

The Software Industry

Prof. Dr. Dirk Riehle

Friedrich-Alexander University Erlangen-Nürnberg

COSS B01

Licensed under CC BY 4.0 International

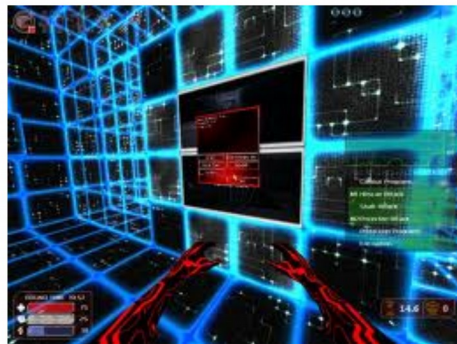
Agenda

1. Definition (software)
2. The software industry
3. A (very) short history
4. The main players
5. Software products
6. Software platforms
7. Software ecosystems

1. What is Software?

Software Definition

- Software (technical definition)
 - Is a set of instructions to make computers act
 - Comes in equivalent forms (source, binary)
 - Is formed into components and programs
 - Programs can be built from components
- Software (economic definition)
 - Is a digital good that can be sold



Software as a Product

- **A Product**
 - Is a man-made artifact sold to customers in a market
 - Has a life-cycle
 - Is born,
 - Grows and matures,
 - Eventually dies
- **A software product** is a product that
 - Is intellectual property
 - Is non-physical, does not rot
 - Has near-zero copying costs
 - Is extremely malleable

2. The Software Industry

The Software Industry

- The software industry
 - Is the set of business that provide
 - Software products and
 - Software services such as
 - Operating services
 - Consulting services
 - Development services
 - Implementation services
 - to other industries as well as itself
- The software industry
 - Is highly concentrated
 - Is highly internationalized
 - Has strong network effects
 - Has a high speed of innovation
 - Is rapidly expanding into new domains

The Software Industry in 2016 [1]

market capitalization	total	\$1.298 trillion
	median	\$744.2 million
	highest	\$415.4 billion (Microsoft)
	lowest	\$177700 (Innovaro Inc.)
earnings per share	median	\$0.20
	highest	\$13.23 per year (IBM)
	lowest	– \$3.40 per year (Wave)
dividend yield	mean	8.913%
	highest	170.3% (Aware)
	lowest	0.07106% (FICO)

[1] <https://www.wolframalpha.com/input/?i=how+big+is+the+software+industry>

“Software is eating the world”
Wall Street Journal
2020-08-20



The CEO Interview

“Industrial companies are in the information business whether they want to be or not.”

—Jeff Immelt

McKinsey&Company



Change and Innovation at the Speed of Software

- Products increasingly include software components
 - Hardware components traditionally have long innovation cycles
 - Software has a significantly faster innovation cycle
 - Innovation speed is continuously increasing
- Products are being adapted to take advantage of software
 - Products can be reconfigured at speed of software
 - Products evolve at speed of software innovation
- Examples increased innovation speed
 - Cars are fully delivered with features switched off
 - Cars evolve at speed of over-the-air update
- Software is eating established industries and products

Societal Significance of Software and Software Systems

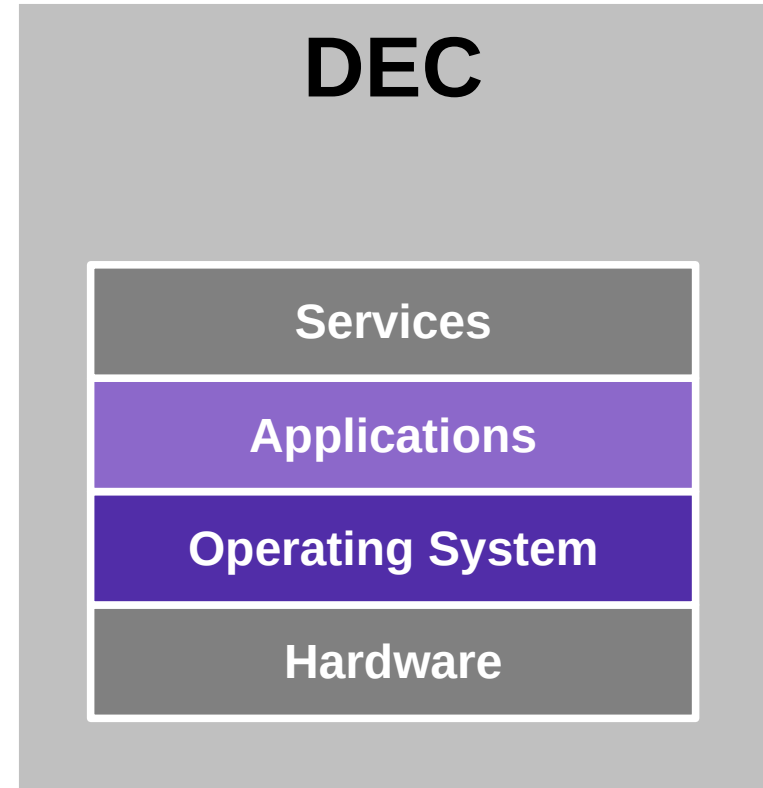
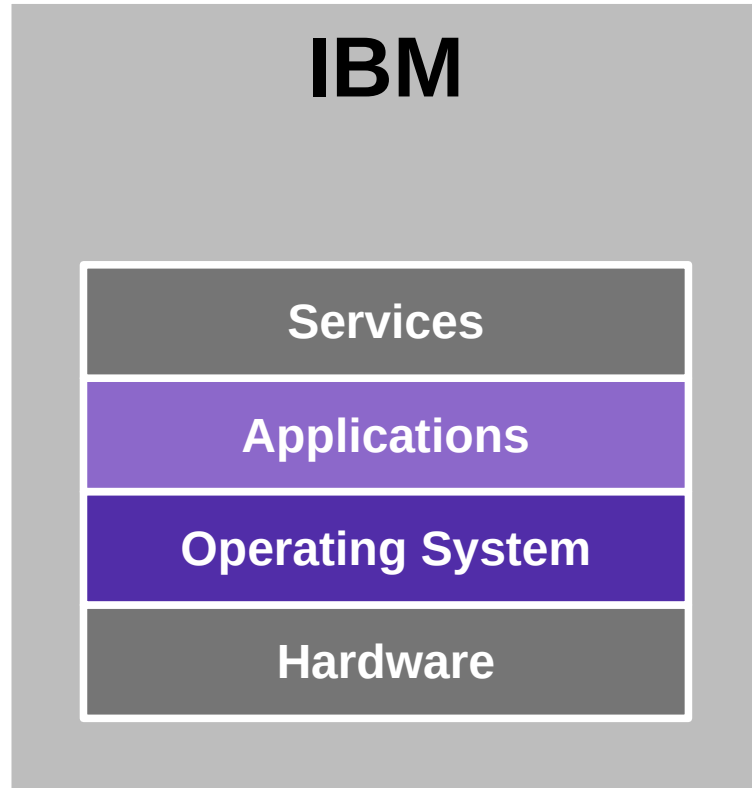
- Software and software systems are changing society
 - Empowerment vs. stratification
 - Enlightenment vs. misinformation
 - Sociability vs. isolation
- Internet and email have become basic utilities

