Software Deployment

Introduction to Software Deployment

Sergio Peignier   sergio.peignier@insa-lyon.fr

Associate Professor
INSA Lyon
Biosciences department

Table of contents

Software deployment overview

Software Deployment: Set of **interrelated steps** that allow **software** to be **available for use**.

Software Deployment Activities

- -1) (Development process)
  Software **design** and **implementation**
- 0) **Release**
- 1) **Installation**
- 2) **Update**
- 3) **Buit-in update**
- 4) **Version tracking**
- 5) **Activation**
- 6) **Deactivation**
- 7) **Uninstall**

Software Deployment Activities

-1) (Development process)

0) Release
Distribute the application's final version (sometimes included in the development process).

1) Installation

2) Update

3) Buit-in update

4) Version tracking

5) Activation

6) Deactivation

7) Uninstall

Software Deployment Activities

-1) (Development process)

0) Release

1) Installation
   Setup the program ready for execution.

2) Update

3) Buit-in update

4) Version tracking

5) Activation

6) Deactivation

7) Uninstall

Software Deployment Activities

- -1) (Development process)
- 0) Release
- 1) Installation
- 2) Update
    Replace a previous software version (all or a part) with a new release.
- 3) Buit-in update
- 4) Version tracking
- 5) Activation
- 6) Deactivation
- 7) Uninstall

Software Deployment Activities

-1) (Development process)
0) Release
1) Installation
2) Update
3) Buit-in update
   Automated update process
4) Version tracking
5) Activation
6) Deactivation
7) Uninstall

Software Deployment Activities

-1) (Development process)
0) Release
1) Installation
2) Update
3) Buit-in update
4) Version tracking
   System allowing to **find** and **install updates**
5) Activation
6) Deactivation
7) Uninstall

Software Deployment Activities

- -1) (Development process)
- 0) Release
- 1) Installation
- 2) Update
- 3) Buit-in update
- 4) Version tracking
- 5) Activation
  Start executing instances.
- 6) Deactivation
- 7) Uninstall

Software Deployment Activities

-1) (Development process)

0) Release

1) Installation

2) Update

3) Buit-in update

4) Version tracking

5) Activation

6) Deactivation
Shut down executing instances.

7) Uninstall

Software Deployment Activities

- -1) (Development process)
- 0) Release
- 1) Installation
- 2) Update
- 3) Buit-in update
- 4) Version tracking
- 5) Activation
- 6) Deactivation
- 7) Uninstall
   Software removal (and dependencies)

Software Deployment Roles

- **Application** developers
- **Build-and-release** engineers and managers
- **Deployment** and **release** coordinators
- **System** admin
- **Database** admin
- …

Pre-internet software deployment

- Expensive
- Infrequent
- Based on **physical storage**
  - ROM cartridge
  - Cassette tape
  - Floppy disk
  - Optical Media (e.g., CD)
  - Flash drive

Internet Software Deployment

- **Fast** deployment to very **large** audience.
- Huge **Flexibility**

New paradigms linked to **software deployment**

- **Continuous delivery**.
- **Software as a service (SaaS)** (in **Cloud computing**)

Continuous Delivery

- Software engineering approach

- Teams work in short cycles to deliver software

- Software is ready to be deployed at any time.

- Release software (new features, bug fixes, …) into production frequently, quickly, safely and sustainably.

- Deployments are applied manually

Deployment pipeline

- **Validation steps** to pass before being **released**.
- **Commit code** should be **releasable** at **any time**
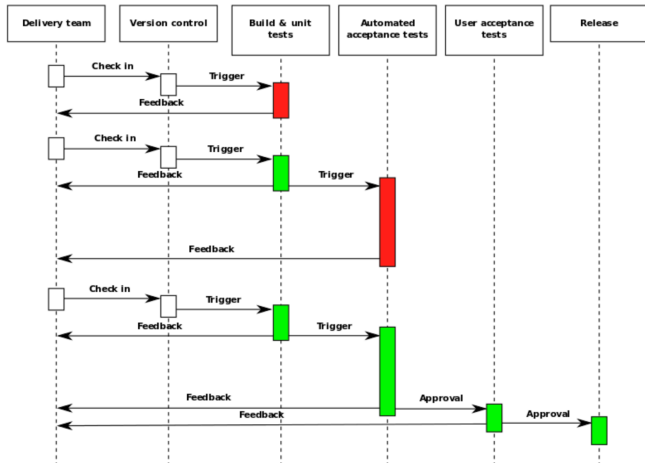
Benefits

- **Fast delivery** of new software **releases** to **clients**.

