



HY454 : ΑΝΑΠΤΥΞΗ ΕΞΥΠΝΩΝ ΔΙΕΠΑΦΩΝ ΚΑΙ ΠΑΙΧΝΙΔΙΩΝ

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ,
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ,
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ**



**ΔΙΔΑΣΚΩΝ
Αντώνιος Σαββίδης**



Γενικές πληροφορίες (1/2)

- Περίπου 20 διαλέξεις, 2 ενότητες, ανισοβαρείς (we do more about games)
 - Γλώσσα προγραμματισμού η C++
- Το project είναι σχετικά μεγάλο και δύσκολο, αλλά αρκετά προκλητικό και διασκεδαστικό
 - Ομάδες το πολύ τριών ατόμων και υποχρεωτικό
 - Τελική εξέταση σε δύο φάσεις: (α) κάθε ομάδα ξεχωριστά ελέγχοντας τη συμμετοχή των μελών στην ανάπτυξη και (β) κοινή παρουσίαση όλων των ομάδων
 - Όποιος δε συμμετέχει ουσιαστικά θα βαθμολογείται αναλόγως (μπορεί και να μηδενίζεται)
 - Θα ανατεθεί στην αρχή της δεύτερης ενότητας
 - ◆ *Super Mario, NES Version, υλοποιείται σε φάσεις*



Γενικές πληροφορίες (2/2)

- Τρόπος βαθμολογίας:
 - 100% από το project (χωρίς τελικό διαγώνισμα)
- Σημειώσεις είναι το περιεχόμενο των διαλέξεων - PowerPoint / PDF - στο site του μαθήματος, την επόμενη μέρα της κάθε διάλεξης
 - Αν και θα χρειαστεί να κρατάτε σημειώσεις για επεξηγήσεις οι οποίες δεν θα υπάρχουν στις διαφάνειες
- Το project που θα αφορά την ανάπτυξη των games
 - Για **rendering** και **input**: βιβλιοθήκη Allegro ή SDL (ή άλλο αντίστοιχο επιλέξετε)
 - Για **physics**: Box2D (<http://box2d.org>)
 - ◆ Θα δούμε εάν το προλάβουμε αυτό
 - Ξεκινήστε από τώρα να ασχολείστε με κάποια από αυτές τις βιβλιοθήκες ώστε να μην τρέχετε



Οι δύο ενότητες

■ Έξυπνες διεπαφές (*Intelligent User Interfaces*)

- Επιπλέον features στο User Interface, πέρα από την προσφορά των βασικών λειτουργιών μίας εφαρμογής, που κρίνονται ως έξυπνα χαρακτηριστικά

■ Παιχνίδια (*Video games*)

- Πως κατασκευάζουμε αλληλεπιδραστικούς κόσμους παιχνιδιών για γρήγορη δράση
 - ◆ δισδιάστατους στο rendering και βασισμένους σε bitmaps
 - ◆ side-scrolling with platforms
 - ◆ rigid-body 2D physics (optional)



Τι θα μάθετε (1/2)

Intelligent User Interfaces

Interactive **configurations**

Adaptive **automations**

Personalized **recommendations**



Τι θα μάθετε (2/2)

Video games

Game industry and history

Direct video access

Tile-based terrains, platforms

Films, sprites, collisions

Animations, timing, animators

Rigid-body physics

Moving platforms

Binary space partition (BSP) trees

Particle systems (effects)

Procedural content generation (PCG)



*Εάν προλάβουμε, θα
μιλήσουμε για κάποιο
από αυτά*



Χρησιμότητα

■ *Intelligent User Interfaces*

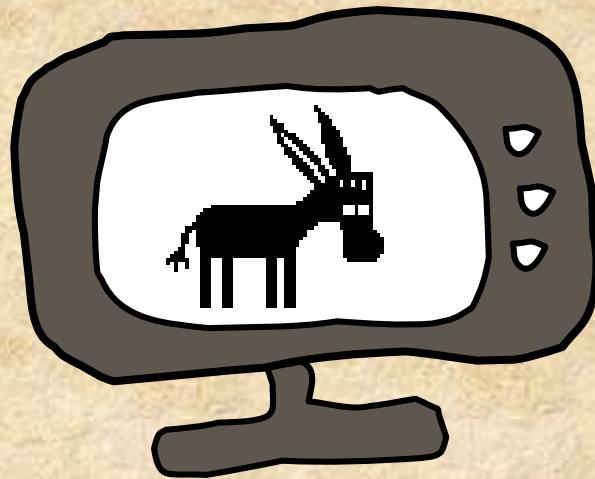
- Τα έξυπνα χαρακτηριστικά των εφαρμογών είναι σήμερα ένας πρωταρχικός **παράγοντας εμπορικής επιτυχίας**
 - ◆ Η ανάπτυξη τους ωστόσο είναι αρκετά πολύπλοκη και απαιτούν ειδική τεχνογνωσία
- Ανήκουν στα δημοφιλή χαρακτηριστικά που επηρεάζουν πολύ την αποδοχή από τους χρήστες, τα γνωστά ως **cool features**



Χρησιμότητα

■ *Video games*

- Τεράστια βιομηχανία με τις περισσότερες εταιρείες
- Το 2019 έφτασε τα 110 δις **ξεπερνώντας αθροιστικά τη βιομηχανία κινηματογράφου, sports media, και web tv**
 - ◆ ~25% αφορούν PC games
 - ◆ **το μεγαλύτερο τμήμα του τομέα μέσων και ψυχαγωγίας** (όπου ανήκει η τηλεόραση και το internet)
- Ανταγωνιστικός τομέας, **με το μεγαλύτερο ποσοστό των τίτλων να μην προσεγγίζουν τους εμπορικούς τους στόχους**
 - ◆ Ωστόσο με τις χειρότερες συνθήκες εργασίες για το χώρο υψηλής τεχνολογίας (κυρίως στα μεγάλα game studios).



