# Integrating Services with an Enterprise Service Bus

Sébastien **Mosser**,
Lecture #2, 24.09.2018

pictures: sxc.hu

CC BY NC

# Previously

Integration is **not** about

**designing beautiful** systems.

It is about **dealing with**

**existing** and **crappy** ones.

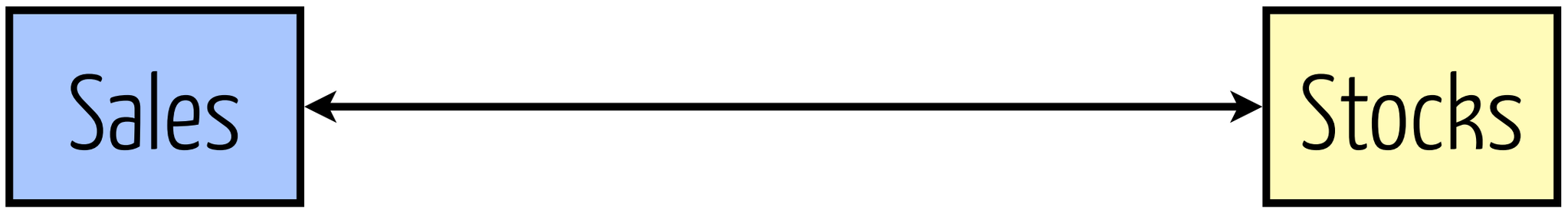
# Service Interface Styles

| | |
|---|---|
| **Remote Procedure Call** | How can clients **execute remote procedures**? |
| **Message** | How can clients interact with remote systems while **avoiding direct coupling** to remote procedure? |
| **Resource** | How can clients manipulate remote data while **minimizing** the need for **a domain-specific API**? |

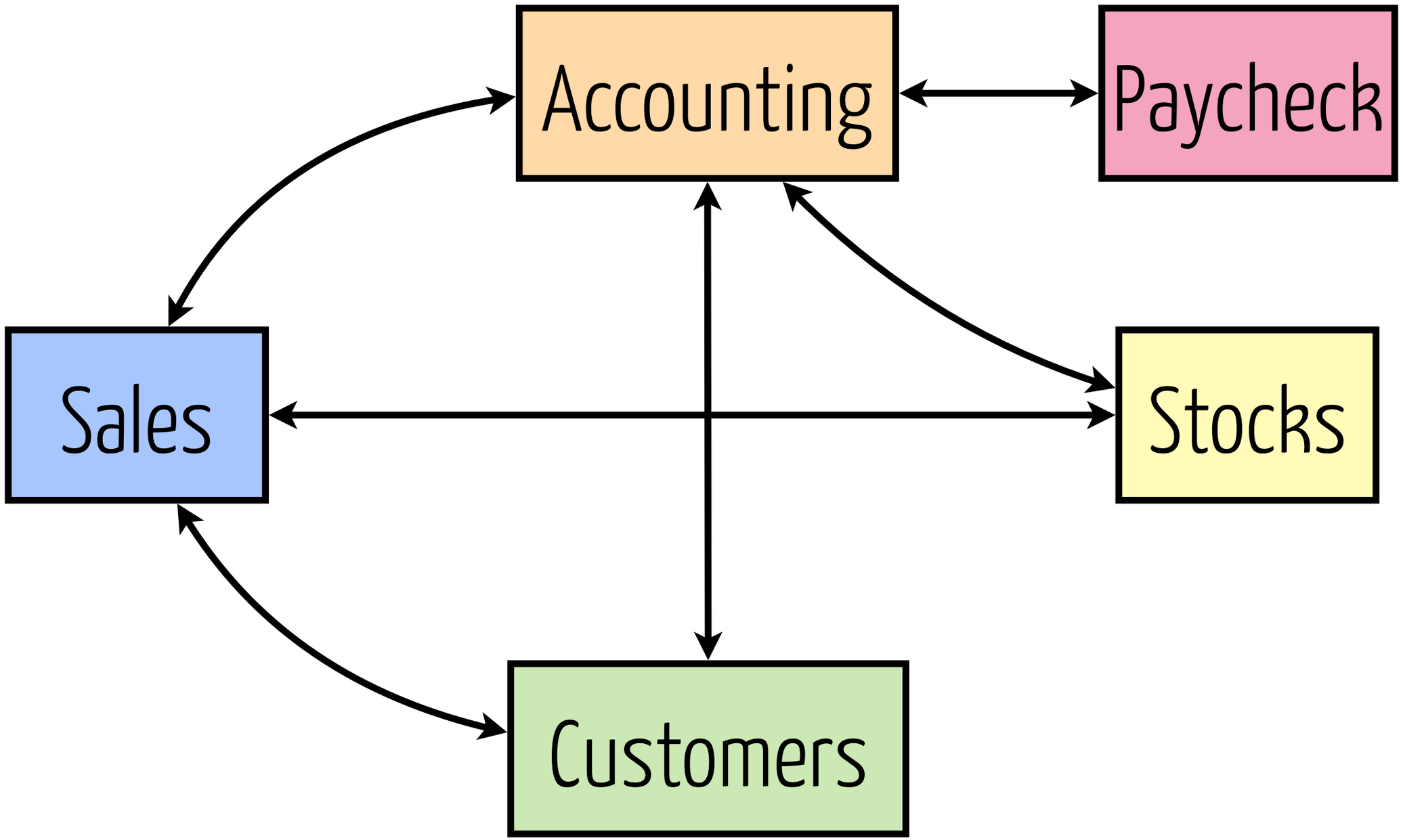
# Integration through Ages

(more than 20 years ago)

# **Point-to-point** Integration [1/2]



# **Accidental** Complexity! [2/2]

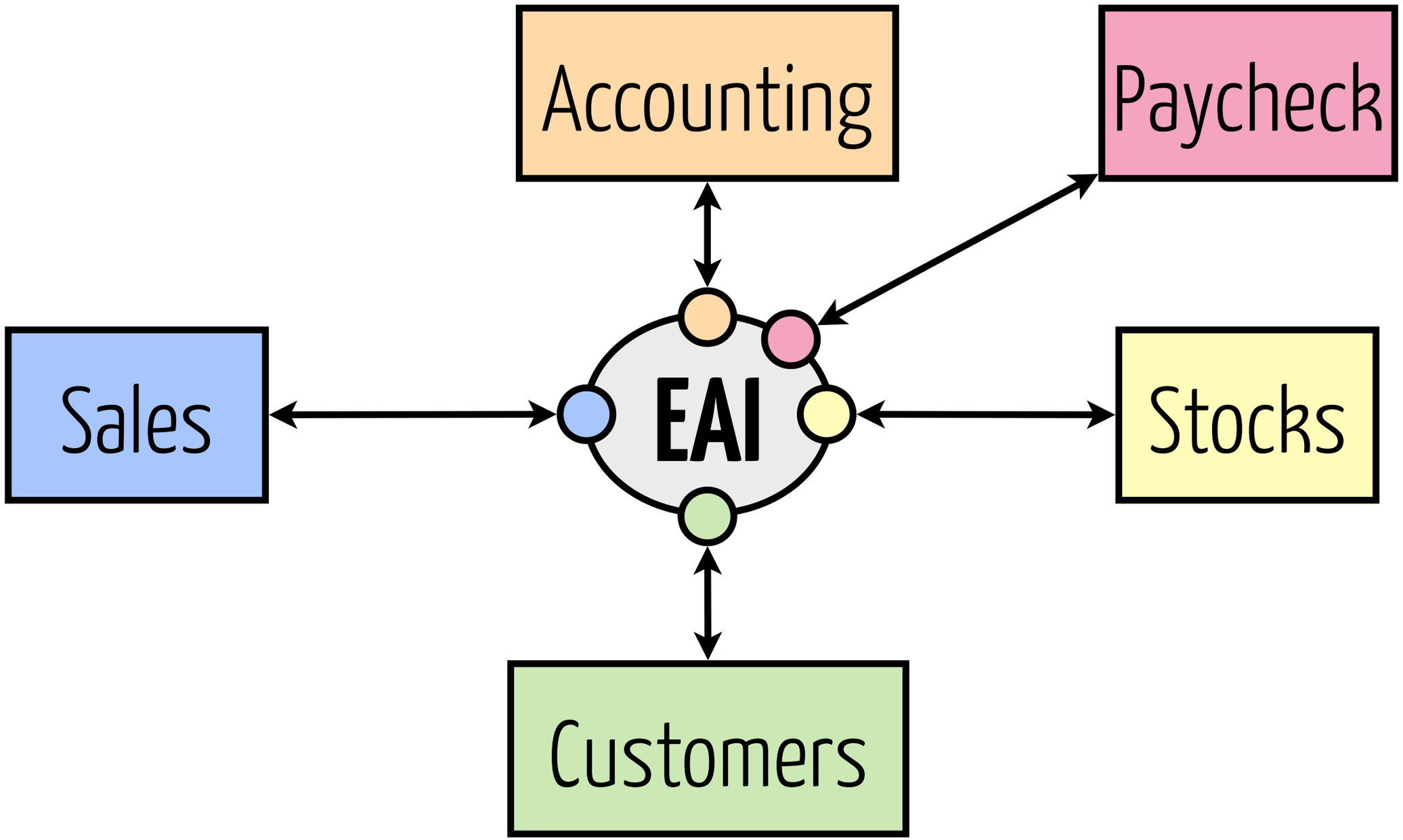



[petals]

