

ΠΑΡΟΥΣΙΑΣΗ ΔΙΔΑΚΤΟΡΙΚΗΣ ΔΙΑΤΡΙΒΗΣ

ΗΜΕΡΟΜΗΝΙΑ:	Δευτέρα, 16 Ιουνίου 2014
ΩΡΑ:	19.00
ΑΙΘΟΥΣΑ:	Αίθουσα Σεμιναρίων (ισόγειο I1-I2) Κτήριο Τμήματος Μηχανικών Η/Υ & Πληροφορικής
ΟΜΙΛΗΤΗΣ:	Διονύσης Αθανασόπουλος

Θ έ μ α

«Σχεδίαση & Ανάπτυξη Υπηρεσιο-κεντρικού Λογισμικού με βάση Θεμελιώδεις Αρχές Αντικειμενοστρέφειας»

*(Designing & Developing Service-Oriented Software with respect to
Fundamental Object-Oriented Principles)*

Επταμελής Εξεταστική Επιτροπή (με αλφαβητική σειρά)

- **Παναγιώτης Βασιλειάδης**, Αναπληρωτής Καθηγητής του Τμήματος Μηχανικών ΗΥ & Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- **Απόστολος Ζάρρας**, Επίκουρος Καθηγητής του Τμήματος Μηχανικών ΗΥ & Πληροφορικής, Πανεπιστήμιο Ιωαννίνων (**Επιβλέπων**)
- **Valerie Issarny**, Research Director, Inria Paris-Rocquencourt, France
- **Barbara Pernici**, Καθηγήτρια, Dipartimento di Elettronica, Informazione, e Biongegneria, Politecnico di Milano
- **Ευαγγελία Πιτουρά**, Καθηγήτρια του Τμήματος Μηχανικών ΗΥ & Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- **Pierluigi Plebani**, Επ. Καθηγητής, Dipartimento di Elettronica, Informazione, e Biongegneria, Politecnico di Milano
- **Ελένη Στρούλια**, Καθηγήτρια, Dep. of Computing Science, University of Alberta, Edmonton, Canada

Περίληψη

Στην παρούσα διατριβή προτείνουμε τεχνικές για τη σχεδίαση και την ανάπτυξη υπηρεσιοκεντρικού λογισμικού (service-oriented software) με βάση θεμελιώδεις αρχές αντικειμενοστρέφειας (fundamental design principles). Υπηρεσιοκεντρικό είναι το είδος του λογισμικού το οποίο χρησιμοποιεί ως βασικά δομικά συστατικά του (components) μονάδες λογισμικού, που έχουν σχεδιαστεί και αναπτυχθεί από τρίτους παρόχους λογισμικού. Αυτά τα δομικά συστατικά είναι γνωστά ως υπηρεσίες (services) και συνήθως προσπελάσσονται από το υπηρεσιοκεντρικό λογισμικό διαμέσου του διαδικτύου (Web) χρησιμοποιώντας την τεχνολογία των υπηρεσιών διαδικτύου (Web-service technology). Οι θεμελιώδεις αρχές αντικειμενοστρέφειας είναι ένα σύνολο από οδηγίες σχεδίασης και ανάπτυξης αντικειμενοστρεφούς λογισμικού των οποίων η παραβίαση δημιουργεί προβλήματα συντήρησης του λογισμικού. Στην παρούσα διατριβή, αυτές οι αρχές οργανώνονται σε δύο ομάδες: (α) τις αρχές σύμφωνα με τις οποίες το λογισμικό πρέπει να αναπτύσσεται με βάση δομές αφαίρεσης (abstractions), και (β) τις αρχές σύμφωνα με τις οποίες το λογισμικό πρέπει να αναπτύσσεται με βάση μικρά και συνεκτικά δομικά συστατικά (cohesive components).

Όταν αυτές οι σχεδιαστικές αρχές παραβιάζονται, τα προβλήματα συντήρησης του υπηρεσιοκεντρικού λογισμικού είναι απροσδόκητα και πιο συχνά σε σχέση με την περίπτωση του αντικειμενοστρεφούς λογισμικού. Αυτό οφείλεται στην ιδιαιτερότητα του υπηρεσιοκεντρικού λογισμικού που σχετίζεται με το γεγονός ότι κάθε βασικό δομικό συστατικό του δεν ανήκει στον ιδιοκτήτη του, αλλά απλά χρησιμοποιείται από το λογισμικό. Η παρούσα διατριβή διαιρείται σε δύο τμήματα. Το πρώτο τμήμα ασχολείται με την εφαρμογή της πρώτης ομάδας σχεδιαστικών αρχών στην περίπτωση του υπηρεσιοκεντρικού λογισμικού, και αντίστοιχα το δεύτερο τμήμα ασχολείται με την εφαρμογή των αρχών της δεύτερης ομάδας.

