

# Idea



# Post-it



# Test Verdi!

**Alberto Brandolini**

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

@ziobrando

<http://ziobrando.blogspot.com>

**Andrea Balducci**

[andrea.balducci@prxm.it](mailto:andrea.balducci@prxm.it)

@andreabalducci

# Idea —> Post-it —> Test verdi



**Alberto Brandolini**

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

@ziobrando

<http://ziobrando.blogspot.com>

**Andrea Balducci**

[andrea.balducci@prxm.it](mailto:andrea.balducci@prxm.it)

@andreabalducci



Grazie a



Sponsor



# About me



**@ziobrando**

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



**avanscoperta**

# About me

Certified DDD Trainer



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer  
Post-it addicted



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer  
Post-it addicted  
Visual thinker



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer

Post-it addicted

Visual thinker

Chaos summoner



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer

Post-it addicted

Visual thinker

Chaos summoner

Idea thief



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer

Post-it addicted

Visual thinker

Chaos summoner

Idea thief

Lean & Agile



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



avanscoperta

# About me

Certified DDD Trainer

Post-it addicted

Visual thinker

Chaos summoner

Idea thief

Lean & Agile

Entrepreneur



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



XPUG - Romagna



**avanscoperta**

# About me

Certified DDD Trainer

Post-it addicted

Visual thinker

Chaos summoner

Idea thief

Lean & Agile

Entrepreneur

...never satisfied



[@ziobrando](#)

[alberto.brandolini@avanscoperta.it](mailto:alberto.brandolini@avanscoperta.it)

DDD-IT



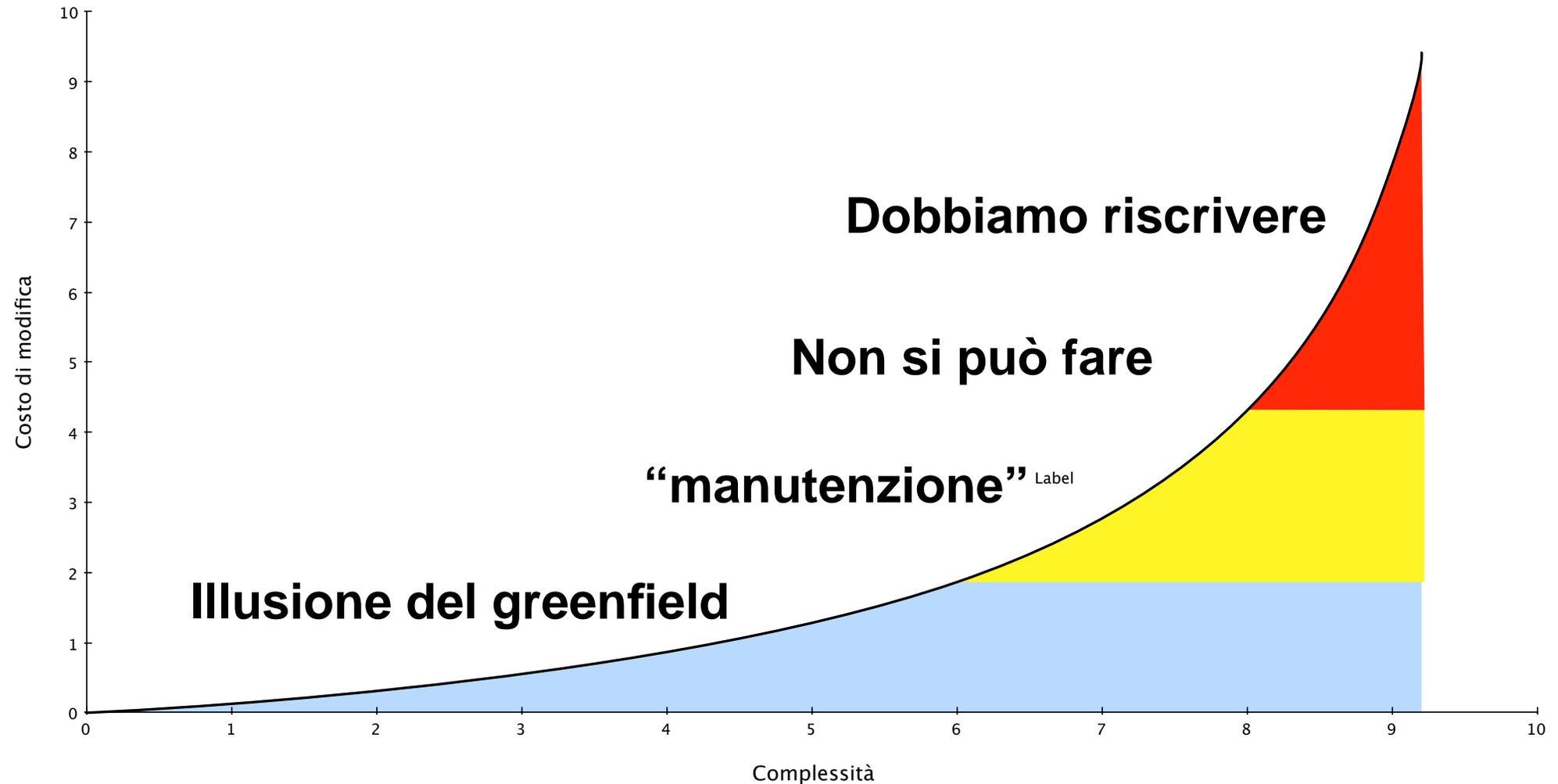
XPUG - Romagna



**avanscoperta**

Il problema

# Costo delle modifiche





# SLAVERY

IT GETS THINGS DONE

# Imponente





**Imponente**

**Flessibile?**



Ooops....

Il miglior modo per  
rimuovere il pattume dal  
sistema è...

**non mettere pattume**  
nel sistema

Altrimenti...



Application

Application

Application

Application

Application

Application

Application

Application

Database

Database

ci ritroviamo con la solita architettura



**...da portare  
sul cloud**





Quando si fa  
refactoring?



Quando si fa  
refactoring?

**Mañana!!**

***Ignorance is the single  
greatest impediment  
to throughput.***

***Dan North***

# ***Learning is the constraint***

**Dan North**

# Piano d'azione

# Piano d'azione

1. Cominciare con il piede giusto

# Piano d'azione

1. Cominciare con il piede giusto
2. Rimuovere gli ostacoli all'apprendimento

# Piano d'azione

1. Cominciare con il piede giusto
2. Rimuovere gli ostacoli all'apprendimento
3. Iterare alla massima frequenza possibile

# Event Storming!



# Event Storming

- Tutti in una stanza
- Rotolone di carta —> “unlimited modelling surface”
- Partiamo dai **domain events**
- Raffiniamo il modello un passo alla volta

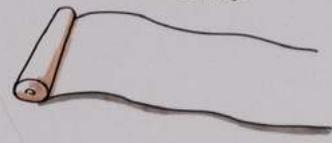
### INVITE THE RIGHT PEOPLE

- THOSE WHO BRING THE QUESTIONS
- THOSE WHO KNOW THE ANSWERS

(!) AND DON'T WORRY IF YOU DON'T GET ALL THE ANSWERS

### UNLIMITED MODELING SPACE

- CAN'T TELL A PROBLEM SIZE BEFORE EXPLORING



### FOCUS ON DOMAIN EVENTS

- SIMPLE SEMANTIC & NOTATION
- EASY TO GRASP BY EVERYBODY IN THE ROOM



### WHAT DOES TRIGGER EVENTS?

- A **COMMAND** FROM A GIVEN **USER**
- OR AN **EXTERNAL SYSTEM**
- OR SIMPLY THE TIME **END OF MONTH**
- OR JUST ANOTHER EVENT **TEXT MESSAGE LATE** **PHONE UPDATED**

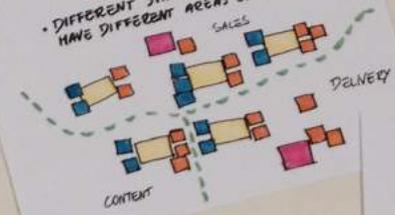
### EXPLORE AGGREGATES OUTSIDE-IN

AN AGGREGATE CAN ACCEPT OR REJECT COMMANDS...  
...AND MOSTLY NEEDS ONLY THE DATA NEEDED FOR THIS PURPOSE



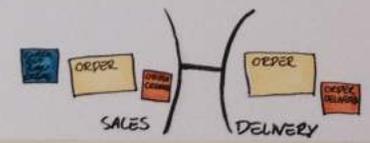
### EXPLORE SUBDOMAINS

- DIFFERENT STAKEHOLDERS WILL HAVE DIFFERENT AREAS OF EXPERTISE



### EXPLORE BOUNDED CONTEXTS

- WHEN DISCUSSIONS ARISE AROUND MEANING AND SEMANTICS, MULTIPLE MODELS ARE PROBABLY INVOLVED



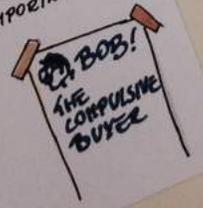
### SKETCH ACCEPTANCE TESTS

- CRITICAL OR AMBIGUOUS USE CASES MIGHT REQUIRE A PRECISE COMPLETION CRITERIA

GIVEN A REGISTERED CUSTOMER BOB AND BOB IS A PREMIUM CUSTOMER WHEN BOB PURCHASES GOODS FOR HOME THEN BOB IS GIVEN A 20% DISCOUNT

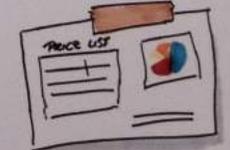
### PERSONAS & USER GOALS

- SOMETIMES WHO IS PERFORMING THE ACTION IS MORE IMPORTANT THAN OTHER DETAILS



### SKETCHING READ MODEL ARTIFACTS

- SOME SCENARIOS REQUIRE A WELL DEFINED SET OF INFORMATION TO BE DISPLAYED TO THE USER





**Tutti in una stanza?**

# Tutti in una stanza!

- Risoluzione anticipata dei conflitti
- Visualizzazione di un artifact
- Possibilità di osservare l'aspetto umano
- Alto livello di energia.

