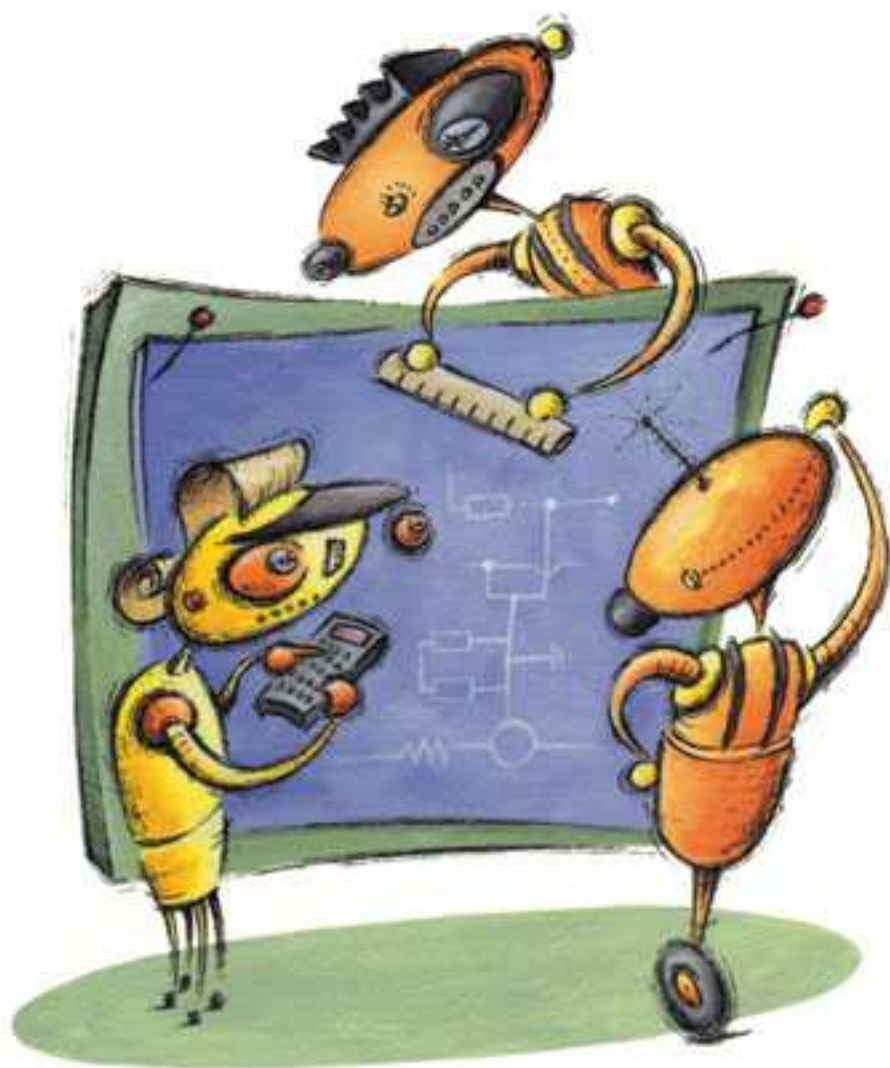


Εισαγωγή στα

# Πολυπρακτορικά συστήματα



**M I C H A E L W O O L D R I D G E**

Μετάφραση – Επιστημονική επιμέλεια ελληνικής έκδοσης: **Ασπασία Δασκαλοπούλου**,  
Επίκουρος Καθηγήτρια Πανεπιστημίου Θεσσαλίας



# Περιεχόμενα

<b>Πρόλογος</b> .....	<b>11</b>
<b>Πρόλογος για την ελληνική έκδοση</b> .....	<b>21</b>
<b>1 Εισαγωγή</b> .....	<b>23</b>
1.1 Το όραμα .....	27
1.2 Μερικές απόψεις για το πεδίο .....	30
1.3 Προβληματισμοί για τα πολυπρακτορικά συστήματα.....	32
<b>2 Ευφυείς πράκτορες</b> .....	<b>39</b>
2.1 Περιβάλλοντα .....	42
2.2 Ευφυείς πράκτορες .....	49
2.3 Πράκτορες και αντικείμενα .....	51
2.4 Πράκτορες και έμπειρα συστήματα.....	54
2.5 Οι πράκτορες ως συστήματα πρόθεσης.....	55
2.6 Αφηρημένες αρχιτεκτονικές για ευφυείς πράκτορες.....	58
2.7 Πώς θα πούμε σε έναν πράκτορα τι να κάνει .....	65
2.8 Σύνθεση πρακτόρων.....	73
<b>3 Πράκτορες παραγωγικού συλλογισμού</b> .....	<b>79</b>
3.1 Οι πράκτορες ως αποδείκτες θεωρημάτων.....	81
3.2 Πρακτοροστρεφής προγραμματισμός.....	88
3.3 Συντρέχουσα MetateM.....	90
<b>4 Πράκτορες πρακτικού συλλογισμού</b> .....	<b>101</b>
4.1 Πρακτικός συλλογισμός = Προσχεδιασμός + Συλλογισμός μέσων-σκοπών .....	101
4.2 Συλλογισμός μέσων-σκοπών .....	107
4.3 Υλοποίηση ενός πράκτορα πρακτικού συλλογισμού .....	113
4.4 HOMER: ένας πράκτορας που σχεδιάζει.....	119
4.5 Το Σύστημα PRS.....	121

<b>5</b>	<b>Αντιδραστικοί και υβριδικοί πράκτορες .....</b>	<b>129</b>
5.1	Ο Brooks και η αρχιτεκτονική υπαγωγής.....	130
5.2	Περιορισμοί των αντιδραστικών πρακτόρων .....	138
5.3	Υβριδικοί πράκτορες .....	139
5.3.1	TouringMachines .....	141
5.3.2	InteRRaP .....	143
<b>6</b>	<b>Πολυπρακτορικές αλληλεπιδράσεις .....</b>	<b>149</b>
6.1	Χρησιμότητες και προτιμήσεις.....	151
6.2	Πολυπρακτορικές αναμετρήσεις.....	153
6.3	Κυρίαρχες στρατηγικές και ισορροπίες Nash.....	156
6.4	Ανταγωνιστικές αλληλεπιδράσεις και αλληλεπιδράσεις μηδενικού αθροίσματος .....	159
6.5	Το δίλημμα του φυλακισμένου .....	160
6.6	Άλλες συμμετρικές αλληλεπιδράσεις $2 \times 2$ .....	169
6.7	Σχέσεις εξάρτησης στα πολυπρακτορικά συστήματα .....	173
<b>7</b>	<b>Σύναψη συμφωνιών.....</b>	<b>177</b>
7.1	Σχεδιασμός μηχανισμού.....	178
7.2	Δημοπρασίες.....	179
7.3	Διαπραγμάτευση .....	187
7.3.1	Τομείς προσανατολισμένοι σε εργασίες.....	190
7.3.2	Τομείς προσανατολισμένοι σε αξίες.....	199
7.4	Επιχειρηματολογία.....	201
<b>8</b>	<b>Επικοινωνία.....</b>	<b>217</b>
8.1	Λεκτικές ενέργειες .....	218
8.1.1	Austin .....	219
8.1.2	Searle.....	220
8.1.3	Η βασισμένη σε πλάνα θεωρία των λεκτικών ενεργειών .....	221
8.1.4	Οι λεκτικές ενέργειες ως ορθολογική δράση .....	222
8.2	Γλώσσες επικοινωνίας πρακτόρων.....	224
8.2.1	KIF .....	224
8.2.2	KQML.....	226
8.2.3	Οι γλώσσες επικοινωνίας πρακτόρων του FIPA.....	232
8.3	Οντολογίες για την επικοινωνία πρακτόρων .....	238
8.4	Γλώσσες συντονισμού.....	243

