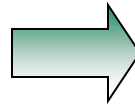


# Κινητοί Πράκτορες

## Κυριότερα παραδείγματα κινητών υπολογισμών

### Απομακρυσμένη εκτίμηση (Remote Evaluation):

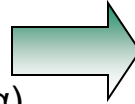
Το αντικείμενο A έχει τη γνώση για την εκτέλεση της εργασίας, αλλά δεν έχει πρόσβαση στους απαιτούμενους πόρους



Το αντικείμενο A στέλνει εντολές σε ένα απομακρυσμένο αντικείμενο B, το οποίο έχει πρόσβαση στους πόρους, το B εκτελεί την εργασία και επιστρέφει το αποτέλεσμα στο A

### Κώδικας μετά από ζήτηση (Code on Demand):

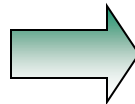
Το αντικείμενο A έχει πρόσβαση στους απαιτούμενους πόρους, αλλά όχι τη γνώση (κώδικα) για την εκτέλεση της εργασίας.



Ο κώδικας μεταφέρεται στη φυσική θέση του A, αποκτά πρόσβαση στους πόρους, η εργασία εκτελείται και το αποτέλεσμα επιστρέφεται στο A

### Κινητοί Πράκτορες (Mobile Agents):

Το αντικείμενο A έχει τη γνώση για την εκτέλεση της εργασίας, αλλά δεν έχει πρόσβαση στους απαιτούμενους πόρους



Το αντικείμενο A μεταναστεύει στη φυσική θέση των πόρων ως μία ολόκληρη υπολογιστική οντότητα (κώδικας και κατάσταση), αποκτά πρόσβαση στους πόρους και εκτελείται η εργασία

# Κινητοί Πράκτορες

## Κινητοί πράκτορες (mobile agents)

Ένας πράκτορας είναι μία οντότητα λογισμικού η οποία βοηθά το χρήστη, δρώντας εκ μέρους του. Η λειτουργία του είναι να επιτρέπει στους χρήστες να αναθέτουν τις εργασίες, τις οποίες θα έκαναν οι ίδιοι, στο ίδιο το λογισμικό.

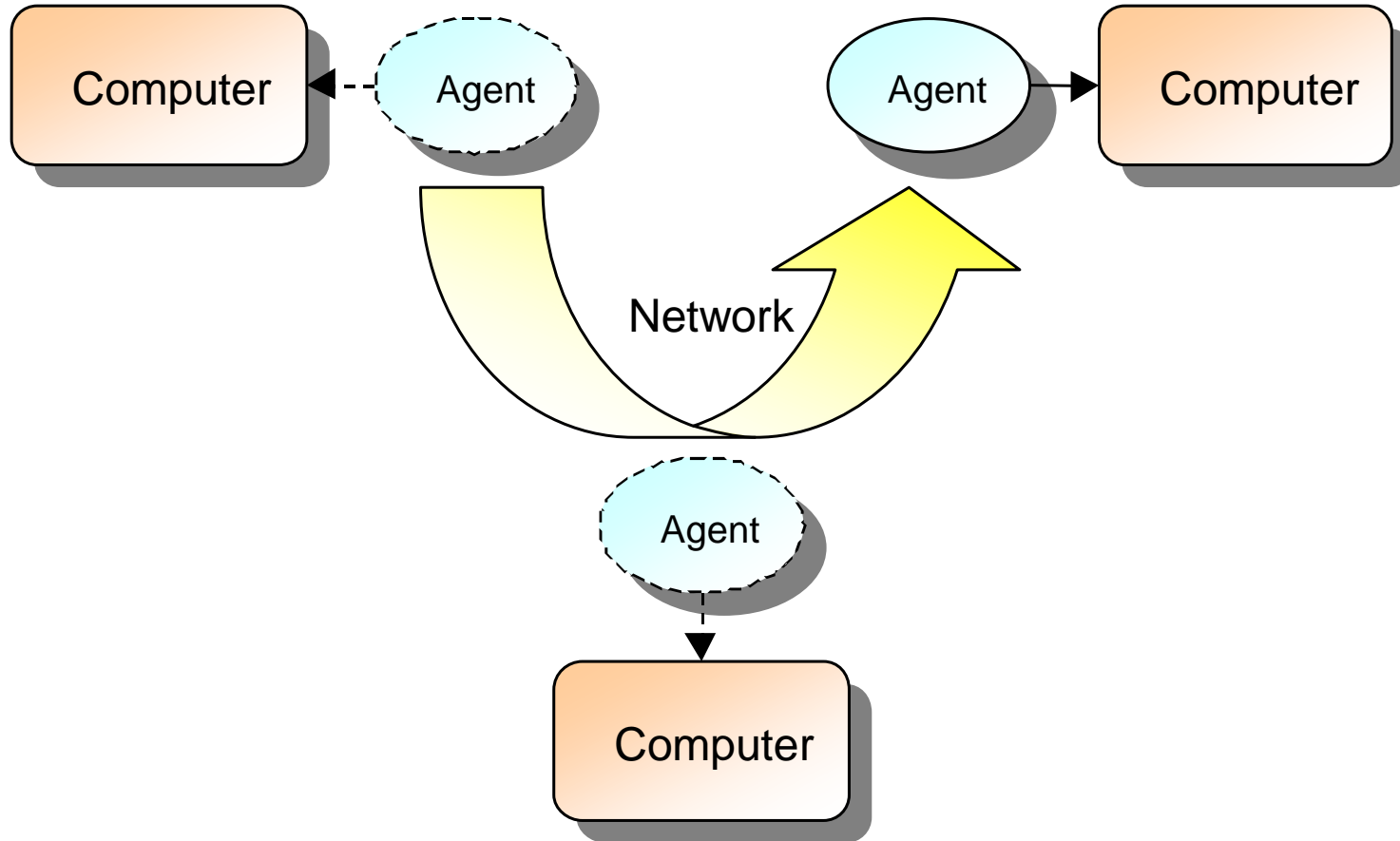
Πιο αυστηρά, ένας πράκτορας είναι ένα αντικείμενο με προσωπικό νήμα εκτέλεσης (thread), με άλλα λόγια ένα ενεργό αντικείμενο το οποίο έχει τη μοναδική ικανότητα να μεταφέρεται αυτόνομα από μία δικτυακή τοποθεσία σε μία άλλη.

