

ΕΣΠΑ 2007-2013: Προγράμματα «ΑΝΤΑΓΩΝΙΣΤΙΚΟΤΗΤΑ & ΕΠΙΧΕΙΡΗΜΑΤΙΚΟΤΗΤΑ» και ΠΕΡΙΦΕΡΕΙΩΝ ΣΕ ΜΕΤΑΒΑΣΗ

Δράση Εθνικής εμβέλειας: «Ενίσχυση Νέων και Μικρομεσαίων Επιχειρήσεων»

Πράξη: «Υποστήριξη Νέων Επιχειρήσεων για Δραστηριότητες Έρευνας & Τεχνολογικής Ανάπτυξης»

Προκήρυξη 2009 – Α' κύκλος

«ΔΙΚΤΥΟ ΚΑΤΑΝΕΜΗΜΕΝΩΝ ΕΥΦΥΩΝ ΠΡΑΚΤΟΡΩΝ ΑΤΜΟΣΦΑΙΡΙΚΗΣ ΠΟΙΟΤΗΤΑΣ (Π1.1)

ΠΡΟΣΤΑΣΙΑ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΑΠΟ ΜΕΛΛΟΝΤΙΚΕΣ ΔΥΣΛΕΙΤΟΥΡΓΙΕΣ ΚΑΙ ΧΡΟΝΙΚΟΣ ΣΥΝΤΟΝΙΣΜΟΣ ΠΡΑΚΤΟΡΩΝ (Π1.2)

ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΚΑΙ ΤΕΛΙΚΟ ΠΕΡΙΒΑΛΛΟΝ ΧΡΗΣΤΗ (Π1.3)»

ΠΑΡΑΔΟΤΕΟ: Π1.1, Π1.2, Π1.3

ΕΝΟΤΗΤΑ ΕΡΓΑΣΙΑΣ: 1

9NEWΕ2009

Κατανεμημένο Ευφύες Σύστημα παρακολούθησης και αντιμετώπισης έντονων αλλαγών της Ατμοσφαιρικής Ρύπανσης και του Δείκτη Δυσφορίας σε Πραγματικό Χρόνο μέσα από μία συνεχή διαδικασία

“AID” Air pollution Intelligent Defense - <http://www.karteco.gr/AID/>



ΚΑΡΤΕΡΗΣ Α- ΚΑΡΤΕΡΗΣ Μ ΟΕ
(kartECO-Σύμβουλοι Μηχανικοί Περιβάλλοντος & Ενέργειας)
ΤΚ 57001, ΤΘ, 60824
Τηλ 2310 365441, Fax: 2310 365442
www.karteco.gr , www.karteco.gr/greenroofs
E-mail: a.karteris@karteco.gr

Υπεργολάβοι: Δημοκρίτειο Πανεπιστήμιο Θράκης,
Τμ. Δασολογίας, Εργαστήριο Δασικής Πληροφορικής,
Ορεστιάδα

2013



ΠΕΡΙΕΧΟΜΕΝΑ

1. Εισαγωγή.....	1
2. Γενικές απαιτήσεις συστήματος.....	2
3. Χρονοδιάγραμμα υλοποίησης εφαρμογής - ΕΕ1.....	4
4. Δίκτυο καταναμεμημένων ευφυών πρακτόρων	7
4.1 Περιγραφή καταναμεμημένου δικτύου ευφυών πρακτόρων.....	7
4.2 Πράκτορες αντίληψης περιβάλλοντος	8
4.3 Πράκτορες αξιολόγησης μετρήσεων.....	11
4.4 Πράκτορας διαχείρισης συστήματος	14
4.5 Πράκτορες λήψης αποφάσεων.....	17
4.5.1 Πράκτορας ελέγχου δείκτη δυσφορίας.....	17
4.5.2 Πράκτορας ελέγχου συγκέντρωσης Διοξειδίου του Αζώτου.....	19
4.5.3 Πράκτορας ελέγχου συγκέντρωσης Όζοντος	22
4.6 Πράκτορες Ενεργειών	23
4.6.1 Πράκτορας υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης.....	23
4.7 Διασύνδεση του Συστήματος – Χρήση σε κινητές συσκευές	25
4.8 Γραφικό περιβάλλον αλληλεπίδρασης – GUI.....	26
4.8.1 GUI πράκτορα αισθητήρα.....	26
4.8.2 GUI πράκτορα αξιολόγησης	27
4.8.3 GUI πράκτορα διαχείρισης συστήματος	28
4.8.4 GUI πράκτορα ελέγχου δείκτη δυσφορίας	30
4.8.5 GUI πράκτορα ελέγχου συγκέντρωσης διοξειδίου του αζώτου και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων.....	30
4.8.6 GUI πράκτορα ελέγχου συγκέντρωσης όζοντος και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων	31
4.8.7 GUI πράκτορα υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου.....	32
5.Ολοκλήρωση του συστήματος – Λειτουργία.....	34

1. Εισαγωγή

Η παρούσα έκθεση αποτελεί σε συνδυασμό με την εφαρμογή που αναπτύχτηκε το τελικό παραδοτέο της Ενότητας Εργασίας 1 (ΕΕ1) «ΑΝΑΠΤΥΞΗ ΛΟΓΙΣΜΙΚΟΥ ΕΥΦΥΩΝ ΠΡΑΚΤΟΡΩΝ ΑΥΤΟΜΑΤΗΣ ΜΑΘΗΣΗΣ» σύμφωνα με το Τεχνικό Παράρτημα του έργου.

Το διάστημα υλοποίησης της ΕΕ1 ήταν από την έναρξη του έργου (26/9/2011) έως και τις 26/1/2013, διάστημα το οποίο και καλύπτεται από την παρούσα έκθεση.

Η παραπάνω ενότητα εργασίας περιλαμβάνει το εξής τρία παραδοτέα:

- Δίκτυο κατανεμημένων ευφύων πρακτόρων ατμοσφαιρικής ποιότητας (Π1.1)
- Προστασία του συστήματος από μελλοντικές δυσλειτουργίες και χρονικός συντονισμός πρακτόρων (Π1.2)
- Προγραμματισμός και τελικό περιβάλλον χρήστη (Π1.3)

Τα παραπάνω παραδοτέα αποτελούν υποσύνολα εργασιών ενός συνολικού αντικειμένου και παρόλο που στην περιγραφή της ενότητας θα γίνει κάποιος διαχωρισμός των εργασιών που υλοποιήθηκαν ανά πακέτο εργασίας, η αναλυτική περιγραφή της ΕΕ θα γίνει συνολικά και όχι ανά ΠΕ.

2. Γενικές απαιτήσεις συστήματος

Κατά τον σχεδιασμό και την κατασκευή του πολυπρακτορικού συστήματος ελήφθησαν υπόψη και υιοθετηθήκαν όλες οι σχεδιαστικές απαιτήσεις που τεθήκαν από την φορέα χρηματοδότησης στην ενότητα εργασίας 2 «ΠΡΩΤΟΚΟΛΛΟ ΣΥΜΠΕΡΙΦΟΡΩΝ ΟΛΩΝ ΤΩΝ ΠΡΑΚΤΟΡΩΝ -ΠΑΡΑΜΕΤΡΟΙ, ΟΡΙΑ ΚΑΙ ΧΑΡΑΚΤΗΡΙΣΜΟΣ ΑΕΡΙΑΣ ΡΥΠΑΝΣΗΣ».

Πιο συγκεκριμένα:

- Υιοθετήθηκε η χρήση των παρακάτω πέντε τύπων πρακτόρων
 - Πράκτορες Αισθητήρες - Αντίληψης Περιβάλλοντος
 - Πράκτορες Αξιολόγησης
 - Πράκτορας Διαχείρισης Συστήματος
 - Πράκτορες Λήψης Αποφάσεων
 - Πράκτορες Ενεργειών
- Δημιουργήθηκαν τα τρία Ασαφή Σύνολα που ακολουθούν και τα οποία χρησιμοποιήθηκαν από τους πράκτορες αξιολόγησης για την αξιολόγηση των μετρήσεων των πρακτόρων αισθητήρων.
 - ασαφές σύνολο “Normal”
 - ασαφές σύνολο “High”
 - ασαφές σύνολο “Critical”

Οι βαθμοί μέλους στα ασαφή σύνολα “Normal” και “Critical” υπολογίζονται με χρήση Ημί-τραπεζοειδών συναρτήσεων βαθμού μέλους, ενώ το ασαφές σύνολο “High” υπολογίζεται με χρήση τριγωνικής συνάρτησης βαθμού μέλους. Οι συναρτήσεις υπολογισμού του βαθμού μέλους που χρησιμοποιήθηκαν είναι οι:

$$\mu_s(x; a, b) = \begin{cases} 1, & x \leq a \\ \frac{b-x}{b-a}, & a < x < b \\ 0, & b \leq x \end{cases}$$

Συνάρτηση 1: Ημι - Τραπεζοειδής συνάρτηση μέλους ασαφούς συνόλου Normal

$$\mu_s(x; f, g) = \begin{cases} 0, & x \leq f \\ \frac{x-f}{g-f}, & f < x < g \\ 1, & g \leq x \end{cases}$$

Συνάρτηση 2: Ημι - Τραπεζοειδής συνάρτηση μέλους ασαφούς συνόλου Critical

$$\mu_s(x; c, d, e) = \begin{cases} 0, & x \leq c \\ \frac{x - c}{d - c}, & c < x \leq d \\ \frac{e - x}{e - d}, & d < x < e \\ 0, & e \leq x \end{cases}$$

Συνάρτηση 3: Τριγωνική συνάρτηση μέλους ασαφούς συνόλου High

Τα όρια (a, b, c, d, e, f, g) που χρησιμοποιούνται στις παραπάνω συναρτήσεις είναι σύμφωνα με το πρωτόκολλο συμπεριφοράς πρακτόρων και φαίνονται στον παρακάτω πίνακα.

Πίνακας 1: Όρια συναρτήσεων ανά ρύπο ή παράγοντα

Παράγοντας	a	b	c	d	e	f	g	Μονάδα
Temperature	20	30	20	30	38	30	38	°C
NO ₂	120	220	120	250	300	270	360	μg/m ³
O ₃	120	180	120	180	210	180	240	μg/m ³
R. Humidity	40	60	40	60	80	60	80	%

- Εκτός από τους πράκτορες αισθητήρες και αξιολόγησης για τους παραπάνω παράγοντες το σύστημα σχεδιάστηκε να υποστηρίζει άμεση προσθήκη πρακτόρων και για το μονοξείδιο του αζώτου (NO).
- Σχεδιάστηκαν και κατασκευάστηκαν οι αυτόνομοι Πράκτορες Λήψης Αποφάσεων:
 - Πράκτορας Ελέγχου Δείκτη Δυσφορίας
 - Πράκτορας Ελέγχου Όζοντος
 - Πράκτορας Ελέγχου Διοξειδίου του αζώτου
- Σχεδιάστηκε και κατασκευάστηκε ο Πράκτορας εικονικής επέμβασης στο κυκλοφοριακό δακτύλιο της Θεσσαλονίκης
- Ελήφθησαν υπόψη κατά τη σχεδίαση οι προϋποθέσεις που τέθηκαν σχετικά με την γενική αρχιτεκτονική του συστήματος όπως:
 - Τα δομικά του συστατικά και οι πράκτορες του συστήματος θα εκτελούνται σε απομακρυσμένους σταθμούς.
 - Φορητότητα του συστήματος.
 - Κεντρικός εποπτικός έλεγχος λειτουργίας και διαθεσιμότητας δεδομένων.
 - Ταυτόχρονη λειτουργία με ασαφή και αλγεβρική λογική.
- Ενσωματώθηκαν όλες οι συμπεριφορές πρακτόρων που αναφέρονται στο κεφάλαιο 3.5 του παραδοτέου Π2.1 του φορέα χρηματοδότησης, καθώς και επιπλέον συμπεριφορές που κρίθηκαν απαραίτητες κατά την σχεδίαση του συστήματος και αναφέρονται στη συνέχεια.

3. Χρονοδιάγραμμα υλοποίησης εφαρμογής - ΕΕ1

Από την ημερομηνία έναρξης του έργου και μέχρι την ολοκλήρωση της ΕΕ1 στις 26/1/2013 έχουν υλοποιηθεί, με βάση τις επιμέρους εκθέσεις προόδου του έργου στην διαχειριστική επιτροπή του ΔΠΘ, οι παρακάτω εργασίες ανά πακέτο εργασίας.

- Πακέτο εργασίας Π1.1 «Δίκτυο Κατανεμημένων Ευφύων Πρακτόρων Ατμοσφαιρικής Ποιότητας»: Από 27/9/2011 έως 31/12/2011

- Παραμετροποίηση και εγκατάσταση των απαραίτητων εφαρμογών στο εργαστήριο Δασικής Πληροφορικής του τμήματος Δασολογίας και Διαχείρισης Περιβάλλοντος και Φυσικών Πόρων έτσι ώστε σε αυτό να μπορέσει να ολοκληρωθεί ο σχεδιασμός, ο προγραμματισμός, ο αρχικός έλεγχος και η δικτυακή λειτουργία του συστήματος που θα αναπτυχθεί καθώς και η πιλοτική - δοκιμαστική λειτουργία που προβλέπεται στην ενότητα εργασίας 3.
- Μοντελοποίηση των ασαφών συνόλων με χρήση συναρτήσεων Java ώστε αυτές να ενσωματωθούν στους πράκτορες αξιολόγησης
- Καθορισμός των βασικών αντικειμένων και μεθόδων του κάθε πράκτορα ώστε να ξεκινήσει ο προγραμματισμός τους.

Από 1/1/2012 έως 29/2/2012

- Κατασκευή αυτόνομων προγραμμάτων σε java για ενσωμάτωση των συναρτήσεων ασαφών συνόλων με σκοπό τον έλεγχο αυτών με χρήση ιστορικών δεδομένων ατμοσφαιρικής ρύπανσης τα οποία προήρθαν από το Υπουργείο περιβάλλοντος και αφορούσαν διάφορους σταθμούς της ευρύτερης περιοχής Αττικής.
- Προγραμματισμός του σκελετού του συστήματος και της βασικής δομής των πρακτόρων έτσι ώστε αυτοί να αποκτήσουν βασική λειτουργικότητα και δυνατότητα ανταλλαγής μηνυμάτων.

Από 1/3/2012 έως 30/9/2012

- Ολοκλήρωση της κατασκευής των επιμέρους πρακτόρων του συστήματος καθώς και του πρωτοκόλλου επικοινωνίας μεταξύ των πρακτόρων (τύποι μηνυμάτων και χρήση τους)
- Τυποποίηση της ονοματολογίας και των ρυθμίσεων των πρακτόρων με βάση την δομή την οποία θα εγκατασταθεί στο σύστημα στην συνέχεια του έργου (δικτυακές ρυθμίσεις, μηχανισμοί αναζήτησης πρακτόρων και εξασφάλιση

επικοινωνίας).

Από 1/10/2012 έως 26/1/2013

– Κατασκευή υπό συστήματος το οποίο θα εξάγει από τον μετρητικό σταθμό τις μετρήσεις σε αρχείο κειμένου σύμφωνα με τις οδηγίες και ρυθμίσεις της προμηθεύτριας εταιρείας. Ενσωμάτωση στους πράκτορες αισθητήρες δυνατότητας εισαγωγής τιμών από αρχείο κειμένου με δεδομένη γραμμογράφηση.

- Πακέτο εργασίας Π1.2 «Προστασία του συστήματος από μελλοντικές δυσλειτουργίες και χρονικός συντονισμός πρακτόρων»:

Από 27/9/2011 έως 30/4/2012

– Έλεγχος των αποτελεσμάτων των συναρτήσεων υπολογισμού μέλους που κατασκευάστηκαν σε java έτσι ώστε αυτές να παράγουν υπό οποιοδήποτε συνθήκες τα επιθυμητά αποτελέσματα.