<b>9</b>	<b>Συνεργασία</b> .....	<b>247</b>
9.1	Συνεργατική κατανεμημένη επίλυση προβλήματος .....	248
9.2	Κοινοχρησία εργασιών και κοινοχρησία αποτελεσμάτων.....	251
9.2.1	Κοινοχρησία εργασιών στο Contract Net.....	254
9.3	Κοινοχρησία αποτελεσμάτων.....	257
9.4	Συνδυασμός της κοινοχρησίας εργασιών και αποτελεσμάτων .....	257
9.5	Χειρισμός της ασυνέπειας .....	260
9.6	Συντονισμός.....	261
9.6.1	Συντονισμός μέσω μερικού καθολικού σχεδιασμού .....	263
9.6.2	Συντονισμός μέσω κοινών προθέσεων .....	266
9.6.3	Συντονισμός με αμοιβαία μοντελοποίηση .....	273
9.6.4	Συντονισμός με νόρμες και κοινωνικούς νόμους.....	276
9.7	Πολυπρακτορικός σχεδιασμός ενεργειών και συγχρονισμός.....	283
<b>10</b>	<b>Μεθοδολογίες</b> .....	<b>289</b>
10.1	Πότε είναι κατάλληλη μια λύση που βασίζεται σε πράκτορες;.....	289
10.2	Πρακτοροστρεφείς τεχνικές ανάλυσης και σχεδιασμού .....	290
10.3	Παγίδες κατά την ανάπτυξη πρακτόρων .....	299
10.4	Κινητοί Πράκτορες .....	303
<b>11</b>	<b>Εφαρμογές</b> .....	<b>313</b>
11.1	Πράκτορες διαχείρισης ροής εργασίας και επιχειρηματικών διεργασιών.....	313
11.2	Πράκτορες για κατανεμημένη αίσθηση.....	316
11.3	Πράκτορες για ανάκληση και διαχείριση πληροφοριών .....	317
11.4	Πράκτορες για ηλεκτρονικό εμπόριο .....	325
11.5	Πράκτορες για διασυνδέσεις ανθρώπου και υπολογιστή.....	330
11.6	Πράκτορες για εικονικά περιβάλλοντα .....	331
11.7	Πράκτορες για κοινωνική προσομοίωση.....	331
11.8	Πράκτορες για Χ.....	337
<b>12</b>	<b>Λογική για πολυπρακτορικά συστήματα</b> .....	<b>341</b>
12.1	Γιατί τροπική λογική; .....	342
12.2	Σημασιολογία πιθανών κόσμων για τροπικές λογικές .....	345
12.3	Κανονικές τροπικές λογικές .....	346
12.4	Επιστημική λογική για πολυπρακτορικά συστήματα.....	354
12.5	Υπερ-θέσεις; στόχοι και επιθυμίες .....	357

12.6	Κοινή και καταναεμημένη γνώση .....	358
12.7	Ολοκληρωμένες θεωρίες πρακτόρων.....	360
12.8	Τυπικές μέθοδοι στην πρακτοροστρεφή τεχνολογία λογισμικού.....	367
12.8.1	Τυπικές μέθοδοι για τον προσδιορισμό προδιαγραφών συστημάτων.....	367
12.8.2	Τυπικές μέθοδοι στην υλοποίηση.....	369
12.8.3	Επαλήθευση .....	374
<b>Παράρτημα Α: Ένα μάθημα ιστορίας.....</b>		<b>387</b>
<b>Επίλογος .....</b>		<b>403</b>
<b>Γλωσσάρι.....</b>		<b>405</b>
<b>Βιβλιογραφία.....</b>		<b>417</b>
<b>Ευρετήριο.....</b>		<b>445</b>

## *Εισαγωγή*

Η ιστορία της πληροφορικής μέχρι σήμερα έχει σημαδευτεί από πέντε σημαντικές και συνεχιζόμενες τάσεις:

- *Καθολικότητα* (ubiquity)
- *Διασύνδεση* (interconnection)
- *Νοημοσύνη* (intelligence)
- *Ανάθεση* (delegation)
- *Ανθρωποκεντρισμός* (human-orientation)

Με τον όρο "καθολικότητα" εννοώ απλά ότι η συνεχής μείωση του κόστους της υπολογιστικής ικανότητας έκανε εφικτή την εισαγωγή εργαλείων επεξεργαστικής ισχύος σε χώρους και συσκευές όπου διαφορετικά θα ήταν ασύμφορα οικονομικά, και ίσως ακόμα και αδιανόητα. Αυτή η τάση αναπόφευκτα θα συνεχιστεί, καθιστώντας την επεξεργαστική ισχύ, και κατά συνέπεια κάποιου είδους νοημοσύνη, καθολικά παρούσα.

