



GOBIERNO
DE ESPAÑA

MINISTERIO
DE TRABAJO
E INMIGRACIÓN

SECRETARÍA DE ESTADO
DE LA SEGURIDAD SOCIAL



Gerencia de Informática
de la Seguridad Social

Metodologías Ágiles: Scrum y técnicas de estimación ágil

PreparaTIC - Junio 2009

Jorge Manrubia Díez
jorge.manrubia@giss.seg-social.es

¿Por qué?



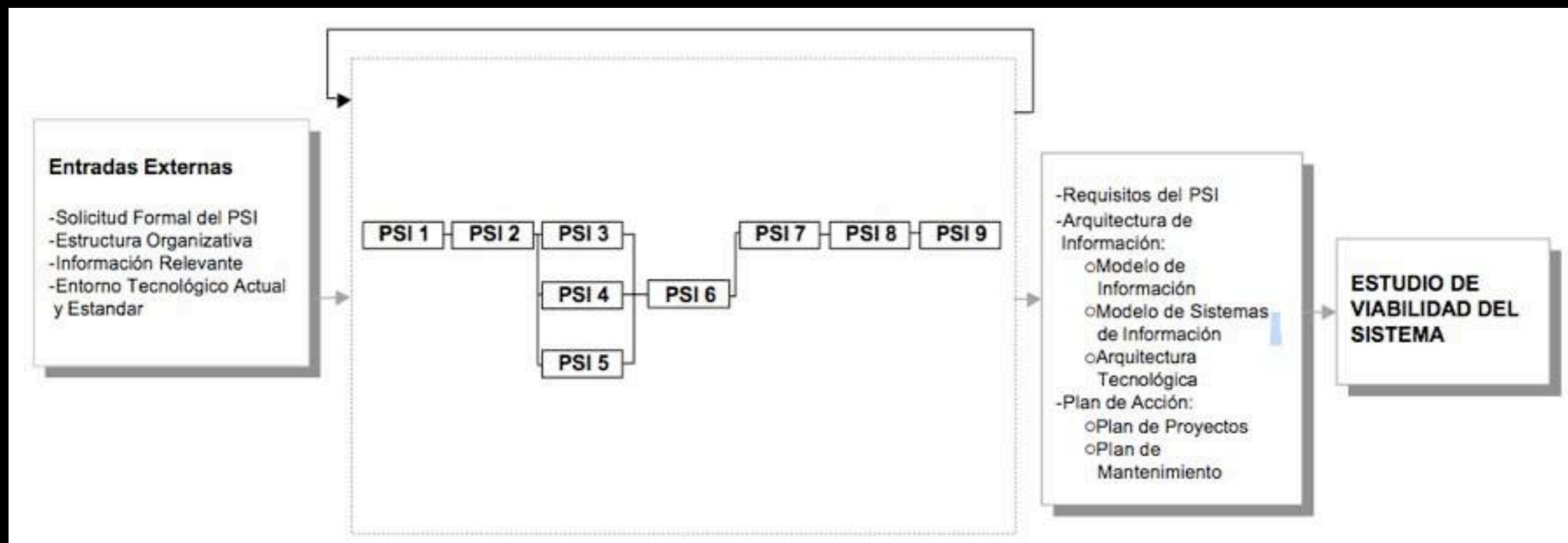
Hacer un programa es como...

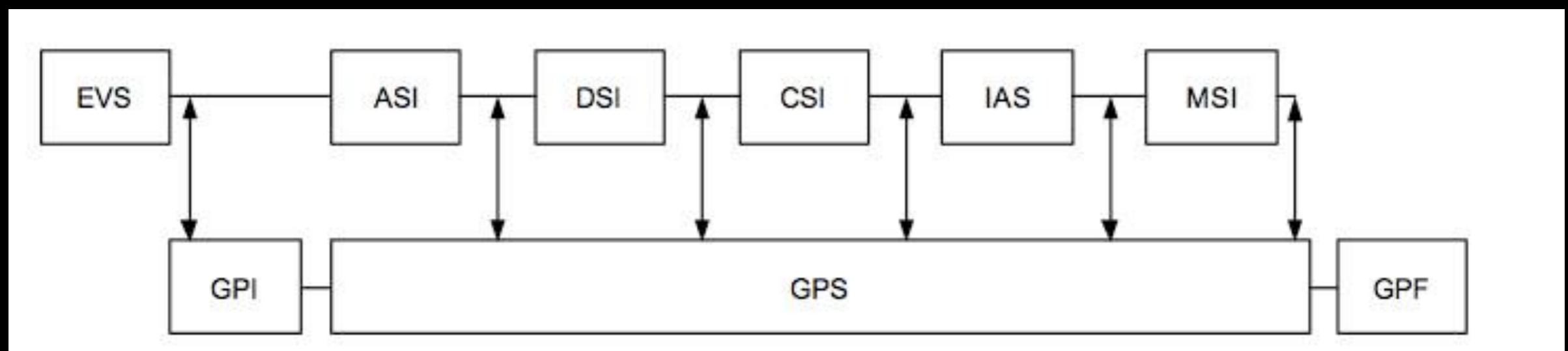
*Can you get a design that is capable of turning the coding into a **predictable** construction activity?
And if so, is cost of doing this sufficiently small to make this approach worthwhile?*

Martin Fowler, “The new methodology” (2005)

