

Κεφάλαιο Δεκατέσσερα
—
Υποδομές Συστατικών

14.1 Οι Συνεισφορές της Υποδομής Συστατικών

- Οι σκοποί μίας υποδομής συστατικών είναι:
 - Η συλλογή χρήσιμων διευκολύνσεων.
 - Η υποχρέωση των περιπτώσεων συστατικών (που εισάγονται δυναμικά στην υποδομή κατά την εκτέλεση), να εκτελούν ορισμένους στόχους μέσω μηχανισμών υπό τον έλεγχο της υποδομής.

14.2.1 OpenDoc - Αρχικές Παρατηρήσεις

- Το OpenDoc σταμάτησε να αναπτύσσεται το 1997 από την Apple. Ήταν από τις λίγες υποδομές συστατικών που αναπτύχθηκαν σε επίπεδο προϊόντος.
- Αποτελεί καλό αντικείμενο μελέτης αφ' ότου δόθηκε ο κώδικάς του στο κοινό από την IBM.
- Αποτελεί διευκόλυνση CORBA με καθορισμούς OMG και θεωρητικά υποστηρίζεται από πολλά βιομηχανικά σύνολα. Οι εφαρμογή από τους καθορισμούς διαφέρει. Οι παρούσες εφαρμογές βρίσκονται σε IBM SOM ενώ η OMG OpenDoc απλώς χρειάζεται CORBA 2.0 και καθορισμούς υπηρεσιών CORBA.
- Η υποδομή OpenDoc μπήκε στη αγορά νωρίς, όταν η δυναμική των σύνθετων κειμένων και οπτικών συστατικών ήταν αφανέρωτη.
- Ως αναφορά την σχέση του με τις διασυνδέσεις OLE αυτές απλά καθορίζουν περιβάλλον συστατικών. Με την ανάπτυξη του Internet Explorer IE στο κέλυφος ActiveX, το IE θα μπορούσε να γίνει η τελική ActiveX υποδομή συστατικού.
- Η JavaBeans έχει μάλλον αδύνατους κανόνες, εστιάζεται στο εύκολο μέρος σε συστατικά που δεν είναι υποδοχείς.

14.2.2 Θεώρηση της OpenDoc

- Το OpenDoc σαν υποδομή ασχολείται με κάθε τα μέρη σύνθετου κειμένου, που και αυτά μπορεί να περιέχουν άλλα μέρη. Τα μέρη περιέχουν τοπικά δεδομένα και (αν είναι υποδοχείς) άλλα μέρη. Κάθε μέρος συνδέεται με ένα εκδότη. Οι χρήστες επιλέγουν τους εκδότες και το είδος των μερών.
- Σε σύγκριση οι διασυνδέσεις OLE δεν υποστηρίζουν τα απλά χαρακτηριστικά των εκδοτών. Αν ένα ενσωμάτωμα ανήκει στην Word πρέπει να αλλάξει σχήμα.
- Στην υποδομή OpenDoc ορίζονται:
 - Δομημένα αρχεία για σύνθετα κείμενα.
 - Μηχανισμοί για ομοιόμορφη μεταφορά δεδομένων.
 - Αρχιτεκτονική γραφής.

Όλ' αυτά είναι παρόμοια στην φύση τους με τους μηχανισμούς διασυνδέσεων OLE (στη μεταφορά δεδομένων COM).

- Στη σύνταξη για αποστολή πακέτων χρησιμοποιείται το SOM.
- Κατανομή και μεταγλώττιση διασυνδέσεων μερών από OMG IDL. Διάθεσις μέσω προτύπου SOM.
- Κατανομή εφαρμογών μερών σαν DLL. Διάθεσις μέσω προτύπου SOM.

Τα μέρη εκτελούν στη ίδια επεξεργασία του υποδοχέα τους, δηλαδή υπάρχει μία επεξεργασία για κάθε σύνθετο κείμενο.

- Ο μηχανισμός συμπλήρωσης δομής αρχείων, γίνεται με την ξεχωριστή εφαρμογή Bento. Σ' αυτό υπάρχει υποστήριξη αριθμού έκδοσης και φύλαξη δένδρου εκδόσεων. Αντί εκτέλεσης αλλαγής φυλάσσεται το μέρος που έχει μεταβληθεί.
- Η αρχιτεκτονική γραφής είναι το Open Scripting Architecture από την AppleScript. Το κύριο πλαίσιο γραφής καθορίζεται από:
 - Ανεξαρτησία γλώσσας γραφής (μεταγλώττιση γραφής σε αποστολές ακολουθειών ιδεατών σημαντικών συμβάντων, σε προορισμούς ορισμένους με καθοριστές αντικειμένων).
 - Σημαντικά συμβάντα (με καταχωρημένη σημασιολογία).
 - Λογικούς χαρακτηρισμούς κειμένου (βλέπε COM monikers).

