



Corso di Progettazione di Applicazioni Web e Mobile

Hello!

I am Diego Bonura

Mi occupo di:

- Frontend
- Backend
- Mobile
- IoT
- Multimedia

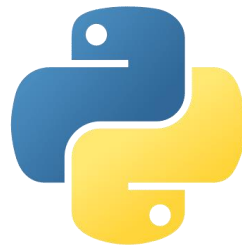
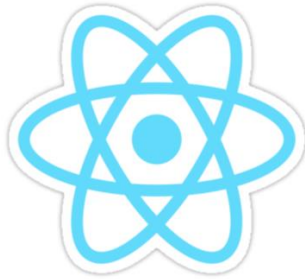
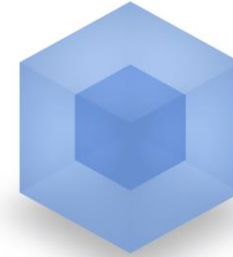
diego@studiograficod2.it



LOCCIONI



Cosa riconoscete?





Programma

- Sviluppo mobile: di cosa si tratta
- Architettura di una applicazione mobile
- Dalla prototipazione al deploy
- Testing
- ...

A decorative network diagram in the top-left corner, consisting of various sized circles (nodes) connected by thin lines (edges). Some nodes are solid grey, while others are hollow white with a grey border. The connections form a complex, interconnected web.

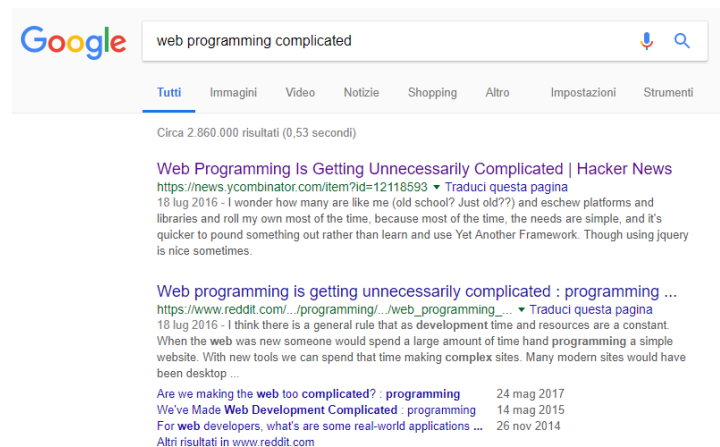
1.



Mobile development



“


Quanto è complesso sviluppare applicazioni web/mobile?




Google  

[Tutti](#) [Immagini](#) [Video](#) [Notizie](#) [Shopping](#) [Altro](#) [Impostazioni](#) [Strumenti](#)

