

# Introduzione a Scrum

## Seconda parte



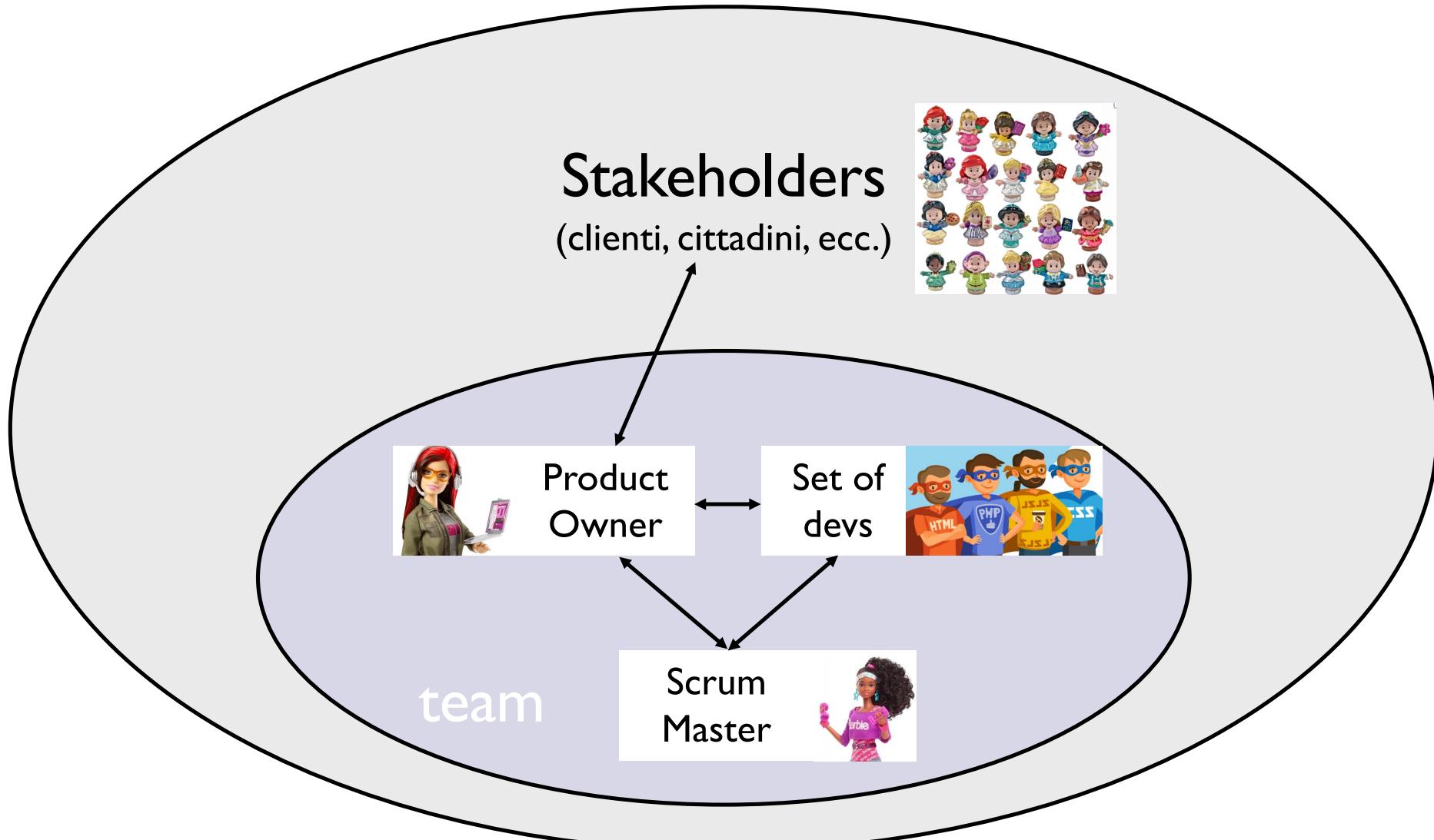
*Corso di Ingegneria del Software  
CdL Informatica Università di Bologna*

# Obiettivi della lezione

- Presentazione del metodo agile Scrum (2)
- Eventi
- Artefatti
- Varianti di Scrum

<https://www.scrum.org/resources/scrum-guide>

# Il team Scrum



# Scrum: gli eventi

## Ruoli

- Product owner
- ScrumMaster
- Team

## Eventi

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

## Artefatti

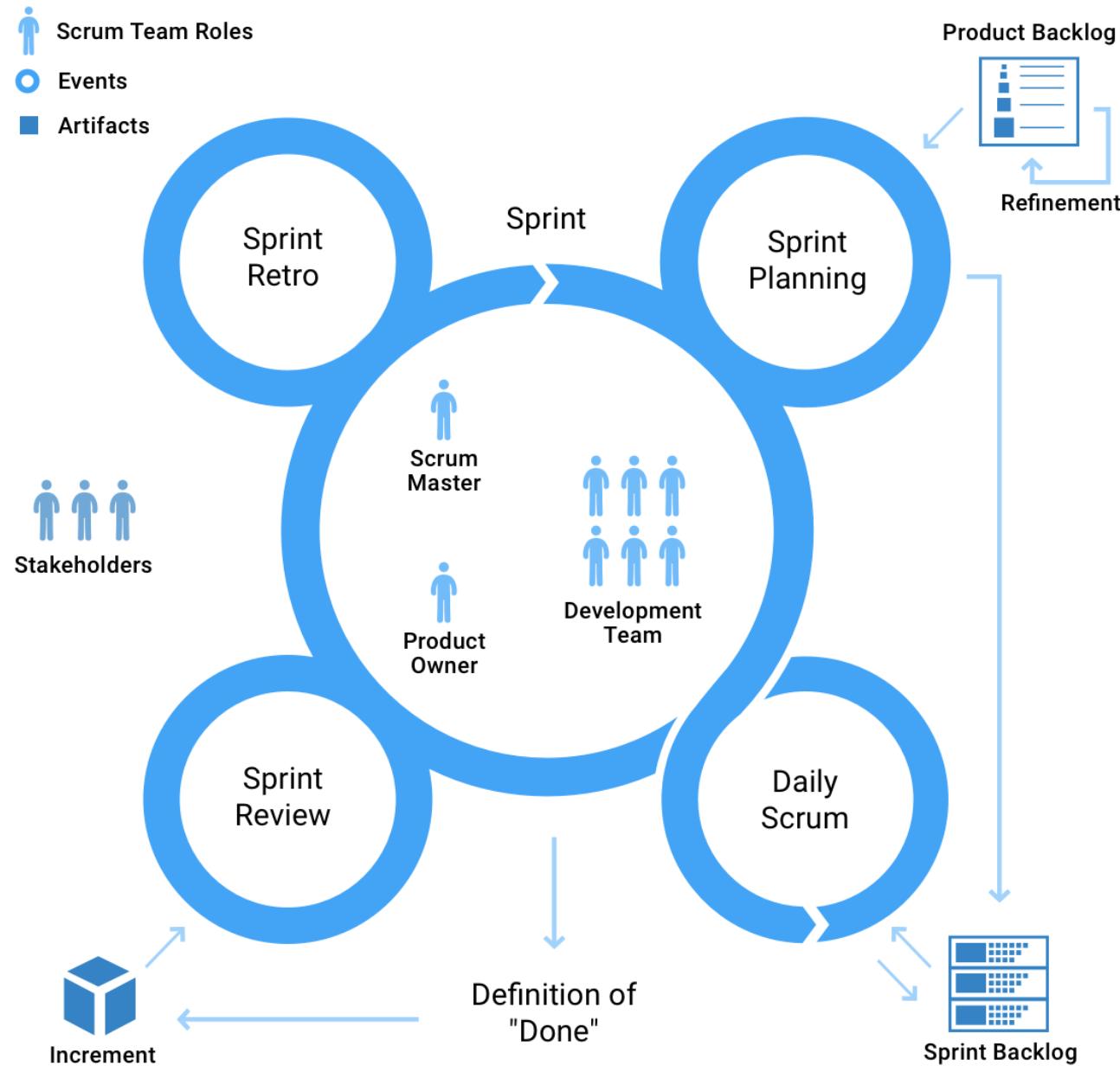
- Product backlog
- Sprint backlog
- Burndown charts

