MOBA1: MOBILE APPLICATIONS 1

DEVICES AND PLATFORMS

OVERVIEW

- Introduction to MOBA1
- Mobile Devices
- Mobile Platforms
- Mobile Applications

OVERVIEW

- Introduction to MOBA1
- Mobile Devices
- Mobile Platforms
- Mobile Applications

TOPICS MOBA1

1. Introduction, Mobile Platform
2. Design & Development of Mobile Apps (1)
3. Design & Development of Mobile Apps (2)
4. Kotlin
5. Android (1)
6. Android (2)
7. Android (3)
8. Android (4)
9. Beacons

TOPICS MOBA1

10. Mobile Web: Overview, Basics
11. Mobile Web: Style & Layout
12. Mobile Web: Device APIs, Cordova/Capacitor
13. Presentations: Projects and Labs
14. Presentations: Projects and Labs

OUTLOOK: MOBA2 (PLANNED)

1. Mobile Web: Component driven UIs
2. Mobile Web: React.js
3. Mobile Web: Ionic/React
4. React Native (1): Basics, Navigation
5. React Native (2): Components and APIs
6. iOS Introduction
7. Swift
8. SwiftUI (1)
9. SwiftUI (2)
10. SwiftUI (3)
11. Presentations: Projects and Labs
12. Presentations: Projects and Labs

LESSONS AND SELF-STUDY

- Lecture: two lessons per week
- Practical exercises: also two lessons
- Self-study

PREVIOUS KNOWLEDGE

- HTML, CSS, JavaScript (WBE)
- Programming in C and Java

ASSESSMENT

Weight	Type
20%	Exercises and presentations during class time
80%	Final exam

EXERCISES AND PRESENTATIONS

- Small mobile project
- Exercises on most topics
- Presentation: project results

EXERCISES AND PRESENTATIONS

- Small mobile project / presentation (teams of 2)
  up to 5 points (project: 3, presentation: 2)
- Solved exercises / presentation
  up to 5 points

Points acquired during class time count for the semester finals (weighted, 20%)

If the practical results are inferior to the semester finals, only the latter count for the course grade

FINAL EXAM

- Mobile Project, 3h, on-site
- Open book, internet allowed except communication

Changes are possible depending on the Corona situation...

DOCUMENTS

In electronic form in Moodle:

- Slides
- Exercises
- Additional material

MOODLE

https://moodle.zhaw.ch/course/view.php?id=2155

- Weekly schedule
- Documents
- Course details and conditions
- Forum (?)

PRACTICAL LESSONS AND LECTURES

PRACTICAL LESSONS

PRACTICAL LESSONS

- Tasks explained
- You work with your notebooks
- Teamwork is appreciated

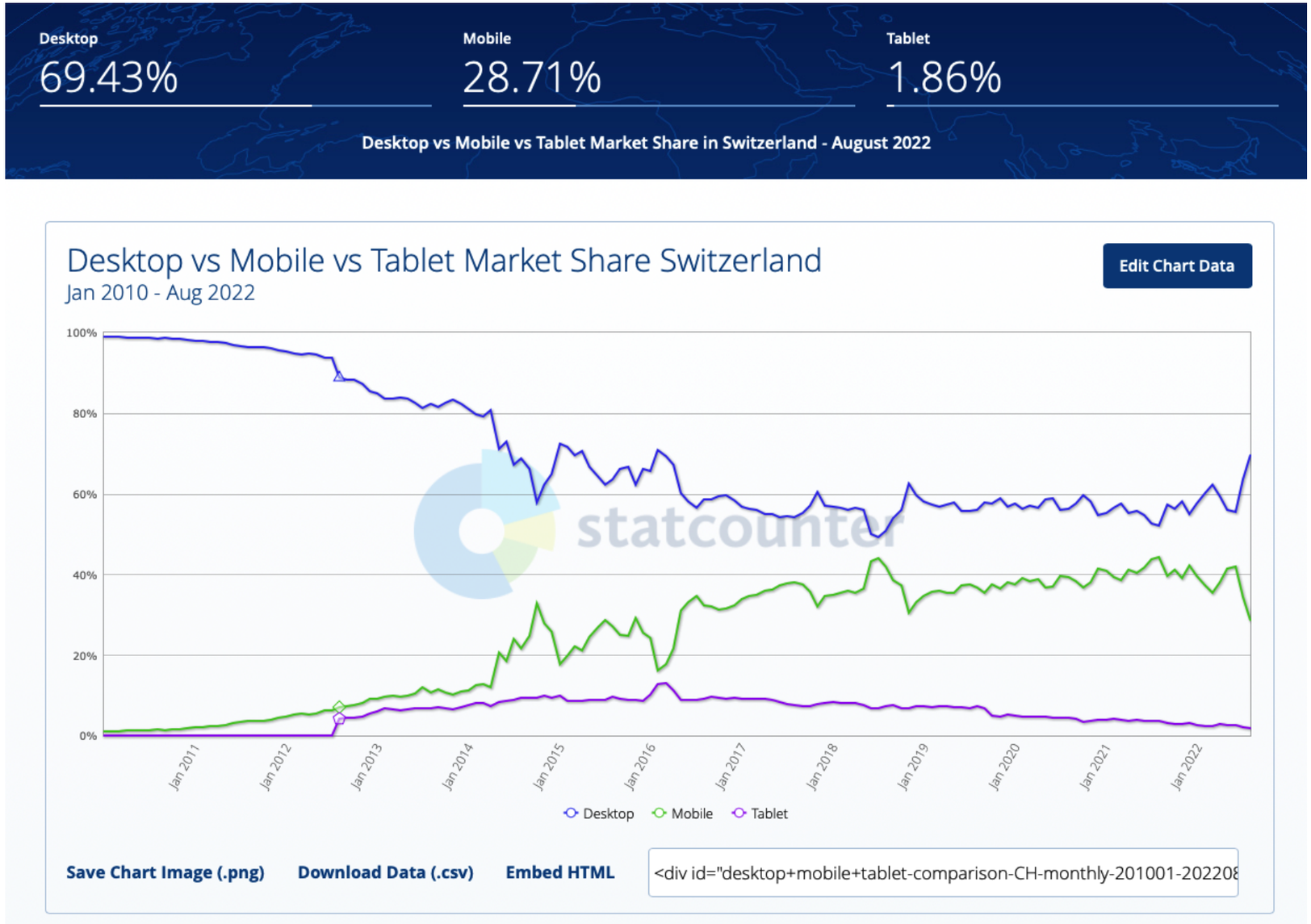
LECTURES

- Private conversations disturbing
- Interaction and participation appreciated
- Take notes

OVERVIEW

- Introduction to MOBA1
- Mobile Devices
- Mobile Platforms
- Mobile Applications

