ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

# Απόκτηση της Απαιτούμενης Κοινής Γνώσης σε Δίκτυα Διαχείρισης Τηλεπικοινωνιών

εργασία που υποβλήθηκε από τον

Σταθόπουλο Β. Κωνσταντίνο

σαν μερική απαίτηση για την απόκτηση
Μεταπτυχιακού Διπλώματος Εξειδίκευσης

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

# Απόκτηση της Απαιτούμενης Κοινής Γνώσης σε Δίκτυα Διαχείρισης Τηλεπικοινωνιών

εργασία που υποβλήθηκε από τον

Σταθόπουλο Β. Κωνσταντίνο

σαν μερική απαίτηση για την απόκτηση
Μεταπτυχιακού Διπλώματος Εξειδίκευσης

17 Νοεμβρίου, 1995

Συγγραφέας:

Σταθόπουλος Β. Κων/νος

Εισηγητική Επιτροπή:

Κουρκουμπέτης Κων/νος, Αναπληρωτής Καθηγητής, Επόπτης

Σταμούλης Γεώργιος, Επίκουρος Καθηγητής, Μέλος

Σερπάνος Δημήτριος, Επίκουρος Καθηγητής, Μέλος

Δεκτή:

Κωνσταντόπουλος Πάνος, Αναπληρωτής Καθηγητής
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

# Απόκτηση της Απαιτούμενης Κοινής Γνώσης σε Δίκτυα Διαχείρισης Τηλεπικοινωνιών

Κωνσταντίνος Β. Σταθόπουλος
Μεταπτυχιακή Εργασία

Πανεπιστήμιο Κρήτης
Τμήμα Επιστήμης Υπολογιστών

## Περίληψη

Το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (Telecommunications Management Network – ΔΔΤ) έχει προταθεί από τη Διεθνή Ενωση Τηλεπικοινωνιών (International Telecommunication Union – ITU) και είναι ένα κατανεμημένο περιβάλλον. Αποτελείται από ένα μεγάλο αριθμό διασυνδεδεμένων διεργασιών οι οποίες συνεργάζονται με σκοπό τη διαχείριση ενός δικτύου τηλεπικοινωνιών. Οι διεργασίες αυτές είναι υπεύθυνες για τη συγκέντρωση και επεξεργασία πολλών ειδών πληροφορίας σχετικά με τους πόρους και τις υπηρεσίες του διαχειριζόμενου δικτύου. Γι' αυτό το λόγο, το ΔΔΤ διασυνδέεται με το προηγούμενο σε πολλά σημεία του προκειμένου να υπάρξει τόσο η ροή της απαραίτητης πληροφορίας προς το ΔΔΤ όσο και η αποστολή των κατάλληλων πράξεων διαχείρισης προς το διαχειριζόμενο δίκτυο.

Η εργασία αυτή ασχολείται με ένα από τα βασικά προβλήματα σε περιβάλλοντα όπως το ΔΔΤ το οποίο είναι η απόκτηση της γνώσης που πρέπει να κατέχουν από κοινού οι διεργασίες του ΔΔΤ προκειμένου να απαντώνται ερωτήματα όπως: Για ένα συγκεκριμένο πόρο του δικτύου, ποια είναι η διεργασία η οποία κρατάει κάποια συγκεκριμένη πληροφορία – π.χ. τα στατιστικά απόδοσης (performance statistics) – για αυτόν τον πόρο; Με ποιον τρόπο περιγράφεται η παραπάνω πληροφορία στη ζητούμενη διεργασία· με άλλα λόγια, ποιο είναι το μοντέλο της πληροφορίας που υποστηρίζει η διεργασία; Ποια είναι η διεύθυνση στην οποία η διεργασία περιμένει να εξυπηρετήσει αιτήσεις για την πληροφορία που κρατάει;

Ολα τα παραπάνω ερωτήματα αναφέρονται σε πληροφορία η οποία καλείται Κοινή Γνώση Διαχείρισης (ΚΓΔ) και αποτελείται από την πληροφορία που χρειάζονται οι διεργασίες του

ΔΔΤ προκειμένου να επικοινωνήσουν και να συνεργαστούν, όχι μόνο μέσα στο ίδιο ΔΔΤ αλλά και μεταξύ διαφορετικών ΔΔΤ. Η πληροφορία αυτή μπορεί να αλλάζει κατά τη διάρκεια λειτουργίας ενός ΔΔΤ (για παράδειγμα, μία διεργασία μπορεί να μετακινηθεί σε μία καινούρια διεύθυνση λόγω κάποιας βλάβης). Για το λόγο αυτό πρέπει όχι μόνο να φυλάσσεται και να διατίθεται στις διεργασίες αλλά και να είναι δυνατό να ενημερώνεται ανά πάσα στιγμή.

Η *Υπηρεσία Καταλόγου* (Directory Service – X.500) που έχει προταθεί από την *ITU* και τον Διεθνή Οργανισμό Τυποποίησης (*International Standards Organization – ISO*) παρέχει μια κατανεμημένη βάση πληροφορίας καθώς και μία αρχιτεκτονική και ένα σύνολο πρωτοκόλλων τα οποία υποστηρίζουν συγκεκριμένους μηχανισμούς για πρόσβαση σε αυτή τη βάση πληροφορίας. Η βάση αυτή – η οποία καλείται Κατάλογος – μπορεί να κρατάει πληροφορία για ένα μεγάλο αριθμό και είδος οντοτήτων (π.χ. διεργασίες, πόρους του δικτύου, ανθρώπους) οι οποίες εγγράφονται με ιεραρχικό τρόπο (ανά χώρα, οργανισμό, γεωγραφική περιοχή κ.ά.). Βάσει αυτής της ιεραρχικής μεθοδολογίας εγγραφής, οποιαδήποτε οντότητα – για την οποία φυλάσσεται πληροφορία στο Κατάλογο – αποκτά ένα μοναδικό όνομα το οποίο δεν είναι άλλο από το αναγνωριστικό της αντίστοιχης εγγραφής του Καταλόγου. Η συνολική πληροφορία του Καταλόγου κατανέμεται σε ένα μεγάλο αριθμό από μονάδες εξυπηρέτησης οι οποίες καλούνται *DSA* (Directory Service Agents). Το βασικό χαρακτηριστικό του X.500 είναι ότι δίνει τη δυνατότητα πρόσβασης στη συνολική πληροφορία του Καταλόγου μέσω οποιουδήποτε DSA χάρη στη χρήση του πρωτοκόλλου DSP (Directory Service Protocol) για την επικοινωνία και μεταφορά πληροφορίας μεταξύ των διαφόρων DSA.

Η συμβολή της εργασίας αυτής αφορά στον προσδιορισμό της απαιτούμενης κοινής γνώσης σε ένα ΔΔΤ (δηλ. της ΚΓΔ) και στην επέκταση του μοντέλου επικοινωνίας των διεργασιών του ΔΔΤ με την προσθήκη ενός μηχανισμού για τη δυναμική απόκτηση της Κοινής Γνώσης Διαχείρισης. Ο μηχανισμός αυτός περιλαμβάνει την αυτόματη ενημέρωση της ΚΓΔ στον Κατάλογο και την αναζήτησή της χρησιμοποιώντας τα πρωτόκολλα του X.500. Οι βασικές οντότητες που εγγράφονται στο X.500 είναι στοιχεία του διαχειριζόμενου δικτύου (όπως μεταγωγείς, δρομολογητές και σταθμοί εργασίας), ομάδες αυτών (υποδίκτυα) καθώς και διεργασίες του ΔΔΤ.

Χρησιμοποιώντας τον παραπάνω μηχανισμό δείχνουμε πώς μπορούμε, δοθέντος ενός πόρου του διαχειριζόμενου δικτύου, να βρούμε τη διεργασία (ή τις διεργασίες) του ΔΔΤ η οποία περιέχει κάποια ζητούμενη πληροφορία σχετική με αυτόν τον πόρο. Επιπλέον περιγράφουμε έναν τρόπο με τον οποίο μπορούμε να παράσχουμε *διαφάνεια ως προς την τοποθεσία* (location transparency). Στο ΔΔΤ η διαφάνεια ως προς την τοποθεσία αναφέρεται σε έναν τρόπο επικοινωνίας των διεργασιών ο οποίος είναι ανεξάρτητος του πού βρίσκεται η κάθε διεργασία και, συνεπώς, κρύβει την ακριβή τοποθέτησή τους μέσα στο ΔΔΤ.

Τα παραπάνω υλοποιήθηκαν και ένα σημαντικό τμήμα τους έχει περιληφθεί σαν μέρος του πακέτου λογισμικού *OSIMIS* το οποίο αποτελεί την ευρύτερα διαδεδομένη – μεταξύ αυτών που διατίθονται δωρεάν – πλατφόρμα ανάπτυξης συστημάτων (που χρησιμοποιούν τα μοντέλα και τα πρωτόκολλα του *ISO*) για τη διαχείριση δικτύων τηλεπικοινωνιών.

Επόπτης: Κων/νος Κουρκουμπέτης
Αναπληρωτής Καθηγητής του Τμήματος Επιστήμης Υπολογιστών
του Πανεπιστημίου Κρήτης

# Acquisition of the Shared Knowledge Required in Telecommunications Management Networks

Costas V. Stathopoulos
M.Sc. Thesis

University of Crete
Department of Computer Science

## Abstract

The Telecommunications Management Network (TMN) has been recommended by the *International Telecommunication Union (ITU)* and is a distributed environment. It consists of a large number of interconnected processes that cooperate in order to manage an underlying telecommunications network. These processes are responsible for collecting and processing many kinds of information regarding the resources and services of the managed network. Hence, the TMN interfaces to the managed network in many points so that the required information is obtained by the former and the appropriate management operations are sent to the latter.

This thesis deals with one of the basic problems in this environment, namely the acquisition of the knowledge that must be shared among the TMN processes in order to answer such questions as: Given a resource of the managed network, which is the process that keeps specific information – e.g. the performance statistics – for that resource? How is that information presented; in other words, what information model does the process support? Which is the address where the process is awaiting for requests regarding the information that it keeps?

All the above questions are related to the so-called Shared Management Knowledge (SMK) which consists of the information that the TMN processes require in order to communicate and cooperate, not only inside the same TMN but also between seperate TMNs. This information may change during the lifetime of a TMN (e.g. a process might migrate to

a new address due to an internal TMN failure). Hence, not only SMK should be stored and available to the processes but also it must be updated whenever it changes.

The *Directory Service* (aka X.500) which is recommended by both the *ITU* and the *International Standards Organization (ISO)* provides a distributed information base as well as the architecture and protocols that support certain mechanisms for accessing this information base. This base – referred to as the "Directory" – is suitable for keeping information concerning a large number and variety of objects (e.g. people, network resources, processes) that are registered in a hierarchical fashion (per country, organization, geographical locality etc.). Due to this hierarchical registration paradigm, every object acquires a unique name that is simply the identifier of the corresponding entry in the Directory. The overall Directory information is distributed in a large number of servers that are called Directory Service Agents (DSAs). An important characteristic of X.500 is that is provides the user with the capability of accessing the whole information base via any DSA. This is accomplished through the use of a sophisticated protocol that supports communication and information transfer between the DSAs.

The contribution of this work concerns the determination of the required shared knowledge in a TMN (i.e. the SMK) as well as the enhancement of the model of interaction between the TMN processes by adding a mechanism for the dynamic acquisition of Shared Management Knowledge. This mechanism includes the automatic update of the SMK in the Directory information base and its search by exploiting the X.500 protocols. The main objects that are registered in the Directory are elements of the managed network (such as switches, routers and workstations), groups of them (i.e. subnetworks) as well as TMN processes.

Taking advantage of the above mechanism, we show how, given a resource of the managed network, we can identify the TMN process (or processes) that contains some information concerning the management of the resource. Additionally, we describe a mechanism that provides *location transparency*. Within a TMN, location transparency refers to a location-independent way of communication between processes that, therefore, hides their exact physical placement in the TMN.

The above were implemented and a significant part of the implementation is included in the *OSIMIS* software package which provides the most widely-known publicly available platform for the development of systems (based on the *ISO* models and protocols) for the management of telecommunication networks.

Supervisor: Costas Courcoubetis
Associate Professor of the Department of Computer Science
of the University of Crete

*στην αδερφή μου Δήμητρα*

## *Ευχαριστίες*

Κατ' αρχήν θα ήθελα να ευχαριστήσω τον καθηγητή μου Κώστα Κουρκουμπέτη για τη στήριξη και την επίβλεψη αυτής της εργασίας. Επίσης ευχαριστώ τους καθηγητές μου Γιώργο Σταμούλη και Δημήτρη Σερπάνο για τη συμμετοχή τους στην επιτροπή εξέτασης καθώς και για τις πολύτιμες διορθώσεις και συστάσεις τους στη σύνταξη της παρούσας αναφοράς.
Είμαι ευγνώμων σε όλα τα μέλη του ερευνητικού προγράμματος *ICM* (RACEII project 2059) της Ευρωπαϊκής Ενωσης για την υποστήριξη αυτής της εργασίας και ιδιαίτερα στους Γιώργο Παύλου και Kevin McCarthy από το πανεπιστήμιο *University College of London* για τις προτάσεις τους στα διάφορα στάδια της υλοποίησης καθώς και για τη συμπερίληψη τμήματος του κώδικα στο πακέτο λογισμικού *OSIMIS*. Θέλω να απευθύνω τις θερμότερες ευχαριστίες μου στους Στέλιο Σαρτζετάκη και David Griffin για τις πολύωρες συζητήσεις που είχαμε τόσο στην αρχή όσο και κατά τη διάρκεια της εκπόνησης αυτής της εργασίας. Η συμβολή τους ήταν πραγματικά ανεκτίμητη.
Ευχαριστώ τέλος, το Ινστιτούτο Πληροφορικής του Ιδρύματος Τεχνολογίας και Ερευνας για την τεχνική και οικονομική υποστήριξη που μου παρείχε κατά τη διάρκεια των μεταπτυχιακών μου σπουδών και το Τμήμα Επιστήμης Υπολογιστών του Πανεπιστημίου Κρήτης για την άψογη οργάνωσή του και το υψηλό επίπεδο σπουδών που μού προσέφερε.

# Περιεχόμενα

**Εισαγωγή** **1**

**1 Σύντομη Ανασκόπηση της Περιοχής** **9**

1.1 Το πλαίσιο διαχείρισης συστημάτων του ISO . . . . . . . . . . . . . . . . . . . . 9

1.1.1 Το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου . . . . . . . . . . . 10

1.1.2 Το μοντέλο της πληροφορίας διαχείρισης . . . . . . . . . . . . . . . . . 12

1.1.3 Το πρωτόκολλο CMIP και η υπηρεσία CMIS . . . . . . . . . . . . . . 15

1.2 Το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (ΔΔΤ) . . . . . . . . . . . . . . . . . 18

1.2.1 Ομαδοποίηση και διαχωρισμός των λειτουργιών του ΔΔΤ . . . . . . . . 22

1.2.2 Το ιεραρχικό μοντέλο διαχείρισης του ΔΔΤ . . . . . . . . . . . . . . . 24

1.2.3 Μερικές παρατηρήσεις πάνω στο ΔΔΤ . . . . . . . . . . . . . . . . . . 26

1.2.4 Παράδειγμα ενός ΔΔΤ για ένα απλό δίκτυο μεταγωγέων . . . . . . . . 27

1.3 Η υπηρεσία καταλόγου (X.500) . . . . . . . . . . . . . . . . . . . . . . . . . . 32

1.3.1 Το μοντέλο λειτουργίας του X.500 . . . . . . . . . . . . . . . . . . . . 32

1.3.2 Το μοντέλο της πληροφορίας του X.500 . . . . . . . . . . . . . . . . . 33

1.4 Η Κοινή Γνώση Διαχείρισης (ΚΓΔ) . . . . . . . . . . . . . . . . . . . . . . . . 39

1.4.1 Προσδιορισμός της Κοινής Γνώσης Διαχείρισης κατά ISO . . . . . . . 39

1.4.2 Πληροφορία που συμπεριλάβαμε στη ΚΓΔ . . . . . . . . . . . . . . . . 42

**2 Το προτεινόμενο μοντέλο και οι βασικές χρήσεις του** **45**

2.1 Η καθολική βάση πληροφορίας διαχείρισης . . . . . . . . . . . . . . . . . . . . 47

2.2 Το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου . . . . . . . . . . . . . 49

2.3 Χρησιμοποιώντας τη Κοινή Γνώση Διαχείρισης . . . . . . . . . . . . . . . . . 51

2.3.1 Πληροφορία για πόρους και διεργασίες στο X.500 . . . . . . . . . . . . 53

2.3.2 Αντιστοίχιση διαχειρίσιμων οντοτήτων σε πόρους . . . . . . . . . . . . 59

2.4 Ο μηχανισμός διαφάνειας ως προς την τοποθεσία . . . . . . . . . . . . . . . . 61
2.5 Παράδειγμα εφαρμογής του νέου μοντέλου . . . . . . . . . . . . . . . . . . . . 63
2.6 Η υπηρεσία του καθολικού CMIS/CMIP . . . . . . . . . . . . . . . . . . . . . 65

**3 Η υλοποίηση** **70**
3.1 Το σύστημα *QUIPU* και οι βασικές οντότητές του . . . . . . . . . . . . . . . . 71
3.1.1 Ορισμός των νέων οντοτήτων του καταλόγου σε ASN.1 . . . . . . . . . 72
3.2 Το πακέτο λογισμικού *OSIMIS* και η θέση των νέων λειτουργιών σε αυτό . . . 79
3.2.1 Οι βασικές ρουτίνες του DUA . . . . . . . . . . . . . . . . . . . . . . . 85

**4 Συμπεράσματα και μελλοντικές επεκτάσεις** **90**
4.1 Μελλοντικές επεκτάσεις . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 92

**Βιβλιογραφία** **95**

# Κατάλογος Σχημάτων


1.1 Το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου του *ISO*. . . . . . . . . . . 10
1.2 Μία απλή υποθετική MIB για ένα μεταγωγέα (switchX). . . . . . . . . . . . . 13
1.3 Το δέντρο ονομασίας (naming tree) μίας MIB. . . . . . . . . . . . . . . . . . . 14
1.4 Η γενική αρχιτεκτονική του ΔΔΤ της *ITU*. . . . . . . . . . . . . . . . . . . . 19
1.5 Το γενικό ιεραρχικό μοντέλο διαχείρισης του ΔΔΤ. . . . . . . . . . . . . . . 23
1.6 Ενα πιθανό ΔΔΤ για ένα δίκτυο τριών μεταγωγέων. . . . . . . . . . . . . . . . 28
1.7 Το μοντέλο λειτουργίας της υπηρεσίας καταλόγου (X.500). . . . . . . . . . . . 33
1.8 Παράδειγμα μιας υποθετικής DIB και της λογικής της οργάνωσης. . . . . . . . 34
1.9 Παράδειγμα των ζευγαριών χαρακτηριστικού-τιμής για μία οντότητα καταλόγου. 36
1.10 Η γενική δομή μίας διεργασίας διαχείρισης. . . . . . . . . . . . . . . . . . . . 41

2.1 Η καθολική ονομασία των διαχειρίσιμων οντοτήτων. . . . . . . . . . . . . . . . 48
2.2 Το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου. . . . . . . . . . . . . 50
2.3 Η σχέση των διαφόρων πόρων. . . . . . . . . . . . . . . . . . . . . . . . . . . 54
2.4 Οντότητες για πόρους και διεργασίες στο X.500. . . . . . . . . . . . . . . . . . 58
2.5 Σχηματική περιγραφή της υπηρεσίας του καθολικού CMIS/CMIP. . . . . . . . 67

3.1 Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν πόρους. . . . . . . . . . 73
3.2 Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν διεργασίες. . . . . . . . 75
3.3 Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν SMAE. . . . . . . . . . 77
3.4 Η γενική αρχιτεκτονική και τα γενικά τμήματα του *OSIMIS*. . . . . . . . . . . 81
3.5 Επικοινωνία μεταξύ DUA και DSA χρησιμοποιώντας το πρωτόκολλο LDAP. . . 88

# Συντομογραφίες

| | |
|---|---|
| ΔΔΤ | Δίκτυο Διαχείρισης Τηλεπικοινωνιών |
| ΔΟ | Διαχειρίσιμη Οντότητα |
| ΚΓΔ | Κοινή Γνώση Διαχείρισης |
| ACSE | Association Control Service Element |
| ASE | Application Service Element |
| ASN.1 | Abstract Syntax Notation One |
| CLNS | Connection-Less Network Service |
| CMIP | Common Management Information Protocol |
| CMIS | Common Management Information Service |
| CMISE | Common Management Information Service Element |
| DAP | Directory Access Protocol |
| DCN | Data Communications Network |
| DIB | Directory Information Base |
| DN | Distinguished Name |
| DSA | Directory Service Agent |
| DSP | Directory Service Protocol |
| DUA | Directory User Agent |
| GDMO | Guidelines for the Definition of Managed Objects |
| GMS | Generic Managed System |
| ICM | Integrated Communications Management |
| IP | Internet Protocol |
| ISO | International Standards Organization |
| ISODE | ISO Development Environment |
| ITU | International Telecommunication Union |
| LDAP | Lightweight Directory Access Protocol |
| LDN | Local Distinguished Name |
| MD | Mediation Device |
| MIB | Management Information Base |
| MO | Managed Object |
| NE | Network Element |
| NELM | Network Element Level Management |

| | |
|---|---|
| NLM | Network Level Management |
| OS | Operations System |
| OSI | Open Systems Interconnection |
| OSIMIS | Open Systems Interconnection Management Information Service |
| OSIRM | Open Systems Interconnection Reference Model |
| pepsy | Presentation Element Parser and Structure-generator Yacc-based |
| PSAP | Presentation Service Access Point |
| QA | Q Adaptor |
| RDN | Relative Distinguished Name |
| RMIB | Remote Management Information Base |
| ROSE | Remote Operations Service Element |
| SMAE | Systems Management Application Entity |
| SMAP | Systems Management Application Process |
| SMASE | Systems Management Application Service Element |
| SMK | Shared Management Knowledge |
| SNMP | Simple Network Management Protocol |
| TCP | Transmission Control Protocol |
| TMN | Telecommunication Management Network |
| WAN | Wide Area Network |
| WS | WorkStation |

# Εισαγωγή

Η διαχείριση των δικτύων τηλεπικοινωνιών είναι ένα θέμα τόσο παλιό όσο και τα πρώτα τηλεφωνικά δίκτυα και αφορά στις αποφάσεις και ενέργειες που εξασφαλίζουν τη σωστή και όσο το δυνατό αποδοτικότερη χρήση και λειτουργία των δικτύων. Τα σημερινά περιβάλλοντα διαχείρησης δικτύων τηλεπικοινωνιών βασίζονται επί το πλείστον σε συστήματα ηλεκτρονικών υπολογιστών και αποτελούνται από διάσπαρτες διεργασίες οι οποίες είναι υπεύθυνες για τη συγκέντρωση και επεξεργασία πολλών ειδών πληροφορίας σχετικά με τους πόρους και τις υπηρεσίες των διαχειριζόμενων δικτύων[1].
Η εργασία αυτή ασχολείται με ένα από τα βασικά προβλήματα σε τέτοια περιβάλλοντα το οποίο αφορά στην απόκτηση της κοινής γνώσης που πρέπει να υπάρχει προκειμένου οι διεργασίες διαχείρισης να συνεργαστούν όσο το δυνατό αποτελεσματικότερα. Προτείνεται μια αρχιτεκτονική η οποία παρέχει τους αναγκαίους μηχανισμούς προκειμένου να απαντώνται ερωτήματα όπως: Για ένα συγκεκριμένο πόρο του δικτύου ποια είναι η διεργασία η οποία κρατάει κάποια συγκεκριμένη πληροφορία – π.χ. τα στατιστικά απόδοσης (performance statistics) – για αυτόν τον πόρο; Με ποιον τρόπο περιγράφεται η παραπάνω πληροφορία στη ζητούμενη διεργασία· με άλλα λόγια, ποιο είναι το μοντέλο της πληροφορίας που υποστηρίζει η διεργασία; Ποια είναι η διεύθυνση στην οποία η διεργασία περιμένει να εξυπηρετήσει αιτήσεις για την πληροφορία που κρατάει;
Ας σημειωθεί ότι το τελευταίο ερώτημα διαφέρει από το πρώτο, το οποίο αναφέρεται σε κάποιο αναγνωριστικό όνομα της διεργασίας που παραμένει το ίδιο σε ένα περιβάλλον διαχείρισης. Αντίθετα, η διεύθυνση μίας διεργασίας είναι δυνατό να αλλάξει και γι' αυτό το λόγο κατά κανόνα δεν χρησιμοποιείται σαν αναγνωριστικό της διεργασίας. Οπως θα φανεί στη συνέχεια, στα παραπάνω περιβάλλοντα, λόγω της ευρείας κατανομής των διεργασιών και της μη συγκεκριμένης τοποθέτησής τους καθώς και λόγω των πολλών δυνατών αρμοδιοτήτων και λειτουργιών τους, οι απαντήσεις στα ερωτήματα που τέθηκαν κάθε άλλο παρά άμεσες είναι.

[1] Στην παρούσα αναφορά, ο όρος "δίκτυο" θα χρησιμοποιείται ισοδύναμα με τον όρο "δίκτυο τηλεπικοινωνιών".

Πολλοί διεθνείς οργανισμοί και κυρίως ο Διεθνής Οργανισμός Τυποποίησης (*International Standards Organization – ISO*) και η Διεθνής Ενωση Τηλεπικοινωνιών (*International Telecommunication Union – ITU*) έχουν εκτεταμένα ασχοληθεί με τη δημιουργία προτύπων σχετικών με τη διαχείριση συστημάτων και δικτύων τηλεπικοινωνιών. Συγκεκριμένα, η *ITU* έχει συντάξει τη σειρά συστάσεων (recommendations) M.30 [Uni93d] η οποία περιγράφει το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (*Telecommunications Management Network – TMN*), τις λειτουργίες του καθώς και τις υπηρεσίες που πρέπει αυτό να παρέχει για τη διαχείριση ενός δικτύου, κυρίως ευρείας περιοχής (*Wide Area Network – WAN*). Ειδικότερα, το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (*ΔΔΤ*) είναι ένα ξεχωριστό – νοητά και όχι κατ' ανάγκη φυσικά – δίκτυο το οποίο διασυνδέεται με ένα διαχειριζόμενο δίκτυο σε πολλά σημεία του, προκειμένου να στέλνει πληροφορία από και προς το τελευταίο με σκοπό τη διαχείρισή του[2]. Εσωτερικά το ΔΔΤ είναι ένα κατανεμημένο σύστημα: ένας μεγάλος αριθμός από διάσπαρτες διεργασίες αποθηκεύει και ενημερώνει την πληροφορία διαχείρισης (management information) που λαμβάνεται για τους πόρους και τις υπηρεσίες του διαχειριζόμενου δικτύου. Οι διεργασίες αυτές καλούνται **αντιπρόσωποι διαχείρισης** (management agents). Επιπλέον, το ΔΔΤ περιέχει διεργασίες, οι οποίες καλούνται **διαχειριστές** (managers), που προσπελαύνουν και επεξεργάζονται την πληροφορία που κρατιέται στους αντιπροσώπους διαχείρισης. Τα δυο παραπάνω είδη διεργασιών του ΔΔΤ, που με ένα όνομα θα καλούμε **διεργασίες διαχείρισης** (management processes), επικοινωνούν μεταξύ τους σύμφωνα με το **μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου** (manager/agent model of interaction) [Org92a] χρησιμοποιώντας την υπηρεσία/πρωτόκολλο επικοινωνίας *CMIS/CMIP* (Common Management Information Service/Protocol) [Org90b][Org90a] τα οποία έχουν προταθεί από τον Διεθνή Οργανισμό Τυποποίησης (*ISO*).
Ενα τυπικό ΔΔΤ μπορεί να έχει εκατοντάδες διεργασίες οι οποίες χρειάζεται να επικοινωνούν μεταξύ τους. Μια από τις βασικές έννοιες σε ένα τέτοιο περιβάλλον είναι η έννοια της **Κοινής Γνώσης Διαχείρησης** (Shared Management Knowledge) η οποία εισήχθη από τον *ISO* [Org92a]. Η Κοινή Γνώση Διαχείρισης (ΚΓΔ) αναφέρεται στην πληροφορία που χρειάζονται οι παραπάνω διεργασίες προκειμένου να επικοινωνήσουν και να συνεργαστούν όχι μόνο μέσα στο ίδιο ΔΔΤ αλλά και μεταξύ διαφορετικών ΔΔΤ.
Τι είδους πληροφορία όμως πρέπει να περιέχεται στη ΚΓΔ; Στην πραγματικότητα τα

[2] Πρέπει να σημειωθεί σε αυτό το σημείο η διαφορά μεταξύ *δικτύου διαχείρισης* και *διαχειριζόμενου δικτύου*. Το πρώτο αναφέρεται σε ένα διασυνδεδεμένο σύστημα διαχείρισης ενώ το δεύτερο αναφέρεται στο δίκτυο τηλεπικοινωνιών που διαχειριζόμαστε με τη βοήθεια του πρώτου.

ερωτήματα που τέθηκαν στην αρχή αναφέρονται σε πληροφορία η οποία περιλαμβάνεται στη ΚΓΔ. Η πληροφορία αυτή μπορεί να αλλάζει κατά τη διάρκεια λειτουργίας ενός ΔΔΤ (για παράδειγμα, μία διεργασία μπορεί να μετακινηθεί σε μία καινούρια διεύθυνση λόγω κάποιας βλάβης). Για το λόγο αυτό πρέπει όχι μόνο να φυλάσσεται και να διατίθεται στις διεργασίες διαχείρισης αλλά και να είναι δυνατό να ενημερώνεται ανά πάσα στιγμή. Λαμβάνοντας υπόψη τη γενική δομή του ΔΔΤ καταλαβαίνουμε ότι η ΚΓΔ αναφέρεται σε κατανεμημένη πληροφορία. Η δε απόκτησή της απαιτεί κάποιο μηχανισμό ο οποίος θα προσφέρει τη δυνατότητα αποθήκευσης και πρόσβασης (ανάγνωσης και γραφής) σε μια κατανεμημένη βάση πληροφορίας.

Η **Υπηρεσία Καταλόγου** (Directory Service – X.500) που έχει προταθεί από την *ITU* και τον *ISO* [Uni93a] έχει σαν βασικό στόχο την παροχή ενός παγκόσμιου on-line καταλόγου. Για το λόγο αυτό παρέχει μια κατανεμημένη βάση πληροφορίας καθώς και ένα σύνολο πρωτοκόλλων τα οποία υποστηρίζουν συγκεκριμένους μηχανισμούς για πρόσβαση σε αυτήν τη βάση πληροφορίας. Η βάση αυτή – η οποία καλείται Κατάλογος – μπορεί να κρατάει πληροφορία για ένα μεγάλο αριθμό και είδος οντοτήτων (π.χ. διεργασίες, πόρους του δικτύου, ανθρώπους) οι οποίες εγγράφονται με ιεραρχικό τρόπο (ανά χώρα, οργανισμό, γεωγραφική περιοχή κ.ά.). Βάσει αυτής της ιεραρχικής μεθοδολογίας εγγραφής, οποιαδήποτε οντότητα – για την οποία φυλάσσεται πληροφορία στο Κατάλογο – αποκτά ένα παγκοσμίως μοναδικό όνομα το οποίο δεν είναι άλλο από το αναγνωριστικό της αντίστοιχης εγγραφής του Καταλόγου. Η συνολική πληροφορία του Καταλόγου κατανέμεται σε ένα μεγάλο αριθμό από μονάδες εξυπηρέτησης οι οποίες καλούνται *DSA* (Directory Service Agents). Το βασικό χαρακτηριστικό του X.500 είναι ότι δίνει τη δυνατότητα πρόσβασης στη συνολική πληροφορία του Καταλόγου μέσω οποιουδήποτε DSA χάρη στη χρήση του πρωτοκόλλου DSP (Directory Service Protocol) για την επικοινωνία και μεταφορά πληροφορίας μεταξύ των διαφόρων DSA.

Η συμβολή της εργασίας αυτής αφορά στον προσδιορισμό της απαιτούμενης κοινής γνώσης σε ένα ΔΔΤ και στην επέκταση του μοντέλου επικοινωνίας διαχειριστή/αντιπροσώπου του *ISO* με την προσθήκη ενός μηχανισμού για τη δυναμική απόκτηση της Κοινής Γνώσης Διαχείρισης. Ο μηχανισμός αυτός περιλαμβάνει την αυτόματη ενημέρωση της ΚΓΔ στον Κατάλογο και την αναζήτησή της χρησιμοποιώντας τα πρωτόκολλα του X.500. Οι βασικές οντότητες που εγγράφονται στο X.500 είναι στοιχεία του διαχειριζόμενου δικτύου (όπως μεταγωγείς, δρομολογητές, σταθμοί εργασίας και μονάδες εξυπηρέτησης του δικτύου[3]), ομάδες αυτών

[3] Network servers.

(υποδίκτυα) καθώς και διεργασίες του ΔΔΤ. Πρέπει να διευκρινίσουμε ότι αναφερόμαστε σε "δυναμική απόκτηση της ΚΓΔ" επειδή δίνεται η δυνατότητα σε κάθε διεργασία να προσπελαύνει τον Κατάλογο κατά τη διάρκεια λειτουργίας της και όχι μόνο τη στιγμή που ξεκινά.
Ακόμα, θα περιγράψουμε το μοντέλο της καθολικής βάσης πληροφορίας διαχείρισης (global management information base) το οποίο πηγάζει από τη συνένωση των δύο μοντέλων πληροφορίας: του καταλόγου και της διαχείρισης. Το βασικό πλεονέκτημά του είναι ότι ενώ στο κλασσικό μοντέλο της πληροφορίας διαχείρισης οι διάφορες μονάδες της πληροφορίας αναγνωρίζονται με ονόματα στο τοπικό επίπεδο του κάθε αντιπροσώπου που περιέχει τη ζητούμενη πληροφορία, στο μοντέλο που θα περιγράψουμε οι διάφορες μονάδες της πληροφορίας μπορούν να αναγνωρίζονται με παγκοσμίως μοναδικά ονόματα[4].
Χρησιμοποιώντας τους παραπάνω μηχανισμούς δείχνουμε πώς μπορούμε, δοθέντος ενός πόρου του διαχειριζόμενου δικτύου, να βρούμε τη διεργασία (ή τις διεργασίες) του ΔΔΤ η οποία περιέχει κάποια ζητούμενη πληροφορία σχετική με αυτόν τον πόρο. Επίσης, περιγράφουμε έναν τρόπο με τον οποίο παρέχεται **διαφάνεια ως προς την τοποθεσία** (location transparency) στο ΔΔΤ. Η διαφάνεια ως προς την τοποθεσία είναι μια από τις πολλές διαφάνειες κατανομής (distribution transparencies) οι οποίες είναι επιθυμητές στα περιβάλλοντα κατανεμημένων συστημάτων. Οι διαφάνειες κατανομής αναφέρονται σε μηχανισμούς οι οποίοι έχουν σκοπό να διευκολύνουν το χρήστη ενός κατανεμημένου συστήματος, μειώνοντας την πολυπλοκότητα που εισάγει το γεγονός της κατανομής των διαφόρων πόρων και διεργασιών του συστήματος. Αυτό συχνά πετυχαίνεται κρατώντας αθέατο από το χρήστη το γεγονός ότι το σύστημα είναι κατανεμημένο. Στο ΔΔΤ η διαφάνεια ως προς την τοποθεσία αναφέρεται σε έναν τρόπο επικοινωνίας των διεργασιών διαχείρισης ο οποίος είναι ανεξάρτητος του πού βρίσκεται η κάθε διεργασία και, συνεπώς, κρύβει την ακριβή τοποθέτησή τους μέσα στο ΔΔΤ. Εδώ θα πρέπει να τονίσουμε ότι, αντίθετα με τα γνωστά μοντέλα διαχείρισης (όπως αυτό του *Internet* [CFSD90]) όπου οι αντιπρόσωποι διαχείρισης για κάποιο κόμβο του δικτύου βρίσκονται σε γνωστή διεύθυνση (δηλ. στον ίδιο το κόμβο!), στο ΔΔΤ – όπως θα δούμε στο πρώτο κεφάλαιο – η περίπτωση αυτή είναι μάλλον η εξαίρεση παρά ο κανόνας.
Τέλος, στηριζόμενοι στο μοντέλο της καθολικής πληροφορίας διαχείρισης και στο μηχανισμό της διαφάνειας ως προς την τοποθεσία, περιγράφουμε την υπηρεσία του καθολικού

[4]Στο εξής, όποτε αναφερόμαστε σε τέτοια παγκοσμίως μοναδικά ονόματα θα χρησιμοποιούμε τον όρο "καθολικά ονόματα" (ο οποίος αποτελεί ελεύθερη μετάφραση του αγγλικού "global names").

CMIS/CMIP. Η υπηρεσία αυτή είναι όμοια, όσον αφορά το πρωτόκολλο και το συντακτικό των μηνυμάτων που χρησιμοποιεί, με το κλασσικό CMIS/CMIP του οποίου αποτελεί μία χρήσιμη επέκταση. Η διαφορά είναι ότι κατά την επικοινωνία διαχειριστή/αντιπροσώπου αποκρύπτεται το στάδιο δημιουργίας της σύνδεσης διαχειριστή/αντιπροσώπου χάρη στη χρήση καθολικών ονομάτων για τον προσδιορισμό της οποιαδήποτε πληροφορίας διαχείρισης στα μηνύματα που ανταλλάσσονται. Η χρήση των καθολικών ονομάτων έχει σαν αποτέλεσμα την δημιουργία της σύνδεσης διαχειριστή/αντιπροσώπου με τρόπο αθέατο στο χρήστη. Ο τελευταίος έχει την ψευδαίσθηση ότι επικοινωνεί πάντα με ένα μοναδικό κεντρικό αντιπρόσωπο ο οποίος κρατάει όλη την πληροφορία διαχείρισης που περιέχεται στο ΔΔΤ.
Ολα τα παραπάνω – με εξαίρεση το καθολικό CMIS/CMIP – υλοποιήθηκαν και ένα σημαντικό μέρος τους αποτελεί τμήμα του πακέτου λογισμικού *OSIMIS* [PMBK95] το οποίο είναι σήμερα η ευρύτερα διαδεδομένη – μεταξύ αυτών που διατίθονται δωρεάν – πλατφόρμα ανάπτυξης συστημάτων που χρησιμοποιούν τα μοντέλα και τα πρωτόκολλα του *ISO* για τη διαχείριση δικτύων.
Πρέπει να σημειωθεί ότι όλοι οι παραπάνω μηχανισμοί μπορούν να λειτουργήσουν και να εφαρμοστούν όχι μόνο στο ΔΔΤ της *ITU* αλλά και σε οποιοδήποτε άλλο δίκτυο διαχείρισης. Ο λόγος που διαλέξαμε το ΔΔΤ σαν σημείο αναφοράς είναι ότι, μέχρι σήμερα, η αρχιτεκτονική του αποτελεί την πιο ευέλικτη, γενικότερη και πληρέστερη προσέγγιση στο πρόβλημα της διαχείρισης των δικτύων τηλεπικοινωνιών.

# Σχέση με άλλες ερευνητικές εργασίες

Τα τελευταία δύο χρόνια υπάρχει αρκετή ερευνητική δραστηριότητα σε θέματα που αφορούν στο ΔΔΤ της *ITU* ενώ ελάχιστες είναι οι εργασίες που ασχολήθηκαν με τον προσδιορισμό, την αποθήκευση ή και την απόκτηση της ΚΓΔ. Η αρχική ιδέα της χρήσης του X.500 για τη μοναδική ονομασία των διεργασιών διαχείρισης αναφέρεται στις συστάσεις του *ISO* [Org92b], [Org95] ενώ μια παρόμοια άλλα αξιοσημείωτη άποψη περιγράφεται στην [Syl93].
Μόλις πριν από ένα χρόνο και κατά τη διάρκεια εκπόνησης της εργασίας αυτής, προτάθηκε από τον *ISO* ένας τρόπος περιγραφής και απόκτησης της ΚΓΔ [Org95]. Η παρούσα εργασία όχι μόνο διατηρεί συμβατότητα με την παραπάνω τυποποίηση αλλά αποτελεί και επέκταση της συμπεριλαμβάνοντας νέα στοιχεία στην ΚΓΔ τα οποία θεωρήθηκαν απαραίτητα σε ένα γενικό περιβάλλον διαχείρισης όπως το ΔΔΤ. Οπως θα δείξουμε στη συνέχεια, κάτι το οποίο αποτελεί το βασικό στόχο της εργασίας μας, οι επεκτάσεις αυτές οδηγούν σε μια πληρέστερη

μέθοδο απόκτησης της ΚΓΔ. Πρέπει να σημειωθεί ότι η παρούσα εργασία αποτέλεσε και πηγή σχολίων τα οποία στάλθηκαν στην επιτροπή του *ISO* που είναι υπεύθυνη για την τελική προώθηση της ISO-10164-16 σε διεθνή τυποποίηση (International Standard) κάτι το οποίο πιθανόν να γίνει μέσα στο 1996.

Η διαφάνεια ως προς την τοποθεσία στα κλασσικά κατανεμημένα συστήματα διασφαλίζεται με τη βοήθεια ειδικών διεργασιών που καλούνται "έμποροι" (traders), η ακριβής λειτουργία των οποίων περιγράφεται στις τυποποιήσεις του *ISO*, τις σχετικές με το μοντέλο της *Ανοιχτής Κατανεμημένης Επεξεργασίας* (*Open Distributed Processing – ODP*) [Org94a]. Στα πλαίσια του *ODP*, ένας έμπορος (trader) είναι μία διεργασία που φυλάσσει πληροφορία η οποία αποτελεί κάποια κοινή γνώση που απαιτείται για τη λειτουργία ενός κατανεμημένου συστήματος και η οποία, προφανώς, σχετίζεται με τις διάφορες διεργασίες του κατανεμημένου συστήματος. Την πληροφορία αυτή ο "έμπορος" τη διαθέτει σε κάθε διεργασία όποτε χρειαστεί ενώ υπάρχει η δυνατότητα της ενημέρωσης της πληροφορίας που ήδη κρατάει. Η υλοποίηση ενός συστήματος "εμπόρων" προϋποθέτει μία κατανεμημένη βάση πληροφορίας όπως αυτή του X.500 καθώς και ένα τρόπο επικοινωνίας μεταξύ των "εμπόρων". Ο ίδιος ο *ISO* έχει προτείνει τη χρήση του X.500 και σε αυτήν την περίπτωση. Τα πλεονεκτήματα ενός συστήματος "εμπόρων" βρίσκονται στην καλύτερη οργάνωση, έλεγχο λειτουργίας και αυτονομία του συστήματος το οποίο είναι υπεύθυνο για την ενημέρωση και διάθεση κάποιας απαιτούμενης κοινής γνώσης σε ένα κατανεμημένο περιβάλλον. Το βασικό μειονέκτημα βρίσκεται στην καθυστέρηση που εισάγει η επικοινωνία των διεργασιών με τους "εμπόρους". Στην πραγματικότητα, το μοντέλο διαχειριστή/αντιπροσώπου που προτείνεται στην παρούσα εργασία συμπεριλαμβάνει τη λειτουργία του "εμπόρου" σε κάθε διεργασία. Επειδή ο βασικός μας στόχος ήταν ο σχεδιασμός ενός συστήματος το οποίο θα χρησιμοποιεί όσο το δυνατό λιγότερα μοντέλα και σύνολα πρωτοκόλλων (στην προκειμένη περίπτωση τα X.500 και X.700[5]) και επειδή η αποτελεσματικότητα και απόδοση της υλοποίησης κρίθηκε ικανοποιητική για τους σκοπούς μας, δεν θεωρήσαμε σκόπιμο να χρησιμοποιήσουμε τις τεχνικές που αναφέρονται στο μοντέλο του *ODP*. Στο τέλος της παρούσας αναφοράς συζητάμε μελλοντικές χρήσεις του *ODP* στα περιβάλλοντα διαχείρισης δικτύων.

