Universidad de Oviedo

School of Computer Science

Software Architecture

Lab. 02

UML diagrams

PlantUML

Introduction to arc42 docs

2020-21

Jose Emilio Labra Gayo
Pablo González
Irene Cid
Paulino Álvarez

School of Computer Science, University of Oviedo

UML

Unified Modeling Language

- Before UML there were several proposals
- UML notation unifies them
- Proposed by OMG (Object Management Group)
- Current version UML 2.5.1 (2017)

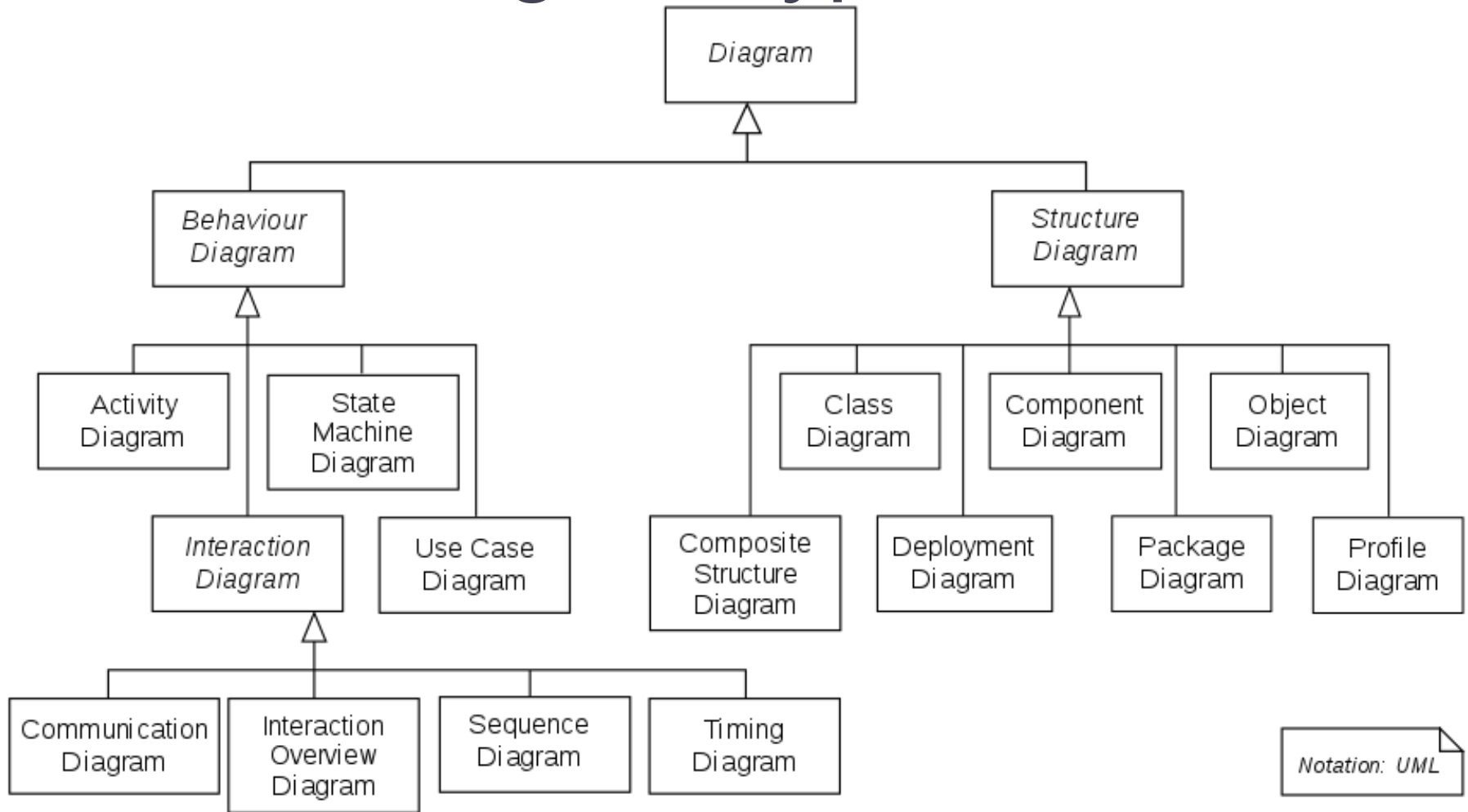Model = abstraction of a problem

- It can have different diagrams
- Diagram = partial graphic representation of a model

