

IN5320 - Development in Platform Ecosystems

Lecture 3: *Platform Ecosystems fundamental concepts*

21th of October 2021

Department of Informatics, University of Oslo
Magnus Li - magl@ifi.uio.no

Today

- Some information about the project
- Platform ecosystems fundamental concepts

The project - evaluation criteria

Now available on the course page - you should plan the project accordingly!

Group projects are evaluated based on your final application, and your presentation of it. The application is evaluated on 1) functionality, 2) technical implementation, and 3) user interfaces. In your presentation, we will also evaluate your reflections upon these aspects and the process of design and development. In the presentation, you will also be asked to evaluate and grade your own work based on these criteria.

Reflections on your application and the design and development process	- Ability to articulate and reflect on positive and negative aspects of your own solution - Ability to elaborate on functional, implementation, and design aspects of your solution - Ability to explain why choices related to these aspects were made - Ability to elaborate and reflect on the project process (coordination, time management, planning, etc.) - Ability to elaborate and reflect on the use of methods for organizing the project work (e.g., from the method toolkit or other resources)
Functionality	- Fully working implementations of the fundamental requirements - Have implemented additional functionality (each assessed based on extensiveness, completeness, and plausible usefulness given the case description) - Additional functionality implemented seems appropriate (useful) given the case description
Technical implementation	- Git has been used to coordinate the development throughout the project - The app platform has been utilized - Use of DHIS2 data model to store data and configurations (as opposed to hard-coding this into the app) - Efficient use of APIs (e.g., server-side filtering, client-side caching) - Modularity and reusable components - Utilizing React capabilities - Readable and consistent naming conventions for functions, components, etc.
User interfaces	- Use of the DHIS2 UI library - Use of the DHIS2 UI style guide when UI library lacks components - Overall 'look and feel' appears user-friendly and robust - Appropriate and user-friendly terminology and error messages - Meaningful and intuitive navigation - Meaningful page layouts - UI consistency

Final exam format

1. DHIS2 as platform (40%)
Answer both of the following questions.

- In the course we have talked about two different types of digital platforms. What type(s) of platform is DHIS2? Discuss how DHIS2 meets the definition and characteristics of the argued type of platform using concepts you see relevant from the course literature (e.g., architecture, multi-sidedness, boundary resources, network effects, etc.)
- Boundary resources is an important concept in the platform literature. Provide some examples of boundary resources for the DHIS2 software, and discuss their role in supporting design and innovation. You can use examples from the project work.

2. Modularization and software platforms (15%)
Explain what is meant by modularization, and discuss how modularization is related to software platform architectures.

3. Platform strategy and generic software solutions (30%)
Explain what is meant by a platform strategy, and discuss why vendors of generic enterprise software solutions (such as SAP and DHIS2) would adopt such a strategy.

4. Platform forking (15%)
Explain what is meant by platform forking, and discuss how it is similar and/or different from “regular” software project forks.

The previous lecture

Summary

- Platform ecosystems take part in, and may be seen as ***information systems***

- They are thus part of **socio-technically complex** environments

- **Standards** are essential to information systems, and are an important part of platform architectures

- In the coming weeks, we will see that platforms has certain **architectural traits** that may promote desirable aspects
  - Reducing complexity, promoting innovation, enabling different levels of design, facilitating integration

Platform ecosystems

Platforms

Two types of *digital platforms*:

- **Transaction platforms**
  Facilitates interaction between participants in a market or network

- **Innovation platforms** (software platforms or platform architectures)
  Facilitates innovation by enabling efficient use and reorganization of resources by a large audience.
  → Main focus of this course

What is a platform?

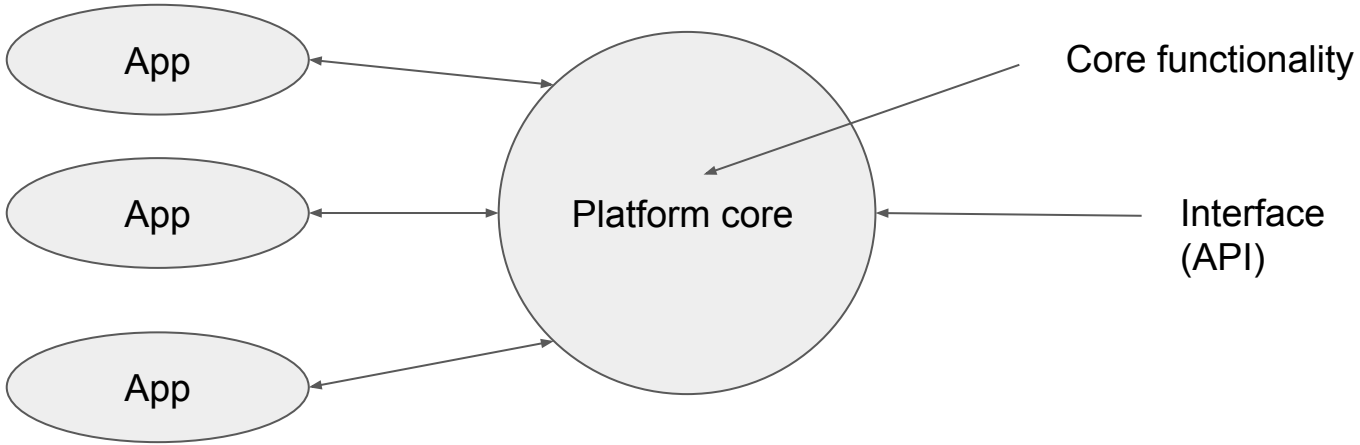
“A software platform is a software-based product or service that serves as a foundation on which outside parties can build complementary products or services” - Tiwana 2013 p5

“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?