3. A (Very) Short History

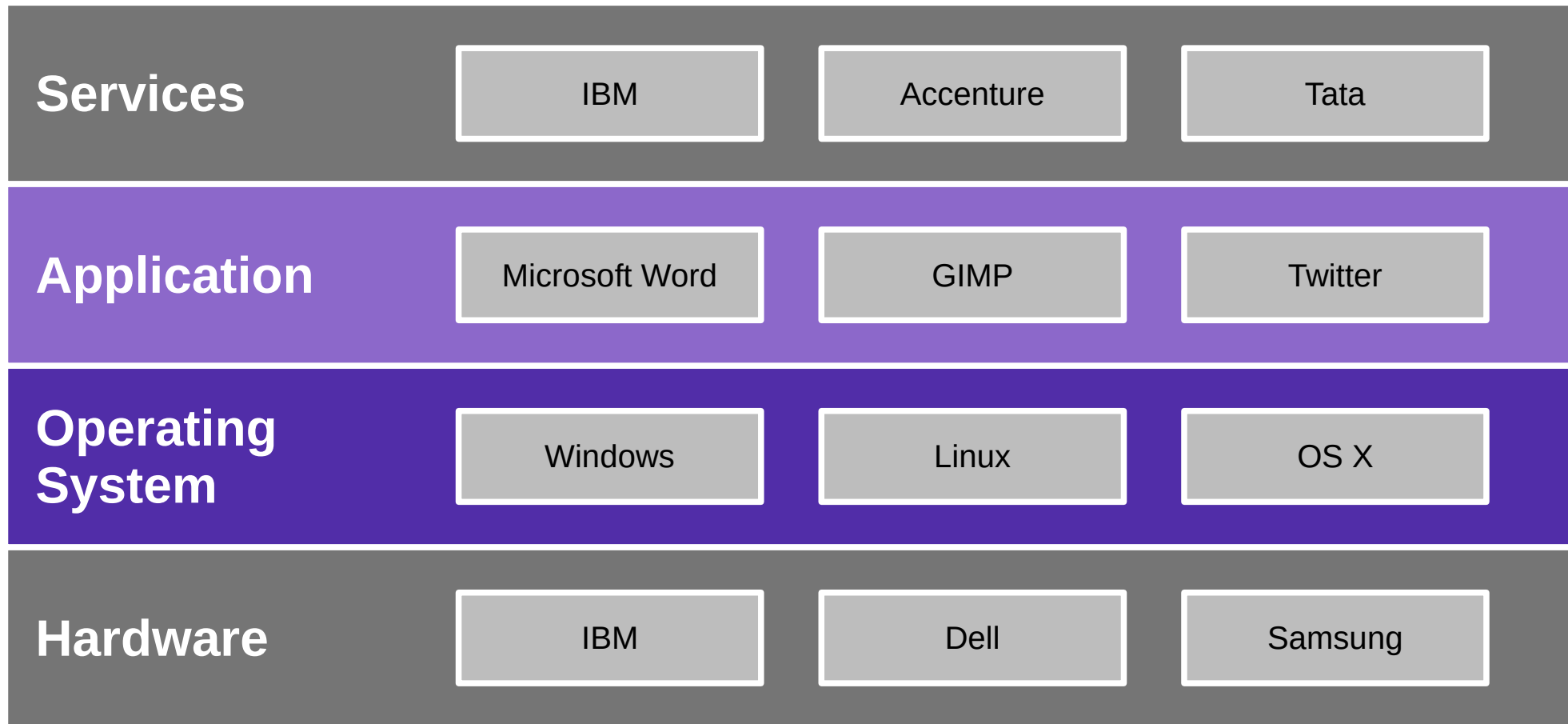
Short History of the Software Industry

- 1959
 - First mentioning of term “software”
- 1969
 - US Dept. of Justice separates hard- from software in landmark decision
- 1980ties
 - From vertical to horizontal integration; growth of platforms and ecosystems
- 1990ties
 - Centralization, dominance of Windows
- 2000ties
 - Diversification, multiple platforms; growth of open source software
- 2010ties
 - Back to vertical integration in the form of cloud computing

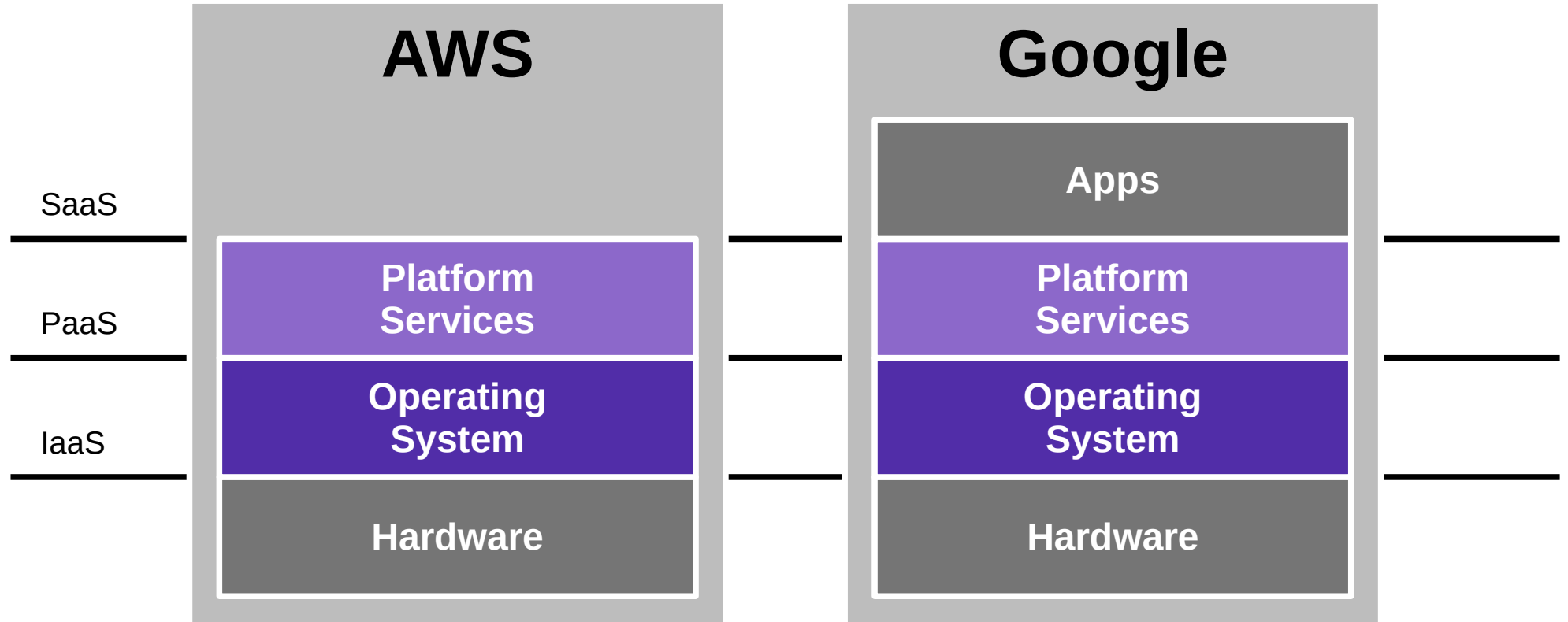
Vertical Integration (Until 1980ties)



Horizontal Integration (Since 1990ties)



Cloud Computing (Since 2000ties)



[illegible]

	Scheduling & Orchestration	Coordination & Service Discovery	Remote Procedure Call	Service Proxy	API Gateway	Service Mesh
Orchestration & Management	 kubernetes CNCF Graduated Nomad VOLCANO	 CoreDNS CNCF Graduated etcd CNCF Incubating NACOS NETFLIX OSS Eureka	 gRPC CNCF Incubating THRIFT SOFA RPC JAX-RS	 envoy CNCF Graduated CONTOUR CNCF Incubating GAMBEL HAProxy NGINX GREENVIEW PORTER SHIPPER NOVA Tengine traefik	 Kong krakenD Platform MuleSoft Sentinel Tyk WSO2 API Manager	 LINKERD CNCF Incubating NETFLIX OSS Zuul istio Consul Grey Matter Kuma POLARIS Istio POLARIS POLARIS

[illegible]

The diagram illustrates the integration of various open-source tools into a cloud-native architecture, categorized into four main functional areas:

- Provisioning:** Tools like Airship, Ansible, Apollo, and Terraform are used to provision infrastructure.
- Automation & Configuration:** Tools like BOSH, Cadence, and CFEngine are used for automation and configuration management.
- Container Registry:** Tools like Harbor, Dragonfly, and Quay are used for container image management.
- Security & Compliance:** Tools like Falco, Sysdig, and Snyk are used for security and compliance monitoring.
- Key Management:** Tools like Spiffe, SPIRE, and Vault are used for key management.

