

Introduzione a Scrum



*Prof. Paolo Ciancarini
Corso di Ingegneria del Software
CdL Informatica
Università di Bologna*

Obiettivi della lezione

- Presentazione del metodo agile Scrum
- Ruoli
- Rituali
- Artefatti
- Varianti di Scrum

Manifesto: valori “agili”

Individui e interazioni

meglio
di

Software che
funziona

meglio
di

Collaborazione
del cliente

meglio
di

Rispondere al
cambiamento

meglio di

Processi e strumenti

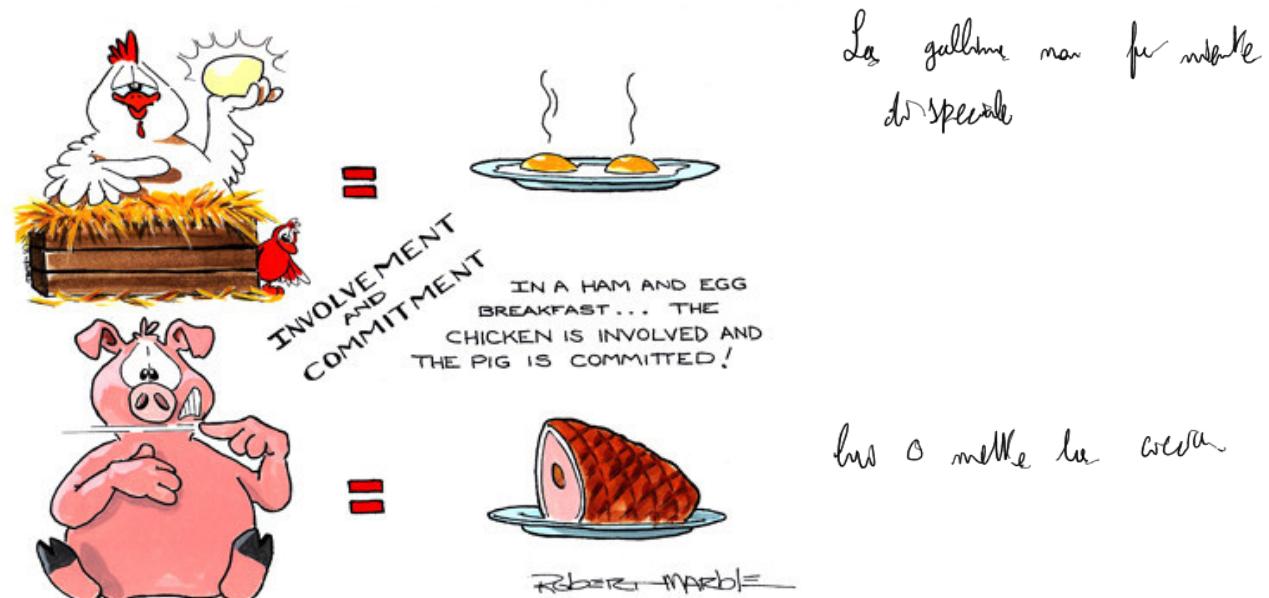
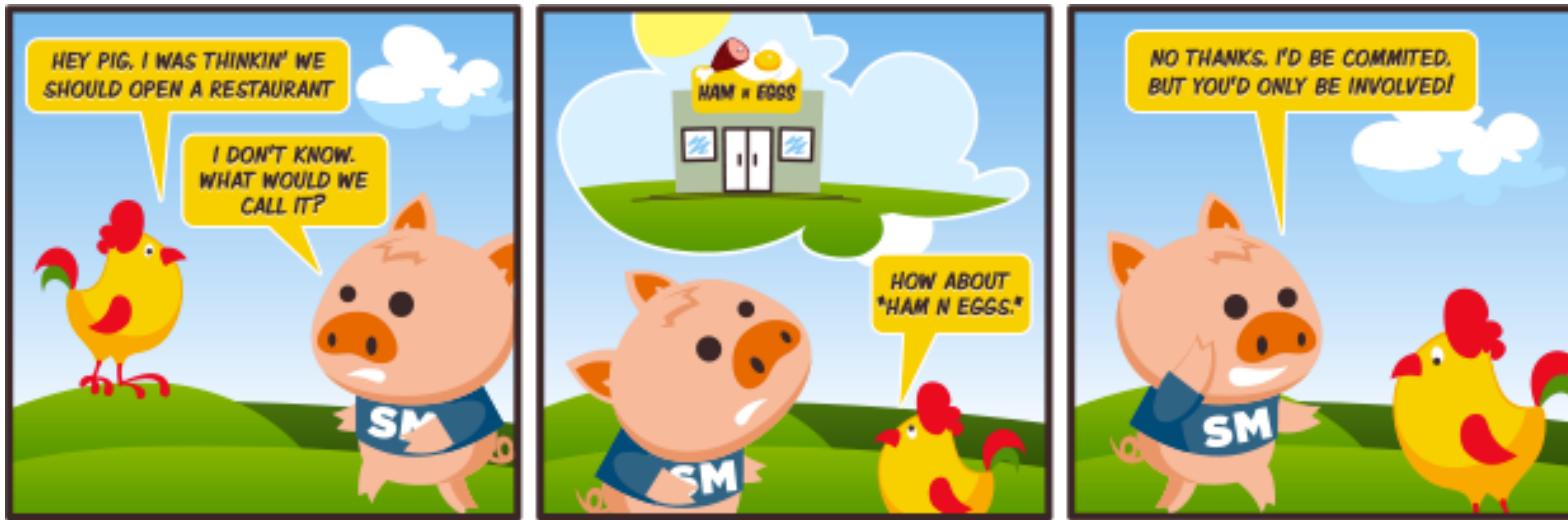
Documentazione
completa

Negoziazione
contrattuale

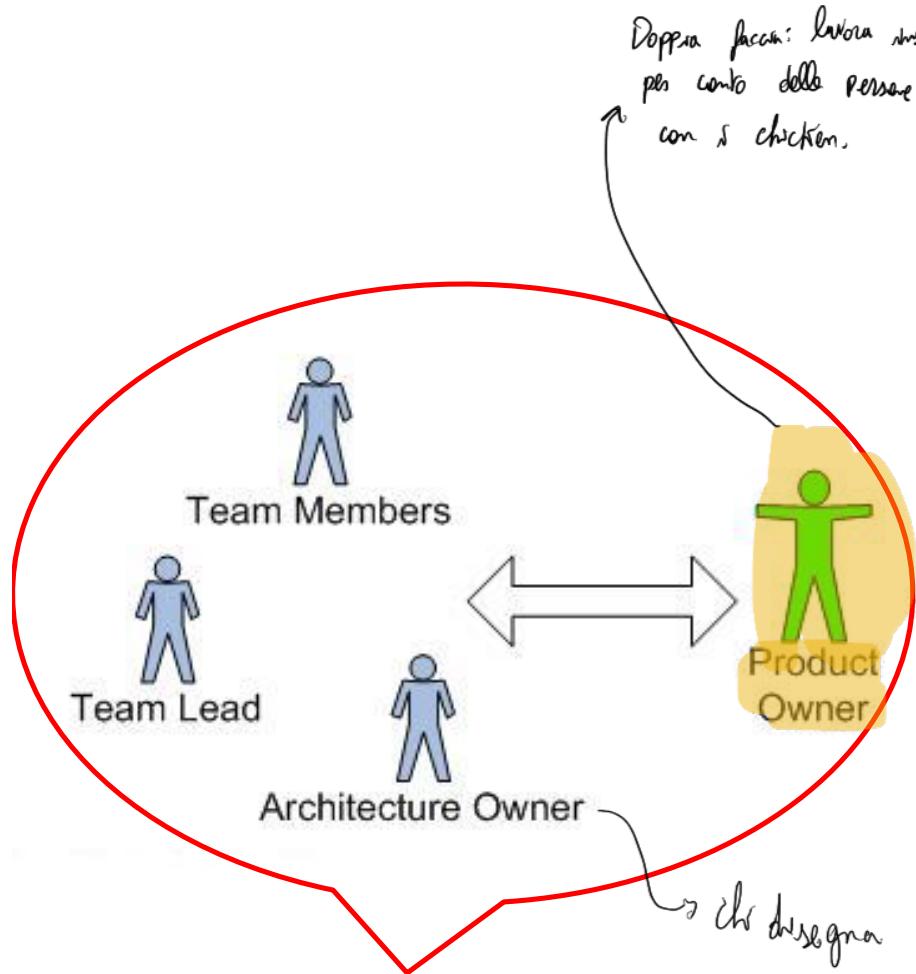
Seguire un piano

Fonte: www.agilemanifesto.org

Galline e maiali

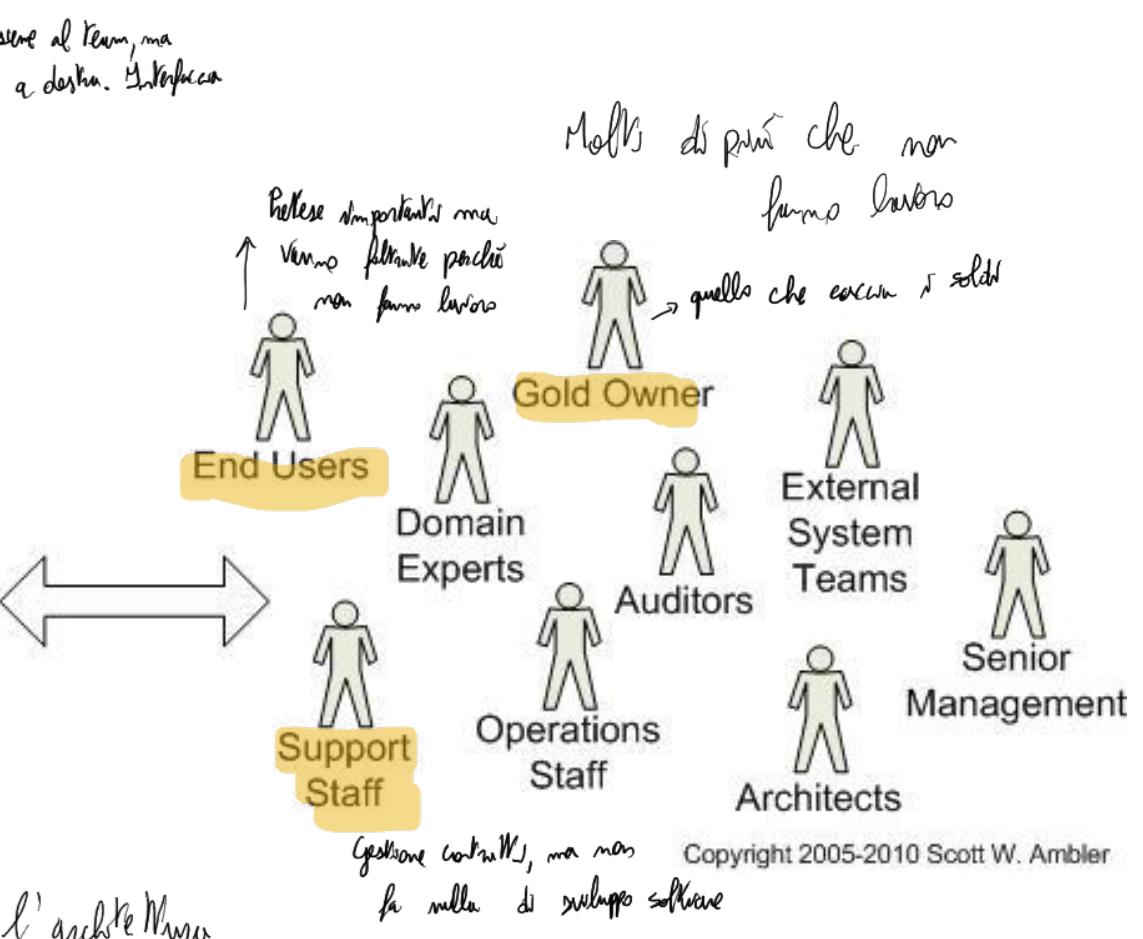


Scrum: ruoli pig e ruoli chicken



Ruoli "pig"

Buona al Senso



Ruoli "chicken"

Staffetta o pacchetto di mischia?

“The... ‘relay race’ approach to product development...may conflict with the goals of maximum speed and flexibility. Instead a holistic or ‘rugby’ approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today’s competitive requirements.”

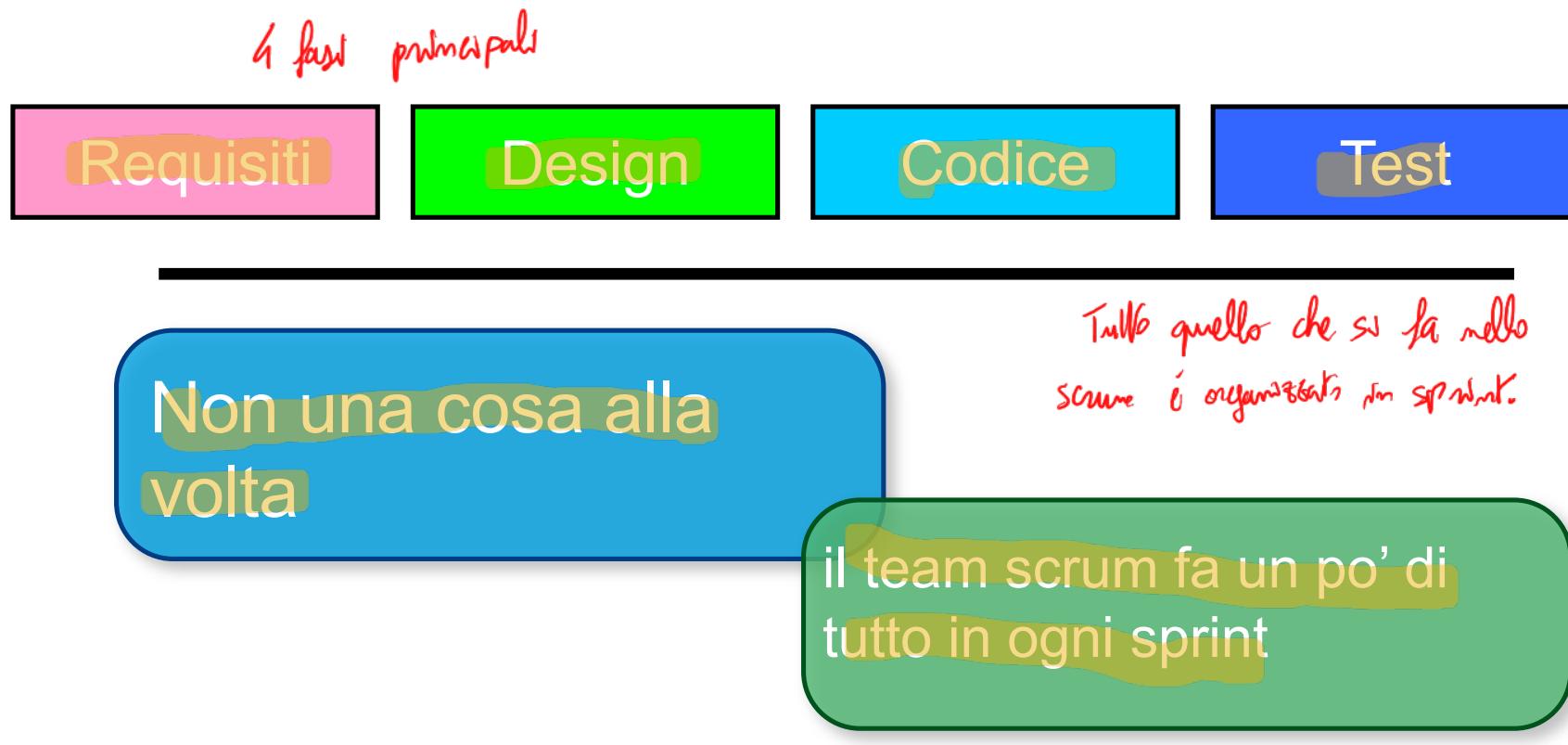
Hirotaka Takeuchi and Ikujiro Nonaka, “The New New Product Development Game”, *Harvard Business Review*, January 1986.



Non affida solo a una persona
ma a tutta la team.



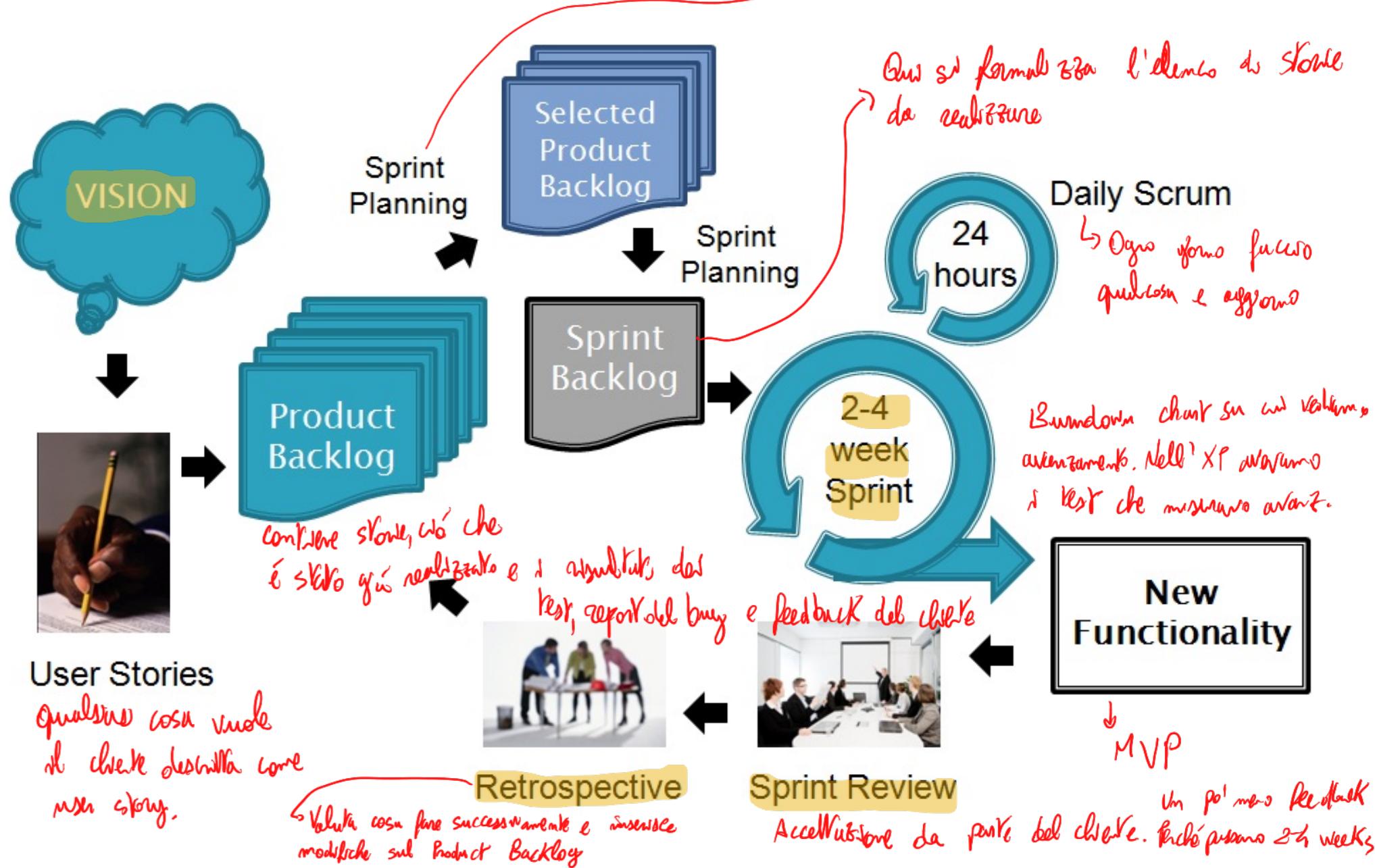
Dalla staffetta al pacchetto di mischia



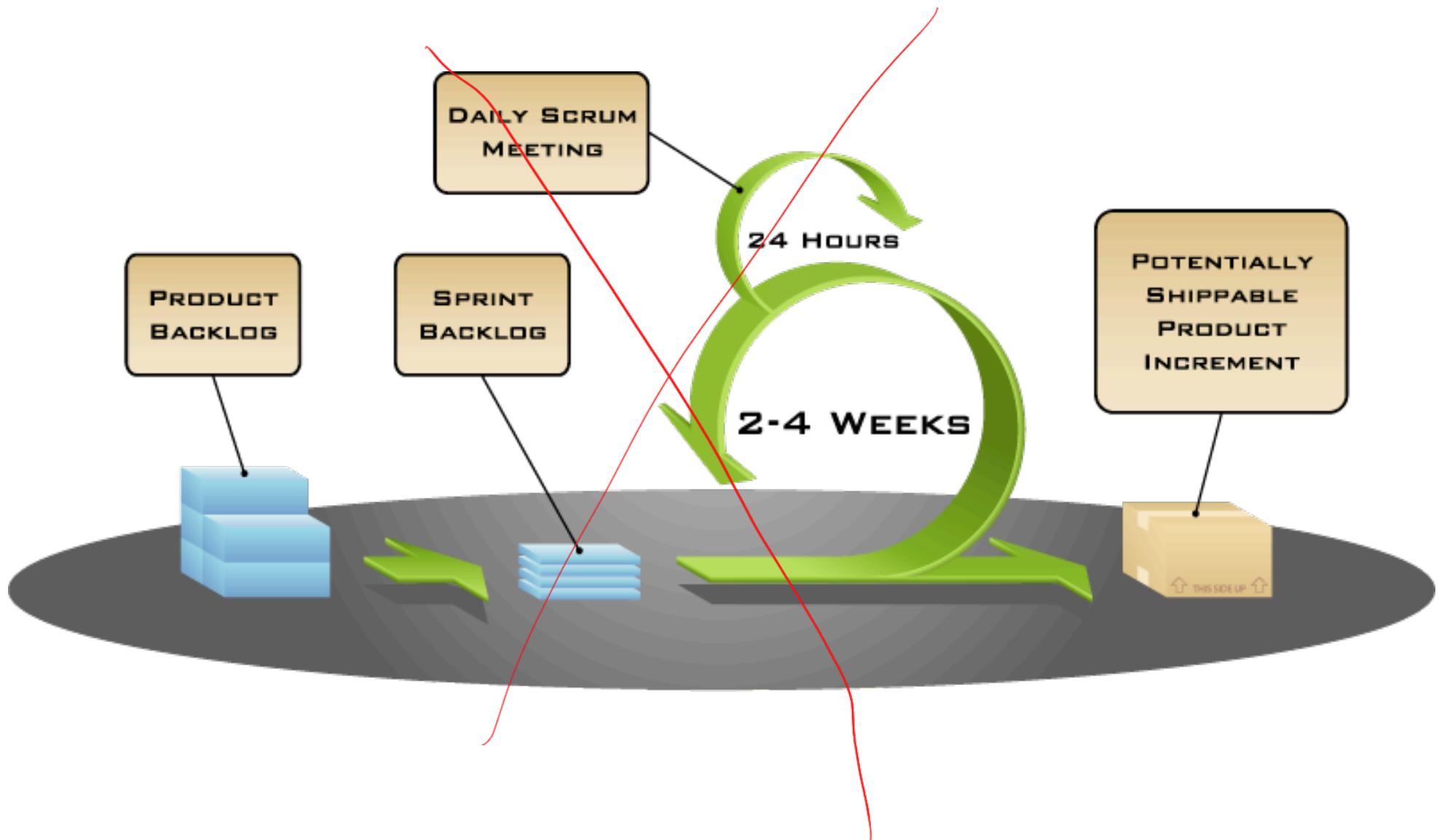
Source: "The New New Product Development Game" by Takeuchi and Nonaka. *Harvard Business Review*, January 1986.