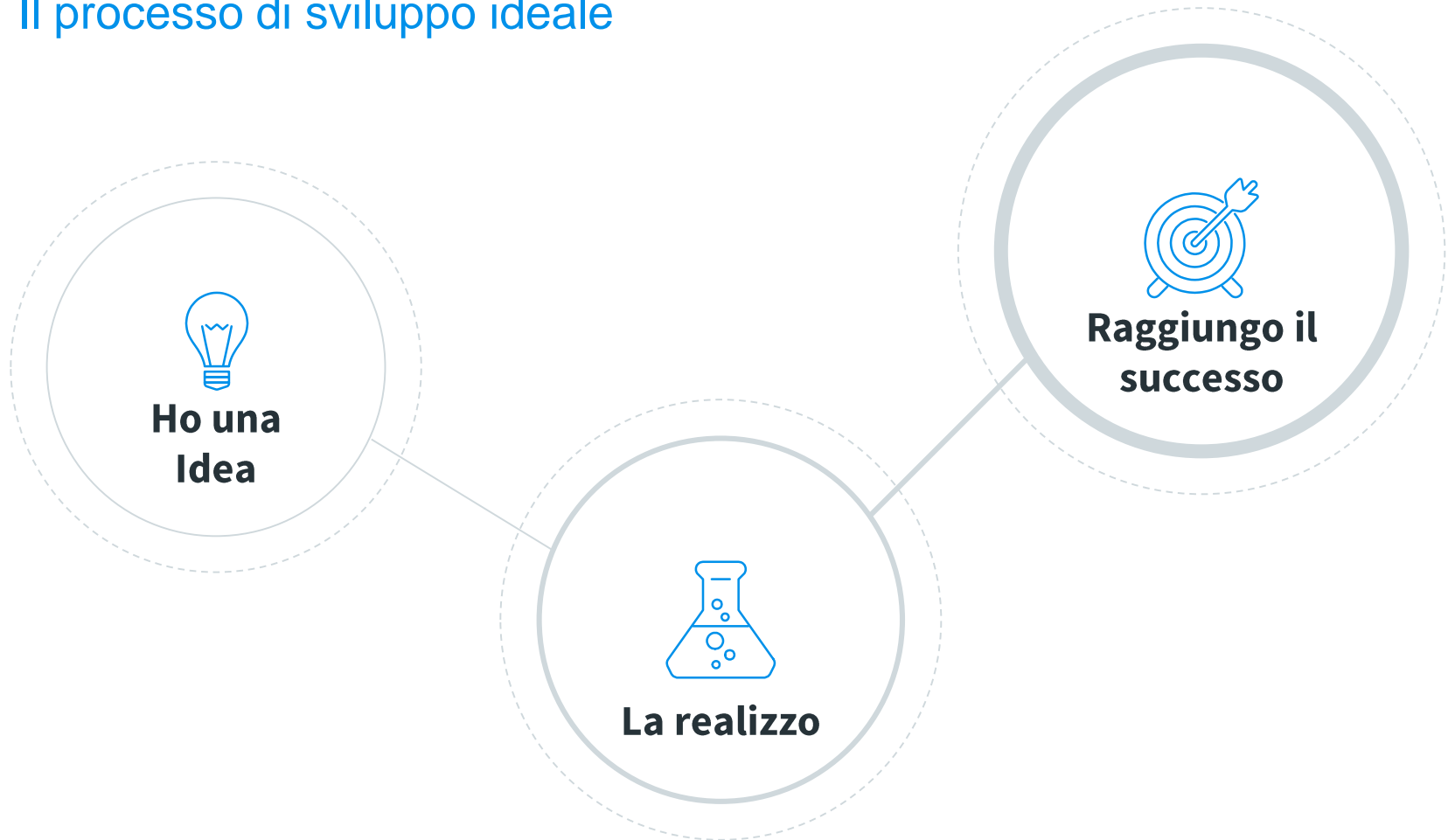
Circa 2.860.000 risultati (0,53 secondi)

Web Programming Is Getting Unnecessarily Complicated | Hacker News
<https://news.ycombinator.com/item?id=12118593>  Traduci questa pagina
18 lug 2016 - I wonder how many are like me (old school? Just old??) and eschew platforms and libraries and roll my own most of the time, because most of the time, the needs are simple, and it's quicker to pound something out rather than learn and use Yet Another Framework. Though using jquery is nice sometimes.

Web programming is getting unnecessarily complicated : programming ...
https://www.reddit.com/.../programming/.../web_programming...  Traduci questa pagina
18 lug 2016 - I think there is a general rule that as development time and resources are a constant. When the web was new someone would spend a large amount of time hand programming a simple website. With new tools we can spend that time making complex sites. Many modern sites would have been desktop ...

Are we making the **web too complicated?** : programming 24 mag 2017
We've Made **Web Development Complicated** : programming 14 mag 2015
For **web developers**, what's are some real-world applications ... 26 nov 2014
Altri risultati in www.reddit.com

Il processo di sviluppo ideale



Il processo di sviluppo reale (semplificato)

Idea

- Strategia
- Monetizzazione

Analisi

- Requirements
- Team
- Roadmap

Design

- UserExperience
- UserInterface

Mock

- Requirements
- Roadmap
- Realizzazione
- Feedback

Sviluppo

- Metodologia
- Backend
- Frontend
- Amministrazione

Testing

- Validazione requirements
- Beta phase
- Analytics

Deploy

- Cloud
- Store

Supporto

Use Cases (semplificato)



- Per l'utente:
 - Facile da rintracciare
 - Facile da installare
 - Facile da usare
 - Riconoscibile (con una propria identità)
 - Sicura
 - Stabile
 - Veloce
 - Poco energivora
 - Leggera nei trasferimenti
 - Con notifiche
 - Facile da condividere
 - Backup automatico



- Per lo sviluppatore:
 - Facile da mantenere
 - Facile da aggiornare
 - Che sia scalabile al crescere degli utenti
 - Che sia economica (cloud/server)
 - Sicura
 - Stabile
 - Che rispetti le linee guida degli store
 - A/B Test facile da integrare
 - Logger e altri servizi facili da integrare



- Per l'amministratore:
 - Facile da usare
 - Facile da analizzare
 - Facile da mantenere