### Χαρακτηριστικά

- ✓ Αυτόνομος (autonomous)
- ✓ Οδηγούμενος απ' το στόχο (goal-driven)
- ✓ Ικανότητα αντίδρασης (reactive)
- ✓ Κοινωνικός (social)
- ✓ Προσαρμοστικός (adaptive)
- ✓ Αξιόπιστος (reliable)
- ✓ Κινητός (mobile)

# Κινητοί Πράκτορες

Κινητοί πράκτορες (mobile agents)



# Κινητοί Πράκτορες

## Χαρακτηριστικά κινητών πρακτόρων

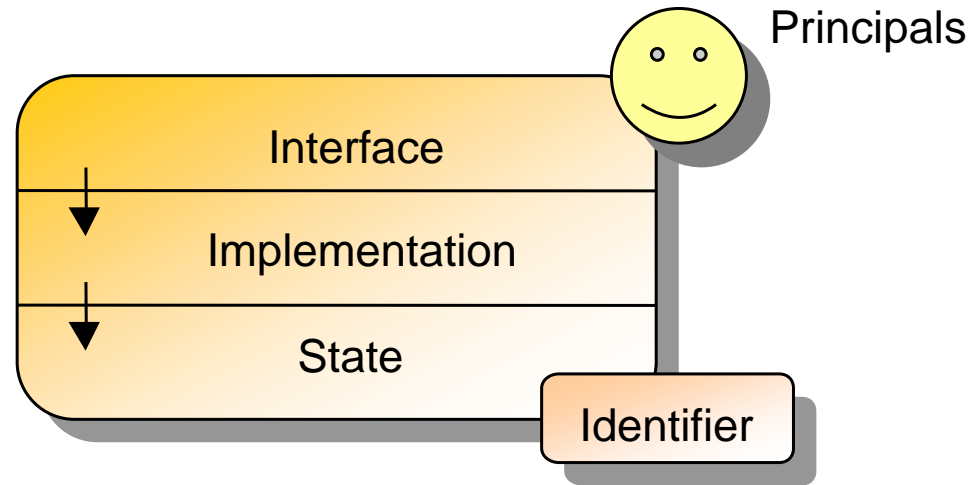
**Κατάσταση (state):** απαραίτητη για την συνέχιση των υπολογισμών του αντιπρόσωπου έπειτα από τη μετανάστευση

**Υλοποίηση (implementation):** απαραίτητη για την εκτέλεση του πράκτορα ανεξάρτητα από την τοποθεσία (π.χ if state=1 then)