SMARTPHONE AND TABLET MARKET



FEATURE PHONE TO SMART PHONE

↓ more ↓

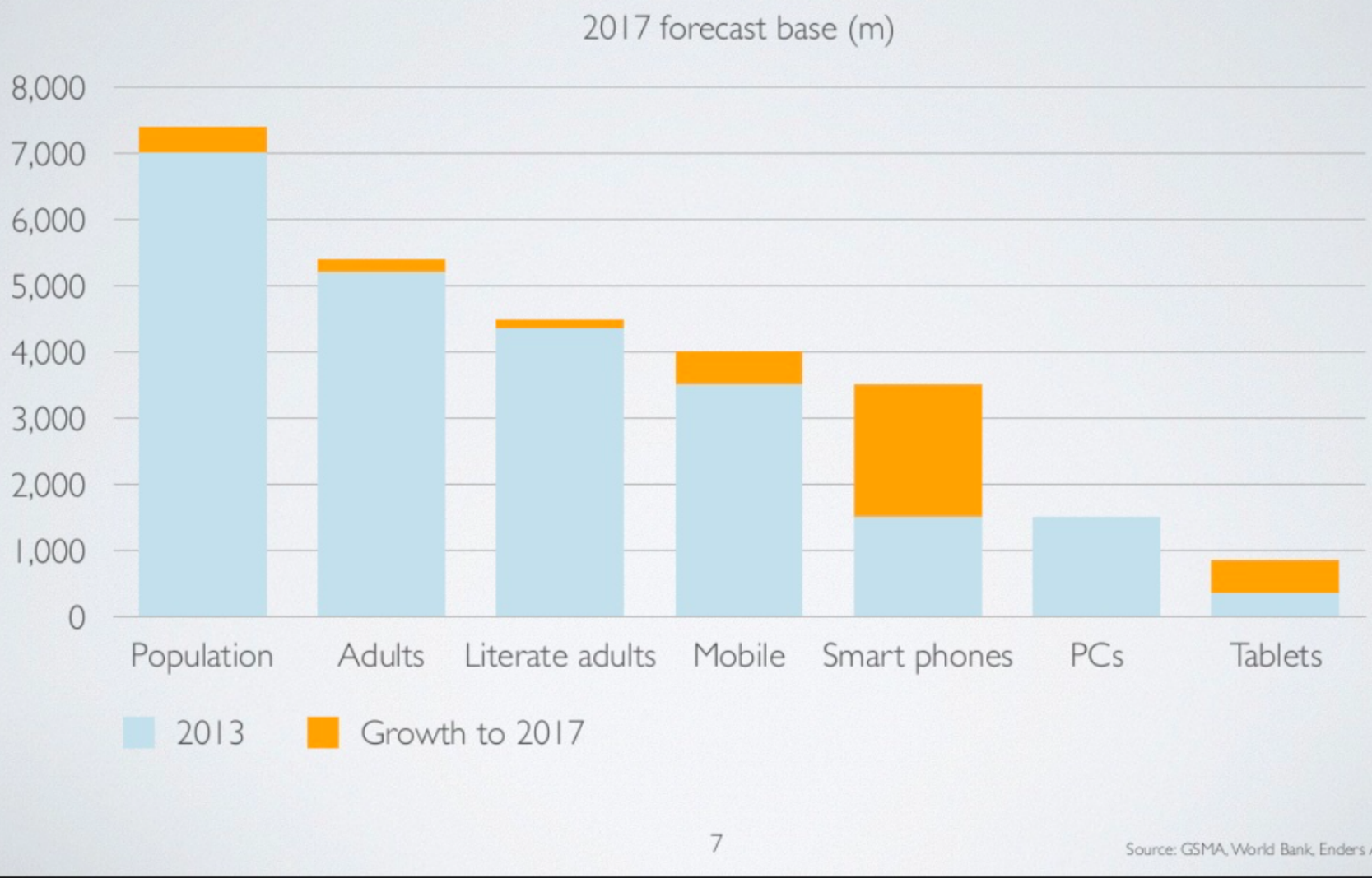
Quelle:
https://www.samsung.com

MOBILE DEVICES

MOBILE DEVICES ??

- Notebooks
- Netbooks
- Feature Phones
- Personal Digital Assistants
- Smartphones
- Tablets
- Smartwatches
- SmartTags, AirTags
- Calculator watches
- Handheld game consoles
- Head-mounted displays
- Wearable computers
- Portable media players
- Digital still cameras (DSC)
- Digital video cameras (DVC) or digital camcorders
- Personal navigation devices (PND)

The world in 2017



7

Source: GSMA, World Bank, Enders Analysis

MOBILE DEVICES IN MOBA 1/2

- In MOBA1/2 we focus on Smartphones
- With some comments on other devices
- Priority on devices with widespread mobile OSs

FEATURES OF MOBILE DEVICES

- Smaller screens (here, smartphones and tablets differ)
- Different input concepts (touch, keyboard, stylus)
- Slow, unstable network connection
- Less powerful processors
- Batteries – minimize power consumption

DEVICE SENSORS

- Camera
- Microphone
- Geolocation, GPS
- Acceleromter
- Gyroscope
- Magnetometer
- Battery state
- Proximity sensor