Τα πρώτα υπολογιστικά συστήματα ήταν απομονωμένες οντότητες που επικοινωνούσαν μόνο με τους ανθρώπους-χειριστές τους, ενώ σήμερα τα συστήματα αυτά είναι συνήθως *διασυνδεδεμένα*. Είναι συνδεδεμένα *μέσω δικτύων* σε μεγάλα *καταναεμημένα* συστήματα (distributed systems). Το Διαδίκτυο (Internet) είναι το προφανές παράδειγμα: γίνεται όλο και πιο σπάνιο να βρει κανείς υπολογιστές που χρησιμοποιούνται σε εμπορικούς ή ακαδημαϊκούς τομείς χωρίς να έχουν τη δυνατότητα πρόσβασης στο Διαδίκτυο. Μέχρι σχετικά πρόσφατα, τα καταναεμημένα συστήματα και τα συστήματα ταυτόχρονου υπολογισμού θεωρούνταν από πολλούς παράξενα και δύσκολα "θηρία" που καλό ήταν να αποφεύγονται. Η πολύ ορατή και ταχεία ανάπτυξη του Διαδικτύου έχει (ελπίζω) διαλύσει αυτή την πεποίθηση για πάντα. Σήμερα, και στο μέλλον, τα καταναεμημένα συστήματα και τα συστήματα ταυτόχρονου υπολογισμού είναι ουσιαστικά ο κανόνας στην εμπορική και βιομηχανική πληροφο-

ρική, οδηγώντας μερικούς ερευνητές και επαγγελματίες στην αναθεώρηση των ιδίων των θεμελιωδών αρχών της επιστήμης των υπολογιστών, καθώς αναζητούν θεωρητικά μοντέλα που αντανακλούν καλύτερα τον υπολογισμό όπως αυτός συμβαίνει πραγματικά, δηλαδή κυρίως με τη μορφή μιας αλληλεπιδραστικής διαδικασίας.

Η τρίτη τάση είναι προς ακόμα περισσότερο *ευφυή* (intelligent) συστήματα. Με αυτό εννοώ ότι η *πολυπλοκότητα* των εργασιών που είμαστε σε θέση να αυτοματοποιήσουμε και να αναθέσουμε σε υπολογιστές αυξάνεται σταθερά. Όλο και περισσότερο κατανοούμε καλύτερα τον τρόπο με τον οποίο μπορούμε να κατασκευάζουμε υπολογιστικά συστήματα για την εκτέλεση εργασιών που θα ήταν αδιανόητες μόλις λίγο καιρό πριν.

Η επόμενη τάση είναι προς ολοένα αυξανόμενη ανάθεση. Για παράδειγμα, η ανάθεση κρίσιμων ως προς την ασφάλεια εργασιών σε υπολογιστικά συστήματα, όπως ο χειρισμός αεροσκαφών, είναι πλέον συνηθισμένη. Μάλιστα, στα αεροσκάφη fly-by-wire<sup>1</sup> συχνά εμπιστευόμαστε περισσότερο την κρίση ενός προγράμματος υπολογιστή από εκείνη έμπειρων πιλότων. Ανάθεση σημαίνει ότι *δίνουμε τον έλεγχο* σε υπολογιστικά συστήματα.

Η πέμπτη και τελευταία τάση είναι η σταθερή απομάκρυνση από μηχανοκεντρικές απόψεις για τον προγραμματισμό, και η σύγκλιση προς έννοιες και μοντέλα που αντανακλούν περισσότερο τον τρόπο με τον οποίο εμείς οι ίδιοι κατανοούμε τον κόσμο. Αυτή η τάση είναι προφανής σε όλους τους τρόπους με τους οποίους αλληλεπιδρούμε με τους υπολογιστές. Για παράδειγμα, στην πρώιμη περίοδο των υπολογιστών οι χρήστες αλληλεπιδρούσαν με τους υπολογιστές ρυθμίζοντας διακόπτες στον πίνακα ελέγχου της μηχανής. Η εσωτερική λειτουργία της συσκευής δεν ήταν με κανένα τρόπο κρυμμένη από το χρήστη — για να χρησιμοποιήσει κανείς με επιτυχία τον υπολογιστή, έπρεπε να κατανοεί πλήρως την εσωτερική δομή και λειτουργία του. Αυτές οι πρωτόγονες — και αντιπαραγωγικές — *διασυνδέσεις* (interfaces) έδωσαν τη θέση τους σε διασυνδέσεις γραμμής διαταγών (command line), όπου μπορούσε κανείς να αλληλεπιδρά με τη συσκευή μέσω ενός συνεχούς διαλόγου στον οποίο ο χρήστης έδινε εντολές τις οποίες μετά εκτελούσε η συσκευή. Αυτές οι διασυνδέσεις κυριάρχησαν μέχρι τη δεκαετία του 1980, όπου έδωσαν τη θέση τους σε διασυνδέσεις με το χρήστη μέσω γραφικών (graphical user interfaces) και στο υπόδειγμα άμεσου χειρισμού (direct manipulation paradigm), σύμφωνα με το οποίο ο έλεγχος της συσκευής από το χρήστη γίνεται με απευθείας χειρισμό εικονιδίων γραφικών που αντιστοιχούν σε αντικείμενα, όπως αρχεία και προγράμματα. Παρομοίως, στην πρώιμη περίοδο των υπολογιστών οι προγραμματιστές δεν είχαν άλλη επιλογή από το να προγραμματίζουν τους υπολογιστές τους με καθαρό κώδικα σε γλώσσα μηχανής, γεγονός που προϋπέθετε λεπτομερή κατανόηση της εσωτερικής δομής και λειτουργίας των μηχανών τους. Τα μεταγενέστερα υποδείγματα προγραμματισμού απομακρύνθηκαν από τέτοιες προσεγγίσεις χαμηλού επιπέδου: δείτε την εξέλιξη των γλωσσών συμβολομεταφραστών (assembler), με τη διαδικασιακή αφαίρεση (proce-

---

<sup>1</sup> ΣτΜ. Τα αεροσκάφη αυτά ελέγχονται από υπολογιστή. Ο άνθρωπος-χειριστής τους, που είναι ο πιλότος, δίνει δεδομένα στον υπολογιστή και αυτός υπολογίζει τις τελικές ενέργειες που πρέπει να εκτελεστούν, π.χ., μείωση ή αύξηση της ταχύτητας, του ύψους πτήσης, κλπ.



dural abstraction), τους αφηρημένους τύπους δεδομένων (abstract data types) και, πιο πρόσφατα, τα αντικείμενα (objects). Όλες αυτές οι εξελίξεις επέτρεψαν στους προγραμματιστές να αντιλαμβάνονται και να υλοποιούν λογισμικό με υψηλότερου επιπέδου — πιο ανθρωποκεντρικές — αφαιρετικές θεωρήσεις.

Αυτές οι τάσεις θέτουν μεγάλες προκλήσεις για τους δημιουργούς λογισμικού. Σε ό,τι αφορά την καθολικότητα και τη διασύνδεση, δεν γνωρίζουμε ακόμα ποιες τεχνικές θα μπορούσαν να χρησιμοποιηθούν για την ανάπτυξη συστημάτων έτσι ώστε να αξιοποιείται η καθολικά παρούσα επεξεργαστική ισχύς. Τα τρέχοντα μοντέλα ανάπτυξης λογισμικού αποδείχθηκαν δυστυχώς ανεπαρκή, ακόμα και όταν αντιμετωπίζουμε σχετικά μικρούς αριθμούς επεξεργαστών. Τι τεχνικές θα χρειαζόμασταν για συστήματα που αποτελούνται από  $10^{10}$  επεξεργαστές; Ο όρος *καθολική υπολογιστική* (global computing) εφευρέθηκε για να περιγράψει τέτοια αδιανόητα μεγάλα συστήματα.