Modelli di processo 2

SCRUM Overview



Il modello Scrum



Scrum in 100 parole

comeXP

- Scrum è un modello di processo per produrre software ottenendo il massimo valore utile nel minor tempo
- Permette al cliente di ispezionare rapidamente e ripetutamente ogni 3-4 settimane versioni funzionanti del software
- Il cliente definisce funzioni da realizzare e loro priorità. Il team di sviluppo decide quotidianamente il modo migliore di produrre le funzioni di più alta priorità.
- Ogni 3-4 settimane nasce una nuova versione che viene esaminata per decidere se continuare lo sviluppo con un altro sprint o produrne un rilascio

→ decisione sui ordini di fabb. è dell'iterazione

Nell'XP ogni fase è una la release.

Trasparenza

→ Solo i pgs devono avere visibilità totale

- Gli aspetti significativi del processo devono essere **visibili** ai responsabili del risultato finale (i “*pigs*”: il team, lo ScrumMaster, il Product Owner)
- La **trasparenza** richiede la condivisione e la comune comprensione di ciò che viene visto

Esempi:

- Tutti i partecipanti debbono condividere un **linguaggio comune di riferimento** al processo di sviluppo
- La **definizione** di ciò che è “Fatto” deve essere condivisa sia da chi esegue sia da chi deve accettare un task

→ Nella XP sulla base dei test d'accettazione.

Bisogna capire cosa significa completo → tutti devono essere d'accordo.

Modelli di processo 2

Scrum

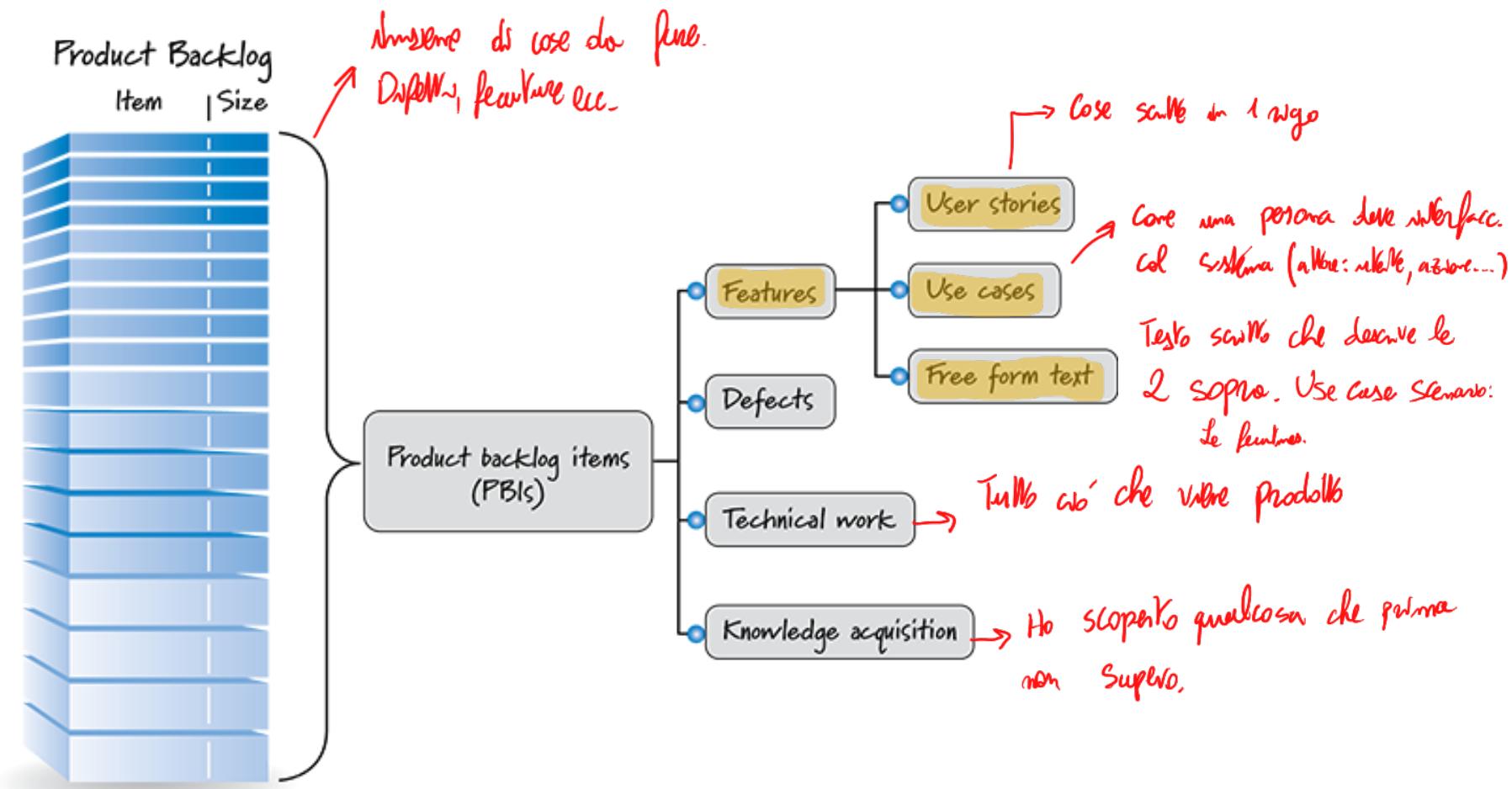
- Ideato da Schwaber e Sutherland [OOPLSA 1995]
- Una strategia di sviluppo di prodotti per cui un team lavora unito per raggiungere un obiettivo comune
- Iterativo e incrementale, basato su controllo empirico del processo

Punti chiave di Scrum

- Team di sviluppatori auto organizzante
- Metodo agile, parzialmente pianificato
- I requisiti sono storie elencate nel “product backlog”, che è una lista ordinata dal PO
- Sviluppo guidato da storie e test
- Il prodotto cresce in “sprint” di durata fissa
- Per ogni sprint, ogni persona del team sceglie i requisiti da realizzare da uno “sprint backlog”
- Meetings: reviews e “retrospettive”

↳ oltre ai daily scrum. Qui è proibito usare le slide.

Il product backlog

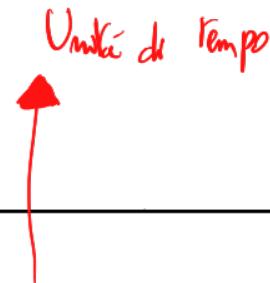


Il product backlog è una lista di user stories

ToDo List			
Story	Estimation	Priority	
As a user I want to be able to reset my password	1	1	
As a user I want to edit items	3	2	
As a user I want to export data	2	3	
As an administrator I want to define KPI's for my sales team	4	4	
As a user I want to view my data on mobile	5	5	
As an administrator I want to send alerts when new leads come in	2	6	
As a user I want to create a report of my data	5	7	
As a user I want to update my reminder settings when a date is added	3	8	
As a user I want filtering enhancements	4	9	
As an administrator I want to configure views of data	5	10	
Total	34		

Oggetto: realizzare al massimo numero di richieste a priorità con rischio minimo

↓
Non sempre priorità
ben diversa

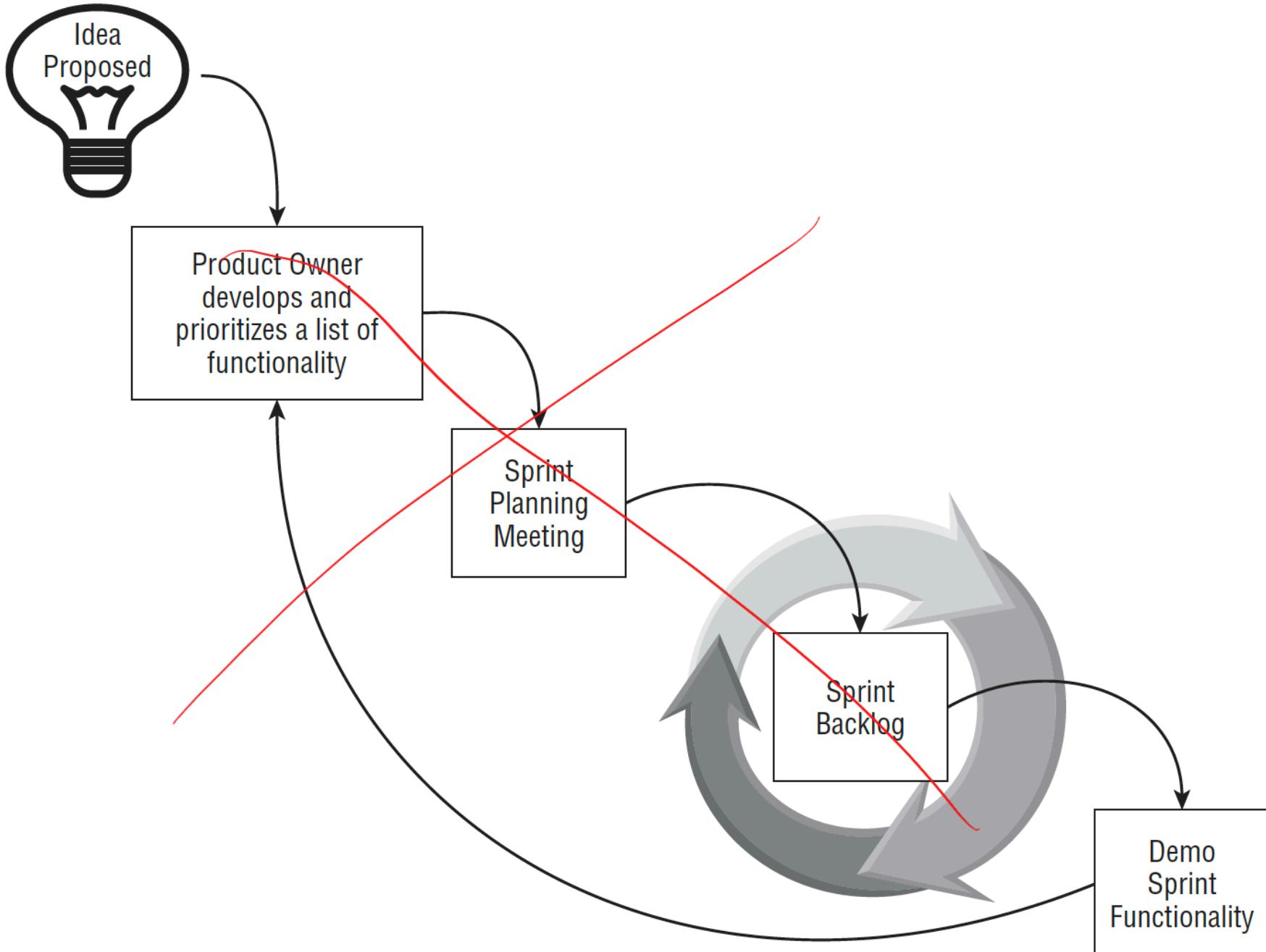


Max Priority

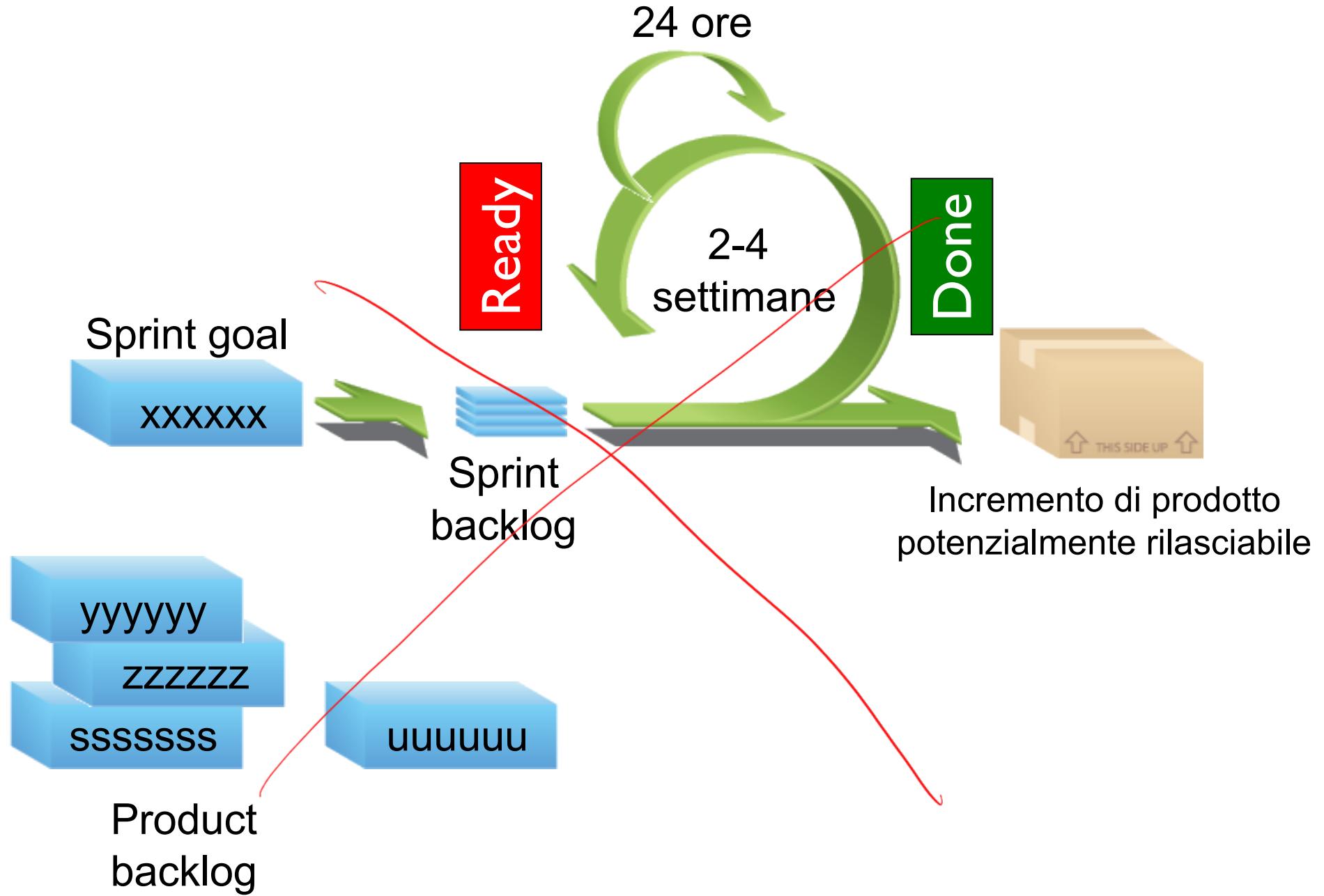
User story map: come affettare il product backlog



Il ciclo di base in Scrum



Un processo Scrum si compone di **sprint**

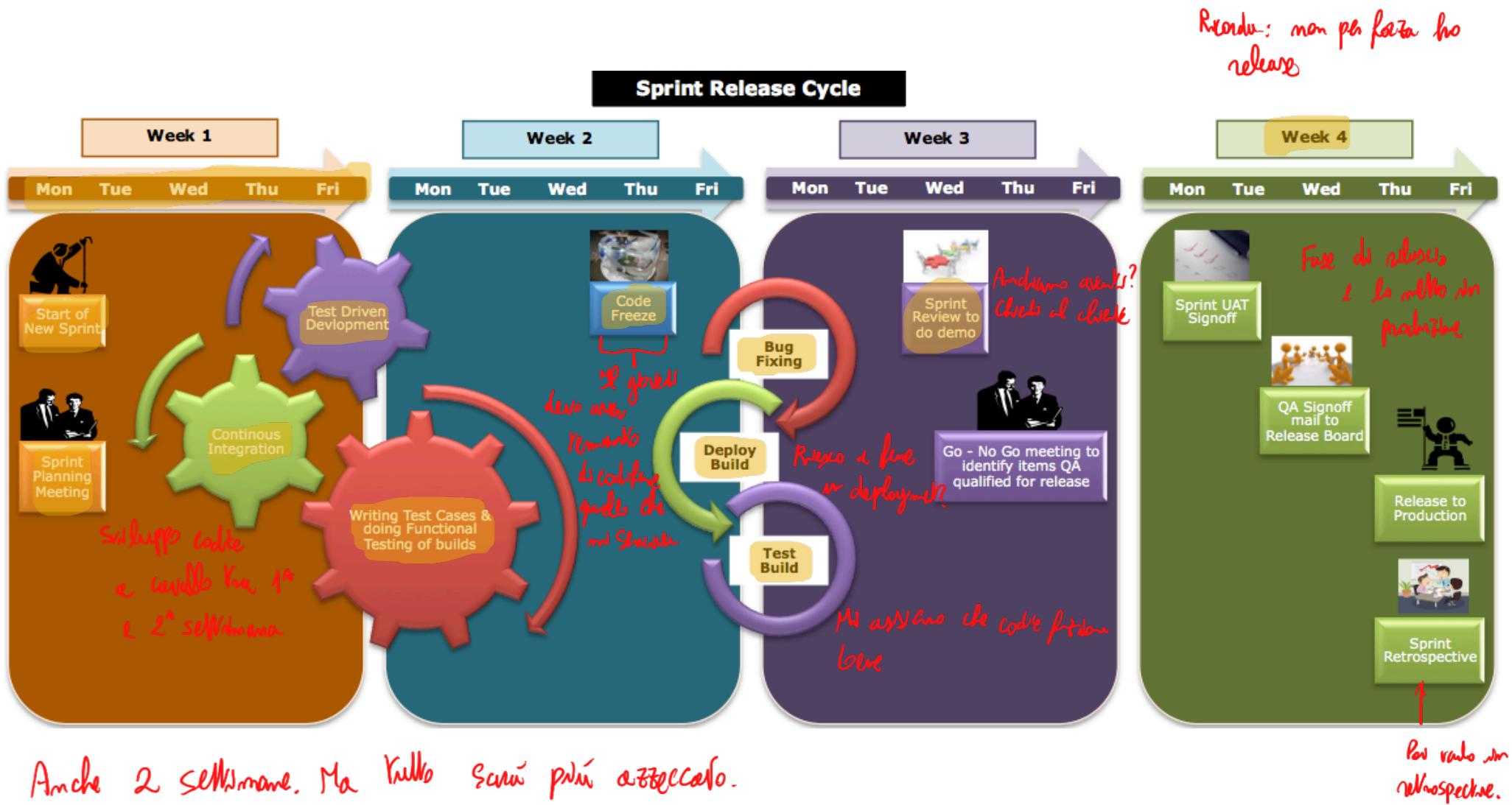


L'iterazione principale: lo sprint

- Un processo Scrum è una serie di sprint (analoghi delle iterazioni RUP)
- Durata 2-4 settimane o un mese max
- Durata costante (migliora il ritmo) *se dura 2W deve sempre durare 2 W.*
- Ogni sprint include design codifica e test
- Ogni sprint estrae funzioni “Ready” dal product backlog e aggiunge codice “Done” al prodotto da mostrare al cliente

Done: già fatto. Ready: Pronto per essere analizzab dello sprint, i cui requisiti sono suff chiari e con stima che mi permette di gestirli. (Altri magari non sono ready). Es. limit search to one field e Search by keyword.

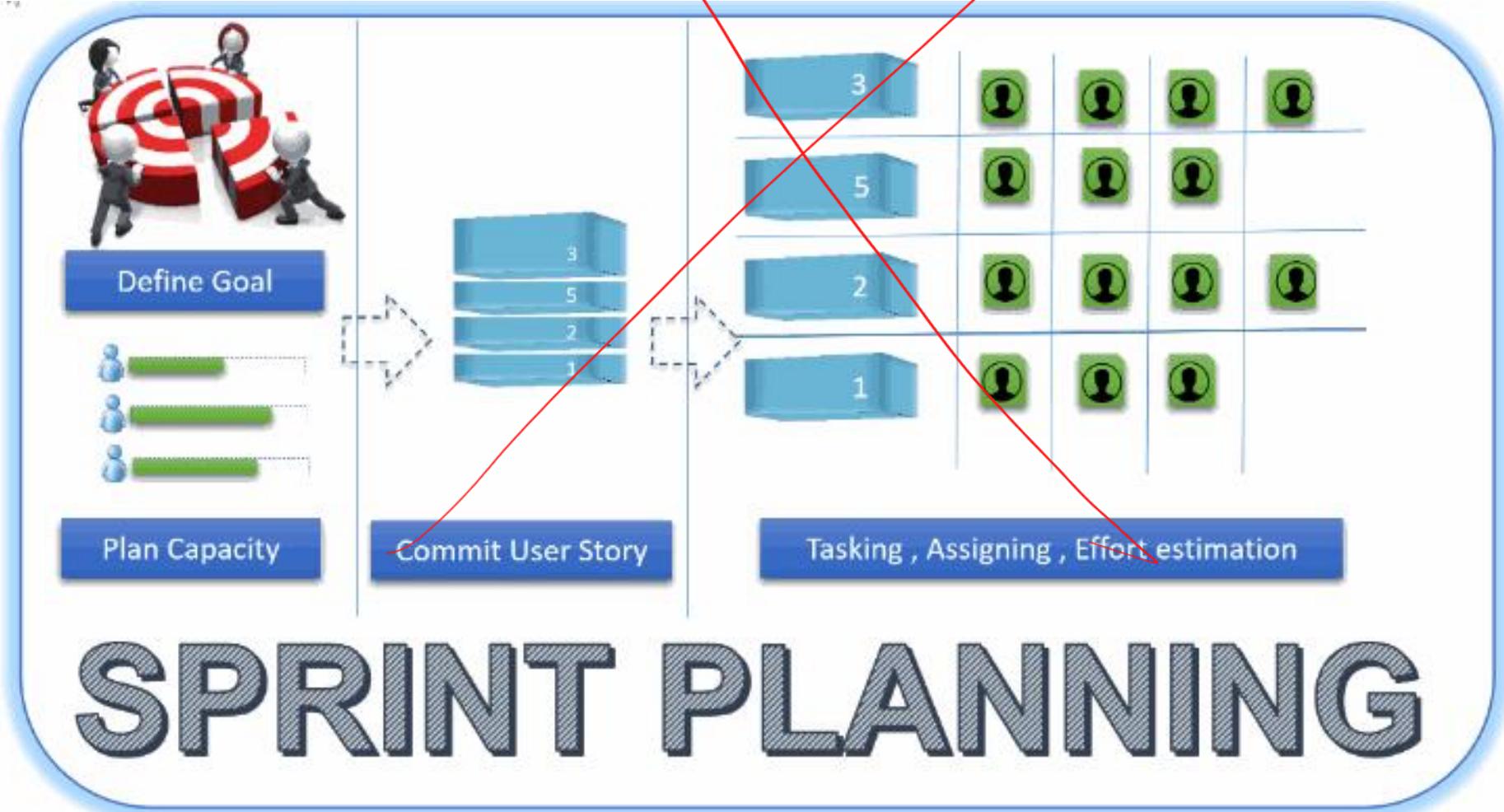
Ciclo delle release in Scrum



Anche 2 settimane. Ma tutto Scrum più attaccato.

