κάτι που, όπως θα αναφερθεί στο κεφάλαιο το σχετικό με ζητήματα υλοποίησης, είναι αρκετά απλό να γίνει για κάθε πράκτορα της παρουσιαζόμενης αρχιτεκτονικής).

#### 4.5.4 Ψηφοφόρος

Πρόκειται για την οντότητα που αντιστοιχεί στον ψηφοφόρο, ο οποίος καλείται να εκφράσει με την ψήφο του την άποψή του επί του θέματος της ψηφοφορίας. Ένας Ψηφοφόρος μπορεί να συμμετέχει σε πολλές ψηφοφορίες ταυτόχρονα.

Ο Ψηφοφόρος έχει τη δυνατότητα να αντιλαμβάνεται το πότε καλείται να ψηφίσει σε μια προγραμματισμένη εκλογή, και το πότε ακριβώς γίνεται η έναρξη της ψηφοφορίας, καθώς και το υπό ποιές συνθήκες καλείται να ψηφίσει σε μια εκλογή (όπως ποιοί είναι οι χρονικοί περιορισμοί, το εκλογικό σύστημα κλπ.). Έχει επίσης τη δυνατότητα να αντιλαμβάνεται βάσει ποιού εκλογικού συστήματος καλείται να ψηφίσει (ποιό είναι και ποιοί κανόνες το διέπουν), και φυσικά τη δυνατότητα ψήφου, αφού διαμορφώσει την άποψή του για το θέμα. Ο Ψηφοφόρος αρχικά επιλέγει να συμμετάσχει ή όχι σε μια ψηφοφορία, ανάλογα με το θέμα της, το οποίο του γίνεται γνωστό ήδη με το μήνυμα προγραμματισμού της ψηφοφορίας, το οποίο αποστέλλει η Γερουσία.

Η διαδικασία κατάθεσης της ψήφου του Ψηφοφόρου μπορεί να αναλυθεί στα ακόλουθα στάδια:



- Συμπλήρωση ψηφοδελτίου

Γίνεται με γνώση των παραμέτρων ψηφοφορίας και του εκλογικού συστήματος (και μελλοντικά και με γνώση μέρους ή του όλου του προεκλογικού υλικού). Κατά τη συμπλήρωση ψηφοδελτίου, ο Ψηφοφόρος πρέπει να εκφράσει με μετρικό τρόπο την "άποψή του".

- Αποστολή ψήφου

Η ψήφος αποστέλλεται στην Εφορευτική Επιτροπή και αυτή την κάνει ή όχι αποδεκτή με βάση τις συνθήκες ψηφοφορίας (δηλαδή το αν είναι εμπρόθεσμη χρονικά και το αν το ψηφοδέλτιο είναι σωστά - σύμφωνα με τις επιταγές του εκλογικού συστήματος - συμπληρωμένο).

Τέλος, άλλη μια δυνατότητα του Ψηφοφόρου είναι να κάνει γνωστή την ύπαρξή του και τις συνθήκες ύπαρξής του, όπως την τοποθεσία και τον φόρτο του μηχανήματος στο οποίο βρίσκεται, (εφόσον του είναι δυνατόν να τον αποτιμήσει).

#### 4.5.4.1 Διαμόρφωση Άποψης Ψηφοφόρου

Η διαμόρφωση της άποψης του Ψηφοφόρου γίνεται από το γνωσιακό του τμήμα, μέσω της αλληλεπίδρασής του με κάποιο υποσύστημα διεξαγωγής νοητικών/ συμπερασματικών διαδικασιών (πιθανά απλά έναν αλγόριθμο), ή με την αξιοποίηση γνώσης που προέρχεται από άλλες πηγές. Τέτοιες πηγές μπορεί να είναι άλλοι Ψηφοφόροι -και μάλιστα Ψηφοφόροι επί του ίδιου θέματος- ή η συγκεντρωμένη εμπειρία του ίδιου του Ψηφοφόρου. Η συγκεντρωμένη αυτή εμπειρία βρίσκεται αποθηκευμένη σε αρχεία προσπελάσιμα από τον Ψηφοφόρο.

Η γνώμη (ή η γνώση) άλλων πέρα από Ψηφοφόρους δε φαίνεται λογικό να ζητείται ή να παρέχεται, καθώς α) οι μεν Υποψήφιοι είτε δεν είναι ουσιαστικά ενεργές οντότητες, είτε θα έπρεπε να θεωρηθούν προκατειλημμένοι ως υποψήφιοι, β) η Εφορευτική Επιτροπή "ζει" μόνο για μια ψηφοφορία, και γ) η Γερουσία δεν έχει ρόλο να παρακολουθεί κάθε Ψηφοφόρο ξεχωριστά, κι άλλωστε ένας ρόλος της μπορεί μεν να είναι να επεξεργάζεται συγκεντρωμένες πληροφορίες από το σύστημα ή να εξάγει στατιστικά, δεν είναι όμως η μόνη που έχει πρόσβαση στις "βάσεις δεδομένων" (αρχεία) με αυτά τα στοιχεία.

Σε περίπτωση που υπάρχει αξιοποίηση της "προεκλογικής περιόδου" από κάποιους Υποψηφίους, ο Ψηφοφόρος μπορεί να αξιοποιεί για τη διαμόρφωση της άποψής του και το "προεκλογικό υλικό" που του αποστέλλουν.

#### 4.5.5 Ψηφοφορία

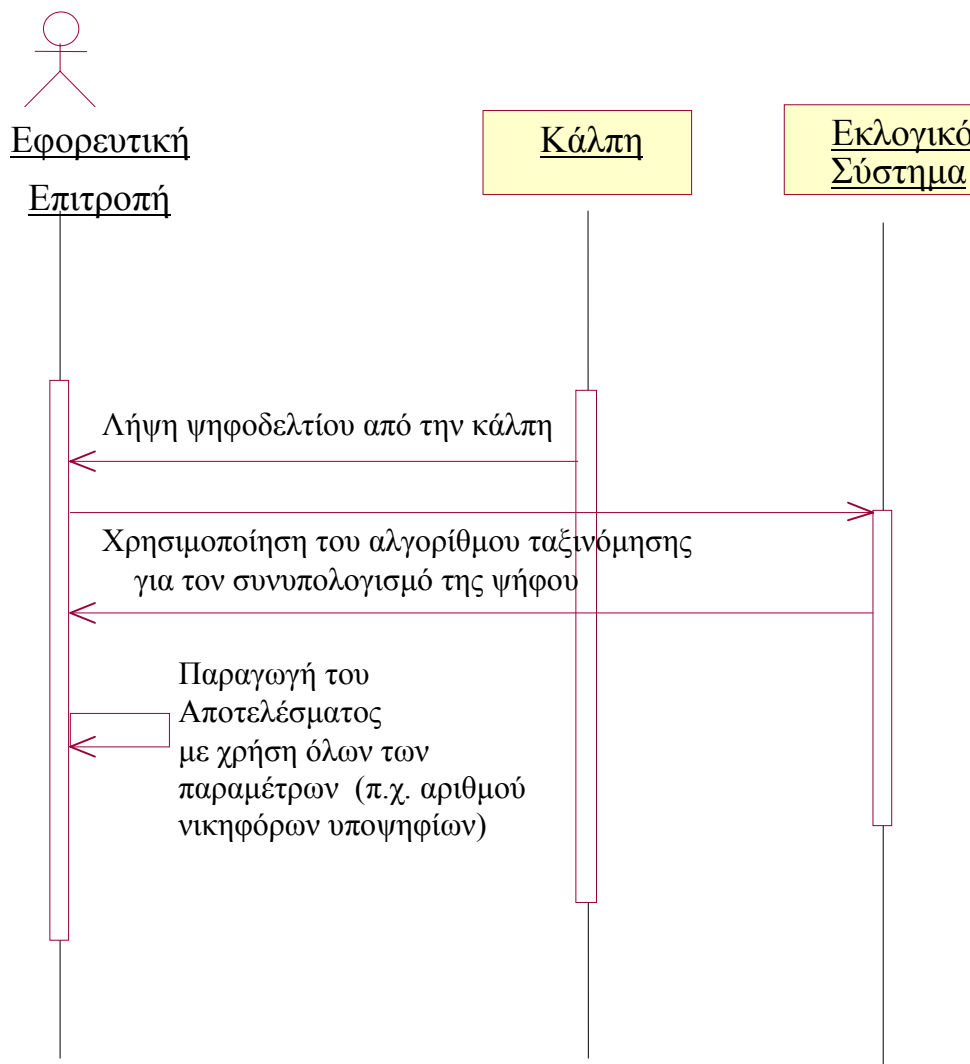
Η οντότητα αυτή μοντελοποιεί μια ψηφοφορία, για συγκεκριμένο θέμα. Παρέχει πληροφορίες για το ποιός είναι ο σκοπός της ψηφοφορίας, ποιές οι υποψηφίες εναλλακτικές προτάσεις, η Εφορευτική Επιτροπή, το Εκλογικό Σύστημα, οι περιορισμοί που ισχύουν και το Ψηφοδέλτιο που χρησιμοποιείται κατά τη διάρκεια της ψηφοφορίας. Επίσης αναφέρει ποιός είναι ο αριθμός των υποψηφίων που θα θεωρηθούν "νικηφόροι" σε αυτοί την ψηφοφορία.

#### 4.5.6 Εκλογικό Σύστημα

Πρόκειται για την οντότητα που αντιστοιχεί στο εκλογικό σύστημα. Το Εκλογικό Σύστημα διευκρινίζει τους κανόνες αποτίμησης των ψηφοδελτίων και εξαγωγής του αποτελέσματος της ψηφοφορίας.

Το Εκλογικό Σύστημα διευκρινίζει τον τύπο των αριθμητικών πεδίων προς συμπλήρωση τα οποία υπάρχουν στο ψηφοδέλτιο. Διαθέτει μια *λειτουργία κατάταξης των υποψηφίων εναλλακτικών λύσεων*, λαμβάνοντας υπ' όψιν τη λίστα των υποψηφίων εναλλακτικών λύσεων και τα συμπληρωμένα από τους Ψηφοφόρους ψηφοδέλτια, και διαθέτοντας μια *συνάρτηση (αλγόριθμο) συνδυασμού/ αποτίμησης των παραπάνω*, η οποία διαφέρει ανάλογα με το ποιό είναι το εκλογικό σύστημα που η συγκεκριμένη οντότητα μοντελοποιεί. Η διατεταγμένη λίστα των υποψηφίων εναλλακτικών λύσεων χρησιμοποιείται από την Εφορευτική Επιτροπή (μαζί με άλλα στοιχεία, όπως ο καθορισμένος αριθμός "νικηφόρων υποψηφίων") για την εξαγωγή του Αποτελέσματος της Ψηφοφορίας (Σχήμα 8).

Το Εκλογικό Σύστημα διαθέτει επίσης μια λίστα στην οποία καθορίζονται "αριθμητικά βάρη" συγκεκριμένων ψηφοφόρων, σε περίπτωση που πρόκειται για κάποιο εκλογικό σύστημα "βεβαρυμένης άποψης ψηφοφόρων" ("weighted" voting system), όπου η άποψη (ψήφος) κάθε ψηφοφόρου μπορεί να έχει διαφορετική βαρύτητα. Σε οποιαδήποτε άλλη περίπτωση, η λίστα αυτή παραμένει κενή.



• Σχήμα 8 Χρήση του Εκλογικού Συστήματος για υπολογισμό του Αποτελέσματος της Ψηφοφορίας.

#### 4.5.7 Ψηφοδέλτιο

Πρόκειται για την οντότητα που αντιστοιχεί στο ψηφοδέλτιο που καταρτίζει η Γερουσία με βάση το Εκλογικό Σύστημα. Το ψηφοδέλτιο παρέχεται από τη Γερουσία στην Εφορευτική Επιτροπή κατά τη δημιουργία της, και προωθείται (από τη Γερουσία) στους Ψηφοφόρους με κατάλληλο μήνυμα προγραμματισμού ψηφοφορίας. Τελικά το ψηφοδέλτιο συμπληρώνεται από τους Ψηφοφόρους και κατατίθεται στην κάλπη.

Το ψηφοδέλτιο μοντελοποιήθηκε ως μια λίστα από ζεύγη της μορφής: *{όνομα υποψηφίου - αριθμητικό πεδίο προς συμπλήρωση}*, όπου το πεδίο προς συμπλήρωση μπορεί να πάρει μια αριθμητική τιμή. Με τον τρόπο αυτό μπορούν να μοντελοποιηθούν μορφές ψηφοδελτίων που αντιστοιχούν σε ευρύ φάσμα εκλογικών συστημάτων (και πάντως σε όλο το φάσμα των

εκλογικών συστημάτων που υλοποιήθηκαν σε αυτή την εργασία), καθώς το πεδίο προς συμπλήρωση μπορεί να πάρει αριθμητική τιμή που θα προσδιορίζει είτε βάρος /ποσοστό επιλογής, είτε σειρά κατάταξης, είτε απλή επιλογή υποψήφιας εναλλακτικής πρότασης χωρίς βάρος ή σειρά κατάταξης (λ.χ. το "1" αντιστοιχεί σε επιλογή, το "0" σε μη επιλογή).

Πέρα από τη βασική λίστα υποψηφίων - αριθμητικών πεδίων, το ψηφοδέλτιο όπως μοντελοποιήθηκε περιλαμβάνει το όνομα του Ψηφοφόρου (προαιρετικά, για λόγους δυνατότητας στατιστικής επεξεργασίας και διατήρησης αρχείων καταγραφής της συμπεριφοράς του συστήματος), το σε ποιά Ψηφοφορία αντιστοιχεί η ψήφος, δυνατότητα παρατηρήσεων /σχολίου από τον Ψηφοφόρο και, τέλος, τον τύπο (βάρη / σειρά κατάταξης κλπ.) των αριθμητικών πεδίων.

#### **4.5.8 Εκλογικό Αποτέλεσμα**

Πρόκειται για την οντότητα που αντιστοιχεί στο εκλογικό αποτέλεσμα μιας ψηφοφορίας. Εξάγεται από την Εφορευτική Επιτροπή και μεταδίδεται από αυτήν στην Γερουσία με κατάλληλο μήνυμα (Σχήμα 8 & Σχήμα 6). Μοντελοποιήθηκε, σε αναλογία με το Ψηφοδέλτιο, ως διατεταγμένη ανάλογα με την σειρά επιλογής από τους Ψηφοφόρους λίστα των υποψηφίων και των αντίστοιχων αριθμητικών πεδίων.

Εκτός από τη διατεταγμένη λίστα, το Αποτέλεσμα αναφέρει σε ποιά Ψηφοφορία αντιστοιχεί, ποιός ήταν ο τύπος των αριθμητικών πεδίων, και ακόμη μια λίστα με τα Ψηφοδέλτια που υπήρχαν στην κάλη ώστε να υπάρχει η δυνατότητα στατιστικής επεξεργασίας του από την Γερουσία.

Τέλος, το Αποτέλεσμα διαθέτει λειτουργίες που το μετατρέπουν σε Υποψήφιο. Έτσι, μια οντότητα που έχει τη δυνατότητα αντίληψης του Αποτελέσματος, είναι σε θέση να καταρτίσει με βάση αυτό δυναμικά μια νέα υποψήφια πρόταση, προκειμένου να δημιουργηθεί μια ιεραρχία από ψηφοφορίες.

#### **4.5.9 Κάλη**

Πρόκειται για την οντότητα που αντιπροσωπεύει το μέσο όπου αποτίθενται οι ψήφοι στην πραγματική ζωή. Την οντότητα τη διαχειρίζεται η Εφορευτική Επιτροπή. Μοντελοποιείται ως μια λίστα από ψηφοδέλτια, με λειτουργίες ενημέρωσης, απαρίθμησης και ανάκτησής τους.

## 4.6 Διαχείριση Ομάδας Πρακτόρων και Ψηφοφορίες σε Ομάδες

Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι, όπως προαναφέρθηκε, μια αρχιτεκτονική πολλών συνεργαζόμενων πρακτόρων, οι οποίοι παίζουν το ρόλο της Γερουσίας, των Ψηφοφόρων, των Υποψηφίων ή των Εφορευτικών Επιτροπών. Πρόκειται δηλαδή για μια "κοινωνία", μια ομάδα από "πολίτες" που οργανώνονται με σκοπό τη διενέργεια ψηφοφοριών.

Όπως κάθε ομάδα, έτσι και η ομάδα των πρακτόρων της παρουσιαζόμενης αρχιτεκτονικής έχει ανάγκη "διαχείρισης", δηλαδή ενός πρωτοκόλλου καθορισμού των προϋποθέσεων και του τρόπου με τον οποίο μέλη εντάσσονται ή αποχωρούν από την ομάδα. Το ρόλο του διαχειριστή της ομάδας τον έχει η Γερουσία, η οποία άλλωστε αποτελεί τη μακροβιότερη οντότητα της αρχιτεκτονικής, και έχει, ως ανάλογο της "δημόσιας αρχής" που συναντάται στον πραγματικό κόσμο, τον πλέον σημαντικό ρόλο στην Αρχιτεκτονική Διενέργειας Ψηφοφοριών.<sup>8</sup>

Ένας πράκτορας που επιθυμεί να συμμετάσχει στην ομάδα που διαχειρίζεται η Γερουσία, πρέπει να το δηλώσει αποστέλλοντας σε αυτήν κατάλληλο μήνυμα. Το μήνυμα αυτό ανήκει σε μια κατηγορία μηνυμάτων για τη διαχείριση ομάδας, τα οποία περιγράφονται σε ακόλουθη ενότητα. Η Γερουσία κάνει ή δεν κάνει αποδεκτή την αίτηση ένταξης, και αποστέλλει επιβεβαίωση ένταξης στο νέο μέλος, με κατάλληλο μήνυμα με το οποίο το ενημερώνει για τη σύνθεση της ομάδας. Οι αιτήσεις ένταξης γίνονται γενικά αποδεκτές, εκτός και αν προέρχονται από Ψηφοφόρους που έχουν πρόσφατα αποβληθεί από την ομάδα. (Το πώς γίνεται αυτό αναφέρεται παρακάτω).

Η αποχώρηση από την ομάδα δηλώνεται πάλι στην Γερουσία, με αποστολή κατάλληλου μηνύματος. Η Γερουσία έχει επίσης τη δυνατότητα να διατάξει την *απομάκρυνση /αποβολή* ενός Ψηφοφόρου-μέλους της ομάδας (κάτι που αποτελεί το ανάλογο της διαγραφής από τους εκλογικούς καταλόγους).

Η απόφαση για αποβολή από την ομάδα, είναι δυνατόν να λαμβάνεται μετά από *αξιολόγηση* των επιδόσεων του Ψηφοφόρου, με στατιστική επεξεργασία της συνεισφοράς του (μέσω των ψήφων του) σε ψηφοφορίες που έλαβαν χώρα στο παρελθόν. Κάτι τέτοιο είναι πιθανόν να είναι επιθυμητό σε ορισμένες εφαρμογές που χρησιμοποιούν την Αρχιτεκτονική Διενέργειας Ψηφοφοριών, και μπορεί να θεωρηθεί -υπό προϋποθέσεις και ανάλογα με την εφαρμογή - πως συνεισφέρει στην αύξηση της "γνώσης" του συστήματος, καθώς έτσι

<sup>8</sup> Στην παρούσα εργασία δεν υπάρχει ενασχόληση με το πως αναδιοργανώνεται η ομάδα ψηφοφοριών σε περίπτωση κατάρρευσης του διαχειριστή της (δηλαδή της Γερουσίας). Η αρχιτεκτονική άλλωστε έχει αποστολή να χρησιμεύσει και ως "εργαλείο ψηφοφοριών" εντός άλλων αρχιτεκτονικών πρακτόρων. Ένας πράκτορας μιας τέτοιας ευρύτερης αρχιτεκτονικής θα είναι υπεύθυνος για την δημιουργία της Γερουσίας. Η απόφαση για το πού θα δημιουργηθεί μια νέα Γερουσία σε περίπτωση κατάρρευσης μιας υπάρχουσας, είναι θέμα που πρέπει να αποφασιστεί από τους πράκτορες της "ευρύτερης αρχιτεκτονικής", με εκλογή, ορισμό ή άλλες μεθόδους γνωστές από τη θεωρία που σχετίζεται με Καταμεμημένα Συστήματα [65].

υπάρχει η δυνατότητα να "συγκλίνουν" οι απόψεις των ψηφοφόρων. (Σε μια εφαρμογή, φερ' ειπείν, που χρησιμοποιεί την Αρχιτεκτονική Διενέργειας Ψηφοφοριών για επίλυση του προβλήματος της ανάκτησης ή της σύγκρισης εικόνων με βάση το περιεχόμενό τους, και όπου Ψηφοφόροι είναι αλγόριθμοι που εκφέρουν γνώμη για την ομοιότητα των εικόνων με βάση διαφορετικό γνώρισμα ο καθένας, η αποβολή ψηφοφόρων μπορεί να οδηγήσει σε επιλογή των "σημαντικών" για το πρόβλημα γνωρισμάτων της εικόνας.)

Κάθε φορά που υπάρχει μεταβολή της σύνθεσης της ομάδας πρακτόρων της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών, η Γερουσία ενημερώνει με κατάλληλο μήνυμα διαχείρισης τους πράκτορες που συμμετέχουν σε αυτή. Η Γερουσία φροντίζει να διαθέτει κατάλογο με τα μέλη της ομάδας, αλλά και με τα (πρώην) μέλη που έχουν πρόσφατα αποβληθεί (κανένας πράκτορας δε μένει επ' άπειρον στον τελευταίο κατάλογο, για να αποφευχθεί ο επ' άπειρον αποκλεισμός του).

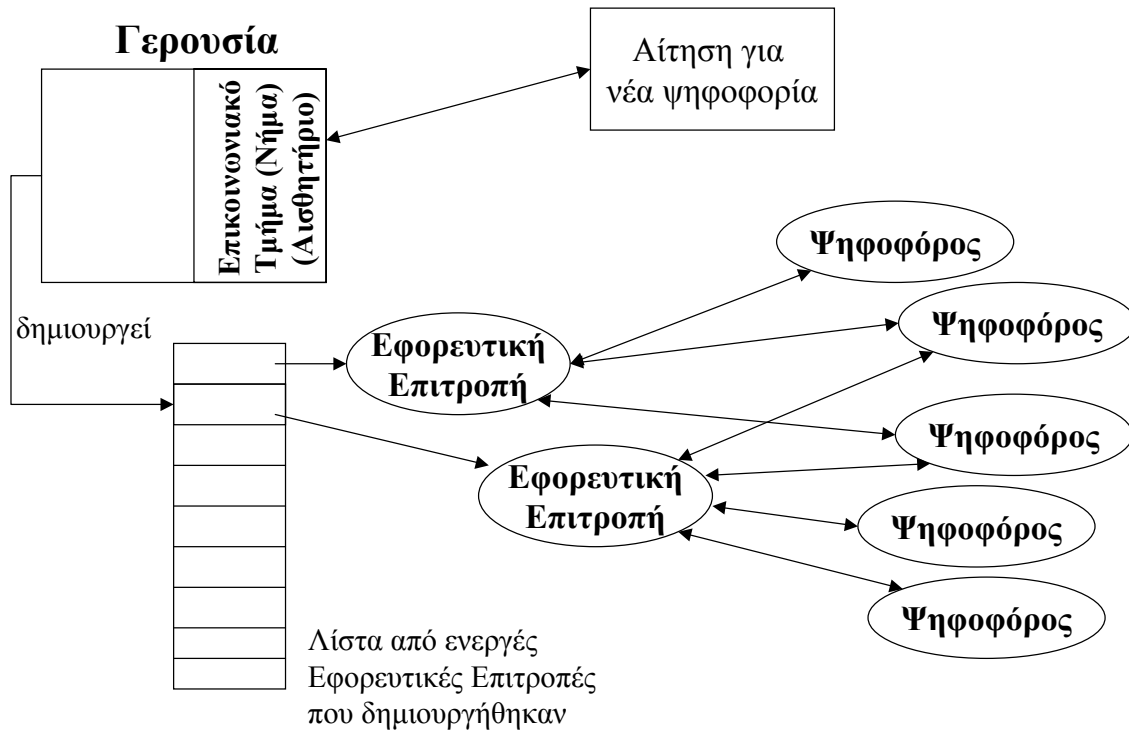
#### 4.6.1 Ψηφοφορία σε Ομάδα

Μια ψηφοφορία εντός της κοινωνίας πρακτόρων της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών γίνεται σε υπο-ομάδα ανά θέμα ψηφοφορίας, καθώς η έννοια της ψηφοφορίας για ένα θέμα προϋποθέτει την δημιουργία ομάδας ψηφοφόρων. Πολλές ψηφοφορίες είναι πιθανό να συμβαίνουν ταυτόχρονα, και πολλοί ψηφοφόροι είναι δυνατόν να συμμετέχουν - όπως αναφέρθηκε - σε περισσότερες από μία ταυτόχρονα διεξαγόμενες ψηφοφορίες. Υπάρχει έτσι η δυνατότητα για διενέργεια παράλληλων ψηφοφοριών που μπορεί να έχουν το ίδιο θέμα αλλά διαφορετική σύνθεση. Ακόμα, το αποτέλεσμα μιας ψηφοφορίας μπορεί να μεταφερθεί ως πιθανή υποψήφια πρόταση σε μια άλλη ψηφοφορία (μέσω λειτουργιών μετατροπής Αποτελέσματος σε Υποψήφιο, τις οποίες διαθέτει η Γερουσία και το ίδιο το Αποτέλεσμα), επιτρέποντας έτσι την θεώρηση διαφορετικών επιπέδων ψηφοφοριών που υπακούουν σε μια ιεραρχία.