# Gli eventi Scrum

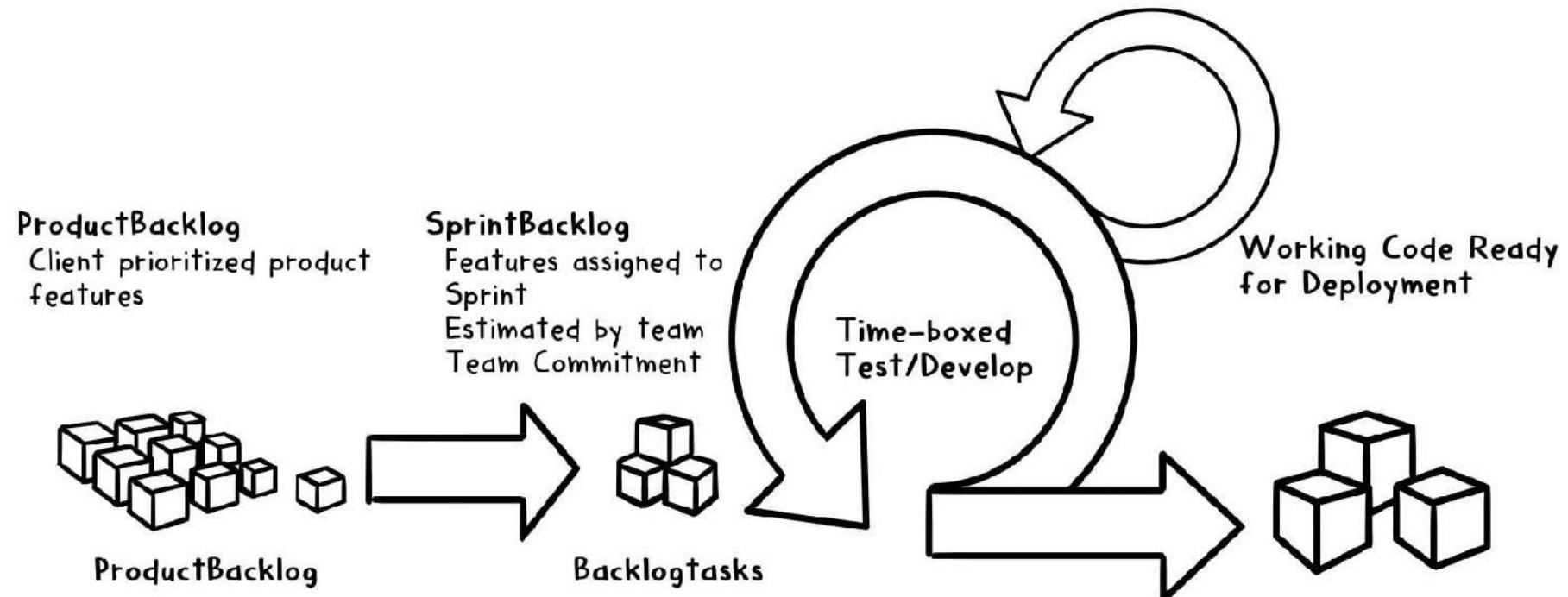
Gli eventi sono riunioni del team (partecipano tutti, incluso PO)

- **Sprint planning:** PO e team decidono cosa fare (sprint backlog) e come rivedere -se occorre- il product backlog
  - Durata: circa un'ora per ogni settimana dello sprint
- **Daily scrum:** ogni sviluppatore dice cosa ha fatto ieri, cosa pianifica per oggi, che impedimenti ha trovato
  - Durata: Max 15 minuti, in piedi
- **Sprint Review:** demo del prodotto: il team mostra al PO cosa è stato o non è stato completato in questo sprint
  - Durata: almeno un'ora, max 3
- **Sprint Retrospective:** riguarda il processo: cosa è andato bene e quali impedimenti sono stati trovati (SM+Team+PO)
  - Durata: almeno un'ora, max 3

# Scrum



# Workflow degli eventi Scrum



**Sprint Planning Meeting**

- Review Product Backlog
- Estimate Sprint Backlog
- Commit

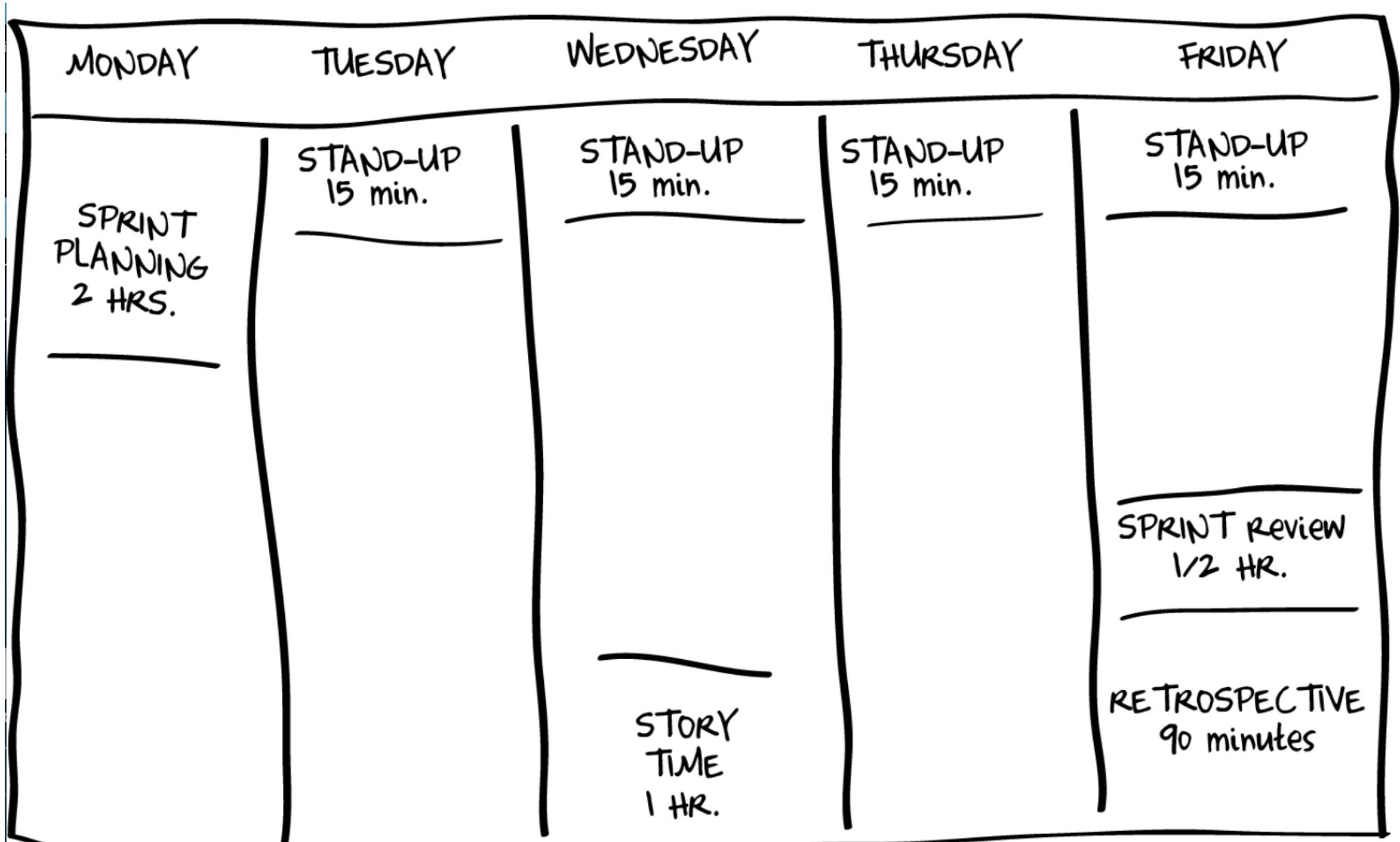
**Daily Scrum Meetings**

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

**Sprint Review Meeting**

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

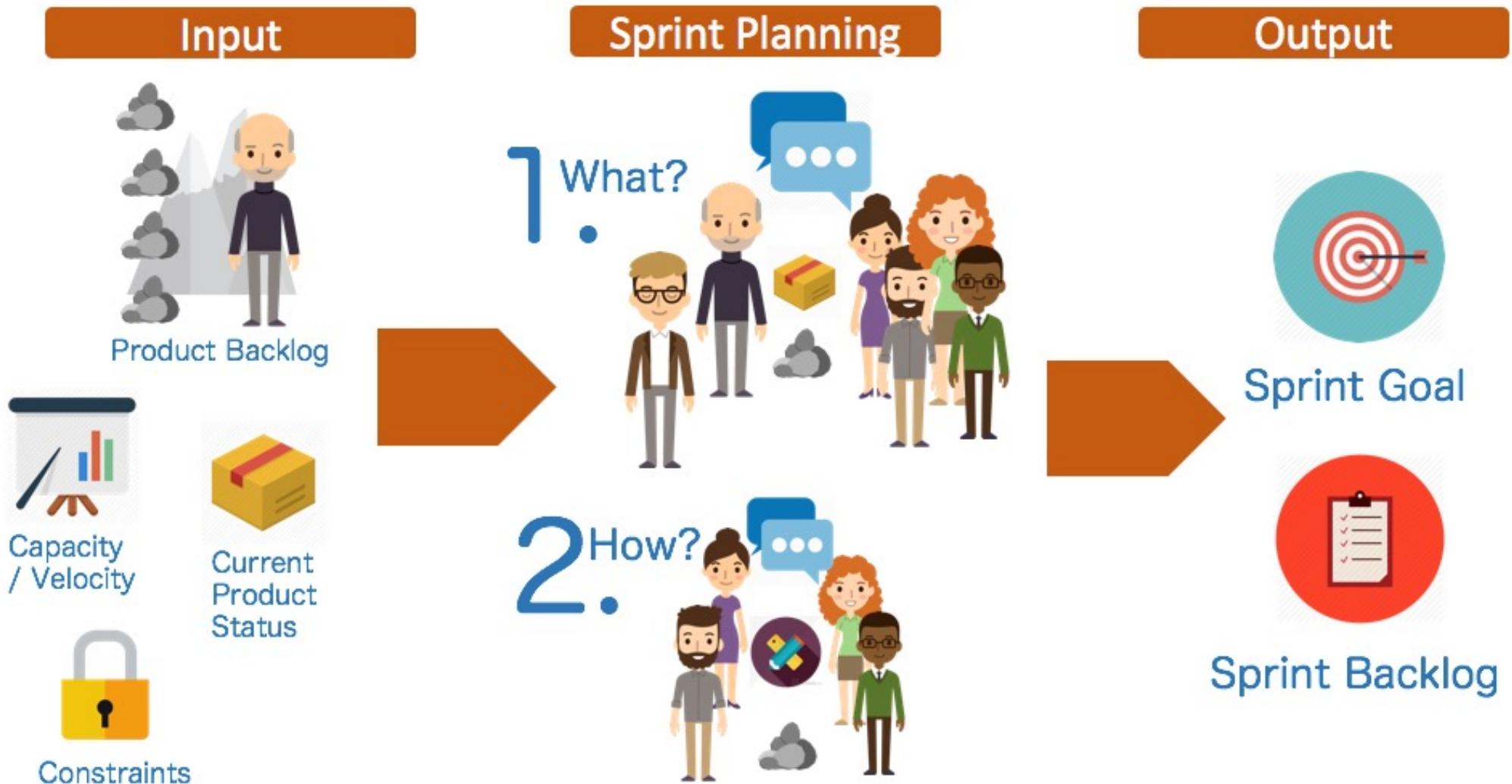
# Struttura di uno sprint su base settimanale

