

Bringing the web offline...

OFFLINE WEB APPLICATIONS




Μέρος πρώτο: Εισαγωγή

Η εξέλιξη του web

- Ήταν:
 - Περιεχόμενο (HTML)
- Προστέθηκαν:
 - Εμφάνιση (XHTML, CSS)
 - Λειτουργικότητα (CGI scripts)
 - Διαδραστικότητα (JavaScript)



Η εξέλιξη του web

- Το web είναι πλέον πλατφόρμα
 - Εφαρμογές αντί σελίδες
 - Interactive
 - Read / write
 - Social
- 

Ένα επόμενο βήμα

- Το disconnected / **offline web**:

Πρόσβαση ακόμα και χωρίς σύνδεση σε:

- Περιεχόμενο
- Λειτουργικότητα


- Ενοποίηση των web εφαρμογών με τις τοπικές εφαρμογές

Γιατί Offline?

- Μα καλά, web εφαρμογή χωρίς internet;
- Μπορεί να μην είμαστε «online» αλλά δεν θα ήταν ωραίο...
 - Να βλέπαμε εκείνο το σημαντικό email από το webmail όσο η σύνδεση έχει «πέσει»;
 - Να γράψουμε ένα άρθρο για το blog μας όσο είμαστε στο (ΚΤΕΛ, τρένο, αεροπλάνο, πλοίο,...);
 - ...βάλτε την δικιά σας ιδέα εδώ!



Τα προβλήματα

- Τοπική αποθήκευση δεδομένων
 - Περιορισμοί των υπάρχοντων προτύπων
 - JavaScript
 - Έλλειψη τεχνογνωσίας
- 



Μέρος Δεύτερο: Τα συστατικά

Απαραίτητα συστατικά για το μαγείρεμα!

1. Τοπική και μόνιμη αποθήκευση δεδομένων (**persistence**)
2. Γνώση συνδεσιμότητας (**connectivity awareness**)
3. Διαφοροποίηση εφαρμογής (**modality**) ²
4. Συγχρονισμός (**synchronization**)

Απαραίτητα συστατικά για το μαγείρεμα! (Bonus)

✦ Ασφάλεια (*security*)

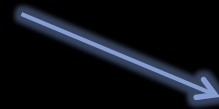
✦ Προστασία προσωπικών δεδομένων (*privacy*)

1^ο Συστατικό: Αποθήκευση δεδομένων

- ✓ Τα αρχεία της εφαρμογής
 - (X)HTML
 - JavaScript
 - CSS
 - Εικόνες
 - ...
- ✓ Τα δεδομένα της εφαρμογής

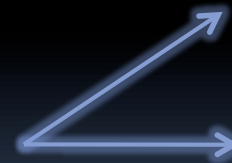
Αποθήκευση δεδομένων: Πώς;

Σήμερα:



- Application Cache
 - LocalServer
- Database (SQLite)

Στο άμεσο μέλλον:
HTML 5



- Name / Value Pair

* Συν κάποιοι άλλοι **έμμεσοι** τρόποι (Flash, browsers' special features)

Υποστήριξη HTML 5

	Database	localStorage	sessionStorage	globalStorage	Application Caches	browserState
Gecko (Firefox, Camino)	x	x	1.8.1 [MOZ-3]	1.8.1 [MOZ-3]	1.9 [MOZ-4]	1.9 [MOZ-5]
WebKit (Safari, Google Chrome, OmniWeb)	3.1 [EIDSON]	x	x	x	x	ημιτελής υποστήριξη
Trident (Explorer)	x	x	8 [MCSFT-2], [MCSFT-3]	8 [MCSFT-2], [MCSFT-3]	x	ημιτελής υποστήριξη
Presto (Opera)	x	x	x	x	x	ημιτελής υποστήριξη