The diagram shows how these tools are integrated into a cloud-native architecture, with each tool having a designated role in the overall system. The tools are organized into a grid, with each tool's logo and name clearly visible. The grid is divided into four main sections, each representing a different functional area. The tools are arranged in a way that shows their relationships and how they work together to manage and secure a cloud-native environment.

The diagram illustrates the integration of various open-source tools into a cloud-native architecture, categorized into four main functional areas:

- Provisioning:** Tools like Airship, Ansible, Apollo, and Terraform are used to provision infrastructure.
- Automation & Configuration:** Tools like BOSH, Cadence, and CFEngine are used for automation and configuration management.
- Container Registry:** Tools like Harbor, Dragonfly, and Quay are used for container image management.
- Security & Compliance:** Tools like Falco, Sysdig, and Snyk are used for security and compliance monitoring.
- Key Management:** Tools like Spiffe, SPIRE, and Vault are used for key management.

The diagram shows how these tools are integrated into a cloud-native architecture, with each tool having a designated role in the overall system. The tools are organized into a grid, with each tool's logo and name clearly visible. The grid is divided into four main sections, each representing a different functional area. The tools are arranged in a way that shows their relationships and how they work together to manage and secure a cloud-native environment.

© 2015 Pearson Education, Inc. or its affiliate(s). All rights reserved.

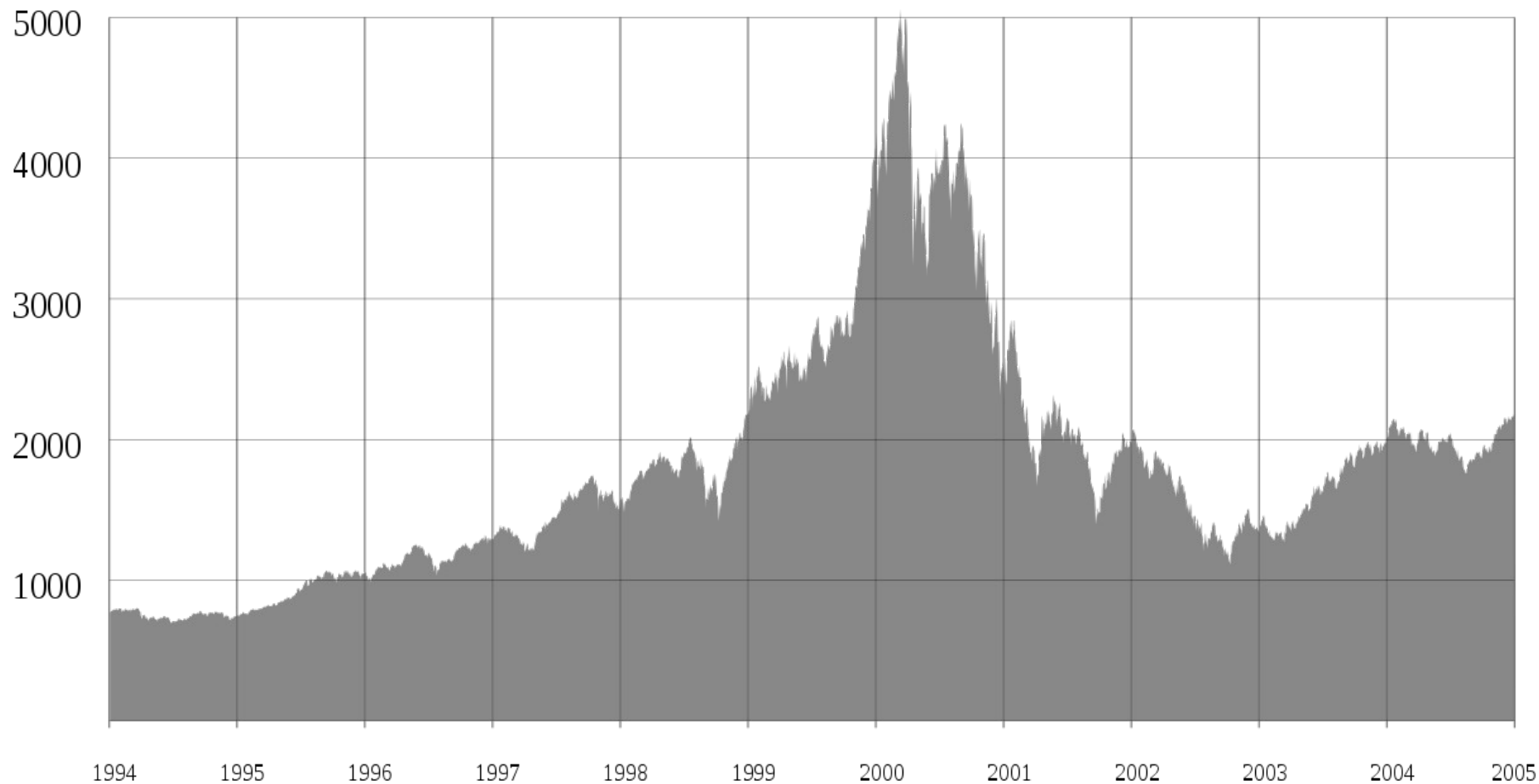
[illegible]

© 2015 Pearson Education, Inc. or its affiliate(s). All rights reserved.

[illegible]

© 2015 Pearson Education, Inc. or its affiliate(s). All rights reserved.

The “Dot-Com” Bubble and Burst (1995-2000)



[1] See https://en.wikipedia.org/wiki/File:Nasdaq_Composite_dot-com_bubble.svg

4. The Main Players

The Main Types of Industry Players

- Standard product providers
 - (Independent) software vendors (ISVs)
 - Produce software products (“standard software” or “commercial off-the-shelf software”)
 - Software service providers (e.g. “Internet companies”)
 - Operate any form of software (and hardware)
- Software consulting firms
 - Development services firms
 - Produce custom software
 - Implementation services firms
 - Configure software products for use by customers
- Non-profit organizations
 - Standards organizations
 - Certification agencies
 - Regulatory bodies

Top 10 Independent Software Vendors (ISVs) in 2019 [1]

Rank ↕		Organization ↕	Sales (B\$) ↕	FY ↕	Market cap (B\$) ↕	Headquarters ↕
1		Microsoft	118.2	2019	946.5	Redmond, WA, US
2		Oracle	39.6	2019	186.3	Redwood City, CA, US
3		SAP	29.1	2019	134.9	Walldorf, Germany
4		Adobe Inc.	9.5	2019	132	San Jose, CA, US
5		Salesforce	13.3	2019	120.9	San Francisco, CA, US
6		VMware	9.0	2019	77.2	Palo Alto, CA, US
7		Intuit	6.4	2019	66.8	Palo Alto, CA, US
8		ServiceNow	2.6	2019	42.9	Santa Clara, CA, US
9		Workday	2.8	2019	41.7	Pleasanton, CA, US
10		Dassault Systèmes	4.1	2019	39.2	Vélizy-Villacoublay, France