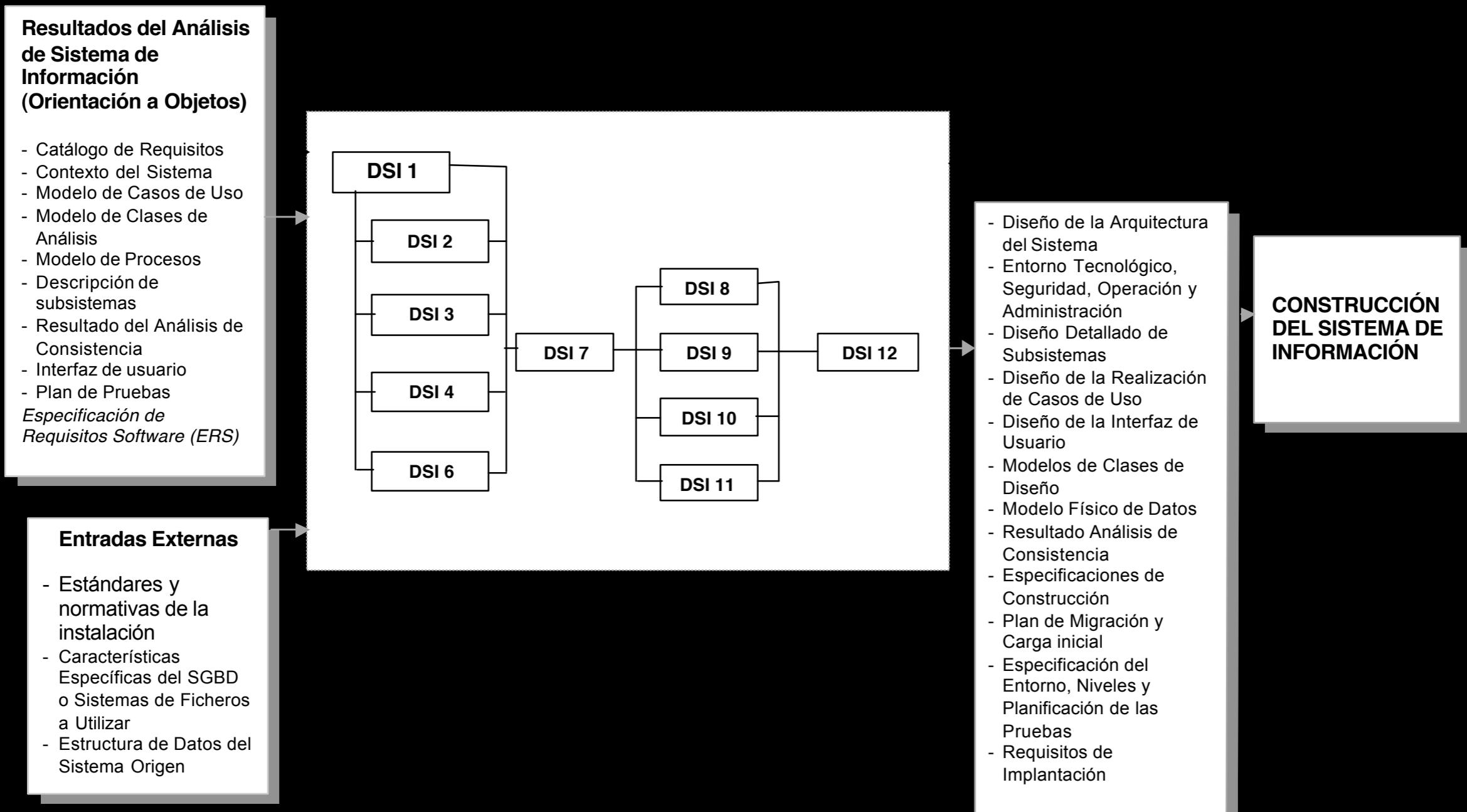
*They were amazed that my industry (software development) was trying to do its work using a completely inappropriate process control model. (...) They said that all complex processes that weren't completely understood required the **empirical model** (not the **defined process control model**)*

Conclusiones de expertos en teoría de control de procesos liderados por Babatunde Ogunnaike tras analizar los procesos de desarrollo software. Ken Schwaber, “Agile Software Development with Scrum” (2001)





DISEÑO DEL SISTEMA DE INFORMACIÓN (ORIENTACIÓN A OBJETOS)



Adaptación continua
del proceso

+

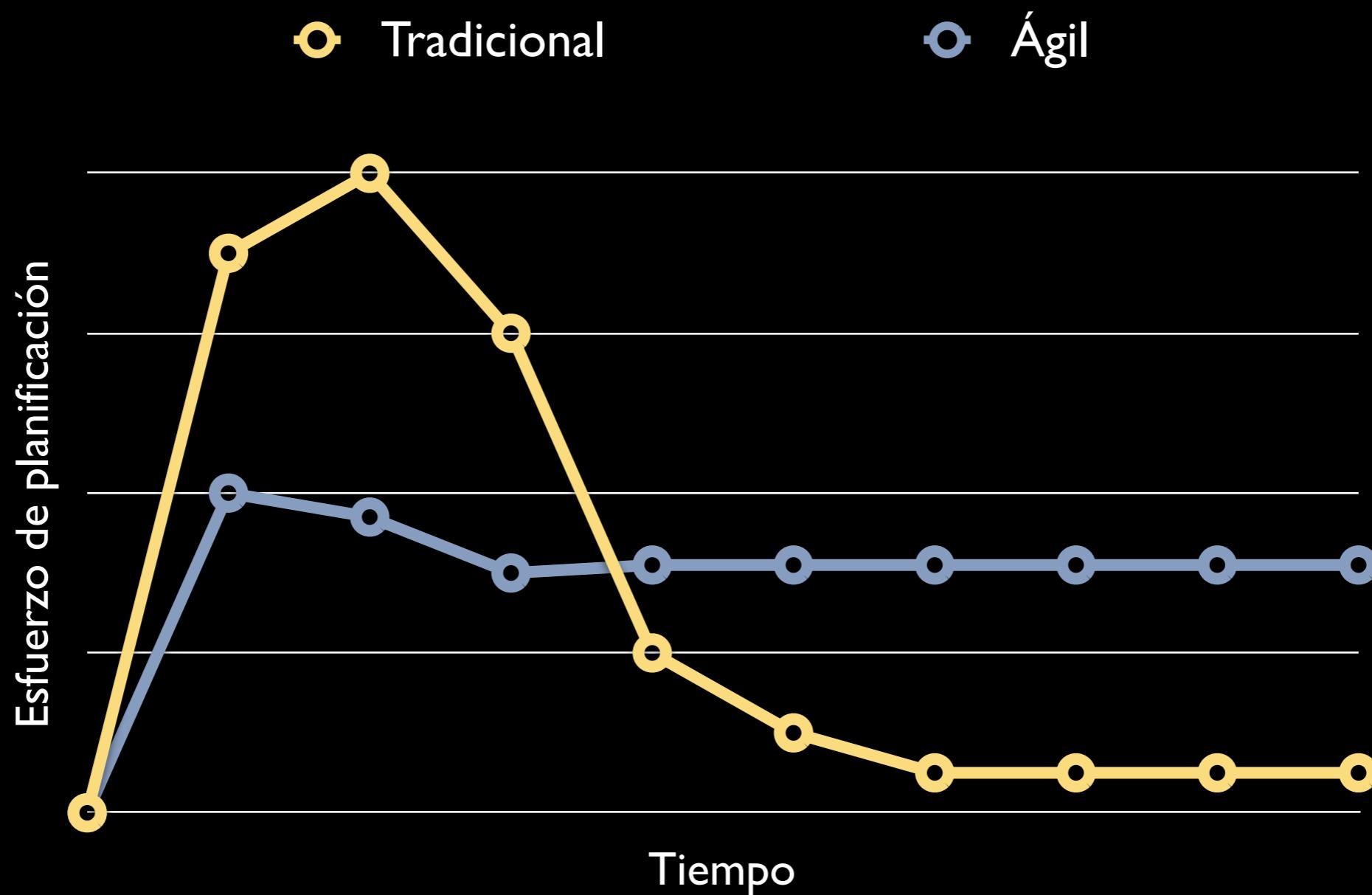
Excelencia técnica



© Scott Adams, Inc./Dist. by UFS, Inc.

© Scott Adams, Inc./Dist. by UFS, Inc.