– Αρχικός έλεγχος της δομής των πρακτόρων και του όλου συστήματος σχετικά με την ορθή μεταφορά των μηνυμάτων, με πιθανά προβλήματα υπερχειλίσης της ουράς μηνυμάτων του κάθε πράκτορα και την πιθανότητα απώλειας επικοινωνίας μεταξύ πρακτόρων

– Παρακολούθηση με συνεχή χρήση του Java VM Monitor των απαιτήσεων του συστήματος σε επεξεργαστική ισχύ και μνήμη σε κανονικές λειτουργίες του συστήματος.

Από 1/5/2012 έως 30/9/2012

– Χρήση ιστορικών δεδομένων από το Υπουργείο Περιβάλλοντος για χρήση τους από το σύστημα και μελέτη της «σταθερότητας» του συστήματος με χρήση του σε πολύ γρήγορους ρυθμούς (εισαγωγή μετρήσεων και λήψη αποφάσεων σε μεσοδιαστήματα τις τάξης των 7 με 20 δευτερολέπτων).

Από 1/10/2012 έως 26/1/2013

– Επανάληψη και ολοκλήρωση όλων των ελέγχων και τυποποίηση των ρυθμίσεων χρονισμού μετά και την ενσωμάτωση στο σύστημα της δυνατότητας εισαγωγής τιμών από τον μετρητικό σταθμό.

- Πακέτο εργασίας Π1.3 «Προγραμματισμός και τελικό περιβάλλον χρήστη»:

Από 27/9/2011 έως 30/9/2012

– Κατασκευή των αντικειμένων, μεθόδων, ιδιοτήτων, συναρτήσεων και δομής του κάθε πράκτορα.

– Σχεδιασμός και προγραμματισμός του γραφικού περιβάλλοντος χρήστη και των

εναλλακτικών τρόπων αλληλεπίδρασης του συστήματος με τον χρήστη.

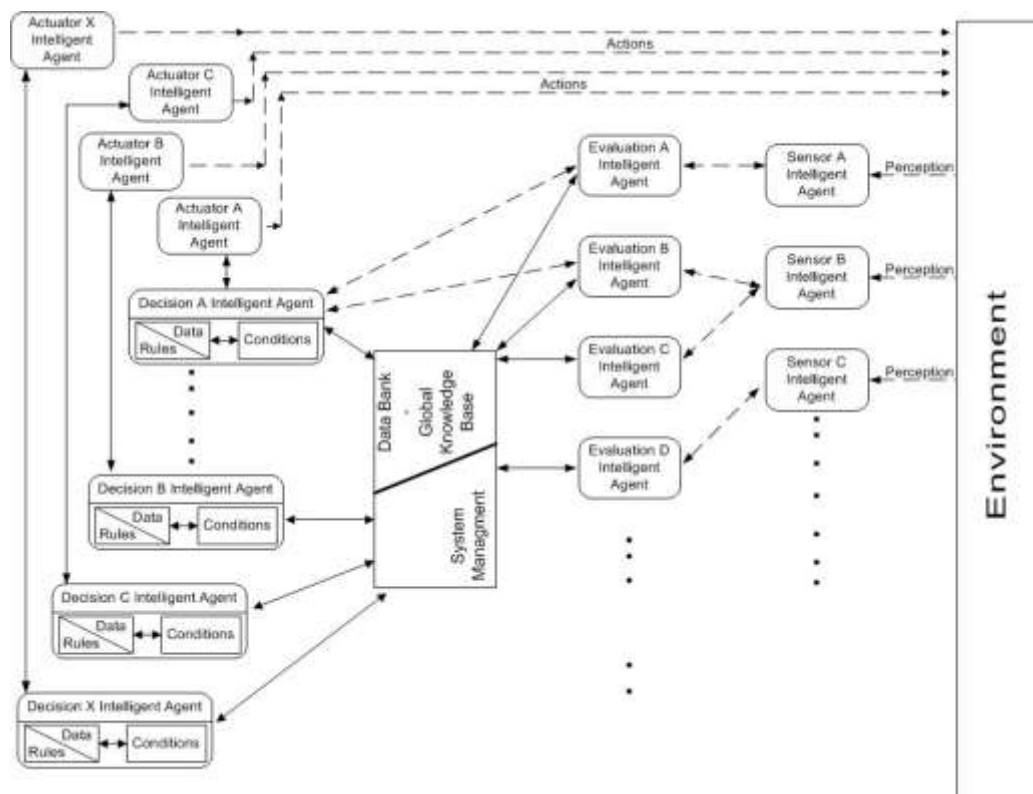
Από 1/10/2012 έως 26/1/2013

- Ολοκλήρωση του προγραμματισμού του συστήματος με ολοκλήρωση του κάθε πράκτορα, του γραφικού περιβάλλοντος και της μεθοδολογίας εισόδου τιμών από τον μετρητικό σταθμό.

4. Δίκτυο κατανεμημένων ευφύων πρακτόρων

4.1 Περιγραφή κατανεμημένου δικτύου ευφύων πρακτόρων

Με στόχο λοιπόν την ικανοποίηση των απαιτήσεων που τέθηκαν από τον φορέα χρηματοδότησης, τις βασικές αρχές αυτονομίας και φορητότητας ενός πολυπρακτορικού συστήματος και λαμβάνοντας υπ' όψιν τα δεδομένα εισόδου, τις περιοχές εγκατάστασης του συστήματος, το διαθέσιμο προϋπολογισμό και τη δυνατότητα μελλοντικής επέκτασης του συστήματος σχεδιάστηκε το παρακάτω κατανεμημένο πολύπρακτορικό σύστημα:



Σχήμα 1: Δομή κατανεμημένου πολυπρακτορικού συστήματος

Το σύστημα αυτονομεί τις λειτουργίες του με σκοπό την καλύτερη και πιο ολοκληρωμένη συνεργασία μεταξύ των τμημάτων που θα εκτελούνται σε διαφορετικά σημεία χωρίς να μειώνεται η λειτουργικότητα του. Δεν υπάρχουν λίγοι και μεγάλοι πράκτορες οι οποίοι θα ενσωματώνουν τους αισθητήρες των παραμέτρων, το μηχανισμό μετασχηματισμού – αντίληψης της μέτρησης, το μηχανισμό λήψης αποφάσεων και τους πράκτορες εκτέλεσης ενεργειών, αλλά όλα αυτά τα συστατικά πλέον διαιρούνται σε μικρότερους αυτόνομους πράκτορες. Τα συστατικά και οι πράκτορες του συστήματος μπορούν να εκτελούνται σε απομακρυσμένους σταθμούς ενός δικτύου, αφού κάθε πράκτορας είναι πλήρως ανεξάρτητος από τον άλλο και η μεταξύ τους επικοινωνία επιτυγχάνεται μέσω της πλατφόρμας υλοποίησης.

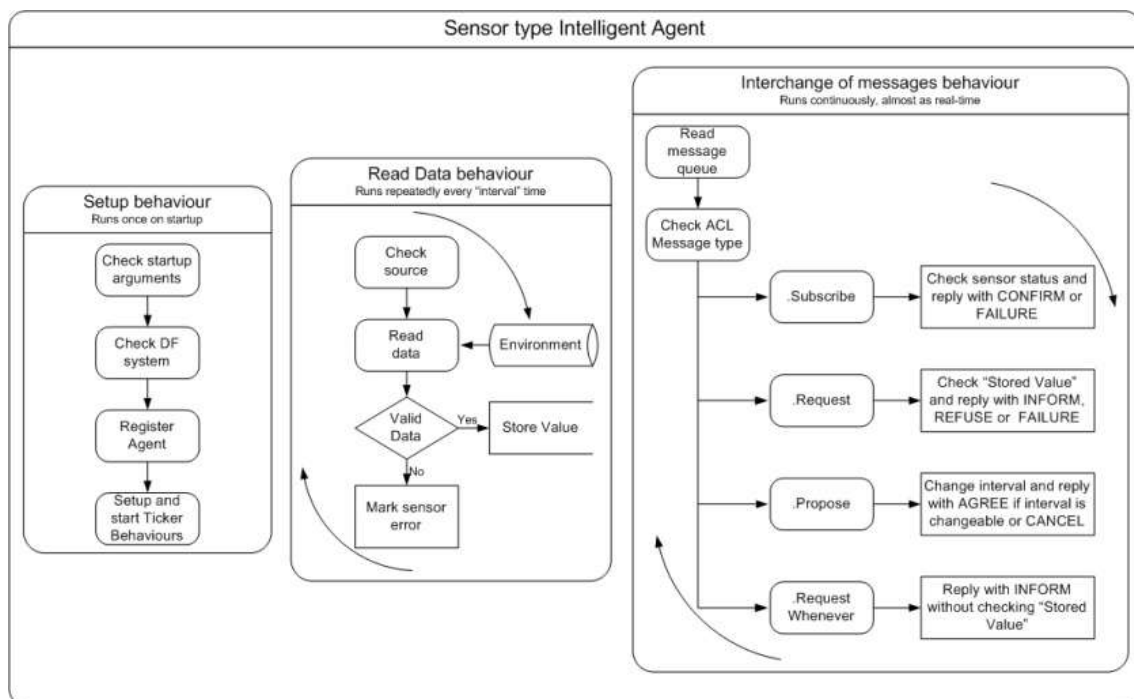
Παρόλη την κατανεμημένη δομή και λειτουργία του συστήματος επιτυγχάνεται κεντρικός εποπτικός έλεγχος λειτουργίας και διαθεσιμότητας δεδομένων με τη χρήση του πράκτορα συστήματος (System agent), ο οποίος εκτός από την κεντρική αποθήκευση όλων των δεδομένων, ώστε αυτά να υπάρχουν διαθέσιμα σε πρώτη ζήτηση, φροντίζει και για τον έλεγχο των συνδεδεμένων πρακτόρων και την ενημέρωση για την κατάσταση του συστήματος.

Το σύστημα μπορεί να λειτουργεί ταυτόχρονα με ασαφή και με αλγεβρική λογική, μιας και κάθε πράκτορας είναι ανεξάρτητος, αντιλαμβάνεται το περιβάλλον του με διαφορετικό τρόπο και μπορεί να απευθυνθεί κάθε φορά στον αντίστοιχο πράκτορα και να ανακτήσει την τιμή την οποία χρειάζεται.

Η αρχιτεκτονική της κάθε κατηγορίας πρακτόρων καθώς και οι λειτουργίες, συμπεριφορές, οι παράμετροι ενεργοποίησης και τα μηνύματα που ανταλλάσσουν μεταξύ τους περιγράφονται στα κεφάλαια που ακολουθούν.

4.2 Πράκτορες αντίληψης περιβάλλοντος

Το πρώτο επίπεδο πρακτόρων του συστήματος είναι οι Πράκτορες Αντίληψης Περιβάλλοντος (ΠΑΠ, Sensor type Agent). Λειτουργία των πρακτόρων αυτών είναι να διαβάζουν τις τιμές των παραμέτρων (οι δώδεκα ατμοσφαιρικοί και μετεωρολογικοί παράγοντες που περιγράφηκαν στο 2^ο κεφάλαιο) και να τις κάνουν διαθέσιμες στο σύστημα, ενώ η δομή τους και τα συστατικά στοιχεία τους φαίνονται στο παρακάτω σχήμα:



Σχήμα 2: Πράκτορας αντίληψης περιβάλλοντος

Όπως φαίνεται και παραπάνω ένας ΠΑΠ αποτελείται από τρεις συμπεριφορές: Η συμπεριφορά εκκίνησης (setup behaviour) εκτελείται μια φορά κατά την αρχική δημιουργία του πράκτορα. Φροντίζει για τον έλεγχο των παραμέτρων ενεργοποίησης του πράκτορα (Startup arguments), τον έλεγχο για την

ύπαρξη του Διεκπεραιωτή Καταλόγου (ΔΚ) και για την μη ύπαρξη δεύτερου ίδιου πράκτορα, για την εγγραφή του στο ΔΚ και τέλος για την ενεργοποίηση των υπολοίπων δύο συμπεριφορών.

Οι παράμετροι ενεργοποίησης του πράκτορα που ελέγχονται από την συμπεριφορά αυτή είναι το αρχείο εισόδου δεδομένων, το χρονικό διάστημα μεταξύ διαδοχικών μετρήσεων (interval time), ο διακόπτης για την ενεργοποίηση του γραφικού περιβάλλοντος του πράκτορα, και τέλος ο διακόπτης για την υποστήριξη αλλαγής του χρονικού διαστήματος μεταξύ διαδοχικών μετρήσεων. Ο τελευταίος διακόπτης υποδηλώνει κατά πόσο ο πράκτορας μπορεί να μεταβάλει τον ρυθμό ανάγνωσης δεδομένων ή όχι. Η δυνατότητα της αλλαγής του interval time του πράκτορα δόθηκε έτσι ώστε αυτός να μπορεί να τροφοδοτεί το σύστημα με μετρήσεις σε μικρότερα χρονικά διαστήματα, όταν η κατάσταση του συστήματος κριθεί κρίσιμη. Όταν ο διακόπτης αυτός είναι αληθής τότε ο πράκτορας θα αποδεχτεί προτάσεις αλλαγής του μεσοδιαστήματος μετρήσεων, ενώ σε διαφορετική περίπτωση θα τις απορρίψει. Ένα απλό παράδειγμα της εντολής εκτέλεσης ενός πράκτορα αισθητήρα σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container temp_sensor:SAgent(c:\temp.txt true 600 false)
```

Εντολή 1: 1^ο Παράδειγμα εκτέλεσης πράκτορα αισθητήρα

Η πιο πάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “temp_sensor”, ο οποίος θα δημιουργήσει και θα εγκατασταθεί σε ένα container με τυχαίο όνομα, θα διαβάζει τιμές θερμοκρασίας από το αρχείο “c:\temp.txt” με ρυθμό μία μέτρηση κάθε 600 δευτερόλεπτα (10 λεπτά) θα έχει γραφικό περιβάλλον και δεν θα έχει την δυνατότητα μεταβολής του χρονικού διαστήματος μεταξύ μετρήσεων. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name aisthitiras_thermokrasias  
temp_sensor:SAgent(c:\temp.txt true 900 true)
```

Εντολή 2: 2^ο Παράδειγμα εκτέλεσης πράκτορα αισθητήρα

Η παραπάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “temp_sensor”, ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “aisthitiras_thermokrasias”, θα διαβάζει τιμές θερμοκρασίας από το αρχείο “c:\temp.txt” με ρυθμό μία μέτρηση κάθε 900 δευτερόλεπτα (15 λεπτά) θα έχει γραφικό περιβάλλον και θα έχει την δυνατότητα μεταβολής του χρονικού διαστήματος μεταξύ μετρήσεων.

Η συμπεριφορά ανάγνωσης δεδομένων (Read Data behaviour) εκτελείται περιοδικά κάθε “interval time” και φροντίζει για την εισαγωγή των τιμών των παραμέτρων. Σε κάθε κύκλο εκτέλεσης ελέγχει για πιθανά σφάλματα ανάγνωσης στο αρχείο εισόδου, διαβάζει την τιμή που αντιστοιχεί στον παράγοντα του περιβάλλοντος που μελετάει ο πράκτορας, ελέγχει την τιμή αυτή και ανάλογα με το αποτέλεσμα του ελέγχου κάνει την τιμή διαθέσιμη στο σύστημα ή μαρκάρει τον πράκτορα ως «μη λειτουργικό».