Software Vendors vs. “Internet Companies” [1]

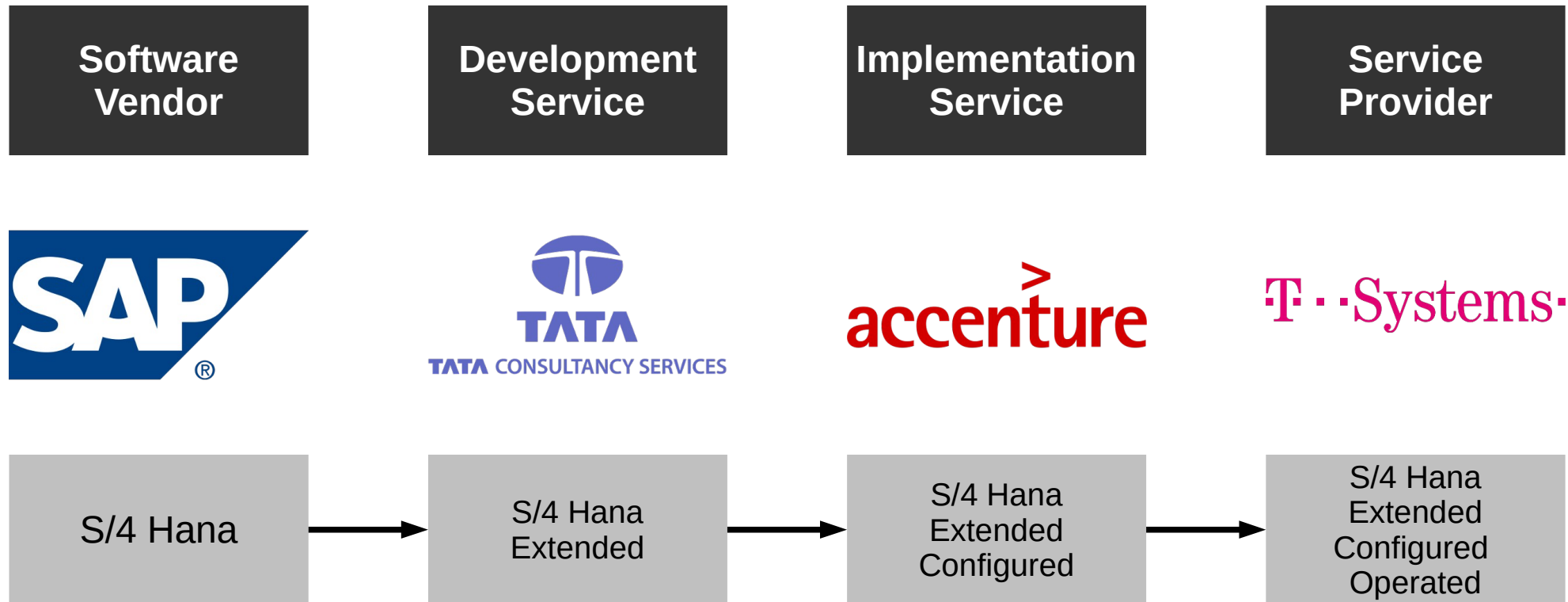


[1] See https://en.wikipedia.org/wiki/List_of_largest_Internet_companies

Software Vendors vs. Service Provider / Operator

- Software vendor
 - Product is a licensed-out artifact
- Examples (before cloud offering)
 - Microsoft
 - Oracle
 - SAP
 - Adobe
- Service provider / operator
 - Product is a standardized service
- Examples
 - Amazon Web Services (if it was broken out)
 - Various Google services
 - Salesforce
 - Facebook

Software Vendors, Consulting Firms, and Service Providers

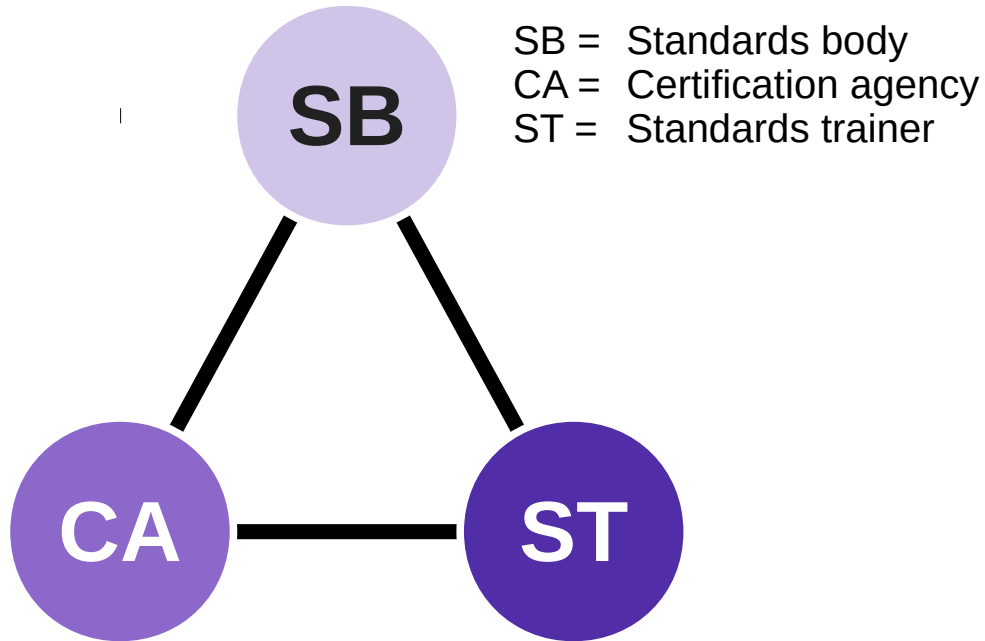


Standards Organizations

- A standards organization
 - Is a public (often non-governmental) organization financed by industry
 - That serves as meeting point and platform to define industry standards
 - Conformance to which may be required before admission to market
 - Is one player (of three) in standards and certification processes
- Examples
 - ISO (“International Standards Organization”)
 - VDE (“Verband der Elektrotechnik”)
 - VDA (“Verband der Automobilindustrie”)

Certification Agencies

- A certification agency
 - Is a non-profit organization that provides certification services for given standards



Regulatory Bodies (Regulators)

- A regulatory body
 - Is a public organization or government agency (state-level, federal-level, union-level)
 - Which by way of laws and directives regulates industries and industry player behavior
 - To protect the public by preventing undesired behavior and enforcing desired one
- Examples
 - European Union
 - European commission → Data protection (GDPR) → Enforcement by Information Commissioner's Office (ICO)
 - U.S.A.
 - Federal and state governments → Antitrust law → Enforcement by Federal Trade Commission (FTC), DoJ

5. Software Products

Consumer vs. Enterprise Customers

- Retail customers (B2C)
 - Are willing to trade time for money
- Enterprise customers (B2B)
 - Are willing to trade money for time