2^ο Συστατικό: Γνώση συνδεσιμότητας

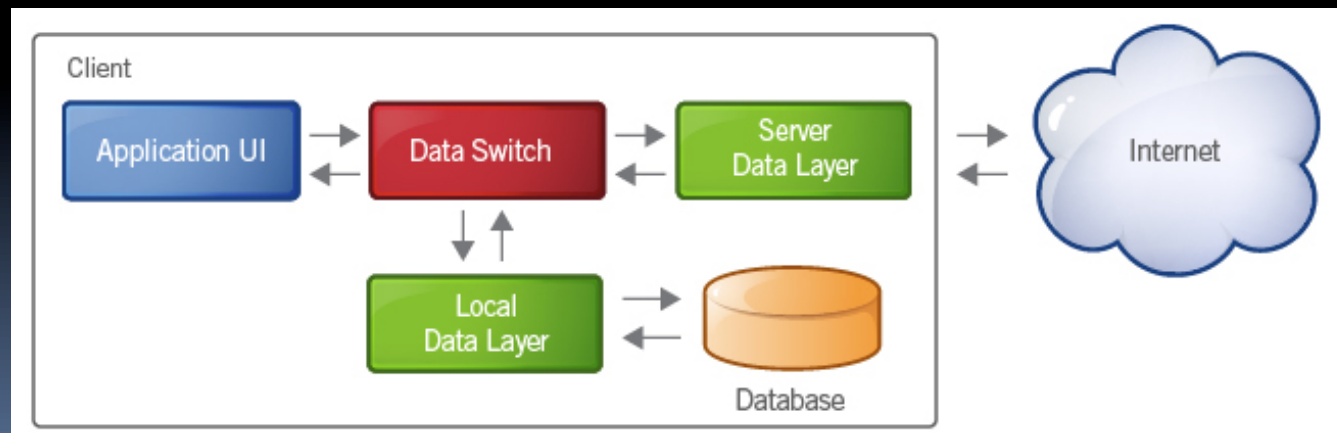
- Πώς ξέρουμε αν έχουμε σύνδεση;
 1. Χειροκίνητα (δήλωση χρήστη)
 2. Dojo Offline
 3. Ή υλοποιούμε κάτι δικό μας


- Καθορισμός ενεργειών όταν:
 1. Η σύνδεση χαθεί
 2. Η σύνδεση επανέλθει
 3. ...

(event-driven μοντέλο)


3^ο Συστατικό: Διαφοροποίηση εφαρμογής

- Εσωτερικά (αρχιτεκτονική)
 - Αλλάζει ο τρόπος λειτουργίας της εφαρμογής ανάλογα με την κατάσταση λειτουργίας;





3^ο Συστατικό: Διαφοροποίηση εφαρμογής

- Εξωτερικά (ως προς την λειτουργικότητα):
 - Όλες οι λειτουργίες λειτουργούν και offline;
 - Υπάρχουν όλα τα δεδομένα τοπικά;
- 

4^ο Συστατικό: Συγχρονισμός

- Χειροκίνητα
 - Ο χρήστης αποφασίζει πότε
 - Τα παλιά δεδομένα πετιούνται
 - Μένουν τα καινούρια
- Αυτόματα:
 1. Χρονολόγηση
 2. Πολιτικές συγχρονισμού δεδομένων
 3. Επικοινωνία client-server
- Υβριδικά

5^ο Συστατικό: Ασφάλεια

★ Τα δεδομένα αποθηκεύονται τοπικά

- Κίνδυνοι:

- Κακόβουλες εφαρμογές
- Κοινόχρηστοι λογαριασμοί
- Φυσική πρόσβαση στον δίσκο

- Λύσεις:

- Άδεια του χρήστη για τοπική αποθήκευση
- Same origin policy
- Encryption



Μέρος Τρίτο: Δουλεύει;

Η δικιά μας demo εφαρμογή

- Εφαρμογή ανταλλαγής μηνυμάτων:
 - Απλοϊκή

Αλλά:

- Εμπεριέχει πολλά από τα πραγματικά προβλήματα:
 - Αυτόματος συγχρονισμός
 - Αυτόματος έλεγχος διαθεσιμότητας δικτύου
 - Προσαρμογή του UI κατάλληλα