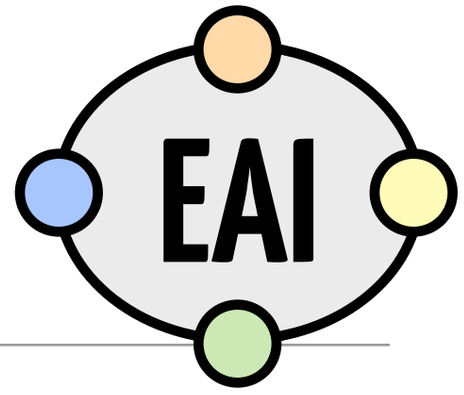
90's

# Enterprise Architecture Integration

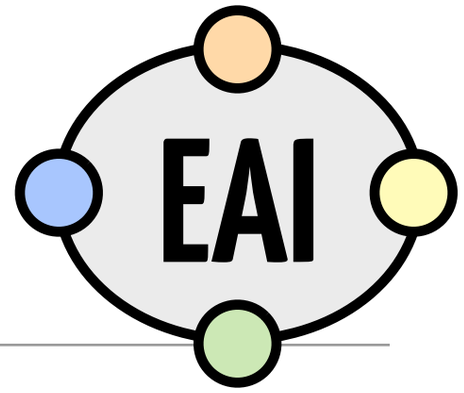
# EAI approach



# **Opening** the Box



# Opening the Box




Connexion

Transport

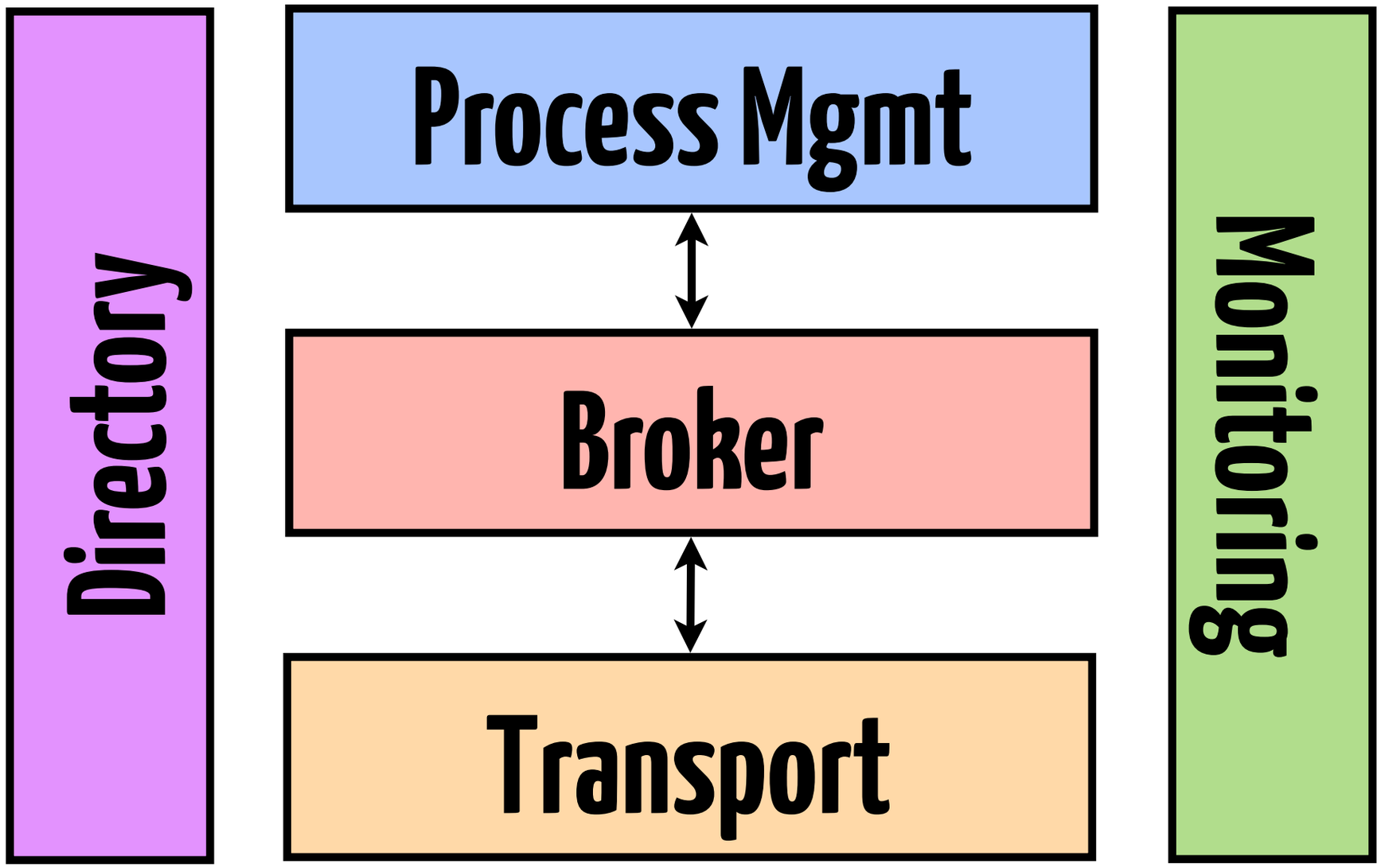
Data Handling

Business Logic

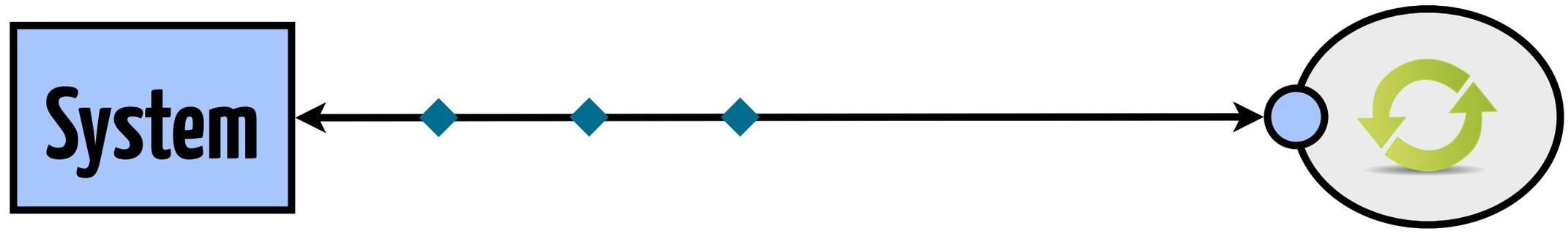
Supervision

Registry

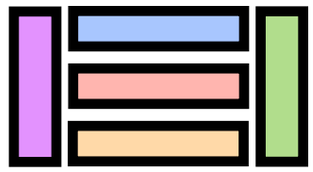
# System **Architecture**

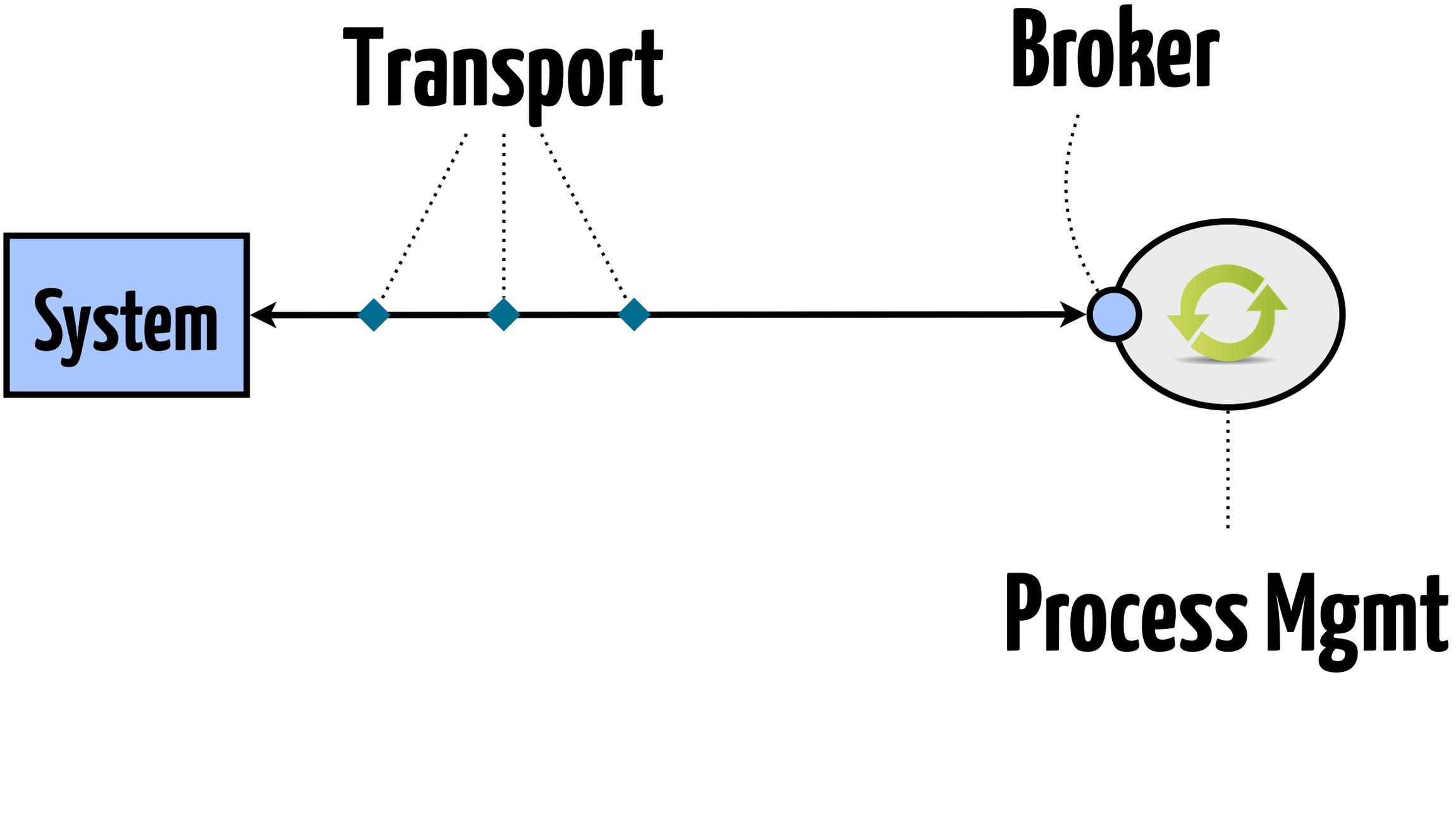


# **Instantiating** the Architecture

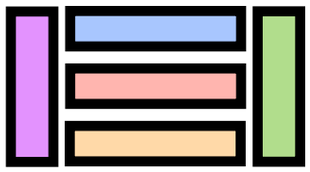


# **Instantiating** the Architecture





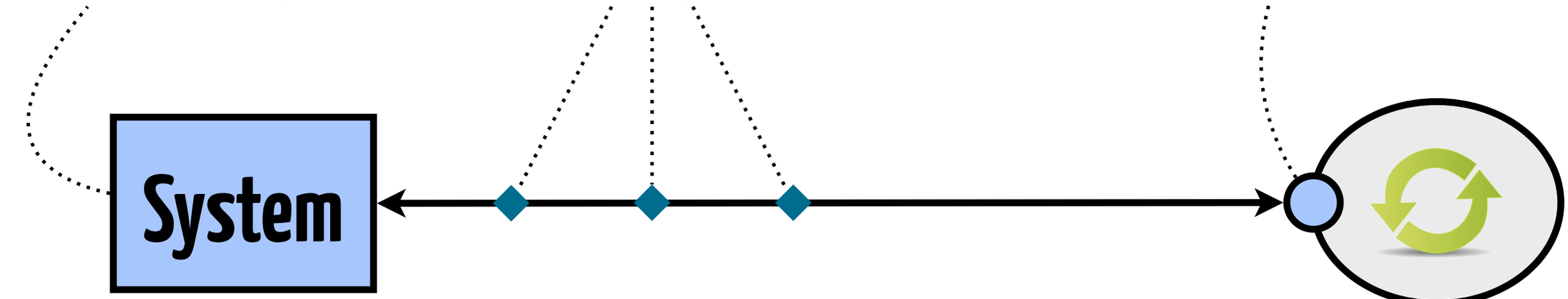