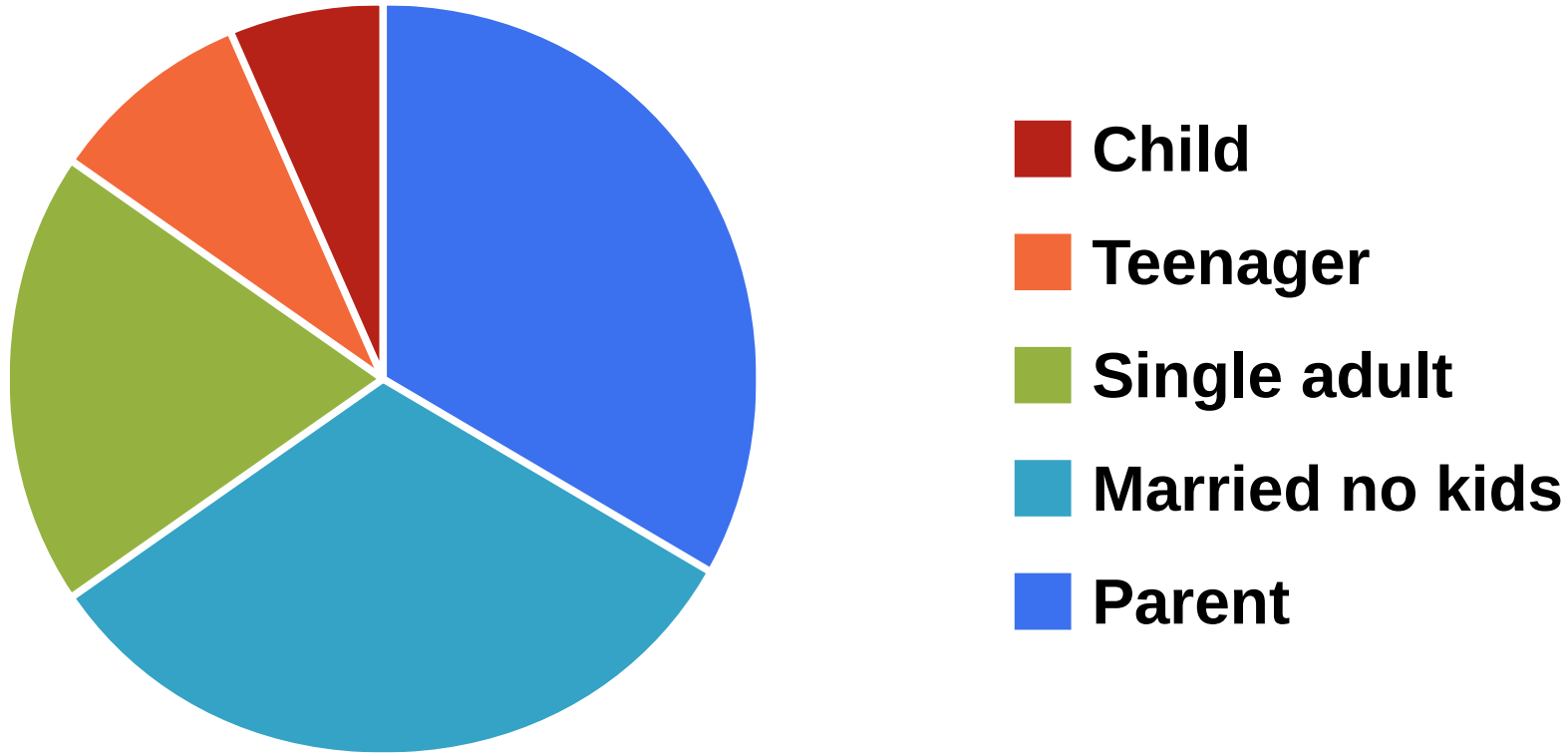
Software Products (Recap)

- **A software product**
 - Is a digital good (software, intellectual property)
 - Is non-physical, does not rot
 - Has near-zero copying costs
 - Is a man-made artifact sold to customers in a market
 - Has a life-cycle (is born, grows and matures, eventually dies)
 - Is both extremely malleable and hard to change

Consumer vs. Enterprise Software Products (and Markets)

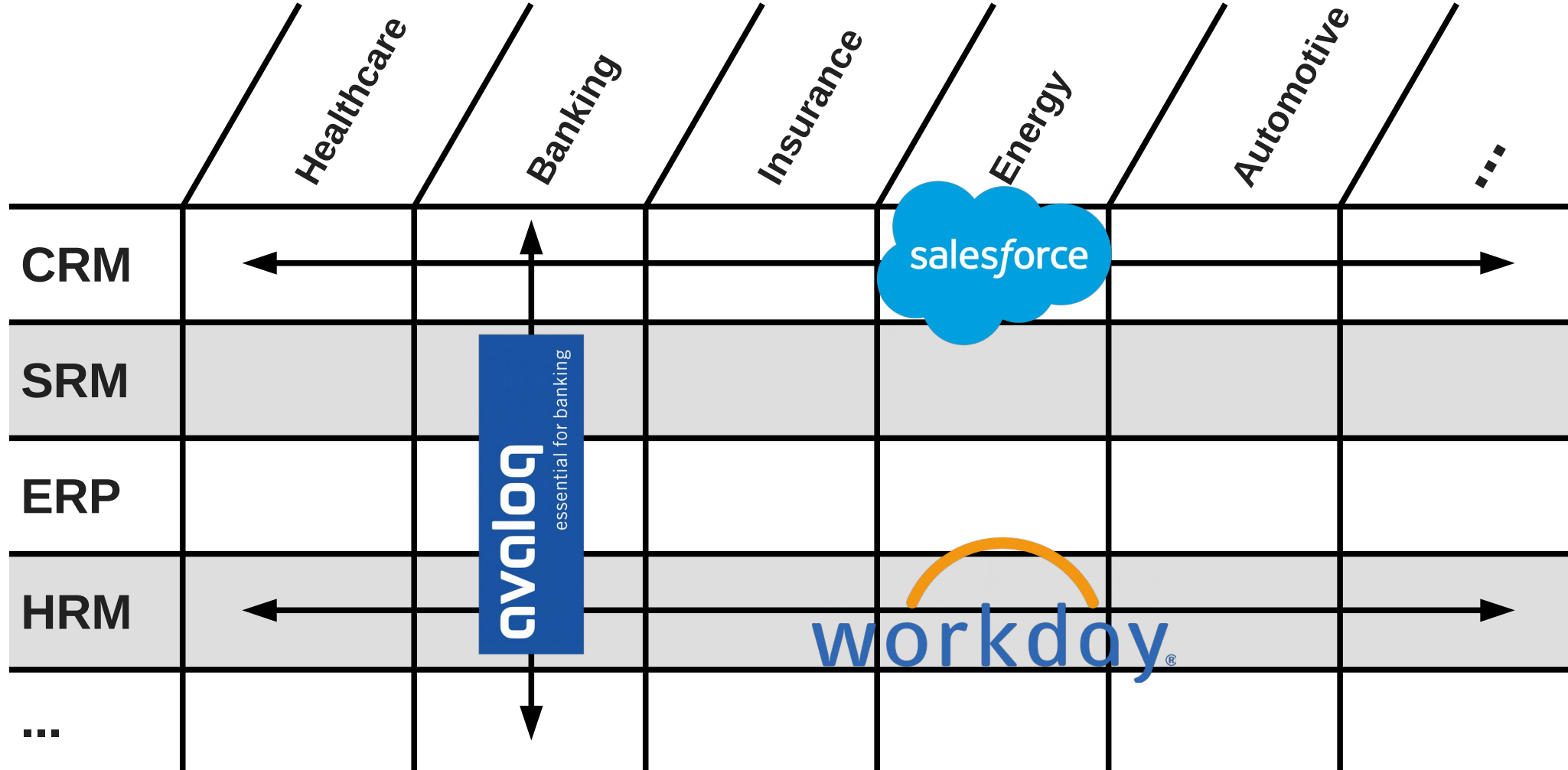
- Consumer (also: retail) products
 - Pricing
 - Comparatively cheap
 - Often free, then subsidized
 - Segmentation
 - Usually by demographics, e.g.
 - By age group
 - By gender
 - Adoption
 - Out of the box
- Enterprise software products
 - Pricing
 - Into million Euros
 - Often the real product behind consumer software
 - Segmentation
 - Horizontal vs. vertical, e.g.
 - By business function
 - By industry
 - Adoption
 - May require implementation project