Fonte: www.scrumalliance.org

La pianificazione dello sprint



Pianificare lo sprint

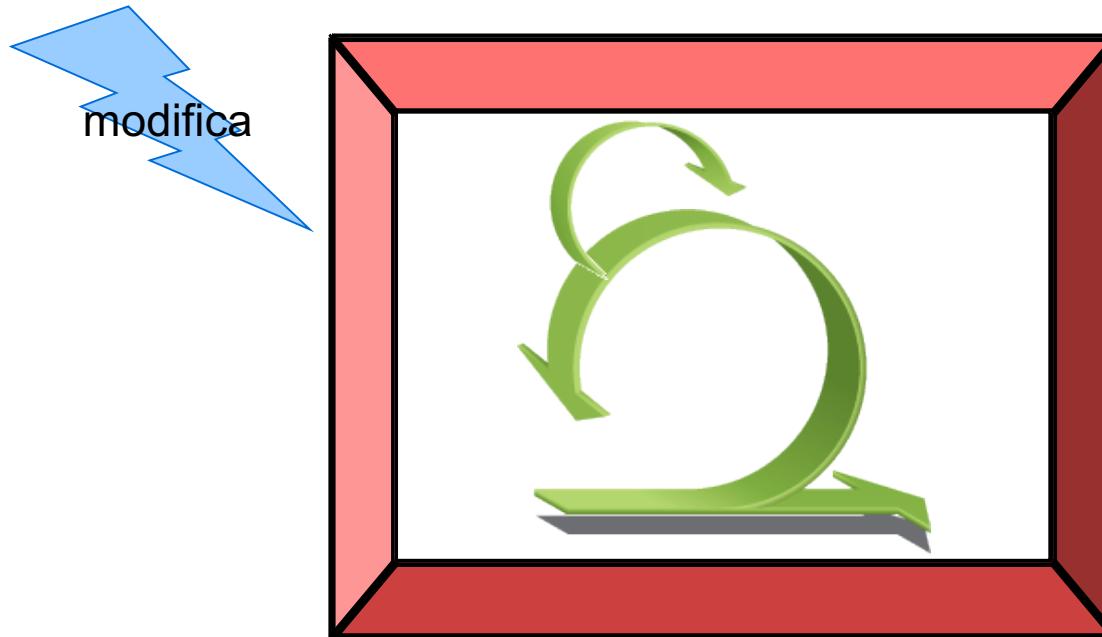
→ Pianificaz. delle attività settimanali.

- Ogni sprint inizia con una pianificazione di gruppo (PO, SM, Team) che dura almeno un'ora per ciascuna settimana di sprint
- Il Team decide quale parte del backlog di prodotto viene estratta per diventare il backlog di sprint: alcune user stories vengono scelte assieme al PO *Product owner*
- Viene concordato da PO e Team la «Definition of Done» per tutti gli elementi del backlog di sprint
- Il Team stima quanto lavoro riuscirà a fare durante lo sprint, tenendo presente la sua «velocità» di sviluppo

NOTA: nell'estrema non tocca cambiare personalità

Niente cambiamenti al team durante lo sprint

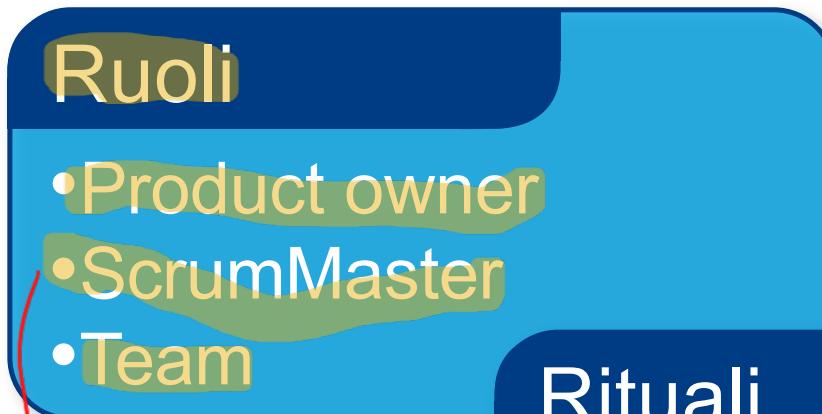
Ora per esempio si lavora su componenti e poi chi ne serve prima se ne va.



Pianificare le durate degli sprint in modo da garantire che il team non cambi

È non possono cambiare i responsabili nello sprint. Se si scopre nel mentre qualcosa, no la produco e poi tra pochi le modifica. Tra uno sprint e l'altro si cambiano cose.

Scrum: ruoli rituali artefatti



esponente Scrum.

Rispondo da persone che lavorano
nel team

Rituali

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Serve ai operatori che mettono
in moto processi di lavoro.

Artefatti

Documenti che
deve sempre
produrre

- Product backlog
- Sprint backlog
- Burndown charts

Scrum: i ruoli

Ruoli

- Product owner
- ScrumMaster
- Team

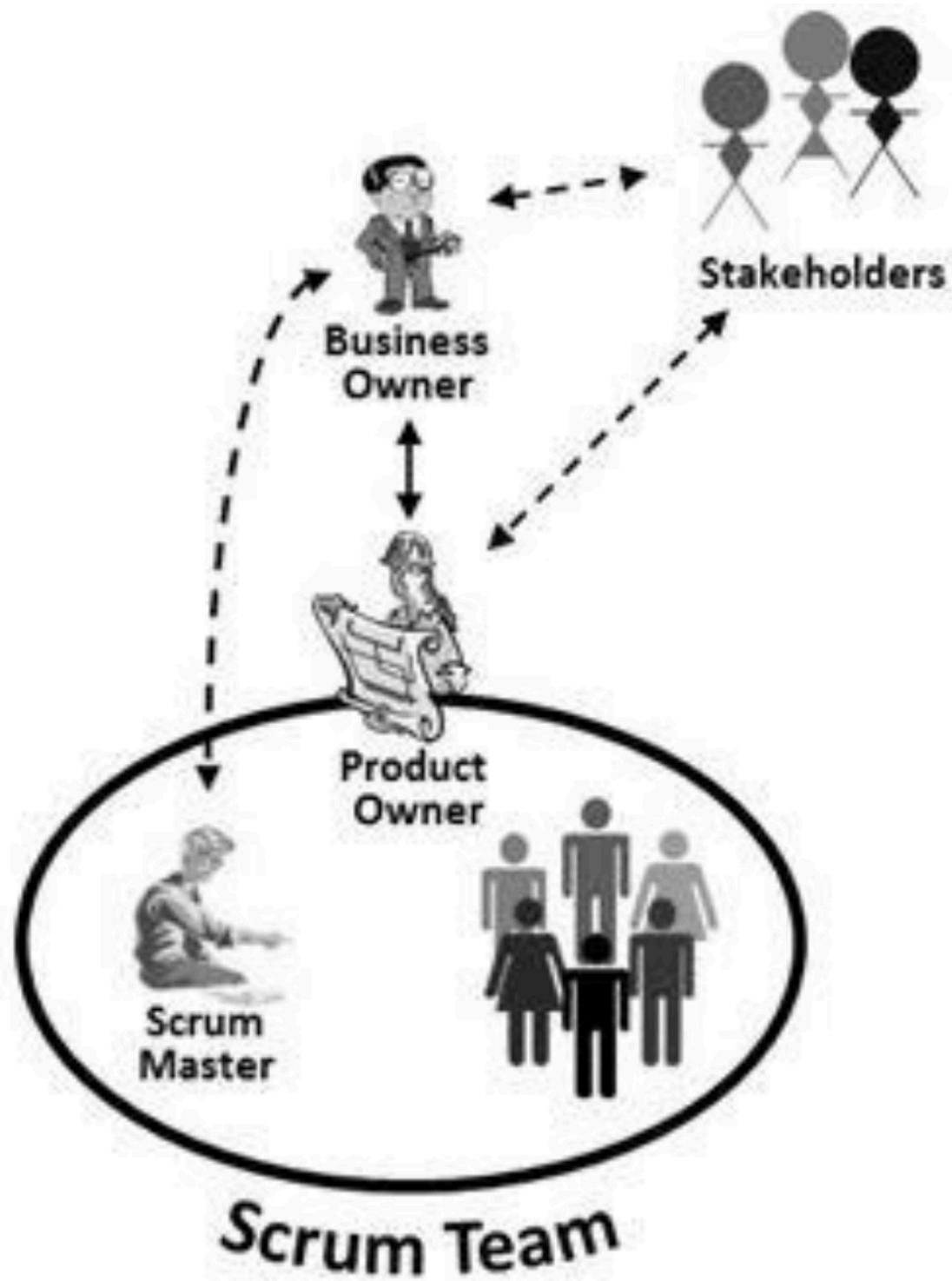
Rituali

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artefatti

- Product backlog
- Sprint backlog
- Burndown charts

I ruoli principali in Scrum



I ruoli core

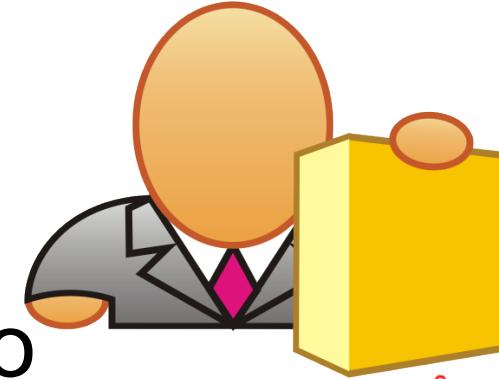
- **Product Owner**: rappresenta gli stakeholder (la voce del cliente), scrive il *product backlog* in forma di *user stories*; è responsabile della definizione di "Fatto".
- **Team di sviluppo**: 3-9 membri con diverse abilità, collettivamente responsabili della consegna di un PSI (Potentially Shippable Increment)
- **Scrum Master**: facilita la corretta esecuzione del processo, elimina gli ostacoli *Fa gli interessi del team*
 - Meglio se non è coperto dalla persona con ruolo Product Owner
 - Non ha responsabilità di gestione del personale o di project management "tradizionale"

Di solito non programma.

Non ha il potere di licenziare

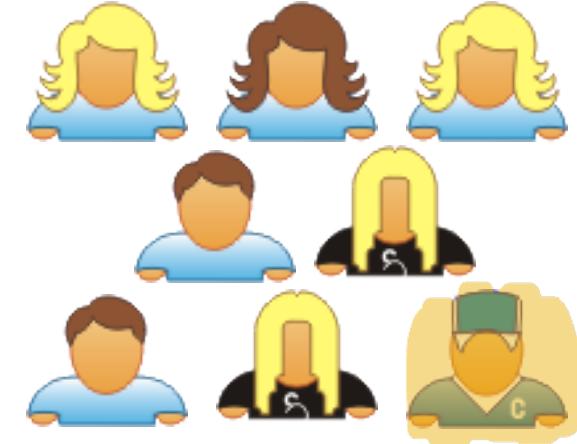
Product owner

Sono praticamente quelle relative all'analisi dei requisiti



- Definisce le feature del prodotto
- Decide i rilasci: date e contenuti
- Responsabile del valore del prodotto (ROI)
Il ritorno dell'investimento
- Mette in priorità le feature rispetto al loro valore di mercato
Le probabilità non sono stabili, non può farlo né chiude né tem.
- Per ogni iterazione rivede lista delle feature e loro priorità ove necessario
- Accetta o rifiuta i risultati mediante la definizione di “Fatto”

Il team di sviluppo



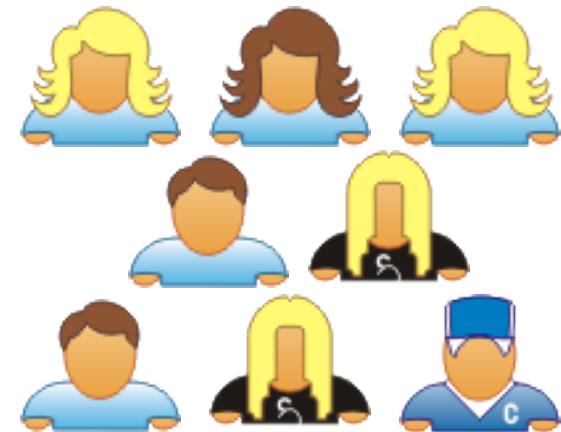
- Auto-organizzante
- Modifiche al team solo in sprint diversi: è stabile per tutto lo sprint
- Tutti presenti nello stesso spazio di lavoro
- Include uno Scrum Master come coach e mentore

I membri del team

→ Max 10

- 5-9 persone
- Varie specialità:
 - Programmatori, testatori, user experience designers, database administrator, ecc.
- Impegno full-time
- Con eccezioni (e.g., database administrator, costoso)

↓
1 persona non può partecipare a più team contemporaneamente



Il team

Non diversificare in sottoteam
Non c'è una gerarchia

Il team include sviluppatori capaci di produrre un incremento del prodotto “Fatto” alla fine di ogni sprint.

Il team **si autogestisce** ed ha le seguenti caratteristiche:

- I membri **si auto-organizzano**; nessuno, nemmeno lo Scrum Master, può dire al Team come trasformare il Product Backlog in incrementi di funzionalità potenzialmente rilasciabili;
- I team sono **cross-funzionali**, cioè **includono tutte le abilità necessarie a creare un incremento di prodotto**;
- I membri del Team **sono tutti Sviluppatori alla pari, senza nessuna eccezione**;
- Il Team è **unitario e non include sottogruppi**, senza nessuna eccezione per attività o domini particolari;
- I membri del Team **possono avere qualche specializzazione personale**, ma l'intero Team resta responsabile dello sviluppo.

ScrumMaster

Concetto di Team Leader nell'XP, ma
nella Scrum bisogna avere certificazioni



- Responsabile dei valori e pratiche Scrum
- Rimuove gli ostacoli al processo
- Assicura il benessere del team
- Supporta la cooperazione di ruoli e funzioni
- Protegge il team da interferenze esterne



ScrumMaster

- Serve il team (non lo dirige)
- Non può licenziare nessuno
- Responsabile del processo Scrum
- Meglio se non partecipa come programmatore

Scrum: i rituali

Ruoli

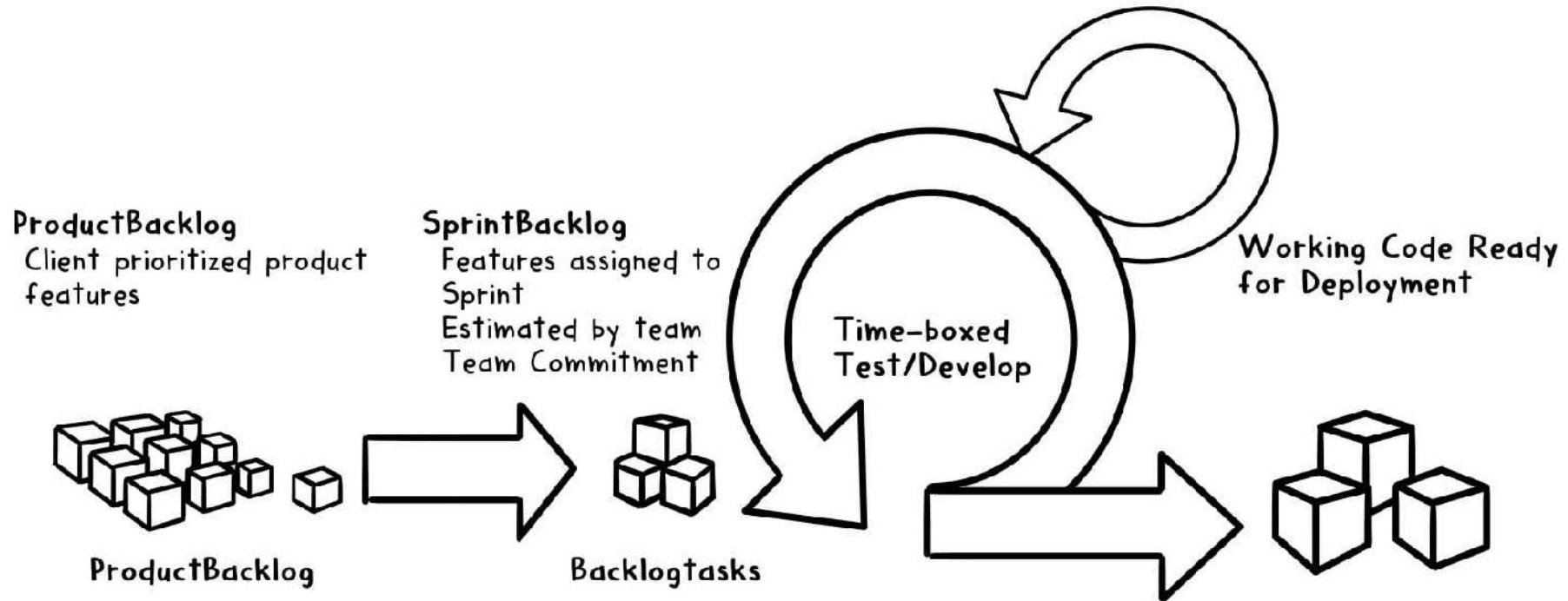
- Product owner
- ScrumMaster
- Team

Rituali

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Artefatti

- Product backlog
- Sprint backlog
- Burndown charts



Sprint Planning Meeting

- Review Product Backlog
- Estimate Sprint Backlog
- Commit

Dévono essere molti meno
sprint back log

Daily Scrum Meetings

- Done since last meeting
- Plan for today
- Roadblocks/Accelerators?

↓
15 min. ritratti
Close che hanno bloccato?

Sprint Review Meeting

- Demo features to all
- Retrospective on the Sprint
- Adjustments

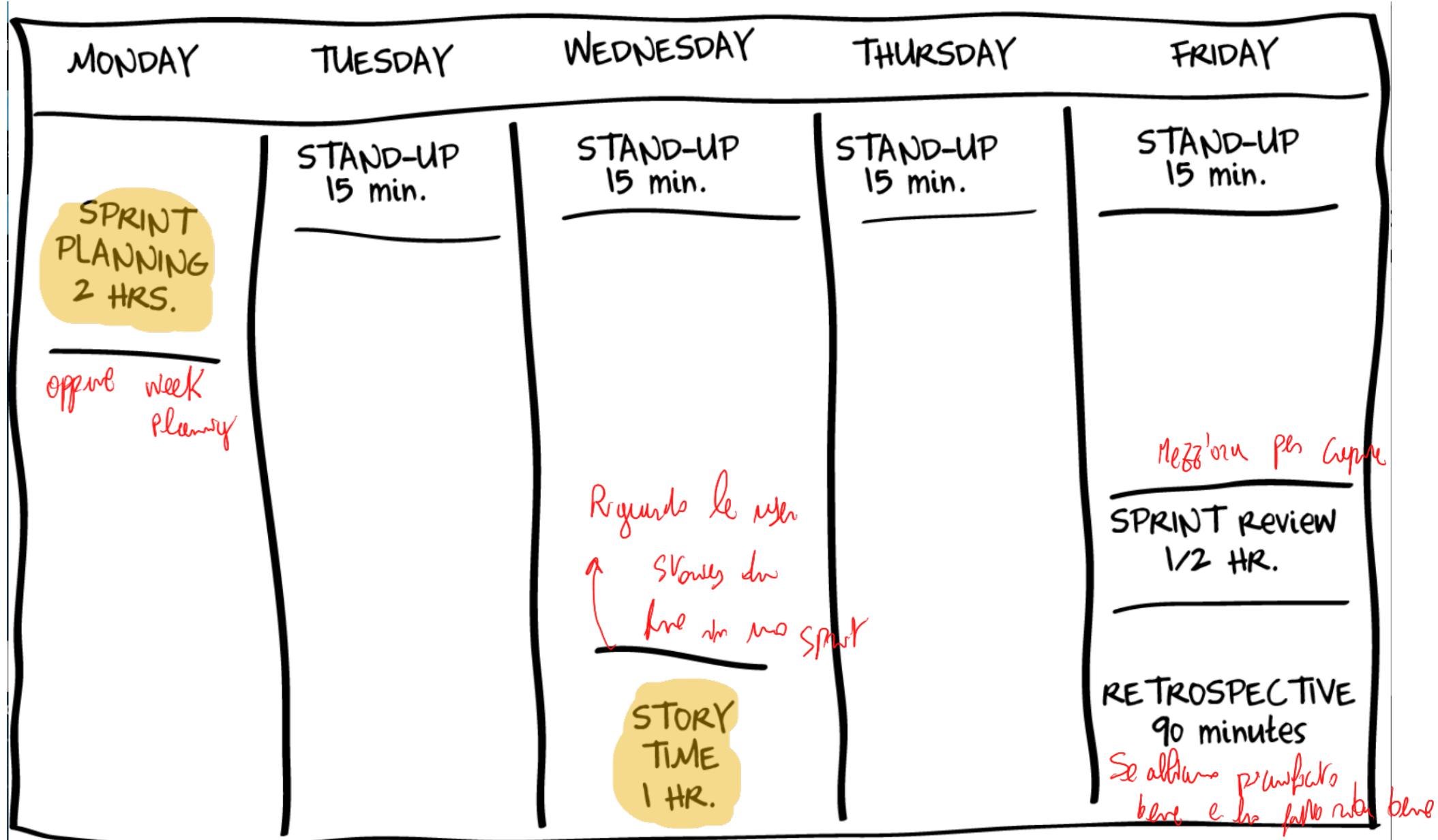
↓
che ho fatto dopo lo sprint?