Οι τάσεις για αυξανόμενη ανάθεση και νοημοσύνη απαιτούν την κατασκευή υπολογιστικών συστημάτων που είναι ικανά να δρουν αποτελεσματικά για λογαριασμό μας. Αυτό, με τη σειρά του, απαιτεί δύο ικανότητες. Η πρώτη είναι η ικανότητα των συστημάτων να λειτουργούν *ανεξάρτητα*, χωρίς την άμεση παρέμβασή μας. Η δεύτερη είναι η ικανότητα των υπολογιστικών συστημάτων να δρουν με τέτοιο τρόπο ώστε να *εκπροσωπούν τα συμφέροντά μας* όταν αλληλεπιδρούν με ανθρώπους ή άλλα συστήματα.

Η τάση για διασύνδεση και κατανομή έχει αναγνωριστεί εδώ και πολύ καιρό ως βασική πρόκληση στη σύγχρονη επιστήμη των υπολογιστών, και πολλές από τις προσπάθειες στον τομέα αυτό τις τελευταίες τρεις δεκαετίες έχουν αφιερωθεί στην ανάπτυξη εργαλείων λογισμικού και μηχανισμών που μας επιτρέπουν να κατασκευάζουμε κατανομημένα συστήματα με μεγαλύτερη ευκολία και αξιοπιστία. Αν, όμως, αυτό συνδυαστεί με την ανάγκη για συστήματα που μπορούν να εκπροσωπούν τα συμφέροντά μας, η κατανομή θέτει άλλα θεμελιώδη προβλήματα. Όταν ένα υπολογιστικό σύστημα που δρα για λογαριασμό μας πρέπει να αλληλεπιδρά με ένα άλλο υπολογιστικό σύστημα που εκπροσωπεί τα συμφέροντα κάποιου άλλου, μπορεί (στην πραγματικότητα, είναι πολύ πιθανό) αυτά τα συμφέροντα να μην ταυτίζονται. Είναι απαραίτητο να δώσουμε σε τέτοια συστήματα την ικανότητα να *συνεργάζονται* και να *πραγματοποιούν συμφωνίες* με άλλα συστήματα, με τον ίδιο περίπου τρόπο με τον οποίο εμείς συνεργαζόμαστε και πραγματοποιούμε συμφωνίες με άλλους στην καθημερινή μας ζωή. Αυτού του είδους η ικανότητα μέχρι πολύ πρόσφατα δεν αποτελούσε αντικείμενο μελέτης της επιστήμης των υπολογιστών.

Όλες μαζί, αυτές οι τάσεις οδήγησαν στην ανάδυση ενός νέου πεδίου στην επιστήμη της πληροφορικής: τα *πολυπρακτορικά συστήματα* (multiagent systems). Η ιδέα του πολυπρακτορικού συστήματος είναι πολύ απλή. Ο πράκτορας (agent) είναι ένα υπολογιστικό σύστημα ικανό για *ανεξάρτητη* δράση για λογαριασμό του χρήστη ή του ιδιοκτήτη του. Με άλλα λόγια, ο πράκτορας μπορεί να καταλάβει μόνος του τι χρειάζεται να κάνει για να ικανοποιήσει τους στόχους για τους οποίους σχεδιάστηκε, αντί να χρειάζεται ρητές υποδείξεις για το τι να κάνει σε κάθε χρονική στιγμή. Το πολυπρακτορικό σύστημα είναι ένα σύστημα που αποτελείται από έναν αριθμό

πρακτόρων οι οποίοι *αλληλεπιδρούν* μεταξύ τους, συνήθως ανταλλάσσοντας μηνύματα μέσω κάποιου δικτύου υπολογιστών. Στη γενικότερη περίπτωση, οι πράκτορες σε ένα πολυπρακτορικό σύστημα θα εκπροσωπούν ή θα δρουν για λογαριασμό χρηστών ή ιδιοκτητών με πολύ διαφορετικούς στόχους και κίνητρα. Για να *αλληλεπιδρούν* με επιτυχία, αυτοί οι πράκτορες θα πρέπει να έχουν την ικανότητα να *συνεργάζονται*, να *συντονίζονται*, και να *διαπραγματεύονται* με άλλους, με τον ίδιο περίπου τρόπο που εμείς συνεργαζόμαστε, συντονιζόμαστε, και διαπραγματευόμαστε με άλλους ανθρώπους στην καθημερινή μας ζωή.

Αυτό το βιβλίο αφορά τα πολυπρακτορικά συστήματα, και αναφέρεται στα δύο βασικά προβλήματα που υπονοήσαμε παραπάνω.

- Πώς μπορούμε να κατασκευάσουμε πράκτορες ικανούς για ανεξάρτητη, αυτόνομη δράση, έτσι ώστε να εκτελούν με επιτυχία τις εργασίες που τους αναθέτουμε;
- Πώς μπορούμε να κατασκευάσουμε πράκτορες ικανούς για αλληλεπίδραση (συνεργασία, συντονισμό, διαπραγμάτευση) με άλλους πράκτορες, έτσι ώστε να εκτελούν με επιτυχία τις εργασίες που τους αναθέτουμε, ιδιαίτερα όταν δεν μπορούμε να υποθέσουμε ότι οι άλλοι πράκτορες μοιράζονται τα ίδια ενδιαφέροντα και τους ίδιους στόχους;

Το πρώτο πρόβλημα είναι αυτό του *σχεδιασμού πράκτορα*, και το δεύτερο πρόβλημα είναι αυτό του *σχεδιασμού κοινωνίας*. Τα δύο προβλήματα δεν είναι ορθογώνια — για παράδειγμα, για να κατασκευάσουμε μια κοινωνία από πράκτορες που συνεργάζονται αποτελεσματικά, ίσως είναι χρήσιμο να δώσουμε στα μέλη της κοινωνίας μοντέλα των άλλων πρακτόρων που υπάρχουν σε αυτή. Η διάκριση μεταξύ των δύο αυτών ζητημάτων συχνά αναφέρεται ως *μικρο/μακρο* διάκριση. Στο υπόλοιπο αυτού του βιβλίου θα εξετάσουμε και τα δύο αυτά ζητήματα λεπτομερώς.