Consumer Market (Segments) [1]

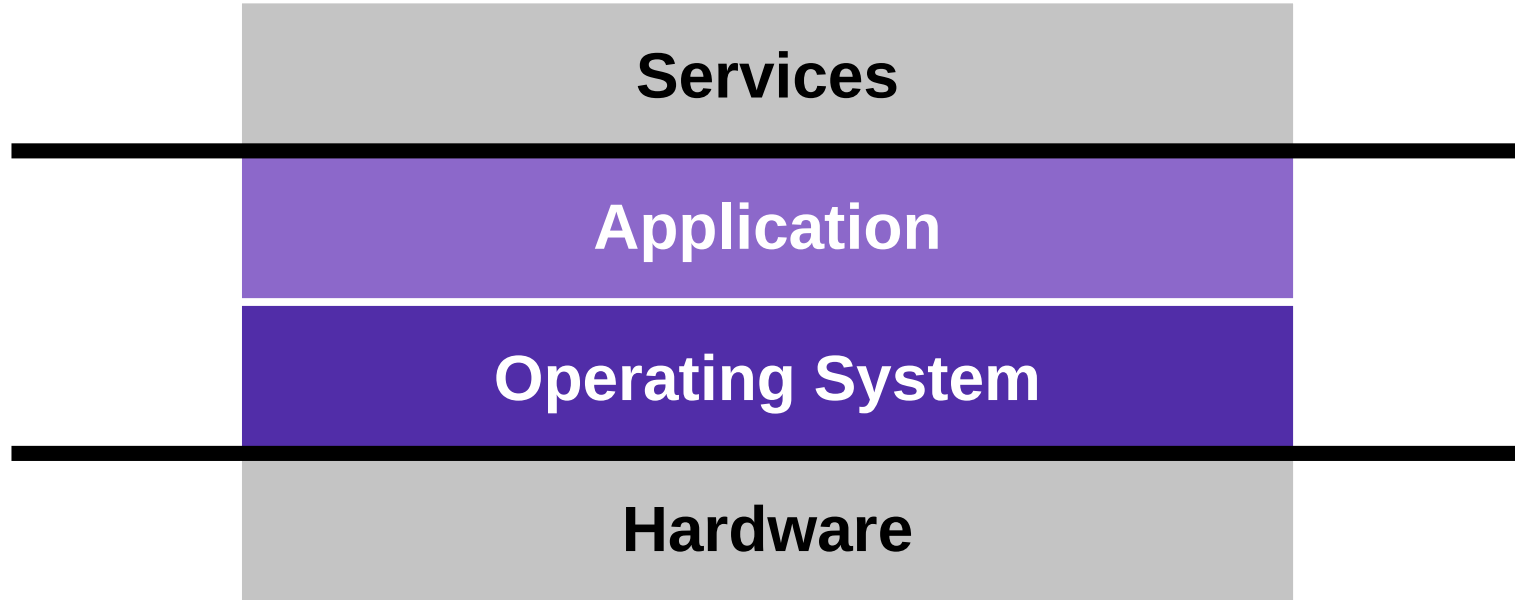


[1] Called customer segments in the business model canvas

Enterprise Software Market Segmentation

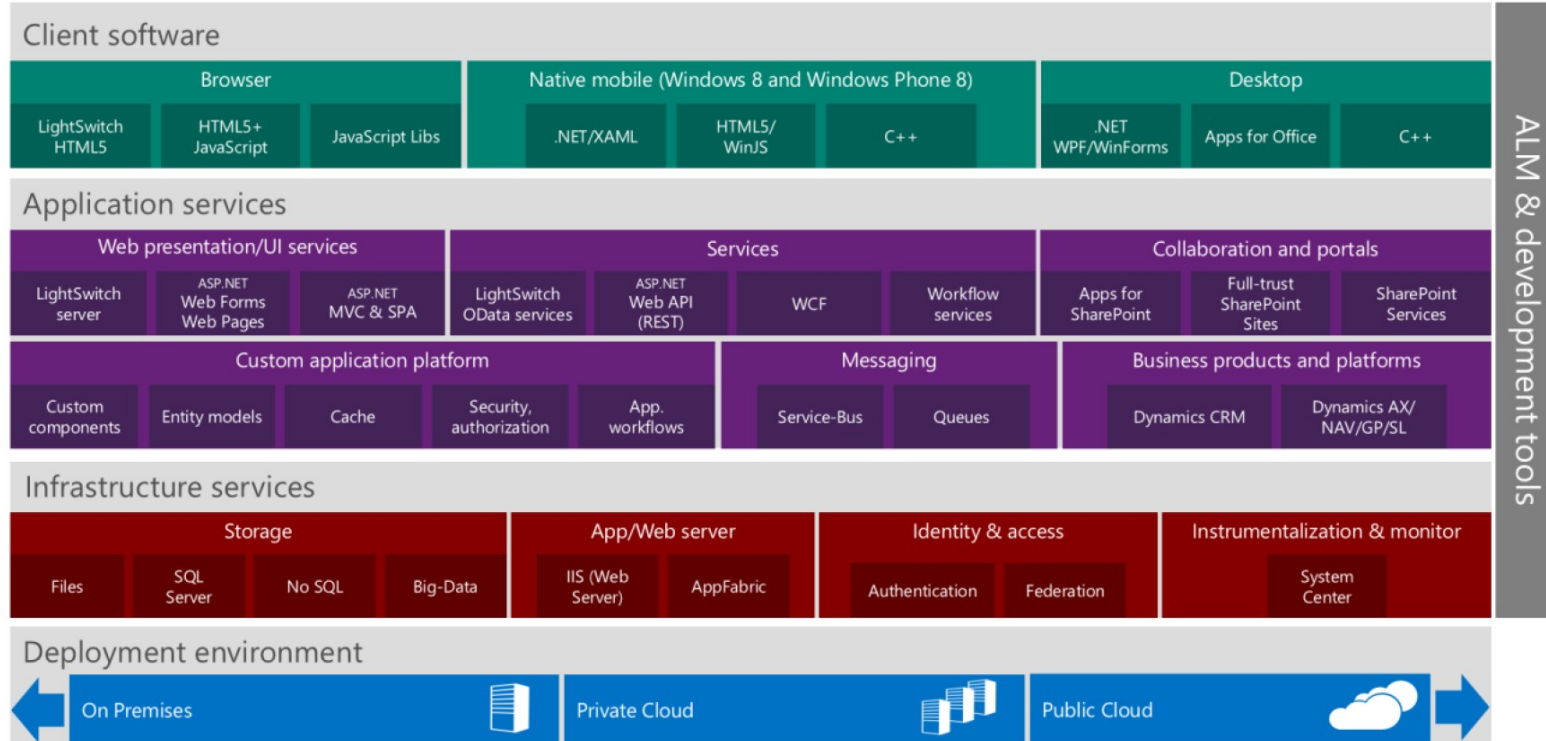


Customers Want to Buy a “Solution”



