



## **IN5320 - Development in Platform Ecosystems**

### *Lecture 7: Platform Ecosystems fundamental concepts*

1st of October 2018

Department of Informatics, University of Oslo

Magnus Li - [magl@ifi.uio.no](mailto:magl@ifi.uio.no)

# Assignment 2

## Assignments

- Mandatory assignment 1
- Mandatory assignment 2
- Guides and resources for assignment 2
- Theoretical assignments
- Devilry (to deliver assignments)



## Resources for Assignment 2

### Content

- Open APIs
- Using React
- Git Guide
- Git with SSH
- CORS Cross Origin

### Open APIs

Here is a list of lists of open APIs that might be useful when solving assignment 2.

- <https://github.com/toddmotto/public-apis>
- <https://any-api.com/>

### Using React

# Group project

Start to form groups now!

[You must register your group here](#) **before 5th of October.**

Cases will be presented on the lecture October 8th (next week)

The week after, you will present initial requirements.

# Today's lecture

1. Platforms and Platform Ecosystem architectures
2. Boundary resources
3. Core characteristics and concepts (Tiwana 2013)
4. Drivers toward platform ecosystems (Tiwana 2013)

# Platform ecosystems

# What is a platform?



coinbase



# What is a platform?

- 1 A raised level surface on which people or things can stand.  
*'there are viewing platforms where visitors may gape at the chasm'*
- 1.2 A raised structure along the side of a railway track where passengers get on and off trains at a station.
- 2 *[usually in singular]* The declared policy of a political party or group.  
*'seeking election on a platform of low taxes'*
- 1.6 A standard for the hardware of a computer system, which determines what kinds of software it can run.

# What is a platform?

Gillespie, 2010 p349:

*“To call one’s online service a platform is not a meaningless claim, nor is it a simple one. Like other structural metaphors (think ‘network’, ‘broadcast’ or ‘channel’) the term depends on a semantic richness that, though it may go unnoticed by the casual listener or even the speaker, gives the term discursive resonance.”*



# Gillespie (2010) : Relevant definitions of “platforms”

**Computational:** “an infrastructure that supports the design and use of particular applications, be they computer hardware, operating systems, gaming devices, mobile devices or digital disc formats.” ibid p349

**Architectural:** “A raised level surface on which people or things can stand, usually a discrete structure intended for a particular activity or operation” (OED)

**Figurative:** “the ground, foundation, or basis of an action, event, calculation, condition, etc. Now also: a position achieved or situation brought about which forms the basis for further achievement” (OED)

**Political:** “The declared policy of a political party or group”, and “An opportunity to voice one's views or initiate action.”

# Gillespie (2010) : Relevant definitions of “platforms”

From this:

- Surface designed for some type of activity
- Neutral in broad sense, less neutral when tied to some specific function  
“Social media platform”, “Advertisement platform”, “Commerce platform”,  
“Health platform”
- Tied to politics.  
What is made easier? Who has access? To what? Based on what? To what ends?

# What is a platform?

Baldwin and Woodard (2008): Platform **architectures** refer to systems that are **partitioned** into:

- 1) A set of stable components
- 2) A set of complementary components that vary

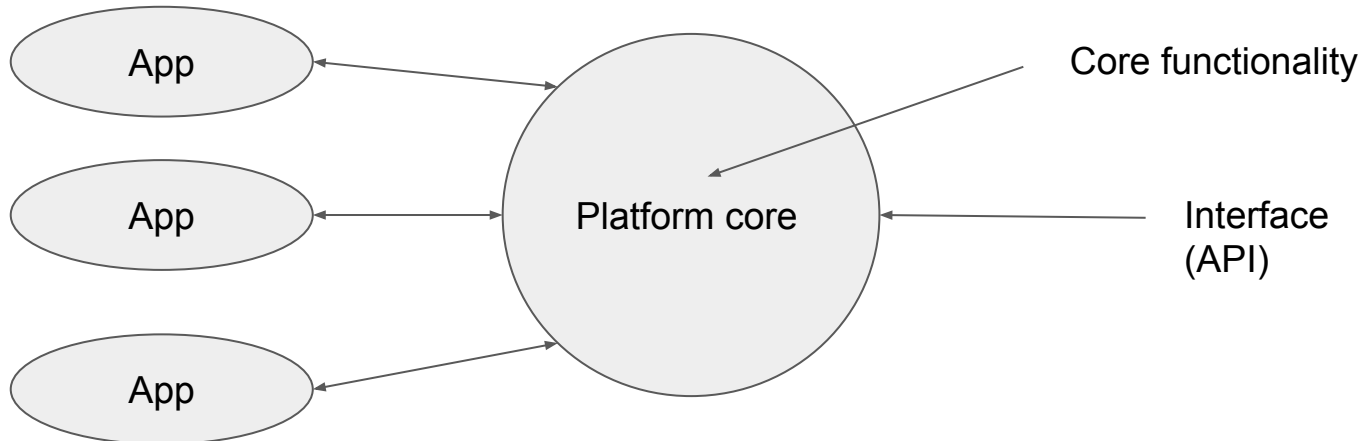
Between these are interfaces that enable interaction. These are part of the platform and should be stable over time.

*“The low-variety components constitute the platform. They are the long-lived elements of the system and thus imply or explicitly establish the system’s interfaces, the rules governing the interactions of the different parts”* (Baldwin and Woodard, 2008, p 19)

# What is a platform?

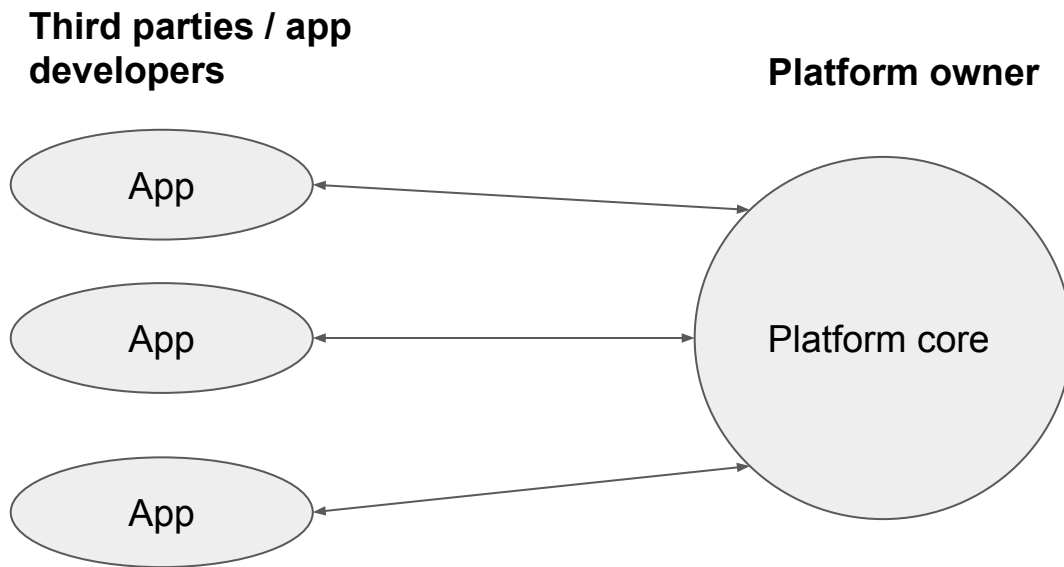
**“A software platform is a software-based product or service that serves as a foundation on which outside parties can build complementary products or services” - Tiwana 2014 p5**

- Provides core functionality which is extendable
- Entails interfaces that allows third parties to develop *apps* that extend the functionality of the platform



