



gd

Gabriele Dragotto  
gabriele.dragotto@studenti.polito.it

# Software Business

---

## Lecture 1 - Introduction

### SOFTWARE

Software is the **set of instructions** or “programs” and data that tell a computer **what to do**

**INDESTRUCTIBILITY - TRASMUTABILITY - REPRODUCIBILITY**

**Types** of software:

1. APP SOFTWARE
2. SYSTEM SOFTWARE
3. UTILITY SOFTWARE

**Characteristics** of software:

1. IMMATERIAL
2. EXPENSIVE TO DEVELOP
3. INEXPENSIVE TO DUPLICATE
4. SHORT LIFE-CYCLE
5. NEED FOR LOCALISATIONS

### SOFTWARE BUSINESS

Software business is business of **selling software** (including systems software, application software and games) either **as licences or as services** and **services related to development** and deployment activities of this software

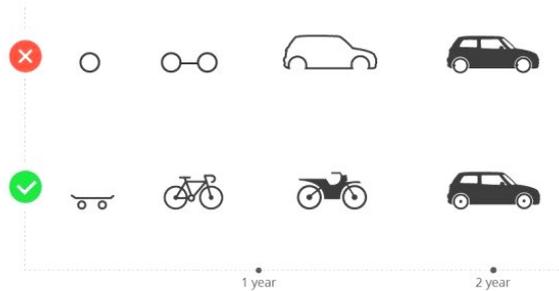
1. SaaS
2. SOFTWARE DEVELOPMENT
3. SOFTWARE LICENSES
4. SOFTWARE

## Key topics

- Business models
  - SWB classics
  - Cloud computing
  - Open Source
  - Platform strategy
  - Revenue models and pricing
  - ...
- SWB strategy and segments
- R&D
  - Differing approaches incl. Open innovation, crowdsourcing
- Software product business and Software service business
- Internationalization
- Managing SWB

### MINIMUM VIABLE PRODUCT

A Minimum Viable Product is a product with **good-enough features** to satisfy **early customers**, and to provide feedback for **future product development**.



**GOOD SOFTWARE** There are some key factors that help making good software business:

- PIONEERING
- WELL-KNOW BRAND
- MICRO TRANSACTION
- LOCK-IN
- FREEMIUM
- DISRUPTION
- UNIQUE VALUE PROPOSITION
- SERVICE SUPPORT
- QUALITY
- MATCHING DEMAND-OFFER
- MOBILIZING
- SOCIAL MEDIALIZING

**LOCK-IN**

In economics **vendor lock-in** makes a customer **dependent on a vendor** for products and services, unable to use another vendor without substantial switching costs.

## Lecture 2 - Business models

**BUSINESS MODEL**

A business model describes the **process** of how an organisation **creates, delivers, and captures value** in economic, social, cultural or other contexts.

*“Is a story explaining how a company works”*

**BUSINESS IDEA**

**What** is sold to **whom** and **how**.

**BUSINESS PLAN**

Written description of the **basic concepts** how a company is aiming at putting its **business idea into operation**.

**BUSINESS MODEL**

How to **extract value** from the market you are in. **3 main** components:

1. **TYPES OF GOODS**
  - 4:** Financial goods, physical goods, intangible goods or human services
2. **BUSINESS ARCHETYPE**
3. **REVENUE MODEL**

**REVENUE MODEL** A revenue model defines how a company is **compensated for its goods and services**. The compensation is usually, but **not necessarily, a payment**.

**SET OF REVENUE MODELS**

**ARCHETYPES**

- **CREATOR**  
Design or creates the product from **inputs**.
- **DISTRIBUTOR**  
Buys product and provide it to customers
- **LESSOR**  
Provide **right of use without owning the service**.  
**EMERGING: Financial lessor**
- **BROKER**  
Facilitates matching of buyers and sellers without owning the product.  
**EMERGING: On-demand economy**

	Types of goods/services offered			
	Financial	Physical	Intangible	Human
Creator	Entrepreneur	Manufacturer	Inventor	n/a
Distributor	Financial trader	Wholesaler, retailer	IP distributor	n/a
Lessor	Financial lessor	Physical lessor	IP lessor	Contractor
Broker	Financial broker	Physical broker	IP broker	HR broker

**HYBRID MODELS**

\*Cross subsidisation

	Types of goods/services offered		Types of goods/services offered		
	Intangible	Human	Physical	Intangible	Human
Creator	Inventor	n/a	Manufacturer	Inventor	n/a
Distributor	IP distributor	n/a	Wholesaler, retailer	IP distributor	n/a
Lessor	IP lessor	Contractor	Physical lessor	IP lessor	Contractor

(a) (b)

- **IP DISTRIBUTOR**  
If a vendor **provides usage rights for another vendor's** software or sells that vendor's IP, it acts as a distributor of intangible goods
- **IP BROKER**  
Large software vendors have **marketplaces** that their partners use to advertise **their solutions**.
- **FINANCIAL LESSOR**  
Software companies **lend money to customers** as to buy licenses.
- **RETAILER**  
Software companies **have retail stores** for selling tangible or intangible goods.
- **CONTRACTOR**  
Software companies **provides human capital** to customers.