Η συμπεριφορά ανταλλαγής μηνυμάτων (Interchange of messages behaviour) αναλαμβάνει την επικοινωνία του πράκτορα με το υπόλοιπο σύστημα. Η εκτέλεση της γίνεται περιοδικά με χρόνο επανάληψης μικρότερο από τον χρόνο που χρειάζεται για την πλήρη ολοκλήρωσης της και όχι ως κυκλική συμπεριφορά συνεχούς εκτέλεσης (cyclic behaviour). Αυτό έχει σαν αποτέλεσμα την συνεχόμενη εκτέλεση της συμπεριφοράς σε ψευδό πραγματικό χρόνο με πολύ μικρή επεξεργαστική επιβάρυνση του συστήματος σε αντίθεση με τις ανελαστικές κυκλικές συμπεριφορές (Blocking cyclic

behaviors). Οι ανελαστικές κυκλικές συμπεριφορές ελέγχουν και εκτελούν σε πραγματικό χρόνο όλες τις λειτουργίες τους επιτυγχάνοντας με αυτόν τον τρόπο τον ελάχιστο χρόνο απόκρισης στο συμβάν που ελέγχουν, επιφέροντας όμως την μέγιστη δυνατή επιβάρυνση του συστήματος και την εκμετάλλευση του 100% της διαθέσιμης επεξεργαστικής ισχύος του. Τα χαρακτηριστικά αυτά δημιουργούν προβλήματα όταν επιθυμούμε να εκτελέσουμε περισσότερους από έναν πράκτορα στο ίδιο σύστημα ή όταν το σύστημα έχει χαμηλές επεξεργαστικές δυνατότητες. Στην παρούσα υλοποίηση επιλέχθηκαν απλές κυκλικές συμπεριφορές με χρόνο επανάληψης μικρότερο από τα 200msec. Ο χρόνος αυτός επιτρέπει αφενός την εκτέλεση όλων των εντολών της συμπεριφοράς με περιοδικότητα, η οποία δεν προκαλεί την παραμικρή καθυστέρηση στην απόκριση του συστήματος δίνοντας στην αίσθηση της εκτέλεσης σε πραγματικό χρόνο και αφετέρου επιτρέπει την απροβλημάτιστη εκτέλεση παράλληλων διεργασιών στο σύστημα μιας και τα 200msec είναι πολύ μεγάλος χρόνος ακόμα και για ένα υπολογιστικό σύστημα μειωμένων δυνατοτήτων.

Η συμπεριφορά αυτή ελέγχει την ουρά μηνυμάτων (message queue) του πράκτορα για εισερχόμενα μηνύματα. Για κάθε μήνυμα που καταφθάνει ελέγχει τον τύπο του (σύμφωνα με το FIPA ACL Message type) και ανάλογα με την τύπο και τον αποστολέα εκτελεί τις ανάλογες λειτουργίες. Συνοπτικά τα μηνύματα που γίνονται δεκτά και οι λειτουργίες που εκτελούνται από τους “Sensor Type” ευφυείς πράκτορες είναι τα εξής:

- Μηνύματα τύπου εγγραφής (subscribe) τα οποία έχουν ως σκοπό την ενημέρωση και τον έλεγχο της κατάστασης του συστήματος. Σε κάθε αίτηση εγγραφής (subscribe request) που δέχεται ο πράκτορας, ανταποκρίνεται με επιβεβαίωση (confirm) ή αποτυχία εγγραφής (failure subscribe) ανάλογα με τη λειτουργική ή μη κατάσταση του, όπως αυτή έχει οριστεί από την συμπεριφορά ανάγνωσης δεδομένων (Read Data behaviour). Με τον τρόπο αυτό ενημερώνει τον πράκτορα που έστειλε την αίτηση εγγραφής για το κατά πόσο ο συγκεκριμένος ΠΑΠ (Sensor Type agent) λειτουργεί σωστά ή όχι.
- Μηνύματα τύπου ανάκτησης τιμής (request) τα οποία έχουν ως σκοπό την ενημέρωση του συστήματος με την τελευταία διαθέσιμη μέτρηση του πράκτορα. Η απάντηση που θα στείλει ο ΠΑΠ θα είναι τύπου πληροφόρησης (inform), άρνησης (refuse) ή αποτυχίας (failure) αναλόγως με την κατάσταση του, τον χρόνο που έχει μεσολαβήσει από το προηγούμενο request και την ύπαρξη διαθέσιμης τιμής. Όταν ο πράκτορας διαθέτει νέα τιμή και λειτουργεί χωρίς πρόβλημα απαντάει με μήνυμα τύπου inform ακολουθούμενο από την τιμή αυτή. Όταν ο πράκτορας διαθέτει τιμή η οποία δεν είναι νέα (έχει ξανά αποσταλεί στο σύστημα) αλλά δεν υπάρχει λειτουργικό πρόβλημα στον πράκτορα τότε αποστέλλει μήνυμα τύπου άρνησης. Η περίπτωση αυτή συμβαίνει όταν κάποιος πράκτορας ξαναζητήσει τιμή από τον ΠΑΠ σε διάστημα μικρότερο από το interval time του. Δηλαδή πριν ο πράκτορας ανανεώσει την τιμή του. Τέλος όταν ο πράκτορας δεν λειτουργεί σωστά (πρόβλημα στο αρχείο δεδομένων, πρόβλημα στην επικοινωνία με την πλατφόρμα) τότε απαντάει με μήνυμα αποτυχίας (failure).
- Μηνύματα τύπου πρότασης (propose) τα οποία έχουν ως σκοπό την αλλαγή του “interval time” του πράκτορα. Σε κάθε τέτοιο μήνυμα ο πράκτορας ελέγχει την εσωτερική δυνατότητα αλλαγής του interval, ανάλογα με τις παραμέτρους ενεργοποίησης του πράκτορα (startup arguments), και αποφασίζει για την αλλαγή. Αντίστοιχα απαντάει με αποδοχή (agree) ή απόρριψη (cancel) στον πράκτορα που έστειλε το propose.
- Τέλος μηνύματα τύπου εσπευσμένης ανάκτησης τιμής (request whenever), τα οποία έχουν ως σκοπό την ενημέρωση του συστήματος με την τελευταία διαθέσιμη μέτρηση του πράκτορα

ανεξάρτητα από την λειτουργική του κατάσταση. Η απάντηση θα είναι τύπου inform ακολουθούμενη από την τελευταία μέτρηση χωρίς να γίνει έλεγχος καλής λειτουργίας του πράκτορα.

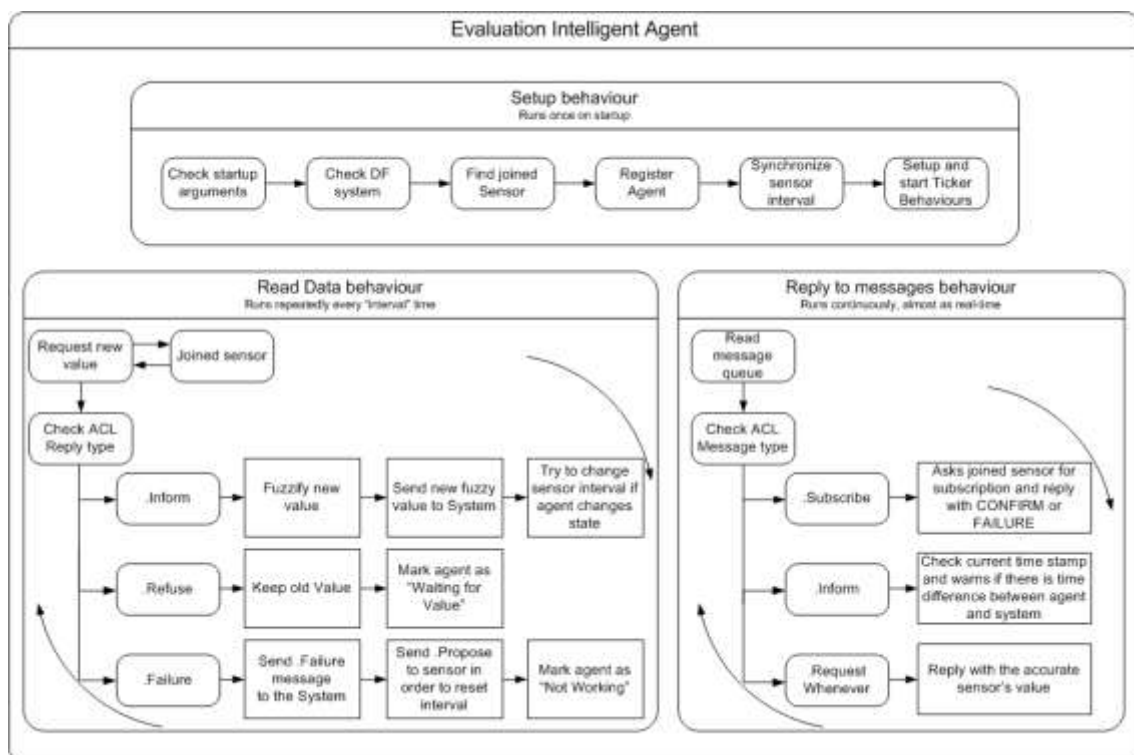
- Μηνύματα εκτός των παραπάνω τύπων αγνοούνται από τον πράκτορα και αφαιρούνται από την ουρά.

Για τις ανάγκες της εφαρμογής, μετά από εντολή του φορέα χρηματοδότησης και με βάση τις προδιαγραφές του σταθμού μέτρησης ατμοσφαιρικών και μετεωρολογικών δεδομένων που θα αποκτηθεί, η συμπεριφορά εισαγωγής – ανάγνωσης στοιχείων διαβάζει τιμές από ένα αρχείο κειμένου στο οποίο θα προβλεφθεί να τοποθετηθούν τα δεδομένα μέτρησης. Οι τιμές θα εγγράφονται στα αρχεία κειμένου από πρόγραμμα που θα το διαθέσει ο προμηθευτής του σταθμού με την αγορά του.

Όλες οι παράμετροι της εφαρμογής ελέγχονται από την ίδια κατηγορία πράκτορα με το όνομα SAgent. Η κατηγορία αυτή (java class SAgent) έχει δυνατότητα διαφοροποίησης από παράμετρο σε παράμετρο με βάση το όνομα του πράκτορα και το αρχείο εισόδου δεδομένων.

4.3 Πράκτορες αξιολόγησης μετρήσεων

Το επόμενο επίπεδο πρακτόρων είναι οι Πράκτορες Αξιολόγησης Μετρήσεων (PIAM, Evaluation Intelligent Agents) οι οποίοι φροντίζουν για την ανάγνωση των μετρήσεων από τους ΠΑΠ (Sensor Type intelligent agents), την επεξεργασία τους με βάση την εσωτερική τους λογική και αντίληψη και την αποστολή στο σύστημα της fuzzy εικόνας του περιβάλλοντος που αυτοί αντιλαμβάνονται. Η δομή του και τα συστατικά στοιχεία ενός PIAM φαίνονται στο επόμενο σχήμα.



Σχήμα 3: Πράκτορας αξιολόγησης μετρήσεων

Κάθε ΠΑΜ είναι λογικά συνδεδεμένος με έναν ΠΑΠ (sensor intelligent agent) από τον οποίο διαβάζει τιμές. Η δομή του συστήματος όμως επιτρέπει την ανάγνωση των μετρήσεων ενός sensor agent από πολλούς Evaluation Intelligent Agents. Είναι δηλαδή δυνατόν ένας ΠΑΠ να συνδέεται με πολλούς διαφορετικούς ΠΑΜ. Αυτό γίνεται για να υπάρχει η δυνατότητα μετασχηματισμού μιας μέτρησης σε διαφορετικές fuzzy τιμές ανάλογα με τις ανάγκες και τη δομή του κάθε προβλήματος που μελετάται.

Ένας ΠΑΜ (Evaluation intelligent agent) αποτελείται από 3 συμπεριφορές.

Η συμπεριφορά εκκίνησης (setup behaviour) εκτελείται μια φορά κατά την αρχική δημιουργία του πράκτορα. Φροντίζει για τον έλεγχο των Startup arguments (συνδεδεμένος Sensor Type intelligent agent, interval time, γραφικό περιβάλλον αλληλεπίδρασης - Gui interface), τον έλεγχο για την ύπαρξη του ΔΚ, για την μη ύπαρξη δεύτερου ίδιου πράκτορα, για την ύπαρξη του πράκτορα από τον οποίο θα «διαβάζει» μετρήσεις στο ΔΚ, για την εγγραφή του στον ΔΚ, τον συγχρονισμό του interval μεταξύ Evaluation Intelligent Agent και του συνδεδεμένου Sensor intelligent agent και τέλος για την ενεργοποίηση των υπολοίπων δύο συμπεριφορών.

Οι παράμετροι ενεργοποίησης του πράκτορα που ελέγχονται από τη συμπεριφορά αυτή είναι το όνομα του συνδεδεμένου πράκτορα, το αρχικό χρονικό διάστημα μεταξύ διαδοχικών μετρήσεων (interval time), και ο διακόπτης για την ενεργοποίηση του γραφικού περιβάλλοντος του. Η δυνατότητα της αλλαγής του interval time είναι ενσωματωμένη στους πράκτορες αυτού του τύπου και ορίζεται ως εξής:

- Όταν ο πράκτορας είναι σε κατάσταση “Normal” ο χρόνος μεταξύ διαδοχικών μετρήσεων ορίζεται στο 100% του αρχικού interval time.
- Όταν ο πράκτορας είναι σε κατάσταση “High” ο χρόνος μεταξύ διαδοχικών μετρήσεων ορίζεται στο 75% του αρχικού interval time.
- Όταν ο πράκτορας είναι σε κατάσταση “Critical” ο χρόνος μεταξύ διαδοχικών μετρήσεων ορίζεται στο 50% του αρχικού interval time.

Όταν υπάρχει αλλαγή στην κατάσταση του πράκτορα, υπολογίζεται ο νέος χρόνος μεταξύ μετρήσεων και αποστέλλεται αντίστοιχο μήνυμα συγχρονισμού στον συνδεδεμένο πράκτορα αισθητήρα. Η αποδοχή του μηνύματος αυτού από τον συνδεδεμένο πράκτορα ή όχι, έχει να κάνει με την λογική και τις ιδιότητες του. Το όνομα του συνδεδεμένου πράκτορα αισθητήρα που δίνεται ως παράμετρος στην εκκίνηση του πράκτορα πρέπει να βρίσκεται ήδη σε λειτουργία και να είναι συνδεδεμένος στην πλατφόρμα αλλιώς ο πράκτορας αξιολόγησης δε θα μπορέσει να ξεκινήσει. Σε περίπτωση δυσλειτουργίας όμως του πράκτορα αισθητήρα, η οποία θα έχει ως αποτέλεσμα την διακοπή λειτουργίας του, ο πράκτορας αξιολόγησης θα συνεχίσει να λειτουργεί αν και όπως γίνεται αντιληπτό δεν θα παρέχει δεδομένα μέχρι την επανέναρξη λειτουργίας του πράκτορα αισθητήρα. Επιπλέον, για να μπορέσει να εκτελεστεί ένας πράκτορας αξιολόγησης πρέπει ήδη να εκτελείται στην πλατφόρμα μας ο πράκτορας διαχείρισης συστήματος που θα εξετάσουμε στην επόμενη παράγραφο. Ένα απλό παράδειγμα της εντολής εκτέλεσης ενός πράκτορα αξιολόγησης σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container temp_eval:tairAgent(temp_sensor 600 true)
```

Εντολή 3: 1^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η εντολή ενεργοποιεί τον πράκτορα με όνομα “temp_eval” ο οποίος θα δημιουργήσει και θα εγκατασταθεί σε ένα container με τυχαίο όνομα, θα διαβάζει τιμές θερμοκρασίας από τον πράκτορα με

όνομα temp_sensor με αρχικό ρυθμό μία μέτρηση κάθε 600 δευτερόλεπτα (10 λεπτά) και θα έχει γραφικό περιβάλλον. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name aksiologisi_thermokrasias  
temp_eval:tairAgent(temp_sensor 900 true)
```

Εντολή 4: 2^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η παραπάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “temp_eval” ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “aksiologisi_thermokrasias”, θα διαβάζει τιμές θερμοκρασίας από τον πράκτορα με όνομα temp_sensor με αρχικό ρυθμό μία μέτρηση κάθε 900 δευτερόλεπτα (15 λεπτά) και θα έχει γραφικό περιβάλλον.

Η κλάση tairAgent αποτελεί την πρότυπη κλάση του πράκτορα αξιολόγησης θερμοκρασίας και μετατρέπει τις μετρήσεις του συνδεδεμένου πράκτορα θερμοκρασίας με βάση τα όρια και τα ασαφή σύνολα που ορίστηκαν στο παραδοτέο της ΕΕ2 από τον φορέα χρηματοδότησης και αναφέρονται και στον Πίνακα 1 του παρόντος.