Στην [TD93] προτείνεται ένας τρόπος ενοποίησης των δύο μοντέλων πληροφορίας και επικοινωνίας (X.500 και X.700). Εκεί περιγράφεται η καθολική βάση πληροφορίας

[5] Ο όρος "X.700" αναφέρεται στο σύνολο των πρωτοκόλλων, λειτουργιών και μοντέλων που έχουν προταθεί για τη διαχείριση συστημάτων. Οπως συμβαίνει και στην περίπτωση του X.500 τα παραπάνω αναφέρονται σε συστάσεις τις οποίες έχουν προτείνει από κοινού ο *ISO* και η *ITU*.

διαχείρισης και παρέχεται ένα ανώτερο επίπεδο λογισμικού το οποίο προσφέρει ένα ενιαίο πρωτόκολλο για την επικοινωνία τόσο με τον Κατάλογο όσο και με τις διεργασίες διαχείρισης. Ο σκοπός του συστήματος της [TD93] είναι η διαχείριση σε ένα κατανεμημένο περιβάλλον όπως το ΔΔΤ και η ταυτόχρονη χρήση του X.500 για την απόκτηση της ΚΓΔ με τρόπο που να κρύβεται τόσο η κατανομή της πληροφορίας στις διεργασίες όσο και το είδος της πληροφορίας που προσπελαύνεται κάθε φορά· με άλλα λόγια ο χρήστης δεν διαχωρίζει κατά πόσο η πληροφορία που προσπελαύνει είναι πληροφορία διαχείρισης ή πληροφορία του Καταλόγου. Η παρεχόμενη υπηρεσία είναι της ίδιας φύσης με την υπηρεσία του καθολικού CMIS/CMIP που περιγράφουμε αλλά κατά πολύ γενικότερη. Εισάγει δε ένα επιπλέον επίπεδο λογισμικού το οποίο παρέχει την ενοποίηση δύο υπηρεσιών: του X.500 και του CMIS/CMIP. Με αυτόν τον τρόπο παρέχεται τελικά μία νέα υπηρεσία υψηλότερου επιπέδου η οποία στην [TD93] καλείται IDMIS (Inter-Domain Management Information Service) αλλά που όμως δεν είναι συμβατή με την κλασσική CMIS/CMIP. Στην εργασία μας το καθολικό CMIS/CMIP αποτελεί απλά μία εφαρμογή που δείχνει πώς η αίσθηση της καθολικής βάσης πληροφορίας διαχείρισης που πετυχαίνεται με τη χρήση της υπηρεσίας καταλόγου σε συνδυασμό με τον μηχανισμό διαφάνειας ως προς την τοποθεσία που υποστηρίζουμε μπορεί να δώσει μία υπηρεσία η οποία είναι συμβατή με την κλασσική CMIS/CMIP με τη διαφορά ότι προσφέρει έναν αυτόματο και αόρατο μηχανισμό δημιουργίας των συνδέσεων κατά την επικοινωνία διαχειριστή/αντιπροσώπου. Ξεκινάμε την παρούσα αναφορά, κάνοντας μια σύντομη περιγραφή των μοντέλων που αναφέραμε παραπάνω. Συγκεκριμένα, το πρώτο κεφάλαιο περιγράφει τα μοντέλα διαχειριστή/αντιπροσώπου και πληροφορίας διαχείρισης του *ISO*. Επίσης, περιγράφεται η αρχιτεκτονική και λειτουργία του ΔΔΤ και της υπηρεσίας καταλόγου της *ITU*. Στο τελευταίο μέρος του κεφαλαίου 1 αναφερόμαστε λεπτομερώς στην ΚΓΔ όπως αυτή περιγράφεται από τον *ISO* και στα επιπλέον στοιχεία τα οποία εμείς θεωρήσαμε ότι πρέπει να περιλαμβάνει. Το κεφάλαιο 2 αποτελεί το κεντρικό μέρος της εργασίας. Περιγράφει την επέκταση που κάνουμε στο μοντέλο διαχειριστή/αντιπροσώπου, την αυτόματη ενημέρωση της ΚΓΔ από τις διεργασίες κατά την έναρξη της λειτουργίας τους και τον τρόπο με τον οποίο πραγματοποιείται στο ΔΔΤ η απόκτηση της ΚΓΔ χρησιμοποιώντας το νέο μοντέλο επικοινωνίας. Επίσης, περιγράφεται η λειτουργία του μηχανισμού διαφάνειας ως προς την τοποθεσία καθώς και του προτεινόμενου καθολικού CMIS/CMIS. Στη συνέχεια, δείχνουμε μια εφαρμογή των παραπάνω σε ένα απλό ΔΔΤ. Στο επόμενο κεφάλαιο παραθέτουμε τις λεπτομέρειες υλοποίησης των παραπάνω μηχανισμών στο πακέτο *OSIMIS*. Η υλοποίηση έγινε χρησιμοποιώντας το πακέτο *QUIPU* [Kil91] το οποίο περιέχει τις αναγκαίες λειτουργίες και βιβλιοθήκες λογισμικού για την παροχή και χρήση της υπηρεσίας καταλόγου. Ιδιαίτερη έμφαση δίνεται στη δομή των νέων οντοτήτων του X.500 που ορίστηκαν για τους σκοπούς αυτής της εργασίας καθώς και στη θέση της υλοποίησης μας στην αρχιτεκτονική του *OSIMIS*. Τέλος, στο κεφάλαιο 4, συζητάμε τις μελλοντικές χρήσεις και πιθανές συμπληρώσεις/επεκτάσεις στους παραπάνω μηχανισμούς.

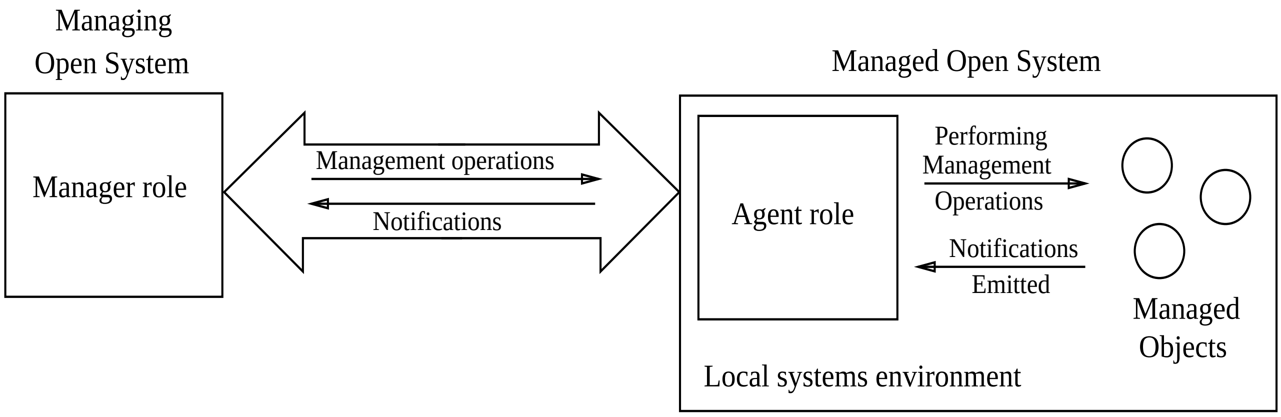
# Κεφάλαιο 1

# Σύντομη Ανασκόπηση της Περιοχής

Στο κεφάλαιο αυτό δίνεται μια σύντομη περιγραφή των μοντέλων που αποτελούν το υπόβαθρο της παρούσας εργασίας και χωρίζεται σε τέσσερα τμήματα. Το πρώτο αναφέρεται στο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου καθώς και στη δομή της πληροφορίας διαχείρισης όπως αυτά έχουν προταθεί από τον *ISO*. Επίσης, περιλαμβάνει μια σύντομη περιγραφή του πρωτοκόλλου *CMIP*. Στο δεύτερο τμήμα παρουσιάζονται συνοπτικά η αρχιτεκτονική του Δικτύου Διαχείρισης Τηλεπικοινωνιών (ΔΔΤ) της *ITU*, η μεθοδολογία που ακολουθεί το ΔΔΤ στη διαχείριση συστημάτων και υπηρεσιών, οι λειτουργίες και οι ιδαιτερότητές του. Στη συνέχεια, γίνεται μια περιληπτική περιγραφή της Υπηρεσίας Καταλόγου (X.500), των πρωτοκόλλων επικοινωνίας και του μοντέλου πληροφορίας που χρησιμοποιεί καθώς και των χαρακτηριστικών και απαιτήσεων της. Τέλος, στο τέταρτο και τελευταίο τμήμα του κεφαλαίου, ορίζεται και περιγράφεται, βάσει των συστάσεων του *ISO*, η Κοινή Γνώση Διαχείρισης (ΚΓΔ) ενώ προσδιορίζουμε τα επιπλέον στοιχεία που πρέπει να περιλαμβάνει σε ένα διασυνδεδεμένο σύστημα διαχείρισης και ειδικότερα στο ΔΔΤ.

## 1.1 Το πλαίσιο διαχείρισης συστημάτων του ISO

Η τυποποίηση ISO/IEC 10040 [Org92a] συνοψίζει το βασικό πλαίσιο διαχείρισης συστημάτων που έχει προταθεί από τον *ISO*. Αυτό περιλαμβάνει τόσο το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου (manager/agent model of interaction) όσο και το μοντέλο της πληροφορίας διαχείρισης (management information model). Η ενότητα αυτή αποτελεί μια σύντομη περιγραφή των παραπάνω ενώ γίνεται και μια συνοπτική αναφορά στο πρωτόκολλο επικοινωνίας (CMIP) και στην υπηρεσία που αυτό παρέχει (CMIS) για τη λειτουργική χρήση




Σχήμα 1.1: Το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου του *ISO*.

των μοντέλων αυτών.

### 1.1.1 Το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου

Στο σχήμα 1.1 απεικονίζεται το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου. Δυο ανοιχτά συστήματα[1], από τα οποία το ένα είναι το διαχειρίζον (managing) και το άλλο το διαχειριζόμενο (managed), επικοινωνούν όπως περιγράφεται παρακάτω.

Το διαχειρίζον σύστημα περιέχει μία διεργασία σε επίπεδο εφαρμογής[2] (application layer) η οποία λειτουργεί σε ρόλο διαχειριστή (manager role) ενώ το διαχειριζόμενο περιλαμβάνει μια διεργασία η οποία λειτουργεί σε ρόλο αντιπροσώπου διαχείρισης (agent role) και έναν αριθμό από διαχειρίσιμες οντότητες (managed objects). Κάθε διαχειρίσιμη οντότητα (ΔΟ) περιέχει πληροφορία για κάποιο πόρο (φυσικό ή λογικό) του δικτύου ο οποίος υπόκειται σε διαχείριση όπως, για παράδειγμα, ένα συνδέσμο (link), ένα πίνακα δρομολόγησης, ένα υποδίκτυο ή κάποια προσφερόμενη υπηρεσία. Με άλλα λόγια, αποτελεί μία αφαίρεση κάποιου στοιχείου

[1] Ο όρος "ανοιχτά συστήματα" αποτελεί μετάφραση του αγγλικού "*open systems*" ο οποίος χρησιμοποιείται από τον *ISO* και αναφέρεται σε συστήματα τα οποία επικοινωνούν με κάποιο πρωτόκολλο το οποίο ακολουθεί το μοντέλο αναφοράς για τη διασύνδεση ανοιχτών συστημάτων (Open Systems Interconnection Reference Model – OSIRM) [Org94b] του *ISO*.

[2] Εδώ ο όρος "διεργασία σε επίπεδο εφαρμογής" είναι ισοδύναμος με τον όρο "διεργασία διαχείρισης" που χρησιμοποιήσαμε στην εισαγωγή. Τις περισσότερες φορές θα χρησιμοποιούμε το δεύτερο ενώ για συντομία θα αναφερόμαστε απλά σε "διεργασίες".

του δικτύου από τη σκοπιά (και για τους σκοπούς) της διαχείρισης.
Μια σύνδεση μεταξύ των δύο συστημάτων του σχήματος 1.1 μπορεί να δημιουργηθεί με δύο τρόπους. Είτε από το διαχειριστή ο οποίος θέλει να εκτελεστεί κάποια πράξη διαχείρισης (management operation) σε κάποια ΔΟ είτε από τον αντιπρόσωπο ο οποίος πρέπει να στείλει μία ειδοποίηση (notification) η οποία προέρχεται από κάποια ΔΟ. Ετσι, ο ρόλος του διαχειριστή περιλαμβάνει τη δυνατότητα τόσο της πυροδότησης πράξεων διαχείρισης όσο και της παραλαβής ειδοποιήσεων. Από την άλλη, ο αντιπρόσωπος[3] εκτελεί τις πράξεις που λαμβάνει από το διαχειριστή για κάποια (ή κάποιες) ΔΟ και προωθεί τις ειδοποιήσεις που εκπέμπουν οι διαχειρίσιμες οντότητες προς τους κατάλληλους διαχειριστές.
Για παράδειγμα, ας θεωρήσουμε μία ΔΟ η οποία περιέχει τα στατιστικά του φόρτου ενός συνδέσμου σε ένα μεταγωγέα (switch) του δικτύου τα οποία περιλαμβάνουν αριθμό εισερχομένων πακέτων στο σύνδεσμο, αριθμό λαθών, αριθμό bits που αντιστοιχεί στα εισερχόμενα πακέτα κ.ά. Ενας διαχειριστής μπορεί να ζητήσει τον αριθμό των πακέτων που εξέρχονται από το σύνδεσμο του μεταγωγέα αποστέλλοντας στον αντιπρόσωπο που περιέχει την παραπάνω ΔΟ μία κατάλληλη πράξη διαχείρισης για την ΔΟ[4]. Επιπλέον, όταν ο αριθμός των λαθών – π.χ. σε εισερχόμενα πακέτα – υπερβεί κάποια προκαθορισμένη τιμή, η ΔΟ μπορεί να εκδόσει μια ειδοποίηση[5] γνωστοποιώντας άμεσα το γεγονός αυτό. Σε κάθε μία από τις παραπάνω περιπτώσεις, ο αντιπρόσωπος του διαχειριζόμενου συστήματος που περιέχει την εν λόγω ΔΟ, αναλαμβάνει να στείλει τα ζητούμενα στατιστικά καθώς και την ειδοποίηση στο διαχειριστή. Η διαφορά είναι ότι στην πρώτη περίπτωση η σύνδεση διαχειριστή/αντιπροσώπου ενεργοποιείται από το διαχειριστή (ο οποίος στέλνει μία πράξη διαχείρισης) ενώ στη δεύτερη από τον αντιπρόσωπο (ο οποίος προωθεί μία ειδοποίηση).
Μία ΔΟ εκτός από την πληροφορία που περιέχει χαρακτηρίζεται και από τη συμπεριφορά (behaviour) της. Η συμπεριφορά μίας ΔΟ είναι άμεσα συνδεδεμένη με τον τρόπο με τον οποίο αντανακλώνται οι διάφορες πράξεις διαχείρισης – που εκτελούνται πάνω στην ΔΟ – στους πραγματικούς – φυσικούς ή λογικούς – πόρους και υπηρεσίες του διαχειριζόμενου δικτύου. Στο προηγούμενο παράδειγμα η ΔΟ που περιέχει τα στατιστικά του φόρτου ενός συνδέσμου έχει την απλή συμπεριφορά της διάθεσης κάποιων στατιστικών στους ενδιαφερόμενους

[3] Ο όρος "αντιπρόσωπος" χρησιμοποιήθηκε για την απόδοση του αγγλικού "*agent*" επειδή, όπως γίνεται φανερό από τα παραπάνω, στην πραγματικότητα ο ρόλος του είναι να αντιπροσωπεύει τους πόρους του δικτύου μέσω των ΔΟ που βρίσκονται υπό τον έλεγχό του.

[4] Οπως θα δούμε στην §1.1.3 όπου περιγράφεται το πρωτόκολλο *CMIS/CMIP*, η εν λόγω πράξη θα αντιστοιχεί σε ένα μήνυμα του τύπου *GET*.

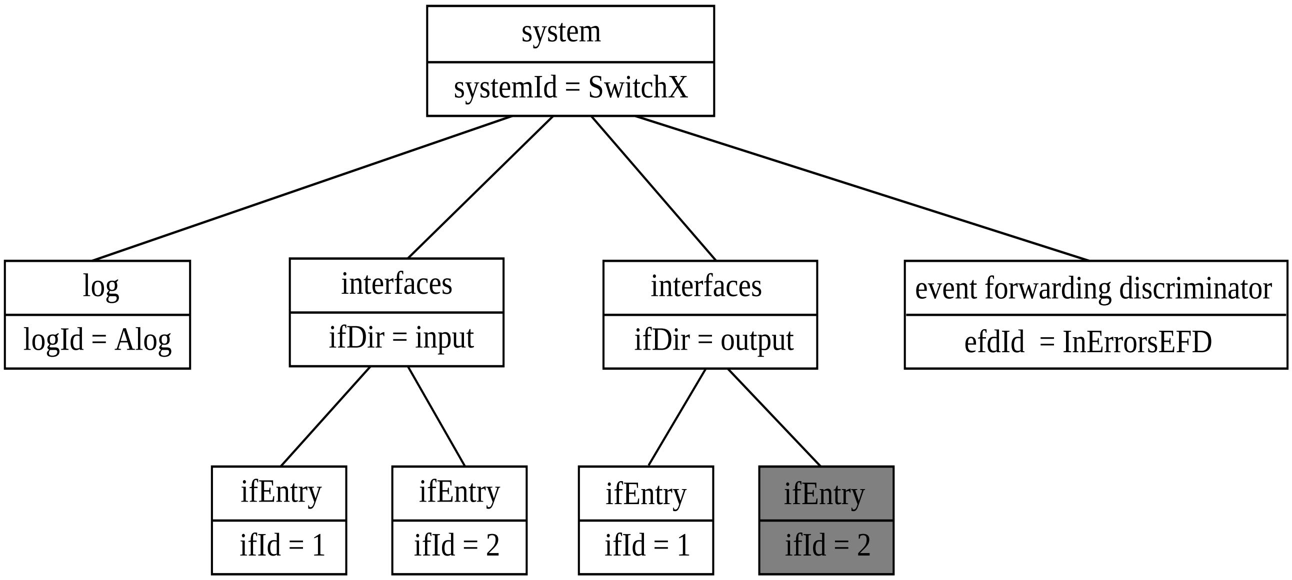
[5] Στέλνοντας ένα μήνυμα του τύπου *EVENT-REPORT* όπως θα δούμε επίσης στην §1.1.3.

διαχειριστές. Προφανώς, η διάθεση των στατιστικών αυτών αφήνει ανεπηρέαστο το σύνδεσμο, η λειτουργία του οποίου απλά παρατηρείται αλλά δεν ελέγχεται από την ΔΟ. Είναι όμως πιθανό μία ΔΟ να δίνει επιπλέον δυνατότητες στο διαχειριστή. Ας υποθέσουμε ότι η παραπάνω ΔΟ περιέχει και την πληροφορία τού αν ο σύνδεσμος λειτουργεί ή όχι. Αν αυτή η πληροφορία αλλαχτεί από κάποιο διαχειριστή τότε είναι δυνατό ο σύνδεσμος να απενεργοποιηθεί εφόσον αυτό περιλαμβάνεται στη συμπεριφορά της ΔΟ (δηλαδή, αν και μόνο αν η ΔΟ αντιδρά στην αλλαγή της πληροφορίας της προκαλώντας κάποια αλλαγή στον πραγματικό πόρο με τον οποίο σχετίζεται).

### 1.1.2 Το μοντέλο της πληροφορίας διαχείρισης

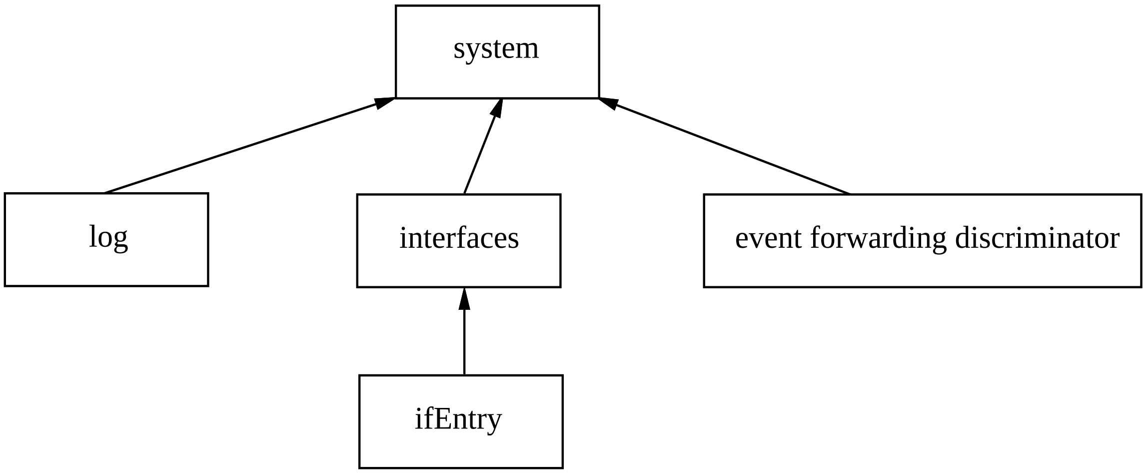
Οι διαχειρίσιμες οντότητες ανήκουν σε κλάσεις. Κάθε κλάση διαχειρίσιμων οντοτήτων (managed object class) ορίζει τις πράξεις τις οποίες επιδέχονται και τις ειδοποιήσεις που μπορούν να εκδόσουν οι οντότητες που ανήκουν σε αυτή καθώς και τις σχέσεις τους με άλλες διαχειρίσιμες οντότητες. Κάθε κλάση διαχειρίσιμων οντοτήτων περιλαμβάνει τις ΔΟ εκείνες που έχουν μία κοινή συμπεριφορά και σχετίζονται με έναν κοινό είδος πόρου (ή υπηρεσίας). Για παράδειγμα, η κλάση *ifEntry* περιλαμβάνει τις ΔΟ που παρέχουν παρακολούθηση και έλεγχο ενός συνδέσμου (interface) του δικτύου μέσω ορισμένων πράξεων διαχείρισης και ειδοποιήσεων. Επιπλέον, κάθε κλάση ορίζει το πλήθος και το είδος των χαρακτηριστικών γνωρισμάτων (attributes) – ή, απλά, χαρακτηριστικών – που ενέχουν οι διαχειρίσιμες οντότητες που ανήκουν σε αυτή. Τα χαρακτηριστικά αυτά αναφέρονται σε καταστάσεις, παραμέτρους και γνωρίσματα του πόρου (ή της υπηρεσίας) με τον οποίο σχετίζεται η ΔΟ. Για παράδειγμα, η ΔΟ για ένα σύνδεσμο ενός μεταγωγέα μπορεί να περιέχει χαρακτηριστικά όπως ο αριθμός των πακέτων που εξέρχονται από το σύνδεσμο, ο αριθμός των πακέτων που φτάνουν με κάποιο λάθος και η κατάσταση λειτουργίας του συνδέσμου. Κάθε χαρακτηριστικό έχει και μια τιμή με την οποία σχηματίζει ένα ζευγάρι χαρακτηριστικού-τιμής. Το ζευγάρι αυτό συνιστά το ελάχιστο κομμάτι της πληροφορίας διαχείρισης.

Το σύνολο των διαχειρίσιμων οντοτήτων που περιέχονται σε έναν αντιπρόσωπο, αποτελεί τη βάση πληροφορίας διαχείρισης (management information base – MIB) του αντιπροσώπου. Η MIB δομείται ιεραρχικά σχηματίζοντας ένα ιδεατό δέντρο, όπως αυτό του σχήματος 1.2. Κάθε κόμβος αντιστοιχεί σε μία ΔΟ ενώ η σχέση των διαχειρίσιμων οντοτήτων υποδηλώνεται από τη σχετική θέση τους στο δέντρο. Στο σχήμα 1.2 η κλάση στην οποία ανήκει κάθε ΔΟ αναγράφεται στην πάνω γραμμή κάθε κόμβου. Για παράδειγμα, η ρίζα της εικονιζόμενης MIB




Σχήμα 1.2: Μία απλή υποθετική MIB για ένα μεταγωγέα (switchX).

ανήκει στην – τυποποιημένη – κλάση *system* η οποία περιέχει χαρακτηριστικά σχετικά με το διαχειριζόμενο σύστημα που αφορά η πληροφορία της MIB. Τέτοια χαρακτηριστικά μπορεί να είναι το όνομα του συστήματος, το χρονικό διάστημα συνεχούς λειτουργίας του, το είδος του υλικού (hardware) και λογισμικού (software) που υποστηρίζει και το πού βρίσκεται.
Για κάθε ΔΟ, ένα χαρακτηριστικό, το οποίο καλείται *χαρακτηριστικό ονομασίας* (naming attribute), χρησιμοποιείται μαζί με την τιμή του για το σχηματισμό του ονόματος της ΔΟ. Αυτό το ζευγάρι χαρακτηριστικού-τιμής πρέπει να είναι μοναδικό ανάμεσα στις διαχειρίσιμες οντότητες με τον ίδιο πατέρα στο δέντρο της MIB, καλείται δε *σχετικά διακεκριμένο όνομα* (relative distinguished name – RDN) της διαχειρίσιμης οντότητας και αποτελεί ένα μέρος του ονόματος της. Στο σχήμα 1.2 το RDN κάθε ΔΟ αναγράφεται στην κάτω γραμμή κάθε κόμβου. Για καθαρά πρακτικούς λόγους παραλείψαμε τα υπόλοιπα ζευγάρια χαρακτηριστικού-τιμής τα οποία ενέχουν οι ΔΟ του σχήματος 1.2.
Η συνένωση όλων των RDN των κόμβων στο μονοπάτι από την ρίζα του δέντρου της πληροφορίας διαχείρισης μέχρι τον κόμβο που αντιστοιχεί στην εν λόγω ΔΟ δίνει στην τελευταία ένα όνομα μοναδικό μέσα στην MIB. Σαν ένα παράδειγμα, ας θεωρήσουμε τη γραμμοσκιασμένη ΔΟ του σχήματος 1.2. Αυτή ανήκει στην κλάση *ifEntry* που – στο




Σχήμα 1.3: Το δέντρο ονομασίας (naming tree) μίας MIB.

παράδειγμα μας – σημαίνει ότι αντιπροσωπεύει μία διασύνδεση (interface) του μεταγωγέα στο δίκτυο. Το σχετικά διακεκριμένο όνομα (RDN) της είναι $\{\textbf{ifId} = \textbf{2}\}$ και είναι μοναδικό μεταξύ των διαχειρίσιμων οντοτήτων που έχουν τον ίδιο πατέρα (δηλ. την οντότητα με RDN $\{\textbf{ifDir} = \textbf{output}\}$). Το όνομα με το οποίο αναγνωρίζεται μοναδικά η γραμμοσκιασμένη οντότητα μέσα στην MIB είναι:
$\{\textbf{systemId} = \textbf{SwitchX}, \textbf{ifDir} = \textbf{output}, \textbf{ifId} = \textbf{2}\}$.
Προαναφέραμε πως η σχέση των διαχειρίσιμων οντοτήτων υποδηλώνεται από τη σχετική θέση τους στο δέντρο της MIB. Θεωρώντας πάλι το σχήμα 1.2, η σχέση των οντοτήτων με RDN $\{\textbf{ifId} = \textbf{1}\}$ και $\{\textbf{ifId} = \textbf{2}\}$ υποδηλώνεται από το γεγονός ότι οι οντότητες αυτές έχουν κοινό πατέρα την ΔΟ με RDN $\{\textbf{ifDir} = \textbf{output}\}$. Η τελευταία ΔΟ αντιπροσωπεύει τα κοινά χαρακτηριστικά όλων των εξόδων του μεταγωγέα. Αρα, γίνεται φανερό ότι οι προηγούμενες δύο διαχειρίσιμες οντότητες αντιπροσωπεύουν δύο διασυνδέσεις του μεταγωγέα και ειδικότερα δύο εξόδους του.
Εδώ πρέπει να προσθέσουμε ότι σε κάθε MIB ισχύουν ορισμένοι κανόνες ονομασίας που καθορίζουν σε ποιες κλάσεις οντοτήτων μπορεί να ανήκουν οι διαχειρίσιμες οντότητες που έχουν πατέρα μία ΔΟ η οποία ανήκει σε κάποια συγκεκριμένη κλάση. Αυτοί οι κανόνες ονομασίας τίθονται κατά τον ορισμό των διαφόρων κλάσεων: κάθε κλάση είναι υποκλάση

κάποιας άλλης[6] σχηματίζοντας έτσι ένα ιδεατό δέντρο το οποίο καλείται *δέντρο ονομασίας* (naming tree) και το οποίο ορίζει μία ιεραρχία στις κλάσεις των διαχειρίσιμων οντοτήτων. Το σχήμα 1.3 απεικονίζει το δέντρο ονομασίας το οποίο αντιστοιχεί στην MIB του σχήματος 1.2. Η δομή οποιασδήποτε MIB μπορεί να περιγραφεί από το δέντρο ονομασίας της. Φυσικά, το δέντρο αυτό δεν δίνει πληροφορία για το ποιες διαχειρίσιμες οντότητες περιέχονται στην MIB ούτε για το πλήθος τους. Δείχνει όμως τα είδη των οντοτήτων που περιέχει η MIB και τη σχετική τους θέση. Τα βέλη των ακμών του δέντρου του σχήματος 1.3 δηλώνουν μία σχέση εγκλεισμού μεταξύ των κλάσεων καθώς και μία αντιστοιχία 1 προς N μεταξύ των διαχειρίσιμων οντοτήτων οι οποίες ανήκουν στις αντίστοιχες κλάσεις. Ετσι, κάθε ΔΟ η οποία ανήκει στην κλάση *system* μπορεί να περιέχει πολλές διαχειρίσιμες οντότητες οι οποίες ανήκουν στην κλάση *interfaces* και, παρόμοια, μία ΔΟ η οποία ανήκει στην κλάση *interfaces* μπορεί να περιέχει πολλές διαχειρίσιμες οντότητες οι οποίες ανήκουν στην κλάση *ifEntry*. Τα παραπάνω αντανακλούν την προφανή σχέση στα στοιχεία του πραγματικού κόσμου που αντιπροσωπεύουν οι διαχειρίσιμες οντότητες – π.χ. ένα σύστημα όντως περιέχει πολλές διασυνδέσεις· επιπλέον, θέτουν έμμεσα τους κανόνες για την ονομασία των διαχειρίσιμων οντοτήτων. Ετσι, το όνομα κάθε οντότητας που ανήκει στην κλάση *ifEntry* θα σχηματίζεται με την προσθήκη του RDN της στο όνομα κάποιας οντότητας της κλάσης *interfaces*. Για περισσότερες λεπτομέρειες σχετικά με το μοντέλο και την οργάνωση της πληροφορίας διαχείρισης ο αναγνώστης παραπέμπεται στις τυποποιήσεις [Org92b][Org92c].

### 1.1.3 Το πρωτόκολλο CMIP και η υπηρεσία CMIS

Τα μοντέλα διαχειριστή/αντιπροσώπου και της πληροφορίας διαχείρισης που περιγράφηκαν απαιτούν την ύπαρξη ενός καλώς ορισμένου πρωτοκόλλου το οποίο θα παρέχει τη δυνατότητα όχι μόνο πρόσβασης αλλά και επέμβασης στην πληροφορία διαχείρισης. Το πρωτόκολλο αυτό έχει οριστεί από τον *ISO*, ονομάζεται CMIP (Common Management Information Protocol) [Org90a] και περιγράφει αναλυτικά όλα τα μηνύματα που πρέπει να ανταλλάσσονται κατά την επικοινωνία διαχειριστή/αντιπροσώπου. Οι βασικοί τύποι μηνυμάτων είναι οι παρακάτω.

- **EVENT-REPORT**: Τα μηνύματα αυτά αντιστοιχούν σε ειδοποιήσεις οι οποίες εκπέμπονται απο κάποια ΔΟ και προωθούνται από τον αντιπρόσωπο στον διαχειριστή. Αντίθετα, όλοι οι επόμενοι τύποι μηνυμάτων αντιστοιχούν σε αιτήσεις για εκτελέσεις

[6] Η υψηλότερη κλάση (στο παράδειγμα μας η *system*) αποτελεί τη μοναδική εξαίρεση δηλαδή, δεν είναι υποκλάση καμίας κλάσης.

πράξεων διαχείρισης οι οποίες στέλνονται από τον διαχειριστή προς τον αντιπρόσωπο.

- **GET**: Τα μηνύματα αυτά αποτελούν αιτήσεις για πρόσβαση και ανάκτηση των τιμών μιας λίστας χαρακτηριστικών (attributes) από κάποιο σύνολο διαχειρίσιμων οντοτήτων.
- **SET**: Τα μηνύματα αυτού του τύπου αντιστοιχούν σε αιτήσεις για αλλαγή των τιμών μιας λίστας χαρακτηριστικών σε κάποιο σύνολο διαχειρίσιμων οντοτήτων.
- **ACTION**: Τα μηνύματα αυτά πυροδοτούν την εκτέλεση συγκεκριμένων ενεργειών (actions) σε κάποιο σύνολο διαχειρίσιμων οντοτήτων. Ενας τρόπος να αντιλαμβανόμαστε τις ενέργειες αυτές είναι σαν τηλε-εκτέλεση διεργασιών (remote process execution) στο διαχειριζόμενο σύστημα.
- **CREATE**: Αντιστοιχούν σε αιτήσεις για τη δημιουργία ενός συνόλου διαχειρίσιμων οντοτήτων.
- **DELETE**: Αντιστοιχούν σε αιτήσεις για τη διαγραφή ενός συνόλου διαχειρίσιμων οντοτήτων.

Εκτός από τα μηνύματα του τύπου **EVENT-REPORT** που δηλώνουν επικοινωνία από τον αντιπρόσωπο προς το διαχειριστή και αναφέρονται πάντα σε κάποια μοναδική ΔΟ, τα παραπάνω μηνύματα αντιστοιχούν σε πράξεις διαχείρισης (management operations) οι οποίες, γενικά, εκτελούνται πάνω σε ένα σύνολο από διαχειρίσιμες οντότητες. Αυτό το σύνολο επιλέγεται εκτελώντας μία αναζήτηση στην MIB του αντιπροσώπου βάσει δύο μηχανισμών (οι οποίοι επιτυγχάνονται με εισαγωγή ειδικών παραμέτρων στα παραπάνω μηνύματα):

1. Δίνοντας την απόσταση (scoping) μέσα στο ιδεατό δέντρο της MIB στην οποία πρέπει να αναζητηθούν οι διαχειρίσιμες οντότητες. Η απόσταση αυτή αναφέρεται στο μήκος του μονοπατιού στο δέντρο της MIB και μετριέται σχετικά με κάποια ΔΟ η οποία χρησιμεύει σαν αρχή για την παραπάνω αναζήτηση. Να σημειωθεί ότι η αναζήτηση γίνεται για οντότητες που ανήκουν σε μια ζητούμενη κλάση.

   Για παράδειγμα, ας θεωρήσουμε έναν αντιπρόσωπο ο οποίος περιέχει την MIB του σχήματος 1.2. Ενας διαχειριστής μπορεί να στείλει στον αντιπρόσωπο αυτόν κάποια πράξη διαχείρισης για να εκτελεστεί σε όλες τις οντότητες οι οποίες ανήκουν στην κλάση *ifEntry* και βρίσκονται ένα επίπεδο χαμηλότερα (στο δέντρο της MIB) από την

ΔΟ με όνομα {**systemId = SwitchX, ifDir = output**}. Χάρη του παραπάνω μηχανισμού η πράξη τελικά θα εκτελεστεί στις ΔΟ με ονόματα:

{**systemId = SwitchX, ifDir = output, ifId = 1**}
{**systemId = SwitchX, ifDir = output, ifId = 2**}

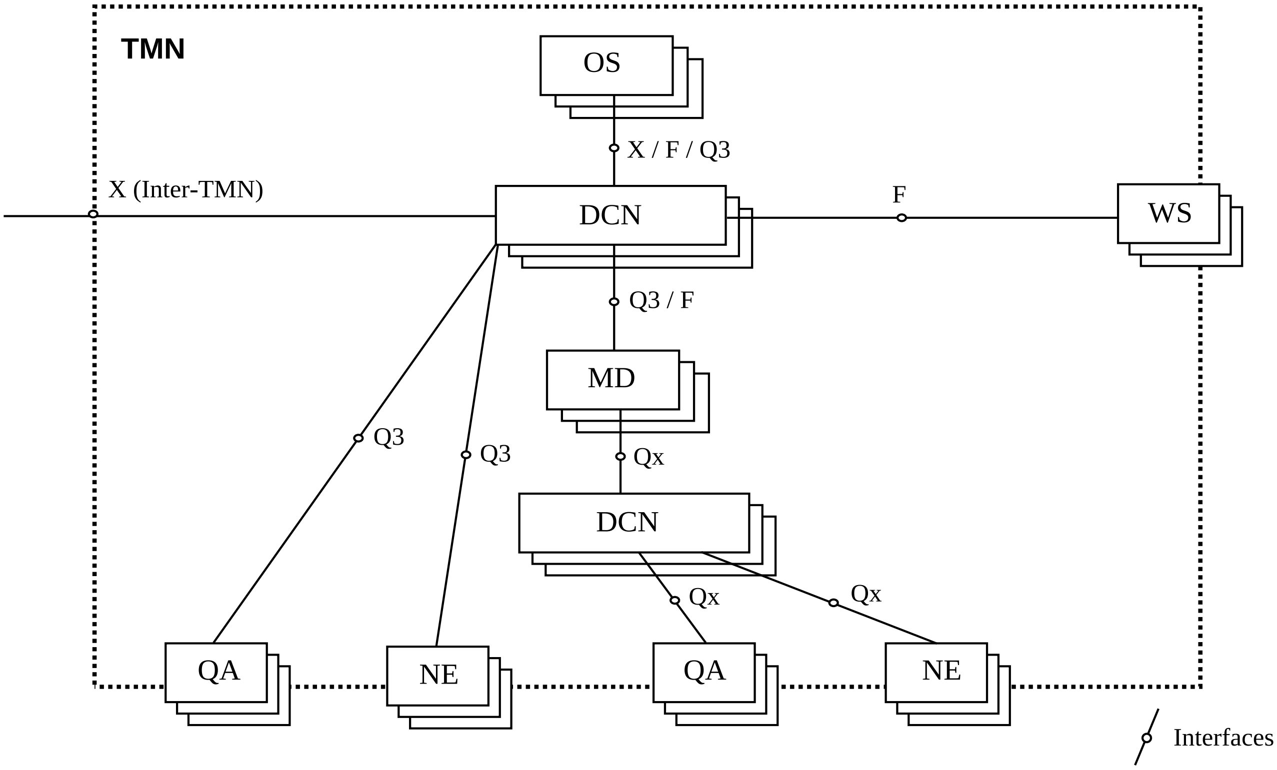
2. Φιλτράροντας (filtering) τις επιλεγμένες από τον προηγούμενο μηχανισμό διαχειρίσιμες οντότητες βάσει της ύπαρξης ή/και των τιμών διαφόρων χαρακτηριστικών της κλάσης.

Επιπλέον, ειδικές παράμετροι χρησιμοποιούνται για το συγχρονισμό των πράξεων που προορίζονται για ένα σύνολο από διαχειρίσιμες οντότητες. Για παράδειγμα, είναι δυνατό να προκαθορίσουμε ότι θέλουμε μία πράξη είτε να εκτελεστεί σε όλες τις διαχειρίσιμες οντότητες του συνόλου είτε να μην εκτελεστεί σε καμία διασφαλίζοντας τη λεγόμενη "ατομικότητα" (atomicity) των πράξεων όταν είναι απαραίτητο. Ολα τα παραπάνω δίνουν ένα ισχυρό πρωτόκολλο το οποίο μπορεί να καταστήσει τα μοντέλα διαχειριστή/αντιπροσώπου και της πληροφορίας διαχείρισης ιδαίτερα λειτουργικά παρέχοντας αξιόλογη υπηρεσία σε συστήματα διαχείρισης όπως το ΔΔΤ που θα περιγράψουμε στην επόμενη ενότητα.
Η υπηρεσία η οποία παρέχεται μέσω του CMIP καλείται CMIS (Common Management Information Service) [Org90b]. Τα CMIS και CMIP είναι αλληλένδετα και συχνά δηλώνονται με τον κοινό όρο "CMIS/CMIP" που αναφέρεται στην υπηρεσία CMIS που παρέχεται "πάνω από" το πρωτόκολλο CMIP. Πρέπει να αναφέρουμε ότι, παρά τη στενή σχέση τους, τα CMIS και CMIP δεν ταυτίζονται. Γενικά, μπορούμε να πούμε ότι το CMIP αναφέρεται στην κωδικοποίηση και σειρά των μηνυμάτων που ανταλλάσσονται κατά την επικοινωνία διαχειριστή/αντιπροσώπου ενώ το CMIS επικεντρώνεται στο περιεχόμενο, τη λειτουργία και τη σημασία τους.
Τέλος, πρέπει να προσθέσουμε ότι το CMIS/CMIP λειτουργεί σε επίπεδο εφαρμογής και αποτελεί μία κατά κανόνα αξιόπιστη υπηρεσία. Αυτό σημαίνει ότι, στις περισσότερες περιπτώσεις, κάθε μήνυμα πρέπει να επιβεβαιωθεί από το απέναντι άκρο το οποίο έτσι ενημερώνει για την επιτυχία ή αποτυχία της επικοινωνίας. Επειδή αυτή η διαδικασία μπορεί να είναι αρκετά χρονοβόρα, σε μερικές περιπτώσεις υπάρχει η ευελιξία της μη επιβεβαίωσης για κάποιο μήνυμα. Για παράδειγμα, ο αντιπρόσωπος που προωθεί μία ειδοποίηση έχει τη δυνατότητα να μην απαιτήσει επιβεβαίωση από το διαχειριστή ότι πράγματι έλαβε την ειδοποίηση.