**Διεπαφή (interface):** απαραίτητη για την επικοινωνία μεταξύ των πρακτόρων (π.χ Class myAgent implements IAgentServer)

**Ταυτότητα (Identifier):** απαραίτητη για την αναγνώριση, διαχωρισμό και εντοπισμό των μετακινούμενων πρακτόρων

**Αρχές (Principals):** απαραίτητη για τον προσδιορισμό αρμοδιοτήτων από την πλευρά των πρακτόρων

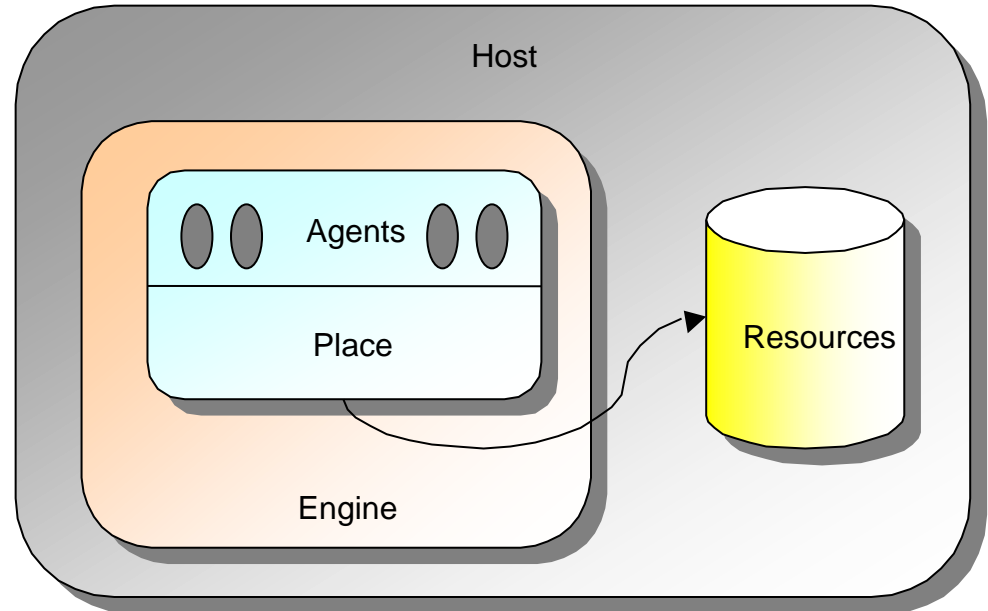


# Κινητοί Πράκτορες

Το περιβάλλον εκτέλεσης των κινητών πρακτόρων

**Engine:** Η εικονική μηχανή που υποστηρίζει μία ή περισσότερες τοποθεσίες (places)

**Resources:** Βάσεις δεδομένων, επεξεργαστές, καθώς και άλλες υπηρεσίες που μπορεί να παρέχονται από το σύστημα

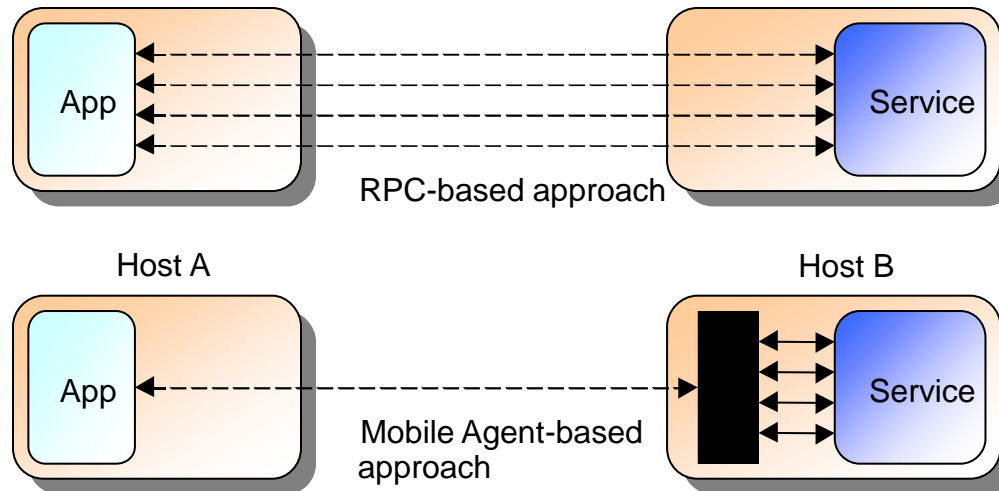


**Place:** Η δικτυακή διεύθυνση μίας δεδομένης τοποθεσίας

# Κινητοί Πράκτορες

Τα πλεονεκτήματα από τη χρήση των κινητών πρακτόρων<sup>1</sup>

## 1) Μειώνουν το φορτίο του δικτύου



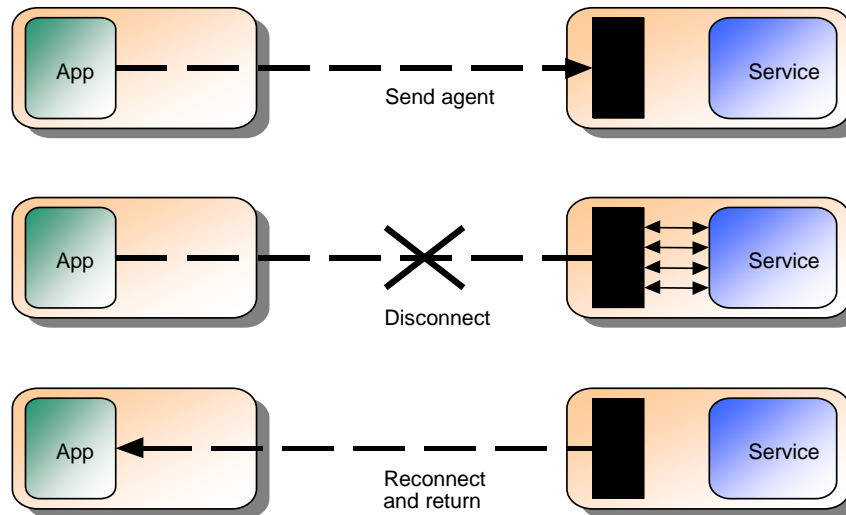
## 2) Αντιμετωπίζουν τη δικτυακή καθυστέρηση

- Οι κινητοί πράκτορες προσφέρουν λύση, αφού αποστέλλονται από ένα κεντρικό ελεγκτή και δρουν τοπικά εκτελώντας τις διαταγές του ελεγκτή.

# Κινητοί Πράκτορες

Τα πλεονεκτήματα από τη χρήση των κινητών πρακτόρων<sup>2</sup>