**Si, TUTTO lo spazio**

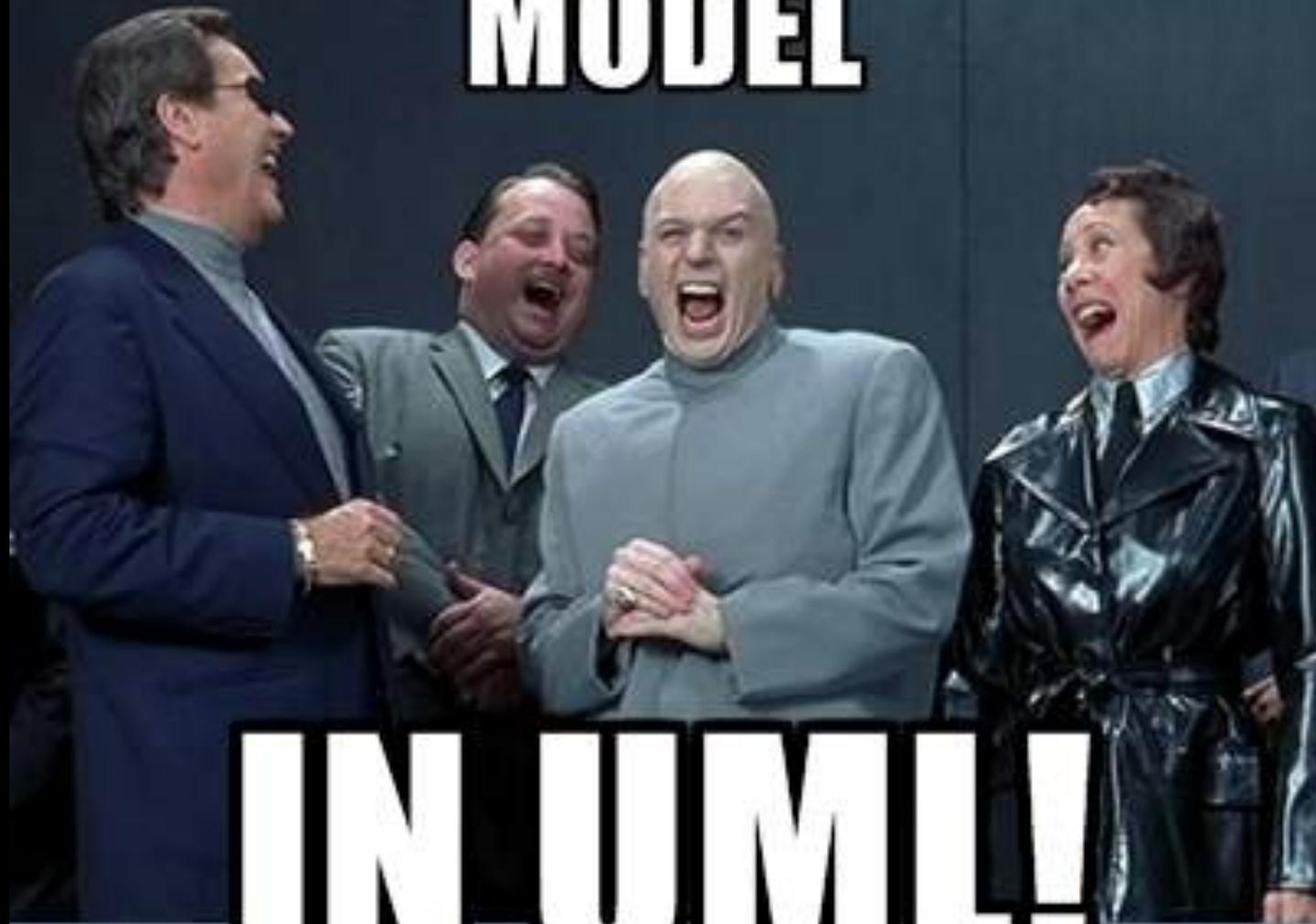


## FOCUS ON DOMAIN EVENTS

- SIMPLE SEMANTIC & NOTATION
- EASY TO GRASP BY EVERYBODY IN THE ROOM



**LET'S SKETCH A  
MODEL**



**IN UML!**

Il dominio...

# Prenotazione risorse

Video!





Che è successo?

# Bounded Contexts

- Lavorando con le **persone** emergono i naturali confini tra
  - le porzioni del dominio —> **subdomains**
  - i modelli in essere —> **Bounded Contexts**

# Ubiquitous Language

- la discussione ci ha portato ad una *precisione* del linguaggio sempre maggiore
- Il linguaggio è **consistente** e preciso all'interno di un **Bounded Context**

# Aggregati ed Invarianti

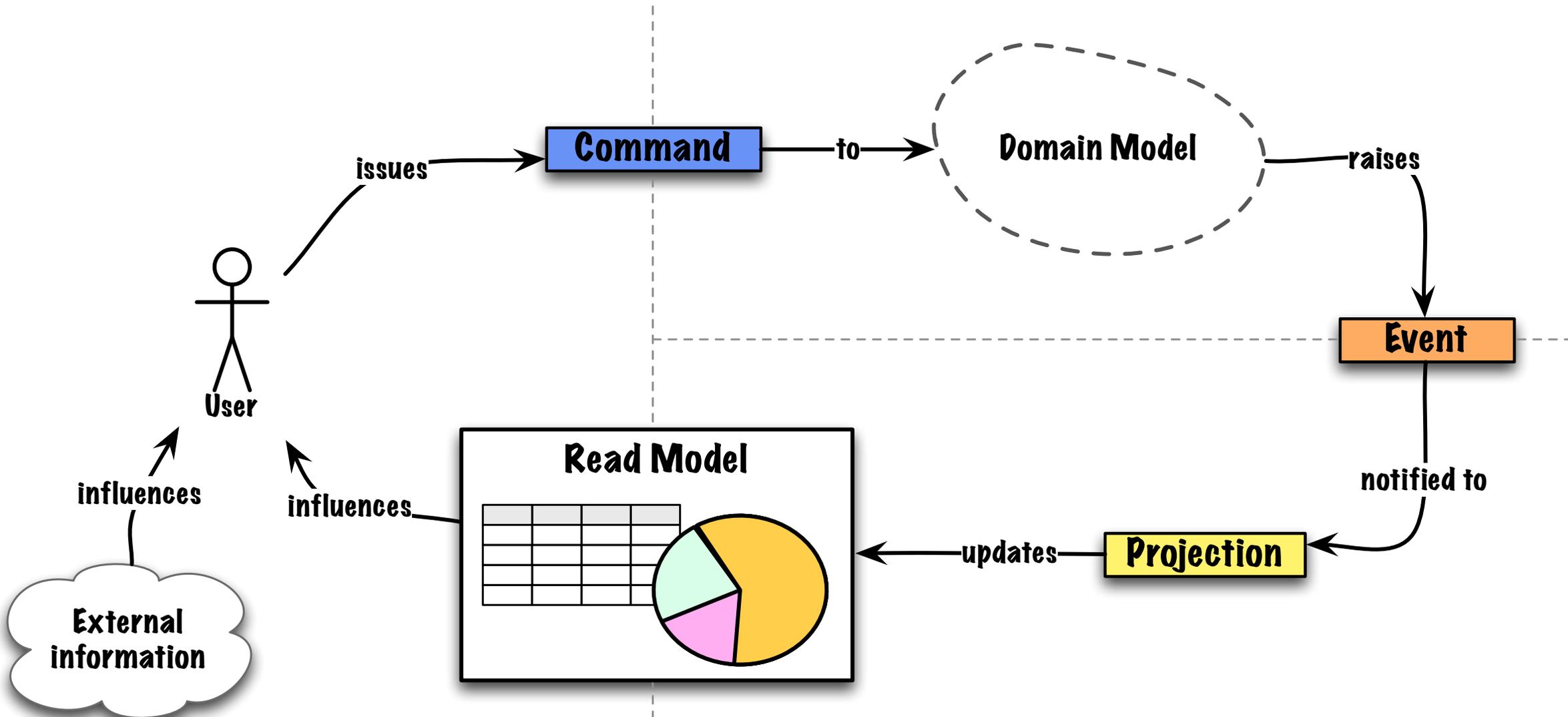
- **aggregati** = unità di consistenza.
- Gli *aggregati* vengono individuati sulla base del **comportamento** del sistema
- Le *invarianti* sono garantite dagli aggregati.

# Roles & Responsibility

**Prima** capiamo cosa sta  
dentro agli aggregati

**Poi** capiamo che  
nome dargli

# Conceptual CQRS



# Read Model

- Gli aggregati prendono decisioni sulla base dello stato corrente.
- Gli utenti prendono decisioni sulla base di quello che vedono...



**Vedo i dati vecchi...**

Cambio!



**dotnet**  
MARCHÉ

**XPUG**  
MARCHÉ

**Dev > Marche**

**<proximo>**

**@andreabalducci**

CommunityDays.it

#CDays14 - Milano 25, 26 e 27 Febbraio 2014

# Event Sourcing



Il nostro **dominio**  
ha una **storia**  
da raccontare...

...ma spesso ci  
accontentiamo  
dei titoli di coda



WINGNUT FILMS PRODUCTION ASSISTANTS CAROLINE MCKAY • CILLA LECKIE  
WINGNUT FILMS RUNNER CARTER NIXON WINGNUT FILMS ACCOUNTANT LINDA KLEIN-NIXON

ACCOUNTANTS AVERIL MAWHINNEY • RICK BAER • EMMA BENDELL

ACCOUNTS ASSISTANTS

DIANNE SUGDEN • ANNE-MARIE DAVENPORT • FA SOLUYAVE

PHILIPPA RACE • MARC TYRON • DOMINIQUE FROMONT

PAYROLL DENISE STEELE • LINUS MURPHY • SEBASTIAN MARR • ANDREW COCHRANE

AERIAL DIRECTOR OF PHOTOGRAPHY PHIL PASTUHOV

CAMERA PILOT ALFIE SPEIGHT

CAMERA OPERATORS

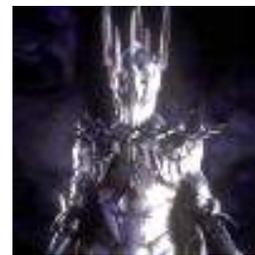
MCCAFFERTY • N MCLEA • WASBENTER • JOE DAY • REWA HARRE • ROB MARSH

EDITORS

LINE

..invece di guardare  
**tutta** la storia.





# Quante informazioni perdiamo ?



Il saldo è 6.684€

<i>Data</i>	<i>Dare</i>	<i>Avere</i>	<i>Descrizione operazione</i>
3/2	160		
4/2		1.500	
4/2		1.400	
10/2	180		
12/2	200		
12/2	180		
14/2	80		
17/2	200		
20/2	16		
20/2	300		
20/2	320		
20/2	75		
21/2	16		
22/2	180		
22/2	200		
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

<i>Data</i>	<i>Dare</i>	<i>Avere</i>	<i>Descrizione operazione</i>
3/2	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
4/2		1.400	ACME Spa - Stipendio Gennaio
10/2	180		Supermercato Le Fornaci - Castelfidardo
12/2	200		Mediaworld Ancona
12/2	180		Pittarello Ancona
14/2	80		Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

<i>Data</i>	<i>Dare</i>	<i>Avere</i>	<i>Descrizione operazione</i>
3/2	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
4/2		1.400	ACME Spa - Stipendio Gennaio
10/2	180		Supermercato Le Fornaci - Castelfidardo
12/2	200		Mediaworld Ancona
12/2	180		Pittarello Ancona
14/2	80		Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Due persone  
Stessa Azienda

<i>Data</i>	<i>Dare</i>	<i>Avere</i>	<i>Descrizione operazione</i>
3/2	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
4/2		1.400	ACME Spa - Stipendio Gennaio
10/2	180		Supermercato Le Fornaci - Castelfidardo
12/2	200		Mediaworld Ancona
12/2	180		Pittarello Ancona
14/2	80		Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Due persone  
Stessa Azienda

Carnivori

<i>Data</i>	<i>Dare</i>	<i>Avere</i>	<i>Descrizione operazione</i>
3/2	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
1/2		1.400	ACME Spa - Stipendio Gennaio
	180		Supermercato Le Fornaci - Castelfidardo
	200		Mediaworld Ancona
	180		Pittarello Ancona
14/2			Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Auto senza telepass

Due persone  
Stessa Azienda

Carnivori

Lui & Lei		Avere	Descrizione operazione
	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
1/2		1.400	ACME Spa - Stipendio Gennaio
	180		Supermercato Le Fornaci - Castelfidardo
	200		Mediaworld Ancona
	180		Pittarello Ancona
14/2			Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Auto senza telepass

Due persone  
Stessa Azienda

Carnivori

Lui & Lei		Avere	Descrizione operazione
	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
1/2		1.400	ACME Spa - Stipendio Gennaio
	180		Supermercato Le Fornaci - Castelfidardo
	200		Mediaworld Ancona
	180		Pittarello Ancona
14/2			Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carpisa Bologna
20/2	75		Hotel Bologna
21/2	16		Autostrade Spa - Casello Ancona
22/2	180		amazon.it
22/2	200		zalando.it
	<b>2.107</b>	<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Due persone  
Stessa Azienda

Auto senza  
telepass

Carnivori

Lei compra entro  
un'ora da lui

Lui & Lei		Avere	Descrizione operazione
	160		Supermercato Le Fornaci - Castelfidardo
4/2		1.500	ACME Spa - Stipendio Gennaio
1/2		1.400	ACME Spa - Stipendio Gennaio
	180		Supermercato Le Fornaci - Castelfidardo
	200		Mediaworld Ancona
	180		Pittarello Ancona
14/2			Ristorante Osteria da Pippo
17/2	200		IperSimply- Castelfidardo
20/2	16		Autostrade Spa - Casello Bologna
20/2	300		Mediaworld Bologna
20/2	320		Carnisa Bologna
20/2			Bologna
21/2			strade Spa - Casello Ancona
22/2			on.it
22/2	200		zalando.it
<b>2.107</b>		<b>2.900</b>	<b>Saldo al 22/2 : 6.684</b>

Auto senza telepass

Due persone  
Stessa Azienda

Carnivori

24/2 spesa a  
CastelfidardoLei compra entro  
un'ora da lui



Event Sourcing?