## 1.2 Το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (ΔΔΤ)

Το Δίκτυο Διαχείρισης Τηλεπικοινωνιών (Telecommunications Management Network – TMN) έχει προταθεί από την *ITU* στη σειρά συστάσεων M.30 (ITU M.30 Recommendations) [Uni93d]. Ο σκοπός του είναι η διαχείριση των δικτύων τηλεπικοινωνιών καθώς και των υπηρεσιών που αυτά προσφέρουν. Στα πλαίσια του ΔΔΤ ο όρος "διαχείριση" αναφέρεται στις λειτουργίες οι οποίες επιτρέπουν την ανταλλαγή και επεξεργασία πληροφορίας με βασικό στόχο την υποστήριξη της σχεδίασης, εγκατάστασης, παρακολούθησης, συντήρησης, λειτουργίας και διοίκησης των δικτύων τηλεπικοινωνιών και των υπηρεσιών τους.
Γενικά, ένα δίκτυο τηλεπικοινωνιών – ή, απλά, δίκτυο – αποτελείται από ένα μεγάλο αριθμό συσκευών όπως μεταγωγείς, δρομολογητές, πολυπλέκτες, σταθμούς εργασίας και μεγάλα υπολογιστικά συστήματα. Τα παραπάνω θα αναφέρονται σαν "στοιχεία δικτύου" (network elements) και θα συνιστούν τα κομμάτια ενός διαχειριζόμενου δικτύου. Το ΔΔΤ είναι ένα ξεχωριστό – λογικά και όχι κατ' ανάγκη φυσικά – δίκτυο το οποίο διασυνδέεται με ένα δίκτυο – το διαχειριζόμενο δίκτυο – σε πολλά σημεία του προκειμένου να μεταφέρεται πληροφορία από και προς το τελευταίο για τους σκοπούς της διαχείρισης. Εσωτερικά, το ΔΔΤ είναι ένα κατανεμημένο σύστημα: πληροφορία διαχείρισης, για ένα μεγάλο αριθμό από διάσπαρτα στοιχεία του διαχειριζόμενου δικτύου και υπηρεσίες οι οποίες μπορεί να παρέχονται μεταξύ πολύ απομακρυσμένων συστημάτων, διατηρείται και ενημερώνεται σε ένα σημαντικό αριθμό από κατανεμημένες διεργασίες. Οι διεργασίες αυτές οι οποίες λειτουργούν σαν αντιπρόσωποι διαχείρισης (management agents) επικοινωνούν με διεργασίες σε ρόλο διαχειριστή (manager) σύμφωνα με το μοντέλο διαχειριστή/αντιπροσώπου που περιγράφηκε στη προηγούμενη ενότητα.
Η βασική ιδέα πίσω από το ΔΔΤ είναι η παροχή μιας οργανωμένης αρχιτεκτονικής για τη διασύνδεση των διαφόρων διεργασιών διαχείρισης τόσο μεταξύ τους όσο και με τις διεργασίες στα διάφορα στοιχεία του δικτύου με σκοπό την εκτέλεση των απαιτούμενων λειτουργιών διαχείρισης (management functions). Το σχήμα 1.4 δείχνει την αρχιτεκτονική του ΔΔΤ η οποία περιλαμβάνει τα παρακάτω δομικά στοιχεία:

- Το δίκτυο διασύνδεσης (Data Communications Network – DCN) του ΔΔΤ αναφέρεται στο φυσικό δίκτυο το οποίο χρησιμοποιείται για την αποστολή πράξεων διαχείρισης και ειδοποιήσεων μεταξύ των υπολοίπων στοιχείων του σχήματος 1.4 τα οποία επικοινωνούν χρησιμοποιώντας το μοντέλο διαχειριστή/αντιπροσώπου που περιγράψαμε. Το δίκτυο αυτό μπορεί να είναι είτε κάποιο τμήμα του διαχειριζόμενου δικτύου είτε κάποιο ξεχωριστό και αυτόνομο δίκτυο τηλεπικοινωνιών το οποίο χρησιμοποιείται αποκλειστικά




Σχήμα 1.4: Η γενική αρχιτεκτονική του ΔΔΤ της *ITU*.

---

από το ΔΔΤ. Είναι φανερό ότι προκειμένου να μπορεί το ΔΔΤ να λειτουργήσει ανεξάρτητα από βλάβες στους συνδέσμους του διαχειριζόμενου δικτύου είναι επιθυμητό το DCN να αποτελεί κάποιο αυτόνομο φυσικό δίκτυο.

- Τα συστήματα λειτουργιών (Operations Systems – OS) αντιστοιχούν στις κεντρικές διεργασίες του ΔΔΤ. Αναλαμβάνουν τον κεντρικό ρόλο της παρατήρησης, του συντονισμού και ελέγχου των διαφόρων λειτουργιών τόσο του διαχειριζόμενου δικτύου όσο και του ιδίου του ΔΔΤ.

- Τα στοιχεία δικτύου (Network Elements – NE) αναφέρονται στους μεταγωγείς, δρομολογητές, σταθμούς εργασίας κ.λπ. του διαχειριζόμενου δικτύου. Κάθε NE, για τους σκοπούς της διαχείρισης, περιέχει μία διεργασία που ενεργεί σε ρόλο αντιπροσώπου (agent) μέσω της οποίας το ίδιο το NE παρατηρείται και ελέγχεται.

Επιπλέον, τα NE περιλαμβάνουν το υλικό και το λογισμικό που είναι υπεύθυνο για τις διάφορες λειτουργίες που επιτελούν και τις υπηρεσίες που παρέχουν στο διαχειριζόμενο δίκτυο. Αυτές οι λειτουργίες και υπηρεσίες αποτελούν το τελικό αντικείμενο της διαχείρισης και δεν είναι μέρος του ΔΔΤ αλλά αντιπροσωπεύονται στο ΔΔΤ από τις διεργασίες διαχείρισης των NE. Ετσι, το κομμάτι του NE που παρέχει αυτή την αντιπροσώπευση είναι μέρος του ΔΔΤ αντίθετα με το κομμάτι που προσφέρει τις υπόλοιπες λειτουργίες και υπηρεσίες του NE. Τα παραπάνω υποδηλώνονται στο σχήμα 1.4 σχεδιάζοντας τα NE έτσι ώστε να τέμνονται από την οριακή γραμμή του ΔΔΤ.

- Τα Q adaptors (QA) αντιστοιχούν σε διεργασίες οι οποίες χρησιμοποιούνται για την επικοινωνία του ΔΔΤ με εξωτερικές διεργασίες διαχείρισης. Οι εξωτερικές αυτές διεργασίες είναι παρόμοιες με αυτές που περιέχονται στα NE με τη βασική διαφορά ότι δεν υποστηρίζουν τον τυποποιημένο τρόπο διασύνδεσης που ορίζεται στο ΔΔΤ (δηλ. μία από τις διασυνδέσεις $Q_x$ και $Q_3$, βλ. παρακάτω). Η βασική ευθύνη των QA είναι η μετάφραση της πληροφορίας και των πράξεων διαχείρισης από τα τυποποιημένα μοντέλα και πρωτόκολλα του ΔΔΤ σε κάποια διαφορετικά – τα οποία συνήθως αναφέρονται ως "ιδιόκτητα" (proprietary) – και αντίστροφα (δηλ. προσαρμόζουν τον "ιδιόκτητο" τρόπο διαχείρισης στον τυποποιημένο, μέσω διασύνδεσης στα τυποποιημένα $Q_x$ και $Q_3$ – απ' όπου και το όνομα *Q adaptors*). Οι λειτουργίες των QA που σχετίζονται με τα "ιδιόκτητα" μοντέλα και πρωτόκολλα δεν είναι μέρος του ΔΔΤ και γι'αυτό τα QA απεικονίζονται στο σχήμα 1.4 με τρόπο παρόμοιο με των NE (δηλ. ένα τμήμα τους βρίσκεται εκτός των ορίων του ΔΔΤ). Στα σημερινά πειραματικά ΔΔΤ, τα QA χρησιμοποιούνται στις περισσότερες περιπτώσεις αντί των NE. Ο λόγος είναι ότι πολύ λίγοι κατασκευαστές εξοπλισμού δικτύων προσφέρουν συστήματα τα οποία έχουν τη δυνατότητα επικοινωνίας χρησιμοποιώντας τα πρωτόκολλα/μοντέλα του *ISO* που περιγράψαμε στην προηγούμενη ενότητα. Αντίθετα, τα περισσότερα συστήματα υλοποιούν αντιπροσώπους διαχείρισης οι οποίοι επικοινωνούν με βάση το πρωτόκολλο *SNMP* [CFSD90] που λειτουργεί πάνω από τα πρωτόκολλα *TCP/IP* του ευρέως γνωστού υπερδικτύου *Internet* [Pos81b], [Pos81a]. Για τη διαχείριση τέτοιων συστημάτων από ένα ΔΔΤ έχουν υλοποιηθεί QA τα οποία δίνουν τη δυνατότητα μετατροπής των πράξεων διαχείρισης του CMIS/CMIP σε πράξεις του SNMP και αντίστροφα [MPBdS95]. Επίσης, υποστηρίζονται κατάλληλες μετατροπές της πληροφορίας διαχείρισης αφού τα δύο μοντέλα πληροφορίας (του *ISO* και του *Internet*) είναι διαφορετικά. Το QA αναλαμβάνει τις αναγκαίες μετατροπές, ώστε από την πλευρά του ΔΔΤ να δίνεται η εντύπωση ότι το QA είναι ένα NE . Ολες οι πράξεις διαχείρισης που στέλνονται στο QA προωθούνται και εκτελούνται τελικά στον αντιπρόσωπο του συστήματος που δεν υποστηρίζει μία τυποποιημένη διασύνδεση με το ΔΔΤ. Λόγω της παραπάνω λειτουργίας, ο αντιπρόσωπος που περιέχει κάθε QA καλείται επίσης *πληρεξούσιος αντιπρόσωπος* (proxy agent) για το αντίστοιχο σύστημα του διαχειριζόμενου δικτύου.

- Οι ενδιάμεσες συσκευές (Mediation Devices – MD) παρεμβάλλονται μεταξύ των NE (ή των QA) και των OS μέσω του DCN. Χρησιμοποιούνται στην περίπτωση που τα NE (ή τα QA) υποστηρίζουν τη διασύνδεση του τύπου $Q_x$ και όχι την $Q_3$ η οποία απαιτείται από τα OS. Ο ρόλος των MD είναι να κάνουν τις αναγκαίες μετατροπές παρέχοντας μία διασύνδεση του τύπου $Q_3$ με τα OS· αυτή η διασύνδεση διαφέρει από την $Q_x$ στο μοντέλο της πληροφορίας διαχείρισης που χρησιμοποιεί και/ή στις λειτουργίες διαχείρισης που υποστηρίζουν τα ASE που επικοινωνούν.

- Ο σταθμός εργασίας (Workstation – WS) αναφέρεται σε ένα σύστημα το οποίο περιέχει μια διεργασία σε ρόλο διαχειριστή και ενα περιβάλλον που μετατρέπει την πληροφορία του ΔΔΤ σε μορφή κατάλληλη για επεξεργασία από τον τελικό χρήστη του ΔΔΤ (δηλ. τον άνθρωπο-διαχειριστή). Οι λειτουργίες που σχετίζονται με το περιβάλλον μετατροπής της πληροφορίας, το οποίο στην πραγματικότητα αποτελεί ένα σύστημα διεπιφάνειας του ΔΔΤ με τον χρήστη (user interface), δεν αποτελούν μέρος του ΔΔΤ. Ετσι, κάποιο μέρος του WS στο σχήμα 1.4 σχεδιάστηκε έξω από το όριο του ΔΔΤ.

Ας διευκρινιστεί ότι καθένα από τα συστήματα (OS, NE, QA, MD, WS) που περιγράφηκαν παραπάνω αντιστοιχεί σε μία μοναδική διεργασία διαχείρισης. Ετσι, στα πλαίσια του ΔΔΤ, η αναφορά σε κάποιο από τα παραπάνω συστήματα θα σημαίνει ισοδύναμα αναφορά σε μία διεργασία διαχείρισης.
Στο σχήμα 1.4 απεικονίζονται και οι διάφορες τύποι διασυνδέσεων (interfaces) μεταξύ των συστημάτων/διεργασιών. Συγκεκριμένα, η διασύνδεση $Q_3$ ορίζεται μεταξύ των OS με διεργασίες του τύπου NE, ή QA, ή MD, ή OS. Η $Q_x$ είναι μεταξύ των MD με διεργασίες του τύπου NE ή QA. Η διασύνδεση $F$ χρησιμοποιείται από τα WS για επικοινωνία με OS ή MD. Τέλος, η διασύνδεση του τύπου $X$ χρησιμοποιείται για την επικοινωνία μεταξύ OS που ανήκουν σε διαφορετικά ΔΔΤ. Θα μπορούσαμε να πούμε ότι τα DCN του σχήματος 1.4 συμβολίζουν ένα μέσο “συγκόλλησης” των ιδίου τύπου διασυνδέσεων που καταλήγουν σε αυτά.

Για λόγους συμβατότητας, κάθε μία από τις παραπάνω διασυνδέσεις έχει αποφασιστεί να γίνεται χρησιμοποιώντας τυποποιημένα πρωτόκολλα επικοινωνίας και μοντέλα πληροφορίας, ενώ ειδική μέριμνα έχει ληφθεί – με τα QA – για περιπτώσεις όπου κάποιο σύστημα δεν ακολουθεί τις απαιτούμενες προδιαγραφές. Ετσι, τα μοντέλα και το πρωτόκολλο που περιγράφηκαν προηγουμένως (διαχειριστή/αντιπροσώπου, ιεραρχικής MIB, CMIS/CMIP) χρησιμοποιούνται για την επικοινωνία των διαφόρων συστημάτων εσωτερικά στο ΔΔΤ.

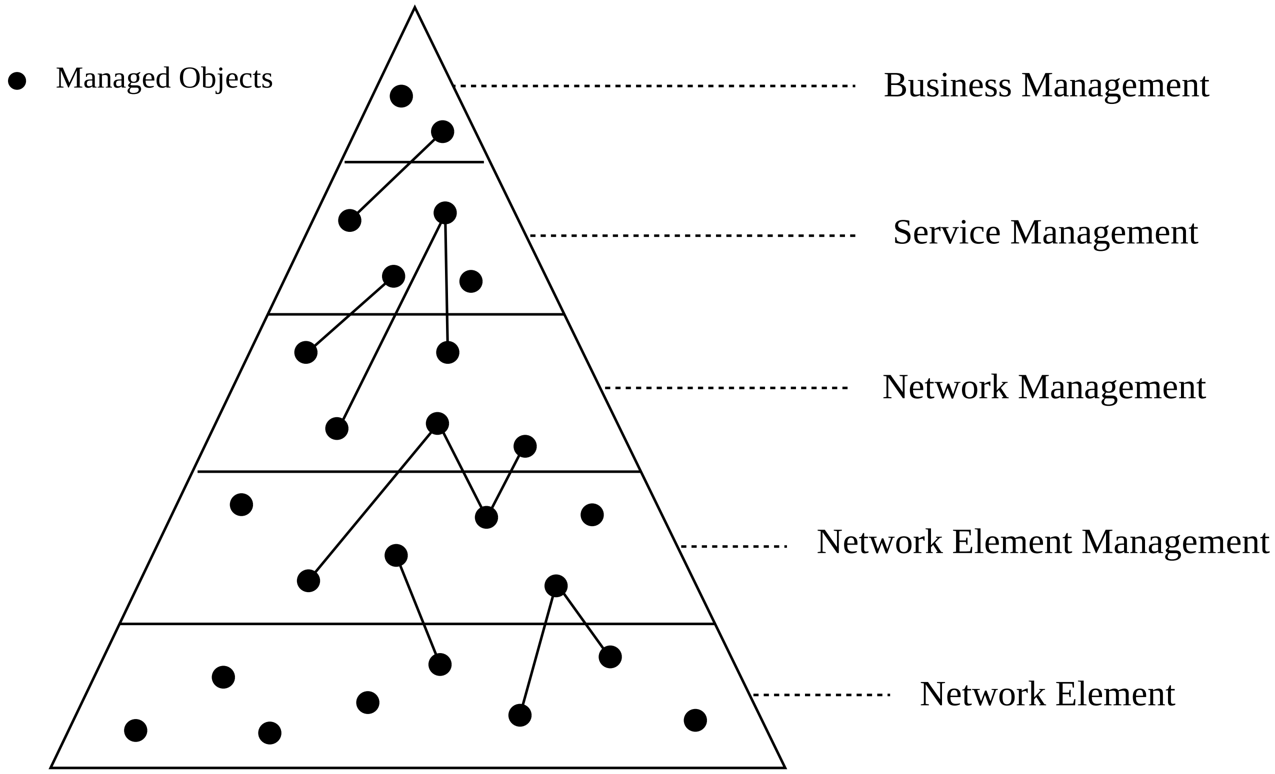
### 1.2.1 Ομαδοποίηση και διαχωρισμός των λειτουργιών του ΔΔΤ

Οι λειτουργίες των διεργασιών των παραπάνω συστημάτων του ΔΔΤ είναι πολλές και ταξινομούνται σε πέντε βασικές λειτουργικές περιοχές διαχείρισης (management functional areas).

- διαχείριση απόδοσης (performance management),
- διαχείριση σφαλμάτων (fault management),
- διαχείριση διαμόρφωσης (configuration management),
- διαχείριση χρέωσης (accounting management) και
- διαχείριση ασφαλείας (security management).

Κάθε πληροφορία που ανταλλάσσεται στο ΔΔΤ μπορεί να χρησιμοποιείται για την υποστήριξη περισσοτέρων από μίας από τις παραπάνω λειτουργικές περιοχές. Επιπροσθέτως, η *ITU* έχει ορίσει έναν περιορισμένο αριθμό (συγκεκριμένα, 19) από υπηρεσίες διαχείρισης (management services) [Uni93f] οι οποίες συνιστούν ένα διαφορετικό τρόπο ομαδοποίησης των λειτουργιών διαχείρισης. Μία υπηρεσία διαχείρισης μπορεί να καλύπτει περισσότερες από μία από τις παραπάνω λειτουργικές περιοχές και χωρίζεται σε διάφορα συστατικά μέρη (management service components) τα οποία με τη σειρά τους αποτελούνται από επιμέρους λειτουργίες διαχείρισης (management functions) [Uni93e]. Μια λειτουργία διαχείρισης είναι το ελάχιστο τμήμα μιας υπηρεσίας διαχείρισης και γενικά αποτελείται από μία σειρά από πράξεις διαχείρισης (συμπεριλαμβανομένων και ειδοποιήσεων) σε μία ή περισσότερες διαχειρίσιμες οντότητες.

Ετσι, για παράδειγμα, μία υπηρεσία διαχείρισης είναι η διαχείριση της κυκλοφορίας (traffic management) σε ένα δίκτυο μεταγωγέων (switching network). Αυτή αποτελείται από




Σχήμα 1.5: Το γενικό ιεραρχικό μοντέλο διαχείρισης του ΔΔΤ.

---

συστατικά (management service components) όπως η παρατήρηση της κατάστασης του δικτύου, η μέτρηση της απόδοσης του δικτύου και ο έλεγχος της πρόσβασης σε αυτό. Καθένα από τα παραπάνω συστατικά χωρίζεται σε έναν αριθμό από λειτουργίες διαχείρισης. Για παράδειγμα, η μέτρηση της απόδοσης του δικτύου περιλαμβάνει τη μέτρηση του φόρτου σε κάθε σύνδεσμο καθώς και κατάλληλους υπολογισμούς σε σχέση με το μέγιστο ανεκτό φόρτο· άλλες απαραίτητες λειτουργίες για τη διαχείριση της κυκλοφορίας είναι ο έλεγχος της ροής της πληροφορίας (flow control) στις προσφερόμενες υπηρεσίες και η επιβολή κάποιου ελέγχου πρόσβασης (access control) σε αυτές. Ολα τα παραπάνω μπορούν να επιτευχθούν με την εκτέλεση πράξεων διαχείρισης σε διάφορες διαχειρίσιμες οντότητες και την εκπομπή κατάλληλων ειδοποιήσεων και, προφανώς, καλύπτουν διάφορες λειτουργικές περιοχές διαχείρισης (π.χ. διαχείριση απόδοσης και διαχείριση ασφάλειας).

### 1.2.2 Το ιεραρχικό μοντέλο διαχείρισης του ΔΔΤ

Εκτός από τον παραπάνω διαμερισμό των λειτουργιών των διαφόρων συστημάτων του ΔΔΤ, η συνολική διαδικασία της διαχείρισης χωρίζεται με ιεραρχικό τρόπο στα πέντε επίπεδα του σχήματος 1.5. Καθένα από τα επίπεδα αυτά περιέχει κάποιο αριθμό από διεργασίες διαχείρισης. Πιο συγκεκριμένα τα επίπεδα στα οποία χωρίζεται η διαδικασία διαχείρισης είναι τα εξής:

- Το επίπεδο των στοιχείων του δικτύου (network element layer) περιέχει τις διαχειρίσιμες οντότητες, που κρατιούνται στους αντιπροσώπους των διαφόρων NE, οι οποίες περιέχουν τη βασική πληροφορία για τους σκοπούς της διαχείρισης κάθε NE. Σε αυτό το επίπεδο ενεργούν εκτός από τα NE και τα QA.

- Το επίπεδο διαχείρισης των στοιχείων του δικτύου (network element management layer) περιλαμβάνει τις διαχειρίσιμες οντότητες οι οποίες, κατά κανόνα, περιέχουν τα αποτελέσματα συγκεκριμένης επεξεργασίας της πληροφορίας του προηγουμένου επιπέδου, ώστε να χρησιμοποιηθούν για την παροχή των διαφόρων υπηρεσιών διαχείρισης σε μεμονωμένα συστήματα του δικτύου. Σε αυτό το επίπεδο ενεργούν τα OS, τα MD και συχνά και τα QA.

- Το επίπεδο διαχείρισης δικτύου (network management layer) περιλαμβάνει τις διαχειρίσιμες οντότητες οι οποίες περιέχουν πληροφορία που προέρχεται απο την επεξεργαοία της πληροφορίας που συγκεντρώνεται στο προηγούμενο επίπεδο. Ο σκοπός αυτών των διαχειρίσιμων οντοτήτων είναι η συμβολή στη διαχείριση ολόκληρων ομάδων πόρων (δηλ. για υποδίκτυα ή ολόκληρο το διαχειριζόμενο δίκτυο).

- Το επίπεδο διαχείρισης υπηρεσιών (service management layer) εξάγει (κατόπιν επεξεργασίας της πληροφορίας του προηγουμένου επιπέδου) και διατηρεί την πληροφορία που απαιτείται για τη διαχείριση των διαφόρων υπηρεσιών που παρέχει το δίκτυο.

- Τέλος, το επίπεδο διαχείρισης επιχειρήσεων (business management layer) περιλαμβάνει την αναγκαία πληροφορία για την οργάνωση των πόρων και των παροχών μιας επιχείρησης που προσφέρει υπηρεσίες πάνω από ένα δίκτυο τηλεπικοινωνιών. Σε αυτό το επίπεδο όπως και στα δύο προηγούμενα ενεργούν μόνο τα OS.

Το βασικό χαρακτηριστικό στην παραπάνω ιεραρχία είναι ότι σε κάθε επίπεδο (εκτός από το χαμηλότερο) οι διαχειρίσιμες οντότητες περιέχουν κατά κανόνα κάποια συγκέντρωση ή

αφαίρεση της πληροφορίας που περιέχεται στο αμέσως κατώτερο επίπεδο. Ο τρόπος με τον οποίο συντελείται η συγκέντρωση ή αφαίρεση αυτή εξαρτάται από την υπηρεσία διαχείρισης την οποία υποστηρίζουν οι διεργασίες που περιέχουν αυτές τις διαχειρίσιμες οντότητες.
Για παράδειγμα, ας υποθέσουμε ότι στο χαμηλότερο επίπεδο έχουμε ένα μεταγωγέα ο οποίος περιλαμβάνει μια διεργασία διαχείρισης σε ρόλο αντιπροσώπου η οποία περιέχει διάφορες διαχειρίσιμες οντότητες που κρατούν πληροφορία όπως το χρονικό διάστημα που έχει παρέλθει από τη στιγμή που άρχισε να λειτουργεί ο μεταγωγέας και τον συνολικό αριθμό των bits που εισήλθαν από κάθε είσοδο του κατά το διάστημα αυτό. Μια τέτοια διεργασία μπορεί είτε να είναι μέρος ενός NE του σχήματος 1.4 είτε, σε περίπτωση που αυτό δεν υποστηρίζει την τυποποιημένη διασύνδεση του ΔΔΤ, να βρίσκεται έξω από το ΔΔΤ και να συνδέεται με αυτό μέσω κάποιου QA.
Μία διεργασία στο επίπεδο διαχείρισης των στοιχείων του δικτύου η οποία ασχολείται με την υποστήριξη, για παράδειγμα, της διαχείρισης της κυκλοφορίας μπορεί να επεξεργάζεται την πληροφορία στο προηγούμενο επίπεδο προκειμένου να υπολογίζει και να κρατάει σε κάποια διαχειρίσιμη οντότητα τον αριθμό τον bits που περνούν από κάθε σύνδεσμο ανά δευτερόλεπτο και το μέσο φόρτο σε όλες τις εισόδους του μεταγωγέα. Μία τέτοια διεργασία μπορεί να περιέχεται, κατά κανόνα, σε ένα OS του σχήματος 1.4. Το βασικό χαρακτηριστικό αυτής της διεργασίας είναι ότι ενεργεί και σε ρόλο διαχειριστή και σε ρόλο αντιπροσώπου. Πραγματικά, η διεργασία αυτή από τη μία πρέπει να στέλνει πράξεις διαχείρισης (*GET*) προκειμένου να διαβάσει την πληροφορία διαχείρισης του παρακάτω επιπέδου (ενεργώντας σαν διαχειριστής) και από την άλλη πρέπει να κρατάει και να ελέγχει ένα σύνολο από διαχειρίσιμες οντότητες (ενεργώντας σε ρόλο αντιπροσώπου). Αυτή η διπλή λειτουργία ισχύει για όλες τις διεργασίες οι οποίες πρέπει να κρατούν και να ελέγχουν κάποια πληροφορία διαχείρισης η οποία προέρχεται από επεξεργασία της πληροφορίας ενός χαμηλότερου επιπέδου.
Στο επίπεδο διαχείρισης δικτύου, μία διεργασία του ΔΔΤ που υποστηρίζει την παραπάνω υπηρεσία διαχείρισης χρησιμοποιεί με όμοιο τρόπο την πληροφορία που περιέχεται στις διαχειρίσιμες οντότητες του παρακάτω επιπέδου[7] προκειμένου να συγκεντρώσει πληροφορία για όλο το δίκτυο. Στη συνέχεια, επεξεργάζεται την πληροφορία για το φόρτο του κάθε συνδέσμου και το μέσο φόρτο των μεταγωγέων του δικτύου και υπολογίζει το φόρτο ολόκληρου του δικτύου. Η διεργασία αυτή θα περιέχεται σε κάποιο OS του ΔΔΤ στο επίπεδο διαχείρισης

[7]Συγκεκριμένα, την πληροφορία των ΔΟ που κρατούνται από τις διεργασίες που ενεργούν στο επίπεδο διαχείρισης των στοιχείων του δικτύου και ασχολούνται με τη διαχείριση της κυκλοφορίας για τους διάφορους μεταγωγείς του δικτύου).

δικτύου.
Τα παραπάνω επεκτείνονται με όμοιο τρόπο στα υψηλότερα επίπεδα διαχείρισης. Ετσι, στο επίπεδο διαχείρισης υπηρεσιών μία διεργασία η οποία ασχολείται με τη διαχείριση μίας υπηρεσίας μπορεί να χρησιμοποιεί την πληροφορία σχετικά με το φόρτο των διαφόρων δικτύων προκειμένου να υπολογίσει το μέγιστο αριθμό χρηστών που μπορούν να χρησιμοποιήσουν μία υπηρεσία μεταξύ δύο απομακρυσμένων δικτύων. Σε αυτήν την περίπτωση ίσως χρειάζεται να ανταλλαχτεί πληροφορία μεταξύ απομακρυσμένων ΔΔΤ (μέσω της διασύνδεσης *X*) η οποία σχετίζεται με τη διαχείριση της κυκλοφορίας στα δίκτυα αυτά.

### 1.2.3 Μερικές παρατηρήσεις πάνω στο ΔΔΤ

Τα παραπάνω συνοψίζουν τη γενική μεθοδολογία που ακολουθεί το ΔΔΤ για τη διαχείριση των δικτύων τηλεπικοινωνιών και των υπηρεσιων τους. Η συνολική διαδικασία χωρίζεται σε έναν αριθμό από υπηρεσίες διαχείρισης οι οποίες υποστηρίζονται σε διάφορα ιεραρχικά επίπεδα διαχείρισης. Ομάδες από διεργασίες ενεργούν σε καθένα από τα επίπεδα αυτά. Επιπλεόν, ομάδες από διεργασίες συνεργάζονται για την παροχή κάποιας υπηρεσίας διαχείρισης ενώ μία διεργασία μπορεί να συμμετέχει σε περισσότερες από μία υπηρεσίες διαχείρισης εκτελώντας κάποιο αριθμό από λειτουργίες. Δύο βασικές παρατηρήσεις μπορούν να γίνουν σε αυτό το σημείο.
Κατ' αρχήν, μια διεργασία διαχείρισης του ΔΔΤ μπορεί να ενεργεί είτε μόνο σε ρόλο διαχειριστή (π.χ. WS), είτε μόνο σε ρόλο αντιπροσώπου (π.χ. NE), είτε και στους δύο. Πράγματι, όπως ήδη εξηγήσαμε, ένα OS στο επίπεδο διαχείρισης των στοιχείων του δικτύου (network element management layer) ενεργεί όχι μόνο σε ρόλο διαχειριστή – επεμβαίνοντας στην πληροφορία διαχείρισης του παρακάτω επιπέδου – αλλά και σε ρόλο αντιπροσώπου – κρατώντας διαχειρίσιμες οντότητες σε μία MIB στην οποία επεμβαίνουν οι διεργασίες του παραπάνω επιπέδου (δηλ. του επιπέδου διαχείρισης δικτύου). Στην ορολογία του *ISO* μία διεργασία διαχείρισης[8] περιλαμβάνει τουλάχιστον μία οντότητα εφαρμογής για τη διαχείριση συστημάτων (Systems Management Application Entities – SMAE). Μία οντότητα εφαρμογής (application entity) είναι το κομμάτι μιας διεργασίας εφαρμογής (application process) το οποίο περιλαμβάνει τις απαραίτητες λειτουργίες για την επικοινωνία με άλλες διεργασίες εφαρμογής σύμφωνα με κάποιο σύνολο πρωτοκόλλων (ένα από αυτά είναι το CMIP για την

[8] Η οποία καλείται και "διεργασία εφαρμογής για τη διαχείριση συστημάτων" ((Systems Management Application Process – SMAP) επειδή αποτελεί μία διεργασία στο επίπεδο εφαρμογής (application layer) [Org94b].

περίπτωση των διεργασίων διαχείρισης). Επειδή καθένας από τους ρόλους αντιπροσώπου και διαχειριστή υποστηρίζει διαφορετικές λειτουργίες (ένας διαχειριστής επεξεργάζεται ειδοποιήσεις και στέλνει πράξεις διαχείρισης, ενώ ένας αντιπρόσωπος προωθεί ειδοποιήσεις και εκτελεί πράξεις διαχείρισης), πρέπει να παρέχεται μέσω κάποιας ξεχωριστής SMAE. Ετσι, σύμφωνα με τα παραπάνω, μία διεργασία διαχείρισης μπορεί να περιλαμβάνει μία ή δύο SMAE.

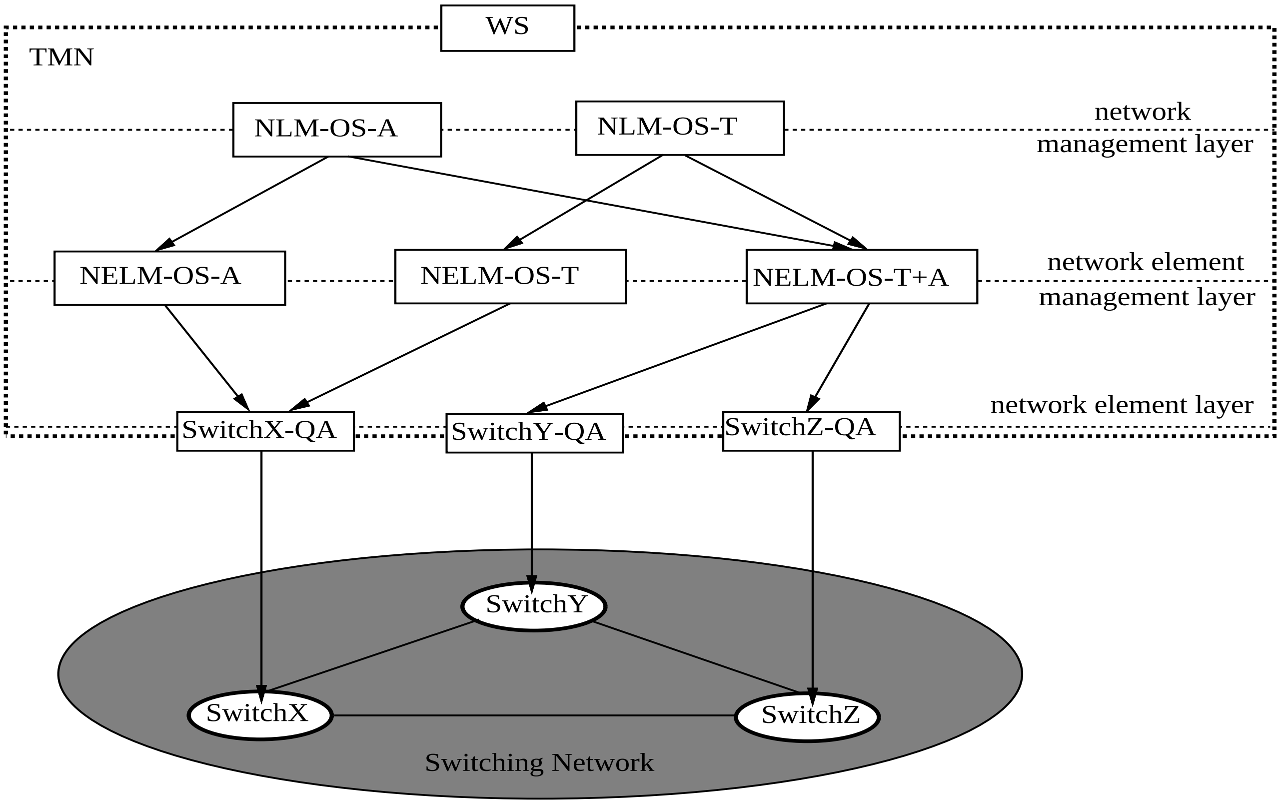
Δεύτερη σημαντική παρατήρηση είναι ότι, εκτός από τις διεργασίες που περιέχονται σε κάποιο NE, γενικά οι διεργασίες του ΔΔΤ είναι μετακινήσιμες (relocatable). Αυτό σημαίνει ότι η τοποθέτησή τους δεν είναι μόνιμη, ενώ υπάρχει ευελιξία στη διαμόρφωση του ΔΔΤ με τρόπο που οι διεργασίες να είναι κοντά (όσον αφορά τη σχετική τοπολογία του ΔΔΤ με το διαχειριζόμενο δίκτυο) στα συστήματα που διαχειρίζονται. Επιπλέον, αν κάποιο σύστημα του ΔΔΤ δεν είναι διαθέσιμο λόγω βλάβης οι διεργασίες που εκτελούνταν σε αυτό μπορούν να μεταφερθούν σε κάποιο άλλο. Τα παραπάνω είναι αναμφισβήτητα πλεονεκτήματα του ΔΔΤ, επειδή προσφέρουν ανοχή σε βλάβες και μεγαλύτερη αξιοπιστία.

### 1.2.4 Παράδειγμα ενός ΔΔΤ για ένα απλό δίκτυο μεταγωγέων

Στο σχήμα 1.6 απεικονίζεται το ΔΔΤ για ένα δίκτυο τριών διασυνδεδεμένων μεταγωγέων με ονόματα *switchX*, *switchY* και *switchZ*. Στη συνέχεια περιγράφουμε τον τρόπο με τον οποίο δομείται και διασυνδέεται το ΔΔΤ του σχήματος 1.6 που ενδεικτικά ασχολείται με την παροχή δύο υπηρεσιών διαχείρισης. Για χάρη γενικότητας, υποθέτουμε ότι καθένας από τους μεταγωγείς περιέχει μία διεργασία διαχείρισης (σε ρόλο αντιπροσώπου) η οποία δεν υποστηρίζει τις τυποποιημένες διασυνδέσεις του ΔΔΤ[9]. Γι'αυτόν το λόγο οι λειτουργίες του επιπέδου των στοιχείων του δικτύου (network element layer) παρέχονται με τη χρήση QA. Αναλυτικά, το ΔΔΤ για το παραπάνω δίκτυο μεταγωγέων περιλαμβάνει τα εξής συστήματα (καθένα από τα οποία περιέχει μία διεργασία διαχείρισης):

- Τρία QA τα οποία περιέχουν τις διαχειρίσιμες οντότητες για τους τρείς μεταγωγείς. Παρόλο που κάποιο QA μπορεί να περιέχει τις διαχειρίσιμες οντότητες για περισσότερα από ένα στοιχεία του δικτύου, στο παράδειγμα μας δείχνουμε την απλή ένα-προς-ένα περίπτωση.

[9]Το παράδειγμα προσαρμόζεται εύκολα στην περίπτωση που οι μεταγωγείς του δικτύου του σχήματος 1.6 περιέχουν κάποια NE.




Σχήμα 1.6: Ενα πιθανό ΔΔΤ για ένα δίκτυο τριών μεταγωγέων.

---

Κάθε QA περιλαμβάνει έναν αντιπρόσωπο ο οποίος ελέγχει την ίδια ουσιαστικά πληροφορία με τον αντιπρόσωπο του αντίστοιχου μεταγωγέα. Η διαφορά μεταξύ τους βρίσκεται στο πρωτόκολλο επικοινωνίας και στο μοντέλο της πληροφορίας που υποστηρίζουν. Ετσι, ο αντιπρόσωπος του QA ακολουθεί τις τυποποιήσεις που χρησιμοποιούνται στο ΔΔΤ – σε αντίθεση με αυτόν του μεταγωγέα – παρέχοντας τη δυνατότητα διαχείρισης του μεταγωγέα από το ΔΔΤ. Ολες οι πράξεις διαχείρισης που στέλνονται στο QA προωθούνται και εκτελούνται τελικά στον αντιπρόσωπο του μεταγωγέα.

- Τρία OS τα οποία λειτουργούν στο επίπεδο διαχείρισης των στοιχείων του δικτύου (network element management layer) και ειδικότερα θα καλούμε NELM-OS. Καθένα τους υποστηρίζει μία ή δύο υπηρεσίες διαχείρισης. Στο συγκεκριμένο ΔΔΤ έχουμε υποθέσει την παροχή δύο διαφορετικών υπηρεσιών διαχείρισης. Αυτές είναι οι *διαχείριση κυκλοφορίας* (traffic management) και *διαχείριση χρέωσης* (accounting management). Στο σχήμα 1.6 το NELM-OS με όνομα *NELM-OS-A* είναι υπεύθυνο για τη υποστήριξη της διαχείρισης της χρέωσης στο μεταγωγέα *switchX* ενώ αυτό με όνομα *NELM-OS-T* υποστηρίζει την διαχείριση της κυκλοφορίας στον ίδιο μεταγωγέα. Τέλος, το *NELM-OS-T+A* υποστηρίζει και τις δύο παραπάνω υπηρεσίες διαχείρισης για τους *switchY* και *switchZ*. Η παραπάνω οργάνωση και λειτουργία των NELM-OS δείχνει τους πολλούς διαφορετικούς τρόπους διασύνδεσης των διεργασιών του ΔΔΤ. Κάθε NELM-OS λειτουργεί σε ρόλο διαχειριστή για τα QA του παρακάτω επιπέδου ενώ αναλαμβάνει και το ρόλο του αντιπροσώπου κρατώντας κατάλληλες διαχειρίσιμες οντότητες τις οποίες χρησιμοποιεί το υψηλότερο επίπεδο διαχείρισης.

- Δύο OS τα οποία λειτουργούν στο επίπεδο διαχείρισης δικτύου (network management layer) – καλούνται NLM-OS (network layer of management OS) – καθένα από τα οποία παρέχει μία από τις παραπάνω υπηρεσίες διαχείρισης για ολόκληρο το δίκτυο των τριών μεταγωγέων. Το *NLM-OS-T* παρέχει τη διαχείριση κυκλοφορίας ενώ το *NLM-OS-A* τη διαχείριση χρέωσης. Οπως είναι φυσικό, το καθένα από τα παραπάνω συστήματα επικοινωνεί με τις διεργασίες του παρακάτω επιπέδου ανάλογα με την υπηρεσία διαχείρισης που θέλει να υποστηρίξει. Ετσι, το *NLM-OS-A* επικοινωνεί με τις *NELM-OS-A* και *NELM-OS-T+A* προκειμένου να επεξεργαστεί την πληροφορία που είναι σχετική με χρέωση.

- Ενα WS το οποίο προσφέρει στον χρήστη του ΔΔΤ (δηλ. τον άνθρωπο-διαχειριστή) τη δυνατότητα επικοινωνίας με οποιοδήποτε από τα παραπάνω OS. Στην ιδανική περίπτωση, όπου όλα λειτουργούν χωρίς απρόβλεπτα λάθη ή βλάβες, ο χρήστης αυτός θα συμβουλεύεται και θα ελέγχει μόνο τις δύο τελευταίες διεργασίες – δηλ. τα NLM-OS – οι οποίες για το λόγο αυτό θα πρέπει να λειτουργούν και σε ρόλο αντιπροσώπου κρατώντας κατάλληλες διαχειρίσιμες οντότητες. Ας σημειωθεί ότι το WS δεν ανήκει σε κάποιο επίπεδο διαχείρισης. Ο βασικός ρόλος του είναι να επικοινωνεί με τα διάφορα OS σε οποιοδήποτε επίπεδο κι αν ενεργούν αυτά.

Ο ακριβής τρόπος με τον οποίο λειτουργεί το παραπάνω ΔΔΤ εξαρτάται από τον τρόπο υλοποίησης των διαφόρων διεργασιών. Ας υποθέσουμε ότι ο άνθρωπος-διαχειριστής – ο οποίος χρησιμοποιεί το WS – θέλει να βρει το μέσο φόρτο μεταξύ των συνδέσμων στο δίκτυο των τριών μεταγωγέων (δηλ. θέλει κάποια πληροφορία η οποία παρέχεται από την υπηρεσία διαχείρισης της κυκλοφορίας). Σε αυτήν την περίπτωση προφανώς θα πρέπει να επικοινωνήσει με το *NLM-OS-T*. Το *NLM-OS-T* μπορεί είτε να έχει ήδη έτοιμη την πληροφορία αυτή[10], είτε να την υπολογίσει οπότε θα πρέπει να συνδεθεί με τους *NELM-OS-T* και *NELM-OS-T+A* και να πάρει τα επιμέρους στοιχεία για το μέσο φόρτο των διαφόρων συνδέσμων μεταξύ των μεταγωγέων. Ομοια, οι διεργασίες *NELM-OS-T* και *NELM-OS-T+A* μπορεί να χρειαστεί να συνδεθούν με τα αντίστοιχα QA δημιουργώντας έτσι νέες συνδέσεις από τα QA στους αντιπροσώπους των μεταγωγέων ζητώντας πιο βασική πληροφορία όπως ο αριθμός των bits που εισέρχονται από κάθε σύνδεσμο και ο χρόνος λειτουργίας του κάθε μεταγωγέα. Βέβαια, αν κάθε διεργασία ανανεώνει από μόνη της την απαιτούμενη πληροφορία, οι παραπάνω συνδέσεις δεν είναι αναγκαίες. Πρέπει να διευκρινιστεί ότι ο ακριβής τρόπος ενημέρωσης της πληροφορίας των διεργασιών (περιοδικά ή κατ' απαίτηση – on demand) αποτελεί μία επιλογή κατά την υλοποίηση των διεργασιών και δεν υπόκειται σε τυποποίηση. Πάντως με τον ένα ή τον άλλο τρόπο τελικά ο διαχειριστής θα πάρει από την MIB του *NLM-OS-T* την απαιτούμενη πληροφορία.

Ας υποθέσουμε τώρα ότι ο διαχειριστής θέλει να ρυθμίσει το μέγιστο επιτρεπτό φόρτο στο δίκτυο των τριών μεταγωγέων. Και σε αυτήν την περίπτωση αρκεί να συνδεθεί με το *NLM-OS-T*. Το *NLM-OS-T* πρέπει να περιέχει την κατάλληλη ΔΟ στην οποία η εκτέλεση κάποιας πράξης διαχείρισης πρέπει να πυροδοτεί την αποστολή μίας νέας πράξης διαχείρισης από το διαχειριστή του *NLM-OS-T* προς τους *NELM-OS-T* και *NELM-OS-T+A κ.ο.κ.* Με αυτόν τον τρόπο, αρκεί ο αντιπρόσωπος να εκτελέσει κάποια πράξη διαχείρισης στην κατάλληλη ΔΟ του *NLM-OS-T* που τελικά θα πυροδοτήσει τις αναγκαίες πράξεις διαχείρισης στους αντιπροσώπους των μεταγωγέων, οι οποίες με τη σειρά τους θα προκαλέσουν τις αναγκαίες αλλαγές σε κάποιες παραμέτρους του λογισμικού του μεταγωγέα και, τελικά, στη λειτουργία του. Τα παραπάνω βέβαια προϋποθέτουν ότι οι MIB των διαφόρων διεργασιών του ΔΔΤ έχουν υλοποιηθεί με τρόπο που η μία πράξη διαχείρισης να πυροδοτεί την επόμενη στο αμέσως χαμηλότερο επίπεδο και τελικά να ενεργοποιείται μία αλλαγή στη λειτουργία του μεταγωγέα. Και πάλι αυτό είναι θέμα υλοποίησης το οποίο σχετίζεται με τον προγραμματισμό ορισμένης συμπεριφοράς (behaviour) των ΔΟ στις MIB των μεταγωγέων.