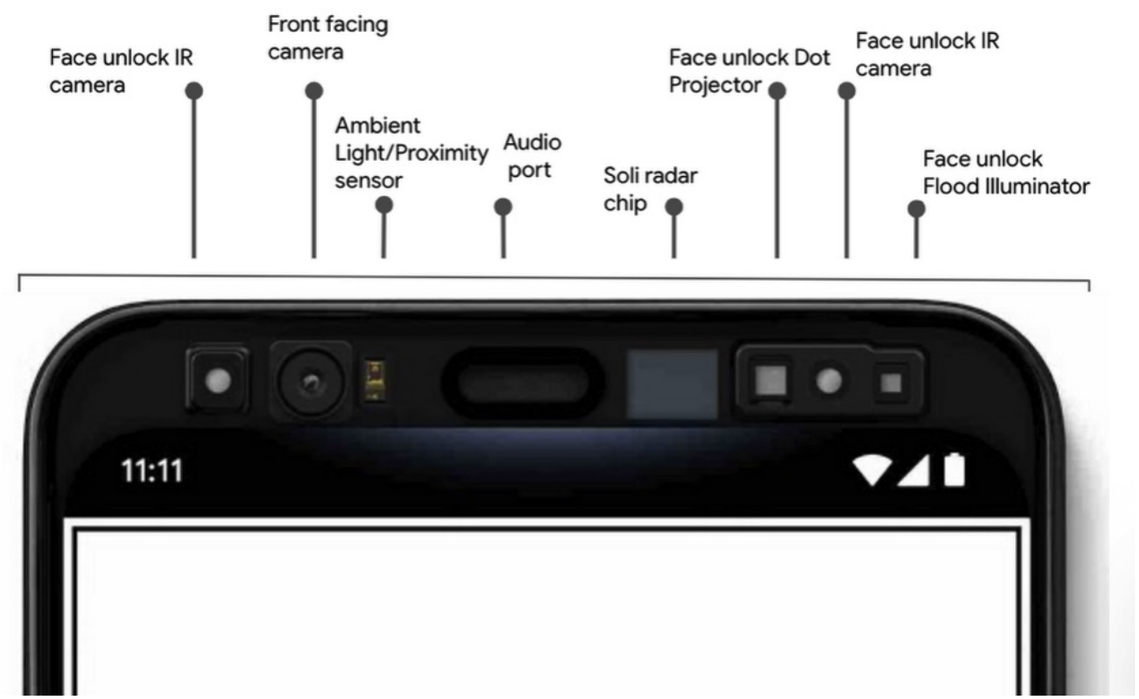
INSIDE IPHONE 13 PRO VS 12 PRO

Teardown

INSIDE SAMSUNG GALAXY S21 ULTRA

Teardown

GOOGLE PIXEL 4 SOLI RADAR



- Google’s Pixel 4 had built-in radar
- Pixel 5 and 6: radar sensor was ditched

PROBLEM: FRONT CAMERA, SENSORS

- Google Pixel 6a
- Fingerprint unlock
- 8 MP front camera

DYNAMIC ISLAND (APPLE)

- iPhone 14 Pro
- Front camera, Face ID sensors integrated in UI

OVERVIEW

- Introduction to MOBA1
- Mobile Devices
- Mobile Platforms
- Mobile Applications

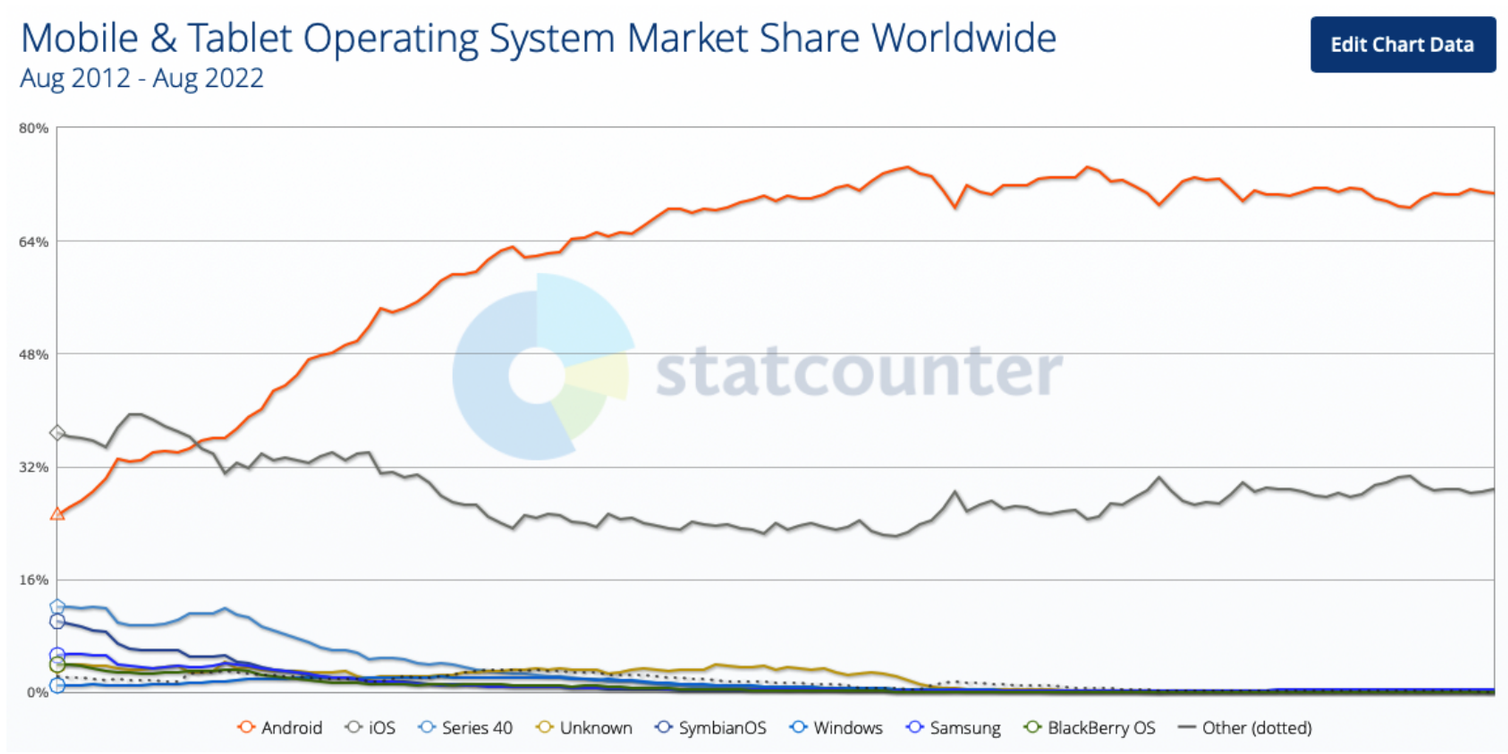
Mobile platform wars over?

- Apple and Google's Android have won: unlikely that other platforms will be relevant
- Apple camped out at the high end, Android taking the rest
- But Google's control of Android is partial
- Facebook and Amazon trying to extract value
- Samsung's position uncertain

41

ANDROID AND IOS

- Dominate the smartphone market
- Also at the top in terms of developer mindshare
- But: Mobile space changes continuously