Θεώρηση της OpenDoc

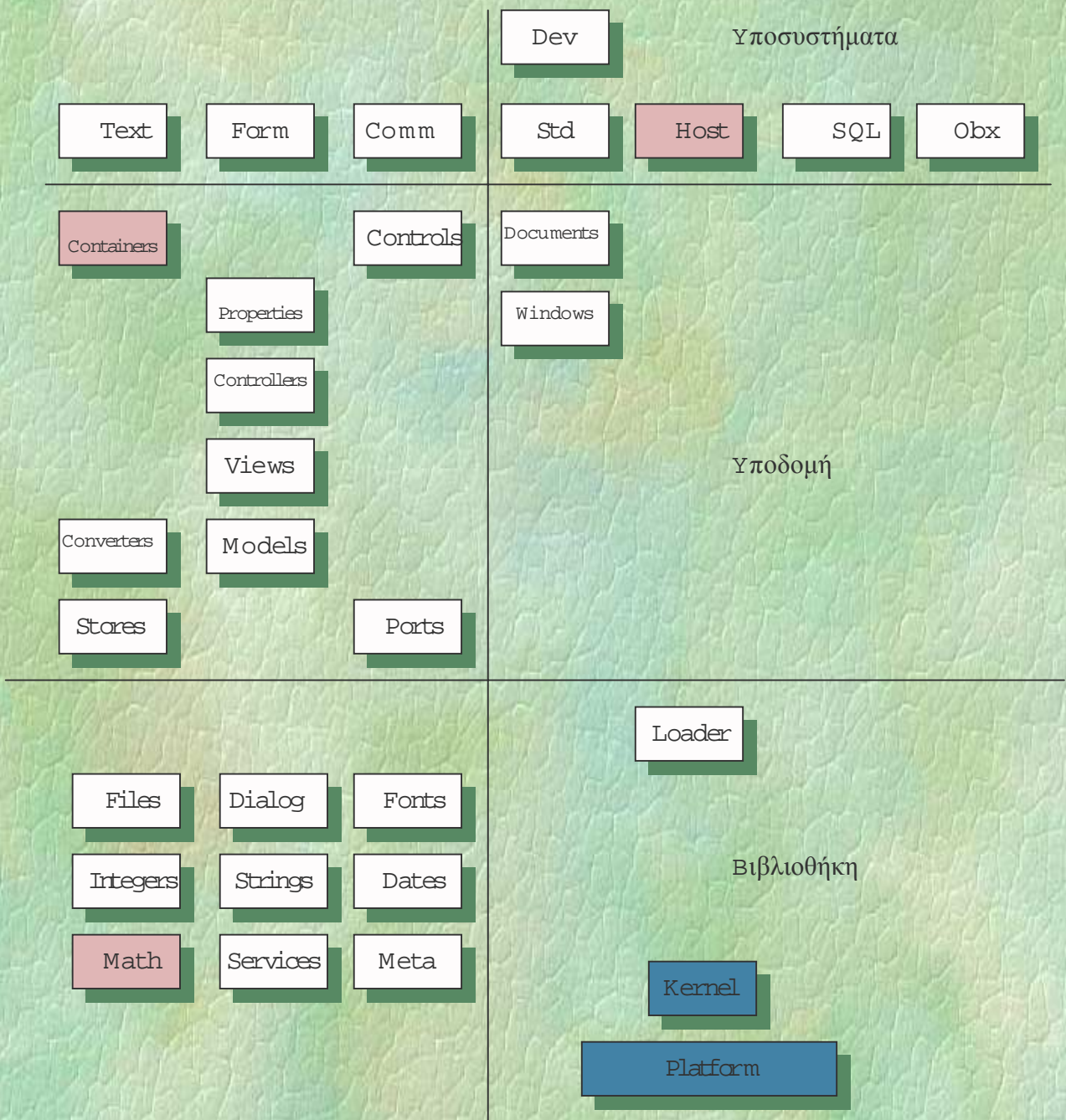
- Τα τρία επίπεδα γραφής της OpenDoc:
 - Απλή γραφή (οδηγεί μερη από έξω).
 - Επιδιορθωτής, που συνάπτει γραφές σε οδηγούς σημαντικών συμβάντων μερών. Αυτά τα μέρη πρέπει να συνάδουν με ορισμένους κανόνες που να επιτρέπουν την προσάρτηση των γραφών. Με την προσάρτηση μία τέτοια γραφή μπορεί να αλλάξει την συμπεριφορά αυτού του μέρους με την λήψη των αντίστοιχων σημαντικών συμβάντων.
 - Καταχωριτής, που απορροφά όλα τα εσωτερικά συμβάντα μέσω του εξωτερικού οδηγού συμβάντων. Έτσι καταχωρούνται ακολουθίες συμβάντων και ξαναπαίζονται αργότερα στο ίδιο ή σε παρόμοιο μέρος.

14.2.3 Προγραμματισμός Ενός Νέου Μέρους της OpenDoc

- Τα νέα OpenDoc μέρη πραγματώνονται με την εφαρμογή νέων εκδοτών μερών, που μπορεί να αντιλαμβάνονται ένα ή περισσότερα σχήματα δεδομένων.
- Υπάρχει μία εντελώς ιδεατή τάξη η ODPart διασύνδεση, που περιέχει 60 περίπου μεθόδους για εφαρμογή από κάθε εκδότη για την υποστήριξη των ιδιοτήτων των μερών όπως:
 - Επιμονή.
 - Αναπαράσταση και διαχείριση πλαισίων και εδρών.
 - Επίδραση με περιέχοντα και περιεχόμενα μέρη.
 - Σύνδεσμοι και οδηγοί συμβάντων.
 - Διασύνδεση χρήστη.
- Υπάρχουν άλλες 70 περιπλέον τάξεις που οι περιπτώσεις τους δημιουργούνται από την OpenDoc.
- Κάθε μέρος OpenDoc βασικά εφαρμόζει το ODPart. Αυτή η τάξη φροντίζει για όλα τα μέρη και επομένως υπάρχει μεγάλο επιπλέον κόστος για την χρήση των απλών μερών. Γι' αυτό δημιουργήθηκε η επιπλέον τάξη υποδομής OpenDoc Part Framework (OPF). Ορίζει μία ημιτελή αρχιτεκτονική αρκετά καλή για την πλειονότητα των μερών. Η χρησιμοποίησή της δεν αποκλείει την χρήση της ODPart που υπερισχύει.

14.3 Η Υποδομή Συστατικών BlackBox (BCF)

- Κατά κύριο λόγο το BlackBox είναι ένα εργαλείο ανάπτυξης συστατικών και περιβάλλον προγραμματισμού, που αναπτύχθηκε από την Oberon microsystems (1997). Μέρος αυτού είναι το BlackBox Component Framework, υποδομή που εστιάζει σε συστατικά για σύνθετα κείμενα με γραφικές διασυνδέσεις χρηστών.
- Στο σχήμα 14.1 δίδεται η περιγραφή της αρχιτεκτονικής του BlackBox. Η υποδομή έχει ένα πυρήνα από επίπεδα διαμορφωμάτων και ένα ανοικτό σύνολο υποσυστημάτων. Κάθε υποσύστημα είναι από μόνο του ένα σύνολο από επίπεδα διαμορφωμάτων. Ο κατασκευαστής συστατικών υπάρχει σαν επέκταση του BCF.
- Τα σκιαζόμενα μέρη του σχήματος 14.1 δηλώνουν εξάρτηση από πλατφόρμα. Τα ελαφρώς σκιαζόμενα περιέχουν φορητές διασυνδέσεις. Οι διαμορφωτές και υποσυστήματα στο αριστερό τήμα του σχήματος περιέχουν καθορισμένες διασυνδέσεις προγραμματισμού. Αυτά στα δεξιά είτε είναι προαιρετικά που δεν δίδονται από συστατικά, είτε υπάρχουν για συγκεκριμένες πλατφόρμες.
- Το BCF εστιάζεται σε εικονικά συστατικά – πλαίσια της Visual Basic, OLE, ActiveX controls, JavaBeans και Live Objects. Αυτά βασίζονται στη οπτική εμφάνιση και αλληλεπίδραση με τα περιέχοντα και περιεχόμενα συστατικά. Η κύρια ιδέα στη οποία η υποδομή κτίζει είναι επομένως οπτική. Οι όψεις αυτές είναι αυτο-υποστηριζόμενες, με δικό τους εκδότη. Περισσότερο πολύπλοκες όψεις υπονοούν διαχωρισμό των προτύπων, όψεων και ελεγκτών.
- Ειδικά χαρακτηριστικά αρχιτεκτονικής BlackBox:
 - Πρότυπο σχεδιασμού μεταφορείς-διάδρομοι-σηματοποιητές (διθητήρες στοιχείων).
 - Αντικείμενα καταλόγου.
 - Ιεραρχικό πρότυπο διαχωρισμού όψεως.
 - Τύποι περιέχοντων.
 - Υπηρεσία μετάδοσης μνημάτων.



Σχ. 14.1 Η αρχιτεκτονική του BCF.