# **Instantiating** the Architecture

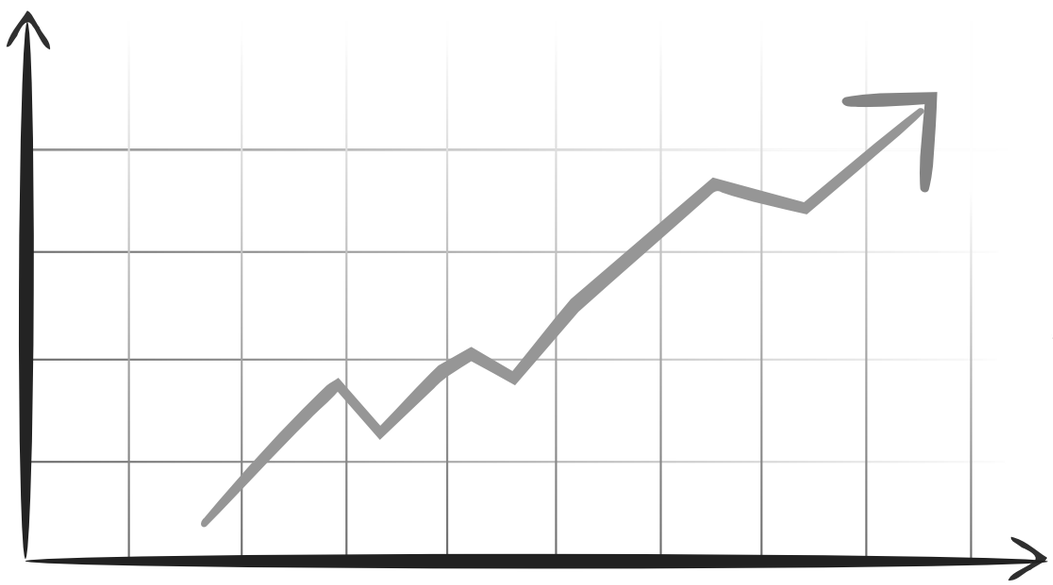


**Registry**

**Transport**

**Broker**

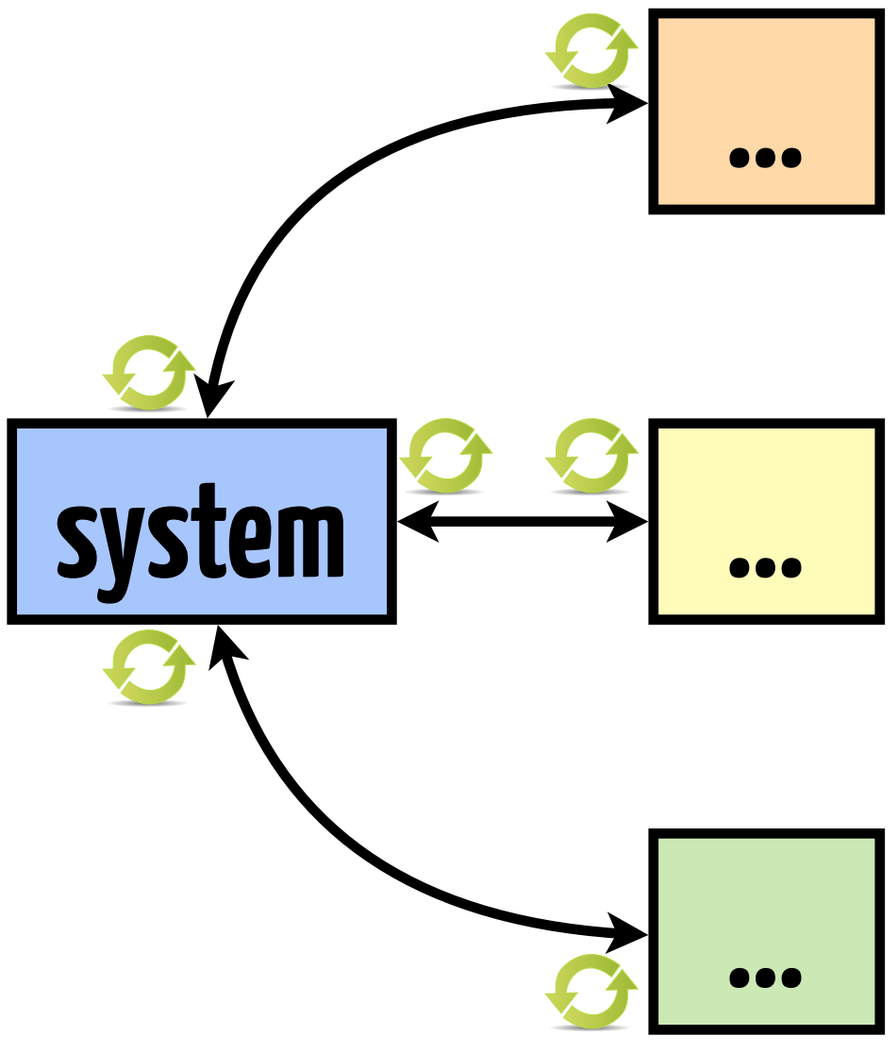


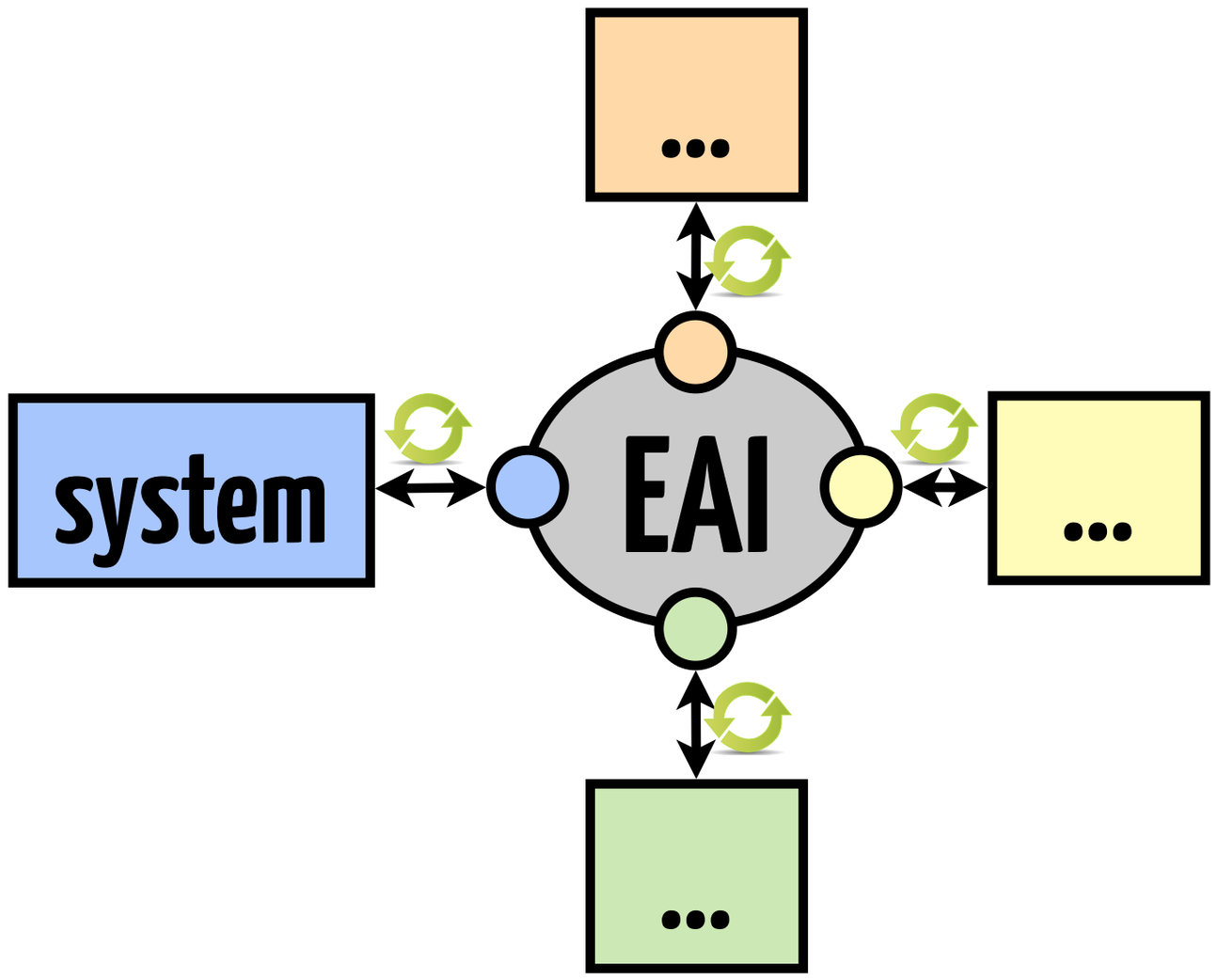



**Process Mgmt**

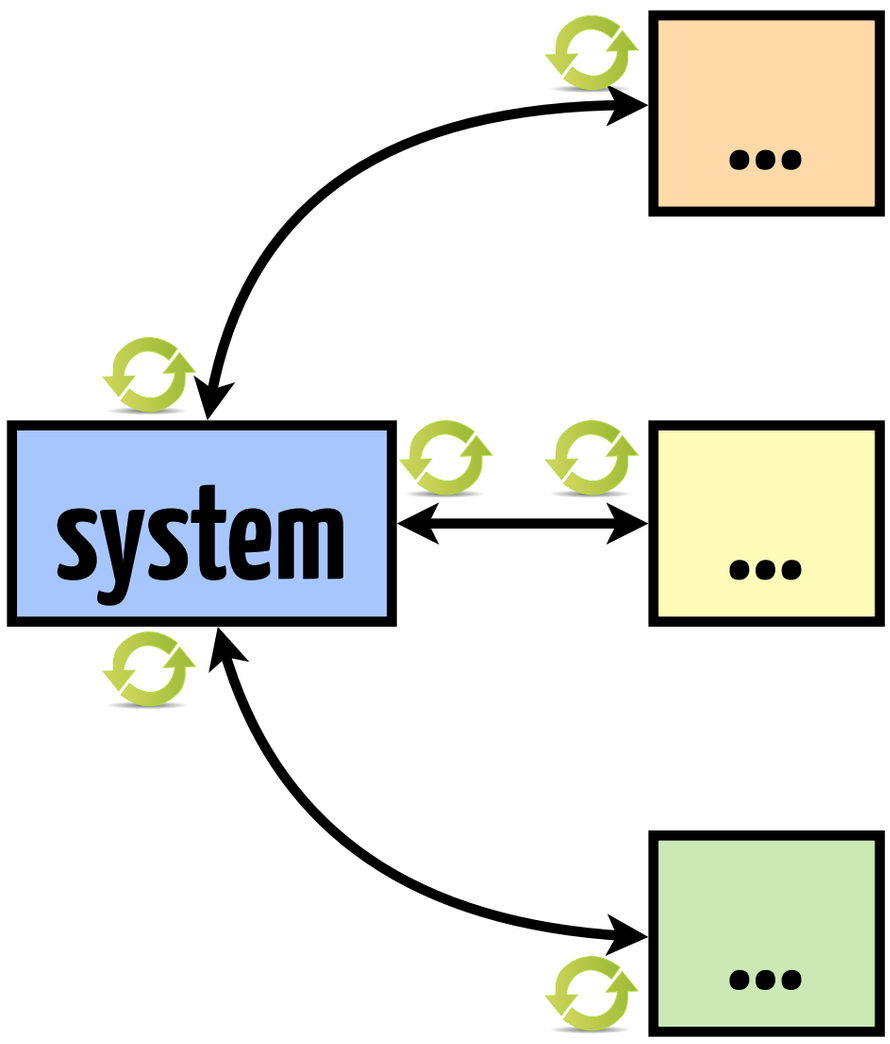
**Monitoring**

# Pros and Cons

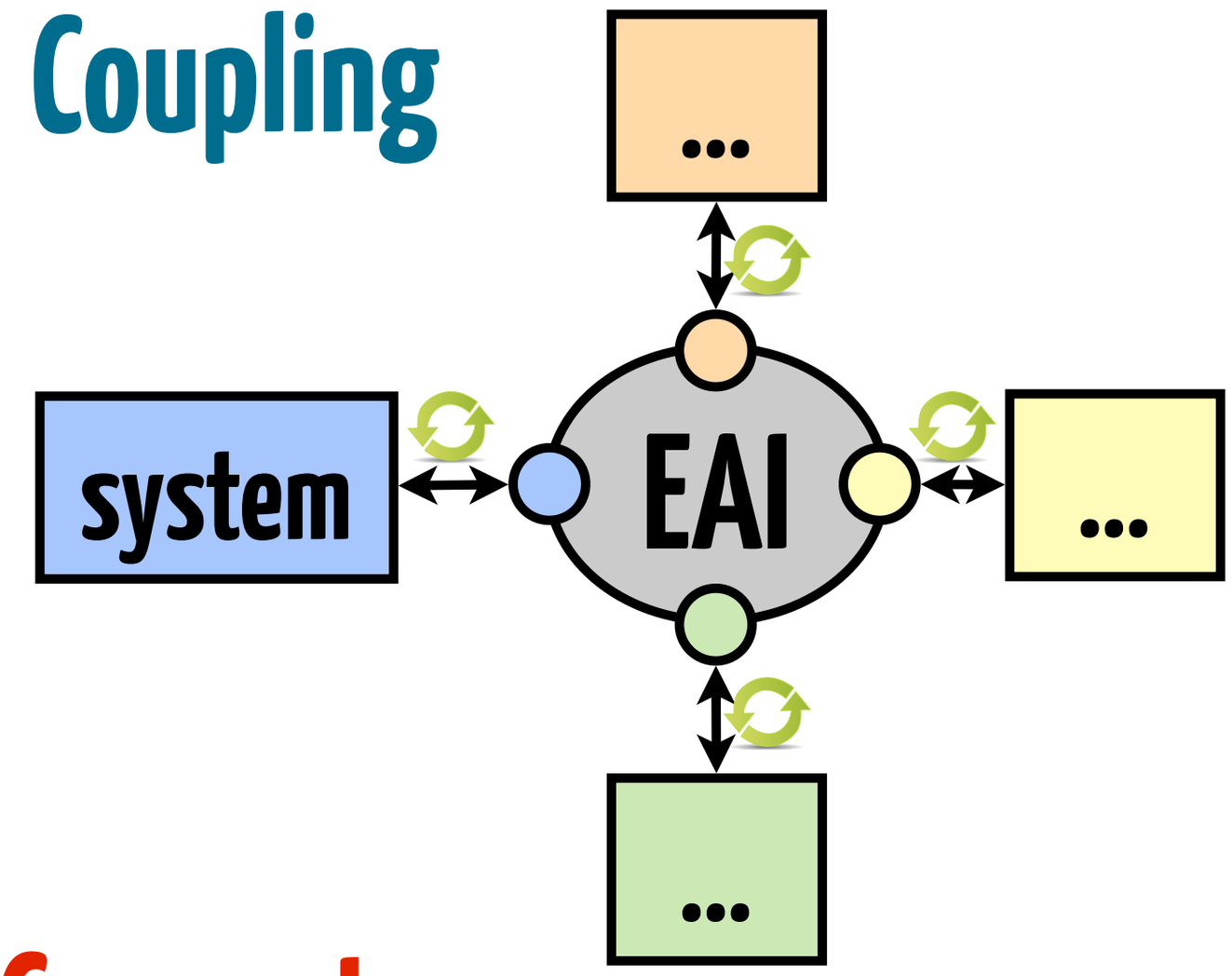


# Pros and Cons

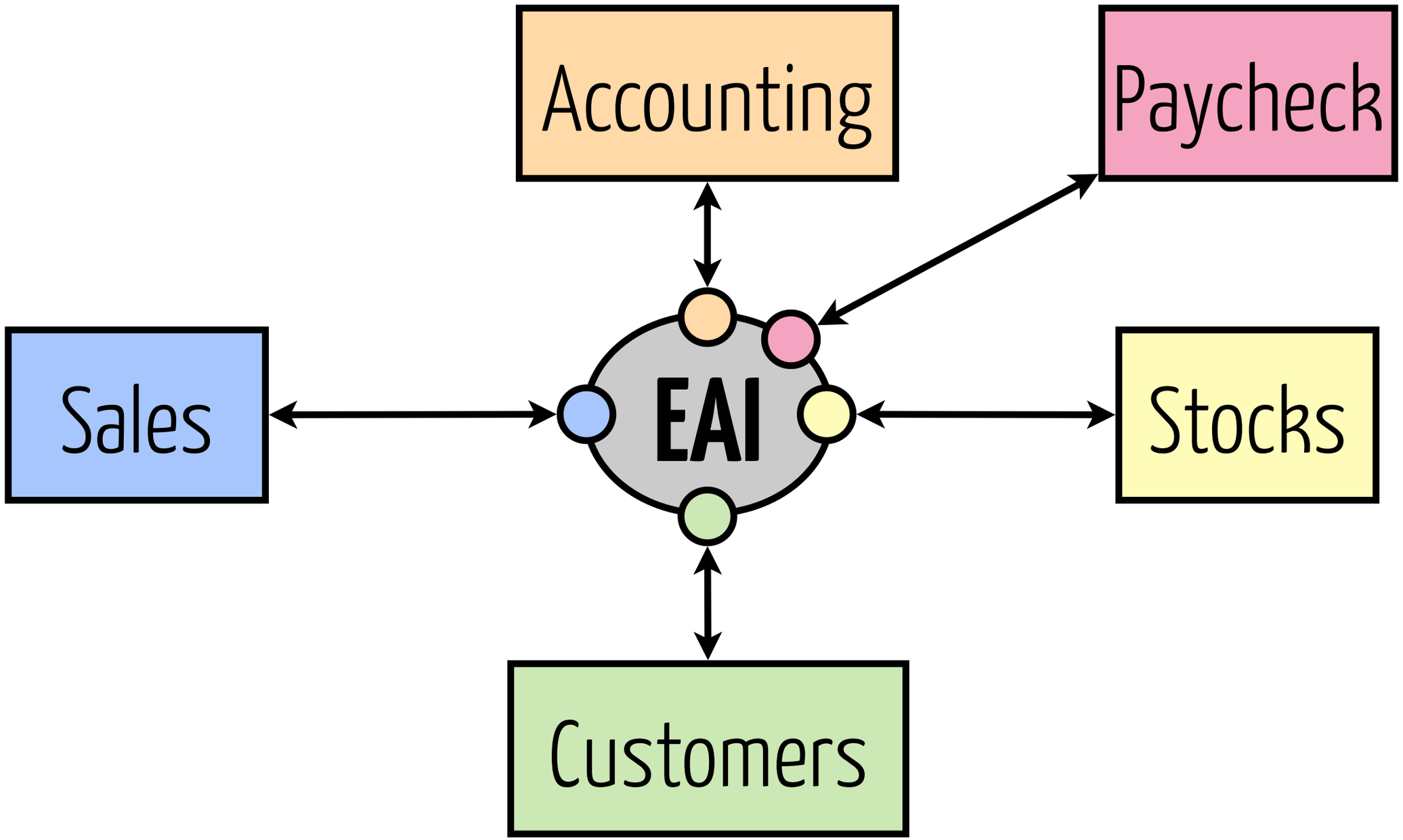



## Coupling




## Connectors

# Still not **shocked**?



# Still not **shocked**?

Accounting

Paycheck

Sales

**SPOF**

Stocks

Customers

[petals]