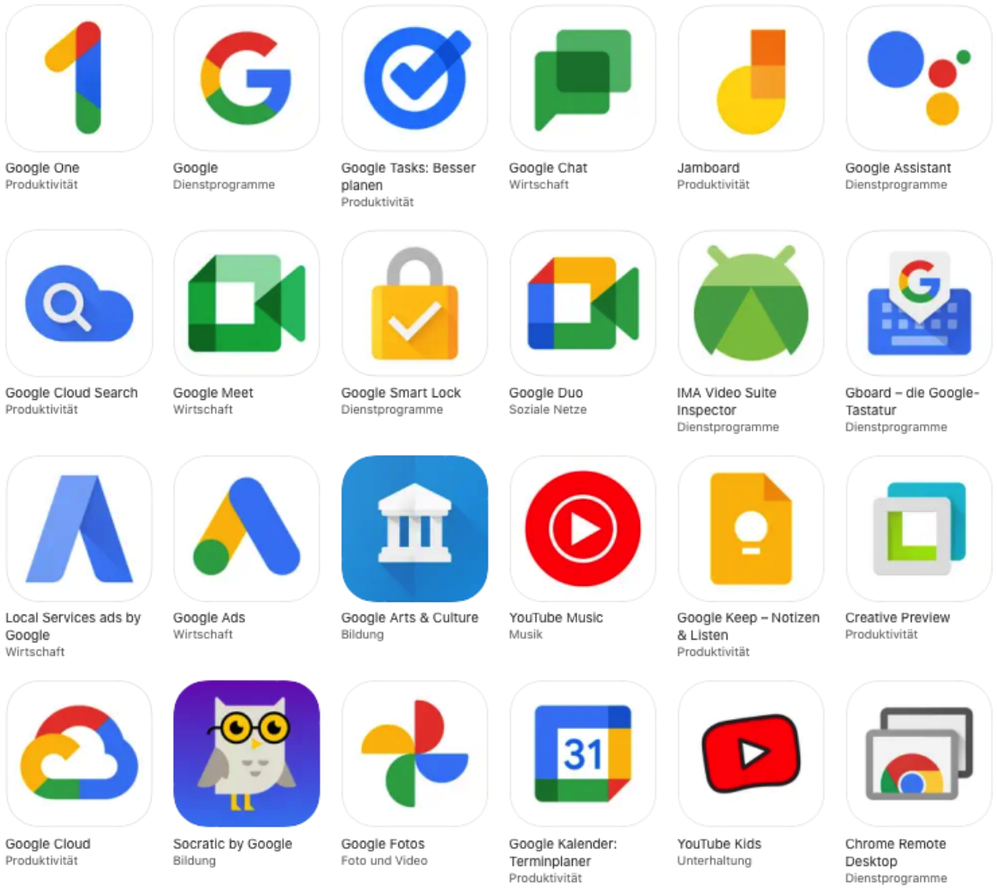
Source:
statcounter

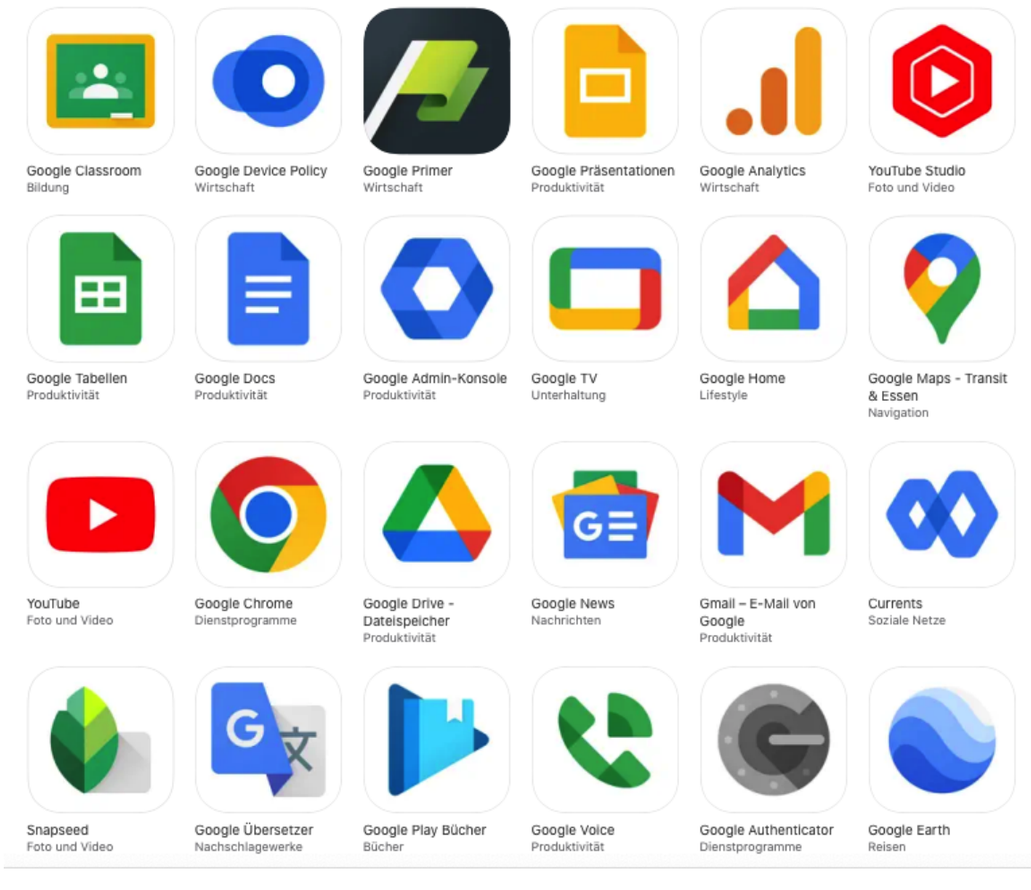
Different focus for innovation

Apple	Google
• Integrated hardware & software	• Systemic fragmentation, little hardware control
• Fingerprints, Bluetooth, Airdrop, 64 bit etc	• Google Now, Maps, Plus, semantic search etc
• Move innovation down the stack (hard for Google)	• Move innovation up the stack (hard for Apple)
• Directed discovery (iBeacon)	• Predictive discovery (Now)

