

Aplicaciones

Híbridas

para

Smartphones

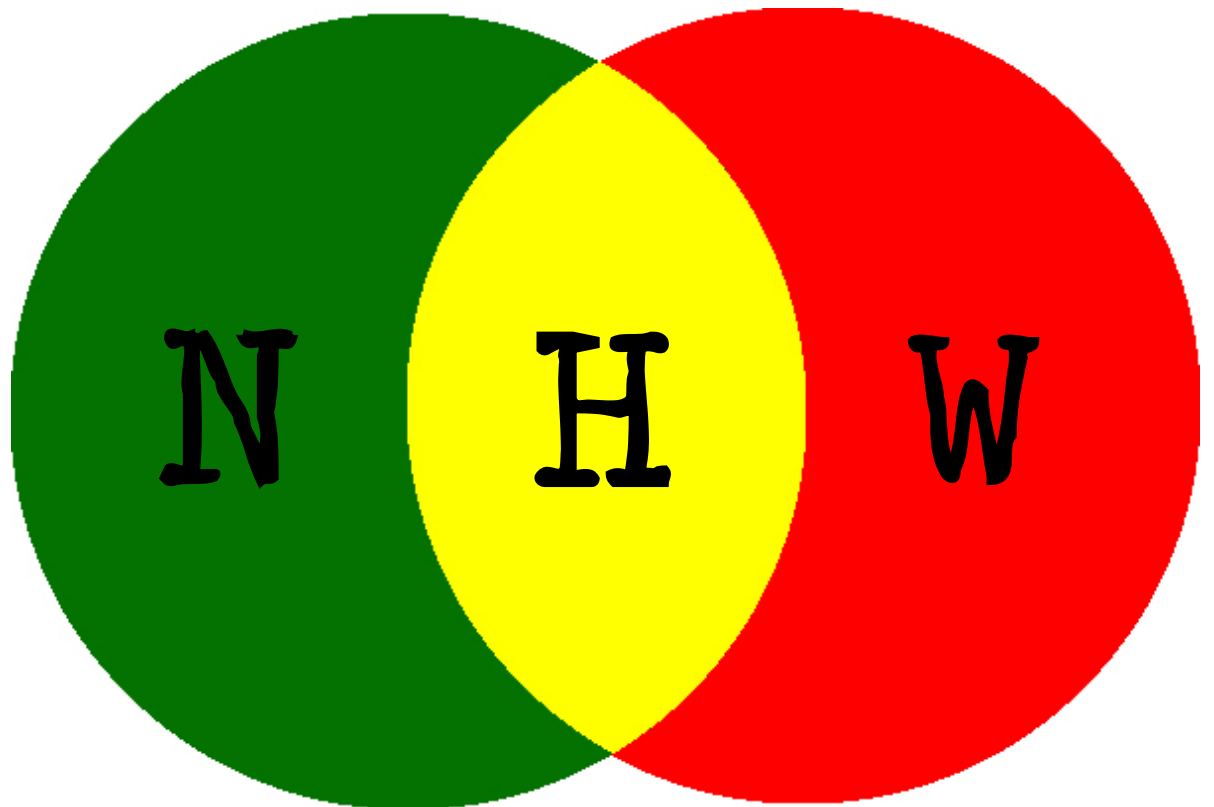
Ing. Federico Kereki



¿Qué son?

Tres tipos de apps

- Nativas
- Web
- Híbridas



Apps Nativas

- Basadas en el software del smartphone
- Específicas para una plataforma
- Diversos lenguajes



symbian

Apps Web

- Corren en un browser
- Dependenden de Internet
- Programadas en HTML+JavaScript



Apps Híbridas

- Aplicaciones web..
 - + browser nativo
 - + capa de intermediación
 - "look" de browser



Problemas comunes

- Hardware
- Capacidad
- Plataformas
- Estándares (?)
- "BYOD"



¿Cuál?

¿Por qué sí nativa?

- Presencia, visibilidad
- Aspecto y uso (UI & UX)
- Performance
- Almacenamiento
- Funcionalidad



¿Por qué no nativa?

- Plataformas
- Reciclaje de Conocimiento
- Actualización
- Costos
- Seguridad



Objective-C



¿Por qué sí web?

- HTML5 + CSS3 + JS
- RRHH, \$\$\$, tiempos
- Reuso de sitios
- Ubicuidad (+/-)
- Actualización



¿Por qué no web?

- "Estándares"
- UI & UX
- Trabajo offline
- "One Page App" (AJAX)
- Performance
- APIs y Hardware



¡Sí a las híbridas!

- Visibilidad
- Desarrollo (+/-) único
- Funcionamiento Offline
- Lucen (+/-) nativas
- Mix: residente + remoto
- Lealtad vs \$\$\$



¿No a las híbridas?

- Funcionalidades nativas (+/-)
- Experiencia y expectativas
- Actualizaciones
- La trampa de Java
- El "riesgo Apple"
- El "caso FaceBook"



facebook

El caso Facebook

- Zuckerberg, 2012:
 - "El mayor error que hicimos como empresa, fue apostar mucho a HTML5 en oposición a nativas."
- pero no fue tan así...