OCL = Object Constraint Language

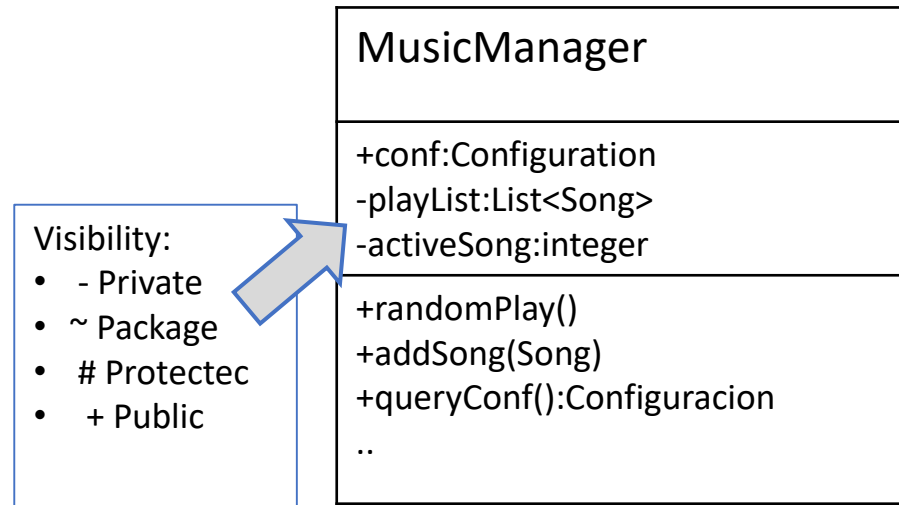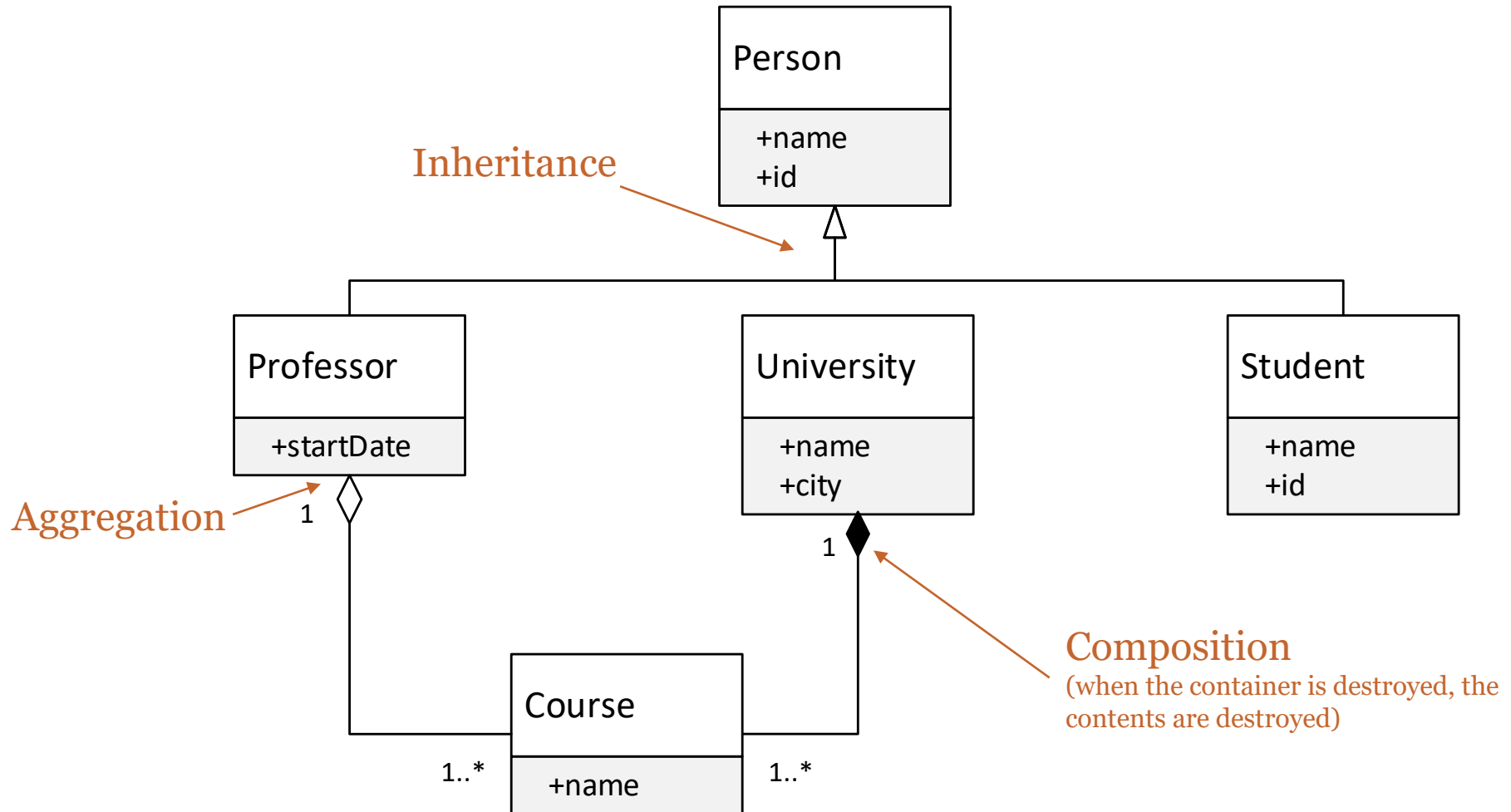- Constraints between objects using formal language