Οι ερευνητές των πολυπρακτορικών συστημάτων μπορεί να ασχολούνται κυρίως με την κατασκευή συστημάτων, αλλά αυτό σε καμία περίπτωση δεν είναι το μοναδικό τους ενδιαφέρον. Όπως και στον τομέα της Τεχνητής Νοημοσύνης (ΤΝ), τα ζητήματα που απασχολούν τον τομέα των πολυπρακτορικών συστημάτων επηρεάζουν βαθύτατα την κατανόησή μας για τον εαυτό μας. Η ΤΝ έχει κατεξοχήν επικεντρωθεί σε θέματα ατομικής νοημοσύνης. Ασφαλώς, όμως, ένα μεγάλο μέρος αυτού που μας κάνει μοναδικούς ως είδος, είναι η *κοινωνική* μας ικανότητα. Όχι μόνο επικοινωνούμε μεταξύ μας σε γλώσσες υψηλού επιπέδου, αλλά επίσης συνεργαζόμαστε, συντονιζόμαστε, και διαπραγματευόμαστε. Πολλά άλλα είδη έχουν κάποιου είδους κοινωνική ικανότητα — τα μυρμήγκια και κάποια άλλα κοινωνικά έντομα είναι ίσως τα γνωστότερα παραδείγματα — όμως κανένα άλλο είδος δεν πλησιάζει την πολυπλοκότητα της δικής μας κοινωνικής ικανότητας. Στα πολυπρακτορικά συστήματα μας ενδιαφέρουν ερωτήσεις όπως οι ακόλουθες:

- Πώς μπορεί να προκύψει συνεργασία σε κοινωνίες ιδιοτελών πρακτόρων;
- Τι είδους κοινές γλώσσες μπορούν να χρησιμοποιήσουν οι πράκτορες για να εκφράσουν τις πεποιθήσεις και τις φιλοδοξίες τους, τόσο σε ανθρώπους όσο και σε άλλους πράκτορες;
- Πώς μπορούν ιδιοτελείς πράκτορες να αναγνωρίσουν πότε οι πεποιθήσεις, οι στόχοι, ή οι ενέργειές τους έρχονται σε σύγκρουση, και πώς μπορούν να επιτύχουν μεταξύ τους συμφωνίες σε θέματα ατομικού συμφέροντος χωρίς να καταλήγουν στη σύγκρουση;
- Πώς μπορούν αυτόνομοι πράκτορες να συντονίζουν τις δραστηριότητές τους έτσι ώστε να πετυχαίνουν συνεργαζόμενοι στόχους;

Όλα αυτά τα ερωτήματα απασχολούν εν μέρει και άλλα γνωστικά πεδία (κυρίως τα οικονομικά και τις κοινωνικές επιστήμες), αλλά αυτό που καθιστά το πεδίο των πολυπρακτορικών συστημάτων μοναδικό και διακριτό είναι ότι υπογραμμίζει το γεγονός πως οι πράκτορες που μας ενδιαφέρουν είναι οντότητες *υπολογισμού* και *επεξεργασίας πληροφοριών*.

### **Το υπόλοιπο αυτού του κεφαλαίου**

Σκοπός του πρώτου κεφαλαίου είναι να σας προσανατολίσει για το υπόλοιπο βιβλίο. Το κεφάλαιο έχει την ακόλουθη δομή:

- Στην επόμενη ενότητα ξεκινάμε με κάποια σενάρια. Σκοπός αυτών των σεναρίων είναι να πάρετε μια αίσθηση για τα μακροπρόθεσμα οράματα που καθορίζουν τη δραστηριότητα στο πεδίο των πρακτόρων.
- Όπως συμβαίνει και στα ίδια τα πολυπρακτορικά συστήματα, δεν επιδιώκουν τον ίδιο σκοπό όλα τα μέλη της ερευνητικής κοινότητας που ασχολείται με τους πράκτορες. Για το λόγο αυτόν συνοψίζουμε τους διαφορετικούς τρόπους με τους οποίους διάφοροι ερευνητές σκέφτονται σχετικά με τα πολυπρακτορικά συστήματα.
- Τέλος, παρουσιάζουμε και αναπτύσσουμε μερικές από τις συνηθέστερες ενστάσεις στο πεδίο των πολυπρακτορικών συστημάτων.

## **1.1 Το όραμα**

Πολύ συχνά είναι δύσκολο να κατανοήσουμε τι κάνουν οι άνθρωποι χωρίς να κατανοήσουμε ποια είναι τα κίνητρά τους. Έτσι, σκοπός αυτής της ενότητας είναι να παρουσιάσουμε μερικά από τα κίνητρα για τη δραστηριότητα της κοινότητας που ασχολείται με πράκτορες. Αυτά τα κίνητρα έχουν τη μορφή μακροπρόθεσμων οραμάτων — είναι δηλαδή ιδέες για το πώς θα μπορούσαν να είναι τα πράγματα. Μια προειδοποίηση: αυτά τα οράματα είναι απλώς οράματα. Κανένα τους δεν είναι πιθανό να πραγματοποιηθεί στο άμεσο μέλλον. Αλλά για καθένα από αυτά γίνεται *ήδη* δου-

λειά προκειμένου να αναπτυχθούν οι τεχνολογίες που πιθανόν απαιτούνται για την πραγματοποίησή τους.

Εξαιτίας κάποιας απροσδόκητης βλάβης του συστήματος, ένας διαστημικός ανιχνευτής που πλησιάζει τον Κρόνο χάνει την επαφή του με το προσωπικό εδάφους του, το οποίο βρίσκεται στη Γη, με αποτέλεσμα τον αποπροσανατολισμό του. Αντί απλώς να εξαφανιστεί στο κενό, ο ανιχνευτής αναγνωρίζει ότι συνέβη βασική βλάβη συστήματος, κάνει διάγνωση και απομόνωση του σφάλματος, και επαναπροσανατολίζεται σωστά έτσι ώστε να ανακτήσει την επαφή του με το προσωπικό εδάφους.