I rituali Scrum

- **Sprint planning:** PO e Team decidono cosa fare (sprint backlog) e come aggiornare il product backlog
 - Durata: circa un'ora per ogni settimana dello sprint
- **Daily scrum o stand-up:** ogni membro del Team dice cosa ha fatto, cosa pianifica per oggi, che impedimenti ha trovato
 - Durata: Max 15 minuti, in piedi
- **Sprint Review:** riguarda il prodotto: cosa è stato o non è stato completato in questo sprint, demo
 - Durata: 4 ore al massimo
- **Sprint Retrospective:** riguarda il processo: cosa è andato bene e quali impedimenti sono stati trovati
 - Durata: 3 ore al massimo

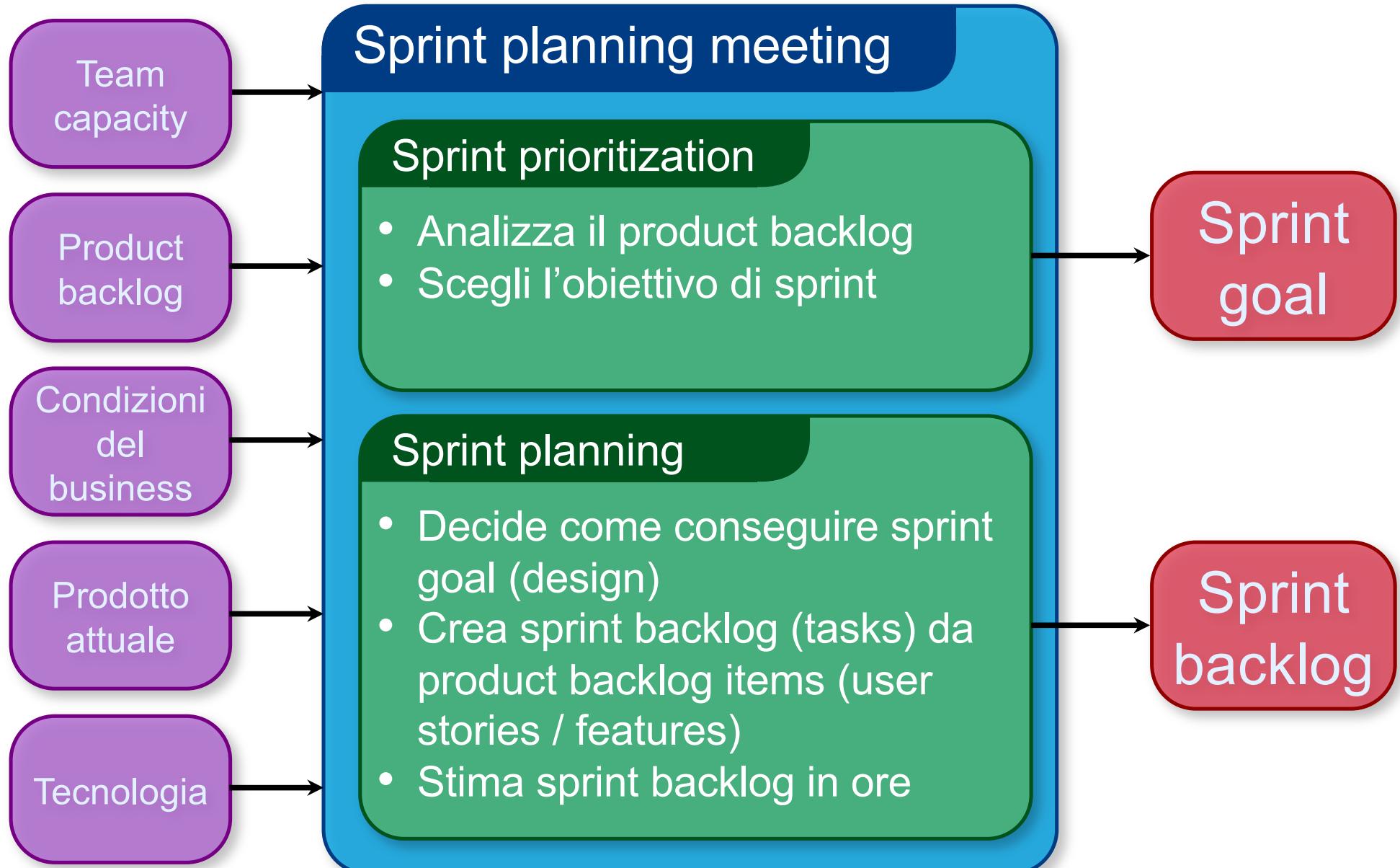
Come viene gestito elenco dei work

Struttura quotidiana di uno sprint su base settimanale

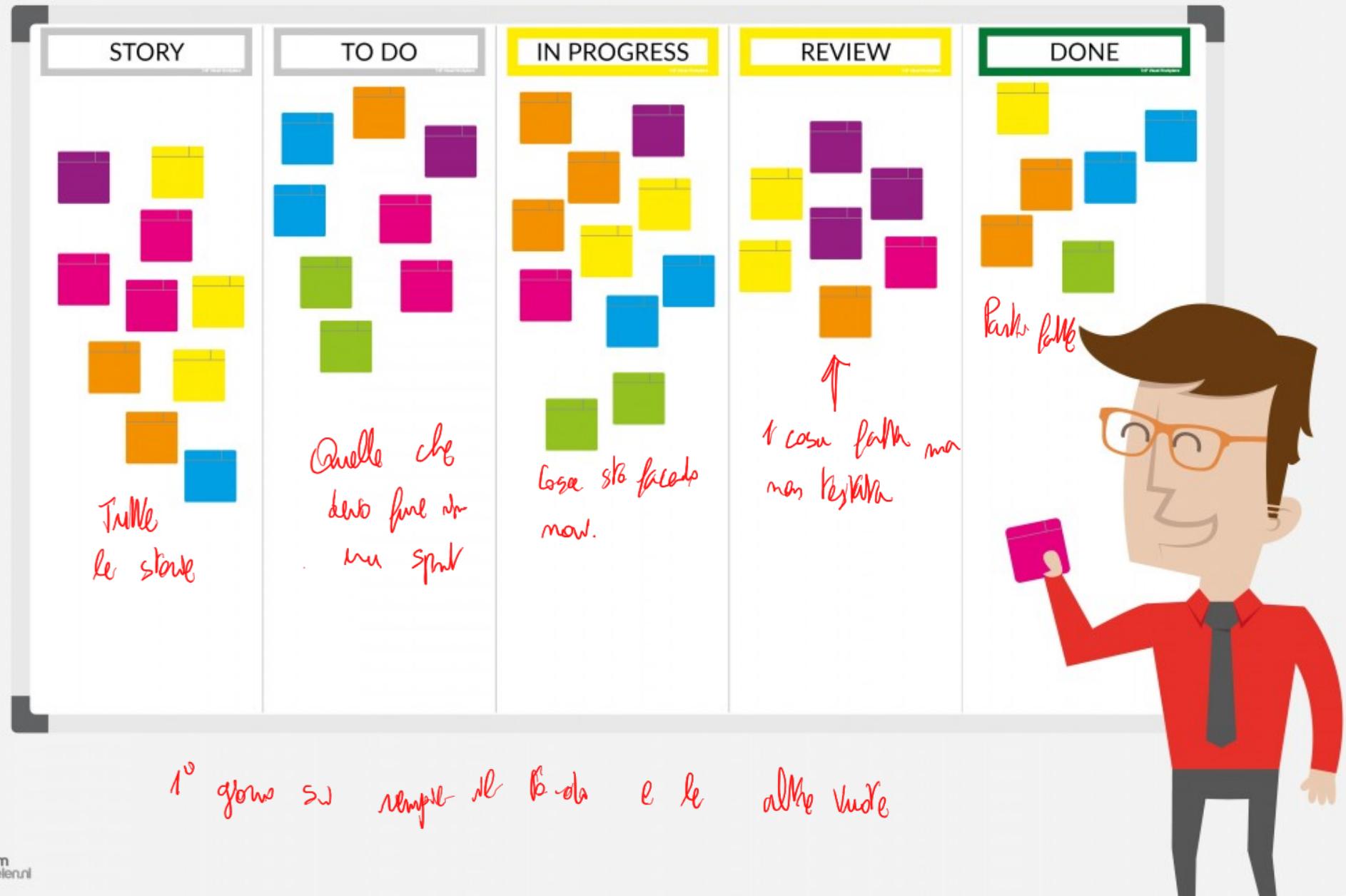


Il meeting di pianificazione dello sprint

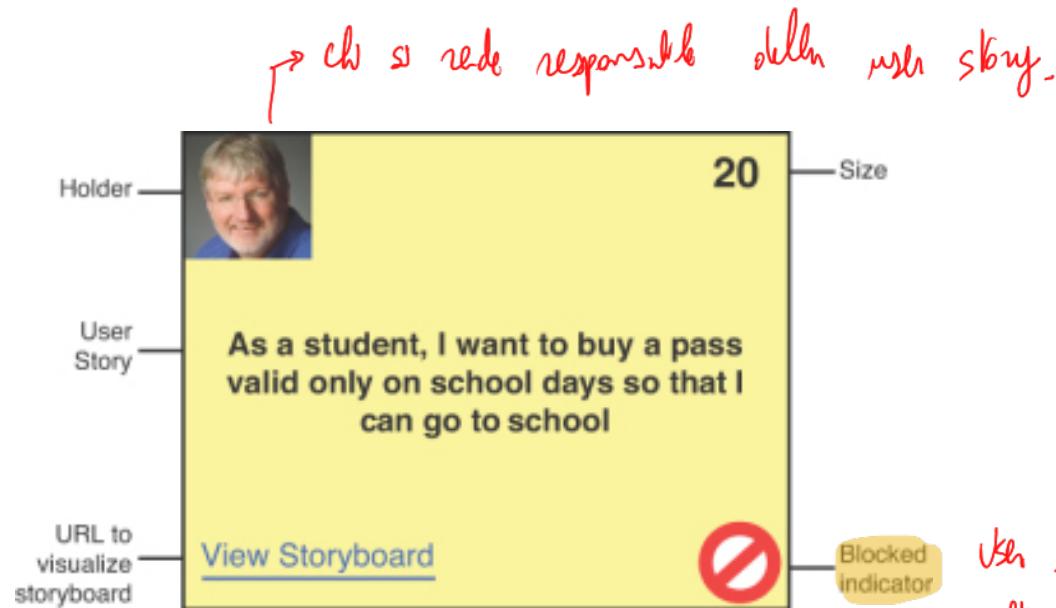




The scrum board



La user story

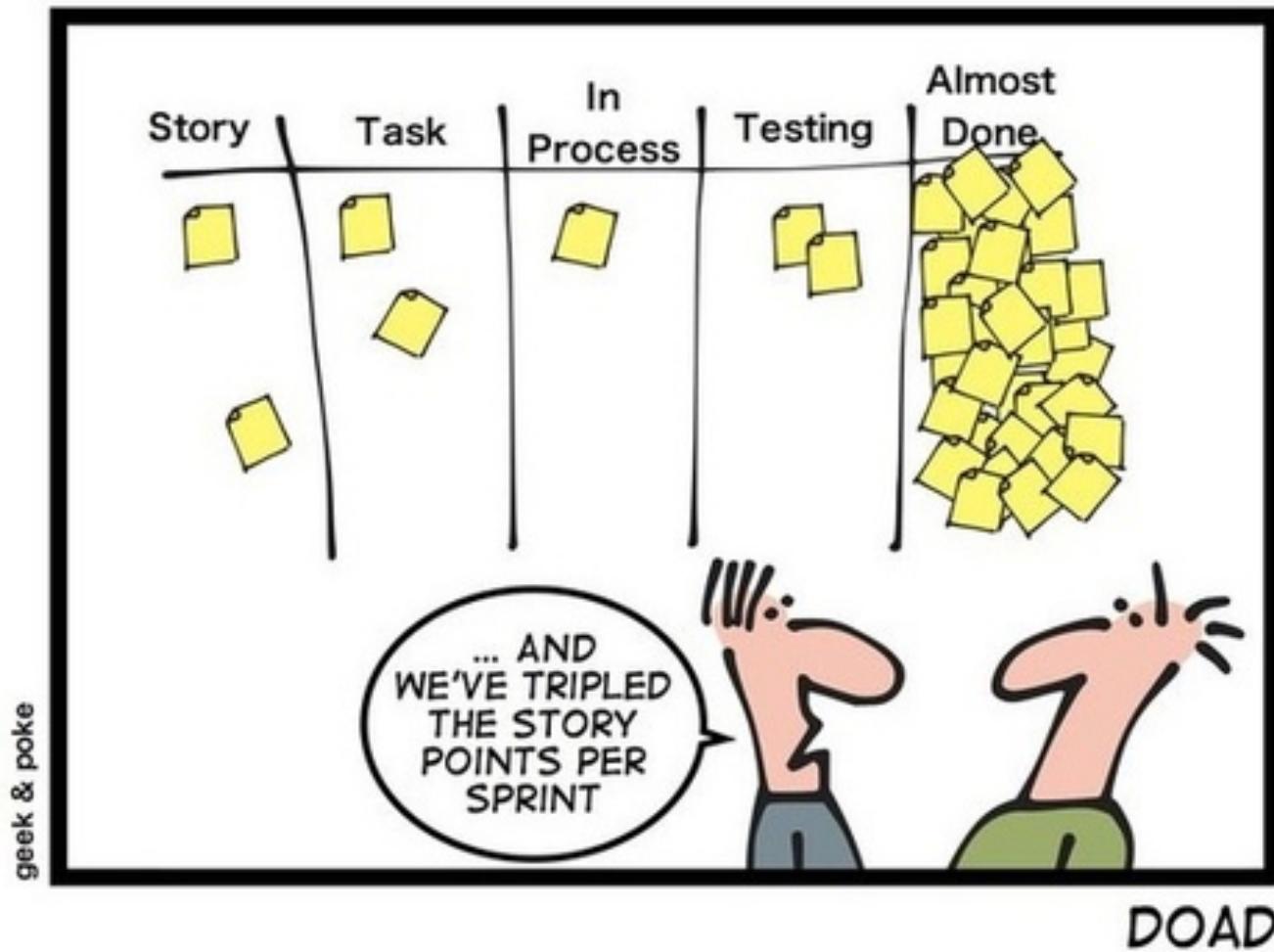


Una Story non può essere sviluppata perché non è finita. Es. non era ready ma me ne sono accorti sulla Story Board.

La definizione di Pronto (definition of Ready)

- Una storia è Ready quanto tutti i membri del Team la capiscono
- È dettagliata abbastanza da poter essere poi testata
→ loro mette i numeri.
- Il PO deve comprenderne il valore per poterla ordinare in priorità con le altre
dove misure ~ dove una priorità deve sapere che pronto ha rispetto alle altre
- Il Team dev'essere in grado di farne una stima in modo da completarla in uno sprint
- La storia è indipendente: il Team può lavorarla avendone il pieno controllo

La definizione di Fatto (Definition of Done)



Definizione di “Fatto”

La lavorazione di una funzione che realizza una user story non è finita a meno che non soddisfi il PO: la definizione di “Fatto” è concordata dal Team con il PO all'inizio del processo

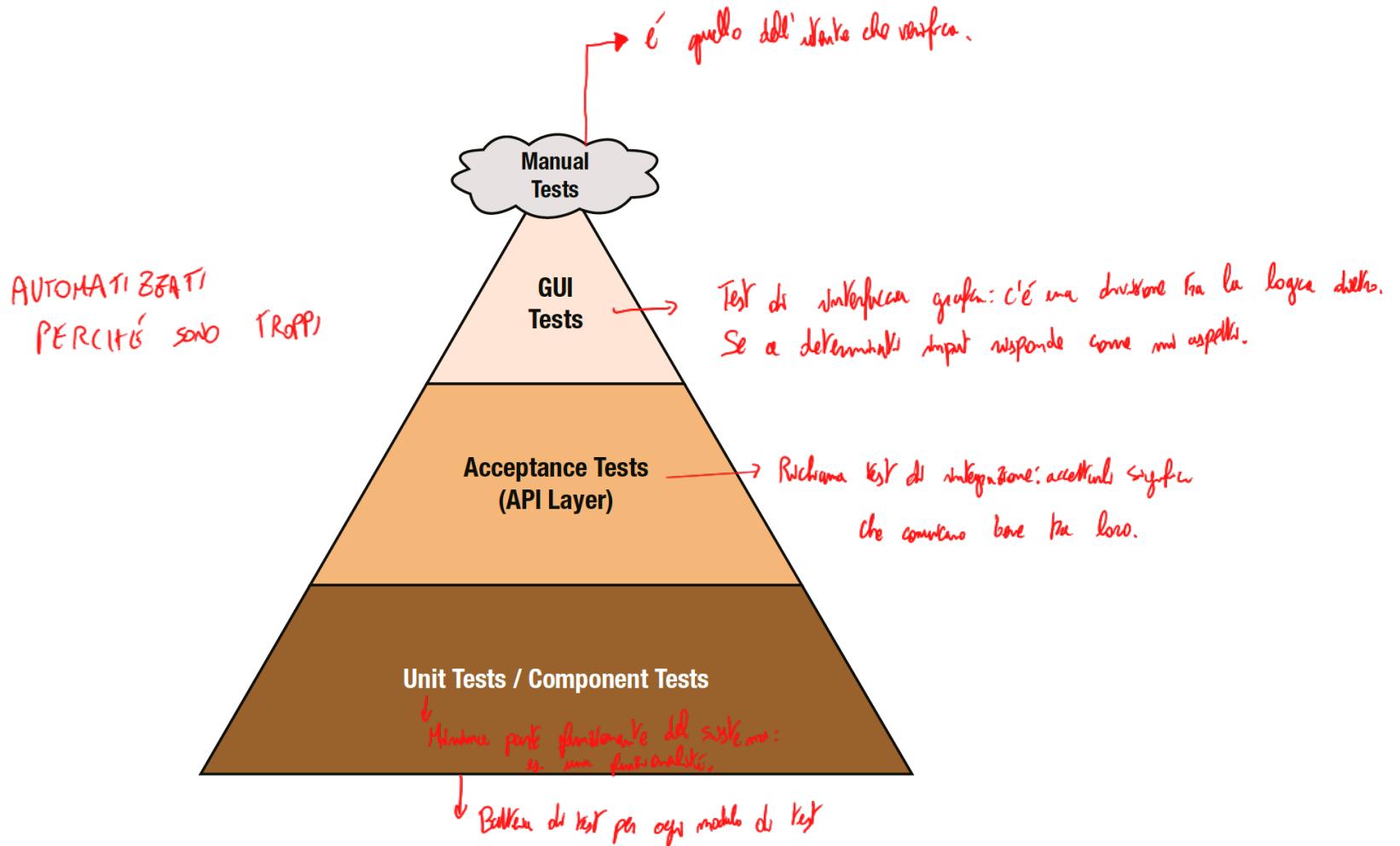
Una possibile definizione di Fatto: *Ese: una combinaz.-di questi elementi*

- i. Codifica della funzionalità richiesta: completata *Codice è pronto*
- ii. Test di unità: scritti ed effettuati
- iii. Test di integrazione: superato
- iv. Test prestazionale: superato
- v. Documentazione (minimale): scritta
- vi. Approvata: dal PO *Product Owner*

NOTA: ha uno sprint e l'altro posso anche definir. che cambia.

[Solo lui quando le user stories non sono pronte ma solo parlando col chiave queste non mi autorizza]
Ready potrebbe essere anche solo questi

Test agile - esempio



Sviluppo test driven e accettazione

Test fai quando il codice non supera.

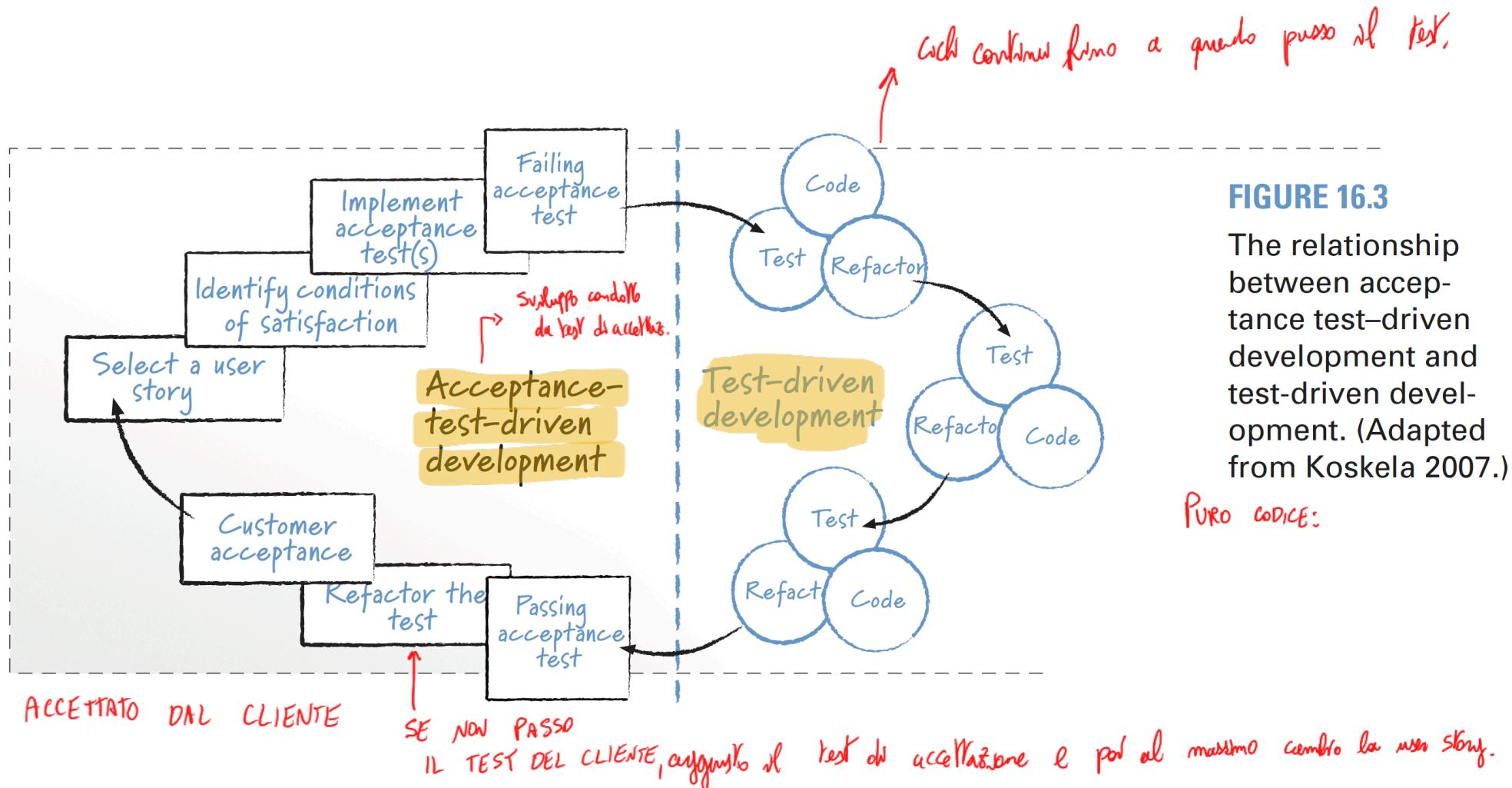
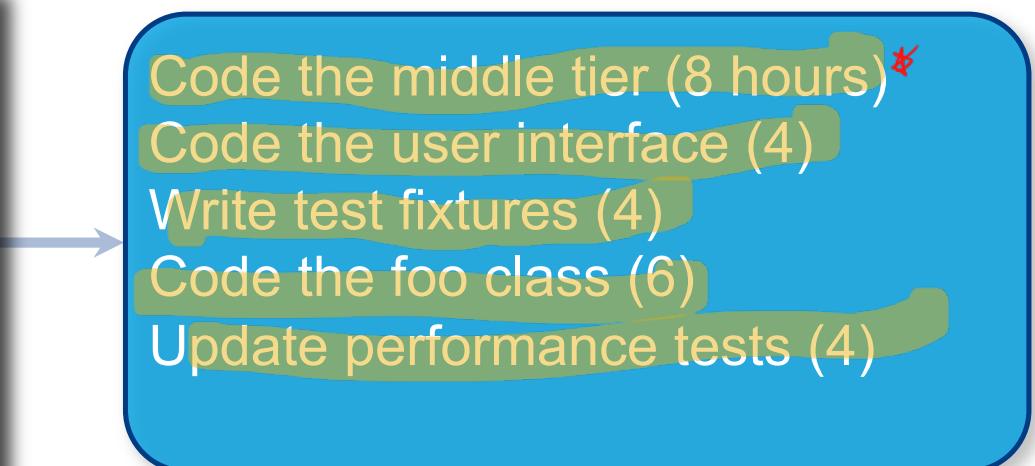
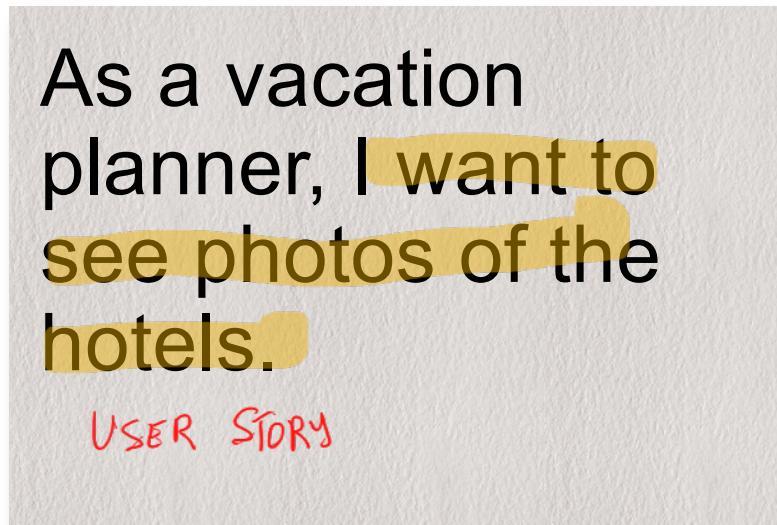


FIGURE 16.3

The relationship between acceptance test-driven development and test-driven development. (Adapted from Koskela 2007.)