Desarrollo ágil: lo que **no** es



El propio proceso se adapta durante el desarrollo

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

- Liberación **temprana y continua** del software de **software funcional**
- Integración total de la **gente de negocio** en los equipos de desarrollo
- **Excelencia técnica**, buenos diseños y simplicidad
- Reuniones cara a cara, **comunicación oral**
- **Autoorganización** de los equipos de desarrollo

• • •

<http://agilemanifesto.org/principles.html>

SCRUM y Técnicas de Estimación Ágil

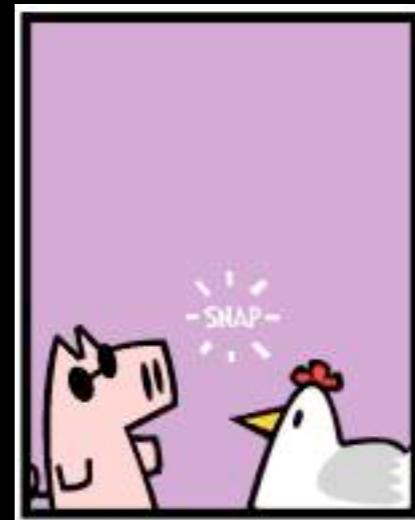
Scrum

- Creada por Ken Schawber, Mike Beedle y Jeff Sutherland (mediados de los 90)
- Equipo de ~7 personas
- Iteraciones (*sprints*) de 30 días



Roles

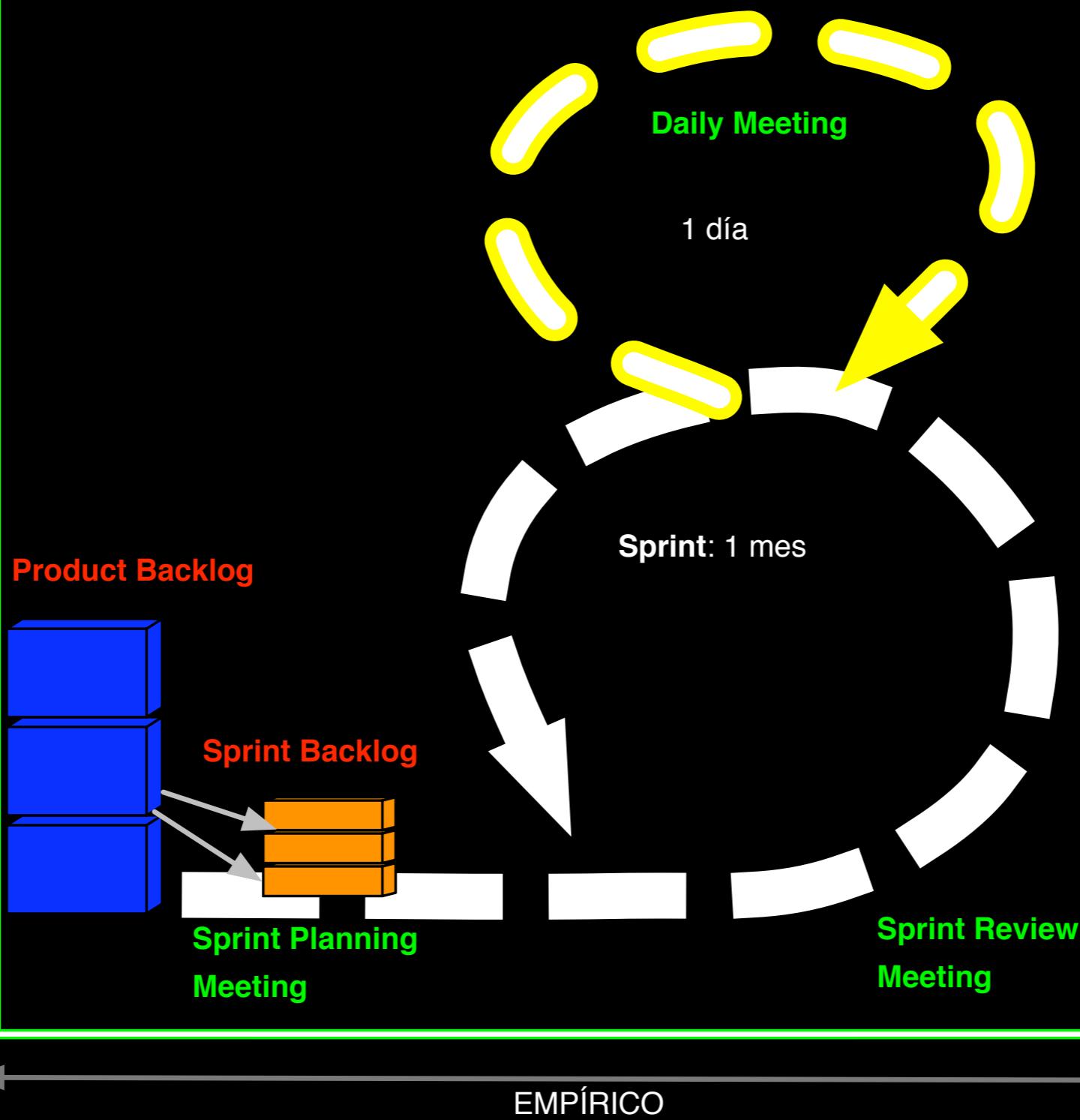
- Product Owner: representa al cliente
- Scrum Master (50% desarrollador)
- Scrum Team
- Chicken (otros)



PRE-GAME

- Product Backlog Inicial
- Release Backlog
- Diseño alto nivel/arquitectura

DESARROLLO (CICLO DE SPRINTS)



POST-GAME

- Documentación de usuario
- Material de formación
- ...

DEFINIDO

EMPÍRICO

DEFINIDO

Backlogs

	Descripción	Items	Estimación
Product Backlog	Lista priorizada de <u>funcionalidades de negocio</u> y técnicas a desarrollar	<ul style="list-style-type: none">• <u>User stories</u>• Características• Bugs	Story points
Sprint Backlog	Selección de items del PB y tareas para desarrollarlos	<ul style="list-style-type: none">• Tareas	Horas

Historias de usuario

Descripción escrita de una funcionalidad
concreta a desarrollar

Como <tipo de usuario> quiero <poder
hacer algo>

Como desarrollador quiero poder transformar
un modelo físico obtenido de ADABAS a un
modelo lógico

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS a un modelo logico

epic



Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS traduciendo sólo las estructuras relacionales

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS admitiendo campos múltiples

Como desarrollador quiero poder transformar un modelo físico obtenido de ADABAS admitiendo campos periódicos

