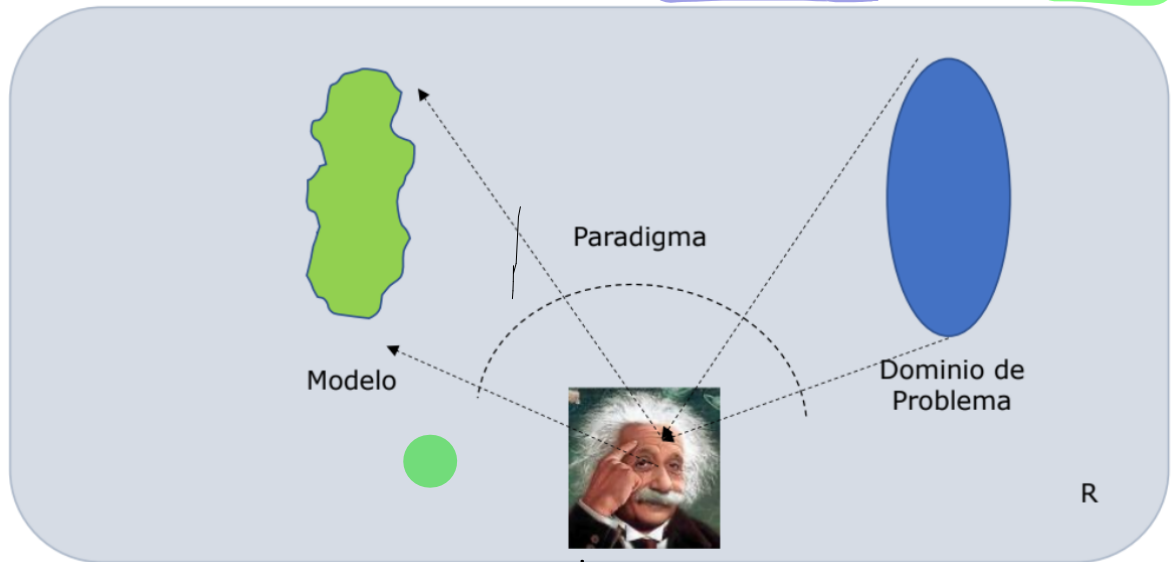


Ingeniería Teórica

¿Qué es el software? → Modelo Computable de un dominio de problema de la realidad



Realidad: Todo aquello que podemos percibir, tocar, ~~haber~~

Dominio de problema: Recorte de la realidad que nos interesa para el negocio que estamos modelando

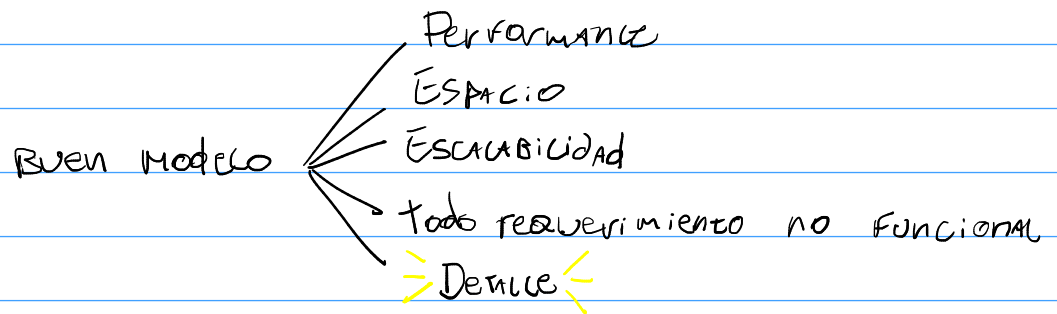
Modelo Representación de aquello que se está modelando

Computable que se puede ejecutar en máquina de Turing → Formal, a-contextual
No solo especifica el que, si no también el como

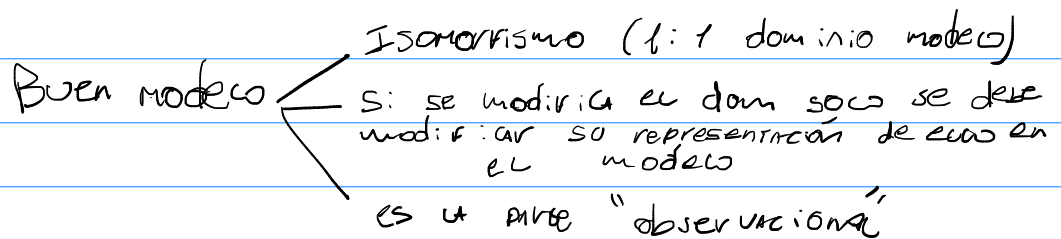
¿Qué hace a un buen modelo?