22

GOOGLE APPS FOR IOS

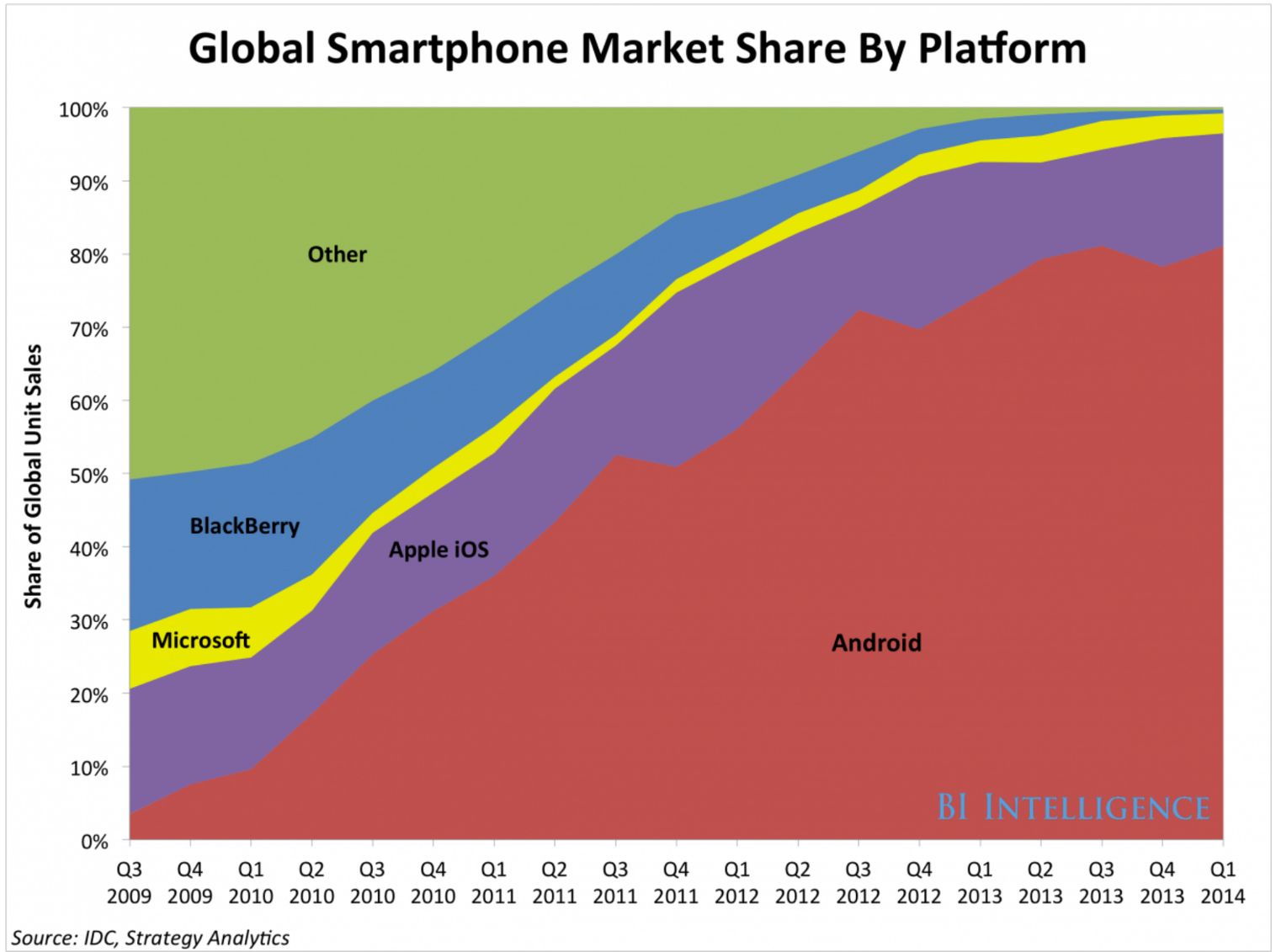




ANDROID'S PURPOSE

- Spreading the use of Google services across the mobile web
- Makes economic sense to offer Android for free
- In 2009 device vendors needed an answer to the iPhone
  - Android was available (and free...)
  - Most device vendors jumped on the Android bandwagon
  - Except for Apple, Nokia, and BlackBerry
  - Nokia and BlackBerry failed

↓ more ↓



Source: IDC, Strategy Analytics

ANDROID DIFFERENTIATION

- Google allowed Android differentiation
- Purpose: Get device vendors to adopt it
- Vendors could create their own interface layer
- They could also change default apps, including the browser
- Samsung TouchWiz, HTC Sense, MotoBlur, ...
- Differentiation is the positive slant on fragmentation

IOS: HISTORY (1)

- iPhone unveiled by Steve Jobs January 9th 2007
- OS named iPhone OS
- November 2010: launch of the fourth generation of the OS
  - renamed to iOS
  - launch of the original iPad
- New version each year

And Then Steve Said, ‘Let There Be an iPhone’

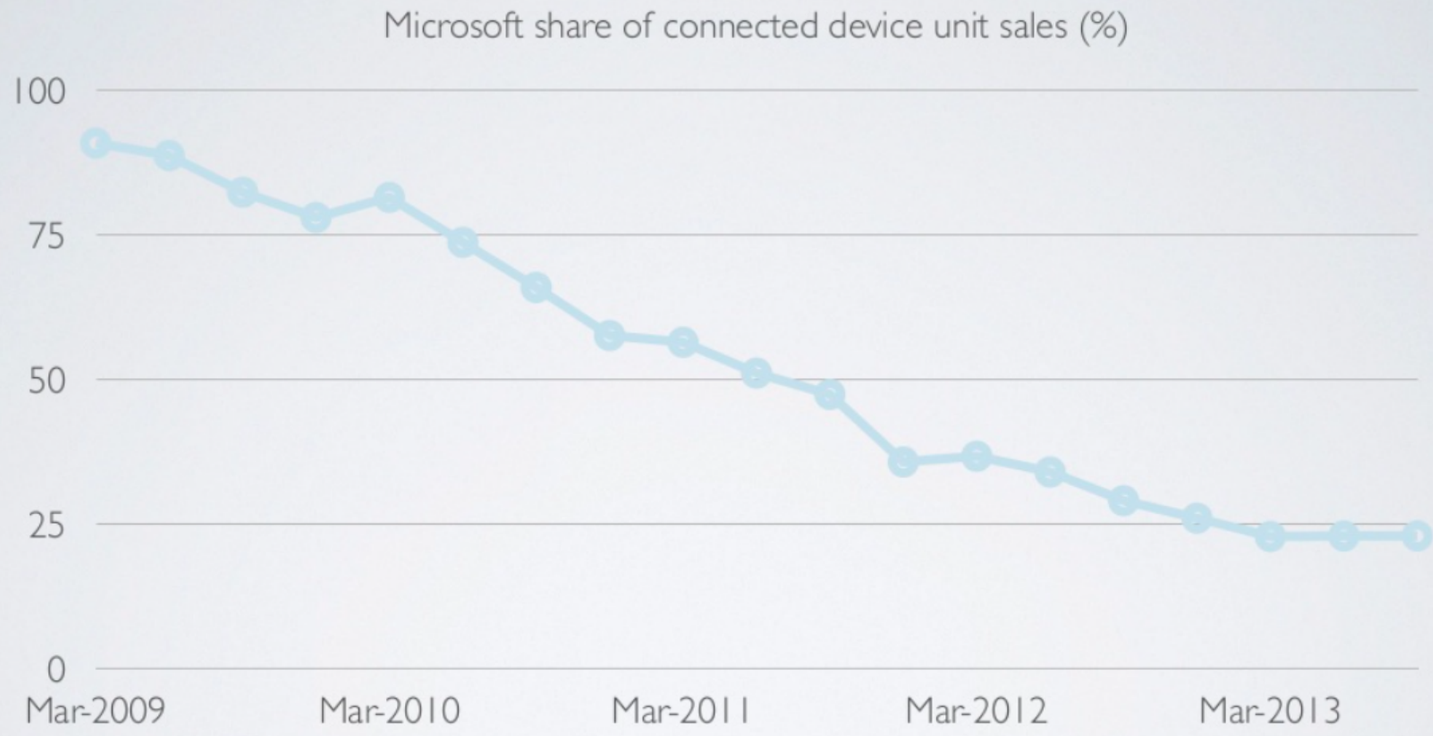
IOS: HISTORY (2)

- iOS 7 was touted as a major UI refresh
- iOS 8 with new frameworks and services, Apple Watch
- iOS 9 with split view on iPad, better Siri, faster graphics
- iOS 10 with haptic feedback, better Siri, better search
- iOS 11...16 with ?? (various improvements)
  - iOS 14 with Widgets on Home Screen
  - iOS 16: lock screen customization

→ Release notes for developers

High adoption rate of each iOS version soon after release

The irrelevance of Microsoft



16

Source: Enders Analysis

MARKET SHARE

Period	Android	iOS	Windows	BlackBerry	Others
2018	85.1%	14.9%	0.0%	0.0%	0.0%
2017	85.0%	14.7%	0.1%	0.0%	0.1%
2016	83.4%	15.4%	0.8%	0.0%	0.4%
2015	78.0%	18.3%	2.7%	0.3%	0.7%
2014	81.2%	15.2%	2.5%	0.5%	0.7%
2013	75.5%	16.9%	3.2%	2.9%	1.5%
2012	59.2%	22.9%	2.0%	6.3%	9.5%