**ΕΞΥΠΝΕΣ ΔΙΕΠΑΦΕΣ,
Σύνολο διαλέξεων 3, Διάλεξη 1η**



Έξυπνες διεπαφές

- **Ορισμοί**
- Κατηγορίες λειτουργικότητας
- Παραδείγματα
- Τεχνικές υποστήριξης χειροκίνητης προσαρμογής



Ορισμοί (1/4)

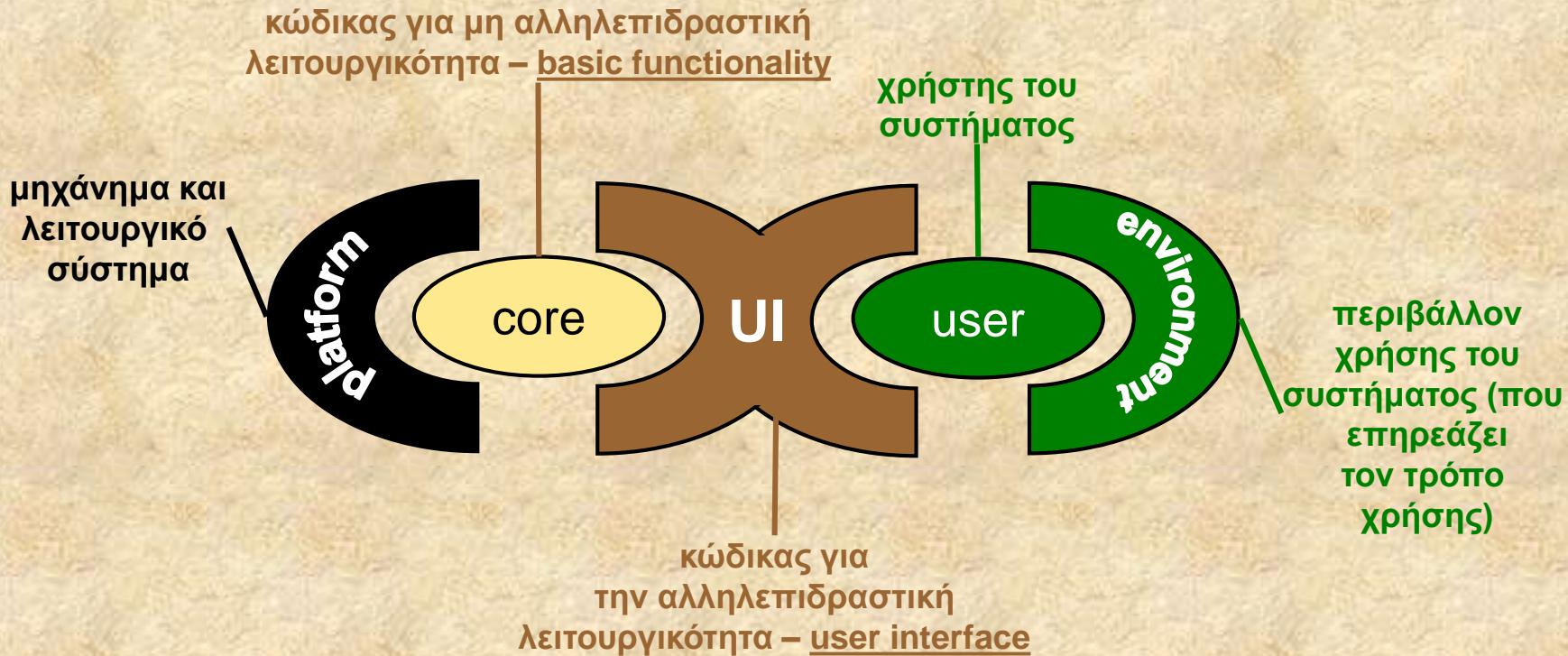
- Όπως και στους ανθρώπους, δεν είναι πολύ ξεκάθαρο (φορμαλιστικά) πως χαρακτηρίζουμε μία διεπαφή ως ευφυή ή έξυπνη
- Υπάρχουν πολλές απόψεις και διαφορετικοί ορισμοί, ενώ υφίστανται συνέδρια με θέμα γύρων από τις έξυπνες διεπαφές
- Δεν υπάρχει λόγος για τον οποίο σε παραλληλισμό με την τεχνητή νοημοσύνη δεν μιλάμε για **ΤΕΧΝΗΤΩΣ ΕΥΦΥΕΙΣ ΔΙΕΠΑΦΕΣ** – artificially intelligent user-interfaces
 - Ίσως επειδή στις τεχνικές κατασκευής δεν υπάρχει καμία σχέση μεταξύ των δύο πεδίων (το γιατί είναι μεγάλη ιστορία...)



Ορισμοί (2/4)

- Η δυνατότητα παροχής στους χρήστες κάποιων επιπλέων ευκολιών αποτελεί το χαρακτηριστικό έξυπνης λειτουργικότητας
 - Όσο πιο εξελιγμένα τα επιπρόσθετα χαρακτηριστικά, τόσο πιο έξυπνη θα χαρακτηρίζαμε μία διεπαφή
 - Αναφερόμαστε σε features πέρα από τις θεμελιώδεις λειτουργίες που μπορεί να παρέχει η εφαρμογή
 - *Στην πράξη διευκολύνουν πολύ τη χρήση των βασικών application features*

Ορισμοί (3/4)





Ορισμοί (4/4)

Ability to classify the user in a larger group and make suggestions for likely choices

Recommendations

Ability to tune the UI and also some application behavior

Ability to track frequent user actions and automate or make easier to apply

Configurations

Automations

User Interface

Functionality

Basic application functionality



Έξυπνες διεπαφές

- Ορισμοί
- ***Κατηγορίες λειτουργικότητας***
- Παραδείγματα
- Τεχνικές υποστήριξης χειροκίνητης προσαρμογής



Κατηγορίες λειτουργικότητας (1/3)

- Χειροκίνητη προσαρμογή από τον χρήστη του συστήματος - ***configuration***
 - Προσφέρονται διάφορες δυνατότητες στον τελικό χρήστη για να προσαρμόσει μόνος τα χαρακτηριστικά της διεπαφής στις δικές του προτιμήσεις και ανάγκες
 - Το είδος των προσφερόμενων λειτουργιών προσαρμογής, καθώς και ο τρόπος υλοποίησης τους, μπορεί να ποικίλει πολύ



Κατηγορίες λειτουργικότητας (2/3)