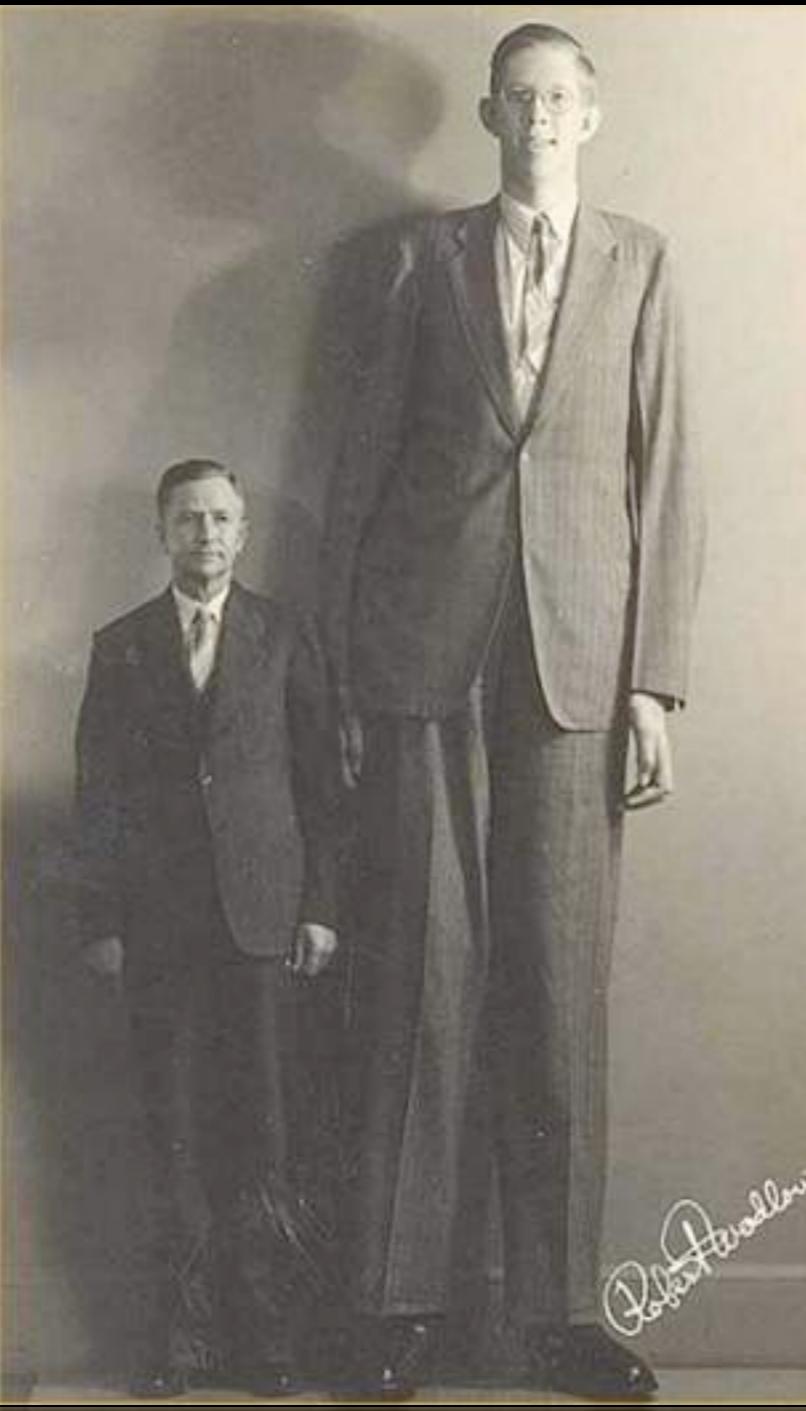
theme

Estimación de Tamaño

≠

Estimación de Duración

Story Point = Esfuerzo necesario +
complejidad + riesgo ...



¿Cuánto miden?
¿Quién es más alto?

1, 2, 3, 5, 8, 13, 20, 30,
50, 100

Escala de estimación

Planning Poker

- La historia se presenta y se discute sobre ella
- Los miembros del equipo escogen carta con la estimación
- Todos dan la vuelta a la carta a la vez
- Los miembros con la estimación más baja y más alta exponen sus razones, y se repite el proceso de estimación



Velocidad del equipo =
Story points / iteración

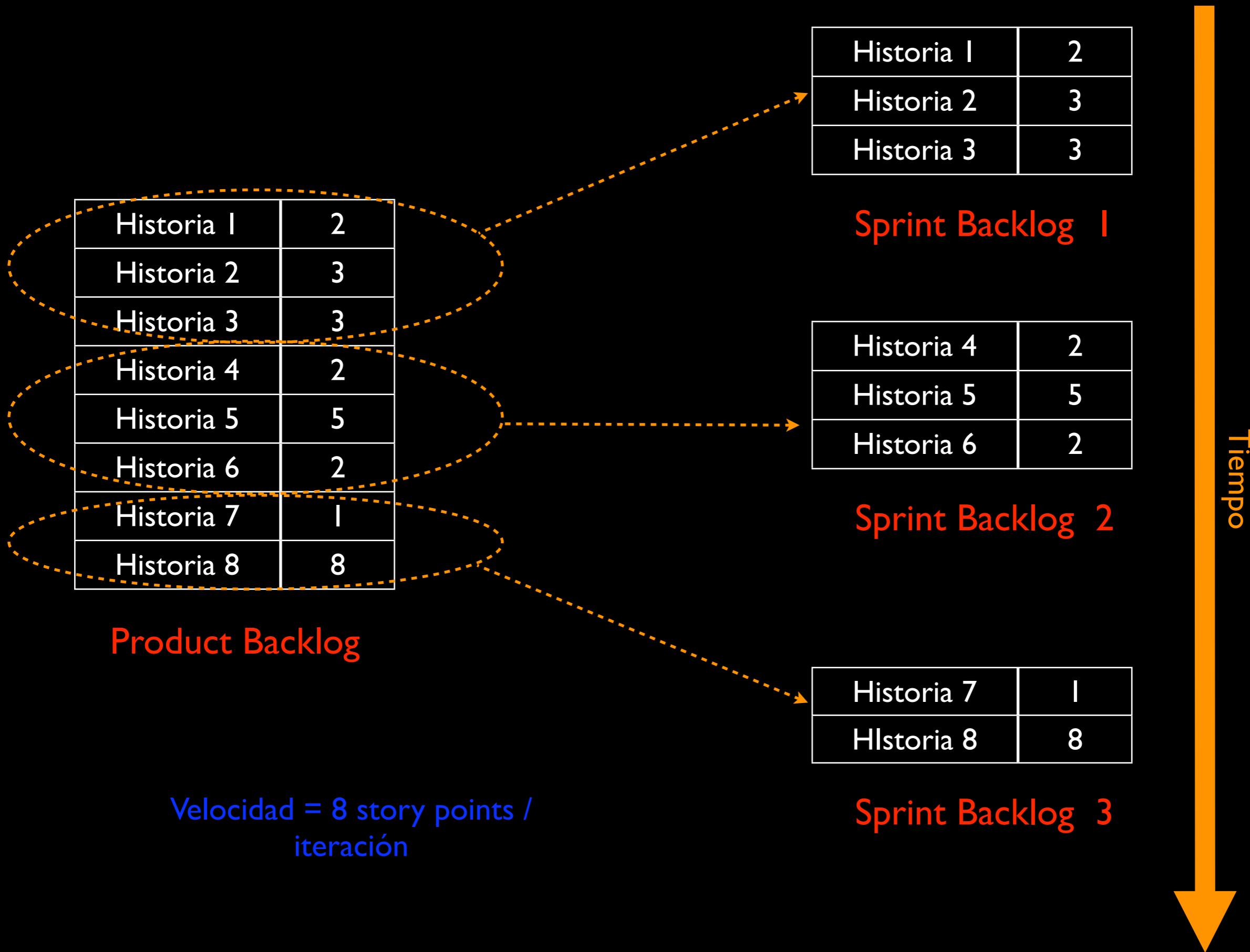


Características
Deseadas

Estimar
tamaño

Estimar
duración

PLANIFICACIÓN



Cómo usuario del plug-in quiero poder obtener un LDM a partir de un modelo de la capa conceptual Rochade