Event Sourcing?

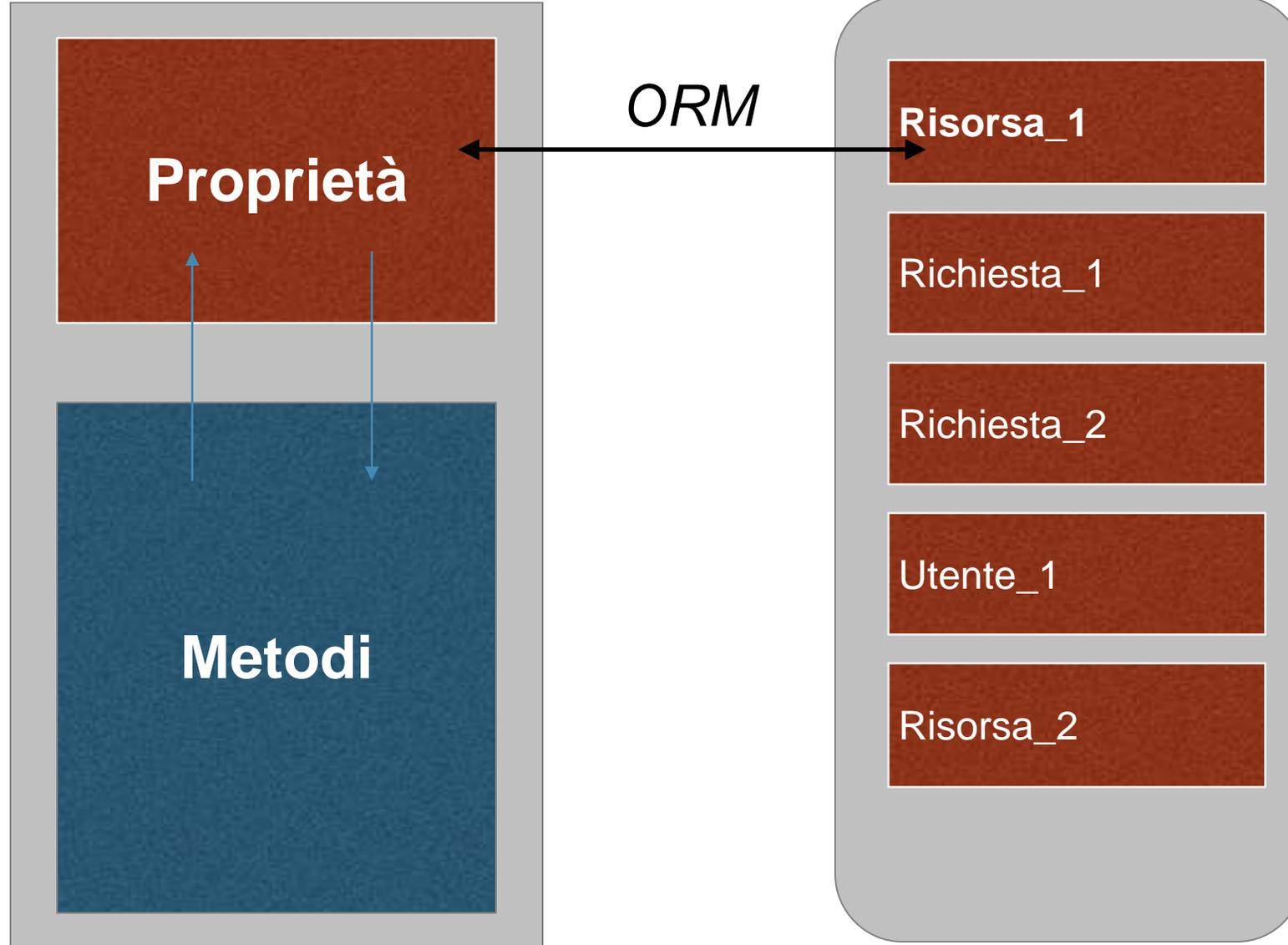
*ne ho sentito parlare...*

# Dominio



*Aggregato*

*DB*



ok.. **basta** una  
tabella di **log**



# basta



# basta

# duplicare tutte le strutture dati

# **basta**

duplicare tutte le strutture dati

tenere tutto sincronizzato

# basta

duplicare tutte le strutture dati

tenere tutto sincronizzato

versionare tutto

# basta

duplicare tutte le strutture dati

tenere tutto sincronizzato

versionare tutto

un bel flag “comando io”

# basta

duplicare tutte le strutture dati

tenere tutto sincronizzato

versionare tutto

un bel flag “comando io”

mapping! mapping! mapping! (cit)

basta uno **storage**  
ad **eventi**



# NEventStore

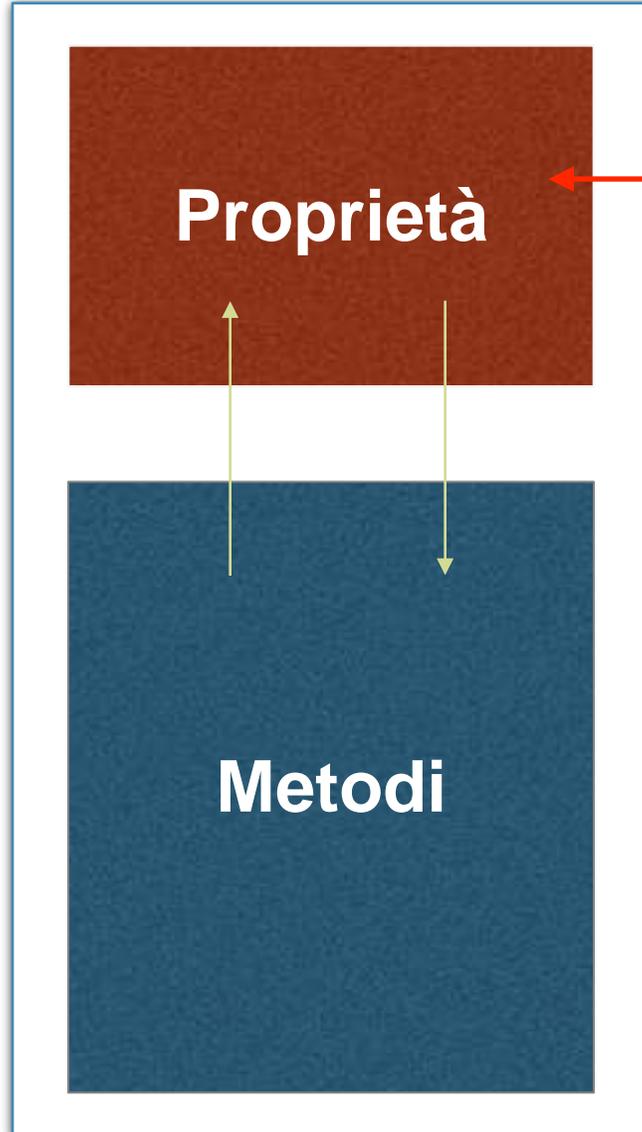
A persistence agnostic Event Store for .NET  
<http://neventstore.org/>