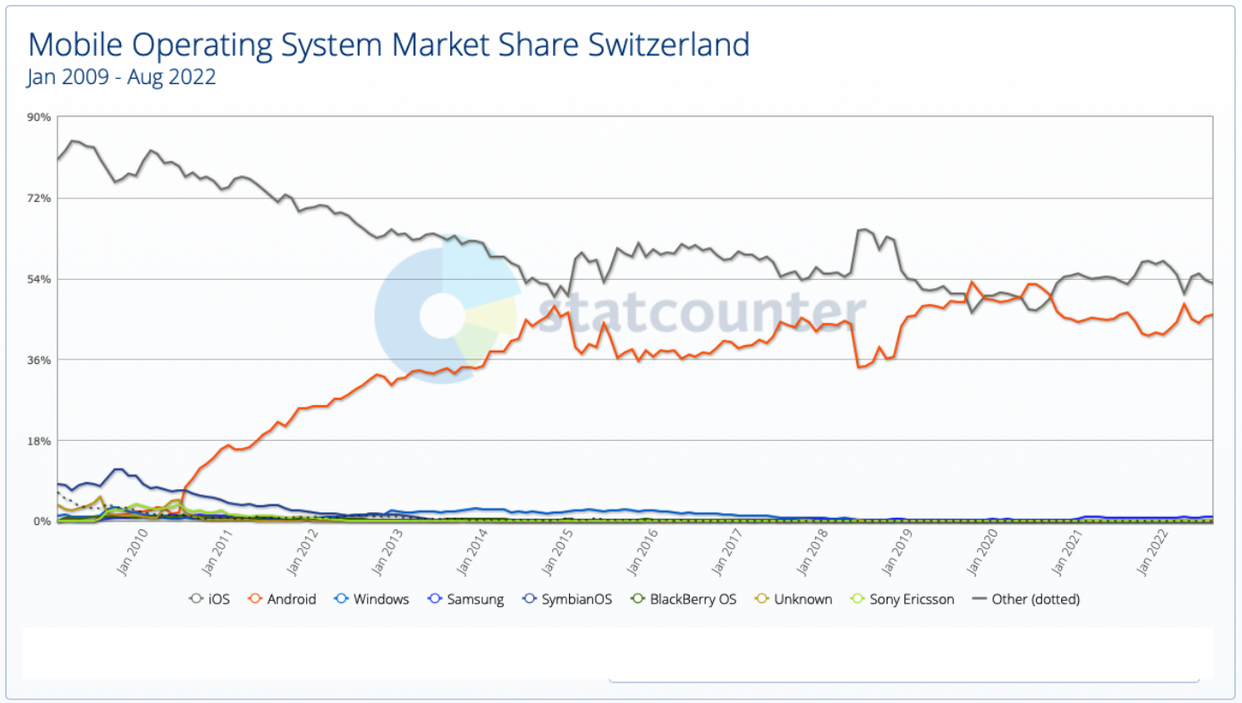
(Source: IDC)

THE UNDERDOGS

Discontinued or irrelevant

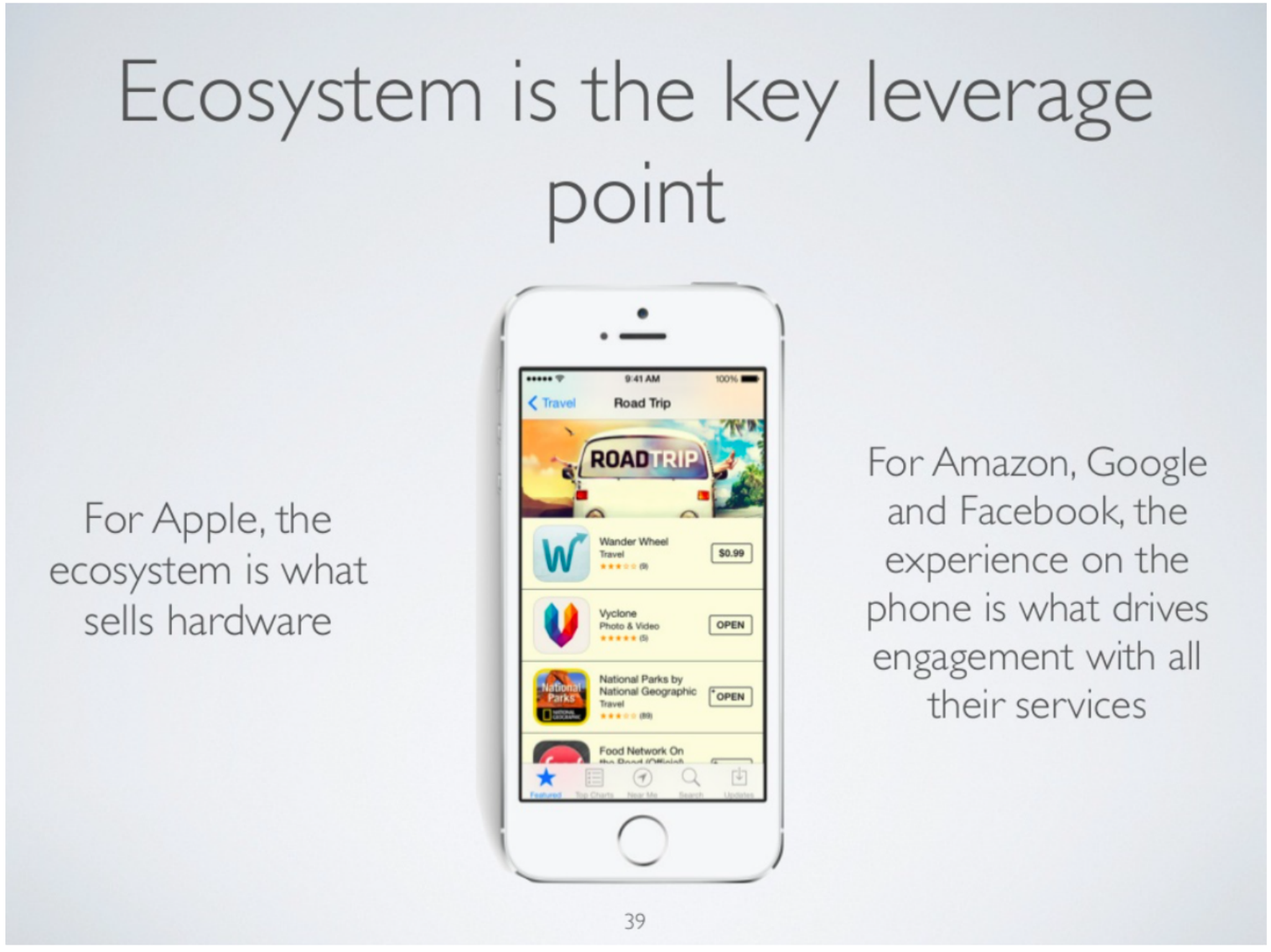
- Firefox OS
- BlackBerry OS
- Ubuntu Phone
- Tizen
- Sailfish OS
- …