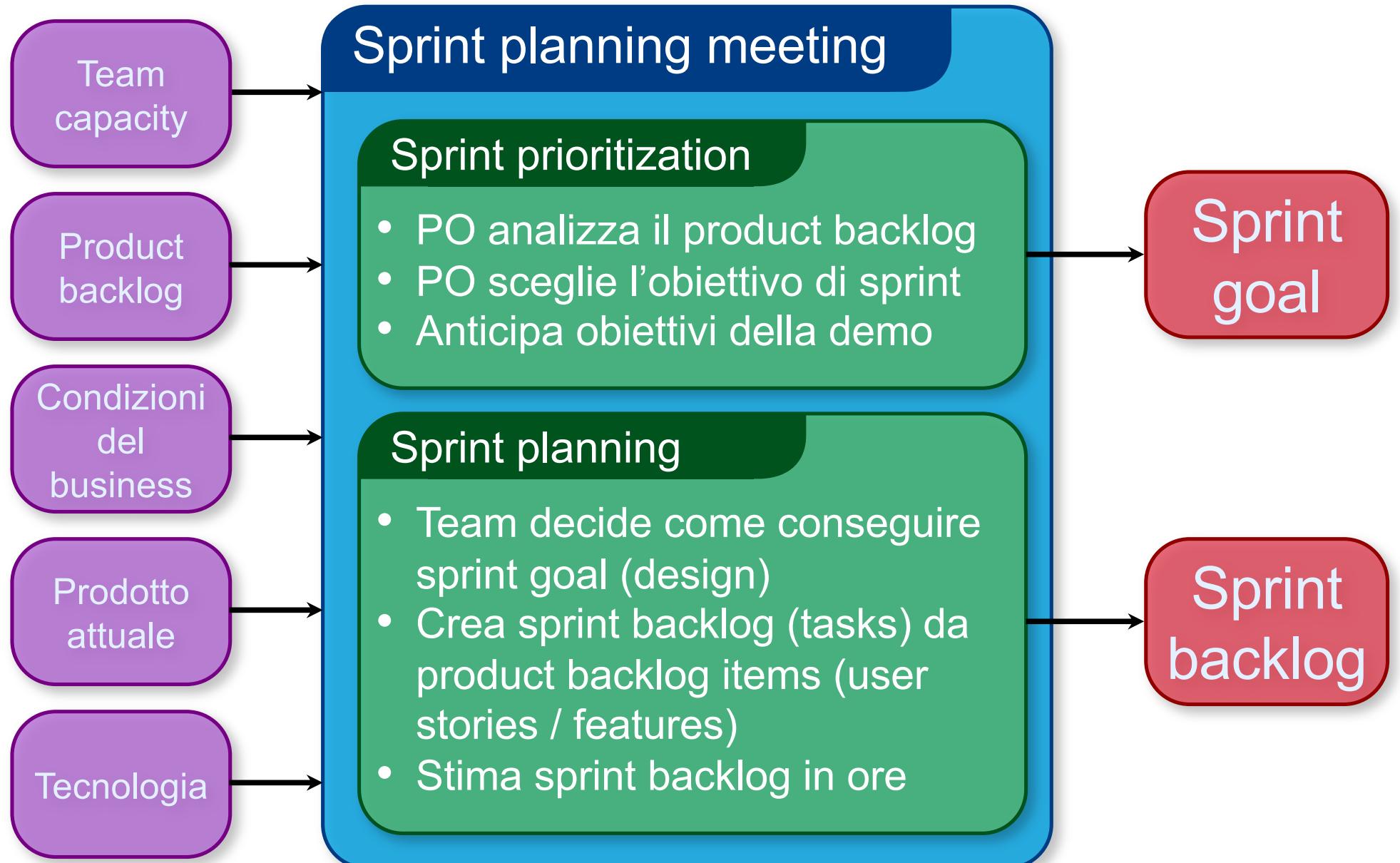


# Pianificare lo sprint

- Ogni sprint inizia con una riunione di pianificazione delle attività (a cura di PO e Team) – questa riunione dura almeno un’ora per ciascuna settimana di sprint
- Il Team **stima** quanto lavoro riuscirà a fare durante lo sprint, tenendo presente la sua «velocità» di sviluppo
- Il Team decide quale parte del backlog di prodotto viene estratta per diventare il backlog di sprint: le user story di maggior priorità vengono scelte assieme al PO
- Tale pianificazione ha per risultato la scelta di alcune user story del backlog di prodotto: tale scelta forma il **backlog di sprint**
- Viene concordata tra PO e Team la «**Definition of Done**» (DoD) per tutti gli elementi del backlog di sprint: il PO accetta come completate solo le US che soddisfano la DoD

# Sprint planning





# Come il PO organizza lo sprint planning

Preparare una sintesi dei bisogni degli stakeholder

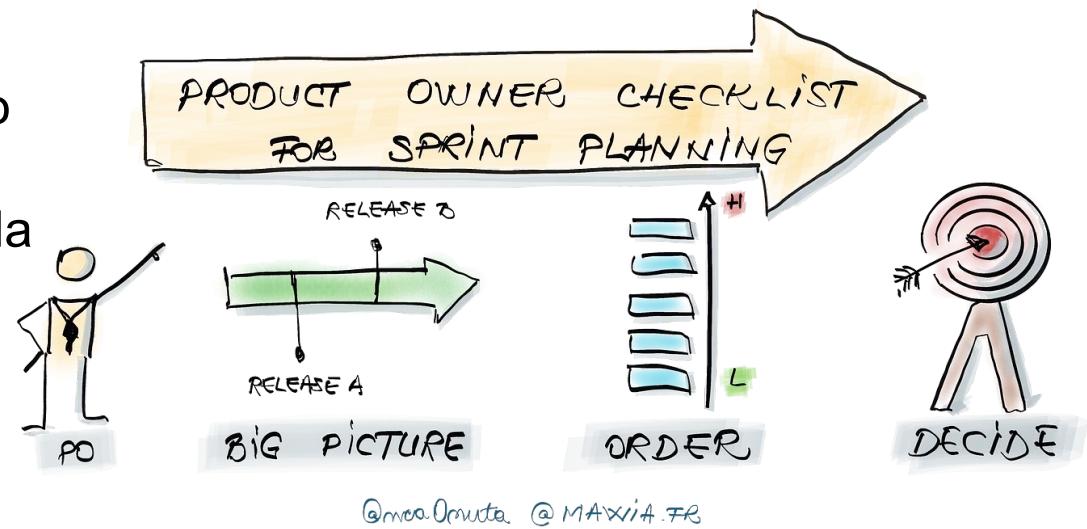
Selezionare e ordinare le User Stories proposte al prossimo sprint

Rivedere le US in modo che abbiano tutte le info: descrizione, criteri di accettazione, legame alla funzione da realizzare, dipendenze da altre US

Rivedere la Definition of Ready e assicurarsi che le US scelte siano pronte

Preparare uno sprint goal e un'agenda per la sprint review

Aiuta il team a definire i task associati a ciascuna US



# La definizione di Pronto (definition of Ready)

Una storia è Ready quando:

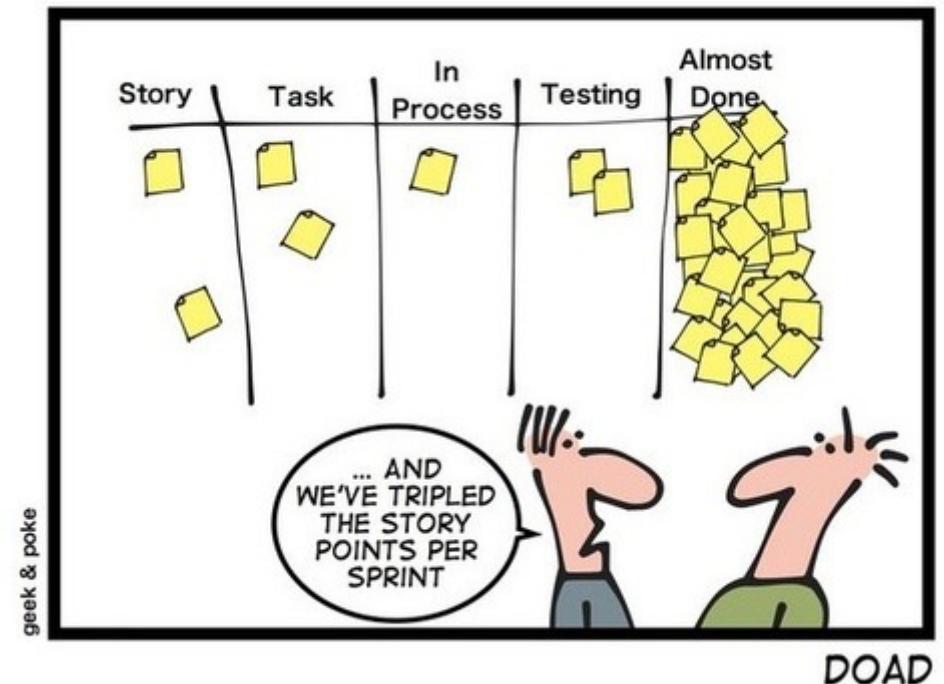
- Tutti i membri del team la capiscono
- È dettagliata abbastanza da poter essere poi testata
- Il PO deve comprenderne il valore per poterla ordinare in priorità con le altre
- Il Team dev'essere in grado di farne una stima in modo da completarla entro uno sprint
- La storia è indipendente dalle altre: il Team può lavorarla avendone il pieno controllo

# La definizione di Fatto (Definition of Done)

La Definition of Done descrive lo stato dell'Increment quando questo soddisfa le metriche di qualità richieste per il prodotto.

La Definition of Done crea trasparenza fornendo a tutti una comprensione condivisa di quale lavoro è stato completato come parte dell'Increment.

Se un elemento non soddisfa la Definition of Done, non può essere rilasciato e nemmeno presentato durante la Sprint Review: ritorna nel Product Backlog



# Esempio di 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 col PO all'inizio del processo

Una possibile definizione di Fatto:

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

# Pianificazione dello sprint

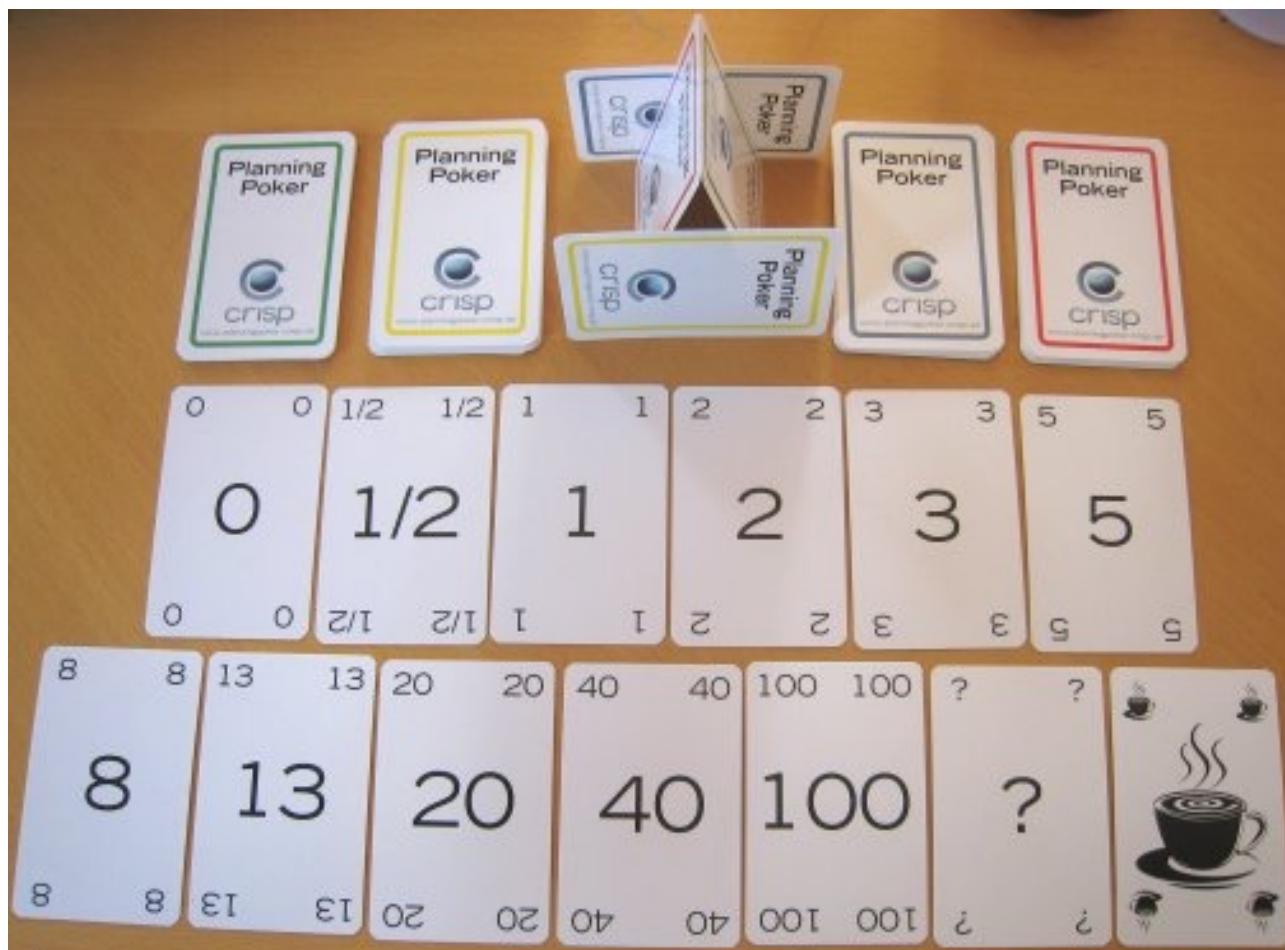
- Il team sceglie dal product backlog gli elementi che verranno realizzati nello sprint
- Si crea lo sprint backlog: i compiti da fare
  - Identificazione e stima di ciascun compito (1-16 hours)
  - Uso di Planning Poker

As a vacation planner, I want to see photos of the hotels.



Code the middle tier (8 hours)  
Code the user interface (4)  
Write test fixtures (4)  
Code the foo class (6)  
Update performance tests (4)