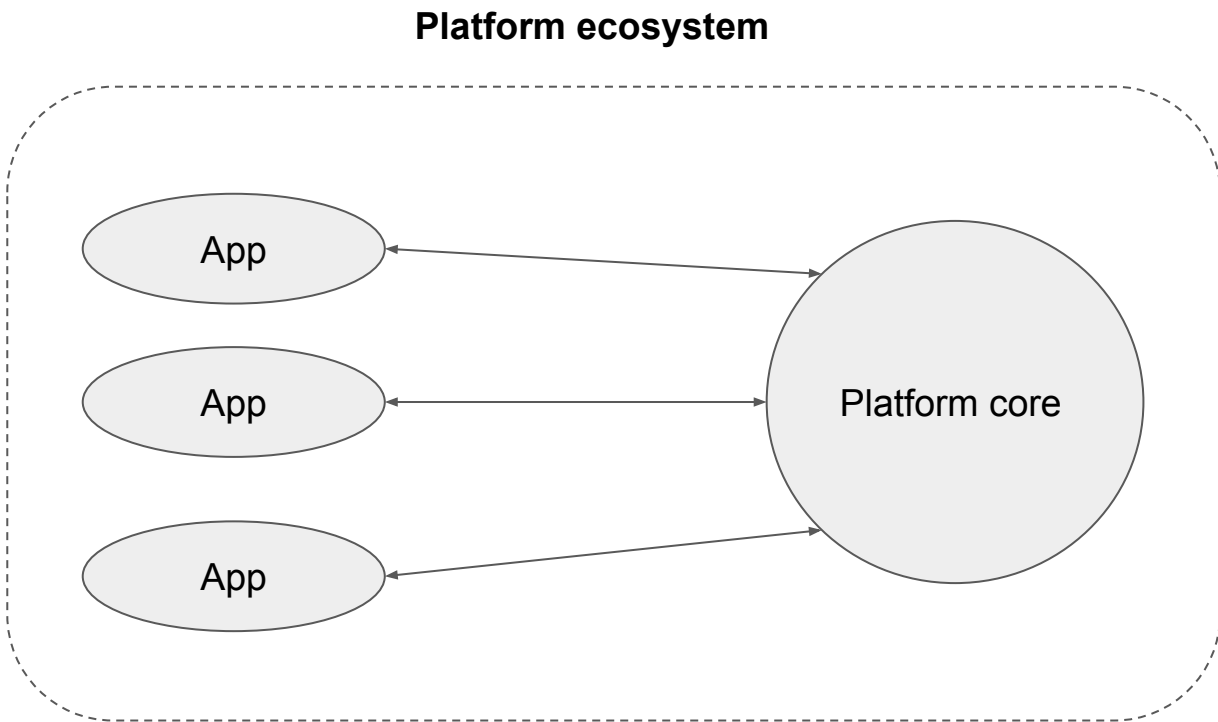
# What is a platform?

- Someone controls the platform core, often referred to as the platform owner.
- This can be one or several firms, and may be proprietary for-profit, or open source.
- The apps are often developed by third parties, that is, other firms or actors.

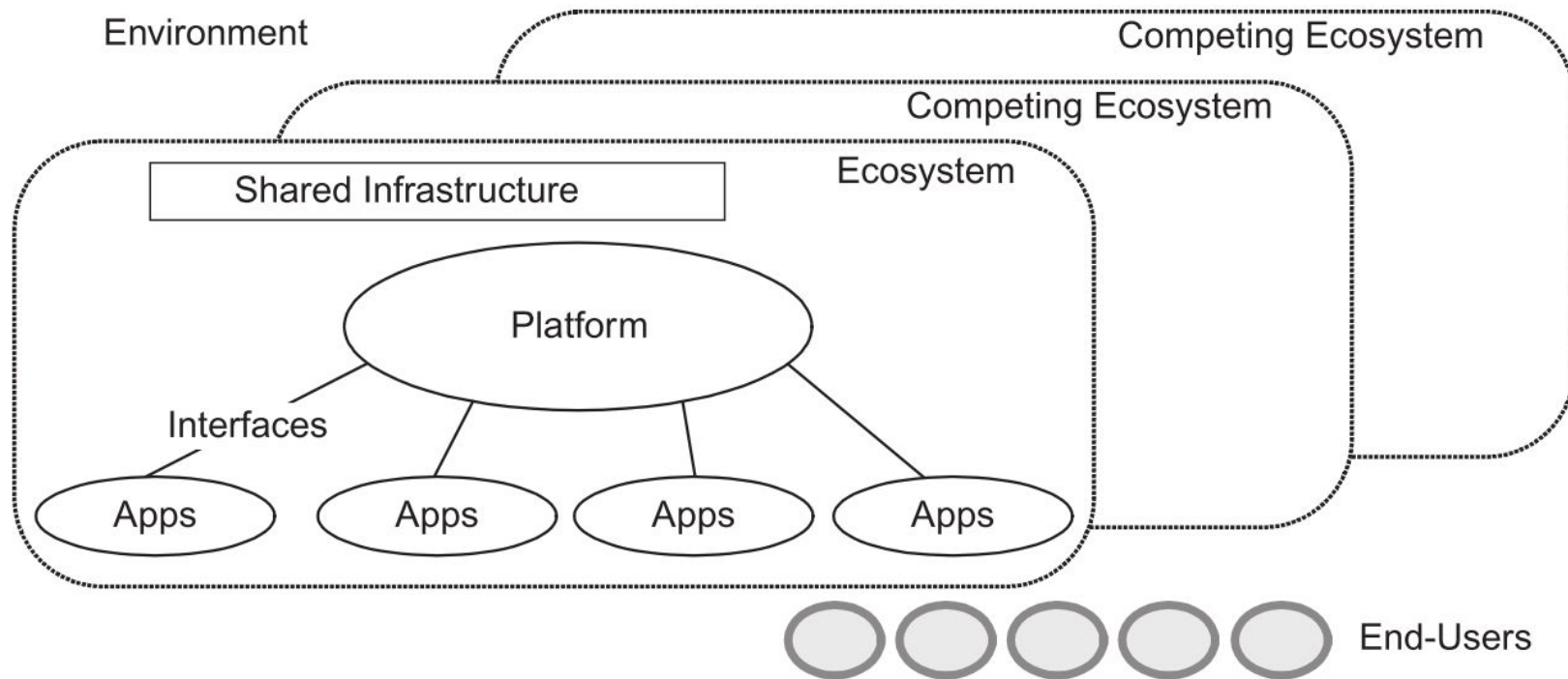


# What is a platform ecosystem?

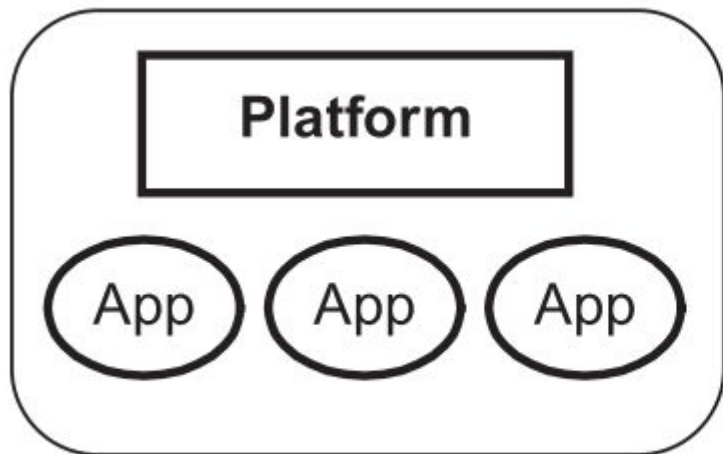
- A platform ecosystem refer to the platform core and the apps that interoperate with it.