- Αυτόματη προσαρμογή με αυτοματισμούς ενεργειών - *automations*
 - Στην περίπτωση αυτή η διεπαφή παρακολουθεί την αλληλεπίδραση με τον χρήστη και καταγράφει συχνά επαναλαμβανόμενες ενέργειες με σκοπό να παρέχει αυτοματισμούς ή συντομεύσεις
 - Κάποιες αλλαγές γίνονται χωρίς παρέμβαση του χρήστη, ενώ κάποιες άλλες απαιτούν δική του επιβεβαίωση



Κατηγορίες λειτουργικότητας (3/3)

■ Έξυπνες συστάσεις - *recommendations*

- Οι κινήσεις και οι επιλογές τους χρήστη στην επιλογή κάποιων αντικειμένων καταγράφονται μαζί με αυτές για μία πληθώρα άλλων χρηστών και ομαδοποιούνται βάσει κοινών ενεργειών
- Εάν εντοπιστούν ομοιότητες μία αλληλουχίας με κάποια ομάδα, η αλληλουχία αυτή συμπληρώνεται με προτάσεις από επιλογές που έκαναν στη συνέχεια οι χρήστες της ίδιας ομάδας



Έξυπνες διεπαφές

- Ορισμοί
- Κατηγορίες λειτουργικότητας
- **Παραδείγματα**
- Τεχνικές υποστήριξης χειροκίνητης προσαρμογής



Παραδείγματα (1/7)

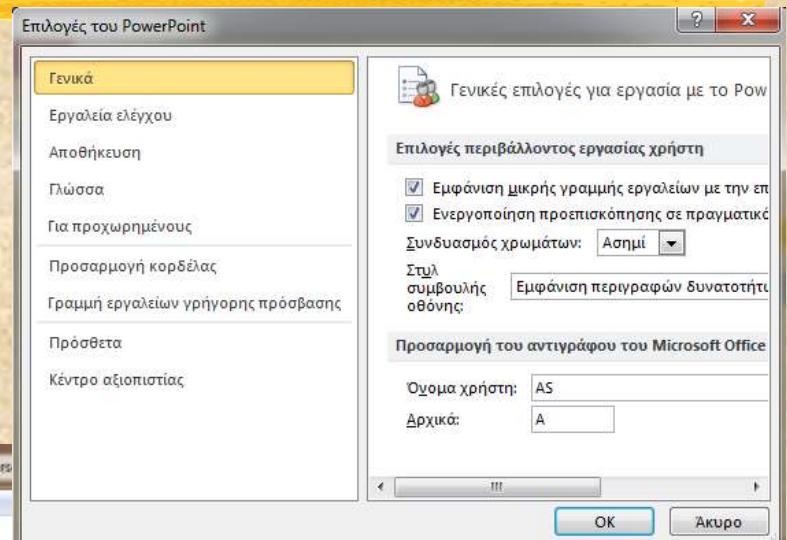
Προσαρμογή των χαρακτηριστικών κάποιων λειτουργιών. Το αποτέλεσμα είναι αντιληπτό εφόσον εκτελεστούν οι σχετικές λειτουργίες μετά τις οποίες προσαρμογές

The screenshot shows the Windows Registry Editor window. The left pane displays a tree view of registry keys under 'PeerNet'. The right pane is a table with columns 'Name', 'Type', and 'Data'. One entry in the table is 'Background' with type 'REG_SZ' and value '0 0 0'. Other entries include 'Default', 'AutoRestartShell', 'CachedLogonsC...', 'DebugServerCo...', 'DisableCAD', 'ForceUnlockLog...', 'LegalNoticeCap...', 'LegalNoticeText', 'PasswordExpiry...', 'PowerdownAft...', 'PreCreateKnow...', and 'ReportBootOk'.

Name	Type	Data
(Default)	REG_SZ	(value not set)
AutoRestartShell	REG_DWORD	0x00000001 (1)
Background	REG_SZ	0 0 0
CachedLogonsC...	REG_SZ	10
DebugServerCo...	REG_SZ	no
DisableCAD	REG_DWORD	0x00000001 (1)
ForceUnlockLog...	REG_DWORD	0x00000000 (0)
LegalNoticeCap...	REG_SZ	
LegalNoticeText	REG_SZ	
PasswordExpiry...	REG_DWORD	0x00000005 (5)
PowerdownAft...	REG_SZ	0
PreCreateKnow...	REG_SZ	{A520A1A4-1780-4FF6-B...
ReportBootOk	REG_SZ	1

Προσαρμογές που μοιάζουν με αρχεία κειμένου (configuration files), στο στυλ key: value

Manual configuration



Εδώ το configuration είναι live, δηλαδή οι επιλογές δοκιμάζονται σε πραγματικό χρόνο βλέποντα άμεσα το αποτέλεσμα που επιφέρουν



Παραδείγματα (2/7)

Automations



Αυτόματη αναδιάταξη των επιλογών ενός menu βάσει της συχνότητας επιλογής των λειτουργιών.

Automations (*program editing automations*)

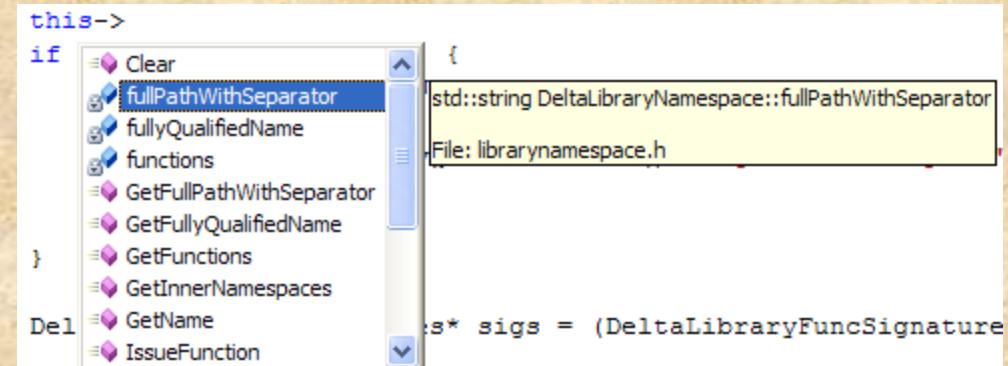
```
SetHandler (
    const std::string& nodeTag,
    Handler f,
    typedef void (*AttributeStringifier::Handler)(const std::string &, void *, std::string *, void *)
);
```