Εδώ θα γίνει συνοπτικά αναφορά στον τρόπο δημιουργίας των υπο-ομάδων ανά ψηφοφορία (Σχήμα 9).

Η Γερουσία, ως υπεύθυνη για την προκύρηξη ψηφοφοριών (Voting Sessions), είναι υπεύθυνη για τον σχηματισμό υπο-ομάδων ψηφοφορίας. Η Γερουσία εμπεριέχει πολυνηματική λειτουργικότητα, που χειρίζεται με συγχρονισμένο τρόπο την δομή των ομάδων. Πιο συγκεκριμένα, υπάρχει ένα νήμα που περιμένει μηνύματα (ο "αισθητήρας" της Γερουσίας), και μόλις μια αίτηση για διενέργεια νέας ψηφοφορίας γίνει δεκτή, και καθοριστούν -ίσως δυναμικά- όλες οι παράμετροι της Ψηφοφορίας, δημιουργείται μια Εφορευτική Επιτροπή που θα είναι υπεύθυνη για την Ψηφοφορία, και τελικά θα μεταδώσει το αποτέλεσμα της στη Γερουσία. Τα μηνύματα που ανταλλάσσονται από τους

συμμετέχοντες στην συγκεκριμένη ψηφοφορία δεν απασχολούν την Γερουσία, αλλά τα διαχειρίζεται ο πράκτορας Εφορευτική Επιτροπή.



• Σχήμα 9 Δυναμική Δημιουργία Υπο-Ομάδων Ψηφοφοριών.

## 4.7 Ανταλλασσόμενα Μηνύματα

Τα μηνύματα που ανταλλάσσονται μεταξύ των βασικών οντοτήτων είναι τα ακόλουθα:

- *Μήνυμα Αίτησης Ψηφοφορίας* (RequestVotingSessionMessage)

Το μήνυμα αυτό αποστέλλεται στην Γερουσία και με αυτό ζητείται ο προγραμματισμός μιας Ψηφοφορίας για κάποιο συγκεκριμένο θέμα. Το μήνυμα περιλαμβάνει το σκοπό της Ψηφοφορίας, μια ποιοτική περιγραφή των χρονικών περιορισμών της Ψηφοφορίας, και πιθανά το Εκλογικό Σύστημα που θα χρησιμοποιηθεί και τους Υποψηφίους ή τις υποψήφιες εναλλακτικές προτάσεις προς ψήφιση.

- *Μήνυμα Προγραμματισμού Ψηφοφορίας* (VotingSessionSchedulingMessage)

Το μήνυμα αυτό αποστέλλεται από την Γερουσία προς όλους τους πράκτορες, πιθανούς Ψηφοφόρους ή Υποψηφίους, προς ενημέρωση για επικείμενη προγραμματισμένη Ψηφοφορία. Το μήνυμα αναφέρει το ποιά είναι αυτή η Ψηφοφορία (Voting Session), προσδιορίζοντας το θέμα της, το Εκλογικό Σύστημα που θα χρησιμοποιηθεί, τους χρονικούς περιορισμούς που ισχύουν για αυτήν και την Εφορευτική Επιτροπή που θα έχει την ευθύνη της συλλογής και καταμέτρησης

των ψηφοδελτίων. Έτσι, οι Ψηφοφόροι που επιθυμούν να ψηφίσουν στην ψηφοφορία μπορούν να αρχίσουν τις "νοητικές" διαδικασίες λήψης απόφασης ψήφου, παρόλο που η καθαυτή διαδικασία κατάθεσης των Ψηφοδελτίων δεν έχει αρχίσει. - οι δε Υποψήφιοι μπορούν να πραγματοποιούν ενέργειες "προεκλογικού" περιεχομένου (όπως για παράδειγμα αποστολή "προεκλογικού υλικού" που μπορεί να περιέχει "υποσχέσεις", ή και "δωροδοκίες" ψηφοφόρων), οι οποίες ποικίλλουν ανάλογα με την πολυπλοκότητα του γνωσιακού τους τμήματος.

- *Μήνυμα Προκήρυξης Κατάθεσης Υποψηφιοτήτων (AskForCandidatesMessage)*

Το μήνυμα αυτό αποστέλλεται από την Γερουσία προς τους πράκτορες, με σκοπό την εξεύρεση νέων Υποψηφίων για μια επικείμενη ψηφοφορία, και καθορίζει το θέμα της ψηφοφορίας. Η αποστολή του μηνύματος ορίζει την αρχή μιας χρονικής περιόδου "υποβολής υποψηφιοτήτων", στην οποία οι Υποψήφιοι μπορούν να καταθέτουν στην Γερουσία την εναλλακτική τους πρόταση (υποψηφιότητα). Το μήνυμα αποστέλλεται ανεξάρτητα από το αν έχουν καθοριστεί υποψήφιοι εναλλακτικές προτάσεις με το μήνυμα αίτησης ψηφοφορίας. Σε περίπτωση που μετά την πάροδο της "περιόδου υποβολής υποψηφιοτήτων" δεν έχει καθοριστεί καμμία υποψήφια εναλλακτική πρόταση (είτε με υποβολή υποψηφιότητας από ενεργή οντότητα - Υποψήφιο, είτε με αρχικό καθορισμό μέσω του μηνύματος αίτησης για ψηφοφορία), η ψηφοφορία δεν προκηρύσσεται. Αντίθετα, αν μετά το πέρας της "περιόδου υποβολής υποψηφιοτήτων" υπάρχουν υποψήφιοι προτάσεις, η Γερουσία αποστέλλει το μήνυμα προγραμματισμού ψηφοφορίας (VotingSessionSchedulingMessage).

- *Μήνυμα Κατάθεσης Υποψηφιότητας (CandidacyMessage)*

Το μήνυμα αυτό αποστέλλεται από τους Υποψηφίους προς τη Γερουσία με σκοπό την κατάθεση υποψηφιότητας. Περιέχει την υποψήφια εναλλακτική πρόταση του Υποψηφίου, μαζί με πληροφορία που αφορά την τοποθεσία στην οποία αυτός βρίσκεται.

- *Μήνυμα Έναρξης Εκλογικής Διαδικασίας (VoteCastingStartsMessage)*

Το μήνυμα αυτό αποστέλλεται από την Εφορευτική Επιτροπή προς όλους τους πράκτορες - πιθανούς Ψηφοφόρους, ώστε να τους ειδοποιήσει για την έναρξη της Ψηφοφορίας που θα διενεργηθεί υπ' ευθύνη του. Το μήνυμα αναφέρει το ποιά είναι αυτή η Ψηφοφορία και ποιά η Εφορευτική Επιτροπή στην οποία πρέπει να αποστέλλουν τα ψηφοδέλτιά τους οι Ψηφοφόροι.

- *Μήνυμα Ψήφου (VoteCastingMessage)*

Το μήνυμα αυτό αποστέλλεται από τον Ψηφοφόρο προς την Εφορευτική Επιτροπή με σκοπό την έκφραση της γνώμης του για το θέμα. Περιέχει το συμπληρωμένο Ψηφοδέλτιο.



- *Μήνυμα Εκλογικού Αποτελέσματος (ResultMessage)*

Το μήνυμα αυτό αποστέλλεται από την Εφορευτική Επιτροπή προς τη Γερουσία με σκοπό την ενημέρωσή της για το αποτέλεσμα της Ψηφοφορίας της οποίας η Εφορευτική Επιτροπή είχε την ευθύνη διεξαγωγής. Το μήνυμα περιέχει το Αποτέλεσμα της Ψηφοφορίας. Το ίδιο μήνυμα αποστέλλεται και από την Γερουσία προς την (πιθανά εξωτερική ως προς την Αρχιτεκτονική) οντότητα που είχε ζητήσει την διεξαγωγή της Ψηφοφορίας.

- *Μήνυμα Αίτησης Ένταξης στην Ομάδα (JoinGroupMessage)*

Το μήνυμα αυτό ανήκει στην κατηγορία των μηνυμάτων διαχείρισης ομάδας πρακτόρων. Αποστέλλεται από ένα πράκτορα (Ψηφοφόρο ή Υποψήφιο) προς τη Γερουσία που διαχειρίζεται την ομάδα πρακτόρων προκειμένου να ζητήσει την ένταξη του στην ομάδα. Το μήνυμα περιέχει πληροφορία για την τοποθεσία του αιτούντος πράκτορα.

- *Μήνυμα Αποδοχής Αίτησης Ένταξης (JoinAcceptedMessage)*

Το μήνυμα ανήκει στην κατηγορία των μηνυμάτων διαχείρισης ομάδας πρακτόρων. Αποστέλλεται από την Γερουσία προς έναν πράκτορα ως θετική απάντηση σε αίτησή του για ένταξη στην ομάδα πρακτόρων που διαχειρίζεται η Γερουσία. Το μήνυμα περιέχει πληροφορία για την σύνθεση της ομάδας, ώστε να είναι το νέο μέλος να μπορεί στο εξής να επικοινωνεί μετά από πρωτοβουλία του με τα υπόλοιπα.

- *Μήνυμα Αποβολής από την Ομάδα (KickoutMessage)*

Το μήνυμα ανήκει στην κατηγορία των μηνυμάτων διαχείρισης ομάδας πρακτόρων. Αποστέλλεται από την Γερουσία προς έναν πράκτορα για να τον ειδοποιήσει πως αποβλήθηκε από την ομάδα πρακτόρων που διαχειρίζεται η Γερουσία.

- *Μήνυμα Αποχώρησης (LeaveGroupMessage)*

Το μήνυμα ανήκει στην κατηγορία των μηνυμάτων διαχείρισης ομάδας πρακτόρων. Αποστέλλεται από έναν πράκτορα προς τη Γερουσία για να της δηλώσει την αποχώρησή του από την ομάδα πρακτόρων στην οποία ως τώρα ανήκε, και την οποία αυτή διαχειρίζεται.

- *Μήνυμα Ενημέρωσης Σύνθεσης Ομάδας (UpdateGroupMessage)*

Το μήνυμα ανήκει στην κατηγορία των μηνυμάτων διαχείρισης ομάδας πρακτόρων. Με το μήνυμα αυτό η Γερουσία αποστέλλει στα μέλη της ομάδας

πρακτόρων την οποία διαχειρίζεται τη σύνθεση της ομάδας μετά από μια μεταβολή της.

- *Μήνυμα Ερώτησης Γνώμης* (AskOpinionMessage)

Με το μήνυμα αυτό ένας Ψηφοφόρος ζητάει την άποψη ενός άλλου Ψηφοφόρου για ένα θέμα ψηφοφορίας. Το μήνυμα περιέχει το θέμα της ψηφοφορίας, και ένα πεδίο που παρέχει περαιτέρω πληροφορίες για αυτήν (λ.χ. τη λίστα με τους υποψηφίους - αν αναμένει απάντηση με μορφή ψηφοδέλιου, ή κάποια άλλη οντότητα την οποία αναμένει ως απάντηση).

- *Μήνυμα Παροχής Γνώμης* (GiveOpinionMessage)

Με το μήνυμα αυτό ένας Ψηφοφόρος ενημερώνει κάποιον άλλο για την άποψή του για ένα θέμα ψηφοφορίας. Το μήνυμα περιέχει το θέμα της ψηφοφορίας, και ένα πεδίο με την άποψη του Ψηφοφόρου. Στο πεδίο αυτό μπορεί να εμπεριέχεται ένα ψηφοδέλτιο που αυτός είχε συμπληρώσει σε παλαιότερη σχετική Ψηφοφορία, ή κάποια άλλη οντότητα η οποία του είχε ζητηθεί ως απάντηση.

- *Μήνυμα Αξιολόγησης Πράκτορα* (EvaluationMessage)

Το μήνυμα αυτό αποστέλλεται από τη Γερουσία προς ένα πράκτορα, με σκοπό να του ανακοινώσει τα αποτελέσματα της αξιολόγησης που του έγινε. Περιέχει το αποτέλεσμα της αξιολόγησης ("φύλλο αξιολόγησης"). Με βάση την παρούσα υλοποίηση, το αποτέλεσμα της αξιολόγησης κατατάσσει τον πράκτορα σε μια από τρεις "ποιοτικές" κατηγορίες: στην κατηγορία του "Ψηφοφόρου-Θύματος" (κατώτατη κατηγορία, που η κατάταξη σε αυτή συνεπάγεται την αποβολή από την ομάδα), στην κατηγορία του "Μέσου Ψηφοφόρου", ή στην κατηγορία του "Πολίτη" (που είναι η ανώτατη κατηγορία, και που κατάταξη σε αυτή μπορεί να συνεπάγεται την πρόσδοση ειδικού βάρους στην άποψη του ψηφοφόρου σε μελλοντικές εκλογές - αν αυτό επιτρέπεται από το εκλογικό σύστημα).

Κάθε μήνυμα αναφέρει τις διευθύνσεις (τα προσδιοριστικά ταυτότητας) του αποστολέα και του παραλήπτη του.

## Κεφάλαιο 5 Ζητήματα Υλοποίησης

Η υλοποίηση ενός οποιουδήποτε πληροφοριακού συστήματος απαιτεί να λαμβάνονται υπόψη πιθανές δυσκολίες και περιορισμοί, όπως η ετερογένεια λειτουργικών συστημάτων, μεταφραστών και διεπιφανειών χρήσης. Επίσης, πρέπει να λαμβάνεται μέριμνα ώστε οι επιλογές υλοποίησης να εξασφαλίζουν την πλήρωση των απαιτήσεων που αναφέρονται στο κεφάλαιο της Αρχιτεκτονικής, όπως η επεκτασιμότητα του συστήματος και η δυνατότητα διασύνδεσής του με άλλες εφαρμογές, καθώς και άλλων γενικών απαιτήσεων - όπως η δυνατότητα επαναχρησιμοποίησης του πηγαίου κώδικα και η ευκολία στην εύρεση και διόρθωση σφαλμάτων.

Το κεφάλαιο αυτό ασχολείται με ζητήματα που έχουν σχέση με την υλοποίηση της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών. Θα αναφερθούν και θα αιτιολογηθούν οι επιλογές για τη γλώσσα υλοποίησης και τον τρόπο επικοινωνίας, θα γίνει αναφορά στον τρόπο υλοποίησης των πρακτόρων, θα παρουσιαστούν τα εκλογικά συστήματα που υλοποιήθηκαν, και θα γίνει μια περιγραφή της ιεραρχίας προγραμματιστικών κλάσεων.

### 5.1 Επιλογή Γλώσσας Προγραμματισμού

Με δεδομένο το γεγονός πως η Αρχιτεκτονική που αναπτύχθηκε είναι μια αρχιτεκτονική πρακτόρων, και με δεδομένη τη σχέση του προγραμματισμού πρακτόρων (Agent Oriented Programming) με τον αντικειμενοστρεφή προγραμματισμό (Object Oriented Programming), φυσιολογική επιλογή ήταν η χρήση μιας αντικειμενοστρεφούς γλώσσας προγραμματισμού, που να επιτρέπει τη χρήση προηγμένων τεχνικών προγραμματισμού και την εφαρμογή οντοκεντρικών σχεδίων προγραμματισμού με φυσικό τρόπο [5][21].

Η γλώσσα που επιλέχθηκε ήταν η γλώσσα προγραμματισμού Java. Η Java επιτρέπει προγραμματισμό δικτύου (κάτι απαραίτητο για την αρχιτεκτονική μας), καθώς και άλλες σημαντικές δυνατότητες, όπως ανεξαρτησία από το λειτουργικό σύστημα, μεταφερσιμότητα κώδικα και δυνατότητα εκτέλεσης μέσω του Παγκόσμιου Ιστού, σε περίπτωση που οι πράκτορες υλοποιηθούν ως Java applets, και χρήση πολλών και συνεχώς επαυξανόμενων έτοιμων βιβλιοθηκών με διάφορες κλάσεις [2][14][83].

Η επιλογή της Java, η οποία συνεχώς εξελίσσεται, επιτρέπει την ενσωμάτωση και χρήση της αρχιτεκτονικής από συστήματα πρακτόρων που είναι αναπτυγμένα σε κάποια από τις συνεχώς εμφανιζόμενες έτοιμες πλατφόρμες ανάπτυξης καταναμημένων συστημάτων

πρακτόρων που χρησιμοποιούν Java, όπως η πλατφόρμα Grasshopper [75], η Voyager [78], ή η JATLite [71], αλλά και τη διασύνδεσή της με ετερογενείς εφαρμογές μέσω των σχετικών τεχνολογιών που υποστηρίζουν τη γλώσσα Java, όπως η τεχνολογία CORBA [70][76] ή η τεχνολογία Jini [72]. Οι αρχικές μάλιστα σκέψεις για την υλοποίηση περιελάμβαναν τη χρήση της επικοινωνιακής ειδικά λειτουργικότητας του Voyager ή του JATLite, αλλά τελικά επιλέχθηκε η χρήση της "καθαρής" γλώσσας Java για λόγους ταχύτητας, απλότητας, και μη επηρεασμού της αρχιτεκτονικής από ενδεχόμενους περιορισμούς των σχετικών πλατφορμών.

## 5.2 Επιλογές Επικοινωνίας

Για μια αρχιτεκτονική κατανεμημένων πρακτόρων, όπως αυτή που αναπτύχθηκε στα πλαίσια της παρούσας εργασίας, η χρησιμοποίηση μιας ευρείας αποδοχής Γλώσσας Επικοινωνίας Πρακτόρων για την επικοινωνία των πρακτόρων αποτελεί οπωσδήποτε θετικό στοιχείο, για λόγους συμπίεσης με τις τεχνολογικές εξελίξεις και για να είναι κατά το δυνατόν "ανοικτή" και επεκτάσιμη.

Η KQML (Knowledge Query and Manipulation Language), η οποία παρουσιάζεται αναλυτικά στο Παράρτημα IV, είναι η Γλώσσα Επικοινωνίας Πρακτόρων με την ευρύτερη σήμερα αποδοχή. Έχει χρησιμοποιηθεί σε πολλά projects με επιτυχία όσον αφορά την ολοκλήρωση εφαρμογών πρακτόρων και τη διαλειτουργικότητα μεταξύ τους. Το σημαντικό πλεονέκτημα που προκύπτει από τη χρήση της είναι ακριβώς η διαλειτουργικότητα και η επικοινωνία μεταξύ ετερογενών πρακτόρων, που πιθανά η ύπαρξή τους ή οι δυνατότητές τους δεν είναι εξ' αρχής γνωστές [17][18][19][73][84].

Όμως, η KQML δυστυχώς επιβαρύνει χρονικά σε μεγάλο βαθμό την επικοινωνία μεταξύ των πρακτόρων, λόγω των πολλών "διαστρωματώσεων" της (Παράρτημα IV: "Επίπεδα στην KQML") που αυξάνουν τον όγκο των μεταδιδόμενων μηνυμάτων. Με δεδομένη αυτή την επιβάρυνση, και το γεγονός ότι η αρχιτεκτονική που παρουσιάζεται στην παρούσα εργασία έχει περιορισμένες ανάγκες μετάδοσης μεταπληροφοριών ή πληροφοριών περιεχομένου τις οποίες θα μπορούσε να καλύψει η KQML με χρήση κάποιας γλώσσας αναπαράστασης γνώσης, δεν προκρίθηκε η επιλογή της για την κάλυψη των επικοινωνιακών αναγκών των πρακτόρων της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών.

Αντί για χρήση μηνυμάτων σε KQML, τα μηνύματα που ανταλλάσσουν οι πράκτορες της παρουσιαζόμενης αρχιτεκτονικής υλοποιήθηκαν ως ιεραρχία κλάσεων αντικειμένων σε "καθαρή" Java, και για τη μετάδοσή τους έγινε χρήση της σχετικής με χρήση του δικτύου λειτουργικότητας της γλώσσας (επικοινωνία με απευθείας χρήση sockets). Έτσι εξασφαλίζεται η βέλτιστη επίδοση σε ταχύτητα επικοινωνίας.

Παρ'όλαυτά, η KQML εύκολα μπορεί να ενσωματωθεί στην παρουσιαζόμενη αρχιτεκτονική στο μέλλον. Κάτι τέτοιο μπορεί απλά να γίνει με τη χρησιμοποίηση κλάσεων σε Java που να αποτελούν KQML "περιτυλίγματα" (wrappers) για τα ήδη υπάρχοντα μηνύματα, τα οποία θα εντάσσονται (ή μάλλον θα αποτελούν) το επίπεδο περιεχομένου (content level) του KQML μηνύματος. Μάλιστα, είναι δυνατή η χρήση έτοιμου κώδικα για KQML wrappers από το KQML επίπεδο της πλατφόρμας JATLite [71].

### 5.3 Πολυνηματικοί Πράκτορες

Όπως έχει ήδη αναφερθεί στο κεφάλαιο της Αρχιτεκτονικής, σε κάθε πράκτορα της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών μπορούν να διακριθούν τρία τμήματα: το τμήμα αίσθησης, που συλλέγει μηνύματα που απευθύνονται προς τον πράκτορα, το γνωσιακό, που ασχολείται με την επιτέλεση των "νοητικών" λειτουργιών του πράκτορα, και το τμήμα δράσης, που ασχολείται με την εκτέλεση των ενεργειών της οντότητας, όπως η αποστολή μηνυμάτων, ή η καταγραφή συμπεριφορών.

Σε επίπεδο υλοποίησης, κάθε ένα από αυτά τα τμήματα αποτελεί ένα αντικείμενο, με βάση τις αρχές του οντοκεντρικού προγραμματισμού [5]. Η συμπεριφορά κάθε ξεχωριστού αντικειμένου καθορίζεται από ένα τουλάχιστον νήμα ελέγχου (thread), ενώ ο ίδιος ο πράκτορας ξεκινά τη ζωή και τη δράση του ως μια προγραμματιστική κλάση-νήμα, από την οποία δημιουργούνται στη συνέχεια τα υπόλοιπα νήματα ελέγχου.

Ο κάθε πράκτορας είναι δηλαδή *πολυνηματικός*, κάτι που επιτρέπει την παράλληλη εκτέλεση επιμέρους λειτουργιών του. Η ύπαρξη ξεχωριστών νημάτων εξασφαλίζει και την δυνατότητα *αυτόνομης* δράσης καθενός από τα τμήματα, αλλά και του ίδιου του πράκτορα, καθώς το κάθε αντικείμενο έχει προγραμματιστικά τη δυνατότητα να αποφασίζει το ίδιο για το πότε και πώς θα εκτελέσει κάποια ενέργεια ή εργασία - ανάλογα με το ποιά είναι η εσωτερική του κατάσταση και η κατάσταση του περιβάλλοντός του, όπως αυτό γίνεται αντιληπτό. Επιπλέον, η επεκτασιμότητα της συνολικής αρχιτεκτονικής έχει όφελος από την χρήση πολλών αντικειμένων-νημάτων, καθώς έτσι δεν υπάρχει ανάγκη -σε περίπτωση που επιζητείται η αύξηση των δυνατοτήτων της- να επαναπρογραμματιστεί κάποια οντότητα (για επαύξηση των λειτουργιών της ή για επαναδιαμοιρασμό υπολογιστικού χρόνου στις υπόλοιπες οντότητες). Σε μια τέτοια περίπτωση, μπορεί απλά να προστεθεί στο σύστημα ένας πράκτορας ή ένα αντικείμενο που να επιτελεί την επιπλέον επιζητούμενη λειτουργικότητα, και να διαθέτει ξεχωριστό νήμα ελέγχου ώστε να δρα παράλληλα με τους ήδη υπάρχοντες πράκτορες ή αντικείμενα. Βέβαια, η δράση πολλών νημάτων ελέγχου πάνω σε κοινή -διαμοιραζόμενη- μνήμη είναι δυνατόν να δημιουργήσει προβλήματα συντονισμού και ταυτόχρονης πρόσβασης στο ίδιο τμήμα μνήμης (ή υπολογιστικό πόρο) από άνω της μίας οντότητες. Τα όποια όμως προβλήματα αντιμετωπίζονται με χρήση της

ενσωματωμένης στη γλώσσα Java λειτουργικότητα αποφυγής τέτοιου είδους προβλημάτων [14], η οποία βασίζεται σε τεχνικές γνωστές από τη θεωρία των λειτουργικών και καταναμημένων συστημάτων [65].

Ειδικά για το νήμα ελέγχου του γνωσιακού τμήματος του πράκτορα πρέπει να αναφερθεί ότι είναι σύνηθες να δημιουργεί ξεχωριστά αντικείμενα-νήματα στα οποία αναθέτει κάποια εργασία, και αναλαμβάνει την διαχείριση των ίδιων και των (πιθανών) αποτελεσμάτων τους. Ο οποιοσδήποτε χρήστης της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών μπορεί να δημιουργήσει το δικό του αντικείμενο-νήμα για το γνωσιακό τμήμα του κάθε πράκτορα της αρχιτεκτονικής, καθορίζοντας έτσι το βαθμό ευφυΐας του, και να το ενσωματώσει στην υπόλοιπη αρχιτεκτονική.