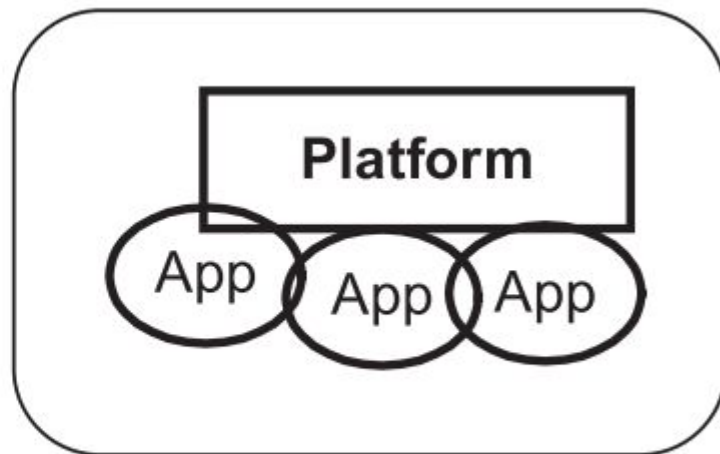
# Platform ecosystems



# Modularization / partitioning



A

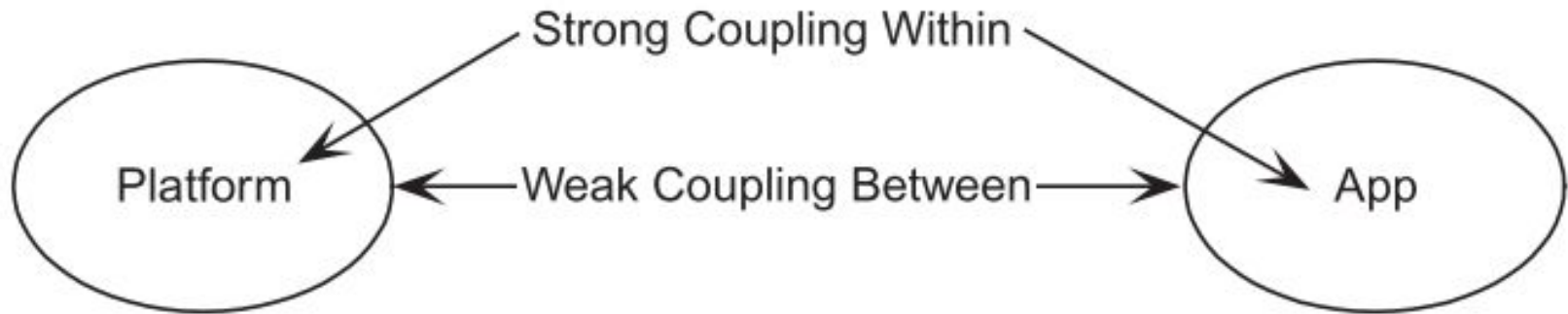


B

Tiwana 2013, p80

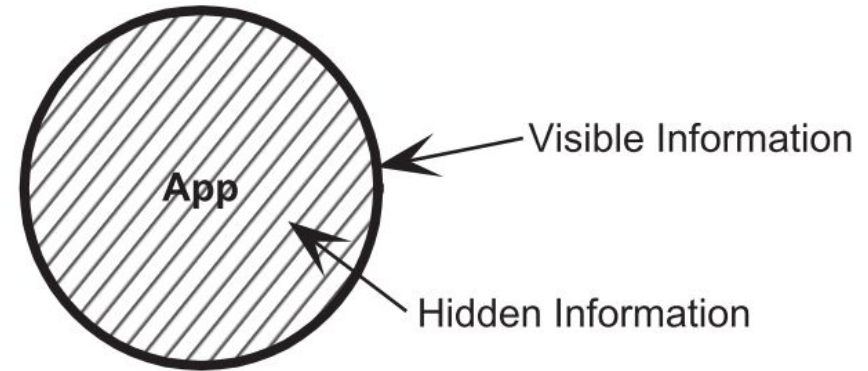
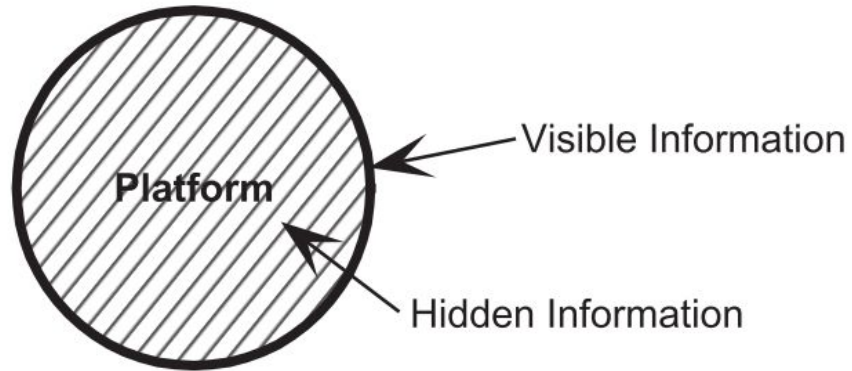