Πρέπει να σημειωθεί ότι το παράδειγμα του ΔΔΤ που παρουσιάσαμε σε αυτή την υποενότητα δεν περιλαμβάνει διεργασίες στα δύο ψηλότερα επίπεδα διαχείρισης του σχήματος 1.5. Το

[10] Αυτό ισχύει μόνο αν περιοδικά ανανεώνει από μόνο του την πληροφορία αυτή σε κάποια ΔΟ.

γεγονός αυτό αφείλεται στο ότι, κυρίως για χάρη απλότητας αλλά και ανάλογα με την περίσταση, ο βαθμός στον οποίο ένα ΔΔΤ αυτοματοποιεί τη διαδικασία της διαχείρισης μπορεί να ποικίλει. Στο παράδειγμά μας ο χρήστης του ΔΔΤ (δηλ. ο διαχειριστής που χρησιμοποιεί το WS) μπορεί να χρησιμοποιήσει τα στοιχεία που συγκεντρώνει στο επίπεδο του δικτύου για να εξάγει συμπεράσματα σχετικά με τις απαιτούμενες ενέργειες, π.χ. για τη διαχείριση ή παροχή μίας υπηρεσίας του δικτύου· δηλαδή, η διαχείριση στο επίπεδο υπηρεσιών δεν συντελείται από το ΔΔΤ του παραδείγματός μας αλλά αφήνεται στο χρήστη του ΔΔΤ.

## 1.3 Η υπηρεσία καταλόγου (X.500)

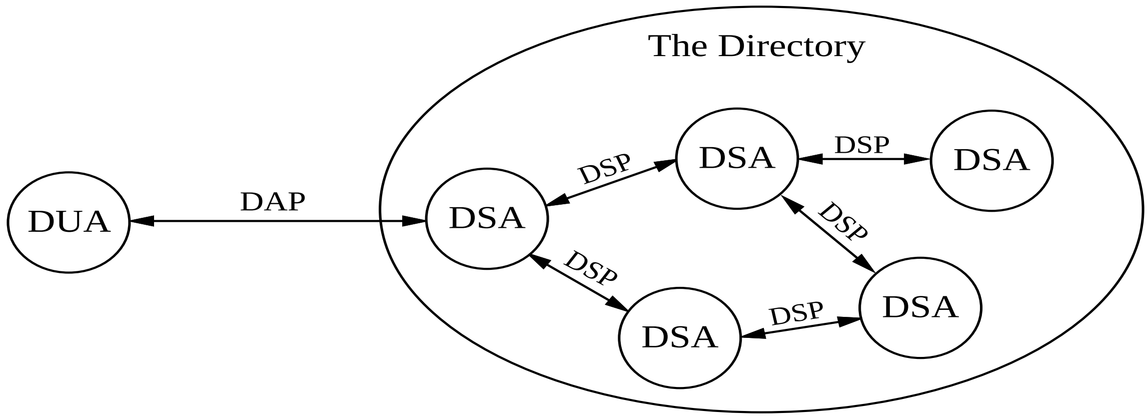
Η **Υπηρεσία Καταλόγου** (Directory Service – X.500) που έχει προταθεί από την *ITU* και τον *ISO* [Uni93a] περιγράφει μια κατανεμημένη βάση δεδομένων – η οποία καλείται Κατάλογος[11] (Directory) – καθώς και μία αρχιτεκτονική και ένα σύνολο πρωτοκόλλων τα οποία παρέχουν τους μηχανισμούς για πρόσβαση στη βάση αυτή. Ο βασικός στόχος του X.500 είναι η παροχή ενός παγκόσμιου on-line καταλόγου ο οποίος θα περιέχει πληροφορία όχι μόνο για ανθρώπους όπως τον αριθμό τηλεφώνου και τις διάφορες διευθύνσεις τους (σπιτιού, εργασίας, e-mail), αλλά και για τα διάφορα διασυνδεδεμένα συστήματα και τις διεργασίες τους που παρέχουν κάποιες υπηρεσίες πάνω από δίκτυα τηλεπικοινωνιών. Ετσι, ο Κατάλογος αποτελείται από ένα μεγάλο αριθμό εγγραφών που περιέχουν πληροφορία για πολλών ειδών οντότητες (π.χ. για ανθρώπους, διεργασίες, συστήματα και υπηρεσίες του δικτύου).

### 1.3.1 Το μοντέλο λειτουργίας του X.500

Το σχήμα 1.7 περιγράφει το μοντέλο λειτουργίας της υπηρεσίας καταλόγου (X.500). Η πληροφορία του Καταλόγου είναι κατανεμημένη σε έναν αριθμό από ειδικές διεργασίες που καλούνται *DSA* (Directory Service Agents). Παρόλο που κάθε DSA κρατάει μόνο ένα τμήμα της συνολικής πληροφορίας που περιέχεται στο Κατάλογο, η κατανομή της πληροφορίας στους επιμέρους DSA δεν γίνεται ορατή στο χρήστη του X.500 χάρη στη χρήση του πρωτοκόλλου *DSP* (Directory Service Protocol) μεταξύ των DSA. Χάρη σε αυτό το πρωτόκολλο, κάθε DSA που δέχεται μία αίτηση για κάποιο κομμάτι πληροφορίας, το οποίο δεν βρίσκεται στο τμήμα του Καταλόγου που αυτός διατηρεί, έχει τη δυνατότητα, βάσει της γνώσης που έχει σχετικά με το "ποιός DSA κρατάει ποιό τμήμα του Καταλόγου", να εξυπηρετήσει την αίτηση με τους παρακάτω τρόπους:

1. Προωθώντας την αίτηση σε κάποιο DSA ο οποίος, είτε είναι αυτός που κρατάει την ζητούμενη πληροφορία, είτε μπορεί με τη σειρά του να προωθήσει την αίτηση σε έναν επόμενο DSA. Επειδή, γενικά, ένας DSA μπορεί να μην ξέρει ακριβώς πού κρατιέται κάποια πληροφορία αλλά να ξέρει κάποιο DSA ο οποίος μπορεί να προωθήσει μία αίτηση τελικά στο ζητούμενο DSA, τα παραπάνω μπορεί να προκαλέσουν μία αλυσιδωτή σειρά αιτήσεων (chaining),

[11] Εφεξής, η ειδική αυτή σημασία του όρου "Κατάλογος" θα υποδηλώνεται από τη χρήση του κεφαλαίου 'Κ' στην αρχή της λέξης.




Σχήμα 1.7: Το μοντέλο λειτουργίας της υπηρεσίας καταλόγου (X.500).

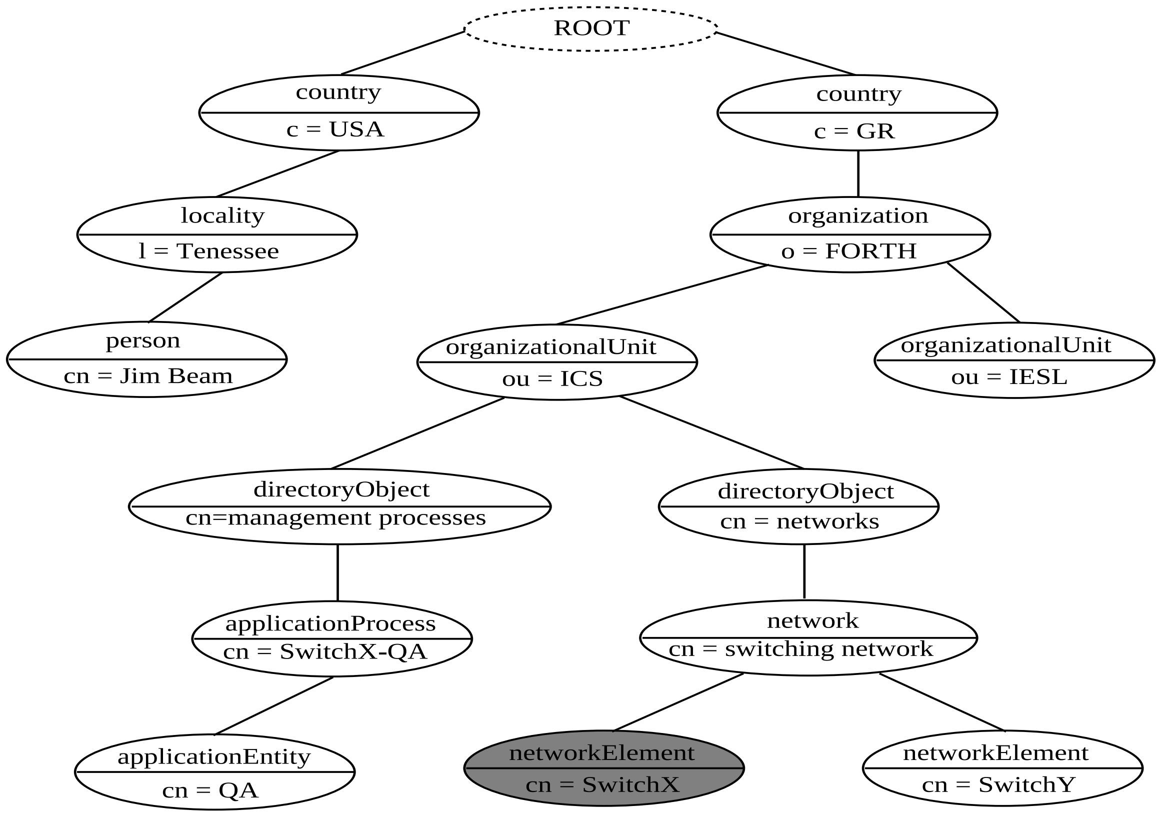
2. Προωθώντας την αίτηση σε πολλούς DSA (multicasting) από τους οποίους κάποιος μπορεί να έχει τη ζητούμενη πληροφορία.

3. Παραπέμποντας τον αιτόντα σε κάποιο καταλληλότερο DSA.

Κάθε χρήστης του X.500 (ο οποίος μπορεί κάλλιστα να είναι μία διεργασία εφαρμογής) έχει τη δυνατότητα πρόσβασης στο Κατάλογο μέσω ενός *DUA* (Directory User Agent) ο οποίος παρέχει μηχανισμούς προσπέλασης, αναζήτησης, ενημέρωσης και τροποποίησης της πληροφορίας του Καταλόγου μέσω του πρωτοκόλλου *DAP* (Directory Access Protocol). Το πρωτόκολλο αυτό χρησιμοποιείται για την επικοινωνία μεταξύ DUA και DSA και παρέχει ένα ισχυρό σύνολο εντολών το οποίο μαζί με την χρήση του DSP δίνει δυνατότητα πρόσβασης και επέμβασης στη συνολική καταναμημένη πληροφορία του Καταλόγου. Τα παραπάνω φυσικά υποστηρίζονται σε συνδυασμό με ειδικούς μηχανισμούς ελέγχου της πρόσβασης (access control) διαφυλάσσοντας την ακεραιότητα της πληροφορίας καθώς και τη μυστικότητά της, όπου κρίνεται αναγκαίο.

### 1.3.2 Το μοντέλο της πληροφορίας του X.500

Προαναφέραμε ότι το X.500 περιέχει πληροφορία για πολλών ειδών οντότητες. Σε κάθε οντότητα αντιστοιχεί μία εγγραφή στον Κατάλογο η οποία καλείται *οντότητα καταλόγου*




Σχήμα 1.8: Παράδειγμα μιας υποθετικής DIB και της λογικής της οργάνωσης.

(directory object). Το σύνολο των οντοτήτων καταλόγου απαρτίζει την Βάση Πληροφορίας του Καταλόγου (Directory Information Base – DIB) η οποία, θεωρητικά, οργανώνεται ιεραρχικά σε μία δενδροειδή δομή όπως αυτή του σχήματος 1.8.

Κάθε οντότητα καταλόγου χαρακτηρίζεται από τις κλάσεις στις οποίες ανήκει. Οι κλάσεις αυτές καθορίζουν τα χαρακτηριστικά (attributes) τα οποία μπορεί να ενέχει η οντότητα. Κάθε κλάση οντοτήτων καταλόγου (directory object class) περιέχει ένα σύνολο υποχρεωτικών και ένα σύνολο προαιρετικών χαρακτηριστικών. Κάθε χαρακτηριστικό μπορεί να λάβει μία ή περισσότερες τιμές σχηματίζοντας ένα ή περισσότερα ζευγάρια χαρακτηριστικού-τιμής τα οποία αποτελούν τα ελάχιστα κομμάτια της πληροφορίας του Καταλόγου.

Για κάθε οντότητα καταλόγου ένα χαρακτηριστικό, το οποίο καλείται *χαρακτηριστικό ονομασίας* (naming attribute), χρησιμοποιείται μαζί με την τιμή του για το σχηματισμό του ονόματος της. Αυτό το ζευγάρι χαρακτηριστικού-τιμής αποτελεί το λεγόμενο *σχετικά διακεκριμένο όνομα* (relative distinguished name – RDN) της οντότητας καταλόγου[12]. Το RDN έχει παρόμοια έννοια με αυτή που περιγράφηκε για τις διαχειρίσιμες οντότητες στην πρώτη ενότητα του παρόντος κεφαλαίου. Δηλαδή, αναγνωρίζει μοναδικά την οντότητα ανάμεσα σε αυτές με τον ίδιο πατέρα – στο δέντρο της DIB – και αποτελεί ένα μέρος του ονόματος της. Η συνένωση όλων των RDN των κόμβων στο μονοπάτι από τη ρίζα του δέντρου της DIB μέχρι τον κόμβο που αντιστοιχεί στην εν λόγω οντότητα καταλόγου δίνει στην τελευταία ένα παγκόσμια μοναδικό όνομα το οποίο καλείται *διακεκριμένο όνομα* (distinguished name – DN).

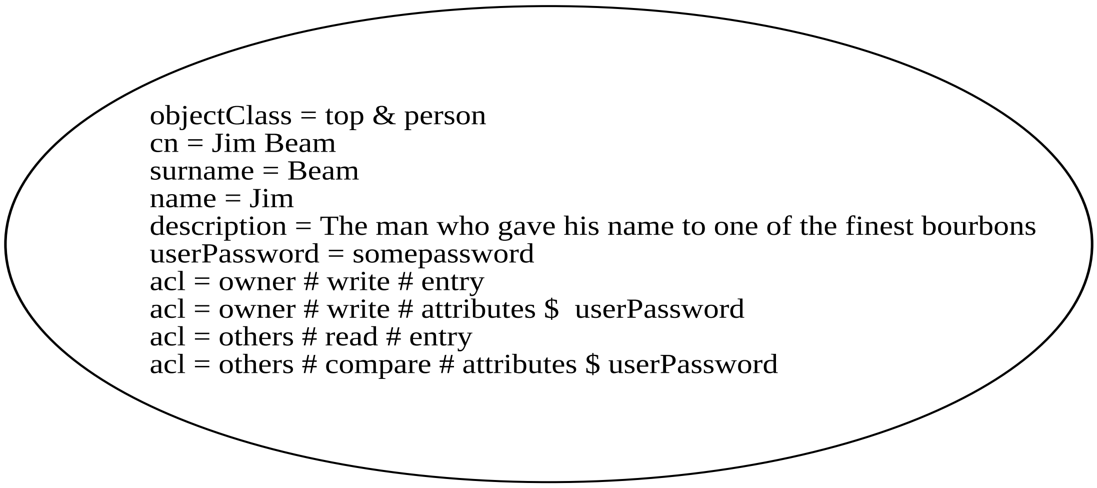
Στο σχήμα 1.8 δίνεται ένα παράδειγμα μιας DIB. Κάθε κόμβος αντιστοιχεί σε μία οντότητα καταλόγου. Η ρίζα του δέντρου του σχήματος 1.8 βρίσκεται μέσα σε μία διακεκομένη κλειστή καμπύλη συμβολίζοντας μία ιδεατή και όχι πραγματική οντότητα η οποία δεν έχει κάποιο όνομα αλλά χρησιμοποιείται απλά για τη λογική σύνδεση όλων των οντοτήτων καταλόγου σε ένα μοναδικό ιδεατό δέντρο.

Στο σχήμα 1.8 η κλάση στην οποία ανήκει κάθε οντότητα καταλόγου αναγράφεται στην πάνω γραμμή του κόμβου ενώ από κάτω αναγράφεται το RDN της οντότητας. Οπως προαναφέραμε, κάθε οντότητα καταλόγου – σε αντίθεση με τις διαχειρίσιμες οντότητες – συνήθως ανήκει σε περισσότερες από μία κλάσεις οι οποίες προσδίδουν επιπλέον χαρακτηριστικά στην οντότητα (π.χ μία ξεχωριστή κλάση μπορεί να περιέχει τα χαρακτηριστικά για τον έλεγχο της πρόσβασης στην οντότητα). Στο σχήμα 1.8 αναγράφεται μόνο μία κλάση η οποία είναι και η πιο αντιπροσωπευτική για το είδος της πληροφορίας που περιέχει η οντότητα καταλόγου. Ετσι, ο γραμμοσκιασμένος κόμβος αντιστοιχεί σε μία οντότητα καταλόγου η οποία περιέχει πληροφορία για ένα στοιχείο του δικτύου, ανήκει στην κλάση *networkElement*, έχει RDN {**cn=SwitchX**} και το διακεκριμένο όνομα της είναι:

{**c=GR, o=FORTH, ou=ICS, cn= networks, cn=switching network, cn=SwitchX**}.

Οπως και στην περίπτωση της MIB, μία βασική παρατήρηση στο σχήμα 1.8 είναι ότι η σχέση των οντοτήτων υποδηλώνεται από τη σχετική τους θέση στο εικονιζόμενο δέντρο. Ετσι, οι οντότητες με RDN {**cn=SwitchX**} και {**cn=SwitchY**} αντιπροσωπεύουν στοιχεία που περιλαμβάνονται στο ίδιο δίκτυο με διακεκριμένο όνομα:

[12] Πρέπει να σημειώσουμε ότι στην ονομασία μιας οντότητας καταλόγου μπορούν να συμμετέχουν περισσότερα από ένα χαρακτηριστικά της οντότητας. Ετσι ένα RDN μπορεί να αποτελείται από πολλά ζευγάρια χαρακτηριστικού-τιμής. Για λόγους απλότητας, η σπάνια αυτή περίπτωση δεν θα μας απασχολήσει.




Σχήμα 1.9: Παράδειγμα των ζευγαριών χαρακτηριστικού-τιμής για μία οντότητα καταλόγου.

---

{**c=GR, o=FORTH, ou=ICS, cn=networks, cn=switching network**}.
Η παραπάνω σχέση των δύο στοιχείων του δικτύου υποδηλώνεται από το γεγονός ότι στο δέντρο της DIB έχουν κοινό πατέρα την οντότητα με το παραπάνω διακεκριμένο όνομα η οποία – όπως γίνεται φανερό από την κλάση στην οποία ανήκει – αντιπροσωπεύει ένα δίκτυο. Επιπλέον, είναι φανερό ότι το παραπάνω δίκτυο ανήκει στον υποοργανισμό *ICS* ο οποίος υπάγεται στον ελληνικό οργανισμό *FORTH*.
Οπως διαφαίνεται από τη λογική οργάνωση της DIB στο σχήμα 1.8, στα υψηλότερα επίπεδα εγγράφονται οντότητες που αντιπροσωπεύουν χώρες, γεωγραφικές περιοχές και οργανισμούς, ενώ στα χαμηλότερα επίπεδα βρίσκονται οντότητες που αντιπροσωπεύουν ανθρώπους, συστήματα και διεργασίες. Τα παραπάνω επιβάλλονται από συγκεκριμένους κανόνες οργάνωσης της πληροφορίας του Καταλόγου οι οποίοι προτείνονται στις σχετικές τυποποιήσεις του X.500 και αποσκοπούν στην όσο το δυνατό αποτελεσματικότερη λειτουργία του X.500.
Σχετικά με την οργάνωση της DIB θα πρέπει ακόμα να τονίσουμε ότι λόγω των πολλών ειδών πληροφορίας που περιέχει ο Κατάλογος μερικές οντότητες παίζουν κυρίως ομαδοποιητικό ρόλο. Ετσι, η οντότητα με RDN {**cn=management processes**} αφορά βασικά στην ομαδοποίηση και εγγραφή όλων των διεργασιών διαχείρισης οι οποίες περιλαμβάνονται σε συστήματα του υποοργανισμού *ICS* κάτω από ένα κοινό υπόδεντρο της DIB.
Προαναφέραμε ότι η πληροφορία που μπορεί να περιέχει μία οντότητα καταλόγου εξαρτάται

από τις κλάσεις στις οποίες ανήκει. Το σχήμα 1.9 δείχνει ένα παράδειγμα της πληροφορίας – με τη μορφή ζευγαριών χαρακτηριστικού-τιμής – που περιέχει η οντότητα καταλόγου που αντιπροσωπεύει έναν άνθρωπο με DN:
**{c=USA, l=Tenessee, cn=Jim Beam}**.
Οπως βλέπουμε στο DN αυτό, το χαρακτηριστικό ονομασίας για την παραπάνω οντότητα καταλόγου είναι το *cn* το οποίο αποτελεί μία βραχυγραφία του τυποποιημένου χαρακτηριστικού *commonName* [Uni93b]. Η οντότητα καταλόγου του σχήματος 1.9 ανήκει σε δύο κλάσεις:

- **top**: Στην τυποποιημένη κλάση αυτή ανήκει κάθε οντότητα του καταλόγου [Uni93c]. Περιέχει τα υποχρεωτικά χαρακτηριστικά *objectClass* και *acl*. Και τα δύο τους μπορούν να πάρουν περισσότερες από μία τιμές στην ίδια οντότητα. Οι τιμές του πρώτου χαρακτηριστικού δηλώνουν τις κλάσεις στις οποίες ανήκει η οντότητα, ενώ οι τιμές του δευτέρου είναι μία λίστα κανόνων ελέγχου της πρόσβασης (access control list) στη συγκεκριμένη οντότητα καταλόγου. Παρακάτω θα αναφέρουμε περισσότερα για το χαρακτηριστικό *acl*.

- **person**: Στην κλάση αυτή ανήκει κάθε οντότητα του καταλόγου η οποία περιέχει πληροφορία για κάποιον άνθρωπο. Περιέχει υποχρεωτικά τα χαρακτηριστικά *cn*, *surName* και *userPassword*, καθώς και διάφορα προαιρετικά χαρακτηριστικά όπως τα *Name* και *description*.

Από τα παραπάνω, το χαρακτηριστικό *acl* είναι ιδιαίτερα σημαντικό. Σύμφωνα με τα όσα είπαμε το χαρακτηριστικό αυτό πρέπει να υπάρχει σε κάθε οντότητα καταλόγου. Οι τιμές του ορίζουν μία λίστα που επιβάλλει συγκεκριμένους κανόνες ελέγχου πρόσβασης στην οντότητα. Κάθε τιμή αποτελείται από τρία μέρη. Το πρώτο προσδιορίζει κάποιο χρήστη (ή μία λίστα χρηστών) του X.500 και το δεύτερο ορίζει τις πράξεις που επιτρέπονται από τον προηγούμενο χρήστη στις οντότητες (ή τα χαρακτηριστικά) που καθορίζονται στο τρίτο μέρος. Συνεπώς, η γραμμή (στην οποία ο χαρακτήρας '#' διαχωρίζει τα παραπάνω τρία μέρη)
`acl = others # compare # attributes $ userPassword`
δηλώνει ότι όλοι οι χρήστες (εκτός από αυτόν που αντιστοιχεί στην ίδια την οντότητα καταλόγου και ο οποίος αναγνωρίζεται από τη λέξη `owner`) μπορούν να συγκρίνουν το χαρακτηριστικό *userPassword*.
Με αφορμή τα παραπάνω πρέπει ακόμα να πούμε ότι κάθε DUA μπορεί να κάνει χρήση του X.500 αφού αναγνωριστεί σαν κάποιος συγκεκριμένος χρήστης ο οποίος αντιστοιχεί σε κάποια

εγγεγραμμένη οντότητα του Καταλόγου. Για να γίνει αυτό χρησιμοποιείται μία ειδική εντολή του πρωτοκόλλου DAP η οποία απαιτεί δύο παραμέτρους: Το DN της οντότητας καταλόγου – η οποία αντιστοιχεί στο χρήστη και συνήθως αναφέρεται σε κάποιον άνθρωπο – και μία μυστική λέξη-κλειδί (password) η οποία αντιστοιχεί στην οντότητα. Με βάση τη γνώση αυτού του μυστικού κλειδιού – η οποία είναι η τιμή του χαρακτηριστικού *userPassword* – πιστοποιείται η ταυτότητα ενός χρήστη στο X.500. Από τα παραπάνω διαφαίνεται ο σκοπός που εξυπηρετούν οι τιμές του χαρακτηριστικού *acl* που είναι η επιβολή ελέγχου στον τρόπο με τον οποίο οι διάφοροι χρήστες αποκτούν πρόσβαση και δικαιώματα επέμβασης στην πληροφορία του Καταλόγου. Περισσότερες λεπτομέρειες υπάρχουν στις σχετικές τυποποιήσεις.
Τέλος, πρέπει να τονιστεί ότι η πληροφορία του Καταλόγου πρέπει να είναι "σχετικά στατική" προκειμένου να λειτουργήσουν αποτελεσματικά τα πρωτόκολλα του X.500. Ως σχετικά στατική χαρακτηρίζεται η πληροφορία που δεν αλλάζει συχνά δηλαδή, για παράδειγμα, δεν αλλάζει μερικές φορές το λεπτό. Οπως θα δούμε στην επόμενη ενότητα, η κοινή γνώση που απαιτείται στα περιβάλλοντα διαχείρισης είναι σχετικά στατική πληροφορία.

## 1.4 Η Κοινή Γνώση Διαχείρισης (ΚΓΔ)

Οι διάφορες διεργασίες, οι οποίες συμμετέχουν και συνεργάζονται σε ένα σύστημα διαχείρισης δικτύων όπως το ΔΔΤ[13], απαιτούν συγκεκριμένη γνώση προκειμένου να προσδιορίσουν τις διεργασίες με τις οποίες πρέπει να επικοινωνήσουν καθώς και τα διάφορα χαρακτηριστικά αυτών των διεργασιών (π.χ. τα πρωτόκολλα επικοινωνίας που υποστηρίζουν, την πληροφορία διαχείρισης που περιέχουν και τις διευθύνσεις στις οποίες "ακούν"). Η γνώση αυτή ονομάζεται Κοινή Γνώση Διαχείρισης (Shared Management Knowledge) [Org92a].

### 1.4.1 Προσδιορισμός της Κοινής Γνώσης Διαχείρισης κατά ISO

Η σύσταση *ISO 10164-16* [Org95] διακρίνει την Κοινή Γνώση Διαχείρισης (ΚΓΔ) στις εξής κατηγορίες:

- Γνώση σχετική με τις κλάσεις των διαχειρίσιμων οντοτήτων (managed object classes). Αυτή περιλαμβάνει την εξής πληροφορία:
  - δοθείσης μίας διεργασίας που ενεργεί σε ρόλο αντιπροσώπου, ποια χαρακτηριστικά υποστηρίζονται και ποια όχι για κάθε κλάση·
  - ποιοι περιορισμοί επιβάλλονται στις διαχειρίσιμες οντότητες μιας συγκεκριμένης κλάσης (π.χ. από όλες τις επιτρεπτές τιμές ενός χαρακτηριστικού της κλάσης ποιο είναι το υποσύνολο των τιμών που υλοποιείται)·
  - ποιες κλάσεις διαχειρίσιμων οντοτήτων διατίθενται από αυτήν[14]· και
  - σε μία διεργασία, κατά τη δημιουργία μίας νέας διαχειρίσιμης οντότητας που ανήκει σε μία συγκεκριμένη κλάση, ποιες είναι οι αρχικές τιμές που εξ ορισμού δίνονται στα χαρακτηριστικά της οντότητας.
- Γνώση των διαθέσιμων διαχειρίσιμων οντοτήτων. Αυτή αποτελείται από την εξής πληροφορία:

[13] Όπως έχουμε προαναφέρει, η συνεργασία των διεργασιών διαχείρισης μπορεί να συμβαίνει και μεταξύ διαφορετικών ΔΔΤ (χρησιμοποιώντας τη διασύνδεση του τύπου X).

[14] Λέμε ότι μία κλάση διαχειρίσιμων οντοτήτων διατίθεται από έναν αντιπρόσωπο αν ο τελευταίος περιέχει κάποιες διαχειρίσιμες οντότητες οι οποίες ανήκουν στην κλάση αυτή και στις οποίες επιτρέπεται να εκτελεστεί κάποια πράξη διαχείρισης.

- δοθέντων μίας διεργασίας και μίας συγκεκριμένης κλάσης, ποιες και πόσες είναι οι διαχειρίσιμες οντότητες της διεργασίας[15] που ανήκουν στην κλάση· και
- ποιες είναι οι διεργασίες που διαθέτουν μία ζητούμενη διαχειρίσιμη οντότητα.

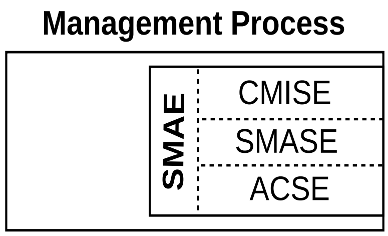
- Γνώση του τρόπου ονομασίας των διαχειρίσιμων οντοτήτων. Περιέχει πληροφορία όπως:
    - για κάθε διεργασία η οποία ενεργεί σε ρόλο αντιπροσώπου, ποιοι είναι οι κανόνες ονομασίας που ισχύουν στην MIB της διεργασίας. Με άλλα λόγια, ποιο είναι το δέντρο ονομασίας (βλ. §1.1.2) που ακολουθεί η διεργασία· και
    - μέχρι πόσες διαχειρίσιμες οντότητες μιας συγκεκριμένης κλάσης μπορεί να έχει η MIB κάθε διεργασίας – που ενεργεί σε ρόλο αντιπροσώπου.

- Γνώση των ορισμών της πληροφορίας διαχείρισης. Ο *ISO* έχει προτείνει μία γλώσσα για την περιγραφή της πληροφορίας διαχείρισης [Org92d]. Με τη βοήθεια της γλώσσας αυτής ορίζονται οι διάφορες κλάσεις διαχειρίσιμων οντοτήτων, οι σχέσεις τους, τα χαρακτηριστικά τους και οι πράξεις διαχείρισης που επιτρέπονται σε κάθε ΔΟ που ανήκει στις κλάσεις αυτές. Η γνώση αυτή αφορά τόσο στους παραπάνω ορισμούς της υπάρχουσας πληροφορίας διαχείρισης όσο και στο ποιοι από αυτούς είναι κατανοητοί από κάθε διεργασία η οποία ενεργεί σε ρόλο διαχειριστή. Το παραπάνω σημαίνει ότι κάποιος διαχειριστής πιθανώς να μη μπορεί να κατανοήσει κάποιο τμήμα της πληροφορίας διαχείρισης ενός αντιπροσώπου. Ακόμα και η γνώση αυτού του γεγονότος είναι αναγκαία.

- Γνώση σχετική με τις επιμέρους διεργασίες διαχείρισης. Αυτή αποτελείται από πληροφορία όπως:
    - η διεύθυνση στην οποία "ακούει" μία οντότητα εφαρμογής (SMAE) κάποιας διεργασίας διαχείρισης για αιτήσεις πράξεων διαχείρισης (στην περίπτωση αντιπροσώπου) ή για ειδοποιήσεις (στην περίπτωση διαχειριστή). Σε συμφωνία με την ορολογία του *ISO* η παραπάνω διεύθυνση θα καλείται διεύθυνση παρουσίασης (presentation address) ή, καλύτερα, PSAP (presentation service access point[16])·

[15] Οπουδήποτε αναφερόμαστε σε διεργασίες που περιέχουν (ή διαθέτουν) διαχειρίσιμες οντότητες εξυπακούεται ότι οι διεργασίες αυτές ενεργούν σε ρόλο αντιπροσώπου.

[16] Η εξήγηση του όρου αυτού είναι πέρα από τα πλαίσια της παρούσας αναφοράς. Ο ενδιαφερόμενος αναγνώστης παραπέμπεται στην τυποποίηση [Org94b] για περισσότερες λεπτομέρειες.




Σχήμα 1.10: Η γενική δομή μίας διεργασίας διαχείρισης.

---

- οι λειτουργίες και εκδόσεις των διαφόρων πρωτοκόλλων επικοινωνίας και υπηρεσιών που υποστηρίζει μία SMAE. Σε αυτό το σημείο πρέπει να αναφέρουμε ότι κάθε οντότητα εφαρμογής του *ISO* περιλαμβάνει διάφορα στοιχεία τα οποία καλούνται *στοιχεία υπηρεσίας εφαρμογής* (Application Service Elements – ASE) καθένα από τα οποία εκτελεί ένα σύνολο λειτουργιών. Στην περίπτωση των SMAE περιλαμβάνονται τρία ASE τα οποία είναι τα SMASE, CMISE και ACSE. Τα πρώτα δύο αναφέρονται σε λειτουργίες οι οποίες προσφέρονται μέσω του CMIS/CMIP ενώ το τρίτο είναι υπεύθυνο για τη σύνδεση (εγκατάσταση, κράτηση, έλεγχο και κλείσιμο αυτής) μεταξύ δύο SMAE. Στο σχήμα 1.10 απεικονίζεται η γενική δομή μίας διεργασίας διαχείρισης η οποία περιέχει μία SMAE με τα παραπάνω ASE. Για κάθε τύπου ASE ορίζεται ένα σύνολο υποχρεωτικών λειτουργιών οι οποίες πρέπει πάντα να επιτελούνται από το ASE, καθώς και ένα σύνολο προαιρετικών λειτουργιών που αποσκοπούν στην αύξηση της λειτουργικότητας του ASE. Συνεπώς, η πληροφορία τού ποιες λειτουργίες επιτελούν τα ASE μίας SMAE είναι επιθυμητή προκειμένου να επικοινωνήσουν αποτελεσματικότερα οι διάφορες διεργασίες διαχείρισης· και
- το πλαίσιο της εφαρμογής (application context) που υποστηρίζει μία SMAE. Αυτό αναφέρεται βασικά στο ποια ASE περιέχει η SMAE.

Η παραπάνω γνώση απαρτίζει την ΚΓΔ που πρέπει να μοιράζονται οι διάφορες διεργασίες διαχείρισης. Προφανώς η παραπάνω πληροφορία δεν αναμένεται να αλλάζει πολύ συχνά.

Είναι δηλαδή σχετικά στατική πληροφορία[17] και συνεπώς η υπηρεσία καταλόγου μπορεί να χρησιμοποιηθεί σαν βάση των διαφόρων μηχανισμών για την αποθήκευση και ανάκτηση της ΚΓΔ. Σε αυτό το σημείο πρέπει να αναφέρουμε τις βασικές απαιτήσεις που τίθενται για τους μηχανισμούς αυτούς.
Πρώτον, οι μηχανισμοί αυτοί πρέπει να επιβάλλουν κάποιον έλεγχο για την πρόσβαση στα διάφορα τμήματα της ΚΓΔ. Για παράδειγμα, η πληροφορία για το ποιες διαχειρίσιμες οντότητες περιέχονται στην MIB ενός αντιπροσώπου διαχείρισης για λόγους ασφαλείας δεν θα πρέπει να είναι διαθέσιμη σε οποιαδήποτε διεργασία. Δεύτερον, πρέπει να προσφέρεται η δυνατότητα σε μία διεργασία να αποκτήσει τμήματα της παραπάνω γνώσης σχετικά με μία άλλη διεργασία χωρίς να χρειάζεται να συνδεθεί με την τελευταία. Γενικά, η ΚΓΔ μεταξύ ενός διαχειριστή και ενός αντιπροσώπου χρειάζεται να αποκτηθεί σε διάφορες στιγμές και ειδικότερα:

- πριν από τη δημιουργία μίας σύνδεσης διαχειριστή/αντιπροσώπου,
- κατά τη διάρκεια της δημιουργίας της σύνδεσης (δηλ. κατά την ανταλλαγή μηνυμάτων για τη δημιουργία της σύνδεσης), και
- κατά τη διάρκεια της επικοινωνίας διαχειριστή/αντιπροσώπου.

Το πότε ακριβώς χρειάζεται κάποιο κομμάτι της ΚΓΔ εξαρτάται από την πληροφορία με την οποία σχετίζεται. Για παράδειγμα, η διεύθυνση (PSAP) μίας διεργασίας πρέπει να είναι γνωστή κατά τη διάρκεια δημιουργίας της σύνδεσης με τη διεργασία. Σαν μια τελευταία απαίτηση, αναφέρουμε την ανάγκη να ενημερώνεται η ΚΓΔ και να τροποποιείται με κατάλληλους μηχανισμούς οποιαδήποτε στιγμή χρειαστεί. Για παράδειγμα, η αλλαγή της διεύθυνσης (PSAP) μίας διεργασίας – που μπορεί να οφείλεται στη αναγκαστική μετακίνηση της διεργασίας λόγω κάποιας βλάβης – πρέπει να μπορεί αυτόματα να γίνει αντιληπτή από οποιαδήποτε διεργασία θέλει να επικοινωνήσει με τη μετακινούμενη.

[17] Εξαίρεση μπορεί να αποτελεί η πληροφορία σχετικά με τις υπάρχουσες διαχειρίσιμες οντότητες σε κάποιο σύστημα στην περίπτωση που αυτές ανήκουν σε ειδικές κλάσεις οι οποίες έχουν το χαρακτηριστικό οι ΔΟ που ανήκουν σε αυτές να δημιουργούνται και να διαγράφονται πολύ συχνά. Η τυποποίηση ISO-10164-16 προβλέποντας αυτόν τον κίνδυνο προτείνει να μην φυλάσσεται στον κατάλογο πληροφορία σχετική με τέτοιες διαχειρίσιμες οντότητες.

### 1.4.2 Πληροφορία που συμπεριλάβαμε στη ΚΓΔ

Οπως αναφέραμε στη §1.1.2 κάθε διαχειρίσιμη οντότητα περιέχει πληροφορία για κάποια υπηρεσία ή κάποιο πόρο – φυσικό ή λογικό – του δικτύου. Σε γενικές γραμμές, η σχέση των διαχειρίσιμων οντοτήτων με τους πόρους[18] που αντιπροσωπεύουν είναι αυθαίρετη. Οχι μόνο επειδή μία διαχειρίσιμη οντότητα μπορεί να αντιπροσωπεύει πολλούς πόρους αλλά και επειδή ένας πόρος μπορεί να αντιπροσωπεύεται από πολλές διαχειρίσιμες οντότητες (κάθε μία από τις οποίες παρέχει μία διαφορετική αφαίρεση του πόρου).

Η βασική παρατήρηση που πρέπει να γίνει στην ταξινόμηση της προηγούμενης υποενότητας είναι ότι δεν αναφέρεται πουθενά η γνώση για το ποιες διαχειρίσιμες οντότητες αντιπροσωπεύουν τους διάφορους πόρους. Ο λόγος για αυτήν την παράλειψη από τις τυποποιήσεις του *ISO* βρίσκεται στο ότι αφενός τέτοιου είδους γνώση θεωρείται ότι προϋπάρχει στα συστήματα διαχείρισης, τα οποία αρχικά διαμορφώνονται με βάση κάποια αντιστοιχία πόρων σε διεργασίες (και, κατ' επέκταση, σε διαχειρίσιμες οντότητες), αφετέρου δε, μέχρι σήμερα, δεν έχει θεωρηθεί σκόπιμο να γίνουν αντικείμενο τυποποίησης οι μηχανισμοί απόκτησης της παραπάνω γνώσης[19].

Ομως η ανάγκη της παραπάνω γνώσης μπορεί εύκολα να φανεί με ένα παράδειγμα. Ας θεωρήσουμε την περίπτωση διαχείρισης ενός δικτύου κατά την οποία πρέπει να παρθούν κάποιες αποφάσεις σχετικές με την αναδιαμόρφωση (reconfiguration) του διαχειριζόμενου δικτύου (π.χ. επαναδρομολόγηση κάποιων συνδέσεων) λόγω κάποιας βλάβης σε αυτό. Προκειμένου να δοθεί μία όσο το δυνατό αποτελεσματικότερη λύση στο παραπάνω σενάριο απαιτείται πληροφορία σχετική με τους κόμβους του δικτύου (π.χ. ο τρόπος διασύνδεσης τους – δηλ. η τοπολογία του δικτύου). Υστερα από κάποια αρχική επεξεργασία, θα επιλεγούν ορισμένοι κόμβοι οι οποίοι πρέπει να αναδιαμορφωθούν. Συνεπώς, οι διεργασίες οι οποίες περιέχουν τις διαχειρίσιμες οντότητες που αντιπροσωπεύουν τους κόμβους αυτούς πρέπει να αναγνωριστούν προκειμένου να εκτελεστούν οι απαιτούμενες πράξεις διαχείρισης στις οντότητες αυτές. Γίνεται φανερό λοιπόν ότι απαιτείται μία αντιστοίχιση από κόμβους σε διεργασίες διαχείρισης και, κατ' επέκταση, σε διαχειρίσιμες οντότητες.

[18] Για τη συνέχεια της ενότητας ο όρος "πόρος" θα περιλαμβάνει και τις υπηρεσίες του δικτύου. Ο διαχωρισμός τους θα γίνει στο επόμενο κεφάλαιο.

[19] Ο λόγος για το τελευταίο είναι ότι η έννοια του πόρου παραμένει σχετικά ασαφής στις τυποποιήσεις του ISO όπου ένας διαχειριζόμενος πόρος μπορεί να είναι οτιδήποτε (π.χ. μεταγωγέας, σύνδεσμος, πίνακας δρομολόγησης). Από την άλλη, οι τυποποιήσεις που σχετίζονται με το ΔΔΤ χωρίζουν του διαχειριζόμενους πόρους σε δύο μεγάλες κατηγορίες οι οποίες υποδηλώνονται και από τα επίπεδα διαχείρισης του ΔΔΤ: στοιχεία του δικτύου (network elements) και δίκτυα. Οπως θα δούμε στο επόμενο κεφάλαιο, στην παρούσα εργασία, χρησιμοποιήσαμε την παραπάνω απλή ταξινόμηση για τους διαχειριζόμενους πόρους

Ως μία πιο διαισθητική περίπτωση, αρκεί να αναφέρουμε ότι οι χρήστες ενός συστήματος διαχείρισης (δηλ. οι άνθρωποι-διαχειριστές) τις περισσότερες φορές επικεντρώνονται σε υπηρεσίες διαχείρισης που θέλουν να παρέχουν πάνω σε γνωστούς πόρους παρά σε γνωστές διεργασίες διαχείρισης. Ετσι, π.χ., πρέπει να υπάρχει ένας τρόπος να αντιστοιχίσουμε σε ένα δεδομένο κόμβο του δικτύου τις κατάλληλες διαχειρίσιμες οντότητες οι οποίες αντιπροσωπεύουν τον κόμβο.