Pianificazione dello sprint

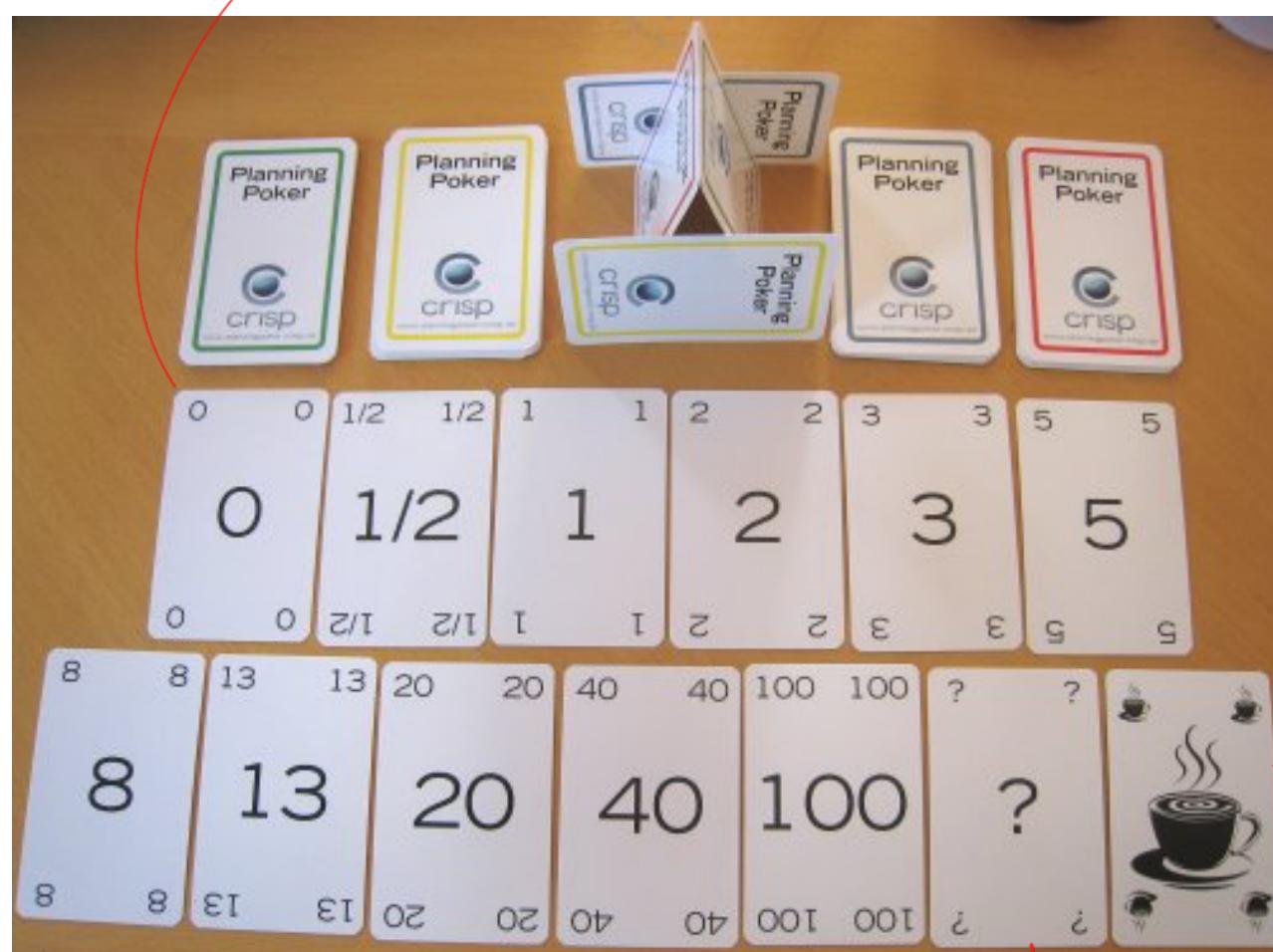
- Il team sceglie dal product backlog gli elementi che verranno certamente realizzati nello sprint
- Studio dell'architettura di alto livello
- Si crea lo sprint backlog: i compiti da fare
 - Identificazione e stima di ciascun compito (1-16 hours)
 - Uso di Planning Poker *Mecanismo per scegliere come lavorare sui task*
↳ Se servono più di 16 ore, non sarà un singolo task



* Parte più complessa

Planning poker

Se una persona vota 2 e una 8, significa
che al voto non è chiuso.



↳ Non ha capito niente

Scrum quotidiano

Stand up meeting

- Modalità
 - quotidiano
 - 15-minuti
 - In piedi
- Niente problem solving *Non si deve provare la soluzione*
 - Chiunque può assistere *(anche chicken)*
 - Possono parlare solo i pig: membri del team, ScrumMaster, Product Owner
 - Scopo: evitare troppi incontri inutili



Serve più copie se ci sono problemi

Tre domande a tutti

1

Che hai fatto ieri?

2

Che farai oggi?

3

Prevedi problemi?

- Non servono allo ScrumMaster per vedere il progresso
- Le risposte sono promesse ai compagni di team

La revisione dello sprint (sprint review)

Fondamentale: riunione informale, mentre slide: non devo far vedere ciò che sono arrivati a fare, faccio la demo e la faccio vedere.

- Il team presenta ciò che ha ottenuto con lo sprint rispetto al prodotto: come questo è “cresciuto”
- E’ una demo delle nuove funzioni o della nuova architettura
- Serve per verificare lo stato del product backlog
- Riunione informale
 - 2 h di preparazione, niente slide
 - Partecipa tutto il team
 - Invitare tutti

→ Invitare tutti a partecipare:
chiare può anche fare domande.
Vediamo come sarà probabile.



→ Per dare confidenza all’altro che al lavoro che doveva fare l’ho fatto.

Sprint retrospective

- Riguarda il processo: come sta andando?
- Occorre analizzare periodicamente cosa va e cosa non va in ciascuno sprint
- Quando: dopo ogni sprint, dopo la review
- Partecipano tutti:
 - ScrumMaster
 - Product Owner
 - Team
 - Forse commitenti, utenti e altri stakeholders

↳ Nessi un forse, ma è bene che ci sia. Questo ho prodotto, questo è ciò che ho affrontato. Come procede il lavoro? Dai 10 punti mi ha fatto 8. Cosa ci ha fatto perdere tempo?

Start / Stop / Continue

Il team discute cosa vorrebbe fare:

Start doing

Ci sono cose che devo iniziare a fare o fermare? O cose che devo continuare?

Stop doing

ES: non ho fatto abbastanza test d'integraz.

Ci sono vari modi alternativi di definire una retrospettiva

Continue doing

Lo scopo della retrospettiva

- Esaminare come è andato l'ultimo sprint riguardo a persone, relazioni, processi e strumenti;
- Rivedere i principali driver architetturali rispetto alle attese degli stakeholder
- Identificare e ordinare gli elementi principali che sono andati bene e le migliorie potenziali;
- Creare un piano per attuare i miglioramenti al modo di lavorare del Team

Cancellare uno sprint

NOTA: Sprint è un'unità atomica.
Requisiti non possono cambiare
Per motivi non dipendenti da me o chiave

- Il Product Owner può cancellare uno sprint durante il suo svolgimento se l'obiettivo dello sprint diventa obsoleto
 - per es. se sono cambiate le condizioni di mercato o se l'organizzazione subisce una modifica
- Di solito non ha senso cancellare uno sprint
 - ↳ Succede qualcosa di estremamente grave

Scrum: gli artefatti

Ruoli

- Product owner
- ScrumMaster
- Team

Rituali

- Sprint planning
- Sprint review
- Sprint retrospective
- Daily scrum meeting

Documenti

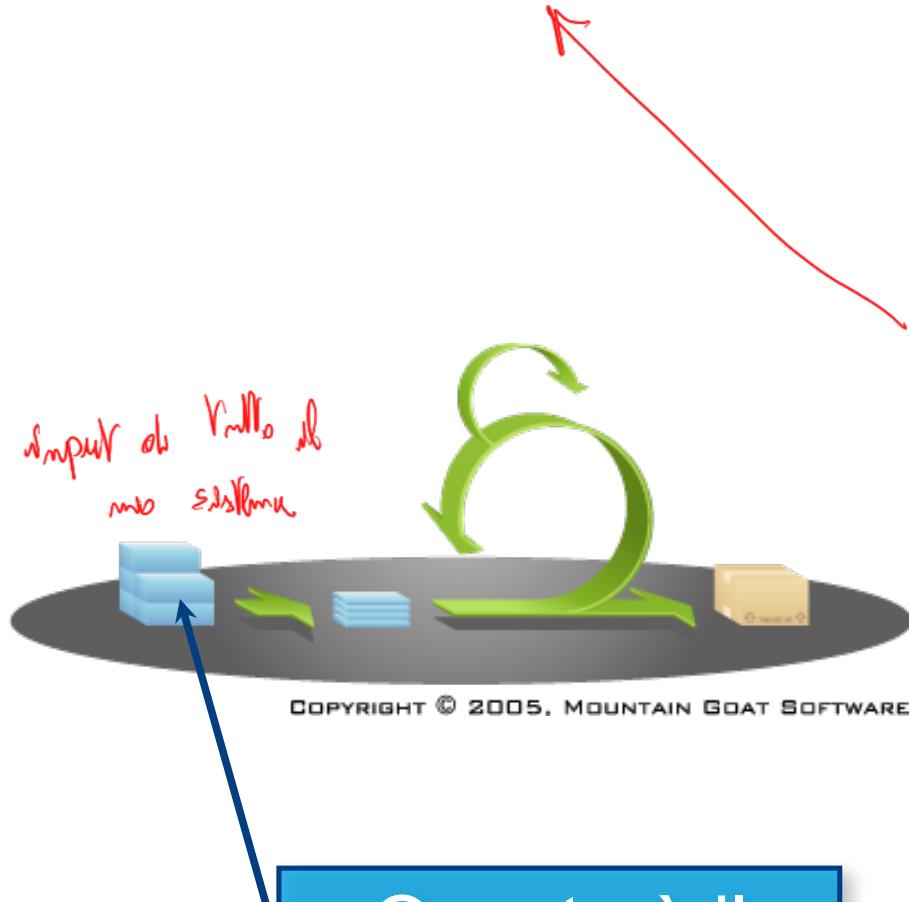
Artefatti

- Product backlog
- Sprint backlog
- Burndown charts

Product backlog

Chiede da sempre un po' di più importanza.

Tutto quantificabile: da termini di tempo, risorse ecc.

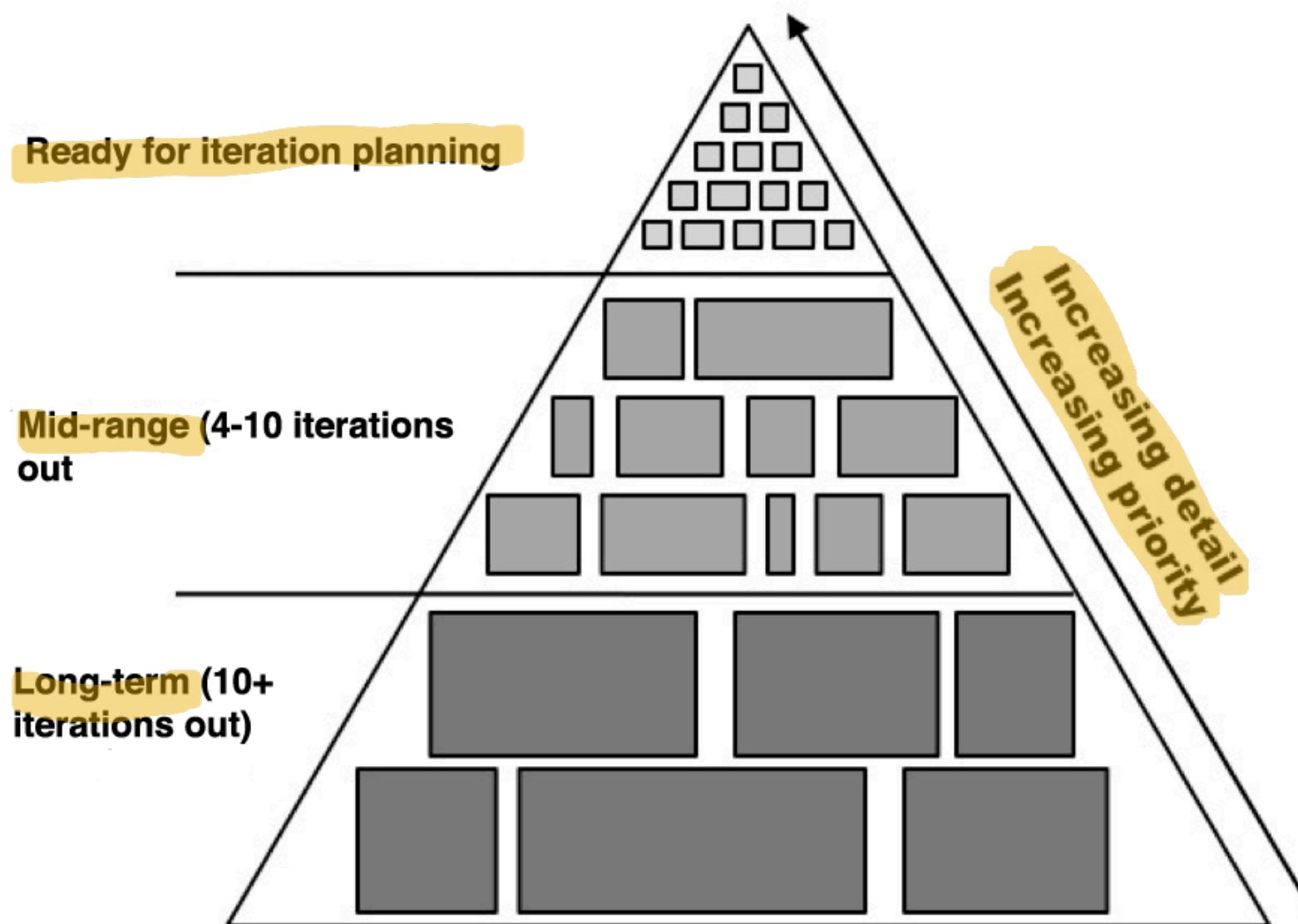


Questo è il
product backlog

- Product backlog = Lista dei requisiti in forma di user stories
- Definita in modo tale che ciascun elemento abbia valore per gli utenti o i committenti del prodotto
- Messa in priorità dal Product Owner
- Priorità ridefinite all'inizio di ogni sprint

La struttura del product backlog

Break down stories as they move up the backlog

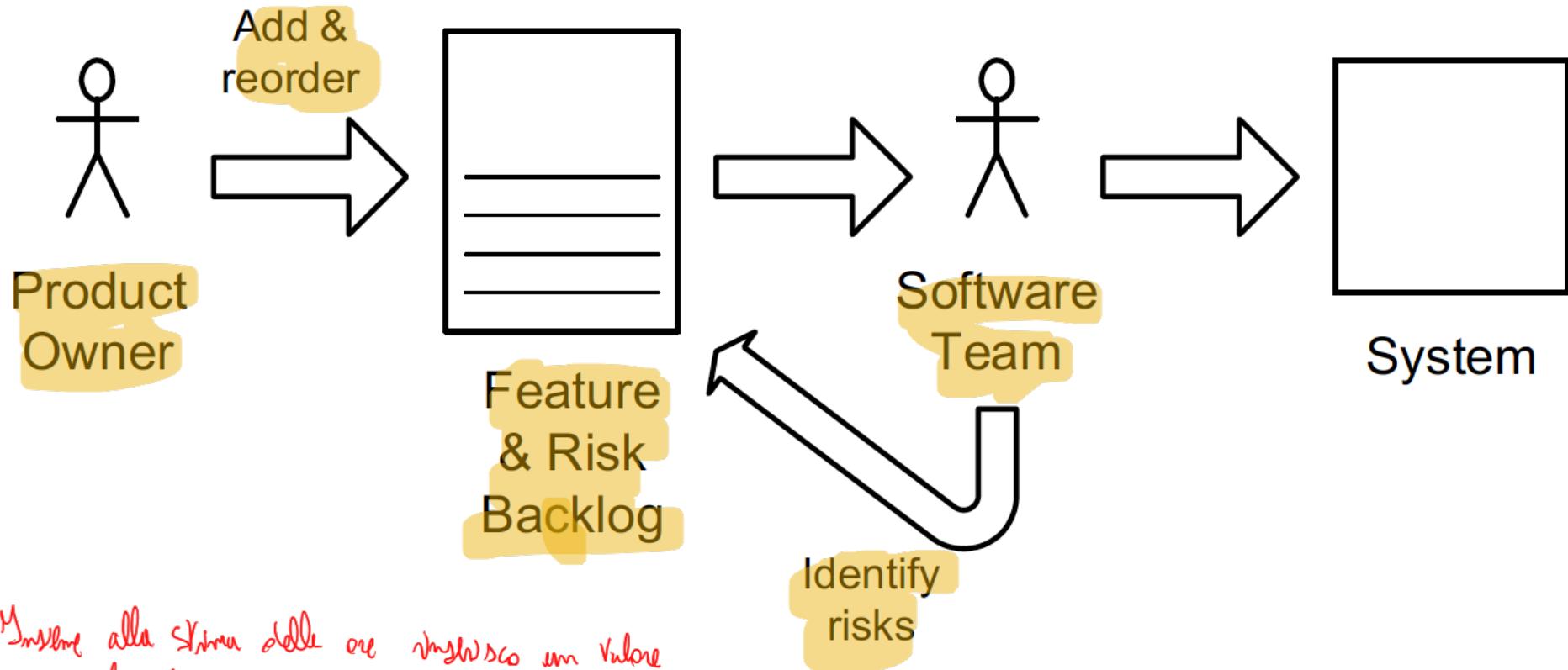


Esempio di product backlog

Elemento backlog	Stima (h)
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50

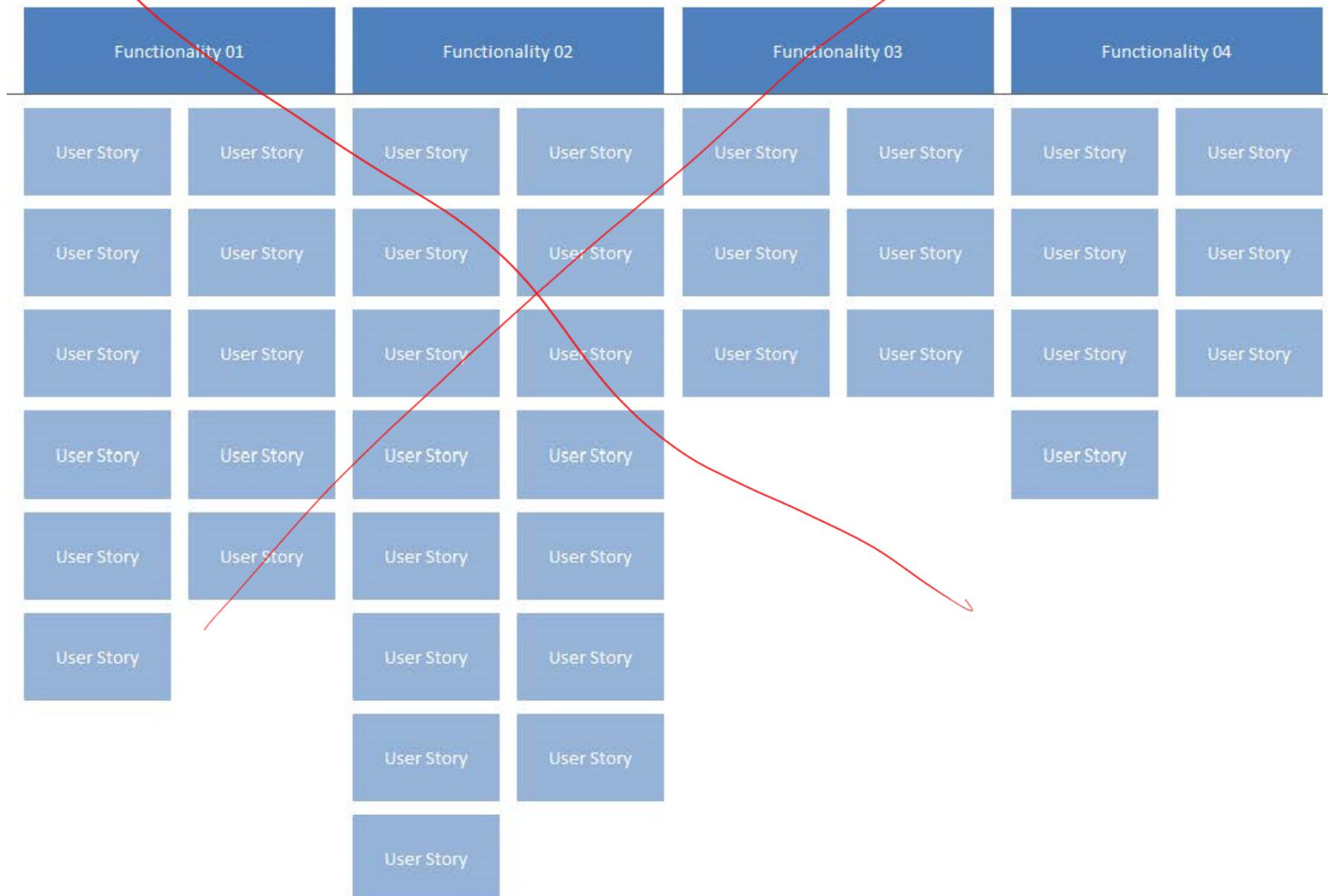
Trend: durata stimante mai verso l'alto.