## 5.4 Εκλογικά Συστήματα που Υλοποιήθηκαν

Η επιλογή των εκλογικών συστημάτων προς υλοποίηση έγινε με κριτήρια α) την συχνότητα χρήσης τους στην καθημερινή ζωή και β) την ανάγκη να δοκιμαστεί ένα ικανοποιητικό σε μέγεθος σύνολο εκλογικών συστημάτων που τα μέλη του να ανήκουν σε διαφορετικές κατηγορίες εκλογικών συστημάτων. Επιλογή ήταν επίσης να επιδιωχθεί ώστε τα εκλογικά συστήματα προς υλοποίηση να επιδέχονται εύκολα τροποποιήσεις, ώστε να είναι δυνατή η κατασκευή παραλλαγών τους.

Με τα κριτήρια αυτά, τα εκλογικά συστήματα που υλοποιήθηκαν είναι τα παρακάτω:

- Μέθοδος Απόλυτης Πλειοψηφίας χωρίς έκφραση προτιμήσεων (Non-Preferential Majority Method).

Η μέθοδος αυτή επιλέγει ως νικήτρια την πρόταση που συγκεντρώνει την μοναδική προτίμηση της απόλυτης πλειοψηφίας των ψηφοφόρων. Ωστόσο, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Απόλυτης Πλειοψηφίας με έκφραση προτιμήσεων (Preferential Majority Method)

Ο κάθε ψηφοφόρος κατατάσσει τις υποψήφιες προτάσεις με βάση την προτίμησή του. Η μέθοδος επιλέγει ως νικήτρια την πρόταση που συγκεντρώνει την πρώτη προτίμηση της απόλυτης πλειοψηφίας των ψηφοφόρων. Αυτό δε σημαίνει ότι δεν μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων

υποψηφίων") ανάλογα με τη συνολική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Πλειοψηφίας χωρίς έκφραση προτιμήσεων (Non-preferential Plurality Method)

Η μέθοδος αυτή επιλέγει ως νικήτρια την πρόταση που συγκεντρώνει την μοναδική προτίμηση της απλής πλειοψηφίας των ψηφοφόρων. Υπάρχει βέβαια η δυνατότητα να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Πλειοψηφίας με έκφραση προτιμήσεων (Preferential Plurality Method)

Ο κάθε ψηφοφόρος κατατάσσει τις υποψήφιες προτάσεις με βάση την προτίμησή του. Η μέθοδος επιλέγει ως νικήτρια την πρόταση που συγκεντρώνει την πρώτη προτίμηση της απλής πλειοψηφίας των ψηφοφόρων. Ωστόσο, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Υπολογισμού Borda (Borda Count Method)

Η μέθοδος αυτή επιτρέπει στον κάθε ψηφοφόρο να κατατάξει τις  $n$  υποψήφιες εναλλακτικές προτάσεις σε μια σειρά κατάταξης. Η  $k$  θέση στη σειρά κατάταξης βαθμολογείται με βαθμό  $n-k$ . Όλες οι επιμέρους ανά ψηφοδέλτιο βαθμολογίες της κάθε υποψήφιας πρότασης αθροίζονται. Νικήτρια είναι η πρόταση με το μεγαλύτερο άθροισμα βαθμολογιών. Ωστόσο, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Επιλογής  $m$  από  $n$  (m-out-of-n Voting Method)

Στη μέθοδο αυτή νικήτρια είναι η υποψήφια πρόταση την οποία επιλέγουν (ως μοναδική προτίμηση ή ως πρώτη προτίμηση) τουλάχιστον  $m$  από τους  $n$  ψηφοφόρους. Μπορεί να βρεθούν πάνω από μία νικήτριες προτάσεις, αν  $m \leq 1/2 n$ . Αν η εφαρμογή το απαιτεί, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους.

- Μέθοδος Έκφρασης Έγκρισης (Approval Voting Method)

Στη μέθοδο αυτή οι ψηφοφόροι έχουν το δικαίωμα να επιλέξουν όσες από τις υποψήφιες προτάσεις εγκρίνουν, χωρίς να τις κατατάξουν σε σειρά προτίμησης. Η νικήτρια είναι η πρόταση που συγκεντρώνει την πλειοψηφία των εγκρίσεων. Βέβαια, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Μέθοδος Πλειοψηφικής Έκφρασης Έγκρισης (Plurality Approval Voting)

Η μέθοδος αυτή προτάθηκε από τους Pirjanian et al. [49] και αποτελεί συνδυασμό approval voting και ψηφοφορίας με έκφραση προτιμήσεων. Ο ψηφοφόρος επιλέγει όσες προτάσεις εγκρίνει, αλλά σε αυτές προσδίδει και αριθμητικά βάρη (από το 0 ως το 1). Για κάθε πρόταση υπολογίζεται το άθροισμα των βαρών και διαιρείται με τον αριθμό των ψηφοφόρων. Νικήτρια είναι η πρόταση με το μεγαλύτερο αποτέλεσμα. Ωστόσο, μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο.

- Ομοφωνία (Unanimity)

Η μέθοδος ψηφοφορίας κατά την οποία η νικήτρια πρόταση είναι αυτή η οποία έχει λάβει τη θετική ψήφο όλων των ψηφοφόρων. Στη συνηθέστερη περίπτωση η μέθοδος απαιτεί από τον ψηφοφόρο να εκφράσει μόνο μια προτίμηση (non-preferential method). Μπορεί να θεωρηθεί όμως και παραλλαγή της μεθόδου με έκφραση προτιμήσεων (είτε με λίστα απόλυτης σειράς προτίμησης, είτε με τοποθέτηση αριθμητικών βαρών), κατά την οποία να απαιτείται η λύση που επιλέγεται να έχει λάβει την πρώτη προτίμηση κάθε ψηφοφόρου.<sup>9</sup> Στην περίπτωση μη επίτευξης ομοφωνίας η ψηφοφορία κηρύσσεται άκυρη και επαναλαμβάνεται (για πεπερασμένο αριθμό επαναλήψεων). Η μέθοδος δεν αποκλείει το ότι μπορεί να δοθεί ως απάντηση ένα μεγαλύτερο σύνολο λύσεων ("νικηφόρων υποψηφίων") ανάλογα με τη συγκεντρωτική σειρά κατάταξής τους, αν η εφαρμογή απαιτεί κάτι τέτοιο. Σε αυτή την περίπτωση όμως, όλοι οι υποψήφιοι πρέπει να έχουν δώσει την ίδια σειρά κατάταξης.

<sup>9</sup> Δεν είναι καθόλου βέβαιο ότι ο ψηφοφόρος θα τοποθετήσει ως πρώτη επιλογή σε μια λίστα έκφρασης προτιμήσεων την υποψήφια πρόταση που θα επέλεγε αν ήταν αναγκασμένος να δώσει μοναδική απάντηση προτίμησης. Αυτό συμβαίνει γιατί ο ψηφοφόρος χρησιμοποιεί διαφορετικές στρατηγικές ανάλογα με το εκλογικό σύστημα που ισχύει, προκειμένου να αναδειχτεί η λύση που επιθυμεί, να καταπνιχθεί η λύση που απεύχεται, ή να επικρατήσει λύση που δεν είναι πρώτη επιλογή του αλλά όμως είναι πλέον πιθανό να επικρατήσει κλπ.



Για κάθε μια από τις παραπάνω μεθόδους (εκτός της ομοφωνίας) ή τις αναφερόμενες παραλλαγές τους, μπορούν εύκολα να θεωρηθούν - και να υλοποιηθούν - άλλες παραλλαγές με χρήση του δικαιώματος αρνησικυρίας (veto).

## 5.5 Ιεραρχία Κλάσεων

Η Αρχιτεκτονική υλοποιήθηκε σαν μια ιεραρχία από 66 Java κλάσεις, οι οποίες αντιστοιχούν στους πράκτορες, τις διάφορες άλλες οντότητες, και τις δομές που χρησιμοποιούνται. Έγινε χρήση των μηχανισμών κληρονομικότητας και αξιοποίησης αφηρημένων κλάσεων που παρέχει η γλώσσα, προκειμένου να υλοποιηθεί το οντοκεντρικό μοντέλο που σχεδιάστηκε, και να επιτευχθεί ο σωστός διαμοιρασμός - αλλά και η αξιοποίηση της κοινής - λειτουργικότητας.

Η ιεραρχία των κλάσεων της Αρχιτεκτονικής παρουσιάζεται στο Σχήμα 10.<sup>10</sup> Οι διάφορες κλάσεις που αντιστοιχούν στους πράκτορες (Γερουσία, Ψηφοφόρους, Εφορευτικές Επιτροπές και Υποψηφίους), κληρονομούν τη λειτουργικότητά τους από μια γενικότερη κλάση (την κλάση "Agent") που αντιστοιχεί στον πράκτορα-"πολίτη" (ο οποίος μπορεί άλλωστε στην καθημερινή ζωή να λειτουργήσει με τον τρόπο που προσδιορίζουν οι "εξειδικεύσεις" του που προαναφέρθηκαν).

Στο σχήμα φαίνεται πως όλοι οι πράκτορες της Αρχιτεκτονικής αποτελούν εξειδικεύσεις της κλάσης "Thread" της Java, κάτι που επιτρέπει τη δράση τους ως αυτόνομα νήματα. Αυτόνομα νήματα είναι και οι κλάσεις που αντιστοιχούν στο γνωσιακό τμήμα, και το τμήμα δράσης και αίσθησης των πρακτόρων (κλάσεις "CognitiveModule", "Receiver" και "Server"). Ακόμη, στο σχήμα φαίνεται η χρήση κληρονομικότητας για τη δημιουργία μιας ιεραρχίας από μηνύματα (υποκλάσεων της κλάσης "Message"), μιας ιεραρχίας από εκλογικά συστήματα (υποκλάσεων της κλάσης "VotingSystem") και μιας ιεραρχίας από κλάσεις που αντιστοιχούν σε "εξαιρέσεις" ("Exceptions") και χρησιμεύουν στην ανίχνευση δυσλειτουργιών κατά την χρησιμοποίηση της Αρχιτεκτονικής.

Εκτός από τις κλάσεις της Αρχιτεκτονικής, αναπτύχθηκε κι ένα σύνολο από προγραμματιστικές κλάσεις που χρησιμεύουν στην διασύνδεση των πρακτόρων με το χρήστη (κλάσεις διεπιφάνειας χρήσης), προκειμένου να διευκολύνεται η επίδειξη της Αρχιτεκτονικής. Είναι σαφές πως η διεπιφάνεια χρήσης της Αρχιτεκτονικής (Εικόνα 1) δε συνιστά τμήμα της Αρχιτεκτονικής, και μπορεί να αντικατασταθεί με οποιαδήποτε άλλη, ανάλογα με τις ανάγκες της εφαρμογής που αξιοποιεί την Αρχιτεκτονική Πρακτόρων για Διενέργεια Ψηφοφοριών.

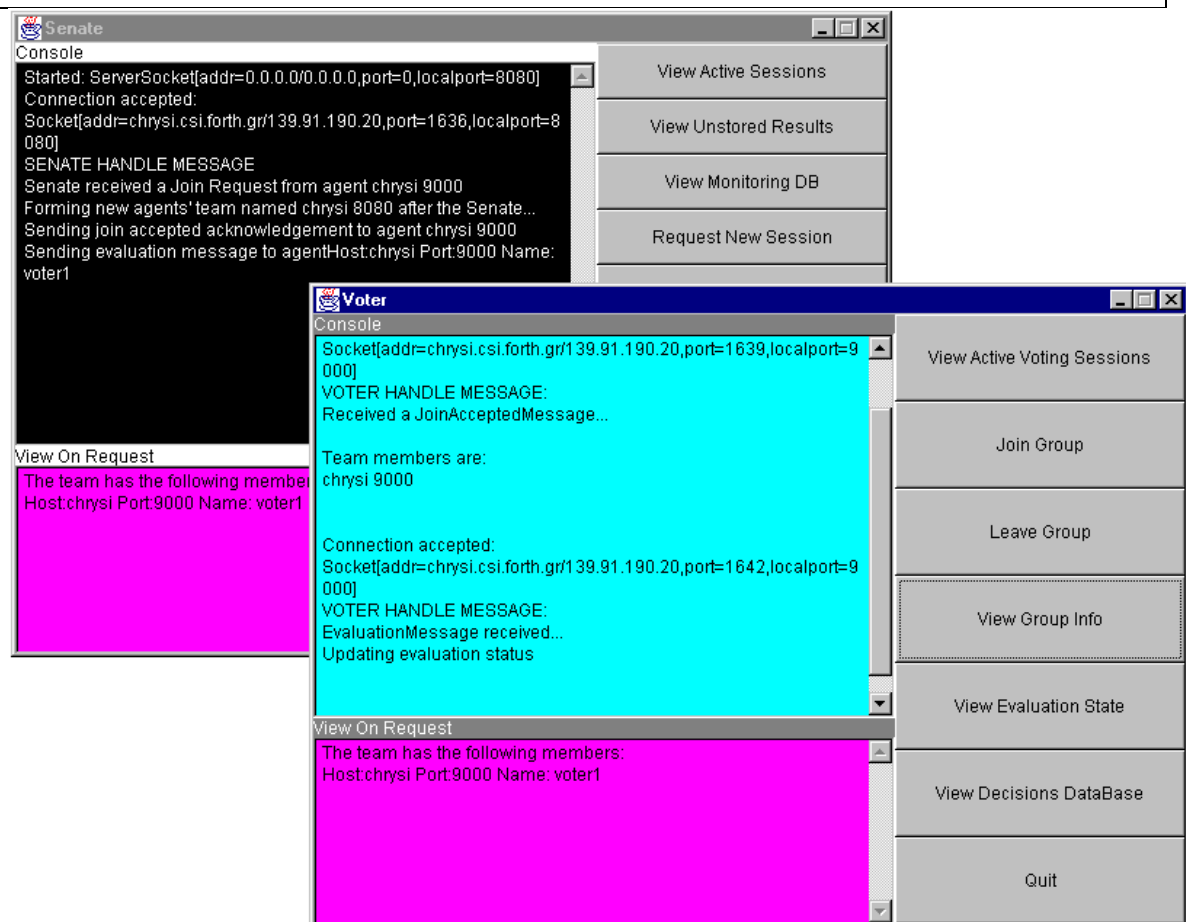
<sup>10</sup> Οι προγραμματιστικές κλάσεις της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών, οι διασυνδέσεις τους, τα γνωρίσματα και η λειτουργικότητά τους, παρουσιάζονται αναλυτικά στην ηλεκτρονική διεύθυνση : <http://www.csd.ucl.ac.uk/~gehalk/Master/Documentation/>.

- class java.lang.Object
  - class Voting.[AgentIdentifier](#) (implements java.io.Serializable)
  - class Voting.[ArithmeticValueKind](#) (implements java.io.Serializable)
  - class Voting.[Ballot](#) (implements java.io.Serializable)
  - class Voting.[BallotBox](#) (implements java.io.Serializable)
  - class Voting.[Constraint](#) (implements java.io.Serializable)
  - class Voting.[EvaluationSheet](#) (implements java.io.Serializable)
  - class Voting.[LocalLoadInfo](#)
  - class Voting.[Message](#) (implements java.io.Serializable)
    - class Voting.[AskForCandidatesMessage](#) (implements java.io.Serializable)
    - class Voting.[AskOpinionMessage](#)
    - class Voting.[CandidancyMessage](#)
    - class Voting.[EvaluationMessage](#)
    - class Voting.[GiveOpinionMessage](#)
    - class Voting.[JoinAcceptedMessage](#)
    - class Voting.[JoinGroupMessage](#)
    - class Voting.[KickoutMessage](#)
    - class Voting.[LeaveGroupMessage](#)
    - class Voting.[RequestVotingSessionMessage](#)
    - class Voting.[ResultMessage](#)
    - class Voting.[UpdateGroupMessage](#)
    - class Voting.[VoteCastingMessage](#)
    - class Voting.[VoteCastingStartsMessage](#)
    - class Voting.[VotingSessionSchedulingMessage](#)
  - class Voting.[NameValueCompare](#) (implements Voting.[Compare](#), java.io.Serializable)
  - class Voting.[NameValuePair](#) (implements java.io.Serializable)
  - class Voting.[Result](#) (implements java.io.Serializable)
  - class Voting.[SessionCommitteePair](#) (implements java.io.Serializable)
  - class Voting.[SessionInitiatorPair](#) (implements java.io.Serializable)
  - class Voting.[SessionResultMonitoring](#) (implements Voting.[Monitoring](#), java.io.Serializable)
  - class Voting.[SessionResultPair](#) (implements java.io.Serializable)
  - class Voting.[Team](#) (implements java.io.Serializable)
  - class java.lang.Thread (implements java.lang.Runnable)
    - class Voting.[Agent](#) (implements java.io.Serializable)
      - class Voting.[Candidate](#)
      - class Voting.[Senate](#)
      - class Voting.[Voter](#)
      - class Voting.[VotingCommittee](#)
    - class Voting.[CognitiveModule](#) (implements java.io.Serializable)
    - class Voting.[Receiver](#) (implements java.io.Serializable)
    - class Voting.[Server](#)
    - class Voting.[StatisticalEvaluation4Voters](#) (implements java.io.Serializable)
    - class Voting.[StatisticalEvaluation4Candidates](#) (implements java.io.Serializable)
- class java.lang.Throwable (implements java.io.Serializable)
  - class java.lang.Exception
    - class Voting.[ConstraintException](#)
    - class Voting.[ElectionsTimeExpiredException](#)
    - class Voting.[InvalidBallotException](#)
    - class Voting.[NoNewCommitteeException](#)
    - class Voting.[NoNewPairException](#)
    - class Voting.[NoNewSessionException](#)
  - class Voting.[TimeConstraintsType](#) (implements java.io.Serializable)
- class java.util.Vector (implements java.lang.Cloneable, java.io.Serializable)
  - class Voting.[SortVector](#)
    - class Voting.[CandidateValuePairsList](#)
    - class Voting.[VoterWeightPairsList](#)
  - class Voting.[VoterWeightCompare](#) (implements Voting.[Compare](#), java.io.Serializable)
  - class Voting.[VoterWeightPair](#) (implements java.io.Serializable)
  - class Voting.[VotingDecisionMonitoring](#) (implements Voting.[Monitoring](#), java.io.Serializable)
  - class Voting.[VotingSession](#) (implements java.io.Serializable)
  - class Voting.[VotingSystem](#) (implements java.io.Serializable)
    - class Voting.[ApprovalVoting](#)

- class Voting.[BordaCount](#)
- class Voting.[M\\_Out\\_Of\\_N\\_VotingMethod](#)
- class Voting.[NonPreferentialMajority](#)
- class Voting.[NonPreferentialPlurality](#)
- class Voting.[PluralityApprovalVoting](#)
- class Voting.[PreferentialMajority](#)
- class Voting.[PreferentialPlurality](#)
- class Voting.[Unanimity](#)

Interfaces : interface Voting.[Monitoring](#) & interface Voting.[Compare](#)

- Σχήμα 10 Η ιεραρχία υλοποιημένων Java κλάσεων.



- Εικόνα 1 Διεπιφάνειες χρήσης Γερουσίας και Ψηφοφόρου. Ο Ψηφοφόρος έχει μόλις γίνει μέλος της ομάδας πρακτόρων την οποία διαχειρίζεται η Γερουσία. Κατά την εισαγωγή του στην ομάδα έλαβε μηνύματα αποδοχής αποδοχής ένταξης, κι αρχικής αξιολόγησής του.

## Κεφάλαιο 6 Πειράματα

Την υλοποίηση της Αρχιτεκτονικής Πρακτόρων για τη Διενέργεια Ψηφοφοριών ακολούθησε η φάση των δοκιμών και της διενέργειας πειραμάτων για τον έλεγχο της λειτουργίας της και τη βελτίωσή της. Οι δοκιμές της Αρχιτεκτονικής έγιναν στα λειτουργικά συστήματα Solaris και Windows NT.

Πολλές ήταν οι ψηφοφορίες-πειράματα που διεξήχθησαν προκειμένου να επαληθευτεί η ορθή λειτουργία των διαφόρων τμημάτων και δυνατοτήτων της Αρχιτεκτονικής. Έγιναν πειράματα δυναμικής κατάθεσης υποψηφιοτήτων από πράκτορες-Υποψηφίους, και πειράματα που περιλάμβαναν την συμμετοχή ψηφοφόρων σε περισσότερες από μια ψηφοφορίες που διενεργούνταν ταυτόχρονα. Ελέγχθηκε η ορθή αποθήκευση των αποφάσεων των ψηφοφόρων και των εκλογικών αποτελεσμάτων από τους ψηφοφόρους και τη Γερουσία αντίστοιχα (αποθήκευση προηγούμενης εμπειρίας προς μελλοντική αξιοποίηση). Επίσης, έγινε έλεγχος της λειτουργίας των μηχανισμών αξιολόγησης πρακτόρων από τη Γερουσία, και διενεργήθηκαν "ακολουθίες" ψηφοφοριών κατά τις οποίες το εκλογικό αποτέλεσμα μιας ψηφοφορίας μετατρέπεται σε υποψήφιο για μια επόμενη (έλεγχος χρήσης της Αρχιτεκτονικής για διενέργεια ιεραρχικών ψηφοφοριών).

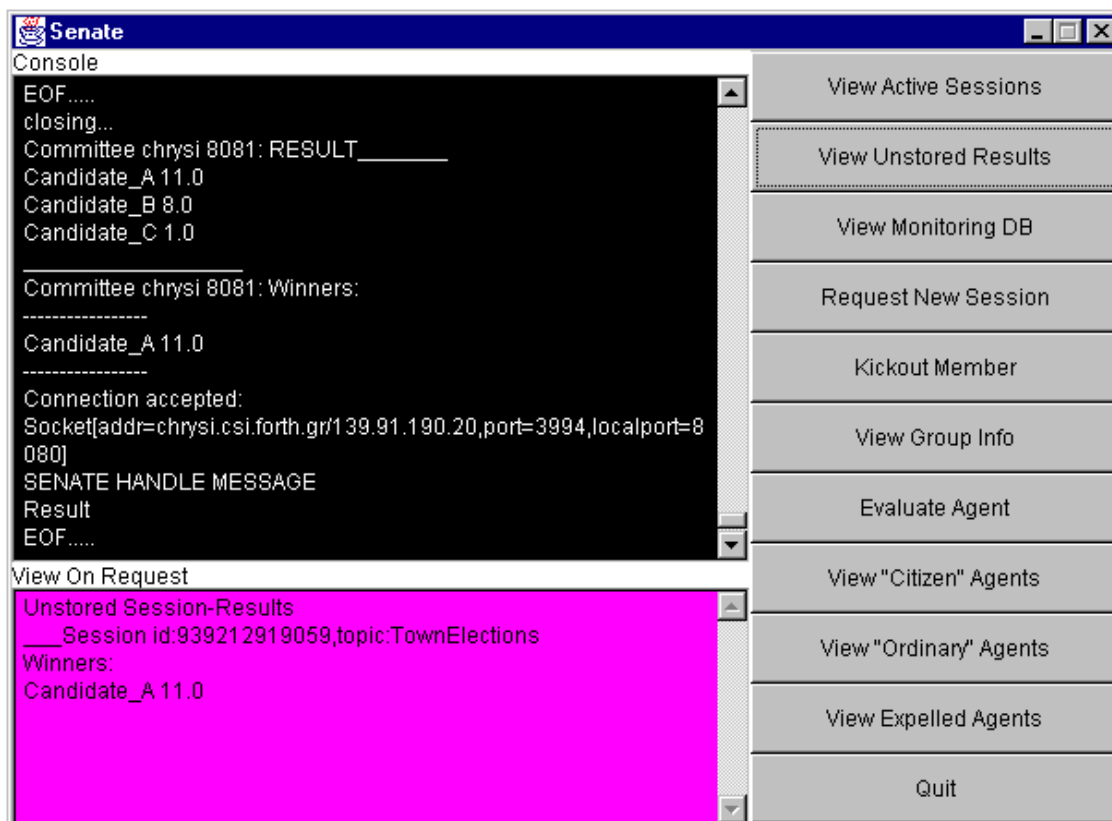
Ο έλεγχος της ορθότητας των υλοποιημένων αλγορίθμων εκλογικών συστημάτων ήταν ένα βασικό αντικείμενο των πειραμάτων. Η χρήση της Αρχιτεκτονικής για τη διενέργεια μάλιστα ψηφοφοριών με το ίδιο αντικείμενο και τους ίδιους ψηφοφόρους, αλλά με διαφορετικό εκλογικό σύστημα, ανέδειξε τη "σχετικότητα της δικαιοσύνης" κατά τις ψηφοφορίες, καθώς υπήρξαν παραδείγματα παραβίασης *Κριτηρίων Δικαιοσύνης* και "ανατροπής" εκλογικών αποτελεσμάτων.

Στη συνέχεια παρουσιάζονται δύο από τα πειράματα που διεξήχθησαν. Στο ένα από τα πειράματα επιδεικνύεται η αλλαγή εκλογικού αποτελέσματος με χρήση διαφορετικού εκλογικού συστήματος και η παραβίαση του *Κριτηρίου της Απόλυτης Πλειοψηφίας*, ενώ στο δεύτερο γίνεται χρήση ψηφοφορίας για να επιλεγεί μια ενέργεια από ένα ρομπότ.