# Coupling



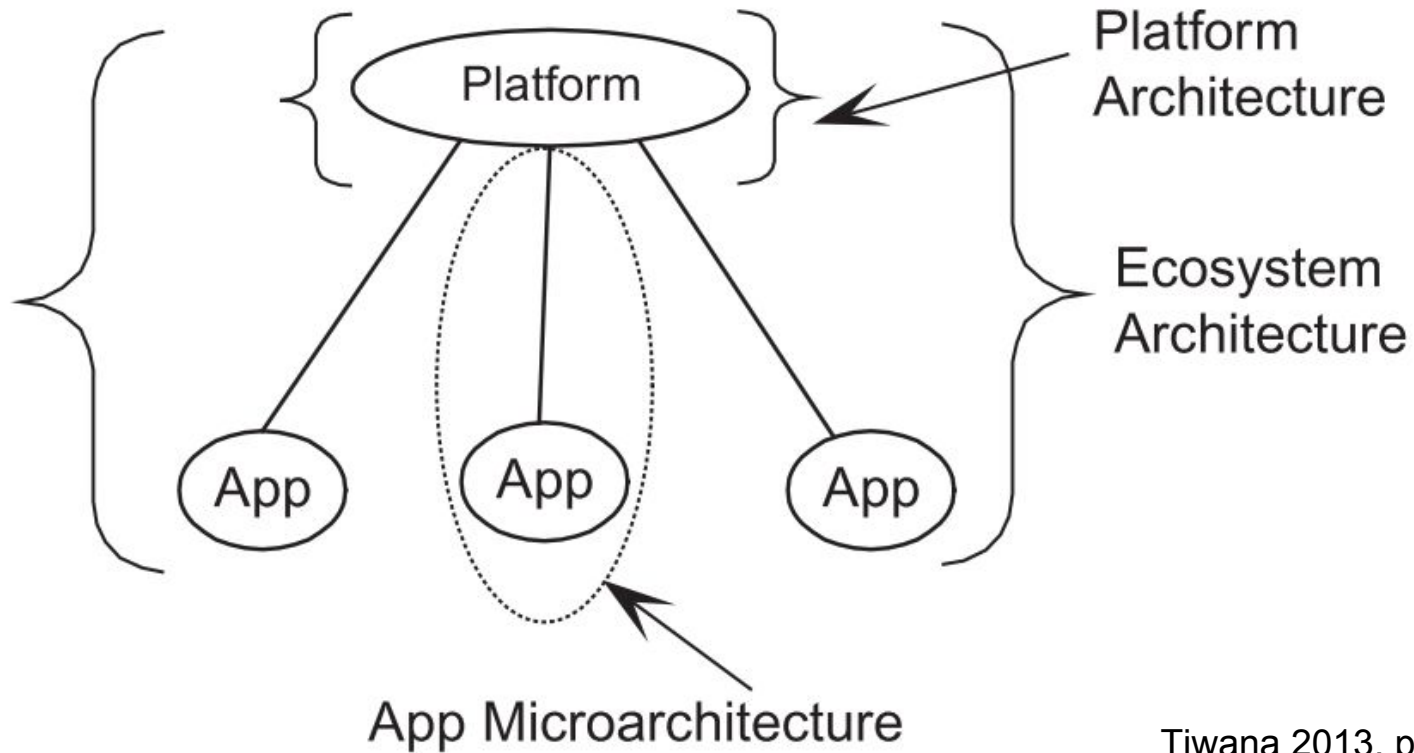
Tiwana 2013, p107

# Coupling



Tiwana 2013, p107

# Two levels of architecture

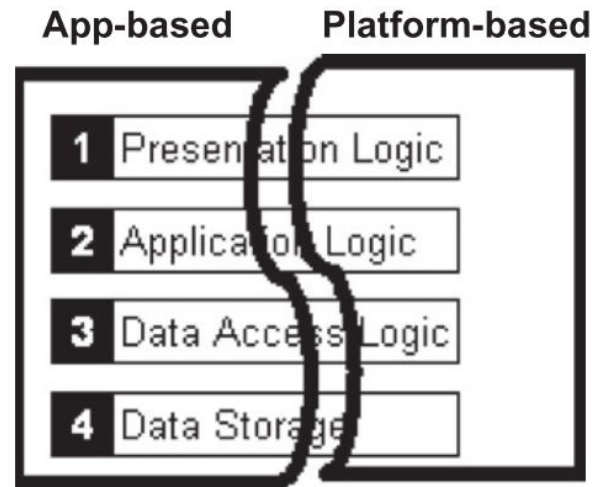


Tiwana 2013, p85

# App micro-architecture

Consists of:

1. Presentation logic (interface between app and end-user)
2. Application logic (the functionality)
3. Data Access logic (processes to interact with data)
4. Data storage (storage of data)



## Why platform?

*“Architecture is therefore a tool for simplifying and precisely describing the interconnections between parts of an ecosystem—potentially reducing structural complexity. It does this by reconfiguring the structure of dependencies between the platform and its apps within an ecosystem.”*

Tiwana 2013, p78

# Why platform? Desirable properties

1. **Simple:** The architecture should be easy to describe on a high level of abstraction. Such as its main components, how they are partitioned, and how they interact
2. **Resilient:** Keep dependencies to a minimum so that one defect app does not affect others.
3. **Maintainable:** Allow changes in components, without breaking others. Stable interfaces are key.
4. **Evolvable:** Support new functionality / innovation over time. Again, stable interfaces are key.

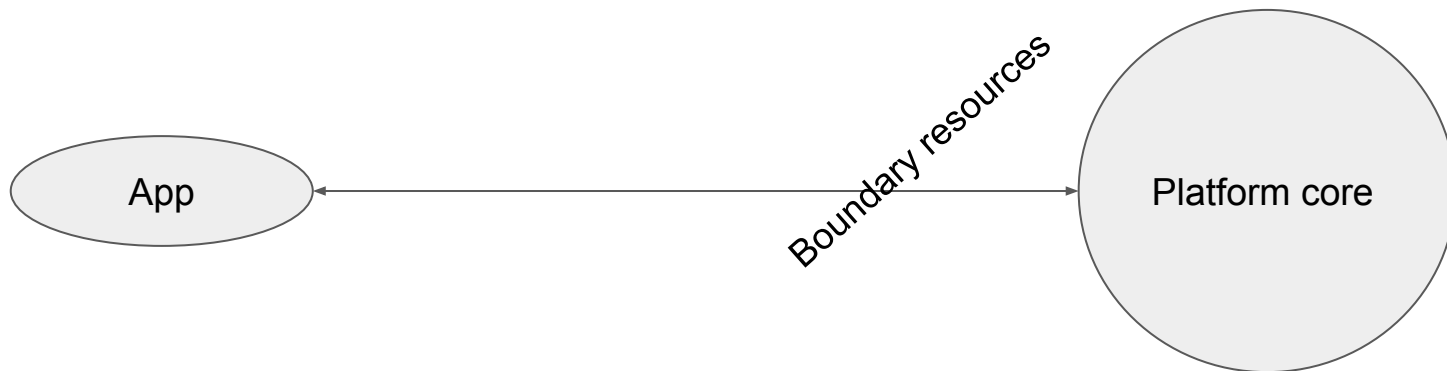
# Boundary resources

*“To successfully build platform ecosystems, the focus of the platform owner must shift from developing applications to providing resources that support third-party developers in their development work” - Ghazawneh & Henfridsson 2013 p 174*

→ Boundary resources: resources enabling third party development through tools and regulations

**Third parties / app  
developers**

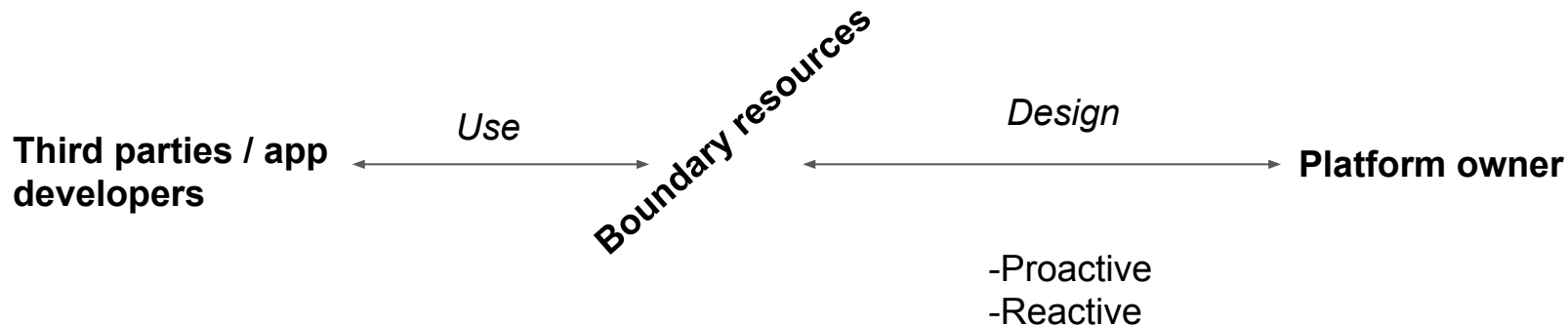
**Platform owner**



# Boundary resources

- To enable innovation, design and development of new functionality to the platform.
- To control the platform and its evolution in some desired direction.

Therefore: boundary resources has to be designed with the balance between these two in mind.





# Platforms, ecosystems and boundary resources - example

Ghazawneh & Henfridsson 2013 describe the evolution of the iOS from a closed operating system toward a platform ecosystem.

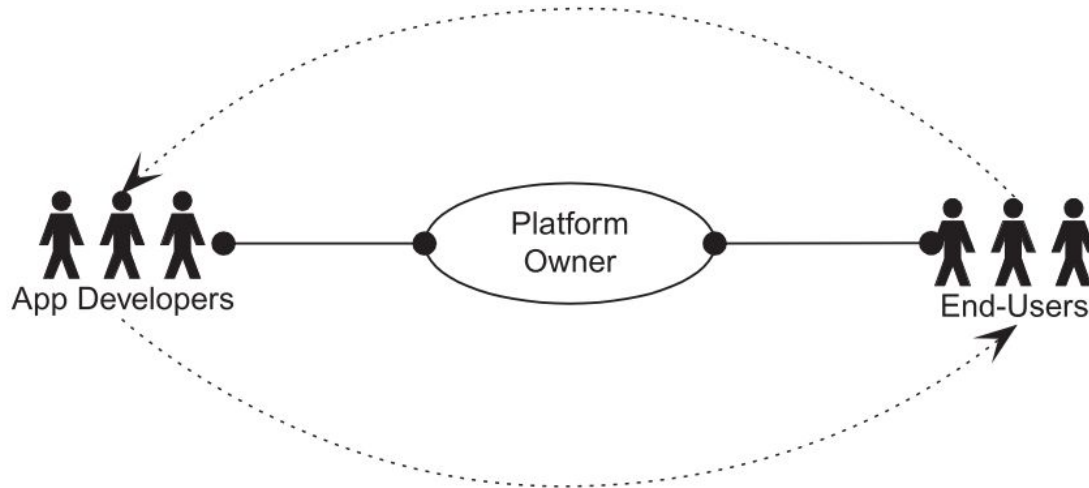
We follow the design of boundary resources, which both enable third-party developers to create apps, and exercise control over the platform.