TABLE 1

Comparing the three software vendors.

	SAP	Microsoft	Google
Main business model	Software as a product	Software as a product	Broker
Main revenue source	Maintenance and support	Indirect licence revenue from its partner ecosystem	Advertising
Emerging business model	Software as a service	Software as a service, broker	Software as a service, retailer, IP licensor for OSs
Main target customers	Businesses	Consumers	Consumers

## CASE: SAP



- ERP PROVIDER
- HYBRID MODEL
  - Providing products + services
- PLATFORM FOR PARTNERS
- IP DISTRIBUTOR
- INVENTOR AND CUSTOMISATION
  - For each company. Customisation covers cost from **R&D and inventing**
- SaaS ANCILLIARY REVENUE STREAM
- SAP EchoHub
  - Kind of app-store for partners

## CASE: MICROSOFT



- 66% SOFTWARE REVENUE
- SHIFT TO SaaS
  - From software to hardware. (Office365, Cloud, Drive)
- B2B and B2C
- PLATFORM STRATEGY
  - Software with hardware makers
- IP Broker
- ADS ANCILLIARY

## CASE: GOOGLE



- NOT SW-Co
  - By definition is not selling software
- ADS COMPANY
- SW & HW MANUFACTURER
- BUNDLING STRATEGY
  - Building an ecosystem

## SAAP

SaaS means that the company **delivers a copy** of the software to the customer, who **gets usage rights** but not ownership.

### ON-PERMISE

#### CHARACTERISTICS

- UPDATES
- RIGHT OF USE
- NEW SOFTWARE
- ADD-ONS

add-ons for additional **services** such as maintenance etc.

- HIGH-UP INVESTMENT

For customers. Shift from **on-permise to SaaS**

## SAAS

Software as a service is a software licensing and delivery model in which software is licensed on a **subscription basis** and is **centrally hosted**

### IP LESSOR

#### CHARACTERISTICS

- HOSTED DEPLOYMENT

SaaS means the software vendor **does not deliver the software**, but the **customer gets both access** to the software and **usage rights**

- SUBSCRIPTION BASED

- CARRIED COSTS

software sup-port, maintenance, and operation.

- BUNDLING
- INTANGIBLE
- PRODUCED AND CONSUMED SIMULTANEOUSLY
- HETEROGENEOUS
- DIVERSIFY REVENUE
- COMPLETE OFFER

**Four** main options:

1. **PRODUCT SMOOTHING**

*Training, maintenance, financing, warranty, smooth the product* without changing it.

2. **CUSTOMISATION**

Opposite to smoothing, **create new features**.

3. **ADAPTING**

Create **new features as to integrate** the product in a existing environment.

4. **REPLACEMENT**

Service that **replace the purchase of a new product**. **SaaS**

#### FOR CUSTOMERS

- FLEXIBLE
- LOWER COMPLEXITY
- ALIGNMENT WITH EXPERIENCE
- NO FAST-TIME PROCESSING
- DATA HOSTED EXTERNALLY
- CUSTOMISATIONS

Key Issue	Software-product Companies	Software-Service Companies
IP-rights	Very important	Less important
Product complementary	Very important	Less important
Returns from scale	A fixed-cost structure allows for higher marginal returns from scale	A variable-cost structure makes increasing returns from scale rare
Abstract knowledge and integrating technology	The company must be able to capture generic product knowledge so that the product can be used in a variety of context.	Knowing clients' characteristics is more important than knowledge abstraction.
Connections with users	Companies have long-term relationships. Users typically technology sophisticated	Companies have project-driven relationships. Users typically technologically unsophisticated

## MOVING FROM PRODUCTS TO SERVICES

- OPEN SOURCE
- NEW PRICING MODELS  
AD-based and SaaS subscription-based.
- LOWER GROSS-MARGIN  
Services require **labor-intensive IT services**.
- DIGITAL DISTRIBUTION
- DIFFERENT BUSINESS MODELS
- SERVICES GROW FASTER

Ways of dealing with the shift:

- A. BEST MIX PRODUCT/SERVICE
- B. SERVICIZING PRODUCTS
- C. PRODUCTISING SERVICES

	1 Traditional	2 Open Source	3 Outsourcing	4 Hybrid	5 Hybrid+	6 SaaS	7 Internet
<b>SOFTWARE</b>	\$4000/user (one time)	\$0/user	\$4000/user (one time)	\$4000/user (one time)			Ads
<b>SUPPORT</b>	\$800/user/year	\$1600/user/year	\$800/user/year	\$800/user/year	\$300/user/month	<\$100/user/month	Transactions Embedded (<\$10/user/Month)
<b>MANAGEMENT</b>			Bid <1300/user/month	\$150/user/month			
			@H	@C	@H	@C	