# CommonDomain

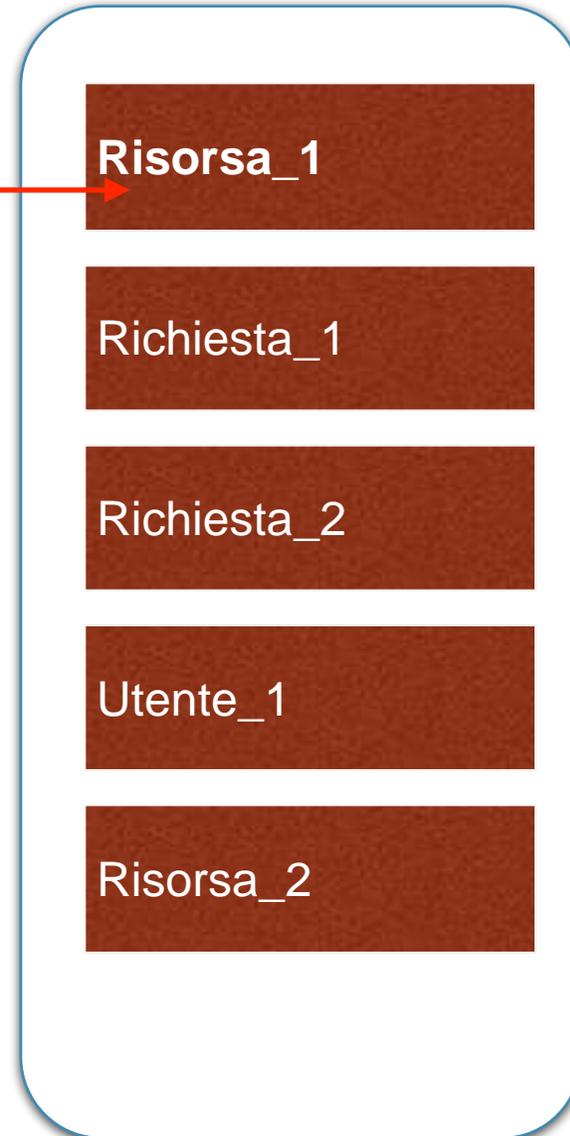
la "colla" tra DDD e NEventStore



# Aggregato



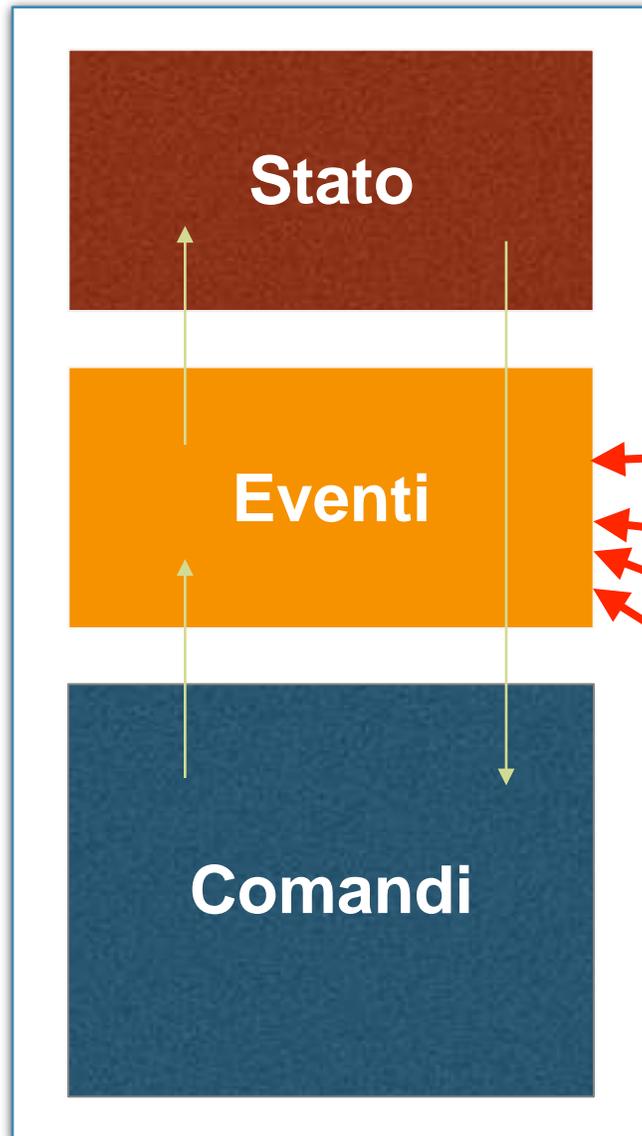
# DB



ORM



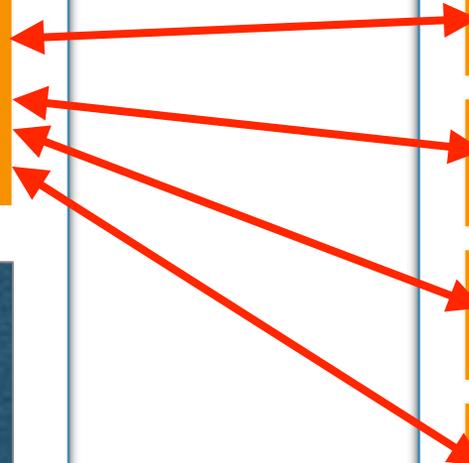
# Aggregato



# EventStore



*Common Domain*



# Evento

Chi

Quando

Dove

Cosa

Perché

# Evento

Chi

Headers: User\_1

Quando

Timestamp: 2014-02-25T15:45

Dove

Aggregato: Presentazione

Cosa

Slide Attiva: 1

Perché

PresentazioneIniziata

# Evento

Chi

Headers: User\_1

Quando

Timestamp: 2014-02-25T15:45

Dove

Aggregato: Presentazione

Cosa

Slide Attiva: 1

**Perché**

PresentazioneIniziata

```
A. var aggregate = repository.GetById<TAggregate>(id);
```

```
B. aggregate.Command(a,b,c);
```

```
C. repository.Save(aggregate, commitId, h => h.Add(k,v));
```

```
A. var aggregate = repository.GetByld<TAggregate>(id);  
B. aggregate.Command(a,b,c);  
C. repository.Save(aggregate, commitId, h => h.Add(k,v));
```

Store Commit



## Commit

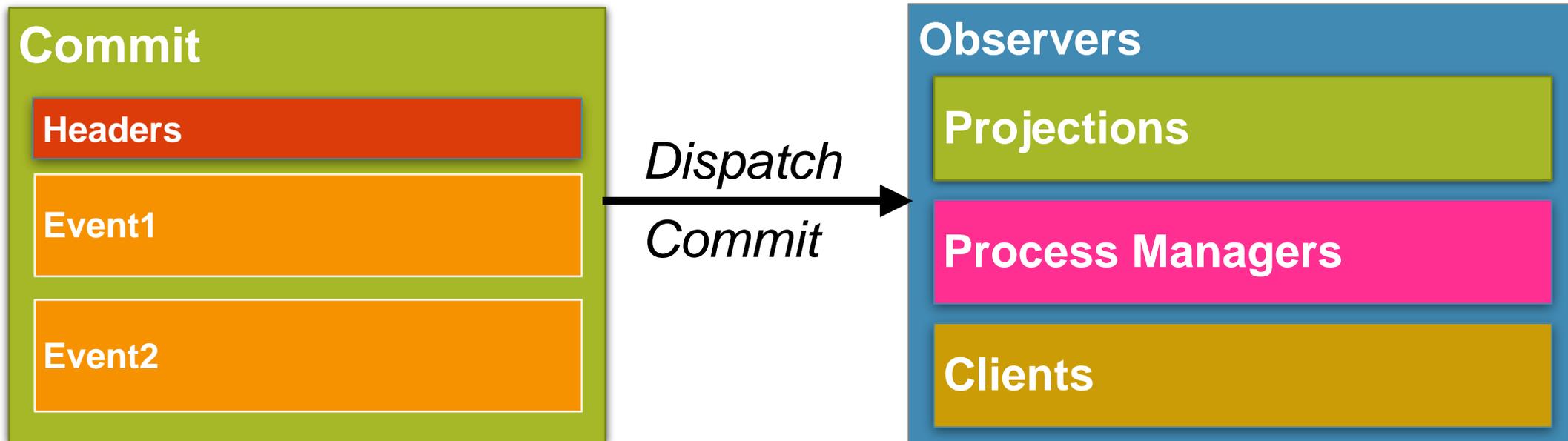
Headers

Event1

Event2

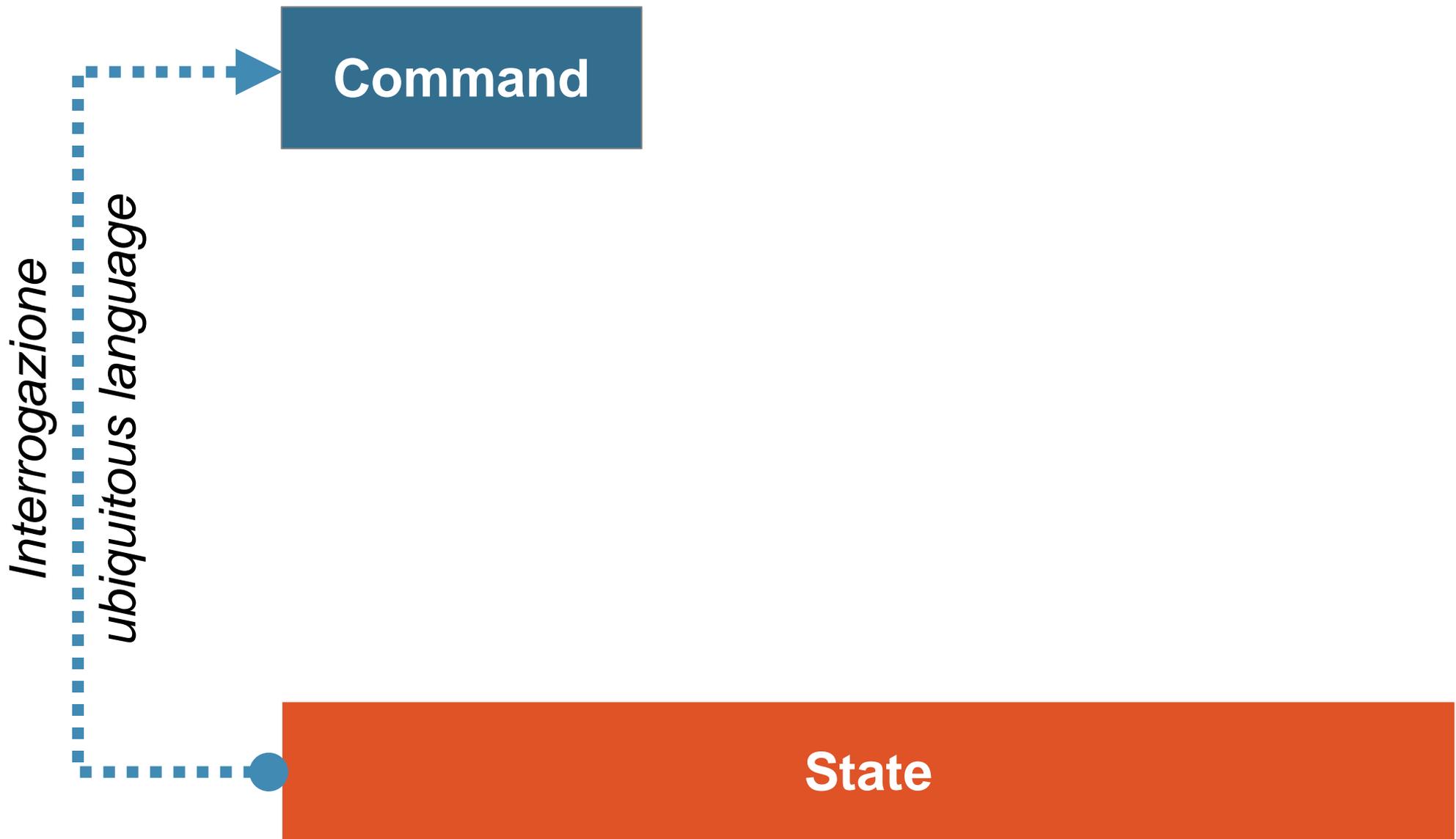
```
A. var aggregate = repository.GetByld<TAggregate>(id);  
B. aggregate.Command(a,b,c);  
C. repository.Save(aggregate, commitId, h => h.Add(k,v));
```