1. Opening up the system with an SDK - and adding a review process
2. Extending the API with additional features.
3. Extended control: Ensuring that competing platforms are not running on their platform through regulations.

# Core characteristics

# Platforms are multisided

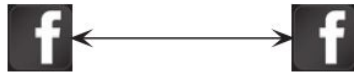
- According to Tiwana, a fundamental characteristic of platforms is that they are **multisided**. That is, they bring together two different groups of actors.
- The platform provides value for both parties by mediating interactions
- For a platform to survive, it must make the interaction between these actors easier than not using the platform



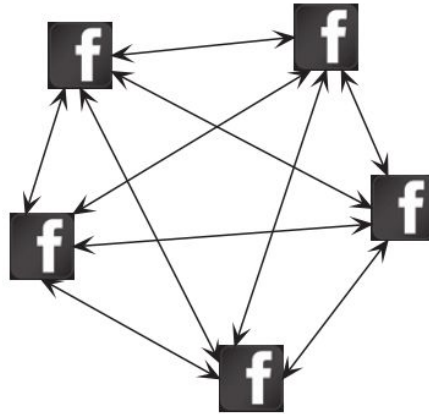
*Tiwana, 2013, p32*

# Network effects

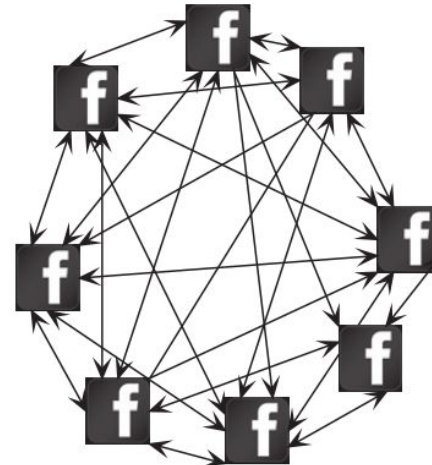
- The added value of a platform when new users adopt it is referred to as network effects.
- For each new user, the value for other users increases. - More exponentially than lineary.



A Two Users



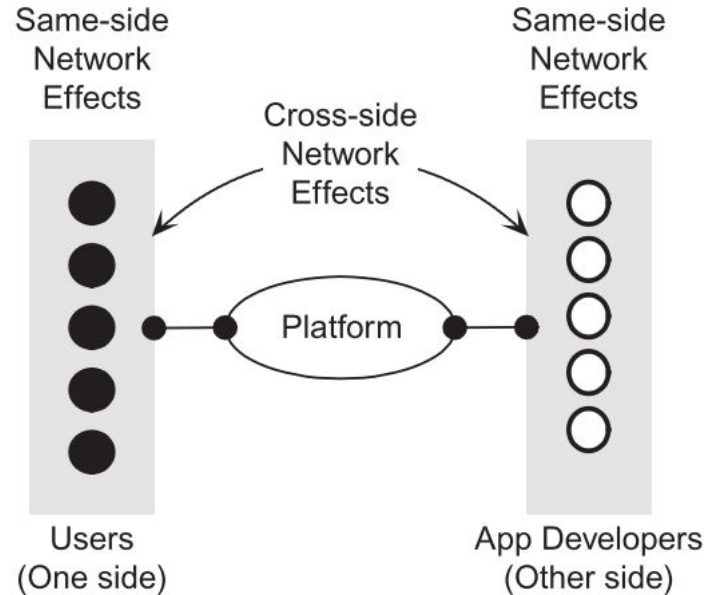
B Five Users



C Eight Users

# Network effects: same-side or cross-side

- Network effects can be same-sided: Value is added to other end-users when a new end-user is added.
- Cross-side: value is added for app-developers when a new end-user is added.

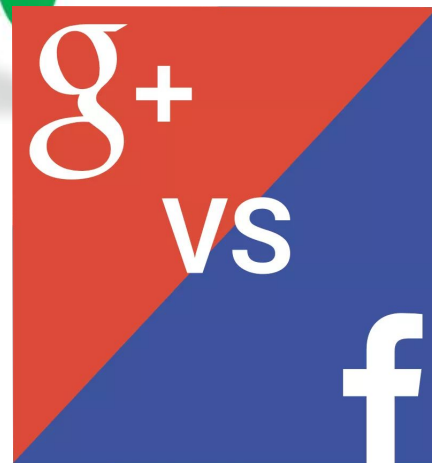


# Network effects: same-side or cross-side

- Network effects can be same-sided: Value is added to other end-users when a new end-user is added.



0 other users



# Network effects: same-side or cross-side

- Cross-side: value is added for app-developers when a new end-user is added.

The screenshot displays the Jamba mobile app store interface. At the top, the Jamba logo is on the left, and a navigation menu includes 'HJEM', 'RINGETONER' (highlighted), 'MUSIKK', 'SPILL & SOFTWARE', 'GRAFIKK', 'VIDEO', and 'EROTIKK'. Below the menu is a search bar with the text 'Søk' and a button 'Velg din mobil' with an 'OK' button.

The main content area is divided into several sections:

- MOBILSTÆSJ UTSALG ALT TIL MOBILEN:** A banner at the top left promoting mobile phone accessories.
- VIDEORINGETONE & RINGETONER:** A section featuring 'Schneffeline' and 'Sweety' ringtones. Each item includes a price (e.g., A1049, A1071, A1070) and a 'Bestill' button.
- BESTSELGERE - REALTONES:** A list of top-selling ringtones, including 'Kongen Av Mallorca', 'Hv Ringer Din Halle', and 'Look Away Lucifer'.
- NYTT - REALTONES:** A list of new ringtones, including 'Only Just Begun', 'Back On The Road (with Paper...', and 'Sillycone Valley'.
- Puppepopp til mobilen!:** A promotional banner on the right side featuring a smartphone and a doll.

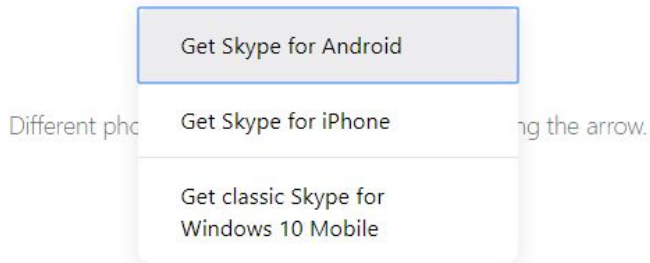
At the bottom, there is a 'VORDAN BESTILLER JEG?' section with three steps: 1. SEND (SENDING) TIL 2125\*, 2. STING SVART PÅ MOTTATT SMS, and 3. DU MOTTAR EN SMS MED DOWNLOAD LINK. A 'GRAFIKK' section is also visible at the bottom left.

# Multihoming

- When a user, app provider, etc. participates in multiple platform ecosystems.