Το πρώτο πείραμα αφορά την ψηφοφορία για μια υποθετική "εκλογή δημάρχου" σε μια μικρή πόλη, στην οποία καλούνται να λάβουν μέρος είκοσι (20) ψηφοφόροι οι οποίοι έχουν για τους υποψηφίους τις προτιμήσεις που αποτυπώνονται στον πίνακα που ακολουθεί.

	Υποψήφιος Α	Υποψήφιος Β	Υποψήφιος Γ
1η Επιλογή	11	8	1
2η Επιλογή	1	10	9
3η Επιλογή	8	2	10

Στον πίνακα φαίνεται ότι η πλειοψηφία των ψηφοφόρων προτιμά ως καλύτερη λύση τον Υποψήφιο Α. Πράγματι, η χρήση της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών για τη διενέργεια της ψηφοφορίας με χρήση της Μεθόδου Πλειοψηφίας χωρίς έκφραση προτιμήσεων έδωσε ως νικητή τον Υποψήφιο Α με 11 ψήφους, όπως φαίνεται στην Εικόνα 2.



- Εικόνα 2 Πείραμα δημοτικών εκλογών: Ανάδειξη του Υποψηφίου Α ως νικητή, με χρήση Μεθόδου Απλής Πλειοψηφίας. Στη γραφική διεπιφάνεια χρήσης φαίνεται το εκλογικό αποτέλεσμα που υπολόγισε η Εφορευτική Επιτροπή, και η μετάδοσή του στην Γερουσία.

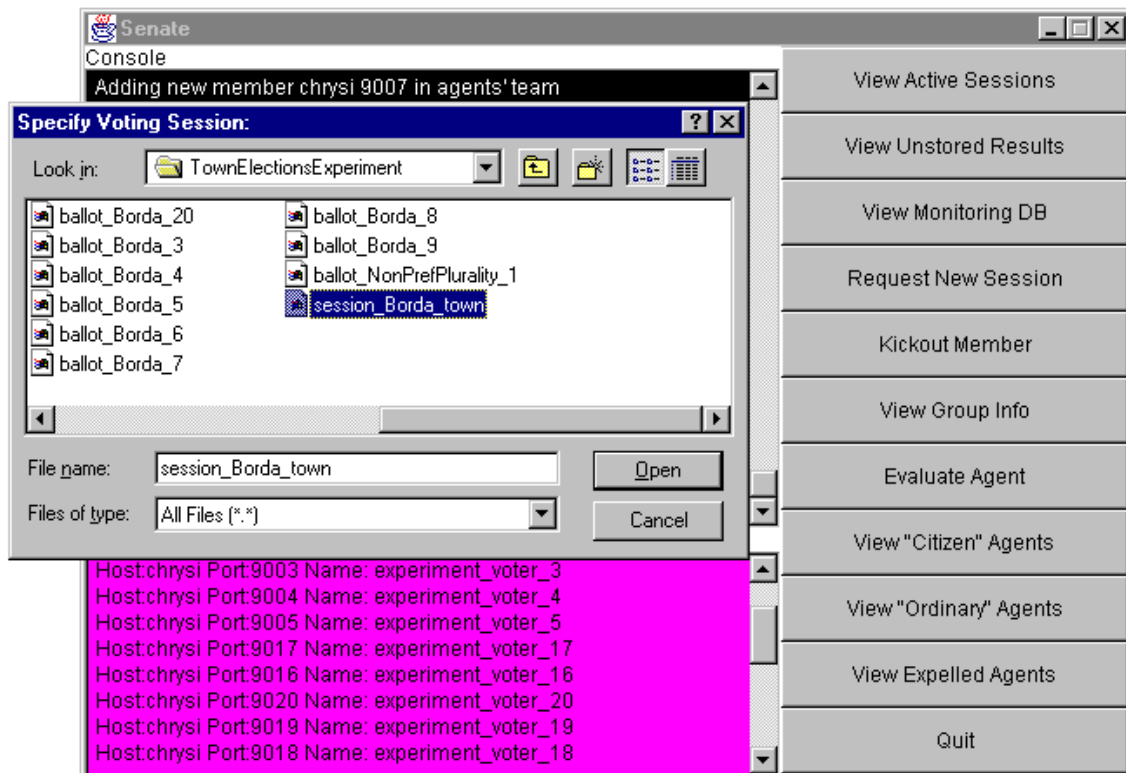
Ο προηγούμενος πίνακας δείχνει πως οι ψηφοφόροι που προτιμούν τον Υποψήφιο Α ως πρώτη επιλογή δεν αποτελούν απλά πλειοψηφία, αλλά και Απόλυτη Πλειοψηφία (11/20 ψηφοφόρους). Όταν όμως χρησιμοποιήθηκε Μέθοδος Υπολογισμού Borda (Εικόνα 3), που αξιολογούσε την πρώτη επιλογή με 3 μόρια, την δεύτερη με 2 μόρια, και την τρίτη με 1 μόριο, ο τελικός νικητής ήταν ο Υποψήφιος Β με 46 μόρια έναντι 43 του Α, όπως φαίνεται στην Εικόνα 4, και όπως ήταν αναμενόμενο από τους υπολογισμούς που εμφανίζει ο παρακάτω πίνακας:<sup>11</sup>

	Υποψήφιος Α	Υποψήφιος Β	Υποψήφιος Γ
1η Επιλογή (3 μόρια)	11	8	1
2η Επιλογή (2 μόρια)	1	10	9
3η Επιλογή (1 μόριο)	8	2	10
<b>Σύνολο μορίων</b>	<b>43</b>	<b>46</b>	<b>31</b>

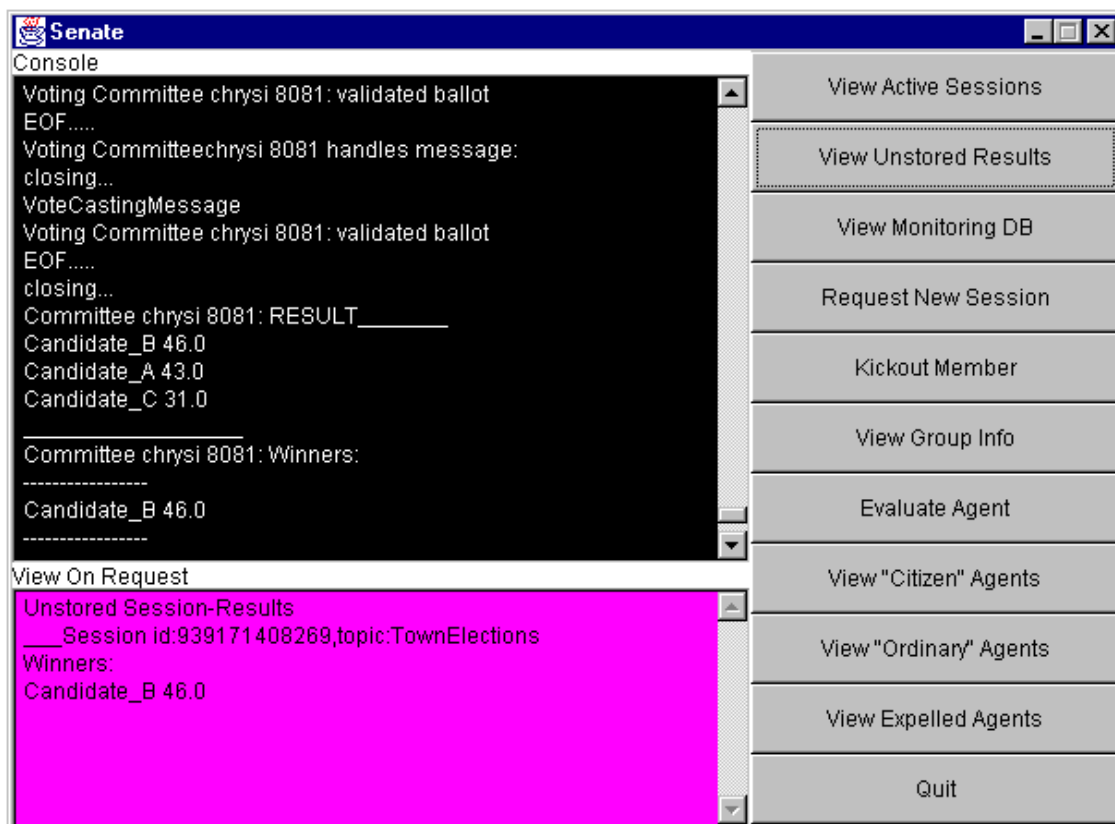
Η ανάδειξη του Β ως νικητή συνιστά μια περίπτωση παραβίασης του Κριτηρίου Δικαιοσύνης, η οποία ανιχνεύτηκε με τη χρήση της Αρχιτεκτονικής Πρακτόρων για τη Διενέργεια Ψηφοφοριών.

Στο δεύτερο πείραμα που θα αναφερθεί εδώ, διεξάγεται ψηφοφορία στην οποία χρησιμοποιείται η Μέθοδος Πλειοψηφικής Έγκρισης (όπως προτείνεται στο [49]) προκειμένου να γίνει επιλογή ενέργειας από ένα ρομπότ. Θεωρείται πως οι Ψηφοφόροι αντιστοιχούν σε *εκτιμητές αντικειμενικής λειτουργίας (objective function estimators)*, οι οποίοι εκτιμούν το πόσο καλά εξυπηρετούνται οι σκοποί τους από κάθε μία από τις παραμέτρους τους (δηλαδή τις ενέργειες - υποψήφιες εναλλακτικές προτάσεις).

<sup>11</sup> Τα ψηφοδέλτια που χρησιμοποιήθηκαν ήταν τα ακόλουθα (ο αριθμός αντιστοιχεί στον ψηφοφόρο, οι **διατεταγμένες** λίστες αντικατοπτρίζουν τις επιλογές του) : 1:{Α, Β, Γ}, 2:{Α, Β, Γ}, 3:{Α, Β, Γ}, 4:{Α, Β, Γ}, 5:{Α, Β, Γ}, 6:{Α, Β, Γ}, 7:{Α, Β, Γ}, 8:{Α, Β, Γ}, 9:{Α, Β, Γ}, 10:{Α, Γ, Β}, 11:{Α, Γ, Β}, 12:{Β, Γ, Α}, 13:{Β, Γ, Α}, 14:{Β, Γ, Α}, 15:{Β, Γ, Α}, 16:{Β, Γ, Α}, 17:{Β, Γ, Α}, 18:{Β, Γ, Α}, 19:{Β, Α, Γ}, 20:{Γ, Β, Α}



- Εικόνα 3 Πείραμα δημοτικών εκλογών: Επιλογή του συστήματος Borda Count για τη διεξαγωγή της ψηφοφορίας.

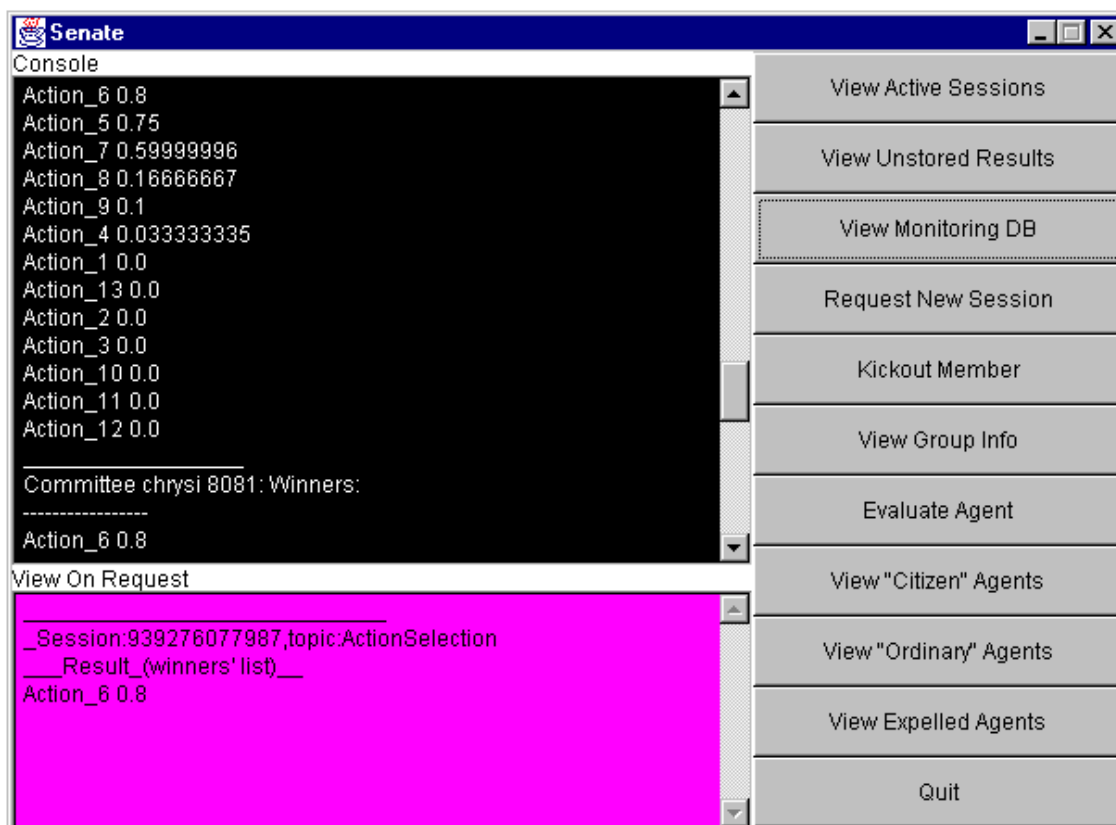


- Εικόνα 4 Πείραμα δημοτικών εκλογών: Ανάδειξη του B ως νικητή με το εκλογικό σύστημα Borda Count.

Στο πείραμα θεωρήθηκαν τρεις ψηφοφόροι και 13 υποψήφιες εναλλακτικές ενέργειας προς επιλογή. Οι επιλογές βαρών των ψηφοφόρων έγιναν με βάση τον παρακάτω πίνακα, οι τιμές του οποίου επιλέχθηκαν ώστε να είναι σε συμφωνία με αντίστοιχες τιμές που εμφανίζονται σε πείραμα που αναφέρεται στο [49]:

Ενέργειες →	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13
Ψηφοφόρος Α	0	0	0	0.1	0.25	0.9	1	0.2	0	0	0	0	0
Ψηφοφόρος Β	0	0	0	0	1	0.7	0.5	0.3	0.3	0	0	0	0
Ψηφοφόρος Γ	0	0	0	0	1	0.8	0.3	0	0	0	0	0	0

Με βάση τη Μέθοδο Πλειοψηφικής Έκφρασης Έγκρισης, και με δεδομένα τα παραπάνω βάρη για τις ενέργειες - υποψήφιες εναλλακτικές προτάσεις, αναμενόμενο πριν το πείραμα ήταν να επιλεγεί η ενέργεια #6 με συγκεντρωτικό βάρος 0.8, έναντι 0.75 της ενέργειας #5. Πράγματι, αυτό ήταν το τελικό αποτέλεσμα της ψηφοφορίας που έγινε με χρήση της Αρχιτεκτονικής Πρακτόρων για τη Διενέργεια Ψηφοφοριών (Εικόνα 5).



• Εικόνα 5 Πείραμα Επιλογής Ενέργειας: Η ενέργεια #6 είναι νικήτρια. Το ζεύγος Ψηφοφορία-Αποτέλεσμα έχει αποθηκευτεί στην βάση της Γερουσίας.



## Κεφάλαιο 7 Συμπεράσματα και Μελλοντική Εργασία

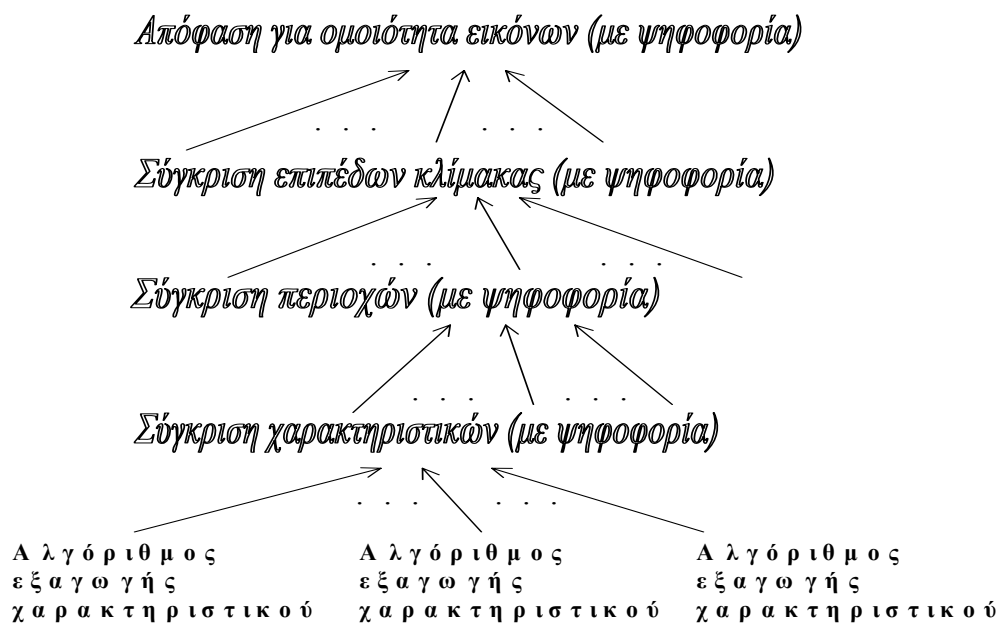
Η ψηφοφορία είναι ίσως η πλέον διαδεδομένη διαδικασία επιλογής αποφάσεων από κοινωνίες. Οι σύγχρονες τεχνολογικές εξελίξεις έχουν κάνει δυνατή την τέλεση ψηφοφοριών με ηλεκτρονικό τρόπο, ενώ η έννοια της ψηφοφορίας είναι οικεία και στην Επιστήμη Υπολογιστών, καθώς χρησιμεύει στην επίλυση προβλημάτων - όπως η διασφάλιση της συνέπειας κατανεμημένων συστημάτων αρχείων ή η επιλογή ενεργειών κατά την πλοήγηση ρομπότ - τα οποία παρουσιάζονται σε διάφορους τομείς της.

Η έννοια της ψηφοφορίας είναι αρκετά πολύπλοκη, καθώς το γεγονός πως στην πραγματικότητα είναι αδύνατον να βρεθεί μια απλή, "δίκαιη" και συνεπής διαδικασία για τον προσδιορισμό του αποτελέσματος μιας εκλογής που αφορά περισσότερους από δυο υποψηφίους ("Θεώρημα του Ανέφικτου του Arrow"), οδήγησε στην ανάπτυξη της Θεωρίας Ψηφοφοριών, που ασχολείται με τη μελέτη των διάφορων μηχανισμών ψηφοφορίας. Η Επιστήμη Υπολογιστών έχει ως τώρα χρησιμοποιήσει αρκετά απλά σχήματα ψηφοφοριών, παραγνωρίζοντας σε κάποιο βαθμό την πολυπλοκότητα της Θεωρίας Ψηφοφοριών, και το γεγονός πως ο ορισμός εκλογικού συστήματος και κατάλληλων εκλογικών διαδικασιών είναι βασικοί παράγοντες που επηρεάζουν το αποτέλεσμα μιας ψηφοφορίας.

Η μεταπτυχιακή αυτή εργασία είχε ως αντικείμενο την ανάλυση, το σχεδιασμό και την υλοποίηση μιας αρκετά γενικής αρχιτεκτονικής, που να επιτρέπει τη διενέργεια ψηφοφοριών με χρήση οποιουδήποτε εκλογικού συστήματος. Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι μια αρχιτεκτονική πολλών συνεργαζόμενων κατανεμημένων πρακτόρων λογισμικού, που σχεδιάστηκε με βάση τις αρχές της οντοκεντρικής σχεδίασης συστημάτων. Οι πράκτορες που συναπαρτίζουν την Αρχιτεκτονική, αντιπροσωπεύουν πραγματικές οντότητες, και ενσωματώνουν στοιχεία της Θεωρίας Ψηφοφοριών. Οι οντότητες της Αρχιτεκτονικής εντοπίστηκαν ως σημαντικές για τη διενέργεια ψηφοφοριών μετά από μελέτη της Θεωρίας Ψηφοφοριών και ενός γενικού παραδείγματος ψηφοφορίας από τον πραγματικό κόσμο. Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών είναι μια επεκτάσιμη αρχιτεκτονική, που μπορεί να χρησιμοποιηθεί ως υποδομή τέλεσης ψηφοφοριών από αρχιτεκτονικές πρακτόρων, ή άλλες εφαρμογές.

Η Αρχιτεκτονική Διενέργειας Ψηφοφοριών θα μπορούσε να ενσωματωθεί σε ένα σύστημα που κάνει χρήση ψηφοφοριών για την σύγκριση και ανάκτηση εικόνων με βάση το περιεχόμενό τους. Η ανάπτυξη ενός τέτοιου συστήματος αποτέλεσε το αρχικό κίνητρο της μεταπτυχιακής αυτής εργασίας, και η ιδέα για αυτό προήλθε από την ψυχολογική θεωρία

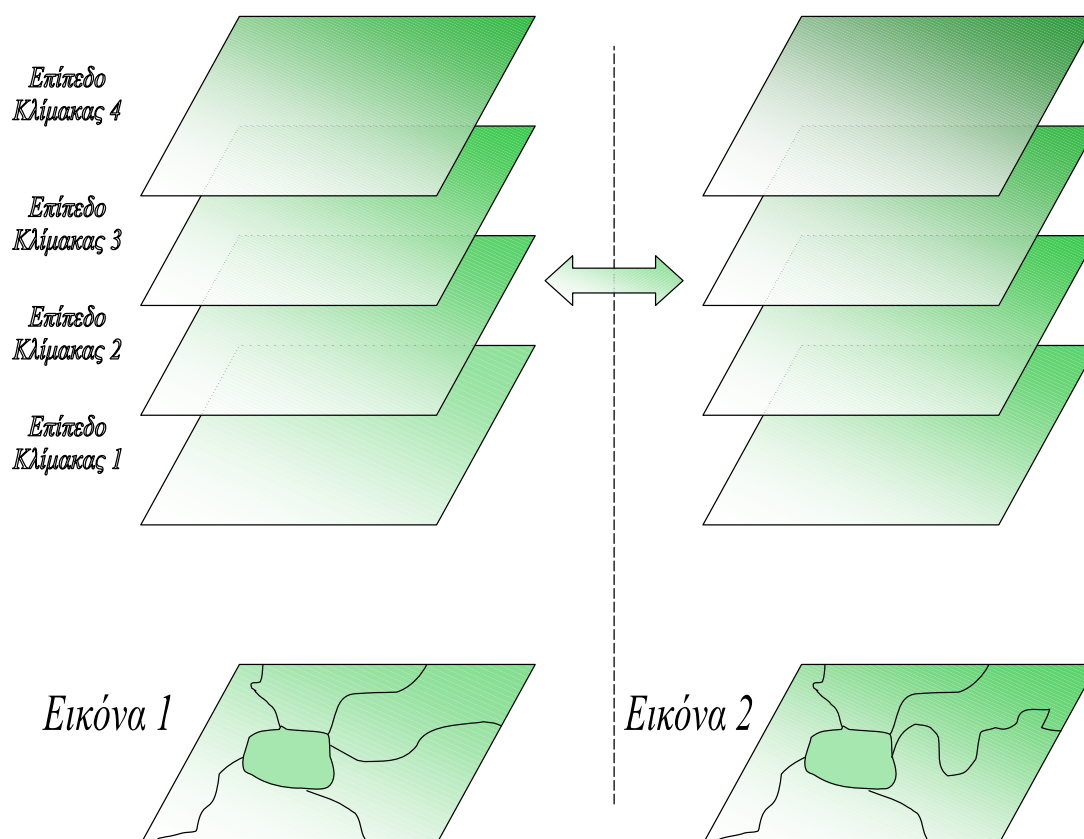
του Πανδαιμονίου, που προσπαθεί να εξηγήσει την οπτική αντίληψη του ανθρώπου κάνοντας την υπόθεση της ύπαρξης μιας ιεραρχίας μονάδων γνώσης, οι οποίες, κατά κάποιον τρόπο, "ψηφίζουν" για την ισχύ των προσλαμβανόμενων ερεθισμάτων τους.



• Σχήμα 11 Πιθανή ιεραρχία επιπέδων ψηφοφοριών για απόφαση περί ομοιότητας εικόνων.

Θα μπορούσε λοιπόν να αναπτυχθεί ένα σύστημα ανάκτησης εικόνων με βάση το περιεχόμενό τους, το οποίο να χρησιμοποιεί για τη λήψη απόφασης περί ομοιότητας εικόνων μια ιεραρχία ψηφοφοριών μεταξύ πρακτόρων-ψηφοφόρων, οι οποίοι θα διαφέρουν ως προς το γνωσιακό τους τμήμα, ανάλογα με το επίπεδο της ιεραρχίας στο οποίο ανήκουν.