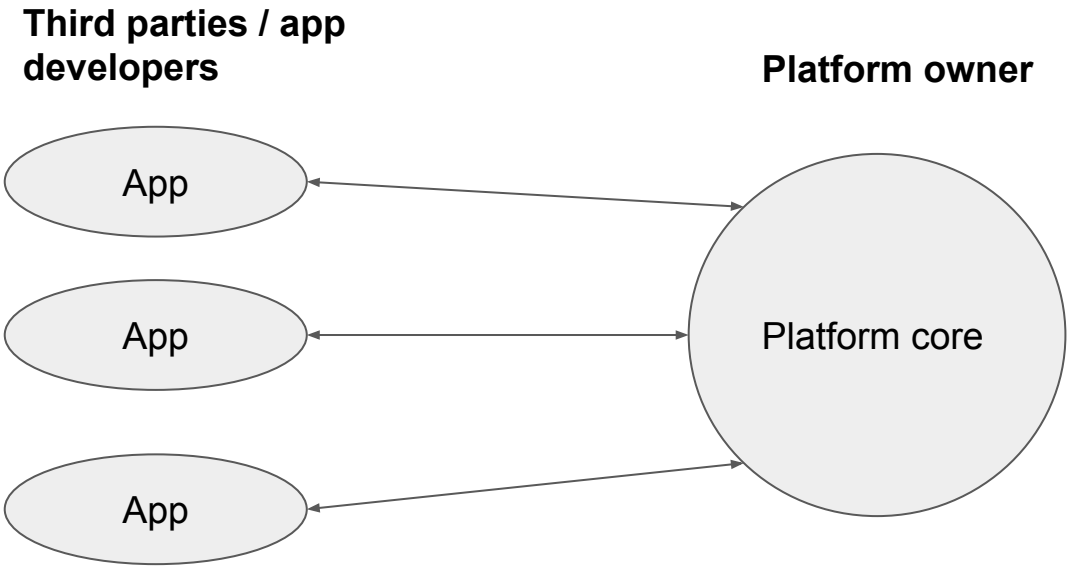
"A software platform is a software-based product or service that serves as a foundation on which outside parties can build complementary products or services" - Tiwana 2013 p5

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

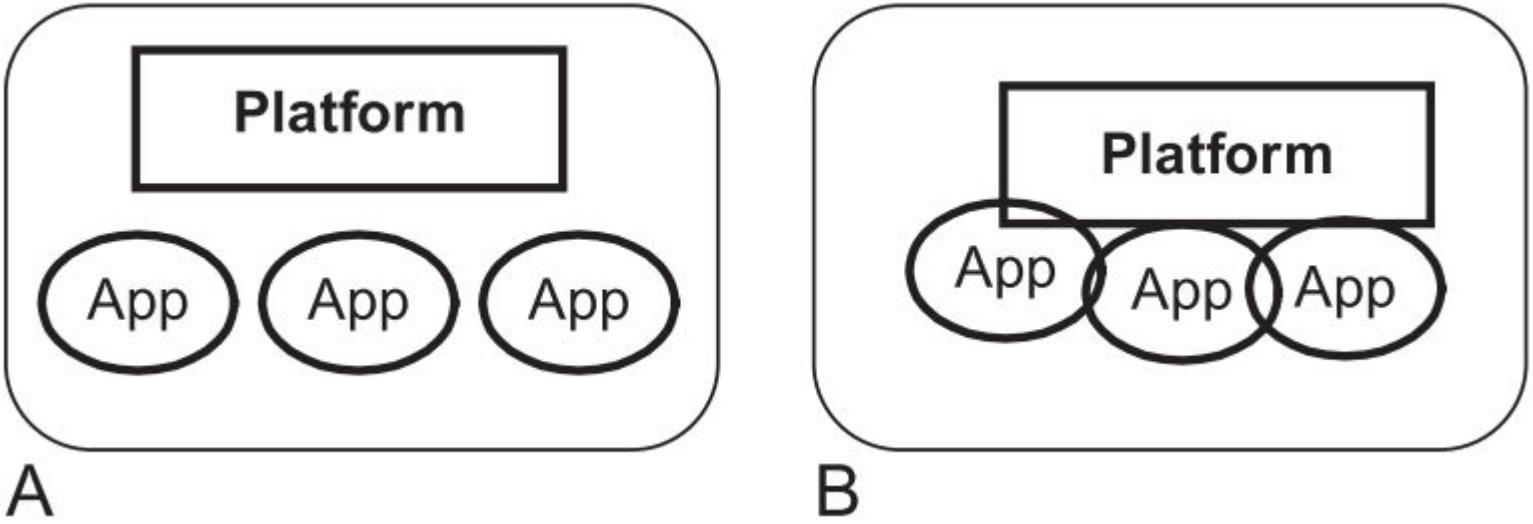


What is a platform?