18.0h

Implementar Asistente de selección de agrupación conceptual	<input type="checkbox"/>	8
Prueba funcional	<input type="checkbox"/>	4
Implementar Asistente de selección de fichero destino	<input type="checkbox"/>	4
Implementar la acción	<input type="checkbox"/>	2
Notas de reunión	<input type="checkbox"/>	5
Otros datos de reunión	<input type="checkbox"/>	4

Las historias seleccionadas para el sprint se desglosan en tareas (estimación en horas)

Diariamente los desarrolladores actualizan su estimación de esfuerzo restante

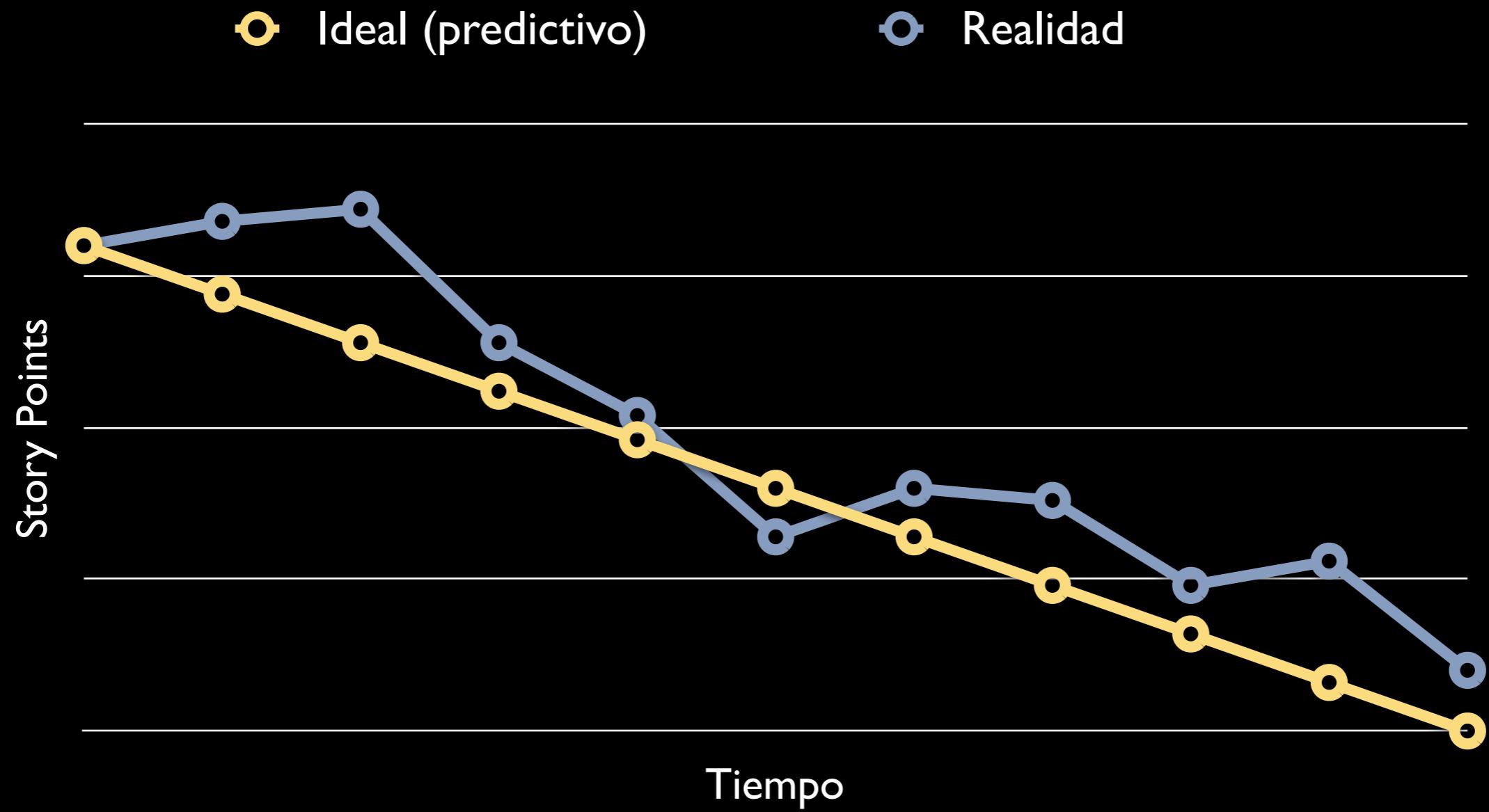


Gráfico de avance del proyecto (Release
Burndown Chart)