Αυτά είναι τα γνωστά type information tooltips του IntelliSense

```
DELTASYMBOLS.NewSymbol(fullName, false);
yFunc = true;
igs = sigs;
```

DeltaSymbol *NewSymbol (const std::string &name, bool storage = true)

Αυτό είναι το actual arguments guidance feature του IntelliSense



Αυτό είναι το auto complete feature του IntelliSense



Παραδείγματα (3/7)

Automations

The screenshot shows a code editor window with a list of method suggestions for the variable 'albit'. The suggestions include:

- @items = [@bag : std::list new(), @grid: []]; albit
- bitmap_construct()/* (width:Number, height:Number, isVideo:Boolean): bitmap*/
- bitmap_destroy()/* (b:bitmap)*/
- bitmap_load()/* (path:String, isVideo:Boolean): bitmap*/
- bitmap_loadimage()/* (path:String): bitmap*/
- bitmapmanager_cleanup()/* () : Void*/
- bitmapmanager_get()/* (rcId:String) : aladin::bitmap*/
- bitmapmanager_load()/* (rcId:String, isVideo:Boolean) : aladin::bitmap*/
- bitmapmanager_loadany()/* (rcId:String) : aladin::bitmap*/
- bitmapmanager_unload()/* (rcId:String)*/
- drawable_black()/* () : Number*/
- drawable.blit()/* (dest:drawable, src:drawable, src_x:Number, src_y:Number, dest_x:Number, dest_y:Number): void*/
- drawable.blitbyblurring()/* (dest:drawable, src:drawable, src_x:Number, src_y:Number, dest_x:Number, dest_y:Number): void*/
- drawable.blitbyblurringsource()/* (dest:drawable, src:drawable, src_x:Number, src_y:Number, dest_x:Number, dest_y:Number): void*/
- drawable.blitbydarkening()/* (dest:drawable, src:drawable, src_x:Number, src_y:Number, dest_x:Number, dest_y:Number): void*/

Αυτό είναι το auto complete feature με υποστήριξη user-defined abbreviations του i-sense στο Sparrow IDE για τη γλώσσα Delta

The screenshot shows a code editor with a completion dropdown for the '#group(albit)' command. The dropdown lists methods grouped under 'albit':

- bitmap_construct
- bitmap_load
- bitmap_loadimage
- bitmap_destroy
- drawable_white
- drawable_black

Details for 'bitmap_construct':

- (width:Number, height:Number, isVideo:Boolean): bitmap
- (path:String, isVideo:Boolean): bitmap
- (path:String): bitmap
- (b:bitmap)
- () : Number
- () : Number

White box library API with optional user-defined auto-completion groups (abbreviations).



Παραδείγματα (4/7)

Απόσπασμα από iOS, Human Interface Guidelines

In iOS, interface elements and layouts can be configured to automatically change shape and size on different devices, during multitasking on iPad, in split view, when the screen is rotated, and more. ***It's essential that you design an adaptable interface that provides a great experience in any environment.***

<https://developer.apple.com/design/adaptivity/>

Device	Portrait dimensions	Landscape dimensions
12.9" iPad Pro	2048px × 2732px	2732px × 2048px
10.5" iPad Pro	1668px × 2224px	2224px × 1668px
9.7" iPad	1536px × 2048px	2048px × 1536px
7.9" iPad mini 4	1536px × 2048px	2048px × 1536px
iPhone X	1125px × 2436px	2436px × 1125px
iPhone 8 Plus	1242px × 2208px	2208px × 1242px
iPhone 8	750px × 1334px	1334px × 750px
iPhone 7 Plus	1242px × 2208px	2208px × 1242px
iPhone 7	750px × 1334px	1334px × 750px
iPhone 6s Plus	1242px × 2208px	2208px × 1242px
iPhone 6s	750px × 1334px	1334px × 750px
iPhone SE	640px × 1136px	1136px × 640px



Παραδείγματα (5/7)

The image shows two screenshots from the game Fallen God: Escape Underworld. The top screenshot displays a menu with various options like Input, Camera, Display, Game, Save options, Story, Credits, Purchase, and Close. A sub-menu for 'Camera' is open, showing 'Edit Preview' and 'Close Apply' buttons. The bottom screenshot shows the same menu, but the 'Edit' button for 'Camera' is highlighted. The sub-menu now lists camera parameters: Up [2,7.5]: 2.93, Behind [9,20]: 20, Angle [-15,15]: -7.5, Apply, Restore, and Close.

menu stage 1 of 23

Input Types Select: low.far(curr)
Camera Edit Preview
Display Close Apply
Game Close
Save options
Story
Credits
Purchase
Close

menu stage 1 of 23

Input Types
Camera Edit
Display Close
Game Up [2,7.5]: 2.93
Save options Behind [9,20]: 20
Story Angle [-15,15]: -7.5
Credits Apply
Purchase Restore
Close

Live camera configuration,
Fallen God: Escape Underworld (iOS)

<https://www.facebook.com/fallen.god.escape.underworld/videos/1193978297313933/>



Παραδείγματα (6/7)

Recommendations, YouTube
choice

YouTube search results for "scorpions rock you like a hurricane". The search bar is highlighted with a red box. The main video thumbnail shows a scorpion on a black background with the text "SCORPIONS". Below it is a progress bar showing 0:06 / 4:16.

AC/DC - thunderstruck (lyric video)
SuperSleepingbee
9.1M views
4:53

Mix - Scorpions - Rock You Like A Hurricane
YouTube
50+ 8:0

Metallica - Nothing Else Matters [Official Music Video]
MetallicaTV
65M views
6:26

Kiss - I was made for lovin' you (lyrics)
KissTV
4:53

suggestions

YouTube search results for "prodigy no good". The search bar is highlighted with a red box. The main video thumbnail shows a person's face in a dark setting. Below it is a progress bar showing 0:00 / 4:16.