14.3.1 Πρότυπο σχεδιασμού μεταφορείς-διάδρομοι-σηματοποιητές

- Οι μεταφορείς είναι αντικείμενα μεταφοράς στοιχείων. Όλες οι πιθανές εφαρμογές για διασύνδεση με ένα μεταφορέα αποτελούν την διάσταση επέκτασης του.
- Ο διάδρομος περιλαμβάνει δρόμο για πρόσβαση σε συγκεκριμένο τόπο. Είναι δημιουργήμα μεταφορέα από τον οποίο παίρνει προτεραιότητα.
- Στις διασυνδέσεις μεταφορέων – διαδρόμων σχηματίζουν σημεία συμφόρησης που προκαλούν αποδυσμευση των επεκτάσεων του μεταφορέα. Ο σηματοποιητής είναι διηθητήρας του σχήματος των στοιχείων. Κάθε αποδυσμευμένος σηματοποιητής έχει διάσταση επέκτασης που είναι ορθογώνιος προς αυτή του μεταφορέα. Δίδεται η ακόλουθη παράσταση σχεδιασμού της υποδομής BlackBox:
 - Σύστημα αρχείων.
 - Μεταφορέας `Store.Reader`, `Stores.Writer` εσωτερίκευση/εξωτερίκευση αντικειμένων.
 - Διάδρομος `Files.Rider` τυχαία πρόσβαση `byte read/write`.
 - Σηματοποιητής `Files.File` ιδεατά αρχεία.
 - Σύστημα προβολής.
 - Μεταφορέας `Ports.Frame` συγχρονισμός μεταφοράς.
 - Διάδρομος `Ports.Rider` περιοχή προβολής.
 - Σηματοποιητής `Ports.Port` ιδεατό `pixelmap` (γραφικά στίγματα).
 - Υποσύστημα κειμένου.
 - Μεταφορέας `TextMappers.Scanner`, `TextMapper.Formatter`
 - Διάδρομος `TextModels.Reader`, `TextModels.Writer`
 - Σηματοποιητής `TextModels.Model`
 - Υποσύστημα σχήματος.
 - Μεταφορέας (μη καθορισμένο σχήμα μεταφορέα).
 - Διάδρομος `FormModels.Reader`, `FormModels.Writer`
 - Σηματοποιητής `FormModels.Model`.

Πρότυπο σχεδιασμού μεταφορείς-διάδρομοι-σηματοποιητές

Το σχήμα 14.2 εξηγεί τις σχέσεις πελατών, σηματοποιητών, διαδρόμων και μεταφορέων.

- Το ακόλουθο παράδειγμα με Component Pascal συνδυάζει μεταφορέα και σηματοποιητή.

```
VAR f:Files.File; w:Stores.Writer; pos:INTEGER;
```

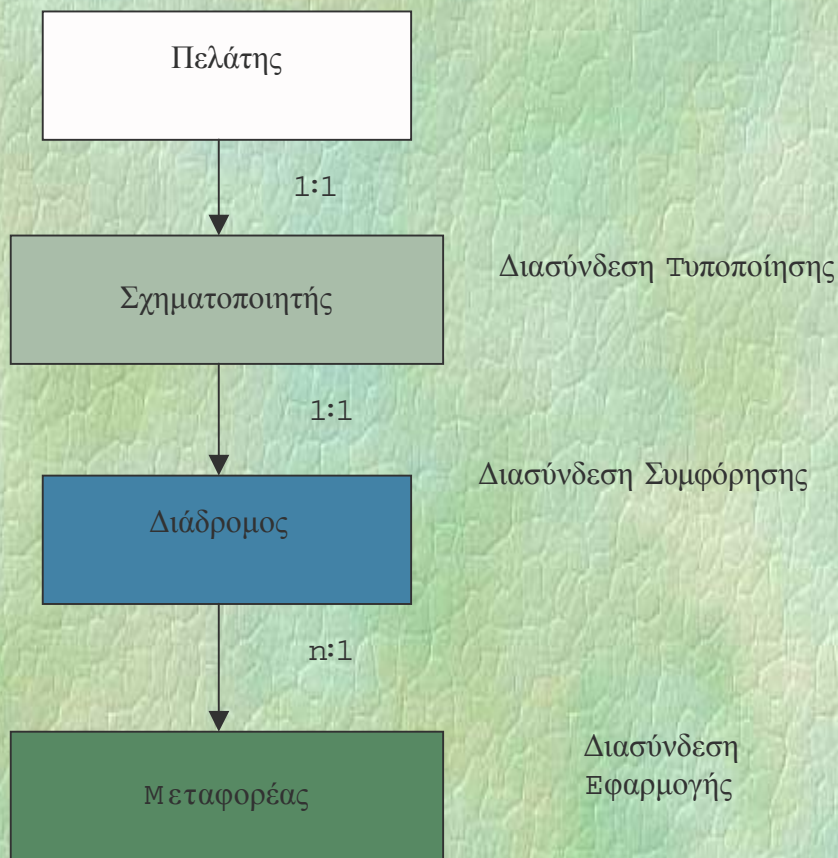
```
...
```

```
f:=...;
```

```
w.ConnectTo(f); w.SetPos(pos)
```

Η μέθοδος `ConnectTo` καλεί νέο διάδρομο από το αρχείο και το συνάπτει στο `Writer`.

Μετά ο σηματοποιητής μπορεί να προχωρήσει σε μία θέση (`pos`) στο αρχείο. Ακολούθως ο `writer` μπορεί να χειριστεί κλήσεις για εξαγωγή αντικειμένων γράφοντας μία γραμμική ακολουθία από `bytes` στο αρχείο.



Σχ 14.2 Διαχωρισμός μεταφορέα, διάδρομου και σηματοποιητή.