# Planning poker



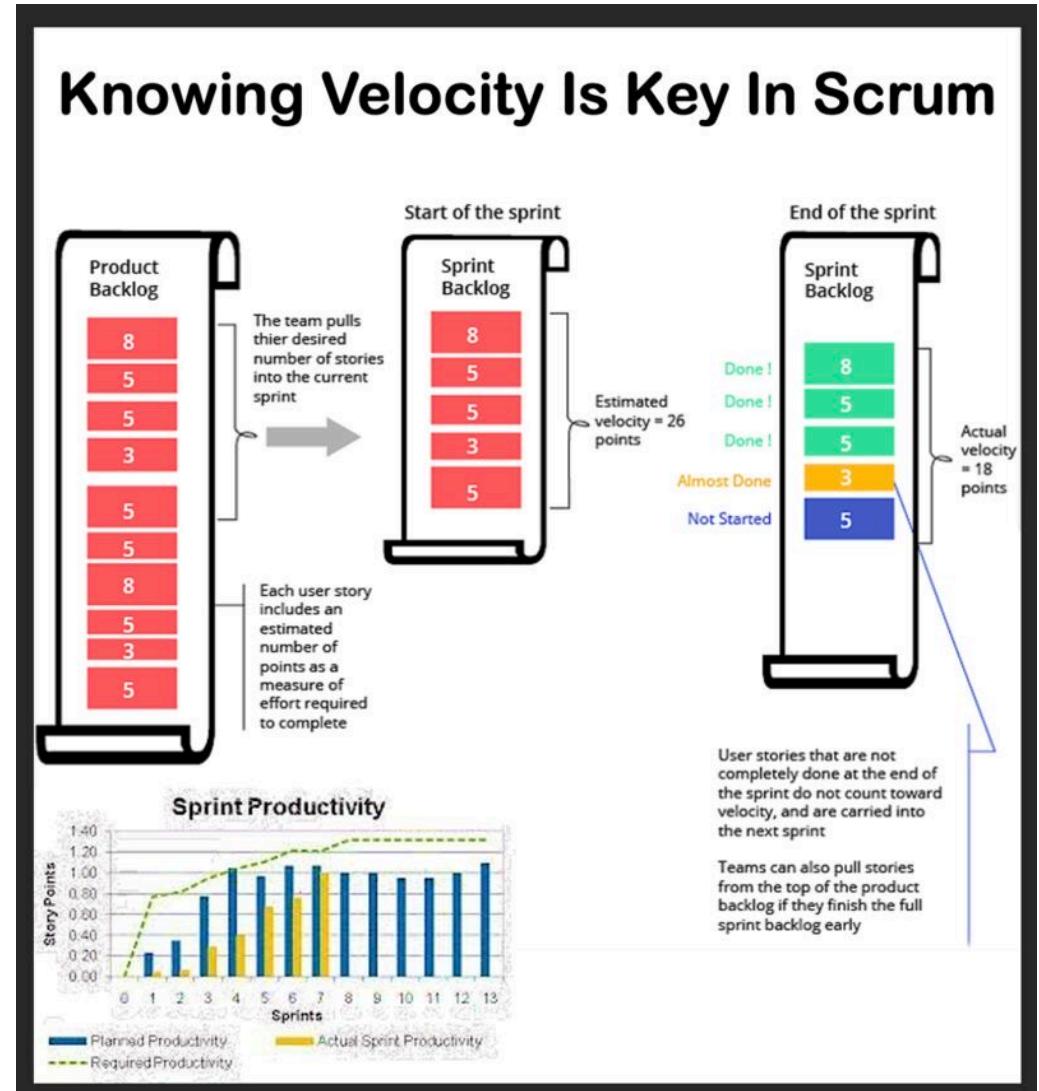
# Pianificazione dello sprint

La “velocità” è la quantità di story point che il team ha completato durante uno sprint

La velocità si calcola alla fine di uno sprint ed è utile per prevedere (stimare) quanto lavoro si riuscirà a fare nello sprint successivo

La velocità si può misurare

- Per ciascun singolo task
- Per ciascuno sprint



# Scrum quotidiano

- Modalità
  - Guardando il board
  - 15-minuti max
  - In piedi
  - Chiunque può assistere
  - Possono parlare solo i pig: membri del team, ScrumMaster, Product Owner
  - Perché in piedi: evitare durata troppo lunga



# Daily scrum: Tre domande a tutti

1

Che hai fatto ieri?

2

Che farai oggi?

3

Prevedi problemi?

Le risposte sono promesse ai compagni di team

# Il PO nel daily scrum

Nell'incontro quotidiano i protagonisti sono i devs

Il PO partecipa al meeting quotidiano, ma

- Non importa se il PO è assente (il meeting inizia comunque)
- Il PO non può inserire nuove US (questo si fa solo all'inizio dello sprint)
- Il PO non pretende di avere chiari gli stati di avanzamento di ciascun dev: gli sviluppatori si autogestiscono
- Il PO partecipa per dare suggerimenti se richiesto

# (Scherzo)

How to make meetings shorter:



# Sprint review: la demo del prodotto

- Il team presenta al PO ciò che ha ottenuto con lo sprint: occorre mostrare come il prodotto è “cresciuto”
- È una demo delle nuove funzioni: serve per verificare lo stato del product backlog
- Riunione informale
  - 2 h di preparazione, niente slide, solo software
- Partecipa tutto il team
- Invitare stakeholder



# Sprint retrospective

- La retrospettiva riguarda il processo (non il prodotto): come sta andando lo sviluppo?
- Occorre analizzare periodicamente cosa va e cosa non va in ciascuno sprint
- Quando: alla fine di ciascuno sprint, dopo la review
- Partecipano tutti:
  - ScrumMaster (coordinatore della retrospettiva)
  - Devs team
  - Product Owner
  - Forse commitenti, utenti e altri stakeholders

# Retrospettiva: Start / Stop / Continue

Il team discute cosa vorrebbe fare:

Start doing

Stop doing

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

# Scrum: gli artefatti

## Ruoli

- Product owner
- ScrumMaster
- Team

## Eventi

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

## Artefatti

- Product backlog
- Sprint backlog
- Burndown charts

# Product backlog



Questo è il  
product backlog

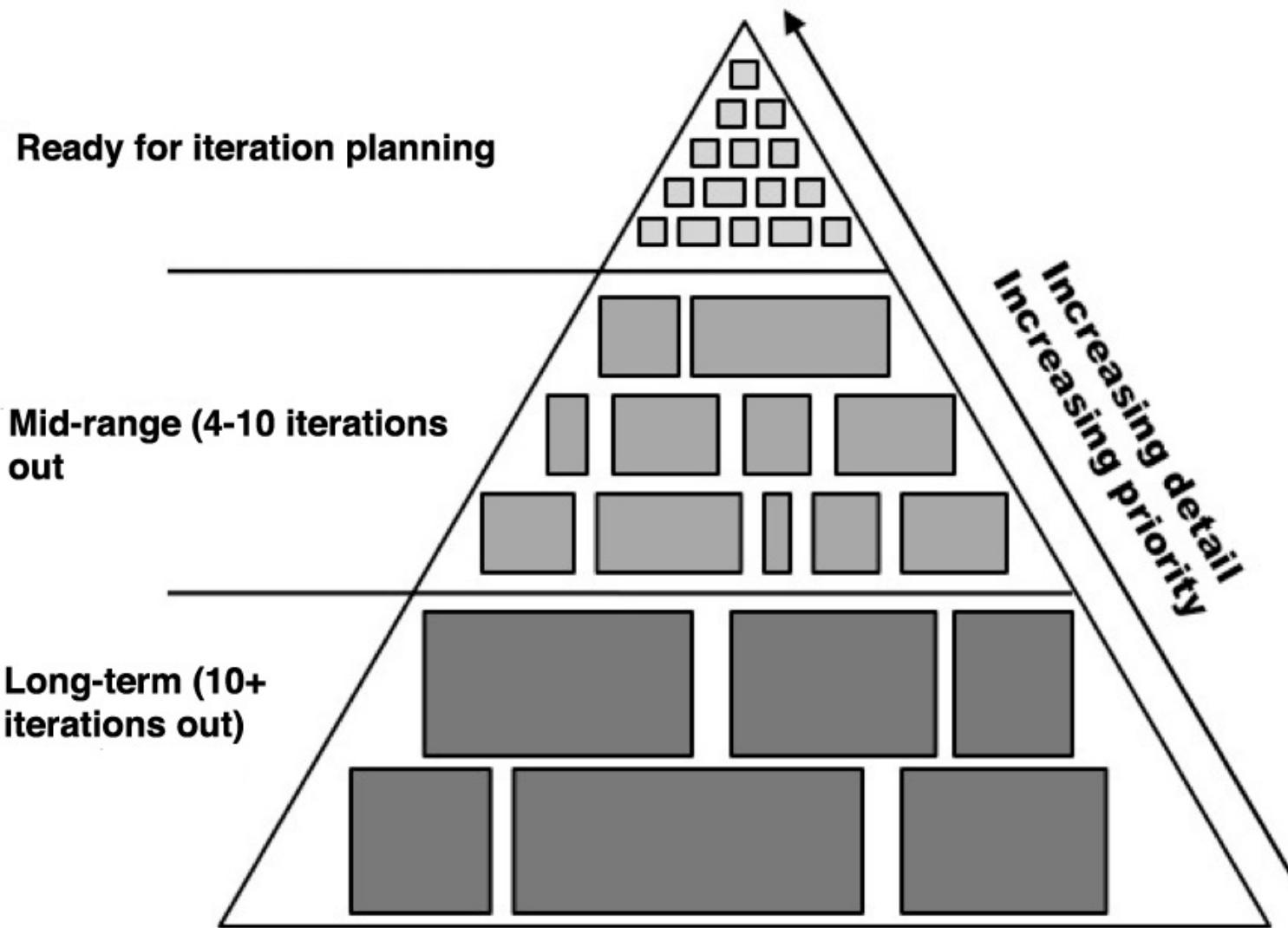
- Product backlog = Lista dei requisiti in forma di user stories
- Definita in modo tale che ciascun elemento abbia valore per gli utenti o i committenti del prodotto
- Messa in priorità dal Product Owner
- Priorità ridefinite all'inizio di ogni sprint