Το βασικό ζήτημα εδώ είναι η ικανότητα του διαστημικού ανιχνευτή να δρα αυτόνομα. Καταρχήν ο ανιχνευτής πρέπει να αναγνωρίσει ότι συνέβη βλάβη, και μετά πρέπει να σκεφτεί τι χρειάζεται να γίνει και με ποιον τρόπο πρέπει να γίνει αυτό. Τέλος, ο ανιχνευτής πρέπει να εκτελέσει τις ενέργειες που επέλεξε, και προφανώς να παρακολουθήσει τι συμβαίνει ώστε να εξασφαλίσει ότι όλα πηγαίνουν καλά. Αν προκύψουν και άλλα προβλήματα, ο ανιχνευτής θα πρέπει να τα αναγνωρίσει και να αντιδράσει κατάλληλα. Παρατηρήστε ότι αυτός είναι ένας τρόπος συμπεριφοράς τον οποίο (εμείς οι άνθρωποι) θεωρούμε εύκολο: εκδηλώνουμε τέτοια συμπεριφορά καθημερινά, αν τύχει να χάσουμε μια πτήση ή αν σκάσει το λάστιχο του αυτοκινήτου μας καθώς οδηγούμε προς τη δουλειά μας. Όπως όμως θα δούμε, είναι πολύ δύσκολο να σχεδιάσουμε προγράμματα υπολογιστή που επιδεικνύουν αυτού του είδους τη συμπεριφορά.

Η αποστολή Deep Space 1 (DS1) της NASA αποτελεί παράδειγμα συστήματος που είναι κοντά σε αυτό το είδος σεναρίου. Ο DS1, που εκτοξεύθηκε από το ακρωτήριο Canaveral στις 24 Οκτωβρίου 1998, ήταν ο πρώτος διαστημικός ανιχνευτής που διέθετε αυτόνομο σύστημα ελέγχου βασισμένο σε πράκτορα (Mussettola *et al.*, 1998). Πριν τον DS1, οι διαστημικές αποστολές απαιτούσαν προσωπικό εδάφους έως και 300 μελών για να παρακολουθεί συνεχώς την εξέλιξή τους. Αυτό το προσωπικό εδάφους έπαιρνε όλες τις απαραίτητες αποφάσεις ελέγχου για λογαριασμό του ανιχνευτή, και τις μετέδιδε συνεχώς στον ανιχνευτή ώστε αυτός να τις εκτελεί. Αν λάβουμε υπόψη τη διάρκεια των συνηθισμένων αποστολών εξερεύνησης πλανητών, αυτή η διαδικασία ήταν δαπανηρή και, αν ποτέ χρειαζόταν *γρήγορη* λήψη αποφάσεων, απλώς δεν ήταν πρακτική. Το αυτόνομο σύστημα ελέγχου του DS1 ήταν σε θέση να παίρνει πολλές σημαντικές αποφάσεις από μόνο του. Έτσι η αποστολή ήταν πιο ασφαλής, ιδιαίτερα σε αιφνίδια απροσδόκητα προβλήματα, και επιπλέον υπήρχε το πολύ επιθυμητό παράπλευρο αποτέλεσμα της μείωσης του συνολικού κόστους της.

Το επόμενο σενάριο αναφέρεται και πάλι σε "εναέριο" περιστατικό, αλλά τουλάχιστον είναι μέσα στα όρια του πλανήτη μας.

Ξαφνικά προκαλείται αστοχία σε ένα βασικό σύστημα ελέγχου εναέριας κυκλοφορίας στο κεντρικό αεροδρόμιο της Ruritania<sup>2</sup>, αφήνοντας τις πτή-

---

<sup>2</sup> ΣτΜ. Συνηθισμένη ονομασία που χρησιμοποιούν αγγλοσάξωνες συγγραφείς για να δηλώσουν μια φανταστική, ανύπαρκτη, χώρα.

σεις στην περιοχή γύρω από το αεροδρόμιο χωρίς υποστήριξη. Ευτυχώς, η βλάβη του συστήματος εντοπίζεται από παρόμοια αυτόνομα συστήματα ελέγχου εναέριας κυκλοφορίας σε γειτονικά αεροδρόμια, τα οποία συνεργάζονται για να βρουν και να χειριστούν όλες τις πτήσεις που επηρεάζονται. Το ενδεχομένως καταστροφικό περιστατικό ξεπερνιέται χωρίς να προκληθεί ατύχημα.

Υπάρχουν πολλά βασικά ζητήματα σε αυτό το σενάριο. Το πρώτο είναι η ικανότητα των συστημάτων να αναλαμβάνουν πρωτοβουλία όταν το απαιτούν οι περιστάσεις. Το δεύτερο είναι η ικανότητα των πρακτόρων να *συνεργάζονται* για την επίλυση προβλημάτων που είναι πέρα από τις δυνατότητες οποιουδήποτε μεμονωμένου πράκτορα. Το είδος συνεργασίας που απαιτείται σε αυτό το σενάριο μελετήθηκε εκτενώς στο ερευνητικό πρόγραμμα "Distributed Vehicle Monitoring Testbed (DVTM)", το οποίο εκπονήθηκε μεταξύ 1981 και 1991 και αφορούσε τη δημιουργία μιας κατανεμημένης πλατφόρμας παρακολούθησης οχημάτων (δείτε, για παράδειγμα, το κείμενο του Durfee, 1988). Το DVTM προσομοιώνει ένα δίκτυο από πράκτορες που παρακολουθούν οχήματα, όπου κάθε πράκτορας είναι ένας επιλυτής προβλημάτων που αναλύει δεδομένα τα οποία λαμβάνει μέσω των αισθητήρων του, έτσι ώστε να προσδιορίζει, να εντοπίζει, και να παρακολουθεί οχήματα που κινούνται στο χώρο. Κάθε πράκτορας συνήθως συσχετίζεται με έναν αισθητήρα, ο οποίος έχει μόνο μερική άποψη του συνολικού χώρου. Κατά συνέπεια, οι πράκτορες πρέπει να συνεργάζονται για να παρακολουθούν την πρόοδο των οχημάτων σε όλο τον αντιληπτό χώρο. Τα συστήματα ελέγχου εναέριας κυκλοφορίας αποτελούν τυπική εφαρμογή της έρευνας γύρω από τους πράκτορες από την εποχή της εργασίας του Cammarata και των συναδέλφων του, στις αρχές της δεκαετίας του 1980 (Cammarata *et al.*, 1983). Μια πρόσφατη πολυπρακτορική εφαρμογή ελέγχου εναέριας κυκλοφορίας είναι το σύστημα OASIS που υλοποιήθηκε για χρήση στο αεροδρόμιο του Σίδνεϊ στην Αυστραλία (Ljunberg και Lucas, 1992).

Οι περισσότεροι από εμάς δεν ασχολούμαστε ούτε με το σχεδιασμό των συστημάτων ελέγχου για τους διαστημικούς ανιχνευτές της NASA, ούτε με το σχεδιασμό συστημάτων για κρίσιμες εφαρμογές ασφάλειας, όπως τα συστήματα ελέγχου εναέριας κυκλοφορίας. Έτσι, ας εξετάσουμε τώρα μια περίπτωση που βρίσκεται πιο κοντά στην καθημερινή ζωή των περισσότερων από εμάς.