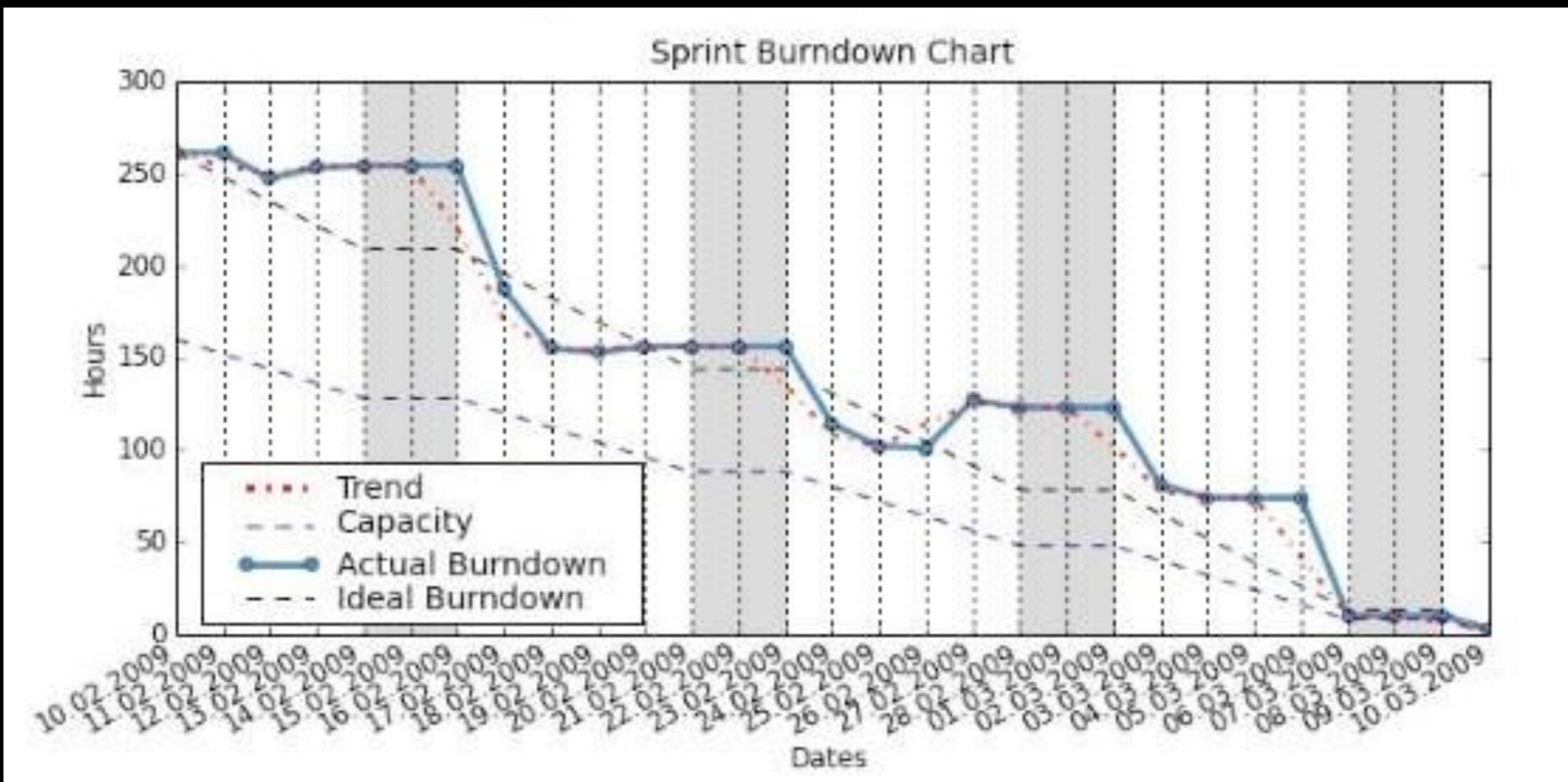


Gráfico de avance del sprint (*Iteration Burndown Chart*)

Daily meeting

- ¿Qué has hecho desde ayer?
- ¿Qué vas a hacer hoy?
- ¿Qué problemas tienes?

15 min. máximo

Excelencia técnica

liberación Frecuente iterativo
entregar Valor refactoring
modelado software Funcional deduda Técnica
planificación pruebas Automáticas
integración Continua
liberación Temprana feedback TDD
colaborativo **simplicidad**
convenios Codificación diseño
codeCoverage patrones
código Legible
mantenibilidad
manejabilidad
copiar y pegar
horizonte

Pruebas

- Exhaustivas
- Automáticas

Pruebas automáticas

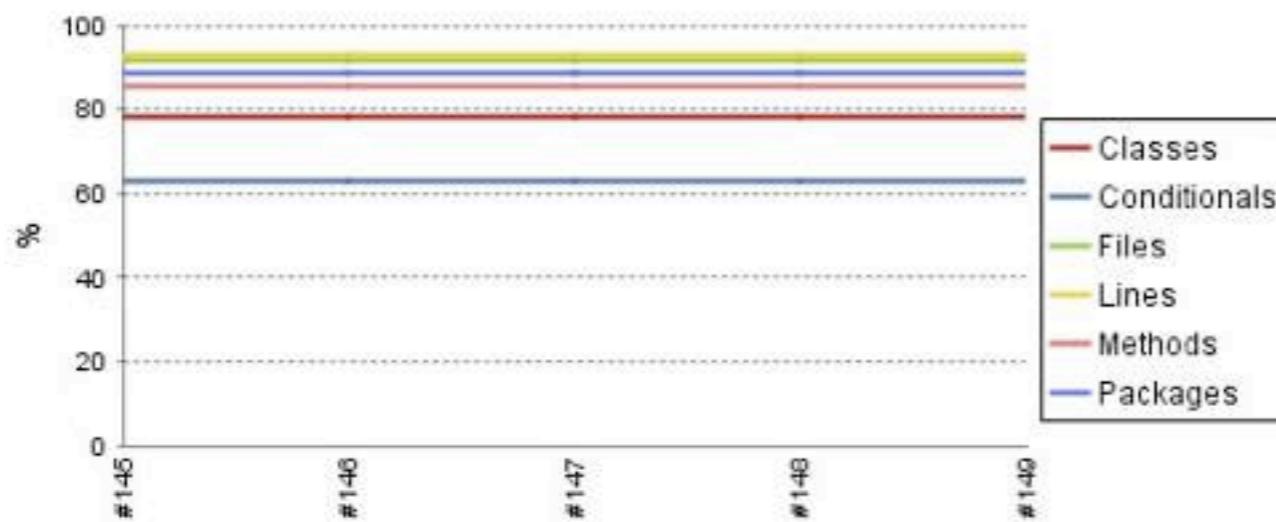


¿Cuánto? → Herramientas
de cobertura de código
(*code coverage*)