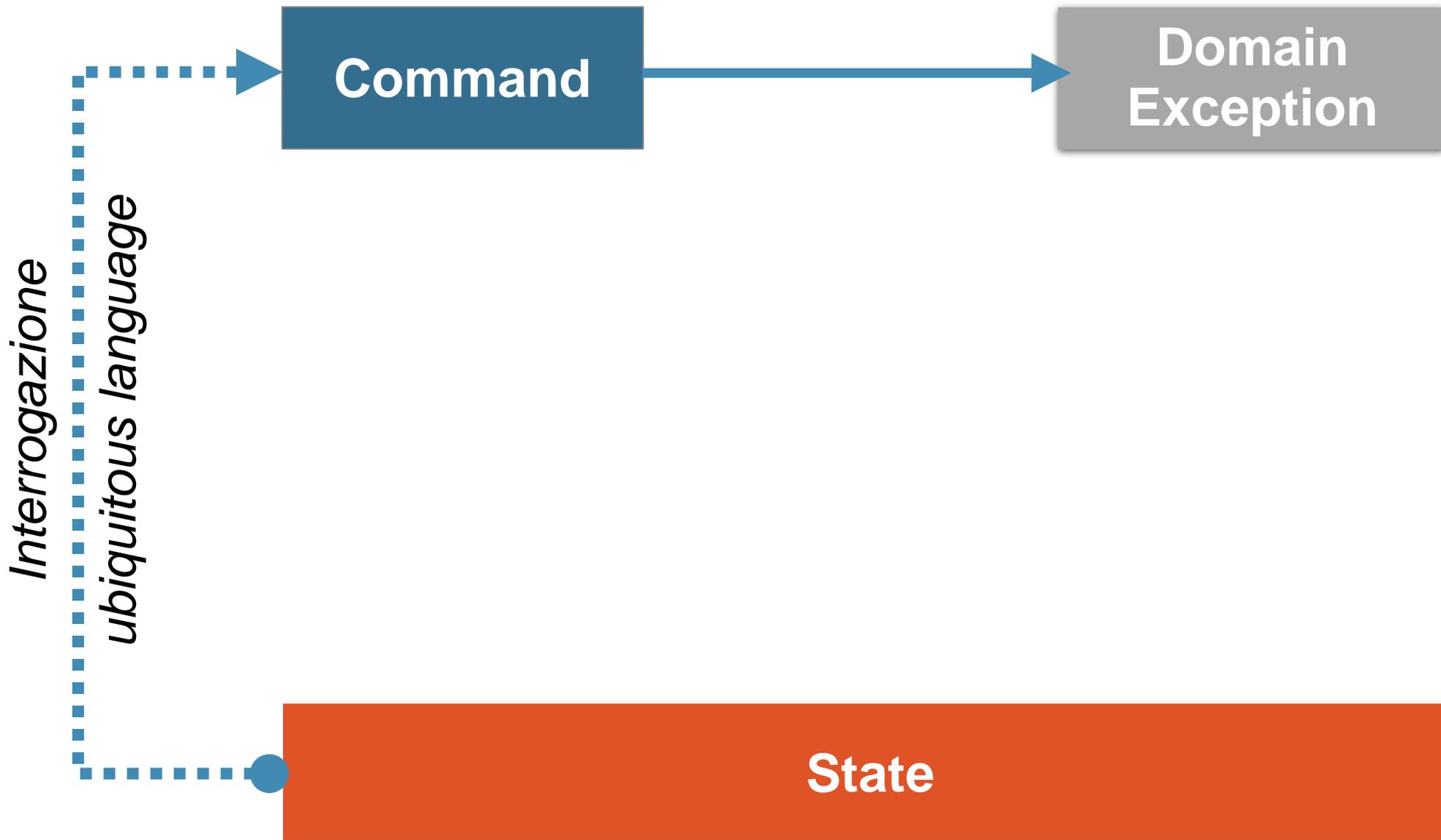
Store Commit

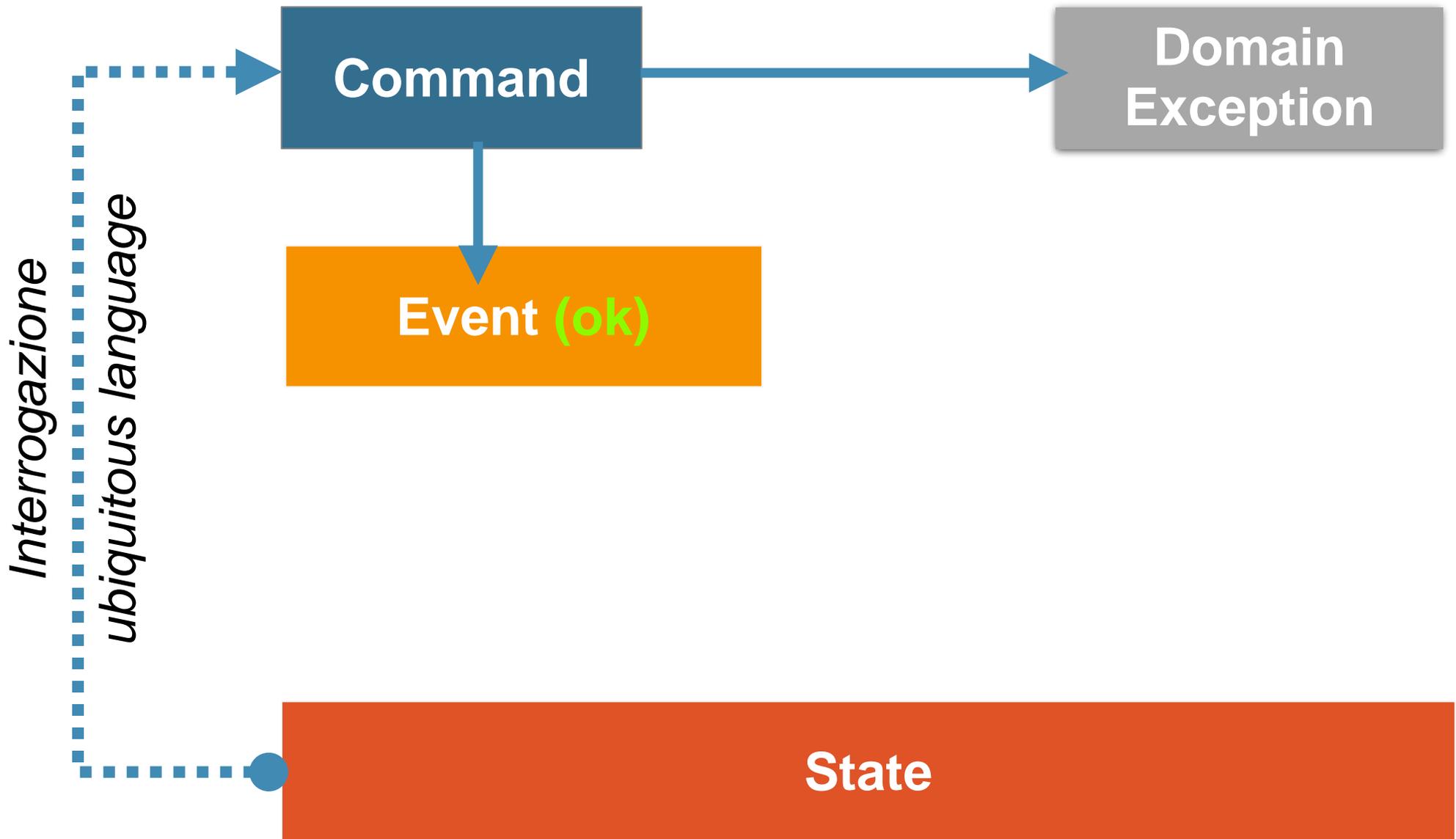


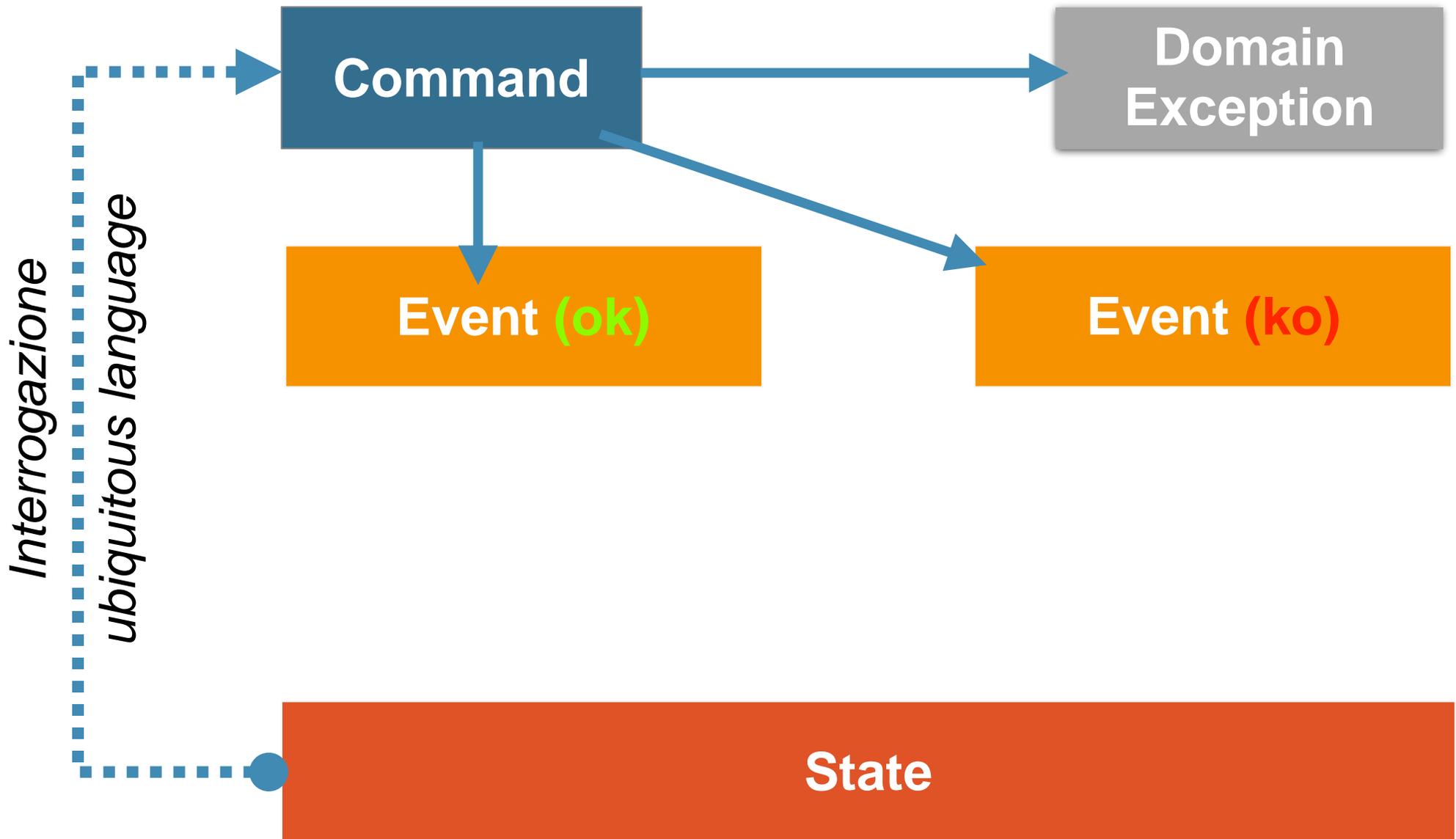
**Command**

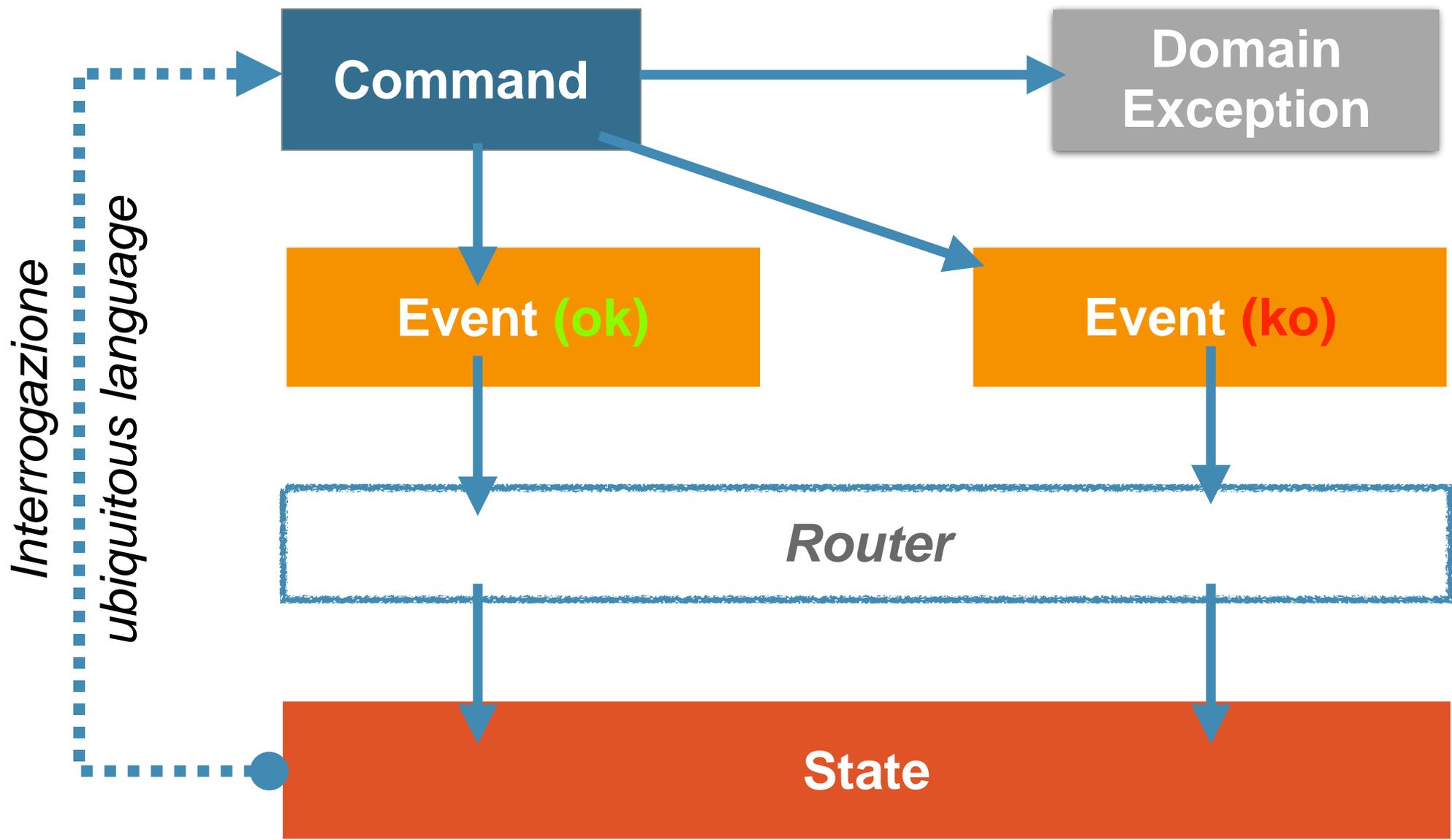
**State**











# Svantaggi



Quantità di codice  
dipende..