1. TRADITIONAL
2. OPENSOURCE
3. OUTSOURCING
4. HYBRID
5. SaaS
6. INTERNET

## BUNDLING

*Product bundling is offering several products for sale as one combined product*

1. **LOCK-IN**  
Keep customers off from the competition
2. **INTRODUCE NEW SOFTWARE**
3. **DISABLE EVENTUAL COMPETITORS**
4. **PENETRATE MARKET**

## PLATFORM

*Build on top of a common platform by **nurturing independent companies**, enabling them **to gain profits**.*

- **API**
- **CHEAP OR COSTLY?**



- **AGGREGATION**  
bring together a broad array of **relevant resources** and help users of the platform to connect with the most appropriate resources.  
**TRANSACTION - RESOURCES - HUB-SPOKE**
- **SOCIAL**  
Similar to aggregation, **enhance human** and long-term relationships  
**RELATIONSHIPS - COMMUNITIES**
- **MOBILISATION**  
They focus on **moving people to act together** to accomplish something beyond the capabilities of any individual participant.  
**RELATIONSHIPS - COMMUNITIES**

## API MODEL

*Provide **services trough access points**.  
A set of **clearly defined methods** of communication between various software components*

- **CHEAP AND FAST**  
Scale on the platform
- **PREMIUM PACKAGE**
- **API AS PRODUCT**

# Lecture 3 - Cloud Business models

## CLOUD COMPUTING

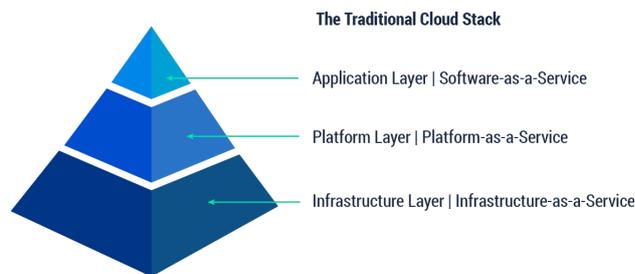
Cloud computing is a computing-infrastructure and software model for enabling **ubiquitous access** to **shared pools** of **configurable resources** which can be **rapidly provisioned** with **minimal management** effort, often over the Internet.

1. ON-DEMAND
2. UBIQUITOUS
3. SHARED POOLS
4. CONFIGURABLE RESOURCES
5. MINIMAL MANAGEMENT

Factors that made it **POSSIBLE**

- COMPUTING POWER
- DISTRIBUTED STORAGE
- VIRTUALIZATION
- HIGH-SPEED INTERNET
- IoT
- INEXPENSIVE SERVERS

## THREE CLOUD LAYERS



## OPEN SOURCE

Open-source is computer software with its source code made **available with a license** in which the copyright holder provides the **rights to study, change, and distribute the software to anyone and for any purpose**

### METHODOLOGY AND PHYLOSOPHY

**POPULAR LICENSES:** GNU, Mozilla public, Eclipse, Apache 2.0, BSD, MIT.

- **SOURCE AVAILABLE**
- **DERIVED WORKS**  
You can develop software starting from the basic one
- **USE, MODIFY, DISTRIBUTE**
- **NO DISCRIMINATIONS**  
Towards groups, industries, purposes
- **MYTHS**  
OSS is free, unreliable, licenses are too liberal,

### WHY GOING OPEN-SOURCE IN BUSINESS

1. FOCUS ON CONTENT
2. FAST DEPLOY
3. INTEROPERABILITY
4. CUSTOMISABILITY
5. COST SAVING

# Lecture 4 - Revenues and pricing

## REVENUE MODELS

### 1. LICENSING

An **fixed term or** perpetual license to access to the software usage rights.  
**VARIATION:** Maintenance, management, sunk costs, **ELA**.

### 2. SUBSCRIPTION

Bundled with maintenance and updates, typical is **SaaS**.  
**VARIATION:** Integration, implementation.

### 3. AD BASED

### 4. AFFILIATE MODEL

Promoting **links to relevant products** and collecting **commissions** on the sales of those products

### 5. FREEMIUM

Software for free, offer and charge for premium services and **enhanced versions**