ECOSYSTEM SIZES

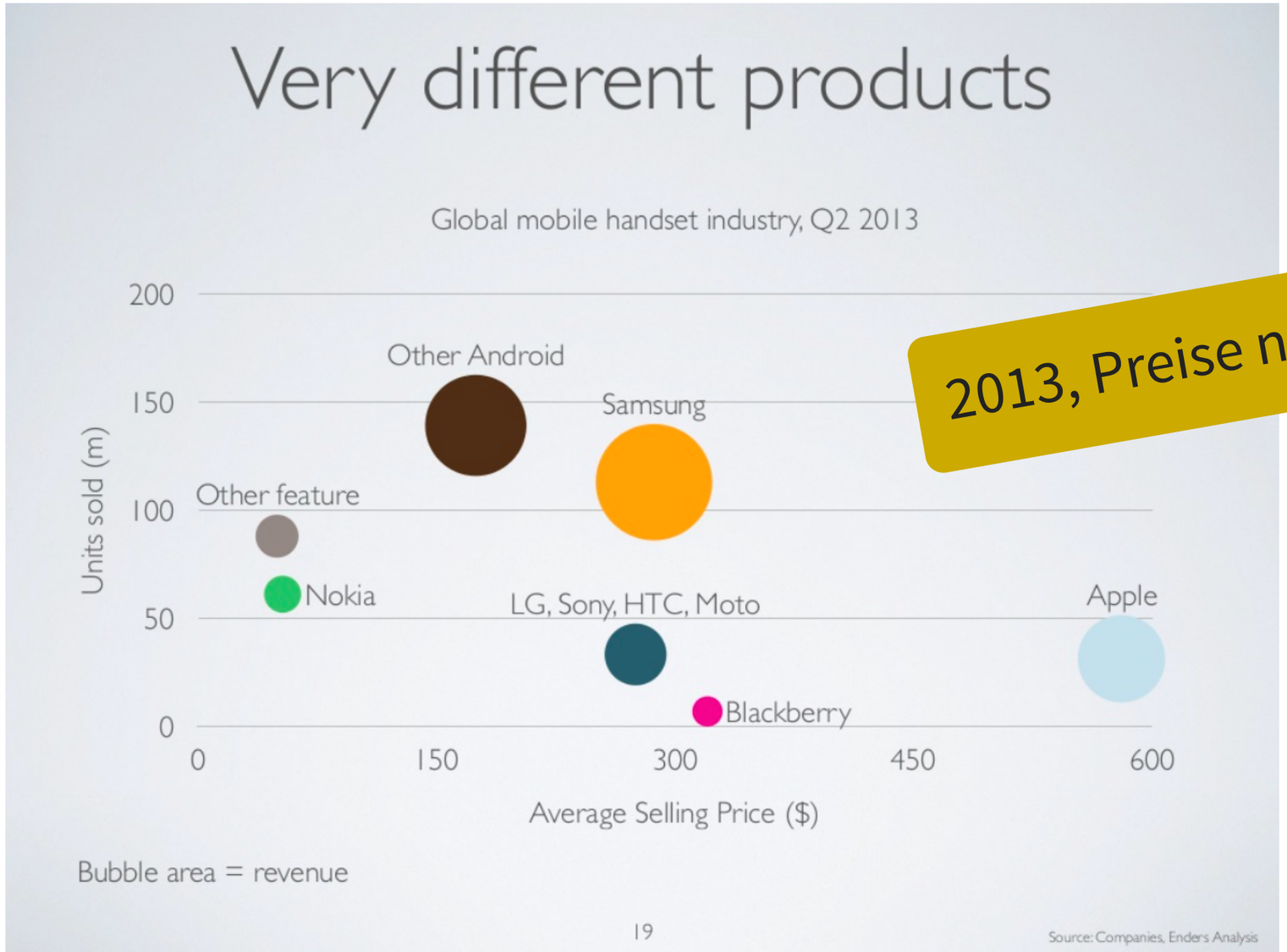


- Regional differences
- Sources
  StatCounter
  kantarworldpanel
  netmarketshare

ECOSYSTEM DIFFERENCES



PRODUCT CATEGORIES



2013, Preise nicht mehr ganz aktuell :)

OVERVIEW

- Introduction to MOBA1
- Mobile Devices
- Mobile Platforms
- Mobile Applications

MOBILE APPLICATIONS

- Traditionally we targeted phones
- Then tablets followed
- Today we have a complete range of device types
  - smartwatch, phone, tablet, TV and automotive
  - Each one poses its own usability challenges