Συμπερασματικά, επειδή περιπτώσεις όπως η παραπάνω, όπου η αρχική πληροφορία που έχει μία διεργασία διαχείρισης είναι κάποιος πόρος του δικτύου, παρουσιάζονται συχνά κατά τη λειτουργία ενός συστήματος διαχείρισης, η ΚΓΔ πρέπει να περιλαμβάνει την εξής πληροφορία:

- Γνώση σχετική με τις διαχειρίσιμες οντότητες που αντιπροσωπεύουν κάποιο πόρο. Ειδικότερα αυτή αποτελείται από την εξής πληροφορία:
  - δοθέντος ενός πόρου του διαχειριζόμενου δικτύου, ποιες είναι οι διεργασίες οι οποίες περιέχουν τις διαχειρίσιμες οντότητες που αντιπροσωπεύουν κάποια αφαίρεση του πόρου αυτού·
  - δοθέντος ενός πόρου του διαχειριζόμενου δικτύου, ποιες είναι οι διαχειρίσιμες οντότητες που αντιπροσωπεύουν κάποια αφαίρεση του πόρου αυτού·
  - δοθέντος ενός πόρου του διαχειριζόμενου δικτύου, ποιες είναι οι διεργασίες οι οποίες παρέχουν μία ορισμένη υπηρεσία διαχείρισης για αυτόν τον πόρο – π.χ. ποιες διεργασίες είναι υπεύθυνες για τη χρέωση (accounting management) των χρηστών του πόρου.

Η παραπάνω πληροφορία αποτελεί πρωταρχική γνώση που πρέπει να έχουν οι διεργασίες του ΔΔΤ. Ανήκει στη ΚΓΔ και, προφανώς, πρέπει να αποκτάται πριν από την επικοινωνία δύο διεργασιών διαχείρισης – εφόσον μόνο αφού αποκτηθεί από μία διεργασία γίνεται γνωστή η άλλη με την οποία θα συντελεστεί η επικοινωνία. Επίσης, ας σημειωθεί ότι δεν συμπεριλάβαμε την αντίστροφη αντιστοίχιση (δηλ. από διαχειρίσιμες οντότητες σε πόρους) θεωρώντας την τελευταία αυτή γνώση μάλλον εξεζητημένη και περιορισμένης χρησιμότητας.

Τέλος, στην παρούσα εργασία περιλάβαμε στη ΚΓΔ πληροφορία η οποία κρίθηκε αναγκαία για λόγους πληρότητας στα πλαίσια του ΔΔΤ. Αυτή αναφέρεται:

- στην κατηγορία που ανήκει κάθε διεργασία μέσα στο ΔΔΤ (π.χ. QA, NE, WS), και
- στην υπηρεσία (ή υπηρεσίες) διαχείρισης που παρέχει η κάθε διεργασία.

# Κεφάλαιο 2

# Το προτεινόμενο μοντέλο και οι βασικές χρήσεις του

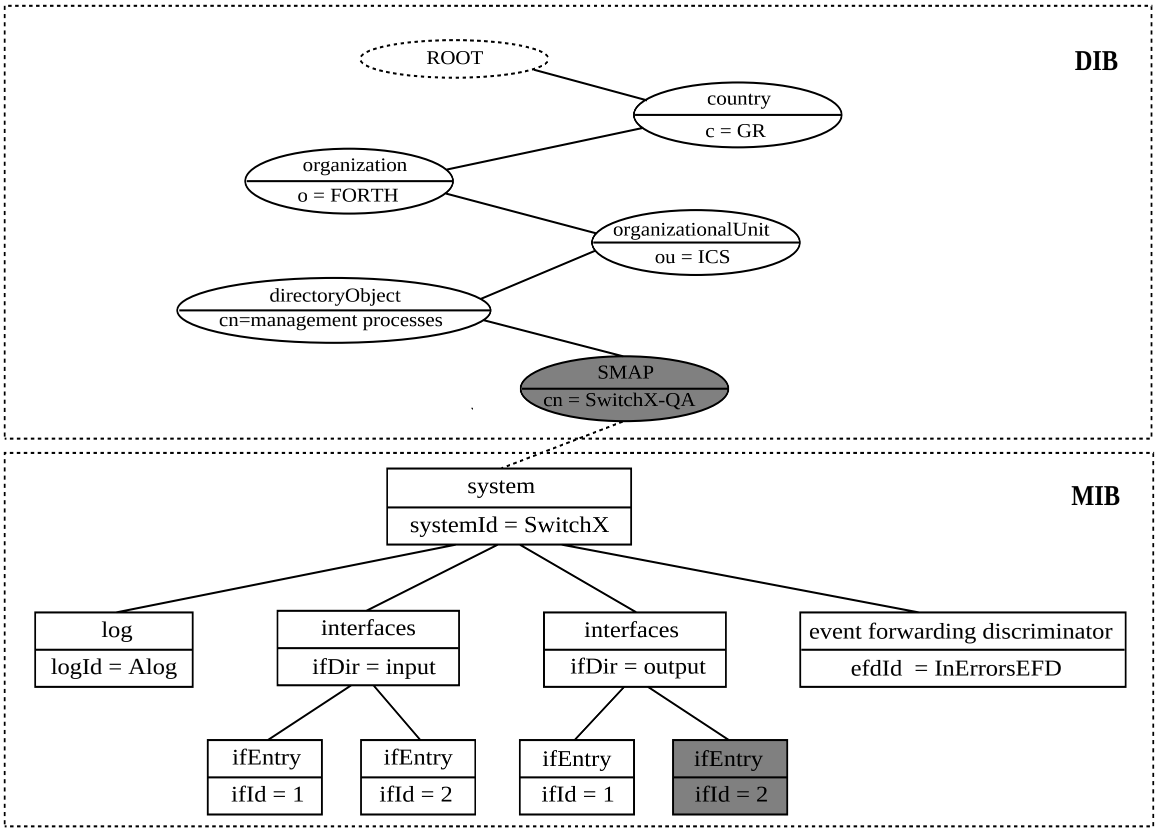
Το κεφάλαιο αυτό αποτελεί τον πυρήνα της παρούσας εργασίας. Περιγράφει το μοντέλο το οποίο προτείνουμε για την απόκτηση της Κοινής Γνώσης Διαχείρισης σε Δίκτυα Διαχείρισης Τηλεπικοινωνιών. Η βασική ιδέα στην οποία στηριχτήκαμε είναι ότι η ΚΓΔ αποτελεί – κατά κανόνα – σχετικά στατική πληροφορία, δηλαδή πληροφορία που δεν αλλάζει πολύ συχνά κατά τη διάρκεια λειτουργίας ενός ΔΔΤ. Για το λόγο αυτό επιλέξαμε να χρησιμοποιήσουμε την κατανεμημένη βάση δεδομένων της υπηρεσίας καταλόγου (X.500), τη δομή της οποίας ήδη περιγράψαμε στο προηγούμενο κεφάλαιο, για την αποθήκευση και διάθεση της ΚΓΔ.
Τα οφέλη που προσφέρει η παραπάνω επιλογή πηγάζουν από το αναμφισβήτητο γεγονός ότι το X.500 παρέχει ένα ισχυρό και καλώς ορισμένο μοντέλο πληροφορίας καθώς και ένα πλήρες σύνολο πρωτοκόλλων, ενώ συγχρόνως παρέχει ένα τυποποιημένο (εγκεκριμένο από τον *ISO* αλλά και την *ITU*) τρόπο αποθήκευσης και απόκτησης πληροφορίας όπως η ΚΓΔ. Η επιλογή μας αυτή δικαιώθηκε καθώς κατά τη διάρκεια εκπόνησης αυτής της εργασίας εκδόθηκε η αρχική πρόταση της τυποποίησης ISO-10164-16 στην οποία το X.500 χρησιμοποιείται για την αποθήκευση και ανάκτηση του μεγαλύτερου τμήματος της ΚΓΔ. Πρέπει να σημειωθεί ότι η παρούσα εργασία αποτέλεσε πηγή σχολίων τα οποία στάλθηκαν στην επιτροπή του *ISO* που είναι υπεύθυνη για την τελική προώθηση της ISO-10164-16 σε διεθνή τυποποίηση (International Standard) κάτι το οποίο πιθανόν να γίνει μέσα στο 1996. Ετσι, οι προσπάθειες μας επικεντρώθηκαν στη σχεδίαση ενός μοντέλου για την επικοινωνία διαχειριστή/αντιπροσώπου το οποίο θα αποτελεί επέκταση του υπάρχοντος συμπεριλαμβάνοντας τις βασικές λειτουργίες για την απόκτηση της ΚΓΔ χρησιμοποιώντας το

X.500.
Ενα από τα βασικά πλεονεκτήματα από τη χρήση του X.500 είναι ότι, από τη στιγμή που εγγράφουμε οντότητες καταλόγου οι οποίες αντιπροσωπεύουν διεργασίες (περιέχοντας κάποια πληροφορία γι'αυτές), κάθε διεργασία διαχείρισης αποκτά ένα μοναδικό όνομα με το οποίο μπορεί να αναγνωρίζεται από κάθε άλλη διεργασία οποιουδήποτε ΔΔΤ.
Η πρώτη ενότητα δείχνει πώς αυτό το γεγονός μπορεί να προσφέρει την ψευδαίσθηση μιας καθολικής βάσης πληροφορίας διαχείρισης (MIB). Στη δεύτερη ενότητα περιγράφουμε τις επεκτάσεις στο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου του *ISO* το οποίο, όπως προαναφέραμε, εμπλουτίζεται με τη δυνατότητα πρόσβασης στο X.500 (μέσω ενός ειδικού DUA) με σκοπό τη δυναμική ενημέρωση της ΚΓΔ και την ικανότητα απόκτησης της ΚΓΔ από κάθε διεργασία διαχείρισης (είτε αυτή ενεργεί σε ρόλο διαχειριστή, είτε σε ρόλο αντιπροσώπου).
Στη συνέχεια, η τρίτη ενότητα του κεφαλαίου παρουσιάζει τον αλγόριθμο της απόκτησης της ΚΓΔ. Στη γενική μορφή του, ο αλγόριθμος, δοθέντος ενός πόρου στον οποίο θέλουμε να εκτελέσουμε κάποια λειτουργία διαχείρισης, βρίσκει τις κατάλληλες διεργασίες – ή ακόμα και τις διαχειρίσιμες οντότητες που περιέχονται σε αυτές τις διεργασίες – οι οποίες σχετίζονται με την απαιτούμενη λειτουργία διαχείρισης στο δεδομένο πόρο. Σε αυτό το σημείο πρέπει να τονίσουμε ότι η πληροφορία σχετικά με το ποιος πόρος αντιπροσωπεύεται από ποια διεργασία περιγράφεται και αποκτάται σαν στοιχείο της ΚΓΔ. Αυτό αποτελεί μία σημαντικότατη καινοτομία σε σχέση με την τυποποίηση ISO-10164-16 καθώς και άλλες σχετικές εργασίες [TD93] οι οποίες δεν παρέχουν κάποιο παρόμοιο μηχανισμό αγνοώντας την ανάγκη απόκτησης της παραπάνω πληροφορίας.
Στην ενότητα 2.4 περιγράφεται ο μηχανισμός ο οποίος προσφέρει διαφάνεια ως προς την τοποθεσία των διεργασιών διαχείρισης. Τα παραπάνω (αλγόριθμος απόκτησης της ΚΓΔ και μηχανισμός διαφάνειας ως προς την τοποθεσία) εφαρμόζονται σε ένα παράδειγμα για την περίπτωση ενός απλού ΔΔΤ. Το κεφάλαιο κλείνει με την περιγραφή του καθολικού CMIS/CMIP το οποίο αποτελεί μία υπηρεσία η οποία είναι ταυτόσημη με αυτή του κλασσικού CMIS/CMIP με τη διαφορά ότι κατά την επικοινωνία διαχειριστή/αντιπροσώπου το στάδιο δημιουργίας της σύνδεσης διαχειριστή/αντιπροσώπου μένει αθέατο χάρη στο μηχανισμό διαφάνειας ως προς την τοποθεσία που βρίσκεται πίσω από τη χρήση καθολικών ονομάτων για τον προσδιορισμό της πληροφορίας διαχείρισης.

## 2.1 Η καθολική βάση πληροφορίας διαχείρισης

Οπως αναφέραμε στο προηγούμενο κεφάλαιο, το X.500 μπορεί να χρησιμοποιηθεί για την απόδοση μοναδικών ονομάτων στις διάφορες διεργασίες. Κάθε είδους διεργασία μπορεί να αντιπροσωπευθεί από μία οντότητα καταλόγου η οποία περιέχει πληροφορία για τη διεργασία με την προϋπόθεση ότι αυτή η πληροφορία είναι σχετικά στατική. Ετσι, κάθε διεργασία διαχείρισης η οποία ενεργεί είτε σε ρόλο διαχειριστή, είτε σε ρόλο αντιπροσώπου, είτε και στους δύο μπορεί να ονομαστεί με κάποιο διακεκριμένο όνομα (Distinguished Name – DN) το οποίο είναι το DN της οντότητας καταλόγου που της αντιστοιχεί. Λαμβάνοντας υπόψη ότι οι διαχειρίσιμες οντότητες χρησιμοποιούν έναν παρόμοιο ιεραρχικό τρόπο ονομασίας στις MIB με αυτόν που χρησιμοποιεί το X.500, μπορούμε να θεωρήσουμε ένα κοινό πεδίο ονομασίας τόσο για τις διαχειρίσιμες οντότητες όσο και για τις οντότητες καταλόγου [Org92b], [Syl93], [TD93].

Στο σχήμα 2.1 απεικονίζονται μία βάση πληροφορίας καταλόγου (DIB) η οποία αποτελείται από οντότητες καταλόγου που αναπαριστώνται με τους ελλειψοειδείς κόμβους του σχήματος και μία βάση πληροφορίας διαχείρισης (MIB) η οποία αποτελείται από διαχειρίσιμες οντότητες που αναπαριστώνται με τους ορθογώνιους κόμβους. Ας θεωρήσουμε τη διεργασία διαχείρισης η οποία αντιπροσωπεύεται από τη γραμμοσκιασμένη οντότητα καταλόγου του σχήματος 2.1 (γραμμοσκιασμένος ελλειψοειδής κόμβος) με διακεκριμένο όνομα
{**c=GR, o=FORTH, ou=ICS, cn=management processes, cn=SwitchX-QA**}.
Επίσης, ας θεωρήσουμε ότι η παραπάνω διεργασία περιέχει την MIB η οποία φαίνεται στο κάτω μέρος του σχήματος 2.1. Κάθε διαχειρίσιμη οντότητα η οποία περιέχεται στην MIB αυτή έχει όπως εξηγήσαμε στην ενότητα 1.1 ένα όνομα μοναδικό μέσα στη συγκεκριμένη MIB. Ετσι, για παράδειγμα, η γραμμοσκιασμένη διαχειρίσιμη οντότητα του σχήματος 2.1 (γραμμοσκιασμένος ορθογώνιος κόμβος) έχει όνομα:
{**systemId = SwitchX, ifDir = output, ifId = 2**}.
Ας σημειωθεί ότι το όνομα αυτό δεν είναι μοναδικό έξω από τη συγκεκριμένη MIB. Πράγματι, δεν μπορούμε να εξασφαλίσουμε ότι κανείς άλλος μεταγωγέας (ή, γενικά, πόρος) σε κανένα άλλο μέρος του κόσμου δεν θα λέγεται “switchX” και επιπλέον είναι πιθανό κάποια διεργασία η οποία αντιπροσωπεύει κάποιον πόρο που επίσης ονομάζεται “switchX” να υποστηρίζει την ίδια – ή παρόμοια – MIB με αυτή του σχήματος 2.1· έτσι, είναι δυνατό μία διαχειρίσιμη οντότητα σε κάποια άλλη διεργασία να έχει ακριβώς το ίδιο όνομα με τη γραμμοσκιασμένη διαχειρίσιμη οντότητα του σχήματος 2.1.




Σχήμα 2.1: Η καθολική ονομασία των διαχειρίσιμων οντοτήτων.

---

Από τη στιγμή όμως που έχουμε τη δυνατότητα αναγνώρισης τής κάθε διεργασίας διαχείρισης με ένα παγκόσμια μοναδικό όνομα – χάρη στο X.500 – μπορούμε να βρούμε ένα μοναδικό όνομα για κάθε διαχειρίσιμη οντότητα το οποίο θα την αναγνωρίζει και έξω από την εκάστοτε MIB. Η μέθοδος η οποία ακολουθούμε είναι πολύ απλή: κάθε διαχειρίσιμη οντότητα αποκτά ένα διακεκριμένο όνομα το οποίο δημιουργείται από τη συνένωση του ονόματος της διεργασίας που περιέχει την οντότητα και του ονόματος της οντότητας μέσα στην MIB της. Ετσι, το DN της γραμμοσκιασμένης οντότητας του σχήματος 2.1 είναι:
{**c=GR, o=FORTH, ou=ICS, cn=management processes, cn=SwitchX-QA, systemId = SwitchX, ifDir = output, ifId = 2**}.
Η διακεκομμένη γραμμή του σχήματος 2.1 απεικονίζει την παραπάνω συνένωση των δύο ονομάτων.

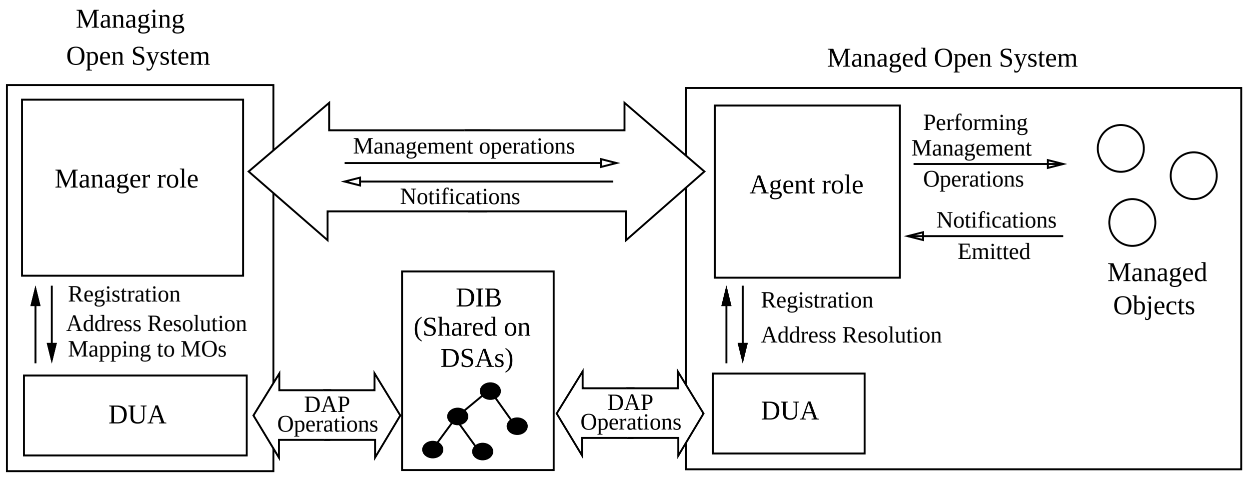
Κατ' επέκταση, μπορούμε να θεωρήσουμε ότι όλες οι επιμέρους MIB – που περιέχονται στις διεργασίες που ενεργούν σε ρόλο αντιπροσώπου και έχουν εγγραφεί στο X.500 – συνδέονται κάτω από το ιδεατό δέντρο της πληροφορίας του X.500 σχηματίζοντας μία **καθολική βάση πληροφορίας διαχείρισης** στην οποία κάθε διαχειρίσιμη οντότητα αναγνωρίζεται με ένα καθολικά διακεκριμένο όνομα. Σε συμφωνία με την ορολογία του X.500 θα καλούμε το όνομα αυτό απλά διακεκριμένο όνομα (DN) της διαχειρίσιμης οντότητας και θα το ξεχωρίζουμε από το όνομα της διαχειρίσιμης οντότητας μέσα στην MIB το οποίο θα καλούμε τοπικά διακεκριμένο όνομα (local DN)[1].

## 2.2 Το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου

Στο σχήμα 2.2 απεικονίζεται το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου όπως επεκτάθηκε με σκοπό την προσφορά της δυνατότητας ενημέρωσης και απόκτησης της ΚΓΔ. Η βασική διαφορά σε σχέση με το μοντέλο που προτάθηκε από τον *ISO* και παρουσιάσαμε στο σχήμα 1.1 είναι η προσθήκη ενός DUA (Directory User Agent) και στα δύο συστήματα του σχήματος (τα οποία στην πραγματικότητα αντιστοιχούν σε δύο διεργασίες διαχείρισης). Η βασική λειτουργία του DUA είναι η επικοινωνία με τους διάφορους DSA της υπηρεσίας καταλόγου με σκοπό την ανάκτηση και ενημέρωση της πληροφορίας που βρίσκεται στον Κατάλογο και που αποτελεί τμήμα της ΚΓΔ. Η παραπάνω επικοινωνία γίνεται με τη χρήση του πρωτοκόλλου DAP όπως προαναφέραμε στην ενότητα 1.3.
Οπως φαίνεται και από το σχήμα 2.2 κάθε διεργασία, η οποία λειτουργεί είτε σε ρόλο διαχειριστή είτε σε ρόλο αντιπροσώπου, χρησιμοποιεί τον DUA της για τα εξής:

- **Ενημέρωση του Καταλόγου.** Κάθε διεργασία κατά την έναρξη της λειτουργίας της ενημερώνει την πληροφορία στις οντότητες του καταλόγου που αντιστοιχούν σε αυτήν και στις SMAE που περιέχει. Η βασική προϋπόθεση είναι να ξέρει η διεργασία τα διακεκριμένα ονόματα (DN) αυτών των οντοτήτων καταλόγου. Ετσι, η κάθε διεργασία είναι υπεύθυνη για την ενημέρωση του τμήματος της ΚΓΔ το οποίο σχετίζεται άμεσα με αυτήν. Στην επόμενη ενότητα θα αναφερθούμε λεπτομερώς στο ποια ακριβώς πληροφορία αποτελεί αυτή τη ΚΓΔ η οποία φυλάσσεται στις οντότητες καταλόγου που

[1] Η ορολογία αυτή χρησιμοποιείται και στην τυποποίηση ISO-10165-1 [Org92b].




Σχήμα 2.2: Το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου.

---

αντιστοιχούν είτε σε διεργασίες διαχείρισης είτε σε SMAE. Να σημειωθεί ότι παρόλο που οι DUA του σχήματος 2.2 περιλαμβάνουν και τη δυνατότητα ενημέρωσης της πληροφορίας που φυλάσσεται στις οντότητες καταλόγου που αντιστοιχούν σε πόρους (πληροφορία που επίσης ανήκει στη ΚΓΔ), όπως θα εξηγήσουμε στην επόμενη παράγραφο, η πληροφορία αυτή ενημερώνεται κατά τη διάρκεια της διαμόρφωσης (configuration) του ΔΔΤ. Η ενημέρωση αυτή αποτελεί τμήμα μίας από τις βασικές υπηρεσίες διαχείρισης η οποία καλείται "διαχείριση του ΔΔΤ" ή "μετα-διαχείριση" (metamanagement). Ετσι, μόνο οι διεργασίες οι οποίες παρέχουν την υπηρεσία της μετα-διαχείρισης ενημερώνουν τις οντότητες καταλόγου που αντιστοιχούν σε πόρους του διαχειριζόμενου δικτύου. Επίσης, όπως θα εξηγήσουμε στο κεφάλαιο 3, ενημέρωση του Καταλόγου γίνεται και στην περίπτωση τερματισμού της λειτουργίας κάποιας διεργασίας προκειμένου να είναι πάντα γνωστό ποιες από τις εγγεγραμμένες διεργασίες είναι διαθέσιμες και λειτουργούν κανονικά.

- **Εύρεση διευθύνσεων.** Οπως έχουμε ήδη αναφέρει, ένα από τα βασικά χαρακτηριστικά του ΔΔΤ είναι ότι οι περισσότερες διεργασίες είναι μετακινήσιμες, δηλαδή μπορούν να μετακομίζουν από μία διεύθυνση σε μία άλλη προσφέροντας έτσι ανοχή σε βλάβες και ευελιξία στη διαμόρφωση του ΔΔΤ. Για κάθε διεργασία διαχείρισης η μοναδική πληροφορία που είναι στατική και δεν αλλάζει ποτέ είναι το διακεκριμένο όνομα της με το οποίο αυτή αναγνωρίζεται. Ετσι, το ερώτημα “ποια είναι η διεργασία που αντιπροσωπεύει κάποιο πόρο” είναι τελείως διαφορετικό από το ερώτημα “ποια είναι η διεύθυνση στην οποία πρέπει να συνδεθούμε προκειμένου να επικοινωνήσουμε με την παραπάνω διεργασία”. Το πρώτο αναφέρεται στο διακεκριμένο όνομα της διεργασίας το οποίο δεν αλλάζει κατά την διάρκεια λειτουργίας ενός ΔΔΤ· το δεύτερο αναφέρεται σε μία διεύθυνση η οποία μπορεί να αλλάξει αρκετές φορές κατά τη διάρκεια λειτουργίας ενός ΔΔΤ. Η λειτουργία της εύρεσης διευθύνσεων ασχολείται με την απάντηση του δεύτερου ερωτήματος, δοθέντος του DN μίας διεργασίας. Ο μηχανισμός διαφάνειας ως προς την τοποθεσία (location transparency) που περιγράφουμε στην ενότητα 2.4 εξηγεί με λεπτομέρεια αυτήν τη λειτουργία.

- **Εύρεση των κατάλληλων διαχειρίσιμων οντοτήτων.** Η λειτουργία αυτή αφορά μόνο διεργασίες που ενεργούν σε ρόλο διαχειριστή, ο σκοπός των οποίων – όπως έχουμε ήδη περιγράψει – είναι η αποστολή πράξεων διαχείρισης σε αντιπροσώπους οι οποίοι περιέχουν κάποιες επιλεγμένες διαχειρίσιμες οντότητες. Στη γενική περίπτωση ένας διαχειριστής δεν ξέρει εκ των προτέρων ποιες ακριβώς είναι οι οντότητες τις οποίες θέλει να διαχειριστεί· με άλλα λόγια, δεν έχει τη δυνατότητα να χρησιμοποιήσει άμεσα κάποια συγκεκριμένα ονόματα οντοτήτων. Οι λόγοι για κάτι τέτοιο μπορεί να είναι πολλοί. Για παράδειγμα, μπορεί να μην ξέρει την ακριβή μορφή της MIB που κρατάει ο αντιπρόσωπος με τον οποίο πρέπει να επικοινωνήσει (και συνεπώς δεν μπορεί να ξέρει τα ονόματα των διαχειρίσιμων οντοτήτων) ή δεν ξέρει καν το όνομα του αντιπροσώπου με τον οποίο πρέπει να επικοινωνήσει για να εκτελέσει κάποια συγκεκριμένη λειτουργία διαχείρισης. Σε πολλές περιπτώσεις γνωρίζει μόνο το όνομα του πόρου τον οποίο επιθυμεί να διαχειριστεί.

  Ετσι, η εύρεση των κατάλληλων διαχειρίσιμων οντοτήτων είναι μία αναγκαία και κρίσιμη λειτουργία η οποία, όπως θα δούμε στην επόμενη ενότητα, προϋποθέτει την απόκτηση της ΚΓΔ με τελικό σκοπό την ανεύρεση των ονομάτων που θα χρησιμοποιηθούν για την πρόσβαση στις επιθυμητές διαχειρίσιμες οντότητες κάποιου αντιπροσώπου.

## 2.3 Χρησιμοποιώντας τη Κοινή Γνώση Διαχείρισης

Οπως εξηγήσαμε και στο πρώτο κεφάλαιο στην υποενότητα 1.4.2, κατά τη διαχείριση των δικτύων από συστήματα όπως το ΔΔΤ, αυτό που συνήθως είναι γνωστό στο διαχειριστή – είτε αυτός είναι κάποιος άνθρωπος, είτε κάποια διεργασία που λειτουργεί σε ρόλο διαχειριστή – είναι κάποιος πόρος (ή κάποια υπηρεσία) ο οποίος αποτελεί το τελικό αντικείμενο διαχείρισης. Ετσι, δοθέντου κάποιου πόρου (ή κάποιας υπηρεσίας), μία σημαντική εργασία κάθε διαχειριστή είναι η εύρεση της κατάλληλης διεργασίας που λειτουργεί σε ρόλο αντιπροσώπου και περιέχει τις διαχειρίσιμες οντότητες – που αντιπροσωπεύουν τον πόρο (ή την υπηρεσία) – πάνω στις οποίες πρέπει να γίνουν κάποιες πράξεις.

Στη γενική περίπτωση, ένας πόρος μπορεί να αντιπροσωπεύεται από πολλές διαχειρίσιμες οντότητες και μία διαχειρίσιμη οντότητα μπορεί να αντιπροσωπεύει περισσότερους από έναν πόρους. Λαμβάνοντας αυτό υπόψη, καθώς και το μέγεθος των συστημάτων διαχείρισης σαν το ΔΔΤ, η παραπάνω αντιστοίχιση αποτελεί μία πολύπλοκη διαδικασία και απαιτεί κάποιες αποφάσεις οι οποίες στηρίζονται στην κατοχή της κατάλληλης ΚΓΔ από τις διάφορες διεργασίες. Πληροφορία όπως το ποιος πόρος αντιπροσωπεύεται από ποιον αντιπρόσωπο διαχείρισης, οι λειτουργίες διαχείρισης που μπορεί να επιτελέσει αυτός ο αντιπρόσωπος και το είδος της πληροφορίας που κρατάει, ανήκει στην ΚΓΔ και, προφανώς, είναι αναγκαία προκειμένου να γίνουν οι σωστές αντιστοιχίσεις.

Στην ενότητα 1.4 προσδιορίσαμε με λεπτομέρεια τη γνώση στην οποία αναφέρεται η ΚΓΔ. Στην παρούσα εργασία χρησιμοποιούμε το X.500 για την αποθήκευση, ανάκτηση και ενημέρωση ενός βασικού υποσυνόλου της γνώσης αυτής. Αυτό το υποσύνολο της ΚΓΔ περιλαμβάνει την πληροφορία που χρειάζεται προκειμένου να γίνονται οι αντιστοιχίσεις από πόρους σε διαχειρίσιμες οντότητες οι οποίες αποτελούν αφαιρέσεις των πόρων αυτών για τους σκοπούς της διαχείρισης. Στην παρούσα ενότητα περιγράφουμε αυτή τη βασική λειτουργία η οποία αποτελεί το σημαντικότερο κομμάτι της εργασίας μας και η οποία βασίζεται στα στοιχεία που συμπεριλάβαμε στη ΚΓΔ στην υποενότητα 1.4.2.

Προτού περιγράψουμε με λεπτομέρεια πώς γίνεται η παραπάνω αντιστοίχιση – η οποία μάλιστα εκτελείται με δυναμικό τρόπο (δηλαδή, κατά τη διάρκεια λειτουργίας του ΔΔΤ) – κάνουμε μία αναλυτική παράθεση της πληροφορίας που φυλάσσουμε στον Κατάλογο και η οποία συνιστά το παραπάνω υποσύνολο της ΚΓΔ. Η πληροφορία αυτή αφορά τόσο σε πόρους του διαχειριζόμενου δικτύου όσο και σε διεργασίες του ΔΔΤ (συμπεριλαμβανομένων και των SMAE που αυτές περιέχουν). Ας σημειωθεί ότι η πληροφορία για τις προσφερόμενες υπηρεσίες του δικτύου – οι οποίες φυσικά είναι αντικείμενο διαχείρισης – παραλείφθηκε για λόγους απλότητας και χωρίς αυτό να μειώνει τη γενικότητα της μεθόδου μας. Προφανώς, παρόμοια πληροφορία με αυτή που κρατιέται για τους διάφορους πόρους του δικτύου μπορεί να φυλαχτεί και για τις υπηρεσίες.

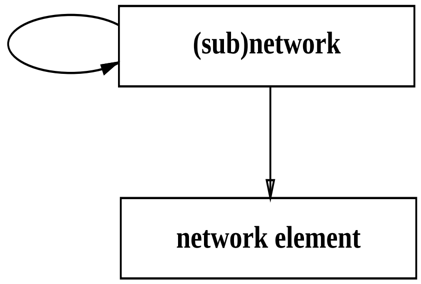
### 2.3.1 Πληροφορία για πόρους και διεργασίες στο X.500

**Πόροι του διαχειριζόμενου δικτύου τηλεπικοινωνιών**

Στην παρούσα αναφορά χρησιμοποιούμε τον όρο "πόροι δικτύου" ή, απλά, "πόροι" για να αναφερθούμε είτε σε στοιχεία του δικτύου (π.χ. μεταγωγείς) είτε σε ομάδες αυτών (δηλ. υποδίκτυα και δίκτυα). Ο παραπάνω απλός διαχωρισμός των διαφόρων πόρων αντανακλά τα δύο επίπεδα διαχείρισης του ΔΔΤ (επίπεδο διαχείρισης στοιχείων του δικτύου και επίπεδο διαχείρισης δικτύου) που ασχολούνται άμεσα με τη διαχείριση πόρων οι οποίοι αποτελούν είτε μεμονωμένα στοιχεία του δικτύου, είτε ολόκληρα δίκτυα (ή υποδίκτυα). Σύμφωνα με τα παραπάνω, οι πόροι του δικτύου και οι σχέσεις τους μπορούν να αναπαρασταθούν με το διάγραμμα του σχήματος 2.3 στο οποίο τα διάφορα βέλη αντιπροσωπεύουν μία σχέση "εγκλεισμού". Ετσι κάθε ορθογώνιο του σχήματος 2.3 δηλώνει μία κατηγορία πόρων ενώ από τα υπάρχοντα βέλη συνάγουμε ότι ένας πόρος του τύπου "δίκτυο" (network) μπορεί να περιέχει:

- είτε πόρους του τύπου "στοιχείο δικτύου" που είναι ο πιο απλός τύπος πόρου,
- είτε απλούστερους πόρους του τύπου "δίκτυο".

Το X.500 μπορεί να χρησιμοποιηθεί για την αποθήκευση πληροφορίας σχετικής με πόρους μέσω της εγγραφής μίας οντότητας καταλόγου για κάθε πόρο. Λαμβάνοντας υπόψη την απλή σχέση των διαφόρων πόρων όπως αυτή δόθηκε στο σχήμα 2.3 μπορούμε να δομήσουμε με κατάλληλο ιεραρχικό τρόπο την πληροφορία του Καταλόγου ώστε κάθε πόρος να ονομάζεται με ένα μοναδικό όνομα. Στο παράδειγμα του σχήματος 1.8 φαίνονται οι οντότητες που αντιστοιχούν σε τρεις πόρους (ο ένας αντιστοιχεί σε ένα δίκτυο ενώ οι άλλοι δύο αντιστοιχούν σε στοιχεία του δικτύου όπως άλλωστε φαίνεται από τις κλάσεις στις οποίες ανήκουν). Η ιεραρχική οργάνωση των οντοτήτων υποδηλώνει και τη σχέση των πόρων που αντιπροσωπεύουν. Ετσι, είναι φανερό ότι το δίκτυο με RDN {**cn=switching network**} περιέχει δύο στοιχεία με RDN {**cn=SwitchX**} και {**cn=SwitchY**}.




Σχήμα 2.3: Η σχέση των διαφόρων πόρων.

Ας σημειωθεί ότι ο παραπάνω ιεραρχικός τρόπος με τον οποίο εγγράφουμε πόρους στον Κατάλογο, αποτελεί μία πολύ απλή μέθοδο η οποία δεν δίνει πολλές πληροφορίες για την τοπολογία του δικτύου. Στη γενική περίπτωση – η οποία είναι πιο κοντά στην πραγματικότητα – οι διασυνδέσεις των διαφόρων πόρων και υποδικτύων σχηματίζουν μάλλον ένα πολύπλοκο γράφημα παρά ένα δέντρο. Για παράδειγμα, ένας κόμβος του δικτύου μπορεί να ανήκει σε περισσότερα από ένα υποδίκτυα. Στην παρούσα εργασία δεν ασχοληθήκαμε με το θέμα της περιγραφής της τοπολογίας του δικτύου από την πληροφορία που φυλάσσουμε στον κατάλογο αλλά με την παροχή της γνώσης του ποιος πόρος αντιπροσωπεύεται από ποιες διεργασίες. Στην εργασία [MJK94] υπάρχει μία πιο ολοκληρωμένη μεθοδολογία στην οποία η πληροφορία του καταλόγου και η οργάνωσή της αντανακλά τους διάφορους τρόπους διασύνδεσης των πόρων και των υποδικτύων (δηλ. αντανακλά την τοπολογία) του εγγεγραμμένου δικτύου. Η πληροφορία που περιγράφουμε στη συνέχεια μπορεί να προστεθεί και στην περίπτωση που ακολουθείται η μέθοδος τής [MJK94] για την εγγραφή πόρων και δικτύων στον Κατάλογο. Για κάθε πόρο κρατάμε την ακόλουθη πληροφορία στη οντότητα καταλόγου που τον αντιπροσωπεύει (η πληροφορία αυτή αποτελεί ένα αναπόσπαστο τμήμα της ΚΓΔ):

- το όνομα (DN) μίας διεργασίας η οποία είναι υπεύθυνη για την παροχή ορισμένης υπηρεσίας διαχείρισης (Management Service) σε αυτόν τον πόρο και
- το όνομα αυτής της υπηρεσίας διαχείρισης·
- όλα τα παραπάνω ζευγάρια διεργασίας/υπηρεσίας διαχείρισης που σχετίζονται με το συγκεκριμένο πόρο.

Κατά τη σχεδίαση ενός ΔΔΤ αποφασίζεται η αντιστοιχία μεταξύ διεργασιών και διαχειριζόμενων πόρων. Βέβαια είναι δυνατό η αντιστοιχία αυτή να αλλάξει κατά τη διάρκεια λειτουργίας του ΔΔΤ αλλά αυτό θα συμβεί μόνο κατόπιν μερικής επανασχεδίασης του ΔΔΤ. Κάθε διεργασία του ΔΔΤ, κατά κανόνα, παρέχει κάποια υπηρεσία διαχείρισης του ΔΔΤ. Η σύσταση M.3200 [Uni93f] ορίζει τις 19 υπηρεσίες διαχείρισης τις οποίες παρέχουν οι διάφορες διεργασίες[2] στο ΔΔΤ. Κάθε διεργασία που ενεργεί σε ρόλο αντιπροσώπου θα μπορούσε να χρησιμοποιεί την λειτουργία ενημέρωσης της ΚΓΔ του μοντέλου που περιγράψαμε στην προηγούμενη ενότητα προκειμένου να ενημερώνει την παραπάνω πληροφορία στις οντότητες καταλόγου που αντιστοιχούν στους πόρους που αντιπροσωπεύει. Αυτή η μέθοδος θα μπορούσε να δημιουργήσει πολλά προβλήματα ακεραιότητας της ΚΓΔ λόγω του γεγονότος ότι η πληροφορία σε μία οντότητα καταλόγου στη γενική περίπτωση θα ενημερωνόταν από πολλές διεργασίες συγχρόνως (εφόσον ένας πόρος γενικά αντιπροσωπεύεται από πολλές διεργασίες). Η πιο σωστή επιλογή είναι να υπάρχουν ειδικές διεργασίες οι οποίες θα αναλαμβάνουν την ενημέρωση της παραπάνω πληροφορίας τόσο κατά τη διαμόρφωση (configuration) όσο και σε περιπτώσεις επαναδιαμόρφωσης (reconfiguration) του ΔΔΤ. Οι διεργασίες αυτές παρέχουν μία ειδική υπηρεσία διαχείρισης η οποία καλείται "μετα-διαχείριση" (metamanagement). Η μετα-διαχείριση αναφέρεται στη διαχείριση του ιδίου του ΔΔΤ και αφορά σε λειτουργίες όπως η αρχική τοποθέτηση και ενεργοποίηση των διεργασιών διαχείρισης στα διάφορα συστήματα του ΔΔΤ, η μετακίνηση των διεργασιών σε ειδικές περιπτώσεις (π.χ. βλάβες του ΔΔΤ) και η επιβολή κάποιου ελέγχου πρόσβασης στην πληροφορία διαχείρισης. Ετσι, η ευθύνη για την ενημέρωση της πληροφορίας των οντοτήτων του καταλόγου που αντιστοιχούν σε πόρους δίνεται στο υποσύστημα μετα-διαχείρισης του ΔΔΤ. Η εργασία [SKGS95] περιγράφει τον τρόπο με τον οποίο επιτελούνται μερικές λειτουργίες της μετα-διαχείρισης σε ένα ΔΔΤ.

### Διεργασίες του ΔΔΤ

Κάθε διεργασία διαχείρισης αντιπροσωπεύεται από μία οντότητα καταλόγου η οποία περιέχει γενική πληροφορία για τη συγκεκριμένη διεργασία. Η πληροφορία αυτή αποτελεί επίσης αναπόσπαστο τμήμα της ΚΓΔ και είναι η παρακάτω:

- η υπηρεσία διαχείρισης που παρέχει η διεργασία στο ΔΔΤ·
- το δομικό στοιχείο του ΔΔΤ που αποτελεί η διεργασία (π.χ. OS)· και,

[2] Για την ακρίβεια, τα OS είναι οι διεργασίες οι οποίες παρέχουν συγκεκριμένες υπηρεσίες διαχείρισης. Οι υπόλοιπες διεργασίες συνήθως δεν ασχολούνται με κάποια συγκεκριμένη υπηρεσία διαχείρισης.

- σε περίπτωση που η διεργασία ενεργεί σε ρόλο αντιπροσώπου, η MIB την οποία υποστηρίζει. Αυτή αντιστοιχεί είτε σε κάποια γνωστή MIB[3] είτε σε μία περιγραφή κάποιου δέντρου ονομασίας το οποίο θα περιγράφει τις κλάσεις διαχειρίσιμων οντοτήτων που περιέχει η MIB και τη σχέση τους

Προαναφέραμε ότι κάθε διεργασία επικοινωνεί με άλλες διεργασίες του ΔΔΤ μέσω κάποιων οντοτήτων εφαρμογής (application entities) – ή, αλλιώς, SMAE – τις οποίες περιέχει. Κατά κανόνα, κάθε διεργασία διαχείρισης μπορεί να περιέχει είτε μία είτε δύο τέτοιες SMAE (ανάλογα με τους ρόλους στους οποίους ενεργεί η διεργασία). Η πληροφορία η οποία σχετίζεται με αυτές είναι πολύ σημαντική και φυλάσσεται στις ειδικές οντότητες καταλόγου οι οποίες αντιπροσωπεύουν τις SMAE. Κάθε οντότητα καταλόγου για κάποια SMAE εγγράφεται μέσα στην ιεραρχική δομή της DIB κάτω από την οντότητα καταλόγου που αντιπροσωπεύει τη διεργασία διαχείρισης η οποία περιέχει την SMAE. Ενα παράδειγμα αυτής της τακτικής φαίνεται πάλι στο σχήμα 1.8 όπου κάτω από τη διεργασία με RDN {**cn=SwitchX-QA**} είναι εγγεγραμμένη η SMAE με RDN {**cn=QA**}. Η πληροφορία που περιέχει κάθε οντότητα που αντιπροσωπεύει μία SMAE είναι η παρακάτω:

- η διεύθυνση (PSAP) στην οποία "ακούει" η SMAE για αιτήσεις πράξεων διαχείρισης (στην περίπτωση αντιπροσώπου) ή για ειδοποιήσεις (στην περίπτωση διαχειριστή)·

- οι λειτουργίες και εκδόσεις των διαφόρων πρωτοκόλλων επικοινωνίας και υπηρεσιών που υποστηρίζει η SMAE·

- το πλαίσιο της εφαρμογής (application context) που υποστηρίζει η SMAE·

- η λίστα των διαχειρίσιμων οντοτήτων που είναι διαθέσιμες μέσω της συγκεκριμένης SMAE· και

- ο ρόλος (διαχειριστή ή αντιπροσώπου) στον οποίο ενεργεί η διεργασία μέσω της συγκεκριμένης SMAE.