- 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.

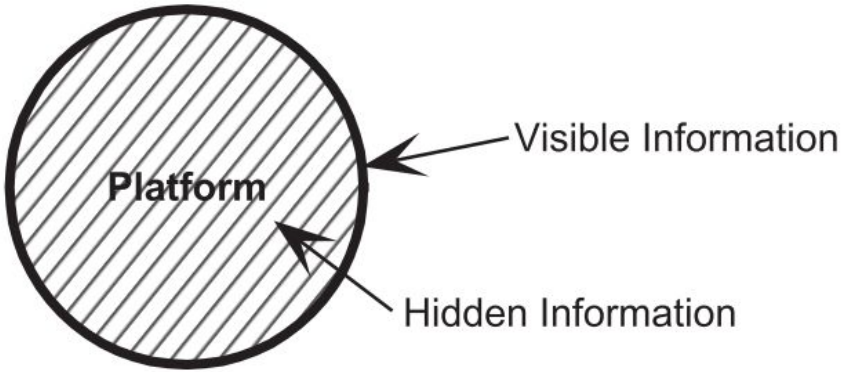


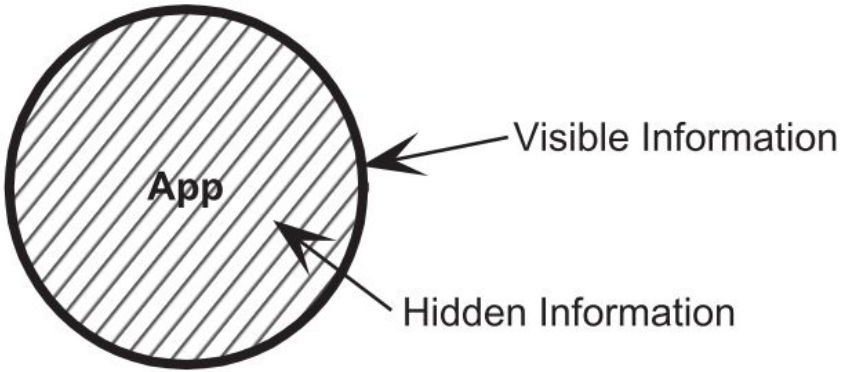
Modularization / partitioning



Tiwana 2013, p80

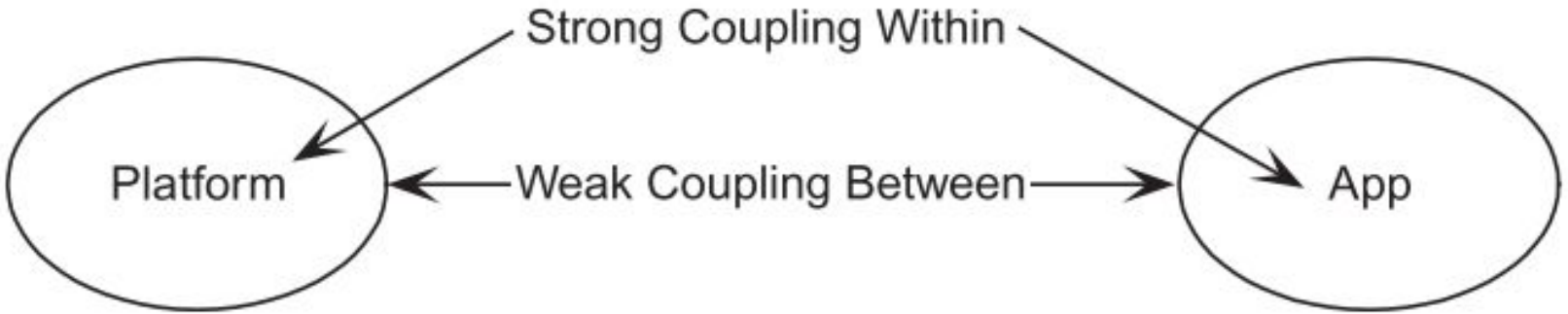