# Quantità di codice

## tanto

# Quantità di codice

**tanto**

**ma dannatamente semplice**

# Vantaggi



# Focus su struttura dati



# Focus su comportamento



# Focus su comportamento business value



# Decisioni prese su stato (stale)

# Decisioni prese su stato corrente



# Decisioni prese su stato corrente by design



# Costo di implementazione esponenziale



# Costo di implementazione lineare



# Costo di implementazione

lineare

eventi++

# Testing

## complesso



# Testing

## facile



# Testing

facile

stato -> comando -> eventi

# Collisioni

transazioni (tutto o niente)

# Collisioni

merge intelligenti / replay



# Collisioni

merge intelligenti / replay

by design



# Integrabilità

**batch / etl / ws ad hoc / trigger**

# Integrabilità

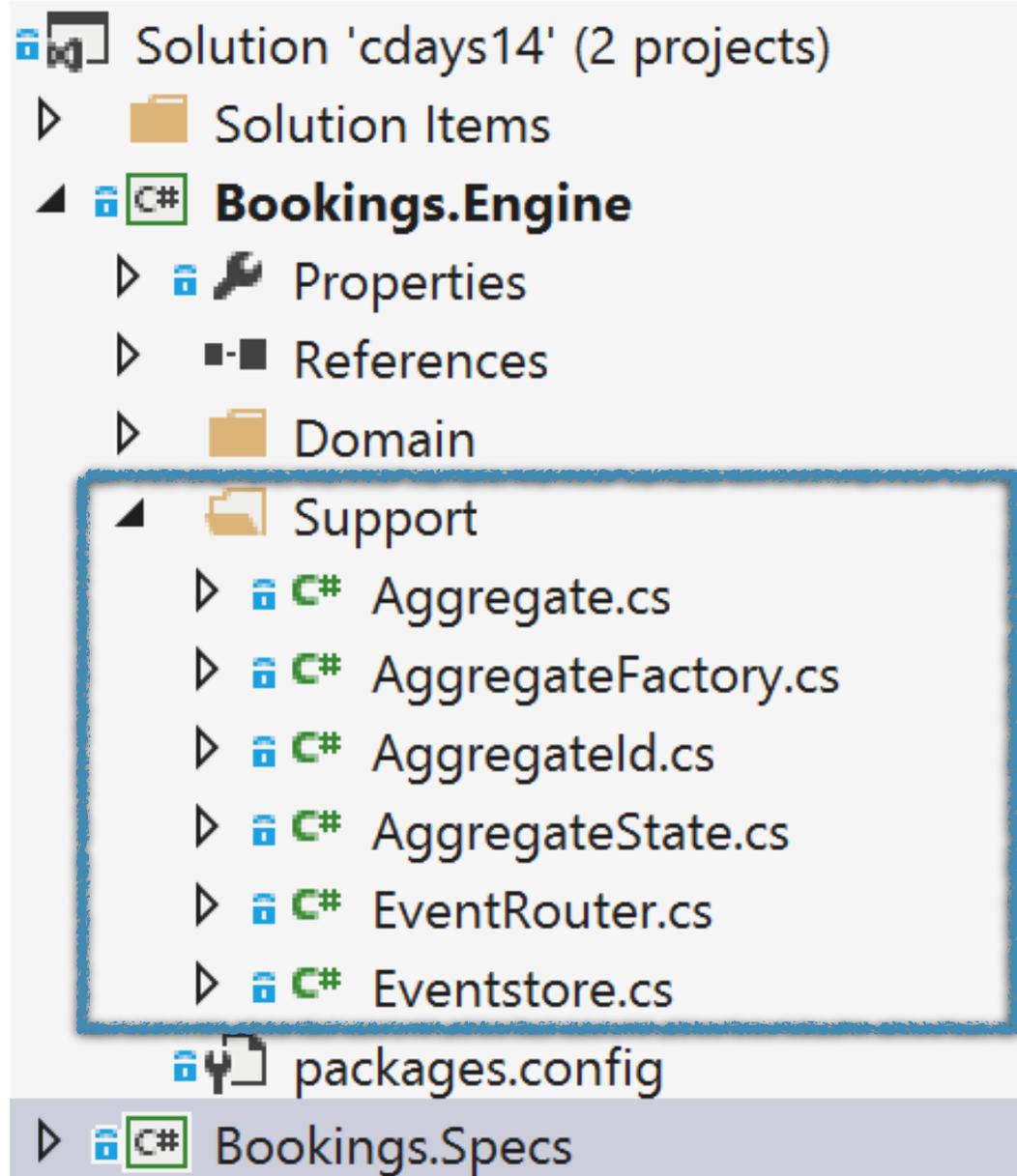
**evento -> subscriber**

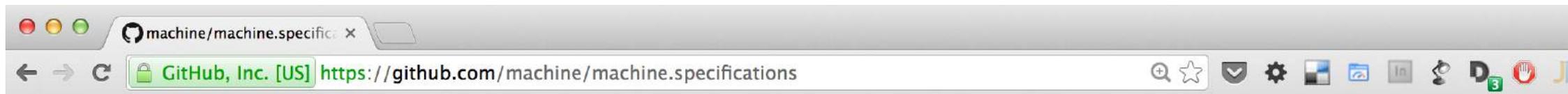
# Integrabilità

evento -> subscriber

push o pull







# Machine.Specifications

Machine.Specifications (MSpec) is a [context/specification](#) framework that removes language noise and simplifies tests. All it asks is that you accept the `= () =>`. Keep up with the [latest news and discussions](#) or follow the maintainers, [@agross](#), CTO of [GROSSWEBER](#) and [@danielmarbach](#).



```
[Subject(typeof(AuthenticationService), "Authentication")]
```

0 references

```
public class When_authenticating_a_user_with_valid_credentials
{
    static AuthenticationService Service;
    static AuthenticationService.Token Token;

    Establish context = () =>
        Service = new AuthenticationService();

    Because of = () =>
        Token = Service.Authenticate("user", "password");

    It authentication_token_should_be_valid = () =>
        Token.IsValid.ShouldBeTrue();

    It authentication_token_should_have_the_user_name_set = () =>
        Token.Username.ShouldBeLike("user");

    Cleanup after = () =>
        Service.Dispose();
}
```

```
[Subject(typeof(AuthenticationService), "Authentication")]
```

0 references

```
public class When_authenticating_a_user_with_valid_credentials
```

```
{
```

```
    static AuthenticationService Service;
```

```
    static AuthenticationService.Token Token;
```

```
    Establish context = () =>
```

```
        Service = new AuthenticationService();
```

```
    Because of = () =>
```

```
        Token = Service.Authenticate("user", "password");
```

```
    It authentication_token_should_be_valid = () =>
```

```
        Token.IsValid.ShouldBeTrue();
```

```
    It authentication_token_should_have_the_user_name_set = () =>
```

```
        Token.Username.ShouldBeLike("user");
```

```
    Cleanup after = () =>
```

```
        Service.Dispose();
```

```
}
```

```
[Subject(typeof(AuthenticationService), "Authentication")]
```

0 references

```
public class When_authenticating_a_user_with_valid_credentials
```

```
{
```

```
    static AuthenticationService Service;
```

```
    static AuthenticationService.Token Token;
```

```
    Establish context = () =>
```

```
        Service = new AuthenticationService();
```

```
    Because of = () =>
```

```
        Token = Service.Authenticate("user", "password");
```

```
    It authentication_token_should_be_valid = () =>
```

```
        Token.IsValid.ShouldBeTrue();
```

```
    It authentication_token_should_have_the_user_name_set = () =>
```

```
        Token.Username.ShouldBeLike("user");
```

```
    Cleanup after = () =>
```

```
        Service.Dispose();
```

```
}
```

```
[Subject(typeof(AuthenticationService), "Authentication")]
```

0 references

```
public class When_authenticating_a_user_with_valid_credentials
```

```
{
```

```
    static AuthenticationService Service;
```

```
    static AuthenticationService.Token Token;
```

```
    Establish context = () =>  
        Service = new AuthenticationService();
```

```
    Because of = () =>  
        Token = Service.Authenticate("user", "password");
```

```
    It authentication_token_should_be_valid = () =>  
        Token.IsValid.ShouldBeTrue();
```

```
    It authentication_token_should_have_the_user_name_set = () =>  
        Token.Username.ShouldBeLike("user");
```

```
    Cleanup after = () =>  
        Service.Dispose();
```

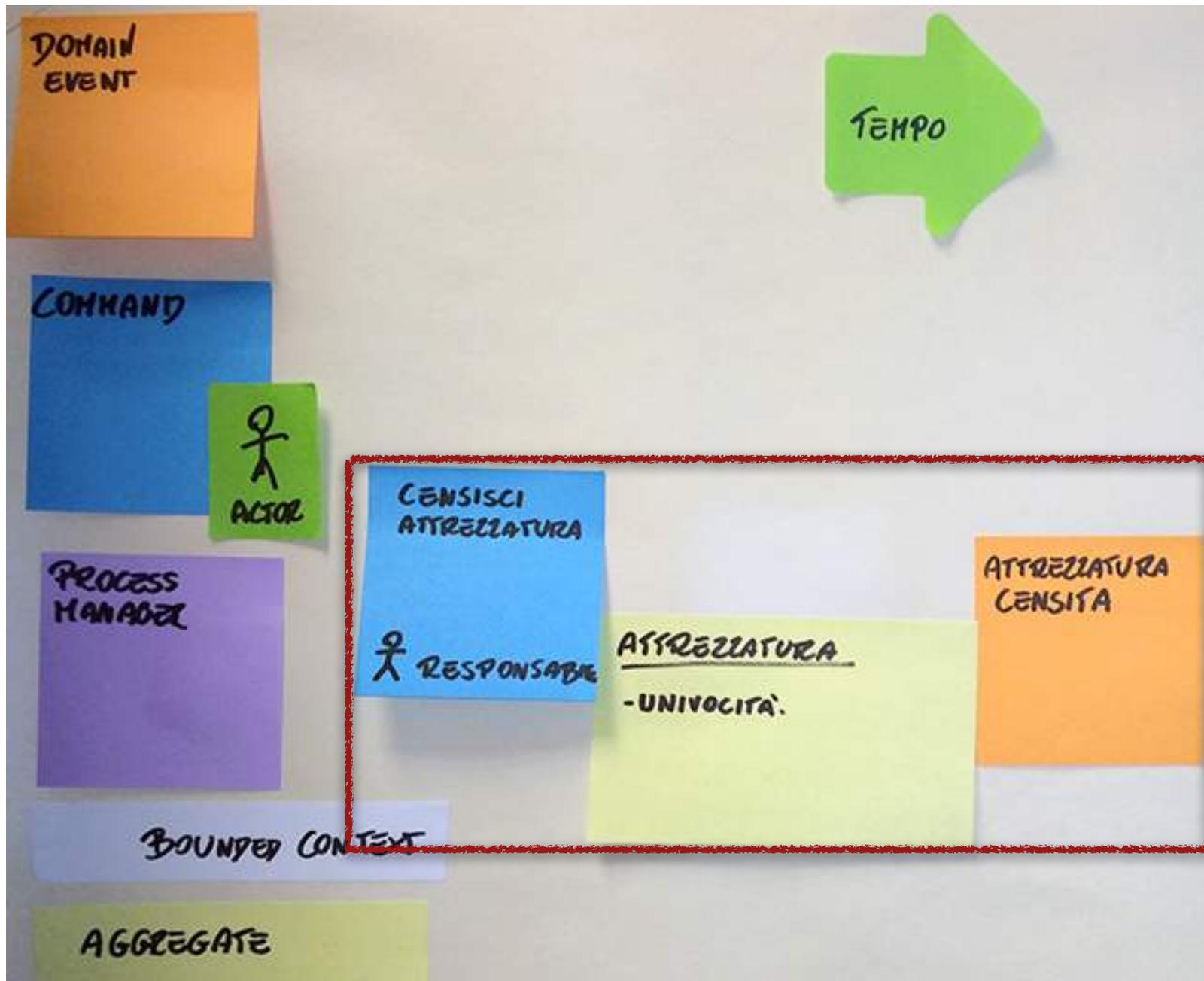
```
}
```

# Nei test runner diventa

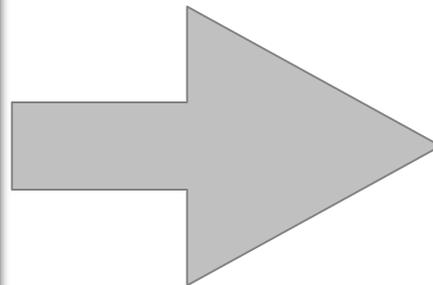
```
[-] ✓ ◊ Bookings.Specs.Specs (2 tests) Success
  [-] ✓ AuthenticationService Authentication, When authenticating a user with valid credentials (... Success
    [-] ✓ authentication token should be valid Success
    [-] ✓ authentication token should have the user name set Success
```