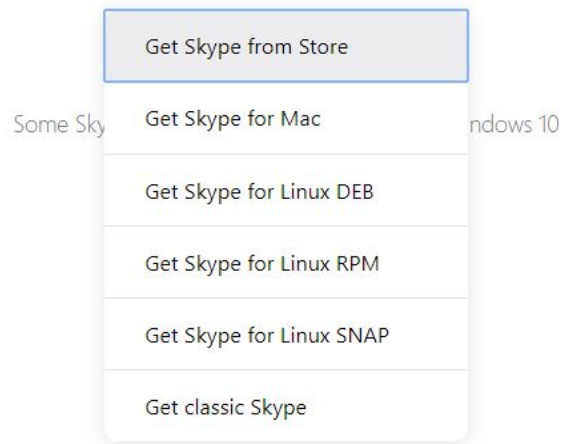
## Skype for Mobile

Available for Android, iPhone and Windows 10 Mobile.



## Skype for Desktop

On Windows 10? Skype is already in your computer.





# Multihoming

Could occur:

- In lack of one clear industry leader (provider-side)
- When one platform ecosystem lack functionality that is important to the end-user.




# Multihoming

- Platform owners have to decide whether to promote or discourage it.
- For instance, the iOS platform ecosystem allow users to instal apps from Google's ecosystem

## App Store Preview

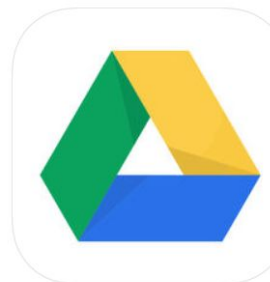
This app is only available on t



**Google Chrome** 17+  
Fast & Secure Web Browser  
[Google, Inc.](#)  
#2 in Utilities  
★★★★☆ 11.5K Ratings  
Free



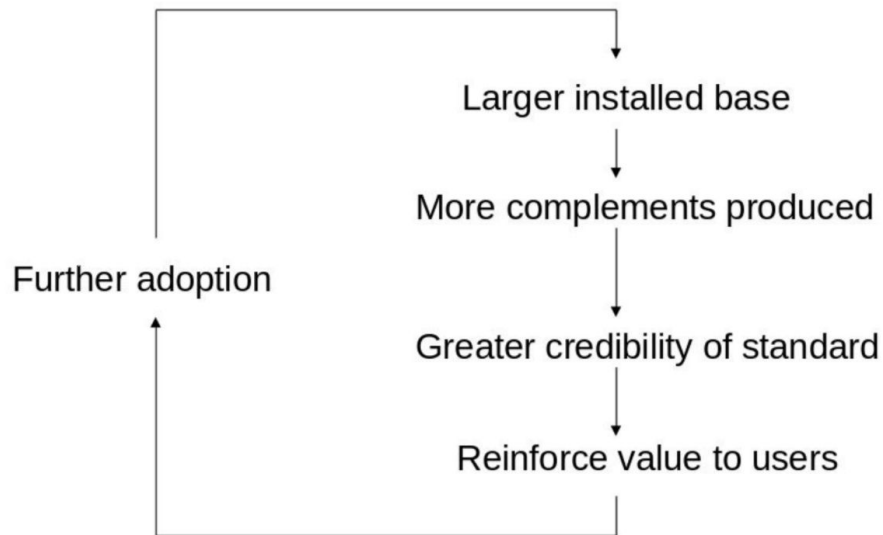
**Google Photos** 4+  
Free Photo and Video Storage  
[Google, Inc.](#)  
#4 in Photo & Video  
★★★★☆ 46.8K Ratings  
Free



**Google Drive** 4+  
Cloud storage space  
[Google, Inc.](#)  
#3 in Productivity  
★★★★☆ 1.4M Ratings  
Free · Offers In-App Purchases

# Tipping

- The point where sufficient users have adopted the platform is called the tipping point or the *critical mass*.
- At this point, the network-effect may create a self-reinforcing feedback loop of further adoption.



# Lock-in

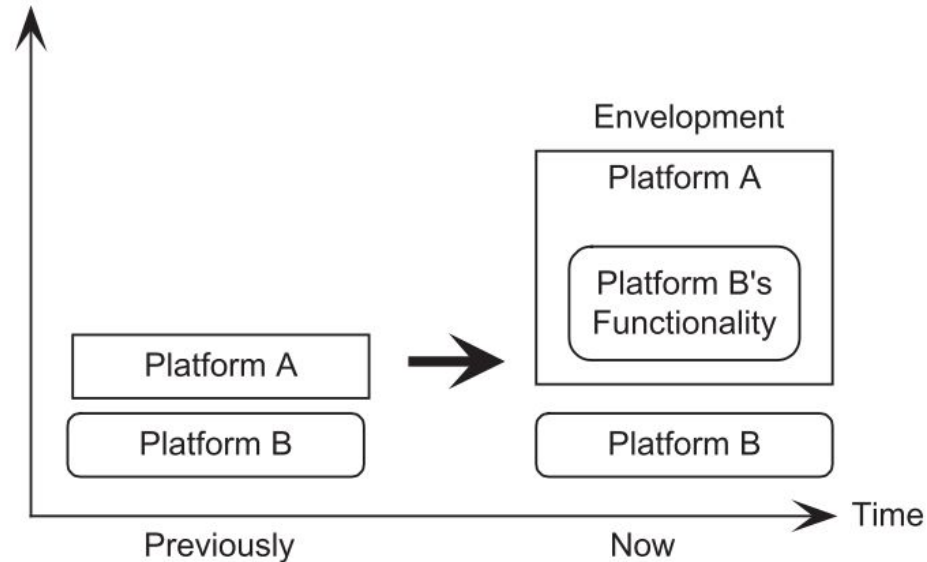
- Challenged by competing platforms, a platform owner need to find ways to keep their existing users.
- Lock-in is often used as a negative term: Locking a customer to one platform or software solution, for example due to high switching-costs.
- Lock-ins can also be created by providing increasing value to the existing user.





