

Modelado - UML

Christian Jaimes

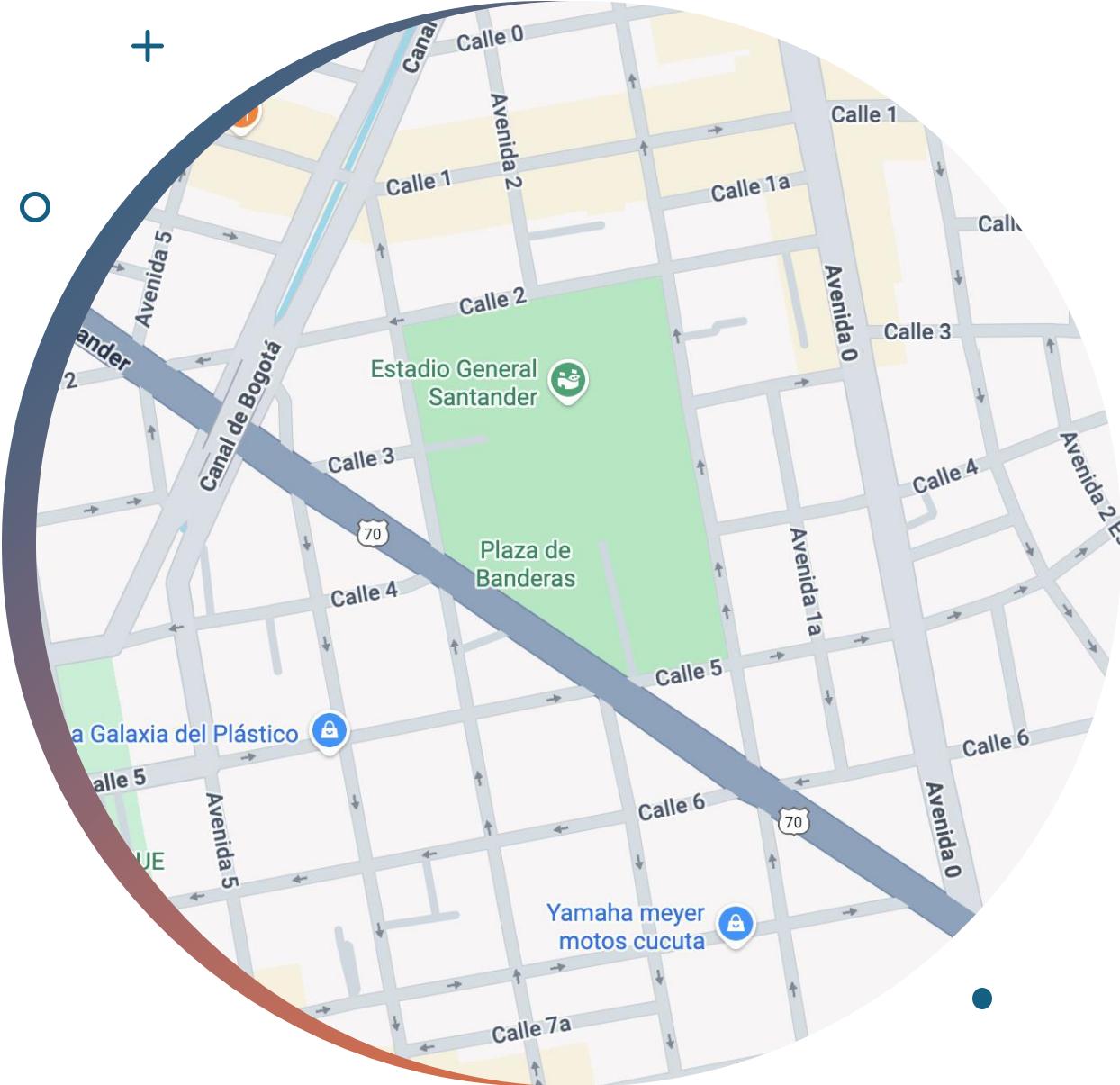
Indice

- Qué es el modelado?
- Qué es UML?
- Diagrama de casos de uso
- Diagrama de clases

Modelado

- Consiste en construir una abstracción de la realidad
- Abstracciones son simplificaciones porque:
 - Ignoran los detalles irrelevantes
 - Solo representan los detalles relevantes
- Lo que es relevante o irrelevante depende del propósito del sistema