### 6. LIMITED SEATS FOR FREE

Only few free licenses and then **pay for more!**

## SAAS REVENUE MODEL

Characteristics of the SaaS business

### 1. LATEST VERSION

### 2. NO INFRASTRUCTURE

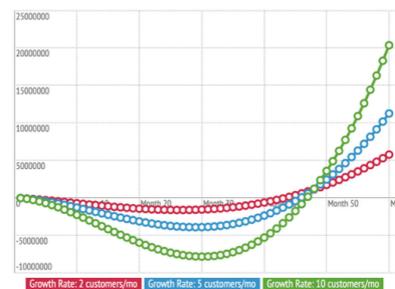
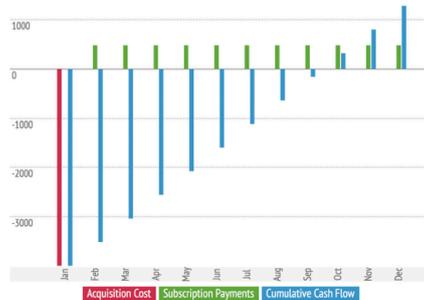
### 3. EASY-TO-DEPLOY

### 4. CUMULATIVE CASH FLOW

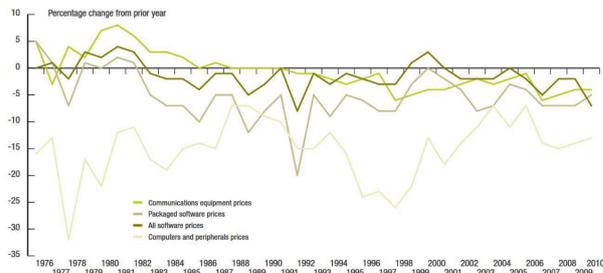
### 5. HYBRID REVENUE MODELS

### 6. DOUBT ABOUT SECURITY

## SAAS CASH FLOW



## CASE: SOFTWARE PRICING



Pricing trends ongoing

- **PAY PER USE**  
Customer charged according to the **metered usage** of software.  
*OVH and AWS*
- **COST BASED PRICING**  
Based on **cost accounting**. **ZERO marginal cost**  
*Outsourcing model*
- **DEMAND-DRIVEN**  
More demand – **higher price**
- **COMPETITION-ORIENTED**  
Depends on the **attractiveness** of the competing product  
*Google and cross-subsidisation*
- **ABILITY-BASED PRICING**  
Small firms use for free/small fee, **large firms pay more**
- **TOLERANCE FOR PAIN**  
Charge by guessing **willingness to pay**. **M&A Threat**
- **OUTCOME BASED**  
Providers are compensated for their role in assisting customers **in achieving successful outcomes** not for supplying products and services.  
*Financial, business-oriented or solution-oriented*
- **PRICE BUNDLING**

## SAAS REVENUE MODELS

### A. PAY PER USE

With pay per use, there's a **unit with a fixed price**, and a customer is **periodically charged** according to the units used

### B. SOFTWARE RENTAL

In software rental, the customer pays a negotiated **subscription fee** to use the software license for a certain **limited time**.

### C. SOFTWARE LICENSING

In packaged licensing, a customer buys a **single license for a single user or computer**, whereas in server-based licensing, the software is bought for a certain number of processors running the software.

Table 2. Comparison of revenue models from the software provider's viewpoint.

Model	Advantages	Disadvantages
Pay per use	<ul style="list-style-type: none"> <li>Doesn't require high initial investments</li> <li>Suits occasional usage</li> <li>Allows shift from capital investment to operational costs</li> <li>Offers low switching costs</li> <li>Lets customers test and evaluate software suitability</li> <li>Doesn't require installation, maintenance, or updates</li> <li>Lowers the expense for your IT personnel</li> <li>Doesn't require your own IT infrastructure</li> </ul>	<ul style="list-style-type: none"> <li>Offers the same price for all customers (non-negotiable)</li> <li>Carries data security concerns</li> <li>Requires estimating actual usage, which is difficult</li> </ul>
Software rental	<ul style="list-style-type: none"> <li>Offers negotiable pricing</li> <li>Doesn't require initial investments</li> <li>Makes costs are predictable</li> <li>Allows shift from capital investment to operational costs</li> <li>Doesn't require separate budgeting</li> <li>Works well when product is needed for a fixed period</li> <li>Lets customers test and evaluate software suitability</li> <li>Doesn't require installation, maintenance, or updates</li> <li>Doesn't require owning your own IT infrastructure</li> <li>Lowers the expense for your own IT personnel</li> <li>Lowers switching costs</li> </ul>	<ul style="list-style-type: none"> <li>Keeps costs the same whether or not the software is used</li> <li>Carries data security concerns</li> <li>Carries risk of software provider falling out of the market</li> </ul>
Software licensing	<ul style="list-style-type: none"> <li>Works well when product is needed for a long period</li> <li>Works well when the software is used for the firm's core business</li> <li>Offers negotiable pricing</li> <li>Keeps data stored and secure within own premises</li> </ul>	<ul style="list-style-type: none"> <li>Requires separate budgeting and decision-making</li> <li>Requires data storage and computing capacity</li> <li>Includes hidden costs</li> <li>Carries high switching costs</li> </ul>

## PRICING STRATEGIES FOR SOFTWARE VENDORS

- NETWORK EFFECT
- INTERNET
- SOFTWARE 3 PROPERTIES
- INTERNET - REDUCED TRANSACTION COST
- WINNER TAKE IT ALL
- LOCK-IN AND SWITCHING

In business because of close relationship between software and processes

### 1. FORMATION

3: Cost-based, demand-driven or competition driven.