# Single Point Of Failure

00's

# Enterprise Service Bus

**Exposed** Service

Sales

Accounting

Paycheck

Enterprise Service Bus

Customers

Stocks

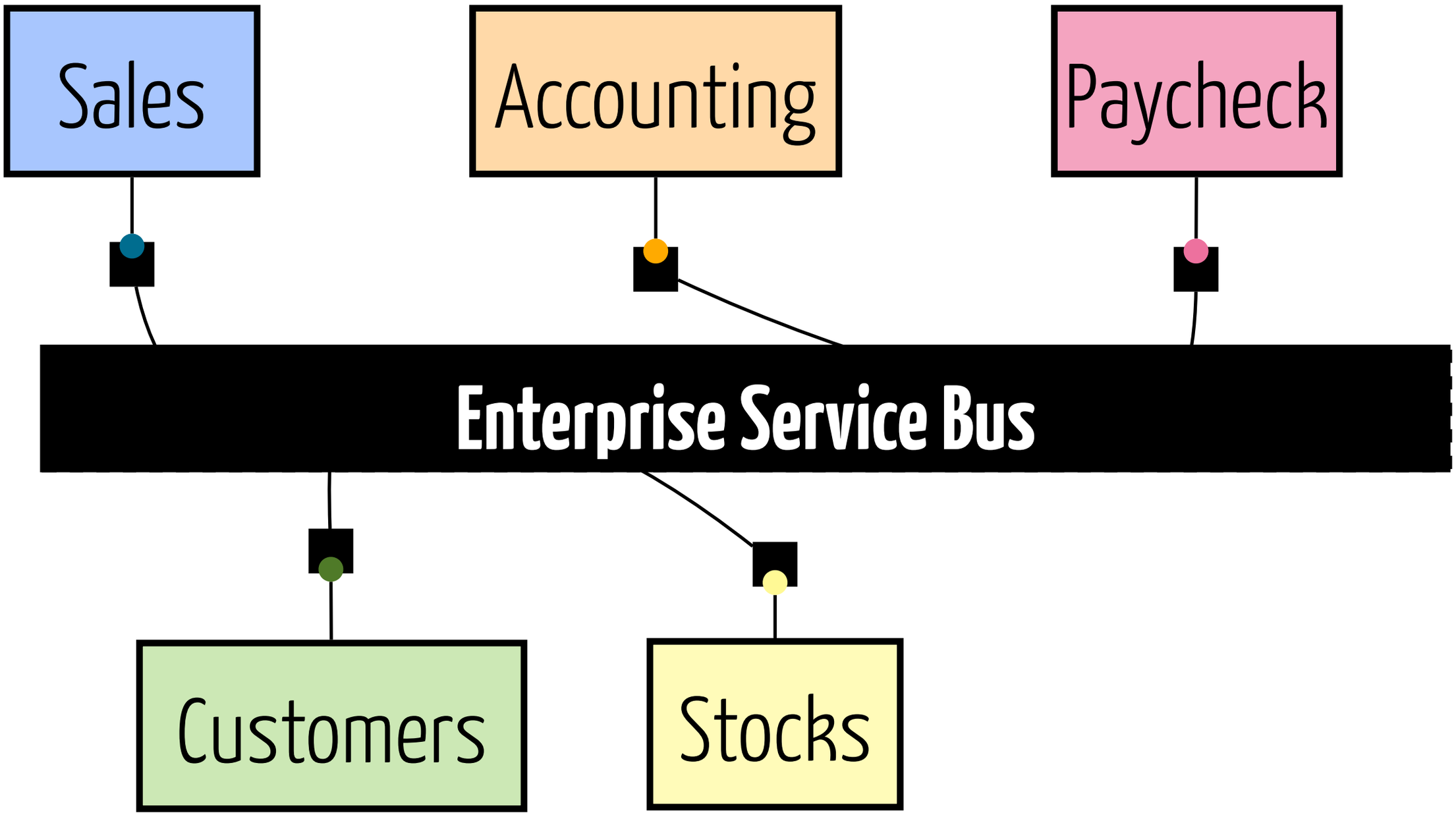
**Standard**

Connectors

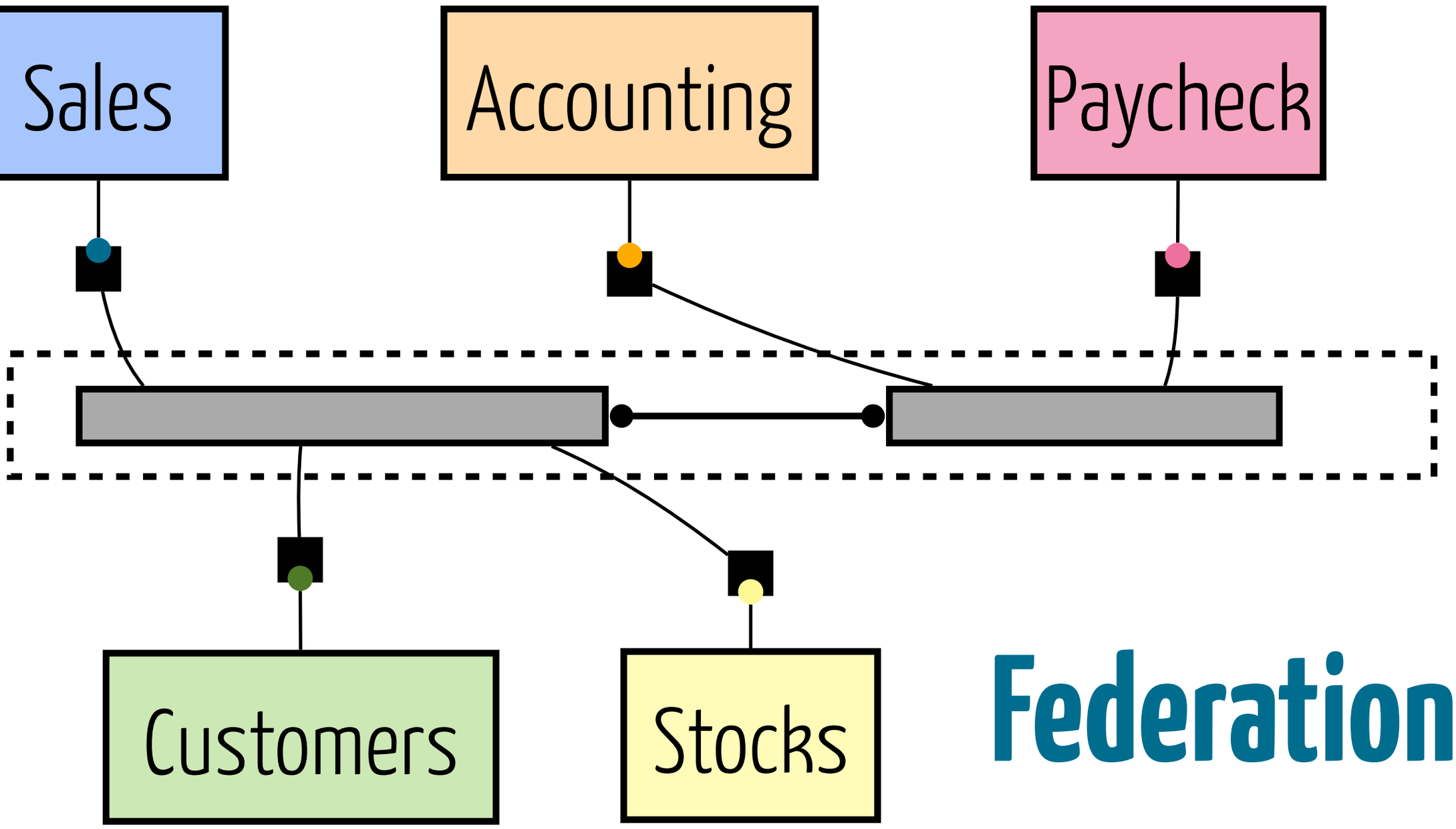
# Wait!

# Wait!

## Single Point Of Failure


Sales
Accounting
Paycheck
Enterprise Service Bus
Customers
Stocks

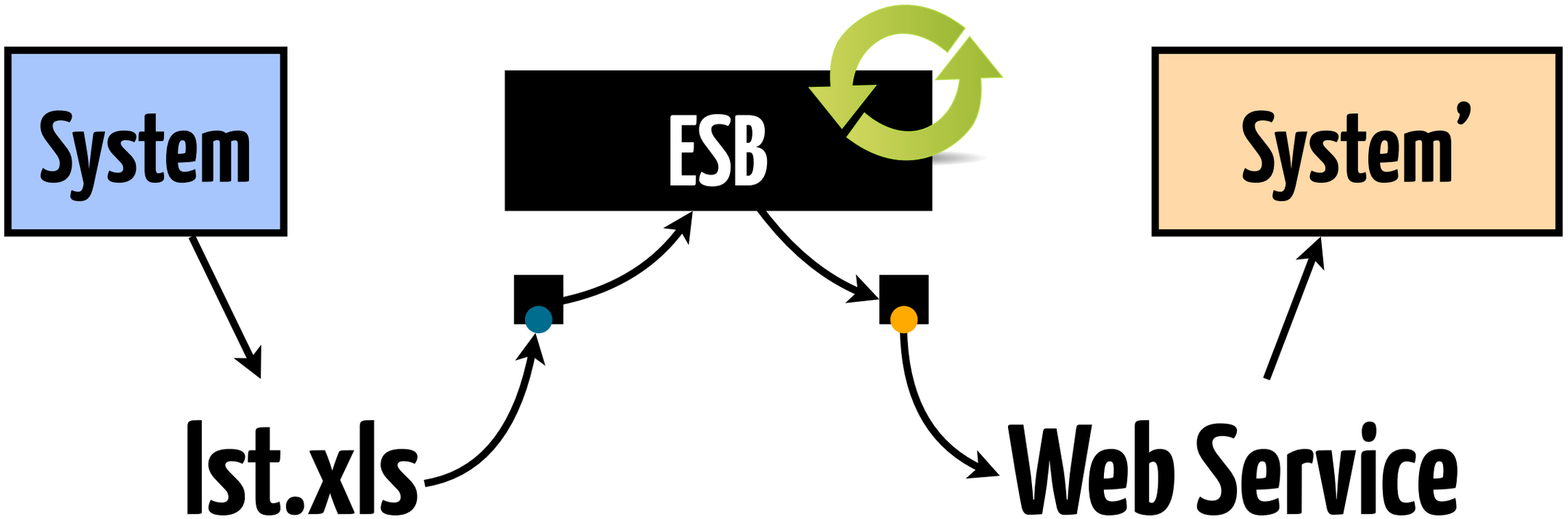
# Distribution




# Federation

# **Connectors** off the **shelf**

# **Connectors** off the **shelf**



# Messaging Systems: Dataflow-driven Development

## Opening the magic box

# Back to the **roots**: Pipe & Filters

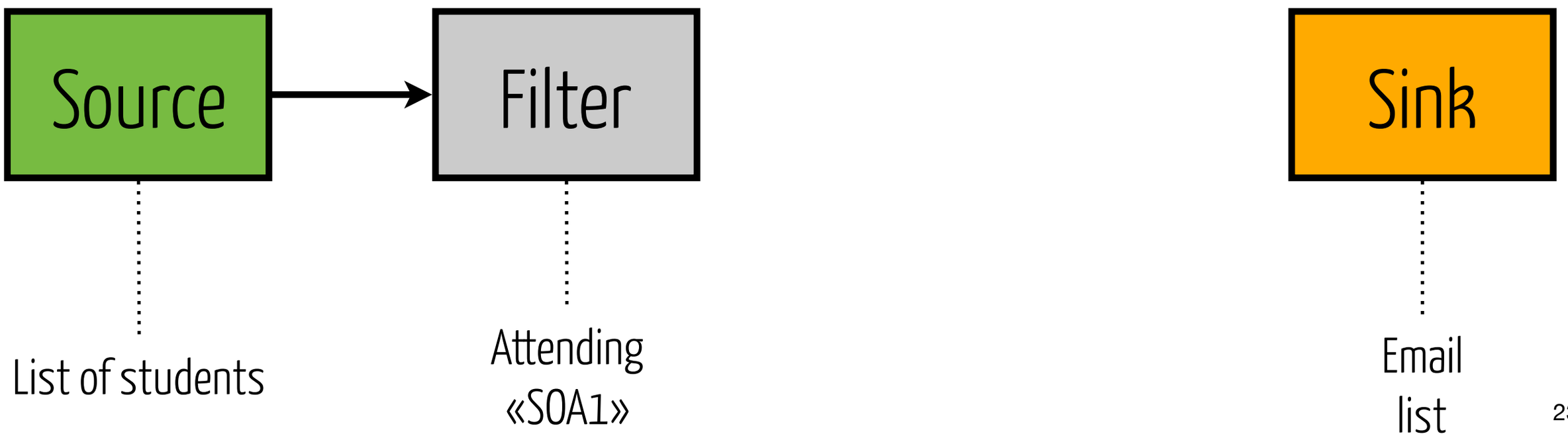
[Khan'78]
[Garlan'96]

**Email** the **students** attending SOA1 who had delivered the first lab.

# Back to the **roots**: Pipe & Filters

[Khan'78]
[Garlan'96]