Variante del product backlog con analisi dei rischi

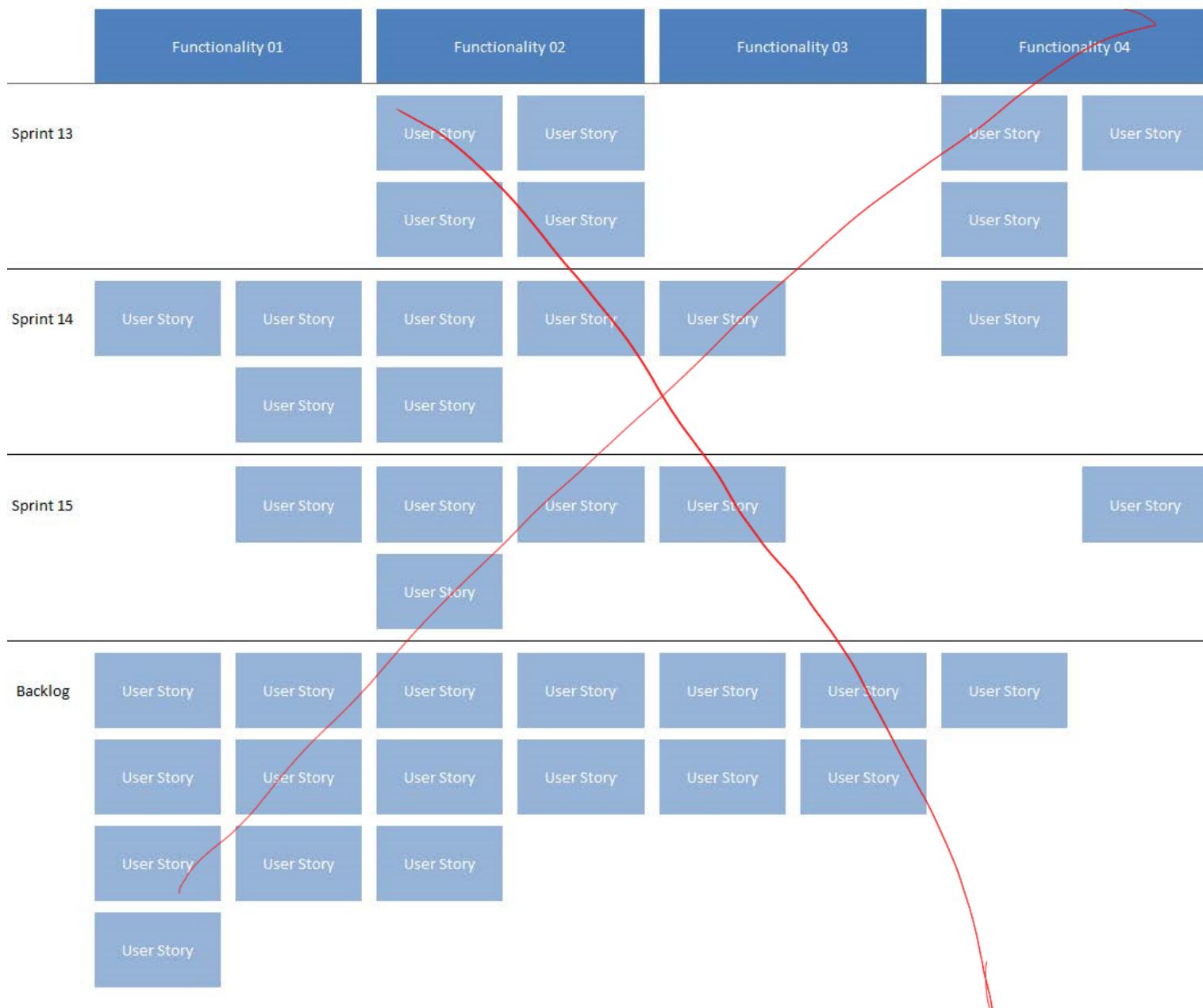


Insieme alla Sistema delle ore risparmio un valore di rischio. Il product owner lavora sulle basi dei wish

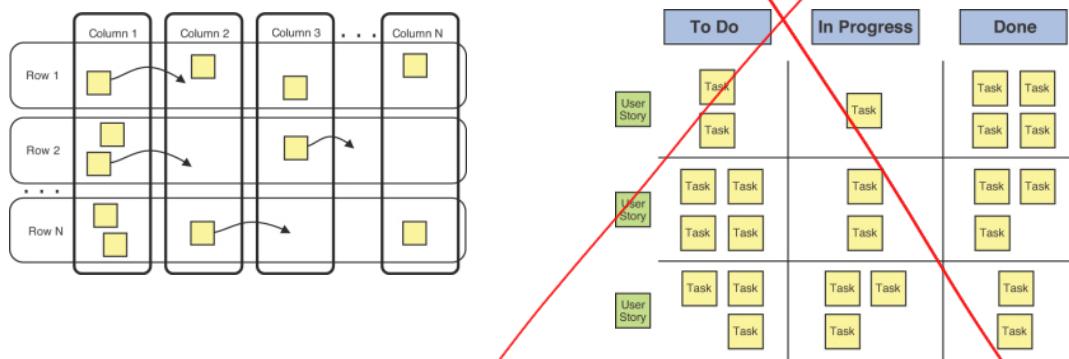
Mappare le user story sulle feature



Mettere in priorità le user story



Come si usa il board



Obiettivo dello sprint (sprint goal)

Qual è l'obiettivo di questa sprint?

Breve descrizione del lavoro da fare durante lo sprint. Esempi:

Database Application

Make the application run on SQL Server in addition to Oracle.

Life Sciences

Support features necessary for population genetics studies.

Financial services

Support more technical indicators than company ABC with real-time, streaming data.

Gestione dello sprint backlog

- I membri del team prenotano lavoro da fare su scelta personale
- Il lavoro non viene assegnato, ma richiesto su base “volontaria”
- La stima del lavoro da fare in termini di effort viene aggiornata quotidianamente

Gestione dello sprint backlog

- Ogni membro del team può modificare lo sprint backlog, che di solito viene conservato in un tabellone detto “taskboard” o “kanban”
- Il lavoro da fare in ogni sprint “emerge” sul tabellone
- Se il lavoro da fare è poco chiaro, conviene definire uno sprint backlog item con una stima maggiore e decomporlo più tardi
- Occorre aggiornare il lavoro da fare man mano che viene fuori

Taskboard

Agilar Taskboard

http://taskboard.agilar.org/taskboard/team/6

Smart Bookmarks ▾ Google Translate Gmail – Inbox Google Calendar Google Reader (100+) Wikipedia Plesk 8.1.0 BIT-HDTV – Home o... site.dev.agilar.org Taskboard DEV iLean DEV

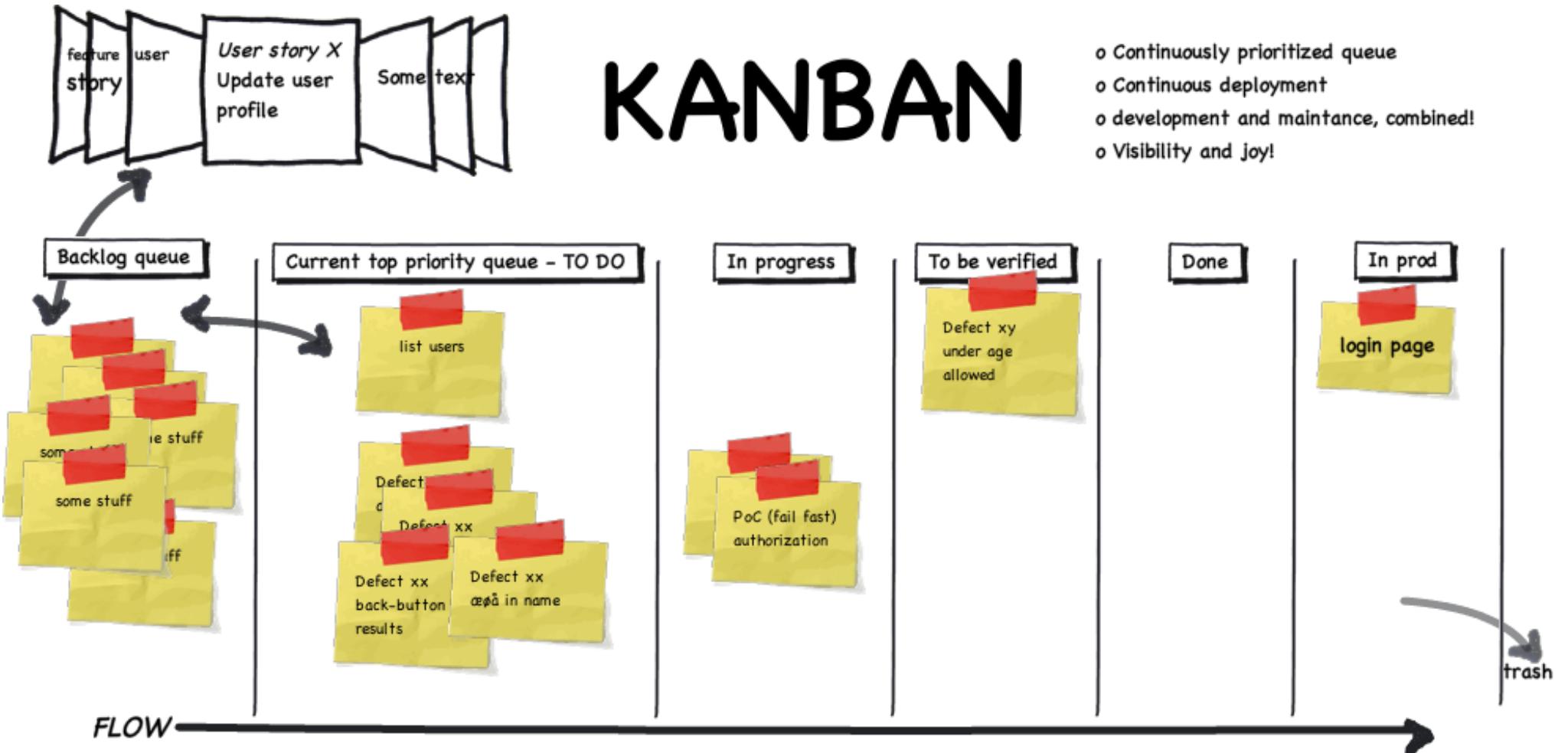
W Wikipedia, the free encyclop... Scrum Alliance – Certified Sc... Agilar Taskboard Dashboard [Hudson] - RFQ detail Gmail – Inbox xquesada@g... +

Product Backlog Add Task PLEASE ANALYSE PLEASE TEST DONE BLOCKED HIGH PRIORITY WAITING DELEGATED BUG FRANCISCO XAVIER MARCOS MARCO EXIT

Agilar Dev Team

	NOT STARTED	IN PROGRESS	FINISHED	
AA018 Legacy Issues And Unplanned Items Priority: 3000 Size: 0	<p>Fix Bubbles</p> <p>Deploy to staging should backup the production database and test a production deploy</p>	<p>PLEASE TEST</p> <p>HUD make deploys know about the status of the last good build</p> <p>FRANCISCO</p>	<p>BUG</p> <p>Sponsorship listings.</p> <p>MARCOS N</p>	<p>When clicking everything goes to the homepage</p>
AP005 Training section and module - dirt road Priority: 2885 Size: 5	<p>Add tests for active resource (see ActiveResource::HttpMock)</p> <p>Add tests to agora training controller (upcoming and past)</p> <p>Add AJAX calendar to start/end date</p> <p>"Your trainings" should only co-trainer but not main trainer</p> <p>You can stop public your profile on website even if you have published trainings</p> <p>Possible to put same BUG and co-trainer</p> <p>The owner of a training should be able to edit and delete it</p>	<p>DONE</p> <p>Add validations on start / end date</p> <p>MARCOS N</p> <p>ONLY users who put their profile on the Agilar website should be eligible for trainings</p> <p>BLOQUE de training - training (website) - link apunta a Agilar</p> <p>Support for external registration mechanism (URL field)</p> <p>FRANCISCO</p> <p>PLEASE TEST</p> <p>Cannot Edit existing trainings</p> <p>MARCOS N</p>	<p>PLEASE TEST</p> <p>Caminar URL de produccion: #course_name</p> <p>MARCOS N</p> <p>PLEASE TEST</p> <p>Mail to trainer and organizer when new registration occurs</p> <p>MARCOS N</p>	<p>Change Trainings heading to "Public Trainings"</p> <p>Add tobias training</p>

Done

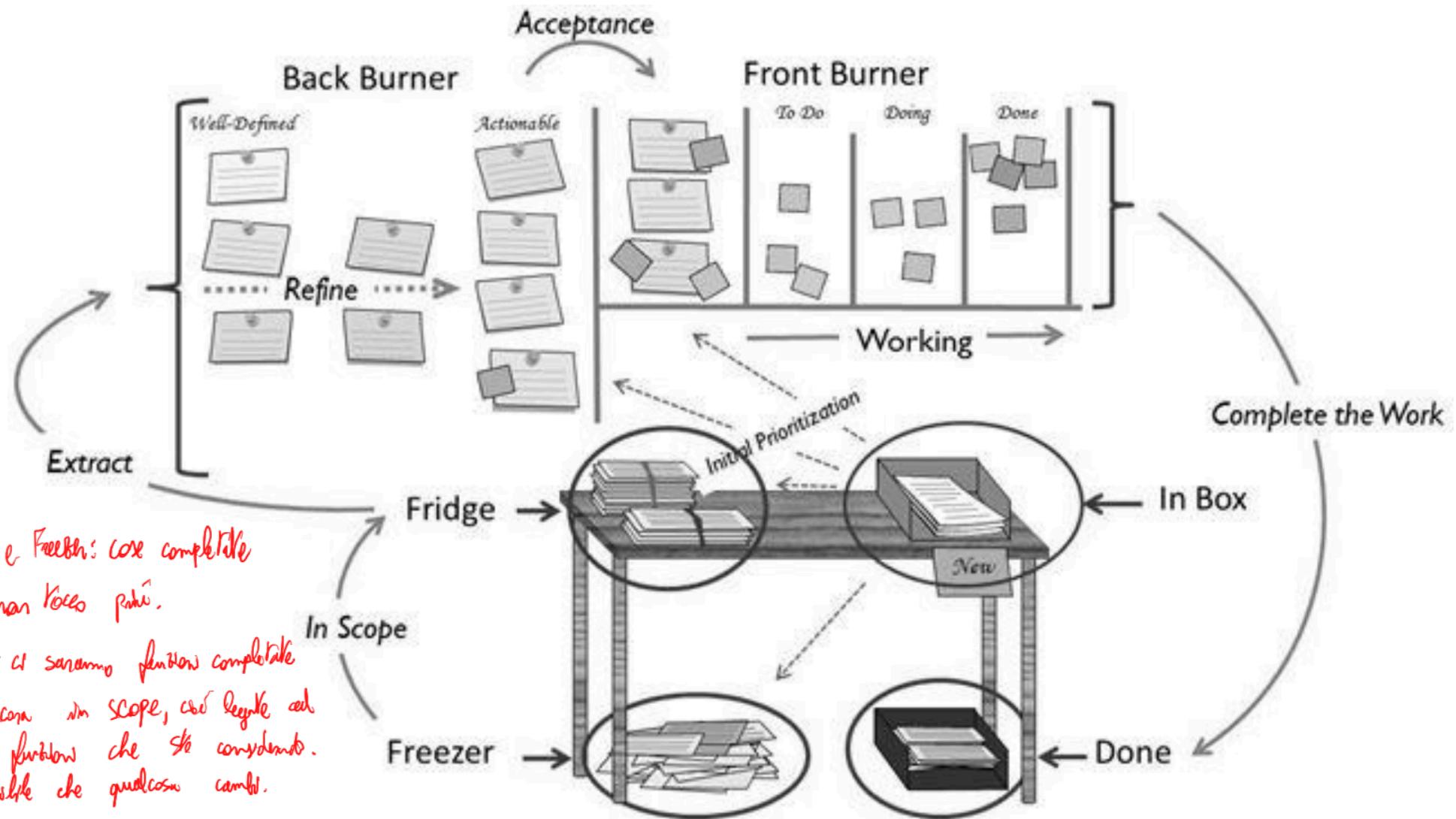


Created by Ole Morten Amundsen using Mockito. Use as you please.

- o Continuously prioritized queue
- o Continuous deployment
- o development and maintenance, combined!
- o Visibility and joy!

Il ciclo dei compiti in uno sprint

Burner: product backlog e lavoro da fare
nello sprint backlog: Plan Phase le incunette



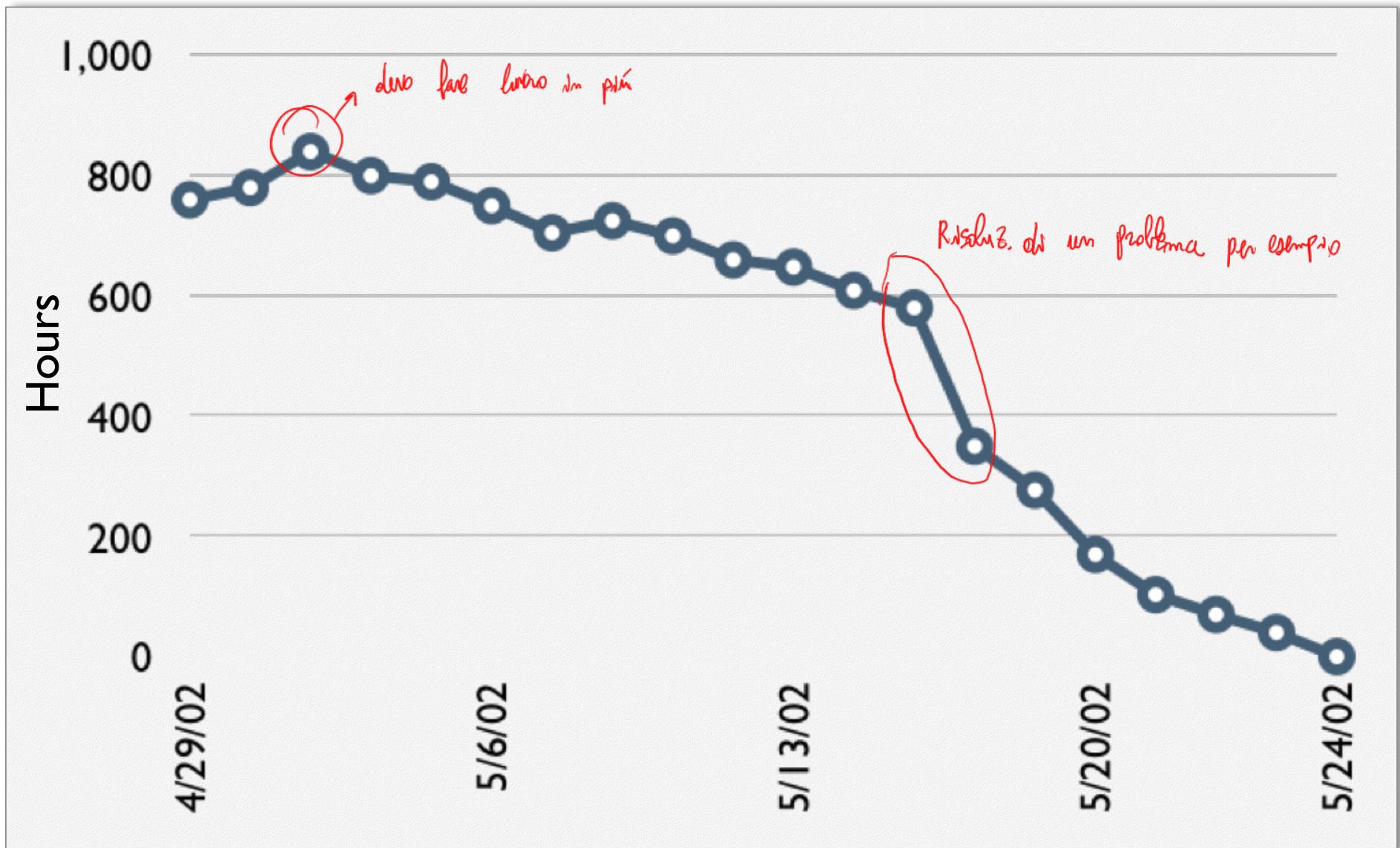
Uno sprint backlog

Vai nel dettaglio sulle ore di lavoro disponibili per giorno

Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	4	
Test the middle tier	8	16	16	11	8
Write online help	12				
Write the foo class	8	8	8	8	8
Add error logging			8	4	

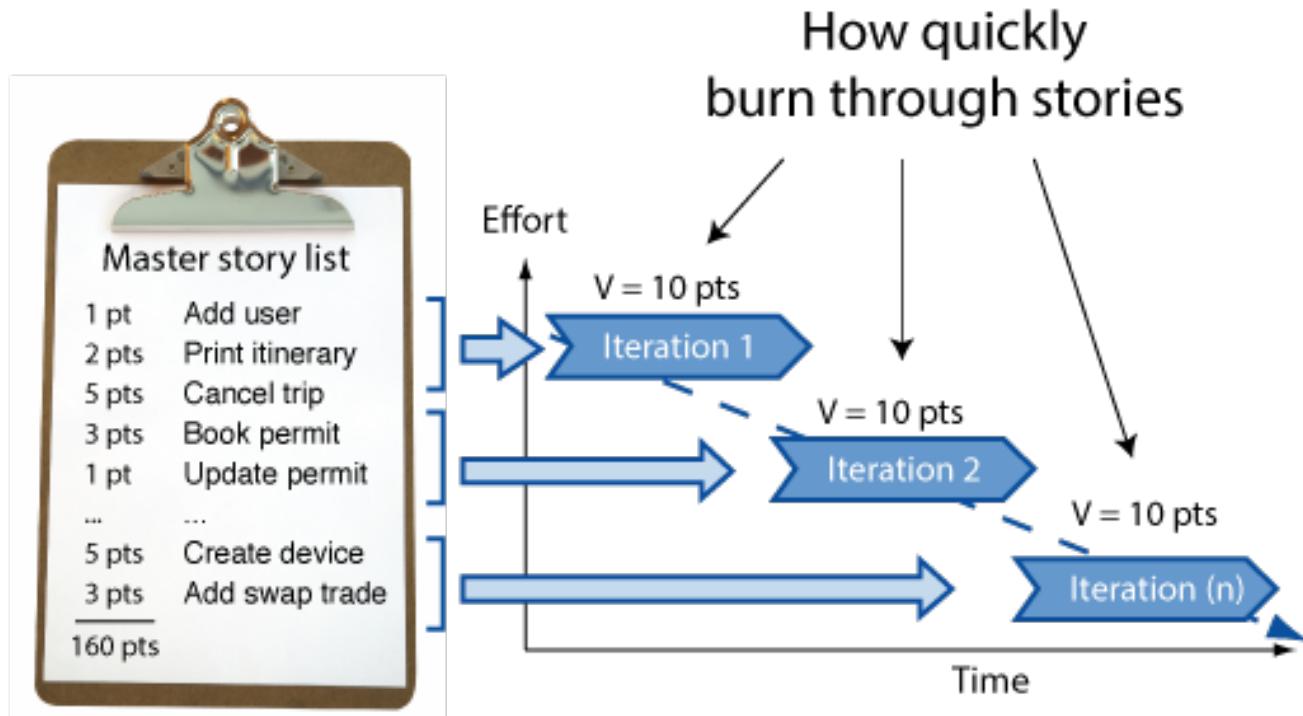
Esempio: burndown perché brucio le attività da fare. Esempio di una burndown squilibrata. Numero ore che lavoro da fare in un sprint