- **Frequent releases** $\rightarrow$ **Quick feedback** $\rightarrow$ focus on **Useful features only**

- High **productivity**, **efficiency**

- **Frequent releases** $\rightarrow$ **few code to check** $\rightarrow$ easier to **spot bugs** $\rightarrow$ Higher **reliability** and **quality**

Drawbacks

- **Clients** may prefer to rely only on **one stable version** (critical stage)

- **Test automation**: Some fields (medical) require **extensive tests** before **releasing a product**
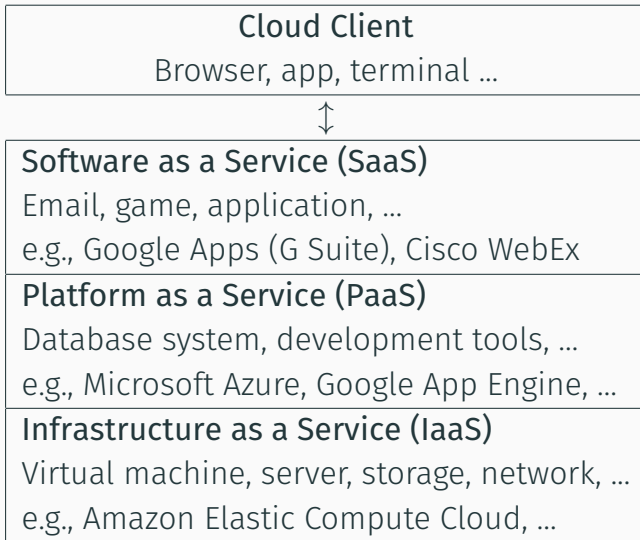
Related concepts

- Continuous Deployment: Deployments are automatically scheduled.

- DevOps: Similar approach, but does not only include software engineering (management, quality assurance...)

- Agile software development: Co-evolution of requirements and solutions through close exchange between clients and developers.

Software as a Service

Cloud Computing

- Make **resources** (computer power/storage) **available** to **many users** over **internet**.

- **No** demand of **active management** by the **users**.

Cloud Client
Browser, app, terminal …

↕

Software as a Service (SaaS)
Email, game, application, …
e.g., Google Apps (G Suite), Cisco WebEx
Platform as a Service (PaaS)
Database system, development tools, …
e.g., Microsoft Azure, Google App Engine, …
Infrastructure as a Service (IaaS)
Virtual machine, server, storage, network, …
e.g., Amazon Elastic Compute Cloud, …

SaaS	Applications
	Middleware
PaaS	
	OS
	Virtualization
IaaS	Servers/storage
	Network
	Data Center

User
↕
Application
↕
Middleware
↕
OS
↕
Firmware
↕
Hardware

Some Definitions

- **Middleware:** "Software glue" layer between applications and the OS.
- **Operating System (OS):** Software managing hardware and software resources.
- **Firmware:** Software providing hardware **low-level control**.
- **Virtualization:** Create a virtual machine with an OS (acts like a real machine)

Software as a Service

- Model of software delivery

- Software license based on a subscription
  (No perpetual license)

- Centrally hosted

- Use via web browsers using a thin client
  (server does most of the job)

- Instant deployment

Software License

Property category including **intangible human creations**.

There are **two types** of rights

- **Industrial property** (trademark, patent)
- Copyright

Legal right of the author of a work to determine its conditions of use

- Often for **limited time**
  Several years after creators' death → **public domain**

- Protects the ideas **original expression**
  but **not** the **ideas themselves**.