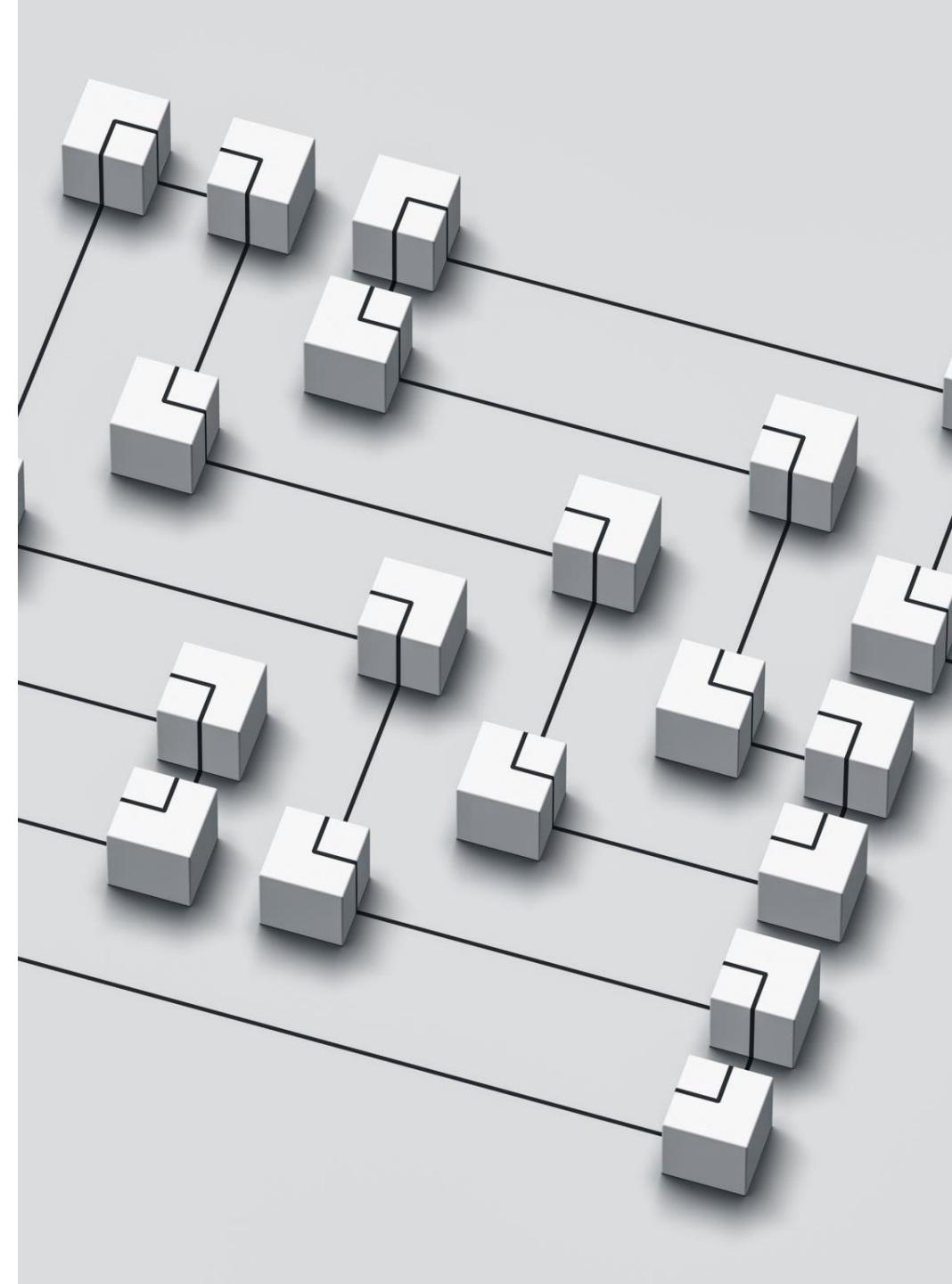


Ejemplo de modelado

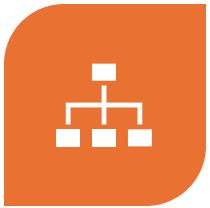
- Un **mapa de calles** es un modelo del mundo real.
- No muestra cada árbol o edificio, solo lo necesario para orientarse.
- Así mismo, en software usamos modelos para enfocarnos en lo importante.

¿Por qué necesitamos modelar software?

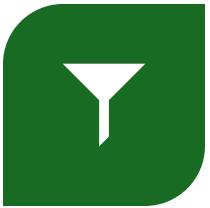
- El software moderno es **muy complejo**.
- Ejemplo: Windows XP tenía más de **40 millones de líneas de código**.
- Un solo programador no puede entender todo el código.
- Los modelos ayudan a **visualizar y simplificar sistemas complejos**.
- Modelar es una herramienta para **manejar la complejidad**.



Sistemas, modelos y vistas



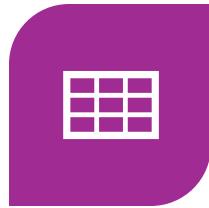
UN MODELO DESCRIBE
UNA PARTE O ASPECTO
DE UN SISTEMA.



UNA VISTA MUESTRA
CIERTOS DETALLES DEL
MODELO.



UNA NOTACIÓN SON LAS
REGLAS (GRÁFICAS O
TEXTUALES) PARA
REPRESENTAR UNA
VISTA.