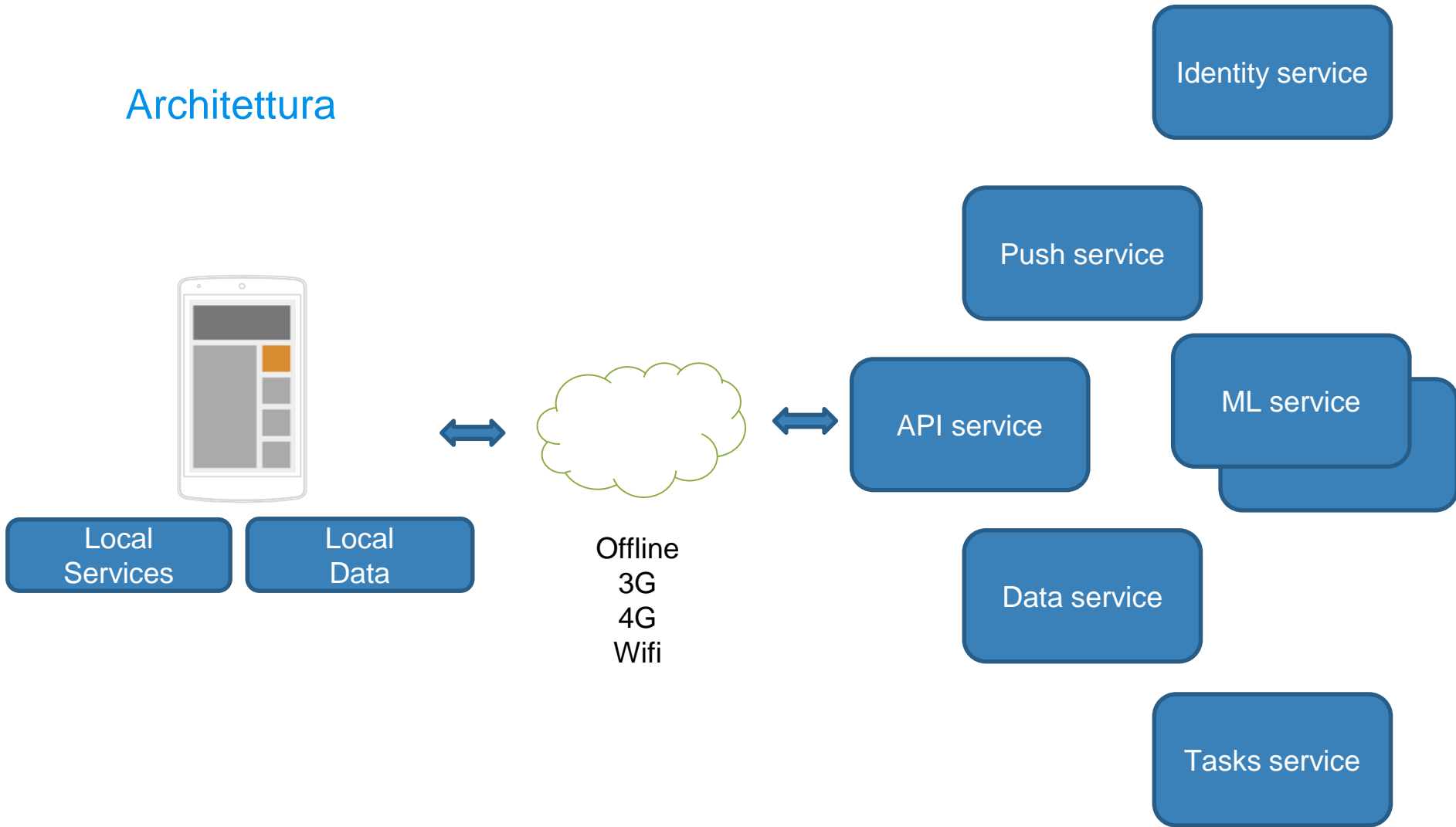
- Per il customer service
 - Facile da usare

A decorative network diagram in the top-left corner, consisting of various sized nodes (some solid grey, some hollow white) connected by thin grey lines, forming a complex web structure.

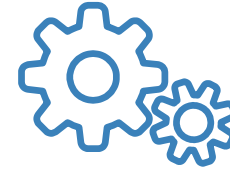
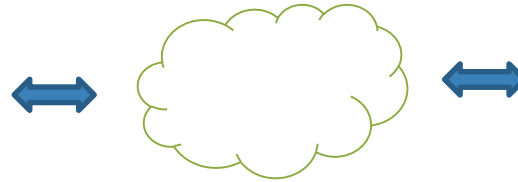
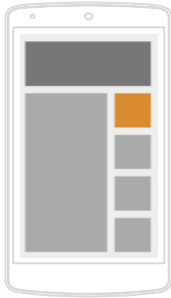
2.