Μετά τον πλέον υγρό και κρύο χειμώνα που έχει ποτέ σημειωθεί στη περιοχή που ζείτε, χρειάζεστε επείγοντως διακοπές σε κάποιο ζεστό και ξηρό μέρος. Αφού προσδιορίσετε τις απαιτήσεις σας στον προσωπικό ψηφιακό βοηθό σας (Personal Digital Assistant, PDA), αυτός ξεκινά μια διαπραγματεύση με πολλές διαφορετικές τοποθεσίες Ιστού οι οποίες παρέχουν υπηρεσίες όπως κρατήσεις πτήσεων και δωματίων ξενοδοχείων, και ενοικιάσεις αυτοκινήτων. Μετά από σκληρές διαπραγματεύσεις για λογαριασμό σας με πολλές τοποθεσίες Ιστού, ο προσωπικός ψηφιακός βοηθός σας παρουσιάζει ένα πακέτο διακοπών.

Από όλα τα σενάρια που αναφέραμε, αυτό είναι ίσως εκείνο που είναι πιο εύκολο να γίνει πραγματικότητα. Υπάρχουν πολλές τοποθεσίες Ιστού που σας επιτρέπουν να ψάξετε για διακοπές της τελευταίας στιγμής, αλλά, απ' όσο γνωρίζω, κατά το χρόνο συγγραφής αυτού του βιβλίου καμιά από αυτές δεν πρόσφερε τη δυνατότητα ενεργούς διαπραγμάτευσης σε πραγματικό χρόνο για τη δημιουργία ενός προσαρμοσμένου πακέτου διακοπών, από ένα εύρος παρόχων υπηρεσιών. Υπάρχουν πολλά βασικά ερευνητικά προβλήματα που πρέπει να επιλυθούν προκειμένου να είναι εφικτό ένα τέτοιο σενάριο. Για παράδειγμα:

- Πώς ορίζετε τις προτιμήσεις σας στον πράκτορά σας;
- Πώς μπορεί ο πράκτοράς σας να συγκρίνει διαφορετικές προσφορές από διαφορετικούς πωλητές;
- Τι αλγορίθμους μπορεί να χρησιμοποιήσει ο πράκτοράς σας για να διαπραγματευτεί με άλλους πράκτορες (ώστε να εξασφαλίσει ότι δεν θα ξοδέψετε μια περιουσία);

Η ικανότητα διαπραγμάτευσης με τον τρόπο που υποδεικνύει το σενάριο είναι ιδιαίτερα πολύτιμη. Για παράδειγμα, κάθε χρόνο η Ευρωπαϊκή Επιτροπή πραγματοποιεί χιλιάδες προκηρύξεις εκδήλωσης ενδιαφέροντος για διάφορα έργα. Η γραφειοκρατία που απαιτείται για τη διαχείριση αυτής της διαδικασίας έχει τεράστιο κόστος. Η ικανότητα αυτοματοποίησης της διαδικασίας προσφοράς και διαπραγμάτευσης θα εξοικονομούσε τεράστια χρηματικά ποσά (των *φορολογουμένων!*). Παρόμοιες καταστάσεις προκύπτουν σε κυβερνητικές οργανώσεις σε ολόκληρο τον κόσμο — ένα καλό παράδειγμα είναι ο στρατός των ΗΠΑ. Έτσι, το να μπορέσουμε να αυτοματοποιήσουμε τη διαδικασία με την οποία οι πράκτορες λογισμικού επιτυγχάνουν αμοιβαία αποδεκτές συμφωνίες σε θέματα κοινού ενδιαφέροντος δεν είναι απλώς ένας αφηρημένος προβληματισμός — μπορεί να επηρεάσει σε σημαντικό βαθμό τη ζωή μας (για παράδειγμα, το ύψος του φόρου που πληρώνουμε).

## 1.2 Μερικές απόψεις για το πεδίο

Το πεδίο των πολυπρακτορικών συστημάτων είναι σε πολύ μεγάλο βαθμό διεπιστημονικό: δέχεται στοιχεία από πολύ διαφορετικούς τομείς, όπως τα οικονομικά, η φιλοσοφία, η λογική, η οικολογία, και οι κοινωνικές επιστήμες. Δεν θα πρέπει λοιπόν να μας εκπλήσσει η ύπαρξη πολλών διαφορετικών απόψεων για το ποιο είναι το αντικείμενο της μελέτης των πολυπρακτορικών συστημάτων. Σε αυτή την ενότητα θα συνοψίσουμε μερικές από τις βασικές απόψεις.

### *Οι πράκτορες ως υπόδειγμα για την τεχνολογία λογισμικού*

Οι μηχανικοί λογισμικού αποκτούν όλο και καλύτερη αντίληψη των χαρακτηριστικών της πολυπλοκότητας στο λογισμικό. Είναι πλέον ευρέως αποδεκτό ότι η *αλληλεπίδραση* αποτελεί μάλλον το πιο σημαντικό χαρακτηριστικό του πολύπλοκου λο-

γισμικού. Συνήθως η ορθή και αποδοτική κατασκευή αρχιτεκτονικών λογισμικού που περιέχουν πολλά συστατικά στοιχεία τα οποία αλληλεπιδρούν με δυναμικό τρόπο, το καθένα με το δικό του νήμα ελέγχου, και τα οποία σχετίζονται με πολύπλοκα, συντονισμένα πρωτόκολλα, είναι κατά πολύ πιο πολύπλοκη από την κατασκευή αρχιτεκτονικών λογισμικού οι οποίες απλώς υπολογίζουν μια συνάρτηση κάποιας εισόδου μέσω ενός μοναδικού νήματος ελέγχου. Δυστυχώς, πολλές (αν όχι οι περισσότερες) εφαρμογές στον πραγματικό κόσμο έχουν ακριβώς αυτά τα χαρακτηριστικά. Κατά συνέπεια, ένα μείζον ερευνητικό θέμα στην επιστήμη των υπολογιστών, τουλάχιστον τις τελευταίες δύο δεκαετίες, είναι η ανάπτυξη εργαλείων και τεχνικών για τη μοντελοποίηση, την κατανόηση, και την υλοποίηση συστημάτων στα οποία η αλληλεπίδραση αποτελεί τον κανόνα. Μάλιστα, πολλοί ερευνητές πλέον πιστεύουν ότι στο μέλλον θα αντιλαμβανόμαστε τον ίδιο τον υπολογισμό κυρίως ως διαδικασία αλληλεπίδρασης. Όπως ακριβώς μπορούμε να κατανοήσουμε ότι πολλά συστήματα αποτελούνται από ουσιαστικά παθητικά αντικείμενα, τα οποία βρίσκονται σε μια κατάσταση και στα οποία μπορούμε να εκτελούμε πράξεις, έτσι μπορούμε να κατανοήσουμε ότι πολλά άλλα συστήματα αποτελούνται από ημιαυτόνομους πράκτορες που αλληλεπιδρούν μεταξύ τους. Αυτή η συνειδητοποίηση οδήγησε στην αύξηση του ενδιαφέροντος για τους πράκτορες ως ένα νέου υποδείγματος για την τεχνολογία λογισμικού.