# Esempio di product backlog

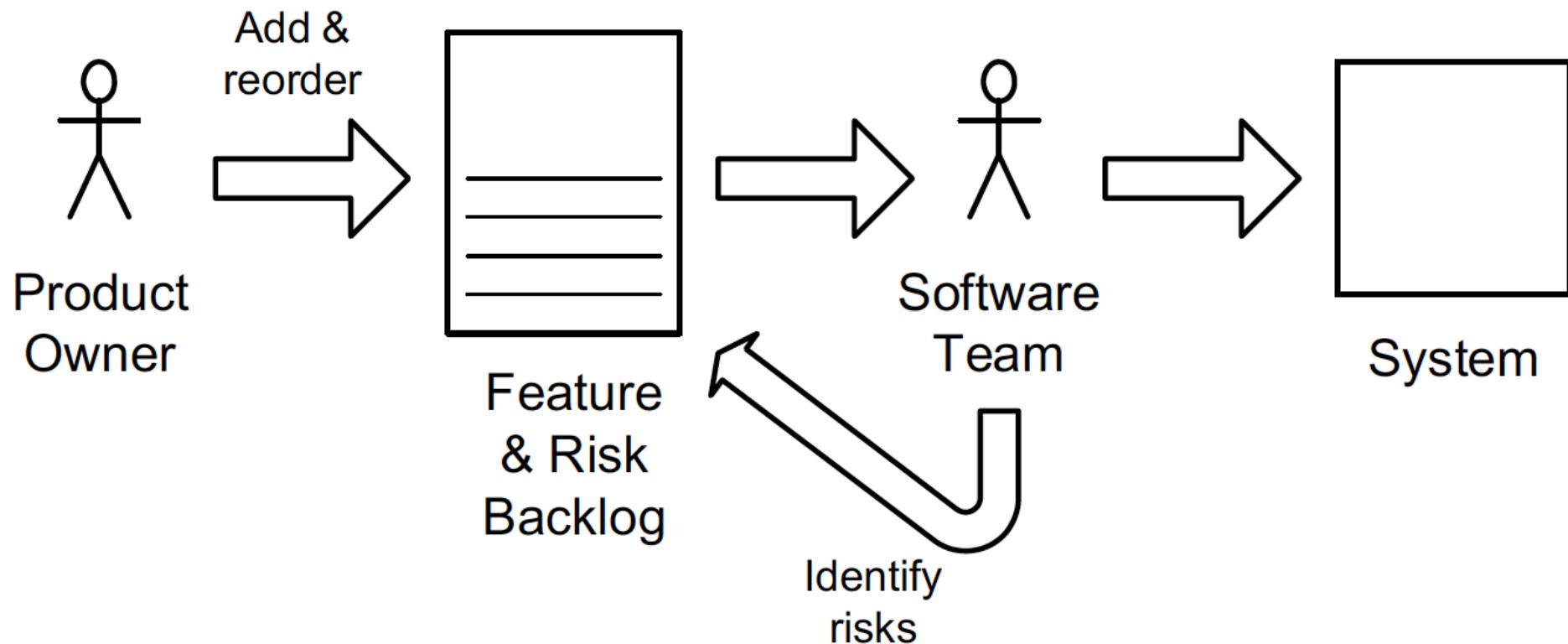
Elemento backlog	Stima (h)
Allow a guest to make a reservation	3
As a guest, I want to cancel a reservation.	5
As a guest, I want to change the dates of a reservation.	3
As a hotel employee, I can run RevPAR reports (revenue-per-available-room)	8
Improve exception handling	8
...	30
...	50

# La dinamica del product backlog

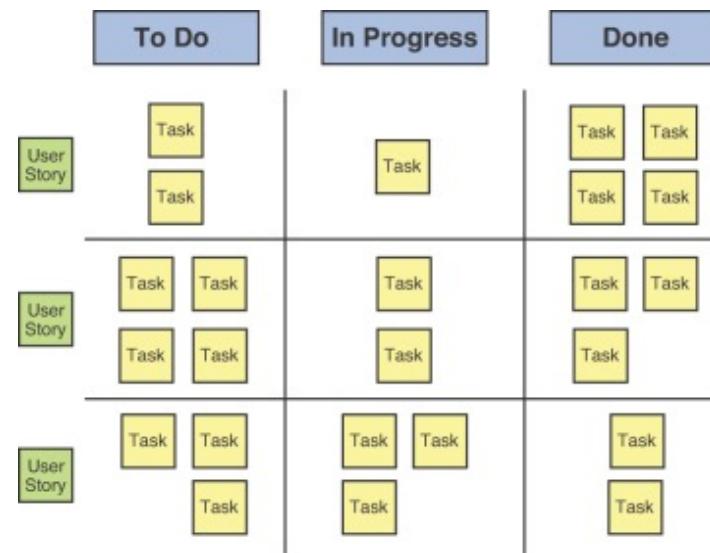
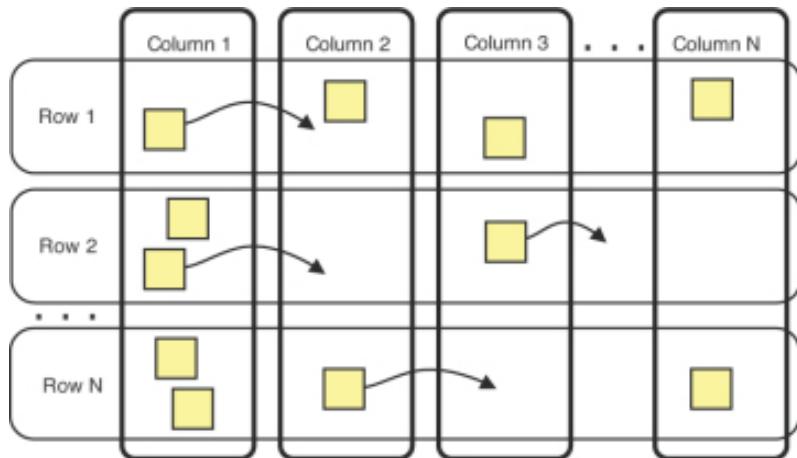
**Break down stories as they move up the backlog**



# Variante del product backlog con analisi dei rischi



# Come si usa il backlog



# The scrum board



# Uno scrum board on line: Jira

The screenshot shows the Jira web interface for a project titled "Progetto di prova 1". The main view is the "Board" tab, which displays three columns: "TO DO" (0 items), "IN PROGRESS" (0 items), and "DONE" (0 items). Below the board, there is a large graphic of a Scrum board with cards and a central title: "Visualize your work with a board". A call-to-action button at the bottom says "Create your first item". To the right, a "Quickstart" sidebar provides links to "Project created", "Get started with list", and other features like "View dates on calendar" and "Plan ahead with timeline". The top navigation bar includes links for "Jira", "Your work", "Projects", "Filters", "Dashboards", "People", "Apps", and a "Create" button. The address bar shows the URL "paolociancarini.atlassian.net".

# Obiettivo dello sprint (sprint goal)

È una breve descrizione del lavoro da fare durante lo sprint. Serve a aumentare la consapevolezza del team, proponendo un obiettivo visibile e condiviso

SMART: **s**pecifico, **m**isurabile, **a**cquisibile, **r**ilevante, **t**empestivo

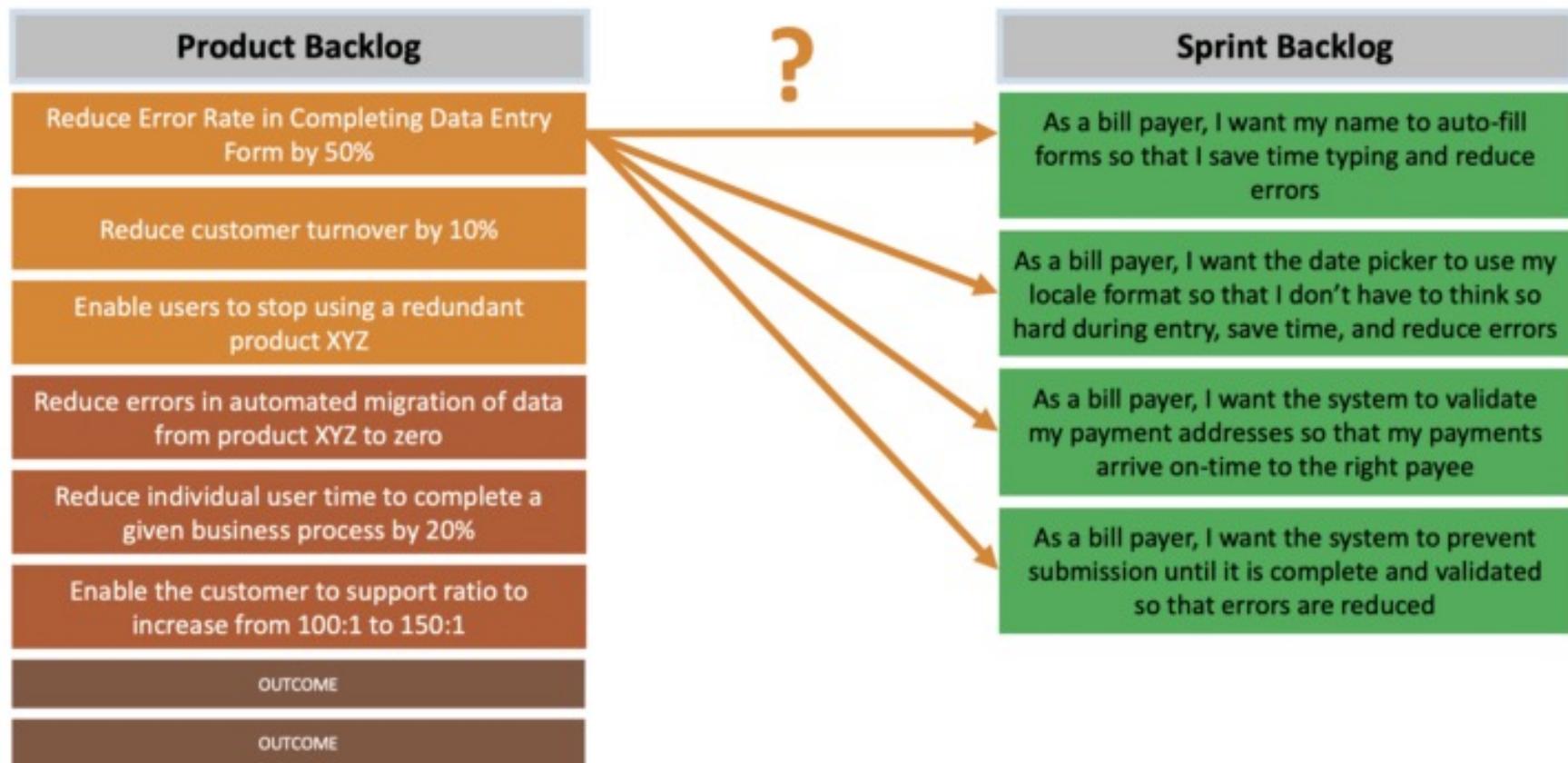
Esempi:

“Impariamo qual è il modo migliore di permettere ad un utente di registrarsi”

“Rendiamo disponibile ai clienti il carrello elettronico”

“Raccogliamo e visualizziamo tutti i tweet disponibili su una specifica trasmissione TV”

# Dal backlog di prodotto a quello di sprint



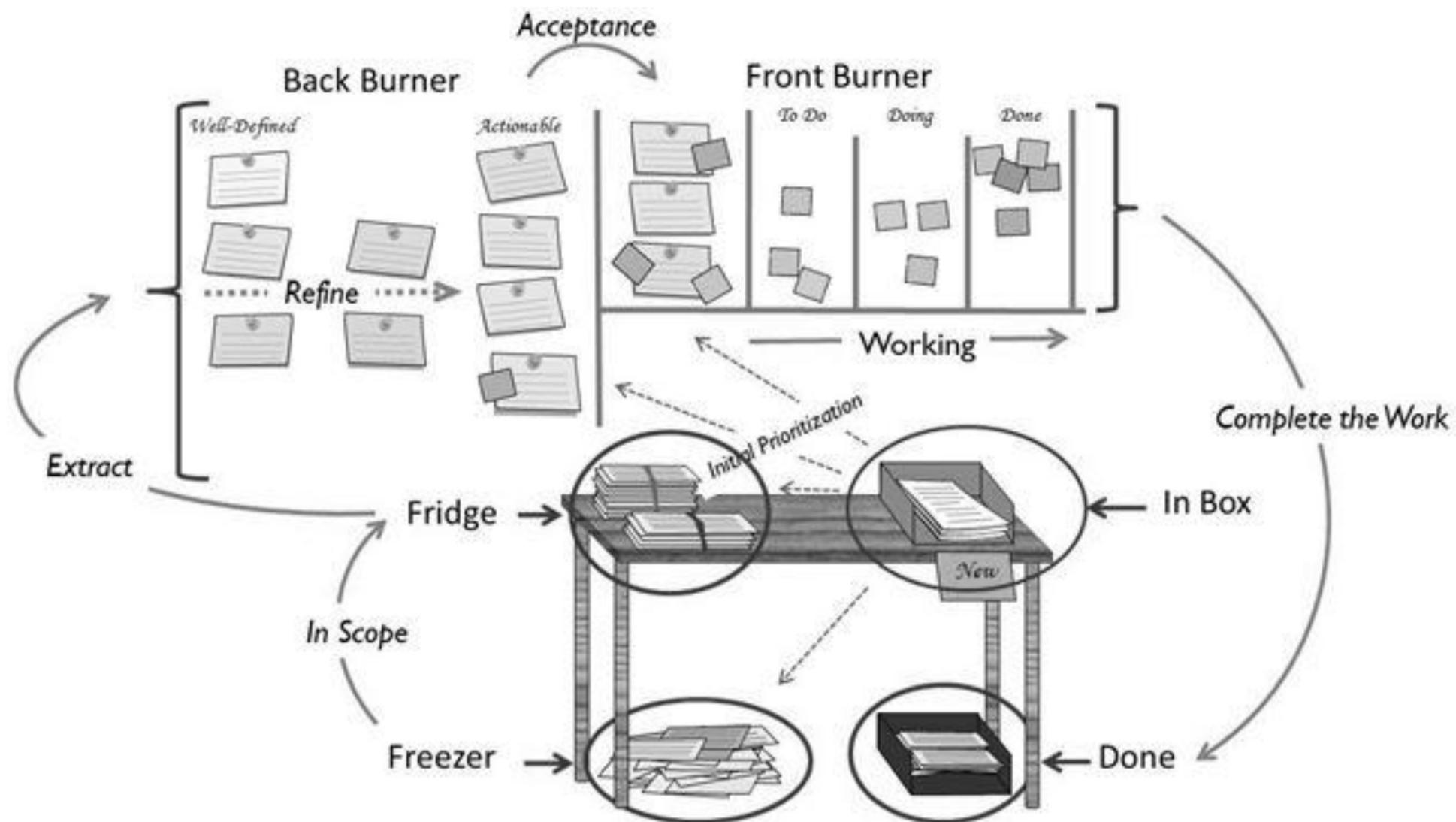
# Gestione dello sprint backlog

- 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 viene conservato nel tabellone condiviso
- Occorre aggiornare il lavoro da fare man mano che viene fuori
- 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

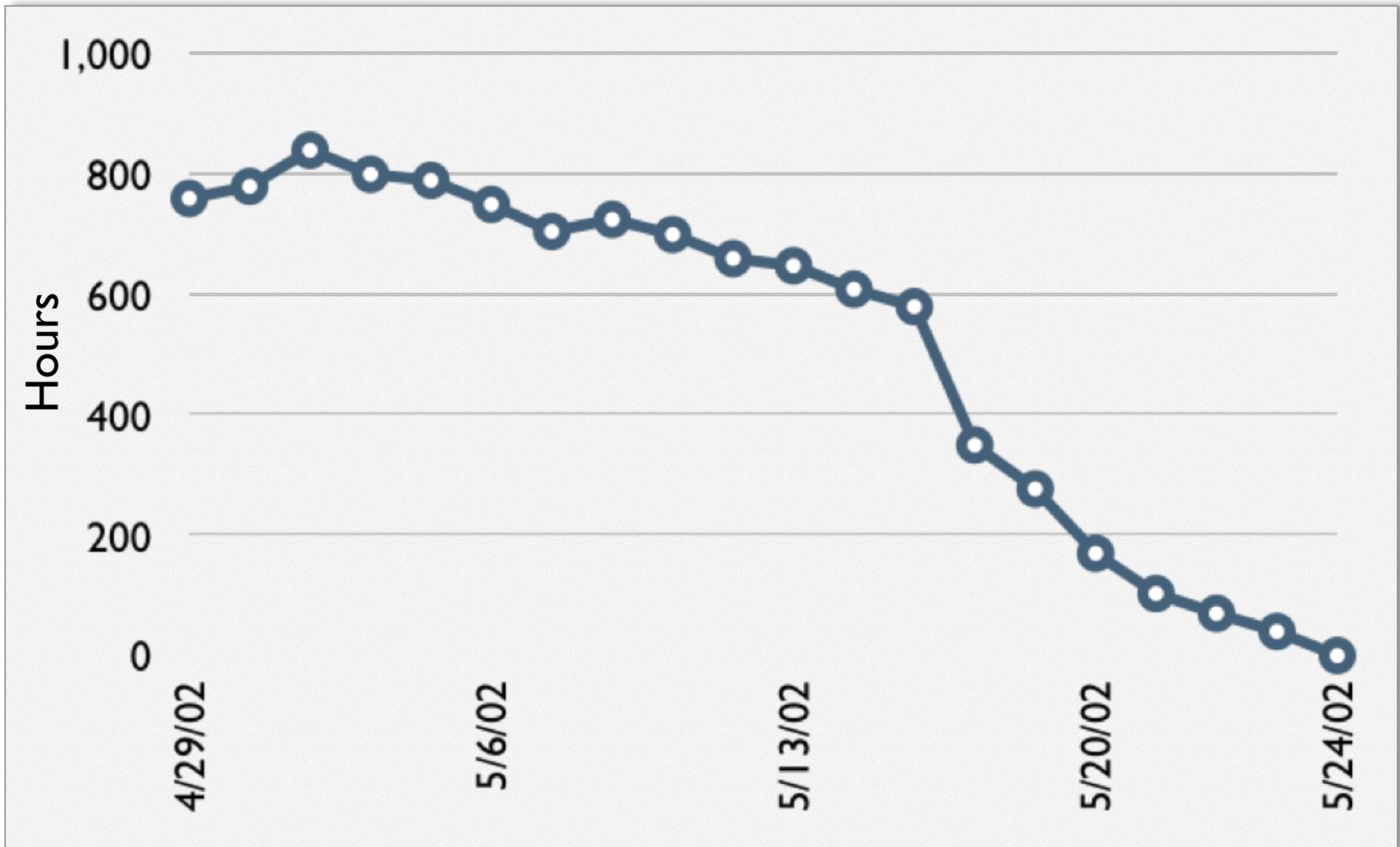
# Il ciclo dei compiti in uno sprint



# Uno sprint backlog

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	

# Un burndown chart di uno sprint



# Scrum e i principi agili

1. **\*\*Soddisfare il cliente\*\*** attraverso la continua consegna di software di valore:

- Scrum favorisce rilasci frequenti grazie agli sprint, che permettono di mostrare e consegnare rapidamente funzionalità funzionanti al cliente.