Architettura

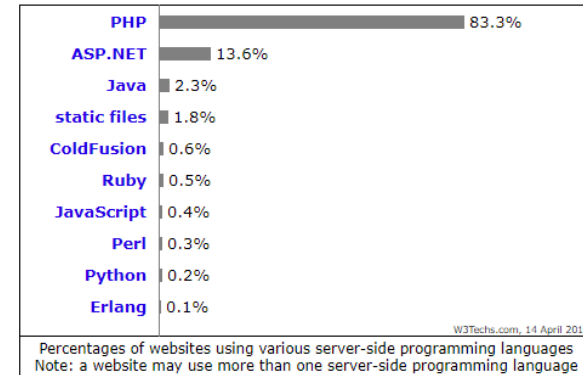
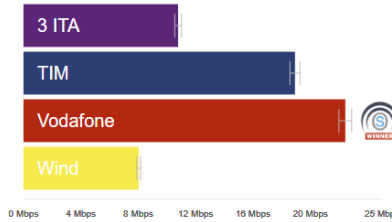
Architettura



Diffusione

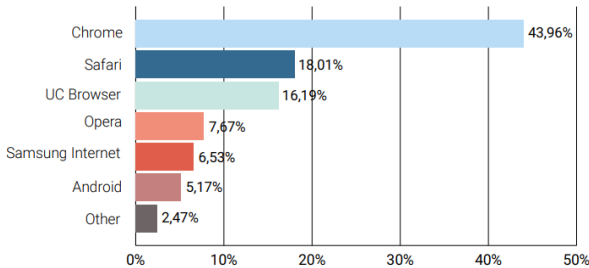


Download Speed: Overall OpenSignal

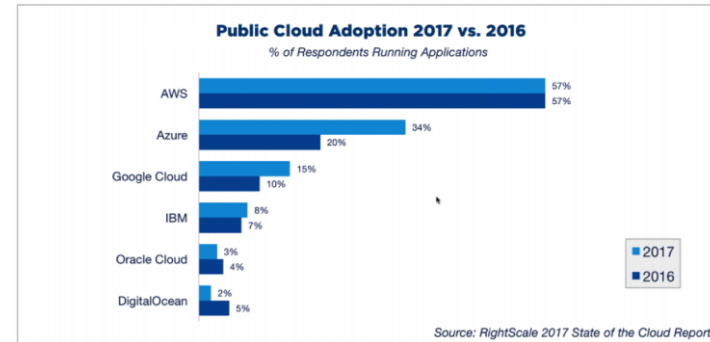


Platform	Market Share Q1 2015	Market Share Q1 2016	Market Share Q1 2017
Android	78.9%	84.1%	86.1%
iOS (Apple)	17.9%	14.8%	13.7%
Other	3.3%	1.1%	0.2%

(Source: gartner.com/newsroom/id/3725117 and gartner.com/newsroom/id/3061917)



Worldwide mobile browser market share June 2016- June 2017 (source: gs.statcounter.com/browser-market-share/mobile/worldwide/#monthly-201606-201706-ba)



Tipologie

Ogni app è un mix di tecnologie differenti



Tipologie

Nativa

Si basa su ambienti di sviluppo e SDK proprietari della piattaforma ed il codice non risulta portabile.

Ibrida

Si basa su ambienti di sviluppo e SDK scelti dallo sviluppatore ed il codice risulta facilmente portabile.

Web (PWA)

Si basa su ambienti di sviluppo e SDK web ed il codice è unico.

Tipologie

Nativa

Pro:

- Performante
- Accesso all'hardware
- GUI specifica
- Presente negli store
- API subito disponibili

Contro:

- Onerosa (per ogni piattaforma ho un SDK)
- Codice non portabile

Ibrida

Pro:

- Sviluppo veloce
- Abbastanza Performante
- GUI specifica in alcuni casi
- Codice quasi portabile
- Presente negli store

Contro:

- Accesso all'hardware limitato
- API non sempre disponibile

Web (PWA)