Η “Read Data behaviour” εκτελείται περιοδικά κάθε “interval time” και φροντίζει για την ανάγνωση της τιμής του συνδεδεμένου πράκτορα, την αξιολόγηση της και την ενημέρωση του συστήματος. Σε κάθε κύκλο εκτέλεσης αποστέλλει ένα μήνυμα τύπου request στον συνδεδεμένο πράκτορα. Ανάλογα με την απάντηση που θα λάβει ενημερώνει το σύστημα και προσπαθεί να τροποποιήσει τον τρόπο λειτουργίας (μεταβολή του interval time αν κριθεί απαραίτητο) του ιδίου καθώς και του συνδεδεμένου πράκτορα.

Σε απάντηση τύπου inform ο πράκτορας κάνει fuzzify την καινούργια τιμή κάνοντας χρήση των ασαφών συνόλων και των συναρτήσεων μέλους χρησιμοποιώντας τα όρια που προαναφέρθηκαν. Ενημερώνει το σύστημα για αντίστοιχο λεκτικό και τον βαθμό μέλους της νέας τιμής και αν υπάρχει αλλαγή στο λεκτικό από την προηγούμενη μέτρηση και λόγος αλλαγής του interval λειτουργίας τότε αλλάζει το interval και στέλνει μήνυμα πρότασης (proposal message) στον συνδεδεμένο πράκτορα, ο οποίος θα λειτουργήσει ανάλογα με τους δικούς του κανόνες.

Σε απάντηση τύπου άρνησης (refuse) το οποίο και σημαίνει ότι δεν άλλαξε η τιμή του συνδεδεμένου πράκτορα γιατί ακόμα δεν ολοκληρώθηκε η αναμονή που ορίζεται από το interval του συνδεδεμένου Sensor agent, ο πράκτορας θα διατηρήσει την παλιά τιμή και θα ενημερώσει το σύστημα ότι είναι σε κατάσταση «αναμονής τιμής».

Τέλος, σε απάντηση τύπου αποτυχίας (failure) ο πράκτορας θα ενημερώσει το σύστημα για τη μη σωστή λειτουργία του συνδεδεμένου πράκτορα, θα του στείλει ένα propose message για να επαναφέρει το interval του στην αρχική του τιμή σε μια προσπάθεια να επανέλθει αυτός σε κανονική λειτουργία και τέλος θα μαρκάρει την κατάσταση του ως «μη λειτουργικός».

Η τελευταία συμπεριφορά, “Reply to messages behaviour”, αναλαμβάνει την διαχείριση και απάντηση μηνυμάτων που λαμβάνει ο πράκτορας από το υπόλοιπο σύστημα. Η εκτέλεση και αυτής γίνεται περιοδικά με interval μικρότερο από τον χρόνο που χρειάζεται για την πλήρη ολοκλήρωση της.

Η συμπεριφορά αυτή ελέγχει την ουρά μηνυμάτων (message queue) του πράκτορα για εισερχόμενα μηνύματα. Για κάθε μήνυμα ελέγχει τον τύπο του (σύμφωνα με το FIPA ACL Message type) και ανάλογα με την τύπο και τον αποστολέα εκτελεί τις ανάλογες λειτουργίες. Συνοπτικά τα μηνύματα που γίνονται δεκτά και οι λειτουργίες που εκτελούνται από έναν “Evaluation Intelligent Agent” είναι τα εξής:

Μηνύματα τύπου subscribe τα οποία έχουν ως σκοπό την ενημέρωση και τον έλεγχο της κατάστασης του συστήματος. Σε κάθε subscribe request που δέχεται ο πράκτορας δημιουργεί και στέλνει ένα αντίστοιχο subscribe message στον συνδεδεμένο sensor agent. Ανάλογα με την απάντηση του συνδεδεμένου πράκτορα και τη λειτουργική κατάσταση του ιδίου, όπως αυτή έχει οριστεί από την “Read Data behaviour”, θα απαντήσει στο σύστημα με confirm ή failure subscribe. Με τον τρόπο αυτό ενημερώνεται το σύστημα για την σωστή λειτουργία του ιδίου αλλά και του συνδεδεμένου “Sensor Type” ευφυούς πράκτορα.

Μηνύματα τύπου inform τα οποία έχουν ως σκοπό τον έλεγχο και τον χρονικό συγχρονισμό μεταξύ των πρακτόρων. Σε κάθε inform μήνυμα ο πράκτορας ελέγχει την διαφορά του τρέχοντος χρόνου του συστήματος με το δικό του και αν αυτή είναι μεγαλύτερη από ένα interval τότε ενημερώνει για πιθανή δυσλειτουργία του συστήματος.

Τέλος, μηνύματα τύπου request whenever τα οποία έχουν ως σκοπό την ενημέρωση των Decision making agents με την πραγματική (crisp) και όχι ασαφή τιμή του συνδεδεμένου πράκτορα. Η υβριδική λειτουργία αυτή επιτρέπει την λήψη αποφάσεων με χρήση της πραγματικής τιμής της κάθε παραμέτρου σε αλγεβρικές πράξεις και μεθόδους, όταν αυτό κριθεί απαραίτητο. Η απάντηση θα είναι τύπου inform χωρίς να γίνει έλεγχος καλής λειτουργίας του πράκτορα μιας και μηνύματα τέτοιου τύπου λαμβάνονται από τους Decision making agents, αφού πρώτα οι τελευταίοι ενημερωθούν από το σύστημα για την λειτουργία των πρακτόρων και την ασαφή τιμή τους.

Μηνύματα εκτός των παραπάνω τύπων αγνοούνται από τον πράκτορα και αφαιρούνται από την ουρά.

Συνολικά οι κλάσεις που κατασκευάστηκαν είναι οι ακόλουθες:

Πίνακας 2: Κλάσεις πρακτόρων αξιολόγησης

Όνομα κλάσης	Παράμετρος αξιολόγησης
tairAgent	Θερμοκρασία περιβάλλοντος
noAgent	Συγκέντρωση μονοξειδίου του αζώτου
no2Agent	Συγκέντρωση διοξειδίου του αζώτου
o3Agent	Συγκέντρωση όζοντος
rhAgent	Σχετική υγρασία

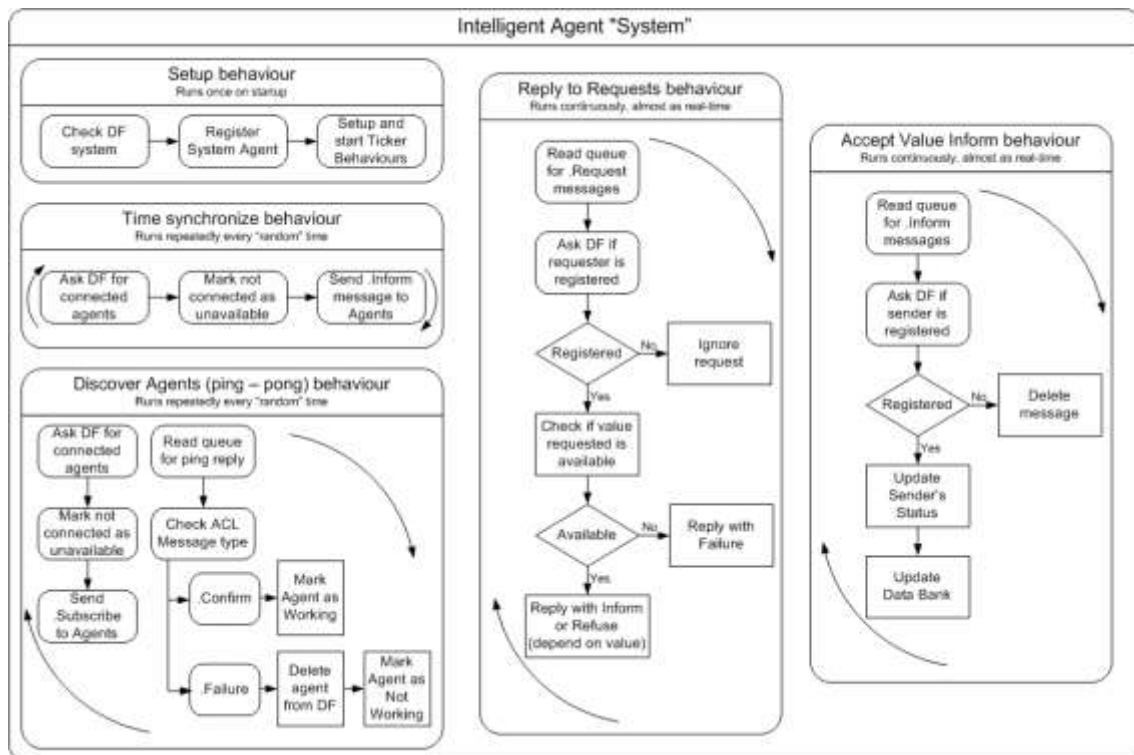
Η κλάση noAgent δεν αποτελεί υποχρέωση βάση του φυσικού αντικείμενου του έργου αλλά κατασκευάστηκε λόγω πιθανής χρήσης της σε σύντομη επέκταση του συστήματος.

4.4 Πράκτορας διαχείρισης συστήματος

Στο επόμενο επίπεδο συναντάται ο Πράκτορας Διαχείρισης Συστήματος (ΠΔΣ - Intelligent Agent System), ο οποίος φροντίζει για το σωστό συγχρονισμό των επιμέρους πρακτόρων, για την αποθήκευση

και διάθεση των μεταβλητών του περιβάλλοντος, όπως αυτές έχουν τροποποιηθεί από τους Evaluation Agents, καθώς και για την συνολική αναπαράσταση της κατάστασης του περιβάλλοντος.

Ο ΠΔΣ αποτελείται από 5 συμπεριφορές.



Σχήμα 4: Πράκτορας διαχείρισης συστήματος

Η “setup behaviour” εκτελείται κατά την εκκίνηση του πράκτορα και ελέγχει την ύπαρξη του ΔΚ, για την μη ύπαρξη δεύτερου ίδιου πράκτορα, για την εγγραφή του στο ΔΚ, και τέλος για την ενεργοποίηση των υπολοίπων συμπεριφορών.