Όπως αναφέραμε στην αρχή του κεφαλαίου, η τάση στην πληροφορική είναι — και θα εξακολουθήσει να είναι — προς τα περισσότερο καθολικά, διασυνδεδεμένα υπολογιστικά συστήματα. Η ανάπτυξη υποδειγμάτων λογισμικού που μπορούν να αξιοποιήσουν το δυναμικό τέτοιων συστημάτων είναι ίσως η μεγαλύτερη πρόκληση για την πληροφορική στην αρχή του 21<sup>ου</sup> αιώνα. Οι πράκτορες φαίνεται να είναι ισχυροί υποψήφιοι για ένα τέτοιο υπόδειγμα.

### ***Οι πράκτορες ως εργαλείο για την κατανόηση των ανθρώπινων κοινωνιών***

Στη δημοφιλή τριλογία επιστημονικής φαντασίας *Foundation* (Γαλαξιακή αυτοκρατορία) του Ισαάκ Ασίμωφ, αποδίδεται σε έναν ήρωα με το όνομα Hari Seldon η εφεύρεση μιας επιστήμης την οποία ο Ασίμωφ αποκαλεί "ψυχοϊστορία" (psychohistory). Η ψυχοϊστορία υποτίθεται ότι είναι ένας συνδυασμός ψυχολογίας, ιστορίας, και οικονομικών, που επιτρέπει στον Seldon να προβλέπει τη συμπεριφορά των ανθρώπινων κοινωνιών μετά από εκατοντάδες χρόνια στο μέλλον. Μάλιστα, με την ψυχοϊστορία ο Seldon μπορεί να προβλέπει την άμεση κατάρρευση μιας κοινωνίας. Η ψυχοϊστορία είναι ενδιαφέρον επινόημα για την πλοκή ενός έργου, αλλά ανήκει αποκλειστικά στην επιστημονική φαντασία. Υπάρχουν υπερβολικά πολλές μεταβλητές και άγνωστες ποσότητες στις ανθρώπινες κοινωνίες για να επιτύχουμε κάτι περισσότερο από την πρόβλεψη πολύ γενικών τάσεων στο βραχυπρόθεσμο μέλλον, αλλά ακόμα και τότε είναι γνωστό ότι η διαδικασία είναι επιρρεπής σε ενοχλητικά λάθη. Αυτή η κατάσταση δεν είναι πιθανό να αλλάξει στο άμεσο μέλλον. Παρόλα αυτά, τα πολυπρακτορικά συστήματα αποτελούν ενδιαφέρον και καινοτόμο εργαλείο

# Εισαγωγή στα Πολυπρακτορικά συστήματα



M I C H A E L W O O L D R I D G E

Τα πολυπρακτορικά συστήματα (multiagent systems) αντιπροσωπεύουν ένα νέο τρόπο σύλληψης και υλοποίησης του κατανεμημένου λογισμικού. Το βιβλίο **Εισαγωγή στα πολυπρακτορικά συστήματα** είναι το πρώτο σύγχρονο εγχειρίδιο για αυτό το σημαντικό θέμα. Το βιβλίο παρέχει μια αναλυτική εισαγωγική παρουσίαση των ευφυών πρακτόρων (intelligent agents) και των πολυπρακτορικών συστημάτων, χωρίς να απαιτεί κάποια εξειδικευμένη γνώση. Παρουσιάζει την ιδέα των πρακτόρων (agents) ως συστημάτων λογισμικού που μπορούν να δρουν αυτόνομα, και αναλύει διεξοδικά τα ζητήματα:

- του τρόπου κατασκευής των πρακτόρων
- του τρόπου επίτευξης συμφωνιών μεταξύ των πρακτόρων
- των γλωσσών που μπορούν να χρησιμοποιούν οι πράκτορες για να επικοινωνούν μεταξύ τους
- της συνεργασίας και του συντονισμού για τα συστήματα πρακτόρων
- της εφαρμογής της τεχνολογίας πρακτόρων

Σχεδιασμένο και γραμμένο ειδικά για φοιτητές και επαγγελματίες των υπολογιστών, το βιβλίο αυτό διαθέτει επίσης μέσω του Ιστού εκτενείς υποστηρικτικούς πόρους διδασκαλίας, οι οποίοι περιλαμβάνουν ένα πλήρες σύνολο από διαφάνειες για διαλέξεις. Εμπλουτισμένο με πάμπολλα άτυπα παραδείγματα και με περισσότερες από 500 παραπομπές, το βιβλίο **Εισαγωγή στα πολυπρακτορικά συστήματα** είναι απαραίτητο για όσους ενδιαφέρονται για το συγκεκριμένο θέμα.

"... μια συνεκτική και υπέροχα διαυγής εισαγωγή στο πεδίο του βασισμένου σε πράκτορες υπολογισμού ... Ο Michael Wooldridge κατάφερε με εξαιρετικό τρόπο να μας δώσει το "απόσταγμα" της θεωρίας και της πρακτικής των πολυπρακτορικών συστημάτων, προς όφελος των μελλοντικών γενεών σπουδαστών."

Nick Jennings, Πανεπιστήμιο του Southampton

## Ο ΣΥΓΓΡΑΦΕΑΣ

Ο Michael Wooldridge είναι Καθηγητής της Επιστήμης Υπολογιστών στο Πανεπιστήμιο του Λίβερπουλ. Πήρε το PhD του το 1992, για την εργασία του με θέμα τα πολυπρακτορικά συστήματα, και από τότε έχει συμμετάσχει ενεργά στην έρευνα για τα πολυπρακτορικά συστήματα.

Για περισσότερες πληροφορίες επισκεφθείτε τις διεύθυνσεις:  
<http://www.csc.liv.ac.uk/~mjw/pubs/imas/>  
<http://www.wiley.com>

 **ΕΚΔΟΣΕΙΣ  
ΚΛΕΙΔΑΡΙΘΜΟΣ**

Στουρνάρα 27B, 10682 Αθήνα, Τηλ. 210-3832044

Επισκεφθείτε μας στο Internet:  
[www.kildarithmos.gr](http://www.kildarithmos.gr)

ISBN 978-960-461-125-6



9 789604 611256