14.3.2 Αντικείμενα Καταλόγου

- Στη BCF γίνεται χρησιμοποίηση κατασκευαστών αντικειμένων και κατασκευαστών μεθόδων, όπου χρειάζονται. Αυτά εξασφαλίζουν τρέχουσες εφαρμογές σε συγκεκριμένο πλαίσιο και ονομάζονται *αντικείμενα καταλόγου* (η αναγκαιότητα για νέα αντικείμενα ωφείλεται στην ιδιότητα της BCF να μην δημοσιοποιεί ονόματα εφαρμογών ιδεατών διασυνδέσεων αποκλείοντας συναρτήσεις που χρειάζονται ονόματα τάξης).
- Κάθε διαμορφωτής με νέα λειτουργικότητα παρέχει επίσης ρυθμιζόμενο αντικείμενο καταλόγου του οποίου ο εντοπισμός είναι ιδεατός. Απ' αυτό μπορεί να ζητηθεί να ανοίξει υπάρχον αρχείο με το όνομα του, ή να δημιουργήσει ένα νέο που αρχικά δεν έχει όνομα.
- Η πρόσβαση στο σύστημα αρχείων γίνεται μέσω της μεταβλητής `File.dir`.
- Για περιπτώσεις αντικειμένων ίδιας ποιότητας (όπως δομημένα στοιχεία και ο προσωρινός καταχωρητής τους σαν περιπτώσεις της ίδιας εφαρμογής), τότε έχουμε κλωνοποίηση με κλώνο άδειο αντίγραφο. Στο παρακάτω παράδειγμα η μεταβλητή `t1` δημιουργείται σαν κλώνος του `t` (που προσθέτη νέες παραμέτρους σε μία καθορισμένη διασύνδεση κειμένου).

```

VAR t,t1:TextModels.Model;
...
t:=...;
t1:=TextModels.Clone(t);
t1.InsertCopy(t,0,42); (*avoid loss of attributions*)
...(*change t1-for example delete all lowercase
characters*)
t.Replace(0,42,t1,0,t1.Length()) (atomically replace with
update)

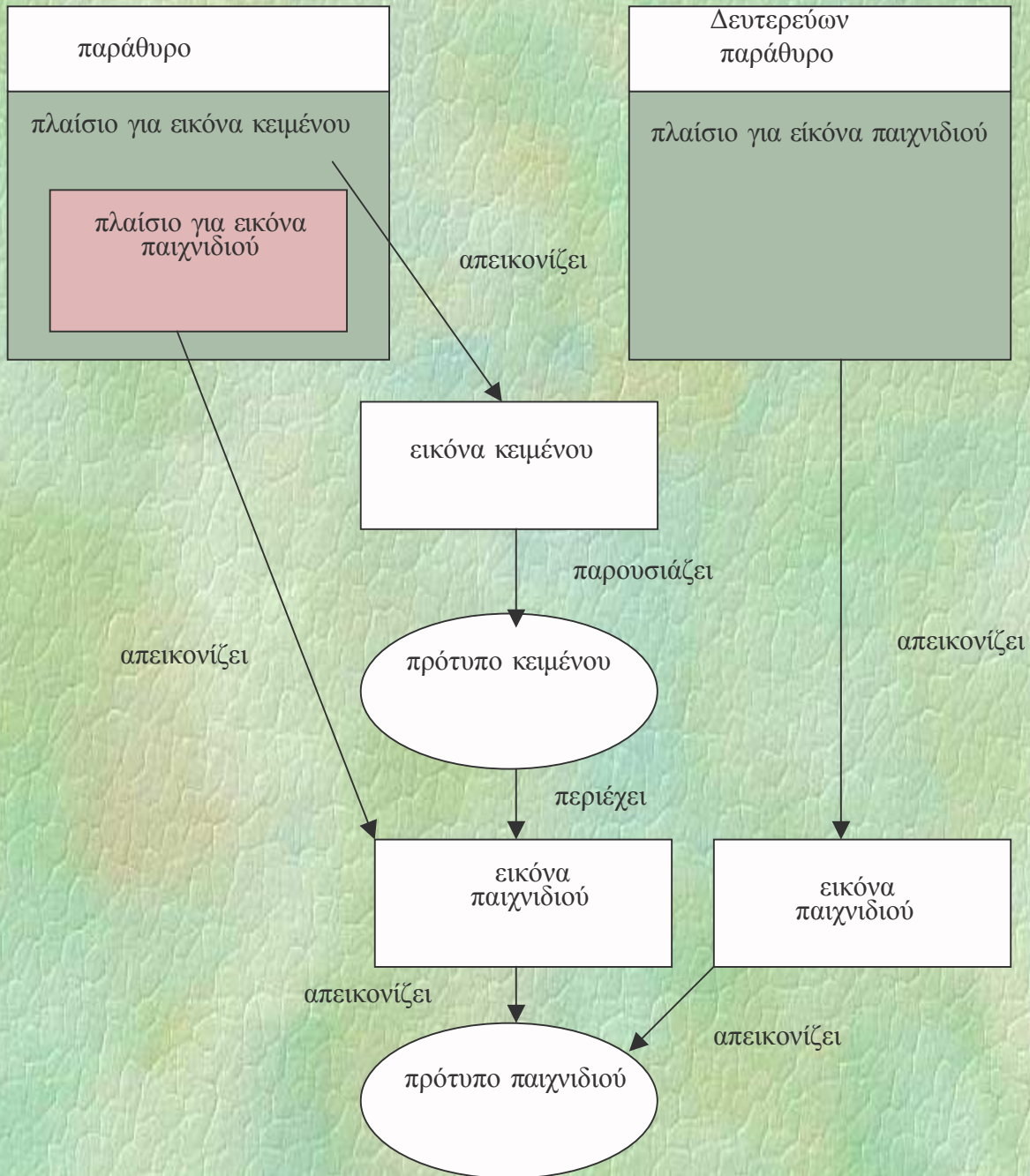
```