14 UML Diagram types

```
                        ┌─────────────┐
                        │  Diagram    │
                        └──────△──────┘
              ┌────────────────┴────────────────┐
      ┌───────────────┐                  ┌───────────────┐
      │  Behaviour    │                  │  Structure    │
      │  Diagram      │                  │  Diagram      │
      └───────△───────┘                  └───────△───────┘
```

Diagram

Behaviour Diagram

Structure Diagram

Activity Diagram

State Machine Diagram

Class Diagram

Component Diagram

Object Diagram

Interaction Diagram

Use Case Diagram

Composite Structure Diagram

Deployment Diagram

Package Diagram

Profile Diagram

Communication Diagram

Interaction Overview Diagram

Sequence Diagram

Timing Diagram

Notation: UML
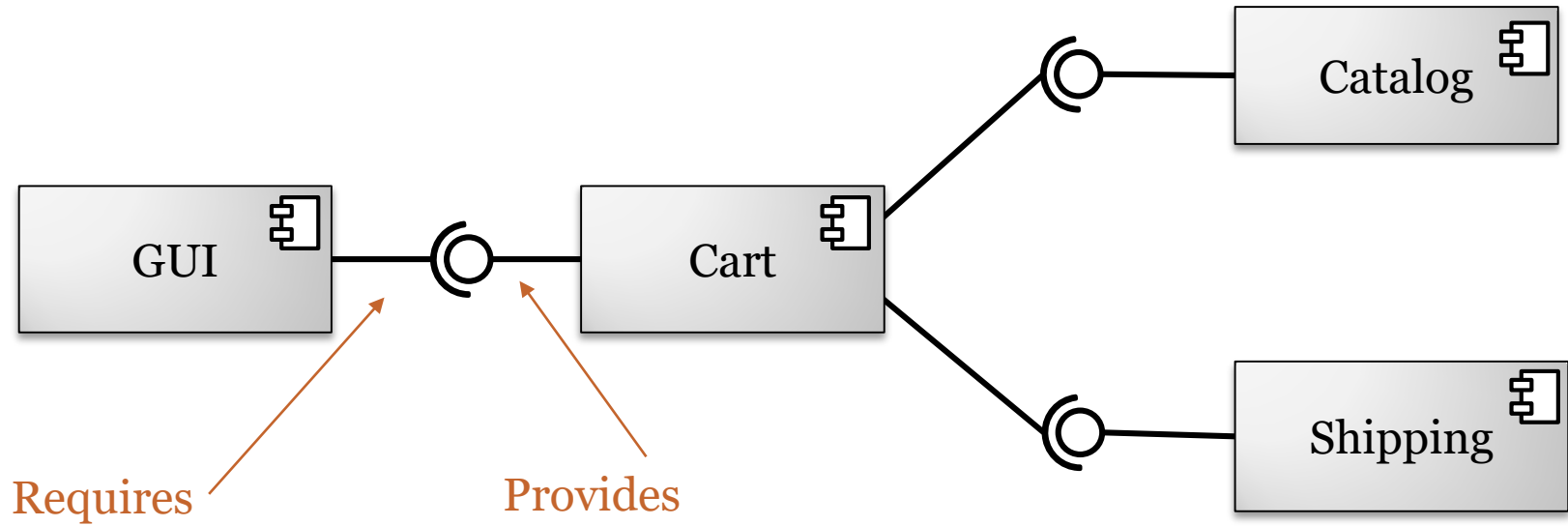
Fuente: Wikipedia

Class diagrams

- Models the static part of the project, without taking into account the time aspect
- Explains the relationships between the different classes.
- Arc42:8-Concepts