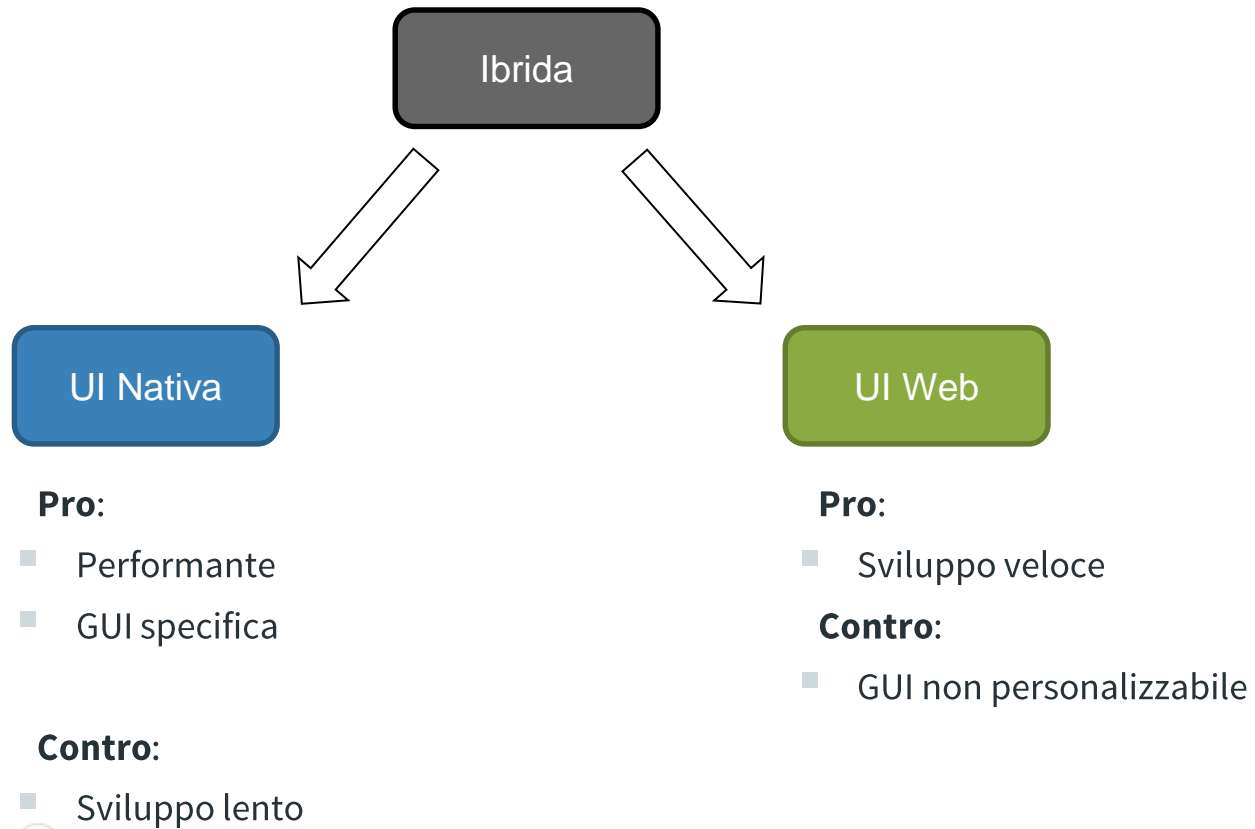
Pro:

- Sviluppo velocissimo
- Un solo codice

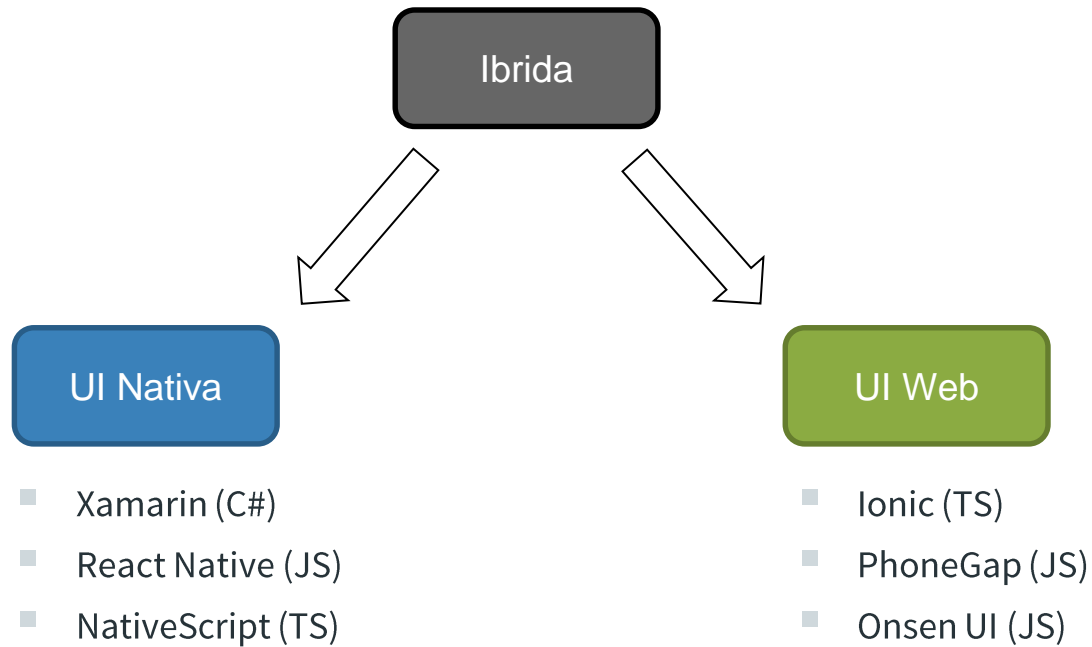
Contro:

- Poco performante
- GUI generica
- Accesso all'hardware limitato
- Non presente negli store

Tipologie ibride



Framework per sviluppo ibrido



Under the hood

AOT/JIT/Marshaling



Compilatore

AOT

ahead-of-time

La compilazione avviene una sola volta



Nativa

AOT

Ibrida

AOT/JIT

Web (PWA)

JIT

JIT

just-in-time

La compilazione avviene ad ogni avvio

- Android permette il JIT
- iOS non permette la compilazione JIT al di fuori della WKWebView

Compilatore

Javascript è un linguaggio debolmente tipizzato

<https://hacks.mozilla.org/2017/02/a-crash-course-in-just-in-time-jit-compilers/>



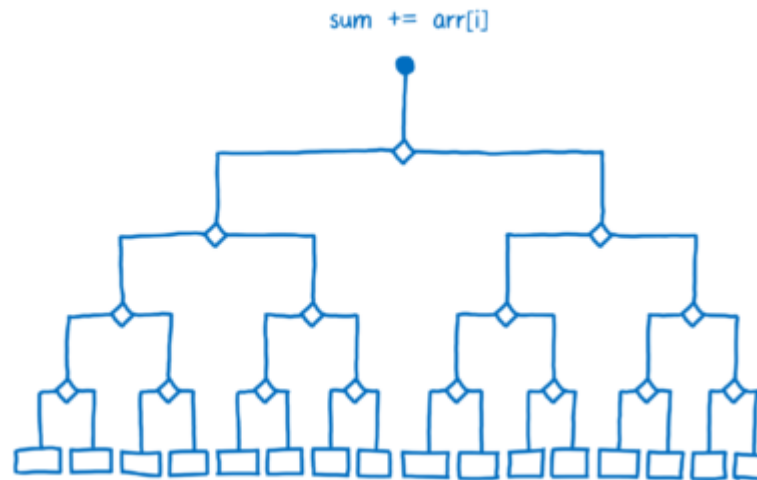
```
function arraySum(arr) {  
  var sum = 0;  
  for (var i = 0; i < arr.length; i++) {  
    sum += arr[i];  
  }  
}
```

is sum an int?

is arr an array?

is i an int?

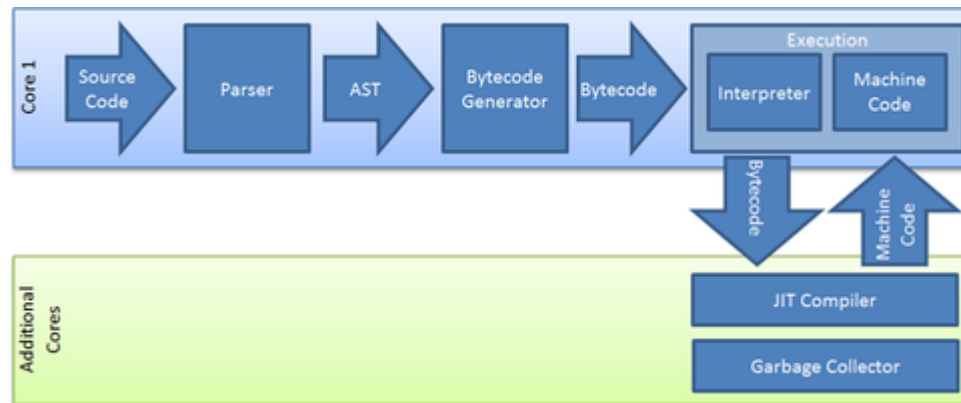
is arr[i] an int?



Compilatore



V8 Engine – Compilazione JIT di un JS





“

Ma come possiamo superare la mancanza di JIT in iOS?