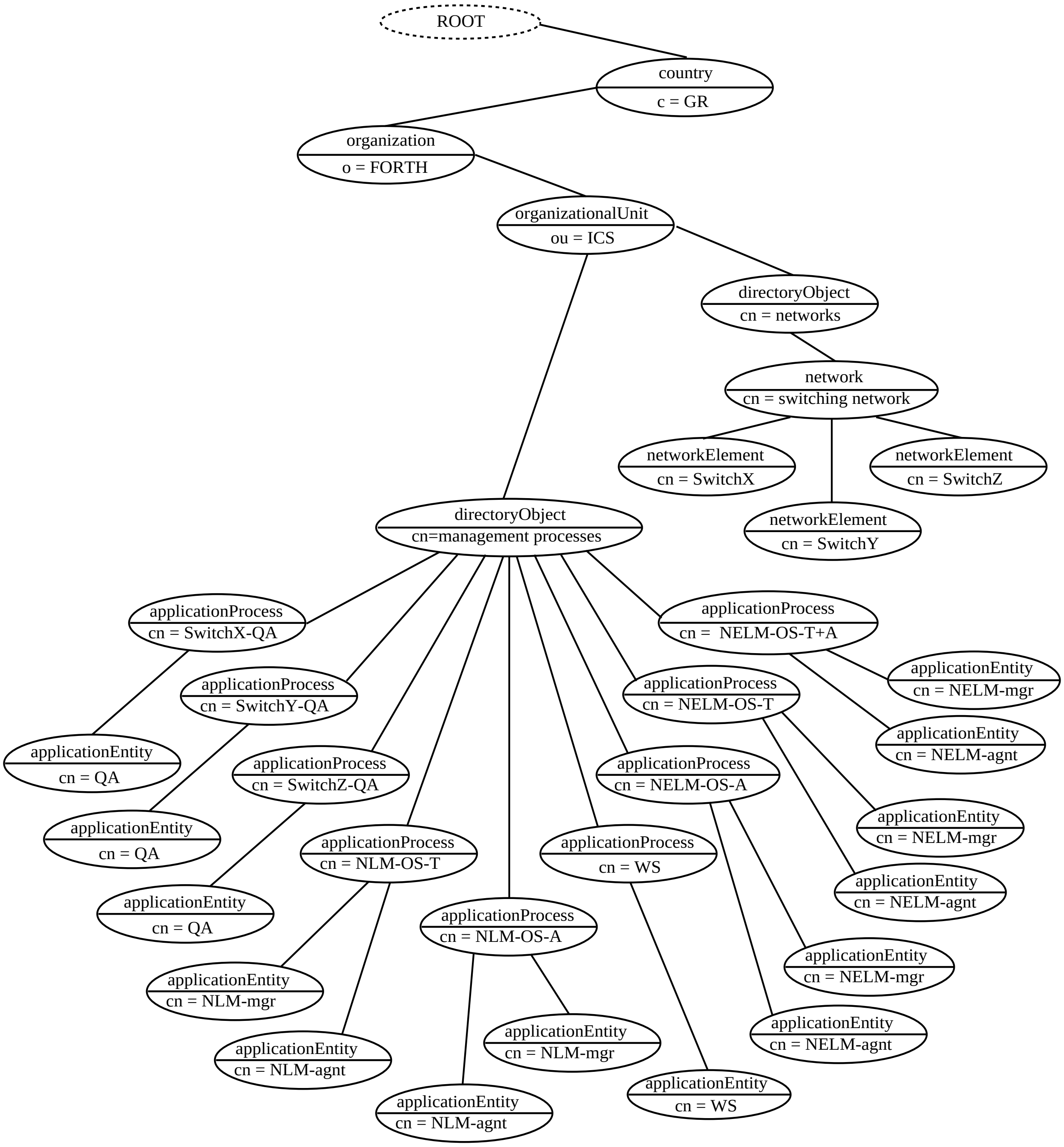
Τα παραπάνω δύο είδη οντοτήτων καταλόγου (δηλ. για διεργασίες και SMAE) περιέχουν το βασικό τμήμα της ΚΓΔ που απαιτήθηκε για τους σκοπούς της παρούσας εργασίας και ειδικότερα για την αντιστοίχιση από πόρους σε διαχειρίσιμες οντότητες και για την επίτευξη

[3] Μελλοντικά, αναμένεται να οριστούν MIB οι οποίες θα τυποποιηθούν από την *ITU* και θα χρησιμοποιούνται για τη διαχείριση κάποιου είδους πόρων (π.χ. μεταγωγέων ATM – Ασύγχρονου Τρόπου Μετάδοσης).

του μηχανισμού της διαφάνειας ως προς την τοποθεσία. Η πληροφορία αυτή ενημερώνεται κατά την έναρξη της λειτουργίας μίας διεργασίας από τον DUA που αυτή περιέχει (βλ. σχήμα 2.2). Πριν περιγράψουμε τη μέθοδο της αντιστοίχισης από πόρους σε διαχειρίσιμες οντότητες δίνουμε ένα παράδειγμα της δομής της DIB που περιέχει την αναγκαία πληροφορία τόσο για τους πόρους όσο και για τις διεργασίες στο περιβάλλον διαχείρισης του σχήματος 1.6.

### Ενα παράδειγμα μιας DIB

Στο σχήμα 2.4 απεικονίζουμε την ακριβή δομή της DIB η οποία περιέχει τόσο τις οντότητες καταλόγου για τους πόρους του διαχειριζόμενου δικτύου του σχήματος 1.6 όσο και τις οντότητες καταλόγου για τις διεργασίες καθώς και τις αντίστοιχες SMAE του ΔΔΤ που περιγράψαμε στην §1.2.4. Οπως μπορούμε να δούμε στο σχήμα 2.4 κάθε διεργασία περιέχει είτε μία είτε δύο SMAE ανάλογα με τους ρόλους στους οποίους ενεργεί.
Κάθε μία από τις απεικονιζόμενες οντότητες καταλόγου έχει ένα μοναδικό όνομα (DN). Πρέπει να διευκρινιστεί ότι το όνομα αυτό (ειδικότερα, το τελευταίο RDN του ονόματος), στην περίπτωση των διεργασιών διαλέγεται μάλλον αυθαίρετα παρά βάσει κάποιου κανόνα. Στην περίπτωση των πόρων υπάρχει ο εξής απλός κανόνας: το RDN σχηματίζεται βάσει του ευρέως χρησιμοποιούμενου ονόματος του μηχανήματος ή του δικτύου. Παρόλα αυτά, κατά την εγγραφή των διαφόρων διεργασιών στο X.500 πρέπει να ληφθούν σοβαρά υπόψη τα πλεονεκτήματα που έχουμε από την επιλογή μνημονικών και συμβολικών ονομάτων. Για παράδειγμα, το RDN {**cn=SwitchX-QA**} αποτελεί ένα μάλλον καλό όνομα για τη διεργασία που αποτελεί ένα QA για το μεταγωγέα *SwitchX*.




Σχήμα 2.4: Οντότητες για πόρους και διεργασίες στο X.500.

### 2.3.2 Αντιστοίχιση διαχειρίσιμων οντοτήτων σε πόρους

Οπως έχουμε ήδη αναφέρει, η σχέση μεταξύ διεργασιών του ΔΔΤ και πόρων που αυτές αντιπροσωπεύουν είναι αυθαίρετη· με άλλα λόγια, όχι μόνο ένας πόρος μπορεί να αντιπροσωπεύεται από πολλές διεργασίες αλλά και μία διεργασία μπορεί να αντιπροσωπεύει πολλούς πόρους. Η παρούσα υποενότητα περιγράφει έναν αλγόριθμο ο οποίος χρησιμοποιεί την πληροφορία της ΚΓΔ που αναφέραμε παραπάνω προκειμένου να εκτελέσει την εξής αντιστοίχιση: Με την προϋπόθεση ότι αρχικά ξέρουμε το DN ενός πόρου στον οποίο θέλουμε να εκτελεστεί κάποια συγκεκριμένη λειτουργία διαχείρισης (management function) – σαν μέρος κάποιας συνολικής υπηρεσίας διαχείρισης – θέλουμε να βρούμε τη διεργασία η οποία κρατάει τις απαραίτητες για τη λειτουργία αυτή διαχειρίσιμες οντότητες και τα ακριβή ονόματα (δηλ. τα DN) των οντοτήτων αυτών.
Προφανώς, η παραπάνω αντιστοίχιση αφορά διαχειριστές οι οποίοι θέλουν να επικοινωνήσουν με κάποιους αντιπροσώπους διαχείρισης που αντιπροσωπεύουν κάποιο διαχειριζόμενο πόρο. Τα αρχικά στοιχεία που έχει ο διαχειριστής (είτε αυτός είναι κάποια διεργασία είτε κάποιος άνθρωπος που χρησιμοποιεί κάποιο WS) είναι τα εξής:

- το DN του πόρου που θέλει να διαχειριστεί·
- την υπηρεσία διαχείρισης την οποία παρέχει καθώς και τη λειτουργία διαχείρισης (μέρος της υπηρεσίας διαχείρισης) που θέλει να εκτελέσει. Σε αυτό το σημείο πρέπει να πούμε ότι, για κάθε υπηρεσία διαχείρισης, οι συστάσεις της *ITU* ορίζουν ένα σύνολο από λειτουργίες διαχείρισης τις οποίες περιλαμβάνει η υπηρεσία διαχείρισης [Uni93f]. Ας υπενθυμίσουμε ότι κάθε λειτουργία διαχείρισης αποτελείται από κάποιες πράξεις διαχείρισης σε ένα σύνολο από διαχειρίσιμες οντότητες (συγκεκριμένων κλάσεων)· και
- κάποια περιγραφή των διαχειρίσιμων οντοτήτων στις οποίες πρέπει να εκτελεστεί η παραπάνω λειτουργία. Στη γενικότερη περίπτωση, η περιγραφή αυτή δεν εξαρτάται από κάποια MIB αλλά αποτελεί μία περιγραφή υψηλού επιπέδου. Για παράδειγμα, μπορεί να προσδιορίζει την κλάση στην οποία ανήκουν οι ζητούμενες διαχειρίσιμες οντότητες. Εννοείται ότι ο διαχειριστής πρέπει να έχει ένα τρόπο να προσδιορίζει μεταξύ των πολλών οντοτήτων μίας κλάσης, που μπορεί να περιέχει κάποια MIB, αυτές που τον ενδιαφέρουν.

Πρέπει να τονίσουμε ότι όλα τα παραπάνω έχουν ιδιαίτερη χρησιμότητα είτε για ανθρώπους διαχειριστές είτε για μελλοντικές διεργασίες διαχείρισης οι οποίες προβλέπεται να ενέχουν την απαραίτητη λογική (π.χ. με κάποια από τις μεθόδους της τεχνητής νοημοσύνης) ώστε να παρέχουν τις δυνατότητες λήψης αποφάσεων και δυναμικού εντοπισμού και διόρθωσης προβλημάτων σε μεγάλα τηλεπικοινωνιακά συστήματα. Ετσι, μία διεργασία η οποία έχει το στατικό ρόλο της επικοινωνίας με κάποιους προκαθορισμένους αντιπροσώπους και της επέμβασης σε κάποιες προκαθορισμένες διαχειρίσιμες οντότητες προφανώς δεν μπορεί να ωφεληθεί από όσα περιγράφονται εδώ.

Αν υποθέσουμε ότι η πληροφορία που αναφέραμε στην προηγούμενη υποενότητα βρίσκεται στον κατάλογο για κάθε πόρο και για κάθε διεργασία του ΔΔΤ (κάτι το οποίο όπως είπαμε γίνεται από το νέο μοντέλο της παραγράφου 2.2), τότε ένας διαχειριστής που έχει τα παραπάνω στοιχεία μπορεί να προσδιορίσει τις διαχειρίσιμες οντότητες στις οποίες αναφέρεται η λειτουργία διαχείρισης που θέλει να εκτελεστεί ως εξής:

1. προσπελαύνοντας την πληροφορία που περιέχεται στην οντότητα καταλόγου που αντιπροσωπεύει το συγκεκριμένο πόρο βρίσκει το όνομα (DN) της διεργασίας (αντιπροσώπου) η οποία παρέχει την απαιτούμενη υπηρεσία διαχείρισης. Ας σημειωθεί ότι σε αυτό το σημείο ο διαχειριστής χρησιμοποιεί ένα ζευγάρι δεδομένων:

   - όνομα (DN) διαχειριζόμενου πόρου, και
   - όνομα υπηρεσίας διαχείρισης.

2. προσπελαύνοντας την πληροφορία που περιέχεται στην οντότητα καταλόγου που αντιπροσωπεύει τη διεργασία που βρέθηκε στο πρώτο βήμα, βρίσκει:

   - την MIB που υποστηρίζει η διεργασία αυτή· και
   - κατ' επέκταση, τα ονόματα των ζητούμενων διαχειρίσιμων οντοτήτων μέσα στην MIB αυτή. Να σημειωθεί ότι αυτό προϋποθέτει ότι η αρχική γνώση που εσωτερικά διέθετε ο διαχειριστής σχετικά με τις ζητούμενες διαχειρίσιμες οντότητες σε συνδυασμό με την γνώση του για την MIB του αντιπροσώπου επιτρέπει την εύρεση των παραπάνω ονομάτων. Πρέπει να διευκρινιστεί ότι η εύρεση αυτή είναι αποτέλεσμα εσωτερικής επεξεργασίας της λαμβανόμενης ΚΓΔ και των αρχικών δεδομένων του διαχειριστή σχετικά με τη λειτουργία διαχείρισης που επιθυμεί να εκτελέσει. Η MIB που υποστηρίζει ένας αντιπρόσωπος μπορεί να δίνεται είτε με τη μορφή του ονόματος μίας ευρέως γνωστής MIB είτε να παραπέμπει σε επιπλέον οντότητες του καταλόγου οι οποίες είναι δυνατό να περιγράφουν την εν λόγω MIB.

Στην πραγματικότητα, η πληροφορία η οποία είναι σχετική με τη δομή των MIB που υποστηρίζουν οι διάφοροι αντιπρόσωποι αποτελεί άλλο ένα κομμάτι της ΚΓΔ το οποίο μπορεί να αποθηκευτεί στον Κατάλογο. Στην παρούσα εργασία δεν ασχοληθήκαμε με το πώς ακριβώς μπορεί να γίνει κάτι τέτοιο παρόλο που γενικά υποθέτουμε ότι η γνώση της δομής των διαφόρων MIB μελλοντικά θα μπορεί να αποκτάται δυναμικά όπως η ΚΓΔ στην οποία αναφερθήκαμε στην προηγούμενη υποενότητα· ο ενδιαφερόμενος αναγνώστης παραπέμπεται στα [Org95] και [DzNT93].

Στη συνέχεια τα DN των προηγουμένων οντοτήτων σχηματίζονται όπως προαναφέραμε από την συνένωση των παραπάνω ονομάτων με το DN της διεργασίας – που είναι γνωστό από το πρώτο βήμα.

Στην περίπτωση που το πρώτο βήμα του αλγορίθμου βρίσκει περισσότερες από μία διεργασίες[4], η επιλογή ανάμεσα σ' αυτές γίνεται στο δεύτερο βήμα με βασικό κριτήριο την MIB που υποστηρίζει η κάθε διεργασία. Κατ' επέκταση, η επιλογή μπορεί να γίνει και βάσει της πληροφορίας που υπάρχει στις SMAE κάθε διεργασίας (π.χ. βάσει των λειτουργιών και των εκδόσεων των πρωτοκόλλων που υποστηρίζει κάθε SMAE).

## 2.4 Ο μηχανισμός διαφάνειας ως προς την τοποθεσία

Ενα από τα βασικά χαρακτηριστικά του ΔΔΤ είναι ότι οι διεργασίες του βρίσκονται κατά κανόνα σε διαφορετικά συστήματα από αυτά που αντιπροσωπεύουν ή διαχειρίζονται. Εκτός από τις διεργασίες που λειτουργούν σαν NE που, προφανώς, πρέπει να βρίσκονται στο ίδιο σύστημα με αυτό που αντιπροσωπεύουν προκειμένου να συγκεντρώσουν από τις διάφορες δομές του λογισμικού που υποστηρίζει το σύστημα τα διάφορα δεδομένα τα οποία αποτελούν τη βάση για τη διαχείριση του εν λόγω συστήματος, οι υπόλοιπες διεργασίες βρίσκονται συνήθως σε διαφορετικά συστήματα τα οποία στην ιδανική περίπτωση θα διασυνδέονται σε ένα αυτόνομο δίκτυο το οποίο αποτελεί το ΔΔΤ.
Είναι προφανές ότι με αυτόν τον τρόπο το ΔΔΤ επηρεάζεται όσο το δυνατό λιγότερο από βλάβες του διαχειριζόμενου δικτύου. Σαν συνέπεια των παραπάνω, οι διεργασίες του ΔΔΤ

[4] Αν και ασυνήθιστο, είναι δυνατό να υπάρχουν, π.χ., δύο OS τα οποία να παρέχουν διαχείριση χρέωσης για κάποιο στοιχείο του δικτύου. Καθένα από τα OS μπορεί να υποστηρίζει κάποιο υποσύνολο από τις λειτουργίες διαχείρισης που απαρτίζουν την διαχείριση της χρέωσης. Το ποιες λειτουργίες διαχείρισης υποστηρίζει κάθε OS, θα γίνεται φανερό από τις ΔΟ που διαθέτει στην MIB του.

είναι κατά κανόνα μετακινήσιμες, δηλαδή μπορούν να τοποθετηθούν και να λειτουργήσουν σε ένα μεγάλο αριθμό από συστήματα του ΔΔΤ προσφέροντας έτσι τη δυνατότητα επαναδιαμόρφωσης (reconfiguration) του ΔΔΤ όποτε αυτό κριθεί αναγκαίο (π.χ. σε περίπτωση μίας εσωτερικής βλάβης στο ΔΔΤ). Ετσι, κάθε διεργασία του ΔΔΤ αναγνωρίζεται από το μοναδικό όνομα της και όχι από τη διεύθυνση της η οποία μπορεί να αλλάζει κατά τη διάρκεια λειτουργίας του ΔΔΤ.

Σε ένα τέτοιο περιβάλλον, η διαφάνεια ως προς την τοποθεσία των διεργασιών είναι ένας αναγκαίος μηχανισμός ο οποίος αναφέρεται σε ένα τρόπο που επιτρέπει την εύρεση της τρέχουσας διεύθυνσης μίας διεργασίας (της οποίας ξέρουμε αρχικά το DN)· ο τρόπος αυτός πρέπει εξ ορισμού να είναι ανεξάρτητος από το γεγονός ότι η διεύθυνση αυτή μπορεί να αλλάζει. Αν θεωρήσουμε ότι οι μετακινήσεις των διαφόρων διεργασιών μέσα στο ΔΔΤ δεν είναι πολύ συχνές καταλαβαίνουμε ότι οι διευθύνσεις τους μπορούν να αποθηκεύονται, να ενημερώνονται και να ανακτώνται σαν πληροφορία του X.500. Για το λόγο αυτό, η διεύθυνση (PSAP) στην οποία βρίσκεται κάθε SMAE οποιασδήποτε διεργασίας, όπως αναφέραμε στην υποενότητα 2.3.1, κρατιέται στον Κατάλογο και, πιο συγκεκριμένα, στις οντότητες καταλόγου που αντιστοιχούν στις SMAE.

Η βασική απαίτηση από ένα μηχανισμό ο οποίος παρέχει στο ΔΔΤ διαφάνεια ως προς την τοποθεσία των διεργασιών είναι ότι δοθέντος του μοναδικού ονόματος (DN) μίας διεργασίας – το οποίο είναι ανεξάρτητο από την τοποθεσία της – με την οποία θέλουμε να επικοινωνήσουμε, πρέπει να μπορεί να εντοπιστεί η διεύθυνση (PSAP) όπου η SMAE που περιέχεται στη διεργασία περιμένει είτε για ειδοποιήσεις (notifications) είτε για αιτήσεις πράξεων διαχείρισης. Επειδή μία διεργασία μπορεί να περιέχει περισσότερες από μία SMAE ο παραπάνω μηχανισμός πρέπει να περιλαμβάνει και μία επιλογή μεταξύ αυτών των SMAE για τον προσδιορισμό της SMAE μέσω της οποίας θα πραγματοποιηθεί τελικά η επικοινωνία. Ετσι, με την προϋπόθεση ότι ο Κατάλογος έχει ενημερωθεί με την πληροφορία που περιγράψαμε στην §2.3.1 σχετικά με τις διάφορες διεργασίες και τις SMAE τους, ο επόμενος αλγόριθμος παρέχει το μηχανισμό με τον οποίο μία διεργασία $A$ (είτε σε ρόλο διαχειριστή είτε σε ρόλο αντιπροσώπου) μπορεί να συνδεθεί με μία διεργασία $B$ (που ενεργεί στον αντίθετο ρόλο) της οποίας το όνομα είναι γνωστό.

1. Δοθέντος του DN της διεργασίας B, η A – η οποία χρησιμοποιεί το μοντέλο επικοινωνίας διαχειριστή αντιπροσώπου του σχήματος 2.2 – εκτελεί μία αναζήτηση στον Κατάλογο με τους ακόλουθους περιορισμούς (δηλ. χρησιμοποιώντας το ανάλογο φίλτρο):

- το σημείο της DIB κάτω από το οποίο γίνεται η αναζήτηση είναι η οντότητα καταλόγου που αντιπροσωπεύει τη διεργασία B·
- ψάξε για οντότητες οι οποίες αντιπροσωπεύουν SMAE (δηλ. ανήκουν σε κάποια ορισμένη κλάση)·
- από τις παραπάνω SMAE ψάξε για αυτές που υποστηρίζουν τον τύπο των πρωτοκόλλων, και τις λειτουργίες που ταιριάζουν με αυτές που υποστηρίζει η διεργασία A· και
- ψάξε για SMAE που ενεργούν στον αντίθετο ρόλο από αυτόν που ενεργεί η A.

Η παραπάνω αναζήτηση, σε περίπτωση επιτυχίας, θα πρέπει να επιλέξει μία μοναδική SMAE της διεργασίας B και να επιστρέψει τη διεύθυνση (PSAP) της SMAE αυτής.

## 2.5 Παράδειγμα εφαρμογής του νέου μοντέλου

Στην ενότητα αυτή δίνουμε ένα γενικό παράδειγμα της χρήσης των αλγορίθμων που περιγράφηκαν στις δύο προηγούμενες ενότητες για το ΔΔΤ του σχήματος 1.6. Πιο συγκεκριμένα, ας υποθέσουμε ότι στο περιβάλλον του σχήματος 1.6 ο άνθρωπος-διαχειριστής που κάθεται πίσω από τη διεργασία *WS* θέλει να βρεί το μέσο όρο των πακέτων που διακινούνται κάθε ώρα στο δίκτυο των τριών μεταγωγέων. Θα δείξουμε πώς οι προηγούμενοι αλγόριθμοι χρησιμοποιούνται για την εξαγωγή της παραπάνω πληροφορίας από το υπάρχον ΔΔΤ. Η βασική προϋπόθεση βέβαια είναι ότι οι διεργασίες χρησιμοποιούν το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου που περιγράψαμε στην ενότητα 2.2 το οποίο προσφέρει τις δυνατότητες για την εκτέλεση των λειτουργιών που περιγράφηκαν.
Ετσι, υποθέτουμε ότι κατά την έναρξη των διαφόρων διεργασιών του ΔΔΤ έχει γίνει η εγγραφή και ενημέρωση της πληροφορίας των οντοτήτων του καταλόγου που αντιπροσωπεύουν πόρους, διεργασίες και τις SMAE τους. Συνεπώς, η γενική δομή της DIB είναι αυτή που φαίνεται στο σχήμα 2.4 και η περιεχόμενη πληροφορία είναι αυτή που περιγράψαμε στην υποενότητα 2.3.1. Ας θεωρήσουμε λοιπόν ότι αρχικά ο άνθρωπος-διαχειριστής στη διεργασία *WS* – η οποία ενεργεί, προφανώς, σε ρόλο διαχειριστή – ξέρει τα εξής:

- το όνομα (DN) του πόρου στον οποίο θέλει να εκτελέσει κάποια λειτουργία διαχείρισης. Αυτό είναι το όνομα της οντότητας καταλόγου που αντιστοιχεί στο δίκτυο των τριών μεταγωγέων, δηλαδή το {**c=GR, o=FORTH, ou=ICS, cn=networks, cn=switching network**}·

- την υπηρεσία διαχείρισης στην οποία περιλαμβάνεται η λειτουργία διαχείρισης που θέλει να εκτελέσει. Στην περίπτωση που η λειτουργία διαχείρισης είναι η εύρεση του μέσου όρου των πακέτων που διακινούνται κάθε ώρα στο παραπάνω δίκτυο, ας υποθέσουμε ότι η υπηρεσία είναι η διαχείριση της κυκλοφορίας (traffic management)· και
- μία αφηρημένη περιγραφή – η οποία είναι ανεξάρτητη από κάποια MIB – της διαχειρίσιμης οντότητας η οποία περιέχει το μέσο όρο των πακέτων που διακινούνται σε κάποιο δίκτυο (π.χ. στην προκειμένη περίπτωση μπορεί να ξέρει απλά την κλάση στην οποία ανήκει η διαχειρίσιμη οντότητα και το χαρακτηριστικό της κλάσης το οποίο περιέχει τη ζητούμενη πληροφορία).

Δοθέντων των παραπάνω, ο διαχειριστής θέλει να βρει το DN της διαχειρίσιμης οντότητας την οποία πρέπει να προσπελάσει για να αποκτήσει τη ζητούμενη πληροφορία. Αυτό μπορεί να γίνει χρησιμοποιώντας τον αλγόριθμο της υποενότητας 2.3.2 ως εξής:

1. προσπελαύνοντας την ΚΓΔ στην οντότητα {**c=GR, o=FORTH, ou=ICS, cn=networks, cn=switching network**} βρίσκει το DN της διεργασίας που παρέχει τη διαχείριση της κυκλοφορίας για τον εν λόγω πόρο. Εστω ότι το DN της ζητούμενης διεργασίας είναι:

   {**c=GR, o=FORTH, ou=ICS, cn=management processes, cn=NLM-OS-T**}

2. προσπελαύνοντας την ΚΓΔ στην οντότητα {**c=GR, o=FORTH, ou=ICS, cn=management processes, cn=NLM-OS-T**} βρίσκει την MIB που υποστηρίζει η διεργασία η οποία αντιπροσωπεύεται από αυτήν την οντότητα. Βάσει της γνώσης που έχει ο διαχειριστής σχετικά με τη δομή αυτής της MIB και σε συνδυασμό με την αφηρημένη περιγραφή της ζητούμενης διαχειρίσιμης οντότητας βρίσκει το όνομα της τελευταίας μέσα στην MIB. Εστω ότι αυτό είναι το:

   {**networkId=switching network, netstats=perhour**}
   Ετσι, το DN της ζητούμενης διαχειρίσιμης οντότητας είναι:

   {**c=GR, o=FORTH, ou=ICS, cn=management processes, cn=NLM-OS-T, networkId=switching network, netstats=perhour**}

Στη συνέχεια, δείχνουμε πώς ο διαχειριστής ανακαλύπτει τη διεύθυνση της SMAE με την οποία πρέπει να επικοινωνήσει προκειμένου να προσπελάσει την πληροφορία στην παραπάνω διαχειρίσιμη οντότητα. Αυτό γίνεται με την εφαρμογή του αλγορίθμου που περιγράψαμε στην ενότητα 2.4 στο παρακάτω βήμα:

1. Από το όνομα της διαχειρίσιμης οντότητας ο διαχειριστής ξέρει ότι η διεργασία με την οποία θέλει να επικοινωνήσει είναι η:

   **{c=GR, o=FORTH, ou=ICS, cn=management processes, cn=NLM-OS-T}**
   Εκτελεί λοιπόν μία αναζήτηση στον Κατάλογο φιλτράροντας:

   - οντότητες που αντιπροσωπεύουν SMAE που περιέχονται στην παραπάνω διεργασία (δηλ. για οντότητες που είναι εγγεγραμμένες κάτω από την οντότητα με το παραπάνω DN και ανήκουν σε μία κλάση ειδική για οντότητες που αντιπροσωπεύουν SMAE)·
   - από τις παραπάνω, τις SMAE μέσω των οποίων διατίθεται η ζητούμενη διαχειρίσιμη οντότητα·
   - από τις παραπάνω, τις SMAE οι οποίες υποστηρίζουν το πλαίσιο εφαρμογής (application context) και τις λειτουργίες των πρωτοκόλλων που απαιτούνται για τη διασύνδεση μεταξύ WS και OS· και
   - από τις παραπάνω, τις SMAE οι οποίες ενεργούν σε ρόλο αντιπροσώπου.

   Η οντότητα καταλόγου η οποία αντιπροσωπεύει μία SMAE που ικανοποιεί τους παραπάνω περιορισμούς είναι η **{c=GR, o=FORTH, ou=ICS, cn=management processes, cn=NLM-OS-T, cn=NLM-agnt}**. Η πληροφορία στην οντότητα αυτή προσπελαύνεται, ώστε να βρεθεί η διεύθυνση (PSAP) στην οποία "ακούει" η SMAE με την οποία πρέπει να γίνει η επικοινωνία του διαχειριστή.

## 2.6 Η υπηρεσία του καθολικού CMIS/CMIP

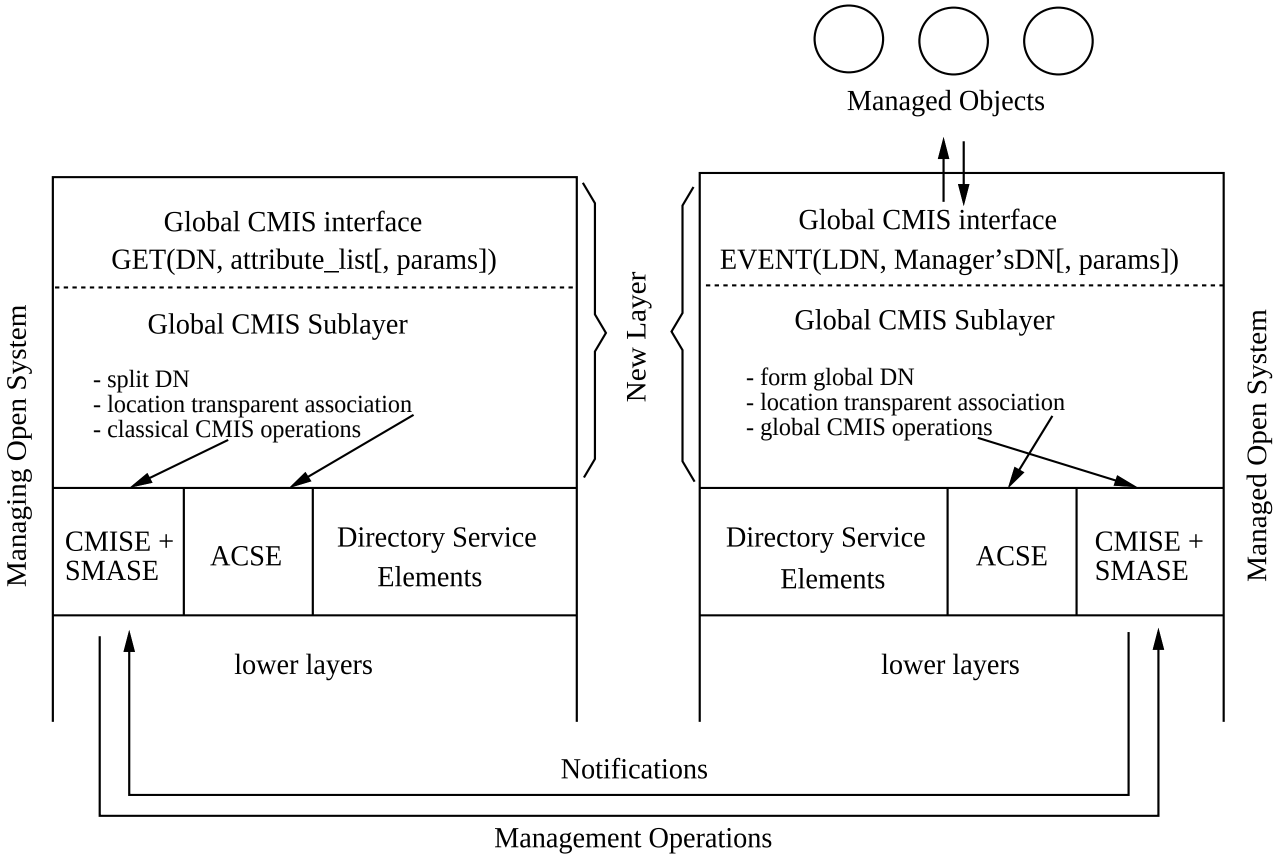
Η τυποποίηση ISO-9595 [Org90b] περιγράφει την υπηρεσία της κοινής πληροφορίας διαχείρισης (Common Management Information Service – CMIS) η οποία παρέχεται μέσω του πρωτοκόλλου διαχείρισης CMIP [Org90a] και γι' αυτό το λόγο συχνά αναφέρεται σαν CMIS/CMIP. Σύμφωνα με την τυποποίηση αυτή, η σύνδεση μεταξύ διαχειριστή/αντιπροσώπου

ακολουθείται είτε από ατήσεις του διαχειριστή για κάποιες πράξεις διαχείρισης, είτε από ειδοποιήσεις του αντιπροσώπου οι οποίες σχετίζονται με κάποιες διαχειρίσιμες οντότητες που περιέχονται στην MIB του αντιπροσώπου. Αυτές οι διαχειρίσιμες οντότητες προσδιορίζονται από τα ονόματα τους μέσα στην MIB τα οποία αναφέρονται και ως τοπικά διακεκριμένα ονόματα (Local Distinguished Names – LDN) σε αντιδιαστολή με τα καθολικά διακεκριμένα ονόματα (ή, απλά, DN) των διαχειρίσιμων οντότητων που ορίσαμε στην ενότητα 2.1.
Στην ενότητα αυτή περιγράφουμε την υπηρεσία του καθολικού CMIS/CMIP το οποίο χρησιμοποιεί τα – καθολικά – DN των διαχειρίσιμων οντοτήτων και το μηχανισμό διαφάνειας ως προς την τοποθεσία που ήδη περιγράψαμε, προκειμένου να παρέχει μία υπηρεσία παρόμοια με αυτή του κλασσικού CMIS/CMIP με τη διαφορά ότι η διαδικασία δημιουργίας της σύνδεσης (association establishment) είναι αθέατη στο χρήστη της υπηρεσίας. Το σχήμα 2.5 απεικονίζει τη θέση της νέας υπηρεσίας στη λογική αρχιτεκτονική του διαχειριζόμενου συστήματος καθώς και του διαχειρίζοντος συστήματος. Στη συνέχεια περιγράφουμε τον τρόπο λειτουργίας της υπηρεσίας του καθολικού CMIS/CMIP κάνοντας σαφή τα σημεία στα οποία διαφέρει από το υπάρχον CMIS/CMIP. Διακρίνουμε τις γνωστές δύο περιπτώσεις:

1. **Αίτηση μίας πράξης διαχείρισης από το διαχειρίζον σύστημα.**
   Στην περίπτωση αυτή, η σύνδεση πυροδοτείται από μία διεργασία η οποία ενεργεί σε ρόλο διαχειριστή και αποστέλλεται μία (ή περισσότερες) πράξη διαχείρισης προς κάποια (ή κάποιες) διαχειρίσιμη οντότητα. Ας υποθέσουμε ότι η πράξη διαχείρισης είναι μία *GET* (βλ. §1.1.3).

   Στο κλασσικό CMIS/CMIP η αποστολή αυτής της πράξης χωρίζεται σε δύο βήματα. Αρχικά, ο διαχειριστής πρέπει να φροντίσει να γίνει η σύνδεση με τον κατάλληλο αντιπρόσωπο. Αυτό απαιτεί τη γνώση της διεύθυνσης της κατάλληλης SMAE του αντιπροσώπου η οποία μπορεί εύκολα να αποκτηθεί με το μηχανισμό που περιγράφηκε στην ενότητα 2.4. Εστω ότι η εντολή που στέλνεται μέσω του ACSE (το οποίο είναι υπεύθυνο για τη δημιουργία των συνδέσεων) είναι – χρησιμοποιώντας έναν απλουστευμένο συμβολισμό – η παρακάτω:

   `ASSOCIATE(PSAP_of_agent, &ASSOCIATION_ID[, params])`
   Από τη στιγμή που η παραπάνω εντολή θα επιστρέψει το αναγνωριστικό σύνδεσης (association identifier) `ASSOCIATION_ID`, δημιουργείται η σύνδεση και ο διαχειριστής μπορεί να στείλει την πράξη διαχείρισης χρησιμοποιώντας αυτό το αναγνωριστικό καθώς και το τοπικό διακεκριμένο όνομα της διαχειρίσιμης οντότητας στην οποία απευθύνεται η




Σχήμα 2.5: Σχηματική περιγραφή της υπηρεσίας του καθολικού CMIS/CMIP.

---

πράξη. Η εντολή που στέλνεται σε αυτό το δεύτερο βήμα θα είναι (περίπου) ως εξής:

`GET(ASSOCIATION_ID, LDN, attribute_list[, params])`

Στο καθολικό CMIS/CMIP ο διαχειριστής δεν χρειάζεται να φροντίσει για τη δημιουργία της σύνδεσης. Απλά αποστέλλει στο υποεπίπεδο διασύνδεσης του καθολικού CMIS (global CMIS interface) (βλ. σχήμα 2.5) μία εντολή της μορφής:

`GET(DN, attribute_list[, params])`

Στη συνέχεια, το καθολικό CMIS αναλαμβάνει να χωρίσει το παραπάνω DN στο όνομα του αντιπροσώπου και στο τοπικό διακεκριμένο όνομα της διαχειριζόμενης οντότητας. Χρησιμοποιώντας το μηχανισμό διαφάνειας ως προς την τοποθεσία βρίσκει την διεύθυνση της SMAE με την οποία πρέπει να επικοινωνήσει και δημιουργεί τη σύνδεση και στέλνει την αίτηση *GET* με τον ίδιο ακριβώς τρόπο όπως στο κλασσικό CMIS (άλλωστε ο τρόπος επικοινωνίας με τα ACSE, CMISE και SMASE δεν αλλάζει). Το πλεονέκτημα είναι ότι η αρχική διαδικασία των δύο βημάτων είναι αθέατη στο διαχειριστή χάρη στη χρήση του καθολικού ονόματος της διαχειρίσιμης οντότητας.

2. **Αποστολή μίας ειδοποίησης από το διαχειριζόμενο σύστημα.**
Σε αυτή την περίπτωση η σύνδεση πυροδοτείται από μία διεργασία σε ρόλο αντιπροσώπου η οποία θέλει να στείλει μία ειδοποίηση σε κάποιο διαχειριστή. Η ειδοποίηση αυτή αφορά σε κάποια διαχειρίσιμη οντότητα. Η κλασσική διαδικασία αποτελείται από δύο βήματα τα οποία είναι παρόμοια με τα αντίστοιχα της πρώτης περίπτωσης. Δηλαδή, πρώτα δημιουργείται η σύνδεση και στη συνέχεια αποστέλλεται η ειδοποίηση με τις παρακάτω εντολές:

```
ASSOCIATE(PSAP_of_manager, &ASSOCIATION_ID[, params])
EVENT-REPORT(ASSOCIATION_ID, LDN[, params])
```

Στο καθολικό CMIS/CMIP ο αντιπρόσωπος απλά αποστέλλει στο υποεπίπεδο διασύνδεσης του καθολικού CMIS μία εντολή της μορφής:

```
EVENT-REPORT(LDN, Manager's_DN[, params])
```

Η πρώτη παράμετρος στην προηγούμενη γραμμή είναι το τοπικό διακεκριμένο όνομα (LDN) της διαχειρίσιμης οντότητας στην οποία αφορά η ειδοποίηση ενώ η δεύτερη παράμετρος είναι το DN του διαχειριστή στον οποίο αποστέλλεται η ειδοποιήση. Η εσωτερική λειτουργία του καθολικού CMIS στο διαχειριζόμενο σύστημα περιλαμβάνει:

- το σχηματισμό του DN της διαχειρίσιμης οντότητας από τη συνένωση του ονόματος του αντιπροσώπου και του παραπάνω LDN,
- τη δημιουργία της σύνδεσης με το διαχειριστή με τη βοήθεια του μηχανισμού διαφάνειας ως προς την τοποθεσία – εφόσον το όνομα του διαχειριστή είναι δεδομένο· και
- την αποστολή της ειδοποίησης χρησιμοποιώντας το DN της διαχειρίσιμης οντότητας με την εντολή:

  ```
  EVENT-REPORT(ASSOCIATION_ID, DN[, params])
  ```

Πρέπει να σημειώσουμε ότι, με την προϋπόθεση ότι όλες οι διεργασίες σε ένα ΔΔΤ χρησιμοποιούν το καθολικό CMIS/CMIP, η αποστολή ειδοποιήσεων χρησιμοποιώντας καθολικά ονόματα διαχειρίσιμων οντοτήτων είναι αναγκαία προκειμένου οι διαχειριστές να λειτουργούν όπως περιγράφηκε στην πρώτη περίπτωση.

Οπως γίνεται φανερό από τα παραπάνω, το καθολικό CMIS/CMIP χρησιμοποιεί τον μηχανισμό διαφάνειας ως προς την τοποθεσία που περιγράψαμε στην ενότητα 2.4. Αυτό όχι μόνο ανακουφίζει τα διάφορα προγράμματα που τρέχουν πάνω από αυτή την υπηρεσία από τον επιπλέον κώδικα για τη δημιουργία των συνδέσεων μεταξύ διαχειριστή/αντιπροσώπου αλλά και κρύβει από αυτά την ακριβή τοποθέτηση των διαφόρων διεργασιών του ΔΔΤ. Κατά τα άλλα η νέα υπηρεσία είναι συμβατή με την κλασσική CMIS/CMIP με τη βασική διαφορά ότι χρησιμοποιούνται τα καθολικά ονόματα των διαχειρίσιμων οντοτήτων. Ετσι, τα διάφορα προγράμματα που κάνουν χρήση αυτής της υπηρεσίας έχουν την ψευδαίσθηση ότι όλες οι διαχειρίσιμες οντότητες αποτελούν τμήμα μίας ενιαίας βάσης πληροφορίας διαχείρισης η οποία φυλάσσεται σε έναν κεντρικό αντιπρόσωπο διαχείρισης.

# Κεφάλαιο 3

# Η υλοποίηση

Στο κεφάλαιο αυτό παρουσιάζουμε μερικές λεπτομέρειες από την υλοποίηση των μηχανισμών που υποστηρίζει το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου που περιγράψαμε στο προηγούμενο κεφάλαιο. Πιο συγκεκριμένα, υλοποιήθηκε η ενημέρωση του Καταλόγου με την ΚΓΔ την οποία παραθέσαμε στην §2.3.1, ο μηχανισμός διαφάνειας ως προς την τοποθεσία και ο μηχανισμός αντιστοίχισης διαχειρίσιμων οντοτήτων σε πόρους. Ο τελευταίος έχει νόημα μόνο για διεργασίες που ενεργούν σε ρόλο διαχειριστή όπως έχουμε ήδη εξηγήσει. Το εγχειρίδιο προγραμματισμού και χρήσης του συστήματος που υλοποιήθηκε υπάρχει διαθέσιμο σε ξεχωριστή αναφορά [Sta95] και ο αναγνώστης παραπέμπεται σε αυτό για περισσότερες λεπτομέρειες.