# Join the dots





**Register  
Resource**



**Resource  
Registered**

```
private Establish context = () =>
{
    var state = new ResourceState()
    {
        Id = new ResourceId(),
        Name = new ResourceName("Surface Pro"),
        Managers = new HashSet<UserId>(new[] { _managerId })
    };

    state.Reservations.Add(
        new ResourceState.Reservation(
            new BookingRequestId(),
            new BookingInterval(
                DateTime.Parse("2014-01-03"),
                DateTime.Parse("2014-01-07")
            )
        )
    );
    Setup(state);
};
```

Because of = () =>

```
Resource.AddReservation(_requestId, _managerId, Interval);
```

```
private static readonly BookingRequestId _requestId = new BookingRequestId();
```

```
private static readonly BookingInterval Interval = new BookingInterval(  
    DateTime.Parse("2014-01-01"),  
    DateTime.Parse("2014-01-10")  
);
```



```
It resource_reservation_failed_event_should_be_raised = () =>
    LastRaisedEventOfType<ResourceReservationFailed>().ShouldNotBeNull();

It resource_reservation_failed_event_should_contain_the_request_id = () =>
    LastRaisedEventOfType<ResourceReservationFailed>().RequestId.ShouldBeLike(_requestId);

It invariants_should_be_satisfied = () =>
    Resource.EnsureAllInvariants();
```



```
8 namespace Bookings.Engine.Domain.Bookings.Resource
9 {
10     public class ResourceState : AggregateState<ResourceId>
11     {
12         stato
13
14         public ResourceName Name { get; protected internal set; }
15         public HashSet<UserId> Managers { get; protected internal set; }
16         public IList<Reservation> Reservations { get; private set; }
17
18         Handler eventi
19
20         public void On(ResourceRegistered e) {...}
21
22         public void On(ResourceManagerAdded e) {...}
23
24         public void On(ResourceReserved e) {...}
25
26         public void On(ResourceReservationFailed e) {...}
27
28         public bool IsManager(UserId userId) {...}
29
30         Interrogazioni allo stato in UL
31
32         public bool IsResourceAvailable(BookingInterval interval) {...}
33     }
34 }
```

```
public class ResourceAggregate : Aggregate<ResourceState, ResourceId>
{
    public ResourceAggregate() : base (new ResourceState())...

    public void Register(ResourceId id, ResourceName name, UserId managerId)...

    public void AddReservation(
        BookingRequestId requestId,
        UserId approvedByUserId,
        BookingInterval interval
    )
    {
        if (!State.IsManager(approvedByUserId))
        {
            throw new DomainException("Reservation not allowed");
        }

        if (State.IsResourceAvailable(interval))
        {
            RaiseEvent(new ResourceReserved(requestId, approvedByUserId, interval));
        }
        else
        {
            RaiseEvent(new ResourceReservationFailed(requestId, "Unavailable"));
        }
    }
}
```

```
public class ResourceState : AggregateState<ResourceId>
{
    public class Reservation[...]

    public ResourceName Name { get; protected internal set; }
    public HashSet<UserId> Managers { get; protected internal set; }
    public IList<Reservation> Reservations { get; private set; }

    public ResourceState()[...]
```

```
public void On(ResourceRegistered e)
{
    Id = e.ResourceId;
    Name = e.Name;
}
```

```
public void On(ResourceManagerAdded e)
{
    Managers.Add(e.ManagerId);
}
```

```
public class ResourceState : AggregateState<ResourceId>
{
    demo
    public override void EnsureAllInvariants()
    {
        base.EnsureAllInvariants();

        if (this.Name == null)
            throw new DomainException("Resource name not set");

        if (!this.Managers.Any())
            throw new DomainException(string.Format("Resource {0} needs a manager", this.Name));
    }

    public bool HasBeenRegistered ...
}
```





44 ✓ 44 - 0 ✖ 0 ? 0

[-] ✓  <Bookings.Specs> (44 tests)	Success
[-] ✓  Bookings.Specs.Evenstore (3 tests)	Success
[+] ✓ Store, StoreSpecs (3 tests)	Success
[-] ✓  Bookings.Specs.Specs (2 tests)	Success
[+] ✓ AuthenticationService Authentication, When authenticating a user with valid credenti...	Success
[-] ✓  Bookings.Specs.Specs.BookingRequestSpecs (9 tests)	Success
[+] ✓ Given a new BookingRequest, when is created (7 tests)	Success
[+] ✓ Given an unauthorized BookingRequest, when is authorized (2 tests)	Success
[-] ✓  Bookings.Specs.Specs.ResourceSpecs (17 tests)	Success
[+] ✓ Given a resource with bookings, when a reservation is made for an interval already re...	Success
[+] ✓ Given a resource without bookings, when is booked (6 tests)	Success
[+] ✓ Given new Resource, when is created (7 tests)	Success
[-] ✓  Bookings.Specs.Unit (13 tests)	Success
[+] ✓ BookingIntervalTests (8 tests)	Success
[+] ✓ ResourceNameTests (5 tests)	Success

Total coverage: 86%  

Symbol	Coverage (%)	Uncovered/Total Stmts.
<input type="checkbox"/> Bookings.Engine	<span style="background-color: green; color: white; padding: 2px;">80%</span>	45/222
<input type="checkbox"/> Bookings.Engine.Support	<span style="background-color: green; color: white; padding: 2px;">70%</span>	18/60
<input type="checkbox"/> Bookings.Engine.Domain.Bookings.Resource	<span style="background-color: green; color: white; padding: 2px;">75%</span>	24/97
<input type="checkbox"/> Bookings.Engine.Domain.Bookings.BookingRequest	<span style="background-color: green; color: white; padding: 2px;">91%</span>	3/35
<input type="checkbox"/> Bookings.Engine.Domain	<span style="background-color: green; color: white; padding: 2px;">100%</span>	0/3
<input type="checkbox"/> Bookings.Engine.Domain.Bookings.BookingRequest.Events	<span style="background-color: green; color: white; padding: 2px;">100%</span>	0/7
<input type="checkbox"/> Bookings.Engine.Domain.Bookings.Resource.Events	<span style="background-color: green; color: white; padding: 2px;">100%</span>	0/20
<input type="checkbox"/> Bookings.Specs	<span style="background-color: green; color: white; padding: 2px;">93%</span>	13/198

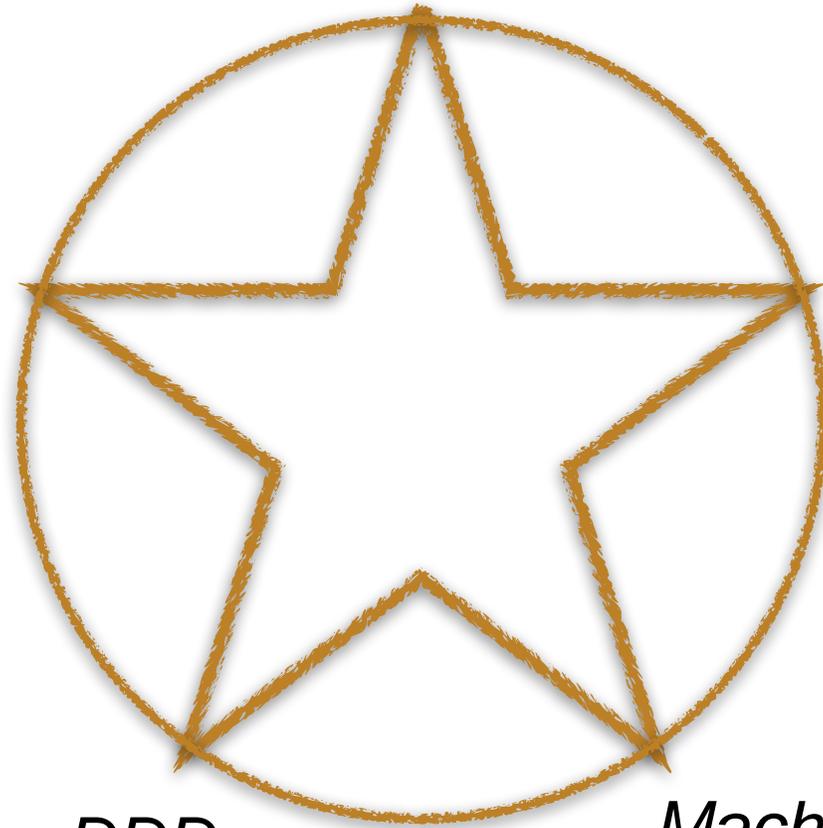
# *Ricapitolando*



*Event Storming*

*NEventStore*

*Common  
Domain*



*DDD*

*Machine  
Specifications*



*<http://github.com/andreabalducci/cdays14>*

# Elogio alla community

“...gli user group sono stati lo strumento che maggiormente mi hanno permesso di imparare sia come scrivere software in modo nettamente migliore, sia come affrontare le altre componenti relative alla gestione di un prodotto o di un progetto. Ciò è stato possibile grazie al dialogo e al confronto con persone intelligenti e disponibili a scambiare gratuitamente le proprie idee, opinioni ed esperienze, col coraggio di riportare anche quelle fallimentari, invece di chiudersi nel proprio orticello cercando di nascondere i propri segreti. Questo è lo scambio culturale, che porta ad un’effettiva crescita, a cui tutti coloro che svolgono la nostra professione farebbero bene a partecipare e contribuire.”

*Elogio alla community - @ste8*

Conclusioni

# Emergenza rapida...

- ... della complessità
- del problema business
- dei conflitti

Troppa complessità



EventStorming non è la  
pillola per diventare  
intelligenti

EventStorming ci  
permette di scoprire  
**dove** abbiamo **torto**  
...il prima possibile

# rapporto costi/benefici

- Alcuni domini chiamano a gran voce EventSourcing
- ...su altri non è che ci sia poi tutta questa complessità.
- ma...