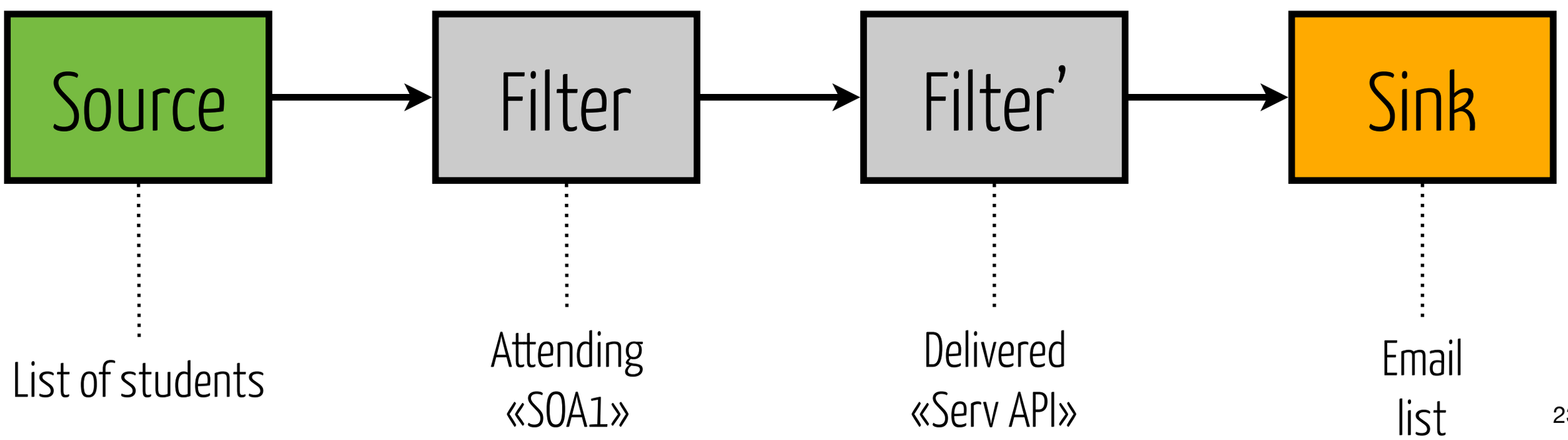
**Email** the **students** attending SOA1 who had delivered the first lab.



# Back to the **roots**: Pipe & Filters

[Khan'78]
[Garlan'96]

**Email** the **students** attending SOA1 who had delivered the first lab.

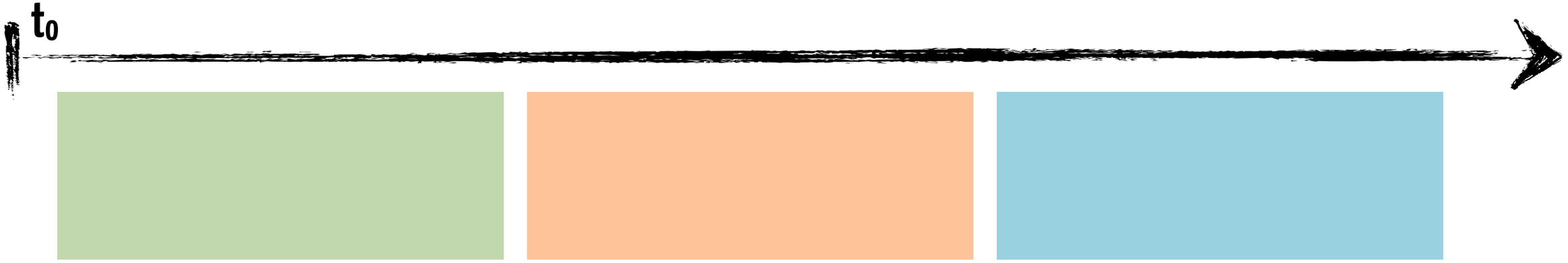


# Filter

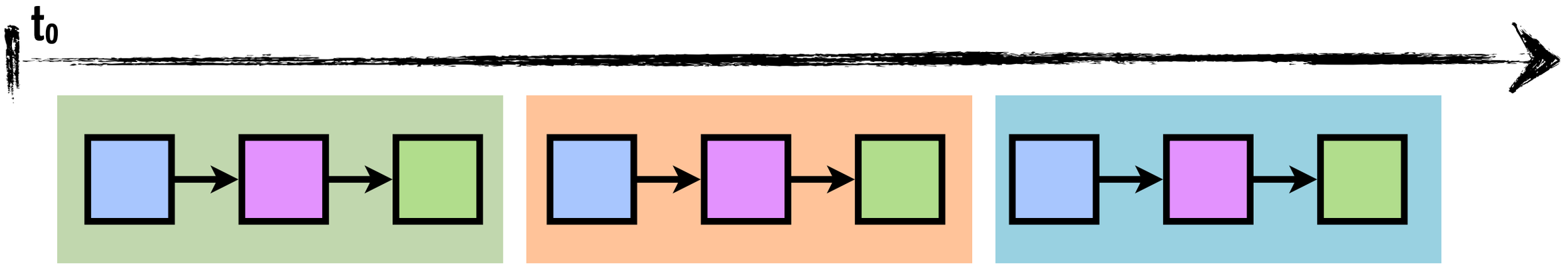
```
cat /etc/passwd | \
cut -d ':' -f 1 | \
grep seb | \
wc -l
```