- ↳ Eje funcional → ¿Buena representación del dom.?
- ↳ Eje Descriptivo → ¿Está bien descrito el modelo?
- ↳ Eje Implementativo → ¿Cómo ejecuta? (en términos técnicos)

En eje DESCRIPTIVO \rightarrow se lo tiene que poder entender y cambiar

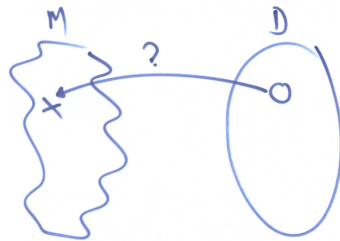


Eje FUNCIONAL \rightarrow debe representar correctamente toda cosa de aquello que modela



CONSULTAR! Porque no podría representar un mes y un año en Ruby, JS ó Go

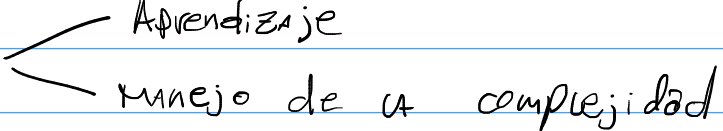
se refiere a representar en el modelo el dominio?



¿Porqué querría hacer esto si el fin de mi programa es modelar la realidad y no modelar la realidad a mi programa?

Ingeniería de software

↳ Aplicación del enfoque científico empírico para encontrar soluciones económicas y eficientes a los problemas prácticos de software

Para esto hay que ser expertos en 
Aprendizaje
manejo de la complejidad

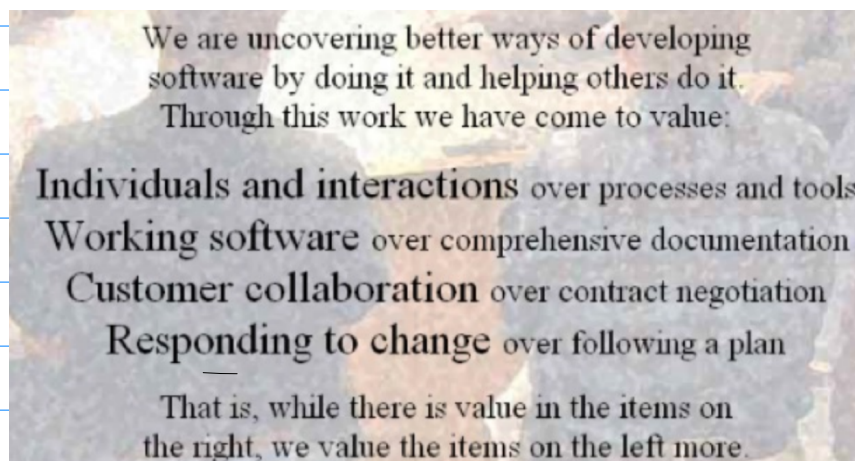
Para ser expertos en el aprendizaje necesitamos lo siguiente

- ↳ Iteración
- ↳ Comentario
- ↳ Incrementalismo
- ↳ Experimentación
- ↳ Empirismo

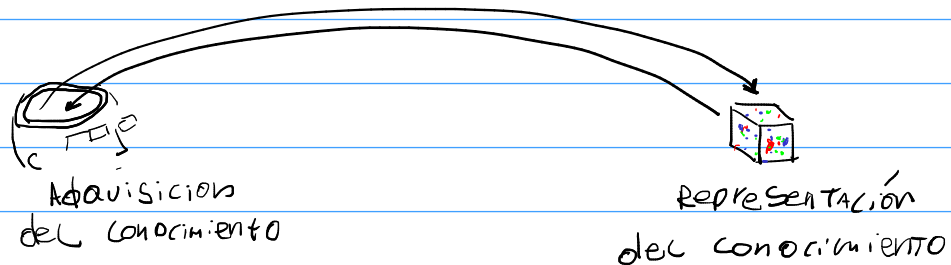
"In science if you know what you are doing you should not be doing it. In engineering if you do not know what you are doing you should not be doing it"

The Art of Science and Engineering - Richard Hamming

Manifiesto For Agile software development



Entonces el desarrollo de software podemos decir que es similar a un proceso de aprendizaje



Esto implica que el desarrollo de software es un proceso de aprendizaje, lo que implica

1. es iterativo
2. es incremental
3. Se genera a partir de hechos concretos
4. El conocimiento generado debe ser organizado

CARACTERÍSTICAS

- ↳ dominio expresado en lenguaje ambiguo y contextualizado
- ↳ desarrollo implica desambiguar y descontextualizar el conocimiento del problema
- ↳ Proceso de desarrollo implica hacer explícito y externo el conocimiento implícito e internalizado de los expertos en el dominio
- ↳ El CAMBIO es una característica esencial del SOFTWARE, no accidental porque
 - cambia el dominio del problema
 - cambia nuestro entendimiento del dominio del problema
 - cambia la manera de modelar lo que entendemos del dominio del problema

Conclusiones

> SOFTWARE = Modelo computable

> Desarrollo de SOFTWARE = Proceso de aprendizaje

↳ ITERATIVO

↳ INCREMENTAL

↳ FEEDBACK INMEDIATO

Estos son los pilares de la materia