Code Coverage

Cobertura Coverage Report

Trend



Project Coverage Summary

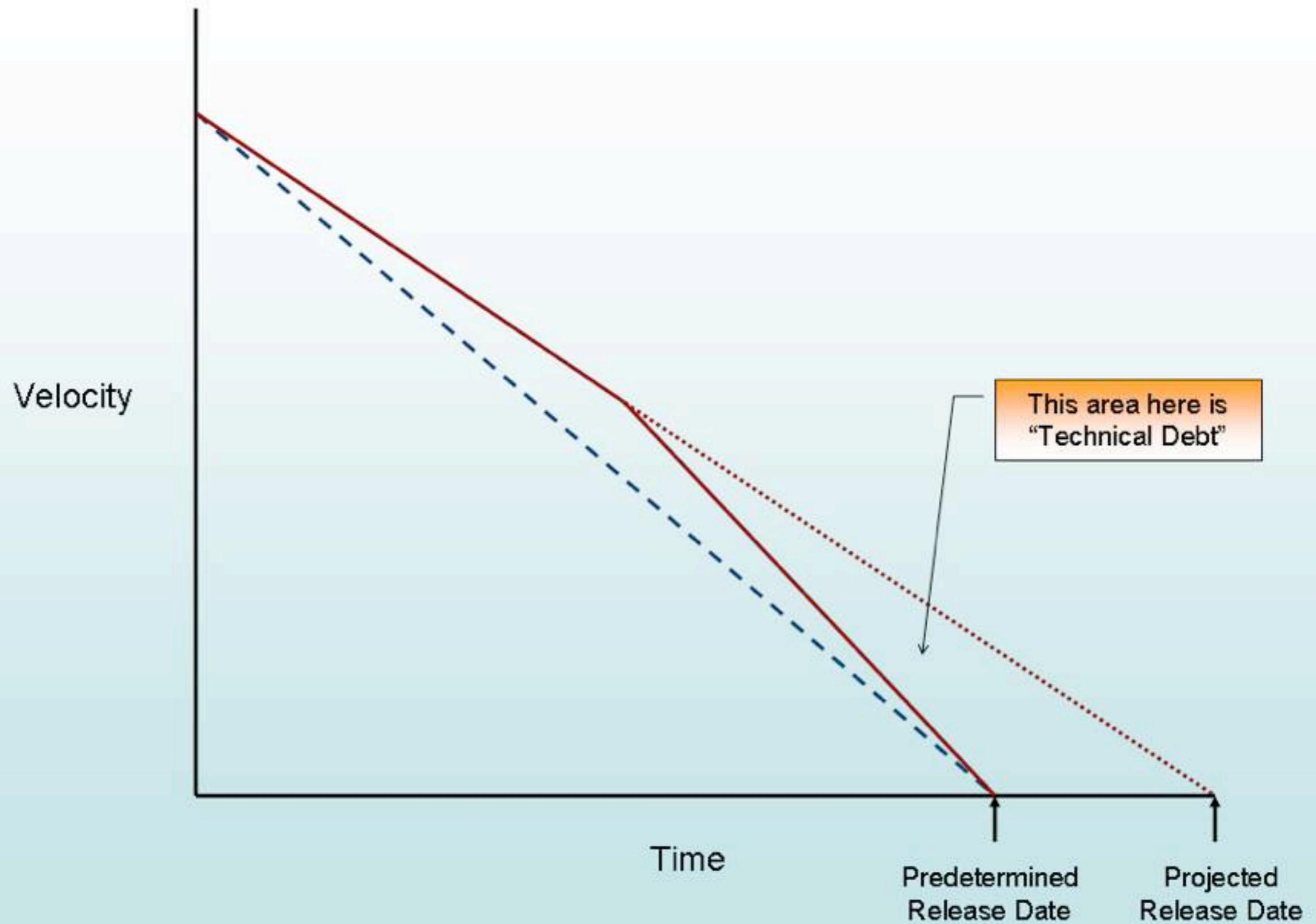
Name	Packages	Files	Classes	Methods	Lines	Conditionals
Cobertura Coverage Report	89% (8/9)	92% (23/25)	78% (57/73)	86% (139/162)	93% (864/930)	63% (1035/1639)

Coverage Breakdown by Package

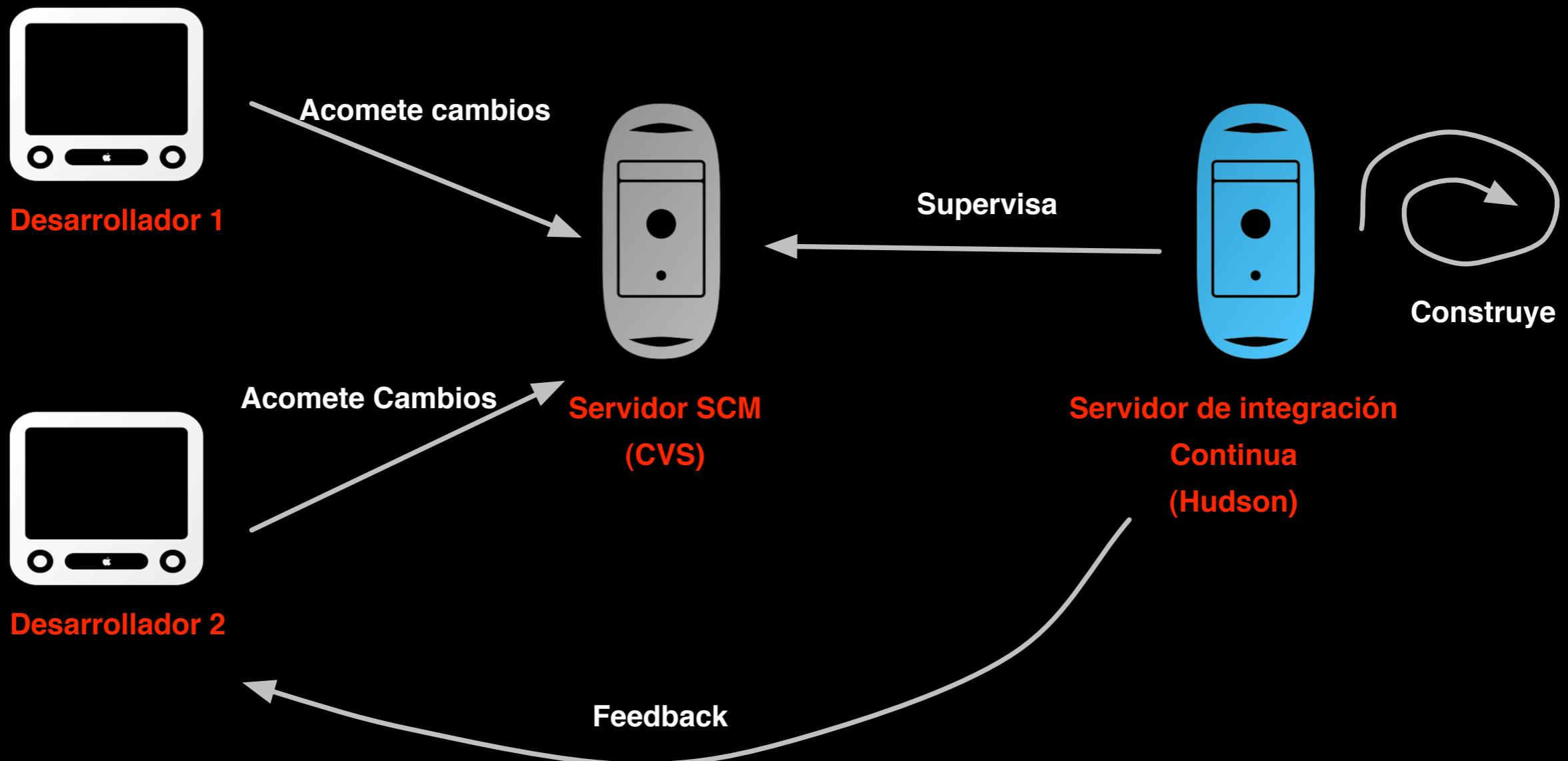
Name	Files	Classes	Methods	Lines	Conditionals
es.qiss.diccionarios.mcd.bus.ldm.emf.util.mcd	100% (7/7)	100% (7/7)	100% (22/22)	97% (63/65)	75% (41/55)
es.qiss.diccionarios.mcd.bus.ldm	0% (0/1)	0% (0/1)	0% (0/2)	0% (0/8)	100% (0/0)
es.qiss.diccionarios.mcd.bus.ldm.transformation.read.basic	100% (1/1)	100% (16/16)	100% (16/16)	89% (49/55)	54% (62/114)
es.qiss.diccionarios.mcd.bus.ldm.transformation	100% (1/1)	100% (20/20)	100% (62/62)	98% (618/630)	64% (846/1326)
es.qiss.diccionarios.mcd.bus.ldm.emf.util	N/A	N/A	N/A	100% (0/0)	100% (0/0)
es.qiss.diccionarios.mcd.bus.ldm.emf.util.ldm.read	100% (5/5)	100% (5/5)	94% (16/17)	94% (51/54)	85% (35/41)
es.qiss.diccionarios.mcd.bus.ldm.rochade.provider	100% (1/1)	100% (1/1)	100% (4/4)	100% (12/12)	50% (2/4)
es.qiss.diccionarios.mcd.bus.ldm.utils	50% (1/2)	6% (1/16)	10% (2/20)	18% (6/33)	7% (3/42)
es.qiss.diccionarios.mcd.bus.ldm.infrastructure	100% (2/2)	100% (2/2)	100% (5/5)	100% (25/25)	100% (24/24)
es.qiss.diccionarios.mcd.bus.ldm.emf.util.ldm	100% (5/5)	100% (5/5)	86% (12/14)	83% (40/48)	67% (22/33)