Coupling and abstraction





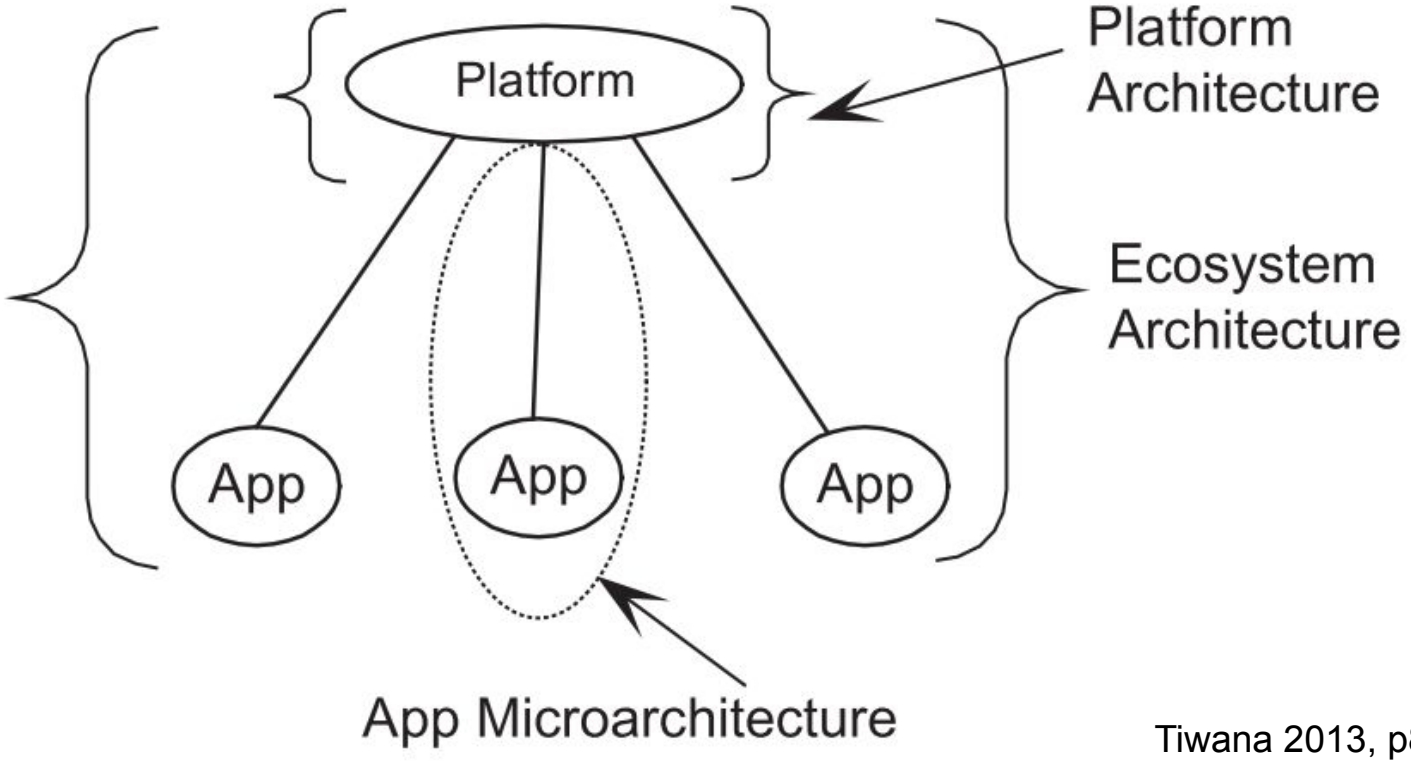
Tiwana 2013, p107

Coupling



Tiwana 2013, p107

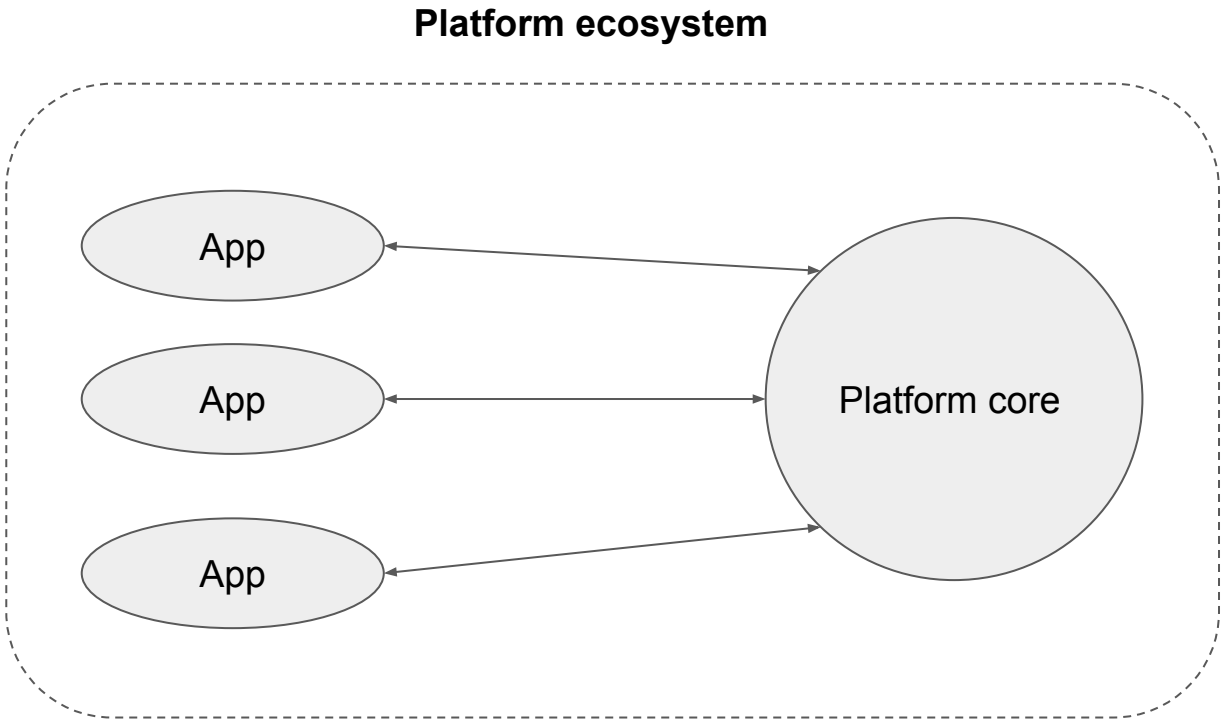
Two levels of architecture



Tiwana 2013, p85

What is a platform ecosystem?