# Pipe

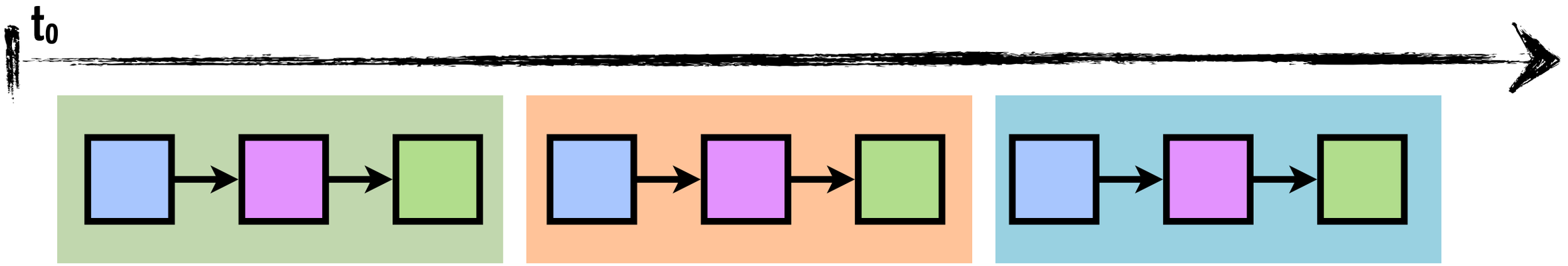
# **Handling** Messages

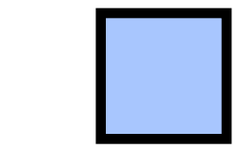


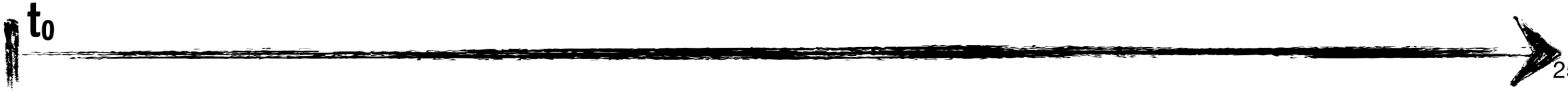
# **Handling** Messages



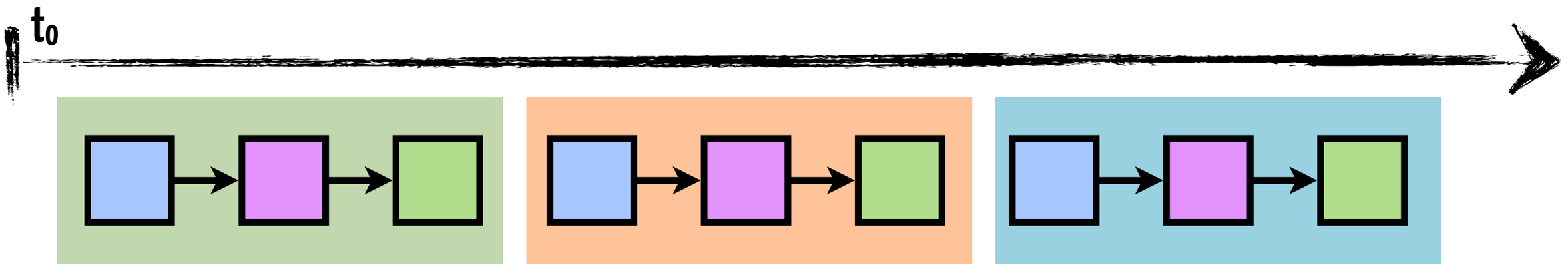
# **Handling** Messages

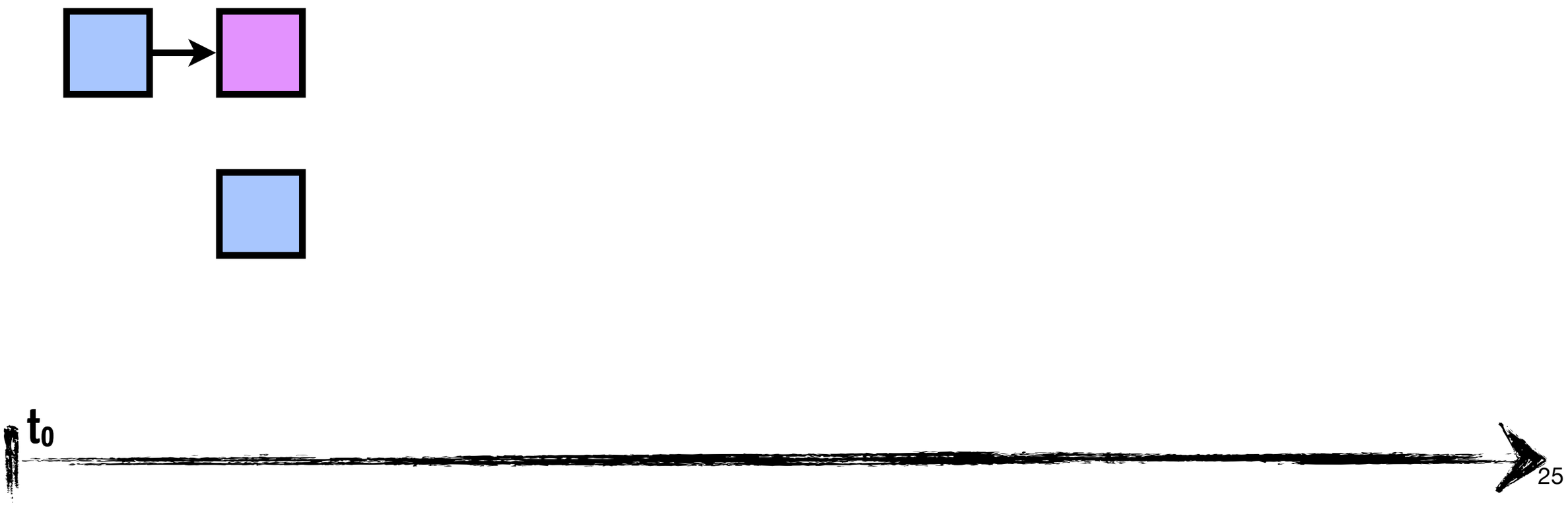


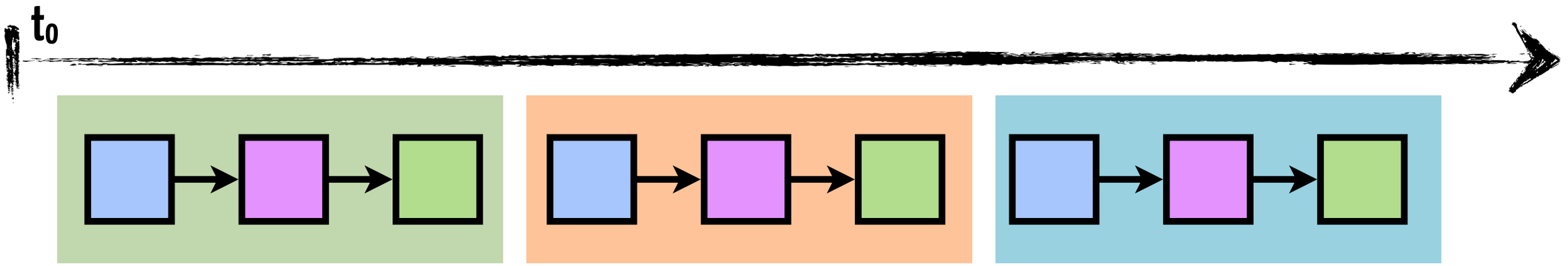


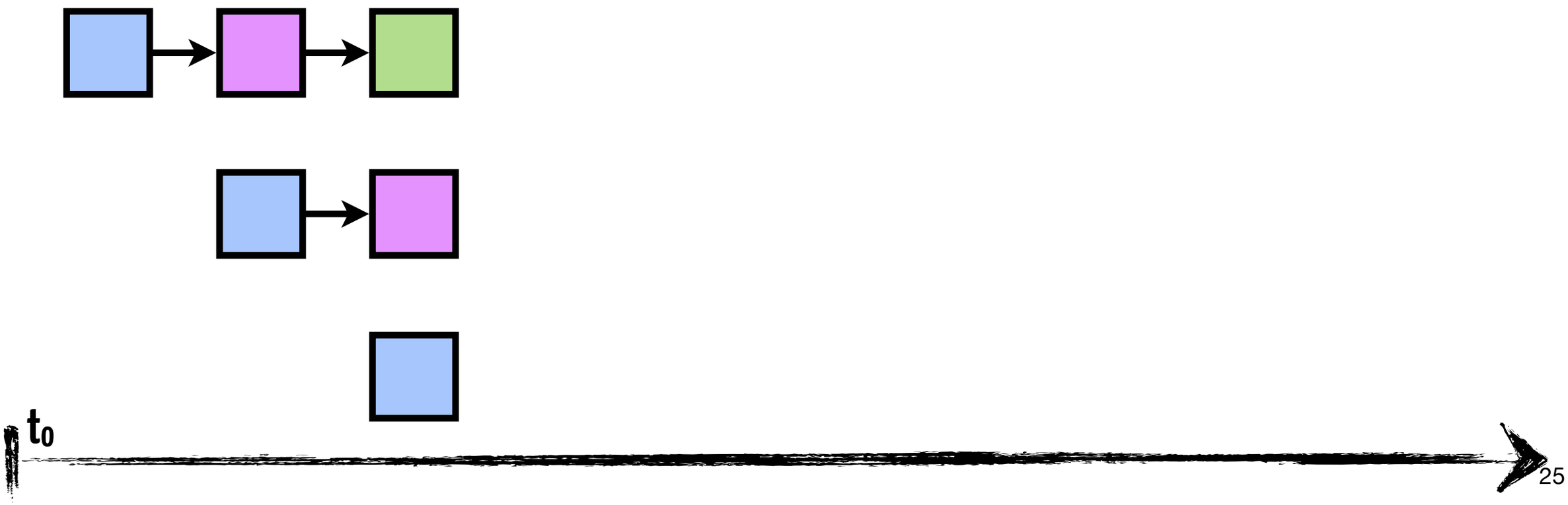
# **Handling** Messages



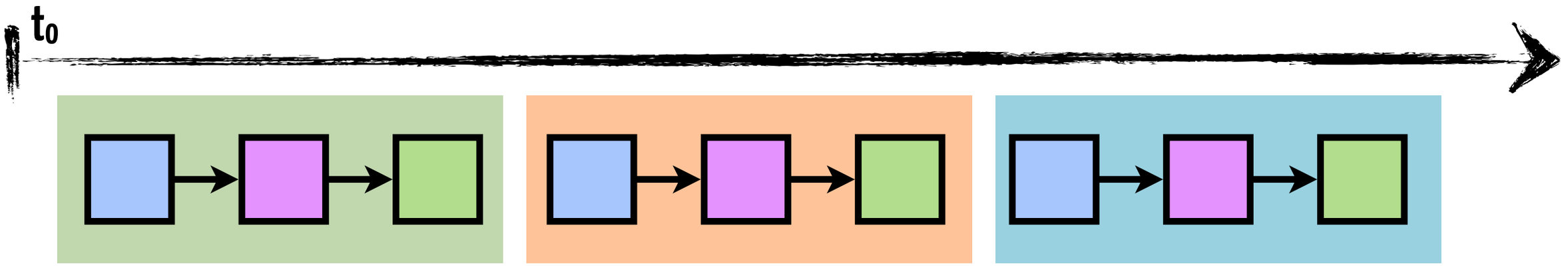



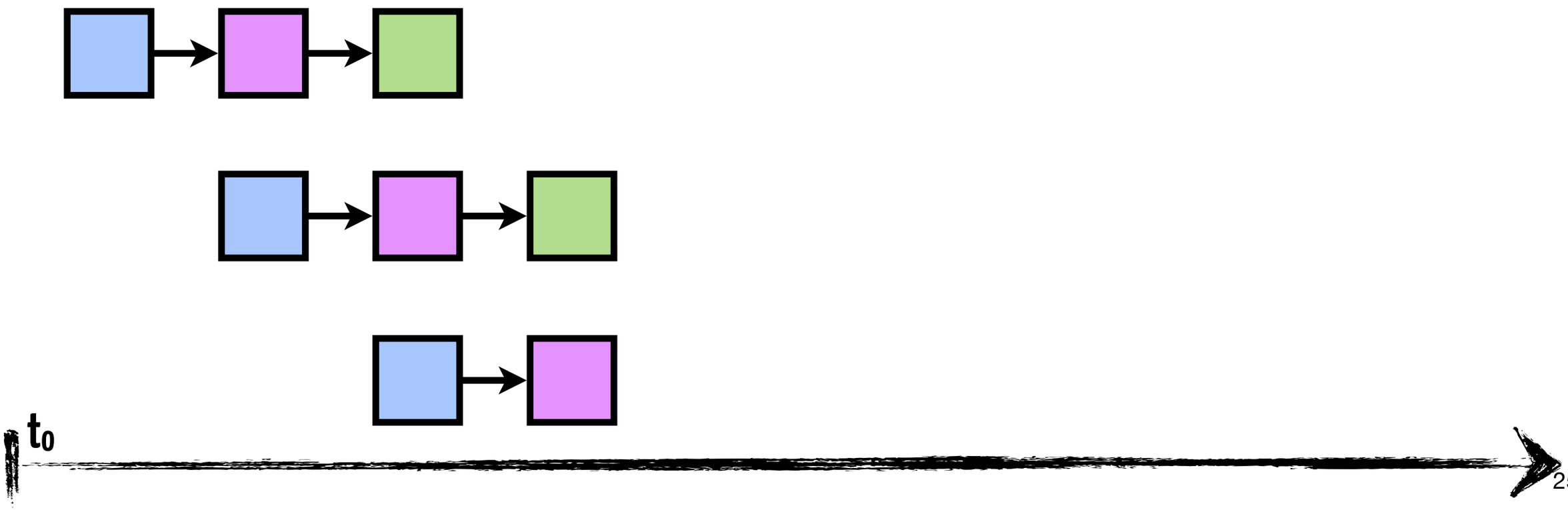
# **Handling** Messages



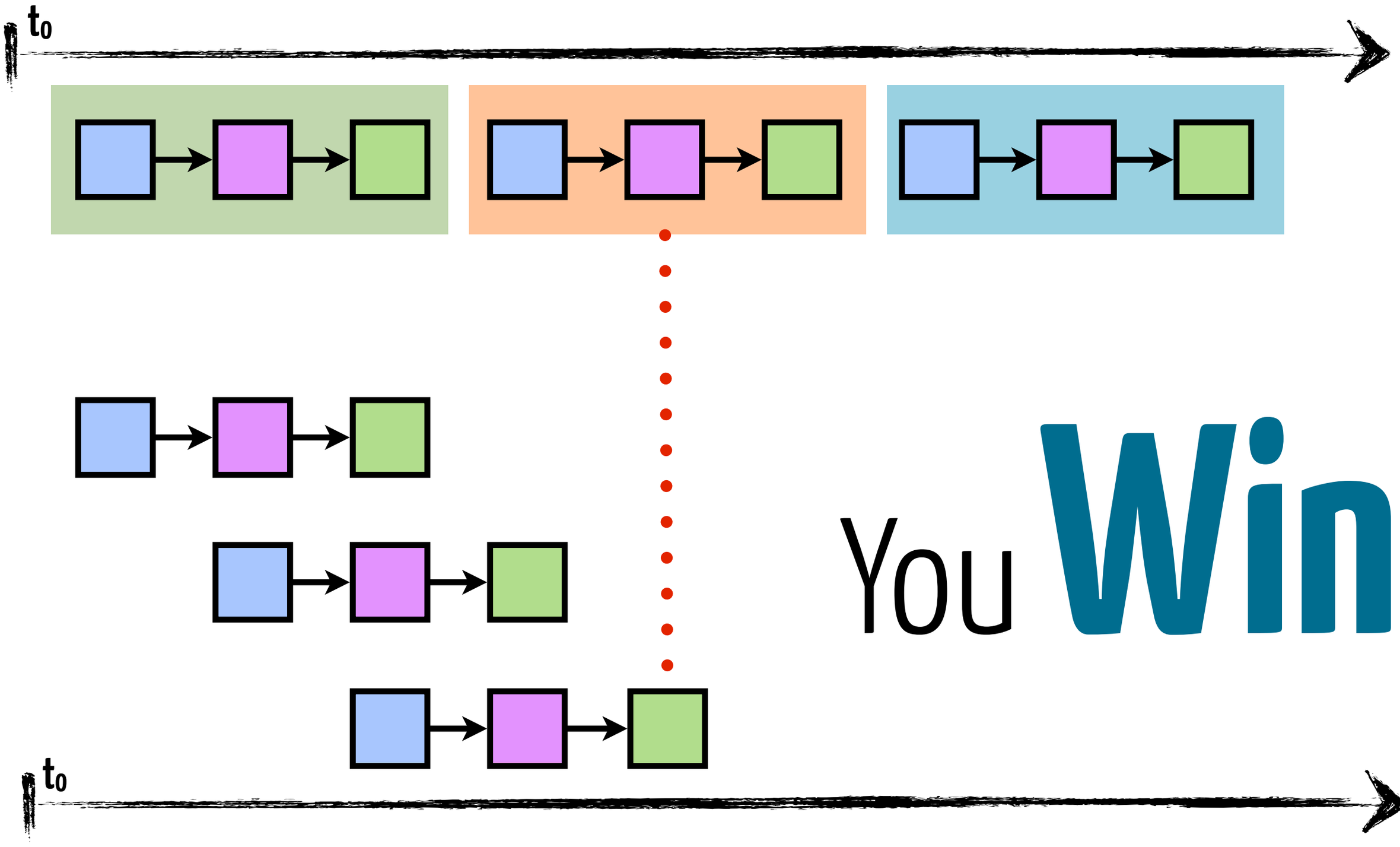



# **Handling** Messages



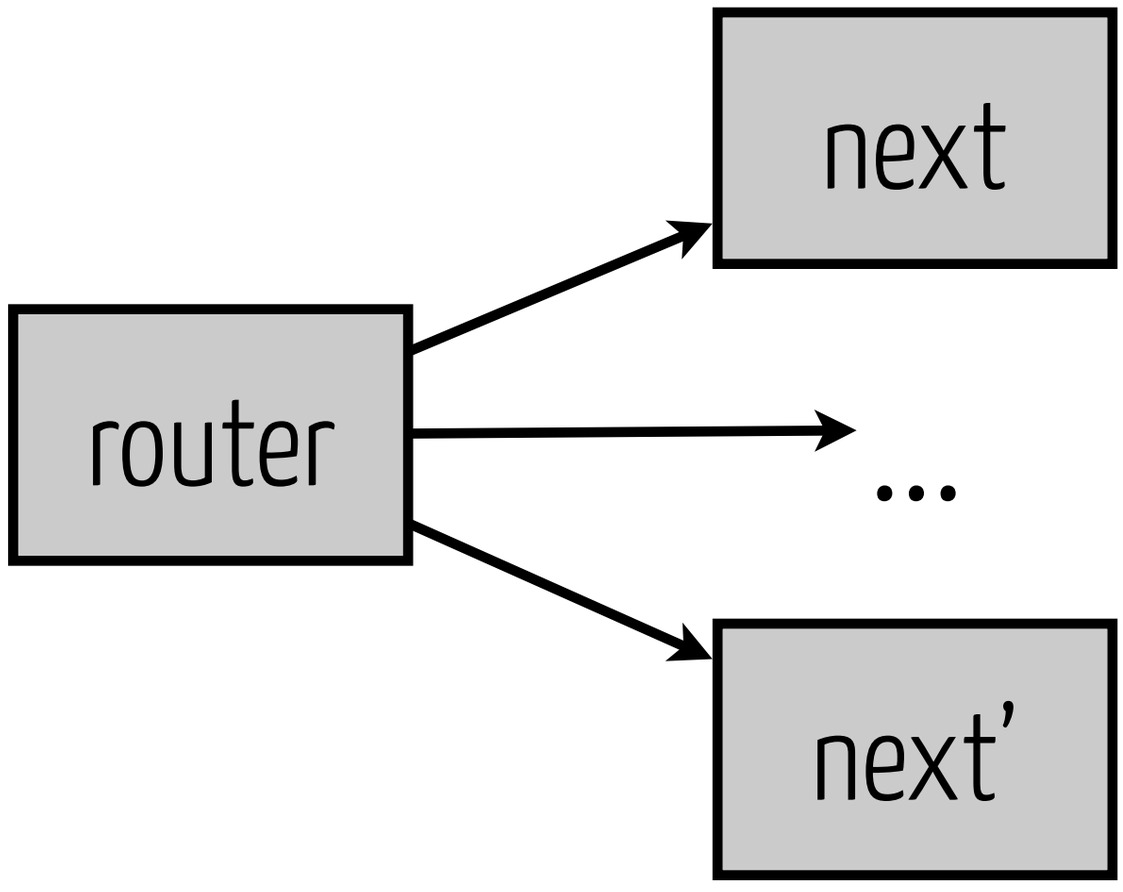



# **Handling** Messages

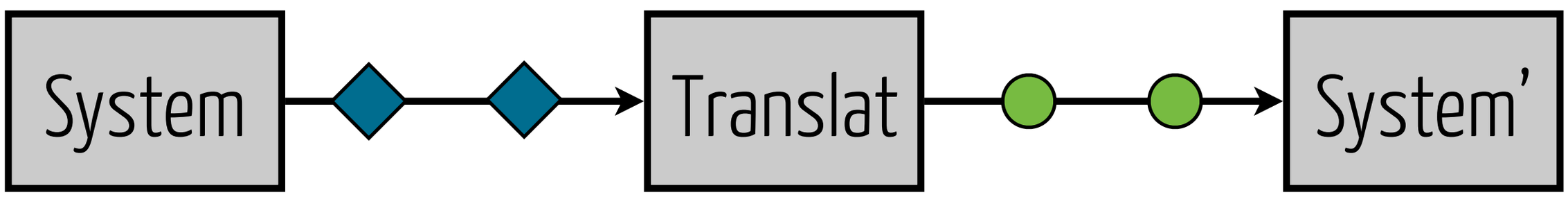



You **Win**!