Μια τέτοια πιθανή ιεραρχία για σύγκριση δύο εικόνων, η οποία θα μπορούσε να υποστηριχθεί από την Αρχιτεκτονική Διενέργειας Ψηφοφοριών, παρουσιάζεται στο Σχήμα 11. Η ιεραρχία χρησιμοποιεί ψηφοφορίες για τη σύγκριση χαρακτηριστικών των εικόνων, περιοχών των εικόνων, και επιπέδων κλίμακας εντός ενός χώρου κλιμάκων (scale space). Στο κατώτερο επίπεδο ψηφοφοριών, οι ψηφοφόροι μπορούν να βασίζονται σε γνώση προερχόμενη από αλγορίθμους εξαγωγής χαρακτηριστικών των εικόνων. Στη συνέχεια, η ομοιότητα χαρακτηριστικών μπορεί να οδηγήσει στην εξαγωγή συμπερασμάτων για την ομοιότητα ολόκληρων περιοχών (τμημάτων) των εικόνων, σε δεύτερο επίπεδο. Οι συγκριτές-ψηφοφόροι στο επίπεδο της ιεραρχίας που αφορά τη σύγκριση κλιμάκων, τέλος, μπορεί να ψηφίζουν για ομοιότητα μεταξύ διαφορετικών επιπέδων κλιμάκων των εικόνων (Σχήμα 12). Γενικά, το κάθε επίπεδο μπορεί να χρησιμοποιεί τα αποτελέσματα των προηγούμενων επιπέδων ψηφοφορίας ως τροφοδότες των γνωσιακών τμημάτων των ψηφοφόρων του, και στο τέλος της ιεραρχίας λαμβάνεται η απόφαση για ομοιότητα ή μή των εικόνων.



- Σχήμα 12 Σύγκριση τμηματοποιημένων εικόνων εντός του χώρου κλιμάκων. Τα επίπεδα κλίμακας των δύο εικόνων (όχι απαραίτητα τα αντίστοιχα) συγκρίνονται ανά δύο μεταξύ τους. Οι συγκριτές μπορεί να ψηφίζουν για ομοιότητα, στο επίπεδο της ιεραρχίας ψηφοφοριών που αφορά τη σύγκριση κλιμάκων.

Η χρήση της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών για ένα πληροφοριακό σύστημα ανάκτησης εικόνων με βάση το περιεχόμενό τους, παρέχει το πλεονέκτημα της δυνατότητας χρήσης διαφορετικών εκλογικών συστημάτων, προκειμένου να συναχθούν πειραματικά συμπεράσματα σχετικά με το πώς μπορεί να γίνεται ο καταλληλότερος συγκερασμός των διάφορων πληροφοριών που υπάρχουν σχετικά με τις εικόνες, ώστε να λαμβάνονται οι ορθότερες το δυνατόν αποφάσεις σχετικά με την ομοιότητα. Ένα άλλο πλεονέκτημα της χρήσης της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών για ένα τέτοιο σύστημα, είναι το ότι αυτή επιτρέπει την εξαγωγή συμπερασμάτων σχετικά με το βαθμό σημασίας των διάφορων χαρακτηριστικών της εικόνας (ή και των υπόλοιπων -ανωτέρου επιπέδου- πληροφοριών σχετικά με αυτή), καθώς μπορεί να παρακολουθείται το πώς συμπεριφέρονται οι αντίστοιχοι ψηφοφόροι σε μια ακολουθία από ψηφοφορίες. Στην εξαγωγή τέτοιων συμπερασμάτων μπορεί να βοηθήσει η χρήση εκλογικών συστημάτων που επιτρέπουν την απόδοση βάρους από τους ψηφοφόρους στις επιλογές τους, αλλά και εκλογικών συστημάτων που επιτρέπουν την απόδοση διαφορετικού βάρους στα ψηφοδέλτια διαφορετικών ψηφοφόρων (weighted systems).

Πέρα από την ανάπτυξη συστήματος για ανάκτηση εικόνων με βάση το περιεχόμενό τους, υπάρχουν αρκετές επεκτάσεις και βελτιώσεις που μπορεί να αποτελούν τμήματα μελλοντικής εργασίας.

Η επιλογή και υλοποίηση κατάλληλων αλγορίθμων για τη στατιστική επεξεργασία των αποτελεσμάτων των εκλογών και των υπόλοιπων καταγραφών των αποφάσεων των πρακτόρων, ώστε να χρησιμοποιούνται για την αξιολόγηση και αυτοαξιολόγηση των πρακτόρων, είναι μια βελτίωση ουσιαστικής σημασίας για τη λειτουργία του συστήματος στα πλαίσια μιας εφαρμογής.<sup>12</sup> Η ενσωμάτωση των αλγορίθμων στην αρχιτεκτονική είναι εύκολη, λόγω της χρήσης οντοκεντρικού σχεδιασμού (και οντοκεντρικής γλώσσας υλοποίησης). Πιθανότατα, βέβαια, η επιλογή των αλγορίθμων πρέπει να γίνεται σε σχέση με το ποιά είναι η εφαρμογή που χρησιμοποιεί την Αρχιτεκτονική Διενέργειας Ψηφοφοριών.

Η μελλοντική εργασία για την Αρχιτεκτονική Διενέργειας Ψηφοφοριών μπορεί να περιλαμβάνει την επαύξηση των δυνατοτήτων της Γερουσίας, ώστε αυτή να είναι ικανή να επιλέγει μετά από σειρά συλλογισμών, το Εκλογικό Σύστημα που είναι πλέον κατάλληλο για την τέλεση μιας Ψηφοφορίας που έχει συγκεκριμένες παραμέτρους. Επίσης, μια ενδιαφέρουσα επέκταση των δυνατοτήτων της Γερουσίας θα μπορούσε να είναι η κατασκευή ενός νέου Εκλογικού Συστήματος. Οι Υποψήφιοι θα μπορούσαν επίσης, στα πλαίσια μελλοντικής εργασίας, να αποκτήσουν επιπλέον δυνατότητες "ενεργητικής δράσης", όπως την κατασκευή και αποστολή προεκλογικού υλικού (ο καθορισμός της φύσης του οποίου αποτελεί ένα ενδιαφέρον πρόβλημα). Σε κάθε περίπτωση, η αρχιτεκτονική μπορεί να ενσωματώσει εύκολα οποιεσδήποτε άλλες βελτιώσεις και επεκτάσεις αφορούν το γνωσιακό τμήμα των πρακτόρων (το οποίο είναι άλλωστε απλό να αλλάζει εξ'ολοκλήρου, αν η εφαρμογή που χρησιμοποιεί την Αρχιτεκτονική Διενέργειας Ψηφοφοριών απαιτεί κάτι τέτοιο).

Τέλος, η χρησιμοποίηση της KQML ως γλώσσας επικοινωνίας των πρακτόρων της Αρχιτεκτονικής, παρόλο που δεν προκρίθηκε στην παρούσα εργασία για λόγους που εξηγήθηκαν στην ενότητα τη σχετική με τις επιλογές επικοινωνίας, είναι πιθανό να αποτελέσει αντικείμενο μελλοντικής εργασίας. Κάτι τέτοιο μπορεί να γίνει εφόσον κριθεί πως επιβάλλεται λόγω αυξημένων απαιτήσεων διαλειτουργικότητας με πράκτορες που χρησιμοποιούν KQML, ή λόγω ύπαρξης αυξημένης ανάγκης μετάδοσης πληροφοριών περιεχομένου (λ.χ. γνώσης και γνώμης των πρακτόρων), τις οποίες θα μπορούσε να καλύψει η KQML με χρήση κάποιας γλώσσας αναπαράστασης γνώσης (πιθανότατα της KIF). Έχει ήδη αναφερθεί πως σε μια τέτοια περίπτωση η KQML εύκολα μπορεί να ενσωματωθεί στην

<sup>12</sup> Επί του παρόντος χρησιμοποιείται ένας απλός αλγόριθμος αξιολόγησης των πρακτόρων, που βασίζεται στην υπέρβαση αυθαίρετων ποσοστίων "κατωφλίων" επιτυχούς επιλογής "πρώτων νικητών" (υπερψήφισης δηλαδή των τελικών νικητών) από τους Ψηφοφόρους, προκειμένου να κατατάσσονται αυτοί μια από τις τρεις κατηγορίες αξιολόγησης των πρακτόρων. Αντίστοιχα, η κατάταξη των Υποψηφίων σε κατηγορίες γίνεται με βάση αυθαίρετα ποσοστά επιλογής της πρότασής τους επί του συνόλου των εκλογών στις οποίες έχουν λάβει μέρος.

αρχιτεκτονική, απλά με τη χρησιμοποίηση Java κλάσεων ως "περιτυλίγματα" KQML των υπαρχόντων μηνυμάτων, τα οποία θα αποτελούν το επίπεδο περιεχομένου του KQML μηνύματος. Παρ'όλα αυτά, η χρήση της KQML και μιας γλώσσας αναπαράστασης γνώσης πιθανά θα καθιστούσε άχρηστη την ύπαρξη των μηνυμάτων AskOpinionMessage και GiveOpinionMessage, καθώς είναι μηνύματα μετάδοσης περιεχομένου (γνώμης /πεποιθήσεων των πρακτόρων), το οποίο μπορεί να κωδικοποιείται άμεσα από την κατάλληλη επικοινωνιακή πράξη (performative) της KQML και τη γλώσσα αναπαράστασης γνώσης.

## Παράρτημα Ι : Γνωρίσματα και Λειτουργίες Βασικών Οντοτήτων

---

Στο Παράρτημα αυτό παρουσιάζονται τα γνωρίσματα και οι κυριότερες λειτουργίες των πρακτόρων και των βασικότερων οντοτήτων της Αρχιτεκτονικής Διενέργειας Ψηφοφοριών. Οι διάφορες κλάσεις πρακτόρων κληρονομούν από την κλάση "Agent" την κατάλληλη λειτουργικότητα για αποστολή, λήψη και διαχείριση μηνυμάτων, καθώς και για την διενέργεια γνωσιακών λειτουργιών.

Μια πλήρης παρουσίαση όλων των υλοποιημένων Java κλάσεων της Αρχιτεκτονικής, γίνεται στην διεύθυνση "<http://www.csd.uch.gr/~gehalk/Master/Documentation/>".

### Γερουσία

#### *Γνωρίσματα*

*senate\_id* : *AgentIdentifier*

Προσδιοριστικό ταυτότητας της Γερουσίας.

*list\_of\_active\_sessions* : *Vector*

Λίστα με τις Ψηφοφορίες που έχει δημιουργήσει η Γερουσία.

*list\_of\_committees* : *Vector*

Λίστα με τις Εφορευτικές Επιτροπές που έχει ορίσει η Γερουσία.

*list\_of\_ses\_com\_pairs* : *Vector*

Λίστα με ζεύγη Ψηφοφοριών και αντίστοιχων Εφορευτικών Επιτροπών.

*list\_of\_session\_result\_pairs* : *Vector*

Λίστα με ζεύγη Ψηφοφοριών και αντίστοιχων Αποτελεσμάτων.

*list\_of\_session\_initiator\_pairs* : *Vector*

Λίστα με ζεύγη Ψηφοφοριών και των αντίστοιχων οντοτήτων που ζήτησαν την διεξαγωγή τους.

*list\_of\_citizen\_status\_agents* : *Vector*

Λίστα πρακτόρων που έχουν αξιολογηθεί ως "Πολίτες".

*list\_of\_sheep\_status\_agents* : *Vector*

Λίστα πρακτόρων που αξιολογήθηκαν ως ευρισκόμενοι στη "μέση κατάσταση" ψηφοφόρου.

*list\_of\_expelled\_agents* : *Vector*

Λίστα πρακτόρων που πρόσφατα αποβλήθηκαν, αφού έλαβαν τη χειρίστη αξιολόγηση.

*session\_result\_monitoring\_base\_name* : *String*

Το όνομα της βάσης όπου αποθηκεύονται τα Αποτελέσματα των Ψηφοφοριών που διενεργούνται.

*team : Team*

Η ομάδα πρακτόρων την οποία διαχειρίζεται η Γερουσία.

*cognitive\_module : CognitiveModule*

Το γνωσιακό τμήμα του πράκτορα.

*senate\_ui : Object*

Αναφορά στη διεπιφάνεια χρήσης η οποία είναι πιθανό να συνδέεται με τον πράκτορα.

### **Λειτουργίες**

*newVotingSession (voting\_reason : String, voting\_system : VotingSystem, constraints : Constraint) : votingSession*

Δημιουργία νέας Ψηφοφορίας μετά από κατάλληλο εξωτερικό ερέθισμα, και εισαγωγή της στη λίστα Ψηφοφοριών.

*createVotingCommittee (voting\_reason : String, voting\_system : VotingSystem, time\_constraints : Constraint, alternatives\_list : Vector, ballot : Ballot, voting\_committee\_id : AgentIdentifier, vot\_session : VotingSession, candidate\_selection\_threshold : int) : VotingCommittee*

Δημιουργία μιας νέας Εφορευτικής Επιτροπής για την εποπτεία συγκεκριμένης ψηφοφορίας. Εισαγωγή της Επιτροπής και του ζεύγους Ψηφοφορίας-Επιτροπής στις σχετικές λίστες που διαχειρίζεται η Γερουσία.

*createConstraints (time\_constraint : TimeConstraintsType, pre\_elections\_constraint : TimeConstraintsType, voting\_session : VotingSession) : Constraint*

Ποσοτική αποτίμηση των ποιοτικά προσδιορισθέντων χρονικών περιορισμών μιας Ψηφοφορίας. Συσχέτιση των υπολογισθέντων περιορισμών με τη δεδομένη Ψηφοφορία.

*constructBallot (alternatives\_list : Vector, type\_of\_arithmetic\_value : ArithmeticValueKind, voting\_session : VotingSession) : Ballot*

Κατασκευή ενός Ψηφοδέλτιου για συγκεκριμένη Ψηφοφορία, με βάσει μια λίστα υποψηφίων εναλλακτικών προτάσεων και τον τύπο των αριθμητικών πεδίων τιμών για το Ψηφοδέλτιο.

*scheduleVotingSession (alternatives\_list : Vector, ballot : Ballot, time\_constraints : Constraint, voting\_reason : String, voting\_system : VotingSystem, voting\_ses : VotingSession, voting\_committee : VotingCommittee)*

Κατασκευή και αποστολή κατάλληλου μηνύματος για την ενημέρωση των πρακτόρων (πιθανών Ψηφοφόρων ή Υποψηφίων) για τον προγραμματισμό μιας Ψηφοφορίας.

*chooseVotingSystem (voting\_purpose : String) : VotingSystem*

Η Γερουσία, μετά από συμπερασματική διαδικασία και βασισμένη στο σκοπό της ψηφοφορίας, επιλέγει ένα Εκλογικό Σύστημα από μια υπάρχουσα βάση με Εκλογικά Συστήματα.

*produceVotingSystem (voting\_purpose : String) : VotingSystem*

Κατασκευή νέου Εκλογικού Συστήματος, μετά από συμπερασματική / νοητική διαδικασία.

*insertSessionIntoSessionsList (voting\_session : VotingSession) : boolean*

Εισαγωγή Ψηφοφορίας στη λίστα Ψηφοφοριών που διαχειρίζεται η Γερουσία.

*insertCommitteeIntoList (voting\_committee : VotingCommittee) : boolean*

Εισαγωγή Εφορευτικής Επιτροπής στη λίστα Εφορευτικών Επιτροπών που διαχειρίζεται η Γερουσία.

*insertPairIntoList (pair : SessionCommitteePair) : boolean*

Εισαγωγή ζεύγους Ψηφοφορίας - Εφορευτικής Επιτροπής στη σχετική λίστα που διαχειρίζεται η Γερουσία.

*insertPairIntoList (voting\_session : VotingSession, voting\_committee : VotingCommittee) : boolean*

Δημιουργία και εισαγωγή ζεύγους Ψηφοφορίας - Εφορευτικής Επιτροπής στη σχετική λίστα που διαχειρίζεται η Γερουσία.

*insertSessionInitiatorPair( pair : SessionInitiatorPair) : boolean*

Εισαγωγή ζεύγους Ψηφοφορίας και οντότητας που την ζήτησε στην αντίστοιχη λίστα.

*removeCommitteeFromList(vot\_com : VotingCommittee ) : boolean*

Λειτουργία εξαγωγής εφορευτικής επιτροπής από την αντίστοιχη λίστα.

*removeSessionCommitteePairFromList(pair : SessionCommitteePair) : boolean*

Λειτουργία εξαγωγής ζεύγους Ψηφοφορίας-Εφορευτικής Επιτροπής από την αντίστοιχη λίστα.

*removeSessionFromList(inactive\_ses\_id : int) : boolean*

Λειτουργία εξαγωγής της Ψηφοφορίας με προσδιοριστικό inactive\_ses\_id από την αντίστοιχη λίστα.

*removeSessionInitiatorPair(int session\_id) : boolean*

Εξαγωγή ζεύγους Ψηφοφορίας και οντότητας που την ζήτησε από την αντίστοιχη λίστα.

*getMatchingSession (voting\_committee : VotingCommittee) : VotingSession*

Επιστρέφει την Ψηφοφορία που αντιστοιχεί στη δεδομένη Εφορευτική Επιτροπή.

*getMatchingCommittee (voting\_session : VotingSession) : VotingCommittee*

Επιστρέφει την Εφορευτική Επιτροπή που αντιστοιχεί στη δεδομένη Ψηφοφορία.

*monitorSessionResults(session\_result\_monitoring\_base\_name : String)*

Ενημέρωση κατάλληλης βάσης με ζεύγη Ψηφοφοριών και αντίστοιχων Αποτελεσμάτων (που έχουν διαμνηυθεί στη Γερουσία).

*handleMessage(message : Message)*

Διαχείριση λαμβανομένου μηνύματος.

*sendMessage(msg : Message, receiver : AgentIdentifier)*

Αποστολή μηνύματος σε πράκτορα.

*sendMessage(msg : Message, receivers\_list : Vector)*

Αποστολή μηνύματος σε λίστα από πράκτορες.

## **Εφορευτική Επιτροπή**

### ***Γνωρίσματα***

*result : Result*

Το Αποτέλεσμα που εξάγει η Εφορευτική Επιτροπή.

*senate : Senate*

Η Γερουσία που δημιούργησε την Εφορευτική Επιτροπή.

*committee\_identifier : AgentIdentifier*

Το προσδιοριστικό ταυτότητας της Εφορευτικής Επιτροπής.



*constraints* : *Constraint*

Οι (χρονικοί) περιορισμοί που ισχύουν στην Ψηφοφορία που εποπτεύει η Εφορευτική Επιτροπή.

*voting\_session* : *VotingSession*

Η Ψηφοφορία που διεξάγεται υπό την εποπτεία της Εφορευτικής Επιτροπής.

*candidate\_selection\_threshold* : *int*

Ο αριθμός των "νικηφόρων" υποψηφίων προτάσεων που συμπεριλαμβάνονται στο Αποτέλεσμα που εξάγει η Εφορευτική Επιτροπή.

*ballot\_box* : *BallotBox*

Η Κάλπη που χρησιμοποιεί η Εφορευτική Επιτροπή.

*alternative\_value\_pairs\_list* : *CandidateValuePairsList*

Λίστα με ζεύγη υποψηφίων εναλλακτικών προτάσεων, και των αντίστοιχων τιμών που συνολικά αντιστοιχούνται σε αυτές, κατά την καταμέτρηση με βάση το εκλογικό σύστημα.

*alternatives\_list* : *Vector*

Λίστα των υποψηφίων εναλλακτικών προτάσεων

*team\_of\_voters* : *Team*

Η ομάδα πρακτόρων την οποία διαχειρίζεται η Γερουσία.

*cognitive\_module* : *CognitiveModule*

Το γνωσιακό τμήμα του πράκτορα.

*committee\_ui* : *Object*

Αναφορά στη διεπιφάνεια χρήσης η οποία είναι πιθανό να συνδέεται με τον πράκτορα.

## **Λειτουργίες**

*VotingCommittee* (*alternatives\_list* : *Vector*, *ballot* : *Ballot*, *time\_constraints* : *Constraint*, *voting\_reason* : *String*, *voting\_system* : *VotingSystem*, *senate* : *Senate*, *committee\_identifier* : *AgentIdentifier*, *voting\_session* : *VotingSession*, *candidate\_selection\_threshold* : *int*)

Κατασκευαστής της Εφορευτικής Επιτροπής. Με αυτόν η Επιτροπή δέχεται από την Γερουσία τις παραμέτρους της Ψηφοφορίας προς διεξαγωγή.

*formStartVotingSessionMsg* (*voting\_session* : *VotingSession*)

Η λειτουργία αυτή χρησιμοποιείται για την κατασκευή και την αποστολή μηνύματος που αναγγέλλει την έναρξη της Ψηφοφορίας την οποία εποπτεύει η Εφορευτική Επιτροπή.

*acceptValidBallot* (*vote\_casting\_msg* : *VoteCastingMessage*) : *boolean*

Εξαγωγή Ψηφοδελτίου από μήνυμα ψήφου, έλεγχος εγκυρότητάς του και εισαγωγή του στην Κάλπη (αν όντως είναι έγκυρο).

*resultExtraction* (*voting\_system* : *VotingSystem*, *ballot\_box* : *BallotBox*, *session* : *VotingSession*) : *Result*

Λειτουργία εξαγωγής του Αποτελέσματος, βασισμένη στο Εκλογικό Σύστημα, τα περιεχόμενα της Κάλπης, και στον αριθμό των "νικηφόρων" υποψηφίων προτάσεων για αυτήν την εκλογή.

*informSenateOnResult* (*senate* : *Senate*, *result* : *Result*)

Ενημέρωση της Γερουσίας για το Αποτέλεσμα, με κατασκευή και αποστολή κατάλληλου μηνύματος.

*checkConstraints (constraints : Constraint)*

Ελέγχει τους περιορισμούς της Ψηφοφορίας, ώστε να αναγγείλει το χρόνο έναρξης και να αντιληφθεί το χρόνο λήξης της Ψηφοφορίας

*getVotingSession () : VotingSession*

Επιστρέφει την Ψηφοφορία την οποία εποπτεύει η Εφορευτική Επιτροπή.

*getVotingCommitteeId () : AgentIdentifier*

Επιστρέφει το προσδιοριστικό ταυτότητας της Επιτροπής.

*getResult () : Result*

Επιστρέφει το εκλογικό Αποτέλεσμα της Ψηφοφορίας που εποπτεύει η Εφορευτική Επιτροπή.

*getConstraints () : Constraint*

Επιστρέφει τους περιορισμούς που ισχύουν στην Ψηφοφορία που εποπτεύει η Επιτροπή.

*getCandidateSelectionThreshold () : int*

Επιστρέφει τον αριθμό των "νικηφόρων" υποψηφίων εναλλακτικών προτάσεων που ισχύει για την Ψηφοφορία που εποπτεύεται από την Εφορευτική Επιτροπή.

*xformR2C(result2xform : Result, candidate\_id : AgentIdentifier) : Candidate*

Συνάρτηση μετατροπής της πρώτης νικήτριας επιλογής του Αποτελέσματος μιας ψηφοφορίας σε Υποψήφιο (που καταθέτει την αντίστοιχη με την επιλογή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*xformR2C(result2xform : Result) : Candidate*

Συνάρτηση μετατροπής της πρώτης νικήτριας επιλογής του Αποτελέσματος μιας ψηφοφορίας σε Υποψήφιο (που καταθέτει την αντίστοιχη με την επιλογή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*xformR2VectorOfC(Result result2xform):Vector*

Συνάρτηση μετατροπής όλων των νικητών (αν αυτοί είναι περισσότεροι από έναν) που αναφέρονται στο Αποτέλεσμα μιας ψηφοφορίας, σε Υποψηφίους (που καταθέτει ο καθένας την αντίστοιχη με τον κάθε νικητή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*handleMessage(message : Message)*

Διαχείριση λαμβανομένου μηνύματος.

*sendMessage(msg : Message, receiver : AgentIdentifier)*

Αποστολή μηνύματος σε πράκτορα.

*sendMessage(msg : Message, receivers\_list : Vector)*

Αποστολή μηνύματος σε λίστα από πράκτορες.

## Ψηφοφόρος

### Γνωρισμάτα

*senate : Senate*

Η Γερουσία η οποία συναλλάσσεται με τον Ψηφοφόρο.

*voter\_id : AgentIdentifier*

Προσδιοριστικό ταυτότητας του Ψηφοφόρου.

*active\_voting\_sessions\_list* : *Vector*

Λίστα με τις Ψηφοφορίες στις οποίες συμμετέχει ο Ψηφοφόρος.

*evaluation\_state* : *EvaluationSheet*

Χαρακτηρισμός του Ψηφοφόρου με βάση την αξιολόγηση που του έχει γίνει.

*team* : *Team*

Η ομάδα στην οποία εντάσσεται ο Ψηφοφόρος.

*ballots2cast* : *Vector*

Λίστα από συμπληρωμένα, έτοιμα προς αποστολή Ψηφοδέλτια, για τις Ψηφοφορίες στις οποίες συμμετέχει ο Ψηφοφόρος.

*voting\_decision\_monitoring\_base\_name*

Το όνομα της βάσης όπου αποθηκεύονται οι αποφάσεις (τα τελικώς αποστέλλομενα Ψηφοδέλτια) του Ψηφοφόρου.