- ▲ No. JavaScriptCore on iOS 7+ won't be able to JIT compile for you, because iOS disallows mapping writable/executable pages of memory as a hard rule, and that's a requirement for JIT. Only
- 4 MobileSafari.app, Web.app and a handful of other system apps carry an entitlement that allows them to JIT compile. The new WKWebView in iOS 8 is rendered in a separate process that is
- ▼ allowed to JIT compile, so JavaScript in a WKWebView is faster than a UIWebView or plain JSContext.



share edit flag

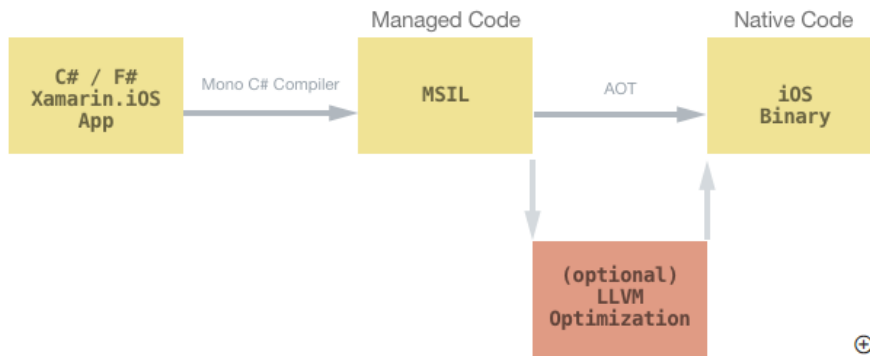
answered Jan 4 '15 at 20:52

Xamarin approach (C#)

AOT

When you compile any Xamarin platform application, the Mono C# (or F#) compiler will run and will compile your C# and F# code into Microsoft Intermediate Language (MSIL). If you are running a Xamarin.Android, a Xamarin.Mac application, or even a Xamarin.iOS application on the simulator, the [.NET Common Language Runtime \(CLR\)](#) compiles the MSIL using a Just in Time (JIT) compiler. At runtime this is compiled into a native code, which can run on the correct architecture for your application.

However, there is a security restriction on iOS, set by Apple, which disallows the execution of dynamically generated code on a device. To ensure that we adhere to these safety protocols, Xamarin.iOS instead uses an Ahead of Time (AOT) compiler to compile the managed code. This produces a native iOS binary, optionally optimized with LLVM for devices, that can be deployed on Apple's ARM-based processor. A rough diagram of how this fits together is illustrated below:

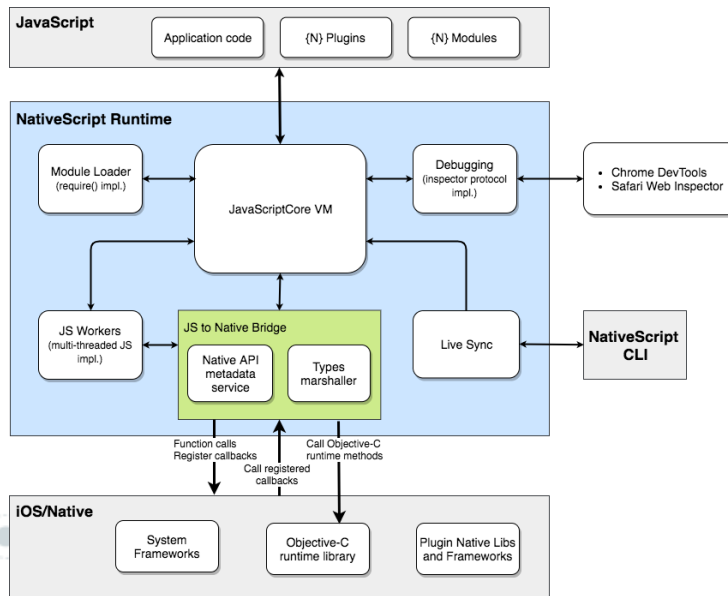


<https://docs.microsoft.com/en-us/xamarin/ios/internals/architecture>

Come superare il limite di iOS ed usare JS?

