Event Aggregator

Design Pattern

pluralsight
see what you can learn

Motivating Example

- **An order entry system**

- **Multiple screens**
    - Order, Order Details, Order History, Shipping

- **Need to be notified**
    - When an order is selected
    - When an order is created
    - When an order update is cancelled
    - When an order is saved
    - When an order is deleted

Intent

Event
Aggregator

- **Simplify event registration by providing a single centralized store**

- **Reduce coupling between publishers and subscribers**

- **Reduce friction for introducing new events**

- **Reduce memory management issues related to eventing**
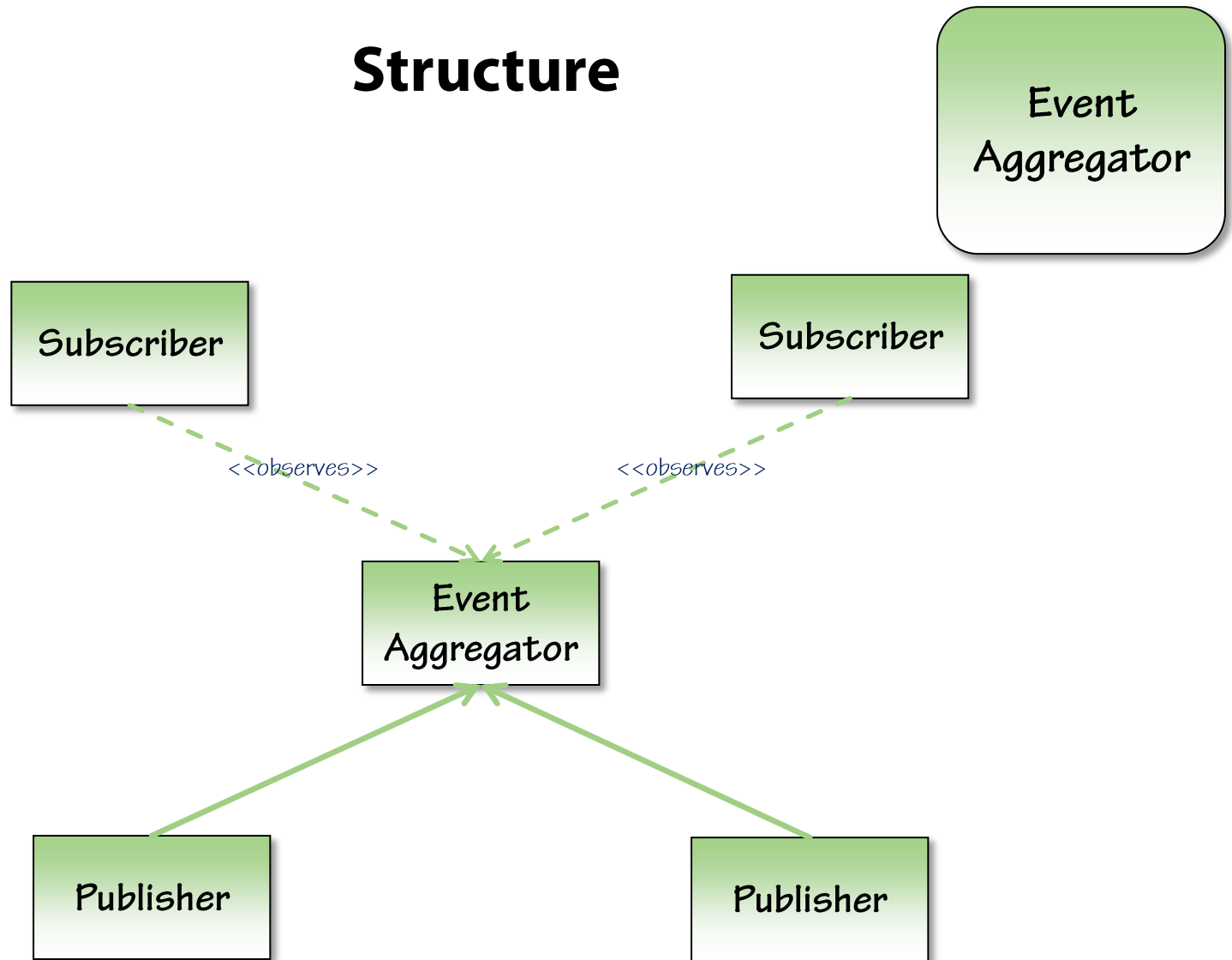
Applicability

Event
Aggregator

- **You are building a composite application**

- **You have complex screens**

- **You have many publishers and subscribers**

- **You have many events**

- **New events are added frequently**

- **Static events are a red flag**

Issues with the current implementation

Event
Aggregator

- Publishers and Subscribers are tightly coupled

- Introducing a new events / subscribers is difficult

- Several implicit memory management issues

Structure

Event
Aggregator

Subscriber

Subscriber

<<observes>>

<<observes>>

Event
Aggregator

Publisher

Publisher

pluralsight
see what you can learn

How It Gets Used

Event Aggregator

- **Publishers and Subscribers each hold a reference to the aggregator.**

- **Publishers call publication methods to notify subscribers.**

- **Subscribers call subscription methods to receive notifications**

Consequences

Event Aggregator

- **Publishers and subscribers are completely decoupled**

- **New events can more easily be introduced in the system (Open/Closed principle)**

- **Memory leaks due to eventing are reduced**

Known Uses / Variations

Event
Aggregator

- **StoryTeller - http://storyteller.tigris.org/**

- **MVVM Light – mvvmlight.codeplex.com**

- **Prism – microsoft.com/prism**

Related Patterns

- **Pub/Sub – http://en.wikipedia.org/wiki/Publish/subscribe**

- **Observer – http://en.wikipedia.org/wiki/Observer_pattern**

Summary

Event Aggregator

- **EventAggregator is useful for coordinating notifications in complex systems**

- **It reduces coupling, improves maintainability and testability**

- **It may be overkill for simple systems**

More information

Event
Aggregator

- http://martinfowler.com/eaaDev/EventAggregator.html

- http://codebetter.com/blogs/jeremy.miller/archive/2009/07/23/how-i-m-using-the-event-aggregator-pattern-in-storyteller.aspx

- http://msdn.microsoft.com/en-us/library/ff921122(v=PandP.40).aspx
[see Event Aggregation topic]