*cognitive\_module* : *CognitiveModule*

Το γνωσιακό τμήμα του πράκτορα.

*voter\_ui* : *Object*

Αναφορά στη διεπιφάνεια χρήσης η οποία είναι πιθανό να συνδέεται με τον πράκτορα.

## **Λειτουργίες**

*vote* (*ballot* : *Ballot*, *voting\_committee* : *VotingCommittee*)

Ο Ψηφοφόρος κατασκευάζει και αποστέλλει στην Εφορευτική Επιτροπή μήνυμα με το συμπληρωμένο Ψηφοδέλτιο.

*perceiveVotingSession* (*voting\_session* : *VotingSession*)

Λειτουργία για την ανάλυση και την κατανόηση των πληροφοριών που σχετίζονται σε μια πρόσφατα γνωστοποιημένη Ψηφοφορία.

*perceiveVotingSystem* (*voting\_system* : *VotingSystem*)

Λειτουργία για την ανάλυση και την κατανόηση των πληροφοριών που σχετίζονται με το Εκλογικό Σύστημα που δίδεται ως όρισμα.

*checkForParticipation*(*voting\_reason* : *String*, *voting\_constraints* : *Constraint*) : *boolean*

Λειτουργία λήψης απόφασης για συμμετοχή ή μη στην ψηφοφορία για τον σκοπό *voting\_reason*.

*askOpinion* (*voters\_list* : *Vector*, *voting\_reason* : *String*, *alternatives* : *Vector*, *voting\_system* : *VotingSystem*)

Λειτουργία για την δημιουργία και αποστολή μηνύματος για την επερώτηση της γνώμης / άποψης ενός ή περισσοτέρων Ψηφοφόρων πάνω σε ένα θέμα, παρέχοντας πληροφορίες για το Εκλογικό Σύστημα και τις υποψήφιες εναλλακτικές προτάσεις.

*insertSessionInList* (*voting\_session* : *VotingSession*) : *boolean*

Εισάγει μια Ψηφοφορία στη λίστα Ψηφοφοριών.

*getSession* (*voting\_reason* : *String*) : *VotingSession*

Επιστρέφει την Ψηφοφορία με συγκεκριμένο θέμα, από την λίστα Ψηφοφοριών στις οποίες συμμετέχει ο Ψηφοφόρος.

*identifySelf* (*voter\_id* : *AgentIdentifier*, *receiver\_id* : *AgentIdentifier*, *local\_load\_info* : *LocalLoadInfo*)

Λειτουργία που χρησιμοποιείται για την κατασκευή και αποστολή μηνύματος που να κάνει γνωστές τις συνθήκες ύπαρξης του Ψηφοφόρου.

*monitorVotingDecision(voting\_decision\_monitoring\_base\_name : String, casted\_ballot : Ballot)*

Ο Ψηφοφόρος φροντίζει για την αποθήκευση του Ψηφοδελτίου που απέστειλε σε κατάλληλη βάση.

*reachVotingDecision(voting\_reason : String, alternatives\_list : Vector, voting\_session : VotingSession, ArithmeticValueKind type\_of\_value) : Ballot*

Λειτουργία που ενεργοποιεί κατάλληλο γνωσιακό νήμα για λήψη απόφασης σχετική με την ψήφο για συγκεκριμένη ψηφοφορία, και την συνακόλουθη κατάλληλη συμπλήρωση του Ψηφοδελτίου.

*handleMessage(message : Message)*

Διαχείριση λαμβανομένου μηνύματος.

*sendMessage(msg : Message, receiver : AgentIdentifier)*

Αποστολή μηνύματος σε πράκτορα.

*sendMessage(msg : Message, receivers\_list : Vector)*

Αποστολή μηνύματος σε λίστα από πράκτορες.

## Υποψήφιος

### **Γνωρίσματα**

*candidate\_solution\_name : String*

Το όνομα της υποψήφιας εναλλακτικής πρότασης που αντιπροσωπεύει ο Υποψήφιος.

*candidate\_identifier : AgentIdentifier*

Το προσδιοριστικό ταυτότητας του Υποψηφίου (περιλαμβάνει και το όνομα του Υποψηφίου).

*evaluation\_state : EvaluationSheet*

Χαρακτηρισμός του Υποψηφίου με βάση την αξιολόγηση που του έχει γίνει.

*team : Team*

Η ομάδα στην οποία εντάσσεται ο Υποψήφιος.

*cognitive\_module : CognitiveModule*

Το γνωσιακό τμήμα του πράκτορα.

*candidate\_ui : Object*

Αναφορά στη διεπιφάνεια χρήσης η οποία είναι πιθανό να συνδέεται με τον πράκτορα.

### **Λειτουργίες**

*getCandidateName () : String*

Επιστρέφει το όνομα του υποψηφίου.

*getCandidateSolutionName() : String*

Επιστρέφει το όνομα της υποψήφιας εναλλακτικής πρότασης που καταθέτει ο υποψήφιος.

*setCandidateSolutionName(candidate\_solution\_name : String)*

Θέτει το όνομα της υποψήφιας εναλλακτικής πρότασης που καταθέτει ο υποψήφιος.

*getCandidateId () : AgentIdentifier*

Επιστρέφει το προσδιοριστικό ταυτότητας του Υποψηφίου.

*handleMessage(message : Message)*

Διαχείριση λαμβανομένου μηνύματος.

*sendMessage(msg : Message, receiver : AgentIdentifier)*

Αποστολή μηνύματος σε πράκτορα.

*sendMessage(msg : Message, receivers\_list : Vector)*

Αποστολή μηνύματος σε λίστα από πράκτορες.

## Ψηφοφορία

### ***Γνωρίσματα***

*session\_id : int*

Μοναδικό προσδιοριστικό της Ψηφοφορίας.

*voting\_system : VotingSystem*

Το εκλογικό Σύστημα που χρησιμοποιείται στην Ψηφοφορία.

*constraints : Constraint*

Οι (χρονικοί) περιορισμοί που ισχύουν κατά την Ψηφοφορία.

*voting\_reason : String*

Ο σκοπός / θέμα της Ψηφοφορίας.

*altrernatives\_list : Vector*

Η λίστα με τις υποψήφιες εναλλακτικές λύσεις για την Ψηφοφορία.

*ballot : Ballot*

Το Ψηφοδέλτιο που χρησιμοποιείται στην Ψηφοφορία.

*voting\_committee : VotingCommittee*

Η Εφορευτική Επιτροπή που έχει την ευθύνη της Ψηφοφορίας.

*candidate\_selection\_threshold : int*

Ο αριθμός των "νικηφόρων" υποψηφίων.

### ***Λειτουργίες***

*VotingSession (constraints : Constraint, voting\_system : VotingSystem, voting\_reason : String)*

Κατασκευαστής της Ψηφοφορίας.

*provideSessionWithBallot (ballot : Ballot)*

Καθορίζει το ισχύον Ψηφοδέλτιο για την Ψηφοφορία

*provideSessionWithAlternatives (alternatives\_list : Vector)*

Παρέχει τη λίστα των υποψηφίων εναλλακτικών λύσεων

*provideSessionWithCommittee (committee : VotingCommittee)*

Καθορίζει την Εφορευτική Επιτροπή της Ψηφοφορίας.

*getVotingCommittee () : VotingCommittee*

Επιστρέφει την Εφορευτική Επιτροπή της Ψηφοφορίας.

*getVotingSystem () : VotingSystem*

Επιστρέφει το Εκλογικό Σύστημα της Ψηφοφορίας.

*getVotingReason () : String*

Επιστρέφει το θέμα της Ψηφοφορίας.

*getConstraints () : Constraint*

Επιστρέφει τους ισχύοντες (χρονικούς) περιορισμούς της Ψηφοφορίας.

*getBallot () : Ballot*

Επιστρέφει το Ψηφοδέλτιο που χρησιμοποιείται στην Ψηφοφορία.

*getAlternativesList () : Vector*

Επιστρέφει τη λίστα των υποψηφίων εναλλακτικών προτάσεων.

*getCandidateSelectionThreshold () : int*

Επιστρέφει τον αριθμό των νικηφόρων υποψηφίων.

## Εκλογικό Σύστημα

### Γνωρίσματα

*arithmetic\_value\_type : ArithmeticValueKind*

Ο τύπος των αριθμητικών πεδίων τιμών που αναγράφονται στα Ψηφοδέλτια για την Ψηφοφορία που ισχύει αυτό το Εκλογικό Σύστημα.

*list\_of\_voters\_n\_weights : VoterWeightPairsList*

Λίστα από ζεύγη ψηφοφόρων και αριθμητικών βαρών ψηφοφόρων, που χρησιμοποιούνται σε περίπτωση χρήσης ενός συστήματος βεβαρυμένης άποψης ψηφοφόρων (weighted systems), όπου η άποψη κάθε ψηφοφόρου μπορεί να αποτιμηθεί με διαφορετικό "βάρος". Σε κάθε άλλη περίπτωση η λίστα είναι κενή.

### Λειτουργίες

*sortingAlgorithm (ballot : Ballot, intermediate\_results\_list : CandidateValuePairsList) : CandidateValuePairsList*

Αλγόριθμος κατάταξης των υποψηφίων εναλλακτικών προτάσεων με βάση τις ψήφους που έλαβαν.

*getArithmeticValueType () : ArithmeticValueKind*

Επιστρέφει τον τύπο των αριθμητικών πεδίων τιμών που αναγράφονται στα Ψηφοδέλτια για την Ψηφοφορία που ισχύει αυτό το Εκλογικό Σύστημα.

## Ψηφοδέλτιο

### Γνωρίσματα

*voter\_identification : AgentIdentifier*

Προσδιοριστικό ταυτότητας του Ψηφοφόρου (συμπληρώνεται προαιρετικά).

*comments* : *String*

Σχόλια επί του ψηφοδελτίου (προαιρετικά).

*alternatives\_and\_values\_pair\_list* : *CandidateValuePairsList*

Λίστα υποψηφίων προτάσεων και αριθμητικών τιμών.

*voting\_session* : *VotingSession*

Η Ψηφοφορία στην οποία χρησιμοποιείται το Ψηφοδέλτιο.

*type\_of\_values* : *ArithmeticValueKind*

Ο τύπος των αριθμητικών πεδίων τιμών του Ψηφοδελτίου.

### **Λειτουργίες**

*Ballot* (*voter\_id* : *AgentIdentifier*, *alternatives\_and\_values\_pair\_list* : *CandidateValuePairsList*, *voting\_session* : *VotingSession*, *comments* : *String*, *type\_of\_values* : *ArithmeticValueKind*)

Κατασκευαστής του Ψηφοδελτίου.

*getVoterId* () : *AgentIdentifier*

Επιστρέφει τον προσδιοριστή ταυτότητας του Ψηφοφόρου.

*getComments* () : *String*

Επιστρέφει τα αναγραφόμενα στο ψηφοδέλτιο σχόλια.

*getVotingSession* () : *VotingSession*

Επιστρέφει την Ψηφοφορία για την οποία προορίζεται το Ψηφοδέλτιο.

*getTypeOfValues* () : *ArithmeticValueKind*

Επιστρέφει τον τύπο των αριθμητικών πεδίων τιμών του Ψηφοδελτίου.

*getAlternativeValuePairsList* () : *CandidateValuePairsList*

Επιστρέφει τη λίστα των ζευγών υποψηφίων προτάσεων - αριθμητικών τιμών.

*insertAlternativeValuePair* () : *boolean*

Εισάγει ένα ζεύγος υποψήφιας πρότασης - αριθμητικής τιμής στη σχετική λίστα.

*getNameValuePair* (*candidate\_solution\_name* : *String*) : *NameValuePair*

Επιστρέφει ένα ζεύγος υποψήφιας πρότασης - αριθμητικής τιμής από τη σχετική λίστα.

*getValue* (*candidate\_solution\_name* : *String*) : *float*

Επιστρέφει την αριθμητική τιμή που αντιστοιχεί στην υποψήφια πρόταση *candidate\_solution\_name*.

## **Εκλογικό Αποτέλεσμα**

### **Γνωρίσματα**

*alternatives\_and\_values\_pair\_list* : *CandidateValuePairsList*

Διατεταγμένη λίστα νικηφόρων υποψηφίων προτάσεων και αριθμητικών τιμών.

*voting\_session* : *VotingSession*

Η ψηφοφορία στην οποία αναφέρεται το Αποτέλεσμα.

*type\_of\_values* : *ArithmeticValueKind*

Ο τύπος των αριθμητικών πεδίων τιμών του ψηφοδελτίου.

*ballots\_list* : *Vector*

Λίστα από τα ψηφοδέλτια που υπήρχαν στην κάλπη.

### **Λειτουργίες**

*Result* (*voting\_session*: *VotingSession*, *alternatives\_and\_values\_pair\_list*: *CandidateValuePairsList*, *type\_of\_values* : *ArithmeticValueKind*, *ballots\_list* : *Vector*)

Κατασκευαστής του Αποτελέσματος.

*xformR2C(candidate\_id : AgentIdentifier) : Candidate*

Συνάρτηση μετατροπής της πρώτης νικήτριας επιλογής του Αποτελέσματος σε Υποψήφιο (που καταθέτει την αντίστοιχη με την επιλογή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*xformR2C() : Candidate*

Συνάρτηση μετατροπής της πρώτης νικήτριας επιλογής του Αποτελέσματος σε Υποψήφιο (που καταθέτει την αντίστοιχη με την επιλογή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*xformR2VectorOfC() : Vector*

Συνάρτηση μετατροπής όλων των νικητών (αν αυτοί είναι περισσότεροι από έναν) που αναφέρονται στο Αποτέλεσμα, σε Υποψηφίους (που καταθέτει ο καθένας την αντίστοιχη με τον κάθε νικητή εναλλακτική πρόταση). Χρησιμοποιείται σε περίπτωση που διενεργείται μια ακολουθία ψηφοφοριών.

*getVotingSession () : VotingSession*

Επιστρέφει το voting session στο οποίο αντιστοιχεί το Αποτέλεσμα.

*getTypeOfValues () : ArithmeticValueKind*

Επιστρέφει τον τύπο των αριθμητικών τιμών που υπήρχαν στο ψηφοδέλτιο.

*getAlternativeValuePairsList () : CandidateValuePairsList*

Επιστρέφει τη λίστα των ζευγών νικηφόρων υποψηφίων προτάσεων - αριθμητικών τιμών.

*getNameValuePair (candidate\_solution\_name : String) : NameValuePair*

Επιστρέφει ένα ζευγάρι νικηφόρας πρότασης υποψηφίου - αντίστοιχης αριθμητικής τιμής από τη σχετική λίστα.

*getValue (candidate\_solution\_name : String) : float*

Επιστρέφει την αριθμητική τιμή που αντιστοιχεί στην νικηφόρο υποψήφια πρόταση candidate\_name.

*getBallotsList () : Vector*

Επιστρέφει τη λίστα των ψηφοδελτίων που υπήρχαν στην κάλπη.

## **Κάλπη**

### **Γνωρίσματα**

*ballots\_list* : *Vector*

Λίστα των Ψηφοδελτίων που βρίσκονται στην Κάλπη.



### **Λειτουργίες**

*insertBallotInBox (ballot : Ballot)*

Εισάγει Ψηφοδέλτιο στην Κάλπη.

*getNextBallot () : Ballot*

Εξάγει Ψηφοδέλτιο από την Κάλπη.

*getBallot (voter: Voter) : Ballot*

Επιστρέφει το Ψηφοδέλτιο που κατατέθηκε από τον Ψηφοφόρο voter.

*getBallotsList () : Vector*

Επιστρέφει τη λίστα των Ψηφοδελτίων που βρίσκονται στην Κάλπη.

## Παράρτημα ΙΙ : Γλωσσάρι Όρων Θεωρίας Ψηφοφοριών

---

**Ανελικρινής Ψήφος (Insincere Vote):** Μια ψήφος που δεν είναι ειλικρινής.

**Ανεξαρτησία από Αδιάφορες Εναλλακτικές Επιλογές (Independence from Irrelevant Alternatives):** Ισχύει στην περίπτωση που το εκλογικό σύστημα παράγει πάντοτε το ίδιο αποτέλεσμα, με δεδομένη την ίδια πάντοτε συνολική κατάταξη των προτιμήσεων των ψηφοφόρων.

**Αποδοτικότητα Condorcet:** Ποσοστό εκλογικών αναμετρήσεων στις οποίες αναδεικνύεται "νικητής κατά το Condorcet", αν υπάρχει τέτοιος.

**Αποτελεσματικότητα (Effectiveness):** Η δυνατότητα ενός ψηφοδελτίου να επηρεάσει το αποτέλεσμα μιας ψηφοφορίας.

**Αποφασιστικό Εκλογικό Σύστημα (Deterministic Voting System):** Ένα εκλογικό σύστημα που πάντα παράγει νικητή.

**Ειλικρινής Ψήφος (Sincere Vote):** Ένας ψηφοφόρος δίδει ειλικρινή ψήφο, αν, για κάθε ζεύγος επιλογών, δίδει μεγαλύτερο ή τουλάχιστον ίδιο βάρος στην επιλογή που προτιμά περισσότερο.

**Εκλογικές Μέθοδοι Έκφρασης Θέσης (Positional Voting Methods):** Οι μέθοδοι (τα εκλογικά συστήματα) που εκλέγουν νικητή με βάση σειρές προτίμησης καταγεγραμμένες από τους ψηφοφόρους.

**Εκφραστικότητα (Expressiveness):** Ο βαθμός κατά τον οποίο ένα εκλογικό σύστημα επιτρέπει στους ψηφοφόρους να εκφράσουν τις ειλικρινείς τους προτιμήσεις, χωρίς να θυσιάσουν την αποτελεσματικότητα της ψήφου τους.

**Επιτηδευμένη Ψήφος (Sophisticated Vote):** Μία ψήφος η οποία καθορίζεται, με χρήση ανάλυσης θεωρίας παιγνίων που λαμβάνει υπόψη τις αναμενόμενες στρατηγικές των υπόλοιπων ψηφοφόρων, ώστε να είναι η καλύτερη δυνατή. Είναι ένας τύπος "στρατηγικής ψήφου".

**Κανόνας της Απόλυτης Πλειοψηφίας (Majority Rule):** Σύμφωνα με αυτόν, ο υποψήφιος που λαμβάνει την απόλυτη πλειοψηφία των ψήφων είναι ο νικητής.

**Κριτήριο Condorcet:** Ικανοποιείται από ένα εκλογικό σύστημα, αν αυτό επιλέγει τον "νικητή κατά το Condorcet".

**Λογική (Rational):** Η ιδιότητα αυτή χαρακτηρίζει τους ψηφοφόρους, αν προβαίνουν σε ενέργειες που πιστεύουν ότι θα οδηγήσουν στο αποτέλεσμα που προτιμούν - ανεξάρτητα από το αν είναι λογικό να προτιμούν αυτό το αποτέλεσμα.

**Μικροχειραγώγηση (Micromanipulation):** Χειραγώγηση της ψηφοφορίας από ξεχωριστούς ψηφοφόρους.

**Μονοτονία (Monotonicity):** Ένα εκλογικό σύστημα είναι μονότονο αν, στην περίπτωση που κάποιος ψηφοφόρος αυξήσει την σειρά προτίμησης του για τον νικητή υποψήφιο, αυτός θα ανακηρυχθεί ξανά νικητής, όπως επίσης στην περίπτωση που κάποιος ψηφοφόρος μειώσει την σειρά προτίμησης του για κάποιον ηττημένο υποψήφιο, αυτός θα παραμείνει ηττημένος.

**Νικητής κατά το Condorcet:** Ο υποψήφιος που είτε πλειοψηφεί επί του αντιπάλου, είτε ισοψηφεί μαζί του, σε όλες τις κατά ζεύγος συγκρούσεις με τους συνυποψηφίους του.

**Πλειοψηφική Εκλογή (Plurality Voting):** Η εκλογική διαδικασία κατά την οποία κάθε ψηφοφόρος ψηφίζει μια μοναδική εναλλακτική πρόταση (ή, όταν η εναλλακτικές προτάσεις πρόκειται να επιλεγούν, οι ψηφοφόροι ψηφίζουν τις η πλέον προτιμητέες κατ' αυτούς προτάσεις), και κατά την οποία τελικά ανακηρύσσεται νικήτρια η πρόταση που λαμβάνει τις περισσότερες ψήφους.

**Σταθερότητα (Stability):** Η ιδιότητα παραγωγής του ίδιου αποτελέσματος από το ίδιο σύνολο ψηφοδελτίων, όσες φορές κι αν γίνει καταμέτρησή τους.

**Στρατηγική Ψήφος (Strategic Vote):** Η διαμόρφωση του ψηφοδελτίου με τέτοιο τρόπο, ώστε να προάγεται τελικά η πλέον προτιμητέα επιλογή του ψηφοφόρου, χωρίς απαραίτητα να είναι ειλικρινής η ψήφος.

**Στρατηγική της Ψήφου (Voting Strategy):** Ένα σχέδιο δράσης, που καταρτίζεται πριν την ψηφοφορία, και περιγράφει τις ενέργειες που πρέπει να κάνει ο ψηφοφόρος προκειμένου να επιτευχθεί το επιθυμητό εκλογικό αποτέλεσμα.

**Συνέπεια (Consistency):** Ισχύει για ένα εκλογικό σύστημα αν, σε περίπτωση που το εκλογικό σώμα διαιρεθεί τυχαία σε δύο τμήματα και ξεχωριστές εκλογές με χρήση του εκλογικού συστήματος σε καθένα από αυτά επιλέξουν τον ίδιο νικητή, αυτός είναι ο ίδιος με

το νικητή που θα αναδείκνυε ολόκληρο το εκλογικό σώμα με χρήση του ίδιου εκλογικού συστήματος.

**Συντονισμένη Μακροχειραγώγηση (Co-ordinated Macromanipulation):** Χειραγώγηση της ψηφοφορίας από συνασπισμούς ψηφοφόρων.

**Σύστημα Αναλογικής Εκπροσώπησης (Proportional Representation System):** Ένα εκλογικό σύστημα που προάγει την μέγιστη δυνατή αντιστοίχιση μεταξύ του ποσοστού (επί του εκλογικού σώματος) των υποστηρικτών ενός κόμματος, και του ποσοστού των εδρών που καταλαμβάνει το κάθε κόμμα στο σώμα αντιπροσώπων που εκλέγεται.

**Χειραγωγησιμότητα (Manipulability):** Ισχύει για ένα εκλογικό σύστημα που επιτρέπει στους ψηφοφόρους να πετύχουν κάποιο προτιμότερο για αυτούς αποτέλεσμα με το να ψηφίσουν κατά τρόπο που δεν εκφράζει τις πραγματικές τους επιλογές.

**Ψηφοφορία Έγκρισης (Approval Voting):** Εκλογική διαδικασία κατά την οποία ο ψηφοφόρος επιλέγει (εγκρίνει) όσους υποψηφίους επιθυμεί, χωρίς να έχει οποιονδήποτε περιορισμό αριθμού επιλέξιμων υποψηφίων, και χωρίς να τους κατατάσσει σε σειρά προτίμησης.

## Παράρτημα III : Παραδείγματα Παραβίασης Κριτηρίων Δικαιοσύνης

### Παραβίαση Κριτηρίου Απόλυτης Πλειοψηφίας

Έστω ότι σε μια ψηφοφορία με 100 ψηφοφόρους χρησιμοποιείται το Εκλογικό Σύστημα Υπολογισμού Borda. Ο πίνακας δείχνει ένα πιθανό αποτέλεσμα της ψηφοφορίας:

	Υποψήφιος Α	Υποψήφιος Β	Υποψήφιος Γ
1η Επιλογή (2 μόρια)	52	39	9
2η Επιλογή (1 μόριο)	2	30	68
3η Επιλογή (0 μόρια)	46	31	23
<b>Σύνολο μορίων</b>	<b>106</b>	<b>108</b>	<b>86</b>

Όπως φαίνεται στον πίνακα, ο Α έχει την απόλυτη πλειοψηφία στις πρώτες προτιμήσεις των ψηφοφόρων, αλλά το εκλογικό σύστημα δίδει τον Β ως νικητή. Το γεγονός αυτό συνιστά παραβίαση του Κριτηρίου Απόλυτης Πλειοψηφίας.

### Παραβίαση Κριτηρίου Condorcet

Έστω τέσσερις υποψήφιοι για τη θέση του δημάρχου μιας πολύ μικρής πόλης, ο Α, ο Β, ο Γ και ο Δ. Υπάρχουν 20 εγγεγραμμένοι ψηφοφόροι. Η τοπική εφημερίδα έκανε μια έρευνα που δημοσιεύτηκε μετά τις εκλογές, και ρώτησε τους ψηφοφόρους ποιόν προτιμούσαν για δήμαρχο αν η εκλογική σύγκρουση γινόταν κατά ζεύγη. Τα αποτελέσματα ήταν τα εξής:

*11 ψηφοφόροι προτιμούν τον Γ παρά τον Α*

*11 ψηφοφόροι προτιμούν τον Γ παρά τον Β*

*17 ψηφοφόροι προτιμούν τον Γ παρά τον Δ*

Έτσι, φαίνεται ότι ο Γ θα κέρδιζε σε κάθε πρόσωπο με πρόσωπο αναμέτρηση.

