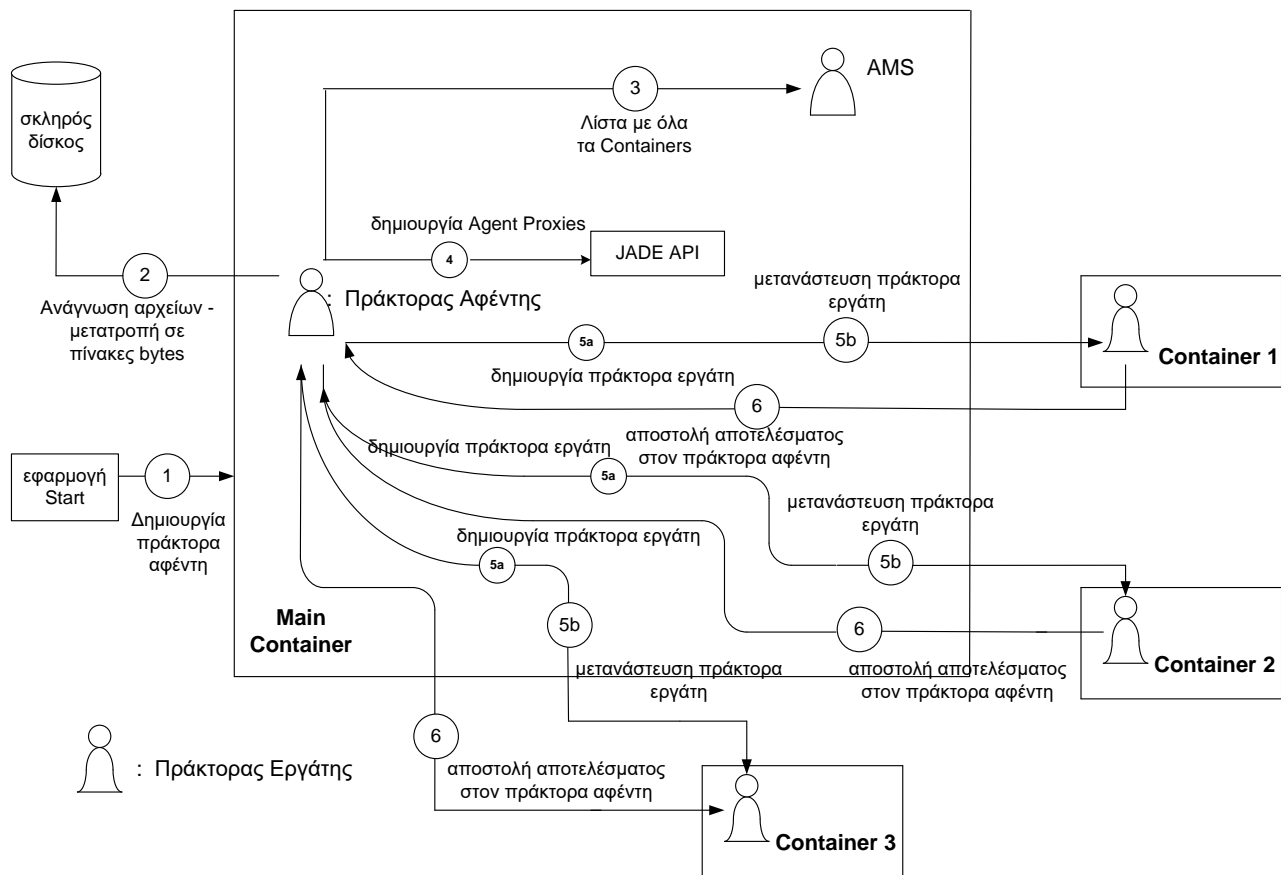


Άσκηση 2: Παράλληλη Κατανεμημένη Παραμετρική Προσομοίωση με Πράκτορες Λογισμικού

Η πιο απλή αρχιτεκτονική για την αξιοποίηση της τεχνολογίας των κινητών πρακτόρων στις Υπολογιστικές Τεχνικές για Συστήματα Μετάδοσης Πληροφορίας είναι αυτή του αφέντη – εργάτη (master – worker). Στην αρχιτεκτονική αυτή υπάρχει ένας βασικός πράκτορας, που ονομάζεται αφέντης (master), ο οποίος είναι υπεύθυνος για την ομαλή ροή της εφαρμογής. Ο πράκτορας αυτός είναι συνήθως σταθερός (stationary) και δημιουργεί μία ομάδα από κινητούς πράκτορες, στον καθένα από τους οποίους αναθέτει και ένα τμήμα της εργασίας. Οι πράκτορες αυτοί ονομάζονται εργάτες (workers) και σκοπός τους είναι να μεταβούν σε απομακρυσμένους υπολογιστές, όπου και θα εκτελέσουν τοπικά το τμήμα αυτό της εργασίας που τους έχει ανατεθεί. Με αυτόν τον τρόπο επιτυγχάνεται η κατανομή του φόρτου εργασίας και η παράλληλη εκτέλεση ανεξάρτητων μεταξύ τους τμημάτων της εφαρμογής. Στη συνέχεια, οι εργάτες επιστρέφουν με τα αποτελέσματα και παίρνουν ένα νέο τμήμα της εργασίας από τον αφέντη, αν αυτό υπάρχει. Στο τέλος, ο αφέντης ενώνει τα επιμέρους αποτελέσματα και βγάζει την τελική λύση.