Up next

Bomfunk MC's - Freestyle (Video Original Version)
BomfunkMCsVEVO
1.6M views
4:53

Mix - The Prodigy - No Good (Start The Dance) (Official YouTube
50+ 8:0

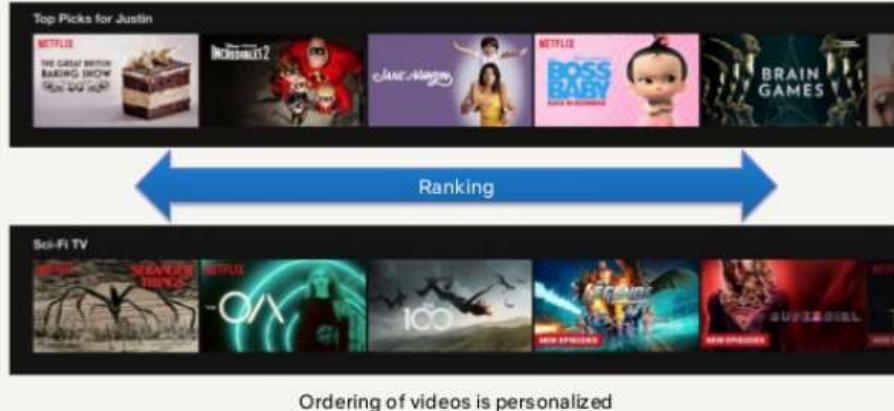
Coolio - Gangsta's Paradise (feat. L.V.) [Official Music Video]
Tommy Boy
224M views
4:17

The Prodigy - Phoenix Festival 1996
vtraduction23
1.5M views
58:16



Παραδείγματα (7/7)

From what we recommend



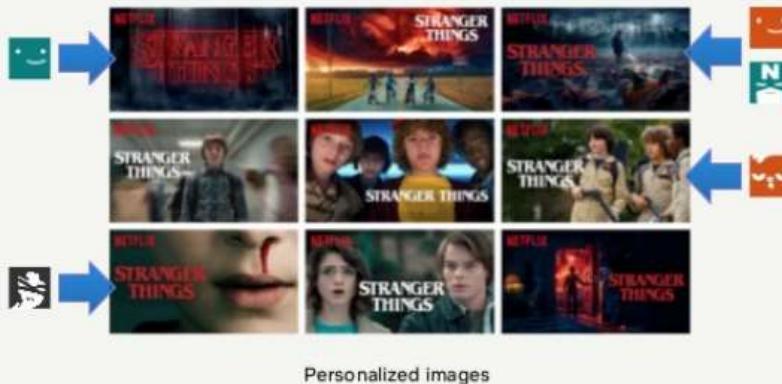
Recommendations, Netflix



... to how we construct a page



... to what images to select





Ευφυείς διεπαφές

- Ορισμοί
- Κατηγορίες λειτουργικότητας
- Παραδείγματα
- **Τεχνικές υποστήριξης χειροκίνητης προσαρμογής**



Χειροκίνητη προσαρμογή

- **Αρχεία προσαρμογής** – configuration files
 - Παρέχονται αρχεία κειμένου για τον ορισμό των προσαρμογών
- **Διαλογικά εργαλεία** – interactive configuration tools
 - Παρέχονται ειδικές διεπαφές – εργαλεία για την προσαρμογή
- **Προγραμματισμός με μικρογλώσσες** - scripting languages
 - Προσφέρεται ειδική γλώσσα προγραμματισμού για πιο πολύπλοκες προσαρμογές

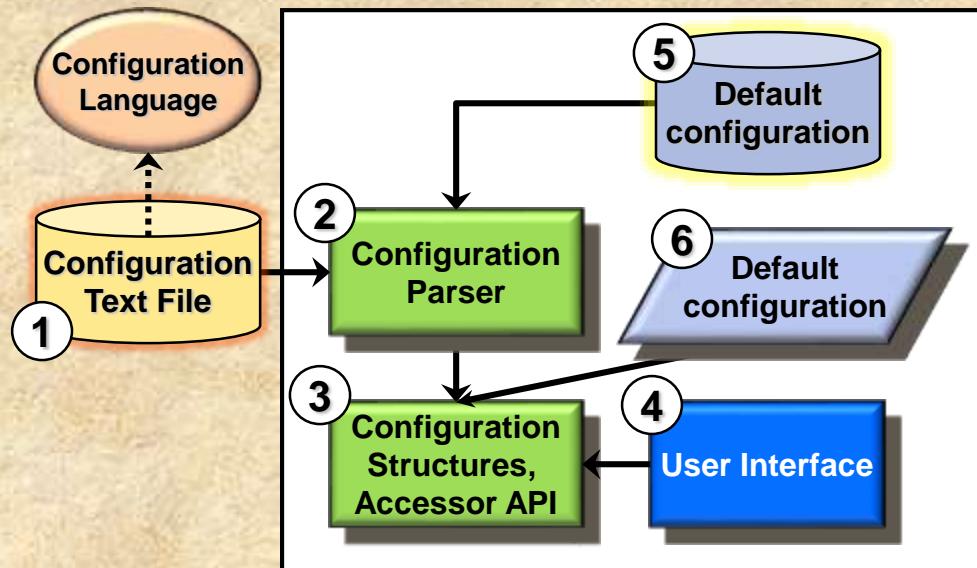


Αρχεία προσαρμογής (1/10)

- Μία από τις παλαιότερες τεχνικές, αλλά και από τις πιο ισχυρές και ευέλικτες σε δυνατότητες.
- Για να υποστηριχθεί χρειάζεται:
 - **Επιλογή των προσαρμόσιμων χαρακτηριστικών της διεπαφής** (π.χ. χρώματα, backgrounds, μηνύματα, χωροταξία, διαθέσιμες λειτουργίες, κλπ).
 - **Τον ορισμό της γλώσσας προσαρμογής** (configuration language), συνήθως μία απλή κανονική γλώσσα, με ελάχιστα keywords και πολύ απλό συντακτικό.
 - **Την κατασκευή ενός κατάλληλου parser**, που θα διαβάζει τα configuration files και θα εφαρμόζει τις όποιες προσαρμογές πάνω στη διεπαφή.
- Υπάρχουν περιπτώσεις στις οποίες τα configuration files είναι αρκετά πολύπλοκα, προσφέροντας μεγάλες δυνατότητες προσαρμογών
 - ...εφόσον ο χρήστης γνωρίζει καλά την γλώσσα που προσφέρεται και τις δυνατότητες της



Αρχεία προσαρμογής (2/10)



1. Το εξωτερικό αρχείο προσαρμογής.
2. Ο parser που διαβάζει τα configuration files.
3. Οι εσωτερικές δομές των configuration data (μία ή περισσότερες κλάσεις).
4. Η υλοποίηση της διεπαφής (έχει τη δική της κατάτμηση).
5. Το default configuration ως εξωτερικό αρχείο.
6. Το default configuration ως εσωτερικά δεδομένα ή εντολές αρχικοποίησης.