Ο πράκτορας διαχείρισης συστήματος δεν χρησιμοποιεί παραμέτρους ενεργοποίησης κατά την εκτέλεση του. Αυτό γίνεται γιατί όλα τα στοιχεία που είναι απαραίτητα για την λειτουργία του συστήματος και του πράκτορα, ανακτούνται δυναμικά κατά την εκτέλεση του πράκτορα από την υπηρεσία ευρετηρίου της πλατφόρμας. Τα ονόματα των διαφόρων πρακτόρων και η κατάσταση τους ελέγχονται περιοδικά από τον ΠΔΣ (system agent) και δεν χρειάζεται να είναι γνωστά σ’ αυτόν κατά την εκτέλεση του. Ένα παράδειγμα της εντολής εκτέλεσης του πράκτορα διαχείρισης συστήματος σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container system:sysAgent
```

Εντολή 5: Παράδειγμα εκτέλεσης πράκτορα συστήματος

Η πιο πάνω εντολή ενεργοποιεί τον πράκτορα system στο ίδιο υπολογιστικό σύστημα μ’ αυτό που εκτελείται η πλατφόρμα JADE.

Ο πράκτορας διαχείρισης συστήματος πρέπει να εκτελεστεί πριν από την εκτέλεση οποιουδήποτε πράκτορα αξιολόγησης ή πράκτορα λήψης αποφάσεων μιας και αυτοί κατά την εκτέλεση τους θα

προσπαθήσουν να επικοινωνήσουν με τον πράκτορα συστήματος για την ενημέρωση του με τις μετρήσεις του περιβάλλοντος, ή για ανάκτηση τιμών προκειμένου να λάβουν αποφάσεις.

Η “Time synchronize behaviour” εκτελείται περιοδικά ανά τυχαία «σύντομα» χρονικά διαστήματα (παραπλήσια του interval των υπολοίπων πρακτόρων). Σκοπός της συμπεριφοράς αυτής είναι ο έλεγχος του σωστού χρονικού συγχρονισμού του συστήματος.

Το όλο σύστημα βασίζεται στον έλεγχο και αξιολόγηση μετρήσεων, οι οποίες λαμβάνονται σε ορισμένα χρονικά διαστήματα και έχουν αξία όταν η παλαιότητα τους δεν υπερβαίνει το time interval των πρακτόρων. Για τον λόγο αυτό η συγκεκριμένη συμπεριφορά σε κάθε εκτέλεση της ελέγχει το DF για συνδεδεμένους πράκτορες και στέλνει στη λίστα των συνδεδεμένων πρακτόρων μήνυμα τύπου inform με το τρέχοντα χρόνο του συστήματος. Οι πράκτορες όταν λαμβάνουν αυτό το μήνυμα ελέγχουν, όπως αναφέρθηκε στην παραπάνω παράγραφο, το χρόνο του συστήματος με το δικό τους και όταν υπάρχει διαφορά ενημερώνουν ανάλογα.

Η “Discover Agents behaviour” (ping – pong behaviour) εκτελείται επίσης περιοδικά ανά «σύντομα» τυχαία χρονικά διαστήματα και έχει ως σκοπό την ανίχνευση των συνδεδεμένων πρακτόρων, τον έλεγχο επικοινωνίας μ’ αυτούς καθώς και την εκκαθάριση του DF από πράκτορες οι οποίοι λόγω του οποιουδήποτε προβλήματος αποσυνδέθηκαν από το σύστημα. Σε κάθε εκτέλεση της λοιπόν, ελέγχει το DF για συνδεδεμένους πράκτορες, ενημερώνει την λίστα πρακτόρων με αυτούς που αναμένονταν να είναι συνδεδεμένοι αλλά δεν είναι και στέλνει ένα μήνυμα τύπου Subscribe σε όλους αυτούς που δηλώνονται στο DF ως συνδεδεμένοι. Ταυτόχρονα ελέγχει συνέχεια την ουρά μηνυμάτων του πράκτορα για απαντήσεις στα μηνύματα αυτά, οι οποίες είναι τύπου subscription reply messages. Αν η απάντηση σε κάποιο subscribe μήνυμα είναι τύπου confirm τότε έχουμε επιτυχή επικοινωνία με τον πράκτορα και αυτός παραμένει στο DF και στη στην λίστα με τους πράκτορες «εν λειτουργία». Αν η απάντηση είναι τύπου failure τότε ο πράκτορας διαγράφεται από το DF και τη λίστα των «εν λειτουργία» πρακτόρων. Επίσης αν δεν υπάρξει απάντηση σε κάποιο subscribe μήνυμα για περισσότερο από ένα ορισμένο χρόνο (2sec στην παρούσα υλοποίηση) τότε ο πράκτορας θεωρείται ότι έχει χάσει την επικοινωνία του με το σύστημα και εκτελούνται ενέργειες ανάλογες με αυτές της απάντησης τύπου failure reply.

Η συγκεκριμένη συμπεριφορά εκτελείται σε τυχαία χρονικά διαστήματα και η ανίχνευση και η αποστολή του μηνύματος ανίχνευσης του κάθε πράκτορα γίνεται επίσης σε τυχαία χρονικά διαστήματα προσπαθώντας να αποφευχθεί συμφόρηση και μαζική αποστολή και λήψη μηνυμάτων.

Η συμπεριφορά “Reply to requests behaviour” φροντίζει να απαντάει στους πράκτορες που ζητούν τιμές για την κατάσταση του περιβάλλοντος. Η εκτέλεση και αυτής γίνεται περιοδικά με interval μικρότερο από τον χρόνο που χρειάζεται για την πλήρη ολοκλήρωση της, εκμεταλλευόμενη την δυνατότητα της εκτέλεσης σε ψευδό πραγματικό χρόνο όπως αναφέρθηκε στην παράγραφο 4.2. Σε κάθε κύκλο εκτέλεσης, ανακτά μηνύματα τύπου request από την ουρά μηνυμάτων, ελέγχει αν ο αποστολέας είναι εξουσιοδοτημένος – εγγεγραμμένος στο DF της πλατφόρμας και ανάλογα με το αποτέλεσμα του ελέγχου αγνοεί το μήνυμα ή συνεχίζει τον έλεγχο της βάσης με τις αποθηκευμένες τιμές για ύπαρξη πρόσφατης μέτρησης για την παράμετρο που ζητήθηκε.

Αν υπάρχουν πρόσφατες διαθέσιμες μετρήσεις για τον παράγοντα που έγινε το request τότε αποστέλλεται απάντηση τύπου inform ακολουθούμενη με τον βαθμό μέλους και το ασαφές σύνολο στο οποίο ανήκει ο παράγοντας τη στιγμή εκείνη. Αν υπάρχει μέτρηση η οποία έχει ξανά αποσταλεί αλλά ο πράκτορας αξιολόγησης λειτουργεί κανονικά και απλά δεν έχει ενημερώσει ακόμα το σύστημα με νέα

μέτρηση, τότε η απάντηση είναι τύπου refuse. Αν δεν υπάρχει διαθέσιμη μέτρηση τότε αποστέλλεται απάντηση τύπου failure.

Η συμπεριφορά “Accept Value Inform behaviour” φροντίζει να διαβάζει τα μηνύματα που στέλνονται από τους Evaluation Agents και αφορούν την κατάσταση του περιβάλλοντος ενημερώνοντας τη βάση με τις τιμές περιβάλλοντος (DataBank) του συστήματος. Η εκτέλεση και αυτής γίνεται περιοδικά με interval μικρότερο από τον χρόνο που χρειάζεται για την πλήρη ολοκλήρωση της. Σε κάθε κύκλο της ανακτά μηνύματα τύπου Inform από την ουρά μηνυμάτων, ελέγχει αν ο αποστολέας είναι εξουσιοδοτημένος – εγγεγραμμένος στο DF της πλατφόρμας και ανάλογα με το αποτέλεσμα του ελέγχου αγνοεί το μήνυμα ή συνεχίζει την ενημέρωση του databank με την καινούργια τιμή καθώς και της λίστας πρακτόρων με τα στοιχεία της τελευταίας ενημέρωσης.

Όλα τα δεδομένα που καταφθάνουν στον πράκτορα συστήματος, αποθηκεύονται σ’ αυτόν και στην συνέχεια αποστέλλονται από αυτόν στους πράκτορες λήψης αποφάσεων, αφορούν τα ασαφή σύνολα και τους βαθμούς μέλους που παράγουν οι πράκτορες αξιολόγησης του περιβάλλοντος. Δεν αποθηκεύονται πραγματικές μετρήσεις και όταν κάποιος πράκτορας λήψης απόφασης χρειαστεί πραγματικές μετρήσεις απευθύνεται απευθείας στον πράκτορα αξιολόγησης.

4.5 Πράκτορες λήψης αποφάσεων

Οι Πράκτορες Λήψης Αποφάσεων ΠΛΑ (Decision type Intelligent Agents) χαρακτηρίζονται από μεγάλη ανομοιομορφία των παραγόντων από τους οποίους επηρεάζεται η κάθε απόφαση, της σημαντικότητας του καθενός, του αριθμού και των ελέγχων που απαιτεί ο καθένας. Επιπλέον εμφανίζονται πολλαπλοί πράκτορες να εξετάζουν το ίδιο θέμα για διαφορετικές ομάδες πληθυσμού με διαφορετικές βαρύτητες παραγόντων για κάθε μία και διαφορετικές «προσαρμοσμένες» αποφάσεις.

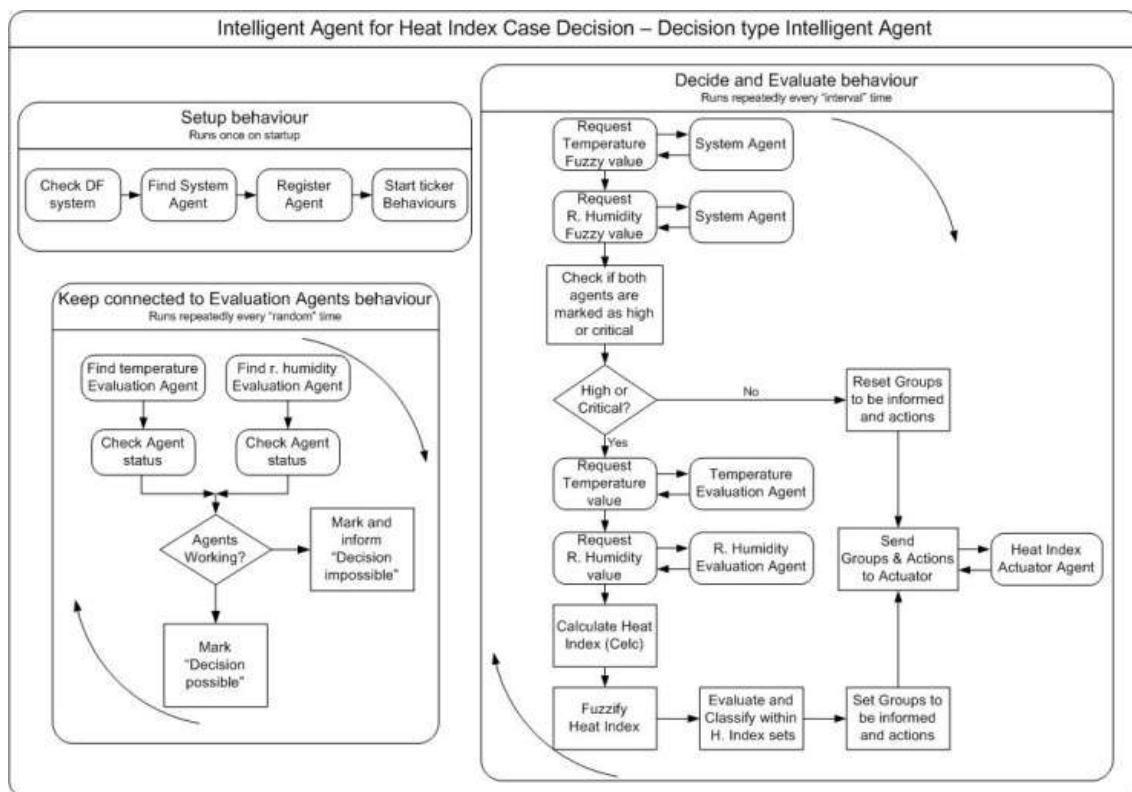
Ο παραπάνω λόγος σε συνδυασμό με την ανάγκη για την δημιουργία ενός ανοικτού συστήματος στο οποίο η διαχειριστική ομάδα θα μπορεί να προσθέτει εξατομικευμένους -custom- πράκτορες λήψης αποφάσεων οδήγησε σε ΠΛΑ (Decision agents) που δεν έχουν κοινή δομή και λειτουργία, όπως όλοι οι προηγούμενοι πράκτορες του συστήματος, αλλά ο καθένας χρησιμοποιεί την πλατφόρμα για να αντλήσει πληροφορίες και δεδομένα και μετά από «εξατομικευμένη» επεξεργασία να παράγει δράσεις και αποτελέσματα. Όλοι οι ΠΛΑ πρέπει να είναι συμβατοί με το FIPA πρότυπο επικοινωνίας και να απευθύνονται στο system agent ή στους Evaluation Agents σύμφωνα με τα μηνύματα που αναφέρθηκαν στις προηγούμενες περιγραφές πρακτόρων.

4.5.1 Πράκτορας ελέγχου δείκτη δυσφορίας

Ο πρώτος πράκτορας λήψης αποφάσεων (ΠΛΑ - Decision type Intelligent Agent) που υλοποιήθηκε στα πλαίσια του συστήματος είναι ο πράκτορας ελέγχου του δείκτη δυσφορίας.

Ο πράκτορας λήψης αποφάσεων Heat Index φροντίζει για τον συνεχή έλεγχο των τιμών της θερμοκρασίας και της σχετικής υγρασίας και όταν αυτές βρίσκονται σε κατάσταση High ή Critical, με οποιονδήποτε βαθμό μέλους, φροντίζει για τον υπολογισμό του δείκτη δυσφορίας, την κατάταξη του σε νέα ασαφή σύνολα με βάση τις οδηγίες και την μοντελοποίηση που έγινε από τον φορέα χρηματοδότησης στην ΕΕ2 και την ενημέρωση μέσω του γραφικού του περιβάλλοντος. Η δομή και οι

λειτουργίες του ΠΛΑ που εξετάζει και προτείνει δράσεις σχετικά με τον Δείκτη Δυσφορίας φαίνονται παρακάτω.



Σχήμα 5: Πράκτορας ελέγχου δείκτη δυσφορίας

Βλέπουμε ότι ο συγκεκριμένος πράκτορας αποτελείται από 3 συμπεριφορές.

Η “setup behaviour” εκτελείται κατά την εκκίνηση του πράκτορα και ελέγχει την ύπαρξη του DF, την εύρεση του system agent, για την εγγραφή του στο DF, και τέλος για την ενεργοποίηση των υπολοίπων συμπεριφορών. Να σημειωθεί ότι ο συγκεκριμένος πράκτορας μπορεί να εκτελείται πολλές φορές σε διαφορετικά σημεία του συστήματος οπότε δεν ελέγχει και δεν αποκλείει διπλότυπες (πολλαπλές) εμφανίσεις του.

Ο πράκτορας ελέγχου του δείκτη δυσφορίας δε χρησιμοποιεί παραμέτρους ενεργοποίησης κατά την εκτέλεση του. Αυτό γίνεται γιατί όλα τα στοιχεία που είναι απαραίτητα για την λειτουργία του, ανακτούνται δυναμικά κατά την εκτέλεση του πράκτορα από την υπηρεσία ευρετηρίου της πλατφόρμας. Ένα απλό παράδειγμα της εντολής εκτέλεσης του πράκτορα ελέγχου του δείκτη δυσφορίας σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container HI_eval:hiAgent
```

Εντολή 6: 1^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η εντολή αυτή ενεργοποιεί τον πράκτορα με όνομα “HI_eval” ο οποίος θα δημιουργήσει και θα εγκατασταθεί σε ένα container με τυχαίο όνομα. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name elegxos_HeatIndex HI_eval:hiAgent
```

Εντολή 7: 2^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η πιο πάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “HI_eval” ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “elegxos_HeatIndex”.

Και στα δύο παραπάνω παραδείγματα ο πράκτορας ενεργοποιείται με ενεργό το γραφικό του περιβάλλον μιας και αυτό δε δύναται να απενεργοποιηθεί αφού η γραφική απεικόνιση της θερμοκρασίας αποτελεί αναπόσπαστο μέρος του πράκτορα.

Η “keep connected to Evaluation Agents behaviour” επιτρέπει στον decision agent να ελέγχει ανά τυχαία χρονικά διαστήματα αν οι evaluation agents οι οποίοι θα χρειαστούν για τη λήψη αποφάσεων είναι συνδεδεμένοι στο σύστημα και λειτουργούν σωστά. Σε κάθε κύκλο εκτέλεσης γίνεται αναζήτηση των Evaluation agents που χρησιμοποιεί ο πράκτορας (θερμοκρασίας και σχετικής υγρασίας) στο DF, ελέγχεται το λειτουργικό status τους και ανάλογα μ’ αυτό ο πράκτορας αποφασίζει κατά πόσο μπορεί να ελέγξει το περιβάλλον ή όχι.

Κύρια συμπεριφορά του πράκτορα είναι η “Decide and Evaluate behaviour”, η οποία εκτελείται περιοδικά κάθε “interval time” και φροντίζει για την αξιολόγηση των μετρήσεων, την εκτίμηση του φαινομένου που εξετάζει και την επιλογή δράσεων.

Κατά την εκτέλεση της, αρχικά διαβάζει από τον system agent την ασαφή τιμή του παράγοντα της θερμοκρασίας και της σχετικής υγρασίας του περιβάλλοντος. Όταν και οι δύο παράγοντες ανήκουν στο ασαφές σύνολο normal τότε ο δείκτης δυσφορίας είναι στην κλίμακα «Comfortable» οπότε ο πράκτορας μηδενίζει τις «ομάδες ενημέρωσης» και τις «προτεινόμενες δράσεις» στέλνοντας το ανάλογο μήνυμα στον συνδεδεμένο actuator agent. Σε περίπτωση που η κατάσταση των δύο παραμέτρων αποτελεί συνέχεια της προηγούμενης (μη αλλαγή κατάστασης του περιβάλλοντος) τότε ο πράκτορας δεν χρειάζεται να προβεί σε καμία ενέργεια.

Αν ο system agent στείλει τιμές μεγαλύτερες από Normal (high ή critical) τότε ο πράκτορας θα στείλει μήνυμα τύπου request whenever στους αντίστοιχους evaluation agents με σκοπό να ανακτήσει την πραγματική τιμή της θερμοκρασίας και σχετικής υγρασίας. Εφαρμόζοντας τις τιμές αυτές υπολογίζει τον δείκτη δυσφορίας (βλέπε Π2.1), τον οποίο στη συνέχεια μετατρέπει σε ασαφές σύνολο με βάση τις κατηγορίες slightly uncomfortable, uncomfortable, Very uncomfortable, intolerable and Death danger. Τέλος, το γραφικό περιβάλλον του πράκτορα εμφανίζει την αντίστοιχη κατηγορία, στην οποία ανήκει ο δείκτης δυσφορίας καθώς και το βαθμό μέλους της.

4.5.2 Πράκτορας ελέγχου συγκέντρωσης Διοξειδίου του Αζώτου

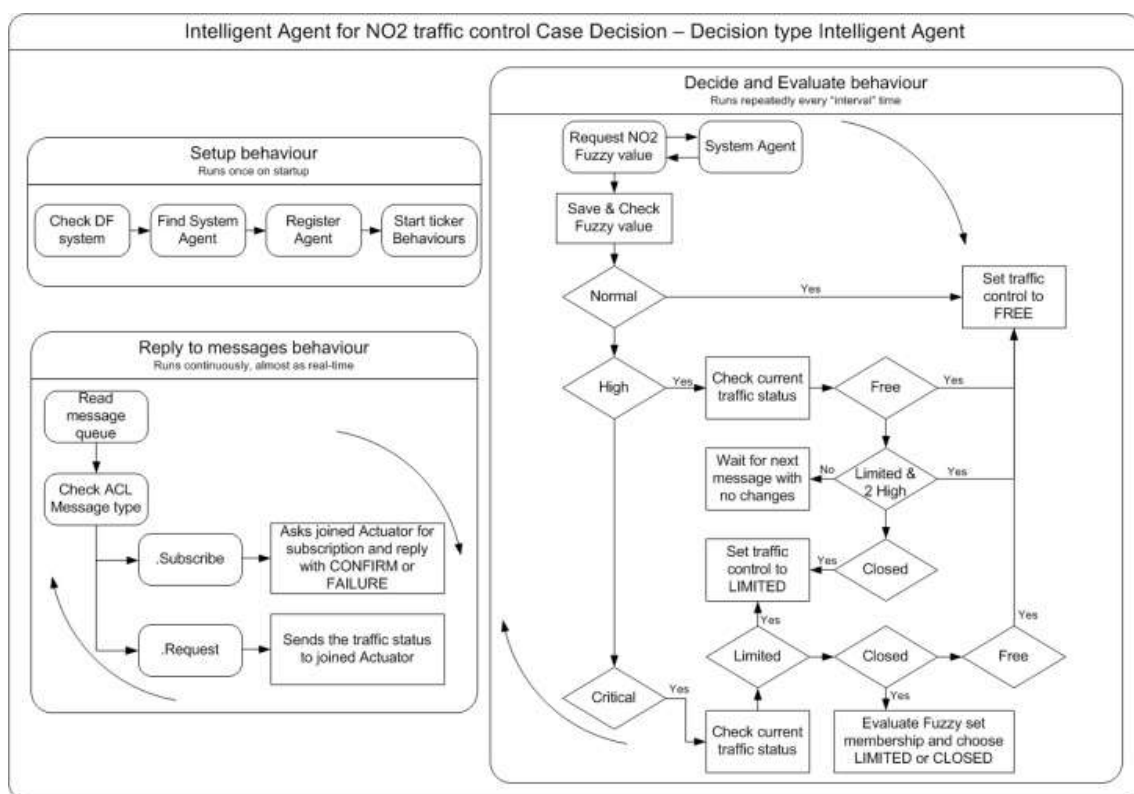
Ο δεύτερος πράκτορας λήψης αποφάσεων, (ΠΛΑ Decision type Intelligent Agent), που υλοποιήθηκε στα πλαίσια του συστήματος είναι ο πράκτορας ελέγχου της συγκέντρωσης Διοξειδίου του Αζώτου στην ατμόσφαιρα και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων στο κέντρο της πόλης της Θεσσαλονίκης.

Ο συγκεκριμένος πράκτορας έχει ως σκοπό τη συνεχή παρακολούθηση των ορίων του διοξειδίου του αζώτου και τον έλεγχο του κυκλοφοριακού δακτυλίου όταν παρουσιάζεται υπέρβαση των ορίων επιφυλακής ή συναγερμού.

Ο πράκτορας ελέγχου της συγκέντρωσης Διοξειδίου του Αζώτου ελέγχει το ασαφές σύνολο στο οποίο ανήκει η συγκέντρωση του συγκεκριμένου ρύπου, όπως αυτή γίνεται αντιληπτή από τον πράκτορα αξιολόγησης μετρήσεων που είναι υπεύθυνος για το διοξείδιο του αζώτου, και ανάλογα με το αντίστοιχο λεκτικό και τον βαθμό μέλους επιλέγει την επιθυμητή κατάσταση του δακτυλίου.

Όταν η συγκέντρωση ανήκει στο ασαφές σύνολο “Normal” τότε αναιρείται οποιοσδήποτε προηγούμενος περιορισμός και ο δακτύλιος ανοίγει για όλα τα οχήματα. Όταν η συγκέντρωση ανήκει για περισσότερες από τρεις μετρήσεις στο ασαφές σύνολο “Critical” τότε ανάλογα με τον βαθμό μέλους επιβάλλεται μερική απαγόρευση της κυκλοφορίας, για ΒΜ μικρότερο του 0,8, ή γενική απαγόρευση για βαθμό μέλους από 0,8 και πάνω. Στην περίπτωση του ασαφούς συνόλου “High” το σύστημα αναμένει για τρεις συνεχόμενες μετρήσεις και αν καμία από αυτές δεν επαναφέρει το σύστημα σε κρίσιμη κατάσταση τότε αναιρεί τους περιορισμούς. Γίνεται βέβαια αντιληπτό ότι αν έστω και μία από τις τρεις αυτές μετρήσεις ανήκει στο ασαφές σύνολο “Normal” τότε αυτόματα αναιρείται κάθε περιορισμός και μηδενίζεται κάθε μετρητής.

Η δομή και οι ακριβείς λειτουργίες του Decision Intelligent Agent που εξετάζει και προτείνει δράσεις σχετικά με τη συγκέντρωση Διοξειδίου του Αζώτου στην ατμόσφαιρα φαίνονται στο παρακάτω σχήμα.



Σχήμα 6: Πράκτορας ελέγχου συγκέντρωσης Διοξειδίου του Αζώτου και επέμβασης στον κυκλοφοριακό δακτύλιο
Βλέπουμε ότι ο συγκεκριμένος πράκτορας αποτελείται από 3 συμπεριφορές.

Η “setup behaviour” εκτελείται κατά την εκκίνηση του πράκτορα και ελέγχει την ύπαρξη του DF, για την μη ύπαρξη δεύτερου ίδιου agent, την εύρεση του system agent, για την εγγραφή του στο DF, και τέλος για την ενεργοποίηση των υπολοίπων συμπεριφορών.

Ο πράκτορας ελέγχου συγκέντρωσης διοξειδίου του αζώτου δεν χρησιμοποιεί παραμέτρους ενεργοποίησης κατά την εκτέλεση του. Αυτό γίνεται γιατί όλα τα στοιχεία που είναι απαραίτητα για την λειτουργία του, ανακτούνται δυναμικά κατά την εκτέλεση του πράκτορα από την υπηρεσία ευρετηρίου της πλατφόρμας. Ένα παράδειγμα εντολής εκτέλεσης του πράκτορα συγκέντρωσης διοξειδίου του αζώτου σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container NO2traffic_eval:NO2Agent
```