- Tiwana define the platform ecosystem as the platform core + the apps



What is a platform ecosystem?

- Tiwana define the platform ecosystem as the platform core + the apps

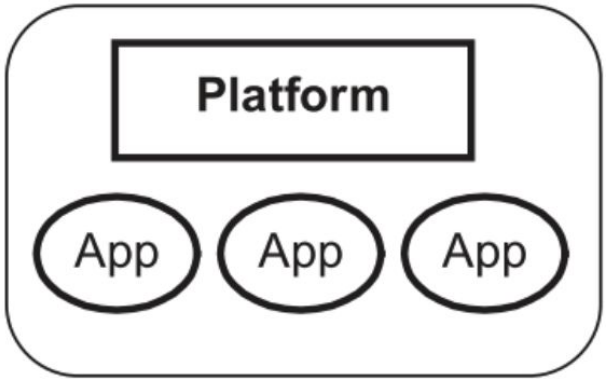
- Other *socio-technical* definitions use “ecosystem” to refer to the platform core, the apps, and all the actors involved in their design, development, and use (i.e., platform owner, third-party developers, and end-users)

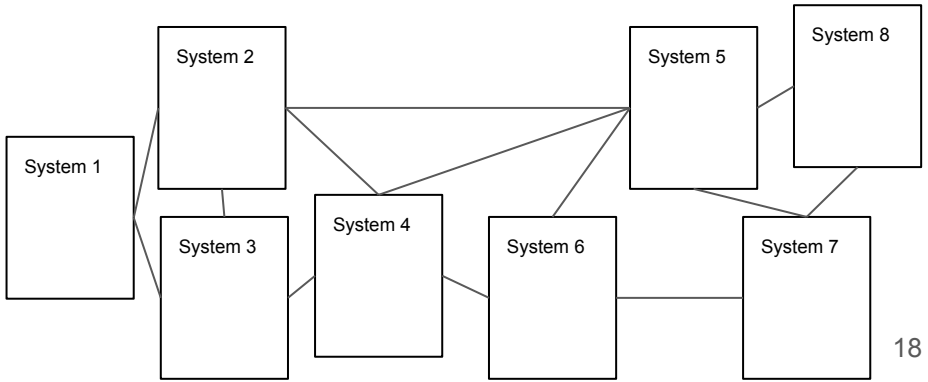


Exercise

For two minutes. Note down some bullet points on paper on the following question:

- Consider the difference between the “silo architecture” versus the platform architecture
- What are benefits with the platform architecture?





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.

→ reusability of code and data?

Core characteristics

Core characteristics (beyond the architecture)

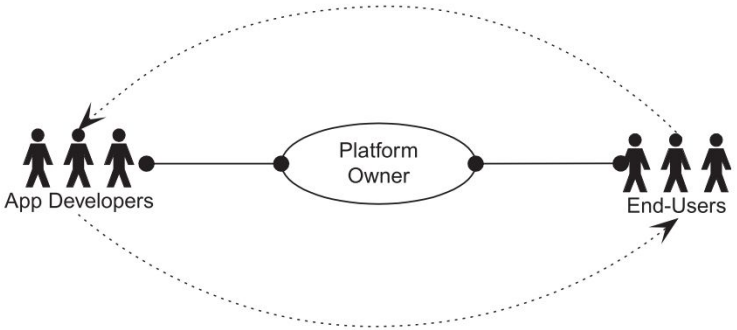
- *Innovation* or *software* platforms are characterized by an architecture with a stable core + APIs - enabling development of apps.

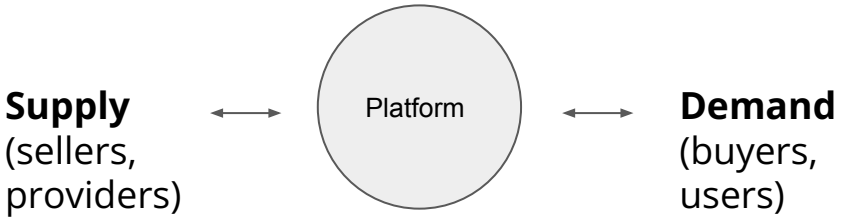
Additional characteristics:

- Multi-sidedness
- Network effects
- Multi-homing
- Tipping-points
- Lock-ins
- Competitive durability
- Envelopment

Platforms are multisided

- According to Tiwana, a fundamental characteristic of platforms is that they are ***multisided***. That is, they bring together two or more different groups of actors.
- The platform provides value for both/all 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

Exercise

For one minute. Note down some bullet points on paper on the following question:

Uber allows you to book car-rides through an app on your phone

- Is it a transaction or innovation platform or neither?
- Consider Tiwana's concept of multi-sidedness: what are the "sides" on Uber?
- How does Uber differ from traditional Taxi services?