Un burndown chart di uno sprint



Burndown charts

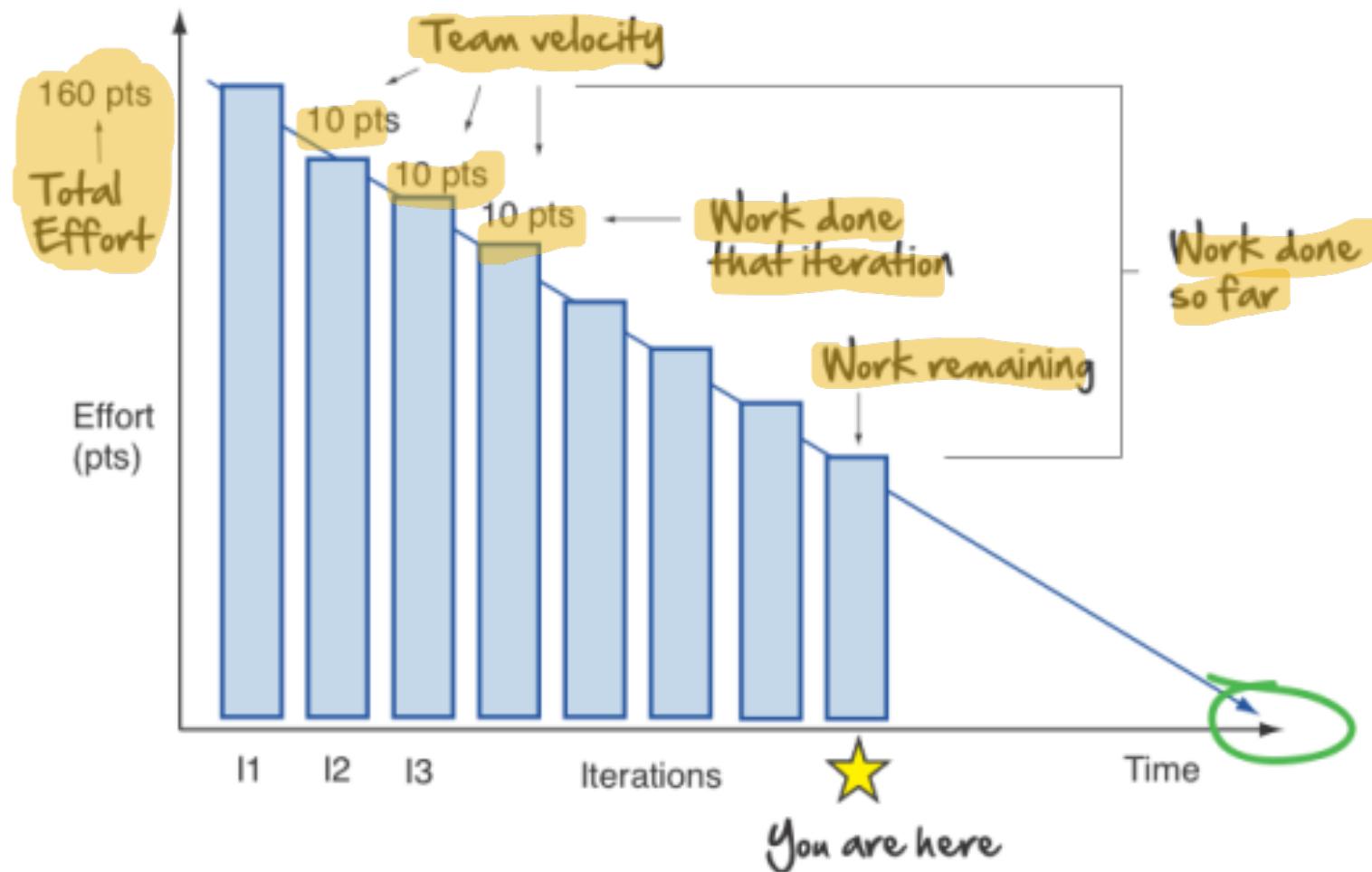
Motivazione degli
effort che devono diminuire



Passo per uno di quei 10 punti.

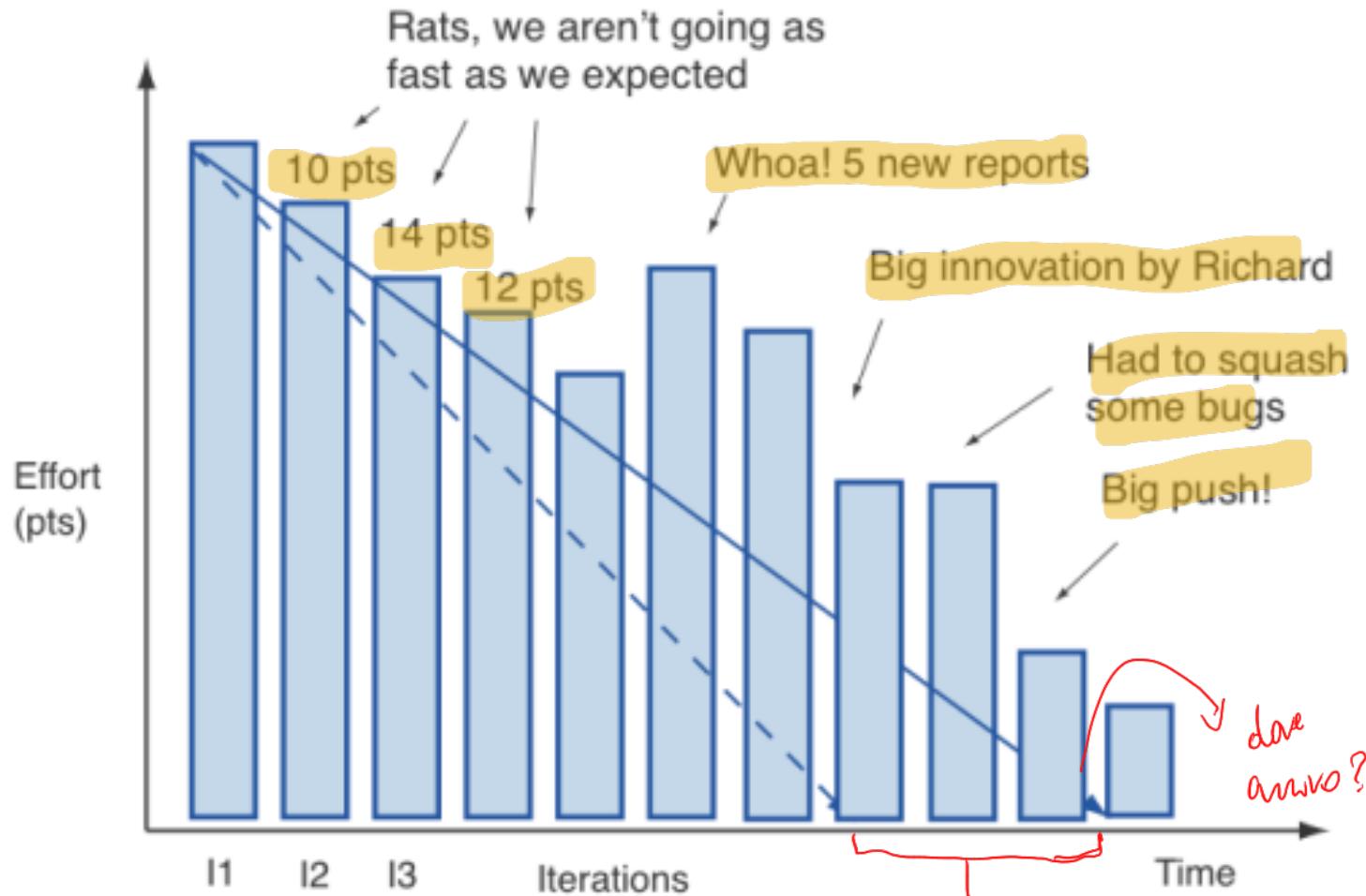
Ideamente, debba essere una linea retta.

Velocity ideale sul burndown chart

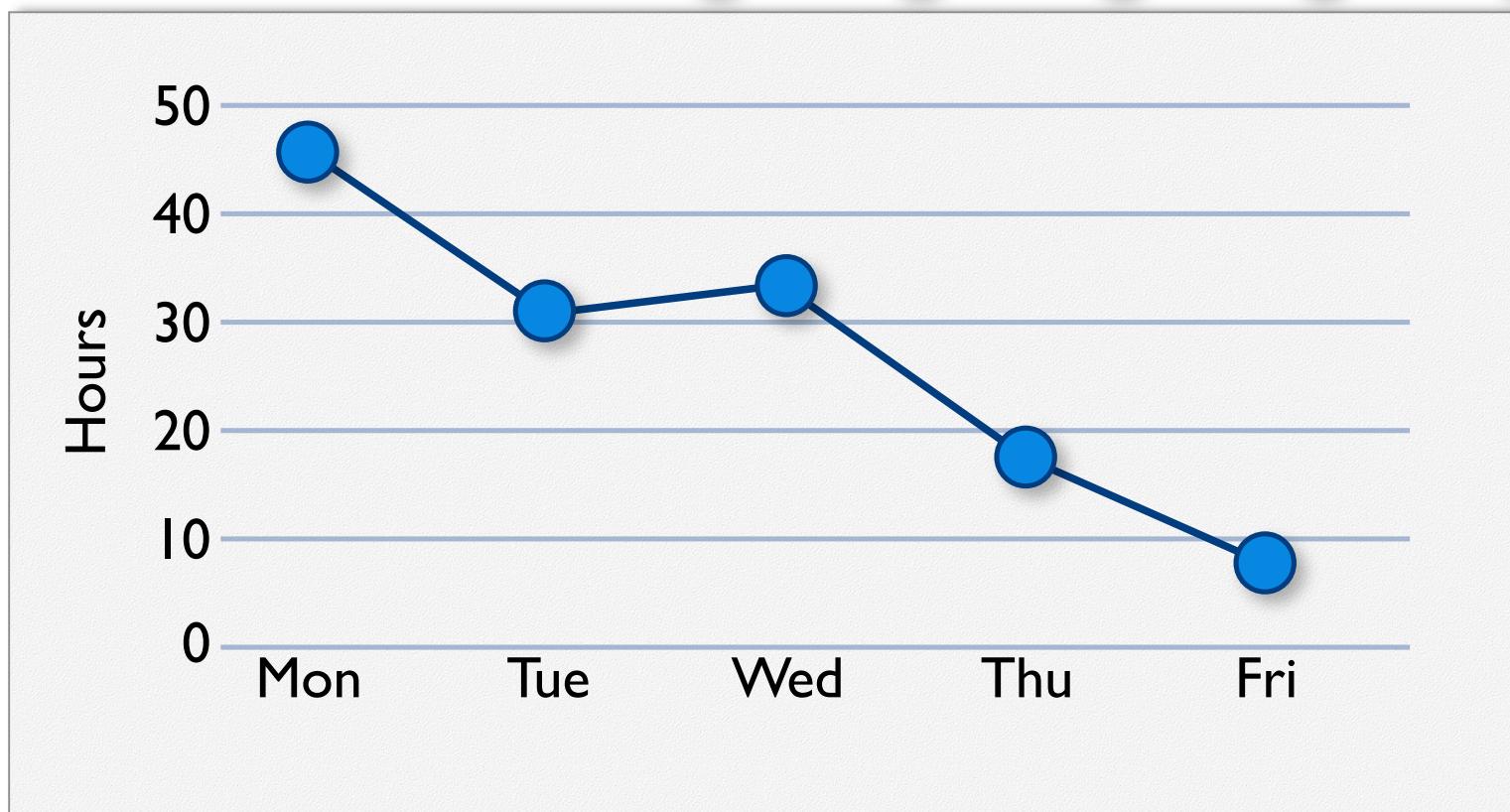


Příručka

Velocity reale sul burndown chart

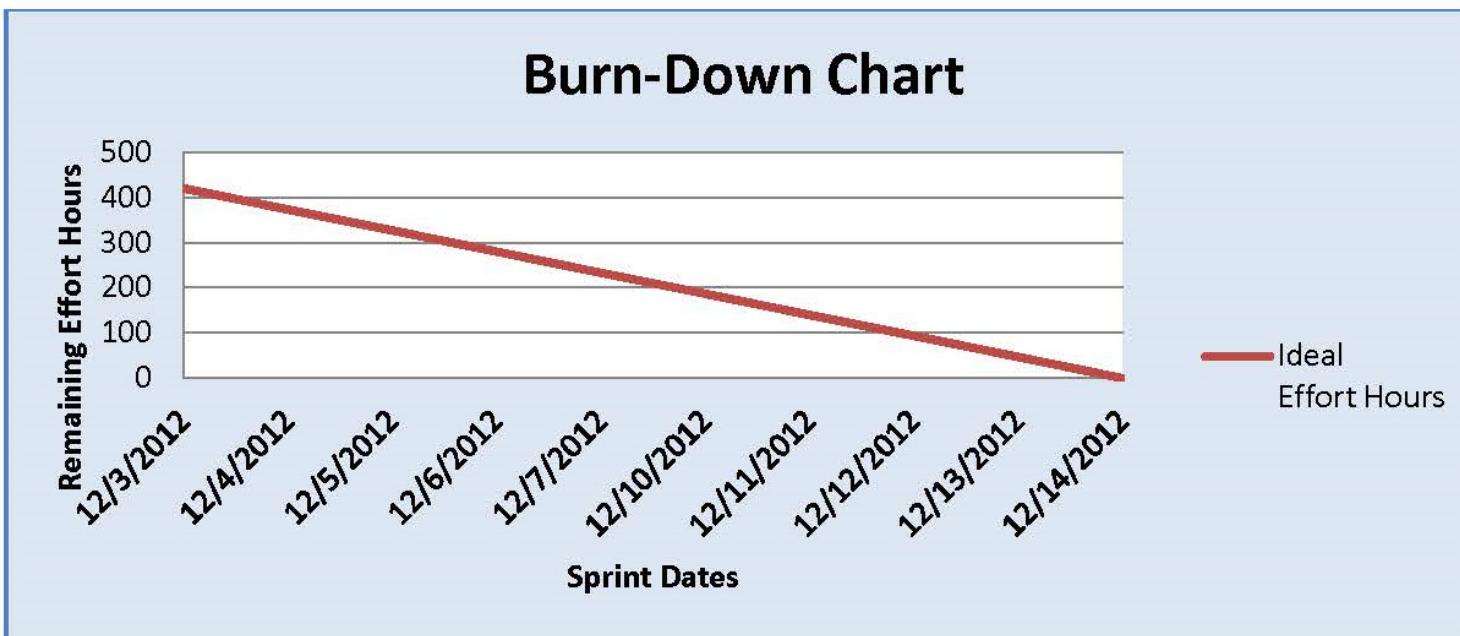


Tasks	Mon	Tues	Wed	Thur	Fri
Code the user interface	8	4	8		
Code the middle tier	16	12	10	7	
Test the middle tier	8	16	16	11	8
Write online help	12				



Esempio

Durata **sprint: 2 settimane, 7 persone, 6h/giorno. Totale sforzo ideale 420 ore**



Ovviamente non finirà così ogni giorno appena le skime

Stima continua dello sforzo rimanente

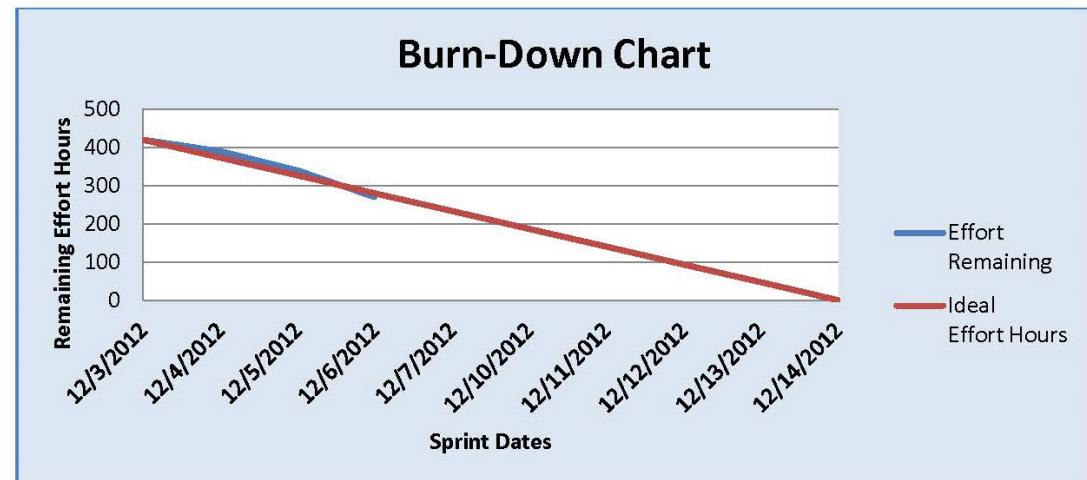
Venne continuamente aggiornato l'effort reale. Aggiornati quotidianamente dalla scrum master

Ogni partecipante sceglie un compito e stima il tempo rimanente.

Per es. dopo il primo giorno sono state spese 6 h sul compito 1. Il programmatore stima che rimangano altre 6 ore (quindi 2h oltre la stima di 10h).

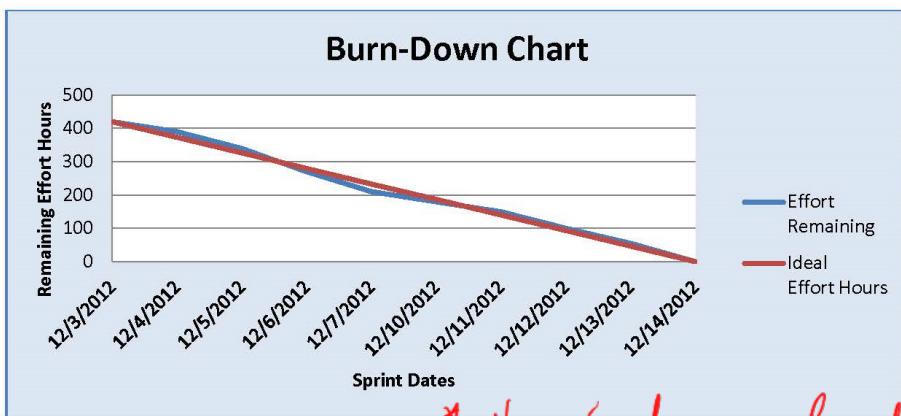
Il burn-down chart viene aggiornato e confrontato con l'ideale (blu vs rosso)

Story Name	Task No	Task Description	Status	Owner	Estimated Effort (in Hours)	Effort Remaining (in Hours)
Story 1	1	POC for Story 1	In Progress	Developer 1	10	6
	2	Requirement Clarification with PO	Closed	BA	8	0
	3	Develop modules	Open	Developer2	12	12



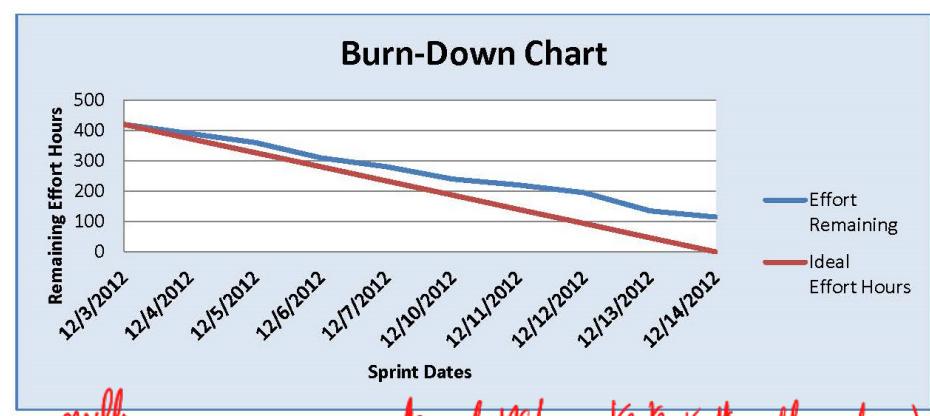
Varie situazioni

Poco svolgimento

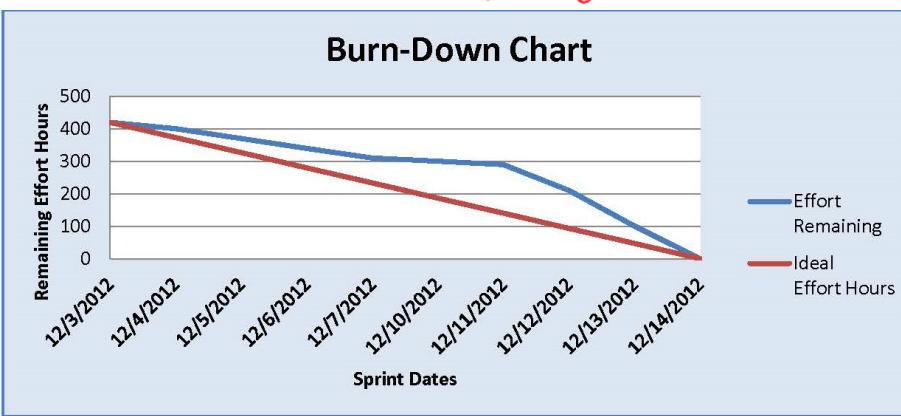


* Non c'è che man ho fatto

Peggiori di tutti



molti, ma non risultati fattibili. Resta tutto alla fine)