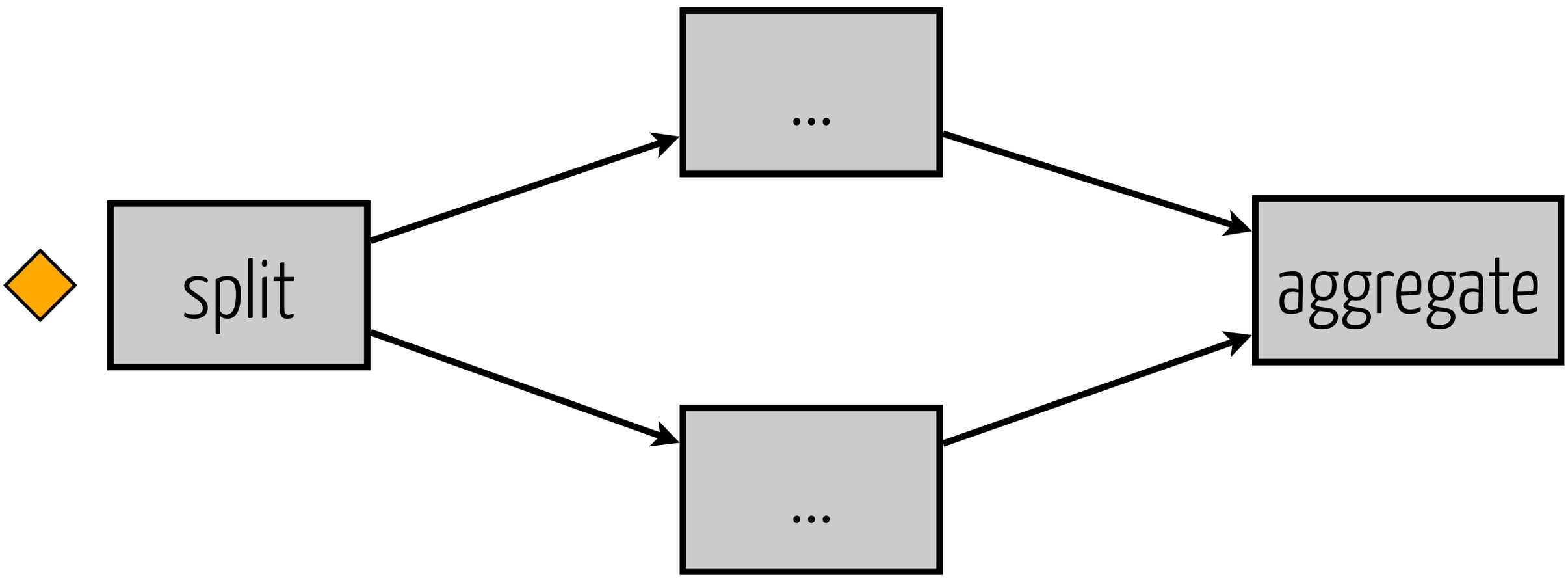
# «Technical» Filter: **Router**



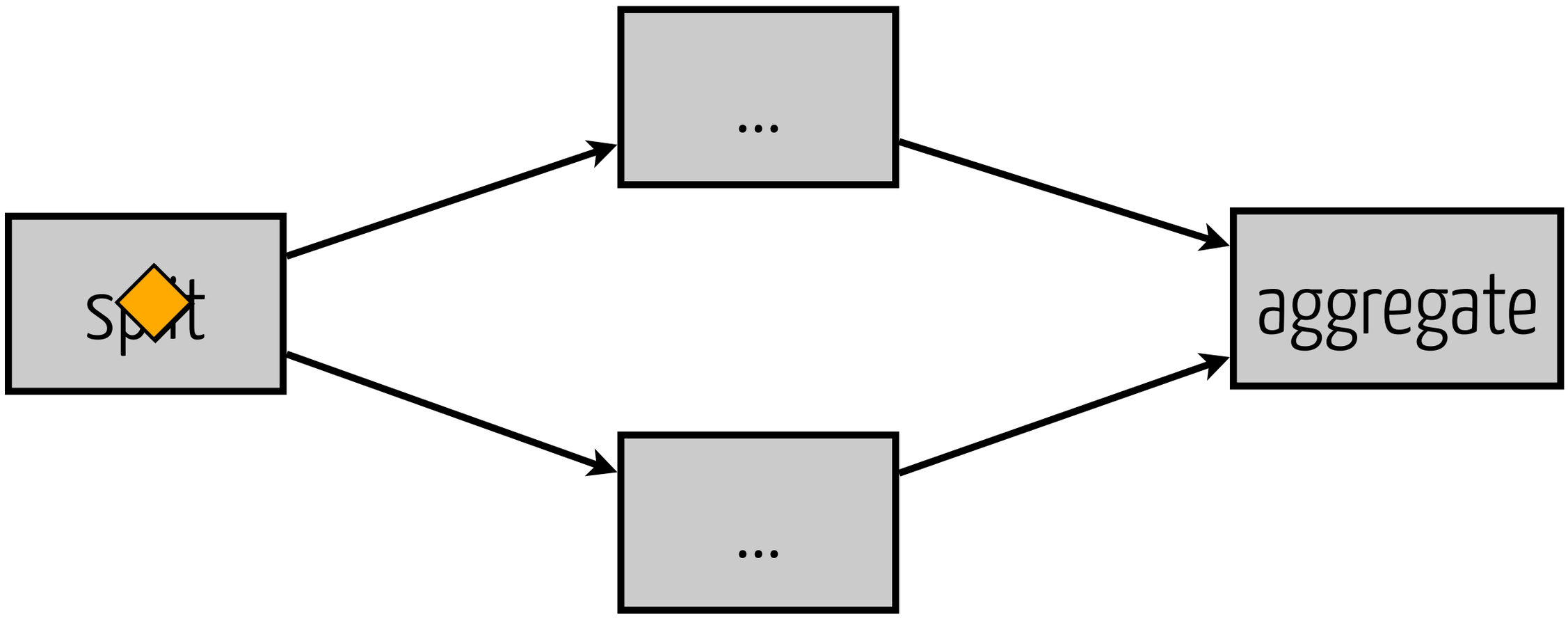
# «Technical» Filter: **Translator**



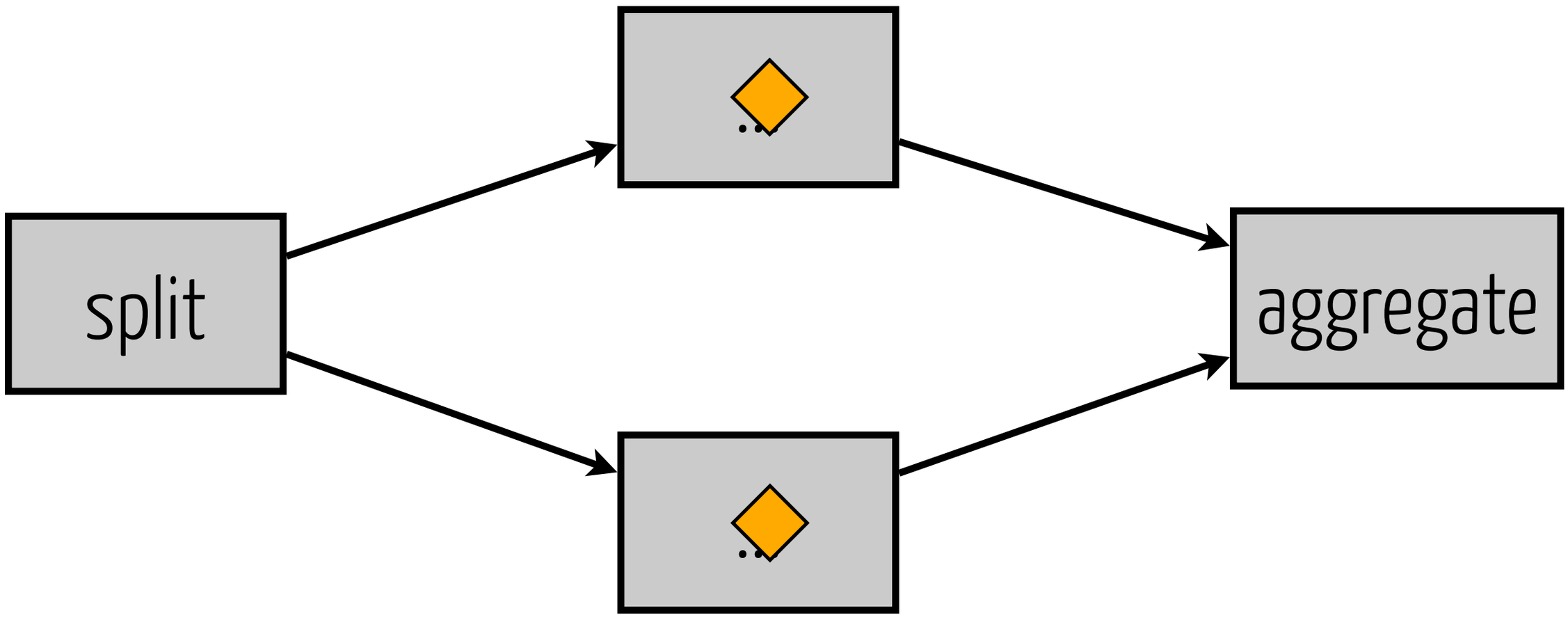
# Data**flow**



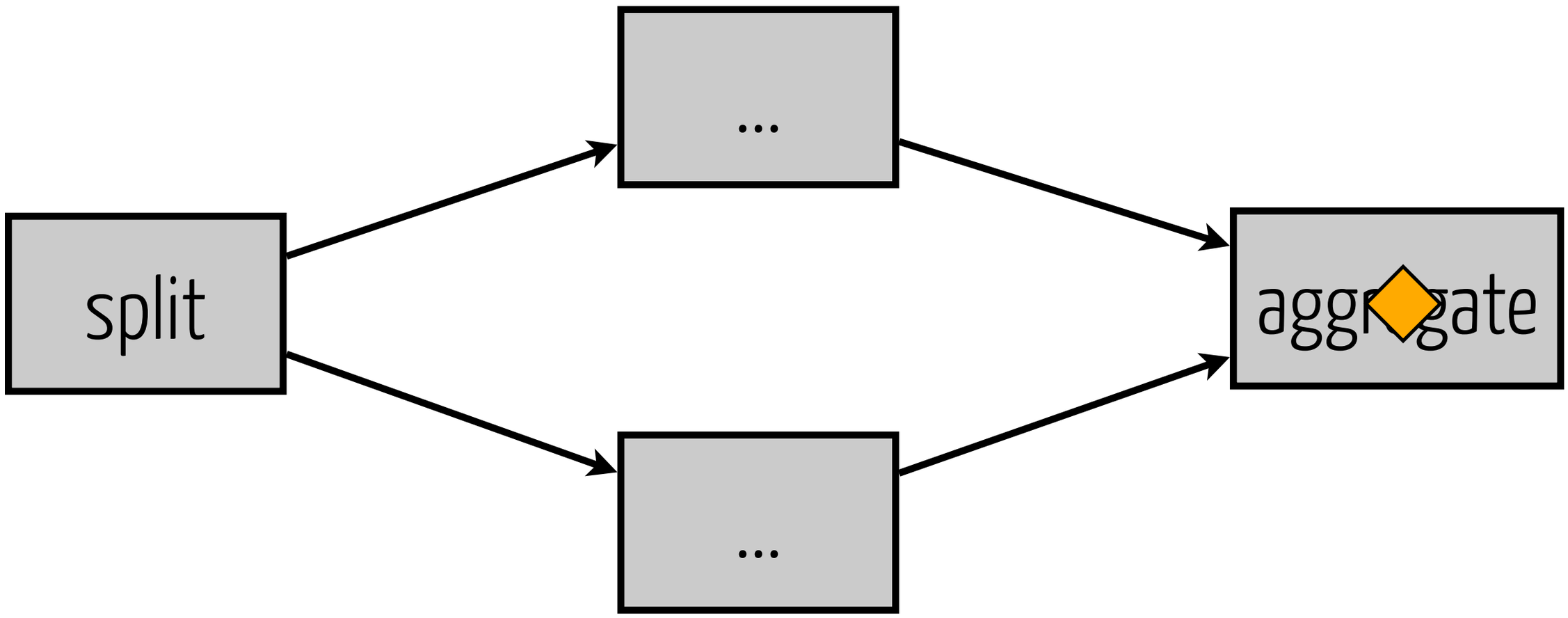
# Data**flow**



# Data**flow**

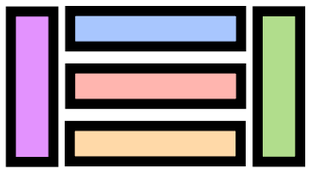


# Data**flow**



# Conclusions

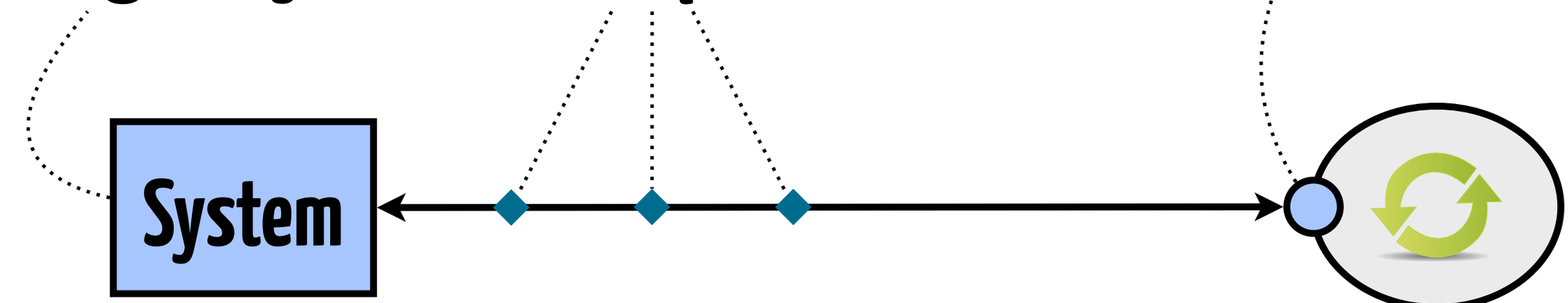
# **Instantiating** the Architecture

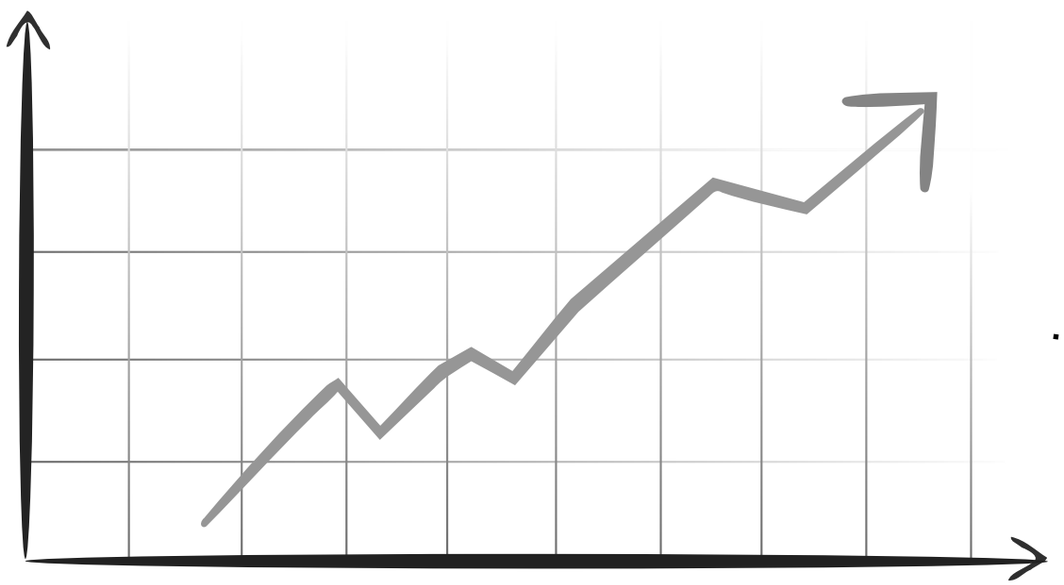


## Registry

## Transport

## Broker



## Process Mgmt

## Monitoring

**Exposed** Service

Sales

Accounting

Paycheck