### 2. STRUCTURE OF FLOW

3: Either the customer makes a **single payment** and thus obtains perpetual rights of use for the software or the pricing model requires regularly **recurring customer payments**. Also the **combination** of the two variants is possible

### 3. ASSESSMENT BASE

1: determining the **assessment or metric base** of the pricing model.

### 4. PRICE DISCRIMINATION

1: offering the same products to **different buyers at different prices**

### 5. PRICE BUNDLING

### 6. DYNAMIC PRICING STRATEGIES

---

## Lecture 5 - Software strategy

### STRATEGY

#### 1. VISION

Is concerned with the **future the organisation seeks to create**. It is an aspiration that will **enthuse, gain commitment** and **stretch performance**

#### 2. STRATEGIC GOALS

#### 3. ACTION PLANS

#### 4. KEY METRICS

Important questions:

- Is software the core of business or facilitator of it?
- Do you want to be a product or service company?
- Do you sell to individuals or enterprises?
- How broad (horizontal) and specialized (vertical)?
- How mainstream are you?
- Do you lead, follow, complement or something else?
- Can you generate recurring revenue stream?

### MARKETS

#### HORIZONTAL

- EASY-TO-CHOOSE
- AMBITIOUS
- HETEROGENOUS TECH STANDARDS
- EARLY ADOPTERS
- UNDERESTIMATED MARKETING EXPENSE

#### VERTICAL

- ENTERPRISE CONSUMERS
- PARTNERING
- PROOF-OF-CONCEPT
- CUSTOMISATION
- SCALING

### SEGMENTATION

Market segmentation is the process of **dividing a broad** consumer or business market into **sub-groups of consumers**

#### 1. A-PRIORI

Basing on **public data** available.

*Demographics, operating variables, purchasing power, situational factors, personal characteristics*

#### 2. NEEDS-BASED

Basing on **drivers** expressed by consumers

#### 3. VALUE-BASED

Basing on **value** expressed by consumers

### FOCUS AND SEGMENTS

#### 1. PROFESSIONAL SOFTWARE

Tailored solution for B2B markets. **++Servitisation —Productisation**

#### 2. ENTERPRISE SOFTWARE

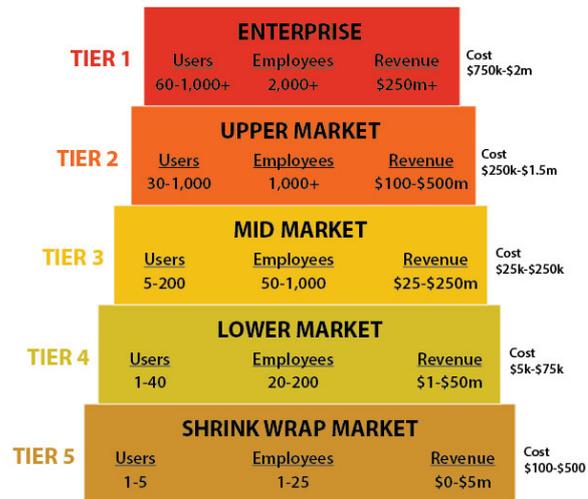
Common core with customisation

#### 3. MASS-MARKET

Homogeneous software little localised. **—Servitisation ++Productisation**

## CASE: ERP MARKET TIER

### Software Market Tier Chart



## METRICS

### SALES

1. REVENUE
2. ARPU  
Average revenue per user

### CUSTOMERS

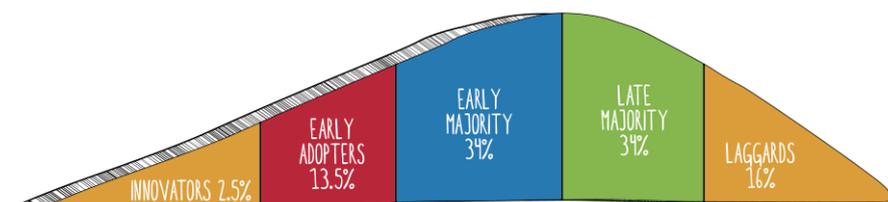
1. CHURN RATE  
the annual rate at which **customers stop subscribing** to a service
2. CAC  
Customer acquisition cost

### FINANCIAL

1. GROSS MARGIN
2. BURN RATE  
the rate at which a new company **spends its initial capital**.
3. OPERATIONS EFFICIENCY  
the **ratio between an output** gained from the business and an **input** to run a business operation