Εντολή 8: 1^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η εντολή αυτή ενεργοποιεί τον πράκτορα αξιολόγησης NO₂ με όνομα “NO2traffic_eval” ο οποίος θα δημιουργηθεί και θα εγκατασταθεί σε ένα container (φάκελο) με τυχαίο όνομα. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name elegxos_daktiliou  
NO2traffic_eval:NO2Agent
```

Εντολή 9: 2^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η πιο πάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “NO2traffic_eval” ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “elegxos_daktiliou”.

Και στα δύο παραπάνω παραδείγματα ο πράκτορας ενεργοποιείται με ενεργό το γραφικό του περιβάλλον μιας και αυτό δεν δύναται να απενεργοποιηθεί αφού η γραφική απεικόνιση αποτελεί αναπόσπαστο μέρος του πράκτορα.

Η επόμενη συμπεριφορά, “Reply to messages behaviour”, αναλαμβάνει τη διαχείριση και απάντηση μηνυμάτων που λαμβάνει ο πράκτορας από το υπόλοιπο σύστημα. Η εκτέλεση και αυτής γίνεται περιοδικά με interval μικρότερο από το χρόνο που χρειάζεται για την πλήρη ολοκλήρωσή της.

Η συμπεριφορά αυτή ελέγχει την ουρά μηνυμάτων (message queue) του πράκτορα για εισερχόμενα μηνύματα. Για κάθε μήνυμα ελέγχει τον τύπο του (σύμφωνα με το FIPA ACL Message type) και ανάλογα με την τύπο και τον αποστολέα εκτελεί τις ανάλογες λειτουργίες. Συνοπτικά τα μηνύματα που γίνονται δεκτά και οι λειτουργίες που εκτελούνται από έναν “Evaluation Intelligent Agent” είναι τα εξής:

Μηνύματα τύπου subscribe τα οποία έχουν ως σκοπό την ενημέρωση και τον έλεγχο της κατάστασης του συστήματος. Σε κάθε subscribe request που δέχεται ο πράκτορας δημιουργεί και στέλνει ένα αντίστοιχο subscribe message στον συνδεδεμένο πράκτορα εκτέλεσης εντολής (Actuator Agent). Ανάλογα με την απάντηση του συνδεδεμένου πράκτορα και τη λειτουργική κατάσταση του ίδιου, θα απαντήσει στο σύστημα με confirm ή failure subscribe.

Μηνύματα τύπου request τα οποία έχουν ως σκοπό την ενημέρωση του συνδεδεμένου πράκτορα εκτέλεσης εντολής (Actuator Agent) για την κατάσταση του πράκτορα και την εντολή που αυτός έχει

αποφασίσει. Η απάντηση θα είναι τύπου inform εκτός της περίπτωσης δυσλειτουργίας του συστήματος οπότε και η απάντηση θα είναι τύπου failure.

Μηνύματα εκτός των παραπάνω τύπων αγνοούνται από τον πράκτορα και αφαιρούνται από την ουρά.

Κύρια συμπεριφορά του πράκτορα είναι η “Decide and Evaluate behaviour”, η οποία εκτελείται περιοδικά κάθε “interval time” και φροντίζει για τη λήψη των μετρήσεων του διοξειδίου του αζώτου από τον Πράκτορα Διαχείρισης Συστήματος (ΠΔΣ), την εκτίμηση της συγκέντρωσης και την επιλογή της λειτουργικής κατάστασης του δακτυλίου.

Κατά την εκτέλεση της, αρχικά διαβάζει από τον system agent την ασαφή τιμή του παράγοντα της συγκέντρωσης διοξειδίου του αζώτου. Όταν ο παράγοντας ανήκει στο ασαφές σύνολο normal τότε η λειτουργία του δακτυλίου ορίζεται ως «Ελεύθερη» (FREE) ανεξάρτητα από την προηγούμενη του κατάσταση. Εάν ο παράγοντας ανήκει στο ασαφές σύνολο High τότε γίνεται περαιτέρω έλεγχος στις προηγούμενες μετρήσεις του διοξειδίου του αζώτου. Αν η προηγούμενη μέτρηση του πράκτορα ήταν Normal ή High για περισσότερες από δύο μετρήσεις τότε η λειτουργία του δακτυλίου ορίζεται ως «Ελεύθερη» (FREE). Αν η προηγούμενη μέτρηση ήταν Critical ή λιγότερο από 2 συνεχόμενα High τότε ο πράκτορας δεν αλλάζει την κατάσταση του δακτυλίου και περιμένει την επόμενη μέτρηση ώστε να συμπληρωθεί η τρίωρη βάση για την λήψη απόφασης. Τέλος αν ο παράγοντας ανήκει στο ασαφές σύνολο Critical τότε γίνεται έλεγχος στην τρέχουσα κατάσταση λειτουργίας του δακτυλίου. Αν ο δακτύλιος ήταν σε κατάσταση περιορισμένης κυκλοφορίας τότε παραμένει σε αυτήν. Αν ήταν σε κατάσταση απαγόρευσης κυκλοφορίας τότε ελέγχεται ο βαθμός μέλους της μέτρησης και ο δακτύλιος παραμένει σε κατάσταση απαγόρευσης αν ο BM είναι μεγαλύτερος από 0.8 ή μειώνεται σε κατάσταση περιορισμού αν είναι μικρότερος από 0.8. Αν ο δακτύλιος ήταν σε κατάσταση ελεύθερης κυκλοφορίας τότε παραμένει σε αυτή εκτός της περίπτωσης των τριών συνεχόμενων critical μετρήσεων οπότε και επιλέγεται η κατάσταση περιορισμένης κυκλοφορίας.

4.5.3 Πράκτορας ελέγχου συγκέντρωσης Όζοντος

Ο τρίτος πράκτορας λήψης αποφάσεων που υλοποιήθηκε στα πλαίσια του συστήματος είναι ο πράκτορας ελέγχου της συγκέντρωσης όζοντος στην ατμόσφαιρα και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων στο κέντρο της Θεσσαλονίκης.

Ο συγκεκριμένος πράκτορας έχει ως σκοπό τη συνεχή παρακολούθηση των ορίων του όζοντος και τον έλεγχο του κυκλοφοριακού δακτυλίου όταν παρουσιάζεται υπέρβαση των ορίων επιφυλακής ή συναγερμού. Η δομή του και η λογική του είναι πανομοιότυπη με τον πράκτορα ελέγχου συγκέντρωσης Διοξειδίου του Αζώτου που περιγράφηκε παραπάνω και ως εκ τούτου δεν θα γίνει αναφορά στις συμπεριφορές του πράκτορα.

Ένα παράδειγμα εντολής εκτέλεσης του πράκτορα συγκέντρωσης όζοντος σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container O3traffic_eval:O3Agent
```

Εντολή 10: 1^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η εντολή αυτή ενεργοποιεί τον πράκτορα αξιολόγησης O_3 με όνομα “O3traffic_eval” ο οποίος θα δημιουργηθεί και θα εγκατασταθεί σε ένα container (φάκελο) με τυχαίο όνομα. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name elegxos_daktiliou  
O3traffic_eval:O3Agent
```

Εντολή 11: 2^ο Παράδειγμα εκτέλεσης πράκτορα αξιολόγησης

Η πιο πάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “O3traffic_eval” ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “elegxos_daktiliou”.

Και στα δύο παραπάνω παραδείγματα ο πράκτορας ενεργοποιείται με ενεργό το γραφικό του περιβάλλον μιας και αυτό δεν δύναται να απενεργοποιηθεί αφού η γραφική απεικόνιση αποτελεί αναπόσπαστο μέρος του πράκτορα.

4.6 Πράκτορες Ενεργειών

Τελευταία κατηγορία πρακτόρων είναι οι Πράκτορες Ενεργειών, (ΠΕ - Actuator type Intelligent Agents). Αυτοί καλούνται να υλοποιήσουν τις αποφάσεις που έχουν λάβει οι προηγούμενοι σε σειρά, πράκτορες λήψης αποφάσεων.

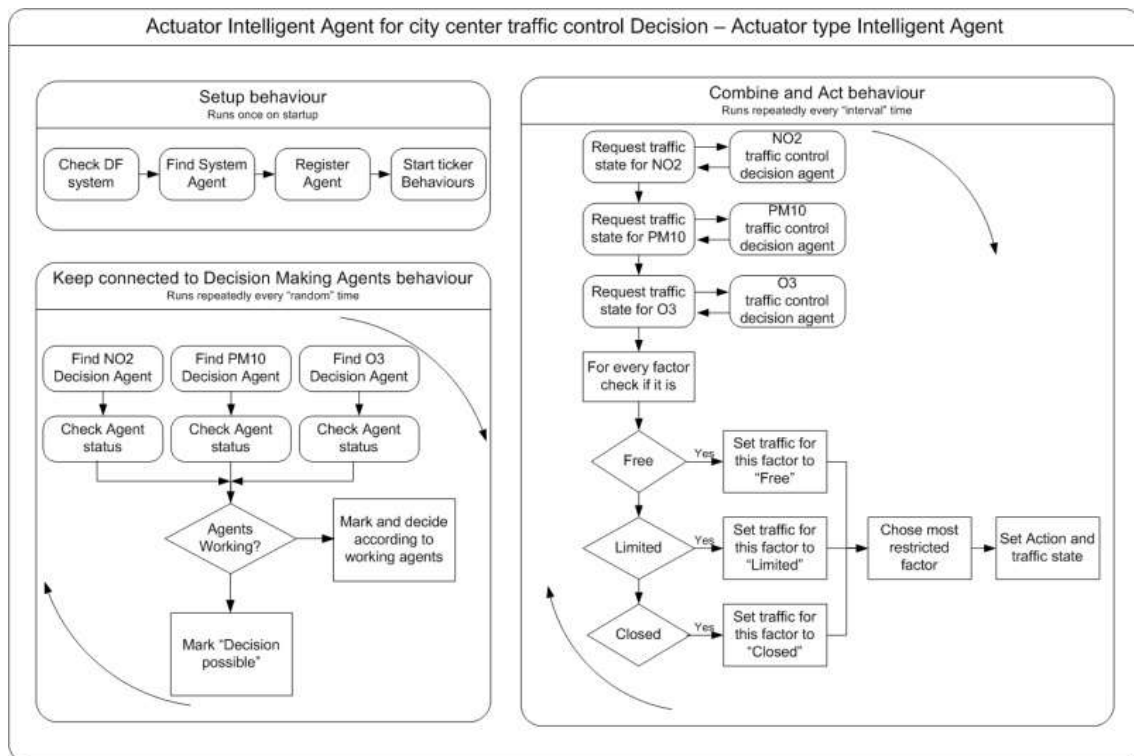
Οι Πράκτορες Ενεργειών, όπως και οι πράκτορες Λήψης Αποφάσεων, δεν έχουν κοινή δομή και λειτουργία, αλλά ο καθένας χρησιμοποιεί την πλατφόρμα για να αντλήσει πληροφορίες και δεδομένα και μετά από «εξατομικευμένους» ελέγχους να αποφασίζουν για την εκτέλεση ή όχι μιας εντολής. Όλοι οι Πράκτορες Ενεργειών πρέπει να είναι συμβατοί με το FIPA πρότυπο επικοινωνίας και να απευθύνονται αποκλειστικά στους Decision type agents που επιθυμούν σύμφωνα με τα μηνύματα που αναφέρθηκαν στις προηγούμενες περιγραφές πρακτόρων.

4.6.1 Πράκτορας υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης

Όπως αναφέρθηκε παραπάνω, σκοπός των decision making agents που ελέγχουν την συγκέντρωση διοξειδίου του αζώτου και του όζοντος είναι να προσδιορίσουν την επιθυμητή κατάσταση του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης, ανεξάρτητα και χωρίς να επηρεάζει ο ένας τον άλλον. Η επιλογή της κατάστασης και η εφαρμογή της θα γίνει από τον αντίστοιχο Actuator. Αυτή είναι λοιπόν η λειτουργία του πράκτορα υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης.

Ο πράκτορας διαβάζει τις αποφάσεις των δύο decision agents που αναφέρθηκαν παραπάνω, ελέγχει αν έχει υπάρξει μεταβολή σε κάποιο παράγοντα, επιλέγει την πιο «αυστηρή» απόφαση αν υπάρχει λόγος περιορισμού της κυκλοφορίας οχημάτων και εφαρμόζει την απόφαση. Στο παρόν στάδιο η εφαρμογή της απόφασης περιορίζεται στην εμφάνιση σχετικού χάρτη με τις περιοχές πρέπει να εφαρμοστούν οι περιορισμοί ενώ η δομή του συστήματος επιτρέπει μέχρι και την “on-line” ενημέρωση των αρμόδιων αρχών τροχαίας, δήμου κ.τ.λ..

Η δομή και οι ακριβείς λειτουργίες του Actuator Intelligent Agent υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου φαίνονται παρακάτω.



Σχήμα 7: Πράκτορας υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου

Βλέπουμε ότι ο συγκεκριμένος πράκτορας αποτελείται από 3 συμπεριφορές.

Η “setup behaviour” εκτελείται κατά την εκκίνηση του πράκτορα και ελέγχει την ύπαρξη του DF, για την μη ύπαρξη δεύτερου ίδιου agent, την εύρεση του system agent, για την εγγραφή του στο DF, και τέλος για την ενεργοποίηση των υπολοίπων συμπεριφορών.

Ο πράκτορας υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου δεν χρησιμοποιεί παραμέτρους ενεργοποίησης κατά την εκτέλεση του. Αυτό γίνεται γιατί όλα τα στοιχεία που είναι απαραίτητα για την λειτουργία του, ανακτούνται δυναμικά κατά την εκτέλεση του πράκτορα από την υπηρεσία ευρετηρίου της πλατφόρμας. Ένα απλό παράδειγμα της εντολής εκτέλεσης του πράκτορα υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου σε γραμμή εντολών είναι το παρακάτω:

```
C:\>java jade.Boot -container traffic_control:trafficAgent
```

Εντολή 12: 1^ο Παράδειγμα εκτέλεσης πράκτορα υλοποίησης αποφάσεων

Η εντολή αυτή ενεργοποιεί τον πράκτορα με όνομα “traffic_control” ο οποίος θα δημιουργήσει και θα εγκατασταθεί σε ένα container με τυχαίο όνομα. Στο παράδειγμα αυτό ο πράκτορας και η πλατφόρμα διαχείρισης (JADE) τρέχουν στο ίδιο υπολογιστικό σύστημα.

```
C:\>java jade.Boot -host 192.168.1.48 -container -container-name elegxos_daktiliou
traffic_control:trafficAgent
```

Εντολή 13: 2^ο Παράδειγμα εκτέλεσης πράκτορα υλοποίησης αποφάσεων

Η παραπάνω εντολή ενεργοποιεί τον πράκτορα με όνομα “traffic_control” ο οποίος θα συνδεθεί στο σύστημα που τρέχει στον υπολογιστή με ip διεύθυνση την 192.168.1.48, θα δημιουργήσει και θα εγκατασταθεί σε ένα container με όνομα “elegxos_daktiliou”.

Και στα δύο παραπάνω παραδείγματα ο πράκτορας ενεργοποιείται με ενεργό το γραφικό του περιβάλλον μιας και αυτό δεν δύναται να απενεργοποιηθεί αφού η γραφική απεικόνιση του χάρτη του δακτυλίου αποτελεί αναπόσπαστο μέρος του πράκτορα.

Η “keep connected to Decision Making Agents behaviour” επιτρέπει στον Actuator να ελέγχει ανά τυχαία χρονικά διαστήματα αν οι Decision Making Agents, οι οποίοι θα χρειαστούν για την επιλογή της απόφασης που θα εκτελεστεί είναι συνδεδεμένοι στο σύστημα και λειτουργούν σωστά. Σε κάθε κύκλο εκτέλεσης γίνεται αναζήτηση των Decision Making Agents που χρησιμοποιεί ο πράκτορας (ελέγχου συγκέντρωσης διοξειδίου του αζώτου και όζοντος) στο DF, ελέγχεται η λειτουργική τους κατάσταση και ανάλογα μ’ αυτή ο πράκτορας αποφασίζει κατά πόσο μπορεί να επιλέξει απόφαση ή όχι.

Κύρια συμπεριφορά του πράκτορα είναι η “Combine and Act behaviour”, η οποία εκτελείται περιοδικά κάθε “interval time” και φροντίζει για την ανάγνωση των επιθυμητών δράσεων από τους πράκτορες, την εκτίμηση του φαινομένου που εξετάζει και την επιλογή δράσεων.

Κατά την εκτέλεση της και εφόσον η “keep connected to Decision Making Agents behaviour” δηλώνει ότι είναι δυνατή η επιλογή κατάστασης για τον δακτύλιο, διαβάζει από τους πράκτορες ελέγχου συγκέντρωσης διοξειδίου του αζώτου και όζοντος τους περιορισμούς στην κίνηση των οχημάτων που αυτοί έχουν αποφασίσει. Ελέγχει για κάθε παράγοντα αν υπάρχει κάποιος περιορισμός και επιλέγει τον υψηλότερο από αυτούς. Τέλος το γραφικό περιβάλλον του πράκτορα εμφανίζει χάρτη με την αντίστοιχη κατηγορία και σήμανση με βάση τον περιορισμό που επιλέχθηκε.

4.7 Διασύνδεση του Συστήματος – Χρήση σε κινητές συσκευές

Όπως αναφέρεται και στην περιγραφή του φυσικού αντικείμενου του έργου, το σύστημα θα πρέπει να έχει την δυνατότητα να πληροφορεί σχετικά με την τρέχουσα κατάσταση και το επίπεδο της ατμοσφαιρικής ρύπανσης στην περιοχή μελέτης εκτός από τους εκάστοτε χρήστες του και το ευρύ κοινό και μέσω ιδικά σχεδιασμένης ιστοσελίδας αλλά και μέσω κινητών – φορητών συσκευών (κινητά τηλέφωνα, tablets, palmtops).

Για τις ανάγκες της εμφάνισης της κατάστασης του συστήματος και στο ευρύ κοινό ο φορέας υλοποίησης σχεδίασε κατάλληλη σχετική ιστοσελίδα, η οποία θα αντλεί δεδομένα από το πολυπρακτορικό σύστημα και θα παρουσιάζει στους ενδιαφερόμενους όλες τις απαραίτητες πληροφορίες.

Η διασύνδεση της ιστοσελίδας με το πολυπρακτορικό σύστημα τυποποιήθηκε ώστε να γίνεται μέσω αρχείων κειμένου, τα οποία ενημερώνονται από τους πράκτορες σε κάθε εκτέλεση τους και στην τελική λειτουργία του συστήματος θα αποθηκεύονται σε κατάλληλο χώρο που θα υποδείξει ο φορέας χρηματοδότησης κατά την φάση της πιλοτικής λειτουργίας του συστήματος. Κατά τη φάση αυτή θα μελετηθεί η σταθερότητα και λειτουργικότητα της συγκεκριμένης επιλογής και σε περίπτωση που αυτή κριθεί ότι δεν καλύπτει τις ανάγκες της εφαρμογής τότε θα κατασκευαστεί ένας απλός πράκτορας, ο οποίος θα συλλέγει όλα τα απαραίτητα δεδομένα σε ένα σημείο και θα τα κάνει διαθέσιμα σε κάποιο

script που θα κατασκευαστεί από το φορέα χρηματοδότησης και θα αναλαμβάνει την ενημέρωση της ιστοσελίδας.

Η ίδια διαδικασία αξιολόγησης θα γίνει και για την χρήση της ιστοσελίδας σε φορητές συσκευές οπότε και εδώ αν η λειτουργικότητα δεν κριθεί ικανοποιητική θα κατασκευαστεί αντίστοιχο πρόγραμμα για κινητές συσκευές, το οποίο θα παρουσιάζει τις ανάλογες πληροφορίες σε ποιο εύχρηστη μορφή.

4.8 Γραφικό περιβάλλον αλληλεπίδρασης – GUI

Όλοι οι πράκτορες που αναπτύχθηκαν για την υλοποίηση του συγκεκριμένου πληροφοριακού συστήματος διαθέτουν σύγχρονο και φιλικό γραφικό περιβάλλον αλληλεπίδρασης και επικοινωνίας με τον χρήστη (GUI) το οποίο έχει σκοπό την παροχή πληροφοριών για την κατάσταση του περιβάλλοντος, την λειτουργική κατάσταση του συστήματος και τις αποφάσεις που λαμβάνονται.

Τα μηνύματα που ανταλλάσσει κάθε πράκτορας, οι μετασχηματισμοί που κάνει στα δεδομένα καθώς και οι αποφάσεις που παίρνει, εμφανίζονται και στο παράθυρο εντολών (command prompt) το οποίο ενεργοποιείται αυτόματα για κάθε πράκτορα. Οι πράκτορες αισθητήρες και οι πράκτορες αξιολόγησης του περιβάλλοντος είναι δυνατόν να ενεργοποιηθούν χωρίς το γραφικό τους περιβάλλον (μέσω της κατάλληλης παραμέτρου στην ενεργοποίηση του πράκτορα) και η επικοινωνία με τον χρήστη να γίνεται μόνο μέσω της γραμμής εντολών. Η δυνατότητα αυτή δεν παρέχεται στον κεντρικό πράκτορα του συστήματος, στους πράκτορες λήψης αποφάσεων και στους πράκτορες υλοποίησης και εφαρμογής αποφάσεων. Η λειτουργία των πρακτόρων αυτών και η λογική τους βασίζονται στον εποπτικό έλεγχο και για τον λόγο αυτό το γραφικό τους περιβάλλον ενεργοποιείται αυτόματα.

Τέλος πρέπει να αναφέρουμε ότι σε όλους τους πράκτορες το πλήκτρο κλεισίματος του παραθύρου είναι ανενεργό για προστασία του πράκτορα από λανθασμένο τερματισμό κάτι που θα προκαλούσε δυσλειτουργία σε όλο το σύστημα.

4.8.1 GUI πράκτορα αισθητήρα

Το γραφικό περιβάλλον αυτού του τύπου των πρακτόρων χωρίζεται σε τέσσερις περιοχές. Στην περιοχή γενικών πληροφοριών, στην περιοχή τελευταίου μηνύματος, στην περιοχή τελευταίας ενέργειας και τέλος στην περιοχή ιστορικού.

Στο παρακάτω σχήμα διακρίνεται αρχικά η πρώτη περιοχή των γενικών πληροφοριών στο πάνω μέρος, η οποία περιλαμβάνει το πλήρες όνομα του πράκτορα (συμπεριλαμβανομένου και του συστήματος στο οποίο είναι συνδεδεμένος), το λειτουργικό του status, την τιμή της τελευταίας μέτρησης, την ένδειξη παλαιότητας της τελευταίας μέτρησης (OLD VALUE, αν η μέτρηση έχει αποσταλεί στο σύστημα, NEW VALUE αν η μέτρηση δεν έχει ακόμα αποσταλεί στο σύστημα), τον ακριβή χρόνο στον οποίο πάρθηκε η τελευταία μέτρηση, τον χρόνο που μεσολαβεί μεταξύ των μετρήσεων (Sensor interval), τη δυνατότητα του πράκτορα να μεταβάλλει τον χρόνο αυτό, τον τρέχοντα χρόνο του συστήματος στο οποίο εκτελείται ο πράκτορας, τον χρόνο στον οποίο ενεργοποιήθηκε ο πράκτορας και τέλος τον αριθμό των εκτελέσεων του πράκτορα για το τρέχον Sensor interval (η ένδειξη αυτή μηδενίζεται κάθε φορά που ο πράκτορας θα αλλάξει ρυθμό επεξεργασίας μετρήσεων με σκοπό την απεικόνιση την «σταθερότητας» των μετρήσεων). Στη συνέχεια βρίσκεται η περιοχή εμφάνισης του τελευταίου μηνύματος. Στην περιοχή αυτή ο

πράκτορας θα εμφανίσει το πλήρες κείμενο της τελευταίας «συνομιλίας» - ανταλλαγής μηνύματος που είχε με κάποιον άλλον πράκτορα.



Σχήμα 8: GUI πράκτορα αντίληψης περιβάλλοντος

Η επόμενη περιοχή εμφανίζει την τελευταία ενέργεια του πράκτορα. Οι κύριες ενέργειες που εκτελούν οι πράκτορες αντίληψης περιβάλλοντος είναι η αλλαγή του χρόνου ανάγνωσης μετρήσεων και η σήμανση των μετρήσεων ως παλιές ή νέες.

Στην τελευταία περιοχή υπάρχουν τα δύο πλήκτρα εμφάνισης του ιστορικού ανταλλαγής μηνυμάτων και ενεργειών του πράκτορα. Κάθε μήνυμα που λαμβάνει ή στέλνει ο πράκτορας όπως και κάθε εντολή την οποία εκτελεί, εκτός από τις περιοχές στις οποίες εμφανίζεται τη στιγμή της εκτέλεσης, αποθηκεύεται και στο ιστορικό (History) του πράκτορα για πιθανή μετέπειτα επεξεργασία.

Τέλος πρέπει να αναφερθεί ότι στους πράκτορες αυτούς (όπως και σε όλους τους άλλους) το πλήκτρο κλεισίματος του παραθύρου είναι ανενεργό για προστασία του πράκτορα από λανθασμένο τερματισμό κάτι που θα προκαλούσε δυσλειτουργία σε όλο το σύστημα.

4.8.2 GUI πράκτορα αξιολόγησης

Το γραφικό περιβάλλον αυτού του τύπου των πρακτόρων χωρίζεται σε πέντε περιοχές. Στις τέσσερις περιοχές που περιγράφηκαν παραπάνω (γενικών πληροφοριών, τελευταίου μηνύματος, τελευταίας ενέργειας και ιστορικού) και στην περιοχή εμφάνισης των ασαφών συνόλων.

Στο σχήμα που ακολουθεί μπορεί να διακριθεί η πρώτη περιοχή των γενικών πληροφοριών με το πλήρες όνομα του πράκτορα (συμπεριλαμβανομένου και του συστήματος στο οποίο είναι συνδεδεμένος), το όνομα του συνδεδεμένου πράκτορα αντίληψης περιβάλλοντος, το λειτουργικό του status, το χρόνο που μεσολαβεί μεταξύ των μετρήσεων (Agent interval), την τιμή της τελευταίας μέτρησης, και τέλος τον τρέχοντα χρόνο του συστήματος, στο οποίο εκτελείται ο πράκτορας.

Η επόμενη περιοχή είναι η περιοχή εμφάνισης των ασαφών συνόλων. Εδώ εμφανίζεται το ασαφές σύνολο στο οποίο ανήκει ο πράκτορας με βάση την μέτρηση που έλαβε καθώς και ο βαθμός μέλους και για τα τρία ασαφή σύνολα "Normal", "High" και "Critical". Η περιοχή αυτή μεταβάλλει ομαλά το χρώμα

της από ανοιχτό κίτρινο μέχρι πολύ σκούρο κόκκινο καθώς αλλάζει το ασαφές σύνολο ή ο βαθμός μέλους της κάθε μέτρησης.



Σχήμα 9: GUI πράκτορα αξιολόγησης μετρήσεων

Στη συνέχεια βρίσκεται η περιοχή εμφάνισης του τελευταίου μηνύματος. Και αυτός ο πράκτορας στην περιοχή αυτή θα εμφανίσει το πλήρες κείμενο της τελευταίας «συνομιλίας» - ανταλλαγής μηνύματος που είχε με κάποιον άλλον πράκτορα.

Η επόμενη περιοχή εμφανίζει την τελευταία ενέργεια του πράκτορα. Οι κύριες ενέργειες που εκτελούν οι πράκτορες αξιολόγησης μετρήσεων είναι η αλλαγή του χρόνου ανάγνωσης μετρήσεων, ο υπολογισμός των ασαφών συνόλων και η διαχείριση των μηνυμάτων καλής λειτουργίας του συστήματος.

Στην τελευταία περιοχή υπάρχουν τα δύο πλήκτρα εμφάνισης του ιστορικού ανταλλαγής μηνυμάτων και ενεργειών, τα οποία έχουν ακριβώς την ίδια λειτουργία με τους πράκτορες αισθητήρες.

4.8.3 GUI πράκτορα διαχείρισης συστήματος

Το γραφικό περιβάλλον αυτού του πράκτορα χωρίζεται σε τρεις περιοχές. Στην περιοχή γενικών πληροφοριών, στην περιοχή πληροφοριών πρακτόρων και στην περιοχή ιστορικού και αναφορών.