**CRUD**

Assumo **ORA** di non avere  
bisogno della storia del dato  
in *futuro*



**Quanto costa la colla?**

***Ignorance is the single  
greatest impediment  
to throughput.***

**Dan North**

# Customer Satisfaction

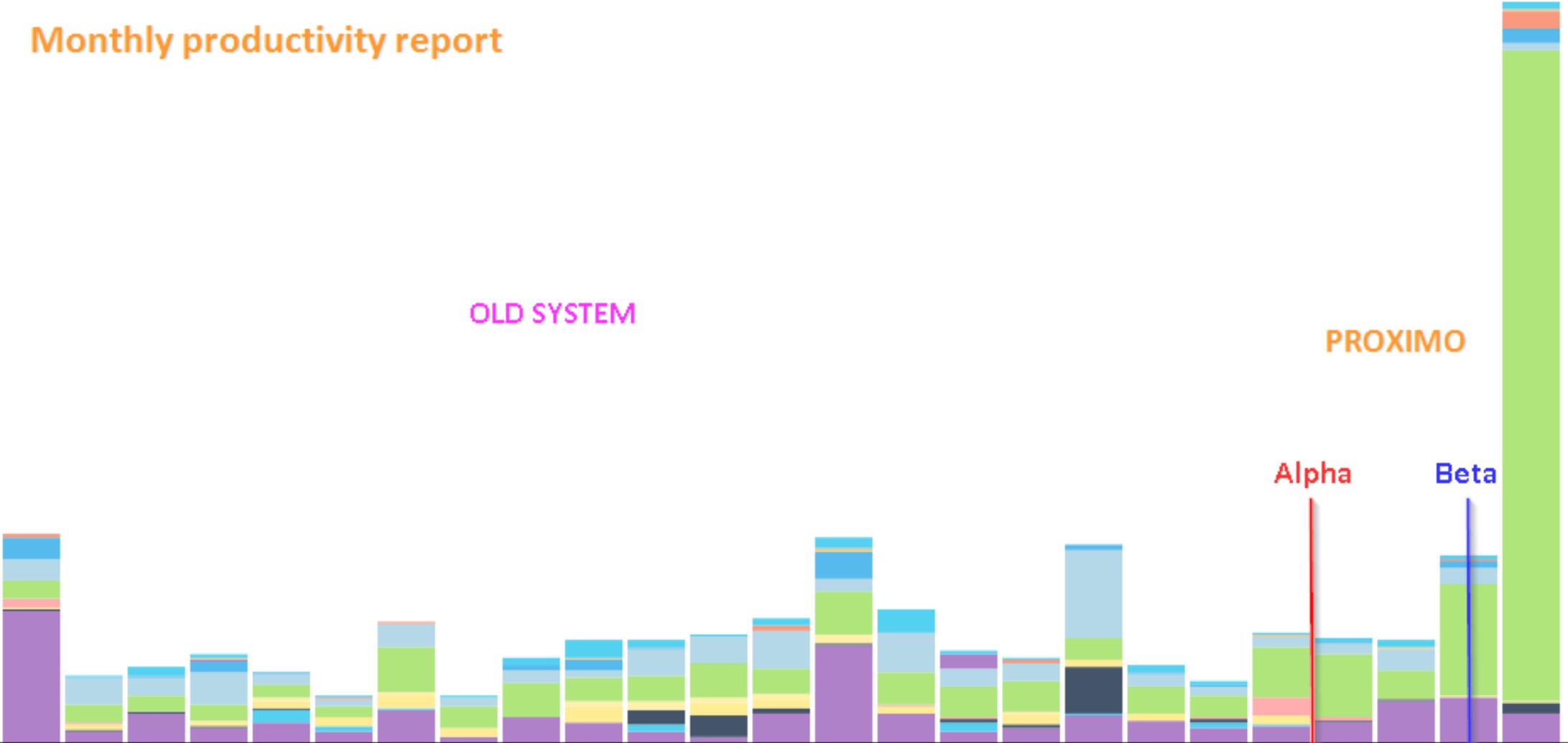
Monthly productivity report

OLD SYSTEM

PROXIMO

Alpha

Beta



# Domain Events

# Domain Events

- Lo-Fi —> Gli esperti *ci correggono*

# Domain Events

- Lo-Fi —> Gli esperti *ci correggono*
- Transizioni di stato —> Maggior precisione e ricchezza del linguaggio

# Domain Events

- Lo-Fi —> Gli esperti *ci correggono*
- Transizioni di stato —> Maggior precisione e ricchezza del linguaggio
- Non ragiono più in tabelle. —> non costringo gli utenti ad adattarsi.

# Domain Events

- Lo-Fi —> Gli esperti *ci correggono*
- Transizioni di stato —> Maggior precisione e ricchezza del linguaggio
- Non ragiono più in tabelle. —> non costringo gli utenti ad adattarsi.
- Faccio le domande giuste



UX



# Scenari classici

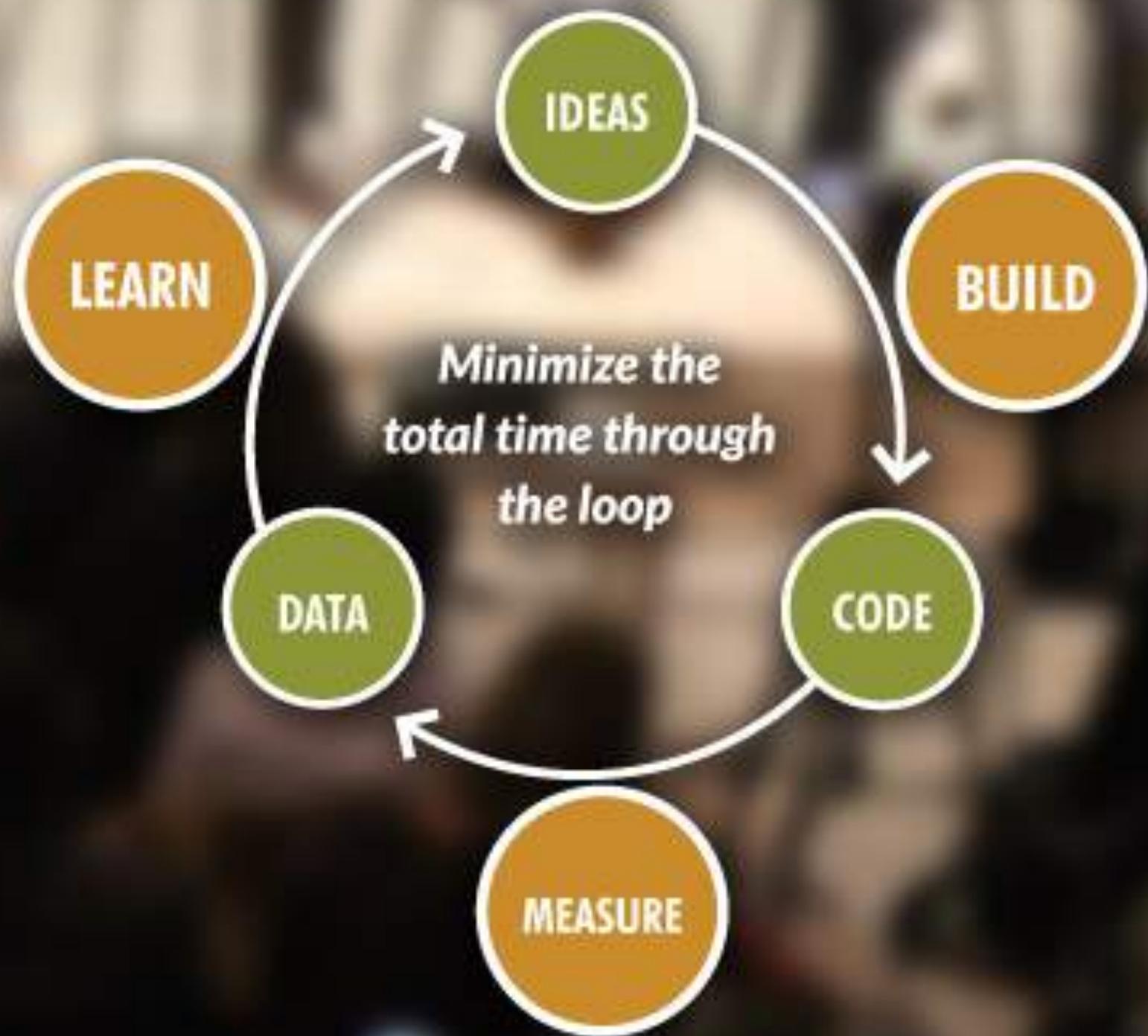
- Analisi redditività
- Fraud detection
- Analisi temporali
- Predittività e correlazioni (scenario walmart)
- ...



gli esperti ci aiutano  
a capire

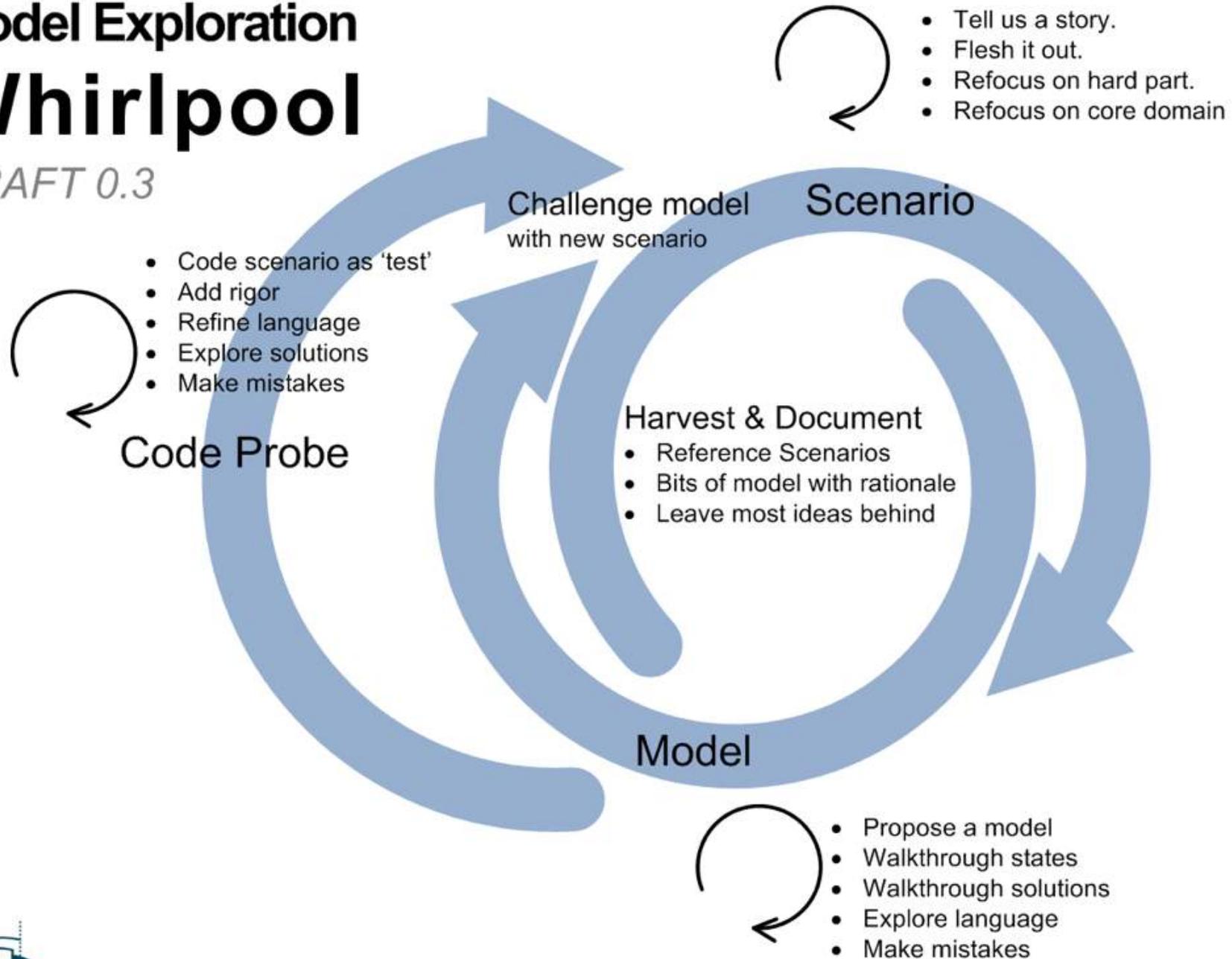
e noi aiutiamo loro

Cicli corti



# Model Exploration Whirlpool

DRAFT 0.3



# Per saperne di più...

- <http://ziobrando.blogspot.it/2013/11/introducing-event-storming.html>
- <https://it.groups.yahoo.com/neo/groups/DDD-IT/>



19 marzo a Roma / 2 aprile a Bologna

Grazie!

# Q&A

Tutto il materiale di questa sessione su  
<http://www.communitydays.it/>

Lascia il feedback su questa sessione,  
potrai essere estratto per i nostri premi!

Seguici su

Twitter @CommunityDaysIT

Facebook <http://facebook.com/cdaysit>

#CDays14