## 3) Εκτελούνται ασύγχρονα και αυτόνομα



## 4) Προσαρμόζονται δυναμικά

Οι κινητοί πράκτορες έχουν την ικανότητα να αντιλαμβάνονται το περιβάλλον εκτέλεσής τους και να αντιδρούν αυτόνομα στις αλλαγές.

Πολλαπλοί πράκτορες έχουν τη μοναδική ικανότητα να κατανέμουν τους εαυτούς τους μεταξύ των διαθέσιμων υπολογιστών του δικτύου προκειμένου να πετύχουν το βέλτιστο τρόπο για την επίλυση ενός προβλήματος.

# Κινητοί Πράκτορες

Τα πλεονεκτήματα από τη χρήση των κινητών πρακτόρων<sup>2</sup>

## 5) Είναι ετερογενείς από τη φύση τους

Η δικτυακή επεξεργασία είναι από τη φύση της ετερογενής, συχνά τόσο από την πλευρά του υλικού όσο και λογισμικού.

Καθώς οι κινητοί πράκτορες είναι ανεξάρτητοι από το επίπεδο συστήματος και το επίπεδο μεταφοράς (transport layer) και εξαρτώνται μόνο από το περιβάλλον εκτέλεσής τους, παρέχουν τις κατάλληλες συνθήκες για την ολοκλήρωση των ετερογενών μερών του συστήματος.

## 6) Παρουσιάζουν ευρωστία και αντοχή σε σφάλματα

Η ικανότητα των κινητών πρακτόρων να αντιδρούν σε ανεπιθύμητες καταστάσεις και γεγονότα καθιστά πιο εύκολη τη δημιουργία εύρωστων συστημάτων.

Εάν ένας υπολογιστής σταματήσει να λειτουργεί, οι πράκτορες που τρέχουν σε αυτόν είναι δυνατό να ενημερωθούν έγκαιρα, να μεταναστεύσουν και να συνεχίσουν την εργασία τους από το σημείο που είχε σταματήσει.