## INNOVATION DIFFUSION

### DIFFUSION OF INNOVATION MODEL



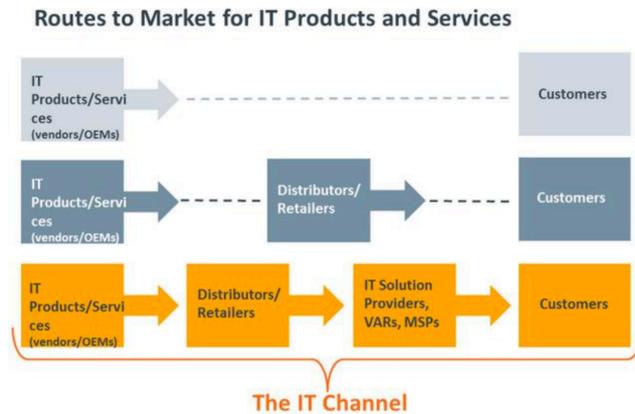
ESSENTIAL MARKETING MODELS [HTTP://BIT.LY/SMARTMODELS](http://bit.ly/smartmodels)

- EARLY ADOPTERS  
Focus on the **primary user**

## CHANNEL

Marketing channel is the **people, organizations, and activities** necessary to transfer the ownership of goods from the point of **production** to the point of **consumption**.

## IT CHANNEL



## ADS CHANNEL

Company \$1MM Sales (Ranked on % Most Effective)	% Currently Use	% Most Effective
Website	86.5%	38.2%
Email for marketing/promotion	70.2%	25.3%
Search engine optimization (SEO)	57.9%	20.8%
Social Media	78.1%	20.2%
Email for customer service	62.4%	16.9%
Videos and photos	60.1%	16.9%
Blogs and white papers	55.1%	14.0%
Paid banner ads on search engines and/or other websites	29.8%	13.5%
Paid search words	28.7%	12.9%
Online events (webinars and shows)	30.3%	12.4%
Email for prospecting	53.4%	11.8%
Online store or other ecommerce solution	23.6%	10.7%
Mobile apps	23.6%	7.9%
Mobile/SMS communication	18.5%	3.9%

- **SOCIAL MEDIA**

71% of consumers receiving a **quick brand response on social media** say they would likely recommend that brand to others

- **PERSONAL RECOMMENDATION**

72% of technology buyers look to **industry peers** while researching relevant content for B2B purchase decisions.

## PARTNERING

- **JOINT VENTURE**

**PRO:** Legally independent, joined by both firms, platform strategies.

- **MERGERS AND ACQUISITIONS**

**PRO:** Shape the market, secure market share, capabilities, platform strategies.

- **STRATEGIC ALLIANCE**

Two or more independent organizations pursue agreed goals.

**Equity or non-equity - Vertical or horizontal**

**PRO:** Extend operations and resources, learn, link with different distributors, internationalisation, platform strategies.

**Failing** because:

- **LACK OF TRUST**
- **LACK OF MANAGEMENT**
- **HIGH EXPECTATIONS**
- **LOW COMPATIBILITY**
- **POOR INTEGRATION**

# Lecture 6 - Internationalisation

	ADVANTAGES	RISKS
EXPORT	<ul style="list-style-type: none"> <li>• Internet facilitated</li> <li>• Leverage on <b>home country</b> economies of scales and supply</li> </ul>	<ul style="list-style-type: none"> <li>• No location advantages</li> <li>• <b>Bargaining of export</b> partners</li> <li>• Transportation and <b>trade barriers</b></li> </ul>
JOINT VENTURE	<ul style="list-style-type: none"> <li>• <b>Shared</b> risks</li> <li>• <b>Complementary resources</b></li> </ul>	<ul style="list-style-type: none"> <li>• <b>Management issues</b></li> <li>• <b>Where is good partner?</b></li> <li>• <b>Hard to coordinate</b></li> <li>• <b>Loose competitive advantage</b></li> </ul>
WHOLLY OWNED SUBSIDIARIES	<ul style="list-style-type: none"> <li>• <b>Integrated</b> coordination</li> <li>• <b>Full control</b></li> </ul>	<ul style="list-style-type: none"> <li>• Substantial <b>investment</b></li> <li>• <b>Where is good partner?</b></li> <li>• <b>Greenfield resource consuming</b></li> </ul>

## INTERNATIONALISATION

### 1. OUTSOURCE + MAKE

**Core competencies remain in-house** while non-core ones are **outsourced**.

2. COST SAVING
3. TAX SAVING
4. GREAT NEW MARKETS
5. ECONOMIES OF SCALE
6. GOVERNMENT INCENTIVES
7. DIGITAL DISTRUPTION

- **PARTNERING**
- **CUSTOMERS SEARCH**
- **STAKEHOLDERS**
- **ENVIRONMENTAL INSECURITIES**  
Political instability, economical one...

## ENTRY MODE TYPES

### 1. EXPORTING

Direct or indirect exporting, with or without **a third party**.