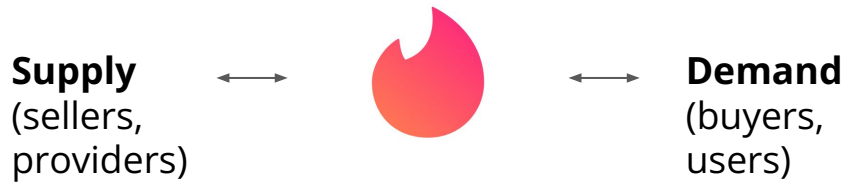


Exercise

For one minute. Note down some bullet points on paper on the following question:

Tinder is a dating application that allows single people to meet other single people.

- Is it a transaction or innovation platform or neither?
- Consider Tiwana's concept of multi-sidedness: what are the "sides" on Tinder?



Exercise

For one minute. Note down some bullet points on paper on the following question:

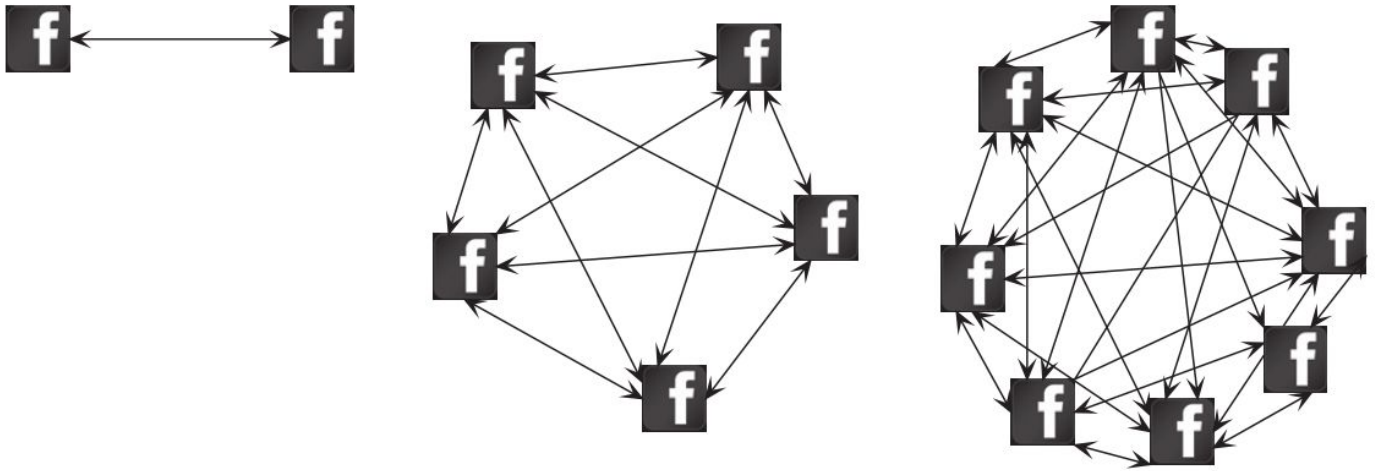
Voi allows you to book e-scooters on your phone

- Is it a transaction or innovation platform or neither?
- Consider Tiwana's concept of multi-sidedness: what are the "sides" on Voi?



Network effects

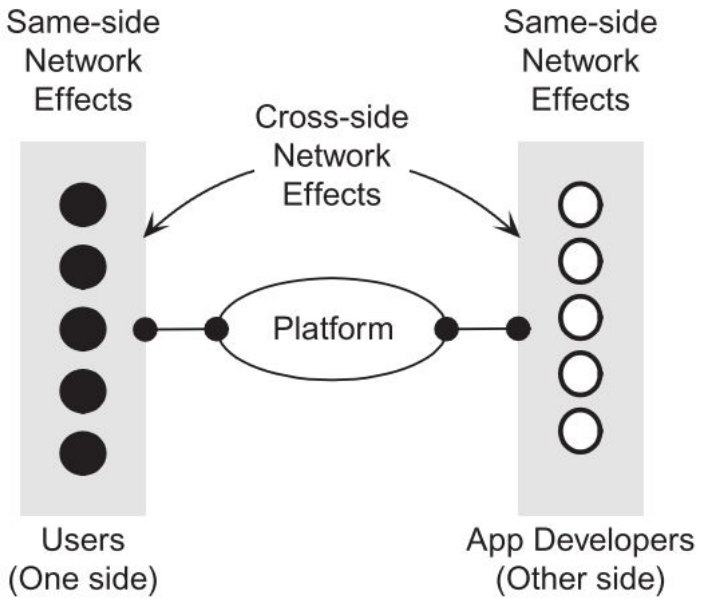
- The added value of a platform when new users or providers adopt it is referred to as network effects.
- For each new user or provider, the value for other users and providers increases. - More exponentially than lineary.



Tiwana, 2013, p34

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: e.g., value is added for app-developers when a new end-user is added.