- Αποφεύγουμε να έχουμε hard-coded default values στην υλοποίηση της διεπαφής.
- Επειδή μπορεί ένα εξωτερικό αρχείο να σβηστεί από τον χρήστη, το default configuration μπορεί να είναι:
 - ένα constant string με το περιεχόμενο που θα είχε εάν ήταν εξωτερικό αρχείο,
 - σύνολο εντολών αρχικοποίησης (να το αποφεύγετε, άδικος κώδικας),
 - ή ένα ενσωματωμένο αρχείο σε κάποιο data package της εφαρμογής (καλύτερη λύση)



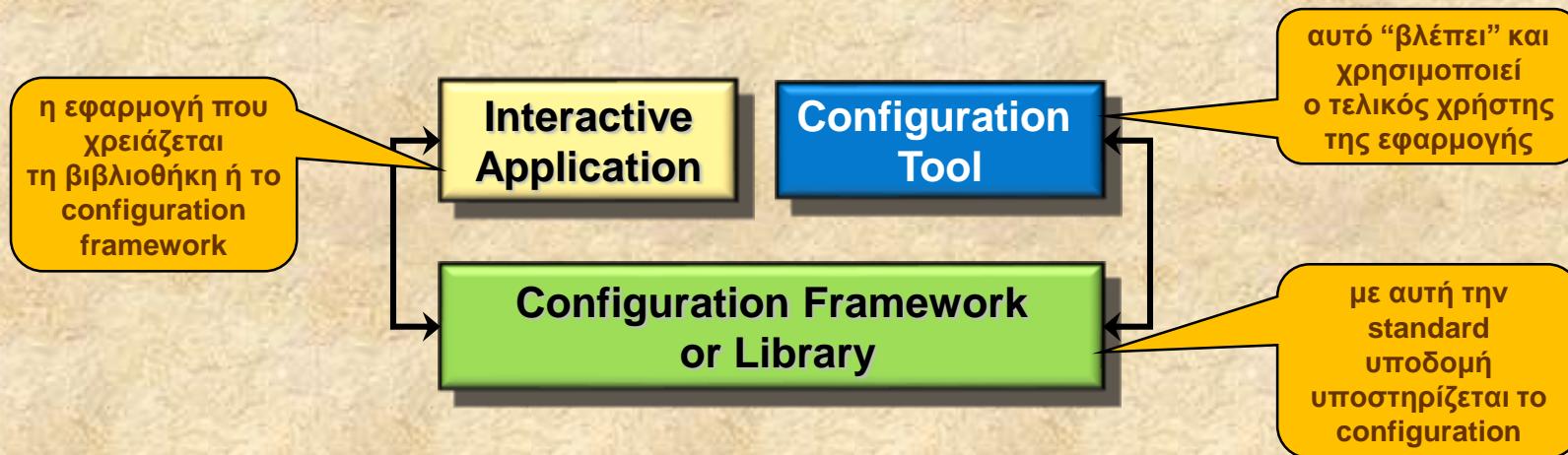
Αρχεία προσαρμογής (3/10)

- Τεχνικά πλεονεκτήματα
 - **Δυνατότητα κλιμάκωσης – scalability**
 - ◆ Μπορεί να εισάγονται εύκολα στη γλώσσα νέα χαρακτηριστικά της διεπαφής που μπορούν να είναι προσαρμόσιμα (απαιτούνται αλλαγές στα εμπλεκόμενα τμήματα)
 - **Αποφεύγονται εντολές αρχικοποίησης** και hard-coded values στον κώδικα (είναι *malpractice*)
 - ◆ Σε μεγάλα User Interfaces κάτι τέτοιο μπορεί να γίνει πονοκέφαλος
 - Υπάρχει η **δυνατότητα ύπαρξης πολλών εναλλακτικών configuration files**, για διαφορετικές περιπτώσεις χρήσης ή χρήστες του ίδιου συστήματος
 - ◆ η ενεργοποίηση απαιτεί συνήθως file renaming, η μπορεί να αυτοματοποιηθεί με μία αρχική οθόνη επιλογής του active configuration
 - Υλοποιείται **εύκολα η ενεργοποίηση κάποιου άλλου configuration** ενώ το σύστημα ήδη τρέχει
 - ◆ Αυτό, όπως θα δούμε αργότερα, μπορεί να προϋποθέτει κατάλληλη υποδομή στην υλοποίηση του User Interface

Αρχεία προσαρμογής (4/10)

■ Configuration frameworks (1/7)

- Έτοιμες υποδομές τις οποίες μπορεί ο κατασκευαστής να τις χρησιμοποιήσει
- Συνήθως παρέχεται η γλώσσα, ο *parser* και τα configuration *structures* με το *accessor API*





Αρχεία προσαρμογής (5/10)

■ Configuration frameworks (2/7)

- Παράδειγμα: **X Attribute Resource Files** για το **Xt / Xlib** (X windowing system - *tribute to history*)
 - ◆ Από τα πλέον προηγμένα πλαίσια προσαρμογών για GUI libraries με πολύπλοκους κανόνες και pattern matching
 - ◆ Επιτρέπει αναφορά στα configuration files σε στιγμιότυπα αντικειμένων που υπάρχουν στον κώδικα μέσω αναγνωριστικών ονομάτων
 - ◆ Υποστηρίζει προσαρμογή όλων των χαρακτηριστικών γνωρισμάτων (attributes) των αντικειμένων (background / foreground color, font, border width, label, κλπ)
 - ◆ Υποστηρίζει κανόνες προσαρμογής για οικογένειες αντικειμένων (π.χ. όλα τα Buttons, όλα τα αντικείμενα που έχουν ως parent ένα συγκεκριμένο window, κλπ).