14.3.3 Ιεραρχικό Πρότυπο Εικόνων

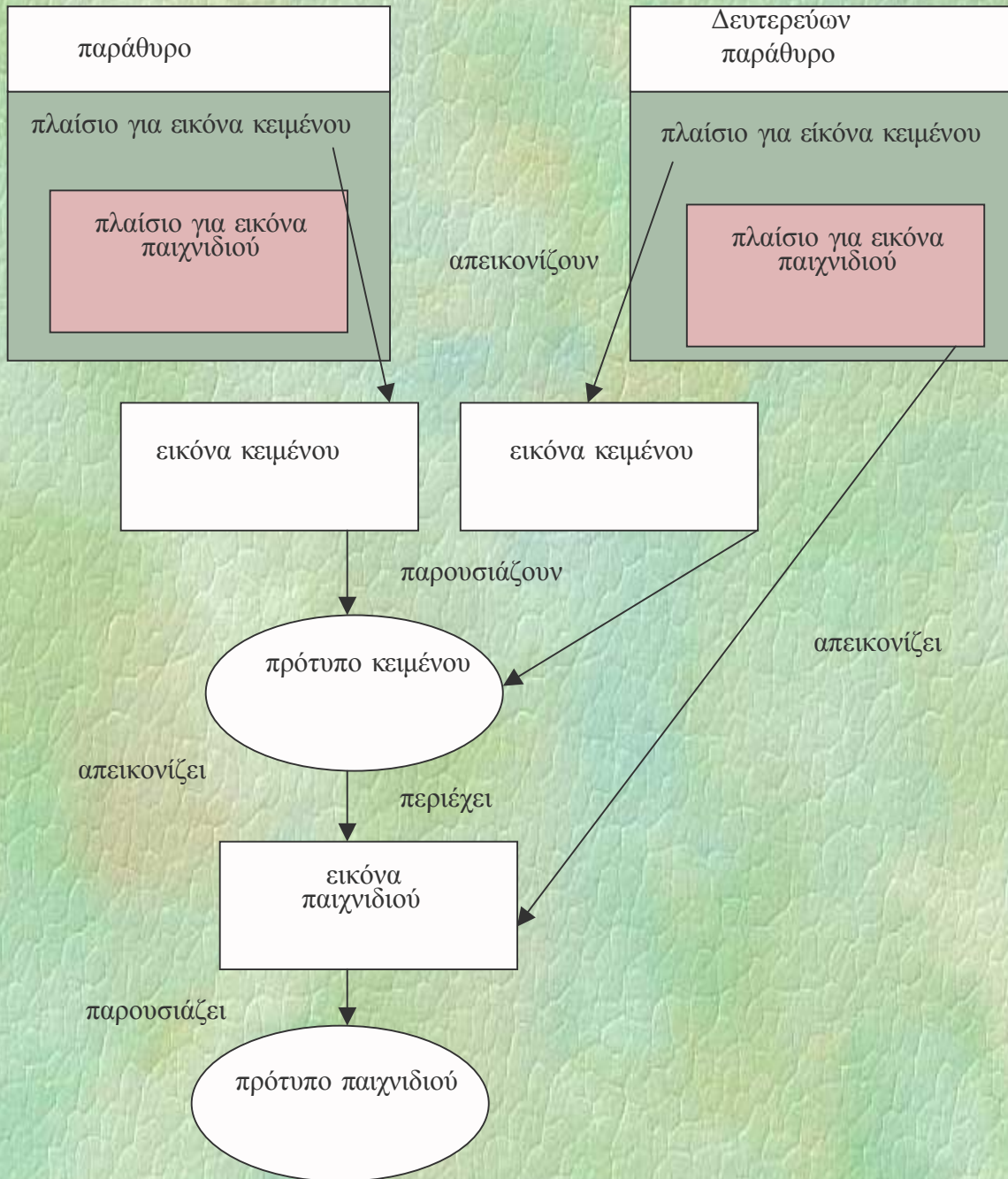
- Στη υποδομή BlackBox μία ιεραρχική έκδοση του προτύπου ελεγκτή εικόνων είναι το Hierarchical Model View Controller (HMVC). Αυτό περιλαμβάνει
 - πολύ ελαφρά συστατικά απεικόνισης (έτσι θα μπορούσαμε να έχουμε απλή εφαρμογή της μεθόδου Restore) καθώς και
 - Βαρείά συστατικά υποδοχέων.
- Οι απεικονίσεις στη BlackBox παρέχουν οπτική παρουσίαση στοιχείων και μπορεί να είναι εξαρτώμενες του πλαισίου και ικανές αλληλεπιδράσεων. Μπορούν να ενσωματωθούν επαναληπτικά, με μερικές να ενεργούν και σαν υποδοχείς. Ένα σύνθετο κείμενο BlackBox αποτελεί μία εξωτερική απεικόνιση. Εκτός της οπτικής πληροφορίας μία απεικόνιση μπορεί να καλύπτει άλλα αυθαίρετα αντικείμενα.
- Οι απεικονίσεις μπορούν να παρουσιάζουν και να διευθύνουν στοιχεία με διαφορετικό τρόπο και προοπτική, μέσω διαφορετικών προτύπων. Ο τύπος ρύθμισης των απεικονίσεων μπορεί επίσης να είναι ξεχωριστός με ελεγκτές της επιλογής του χρήστη.
- Οι ρυθμίσεις που γίνονται στις απεικονίσεις, τα πρότυπα στοιχείων και τους ελεγκτές πρέπει να φυλάγονται στο περιέχον κείμενο, για δύο κυρίως λόγους:
 - Τα ίδια στοιχεία μπορεί να δίδονται με διαφορετικούς τρόπους ταυτόχρονα.
 - Μία απεικόνιση μπορεί να έχει και σταθερές και ρυθμιζόμενες κατά την εκτέλεση ιδιότητες.

Οι ελεγκτές θα πρέπει να μπορούν να φυλάγουν το τύπο υποδοχέα στον οποίο ανήκουν.

- Το HMVC υπονοεί ιεραρχικά εμπλεκόμενες απεικονίσεις (κανονικές και ενσωματωμένες). Μεταξύ όλων των απεικονίσεων πρέπει φυσικά να υπάρχει συμβατότητα.
- Το σχήμα 14.3 δείχνει την υποστήριξη σε ένα πρότυπο όπου η απεικόνιση ενός παιχνιδιού είναι ενσωματωμένη σε ένα κείμενο στο βάθος, ενώ μπροστά βρίσκεται η κανονική απεικόνιση του.
- Στον ιεραρχικό έλεγχο η BlackBox υπάρχει η ριζική απεικόνιση – συνήθως κείμενο, με δευτερεύοντα παράθυρα να παρουσιάζουν τμήματα του ή και ενσωματώσεις. Οι λειτουργίες των δευτερευόντων παραθύρων είναι ανεξάρτητες. Κάθε ριζικό παράθυρο όμως επιρραάζει τα δευτερεύοντα του.
- Κάθε απεικόνιση αντιπροσωπεύεται από ένα αντικείμενο πλαισίου (frame). Τα πλαίσια συνδέουν σε ιδεατές θόνες-θύρες επικοινωνίας (ports). Στο σχήμα 14.4 εξηγείται πώς το δευτερεύον παράθυρο παρουσιάζει το αρχικό κείμενο.



Σχ. 14.3 υποστήριξη σε ένα πρότυπο όπου η απεικόνιση ενός παιχνιδιου είναι ενσωματωμένη σε ένα κείμενο στο βάθος, ενώ μπροστά βρίσκεται η κανονική απεικόνισή του.



Σχ. 14.4 Πληροφορίες εικόνες και πλαίσια με δύο παράθυρα στο ίδιο κείμενο.