# Κινητοί Πράκτορες

Εφαρμογές που επωφελούνται από τη χρήση των κινητών πρακτόρων:

- Εφαρμογές ελέγχου ροής (Workflow applications)
- Ηλεκτρονικό εμπόριο (E-Commerce)
- Κατανεμημένη ανάκτηση πληροφοριών
- Διάδοση πληροφοριών
- Απομακρυσμένη εκτέλεση και παρακολούθηση διεργασιών
- Παράλληλη επεξεργασία

# Κινητοί Πράκτορες

Η απήχηση της Java για την υλοποίηση multi-agent συστημάτων<sup>1</sup>

- **Ανεξαρτησία από πλατφόρμα**
  - Η Java είναι σχεδιασμένη να λειτουργεί σε ετερογενή περιβάλλοντα.
  - Επιτρέπει τη δημιουργία ενός αντιπρόσωπου που μπορεί να εκτελεστεί σε οποιαδήποτε πλατφόρμα που έχει εγκατεστημένη την εικονική μηχανή της Java.
- **Ασφαλής εκτέλεση**
  - Η Java έχει ένα μοντέλο δεικτών (pointers) που εξαφανίζει την πιθανότητα της εγγραφής θέσεων της μνήμης ή την καταστροφή δεδομένων
  - Η αρχιτεκτονική ασφάλειας της Java επιτρέπει την ασφαλή εκτέλεση ενός μη-έμπιστου αντιπρόσωπου διότι ο πράκτορας δε μπορεί να έχει πρόσβαση σε απόρρητες πληροφορίες.
- **Δυναμικό φόρτωμα κλάσεων**
  - Η εικονική μηχανή μπορεί να φορτώνει και να ορίζει κλάσεις κατά την διάρκεια εκτέλεσης ενός προγράμματος.
- **Πολυνηματικός προγραμματισμός (Multithreaded programming)**
  - Οι αντιπρόσωποι είναι εξ' ορισμού αυτόνομοι
  - Η Java όχι μόνο υποστηρίζει πολυνηματικό προγραμματισμό, αλλά υποστηρίζει επίσης ένα σύνολο από δυνατότητες συγχρονισμού (synchronization primitives) δομημένες στην ίδια τη γλώσσα.

# Κινητοί Πράκτορες

Η απήχηση της Java για την υλοποίηση multi-agent συστημάτων<sup>2</sup>

- **Σειριοποίηση αντικειμένων**

- Ένα βασικό χαρακτηριστικό των κινητών πρακτόρων είναι η δυνατότητά τους για σειριοποίηση και αποσειριοποίηση.
- Η Java παρέχει ένα ενδογενή μηχανισμό σειριοποίησης ο οποίος αναπαριστά την κατάσταση ενός αντικειμένου σε σειριοποιημένη μορφή με αρκετά λεπτομερή τρόπο ώστε το αντικείμενο να μπορεί να ανακατασκευαστεί αργότερα.