Visibility:
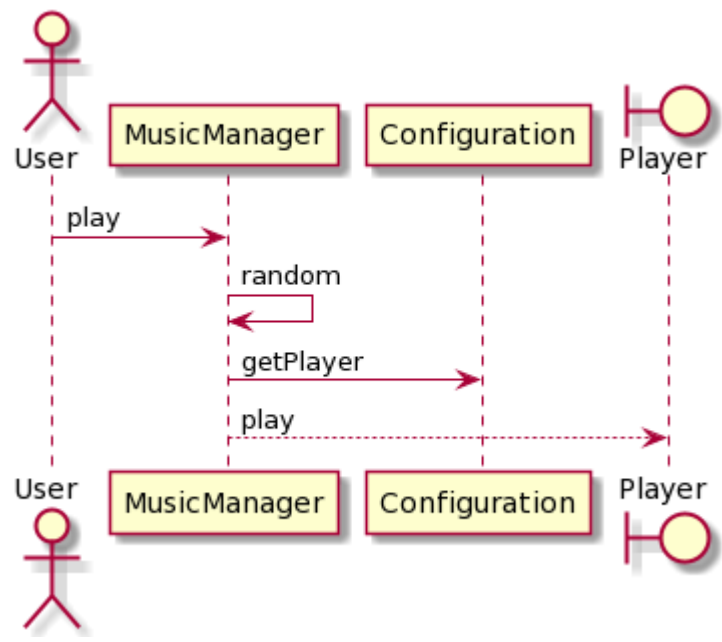- - Private
- ~ Package
- # Protectec
- + Public

MusicManager
+conf:Configuration -playList:List<Song> -activeSong:integer
+randomPlay() +addSong(Song) +queryConf():Configuracion ..

Example

UML Component diagram

GUI

Cart

Catalog

Shipping

Requires

Provides

Sequence diagram

Models communication between some objects at a given time

Objects can send two types of messages: synchronous or asynchronous
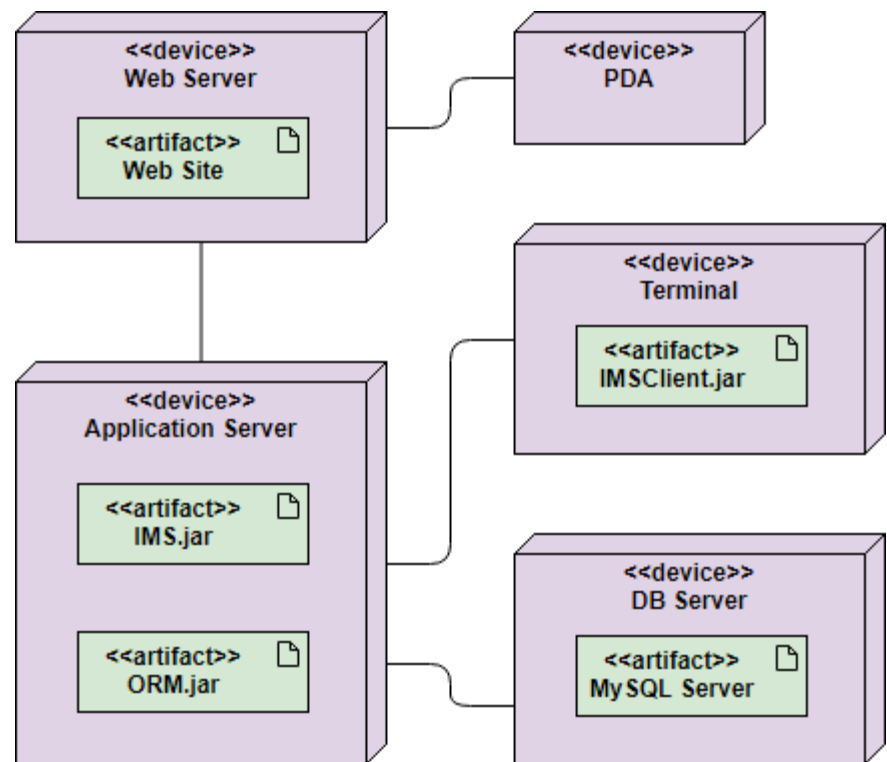
Arc42:6-RuntimeView