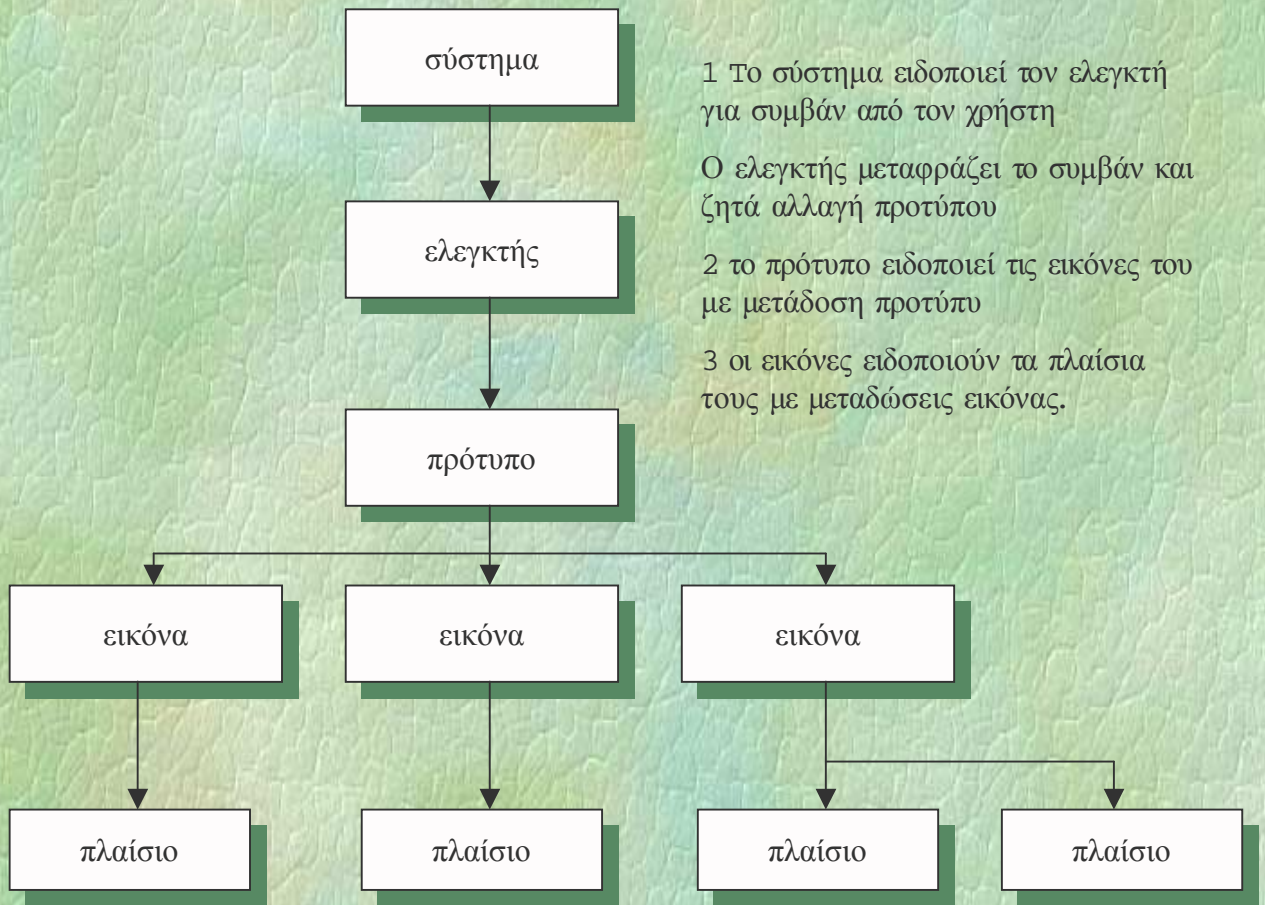
14.3.4 Υποδοχές πληροφοριών

- Στη παροχή - υποδοχή πληροφοριών υπάρχουν δύο στάδια.
 - Κατασκευή περίπτωσης συστατικού, όπως σχεδιασμός νέου εντύπου οθόνης.
 - Χρήση προκατασκευασμένης περίπτωσης συστατικού, όπως η συμπλήρωση ενός εντύπου οθόνης.
 - Τα πύ πάνω ισχύουν για αντικειμενοστρεφείς γλώσσες. Στη περίπτωση όμως συστημάτων σύνθετων κειμένων τα δύο στάδια ενοποιούνται. Για παράδειγμα τα εξωτερικά επίπεδα ενός σύνθετου κειμένου μπορούν να παγώσουν, ενώ σε εσωτερικά επίπεδα μπορούν να δημιουργηθούν νέες περιπτώσεις. Το μειονέκτημα τέτοιας υποδομής είναι ότι μπορεί να προκληθεί ζημιά στη διάρκεια της συμπλήρωσης λ.χ. μίας περίπτωσης.
 - Το πλαίσιο της BlackBox, container mode (τύπος υποδοχής), εξασφαλίζει πλήρη έλεγχο στα ενοπιημένα στάδια περιπτώσεων συστατικών. Αυτό περιλαμβάνει:
 - Edit mode. Τα περιεχόμενα του υποδοχέα συλλέγονται, εκδίδονται και εστιάζονται (ενεργοποιούνται) - επιλογή εκδότη κειμένου με ενσωματωμένα περιεχόμενα.
 - Layout mode. Τα περιεχόμενα του υποδοχέα συλλέγονται, εκδίδονται, αλλά δεν εστιάζονται - επιλογή οπτικού σχεδιαστή ή κατασκευαστή συστατικών.
 - Browser mode. Τα περιεχόμενα του υποδοχέα συλλέγονται, εστιάζονται αλλά δεν εκδίδονται (όπως τα υπερκείμενα της HTML) - επιλογή για κειμενογράφηση σε ανοικτή γραμμή.
 - Mask mode. Τα περιεχόμενα του υποδοχέα μόνο εστιάζονται - επιλογή για προκατασκευασμένο έντυπο.
 - Palette mode. Τα περιεχόμενα του υποδοχέα συλλέγονται μόνο.
 - Frozen mode. Καμμιά λειτουργία δεν επιτρέπεται στα περιεχόμενα.
- επίσης είναι πιθανοί οι συνδυασμοί.
- Μόνο έκδοσης περιεχομένων.
 - Έκδοσης και ενεργοποίησης.
- Όλοι οι πύ πάνω συνδυασμοί εναπόκεινται στον χρήστη.

14.3.5 Διαδοχικά Μηνήματα σε μη εγγεγραμμένους αποδέκτες

- Αποδέσμευση πηγών συμβάντων σε αντικείμενα μηνυμάτων.
- Εσωτερικές συνδέσεις των περιπτώσεων συστατικών.
- Η επικοινωνία γίνεται με πολλαπλούς μη εγγεγραμμένους αποδέκτες.
- Η ακολουθία αιτιότητας (με την σειρά που προκλήθηκαν) σ' αυτή την επικοινωνία είναι σημαντική.
- Μία βαρυφορτωμένη επίλυση είναι το πέρασμα των μηνυμάτων σε σειρά αναμονής και η παροχή των νημάτων τους στους δέκτες. Μετά τα μηνήματα στέλνονται σε αιτιατή σειρά και τα νήματα αφήνουν την εκτέλεση τους.
- Σε σύνθετο κείμενο αυτό σημαίνει ξεχωριστό νήμα για κάθε ενσωματωμένη περίπτωση.
 - Το πρότυπο ειδοποιεί τις εικόνες του με μετάδοση.
 - Κάθε ενδιαφερόμενη εικόνα τότε και πάλι μεταδίδει το μήνυμα σε κάθε ενδιαφερόμενο πλαίσιο που προβάλλεται σε κάποια οθόνη.
- Κανένα μήνυμα προτύπου δεν στέλλεται πριν τελειώσει την μετάδοση του το προηγούμενο.
- Το αντίστοιχο ισχύει και για τα μηνήματα εικόνων.
- Τα μηνήματα κατέχουν την διεύθυνση της πηγής τους, έτσι το σύστημα ελαττώνει την διανομή μηνυμάτων σε πραγματικά ενδιαφερόμενους δέκτες.
- Οι μεταδώσεις σε περιοχή, όπου οι δέκτες ανήκουν όλοι σε ένα κείμενο γίνονται όταν αυτοί δεν έχουν στοιχεία διευθύνσεων ώστε να αποφασίζουν για τον αποστολέα.
- Η μετάδοση μηνυμάτων εικόνας προς όλους γίνεται σε όλα τα ανοικτά κείμενα.