Στο προηγούμενο κεφάλαιο κάναμε σαφές ότι το βασικό στοιχείο του νέου μοντέλου που περιγράψαμε είναι η διασύνδεση των διεργασιών διαχείρισης με το X.500 μέσω ενός ειδικού DUA (Directory User Agent). Το σύστημα *QUIPU*[1] [Kil91] αποτελεί σήμερα το ευρύτερα διαδεδομένο και χρησιμοποιούμενο σύστημα για την παροχή της υπηρεσίας καταλόγου. Διαθέτει έναν DSA, διάφορους DUA και υλοποιεί τα πρωτόκολλα DSP και DAP για τη διασύνδεση των DSA μεταξύ τους και με τους DUA, αντίστοιχα. Επιπλέον, παρέχει μία βιβλιοθήκη για τον προγραμματισμό νέων DUA οι οποίοι βέβαια θα μπορούν να επικοινωνούν μέσω του DAP με τους DSA. Ετσι, το *QUIPU* χρησιμοποιήθηκε τόσο για την υλοποίηση του DUA του σχήματος 2.2 όσο και για τη δημιουργία των νέων οντοτήτων καταλόγου οι οποίες κρατούν την πληροφορία που περιγράψαμε στην §2.3.1.

Το πακέτο λογισμικού *OSIMIS* [PMBK95] είναι σήμερα το πιο γνωστό περιβάλλον ανάπτυξης

[1] Το όνομα *QUIPU* δεν σχηματίζεται από κάποια αρχικά. Η προέλευσή του εξηγείται στο [Kil91].

συστημάτων διαχείρισης το οποίο διατίθεται δωρεάν μαζί με πηγαίο κώδικα (source code) και βιβλιοθήκες· παρέχει ένα προγραμματιστικό περιβάλλον πάνω στο οποίο έχουν αναπτυχθεί αρκετές διεργασίες διαχείρισης (τόσο διαχειριστές όσο και αντιπρόσωποι) οι οποίες ακολουθούν τα μοντέλα και τα πρωτόκολλα του *ISO* (X.700) για διαχείριση συστημάτων τα οποία περιγράφηκαν στο πρώτο κεφάλαιο. Το προτεινόμενο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου υλοποιήθηκε σαν επέκταση του προγραμματιστικού περιβάλλοντος του *OSIMIS* και αυτή τη στιγμή αποτελεί τμήμα της τελευταίας έκδοσης του πακέτου. Η γλώσσα προγραμματισμού που χρησιμοποιήσαμε είναι η *C* [KR88].
Στην επόμενη ενότητα κάνουμε μία σύντομη περιγραφή των δυνατοτήτων του *QUIPU* και δίνουμε αναλυτικά το συντακτικό των διαφόρων οντοτήτων καταλόγου (δηλαδή τις νέες κλάσεις και τα χαρακτηριστικά τους) που χρησιμοποιήσαμε για την αποθήκευση της ΚΓΔ στο X.500. Στη δεύτερη και τελευταία ενότητα περιγράφουμε γενικά το πακέτο *OSIMIS* και τις προγραμματιστικές δυνατότητες που προσφέρει και παραθέτουμε τις βασικές ρουτίνες πρόσβασης στο X.500 που υλοποιήθηκαν. Τέλος περιγράφουμε πώς προσαρμόστηκαν στο *OSIMIS* η ενημέρωση της πληροφορίας του Καταλόγου και οι δύο μηχανισμοί που περιγράφηκαν στο προηγούμενο κεφάλαιο. Το ακριβές συντακτικό και οι παράμετροι όλων των ρουτινών που περιέχονται στην βιβλιοθήκη του συστήματός μας δίνονται αναλυτικά στο [Sta95].

## 3.1 Το σύστημα *QUIPU* και οι βασικές οντότητές του

Το σύστημα *QUIPU* αποτελεί τόσο μία υλοποίηση των βασικών διεργασιών και πρωτοκόλλων της υπηρεσίας καταλόγου όσο και ένα περιβάλλον ανάπτυξης νέων οντοτήτων καταλόγου και νέων DUA όπως ο DUA που περιλαμβάνεται στο σχήμα 2.2. Το *QUIPU* είναι τμήμα του πακέτου λογισμικού *ISODE* (ISO Development Environment) [MRR91] το οποίο είναι ένα ευρέως διαδεδομένο πακέτο υλοποίησης των ανωτέρων επιπέδων (δηλ. κυρίως των επιπέδων εφαρμογής και παρουσίασης) του μοντέλου αναφοράς για τη διασύνδεση ανοικτών συστημάτων (open systems interconnection reference model – OSIRM) [Org94b], το οποίο έχει οριστεί από τον *ISO*, καθώς και διαφόρων εφαρμογών που ακολουθούν τις τυποποιήσεις του *ISO*. Ενας λόγος που το *ISODE* είναι σήμερα η πιο διαδεδομένη πλατφόρμα λειτουργίας και ανάπτυξης εφαρμογών που ακολουθούν το OSIRM είναι η δυνατότητα εγκατάστασής του σε μία μεγάλη ποικιλία μηχανών με διάφορες αρχιτεκτονικές και διαφορετικά λειτουργικά συστήματα – τα οποία βασίζονται στο γνωστό λειτουργικό σύστημα *UNIX*. Το βασικό πλεονέκτημα που προσφέρει το *ISODE* είναι ότι παρέχει τις ισχυρές, ώριμες και τυποποιημένες εφαρμογές του

*ISO* πάνω από μία μεγάλη ποικιλία τεχνολογιών δικτύων όπως το TCP/IP [Pos81a] [Pos81b], το X.25 [Uni88] και το CLNS [Org87]· με άλλα λόγια, οι διάφορες υπηρεσίες του *ISO* παρέχονται όχι μόνο πάνω από τα – λιγότερο διαδεδομένα – δίκτυα που έχουν προταθεί από τον *ISO* (X.25, CLNS) αλλά και πάνω από το *Internet* (TCP/IP) το οποίο αποτελεί σήμερα το πιο δημοφιλές δίκτυο. Πραγματικά, το πιο ελκυστικό σημείο στο OSIRM είναι ότι περιλαμβάνει πλήρεις εφαρμογές οι οποίες έχουν γενικότητα και μεγάλη ευελιξία για την παροχή ολοκληρωμένων υπηρεσιών. Από την άλλη, οι αντίστοιχες τεχνολογίες δικτύων του *ISO* έχουν υποσκελιστεί από το δίκτυο *Internet*. Το *ISODE* προσφέρει τη δυνατότητα να χρησιμοποιήσουμε πρωτόκολλα όπως το CMIP και τα DSP και DAP πάνω από το *Internet*. Μεγαλύτερη αναφορά στην παραπάνω προσέγγιση και τα πλεονεκτήματα που προσφέρουν οι υπηρεσίες του *ISO* γίνεται στο βιβλίο [Mar90].
Επιπλέον, το *ISODE* περιλαμβάνει ένα περιβάλλον ανάπτυξης νέων εφαρμογών και πρωτοκόλλων πάνω από τις παραπάνω τεχνολογίες. Για το σκοπό αυτό παρέχει διάφορους μεταφραστές της γλώσσας ASN.1 [Org90c]. Η ASN.1 (Abstract Syntax Notation 1) είναι μία γλώσσα που χρησιμοποιείται για την αναπαράσταση δομών δεδομένων με έναν τρόπο ανεξάρτητο μηχανής. Με τη χρήση της ASN.1 μπορούν να οριστούν τόσο τα διάφορα πρωτόκολλα επικοινωνίας και η δομή των μηνυμάτων που ανταλλάσσουν όσο και η δομή των διαφόρων οντοτήτων που συνθέτουν μία βάση πληροφορίας όπως η βάση πληροφορίας καταλόγου (MIB) και η βάση πληροφορίας διαχείρισης (DIB).

### 3.1.1 Ορισμός των νέων οντοτήτων του καταλόγου σε ASN.1

Στην υποενότητα αυτή ορίζουμε τις διάφορες οντότητες καταλόγου χρησιμοποιώντας τη γλώσσα ASN.1. Μία λεπτομερής περιγραφή της ASN.1 υπάρχει στο [Org90c]. Εδώ απλά εξηγούμε μόνο ό,τι χρειάζεται προκειμένου να γίνουν κατανοητοί οι ορισμοί που ακολουθούν.

#### Οντότητες για πόρους

Στο σχήμα 3.1 φαίνονται οι ορισμοί σε ASN.1 για τις οντότητες καταλόγου οι οποίες αντιστοιχούν σε πόρους. Στις γραμμές 1-3 του σχήματος 3.1 ορίζεται η κλάση οντοτήτων καταλόγου στην οποία ανήκουν οι οντότητες που αντιστοιχούν σε διαχειριζόμενους πόρους. Το όνομα της κλάσης είναι *managedResource*. Είναι υποκλάση (subclass) της τυποποιημένης κλάσης *top* και ενέχει δύο υποχρεωτικά χαρακτηριστικά τα οποία ονομάζονται *responsibleSMAP* και *cannonicalName*. Το πρώτο χαρακτηριστικό αναφέρεται στη διεργασία

```
(1)   managedResource          OBJECT-CLASS
(2)   SUBCLASS OF              top
(3)   MUST CONTAIN             {responsibleSMAP, cannonicalName}

(4)   responsibleSMAP                     ATTRIBUTE
(5)   WITH ATTRIBUTE-SYNTAX               resposibleSMAPSyntax
(6)   MULTI  VALUE

(7)   responsibleSMAPSyntax::=  SEQUENCE {
(8)           DistinguishedName,          -- DistinguishedName is defined in the standard
(9)           tMNManagementService        OPTIONAL}

(10)  tMNManagementService::=   ENUMERATED {
(11)          CustomerAdministration (0),
(12)          Management of the Security of the TMN (1),
(13)          Traffic Management (2),
(14)          Switching Management (3),
(15)          Accounting Management (4),
(16)          Restoration and Recovery (5)   }
```

Σχήμα 3.1: Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν πόρους.

---

που παρέχει κάποια υπηρεσία διαχείρισης για τον πόρο ενώ το δεύτερο είναι το τυποποιημένο χαρακτηριστικό ονομασίας το οποίο χρησιμοποιείται στο σχηματισμό του RDN των οντοτήτων (βλ. και ενότητα 1.3).

Στις γραμμές 4-6 ορίζεται το χαρακτηριστικό *responsibleSMAP*. Ετσι, ορίζεται το συντακτικό του (γραμμή 5) το οποίο δηλώνεται από το αναγνωριστικό *responsibleSMAPSyntax* και επιπλέον ορίζεται ότι το χαρακτηριστικό αυτό μπορεί να έχει περισσότερες από μία τιμές (multi value – γραμμή 6) σε κάποια συγκεκριμένη διαχειρίσιμη οντότητα. Αυτό είναι σύμφωνο με τη δυνατότητα κάποιος πόρος να αντιπροσωπεύεται από περισσότερες από μία διεργασίες διαχείρισης (κάθε μία από τις οποίες μπορεί να παρέχει διαφορετική υπηρεσία διαχείρισης).

Στις γραμμές 7-9 ορίζεται το συντακτικό *responsibleSMAPSyntax* σαν μία ακολουθία (sequence) που αποτελείται το πολύ από δύο πεδία. Το πρώτο είναι ένα διακεκριμένο όνομα (το οποίο προφανώς αντιστοιχεί στο όνομα μίας διεργασίας διαχείρισης) και το δεύτερο είναι του τύπου *tMNManagementService*. Ο τελευταίος τύπος ορίζεται στις γραμμές 10-16 και είναι ένας απαριθμητός τύπος (enumerated) τα στοιχεία του οποίου αντιστοιχούν στις υπάρχουσες υπηρεσίες διαχείρισης. Στο σχήμα 3.1 παραθέτουμε δειγματοληπτικά ένα υποσύνολο των υπηρεσιών διαχείρισης που έχει ορίσει η *ITU*. Ο πλήρης κατάλογος υπάρχει στο [Uni93f]. Ας σημειωθεί ότι το πεδίο στη γραμμή 9 του σχήματος 3.1 είναι προαιρετικό (optional). Αυτό σημαίνει ότι η τιμή του χαρακτηριστικού *responsibleSMAP* είναι είτε ένα ζεύγος "ονόματος διεργασίας/υπηρεσίας διαχείρισης που προσφέρει η διεργασία" είτε, απλά, το όνομα μίας διεργασίας. Ο λόγος είναι ότι μία διεργασία μπορεί να μην παρέχει κάποια συγκεκριμένη υπηρεσία διαχείρισης[2] και, συνεπώς, το αντίστοιχο πεδίο μπορεί να παραληφθεί.

### Οντότητες για διεργασίες και για SMAE

Στο σχήμα 3.2 φαίνονται οι ορισμοί σε ASN.1 για τις οντότητες καταλόγου οι οποίες αντιστοιχούν σε διεργασίες διαχείρισης. Στις γραμμές 1-4 του σχήματος 3.2 ορίζεται η κλάση οντοτήτων στην οποία ανήκουν οι διεργασίες διαχείρισης. Το όνομα της είναι *managementProcess*. Η κλάση αυτή είναι υποκλάση της τυποποιημένης *applicationProcess* στην οποία ανήκουν οι διεργασίες σε επίπεδο εφαρμογής. Κάθε διεργασία διαχείρισης ανήκει στην κλάση *applicationProcess* – εφόσον είναι μία διεργασία επιπέδου εφαρμογής – καθώς και στην *managementProcess*. Η τελευταία ενέχει τα εξής χαρακτηριστικά:

- *tMNBuildingBlock*. Το χαρακτηριστικό αυτό δηλώνει τον τύπο της διεργασίας στα πλαίσια του ΔΔΤ. Το συντακτικό του είναι του τύπου *tMNBlockSyntax* (γραμμή 6) και μπορεί να έχει μία μοναδική τιμή (γραμμή 7) για κάθε οντότητα που αντιπροσωπεύει μία διεργασία διαχείρισης. Το *tMNBlockSyntax* είναι ένας απαριθμητός τύπος ο οποίος λαμβάνει επτά δυνατές τιμές (γραμμές 8-9). Για παράδειγμα, η τιμή *QA* υποδηλώνει ένα QA ενώ η τιμή *NL-OS* υποδηλώνει ένα OS το οποίο λειτουργεί στο επίπεδο διαχείρισης δικτύου (network level management). Ας σημειωθεί ότι μόνο τα OS χωρίζονται ανάλογα με το επίπεδο διαχείρισης στο οποίο λειτουργούν εφόσον οι υπόλοιπες διεργασίες λειτουργούν σε γνωστά επίπεδα (π.χ. τα NE λειτουργούν στο επίπεδο στοιχείου του δικτύου – network element level). Τέλος, το χαρακτηριστικό *tMNBuildingBlock* δηλώνεται ως προαιρετικό (MAY CONTAIN) προκειμένου η συγκεκριμένη κλάση οντοτήτων να μπορεί να αναφέρεται και σε διεργασίες που δεν ανήκουν στο ΔΔΤ αλλά σε κάποιο άλλο σύστημα διαχείρισης.

[2]Οπως είχαμε αναφέρει, μόνο τα OS παρέχουν συγκεκριμένες υπηρεσίες διαχείρισης. Ετσι, ένα NE ή ένα QA δεν παρέχει κάποια συγκεκριμένη υπηρεσία διαχείρισης αλλά, αντίθετα, η πληροφορία που περιέχει χρησιμοποιείται από πολλά OS για την παροχή των διαφόρων υπηρεσιών διαχείρισης. Σε αυτήν την περίπτωση, ο αλγόριθμος που περιγράψαμε στην §2.3.2, επίσης δουλεύει (εφόσον αναμένεται να υπάρχει το πολύ ένα NE – ή QA – που να αντιπροσωπεύει κάποιο πόρο χωρίς να σχετίζεται με κάποια υπηρεσία διαχείρισης) ψάχνοντας για τη διεργασία που δεν παρέχει συγκεκριμένη υπηρεσία διαχείρισης για κάποιο πόρο!

```
(1)   managementProcess        OBJECT-CLASS
(2)   SUBCLASS OF              applicationProcess
(3)   MAY CONTAIN              {tMNBuildingBlock, supportedMIB, tMNMS}
(4)   MUST CONTAIN             {operationalStatus}

(5)   tMNBuildingBlock                          ATTRIBUTE
(6)   WITH ATTRIBUTE-SYNTAX                     tMNBlockSyntax
(7)   SINGLE VALUE

(8)   tMNBlockSyntax::=             ENUMERATED {
(9)          NE(0), QA(1), MD(2), SL-OS(3), NL-OS(4), NE-OS(5), WS(6)      }

(10)  supportedMIB                              ATTRIBUTE
(11)  WITH ATTRIBUTE-SYNTAX                     DistinguishedName
(12)  MULTI VALUE

(13)  tMNMS                                     ATTRIBUTE
(14)  WITH ATTRIBUTE-SYNTAX                     tMNManagementService
(15)  MULTI VALUE

(16)  operationalStatus                         ATTRIBUTE
(17)  WITH ATTRIBUTE-SYNTAX                     operationalStatusSyntax
(18)  SINGLE VALUE

(19)  operationalStatusSyntax::=    BIT STRING {
(20)         enabled(0), disabled(1)  }
```

Σχήμα 3.2: Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν διεργασίες.

---

- *supportedMIB*. Το χαρακτηριστικό αυτό δηλώνει τις MIB που υποστηρίζει η διεργασία διαχείρισης (εφόσον αυτή ενεργεί σε ρόλο αντιπροσώπου). Είναι προαιρετικό – αφού μερικές διεργασίες ενεργούν μόνο σε ρόλο διαχειριστή – και το συντακτικό του είναι του τυποποιημένου τύπου *DistinguishedName* (γραμμή 11). Ο λόγος είναι ότι υποθέτουμε ότι οι ορισμοί των διαφόρων MIB θα είναι εγγεγραμμένοι σε οντότητες του καταλόγου. Τρόποι για να γίνει κάτι τέτοιο περιγράφονται στην [DzNT93] αλλά και στην τυποποίηση ISO-10164-16 [Org95]. Ετσι, οι τιμές αυτού του χαρακτηριστικού – οι οποίες φυσικά μπορεί να είναι περισσότερες από μία (γραμμή 12) – θα είναι τα DN των οντοτήτων καταλόγου που περιέχουν τους ορισμούς όλων των δυνατών διαχειρίσιμων οντοτήτων της MIB!

- *tMNMS*. Το χαρακτηριστικό αυτό δηλώνει την υπηρεσία διαχείρισης την οποία παρέχει η συγκεκριμένη διεργασία διαχείρισης. Το συντακτικό του είναι του τύπου *tMNManagementService* (γραμμή 14) το οποίο ορίστηκε στο σχήμα 3.1. Λαμβάνοντας υπόψη την πιθανότητα μία διεργασία να μπορεί να παρέχει περισσότερες από μία υπηρεσίες διαχείρισης, το χαρακτηριστικό αυτό ορίζεται να μπορεί να λαμβάνει περισσότερες από μία τιμές στην ίδια οντότητα (γραμμή 15). Είναι δε προαιρετικό, επειδή μία διεργασία μπορεί να μην παρέχει κάποια συγκεκριμένη υπηρεσία διαχείρισης.

- *operationalStatus*. Το χαρακτηριστικό αυτό είναι υποχρεωτικό της κλάσης και δηλώνει την κατάσταση της διεργασίας, δηλαδή αν είναι ενεργή (enabled) ή όχι (disabled). Εχει μία μοναδική τιμή (γραμμή 18) η οποία είναι μία από τις δύο δυνατές τιμές του συντακτικού *operationalStatusSyntax* (γραμμές 19-20).

  Η πληροφορία για το κατά πόσο μία διεργασία είναι ενεργή (enabled) ή όχι δεν μας απασχόλησε στα προηγούμενα κεφάλαια. Ο σκοπός που τη χρησιμοποιούμε είναι βασικά για να διευκολύνουμε τη διαδικασία ενημέρωσης του καταλόγου κατά τον τερματισμό της λειτουργίας μίας διεργασίας. Οποτεδήποτε μία διεργασία τερματίζει, αντί να διαγράψουμε τις οντότητες του καταλόγου που σχετίζονται με αυτήν, αλλάζουμε απλά την τιμή του χαρακτηριστικού *operationalStatus* υποδηλώνοντας τον τερματισμό της και ακυρώνοντας με αυτό τον τρόπο οποιαδήποτε πληροφορία του καταλόγου σχετικά με αυτήν. Ετσι, οι μηχανισμοί που περιγράψαμε στο προηγούμενο κεφάλαιο ελέγχουν πάντοτε την τιμή του χαρακτηριστικού αυτού.

Στο σχήμα 3.3 φαίνονται οι ορισμοί σε ASN.1 για τις οντότητες καταλόγου οι οποίες αντιστοιχούν σε SMAE. Πιο συγκεκριμένα, ορίζονται τρεις κλάσεις οντοτήτων οι δύο από τις οποίες (γραμμές 10-16) ορίζονται με λεπτομέρεια στο [Org95] και αποτελούν τυποποιημένες κλάσεις οντοτήτων του καταλόγου. Επειδή στην υλοποίηση μας υποστηρίζουμε τις κλάσεις αυτές θα εξηγήσουμε την πληροφορία που περιλαμβάνεται σε κάθε χαρακτηριστικό τους. Στις γραμμές 1-3 του σχήματος 3.3 ορίζεται η κλάση *managementEntity* σαν υποκλάση της τυποποιημένης *applicationEntity*. Στην τελευταία ανήκουν όλες οι οντότητες του καταλόγου οι οποίες αντιστοιχούν σε οντότητες εφαρμογής (application entities) και συνεπώς και οι οντότητες εφαρμογής για τη διαχείριση συστημάτων (systems management application entities – SMAE). Επιπλέον οι SMAE ανήκουν στην κλάση *managementEntity* η οποία ενέχει δύο υποχρεωτικά χαρακτηριστικά. Το πρώτο είναι το *operationalStatus* που αναφέρεται στην κατάσταση της SMAE (ενεργή ή ανενεργή). Φυσιολογικά, το χαρακτηριστικό αυτό

```
(1)   managementEntity        OBJECT-CLASS
(2)   SUBCLASS OF             applicationEntity
(3)   MUST CONTAIN            {operationalStatus,
                               managementRole}

(4)   managementRole                        ATTRIBUTE
(5)   WITH ATTRIBUTE-SYNTAX                 managementRoleSyntax
(6)   SINGLE  VALUE

(7)   managementRoleSyntax::=   BIT STRING {
(8)           agent(0), manager(1)   }

      -- the following are from ISO-10164-16. The detailed ASN.1 definitions exist therein.
(9)   cMISE                   OBJECT-CLASS
(10)  SUBCLASS OF             applicationEntity
(11)  MUST CONTAIN            {supportedCmipVersion,
                               supportedCmipProfiles,
                               supportedCmisFunctionalUnits}

(12)  sMASE                   OBJECT-CLASS
(13)  SUBCLASS OF             applicationEntity
(14)  MUST CONTAIN            {supportedSmaseFunctionalUnits}
(15)  MAY CONTAIN             {mitMoList,
                               supportsMKMglobalNames}
```

Σχήμα 3.3: Οι ASN.1 ορισμοί για οντότητες που αντιπροσωπεύουν SMAE.

---

κληρονομεί την τιμή του από την τιμή του αντίστοιχου χαρακτηριστικού της υπερκείμενης οντότητας (η οποία ανήκει στην κλάση *managementProcess* που περιγράψαμε παραπάνω και αντιστοιχεί στη διεργασία που περιέχει την SMAE).

Στο μέλλον ίσως υπάρξουν υλοποιήσεις διεργασιών διαχείρισης στις οποίες είναι δυνατό κάποιες SMAE να είναι ενεργές ενώ κάποιες άλλες όχι. Σε αυτήν την περίπτωση το χαρακτηριστικό *operationalStatus* θα πρέπει να εξετάζεται ιδιαιτέρως. Η παρούσα υλοποίηση λαμβάνει υπόψη αυτήν την πιθανή μελλοντική απαίτηση εξετάζοντας την τιμή του χαρακτηριστικού *operationalStatus* στην περίπτωση που η τιμή του ιδίου χαρακτηριστικού της υπερκείμενης οντότητας – που αντιστοιχεί στην διεργασία που περιέχει την SMAE – είναι "*enabled*". Σε αντίθετη περίπτωση, η διεργασία είναι εκτός λειτουργίας και, συνεπώς, και όλες οι SMAE που περιέχει θεωρούνται εκτός λειτουργίας.

Το δεύτερο χαρακτηριστικό της κλάσης *managementEntity* ονομάζεται *managementRole*. Η τιμή του μπορεί να πάρει μία από τις δύο τιμές που ορίζονται στις γραμμές 7-8 του σχήματος 3.3 και δηλώνει τον ρόλο στον οποίο ενεργεί μία SMAE.

Στις γραμμές 9-15 ορίζονται οι κλάσεις *cMISE* και *sMASE*. Οπως προαναφέραμε, οι κλάσεις αυτές ορίζονται στην τυποποίηση ISO-10164-16. Τα χαρακτηριστικά της κλάσης *cMISE* είναι τρία και είναι υποχρεωτικά. Περιέχουν πληροφορίες σχετικές με τις δυνατότητες και την έκδοση του CMIS/CMIP που παρέχει μία SMAE. Ας διευκρινίσουμε ότι η παραπάνω πληροφορία σχετίζεται με το CMISE (common management information service element) το οποίο είναι το στοιχείο υπηρεσίας εφαρμογής (application service element) το οποίο αντιστοιχεί στο CMIS/CMIP. Εδώ πρέπει να σημειωθεί ότι στην παρούσα έκδοση του *OSIMIS* οι τιμές των χαρακτηριστικών της κλάσης *cMISE* είναι συγκεκριμένες και ίδιες για κάθε SMAE που αναπτύσσεται. Περισσότερα για το *OSIMIS* θα αναφέρουμε στην επόμενη ενότητα του παρόντος κεφαλαίου.

Τα χαρακτηριστικά της κλάσης *sMASE* είναι τρία. Ολα σχετίζονται με το στοιχείο υπηρεσίας εφαρμογής SMASE (systems management application service element) το οποίο αναφέρεται στην πληροφορία που διατίθεται μέσω της SMAE και στις διάφορες ιδιαιτερότητες της πληροφορίας αυτής. Το χαρακτηριστικό *supportedSmaseFunctionalUnits* είναι υποχρεωτικό και αναφέρεται στις λειτουργίες που υποστηρίζει το SMASE μίας SMAE. Το σύνολο όλων των πιθανών αυτών λειτουργιών αναμένεται να είναι ορισμένο και κατανοητό από κάθε SMAE η οποία πρέπει να μπορεί να αποφασίσει κατά πόσο είναι δυνατή η επικοινωνία με μία SMAE η οποία περιέχει κάποιο συγκεκριμένο SMASE. Να σημειωθεί ότι η πληροφορία που παρέχουν οι τιμές του χαρακτηριστικού *supportedSmaseFunctionalUnits* περιλαμβάνει το ρόλο στον οποίο ενεργεί η SMAE. Το *mitMoList* είναι η λίστα από τις διαχειρίσιμες οντότητες οι οποίες γίνονται ορατές μέσω της SMAE. Τέλος, το *supportsMKMglobalNames* είναι ένα χαρακτηριστικό το οποίο δηλώνει αν η συγκεκριμένη SMAE μπορεί να χειριστεί καθολικά ονόματα για τις διαχειρίσιμες οντότητες ή όχι. Οι δυνατές τιμές είναι δύο: *true* ή *false*.

Στην παρούσα υλοποίηση, τα χαρακτηριστικά της τελευταίας κλάσης δεν χρησιμοποιούνται αλλά αντικαθίστανται από άλλα που θεωρούνται επαρκή. Διάφοροι λόγοι οδηγούν σε αυτό. Οσον αφορά το χαρακτηριστικό *supportedSmaseFunctionalUnits* παρόλο που περιέχει πληροφορία που χρησιμοποιείται στην παρούσα υλοποίηση του *OSIMIS* (δηλ. το ρόλο στον οποίο ενεργεί μία SMAE) αντικαταστήθηκε από το χαρακτηριστικό *managementRole* που περιγράψαμε προηγουμένως. Ο λόγος βρίσκεται στο ότι το σύνολο των τυποποιημένων τιμών του *supportedSmaseFunctionalUnits* (βλ. [Org92a]) εκτός από την πληροφορία για τον ρόλο

στον οποίο ενεργεί μία SMAE περιέχει και επιπλέον πληροφορία η οποία δεν χρησιμοποιείται στην παρούσα υλοποίηση του *OSIMIS* όπως οι λειτουργίες διαχείρισης που επιτελεί μία SMAE.
Σχετικά με την πληροφορία του χαρακτηριστικού *mitMoList*, αυτή δίνεται κυρίως από την τιμή του χαρακτηριστικού *supportedMIB* της υπερκείμενης οντότητας. Επειδή στις υπάρχουσες υλοποιήσεις διεργασιών περιέχεται το πολύ μία SMAE σε ρόλο αντιπροσώπου εξυπακούεται ότι αυτή η SMAE θα κάνει ορατή και όλη την MIB που περιέχει η διεργασία. Το πλεονέκτημα που μας δίνει η χρήση του χαρακτηριστικού *supportedMIB* είναι ότι η πληροφορία για την MIB που υποστηρίζει ένας αντιπρόσωπος γίνεται γνωστή στο επίπεδο των οντοτήτων του καταλόγου που αντιστοιχούν σε διεργασίες· με άλλα λόγια δεν χρειάζεται να ψάχνουμε στην πληροφορία για τις SMAE η οποία βρίσκεται ένα επίπεδο βαθύτερα στο Κατάλογο. Ετσι, το χαρακτηριστικό *mitMoList* δεν χρησιμοποιείται.
Τέλος, το χαρακτηριστικό *supportsMKMglobalNames* θεωρείται πάντα να έχει την τιμή *false* εφόσον δεν ασχοληθήκαμε καθόλου με την υλοποίηση εφαρμογών όπως το καθολικό CMIS/CMIP (την αρχιτεκτονική του οποίου περιγράψαμε στο προηγούμενο κεφάλαιο) και το οποίο είναι παράδειγμα μίας εφαρμογής η οποία θα απαιτούσε την πληροφορία που μας δίνει η τιμή του χαρακτηριστικού αυτού. Παρόλα αυτά, πρέπει να τονίσουμε ότι η υλοποίηση μας υποστηρίζει όλα τα παραπάνω χαρακτηριστικά τα οποία αφήνονται για μελλοντική χρήση.

## 3.2 Το πακέτο λογισμικού *OSIMIS* και η θέση των νέων λειτουργιών σε αυτό

Το *OSIMIS* (OSI[3] Management Information Service) [PMBK95] είναι ένα πακέτο λογισμικού το οποίο αναπτύχθηκε στο πανεπιστήμιο University College of London. Ο σκοπός του *OSIMIS* δεν είναι να παράσχει κάποιο συγκεκριμένο συνόλο από λειτουργίες διαχείρισης του *OSI* αλλά να δώσει μία γενική πλατφόρμα διαχείρισης δικτύων και συστημάτων η οποία να χρησιμοποιεί τα μοντέλα και πρωτόκολλα του *ISO* δείχνοντας έτσι την πλούσια λειτουργικότητα που αυτά προσφέρουν. Συγκεκριμένα, υποστηρίζει μία εκτεταμένη βιβλιοθήκη προγραμματισμού η οποία είναι γραμμένη στη γλώσσα *C++* [Str86] σύμφωνα με την οντοκεντρική μεθοδολογία (object-oriented paradigm). Η βιβλιοθήκη αυτή μπορεί να χρησιμοποιηθεί για την υλοποίηση

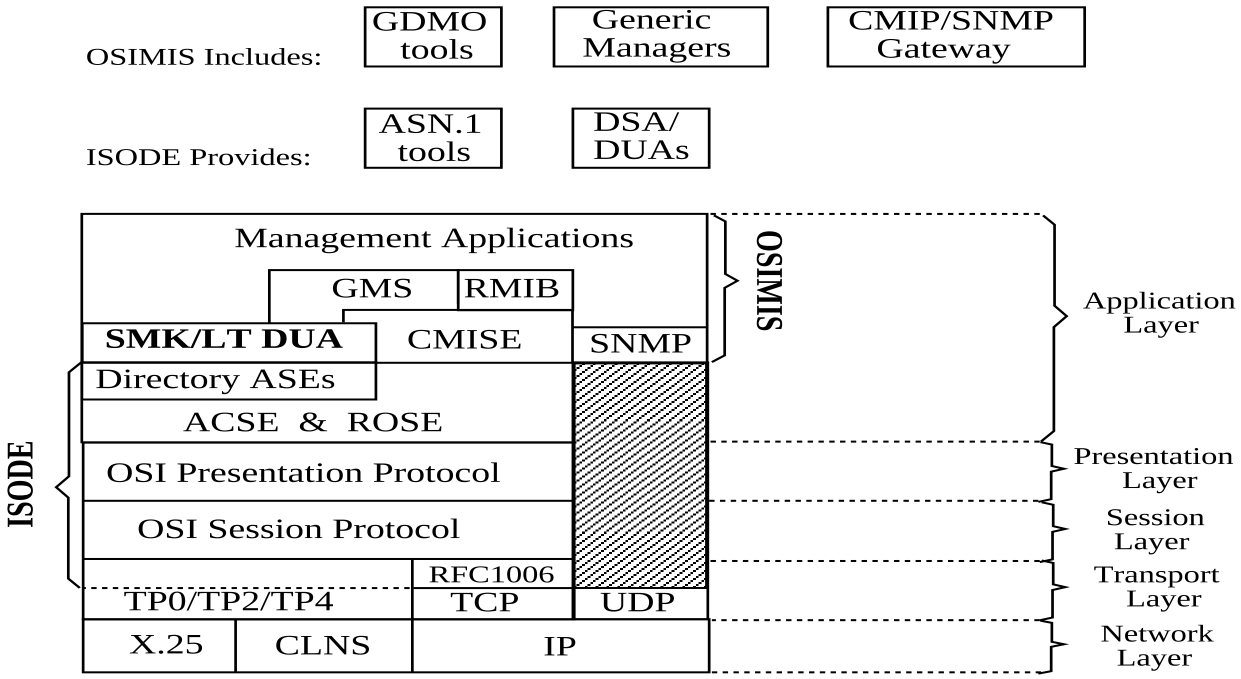
[3]Οπως προαναφέραμε το *OSI* (Open Systems Interconnection) είναι το γενικό μοντέλο του *ISO* στο οποίο βασίζονται και υπάγονται διάφορα πρωτόκολλα για τη διασύνδεση και επικοινωνία συστημάτων. Στην παρούσα εργασία όλα τα πρωτόκολλα και μοντέλα του *ISO* που περιγράψαμε υπάγονται στο *OSI*.

διεργασιών διαχείρισης (τόσο σε ρόλο διαχειριστή όσο και σε ρόλο αντιπροσώπου) και έχει σαν βασικό χαρακτηριστικό ότι κρύβει από τον προγραμματιστή τις λεπτομέρειες της πρόσβασης στην πληροφορία διαχείρισης μέσω του τυποποιημένου πρωτοκόλλου CMIP.

Το *OSIMIS* έχει αναπτυχθεί χρησιμοποιώντας το πακέτο *ISODE* για την παροχή των διαφόρων γενικών υπηρεσιών των ανωτέρων επιπέδων του OSIRM – π.χ. για την παροχή της κωδικοποίησης/αποκωδικοποίησης του επιπέδου παρουσίασης (presentation layer). Το όλο σύστημα αποτελείται από γενικά και ειδικά τμήματα. Τα γενικά τμήματα αποτελούνται από μία οντοκεντρική υποδομή για την ανάπτυξη προγραμμάτων διαχειριστών και αντιπροσώπων καθώς και από ένα σύνολο γενικών εφαρμογών διαχείρισης, δηλαδή εφαρμογών που είναι ανεξάρτητες από κάποια βάση πληροφορίας διαχείρισης (MIB). Τα ειδικά τμήματα είναι αντιπρόσωποι που υποστηρίζουν ειδικές MIB και διαχειριστές που επικοινωνούν με αυτούς. Τα ειδικά αυτά τμήματα αναπτύχθηκαν στο *OSIMIS* κυρίως σαν παραδείγματα της λειτουργικότητας και αποτελεσματικότητας των γενικών τμημάτων. Ετσι, πρέπει να σημειώσουμε ότι η έμφαση δίνεται κυρίως στη γενική προγραμματιστική υποδομή που προσφέρει το *OSIMIS*.

Η βασική προσφορά του πακέτου έγκειται στο σύστημα που παρέχει για την ανάπτυξη αντιπροσώπων διαχείρισης το οποίο ονομάζεται *GMS* (Γενικό Διαχειριζόμενο Σύστημα – Generic Managed System). Το *GMS* κρύβει τελείως από τον προγραμματιστή λεπτομέρειες σχετικές με την παροχή του CMIS, όπως είναι η εύρεση των διαχειρίσιμων οντοτήτων σε έναν αντιπρόσωπο βάσει των μηχανισμών που αναφέρθηκαν στην υποενότητα 1.1.3 (δηλαδή των λεγομένων scoping και filtering μηχανισμών) και ο χειρισμός λαθών, επιτρέποντας στους προγραμματιστές νέων MIB να επικεντρωθούν σε θέματα που αφορούν στην πρόσβαση της πληροφορίας των διαχειριζόμενων πόρων για τη δημιουργία της αφαίρεσης του πόρου που αντιπροσωπεύει κάθε διαχειρίσιμη οντότητα. Η δημιουργία νέων MIB απλοποιείται πολύ με τη χρήση ενός μεταφραστή από τη γλώσσα *GDMO* στη *C++*. Η *GDMO* (Guidelines for the Definition of Managed Objects) είναι μία γλώσσα με τη βοήθεια της οποίας ορίζεται η δομή των διαφόρων MIB που ακολουθούν το μοντέλο πληροφορίας διαχείρισης του *ISO*. Επιπλέον, το *GMS* υποστηρίζει διάφορες βασικές τυποποιημένες λειτουργίες διαχείρισης (management functions) όπως η αναφορά και καταγραφή ειδοποιήσεων (event reporting and logging).

Οπως θα περιγράψουμε αμέσως παρακάτω η λειτουργία της ενημέρωσης της ΚΓΔ που φυλάσσεται στον κατάλογο έχει συμπεριληφθεί στο *GMS*. Με αυτόν τον τρόπο οποιοσδήποτε διαχειριστής ή αντιπρόσωπος αναπτυχθεί χρησιμοποιώντας το σύστημα αυτό ενημερώνει αυτόματα την ΚΓΔ που σχετίζεται με αυτόν κατά την έναρξη της λειτουργίας του. Επιπλέον, ο μηχανισμός διαφάνειας ως προς την τοποθεσία υλοποιήθηκε σαν κομμάτι της διαδικασίας




Σχήμα 3.4: Η γενική αρχιτεκτονική και τα γενικά τμήματα του *OSIMIS*.

δημιουργίας κάθε σύνδεσης (association establishment) η οποία είναι μέρος του CMIS και, συνεπώς, σύμφωνα με όσα προαναφέραμε περιλαμβάνεται και στο *GMS* με τρόπο αθέατο στον προγραμματιστή.

Τα παραπάνω φαίνονται στο σχήμα 3.4 το οποίο δείχνει τα γενικά τμήματα που περιλαμβάνει το *OSIMIS*, τη σχέση τους, τα κομμάτια και τις εφαρμογές που χρησιμοποιεί από το *ISODE* και τη γενική αρχιτεκτονική του όλου συστήματος βάσει των επιπέδων του OSIRM.

Συγκεκριμένα, το σύστημα του σχήματος 3.4 χρησιμοποιεί τα εξής τμήματα του *ISODE*:

- την υλοποίηση του RFC1006 [RC87] σύμφωνα με την οποία δίνεται η δυνατότητα παροχής των διαφόρων υπηρεσιών του *ISO* πάνω από το *Internet* (δηλαδή πάνω από το πρωτόκολλο μεταφοράς TCP)·
- την υλοποίηση των επιπέδων συνόδου (session) και παρουσίασης (presentation)·
- τα στοιχεία υπηρεσιών εφαρμογής (application service elements) ACSE (Association Control Service Element – το οποίο είναι υπεύθυνο για τη δημιουργία συνδέσεων),

ROSE (Remote Operations Service Element – το οποίο είναι υπεύθυνο για την τηλε-εκτέλεση πράξεων σε απομακρυσμένα συστήματα) καθώς και τα σχετικά με την υπηρεσία καταλόγου (Directory Application Service Elements)·

- τους διάφορους μεταφραστές της ASN.1. Πρέπει να τονίσουμε ότι η επεξεργασία της ASN.1 είναι πολύ σημαντική για τις διάφορες κατανεμημένες εφαρμογές του *OSI*. Η προσέγγιση που ακολουθεί σε αυτό το θέμα το πακέτο *ISODE* βασίζεται σε μία βασική δομή, γνωστή ως *Στοιχείο Παρουσίασης* (Presentation Element). Αυτό είναι μία δομή της γλώσσας προγραμματισμού *C* η οποία μπορεί να περιγράφει με αναδρομικό τρόπο κάθε τύπο δεδομένων ο οποίος έχει οριστεί σε ASN.1. Ο βασικός μεταφραστής της ASN.1 – γνωστός σαν "*pepsy*" – παρέχεται μαζί με εμπλουτισμένο συντακτικό (βασισμένο στη γλώσσα *C*) και παράγει ακριβείς αναπαραστάσεις (δηλαδή δομές της *C* οι οποίες αντιστοιχούν στους τύπους της ASN.1) και διάφορες ρουτίνες για την κωδικοποίηση/αποκωδικοποίηση αυτών των αναπαραστάσεων σε/από Στοιχεία Παρουσίασης. Ο ρόλος του επιπέδου παρουσίασης είναι να μετατρέπει τα Στοιχεία Παρουσίασης σε μία ροή δεδομένων σύμφωνα με καλά ορισμένους κανόνες κωδικοποίησης και αντίστροφα·

- την υλοποίηση ενός DSA ο οποίος μπορεί να διασυνδέεται με άλλους DSA χρησιμοποιώντας το πρωτόκολλο DSP και παρέχοντας έτσι έναν κατανεμημένο Καταλόγο·

- την υλοποίηση διαφόρων DUA γενικού σκοπού οι οποίοι χρησιμεύουν για την πρόσβαση στην πληροφορία που κρατιέται σε οποιονδήποτε από τους διασυνδεδεμένους DSA χρησιμοποιώντας το πρωτόκολλο DAP. Σε αυτό το σημείο πρέπει να γίνει σαφές ότι οι DSA και DUA του *ISODE* είναι εφαρμογές γενικού σκοπού στα πλάισια της υπηρεσίας καταλόγου δηλαδή υποστηρίζουν τις διάφορες κλάσεις οντοτήτων του καταλόγου που έχουν καθοριστεί στις τυποποιήσεις. Οι νέες κλάσεις που περιγράψαμε στην προηγούμενη ενότητα αναπτύχθηκαν ξεχωριστά σαν τμήμα αυτής της εργασίας και υποστηρίζονται από τον ειδικό DUA που σχεδιάστηκε για το νέο μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου. Φυσικά για την αναπαράσταση των νέων κλάσεων από την ASN.1 στη γλώσσα *C* χρησιμοποιήθηκε ο μεταφραστής "*pepsy*".

Τα βασικά κομμάτια του *OSIMIS* που φαίνονται στο σχήμα 3.4 είναι τα εξής:

- η προσωπική μας υλοποίηση του ειδικού DUA (SMK/LT DUA) ο οποίος παρέχει τις τρεις λειτουργίες που ήδη περιγράψαμε αναλυτικά στο προηγούμενο κεφάλαιο (δηλαδή την ενημέρωση του Καταλόγου με τη ΚΓΔ για διεργασίες και για πόρους, το μηχανισμό διαφάνειας ως προς την τοποθεσία και το μηχανισμό εύρεσης των διαχειρίσιμων οντοτήτων που αντιπροσωπεύουν στα πλαίσια κάποιας υπηρεσίας διαχείρισης μία αφαίρεση κάποιου πόρου). Ο DUA αυτός υλοποιήθηκε πάνω από τη βιβλιοθήκη του *ISODE* που υλοποιεί τα στοιχεία για την υπηρεσία εφαρμογής του καταλόγου (Directory Service Elements).