### 2. JOINT VENTURE

Two firms partner as to create a new **independent entity**

### 3. SUBSIDIARY

Daughter company

### 4. REPRESENTATIVE OFFICE

**Explorative** tasks in new market

## ENTRY DECISIONS

Entry decision are made on basis of:

- **EFFECTUATION BASED**  
Based on **existing resources** and **means-driven approach**
- **CAUSATION BASED**  
Based on **goal-driven approach**. Predictability and **research-oriented**

## PATHWAYS

### • BORN-GLOBAL

**Focus on "leading" markets**, might ignore home country

### • TRADITIONAL

**From domestic market** to nearby markets

**Relationships** play a fundamental role in **exporting to new markets**, both the **formal and the informal** ones.

# Lecture 7- Research and development

## INNOVATION MANAGEMENT

Apply knowledge and make discoveries as to develop and **improve new or existing products or procedures**

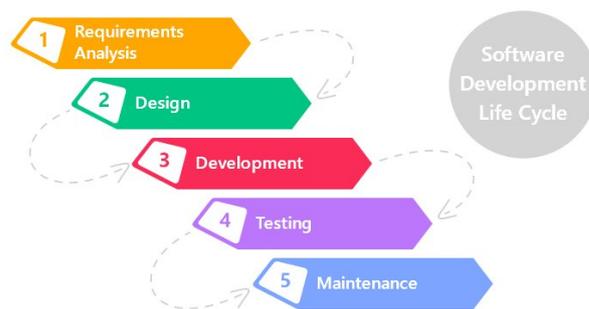
### SDP

Software development process is the process of **dividing software development work** into distinct phases to improve design, product management, and project management

#### TIME SCOPE COSTS

### WATERFALL

The waterfall model is a **sequential development approach**, in which development is seen as **waterfall flow**



### AGILE

"Agile software development" refers to a group of software development methodologies based on **iterative development**, where requirements and solutions evolve via collaboration **between self-organizing cross-functional teams**.



### MOCKUPS

In manufacturing and design, a mockup, or mock-up, is a **scale or full-size model of a design or device**, used for teaching, demonstration, design evaluation, promotion, and other purposes.

### FUN FACTS

1. **EUROPEANS**  
Maximize quality
2. **AMERICAN**  
Go for business
3. **ASIANS**  
Standardize and develop

## SOURCING INNOVATION

### 1. CROWDSOURCING, HACKATON, COMPETITIONS

**External inputs. Split into micro tasks** that enables **diversity** of different contributors, allows **experimentations**, is **cheap easy and fast**.  
**Channel, quality, selection**

2. CONTINUOUS OR UNANTICIPATED R&D
3. FOCUSED OR UNFOCUSED

## INNOVATION PROCESS

### A. SCIENTIFIC MOMENT

Get the thing from sources. **Shared IP or proprietary**

### B. IMPLEMENT

Into your product

### C. BUSINESS PHASE

Market your product.

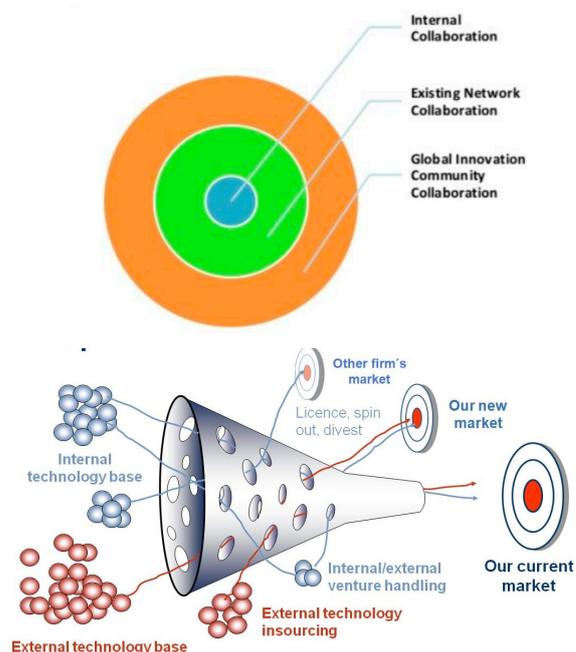
## SHARED IP

- NETWORKING
- ALLIANCES
- NEW INSIGHTS
- INNOVATION FROM OUTSIDE
- NEW MARKETS
- NEW CONTRIBUTORS

## OPEN INNOVATION

*a distributed innovation process based on **managed knowledge flows across organizational boundaries**, using pecuniary and non-pecuniary mechanisms and resources*

### LICENSE - SPINOFF - SALES



## PRODUCT INVENTION OFFERING

- ROUTINE DESIGN
- INNOVATION
- IMPROVEMENT
- EXAPTATION

# Lecture 8 - Managing

## STRATEGIES



## 5 FORCES

Porter's five forces help identify the **attractiveness** of an industry in terms of profitability.