Αρχεία προσαρμογής (6/10)

■ Configuration frameworks (3/7)

```
Prolifics.black:          #000000
Prolifics.blue:          #0000a8
Prolifics.green:         #00a800
Prolifics*foreground:    white
Prolifics*background:   dark slate gray
Prolifics*XmMenuShell*windows_raise.labelXString:      Raise All
Prolifics*XmMenuShell*windows_raise.mnemonic:           R
Prolifics*XmMenuShell*edit_cut.labelXString:            Cut
Prolifics*XmMenuShell*edit_cut.mnemonic:                 t
Prolifics*XmMenuShell*edit_copy.labelXString:           Copy
Prolifics*XmMenuShell*edit_copy.mnemonic:                C
Prolifics*defaultVirtualBindings:                      \n\
osfMenu:          <Key>F26          \n\
osfMenuBar:       <Key>F25          \n\
osfActivate:     <Key>KP_Enter     \n\
osfCancel:       <Key>Escape       \n\
osfDown:         <Key>Down        \n\
osfLeft:         <Key>Left         \n\
osfRight:        <Key>Right        \n\
osfUp:           <Key>Up          \n\
osfBackSpace:    <Key>BackSpace   \n\
osfDelete:       <Key>Delete      \n\
osfHelp:          <Key>F1          \n\

void osfMenu(void);
.
```

- Απόσπασμα από attribute resource files για το Motif. Υποστηρίζει και σχόλια με ! language να υποστηρίζει).
- Μπορούν να γίνονται exported και συναρτήσεις void(*)void) ως actions με αντιστοίχηση σε πλήκτρα.
- Το πρώτο id αριστερά (π.χ. **Prolifics**) είναι το όνομα της εφαρμογής (optional).
- Υπάρχει «συμφωνία» με την ονοματολογία των widgets στην υλοποίηση.
- Ο τελεστής * κάνει matching «οτιδήποτε» στην ιεραρχία των widgets (Kleen star).
- Matching επιτρέπεται σε επίπεδο class, instance και properties



Αρχεία προσαρμογής (7/10)

■ *Configuration frameworks (4/7)*

● **INI (initialisation) files (Windows)**

- ◆ Υποστηρίζει ομάδες από δηλώσεις αρχικοποίησης αναγνωριστικών ονομάτων σε default τιμές
- ◆ Κάθε ομάδα από έχει ένα μοναδικό αναγνωριστικό όνομα, ενώ μπορούν σε διαφορετικές ομάδες να εμφανίζονται παρόμοια ονόματα
- ◆ Προσφέρεται ένα απλό API για την εξαγωγή των τιμών αρχικοποίησης βάση ομάδας και αναγνωριστικού ονόματος
- ◆ Μπορεί να χρησιμοποιηθεί όπως είναι επιθυμητό από τον προγραμματιστή
- ◆ Τα αναγνωριστικά ονόματα είναι πολύ απλά και δεν επιτρέπουν διάφορα dirty tricks (π.χ. dotted names ή constant strings)



Αρχεία προσαρμογής (8/10)

■ Configuration frameworks (5/7)

```
[Applications List]
App1=Microsoft Windows XP Service Pack 1
App2=Compaq Diagnostics for Windows

[Microsoft Windows XP Service Pack 1]
Description=Microsoft Windows XP Service Pack 1

[Compaq Diagnostics for Windows]
Description=Compaq Diagnostics for Windows Setup
Directory=Diags
FileName=cpqdiag.exe
FileNameTarget=WINDIR
FileNameSubDirectory=CPQdiag
FileVersion=3.8.2074
```

Κατασκευαστικά τα INI files είναι 2-level hashing, με το 1^ο επίπεδο να είναι η επιλογή του group και το δεύτερο να είναι η επιλογή του configuration variable

```
using namespace std;
typedef map<string, string> VarMap;
typedef map<string, VarMap> GroupMap;
GroupMap iniFile;
string GetIniValue (string group, string var);
```

- Απόσπασμα από INI file.
- Υποστηρίζει και σχόλια με ;
- Δεν υποστηρίζονται υποομάδες.
- Οι τιμές μπορεί να είναι μόνο string ή integer (για άλλου είδους τιμές κάνετε τις μετατροπές από string).
- Σε σύγκριση με τα attribute resource files είναι ειδικότερο πλαίσιο αλλά πιο ευέλικτο, ωστόσο δε δίνει δυνατότητες προσαρμογής των χαρακτηριστικών της διεπαφής!



Αρχεία προσαρμογής (9/10)

■ Configuration frameworks (6/7)

```
<editor>
  <keywords>
    <class>
      <name>Bold</name>
      <public>Normal</public>
      <private>Italics</private>
      <protected>Underlined</protected>
    </class>
  </keywords>
</editor>
```

- Η XML είναι γνωστή ως mark-up language με την οποία ουσιαστικά μπορεί να περιγράφεται οποιαδήποτε ιεραρχική δομή, με user-defined tags.
- Υπάρχουν διαθέσιμοι διάφοροι XML parsers.

- Κατασκευαστικά τα XML files είναι N-level hashing, με το m-οστό επίπεδο να είναι η επιλογή του attribute σε nesting m, ενώ υποστηρίζεται σε κάθε επίπεδο hashing τόσο με string όσο και με αριθμό (μήπως θυμίζει σε κάτι τους πίνακες της γλώσσας alpha...).
- Δηλ., ουσιαστικά είναι δυναμικά records, με ευθύνη για το parsing – conversion των values από strings στον προγραμματιστή.



Αρχεία προσαρμογής (9/10)

■ Configuration frameworks (7/7)

```
{  
  "font" : {  
    "family" : "Times Roman",  
    "pointSize": 12,  
    "italics" : false,  
    "bold"     : false  
  },  
  "editor" : {  
    "styleDialogue" : true,  
    "rulerDialogue" : false  
  }  
}  
  
var configData = JSON.parse(configText);
```

- Εναλλακτικά η JSON (JavaScript Object Notation) είναι πιο readable από XML και de facto standard για data interchange στο web
- Ενδεχομένως να πρέπει να φτιάξετε δικός σας parser αν και στις πιο πολλές γλώσσες υπάρχουν διαθέσιμοι.
 - **Φτιάξτε έναν parser μόνοι σας, θα χρειαστείτε το πολύ μία εβδομάδα**
 - **Suggestion: allow non-quoted ids and dotted ids as keys!**



Αρχεία προσαρμογής (10/10)