MOBILE APPLICATIONS

- Native
- Web-based
- Hybrid

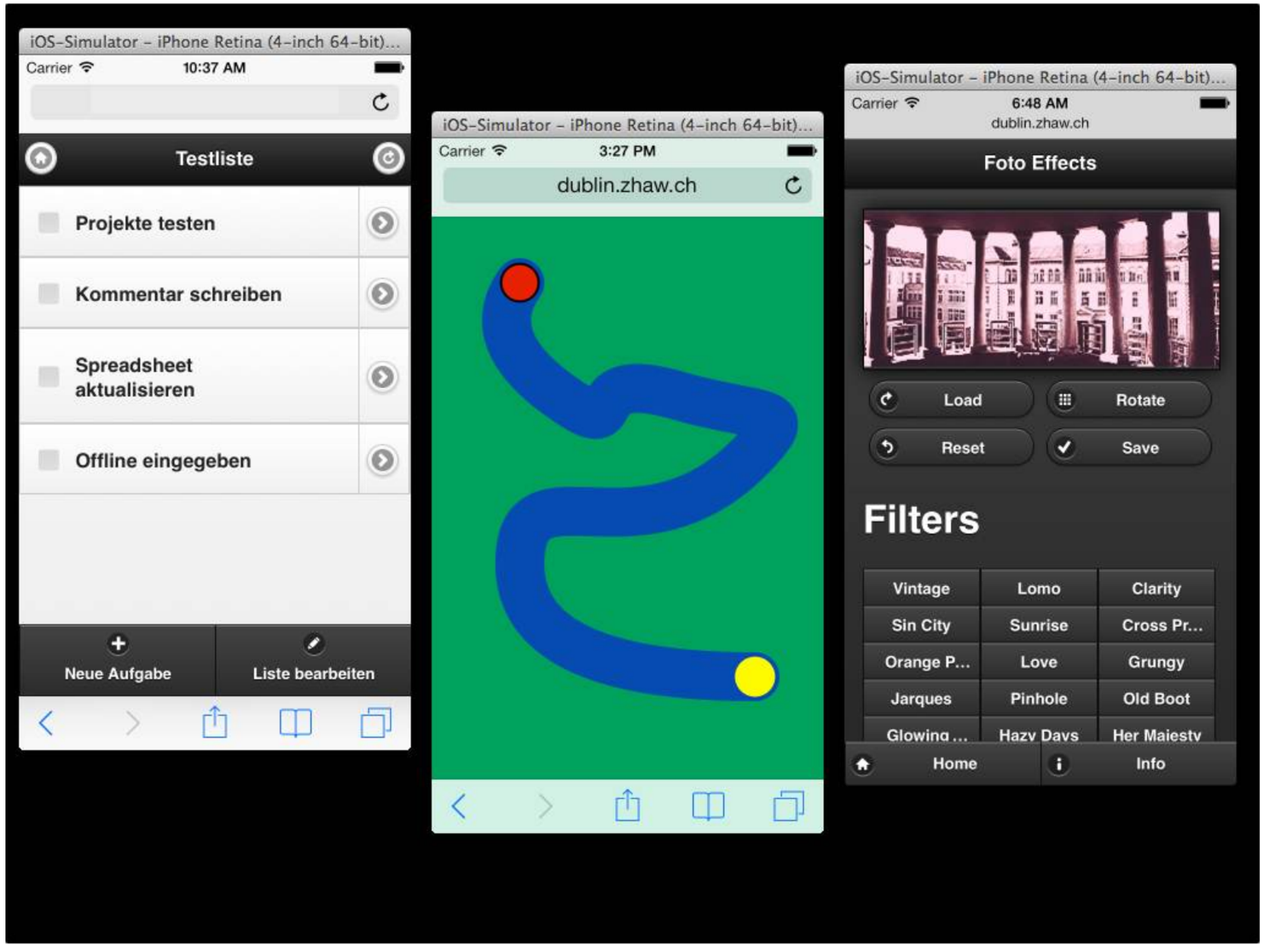
NATIVE MOBILE APPLICATIONS

- Platform specific language
- Platform specific APIs
- Platform specific central app store
- Advantages
  - Usually offer the best performance
  - Deepest integration
  - Best overall user experience
- Disadvantage
  - Most complex development option

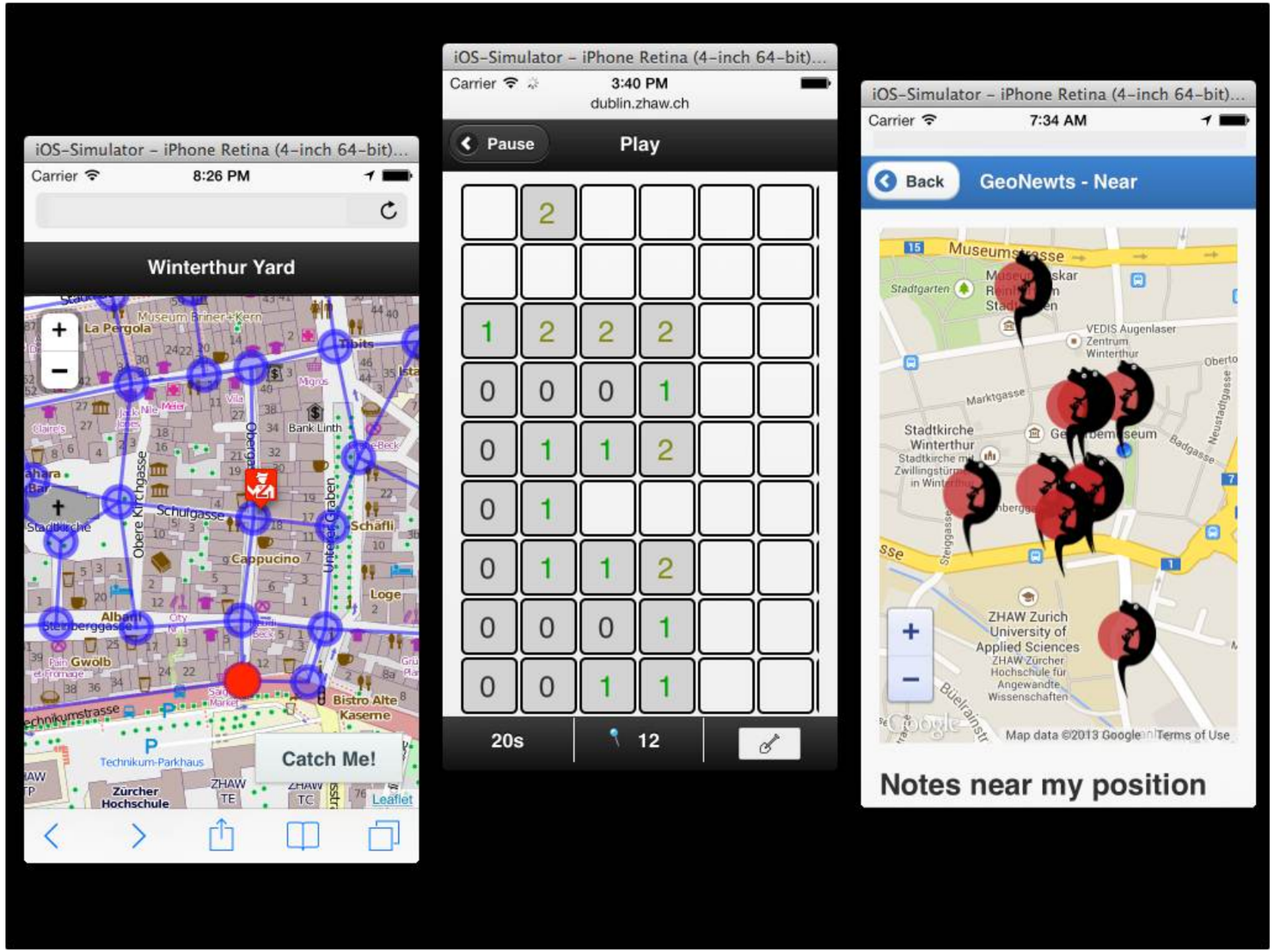
WEB-BASED MOBILE APPLICATIONS

- Based on HTML, JavaScript and CSS
- Do not rely on an app store
- Essentially locally stored mobile sites
- Try to emulate the look-and-feel of an app
- Can be added to the home screen

SAMPLE WBE PROJECTS



SAMPLE WBE PROJECTS



HYBRID MOBILE APPLICATIONS

- Frameworks can build a native wrapper around web apps (examples: Cordova, Capacitor)
- Use native code for enhanced performance and integration
- Use a webview with HTML-based content for other parts
- Allow to revise content and features without using the app stores

SOURCES

- Slides and other material from courses WEB1, WBE
- Mobile Developer’s Guide To The Galaxy, 18th Edition, Open XChange, https://www.open-xchange.com/resources/mobile-developers-guide-to-the-galaxy/
- Mobile is eating the world, Benedict Evans, 2013, http://de.slideshare.net/bge20/2013-11-mobile-eating-the-world