Technology Stacks

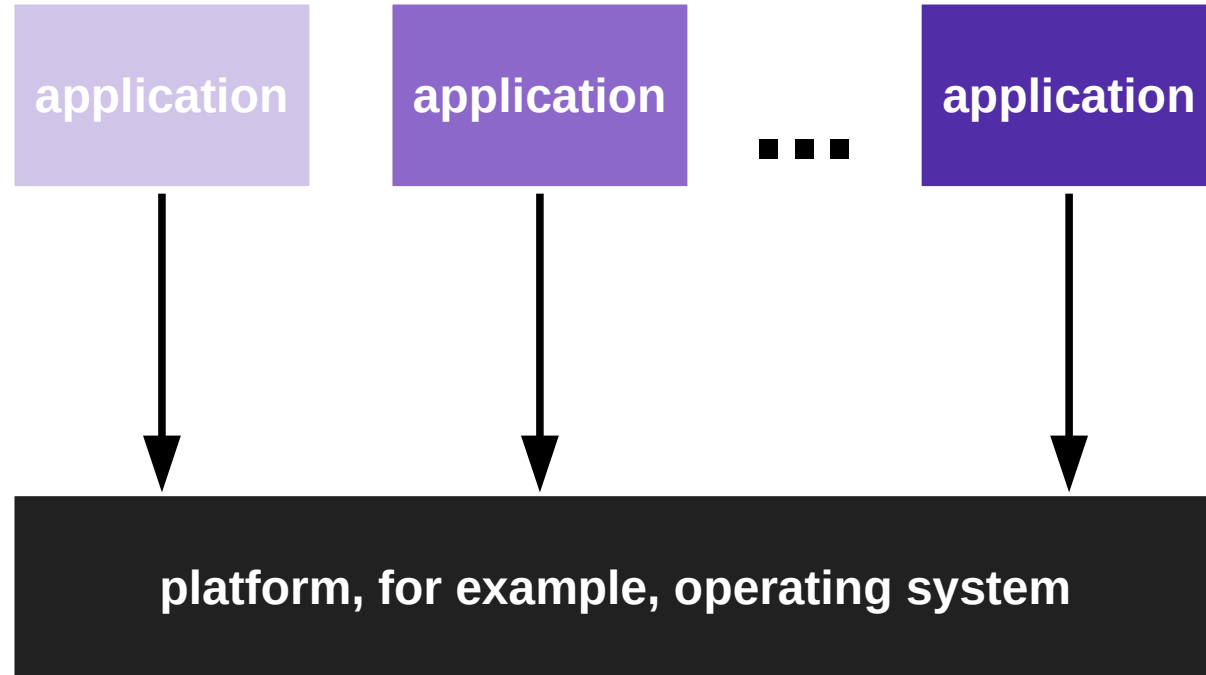
Microsoft Development Platform Technologies



6. Software Platforms

Categories of Software Products

- **Applications**
 - Software that is not built upon
 - Software that delivers immediate business value
 - Top-layer of the solution stack
- **Platforms**
 - Software that is built upon
 - Software that supports other software in delivering business value
 - Everything that is not the top layer
- **Why does everyone want to be a platform?**

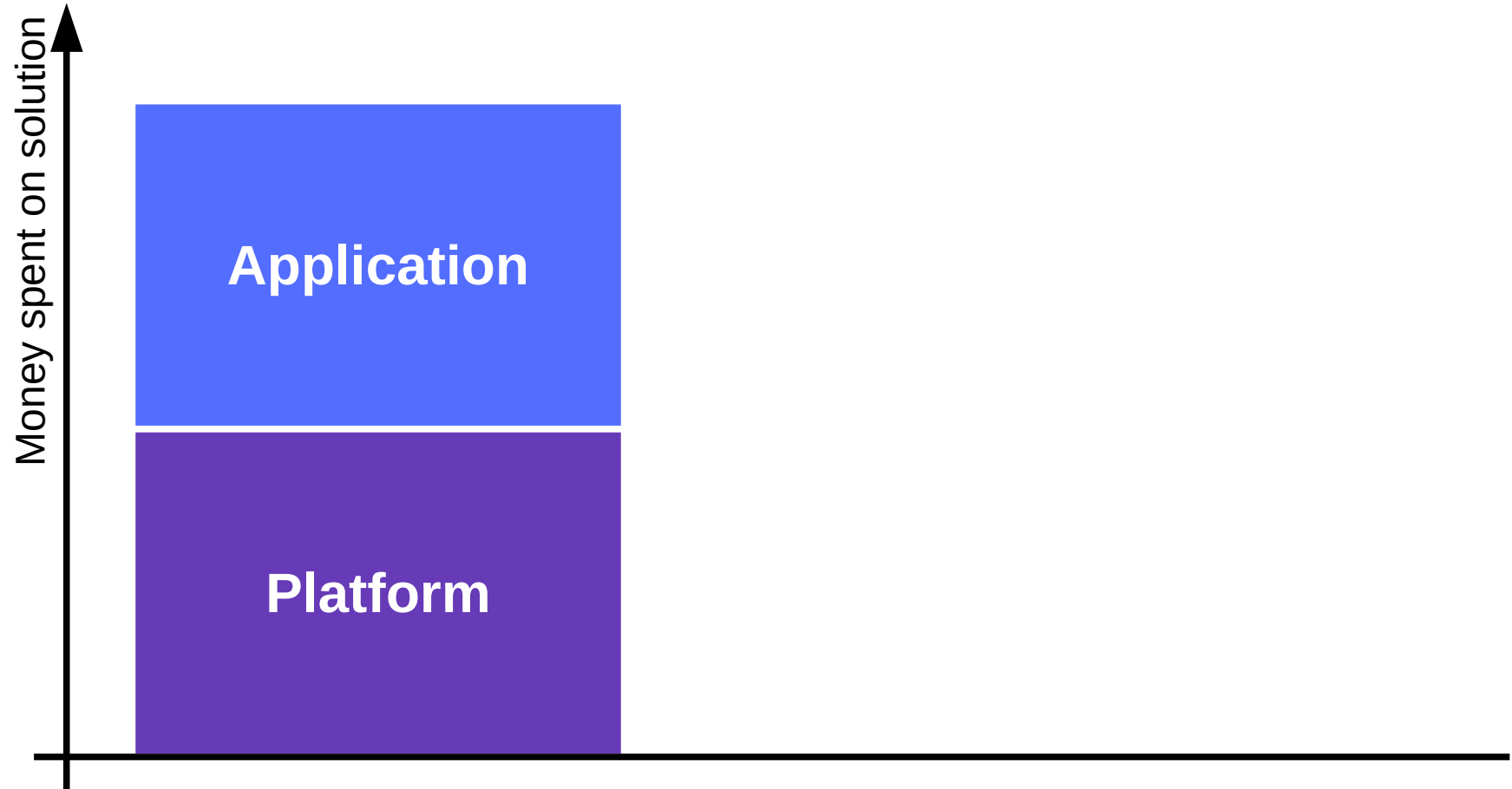


- Software platform
 - Is an environment for the development and deployment of applications
 - Implies split between applications on top of the platform
 - Provides a full set of application-independent life-cycle functions for applications
 - Among many components, the largest collection (i.e. not just a library)
- Customer (user) value of software platforms
 - By definition, a platform in itself is useless
 - Customer value is only created by applications