- Σε πολλές περιπτώσεις τα αρχεία προσαρμογής δεν θέλουμε να είναι διαθέσιμα στους χρήστες
- Τότε η ευχρηστία της γλώσσας δεν μας ενδιαφέρει ιδιαίτερα, καθώς τα αρχεία θα κρύβονται από κάποιο User Interface το οποίο θα χρησιμοποιεί ο χρήστης
- Σε τέτοιες περιπτώσεις μπορούμε να παραβούμε κάποιους κανόνες ή να αποδεχθούμε τη χρήση λύσεων που οδηγούν σε σχετικά «φλύαρα» αρχεία όπως η XML
- Γενικά η JSON είναι ένα de facto standard
- Η επιλογή σας πρέπει να σταθμίζει την πολυπλοκότητα των προσαρμογών, τη δυνατότητα γρήγορης κατασκευής ενός UI, τη δυσκολία της γλώσσας καθώς και τον πραγματικό ρόλο των προσαρμογών



Διαλογικά εργαλεία (1/3)

- ***Interactive configuration tools.*** Η λεπτομέρεια, το πλήθος και το είδος των υποστηριζόμενων προσαρμογών ποικίλει, συνήθως εξαρτώμενο από το κόστος που επενδύει ο κατασκευαστής
- Όταν παρέχονται διαλογικά εργαλεία υφίσταται πάντα και ο τρόπος αποθήκευσης σε configuration files
 - Ωστόσο, αυτά δεν είναι πάντα σε αναγνώσιμη μορφή κειμένου (δηλ. πρακτικά δεν μπορεί ο τελικός χρήστης να επεμβαίνει)
 - Αλλά στα πιο σοβαρά συστήματα τα διαλογικά εργαλεία παράγουν αναγνώσιμα και πλήρως τεκμηριωμένα (ως προς τη δομή) configuration files
 - **Στην πράξη τα διαλογικά εργαλεία πρέπει να είναι ένα επίπεδο πάνω από τα configuration files**

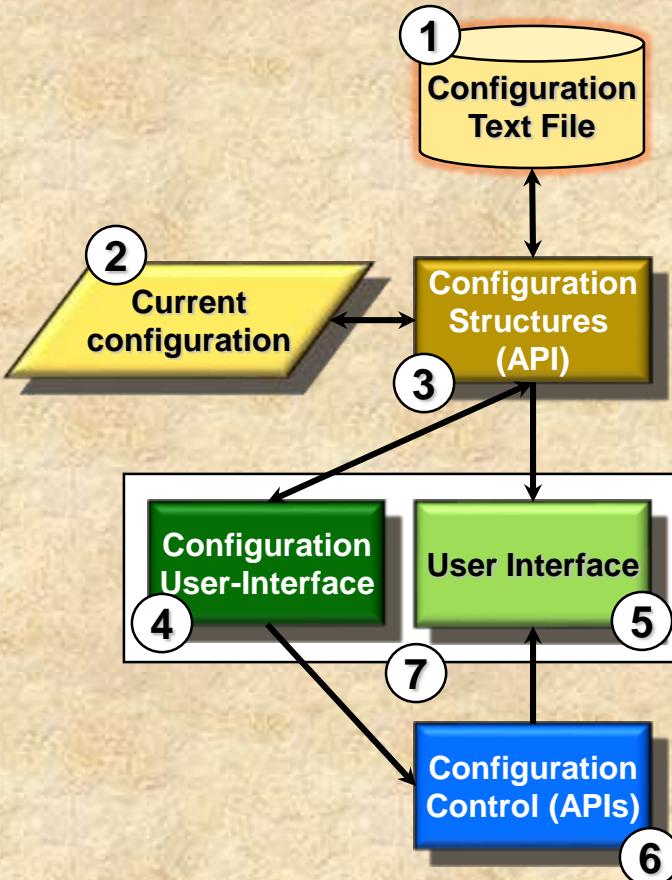


Διαλογικά εργαλεία (1/3)

- Επειδή στα διαλογικά εργαλεία πρέπει να υπάρχει δυνατότητα «πρόβας» (**preview**) αλλά και άμεσης εφαρμογής των προσαρμογών στο User Interface τηνώρα της εκτέλεσης:
 - πρέπει αυτό να προσφέρει κάποιο ειδικό API ώστε να κάνει refresh με νέες προσαρμογές χαρακτηριστικών τη στιγμή που η εφαρμογή τρέχει (**on-the-fly**)
- Ο χρήστης θέλει να μπορεί να δοκιμάζει αλλαγές σε χαρακτηριστικά με live preview χωρίς να είναι αυτές μόνιμες, εκτός και εάν το επιλέξει (**apply**)
 - Πρέπει να υλοποιείται και fallback για επαναφορά σε defaults, καθώς δοκιμάζοντας και κάνοντας apply μπορεί να διαπιστώσει ότι τα πράγματα είναι χειρότερα από πριν



Διαλογικά εργαλεία (2/3)



1. Το εξωτερικό αρχείο προσαρμογής
2. Το στιγμιότυπο της παρούσας προσαρμογής
3. Κλάσεις για την υλοποίησης των configuration data
4. Η υλοποίηση της διεπαφής προσαρμογής
5. Η διαλογικά προσαρμόσιμη διεπαφή (interactive configuration)
6. Ένα standard API το οποίο υλοποιεί κάθε διαφορετικό τμήμα της διεπαφής ώστε να υποστηρίζεται εξωτερικά έλεγχος και μεταβολή των προσαρμόσιμων χαρακτηριστικών
7. Το configuration User Interface, ως αναπόσπαστο τμήμα της συνολικής διεπαφής, υπόκειται και αυτό σε προσαρμογή, που σημαίνει ότι υλοποιεί και το ίδιο ένα configuration control API



Διαλογικά εργαλεία (3/3)

- Στην επόμενη διάλεξη θα δούμε τον τρόπο με τον οποίο μπορούμε να υλοποιήσουμε ένα configuration interface
 - **Επαναχρησιμοποιήσιμο** για όλες τις εφαρμογές
 - **Αυτόματα επεκτάσιμο** ως προς το πλήθος των προσαρμόσιμων χαρακτηριστικών της διεπαφής
 - **Ανεξάρτητο** από τον κώδικα των συγκεκριμένων τμημάτων της διεπαφής
- ➔ *Βασισμένο σε ένα abstract configuration API το οποίο υλοποιούν όλα τα τμήματα της διεπαφής*