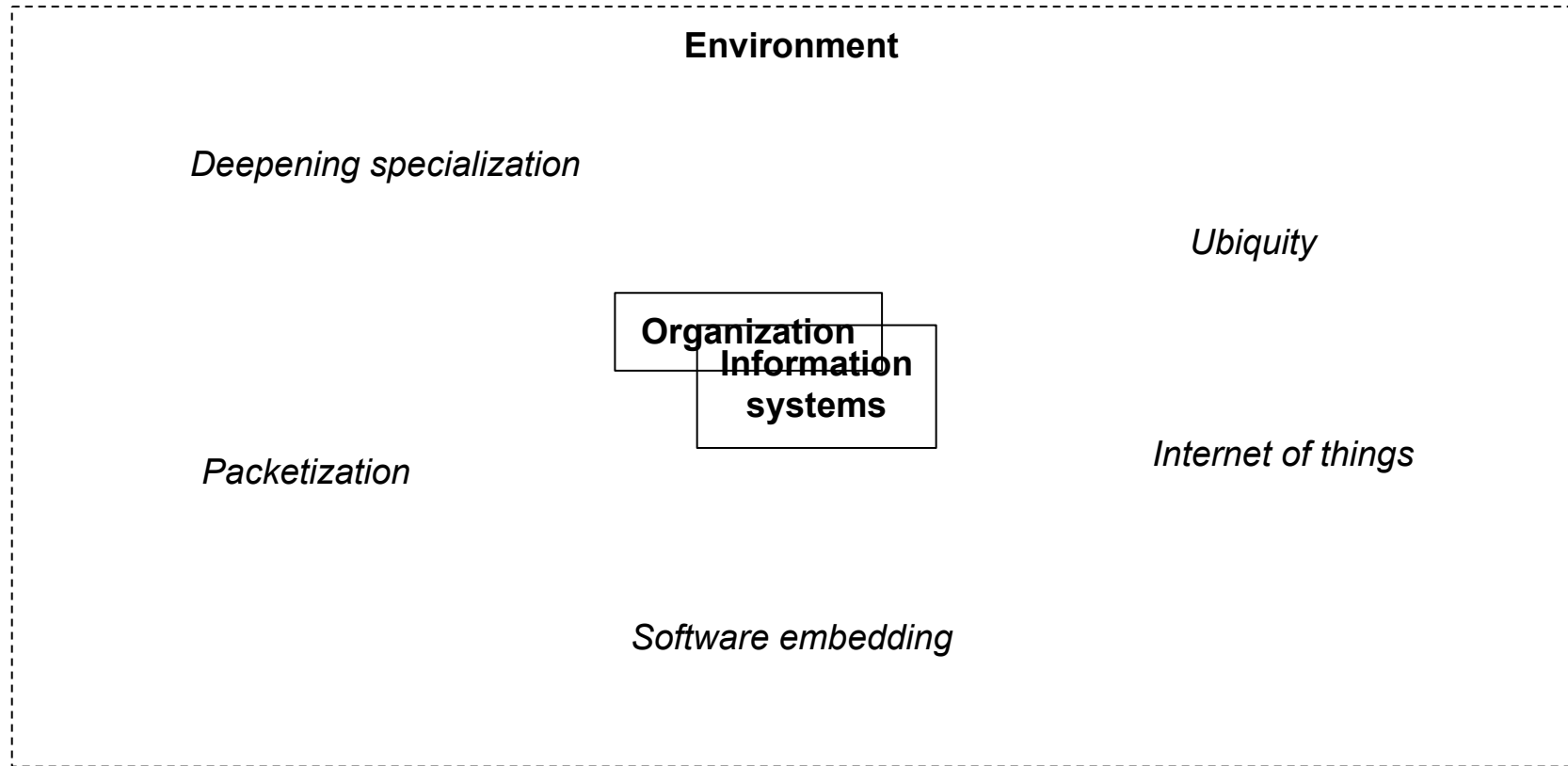
# Envelopment

- By extending a platform to offer services and functionality of another platform, the other platform is enveloped.
- For example, when Apple created their own Map-application for iOS, it enveloped Google Maps.



# Drivers toward platform ecosystems

# Drivers towards platform architectures (Tiwana, 2013)





# Driver 1: Deepening specialization

- As things are getting digitized, no one can cover all.
- Developers need to specialize.
- Complexity is increasing:
  - Lines of code are estimated to double every two years.

*“This creates a greater pressure for companies to more deeply specialize in their core competence and leave the rest to capable partners”*

Tiwana, 2013, p11

- Ecosystems allow these specialized entities to work together on a common platform.
- No longer constrained by geography and coordination

# Driver 2: Packetization

- Internet and technology allows *everything* to be digitized and sent over the network at no time and cost.
- Not only media (music, documents, video). Also human activities, processes, services



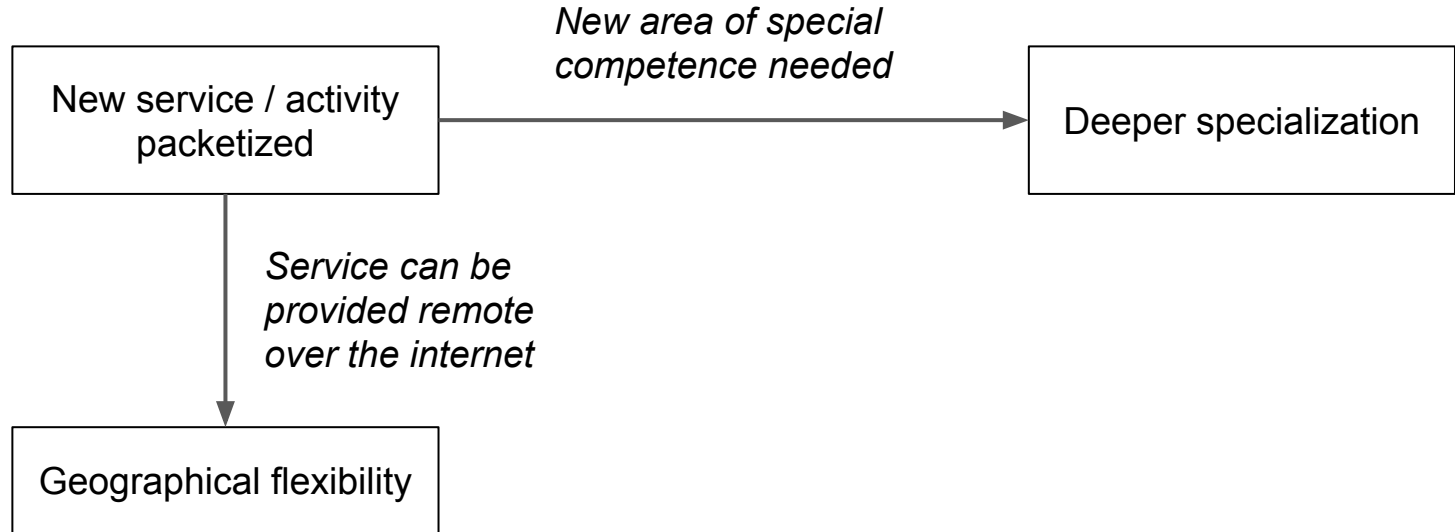
[“Fast-food restaurants begin outsourcing drive-through order-taking”](#) from 2009



Self-service ordering in 2017

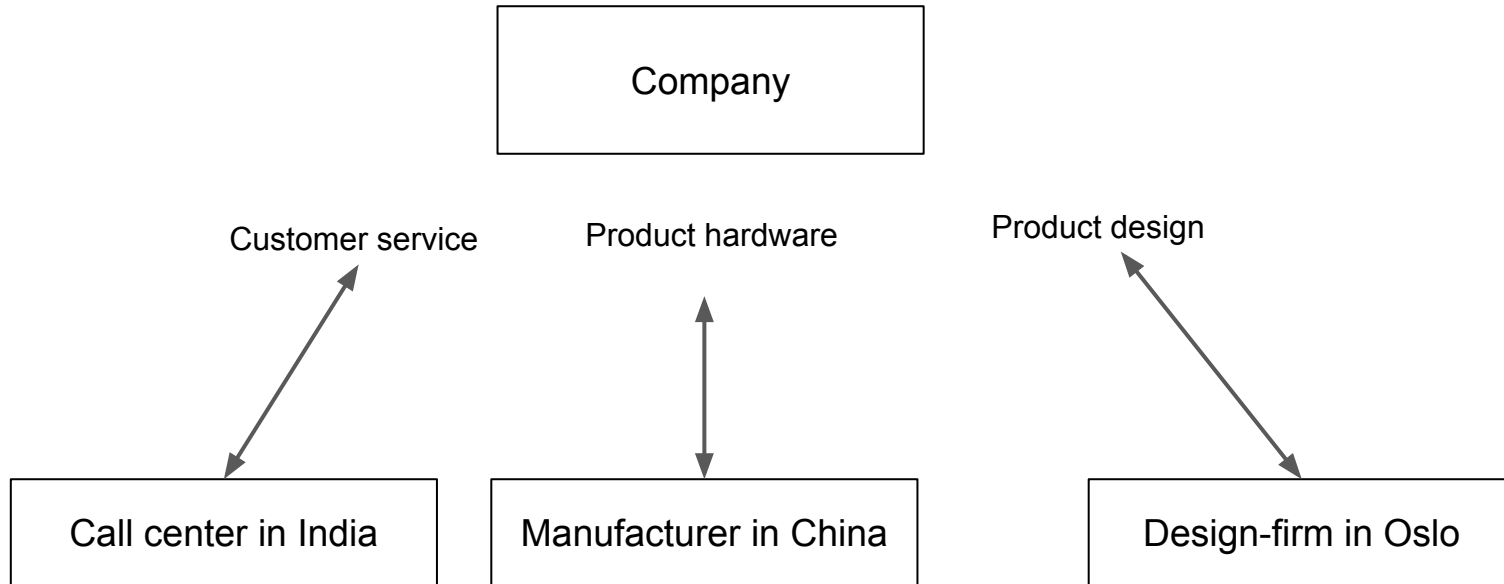
## Driver 2: Packetization

- Packetization increases geographical flexibility.
- Enforces the process of deeper specialization.