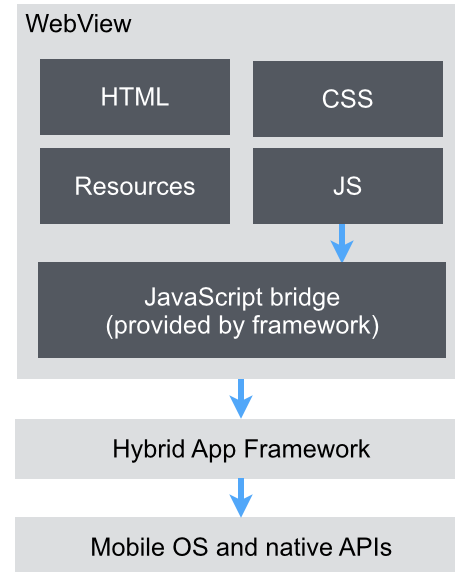
Interprete

L'applicazione viene interpretata a runtime o pre compilata



WebView

L'applicazione vive in un browser contenuto nell'app



Come superare il limite di iOS ed usare JS?

Interprete

L'applicazione viene interpretata o pre compilata

- Approccio complesso
- Performante
- UI Nativa
- Limiti nella compilazione
- Accesso hardware diretto
- Marshalling

WebView

L'applicazione vive in un browser embedded

- Approccio semplice
- Lenta
- UI Web
- Nessun limite (JIT presente)
- Limiti nell'accesso hardware

Marshalling

<https://docs.nativescript.org/runtimes/android/marshalling/overview>
<https://docs.nativescript.org/runtimes/ios/marshalling/Marshalling-Overview>

Cross-platform API

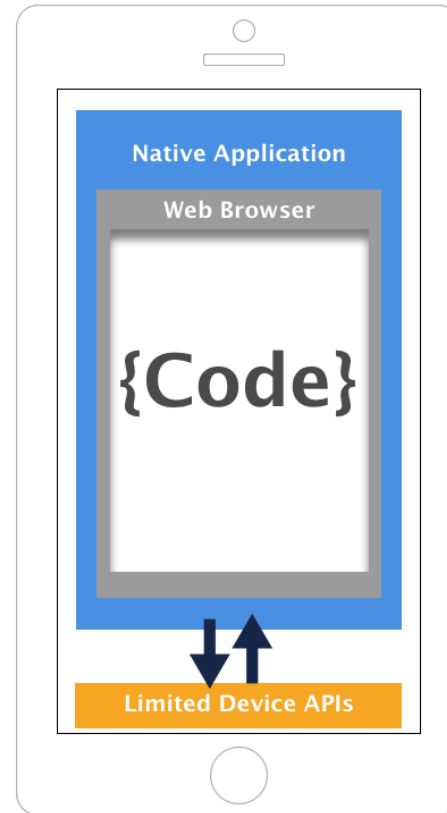
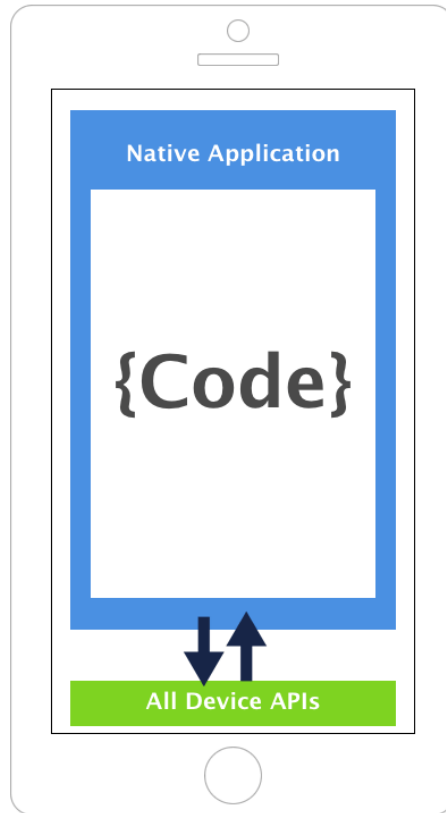
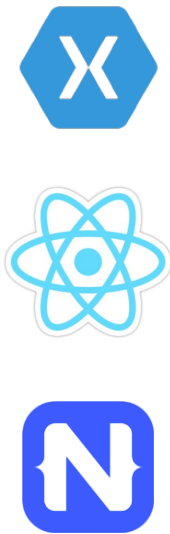


Differenze sostanziali

<https://www.nativescript.org/blog/nativescript-and-xamarin>

Truly Native (Xamarin, NativeScript)

Hybrid Apps (Cordova, PhoneGap)

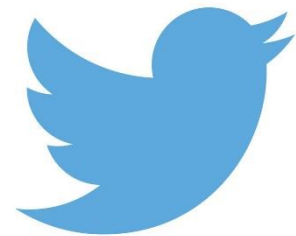




17TH EDITION
DOWNLOAD
RIGHT NOW

Medium

Newsletter 





<https://github.com/denysdovhan/wtfjs>