Αναλυτική αρχιτεκτονική της εφαρμογής

a) Βασική υποδομή

Η εργασία αυτή χωρίζεται σε δύο τμήματα. Στο πρώτο καλείστε να αναπτύξετε τη βασική αρχιτεκτονική της εφαρμογής. Πιο αναλυτικά, καλείστε να αναπτύξετε τις πρώτες μορφές του πράκτορα – αφέντη και των πρακτόρων – εργατών. Ο πράκτορας αφέντης θα είναι ένας stationary πράκτορας, ο οποίος θα δημιουργεί έναν πράκτορα – εργάτη. Ο πράκτορας – εργάτης θα μεταναστεύει σε ένα απομακρυσμένο πρακτορείο, όπου και θα τυπώνει ένα μήνυμα.

b) Λειτουργικότητα

Επεκτείνετε τους προηγούμενους πράκτορες έτσι ώστε:

- Ο πράκτορας – αφέντης να διαβάζει ένα αρχείο εισόδου `input.properties` το οποίο περιέχει δύο νούμερα. Στη συνέχεια να αρχικοποιεί δύο πράκτορες – εργάτες, δίνοντάς τους σαν παράμετρο αρχικοποίησης το ένα νούμερο.
- Οι πράκτορες – εργάτες να μεταβαίνουν σε απομακρυσμένα containers, όπου να δημιουργούν ένα αρχείο, το οποίο περιέχει το νούμερο με το οποίο αρχικοποιήθηκαν. Στη συνέχεια, να σειριοποιούν το αρχείο αυτό και να επιστρέφουν με αυτό στο main container, όπου εκτελείται ο πράκτορας – αφέντης.

c) Παράλληλη Κατανεμημένη Παραμετρική Προσομοίωση

Χρησιμοποιώντας τον κώδικα της Άσκησης 1, καθώς και την πλατφόρμα Κινητών Πρακτόρων Λογισμικού Jade, η οποία έχει αναπτυχθεί στο εργαστήριο σε μοντέλο master-worker, να πραγματοποιήσετε κατανεμημένες προσομοιώσεις του συγκεκριμένου προβλήματος σκέδασης. Πιο συγκεκριμένα, να υπολογιστούν κατανεμημένα:

- Η μεταβολή του μονοστατικού RCS ($\sigma(\varphi=0)$) καθώς μεταβάλλεται η ακτίνα του κυλίνδρου από $\alpha=0.8\lambda$ μέχρι $\alpha=1.0\lambda$ σε 10 βήματα. Η ακτίνα της βοηθητικής επιφάνειας να διατηρηθεί στο ίδιο ποσοστό της ακτίνας του κυλίνδρου.
- Το μέγιστο σφάλμα της οριακής συνθήκης $(\Delta E_{bc})_{\max}$ καθώς μεταβάλλεται η ακτίνα της βοηθητικής επιφάνειας από 0.01α μέχρι 0.99α σε 20 βήματα, διατηρώντας πάντα σταθερή την ακτίνα του κυλίνδρου $\alpha=1.0\lambda$.

Ο κάθε πράκτορας – εργάτης (worker) έχει τη δυνατότητα να εκτελεί τον κώδικα με διαφορετικά αρχεία (ή παραμέτρους) εισόδου και να ενημερώνει τον πράκτορα – αφέντη (master) για τα αποτελέσματα. Ο πράκτορας – αφέντης μπορεί να συγκεντρώνει όλα τα αποτελέσματα και να δημιουργεί ένα αρχείο – πίνακα τιμών σε κάθε περίπτωση. Να χρησιμοποιήσετε αυτό το μοντέλο κατανεμημένης επεξεργασίας που περιγράφεται στο σχήμα, επεκτείνοντας κατάλληλα την πλατφόρμα που σας δίνεται για τις ανάγκες της εφαρμογής σας.

Το API που σας παρέχεται είναι σε γλώσσα Java, λέγεται Jade και παρέχει λειτουργικότητα για ανάπτυξη εφαρμογών με κινητούς πράκτορες λογισμικού κατά το πρότυπο FIPA. Προγραμματιστικά βοηθήματα και βιβλιοθήκες λογισμικού είναι διαθέσιμα στην ιστοσελίδα του μαθήματος (<https://mycourses.ntua.gr/> και <http://mfol.ece.ntua.gr/computational-techniques-for-information-transmission-systems/>).