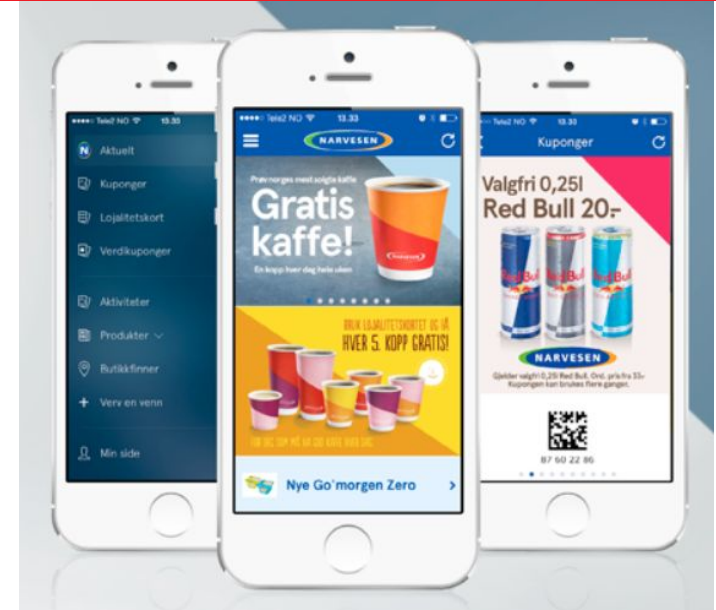
## Driver 2: Packetization

- Packetization increases geographical flexibility.
- Enforces the process of deeper specialization.



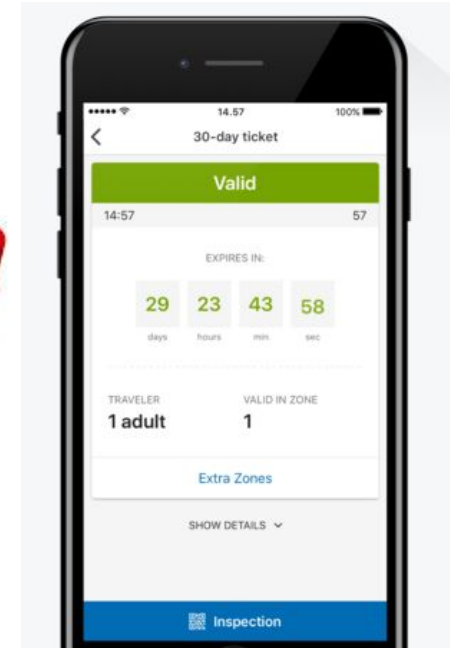
# Driver 3: Software embedding

- Activities or processes are being baked into software
- All aspects of business and society are becoming increasingly embedded in software.
- They are increasingly networked - connected to other software



# Driver 3: Software embedding

- Baking activities or processes into software
- All aspects of business and society are becoming increasingly embedded in software.
- They are increasingly networked - connected to other software



# Driver 3: Software embedding

- Baking activities or processes into software
- All aspects of business and society are becoming increasingly embedded in software.
- They are increasingly networked - connected to other software



# Driver 3: Software embedding

- Software embedding may result in the transformation of products into services
- Instead of offering a finished product, the software or service allows for extensions, updates and so on.



*Cars have moved from a finished product, to a platform of software-based services*



## Driver 3: Software embedding

- Software embedding blurs the boundaries between the physical and digital world.
- Businesses that do not produce digital products increasingly have to focus on digital technologies



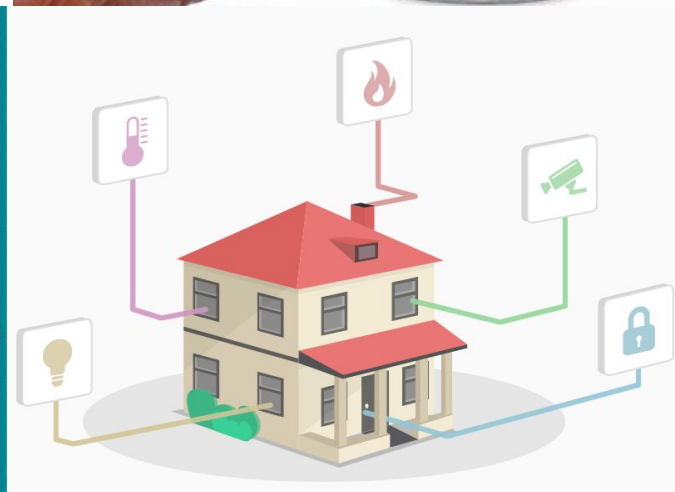
# Driver 3: Software embedding

- Software embedding may result in convergence. That is, businesses that originally does not compete, suddenly are entering each others markets.



# Driver 4: The internet of things

- Sensors and microchips are getting cheaper and smaller.
- All types of objects are increasingly connected to the internet.



## Driver 4: The internet of things

- Number of devices connected to the internet outgrew the world population in 2008
- In 2020 an estimate of 50 billion devices will be connected to the internet
- Data from all these objects can be used to gather data about people and their surroundings, and combined to form novel solutions
- The process is expected to continue: IPV6 allows 100 unique ip-addresses per atom on earth.

## Driver 5: Ubiquity

- Fast and cheap wireless internet connection are increasingly available around the globe.
- Enables packetization by allowing digital packets to travel over the internet protocol
- Enables The internet of things

*“everything [...] can be delivered anywhere”*

Tiwana 2013 p19

- Allows small firms and entrepreneurs to reach out to customers
- Challenges existing large companies

# Drivers towards platform architectures

1. **Deepening specialization**

Technologies get increasingly complex and specialized → Require deeper expertise

2. **Packetization**

Process of digitizing new phenomena such as activities or processes.

3. **Software embedding**

Business activities are put into software.

4. **Internet of things**

It is increasingly easy to connect everyday objects to the web.

5. **Ubiquity**

Fast and cheap networks are available “everywhere”

# Guiding principles

<b>Principle</b>	<b>Key Idea</b>
Red Queen effect	The increased pressure to adapt faster just to survive is driven by an increase in the evolutionary pace of rival technology solutions
Chicken-or-egg problem	The dilemma that neither side will find a two-sided technology solution with potential network effects attractive enough to join without a large presence of the other side
The penguin problem	When potential adopters of a platform with potentially strong network effects stall in adopting it because they are unsure whether others will adopt it as well
Emergence	Properties of a platform that arise spontaneously as its participants pursue their own interests based on their own expertise but adapt to what other ecosystem participants are doing
Seesaw problem	The challenge of managing the delicate balance between app developers' autonomy to freely innovate and ensuring that apps seamlessly interoperate with the platform
Humpty Dumpty problem	When separating an app from the platform makes it difficult to subsequently reintegrate them
Mirroring principle	The organizational structure of a platform's ecosystem must mirror its architecture
Coevolution	Simultaneously adjusting architecture and governance of a platform or an app to maintain alignment between them
Goldilocks rule	Humans gravitate toward the middle over the two extreme choices given any three ordered choices