Διαδοχικά Μηνήματα σε μη εγγεγραμμένους αποδέκτες



Σχ. 14.5 τρία στάδια διαδοχικών μηνυμάτων στην BlackBox.

14.3.6 Προχωρημένες Εφαρμογές Βασιζόμενες σε Σύνθετα Κείμενα

- Οι καθορισμένοι υποδοχείς κειμένου επιτρέπουν τις διαφοροποιήσεις στη διασύνδεση του χρήστη.
- Τα συστατικά έχουν διασυνδέσεις προγραμματισμού.

14.4 Σύγκριση BlackBox με OpenDoc

- Κοινή αντίληψη γενικών υποδοχών και επαναλαμβανόμενης ενσωμάτωσης .
- Κοινή χρήση σύνθετων κειμένων σε όλα.
- Αντίστοιχοι όροι στο πρότυπο σύνθετου κειμένου.
- Λιγότερο περίπλοκο και λιγότερο ισχυρό είναι το BlackBox, με υποστήριξη περιορισμένου προτύπου κειμένου.
- Καθόλου υποστήριξη για σύνδεση αντικειμένων από την BlackBox, που όμως υποστηρίζει υπερκείμενα .
- Η BlackBox έχει ένα ομοιόμορφο σύστημα ανάπτυξης με Component Pascal που είναι μικρή, απλή και αποτελεσματική. Έχει ψηλού και χαμηλού επιπέδου εντολές και ο προγραμματιστής χρειάζεται να ξέρει μόνο ότι θα χρησιμοποιήσει.
- Η OpenDoc χρησιμοποιεί Java και JavaScript για γράψιμο. Η Java δεν είναι καλή για προγραμματισμό συστήματος.
- Η OpenDoc επίσης υποστηρίζει πλατφόρμες COM και Windows. Σύνδεση της COM με Component Pascal για προγραμματισμό συστατικών.
- Η BlackBox σε COM αποτελεί και εξυπηρετητή και υποδοχέα OLE του οποίου οι διασυνδέσεις χρησιμοποιούνται σε Component Pascal ή σε άλλη γλώσσα.

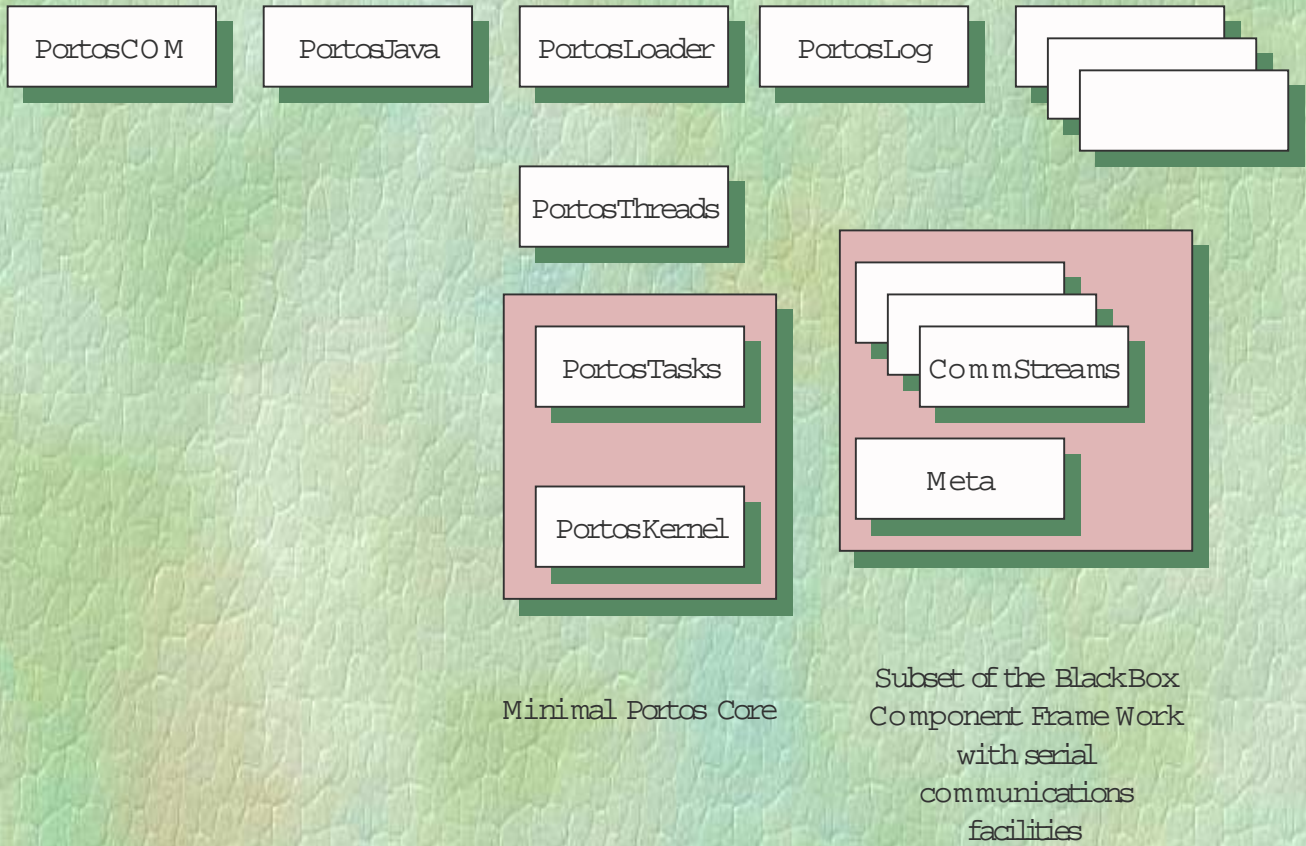
14.5 Portos και Denia - Υποδομή Συστατικών Πραγματικού Χρόνου και το Περιβάλλον Αναπτύξεως τους

- Παρέχει πλήρεις διασυνδέσεις COM.
- Οδηγοί συσκευών με κανόνες OPC είναι συμβατοί με αυτό το λειτουργικό.
- Εκπληρωμένο σε Component Pascal, με ιδιότητες συλλογής αχρήστων, και τοπικές διασυνδέσεις Component Pascal. Επίσης τοπική υποστήριξη Java.
- Εργαλείο ανάπτυξης Denia που βασίζεται σε υποδομή BlackBox.
- Πλήρης συμβατότητα με διασυνδέσεις OLE.