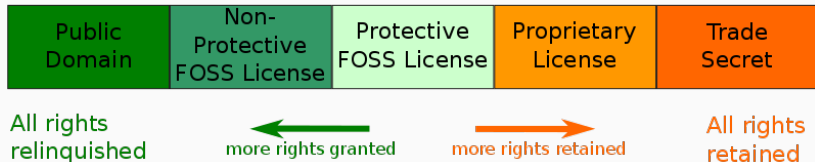- **Depends** on the **countries jurisdiction**

Software → Copyright protected

source code and object code (compiled) are protected

Software Categories under Copyright

- Free and Open Source Software (FOSS)
  Grants right to **modify** and **re-use** the software

- Proprietary software
  Closed source

Rights in Copyright

Public Domain	Non-Protective FOSS License	Protective FOSS License	Proprietary License	Trade Secret

All rights relinquished

← more rights granted

more rights retained →

All rights retained

Legal instrument (contract law) defining software use and distribution.

- Rights/restrictions of software under copyright
- Responsibility terms between parties.
  - Limited liability (pay money for damages)
  - Warranty disclaimers
  - Legal indemnity
- No explicit license → All rights reserved
- EULA: End-User License Agreement

Proprietary Software

- Software **publisher keeps** the **ownership**.

- User must **accept** the **license** to **use** the **software**.

- **Free proprietary software** $\rightarrow$ **Freeware**

Open Source Software

Four freedoms

0) **Run** the program for any purpose
1) **Study** the software functioning and **change** it
2) **Copy** and **redistribute**
3) **Improve** the software and **deploy** to public.

Major categories

Protective/reciprocal/Copyleft license (e.g., GPL)

- Author keeps copyright
- **Redistribution** under the **same license (share alike)**

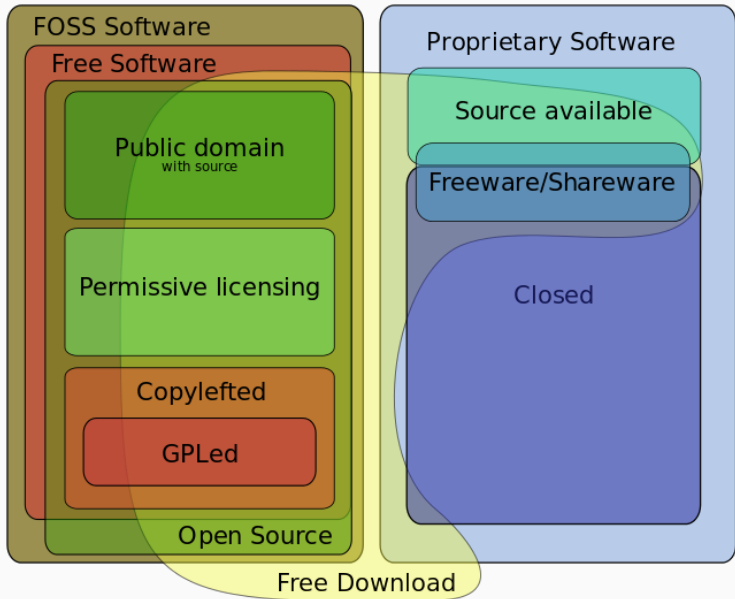### Permissive/copyfree license (BSD-style)

- Author keeps copyright (disclaim warranty)
- **No restriction** on **distribution**, even closed-source

Public-domain

- **Expired copyright**
- **Released** under public domain (waiver statement)
- Can be incorporated to proprietary or free software

Categories

25

Licenses - freedoms

Rights granted	Public domain	Permissive FOSS license (e.g. BSD license)	Copyleft FOSS license (e.g. GPL)	Freeware/Shareware/Freemium	Proprietary license	Trade secret
Copyright retained	No	Yes	Yes	Yes	Yes	Very strict
Right to perform	Yes	Yes	Yes	Yes	Yes	No
Right to display	Yes	Yes	Yes	Yes	Yes	No
Right to copy	Yes	Yes	Yes	Often	No	Many lawsuits are filed by the owner
Right to modify	Yes	Yes	Yes	No	No	No
Right to distribute	Yes	Yes, under same license	Yes, under same license	Often	No	No
Right to sublicense	Yes	Yes	No	No	No	No
Example software	SQLite, ImageJ	Apache web server, ToyBox	Linux kernel, GIMP, OBS	Irfanview, Winamp, *League of Legends*	Windows, the majority of commercial video games and their DRMs, Spotify, xSplit, TIDAL	Server-side Cloud computing Games by Blizzard Entertainment, Rockstar, Activision, etc. PlayStation Network and Xbox Live

Exercise

Search some well-known licenses

To which **family** does this **license belongs**?

Overview