# Κινητοί Πράκτορες

## Μειονεκτήματα της Java

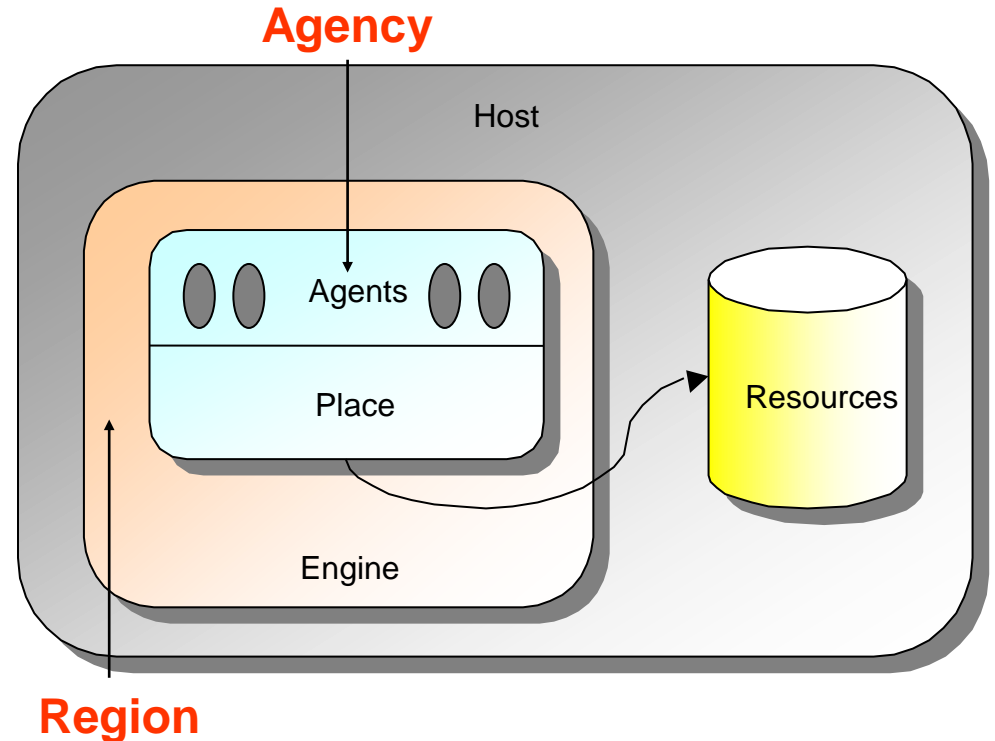
- **Ανεπαρκής υποστήριξη ελέγχου πόρων**
  - Δυστυχώς, η Java δεν παρέχει τρόπους περιορισμού των πόρων του επεξεργαστή και της μνήμης που διατίθεται σε κάποιο αντικείμενο ή νήμα.
  - Παρόμοιο θέμα είναι η ικανότητα ενός πράκτορα να δεσμεύει πόρους εξωτερικούς του προγράμματος, για παράδειγμα με το να ανοίγει αρχεία και διόδους επικοινωνίας (sockets), και παράγοντας παράθυρα.
- **Δεν υπάρχει προστασία των αναφορών ενός αντικειμένου**
  - Οι public μέθοδοι ενός Java αντικειμένου είναι διαθέσιμες σε κάθε άλλο αντικείμενο που έχει αναφορά σε αυτό.
  - Δεν υπάρχει τρόπος σε κάποιον πράκτορα να επιβλέπει και να ελέγχει τα αντικείμενα (πράκτορες) που έχουν πρόσβαση στις μεθόδους του.
- **Δεν υπάρχει υποστήριξη για διατήρηση και επανάκτηση της κατάστασης εκτέλεσης**
  - Είναι προς το παρόν αδύνατη η επανάκτηση της πλήρους κατάστασης εκτέλεσης ενός αντικειμένου με τη Java.
  - Πληροφορίες όπως η κατάσταση του μετρητή προγράμματος (program counter) και της στοίβας εντολών είναι απαγορευτικές για προγράμματα της Java (γλώσσα υψηλού επιπέδου).

# Κινητοί Πράκτορες

Κινητοί πράκτορες – το πρότυπο OMG MASIF  
(Mobile Agent System Interoperability Facilities)<sup>1</sup>

**Περιοχή (Region):** Διευκολύνει τη διαχείριση των πρακτορείων (agencies), τοποθεσιών (places) και πρακτόρων (agents). Μία περιοχή περιέχει μία μόνον περιοχή καταχώρησης (region registry) και πολλά πρακτορεία.

**Περιοχή Καταχώρησης (Region Registry):** Παρέχει τη βάση δεδομένων με πληροφορία σχετική για τα πρακτορεία, τοποθεσίες και τους πράκτορες μιας περιοχής.



# Κινητοί Πράκτορες

## Κινητοί πράκτορες - το πρότυπο MASIF<sup>1</sup>

- **Πρακτορείο (Agency):** Παρέχει το περιβάλλον εκτέλεσης των κινητών πρακτόρων. Κάθε πρακτορείο εκτελείται σε ξεχωριστή εικονική μηχανή της Java και αποτελείται από μία ή περισσότερες τοποθεσίες. Το πρακτορείο παρέχει τις βασικές λειτουργίες, όπως διαχείριση του κύκλου ζωής του πράκτορα, μεταφορά, επικοινωνία και ασφάλεια.
- **Τοποθεσία (Place):** Επιτρέπει την οργάνωση των πρακτόρων, που παρέχουν ίδιες υπηρεσίες. Οι τοποθεσίες ανήκουν στα πρακτορεία. Κάθε τοποθεσία μπορεί να συσχετιστεί με μια πολιτική ασφάλειας (security policy), για τη διευκόλυνση της διαχείρισης της ασφάλειας.
- **Πράκτορας (agent):** Οντότητα λογισμικού, που έχει τη δυνατότητα να δρα αυτόνομα, αποτελούμενη από μία ή περισσότερες Java κλάσεις. Υπάρχουν δύο είδη πρακτόρων: οι κινητοί (mobile) και οι στάσιμοι (stationary).

# Κινητοί Πράκτορες

Ο κύκλος ζωής του κινητού πράκτορα