Στην πραγματικότητα όμως, η καταμέτρηση των αποτελεσμάτων με σύστημα σχετικής πλειοψηφίας έδωσε στον Α εννέα ψήφους και τη θέση του δημάρχου, στον Γ οκτώ ψήφους, στον Δ τρεις ψήφους και στον Β καμμία ψήφο. Η μή εκλογή του Γ συνιστά παραβίαση του Κριτηρίου Condorcet.

### **Παραβίαση Κριτηρίου Μονοτονίας**

Τρεις μαθητές, ο Χάρης, ο Άρης και ο Πάρης είναι υποψήφιοι πρόεδροι της τάξης τους. Η ψηφοφορία γίνεται σε γύρους. Ο τρίτος στην κατάταξη μετά τον πρώτο γύρο υποχρεώνεται να εγκαταλείψει την εκλογική μάχη, και ο νικητής αναδεικνύεται μεταξύ των δύο εναπομείναντων υποψηφίων μετά από δεύτερο γύρο. Στον πρώτο γύρο, ο Χάρης παίρνει 11 ψήφους, ο Άρης παίρνει 8 ψήφους και ο Πάρης 10. Έτσι, ο Άρης εγκαταλείπει, και στον δεύτερο γύρο ο Χάρης παίρνει 11 ψήφους και ο Πάρης 18. Έτσι ο Πάρης είναι ο νικητής.

Όμως, λόγω "τεχνικού" προβλήματος με τα ψηφοδέλτια, ο πρόεδρος της Εφορευτικής κηρύσσει άκυρο το αποτέλεσμα και η ψηφοφορία πρέπει να επαναληφτεί. Στον πρώτο γύρο της επαναληπτικής ψηφοφορίας, όλοι ψήφισαν όπως στον πρώτο γύρο της αρχικής ψηφοφορίας, εκτός από 4 ψηφοφόρους που αποφάσισαν να ενισχύσουν τον Πάρη αντί του Χάρη, μια που ο Πάρης ήταν ο προηγούμενος νικητής. Έτσι, ο Χάρης παίρνει 7 ψήφους, ο Άρης 8, ο Πάρης 14, και ο Χάρης αναγκάζεται να εγκαταλείψει (αντί του Άρη). Όμως, οι 7 ψηφοφόροι του Χάρη, προτιμούν τον Άρη παρά τον Πάρη. Έτσι, στο δεύτερο γύρο ψηφίζουν όλοι τον Άρη, ο οποίος κερδίζει με 15 ψήφους έναντι 14 του Πάρη. Ο Άρης είναι ο τελικός νικητής, παρόλο που οι μόνες αλλαγές στις προτιμήσεις των ψηφοφόρων ήταν οι 4 ψήφοι που ενίσχυσαν τον αρχικό νικητή (τον Πάρη). Το γεγονός συνιστά παραβίαση του Κριτηρίου Μονοτονίας.

### **Παραβίαση Κριτηρίου Ανεξαρτησίας Αδιάφορων Εναλλακτικών**

#### **Επιλογών**

Σε ένα διαγωνισμό ζαχαροπλαστικής, ο νικητής προκύπτει μετά από ψηφοφορία όπου χρησιμοποιείται το Εκλογικό Σύστημα Υπολογισμού Borda. Οι διαγωνιζόμενοι είναι τρεις, ο Νίκος, ο Τάκης και η Μαρία. Η πρώτη προτίμηση ενός μέλους της Κριτικής Επιτροπής

λαμβάνει 3 μόρια, η δεύτερη 2 μόρια και η τρίτη 1 μόριο. Το αποτέλεσμα του διαγωνισμού φαίνεται στον πίνακα:

	Τούρτα Νίκου	Τούρτα Τάκη	Τούρτα Μαρίας
1η Επιλογή (3 μόρια)	27	2	24
2η Επιλογή (2 μόρια)	0	51	2
3η Επιλογή (1 μόριο)	26	0	27
<b>Σύνολο μορίων</b>	<b>107</b>	<b>108</b>	<b>103</b>

Πριν την τελική ανακοίνωση των αποτελεσμάτων όμως, η Μαρία ενοχλημένη ζητά να αποσυρθεί η τούρτα της από το διαγωνισμό. Η Κριτική Επιτροπή το δέχεται, και επαναυπολογίζει τα αποτελέσματα, δίδοντας βέβαια πλέον 2 μόρια για την πρώτη και 1 μόριο για τη δεύτερη επιλογή:

	Τούρτα Νίκου	Τούρτα Τάκη
1η Επιλογή (2 μόρια)	27	26
2η Επιλογή (1 μόριο)	26	27
<b>Σύνολο μορίων</b>	<b>80</b>	<b>79</b>

Έτσι, ο νικητής άλλαξε λόγω της εκ των υστέρων αποχώρησης ενός ηττημένου υποψηφίου. Αυτό συνιστά παραβίαση του Κριτηρίου Ανεξαρτησίας των Αδιάφορων Εναλλακτικών Επιλογών.

## **Παράρτημα IV : Επισκόπηση Γλωσσών Επικοινωνίας Πρακτόρων και Γλωσσών Αναπαράστασης Γνώσης**

---

### **Επιθυμητά χαρακτηριστικά μιας Γλώσσας Επικοινωνίας Πρακτόρων**

Μπορεί να προταθούν επτά κατηγορίες απαιτήσεων για μια καλή γλώσσα επικοινωνίας πρακτόρων (Agents Communication Language - ACL) [19] :

**Μορφής:** Δηλωτική, απλή και επεκτάσιμη συντακτικά, αναγνώσιμη από ανθρώπους, γραμμική γλώσσα ή εύκολα μετατρέψιμη σε γραμμική μορφή (για εύκολη μετάδοση).

**Περιεχομένου:** Πρέπει να γίνεται διαχωρισμός μεταξύ της γλώσσας επικοινωνίας (communication language), που χρησιμεύει για την έκφραση επικοινωνιακών πράξεων, και της γλώσσας περιεχομένου, η οποία χρησιμεύει για την έκφραση δεδομένων για την περιοχή. Η ACL πρέπει να δομείται κατάλληλα ώστε να εξυπηρετείται αυτός ο διαχωρισμός. Επίσης, η γλώσσα πρέπει να υποστηρίζει **ένα καλά καθορισμένο** (ώστε πληθώρα συστημάτων να το χρησιμοποιεί) σύνολο από "πρωτογενείς πράξεις επικοινωνίας" (communication acts/ primitives), το οποίο μπορεί να είναι *επεκτάσιμο*. Μπορεί να χρησιμοποιείται ή όχι μια και μόνη συγκεκριμένη γλώσσα περιεχομένου (αν ναι, μπορεί να περιοριστεί το σύνολο πρωτογενών πράξεων επικοινωνίας και να αυξηθούν οι αρμοδιότητες του "επιπέδου περιεχομένου" της γλώσσας, αλλά τότε όλες οι εφαρμογές που αυτή εξυπηρετεί πρέπει να χρησιμοποιούν την ίδια γλώσσα περιεχομένου).

**Σημασιολογίας:** Η σημασιολογία μιας ACL πρέπει να είναι **μή διφορούμενη (unambiguous)** και να επιδεικνύει **κανονική μορφή** (ομοιότητα στη σημασία πρέπει να οδηγεί σε ομοιότητα στην αναπαράσταση). *Η σημασιολογία πρέπει επιμελώς να καλύπτει τις έννοιες χρόνος και τόπος* (καθώς κατανεμημένοι στο χώρο πράκτορες επικοινωνούν ανά διάφορες χρονικές στιγμές). Επίσης, πρέπει να παρέχει ένα **μοντέλο επικοινωνίας** (χρήσιμο για μοντελοποίηση της απόδοσης - performance modeling).

**Υλοποίησης:** Αποτελεσματική υλοποίηση σε ταχύτητα, βέλτιστη αξιοποίηση της ταχύτητας του δικτύου, εύχρηστη επιφάνεια χρήσης (απόκρυψη των κατώτερων επιπέδων



επικοινωνίας), ταίριασμα με την τρέχουσα τεχνολογία λογισμικού και δυνατότητα μερικής υλοποίησης.

**Δικτύου:** Αξιοποίηση τελευταίας δικτυακής τεχνολογίας - υποστήριξη όλων των βασικών τρόπων διασύνδεσης (point-to-point, multicast, broadcast), υποστήριξη σύγχρονης και ασύγχρονης επικοινωνίας, παροχή πλούσιου συνόλου από πρωτογενείς πράξεις επικοινωνίας χρήσιμου ως υπόβαθρου πρωτοκόλλων αλληλεπίδρασης, ανεξάρτητων από τους μηχανισμούς μετάδοσης.

**Περιβάλλοντος:** Παροχή εργαλείων για αντιμετώπιση της πολύ πιθανής ετερογένειας των πρακτόρων, που μάλιστα πιθανότατα θα δρουν μέσα σε ένα δυναμικά μεταβαλλόμενο περιβάλλον. Διαλειτουργικότητα με άλλες γλώσσες/ πρωτόκολλα. Υποστήριξη τρόπων ανακάλυψης γνώσης μέσα σε μεγάλα δίκτυα. Ευκολία ενσωμάτωσης σε ήδη υπάρχοντα συστήματα.

**Αξιοπιστίας:** Παροχή αξιόπιστης και ασφαλούς επικοινωνίας, και τρόπων πιστοποίησης ταυτότητας (authentication) πρακτόρων. Παροχή μηχανισμών επισήμανσης σφαλμάτων.

Οι παραπάνω απαιτήσεις μπορεί να έρχονται πολλές φορές σε αντίφαση μεταξύ τους - κι είναι ευθύνη του σχεδιαστή της γλώσσας ο κατάλληλος συγκερασμός τους, ανάλογα με τις ανάγκες.

## **KQML (Knowledge Query and Manipulation Language)**

### ***Τί είναι η KQML / Γενικοί σκοποί της***

Η KQML (Knowledge Query and Manipulation Language) είναι μια γλώσσα για την ανταλλαγή πληροφορίας και γνώσης. Η KQML μπορεί να χρησιμοποιηθεί σαν μια γλώσσα αλληλεπίδρασης μεταξύ ενός έξυπνου συστήματος και μιας εφαρμογής, ή μεταξύ δύο ή περισσότερων έξυπνων συστημάτων τα οποία θέλουν να διαμοιραστούν γνώση στην προσπάθεια τους να επιλύσουν από κοινού ένα πρόβλημα. Η KQML ενσωματώνει ταυτόχρονα ένα format μηνύματος και ένα πρωτόκολλο διαχείρισης μηνυμάτων για να υποστηρίξει τον διαμοιρασμό γνώσης μεταξύ πρακτόρων κατά το χρόνο εκτέλεσης. Αναπτύχθηκε στα πλαίσια της χρηματοδοτούμενης από το ARPA "KSE" (Knowledge Sharing Effort), που είναι ένας συνεταιρισμός (consortium) για την ανάπτυξη τεχνικής υποδομής για την υποστήριξη ανταλλαγής γνώσης μεταξύ συστημάτων, και την ανάπτυξη βάσεων γνώσης μεγάλης κλίμακας οι οποίες είναι διαμοιραζόμενες και επαναχρησιμοποιήσιμες [19] [18] [17] [84].

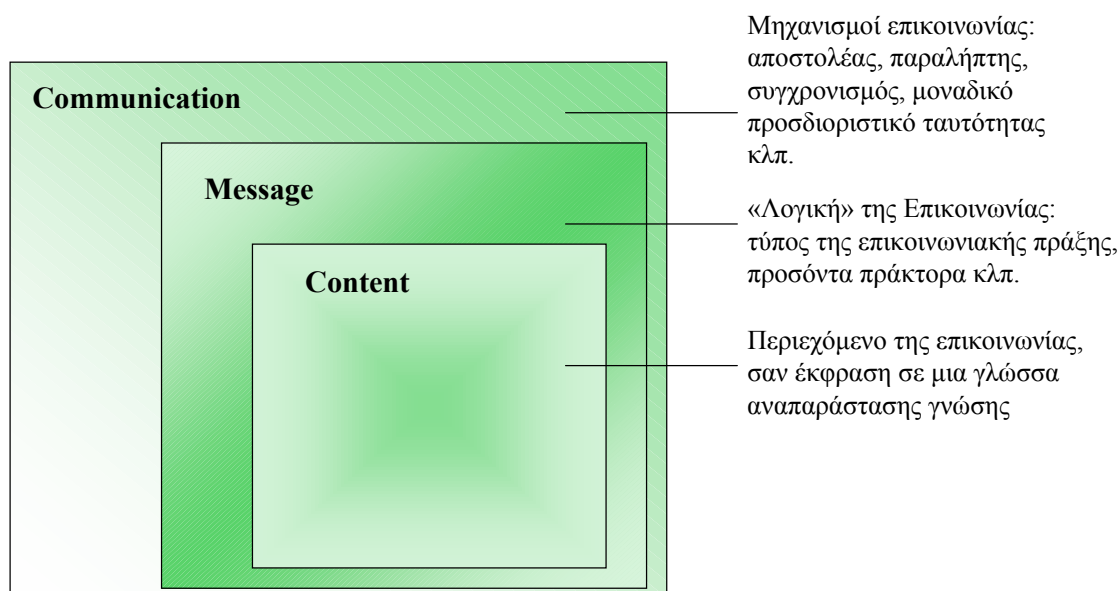
### **Stanford's ACL**

Η γλώσσα "ACL" ("Agent Communication Language") που έχει αναπτυχθεί στο Stanford είναι μια έκδοση της KQML που χρησιμοποιεί αποκλειστικά την KIF σαν γλώσσα περιεχομένου.

### **Επίπεδα στην KQML**

Μια έκφραση της KQML μπορεί να θεωρηθεί πως αποτελείται από μια έκφραση περιεχομένου ενθυλακωμένη σε ένα "περιτύλιγμα" (wrapper) μηνύματος που με τη σειρά του εμπεριέχεται σε ένα "επικοινωνιακό περιτύλιγμα" (communication wrapper), όπως φαίνεται στο Σχήμα 13.

Η γλώσσα δηλαδή αποτελείται από τρία επίπεδα: το επίπεδο περιεχομένου, το επίπεδο μηνύματος και το επίπεδο επικοινωνίας [19] [18] [17].



- Σχήμα 13 Τα επίπεδα της KQML. Οι KQML εκφράσεις μπορεί να θεωρηθούν ότι αποτελούνται από μια έκφραση περιεχομένου, περιεχόμενη σε ένα "περιτύλιγμα" μηνύματος, περιεχόμενο με τη σειρά του σε ένα "επικοινωνιακό περιτύλιγμα".

### **Content Layer**

Το επίπεδο περιεχομένου (content layer) της KQML περιέχει μια έκφραση σε κάποια γλώσσα αναπαράστασης γνώσης (ή αλλιώς γλώσσα περιεχομένου) -όπως η KIF- η οποία κωδικοποιεί την υπό μετάδοση γνώση (μπορεί επίσης να περιέχει και μια ακολουθία από αριθμούς, ή μια "ειδική" γλώσσα κωδικοποίησης γνώσης χρησιμοποιούμενη από τους

επικοινωνούντες πράκτορες). Το μόνο που έχουν να κάνουν οι πράκτορες είναι να συμφωνήσουν στην γλώσσα περιεχομένου. [19][18]

### ***Message Layer***

Το επίπεδο αυτό είναι που χρησιμοποιείται για την κωδικοποίηση της μεταδιδόμενης γνώσης. Τα μηνύματα που στέλνονται αρχίζουν (στην τελευταία specification της KQML) με ένα χαρακτηριστικό που περιγράφει την επιδιωκόμενη επικοινωνιακή πράξη (performative), όπως ask-one ή tell. Η περιγραφή του performative αποτελεί την βασική λειτουργία αυτού του επιπέδου. Ακόμα, στο επίπεδο αυτό τα μηνύματα περιέχουν ένα πεδίο για το ποιό είναι η γλώσσα περιεχομένου και ένα για το ποιό η οντολογία στην οποία αναφέρεται η ανταλλαγή μηνύματος (αν υπάρχει). [19][18]

Η KQML έχει συντακτικό που βασίζεται σε μια ισορροπημένη λίστα παρενθέσεων - μπορεί όμως στο μέλλον να τροποποιηθεί χάριν της απλότητας της γλώσσας.

### ***Communication Layer***

Στο επίπεδο αυτό περιγράφονται ο αποστολέας, ο παραλήπτης, και πληροφορίες που έχουν να κάνουν με τον τρόπο μετάδοσης/ επικοινωνίας (π.χ. blocking/ non-blocking σύνδεση κλπ.)

Ένα παράδειγμα KQML μηνύματος είναι το εξής:

(ask-one

:sender joe

:content (PRICE IBM ?PRICE)

:receiver stock-server

:reply-with ibm-stock

:language LPROLOG

:ontology NYSE-TICKS)

Στο μήνυμα αυτό, τα πεδία receiver, sender, reply-with ανήκουν στο communication layer, το περιεχόμενο περιγράφεται στο πεδίο content, και τα υπόλοιπα πεδία αποτελούν το message layer του μηνύματος.<sup>13</sup>

### ***Facilitators και Facilitator Interface Libraries***

Το πρόβλημα του συγχρονισμού πολλών πρακτόρων που επικοινωνούν μεταξύ τους είναι δύσκολο, και μπορεί να χωριστεί σε δύο κατηγορίες:

- Σε προβλήματα που έχουν να κάνουν με τους μηχανισμούς επίτευξης αξιόπιστης επικοινωνίας με τον κατάλληλο πράκτορα
- Σε προβλήματα που σχετίζονται με την κοινή κατανόηση και χρησιμότητα του περιεχομένου της επικοινωνίας

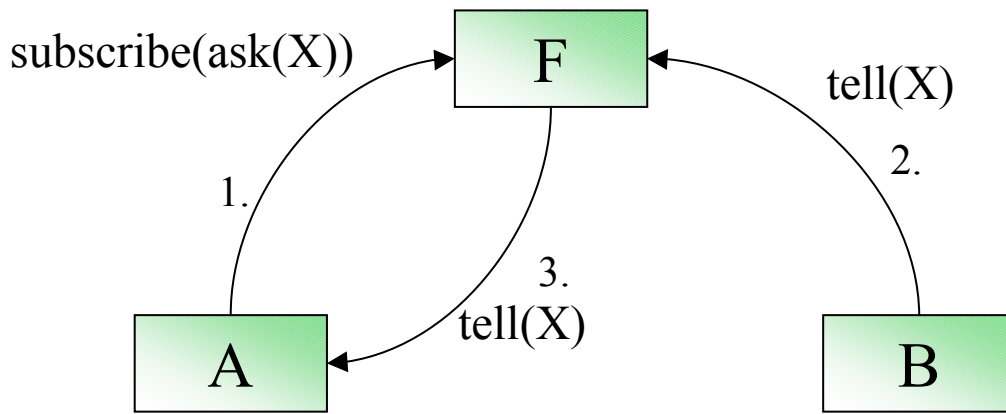
Για την επίλυση της πρώτης κατηγορίας προβλημάτων, μπορούν να χρησιμοποιηθούν ειδικοί πράκτορες, οι διεκπεραιωτές επικοινωνίας (communication facilitators), ενώ για την επίλυση της δεύτερης κατηγορίας προβλημάτων εισάγονται άλλοι ειδικοί πράκτορες, οι "ενδιάμεσοι" (mediator agents).

Οι facilitators και οι mediators επικοινωνούν μεταξύ τους με τη χρήση καλά καθορισμένων πρωτοκόλλων, ανεξάρτητων από την γλώσσα περιεχομένου.

Η επικοινωνία εξυπηρετείται από τους διεκπεραιωτές, χωρίς να επιβαρύνονται οι άλλοι πράκτορες.

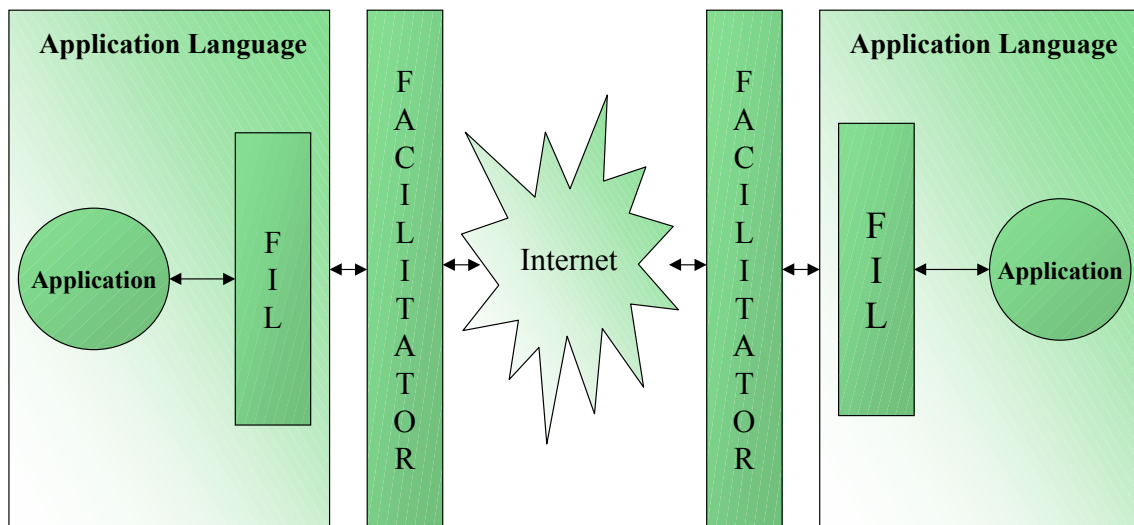
Κάθε εφαρμογή απλά πρέπει να ξέρει να επικοινωνεί με τους διεκπεραιωτές.

<sup>13</sup> Σε παλαιότερες εκδόσεις, η ενθυλάκωση των πεδίων ήταν σαφής και συντακτικά, αλλά τώρα η γλώσσα έχει απλοποιηθεί ως προς αυτό το στοιχείο.



• Σχήμα 14 Επικοινωνία πρακτόρων μέσω facilitators. Στο σχήμα φαίνεται το πώς ο πράκτορας A ζητεί από τον facilitator F να τον ενημερώνει για αλλαγές στη βάση γνώσης B.

Το πρωτόκολλο επικοινωνίας που χρησιμοποιεί η KQML ονομάζεται SKTP - Simple Knowledge Transfer protocol, και χωρίζεται σε τρία επίπεδα: περιεχομένου, μηνύματος και επικοινωνίας. Υλοποιεί τα δύο τελευταία επίπεδα χρησιμοποιώντας τους διεκπεραιωτές (facilitators) και τις Βιβλιοθήκες Διεπιφάνειας Διεκπεραιωτών (Facilitator Interface Libraries - FILs).



• Σχήμα 15 SKTP Αρχιτεκτονική. Κάθε τμήμα της αντιστοιχεί σε ένα από τα κύρια επίπεδα της KQML.

Μια FIL διεκπεραιώνει τρεις λειτουργίες [17]:

- Μεταφράζει σύνολα δηλώσεων που περιγράφουν τις συναλλαγές με την εσωτερική βάση γνώσης, οι οποίες πρέπει να εξαχθούν από ή να εισαχθούν στον πράκτορα
- Εμπεριέχει κώδικα για την παρακολούθηση αυτών των εσωτερικών συναλλαγών και προγραμματίζει την μετάδοση τους στον κατάλληλο διεκπεραιωτή, που την προωθεί στο δίκτυο.

- Εμπεριέχει κώδικα που παρέχει στους διεκπεραιωτές σημεία πρόσβασης για τη μετάδοση μηνυμάτων προς τον πράκτορα.

### ***Performatives***

Στο Σχήμα 16 φαίνονται τα προκαθορισμένα KQML performatives, τα οποία χωρίζονται σε επτά κατηγορίες [18] [17].

<u>Κατηγορία</u>	<u>Όνομα</u>
<b>Basic Query</b>	evaluate, ask-if, ask-about, ask-one, ask-all
<b>Multi-response (query)</b>	stream-about, stream-all, eos
<b>Response</b>	reply, sorry
<b>Generic informational</b>	tell, achieve, cancel, untell, unachieve
<b>Generator</b>	standby, ready, next, rest, discard, generator
<b>Capability - definition</b>	advertise, subscribe, monitor, import, export
<b>Networking</b>	register, unregister, forward, broadcast, route

• Σχήμα 16 Πίνακας KQML performatives.

### ***Αξιολόγηση της KQML ως ACL***

Στην ενότητα αυτή θα γίνει μια προσπάθεια αξιολόγησης της KQML ως γλώσσας Επικοινωνίας Πρακτόρων, με βάση τις απαιτήσεις από μια ACL που αναφέρθηκαν πιο πάνω [19].

**Μορφή:** Τα performatives είναι οι μοναδικές πρωτογενείς πράξεις επικοινωνίας που χρησιμοποιούνται στην KQML, και είναι πράγματι δηλωτικά των χρησιμοποιούμενων επικοινωνιακών πράξεων και των συνεπακολλούθων τους. Επιπλέον, τα μηνύματα σε KQML είναι γραμμικές ακολουθίες χαρακτήρων με σύνταξη που μοιάζει με Lisp, και έτσι είναι ευανάγνωστα, αναλύονται εύκολα και μετατρέπονται εύκολα σε άλλο format. Το συντακτικό της γλώσσας είναι απλό και επεκτάσιμο.