VARIOS MODELOS O
VISTAS DE UN MISMO
SISTEMA PUEDEN
SUPERPONERSE.
EJEMPLO:



SISTEMA: AVIÓN



MODELOS: SIMULADOR
DE VUELO, MAQUETA A
ESCALA

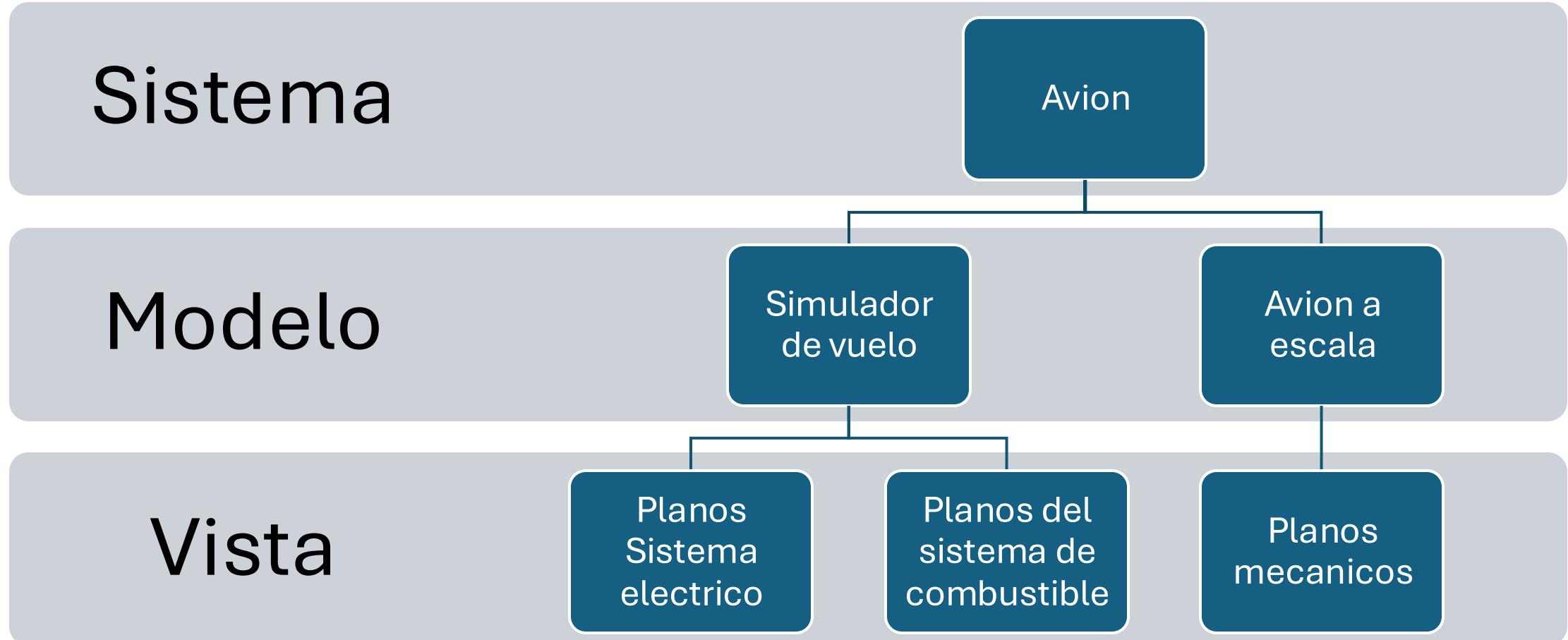


VISTAS: PLANOS,
CABLEADO ELÉCTRICO,
SISTEMA DE
COMBUSTIBLE

Relación entre sistema, modelo y vista

- El **sistema** es la realidad completa (por ejemplo, el avión).
- Un **modelo** representa cómo funciona o se ve (ej. simulador).
- Una **vista** muestra detalles específicos (ej. el cableado).
- En UML:
 - Sistema → lo que queremos construir o entender.
 - Modelo → su representación conceptual.
 - Vista → una perspectiva de ese modelo (clases, secuencia, etc.).

Models, Views and Systems (UML)



Conceptos y fenómenos

Fenómeno: algo del mundo real que podemos percibir.

- Ejemplo: “la clase que estás tomando” o “Un relog Casio clasico”.

Concepto: agrupa fenómenos con características comunes.

- Ejemplo: “clases de ingeniería de software”, “relojes negros”.

Un **concepto** tiene:

- **Nombre:** para distinguirlo.
- **Propósito:** propiedades que definen su pertenencia.
- **Miembros:** los fenómenos que pertenecen a él.



Abstracción y modelado

Abstracción: proceso de enfocarse en los aspectos importantes.

Clasificación: agrupar fenómenos similares en conceptos.

Modelado: desarrollar abstracciones para responder preguntas, ignorando los detalles irrelevantes.

Conceptos en software: tipo e instancia

Tipo: abstracción en programación.

- Ejemplo: int → representa números enteros.

Instancia: un valor concreto del tipo.

- Ejemplo: 1, -2, 0 son instancias de int.

Analogías:

- “Tipo ↔ instancia” es como “concepto ↔ fenómeno”.

TIPO == CONCEPTO

INSTANCIA == FENOMENO

Modelado orientado a objetos

- En el dominio de aplicación modelamos el **problema**.
- En el dominio de solución modelamos la **implementación**.
- UML nos permite representar ambos:
 - Modelo del dominio (ej. aeropuerto, plan de vuelo).
 - Modelo del sistema (ej. controlador de tráfico, pantallas).