14.5.1 Η Δομή του Portos

- Ο πυρήνας του Portos είναι μικρός. Σαν λειτουργικό μπορεί να υπάρχει μόνο του η να φιλοξενείται από άλλο.
- Η δομή δίδεται στο σχέδιο 14.6, όπου τα κουτιά με χονδρές γραμμές αντιπροσωπεύουν μονάδες για χρήση από όλους, ενώ αυτά με λεπτές γραμμές αντιπροσωπεύουν μονάδες με εσωτερικές διασυνδέσεις ή που εφαρμόζουν διασυνδέσεις άλλων μονάδων.
- Οι βασικότερες μονάδες PortosKernel και PortosTasks (υποστήριξη εφαρμογών πραγματικού χρόνου).
- Προαιρετικά μπορεί να προστεθεί η PortosThread που διευθύνει νήματα προτεραιότητας. Το χρονοδιάγραμμα τους μπορεί να συνδεθεί με PortosTasks.
- Επίσης μονάδες από την BlackBox μπορούν να χρησιμοποιηθούν στο Portos:
 - Meta (υπηρεσίες αντανάκλασης).
 - CommStreams (σειριακή επικοινωνία).
- Το PortLog εξάγει πληροφορίες σε μηχανές που τρέχουν Denia ή τερματικά προγράμματα. Διαταγές στο Denia μπορούν να εκτελούνται στο Portos. Έτσι νέες μονάδες μπορούν να εγκαθίστανται στο Portos.
- Το PortosLoader φορτώνει νέες μονάδες (αρχεία κώδικα) κατά την εκτέλεση. Μπορεί να συνδεθεί με Portoskernel.
- Το PortosCOM εφαρμόζει βασικές υπηρεσίες για αντικείμενα COM. Λειτουργεί σαν φορτωτής COM DLL με παρόμοιο τρόπο που το PortosLoader χειρίζεται αρχεία κώδικα της Component Pascal.
- Το PortosJava εφαρμόζει τις τάξεις που ορίζονται με Java. Με αυτό χρησιμοποιούνται το PortosLoader και PortosKernel. Τα αρχεία κώδικα Java μετατρέπονται σε Component Pascal πριν να μεταφερθούν στο Portos.

Various servers
for remote
control
debugging, etc



Σχ. 14.6 Η δομή του Portos.

14.5.2 Προγραμματιστής Πραγματικού Χρόνου

- Η κεντρική μονάδα για προγραμματισμό πραγματικού χρόνου είναι το `PortoTask`:
- Σαν εργαλείο συγχρονισμού χρησιμοποιείται ο συνδυασμός παρακολουθητή και σήματος. Ο τύπος `Synchronizer` αντιστοιχεί σε Java Objects.
- Κάθε `task` αντιπροσωπεύει μία ταυτόχρονη διεργασία που εφαρμόζεται από το αντικείμενο `PortoTasks.Handler`. Τα αντικείμενα `Handler` περιέχουν μία ιδεατή μέθοδο, `RUN` που εφαρμόζει τον κώδικα της διεργασίας.
- Μετά την δημιουργία τους τα `Tasks` είναι ανενεργά. Οι λειτουργίες τους είναι ξεκίνημα, σταμάτημα, αναβολή και νάρκωση για κάποιο διάστημα. Κάθε `task` πρέπει να τερματίζει σε προθεσμία χρόνου ή πριν καταναλώσει ορισμένο υπολογιστικό χρόνο. Αυτοί οι περιορισμοί είναι αυστηροί και χρησιμοποιούνται και στην εφαρμογή `PortoThreads`.

DEFINITION `PortoTasks`;

TYPE

(**monitor/signal synchronization primitive **)

Synchronizer=POINTER TO

EXTENSIBLE RECORD

(*s:Synchronizer*) **TryEnter** (OUT done BOOLEAN),NEW; (**non-blocking enter**)

(*s:Synchronizer*) **Enter**,NEW; (**lock**)

(*s:Synchronizer*) **Exit**,NEW; (**unlock**)

(*s:Synchronizer*) **Notify**,NEW; (**wake up a waiting task**)

(*s:Synchronizer*) **NotifyAll**,NEW; (**wake up all waiting tasks**)

(*s:Synchronizer*)

Awaited():BOOLEAN,NEW; (**a task for waiting for notification**)

(*s:Synchronizer*) **Wait**

(*usec:INTEGER*),NEW; (**await notify with timeout**)

END;

(**client process controlled by a task**)

Handler=POINTER TO ABSRTACT

RECORD

(*h:Handler*)**Run**-,NEW,ABSTRACT; (**implements the task's behaviour**)

(*h:Handler*)**HandleException**-,NEW,EMPTY (**optional exception hadler**)

END;

Προγραμματιστής Πραγματικού Χρόνου

```

(*tasks control the concurrent
execution of handlers*)
Task=POINTER TO A LIMITED RECORD
    (t:Task) Start,NEW; (*make task ready for
scheduling*)
    (t:Task) Stop,NEW; (*make task inactive*)
    (t:Task) Suspend,NEW; (*wait for resumption*)
    (t:Task) Resume,NEW (*resume after suspension*)
END;

VAR current -: Task; (*the currently executing task*)
    (factory functions for normal,periodic, and sporadic
    (interrupt) tasks*)
PROCEDURE
    NewTask(handler:Handler;duration,deadline:INTEGER):Task;
PROCEDURE NewPeriodicTask(handler:Handler;
    duration,deadline:INTEGER; period:INTEGER):Task;
PROCEDURE NewSychedTask(handler:Handler;
    duration,deadline:INTEGER; factor:INTEGER;
    periodicTask:Task):Task;
PROCEDURE
NewSporadicTask (handler:Handler;
duration,deadline:INTEGER;
minPeriod:INTEGER;
eventType:INTEGER):Task;
PROCEDURE Time():INTEGER;
PROCEDURE Sleep(ticks:INTEGER; c:Clock);
END PortosTasks.

```

14.5.3 Περιβάλλον Ανάπτυξης Denia

- Το Denia αποτελεί επέκταση του κατασκευαστή συστατικών της BlackBox όπως έχει περιγραφεί πιο πάνω. Προσθέτει λειτουργικότητα για την μεταφορά αρχείων κώδικα και πληροφοριών προς και από το ενσωματωμένο σύστημα που τρέχει το Portos, ένα μεταγλωττιστή Component Pascal που παράγει κώδικα PowerPC, συμβολική υποστήριξη διόρθωσης λαθών προγράμματος, μεταφραστή για απόμακρες διαταγές, και απόμακρο (οπτικό) έλεγχο. Denia με BlackBox παρέχεται στα Windows.
- Σ' αυτό υπάρχουν μία ή περισσότερες συνδέσεις ανοικτές προς τον επεξεργαστή που τρέχει Portos είτε μέσω σειριακών γραμμών V24 είτε με ταυτόχρονες συνδέσεις TCP/IP, όπως μία σύνδεση για κάθε επεξεργαστή σε σύστημα πολλαπλών επεξεργαστών.
- Το υποσύστημα εντύπων της BlackBox μπορεί να δημιουργήσει κουτιά διαλόγου σαν σιασυνδέσεις χρήστη σε απόμακρες εφαρμογές Portos.