Tiwana, 2013, p35

Exercise

For one minute. Note down some bullet points on paper on the following question:

If you were to establish a new software platform: how would you recruit your first users and developers?

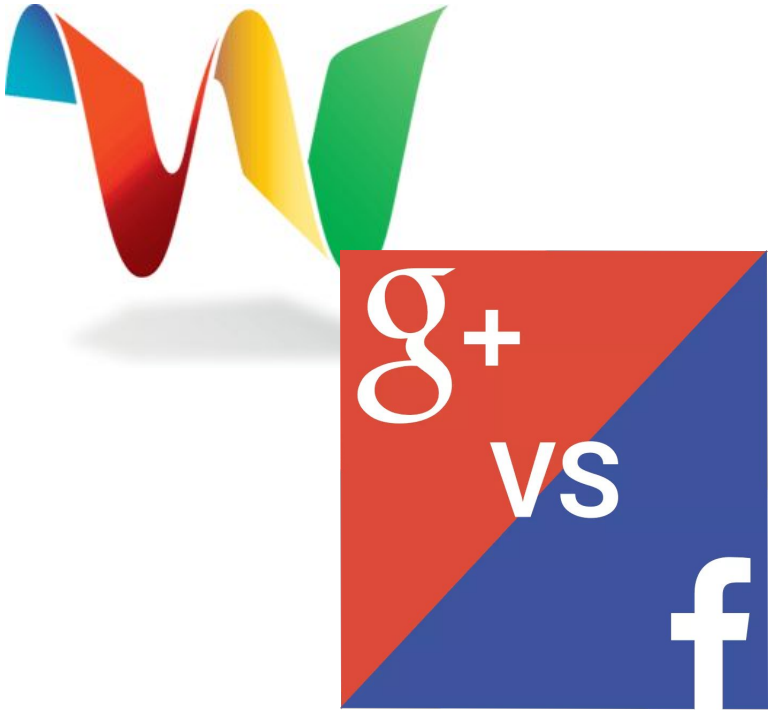
Network effects: same-side or cross-side

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



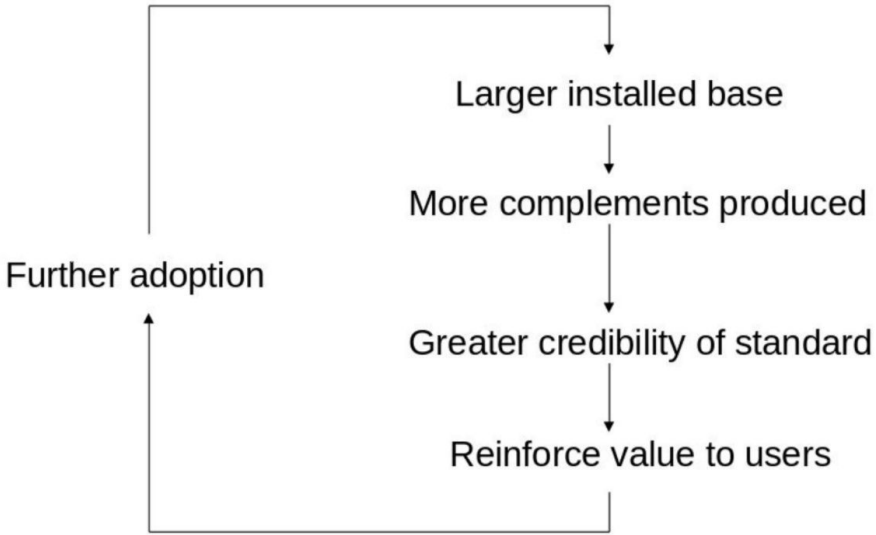
0 other users





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.



Ciborra et al., 2000

Multihoming

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





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.