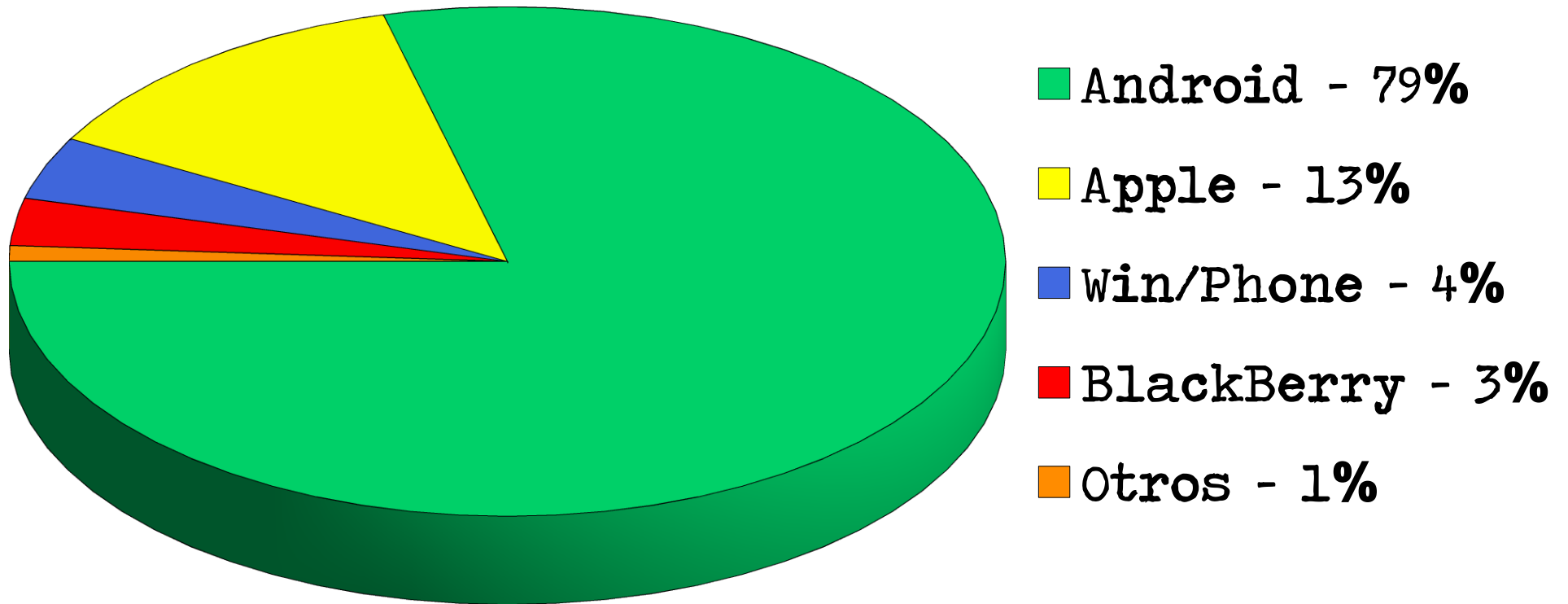
¿Cuándo?

Consideraciones

- ¿App interna/externa?
- ¿App exclusiva/masiva?
- ¿Una/toda plataforma?
- ¿Dispositivos únicos?
- ¿B2E, B2B, B2C, ...?



Mercado



Tendencias

- Estimación: costos 150-200% mayores para nativas
- Hoy: 80% nativas, 5% híbridas, 15% web
- 2016: 50% híbridas (Gartner)



¿Entonces?

Simbólicamente...

$N \succ H \ \& \ H \succ W \ \dashrightarrow \ N$

$N \dashrightarrow \$ \ \& \ t$

$\sim \$ \ \circ \ \sim t \ \dashrightarrow \ ?$

A favor / En contra

	W	H	N
Multipataforma	+	+	-
Fragmentación	0	0	-
API y Hardware	-	0	+
Offline	-	+	+
Performance	0	-	+
UI & UX	-	0	+
BYOD	+	+	-

A favor / En contra

	W	H	N
Tiempo (TTM)	+	0	-
\$\$\$	+	0	-
RRHH	+	0	-
Herramientas	+	0	-
Integración	+	+	0
Visibilidad	0	+	+
Actualización	+	-	-

¿Cómo?

En realidad...

1. Elegir herramientas
2. Programar estilo web
3. Probar y optimizar
4. Probar y optimizar
5. Probar y optimizar (ietc.!)
6. Distribuir

Herramientas

- jQuery Mobile
- Sencha Touch
- Dojox.mobile
- PhoneJS
- Incluso GWT!

Phone**JS**



Sencha

dōjō
toolkit



 **jQuery**
mobile

Soluciones

- PhoneGap
- IBM Worklight
- Appcelerator
Titanium
- Xamarin



Xamarin



titanium™

PhoneGap



Historia

- 2009: Creado por Nitobi
- 2011: Comprado por Adobe
- 2012: "Apache Cordova"
- 2012: PhoneGap Build



Consideraciones

- Desarrollo único, multiplataforma
- Especializaciones nativas
- Muy (!) dinámico
- ¿Performance?
- ¿Desarrollo?



TIZEN™
symbian

APIs

Accelerometer

Camera

Capture

Compass

Connection

Contacts

Device

Events

File

Geolocation

Globalization

InAppBrowser

Media

Notification

SplashScreen

Storage

¿Preguntas?



iGracias!

Ing. Federico Kereki
fkereki@gmail.com