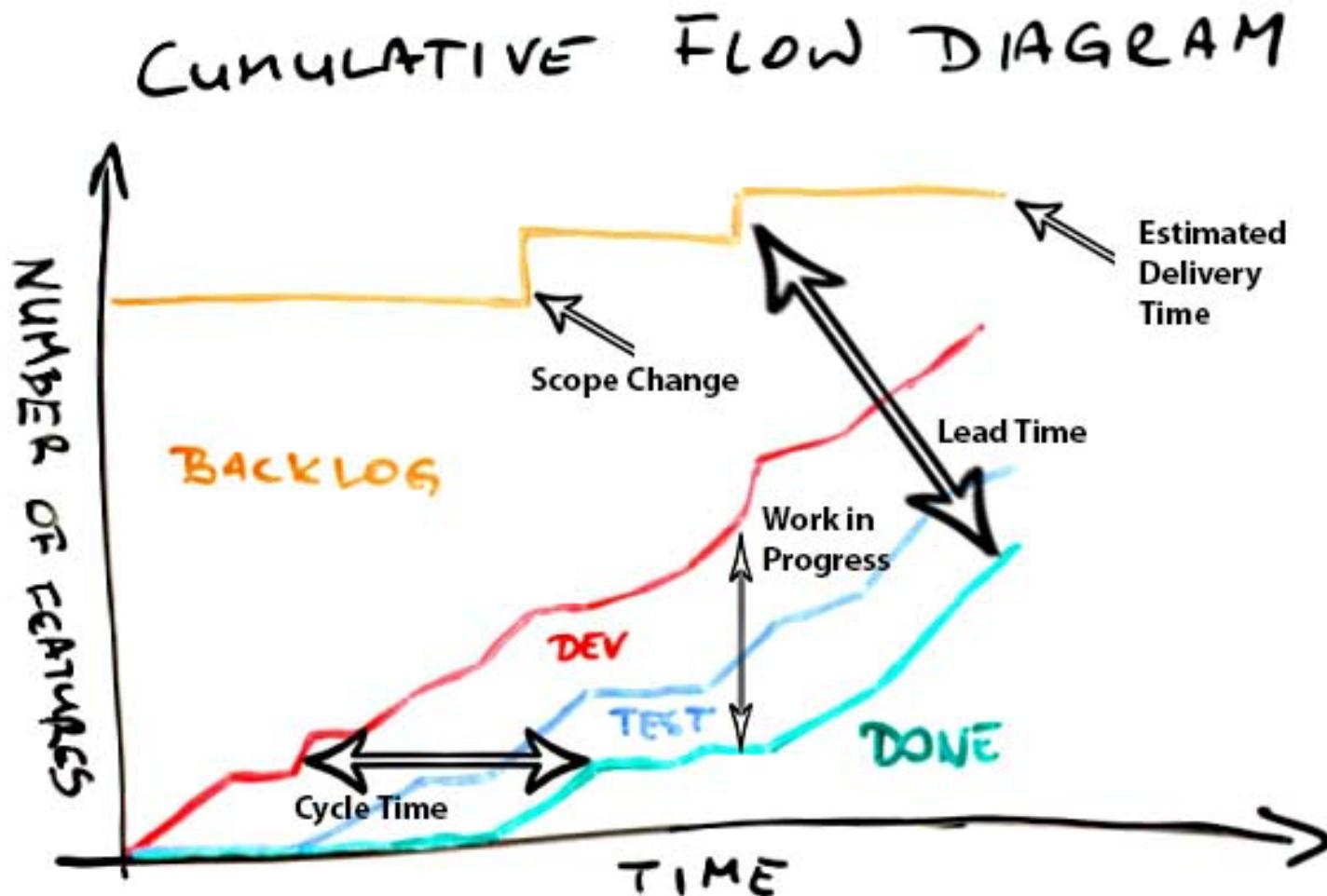


O c'era un problema noto

↓
and best

Non dipende da errori nel task, ma da modo di lavorare comunque pessimo - &

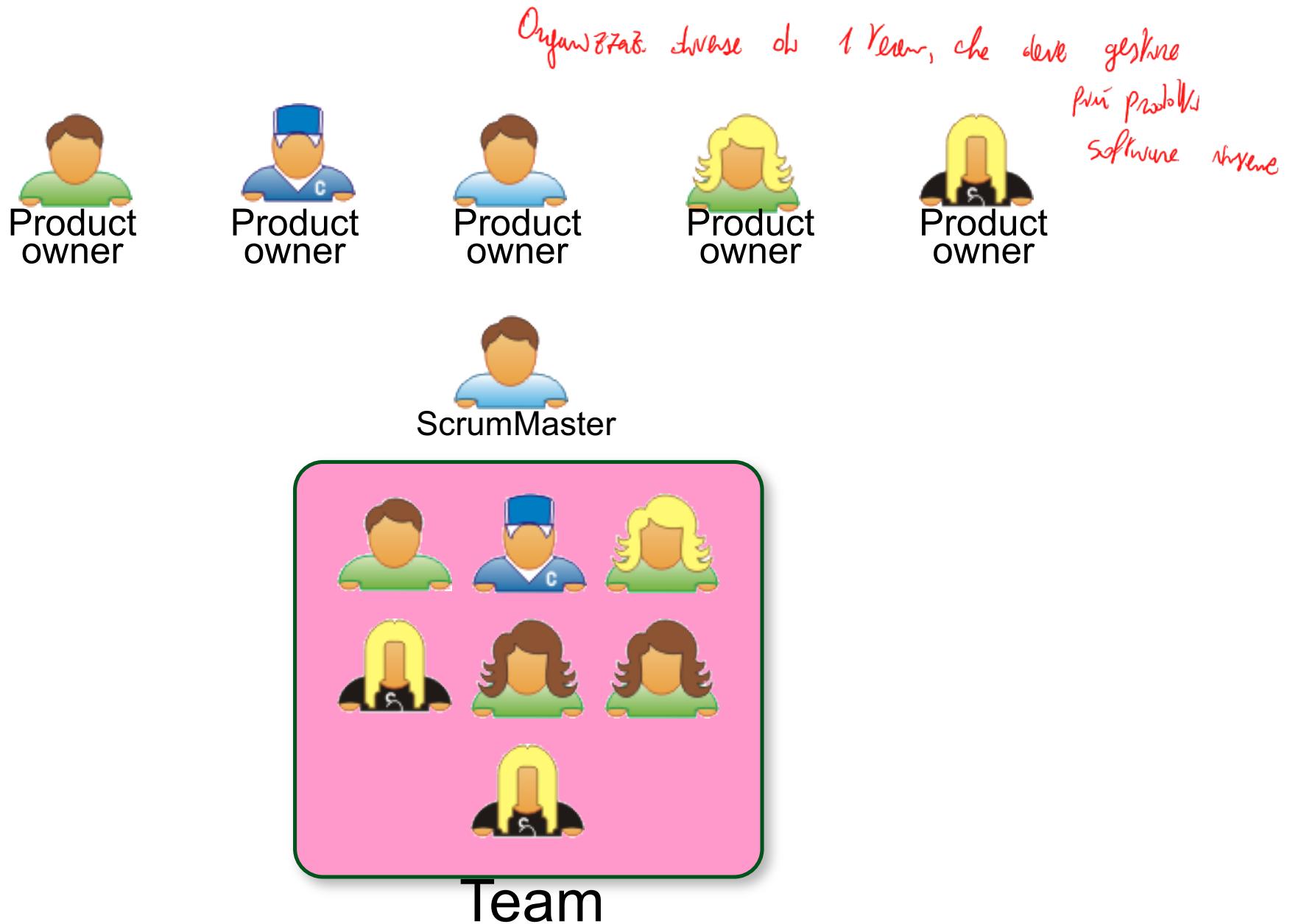
Cumulative flow diagram



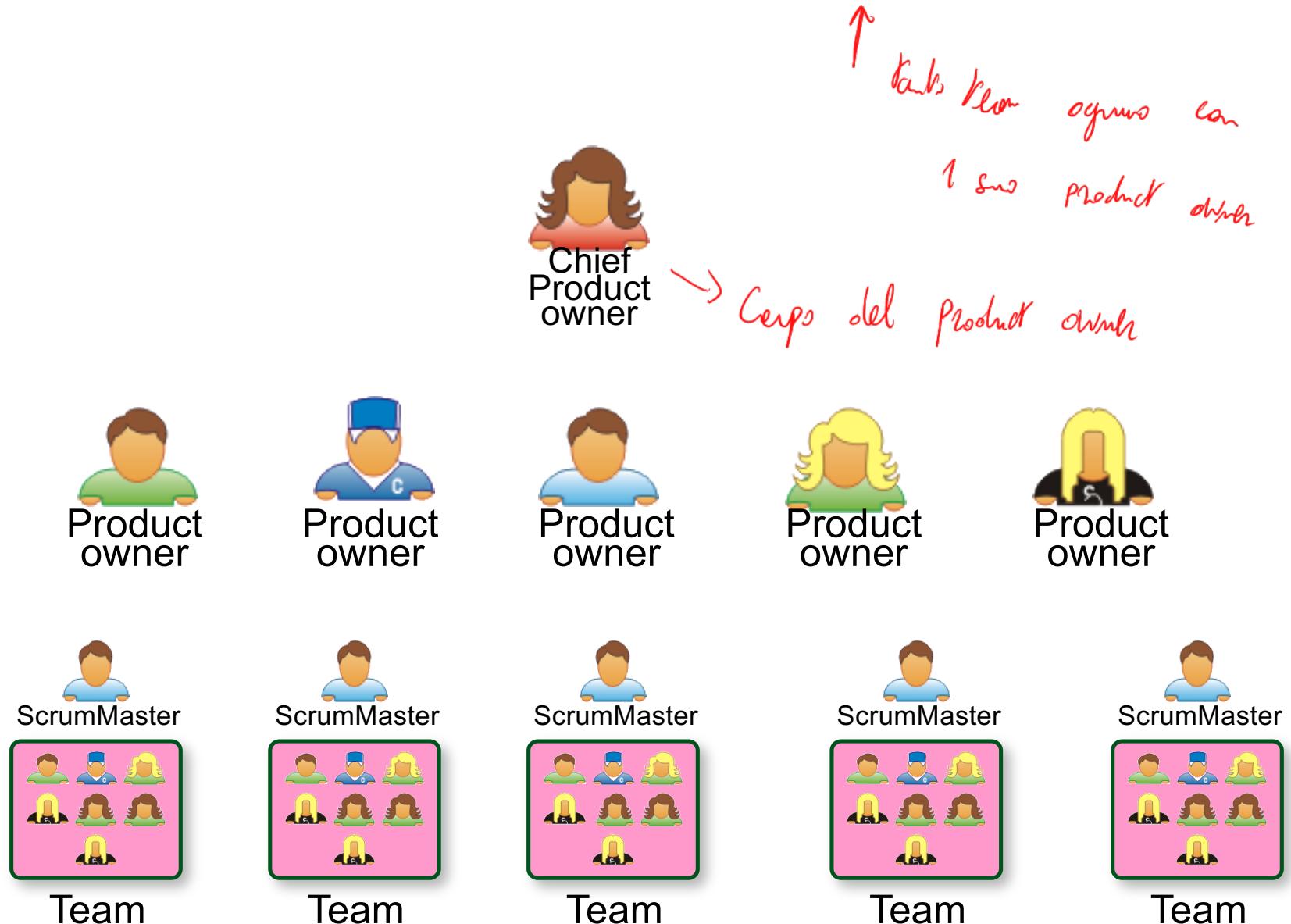
Scalabilità

- Team: 7 ± 2 persone
 - Si scala su grossi sistemi con scrum di scrum
- fattori
 - Tipo dell'applicazione
 - Dimensione dei team
 - Dispersione dei team
 - Durata del progetto
- Alcuni progetti Scrum hanno coinvolto oltre 500 persone

Team unico, molteplici product owner

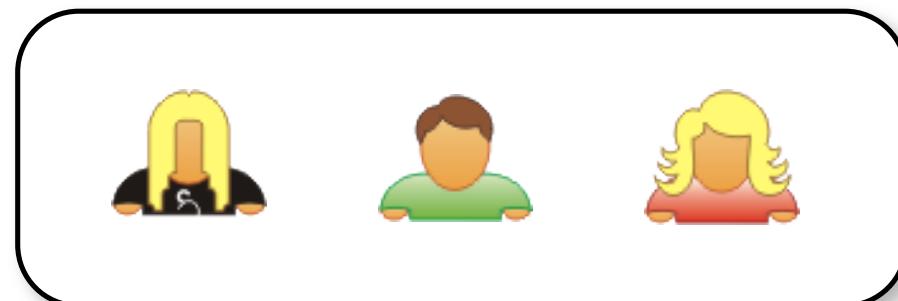


Molteplici team, prodotto unico

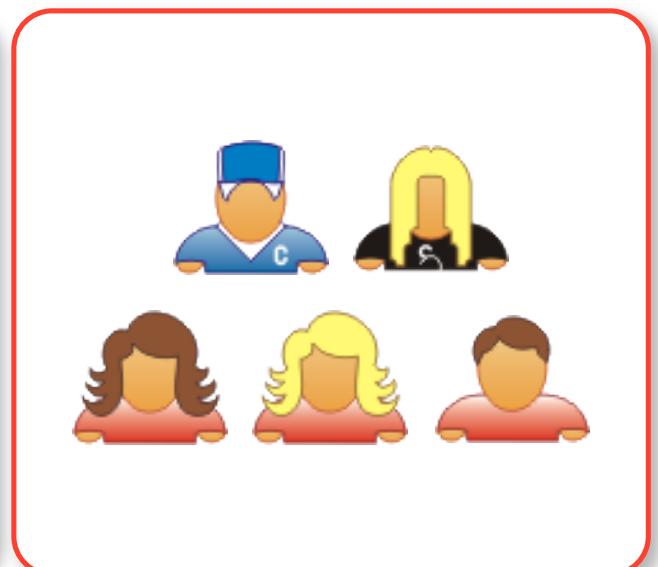
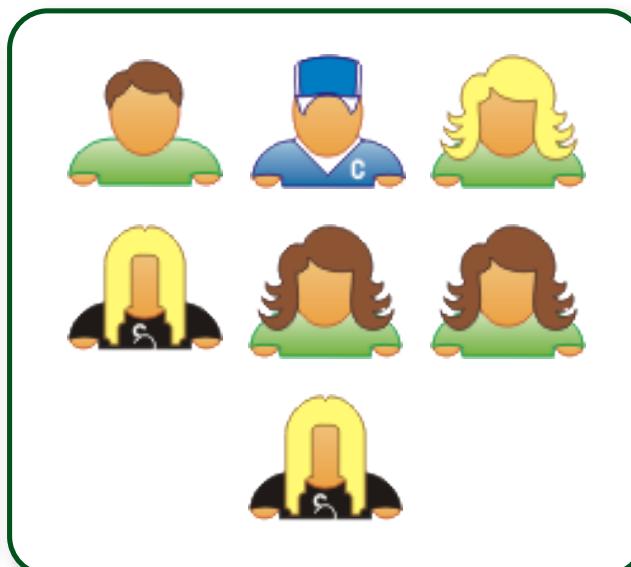
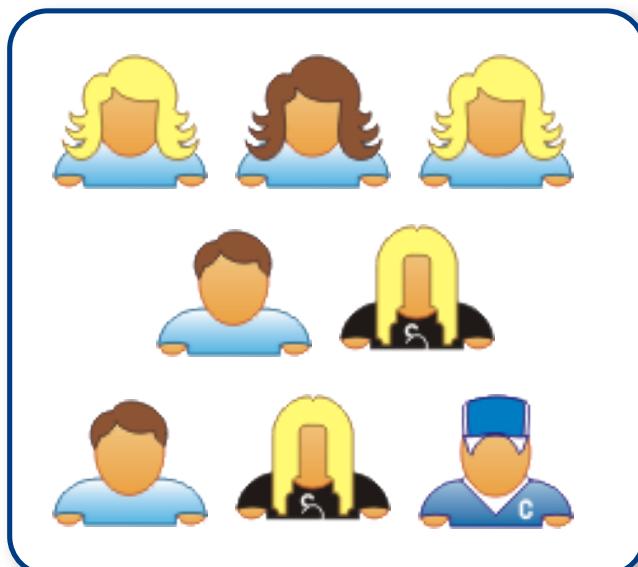


Scrum di scrum

10 team Scrum organizzati in un grande Team scrum con degli ambasciatori
Now: con scrum posso anche gestire
altre cose



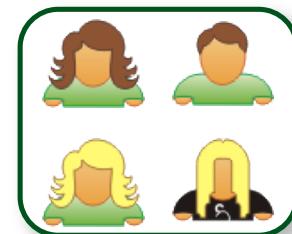
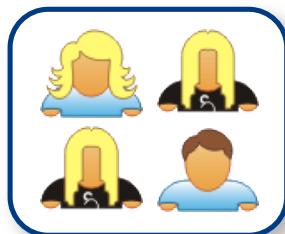
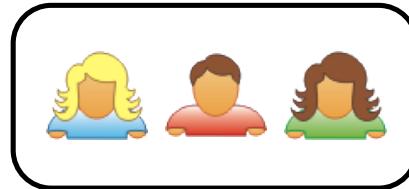
Metascrum
degli ambasciatori



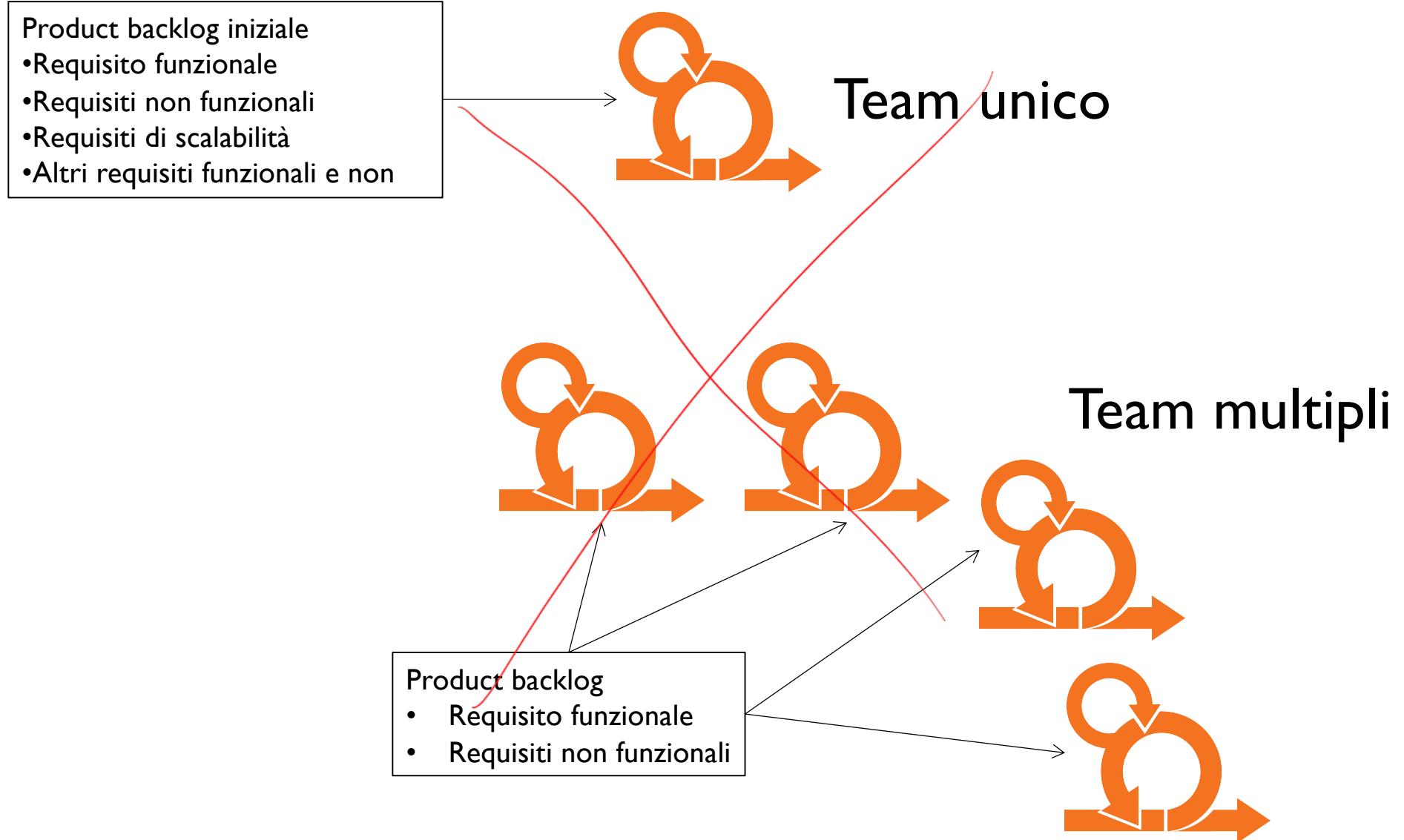
J. Sutherland sullo scrum di scrum

- *Since I originally defined the Scrum of Scrum (Ken Schwaber was at IDX working with me), I can definitively say the Scrum of Scrums is not a "meta Scrum."*
- *The Scrum of Scrums as I have used it is responsible for delivering the working software of all teams to the definition of Done at the end of the Sprint, or for releases during the sprint.*
 - *PatientKeeper delivered to production four times per Sprint. Ancestry.com delivers to production 220 times per two week sprint. Hubspot delivers live software 100-300 times a day.*
- *The Scrum of Scrums Master is held accountable for making this work*
- *So the Scrum of Scrums is a operational delivery mechanism.*

Scrum di scrum di scrum



Coordinare team multipli



Problemi tipici con Scrum

1. Ignoranza dei valori agili e di Scrum
2. Prodotto software non testato alla fine dello sprint
(cattiva definizione di “*Fatto*”)
3. Backlog non pronto all'inizio dello sprint (cattiva definizione di *Ready*)
4. Mancanza di facilitazione (o cattiva facilitazione)
5. Mancanza di supporto da parte dei manager
6. Mancanza di supporto da parte degli stakeholder
7. Gestione caotica degli scrum di scrums

Scrum: patti tra stakeholder e team

Il Team promette agli Stakeholder:

- Il PO nel Team difenderà gli interessi degli Stakeholder
- Il tempo degli Stakeholder sarà usato saggiamente, verranno poste solo questioni utili allo sviluppo
- Il lavoro sarà della massima qualità, compatibilmente coi vincoli posti dall'azienda
- Alla fine di ogni sprint verranno consegnate nuove funzionalità che potranno essere validate dagli Stakeholders

Gli Stakeholder promettono al Team:

- Gli stakeholders saranno disponibili per aiutare il Team quando necessario
- Lo ScrumMaster verrà coadiuvato a rimuovere gli ostacoli
- Priorità e vincoli non cambieranno durante uno Sprint senza il consenso del Team
- Partecipare ad uno sviluppo Scrum non danneggerà la carriera dei membri del Team

Modelli di processo 2 XP o Scrum?

XP	Scrum
Orientato alla qualità (test driven)	Orientato al project management
Iterazione: 1-2 settimane	Sprint: 2-4 settimane
Requisiti sempre modificabili	Req modificabili alla fine dello sprint
Il cliente ordina le storie	Il team ordina le storie
Coaching informale	Scrum master certificato
Buone pratiche tipiche di XP: TDD, Pair programming, planning game, refactoring	Buone pratiche tipiche di Scrum: Retrospettiva post-mortem, uso di strumenti di PM, planning poker

Nota bene: XP e Scrum possono coesistere

Riferimenti utili

- www.scrumalliance.org
- www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Italian.pdf
- www.controlchaos.com
- www.mountaingoatsoftware.com/scrum

Una lista di libri su Scrum

- Larman: *Agile and Iterative Development: A Manager's Guide*
- Cohn: *Agile Estimating and Planning*
- Cohn: *Succeeding with Agile*
- Cohn: *User Stories Applied for Agile Software Development*
- Derby & Larsen: *Agile Retrospectives*
- Highsmith: *Agile Software Development Ecosystems*
- Rubin: *Essential Scrum*
- Schwaber & Beedle: *Agile Software Development with Scrum*
- Schwaber: *Scrum and The Enterprise*
- Schwaber: *Agile Project Management with Scrum*

Domande?