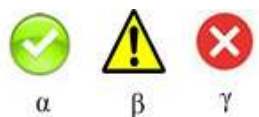
Στο παρακάτω σχήμα διακρίνεται η πρώτη περιοχή των γενικών πληροφοριών με το όνομα του πράκτορα, τον τρέχοντα χρόνο του συστήματος στο οποίο εκτελείται ο πράκτορας και τα μη διαβασμένα μηνύματα που αναμένουν στην ουρά του ΠΔΣ (System agent). Η ένδειξη των μη διαβασμένων μηνυμάτων χρησιμοποιείται ως ένδειξη καλής λειτουργίας του συστήματος.



Σχήμα 10: GUI πράκτορα διαχείρισης συστήματος

Η επόμενη περιοχή εμφανίζει όλους τους πράκτορες αξιολόγησης μετρήσεων του συστήματος με ένδειξη για την κατάσταση λειτουργίας τους και τις μετρήσεις τους. Για κάθε πράκτορα υπάρχουν τρεις πιθανές καταστάσεις λειτουργίας:

- Η κατάσταση κανονικής λειτουργίας η οποία γραφικά αναπαριστάται με πράσινο σήμα σε μορφή «τικ» και σημαίνει ότι ο πράκτορας λειτουργεί κανονικά και υπάρχει διαθέσιμη μέτρηση ή αναμένεται σύντομα.
- Η κατάσταση μερικής λειτουργίας η οποία γραφικά αναπαριστάται με τριγωνικό κίτρινο σήμα με ενσωματωμένο θαυμαστικού και σημαίνει ότι ο πράκτορας είναι εγγεγραμμένος στον ΔΚ, ανταποκρίνεται στην αναζήτηση του ΠΔΣ αλλά δεν αποστέλλει μετρήσεις.
- Η κατάσταση μη λειτουργίας η οποία γραφικά αναπαριστάται με κόκκινο σήμα σε μορφή «X» και σημαίνει ότι ο πράκτορας δεν είναι συνδεδεμένος στο σύστημα.



Σχήμα 11: Γραφική αναπαράσταση λειτουργίας πρακτόρων αξιολόγησης μετρήσεων

Δίπλα στην γραφική απεικόνιση της κατάστασης λειτουργίας του κάθε πράκτορα αξιολόγησης μετρήσεων εμφανίζεται πλαίσιο κειμένου με μηνύματα σχετικά με τον πράκτορα. Σε κατάσταση κανονικής λειτουργίας εμφανίζει το τρέχον ασαφές σύνολο, στο οποίο βρίσκεται η παράμετρος η οποία μελετάται καθώς και τον βαθμό μέλους της ενώ σε κατάσταση μη σωστής λειτουργίας εμφανίζει το πρόβλημα το οποίο ανιχνεύθηκε.

Στην τελευταία περιοχή υπάρχουν το πλήκτρο εμφάνισης του ιστορικού ανταλλαγής μηνυμάτων, το πλήκτρο εμφάνισης ολόκληρου του ιστορικού των ενεργειών ανίχνευσης των συνδεδεμένων πρακτόρων καθώς και το πλήκτρο εμφάνισης της αναλυτικής κατάστασης του περιβάλλοντος.

4.8.4 GUI πράκτορα ελέγχου δείκτη δυσφορίας

Το γραφικό περιβάλλον του πράκτορα ελέγχου του δείκτη δυσφορίας χωρίζεται σε τρεις περιοχές: στην περιοχή γενικών πληροφοριών, στην περιοχή πληροφοριών για την τρέχουσα κατάσταση και στην περιοχή οπτικής αναπαράστασης του δείκτη δυσφορίας.



Σχήμα 12: GUI πράκτορα ελέγχου δείκτη δυσφορίας

Στο παραπάνω σχήμα διακρίνεται η πρώτη περιοχή των γενικών πληροφοριών με το πλήρες όνομα του πράκτορα, τα ονόματα των συνδεδεμένων πρακτόρων (πράκτορες αξιολόγησης μετρήσεων θερμοκρασίας και σχετικής υγρασίας), τον τρέχοντα χρόνο του συστήματος στο οποίο εκτελείται ο πράκτορας και τον χρόνο στον οποίο υπολογίστηκε τελευταία φορά ο δείκτης δυσφορίας. Σε περίπτωση που κάποιος από τους συνδεδεμένους πράκτορες είναι εκτός λειτουργίας αντί για το όνομα του εμφανίζεται ανάλογο μήνυμα.

Στην επόμενη περιοχή εμφανίζονται πληροφορίες για τη τρέχουσα κατάσταση του δείκτη δυσφορίας. Εμφανίζεται η κλάση του δείκτη, η θερμοκρασία που αντιπροσωπεύει την τρέχουσα κλάση και μήνυμα για το κατά πόσο ο τρέχον δείκτης δυσφορία είναι μεγαλύτερη ή μικρότερη από την πραγματική θερμοκρασία. Η περιοχή αυτή αλλάζει αυτόματα το χρώμα του φόντου σε κάθε υπολογισμό του δείκτη έτσι ώστε η οπτική της αναπαράσταση να συμφωνεί με την κλίμακα θερμοκρασίας που εμφανίζεται στην τελευταία περιοχή του πράκτορα.

4.8.5 GUI πράκτορα ελέγχου συγκέντρωσης διοξειδίου του αζώτου και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων

Ο πράκτορας ελέγχου της συγκέντρωσης Διοξειδίου του Αζώτου στην ατμόσφαιρα και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων στο κέντρο της Θεσσαλονίκης, ελέγχει τη συγκέντρωση του διοξειδίου του αζώτου και μετά από υπολογισμούς επιλέγει την ανάλογη λειτουργική κατάσταση του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης.

Το γραφικό του περιβάλλον χωρίζεται σε τρεις περιοχές: στην περιοχή γενικών πληροφοριών, στην περιοχή πληροφοριών για την κατάσταση των τελευταίων τριών ωρών και στην περιοχή οπτικής αναπαράστασης της απόφασης του πράκτορα.



Σχήμα 13: GUI πράκτορα ελέγχου συγκέντρωσης διοξειδίου του αζώτου και επέμβασης στον έλεγχο της κυκλοφορίας

Στο παραπάνω σχήμα διακρίνεται αρχικά η πρώτη περιοχή των γενικών πληροφοριών με το πλήρες όνομα του πράκτορα, τον τρέχοντα χρόνο του συστήματος στο οποίο εκτελείται ο πράκτορας και τον χρόνο στον οποίο ενημερώθηκε τελευταία φορά ο αντίστοιχος πράκτορας υλοποίησης της απόφασης.

Στην επόμενη περιοχή εμφανίζονται τα ασαφή σύνολα με τους αντίστοιχους βαθμούς μέλους των τριών τελευταίων μετρήσεων του πράκτορα αξιολόγησης περιβάλλοντος, ο οποίος ελέγχει το διοξείδιο του αζώτου.

Η τελευταία περιοχή εμφανίζει την απόφαση του πράκτορα με χρήση των λεκτικών “FREE”, “LIMITED” και “CLOSED”. Η περιοχή αυτή αλλάζει αυτόματα το χρώμα του φόντου ανάλογα με την απόφαση του πράκτορα έτσι ώστε να γίνεται άμεση οπτική αναπαράσταση της απόφασης. Σε περίπτωση που ο πράκτορας αξιολόγησης περιβάλλοντος που ελέγχει το διοξείδιο του αζώτου είναι εκτός λειτουργίας ή δεν λειτουργεί σωστά ο πράκτορας διαχείρισης συστήματος αντί για την απόφαση εμφανίζεται ανάλογο μήνυμα που ενημερώνει για την αδυναμία λήψης απόφασης.

4.8.6 GUI πράκτορα ελέγχου συγκέντρωσης όζοντος και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων

Ο πράκτορας ελέγχου της συγκέντρωσης όζοντος στην ατμόσφαιρα και επέμβασης στον έλεγχο της κυκλοφορίας των οχημάτων, ελέγχει τη συγκέντρωση του όζοντος και μετά από υπολογισμούς επιλέγει την ανάλογη λειτουργική κατάσταση του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης. Το γραφικό του περιβάλλον είναι πανομοιότυπο με του πράκτορα ελέγχου συγκέντρωσης διοξειδίου του αζώτου που περιγράφηκε παραπάνω και εκτελεί ακριβώς τις ίδιες λειτουργίες.

4.8.7 GUI πράκτορα υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου

Ο πράκτορας υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου ελέγχει τους πράκτορες λήψης αποφάσεων που εμπλέκονται με την κατάσταση λειτουργίας του δακτυλίου (συγκέντρωσης διοξειδίου του αζώτου και όζοντος) και επιλέγει την απόφαση που πρέπει να εφαρμόσει.

Το γραφικό του περιβάλλον χωρίζεται σε τρεις περιοχές: στην περιοχή γενικών πληροφοριών, στην περιοχή εμφάνισης πληροφοριών για την κατάσταση λειτουργίας και αποφάσεων των δύο συνδεδεμένων πρακτόρων λήψης αποφάσεων και στην περιοχή οπτικής αναπαράστασης της απόφασης του πράκτορα. Στην τελευταία περιοχή εκτός από την λεκτική οπτική αναπαράσταση υπάρχει και η επιλογή της εμφάνισης του χάρτη του κυκλοφοριακού δακτυλίου σε μορφή ανάλογη της απόφασης.

Το γραφικό περιβάλλον ενός τέτοιου πράκτορα φαίνεται στο ακόλουθο σχήμα:



Σχήμα 14: GUI πράκτορα υλοποίησης αποφάσεων ελέγχου του κυκλοφοριακού δακτυλίου της Θεσσαλονίκης
Στο παραπάνω σχήμα διακρίνεται αρχικά η πρώτη περιοχή των γενικών πληροφοριών με το πλήρες όνομα του πράκτορα, τον τρέχοντα χρόνο του συστήματος στο οποίο εκτελείται ο πράκτορας και τέλος τον χρόνο στον οποίο ο πράκτορας υπολόγισε και εφάρμοσε κάποια απόφαση.

Στην επόμενη περιοχή απεικονίζονται οι καταστάσεις και οι επιμέρους αποφάσεις των δύο συνδεδεμένων πρακτόρων λήψης αποφάσεων. Για κάθε έναν από αυτούς απεικονίζεται το όνομα με το οποίο είναι εγγεγραμμένος στο DF, η τρέχουσα απόφαση του και ο χρόνος στον οποίο έγινε η τελευταία ενημέρωση του τρέχοντα πράκτορα υλοποίησης αποφάσεων. Σε περίπτωση που ένας πράκτορας λήψης αποφάσεων δεν είναι συνδεδεμένος το σύστημα τότε εμφανίζεται αντίστοιχο μήνυμα και δεν υπολογίζεται στην λήψη της απόφασης. Επίσης, σε περίπτωση που ένας ΠΛΑ είναι εγγεγραμμένος στο DF αλλά για τον οποιονδήποτε λόγο δεν απαντάει στα μηνύματα του πράκτορα υλοποίησης αποφάσεων πάλι εμφανίζεται αντίστοιχο μήνυμα και δεν υπολογίζεται στην λήψη της απόφασης.

Η τελευταία περιοχή εμφανίζει την απόφαση του πράκτορα με χρήση των λεκτικών "FREE", "LIMITED" και "CLOSED". Η περιοχή αυτή αλλάζει αυτόματα το χρώμα του φόντου ανάλογα με την απόφαση του πράκτορα έτσι ώστε να γίνεται άμεση οπτική αναπαράσταση της απόφασης.

Το πλήκτρο εμφάνισης χάρτη του δακτυλίου ανάλογα με την λειτουργική του κατάσταση, εμφανίζει επιπλέον παράθυρο, στο οποίο απεικονίζεται χάρτης με τα όρια του δακτυλίου σε χρώματα ανάλογα με την τρέχουσα απόφαση.

Πρέπει να αναφερθεί ότι σε περίπτωση υλοποίησης απόφασης τύπου «LIMITED» ή «CLOSED» ο ανάλογος χάρτης θα εμφανιστεί αυτόματα και ο χρήστης δεν έχει την δυνατότητα τερματισμού του σχετικού παραθύρου. Απόκρυψη του χάρτη επιτρέπεται μόνο στην κατάσταση «FREE» οπότε στη συνέχεια αν χρειάζεται, μπορεί με χρήση του πλήκτρου που αναφέρεται να επανεμφανιστεί ο χάρτης.

5.Ολοκλήρωση του συστήματος – Λειτουργία

Με την ολοκλήρωση της ΕΕ1 ολοκληρώθηκε και η ανάπτυξη του πολυπρακτορικού συστήματος, όπως αυτό προδιαγράφηκε από τις απαιτήσεις του φορέα χρηματοδότησης.

Το σύστημα σχεδιάστηκε και αναπτύχθηκε εξ αρχής σε δικτυακό περιβάλλον (η ανάπτυξη του έγινε στο Εργαστήριο Δασικής Πληροφορικής του τμήματος Δασολογίας και Διαχείρισης Περιβάλλοντος και Φυσικών Πόρων του ΔΠΘ) και κατά την διάρκεια της ανάπτυξης έγιναν επανειλημμένοι έλεγχοι για να επιβεβαιωθεί η ορθή δικτυακή του λειτουργία. Κατά την χρήση του όμως σε πραγματικές συνθήκες θα υπάρξουν διαφορετικά δικτυακά δεδομένα (Dsl γραμμή και όχι πανεπιστημιακό δίκτυο με μισθωμένα κυκλώματα κ.τ.λ.), οπότε κατά την λειτουργία του συστήματος μέχρι και την λήξη του έργου, μπορεί να κριθεί σκόπιμο να υπάρξει μικρή τροποποίηση σε τμήματα του συστήματος ώστε αυτό να παρουσιάσει την επιθυμητή σταθερότητα και λειτουργικότητα.

Στις ΕΕ που ακολουθούν, δηλαδή ο «ΕΛΕΓΧΟΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΜΕ ΠΑΛΑΙΑ ΔΕΔΟΜΕΝΑ (ΕΕ3)» και η «ΠΙΛΟΤΙΚΗ ΕΦΑΡΜΟΓΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ ΣΕ ΑΣΤΙΚΟ ΚΕΝΤΡΟ (ΕΕ4)» θα πραγματοποιηθεί από τον φορέα χρηματοδότησης η εγκατάσταση του εξοπλισμού, η εγκατάσταση του δικτύου, η ρύθμιση του σταθμού μέτρησης και όλες οι απαραίτητες ενέργειες για την ολοκληρωμένη χρήση και έλεγχο του συστήματος. Κατά το διάστημα αυτό θα γίνουν αν κριθεί απαραίτητο και οι πιθανές, μικρής κλίμακας, τροποποιήσεις των τμημάτων του συστήματος που προαναφέρθηκαν.