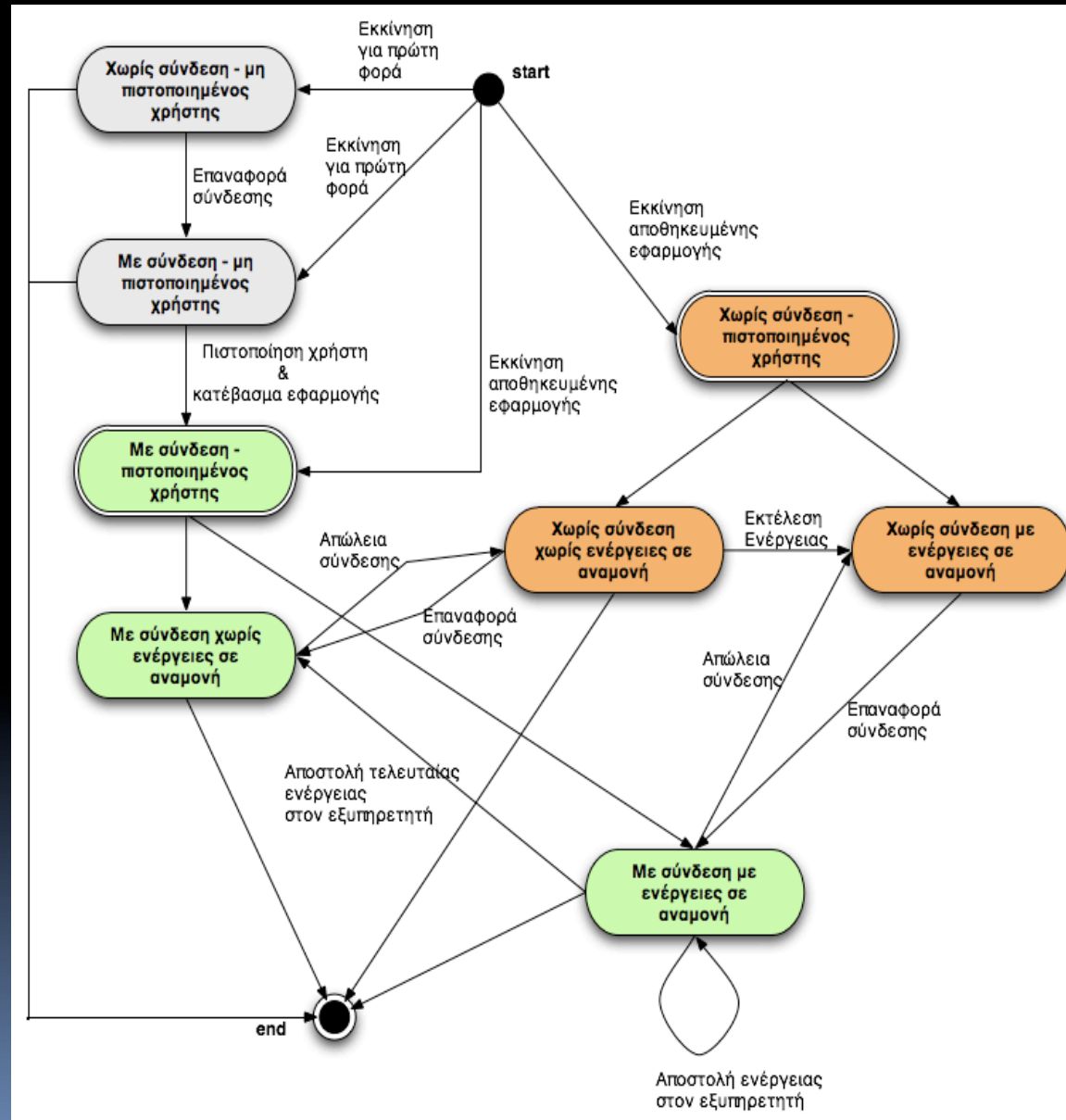
Η δικιά μας demo εφαρμογή

- Δημιουργία νέου μηνύματος
- Ανάγνωση και επεξεργασία μηνυμάτων που είναι «πρόχειρα» (δεν έχουν ακόμα αποσταλεί)
- Αποστολή μηνύματος σε άλλον χρήστη
- Ανάγνωση εισερχόμενων μηνυμάτων
- Διαγραφή εισερχόμενων / εξερχόμενων / πρόχειρων μηνυμάτων

Η δικιά μας demo εφαρμογή

- Gears
 - μόνιμη τοπική αποθήκευση δεδομένων και αρχείων
- Dojo Toolkit
 - έλεγχος συνδεσιμότητας
 - AJAX
 - DOM manipulation (UI)
 - συγχρονισμός δεδομένων (υποστηρικτικά μόνο)
- Ruby On Rails
 - server-side framework

Σενάριο λειτουργίας



Συμπεράσματα (1)

- Σε σχέση με τις σημερινές εφαρμογές:
 - Πιο «δύσκολο»
 - Πιο πολύπλοκο

Αλλά:

Απολύτως εφικτό!

Εφαρμογές που είναι «στον αέρα»

- Google Reader
- Google Docs
- RememberTheMilk
- Buxfer
- Passpack
- Zoho Writer



Μέρος Τέταρτο: Έχει νόημα;

Που θα ήταν χρήσιμο

- Ανάγνωση περιεχομένου:
 - Webmail
 - Analytics
 - Ημερολόγια, to-do's, εφαρμογές οργάνωσης χρόνου
 - ...
- Δημιουργία περιεχομένου:
 - Forums, newsgroups, ...
 - Blogs
 - Photo galleries (flickr, deviantArt, ...)
 - Wiki's
 - Αγγελίες (craigslist, ebay, ...)
 - ...

Άλλες εφαρμογές

- Software As A Service (SaaS)
 - (σε εφαρμογές που δεν έχουμε σκεφτεί ότι θα μπορούσαν να υπάρξουν στο web)
- Caching για ταχύτητα
- Αποφυγή μικροδιακοπών
- Μείωση bandwidth / resources
- Desktop integration