2. **\*\*Accogliere i cambiamenti\*\*** anche in fasi avanzate dello sviluppo:

- Il backlog del prodotto può essere continuamente aggiornato per riflettere nuovi requisiti, facilitando il cambiamento senza interrompere il ciclo di lavoro.

3. **\*\*Rilasciare frequentemente software funzionante\*\***, con cicli brevi:-

Gli sprint di Scrum sono iterazioni brevi che culminano sempre nella presentazione di un incremento di prodotto funzionante.

4. **\*\*Collaborazione quotidiana tra business e team di sviluppo\*\***:

- Scrum promuove la collaborazione continua attraverso meeting giornalieri (daily stand-up) e la partecipazione attiva del PO, che rappresenta gli interessi del business.

5. **\*\*Supportare e motivare gli individui\*\***:

- Scrum enfatizza l'auto-organizzazione dei team, dando autonomia e fiducia ai membri per prendere decisioni su come meglio affrontare il lavoro.

6. **\*\*Favorire la comunicazione faccia a faccia\*\***:

- Scrum incoraggia la comunicazione diretta e continua tra il team e gli stakeholder, grazie ai daily stand-up e le review.

# Scrum e i principi agili

## 7. \*\*Misurare il progresso con il software funzionante\*\*:

- La principale misura di progresso in Scrum è l'incremento funzionante che viene rilasciato al termine di ogni sprint, piuttosto che documentazione o piani formali.

## 8. \*\*Promuovere uno sviluppo sostenibile\*\*:

- Scrum fissa un ritmo sostenibile di lavoro con sprint di durata fissa, evitando sovraccarichi di lavoro e aiutando il team a mantenere un ritmo costante.

## 9. \*\*Mantenere alta l'attenzione alla qualità e all'eccellenza tecnica\*\*:

- Scrum richiede revisioni continue, con sprint review e retrospettive, per migliorare il prodotto e il processo, assicurando qualità tecnica e mantenimento del debito tecnico.

## 10. \*\*Semplicità, ovvero massimizzare il lavoro non svolto\*\*:

- Il PO prioritizza il backlog per garantire che il team lavori solo sulle funzionalità più importanti, minimizzando gli sforzi su attività che non portano valore immediato.

## 11. \*\*I team auto-organizzati producono i migliori risultati\*\*:

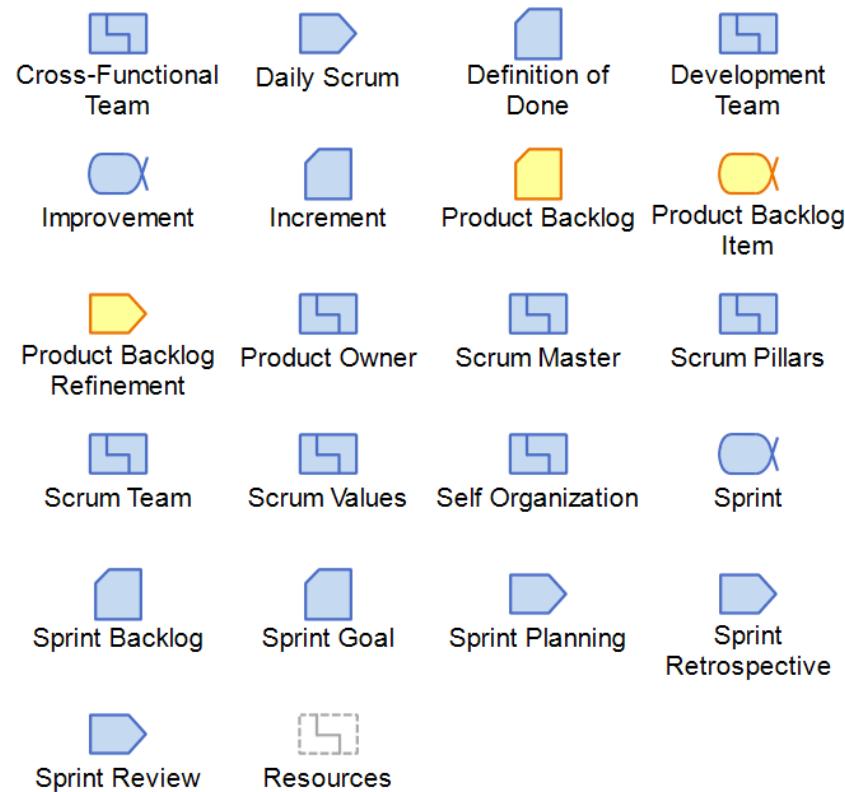
- Scrum incoraggia team cross-funzionali e auto-organizzati, dove i membri del team decidono autonomamente come affrontare i task dello sprint.

## 12. \*\*Riflettere e adattarsi regolarmente\*\*:

- La retrospettiva alla fine di ogni sprint è un momento dedicato all'auto-miglioramento, in cui il team riflette su cosa è andato bene e cosa può essere migliorato, adattando il processo.

# Scrum Essentials

Scrum is a framework for developing, delivering, and sustaining complex products.

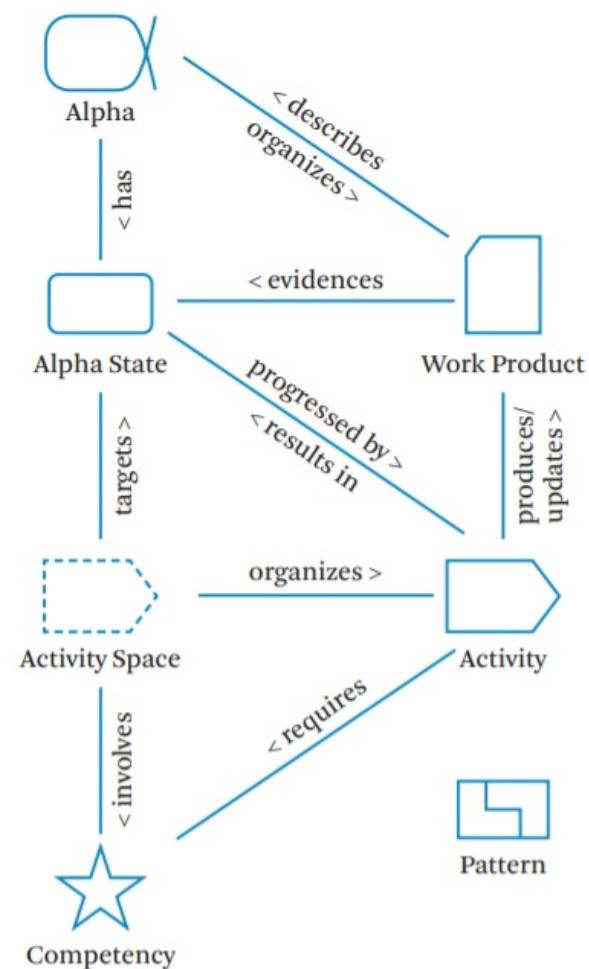


2.04

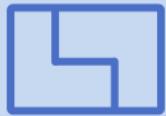
Customer

Solution

Endeavor



Legenda dei simboli Essence



## Scrum Pillars

Scrum is founded on empirical process control theory, or empiricism. Three pillars uphold every implementation of empirical process control:

- Transparency
- Inspection
- Adaptation

In turn they are the foundation of every successful Scrum implementation.

Applies to:  Team and

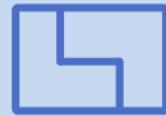
Way of Working



 IVAR JACOBSON  
INTERNATIONAL  
Generated by IJI Practice Workbench™

**scruminc.**

2.04



## Scrum Team

The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master.

Scrum Teams are:

- Self organizing
- Cross-functional
- Flexible
- Creative
- Productive

Scrum Teams deliver products iteratively and incrementally, maximizing opportunities for feedback.

Applies to:  Team



 IVAR JACOBSON  
INTERNATIONAL  
Generated by IJI Practice Workbench™

2.04

# Riferimenti utili

- [scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Italian.pdf](https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Italian.pdf)
- [scrumprimer.org/primers/it\\_scrumprimer20.pdf](https://scrumprimer.org/primers/it_scrumprimer20.pdf)
- [www.scrum.org](https://www.scrum.org)
- [www.scrumalliance.org](https://www.scrumalliance.org)
- [www.mountaingoatsoftware.com/scrum](https://www.mountaingoatsoftware.com/scrum)
- [www.controlchaos.com](https://www.controlchaos.com) (repository di articoli storici su Scrum)

# Una lista di libri su Scrum

- Derby & Larsen, Agile Retrospectives, 2006
- Rubin, *Essential Scrum*, 2012

# Domande?