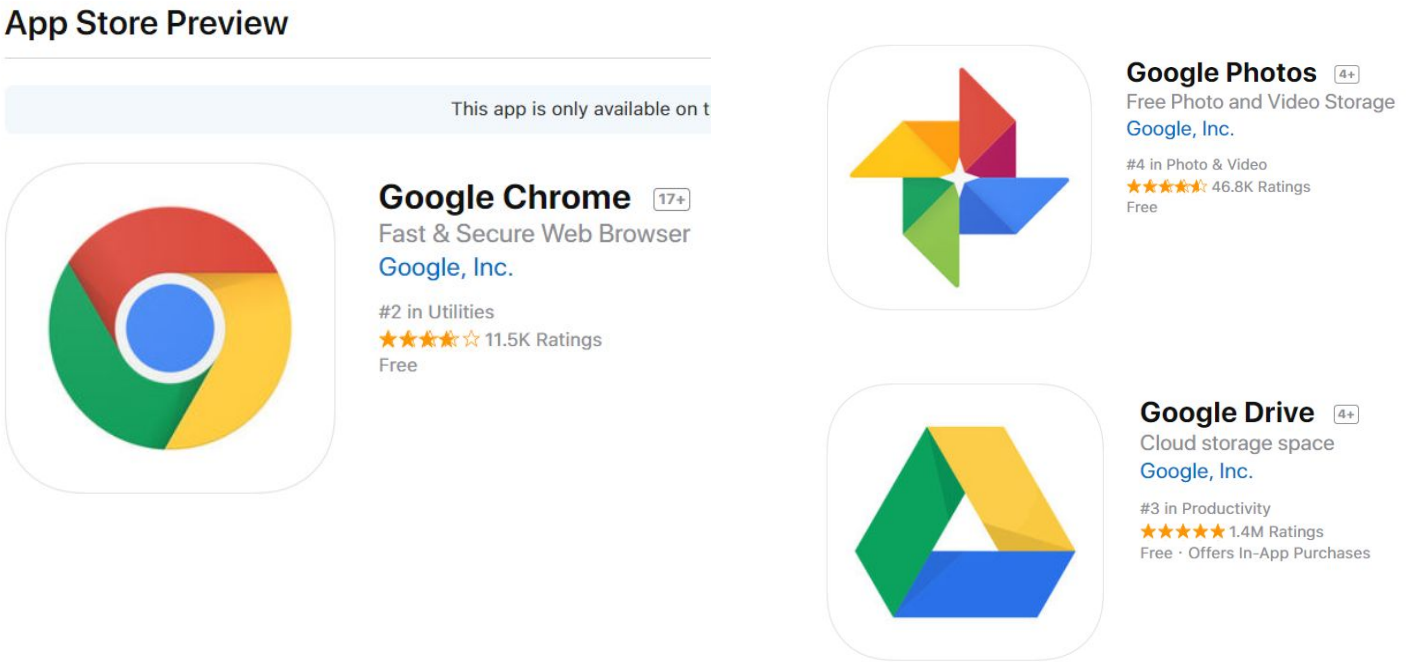






Multihoming

- 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



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.



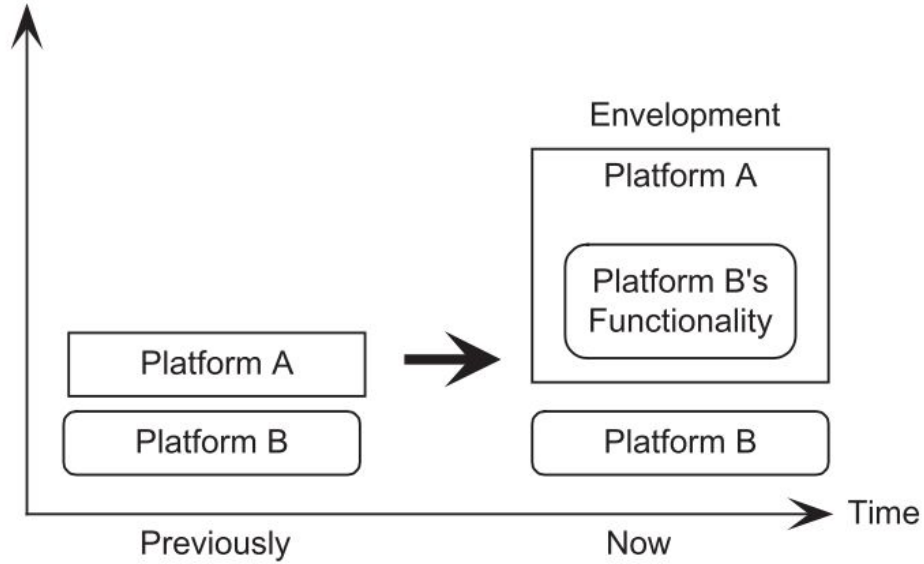
Competitive durability

- The ability to remain competitive over time.
- Strengthening network effects
- Adding new functionality



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.



Exercise

For two minutes. Note down some bullet points on paper on the following question:

What type(s) of platform is DHIS2?

How do the different characteristics relate to DHIS2? (e.g., architecture, multi-sidedness, network effects, etc.)

Guiding principles

Table 2.3 Summary of the Nine Guiding Principles in Platform Markets

Principle	Key Idea
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

Summary

Platforms

Two types of *digital platforms*:

- **Transaction platforms**
  Facilitates interaction between participants in a market or network

- **Innovation platforms** (software platforms or platform architectures)
  Facilitates innovation by enabling efficient use and reorganization of resources by a large audience.
  Main focus of this course

Core characteristics (beyond the architecture)

- *Innovation* or *software* platforms are characterized by an architecture with a stable core + APIs - enabling development of apps.
- Their emergence is driven by several environmental factors.

Additional characteristics:

- Multi-sidedness
- Network effects
- Multi-homing
- Tipping-points
- Lock-ins
- Competitive durability
- Envelopment

Related exam questions

1. DHIS2 as platform (40%) Answer both of the following questions.

- In the course we have talked about two different types of digital platforms. What type(s) of platform is DHIS2? Discuss how DHIS2 meets the definition and characteristics of the argued type of platform using concepts you see relevant from the course literature (e.g., architecture, multi-sidedness, boundary resources, network effects, etc.)

- Boundary resources is an important concept in the platform literature. Provide some examples of boundary resources for the DHIS2 software, and discuss their role in supporting design and innovation. You can use examples from the project work.

2. Modularization and software platforms (15%) Explain what is meant by modularization, and discuss how modularization is related to software platform architectures.