Για την εφαρμογή της πρώτης ομάδας αρχών, προτείνουμε ότι το υπηρεσιοκεντρικό λογισμικό πρέπει να αναπτύσσεται με βάση αφηρημένες υπηρεσίες (abstract services). Ωστόσο, στην πράξη, αφηρημένες υπηρεσίες δεν είναι διαθέσιμες. Έτσι, προτείνουμε μία μέθοδο εξόρυξης (mining) αφηρημένων υπηρεσιών από ένα σύνολο υπαρχόντων υπηρεσιών. Η προτεινόμενη μέθοδος βασίζεται σε έναν αλγόριθμο σύγκρισης και συσχέτισης υπηρεσιών (service matching). Δεδομένου ότι τα δεδομένα εισόδου και εξόδου των λειτουργιών των υπηρεσιών (service operations) είναι πολύπλοκες XML γραμματικές (XML schemas), προτείνουμε έναν αποτελεσματικό (effective) και αποδοτικό (efficient) αλγόριθμο σύγκρισης τέτοιων γραμματικών.

Η προτεινόμενη μέθοδος εξόρυξης αφηρημένων υπηρεσιών συγκρίνει υπηρεσίες με δύο τρόπους: (i) προοδευτικά σε ζευγάρια (incremental pairwise) ή (ii) περισσότερες από δύο υπηρεσίες ταυτόχρονα (holistic). Συγκρίνοντας αυτούς τους δύο τρόπους εξόρυξης αφηρημένων υπηρεσιών, τα πειραματικά αποτελέσματα έδειξαν, ότι ο πρώτος τρόπος είναι πιο αποδοτικός αλλά λιγότερο αποτελεσματικός σε σχέση με το δεύτερο τρόπο.

Για την εφαρμογή της δεύτερης ομάδας αρχών, προτείνουμε μία μέθοδο διάσπασης (decomposition) μίας υπηρεσίας σε μικρότερες και πιο συνεκτικές υπηρεσίες. Για να υπολογίσουμε τη συνεκτικότητα μίας υπηρεσίας, προτείνουμε τρεις εναλλακτικές μετρικές. Η πρώτη μετρική, που εφαρμόζεται στα μηνύματα των λειτουργιών υπηρεσιών (message-level metric), βρίσκει σχέσεις μεταξύ των λειτουργιών που έχουν παρόμοια δεδομένα εισόδου και εξόδου. Η δεύτερη μετρική, που βασίζεται στην ανταλλαγή μηνυμάτων μεταξύ των λειτουργιών υπηρεσιών (conversation-level metric), βρίσκει τέτοιες σχέσεις ώστε τα δεδομένα εξόδου μίας λειτουργίας να είναι παρόμοια με τα δεδομένα εισόδου μιας άλλης λειτουργίας. Η τρίτη μετρική βρίσκει σχέσεις μεταξύ λειτουργιών των οποίων τα ονόματα χαρακτηρίζονται από όρους (terms), που ανήκουν σε παρόμοια θεματικά πεδία (domain-level metric). Αποτιμώντας την αποτελεσματικότητα και τη χρησιμότητα της προτεινόμενης μεθόδου σε πραγματικές υπηρεσίες διαδικτύου, τα πειραματικά αποτελέσματα έδειξαν, ότι η μέθοδος μπορεί να βελτιώσει τη συνεκτικότητα μίας υπηρεσίας διασπώντας την σε μικρότερες και πιο συνεκτικές υπηρεσίες. Επίσης, τα πειραματικά αποτελέσματα έδειξαν, ότι όταν η προτεινόμενη μέθοδος συνδυάζεται με την τρίτη μετρική, τότε η αποτελεσματικότητα της μεθόδου είναι υψηλότερη σε σχέση με τους συνδυασμούς της μεθόδου με τις άλλες δύο μετρικές.