- η υλοποίηση του CMISE το οποίο είναι το στοιχείο εφαρμογής που παρέχει την υπηρεσία CMIS/CMIP. Οπως γίνεται φανερό από τη σχετική του θέση στο σχήμα 3.4, το CMISE χρησιμοποιεί τον ειδικό DUA που αναπτύξαμε. Η χρήση αυτή αφορά στο σημείο δημιουργίας της σύνδεσης διαχειριστή/αντιπροσώπου όπου γίνεται η χρήση του μηχανισμού διαφάνειας ως προς την τοποθεσία. Επιπλέον, το CMISE χρησιμοποιεί τα ACSE και ROSE τα οποία είναι απαραίτητα για την επικοινωνία διαχειριστή/αντιπροσώπου·

- το σύστημα *GMS* το οποίο, όπως ήδη αναφέραμε, χρησιμοποιείται για τη δημιουργία αντιπροσώπων διαχείρισης οι οποίοι μπορούν να "ακούν" για αιτήσεις πράξεων διαχείρισης. Βασικά, το *GMS* προσφέρει έναν εύκολο τρόπο ανάπτυξης νέων αντιπροσώπων με νέες διαχειρίσιμες οντότητες, δηλαδή, νεές κλάσεις, χαρακτηριστικά, ειδοποιήσεις, ενέργειες (actions) και λειτουργίες διαχείρισης. Οπως φαίνεται στο σχήμα 3.4 το *GMS* βρίσκεται πάνω από το CMISE προκειμένου να παρέχει τη δυνατότητα επικοινωνίας κάθε αντιπροσώπου με κάποιο διαχειριστή χρησιμοποιώντας το πρωτόκολλο CMIP. Με αυτόν τον τρόπο το *GMS* χρησιμοποιεί το μηχανισμό διαφάνειας ως προς την τοποθεσία κατά τη δημιουργία σύνδεσης – προφανώς στην περίπτωση που ένας αντιπρόσωπος θέλει να στείλει μία ειδοποίηση σε κάποιο διαχειριστή. Επιπλέον, εκτός από αυτόν τον έμμεσο τρόπο χρήσης του DUA από το *GMS*, το σύστημα χρησιμοποιεί άμεσα τον ειδικό μας DUA (βλ. τμήμα του σχήματος στο οποίο το *GMS* είναι πάνω από τον DUA). Το τελευταίο γίνεται για να ενημερώνεται η ΚΓΔ στο Καταλόγο τόσο για τις διεργασίες όσο και για τους πόρους. Ετσι, κάθε αντιπρόσωπος που υλοποιείται με το *GMS* έχει τη δυνατότητα να ενημερώνει την ΚΓΔ που σχετίζεται με αυτόν·

- το σύστημα της απομακρυσμένης MIB (remote MIB - RMIB). Αυτή γενικά προσφέρει ένα CMIS υψηλού επιπέδου το οποίο έχει υλοποιηθεί πάνω από το γνωστό CMIS και κρύβει αρκετές λεπτομέρειες της επικοινωνίας διαχειριστή/αντιπροσώπου. Μερικές τέτοιες λεπτομέρειες είναι η δημιουργία συνδέσεων, η ακριβής ονομασία των διαχειρίσιμων οντοτήτων (χρησιμοποιούνται σύντομα και μνημονικά ονόματα), η αντιμετώπιση λαθών κατά την επικοινωνία και η συναρμολόγηση διασυνδεδεμένων απαντήσεων. Η κεντρική ιδέα είναι η δημιουργία μίας – ιδεατά – μόνιμης σύνδεσης από έναν διαχειριστή σε κάποιον αντιπρόσωπο – ή και αντίστροφα – με τον οποίο η επικοινωνία είναι μάλλον συχνή. Ολες οι λεπτομέρεις γύρω από την υλοποίηση και περιγραφή των λειτουργιών της απομακρυσμένης MIB υπάρχουν στο [Tin93]·

- η υλοποίηση του πρωτοκόλλου *SNMP* [CFSD90] το οποίο αποτελεί το πρωτόκολλο διαχείρισης που έχει προταθεί για χρήση στο Internet. Το SNMP, αν και απλούστερο από το CMIP και περιορισμένων δυνατοτήτων, έχει υιοθετηθεί από πολλούς κατασκευαστές συστήματων για την παροχή αντιπροσώπων διαχείρισης μέσω των οποίων γίνεται η διαχείριση πάνω στα συστήματα αυτά. Ετσι, το *OSIMIS* περιλαμβάνει την υλοποίηση του SNMP και συγχρόνως παρέχει μία εφαρμογή η οποία μετατρέπει πράξεις και πληροφορία διαχείρισης από το μοντέλο του Internet στο μοντέλο του *ISO*· δηλαδή, το *OSIMIS* παρέχει ένα QA (βλ. ενότητα 1.2) για το πρωτόκολλο SNMP [MPBdS95]·
- διάφορες γενικές εφαρμογές σε ρόλο διαχειριστή·
- η υλοποίηση ενός μεταφραστή της γλώσσας GDMO ο οποίος παράγει κώδικα σε *C++* καλύπτοντας το συντακτικό μέρος των ορισμών των διαχειρίσιμων οντοτήτων και αφήνοντας για την υλοποίηση μόνο το σημαντικό μέρος τους (δηλαδή, αυτό που ορίζεται σαν συμπεριφορά – behaviour – της διαχειρίσιμης οντότητας).

Οπως φάνηκε στα παραπάνω, η θέση του ειδικού DUA μας είναι εσωτερική στο *OSIMIS* και επηρεάζει κυρίως το *GMS* και το CMISE με τρόπο αθέατο στον προγραμματιστή νέων διεργασιών που χρησιμοποιεί το *GMS*. Ομως, το *GMS* δεν χρησιμοποιεί όλες τις δυνατότητες του DUA. Ετσι, ο μηχανισμός εύρεσης των διαχειρίσιμων οντοτήτων που αντιπροσωπεύουν κάποιο πόρο και σχετίζονται με κάποια υπηρεσία διαχείρισης είναι κάτι που, παρόλο που υλοποιήθηκε στον DUA, δεν χρησιμοποιείται από κανένα από τα προγραμματιστικά υποσυστήματα (GMS, RMIB) του *OSIMIS*. Αυτό είναι φυσιολογικό εφόσον, όπως προαναφέραμε, ο εν λόγω μηχανισμός στα σημερινά – μη έξυπνα – συστήματα διαχείρισης χρησιμοποιείται βασικά από εφαρμογές τις οποίες χειρίζονται άνθρωποι-διαχειριστές (π.χ.

WS). Επιπλέον, η δυνατότητα ενημέρωσης της πληροφορίας στις οντότητες του καταλόγου που αντιστοιχούν σε πόρους δεν γίνεται από κάθε αντιπρόσωπο άλλα αφήνεται για μελλοντική χρήση από ένα σύστημα το οποίο θα είναι υπεύθυνο για τη "μετα-διαχείριση" του ΔΔΤ [SKGS95]. Επειδή είναι πολύ πιθανό κάποιος προγραμματιστής να χρειαστεί να χρησιμοποιήσει τις επιπλέον δυνατότητες του ειδικού DUA, η επόμενη υποενότητα παραθέτει όλες τις βασικές ρουτίνες που έχουν υλοποιηθεί σαν τμήμα του DUA του σχήματος 3.4 και αναφέρει τις χρήσεις τους.

### 3.2.1 Οι βασικές ρουτίνες του DUA

Η υποενότητα αυτή δεν αποτελεί εγχειρίδιο χρήσης των διαφόρων ρουτινών που υλοποιήσαμε σαν τμήμα αυτής της εργασίας. Προσφέρει μόνο μία γενική περιγραφή των βασικότερων ρουτινών δίνοντας ιδιαίτερη έμφαση στη λειτουργία που επιτελούν και στον τρόπο χρήσης τους. Το ακριβές συντακτικό των ρουτινών που περιγράφουμε και των παραμέτρων τους υπάρχει στο [Sta95] και ο προγραμματιστής παραπέμπεται εκεί για περισσότερες λεπτομέρειες.

#### Ενημέρωση της ΚΓΔ για διεργασίες και πόρους

Η ρουτίνα **dir_smk_update** χρησιμοποιείται για την ενημέρωση της ΚΓΔ η οποία περιέχεται στις οντότητες καταλόγου που αντιπροσωπεύουν διεργασίες και SMAE. Η **res_smk_update** χρησιμοποιείται για την ενημέρωση της ΚΓΔ η οποία περιέχεται στις οντότητες καταλόγου που αντιπροσωπεύουν πόρους.

Η **dir_smk_update** χρησιμοποιείται στο *GMS* τόσο κατά την έναρξη όσο και κατά το τέλος της λειτουργίας μίας διεργασίας διαχείρισης προκειμένου να ενημερώσει τον Κατάλογο με την πληροφορία για τις υπάρχουσες διεργασίες του ΔΔΤ και τις SMAE τους. Οπως προαναφέραμε, στην περίπτωση του τέλους της λειτουργίας μίας διεργασίας αντί να διαγράψουμε την πληροφορία η οποία είναι σχετική με αυτή τη διεργασία την ακυρώνουμε σημειώνοντας τη διεργασία σαν ανενεργή (δίνοντας στο χαρακτηριστικό *operationalStatus* την τιμή "disabled"). Στη γενική περίπτωση η ρουτίνα **dir_smk_update** μπορεί να καλείται οποιαδήποτε στιγμή κατά τη διάρκεια λειτουργίας μίας διεργασίας εφόσον αυτό είναι δυνατό. Για παράδειγμα, αν μελλοντικά υλοποιηθεί ένας μηχανισμός χάρη στον οποίο μία διεργασία θα μπορεί να "μετακομίζει" σε νέα διεύθυνση, χωρίς αυτό να σημαίνει ότι θα πρέπει να είναι

κατά τη μετακόμιση ανενεργή[4], η ρουτίνα αυτή μπορεί να καλείται οποτεδήποτε.
Η **res_smk_update** παρέχεται για χρήση από τις διάφορες διεργασίες που παρέχουν την υπηρεσία της μετα-διαχείρισης στο ΔΔΤ. Το σύστημα που περιγράφεται στην εργασία [SKGS95] χρησιμοποιεί τη ρουτίνα αυτή κάτα την αρχική διαμόρφωση του ΔΔΤ πριν την ενεργοποίηση των διαφόρων διεργασιών διαχείρισης.

### Αναζήτηση κάποιου κομματιού της ΚΓΔ

Η ρουτίνα **dir_smk_search** χρησιμοποιείται για την αναζήτηση κάποιας συγκεκριμένης ΚΓΔ σε κάποιο σημείο του Καταλόγου. Για παράδειγμα, χρησιμοποιώντας αυτήν τη ρουτίνα μπορούμε να βρούμε τις διεργασίες που υποστηρίζουν κάποια κλάση διαχειρίσιμων οντοτήτων μεταξύ των διεργασιών οι οποίες ενεργούν σαν QA σε κάποιο ΔΔΤ. Αυτό γίνεται με τη χρήση κατάλληλων φίλτρων τα οποία περνάμε σαν παραμέτρους. Επίσης ανάλογα με τις παραμέτρους που δίνουμε κατά την κλήση της **dir_smk_search** γίνεται σαφές αν η οντότητα καταλόγου στην οποία ψάχνουμε κάποια πληροφορία αντιστοιχεί σε πόρο, διεργασία ή SMAE.
Η ρουτίνα **lt_read** η οποία χρησιμοποιείται για την εύρεση της διεύθυνσης στην οποία "ακούει" μία SMAE κάποιας διεργασίας χρησιμοποιεί την προηγούμενη γενική ρουτίνα αναζήτησης για το σκοπό αυτό. Η **lt_read** χρησιμοποιείται στη παρούσα έκδοση του *OSIMIS* στην υλοποίηση του CMISE και παρέχει τον τρόπο με τον οποίο βρίσκεται η διεύθυνση στην οποία ακούει μία διεργασία διαχείρισης με την οποία θέλουμε να επικοινωνήσουμε. Το κομμάτι της δημιουργίας της σύνδεσης (association establishment) του CMISE, το οποίο χρησιμοποιεί αυτή τη ρουτίνα, υλοποιεί το μηχανισμό διαφάνειας ως προς την τοποθεσία σύμφωνα με τον οποίο η διεύθυνση επικοινωνίας γίνεται γνωστή με δυναμικό τρόπο από τον Κατάλογο και όχι με στατικό όπως σε παλαιότερες εκδόσεις του *OSIMIS*. Οι παλαιές αυτές εκδόσεις του *OSIMIS* χρησιμοποιούσαν ένα αρχείο (το οποίο ονομάζεται "*isoentities*") και το οποίο εκ των προτέρων έπρεπε να έχει ενημερωθεί με τις διευθύνσεις όλων των διεργασιών διαχείρισης που λειτουργούσαν στο ΔΔΤ προκειμένου να επικοινωνήσει μία διεργασία με οποιαδήποτε άλλη. Η χρήση αυτού του αρχείου ήταν πολύ περιοριστική για περιβάλλοντα όπως το ΔΔΤ: η αλλαγή της διεύθυνσης μίας και μόνο διεργασίας απατούσε συχνά την ενημέρωση μίας πλειάδας αρχείων το οποία κρατούνταν σε διαφορετικά συστήματα. Επιπλέον, καθιστούσε ανέφικτους μηχανισμούς όπως η δυναμική μετακόμιση διεργασιών σε περίπτωση κάποιας βλάβης στο ΔΔΤ.

[4]Ένας τέτοιος μηχανισμός θα πρέπει να προσφέρει διαφάνεια ως προς τη μετακόμιση (migration transparency) και είναι επιθυμητός σε συστήματα όπως το ΔΔΤ.

### Λίστα των υπαρχόντων διεργασιών διαχείρισης

Χρησιμοποιώντας τη ρουτίνα **dir_smk_search** μπορούμε να παράσχουμε μία σειρά από ρουτίνες οι οποίες χρησιμοποιούνται για τη δημιουργία διαφόρων λιστών. Ενδεικτικά, παρέχουμε δύο λίστες χρησιμοποιώντας τις παρακάτω ρουτίνες. Η βασική τους χρήση βρίσκεται μάλλον σε εφαρμογές για τον άνθρωπο-διαχειριστή:

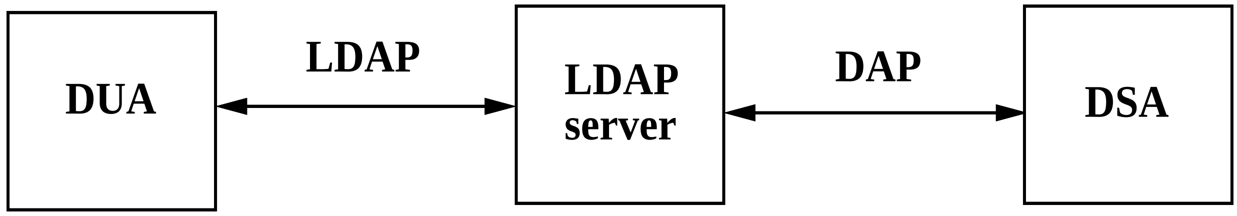
- **λίστα όλων των ενεργών διεργασιών διαχείρισης ενός ΔΔΤ (dir_get_mp).** Η κλήση της **dir_get_mp** επιστρέφει τα DN όλων των ενεργών (*operationalStatus = enabled*) διεργασιών. Οι οντότητες που αντιστοιχούν σε αυτές τις διεργασίες βρίσκονται κάτω από το ίδιο υπόδεντρο του Καταλόγου – σε ένα σωστά οργανωμένο Κατάλογο αυτό σημαίνει ότι οι διεργασίες ανήκουν στο ίδιο ΔΔΤ.
- **λίστα όλων των διεργασιών που αντιπροσωπεύουν έναν πόρο (mp_per_res).** Η κλήση της **mp_per_res** επιστρέφει τα DN των διεργασιών οι οποίες αντιπροσωπεύουν κάποιο πόρο. Αυτό σημαίνει ότι τα ονόματα αυτών των διεργασιών εμφανίζονται στις τιμές των χαρακτηριστικών *responsibleSMAP* της οντότητας καταλόγου που αντιστοιχεί στον εν λόγω πόρο.

Και οι δύο παραπάνω ρουτίνες αντιστοιχούν σε κλήσεις της **dir_smk_search** χρησιμοποιώντας κατάλληλες παραμέτρους ενώ περιλαμβάνουν την απαραίτητη επεξεργασία στα αποτελέσματα που επιτρέφει κάθε κλήση τής **dir_smk_search**.

### Η υπηρεσία καταλόγου στο *Internet*: το πρωτόκολλο *LDAP*

Οσο και αν η ύπαρξη πολλών τεχνολογιών δικτύων είναι γεγονός, η σημερινή πραγματικότητα καθιστά τα δίκτυα που βασίζονται στα πρωτόκολλα TCP/IP τη συντριπτική πλειοψηφία. Ετσι και οι διάφορες διεργασίες του *OSIMIS* χρησιμοποιούνται σήμερα, κατά κανόνα, πάνω από τέτοια δίκτυα τα οποία διασυνδέονται μεταξύ τους σχηματίζοντας το γνωστό Internet. Αυτό είναι δυνατό χάρη στη χρήση του μηχανισμού που περιγράφεται στο RFC[5] 1006 [RC87] και ο οποίος υλοποιείται από το *ISODE* με αποτέλεσμα οι διάφορες υπηρεσίες του *ISO* να μπορούν να παρέχονται στο Internet.
Στην περίπτωση της υπηρεσίας καταλόγου έχει προταθεί στο Internet η χρήση ενός πρωτοκόλλου το οποίο ονομάζεται *ελαφρύ DAP* ή *LDAP* (Leightweight Directory Access

[5] Τα RFC (Request For Comments) αποτελούν κάτι σαν τυποποιήσεις ή συστάσεις των διαφόρων πρωτοκόλλων που χρησιμοποιούνται στο Internet.




Σχήμα 3.5: Επικοινωνία μεταξύ DUA και DSA χρησιμοποιώντας το πρωτόκολλο LDAP.

Protocol) και το οποίο περιγράφεται στο RFC 1487 [YHK93]. Το LDAP έχει προταθεί για την πρόσβαση στην υπηρεσία καταλόγου πάνω από το TCP/IP. Η βασική προϋπόθεση για τη χρήση του LDAP είναι να υπάρχει κάποιος DSA ο οποίος να μπορεί να επικοινωνεί χρησιμοποιώντας το πρωτόκολλο DAP πάνω από το TCP/IP. Ο DSA που παρέχει το *ISODE* (πιο συγκεκριμένα, το τμήμα του *ISODE* το οποίο είναι γνωστό σαν *QUIPU*) σύμφωνα με τα όσα έχουμε αναφέρει μπορεί να παρέχει την υπηρεσία καταλόγου πάνω από τα πρωτόκολλα TCP/IP.

Το βασικό πλεονέκτημα που προσφέρει το LDAP σε σχέση με το DAP είναι ότι παρακάμπτει την πολυπλοκότητα των επιπέδων συνόδου (session) και παρουσίασης (presentation) χρησιμοποιώντας απλούς κανόνες για την κωδικοποίηση των διαφόρων μηνυμάτων τα οποία ανταλλάσσονται μεταξύ DUA και DSA. Το μειονέκτημα κατά τη χρήση του LDAP είναι ότι δεν προσφέρει τις δυνατότητες ελέγχου της ποιότητας της παροχής της υπηρεσίας καταλόγου που προσφέρει το DAP. Για παράδειγμα, χρησιμοποιώντας το κλασσικό DAP ο DUA μπορεί να ορίσει αν θέλει να έχει πρόσβαση μόνο σε κύρια πληροφορία[6] ενός DSA ή αν δεν έχει πρόβλημα η πληροφορία την οποία προσπελαύνει να αποτελεί ένα αντίγραφο της κύριας πληροφορίας κάποιου DSA. Το επιπλέον στοιχείο που εισάγεται με το LDAP είναι η μονάδα εξυπηρέτησης του LDAP (LDAP server) μεταξύ DUA και DSA. Αυτή μετατρέπει τα μηνύματα του DUA (τα οποία στέλνονται με το LDAP) σε μηνύματα για τον DSA (τα οποία στέλνονται με το κλασσικό DAP) και αντίστροφα. Η παραπάνω λειτουργία φαίνεται στο σχήμα 3.5.

Το όφελος που έχουμε είναι σημαντικό από άποψη πολυπλοκότητας του DUA: Οι DUA που

[6]Κάθε κομμάτι πληροφορίας του καταλόγου μπορεί να κρατιέται σε περισσότερους από έναν DSA. Ενας από αυτούς τους DSA ορίζεται ο κύριος (master) DSA για αυτήν την πληροφορία και είναι ο DSA ο οποίος διαθέτει πάντα την πιο ενημερωμένη έκδοση της πληροφορίας. Ως κύρια πληροφορία ενός DSA εννοούμε την πληροφορία στην οποία ο DSA αυτός είναι κύριος.

χρησιμοποιούν LDAP είναι προφανώς πολύ απλούστεροι, εφόσον δεν περιλαμβάνουν πολύπλοκες κωδικοποιήσεις στο επίπεδο παρουσίασης του *OSI*. Ετσι, ο όγκος του εκτελέσιμου προγράμματος του DUA μειώνεται δραστικά. Επειδή στην περίπτωση των διεργασιών διαχείρισης ο όγκος των εκτελέσιμων προγραμμάτων είναι αρκετά μεγάλος, προσφέρθηκε εναλλακτικά μία υλοποίηση του DUA του σχήματος 3.4 η οποία χρησιμοποιεί το LDAP. Η υλοποίηση αυτή δίνει εκτελέσιμα τα οποία είναι περίπου κατά 0,5 MBytes μικρότερα από τα αντίστοιχα που χρησιμοποιούν το κλασσικό DAP. Η προγραμματιστική βιβλιοθήκη παρέμεινε ίδια ενώ κατά την εγκατάσταση του *OSIMIS* επιλέγεται αν θα χρησιμοποιηθεί το DAP ή το LDAP. Η επιλογή του LDAP, όπως είναι φυσικό, απαιτεί και τη χρήση ενός πακέτου υλοποίησης του πρωτοκόλλου LDAP το οποίο θα περιλαμβάνει και τον αντίστοιχο LDAP server. Το πακέτο που χρησιμοποιήθηκε ήταν η έκδοση LDAP 3.0 του πανεπιστημίου του Michigan το οποίο διατίθεται δωρεάν.

# Κεφάλαιο 4

# Συμπεράσματα και μελλοντικές επεκτάσεις

Η υπηρεσία καταλόγου του *ISO* παρέχει τη δυνατότητα πρόσβασης (ανάγνωσης και γραφής) σε μία κατανεμημένη βάση δεδομένων (η οποία καλείται Κατάλογος) σε παγκόσμιο επίπεδο. Στην παρούσα εργασία εκμεταλλευτήκαμε τα πρωτόκολλα της υπηρεσίας καταλόγου προκειμένου να αποθηκεύσουμε ένα σημαντικό τμήμα της πληροφορίας που αναφέρεται ως "κοινή γνώση διαχείρισης" σε δίκτυα διαχείρισης τηλεπικοινωνιών. Στηριζόμενοι στο γεγονός ότι η κοινή γνώση διαχείρισης αποτελεί μία σχετικά στατική πληροφορία επεκτείναμε το μοντέλο επικοινωνίας διαχειριστή/αντιπροσώπου του *ISO* συμπεριλαμβάνοντας σε αυτό την αυτόματη και κατά βούληση ενημέρωση της ΚΓΔ καθώς και τη δυναμική απόκτησή της από τις διάφορες διεργασίες διαχείρισης.

Χρησιμοποιώντας το νέο αυτό μοντέλο δείξαμε πώς μπορούμε να παράσχουμε ένα μηχανισμό που να βρίσκει τις διαχειρίσιμες οντότητες που αντιπροσωπεύουν μία ζητούμενη αφαίρεση κάποιου διαχειριζόμενου πόρου ενώ επιπλέον υλοποιήσαμε ένα μηχανισμό διαφάνειας ως προς την τοποθεσία κατά την επικοινωνία διαχειριστή/αντιπροσώπου.

Τα παραπάνω στηρίχτηκαν στην απόκτηση της ΚΓΔ μέσω της υπηρεσίας καταλόγου. Σημαντικό είναι το γεγονός ότι χάρη στην παγκοσμιότητα της υπηρεσίας καταλόγου είναι δυνατό να αποκτηθεί η ΚΓΔ που απαιτεί μία διεργασία προκειμένου να επικοινωνήσει με κάποια άλλη η οποία ανήκει ακόμα και σε κάποιο διαφορετικό ΔΔΤ. Οι μηχανισμοί που περιγράψαμε υλοποιήθηκαν και δοκιμάστηκαν τόσο από άποψη χρησιμότητας όσο και απόδοσης σε σημερινά πειραματικά ΔΔΤ τα οποία λειτούργησαν στα πλαίσια του έργου ICM (Integrated Communications Management) – έργο R2059 του RACEII – της Ευρωπαϊκής Ενωσης.

Πρέπει να σημειωθεί ότι ο μηχανισμός εύρεσης των διαχειρίσιμων οντοτήτων που αντιπροσωπεύουν μία ζητούμενη αφαίρεση κάποιου διαχειριζόμενου πόρου, στα σημερινά ΔΔΤ, χρησιμοποιείται κυρίως από τον άνθρωπο-διαχειριστή ο οποίος στη συνήθη περίπτωση γνωρίζει τον πόρο που θέλει να διαχειριστεί και όχι τη διεργασία που περιέχει τις διαχειρίσιμες οντότητες στις οποίες πρέπει να εκτελεστούν τελικά κάποιες πράξεις. Από την άλλη οι διάφορες διεργασίες οι οποίες ενεργούν σε ρόλο διαχειριστή αναλαμβάνουν προκαθορισμένες λειτουργίες και επικοινωνούν με αντιπροσώπους διαχείρισης οι οποίοι είναι γνωστοί εκ των προτέρων με αποτέλεσμα να μη χρειάζονται τον παραπάνω μηχανισμό.
Σε μελλοντικά συστήματα διαχείρισης αναμένεται η όλη διαδικασία της διαχείρισης των διαφόρων υπηρεσιών και πόρων να είναι περισσότερο αυτοματοποιημένη. Τα ΔΔΤ θα πρέπει να ενέχουν την απαραίτητη λογική και εξυπνάδα για τη δυναμική λήψη αποφάσεων καθώς και για το δυναμικό προσδιορισμό των πόρων και των υπηρεσιών που πρέπει να διαχειριστούν. Σε τέτοια "έξυπνα" συστήματα – τα οποία πιθανώς θα χρησιμοποιούν κάποιες από τις τεχνικές της τεχνητής νοημοσύνης – η εύρεση των κατάλληλων διεργασιών με τις οποίες πρέπει να υπάρξει επικοινωνία θα γίνεται δυναμικά στις διάφορες διεργασίες-διαχειριστές. Ετσι, οι παραπάνω μηχανισμοί θα είναι αναγκαίοι για τη σωστή λειτουργία των ΔΔΤ αυτών.
Τέλος, στην παρούσα εργασία στηριζόμενοι στο μηχανισμό διαφάνειας ως προς την τοποθεσία δείξαμε τη δυνατότητα παροχής της υπηρεσίας ενός καθολικού CMIS το οποίο βασίζεται στην ψευδαίσθηση μίας καθολικής πληροφορίας διαχείρισης, αποκρύπτοντας από τις διεργασίες διαχείρισης τις λεπτομέρειες δημιουργίας των συνδέσεων. Χάρη στην υπηρεσία του καθολικού CMIS το γεγονός της κατανομής των διαφόρων διεργασιών και της πληροφορίας διαχείρισης αποκρύπτεται από τις επιμέρους διεργασίες οι οποίες αντιλαμβάνονται το ΔΔΤ σαν ένα κεντρικοποιημένο σύστημα. Το σημαντικό κέρδος είναι ότι αυτό επιτεύχθηκε χρησιμοποιώντας την υπηρεσία καταλόγου από τις διεργασίες που ακολουθούν τα κλασσικά πρωτόκολλα και μοντέλα του *ISO* για τη διαχείριση συστημάτων, αποφεύγοντας τη χρήση νέων μοντέλων όπως αυτό της *Ανοιχτής Κατανεμημένης Επεξεργασίας* (*Open Distributed Processing – ODP*) [Org94a] το οποίο, αν και προσφέρει πολλές από τις αναγκαίες λειτουργίες σε κατανεμημένα συστήματα, δεν κρίθηκε απαραίτητο, αφού μάλλον θα περιέπλεκε παρά θα διευκόλυνε τους μηχανισμούς που προτάθηκαν στα πλαίσια αυτής της εργασίας.

## 4.1 Μελλοντικές επεκτάσεις

Οι μηχανισμοί που περιγράψαμε μπορούν να χρησιμοποιηθούν τόσο στο ΔΔΤ όσο και σε διαφορετικά περιβάλλοντα διαχείρισης τα οποία ακολουθούν παρόμοια ιεραρχική μεθοδολογία για τη διαχείριση δικτύων και υπηρεσιών από ένα σύνολο διασυνδεδεμένων διεργασιών. Παρόλο που τα όσα περιγράψαμε λειτούργησαν σε περιβάλλοντα διαχείρισης πόρων, όπως προαναφέραμε, η αποθήκευση της ΚΓΔ σχετικά με διαχειριζόμενες υπηρεσίες με τρόπο παρόμοιο με την περίπτωση των πόρων είναι δυνατή. Στα μελλοντικά δίκτυα (π.χ. ATM), όπου η διαχείριση των παρεχομένων υπηρεσιών θα είναι το κεντρικό αντικείμενο ενός ΔΔΤ, αναμένεται ένας μηχανισμός εύρεσης των διαχειρίσιμων οντοτήτων που αντιπροσωπεύουν τις υπηρεσίες που θέλουμε να διαχειριστούμε παρόμοιος με αυτόν που περιγράψαμε για την περίπτωση των πόρων.
Λόγω της κατανεμημένης φύσης συστημάτων όπως είναι το ΔΔΤ, μελλοντικά θα είναι αναγκαία η παροχή μηχανισμών όπως η διαφάνεια βλαβών (failure transparency) και η διαφάνεια μετακόμισης (migration transparency) των διεργασιών διαχείρισης. Η διαφάνεια βλαβών αναφέρεται στους μηχανισμούς οι οποίοι θα εξασφαλίζουν τη συνεχή λειτουργία και παροχή των υπηρεσιών διαχείρισης από το ΔΔΤ ακόμα και στην περίπτωση βλαβών είτε σε συνδέσμους, είτε σε συστήματα, είτε σε διεργασίες του ΔΔΤ. Οι μηχανισμοί αυτοί προφανώς θα συμπεριλαμβάνουν λειτουργίες όπως η συντήρηση αντιγράφων (replication) των διαφόρων διεργασιών του ΔΔΤ και η αντικατάσταση μίας διεργασίας που δεν είναι προσβάσιμη λόγω βλάβης από κάποιο αντίγραφό της.
Η διαφάνεια μετακόμισης αναφέρεται στους μηχανισμούς οι οποίοι θα εξασφαλίζουν τη συνεχή πρόσβαση σε μία διεργασία ακόμα και σε περιπτώσεις που αυτή πρέπει να μετακομίσει σε κάποια άλλη διεύθυνση. Η συνεχής πρόσβαση είναι φυσικά ιδεατή και όχι πραγματική που σημαίνει ότι οι διεργασίες που επικοινωνούν με τη διεργασία που μετακομίζει δεν θα πρέπει να αντιλαμβάνονται το γεγονός της μετακόμισης. Αυτή η διαφάνεια μπορεί να προαπαιτείται για την προηγούμενη εφόσον ένας από τους τρόπους αντιμετώπισης κάποιας βλάβης είναι η μετακόμιση των διεργασιών σε άλλα συστήματα του ΔΔΤ. Ο τρόπος με τον οποίο μπορεί να μετακινηθεί μία διεργασία κάπου αλλού στο ΔΔΤ, χωρίς αυτό να επηρεάσει τις υπάρχουσες ή μελλοντικές συνδέσεις της διεργασίας, είναι ένα από τα δύσκολα προβλήματα που πρέπει να λυθούν. Στο μέλλον, αναμένεται ότι οι παραπάνω μηχανισμοί θα παρέχονται στο γενικό μοντέλο του *ODP* και συνεπώς το τελευταίο ίσως αποτελέσει τη βάση για την ανάπτυξη μελλοντικών ΔΔΤ.

Ενα από τα θέματα που παραλείψαμε στην παρούσα εργασία είναι η παροχή κάποιου μηχανισμού ελέγχου της πρόσβασης στην ΚΓΔ. Σε αρκετές περιπτώσεις είναι δυνατό να απαιτηθεί κάποιο κομμάτι της ΚΓΔ να μην είναι διαθέσιμο σε οποιονδήποτε μπορεί να έχει πρόσβαση στον Κατάλογο μέσω κάποιου DUA. Γι' αυτόν το λόγο θα πρέπει να υπάρχει η δυνατότητα επιβολής συγκεκριμένου ελέγχου της πρόσβασης στην πληροφορία του Καταλόγου που αντιστοιχεί στη ΚΓΔ. Η υπηρεσία καταλόγου, όπως είδαμε, προσφέρει τη δυνατότητα ελέγχου της πρόσβασης στην πληροφορία κάθε εγγεγραμμένης οντότητας μέσω των τιμών του χαρακτηριστικού *acl* το οποίο είναι υποχρεωτικό για κάθε οντότητα καταλόγου. Αυτή η δυνατότητα χρησιμοποιήθηκε εν μέρει. Ετσι, στην παρούσα υλοποίηση η ΚΓΔ ενημερώνεται από τις διεργασίες μόνο εφόσον αυτές γνωρίζουν κάποια κωδική λέξη (password) και συνδεθούν στο X.500 χρησιμοποιώντας αυτή τη λέξη προκειμένου να αποκτήσουν τη σχετική άδεια ενημέρωσης του Καταλόγου.
Παρόλα αυτά από τη στιγμή που η ΚΓΔ αποθηκευτεί στον Κατάλογο οποιοσδήποτε μπορεί να τη διαβάσει. Αυτό δεν είναι τόσο καθοριστικό αν ο έλεγχος της πρόσβασης γίνεται στο επίπεδο επικοινωνίας μεταξύ των διεργασιών διαχείρισης. Πραγματικά, αν κάποιος δεν μπορεί να χρησιμοποιήσει την ΚΓΔ προκειμένου να συνδεθεί με οποιαδήποτε διεργασία θελήσει και να εκτελέσει οποιαδήποτε αυθαίρετη πράξη ή να λάβει οποιαδήποτε πληροφορία διαχείρισης, τότε μόνο το γεγονός κατοχής της ΚΓΔ μπορεί να θεωρηθεί αβλαβές για την ασφάλεια ενός ΔΔΤ. Ετσι, περισσότερο επιτακτικοί είναι κάποιοι κατάλληλοι μηχανισμοί πιστοποίησης ταυτότητας (authentication) και κρυπτογράφησης κατά την επικοινωνία διαχειριστή/αντιπροσώπου ώστε να διασφαλιστεί το απόρρητο και η ακεραιότητα της πληροφορίας και της λειτουργίας του ΔΔΤ. Η τελευταία έκδοση του *OSIMIS* περιλαμβάνει ένα μηχανισμό πιστοποίησης ταυτότητας κατά την επικοινωνία δύο διεργασιών διαχείρισης. Γενικά, η ασφάλεια του ΔΔΤ είναι ένα ξεχωριστό και πολύπλοκο θέμα το οποίο περιλαμβάνει ειδικούς μηχανισμούς οι οποίοι ξεφεύγουν από τα όρια αυτής της εργασίας.
Τέλος, μία σημαντική έννοια που σχετίζεται με την παρούσα εργασία είναι η μετα-διαχείριση (metamanagement). Αυτή αναφέρεται στη διαχείριση του ιδίου του ΔΔΤ και σχετίζεται με θέματα όπως η αποκατάσταση εσωτερικών βλαβών του ΔΔΤ και η διαμόρφωση και επαναδιαμόρφωση (configuration and reconfiguration) του ΔΔΤ. Ειδικά για τη διαμόρφωση και επαναδιαμόρφωση του ΔΔΤ αναφέραμε ότι περιλαμβάνουν την ενημέρωση της πληροφορίας του καταλόγου σχετικά με το ποιες διεργασίες αντιπροσωπεύουν ποιους πόρους η οποία αποτελεί τμήμα της ΚΓΔ και αποθηκεύεται στις οντότητες καταλόγου που αντιστοιχούν σε πόρους. Ο λόγος που αυτή η πληροφορία ενημερώνεται από το υποσύστημα του ΔΔΤ που

είναι υπεύθυνο για τη μετα-διαχείριση είναι ότι η εν λόγω πληροφορία δίνεται κατά τη σχεδίαση του ΔΔΤ οπότε και αποφασίζεται ποιες διεργασίες θα αντιπροσωπεύουν ποιους πόρους. Σε περιπτώσεις επανασχεδίασης και επαναδιαμόρφωσης του ΔΔΤ η πληροφορία αυτή πρέπει πάλι να αλλαχτεί και αυτό προφανώς γίνεται από το ίδιο υποσύστημα.
Η υλοποίησή μας περιλαμβάνει μία ρουτίνα η οποία ενημερώνει αυτήν την πληροφορία και η οποία παρέχεται κυρίως για χρήση από τις διεργασίες οι οποίες αναλαμβάνουν τη μετα-διαχείριση (η οποία, ας σημειωθεί, είναι μία από τις υπηρεσίες διαχείρισης που έχει ορίσει η *ITU* στο [Uni93e]). Η εργασία [SKGS95] χρησιμοποιεί τη ρουτίνα αυτή προκειμένου να ενημερώσει την παραπάνω ΚΓΔ στον Κατάλογο. Οπως αναφέρεται στην εργασία [SKGS95] το νέο μοντέλο διαχειριστή/αντιπροσώπου που περιγράψαμε εδώ είναι απαραίτητο και βρίσκει εφαρμογή και στην περίπτωση της μετα-διαχείρισης, όπου η υπηρεσία καταλόγου χρησιμοποιείται για την αποθήκευση και απόκτηση επιπρόσθετης πληροφορίας διαφορετικής από την ΚΓΔ.

# Βιβλιογραφία


[CFSD90] J.D Case, M. Fedor, M.L. Schoffstall, and C. Davin. Simple Network Management Protocol (SNMP). Request for Comments 1157, DDN Network Information Center, May 1990.

[DzNT93] A. Dittrich, U.M. zu Natrup, and P.G. Tsigaridas. Representation of management schema information. Technical Report 6331, GMD–FOKUS, Berlin, 1993.

[Kil91] Stephen E. Kille. *Implementing X.400 and X.500 : the PP and QUIPU Systems.* Artech House, 1991.

[KR88] B. Kernighan and D. Ritchie. *The C Programming Language.* Prentice Hall, second edition, 1988.

[Mar90] Rose T. Marshall. *The Open Book: A practical perspective on OSI.* Prentice Hall, 1990.

[MJK94] G. Mansfield, T. Johannsen, and M. Knopper. Charting networks in the X.500 Directory. Request for Comments 1609, DDN Network Information Center, March 1994.

[MPBdS95] K. McCarthy, G. Pavlou, S. Bhatti, and J. Neuman de Souza. Exploiting the power of OSI management for the control of SNMP-capable resources using generic application level gateways. In *Integrated Network Management IV, Chapman and Hall*, pages 440–453, 1995.

[MRR91] J. Onions M. Rose and C. Robbins. *The ISO Development Environment: User's Manual.* PSI Inc. / X-tel Services Ltd., 7.0 edition, August 1991.

[Org87] International Standards Organization. Information Technology – Open Systems Interconnection – protocol for providing the connectionless-mode network service. Draft International Standard 8473, ISO/IEC, 1987.

[Org90a] International Standards Organisation. Information Technology – Open Systems Interconnection – Common Management Information Protocol Specification. International Standard 9596, ISO/IEC, 1990.

[Org90b] International Standards Organization. Information Technology – Open Systems Interconnection – Common Management Information Services Definition. International Standard 9595, ISO/IEC, 1990.

[Org90c] International Standards Organization. Information Technology – Open Systems Interconnection – specification of Abstract Syntax Notation One (ASN.1). International Standard 8824, ISO/IEC, 1990.

[Org92a] International Standards Organization. Information Technology – Open Systems Interconnection – systems management overview. International Standard 10040, ISO/IEC, 1992.

[Org92b] International Standards Organization. Information Technology – Open Systems Interconnection – structure of management information: Management information model. International Standard 10165-1, ISO/IEC, 1992.

[Org92c] International Standards Organization. Information Technology – Open Systems Interconnection – structure of management information: Definition of management information. International Standard 10165-2, ISO/IEC, 1992.

[Org92d] International Standards Organization. Information Technology – Open Systems Interconnection – structure of management information: Guidelines for the definition of managed objects. International Standard 10165-4, ISO/IEC, 1992.

[Org94a] International Standards Organization. Information Technology – Open Distributed Processing – reference model – Open Distributed Processing – part1: Overview and guide to use. Committee Draft 10746-1, ISO/IEC, 1994.

[Org94b] International Standards Organization. Information Technology – Open Systems Interconnection – Basic Reference Model. International Standard 7498, ISO/IEC, 1994.

[Org95] International Standards Organization. Information Technology – Open Systems Interconnection – systems management: Management knowledge management function. Draft International Standard 10164-16, ISO/IEC, 1995.

[PMBK95] G. Pavlou, K. McCarthy, S. Bhatti, and G. Knight. The OSIMIS platform: making OSI management simple. In *Integrated Network Management IV, Chapman and Hall*, pages 480–493, 1995.

[Pos81a] J.B. Postel. Internet Protocol. Request for Comments 791, DDN Network Information Center, October 1981.

[Pos81b] J.B. Postel. Transmission Control Protocol. Request for Comments 793, DDN Network Information Center, September 1981.

[RC87] M. Rose and D. Case. ISO transport services on top of the TCP. Request for Comments 1006, DDN Network Information Center, May 1987.

[SKGS95] S. Sartzetakis, V. Kalogeraki, D. Griffin, and C. Stathopoulos. Managing the TMN. In *Proceedings of the 3rd international conference on Intelligence in Broadband Services and Networks*, 1995.

[Sta95] C. Stathopoulos. *Shared Management Knowledge and Location Transparency Support via the X.500 Directory in OSIMIS: User's Manual.* Institute of Computer Science, Foundation for Research and Technology – Hellas, 2.5 edition, August 1995.

[Str86] B. Stroustrup. *The C++ Programming Language.* Addison-Wesley, 1986.

[Syl93] M. Sylor. Junction objects. In *Integrated Network Management III*, 1993.

[TD93] M. Tschichholz and W. Donnelly. The PREPARE management information service. In *Proceedings of the 5th RACE IS&N Conference*, 1993.

[Tin93] T. Tin. *The Remote MIB Manager Support API: User's Guide.* Unipro Limited, 1.1 edition, July 1993.

[Uni88] International Telecommunication Union. Interface between DTE and DCE terminals operating in packet mode. Recommendation X.25, ITU-T, 1988.

[Uni93a] International Telecommunication Union. Information Technology – Open Systems Interconnection – the Directory: Models. Recommendation X.501, ITU-T, 1993.

[Uni93b] International Telecommunication Union. Information Technology – Open Systems Interconnection – the Directory: Selected attribute types. Recommendation X.520, ITU-T, 1993.

[Uni93c] International Telecommunication Union. Information Technology – Open Systems Interconnection – the Directory: Selected object classes. Recommendation X.521, ITU-T, 1993.

[Uni93d] International Telecommunication Union. Principles for a Telecommunications Management Network. Recommendation M.3010, ITU-T, 1993.

[Uni93e] International Telecommunication Union. TMN interface specification methodology. Recommendation M.3020, ITU-T, 1993.

[Uni93f] International Telecommunication Union. TMN Management Services: Overview. Recommendation M.3200, ITU-T, 1993.

[YHK93] W. Yeong, T. Howes, and S. Kille. X.500 Lightweight Directory Access Protocol. Request for Comments 1487, DDN Network Information Center, July 1993.