## SECURING IP

### 1. COPYRIGHTS

Does not include **ideas, methods, approaches**

### 2. PATENTS

The right to **exclude others** from making, using, offering for sale, or selling.

**TEDIOUS TO APPLY - SHARING AGREEMENT OI - JOINT VENTURES**

### 3. TRADEMARK

**a word, name, symbol, or device** that is used in trade with goods to indicate the source of the goods

## PRIORITIES

The priority of management area	Professional software services	Enterprise solutions	Mass-market software
I	<b>People management</b>	Partnering	Marketing
II	Human resource assignment	Service strategy	Partnering
III	Development	Marketing	Globalisation
IV	Marketing	<b>People management</b>	<b>People management</b>
V	Partnering	Development	Development

# Appendix: International operations of software firms

## TOPICS

- SOFTWARE COMPLEXITY AND OFFSHORING
- INTERNATIONALISATION OF PLATFORM PROVIDERS

## FIRST PAPER: COMPLEXITY

1. Relationship between complexity and offshoring?
2. How do software firms coordinate their offshoring activity to cope with the software in question?

### BACKGROUND

- DEVELOPMENT AND DISTRIBUTION IS GLOBAL
- GEOGRAPHICAL DISTRIBUTION IS NOT A PROBLEM
- LOCALISATION COMPLEXITY

Identify the **key requirements** that are **sensitive to local context**. The software has to be **maintained globally**

- VARIANCE IN CONSUMERS NEED

## OFFSHORING

*Moving certain activities to another country, either a foreign unit or a third-party supplier*

- OUTSOURCING FROM THIRD PARTIES

Using **foreign** distributor as to access to a **new knowledge base**

- INSOURCING

Use own personal to cope with **different requirements**

## SOFTWARE COMPLEXITY

*Software complexity is **low for mass markets** and tends to be **high for localised markets***

- INTERNAL COMPLEXITY

Type and numbers of dependencies within the software codes

- EXTERNAL COMPLEXITY

Dependencies of the software with **its development and use environment**.

## FINDINGS

- CORE-COMPETENCES IN HOUSE

Development **always in house**.

- OUTSOURCED ACTIVITIES

4: Localisation, customisation to specific needs, integration with existing IT systems and product supports

- FIRM ORGANISATION

**is affected** on how competencies are outsourced

- LOW-COMPLEXITY BUSINESSES. LOW-OUTSOURCING

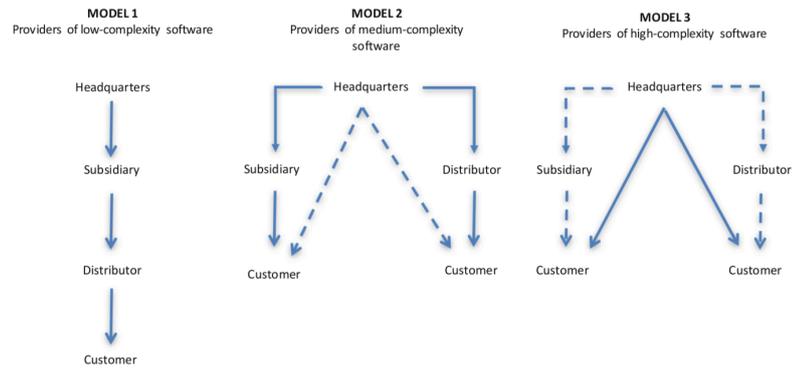
Outsource only a **limited portion of activities** (*support*)

- MEDIUM-COMPLEXITY BUSINESSES. HIGH-OUTSOURCING

Outsource typically **localisation, customisation, integration and support**.

- HIGH-COMPLEXITY BUSINESSES. LIMIT

Outsource very very **small activities**, because of the need to know the **client requirements**.



## SECOND PAPER: INTERNATIONALIS ATION

How digitalisation with **disruptive services** influences **foreign expansion and operations**

### BACKGROUND

- DIGITALISATION
- PLATFORMS PROVIDE ALTERNATIVE TO PHYSICAL
- DIGITAL SERVICES ARE DISRUPTIVE
- HOW DISRUPTIVENESS INFLUENCES INTERNATIONAL EXPANSION

### LAYERED ARCHITECTURE

1. CONTENT LAYER
2. SERVICE LAYER
3. NETWORK LAYER
4. DEVICE LAYER

### FINDINGS

#### • LAYER INTERDEPENDENCIES

When network layer was developed, game company was able to **expand its market**

#### • NETWORK INTERNATIONALISATION

Firms are required to internationalise as to **gain access to new layers.**

#### • MOVE MORE ACTIVELY

When all the layers are well developed.

#### • ECOSYSTEMS

Digital services form an ecosystem within which firms may come together, interact, and offer different kinds of services, components, and technologies