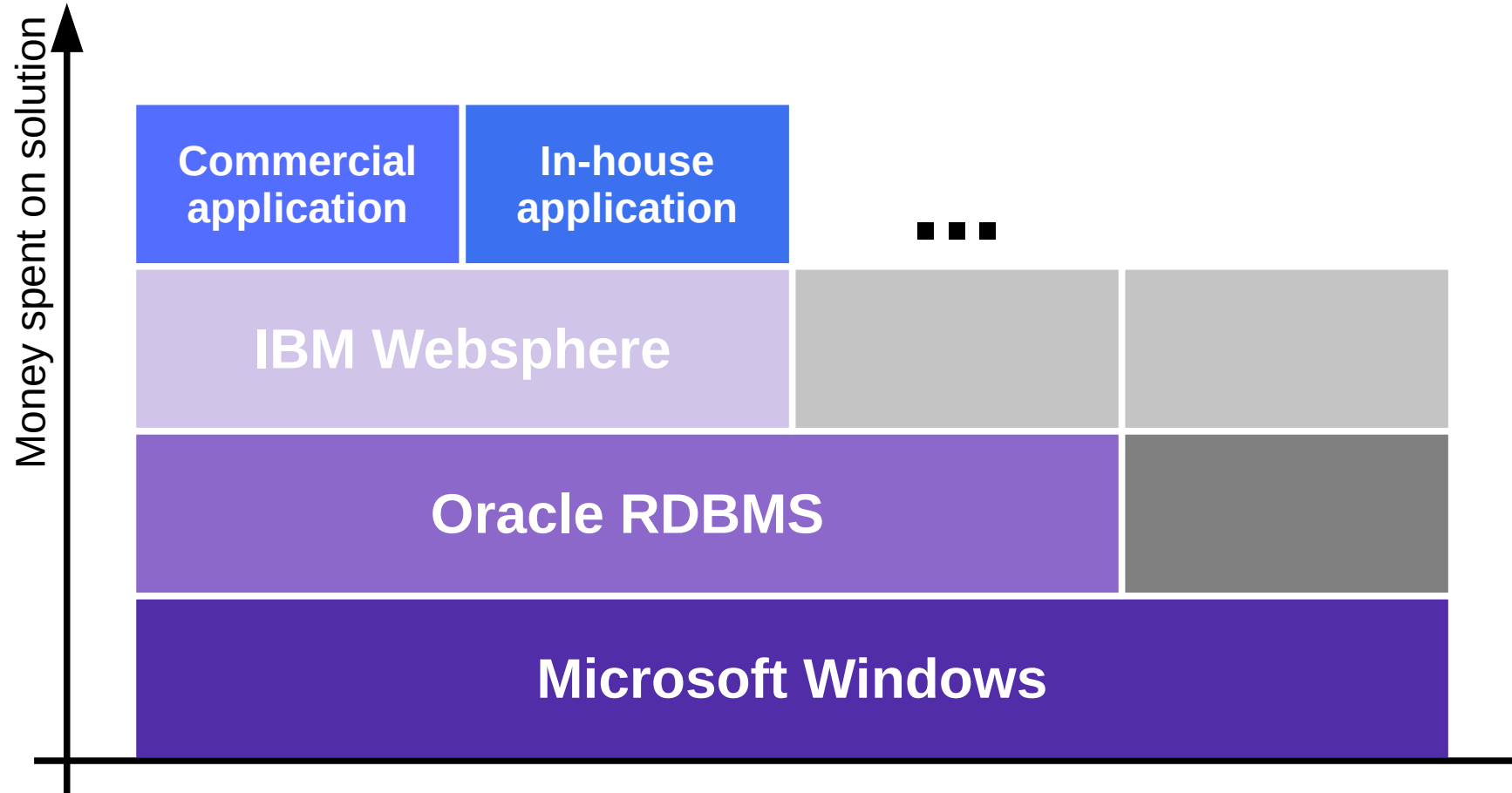
Software Platforms as a Product

- Platforms are valuable
 - Platforms are needed by the applications running on top of it
 - Platforms can simplify IT department operations costs
- An application license sale implies a platform sale

Pricing Power 1 / 2



Pricing Power 2 / 2 [1]



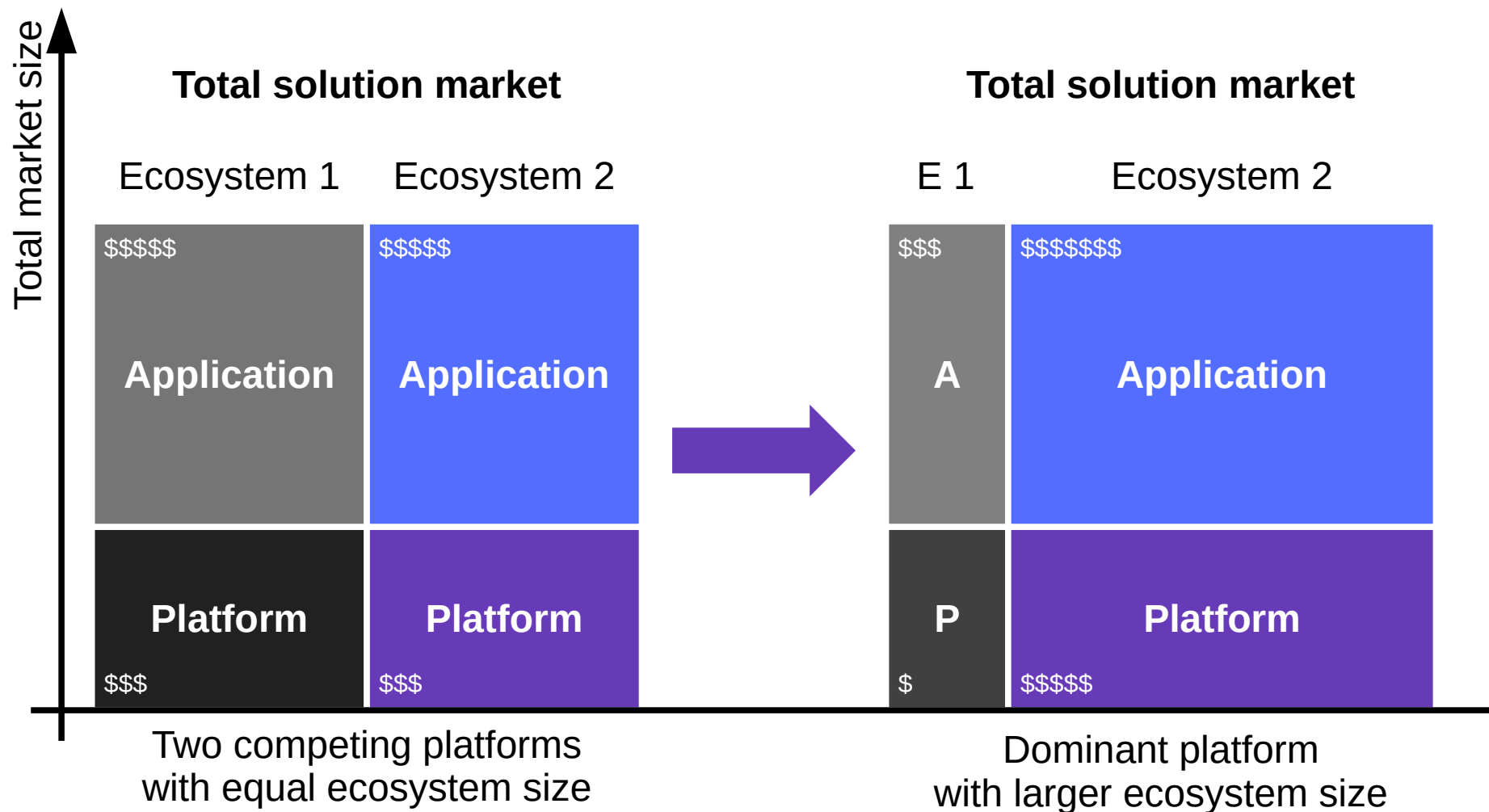
[1] Not drawn to scale

7. Software Ecosystems

Software Ecosystem

- **Software ecosystem**
 - The totality of actors (businesses and individuals)
 - Software applications and components
 - Their relationships and goals
 - On and around a software platform

The Software Ecosystem Wars



Summary

1. Definition (software)
2. The software industry
3. A (very) short history
4. The main players
5. Software products
6. Software platforms
7. Software ecosystems

Thank you! Questions?

dirk.riehle@fau.de – <http://osr.cs.fau.de>

dirk@riehle.org – <http://dirkriehle.com> – [@dirkriehle](#)

Credits and License

- Original version
 - © 2020 Dirk Riehle, some rights reserved
 - Licensed under [Creative Commons Attribution 4.0 International License](#)
- Contributions
 - None yet