**Περιεχόμενο:** Με βάση το διαχωρισμό της KQML σε επίπεδα, τα μηνύματα της γλώσσας διαχωρίζουν το περιεχόμενο από την καθαυτή επικοινωνιακή λειτουργικότητά τους. Η

γλώσσα ορίζει ένα μέσου μεγέθους (ούτε υπερβολικά μικρό, ούτε υπερβολικά μεγάλο) "ελάχιστο σύνολο" (minimum set) από performatives τα οποία αποτελούν το επίπεδο μηνυμάτων (message layer) και ερμηνεύονται ως "εκφράσεις ομιλίας" (speech acts). Το επίπεδο επικοινωνίας (communication layer) της γλώσσας κωδικοποιεί μια σειρά από ορίσματα που περιγράφουν της χαμηλού επιπέδου παραμέτρους επικοινωνίας.

**Σημασιολογία:** Ο εφοδιασμός της γλώσσας με σημασιολογία (semantics) είναι ακόμα ένα ανοικτό θέμα. Υπάρχουν μόνο περιγραφές σε φυσική γλώσσα για τη σημασία και τη χρήση των performatives.

**Υλοποίηση:** Η δημιουργία μιας KQML διεπαφής για μια εφαρμογή, έχει κυρίως να κάνει με την παροχή συναρτήσεων διαχείρισης των performatives, ώστε να γίνονται αντικείμενο επεξεργασίας από την εφαρμογή. Το επικοινωνιακό κόστος από τη χρήση KQML είναι μεγάλο, λόγω των ενθυλακωμένων επιπέδων που αυξάνουν το μέγεθος των μηνυμάτων. Έχουν προστεθεί στη γλώσσα τρόποι συμπίεσης των μηνυμάτων.

**Δίκτυο:** Υποστηρίζεται peer-to-peer επικοινωνία και broadcasting/ multicasting μηνυμάτων, ασύγχρονη/ σύγχρονη επικοινωνία, και blocking/ non blocking αποστολή μηνύματος από μια εφαρμογή, με την κατάλληλη απόδοση τιμών σε πεδία του KQML μηνύματος

**Περιβάλλον:** Η KQML μπορεί να χρησιμοποιήσει οποιοδήποτε πρωτόκολλο μετάδοσης πληροφορίας σαν το μηχανισμό μετάδοσής της (HTTP, SMTP, TCP/ IP κλπ.). Δεν υπάρχουν περιορισμοί όσον αφορά την γλώσσα περιεχομένου (γλώσσα αναπαράστασης γνώσης) που χρησιμοποιείται (πρέπει βέβαια να υλοποιούνται συναρτήσεις χρήσης των performatives για τη γλώσσα περιεχομένου που χρησιμοποιεί η εφαρμογή). Δεν έχουν εμφανιστεί πολλές άλλες ACLs, ώστε να γίνουν προσπάθειες για επίτευξη διαλειτουργικότητας με αυτές (έχει γίνει όμως προσπάθεια στην περίπτωση της Shoham's Agent Oriented Programming). Η ύπαρξη των facilitators στο περιβάλλον της KQML μπορεί να χρησιμεύσει για την ανεύρεση γνώσης σε μεγάλο δίκτυο.

**Αξιοπιστία:** Η παροχή μηχανισμών ασφάλειας και πιστοποίησης ταυτότητας είναι ανοικτό θέμα.

### ***Εφαρμογές και KQML software***

Η KQML έχει χρησιμοποιηθεί σε πολλά συστήματα επίδειξης και πρωτότυπα. Οι εφαρμογές είχαν σχέση με ταυτόχρονη σχεδίαση hardware και software συστημάτων, με στρατιωτικές εφαρμογές για χρονικό προγραμματισμό δράσης (scheduling), με ευέλικτες αρχιτεκτονικές για ετερογενή πληροφοριακά συστήματα μεγάλης κλίμακας, με

ολοκλήρωση συστημάτων πρακτόρων και ανάκτηση πληροφοριών υπό συνεργασία. Η γλώσσα αποδείχτηκε αρκετά αποδοτική στην ολοκλήρωση διαφορετικών εργαλείων και συστημάτων, και στην επίτευξη "υψηλού επιπέδου" επικοινωνίας μεταξύ διασκορπισμένων στο δίκτυο οντοτήτων [19].

Μερικά APIs που έχουν χρησιμοποιήσει KQML είναι το JATLite (Stanford) που παρέχει templates γραμμένα σε Java για την κατασκευή πρακτόρων, το MAGENTA (Stanford) που είναι ένα ACL API που υποβοηθεί στην επικοινωνία πρακτόρων σε ετερογενή περιβάλλοντα, και το TKQML/Tackal (UMBC) που αποτελεί μια ολοκλήρωση της KQML με Tcl/Tk. [74]

### **KIF (Knowledge Interchange Format)**

Η KIF είναι μια *computer-oriented* γλώσσα (δεν αποσκοπεί στην αλληλεπίδραση με χρήστες) για την ανταλλαγή γνώσης ανάμεσα σε κατανεμημένα ετερογενή προγράμματα, που μπορεί να είναι γραμμένα από διαφορετικούς προγραμματιστές, σε διαφορετικές χρονικές στιγμές και σε διαφορετικές γλώσσες. *Η KIF δεν αποσκοπεί στο να αποτελεί την γλώσσα για την εσωτερική αναπαράσταση γνώσης σε ένα πρόγραμμα, ούτε καν ανάμεσα σε στενά συσχετιζόμενα προγράμματα. Οι εσωτερικοί τρόποι αναπαράστασης που ένα πρόγραμμα χρησιμοποιεί, αντιστοιχούνται σε αναπαράσταση σε KIF όταν αυτό θέλει να επικοινωνήσει με άλλο ετερογενές πρόγραμμα.*

**Η σημασιολογία της KIF είναι δηλωτική** - η σημασία των εκφράσεών της μπορεί να γίνει κατανοητή χωρίς να είναι αναγκαία η χρήση μεταφραστή για το χειρισμό τους.

Η KIF είναι **λογικά περιεκτική** - παρέχει δηλαδή τη δυνατότητα έκφρασης πολύπλοκων εκφράσεων σε κατηγορηματικό λογισμό πρώτης τάξης.

**Παρέχει τη δυνατότητα αναπαράστασης γνώσης για την αναπαράσταση γνώσης.** Έτσι μπορεί να γίνονται σαφείς όλες οι αποφάσεις για την αναπαράσταση γνώσης, και μπορούν να εισάγονται νέες δομές για την αναπαράσταση γνώσης χωρίς να αλλάζει η γλώσσα. Επιπλέον, κάνει δυνατή την αναπαράσταση μή μονότονων (μή τετριμμένων, πολύπλοκων) συμπερασματικών κανόνων, και παρέχει τη δυνατότητα ορισμού αντικειμένων, συναρτήσεων και σχέσεων (relations). [22] [73]

Επιπλέον αυτών, η KIF προσπαθεί να ικανοποιήσει - σε μικρότερη προτεραιότητα - και τις παρακάτω απαιτήσεις [22]:



- *Μεταφρασιμότητα* - Κεντρική λειτουργική απαίτηση για την γλώσσα είναι η παροχή πρακτικών τρόπων για την μετάφραση δηλωτικών βάσεων γνώσης από και προς γλώσσες αναπαράστασης γνώσης
- *Αναγνωσιμότητα* - Αν και η KIF δεν αποσκοπεί τόσο στη χρήση της για αλληλεπίδραση με ανθρώπους, η αναγνωσιμότητά της από αυτούς εξυπηρετεί τη χρήση της για την περιγραφή της σημασιολογίας γλωσσών αναπαράστασης γνώσης, τη χρήση της ως γλώσσα για τη δημοσίευση περιεχομένων βάσεων γνώσης, καθώς και τη χρήση της για την υποβοήθηση των ανθρώπων-χρηστών στην κατανόηση/ μετάφραση βάσεων γνώσης.

### **Χρησιμοποίηση ως εσωτερική γλώσσα παράστασης γνώσης**

Υπάρχουν δύο συντακτικές παραλλαγές της KIF, η γραμμική (linear) και η δομημένη (structured). Στη linear KIF όλες οι εκφράσεις είναι ακολουθίες ASCII χαρακτήρων (με προφανή πλεονεκτήματα στην αποθήκευση και τη μετάδοση), ενώ στην δομημένη KIF οι νόμιμες εκφράσεις της γλώσσας είναι δομημένα αντικείμενα. Η δομημένη KIF έχει ειδική χρήση στην επικοινωνία μεταξύ αντικειμένων που βρίσκονται στον ίδιο χώρο διευθύνσεων [22].

Η KIF έχει χρησιμοποιηθεί για την μεταφορά γνώσεων μεταξύ πρακτόρων που δρουν σε κατακευματμένα περιβάλλοντα ψηφιακών βιβλιοθηκών<sup>14</sup>, και αποτελεί την γλώσσα περιεχομένου που χρησιμοποιεί η "ACL" - version της KQML που έχει αναπτύξει το Stanford.

## **Άλλες ACLs και γλώσσες αναπαράστασης γνώσης**

### **Άλλες ACLs**

Εκτός της KQML λίγες είναι οι υπόλοιπες ACLs που έχουν αναπτυχθεί. Μπορούν να χωριστούν σε θεωρητικές και εφαρμοσμένες προσεγγίσεις. Οι πρώτες προσφέρουν δομές και φορμαλισμούς γενικά περί της έννοιας "πράκτορας" και περί της επικοινωνίας μεταξύ πρακτόρων ειδικότερα, ενώ οι δεύτερες επικεντρώνουν το ενδιαφέρον τους στην διαμόρφωση πρωτοκόλλων για την επικοινωνία διάφορων εφαρμογών και εργαλείων.

Παραδείγματα ACLs είναι [19]:

<sup>14</sup> Στην University of Michigan Digital Library - Η KIF είναι η βάση της Conspectus Language που χρησιμοποιείται εκεί.

Η AOP (Agent Oriented Programming), η οποία είναι περισσότερο θεωρητική προσέγγιση, στην οποία οι πράκτορες θεωρούνται οντότητες με πεποιθήσεις, ικανότητες, επιλογές και δεσμεύσεις. Φέρει όμως μαζί της μια καλά ορισμένη σημασιολογικά και συντακτικά γλώσσα, την AGENT-0, που διαθέτει performatives για την επικοινωνία των πρακτόρων. Η AGENT-0 έχει όμως μια σειρά από μειονεκτήματα, ένα από τα οποία είναι πως δεν διαθέτει τη δυνατότητα για "planning" (εκπόνηση σχεδίου ενεργειών). Προσπάθειες για ξεπέρασμα των προβλημάτων της AGENT-0 οδήγησαν στη δημιουργία μιας άλλης γλώσσας, της PLACA, ενώ η AGENT-K είναι μια γλώσσα που επεκτείνει την AGENT-0 ώστε να χρησιμοποιεί KQML για επικοινωνία.

Η Telescript, που ορίζει περιβάλλον για συναλλαγές μεταξύ εφαρμογών πάνω από το δίκτυο, με επικεντρωμένο ενδιαφέρον στις εφαρμογές ηλεκτρονικού εμπορίου. Όμως, οι πράκτορες στην Telescript δεν επικοινωνούν, απλά μεταφέρουν τον κώδικά τους πάνω από το δίκτυο και εκτελούν προκαθορισμένα scripts στη νέα τους τοποθεσία.

### ***Άλλες γλώσσες αναπαράστασης γνώσης***

Εκτός της KIF έχουν αναπτυχθεί κι άλλες γλώσσες αναπαράστασης γνώσης που μπορεί να χρησιμοποιηθούν ως γλώσσες περιεχομένου σε μια ACL. Τέτοιες είναι η KRSL (Knowledge Representation Specification Language - χρήση κυρίως για planning), η LOOM, η Interlingua και η CLOS.

## Βιβλιογραφία

---

1. "A Conversation with Marvin Minsky about Agents", Communications of the ACM, 37(7), pp. 23-29, 1994.
2. Arnold K. & Gosling J.: "The Java Programming Language", Addison-Wesley, 1996.
3. Arrow K.: "Social Choice and Individual Values", Wiley, New York, 1951.
4. Benaloh J. & Tuinstra D.: "Receipt-free secret-ballot elections", In Proceedings of the Twenty-sixth Annual ACM Symposium on the Theory of Computing, pp. 544-553, May 23-25, 1994.
5. Booch G.: "Object Oriented Analysis and Design with Applications", Benjamin - Cummings Publishing, 1994.
6. Brautigam C.G., A model-free voting approach to cue integration. Ph. D. dissertation, Dept. of Numerical Analysis and Computing Science, KTH (Royal Institute of Technology), August 1998.
7. Brooks R.A.: "Intelligence Without Reason", Computers and Thought, Proc. Of IJCAI-91, Sidney, Australia, 1991.
8. Buckminster Fuller R.: "No more secondhand God", In "No More Secondhand God and other writings", Southern Illinois University Press, Carbondale and Edwardsville, 1963.
9. Chavez A., Moukas A. & Maes P.: "Challenger: A Multi-Agent System for Distributed Resource Allocation", Proceedings of the International Conference on Autonomous Agents, Marina Del Ray, California, 1997.
10. Cohen J.D. & Fisher M.J.: "A Robust and Verifiable Cryptographically Secure Election Scheme (extended abstract)", Technical Report YALEU/DCS/TR-454, Yale University, July 1985.
11. Cohen P.R. & Levesque H.J.: "Teamwork", Nous 25(4), pp.487-512, 1991.
12. Davcev D. and Burkhard W.A.: "Consistency and Recovery Control for Replicated Files", In Proceedings of the Tenth ACM Symposium on Operating Systems, Operating Systems Review, 1985.
13. Drashansky T., Houstis E.N., Ramakrishnan N. & Rice J.R.: "Networked Agents for Scientific Computing", Communications of the ACM, 42(3), pp. 48-52 & 54, March 1999.
14. Eckel B.: "Thinking in Java", Prentice Hall, 1998.
15. Etzioni O. & Weld D.S.: "Intelligent Agents on the Internet: Fact, Fiction and Forecast", IEEE Expert, 10(4), pp. 44-49, 1995.
16. Faith-Cranor L.: "Declared-Strategy voting: An instrument for Group Decision-Making", Dissertation Thesis, Severe Institution of the Washington University, December 1996.
17. Finin T., Fritzson R., McKay D. & McEntire R.: "KQML- A Language and Protocol for Knowledge and Information Exchange"
18. Finin T., Fritzson R. & McKay D.: "A Language and Protocol to Support Intelligent Agent Interoperability", In Proceedings of the CE & CALS Washington '92 Conference, June 1992.
19. Finin T., Labrou Y. & Mayfield J.: "KQML as an agent communication language", in Bradshaw J.(Ed.), "Software Agents", AAAI Press/MIT Press, 1997.
20. Fujioka A., Okamoto T. & Ohta K.: "A practical secret voting scheme for large scale elections", In J. Seberry and Y. Zheng, editors, Advances in Cryptology - AUSCRYPT

- '92, Lecture Notes in Computer Science, vol. 718, pp. 244-251, Springer-Verlag, Berlin, 1993.
21. Gamma E., Helm R., Johnson R. & Vlissides J.: "Design Patterns: Elements of Reusable Object Oriented Software", Addison-Wesley, 1994.
  22. Genesereth M.R. & Fikes R.E.: Knowledge Interchange Format Version 3.0 Reference Manual, June 1992.
  23. Genesereth M.R. & Katchpel S.P.: "Software Agents", Communications of the ACM, 37(7):48-53,147, 1994.
  24. Gifford D.K.: "Weighted Voting for Replicated Data", In Proceedings of the Seventh ACM Symposium on Operating Systems Principles, 1979.
  25. Hayes-Roth B."Architectural foundations for Real-Time Performance in Intelligent Agents", Knowledge Systems Laboratory, Computer Science Department, Stanford University, 1990.
  26. Hayes-Roth B.: "An Architecture for Adaptive Intelligent Systems", Artificial Intelligence, 72 (1995) 329-365, 1995.
  27. Hedberg S. R.: "The first harvest of softbots looks promising", IEEE Expert, August 1995, pp.6-9, 1995.
  28. Hill R., Chen J., Gratch J., Rosenbloom P. & Tambe M.: "Intelligent Agents for the Synthetic Battlefield: A Company of Rotary Wing Aircraft", In Proceedings of Innovative Applications of Artificial Intelligence (IAAI-97), Providence, RI, July 1997.
  29. IBM Almaden Research Team: "Query by Image and Video Content: The QBIC System", Computer, September 1995.
  30. Iversen K.R.: "A Cryptographic Scheme for Computerized General Elections", In Advances in Cryptology - CRYPTO '91, Lecture Notes in Computer Science, vol. 576, pp. 405-419, Springer-Verlag, 1992.
  31. Jajodia S. & Mutchler D.: " Enhancements to the Voting Algorithm", In Proceedings of 13th International Conference on Very Large DataBases, VLDB'87, September 1-4, 1987, Brighton, England, 1987.
  32. Jennings N.R. & Wooldridge M.: "Software Agents", IEE Review, vol. Jan 96, pp. 17-20, 1996.
  33. Lijphart A. & Grofman B.: "Choosing an electoral system", In "Choosing an Electoral System: Issues and Alternatives", chapter 1, pp. 3-12, Praeger, New York, 1984.
  34. Long D.D. & Paris J.-F.: "A Realistic Evaluation of Optimistic Dynamic Voting", In Seventh Symposium on the Reliability in Distributed Systems, Columbus, OH, 10-12 October 1988.
  35. Long D.D. & Paris J.-F.: "On Improving the Availability of Replicated Files", In Symposium on Reliability in Distributed Software and Database Systems, Williamsburg, Va, 17-19 March 1987.
  36. Maes P.,Guttman R.H. & Moukas A.G.: "Agents that Buy and Sell", Communications of the ACM, 42(3), pp. 81-87& 90-91, March 1999.
  37. Maes P.: "Agents that Reduce Work and Information Overload", Communications of the ACM, 37(7), pp. 31-40, 1994.
  38. Maes P.: "Artificial Life Meets Entertainment: Lifelike Autonomous Agents", Communications of the ACM, 38(11), 108-114, 1995.
  39. Maes P.: "Modeling Adaptive Autonomous Agents", Artificial Life Journal, 1(1-2), MIT Press, 1994.
  40. Marsella S., Adibi J., Alonaizon Y., Erdem, A., Hill R., Kaminka G., Tambe M. & Zhun Q.: "Using an Explicit Teamwork Model and Learning in RoboCup: An Extended Abstract", RoboCup'98: Proceedings of the Second Robot World Cup Competition and Conferences, Springer Verlag, 1998.

41. Michalski S. R.: "Machine Learning : An Artificial Intelligence Approach", Springer Verlag, 1984.
42. Newell A.: "Reflections on the knowledge level", Artificial Intelligence, vol. 59, pp. 31-38, 1993.
43. Nurmi H., Salomaa A. & Santean L.: "Secret ballot elections in computer networks", Computers & Security, 36(10), pp. 553-560, 1991.
44. Nurmi H.: "Comparing Voting Systems", D. Reidel Publishing Company, Dordrecht, 1987.
45. Nurmi H.: "Voting Procedures: A Summary Analysis", British Journal of Political Science, 13(2), pp. 181-208, 1983.
46. Orphanoudakis S.C., Chronaki C.E. & Kostomanolakis S.: "I<sup>2</sup>C: A System for the Indexing, Storage and Retrieval of Medical Images by Content", Med.Inform, vol. 19, pp. 109-122, 1994.
47. Orphanoudakis S.C., Chronaki C.E. & Vamvaka D.: "I<sup>2</sup>Cnet: Content-Based Similarity Search in Geographically Distributed Repositories of Medical Images", Computerized Medical Imaging and Graphics, vol. 20(4), pp. 193-207, 1996.
48. Pentland A., Picard R.W. & Sclaroff S.: "Photobook: Content-Based Manipulation of Image Databases", M.I.T. Media Laboratory, Technical Report No. 255, November 1993.
49. Pirjanian P., Christensen H.I. & Fayman J.A.: "Application of Voting to Fusion of Purposive Modules: An Experimental Investigation", Journal of Robotics & Autonomous Systems, vol. 23, No. 4, July 1998, pp. 253-266, 1998.
50. Pirjanian P.: "Multiple Objective Action Selection & Behavior Fusion using Voting", Ph.D.-thesis, Faculty of Technical Sciences, Aalborg University Denmark. April 1998.
51. Quatrani T.: "Visual Modeling with Rational Rose and UML", Addison-Wesley, 1998.
52. Riker W.H.: "Liberalism against Populism", Waveland Press, Inc, Prospect Heights, 1982.
53. Russel S. & Norvig P.: "Artificial Intelligence: A Modern Approach", Prentice Hall Inc., 1995.
54. Sako K., "Electronic voting scheme allowing open objection to the tally", IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E77-A(1), pp. 24-30, 1994.
55. Salomaa A.: "Verifying and recasting secret ballots in computer networks", In H. Maurer, editor, New Results and New Trends in Computer Science, Lecture Notes in Computer Science, vol. 555, pp. 283-289, Springer-Verlag, Berlin, 1991.
56. Sandholm T.W. & Lesser V.R.: "Coalition Formation among Bounded Rational Agents", Proceedings of IJCAI-95, pp. 662-669, Montreal, 1995.
57. Schneier B.: "Applied Cryptography", John Wiley & Sons, New York, 1994.
58. Selfridge, O. G.: "Pandemonium: A paradigm for learning." Proceedings of the Symposium on Mechanization of Thought Processes ( Blake, D. V. and Uttley, A. M., ed.), National Physical Laboratory. Her Majesty's Stationery Office: Teddington, United Kingdom, 1959.
59. Shen W.-M., Adibi J., Adobbati R., Cho B., Erdem A., Moradi H., Salemi B. & Tejada S.: "Building Integrated Robots for Soccer Competition", Proceedings ICMAS'98, pp. 465-466, July 1998.
60. Smith J.R. & Chang S.-F.: "VisualSEEk: a fully automated content-based image query system", In Proceedings ACM Int. Multimedia Conference, 1996.
61. Stone P. & Veloso M.: "Multiagent Systems: A Survey from a Machine Learning Perspective", Carnegie Mellon University CS Technical Report CMU-CS-97-193, December, 1997.

62. Sycara K. & Zeng D.: "Coordination of multiple intelligent software agents", International Journal of Cooperative Information Systems, vol.5, Nos. 2&3, pp.181-211, 1996.
63. Tambe M.: "Teamwork in Real-World, Dynamic Environments", Proceedings of the International Conference on Mutli-Agent Systems (ICMAS), December 1996.
64. Tanaka-Ishii K., Noda I., Frank I., Nakashima H., Hasida K. & Matsubara H.: "Mike: An Automatic Commentary System for Soccer", Proceedings ICMAS'98, pp. 285-292, July 1998.
65. Tanebaum A. S., "Modern Operating Systems", vol. 1&2, Hanser – Prentice Hall, London 1994.
66. Trucco E. & Verri A.: "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.
67. Wooldridge M. & Jennings N.R.: "Intelligent Agents: Theory and Practice", Knowledge Enginnering Review, 10(12), pp.115-152, 1995.
68. Wurman P.R., Wellman M.P. & Walsh W.E.: "The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents", In Proceedings 2nd International Conference on Autonomous Agents, 1998.
69. Ζαμπούλης Ξ.: "Χρήση αυτόνομων οντοτήτων λογισμικού στη διαχείριση πόρων ενός συστήματος επείγουσας προνοσοκομειακής ιατρικής", Μεταπτυχιακή Εργασία, Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης, Ιούνιος 1998.

## Παραπομπές σε Ηλεκτρονικές Διευθύνσεις

---

70. CORBA Success Stories - <http://www.corba.org/>
71. JATLite - <http://java.stanford.edu/>
72. Jini™ Connection Technology - <http://www.sun.com/jini/>
73. Knowledge Interchange Format (KIF) - <http://www-ksl.stanford.edu/knowledge-sharing/kif/>
74. KQML software - <http://www.cs.umbc.edu/kqml/software/>
75. Mobile Agent Platform - The OMG-MASIF conformant mobile agent platform is Grasshopper - <http://www.ikv.de/products/grasshopper/grasshopper.html>
76. Object Management Group Home Page - <http://www.omg.org/>
77. Object Management Group Home Page: What Is OMG-UML and Why Is It Important? <http://www.omg.org/news/pr97/umlprimer.html>
78. ObjectSpace: Voyager Overview - <http://www.objectspace.com/products/vgrOverview.htm>
79. Rational Software: Unified Development Solutions & Programming Tools - <http://www.rational.com>
80. Scientific American: Ask the Experts: Mathematics - <http://www.sciam.com/askexpert/math/math2.html>
81. Software Agents (by Genesereth & Katchpel) - <http://www.cs.umbc.edu/~thurston/genes.html>
82. The Mathematics of Voting - <http://www.sa.ua.edu/ctl/math103/Voting/mathemat.htm#The Mathematics of Voting>
83. The Source for Java™ Technology - <http://java.sun.com/>
84. What is KQML? - <http://www.cs.umbc.edu/kqml/whats-kqml.html>