mbpl.litio.fme.mapl.sud.bcm.somenodiscip.esip.se	100% (2\2)	100% (2\2)	88% (40\48)	83% (55\33)
mbpl.litio.fme.mapl.sud.bcm.somenodiscip.esip.se	100% (5\5)	100% (5\5)	100% (52\52)	100% (54\54)
slidu.mapl.sud.bcm.somenodiscip.esip.se	20% (5\2)	9% (1\1)	10% (5\50)	18% (9\33)
slidu.mapl.sud.bcm.somenodiscip.esip.se	100% (T\T)	100% (T\T)	100% (T\T)	100% (TS\TS)

Refactoring = mejora
continua y temprana
del diseño del código



Integración continua



Construcción = Construcción + Tests + Despliegue

Build Executor Status

No.	Status
1	Building es.qiss.diccionarios.busemfrochade.core #200 

add description

All

+

S	W	Job	Last Success	Last Failure	Last Duration
		es.qiss.diccionarios.apirochade.groovy	1 day 1 hr (#102)	1 day 22 hr (#100)	1 min 31 sec 
		es.qiss.diccionarios.busemfrochade.common	1 hr 2 min (#119)	N/A	28 sec 
		es.qiss.diccionarios.busemfrochade.core	2 hr 0 min (#188)	N/A	5 min 58 sec 
		es.qiss.diccionarios.feature	1 day 22 hr (#82)	N/A	3,3 sec 
		es.qiss.diccionarios.mcd.bus.dbm.oracle	1 hr 2 min (#54)	N/A	2 min 54 sec 
		es.qiss.diccionarios.mcd.bus.excel.conceptual	59 min (#122)	N/A	17 min 
		es.qiss.diccionarios.mcd.bus.excel.mapping	1 day 22 hr (#108)	16 min (#123)	2 min 14 sec 
		es.qiss.diccionarios.mcd.bus.ldm	42 min (#149)	N/A	8 min 51 sec 
		es.qiss.diccionarios.mcd.bus.msl	6 days 9 hr (#119)	22 min (#152)	18 min 
		es.qiss.diccionarios.mcd.capaconceptual.ui	13 min (#87)	N/A	1 min 9 sec 
		es.qiss.diccionarios.mcd.transformation.adabasdbm2ldm	33 min (#291)	23 hr (#284)	11 min 
		es.qiss.diccionarios.mcd.transformation.ldm2msl	1 day 22 hr (#152)	13 min (#169)	23 sec 
		es.qiss.diccionarios.mcd.transformation.ldm2namedldm	59 min (#70)	2 hr 41 min (#68)	20 sec 
		es.qiss.diccionarios.metamodels.mcd	13 hr (#89)	N/A	12 sec 
		es.qiss.diccionarios.metamodels.rochade	2 hr 0 min (#94)	13 hr (#93)	7,1 sec 
		es.qiss.diccionarios.updatesite	1 day 22 hr (#99)	1 day 22 hr (#97)	2 min 4 sec 

Icon: S M L

Legend  for all  for failures  for just latest builds

Icon: Z W R

Build Result Total  Failed Total  Blocked Total  Pending Total 

Y para el cuarto...



Diferenciarse



No aburrir



Sin pasarse

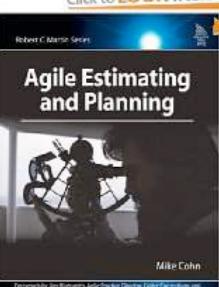
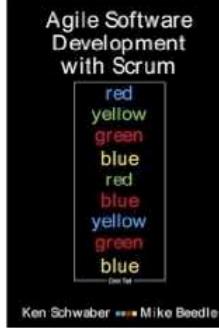
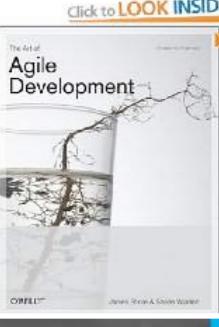
Proceso de Desarrollo

Elemento	Ejemplos
Descripción del proceso	<ul style="list-style-type: none">•Iteraciones cortas, completas•Niveles de planificación: de lanzamiento, de iteración, diario•Planificación priorizada•Historias de usuario, <i>story points</i>, velocidad
Configuración del equipo de desarrollo	<ul style="list-style-type: none">•Implicación del cliente

Aspectos técnicos

Elemento	Ejemplos
Uso de un servidor de integración continua	<ul style="list-style-type: none">• Detección temprana de errores• Software siempre listo para desplegar
Pruebas automáticas	<ul style="list-style-type: none">• Exhaustividad• Automatización de tests de aceptación
Uso de herramienta de cobertura de código	<ul style="list-style-type: none">• Definición de umbrales de cobertura a exigir: ejemplo, 90% código cubierto
Uso de herramientas de análisis de código	<ul style="list-style-type: none">• Detección de código sin documentar• Nombres de identificadores con tamaño suficiente• Detección de métodos demasiado largos

Referencias

	<p><i>Agile Estimating and Planning</i>. Mike Cohn. Prentice Hall, 2005</p>
	<p><i>Agile Software Development with Scrum</i>. Ken Schwaber. Prentice Hall, 2001</p>
	<p><i>The art of Agile Development</i>. James Shore and Shane Warden. Prentice Hall, 2007</p>
	<p><i>User Stories Applied: For Agile Software Development</i>. Mike Cohn. Addison-Wesley Professional, 2004</p>