Deployment diagrams

Represents the final location of
  the components in an app

Elements:

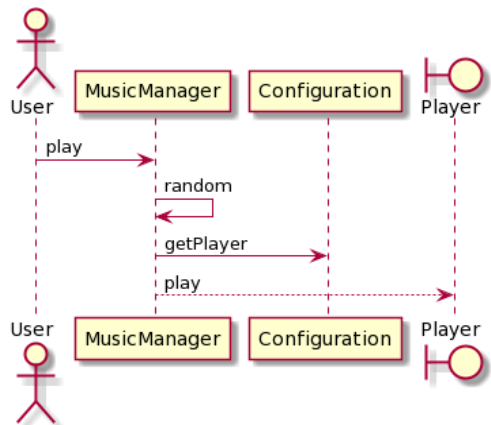 Nodes , Components,
    relationships

Arc42: 07.DeploymentView

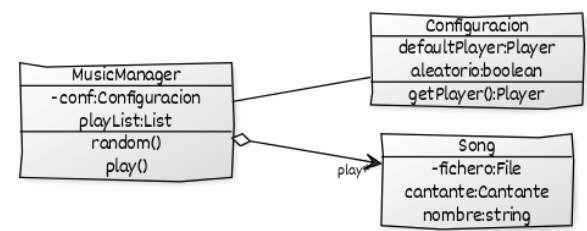Fuente:VisualParadigm

Text-based tools

PlantUML

```
@startuml component
actor User
participant MusicManager
participant Configuration
boundary Player
User -> MusicManager: play
MusicManager -> MusicManager: random
MusicManager -> Configuration : getPlayer
MusicManager --> Player : play
@enduml
```


YUML

```
// Cool Class Diagram
[MusicManager|-conf:Configuracion;
playList:List |random();play()]
[MusicManager]<>-play*>[Song|-
fichero:File;cantante:Cantante;nombre
:string]
[MusicManager]-
[Configuracion|defaultPlayer:Player;a
leatorio:boolean|getPlayer():Player]
]
```
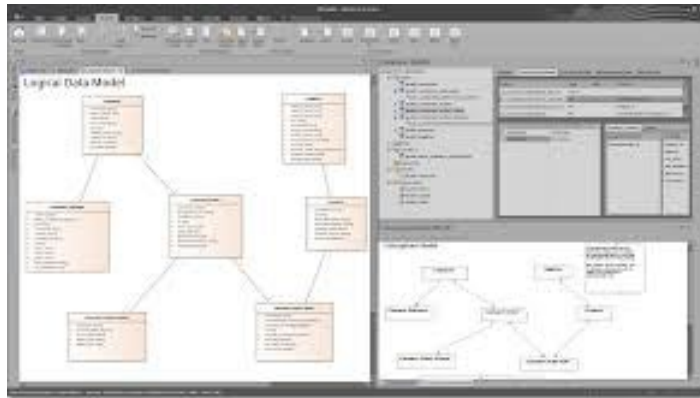
Drawing tools

Powerpoint

MsVisio

UMLet (https://www.umlet.com/)

CASE tools

EnterpriseArchitect

Reverse Enginering with Java/C++

Oracle connection for relational databases

Word, HTML templates

MagicDraw

▫ Java based

▫ UML diagrams

▫ Reverse Engineering Java , C++

Visual Paradigm

▫ Commercial (student license)

Modelio

▫ Open source

▫ Java based

▫ Reverse Engineering Java , C++

Diagramming the architecture

Video:

https://www.youtube.com/watch?v=wgpSdpny-0c

Checklist:

https://c4model.com/assets/software-architecture-diagram-review-checklist.pdf

Arc42 templates

https://arc42.org/

Radarin already follows the template:

https://radarin0webapp.herokuapp.com/docs/

Generation of docs:

> npm run docs

End of presentation