Enterprise Service Bus

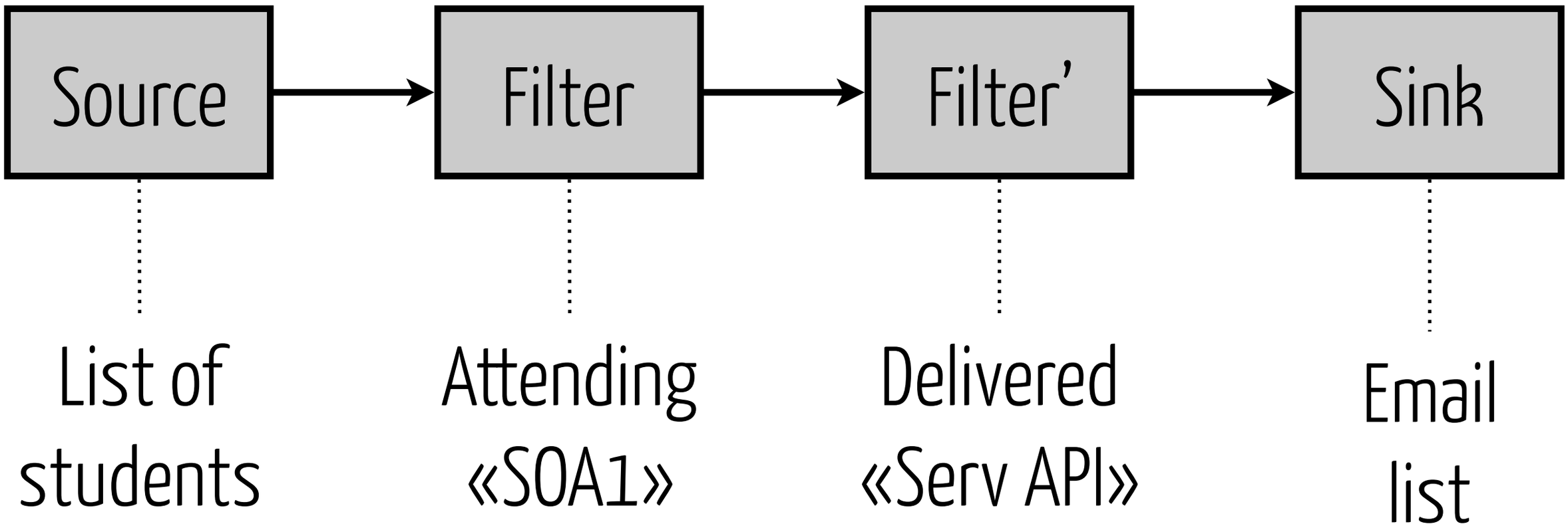
Customers

Stocks

**Standard**

Connectors

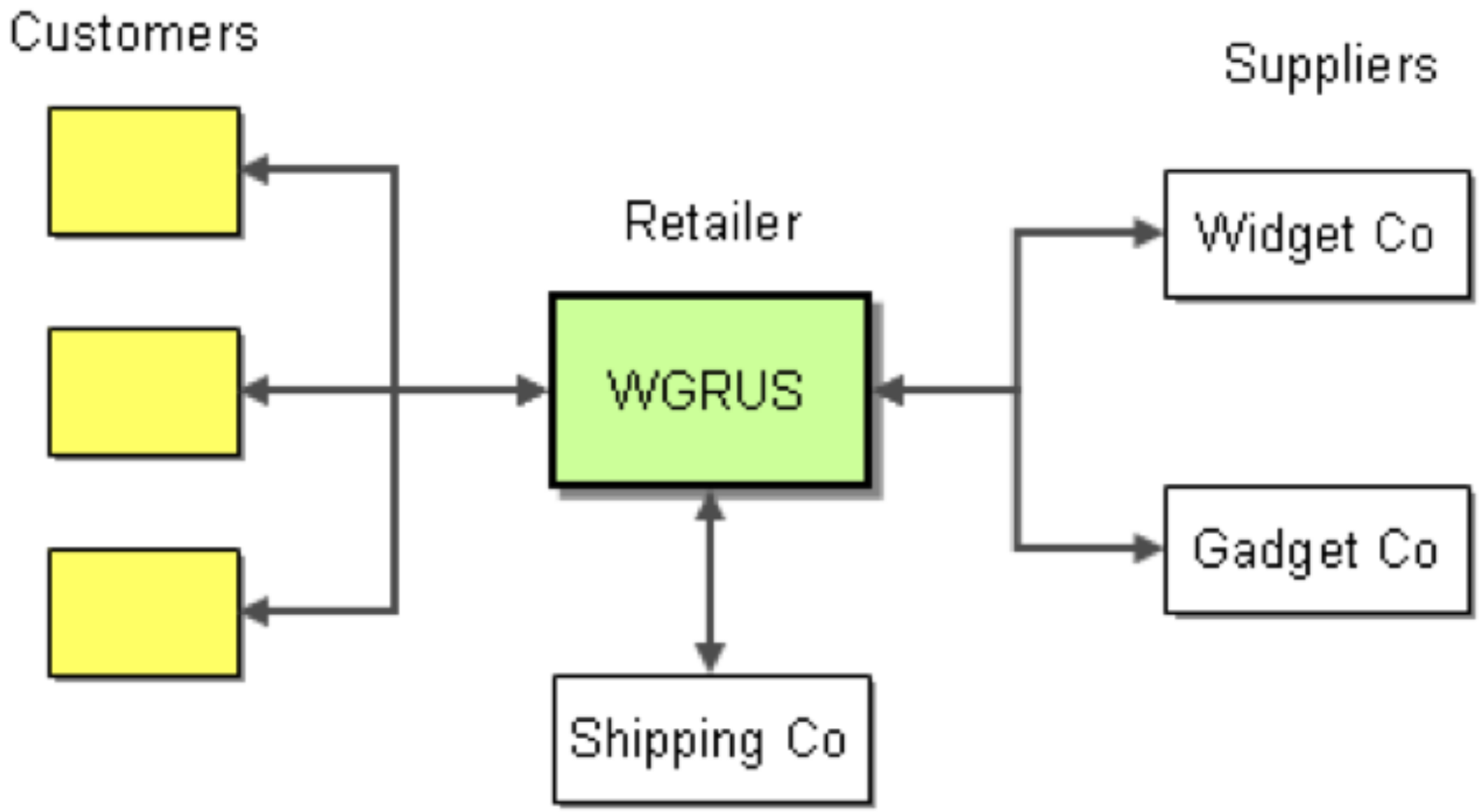
# Back to the **roots**: Pipe & Filters




[Khan'78] [Garlan'96]

# Case Study: Widget-Gadget Corp.

[Enterprise Integration Patterns]



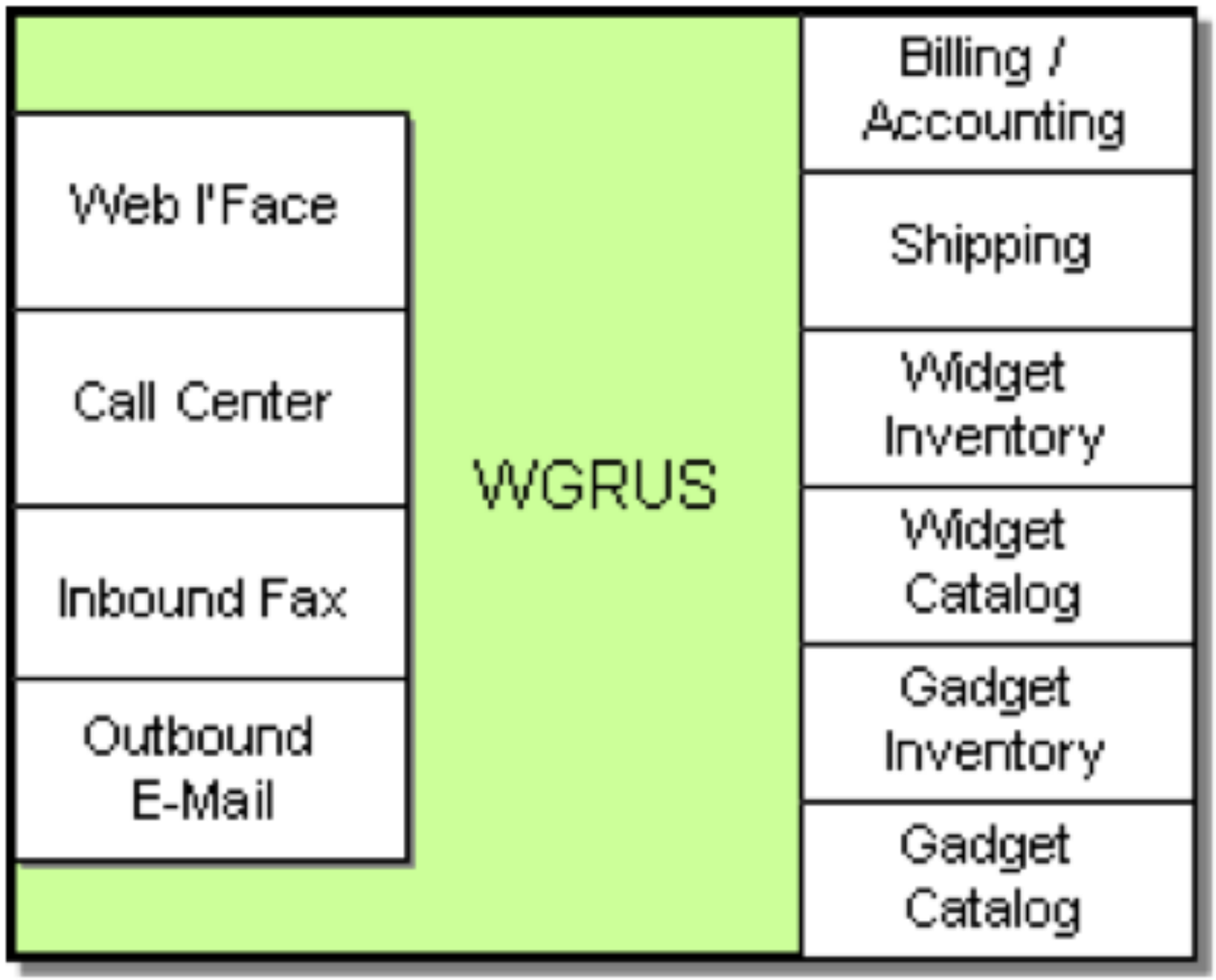

*WGRUS Ecosystem*

[Enterprise Integration Patterns]

# Use Cases

- **Take Orders**
  - Web, Fax or Phone
- **Process Orders**
  - verifying inventory, shipping, ...
- **Check Status**
- **Change Address**
  - Web page to update user's profile
- **New Catalog**
  - support periodical updates

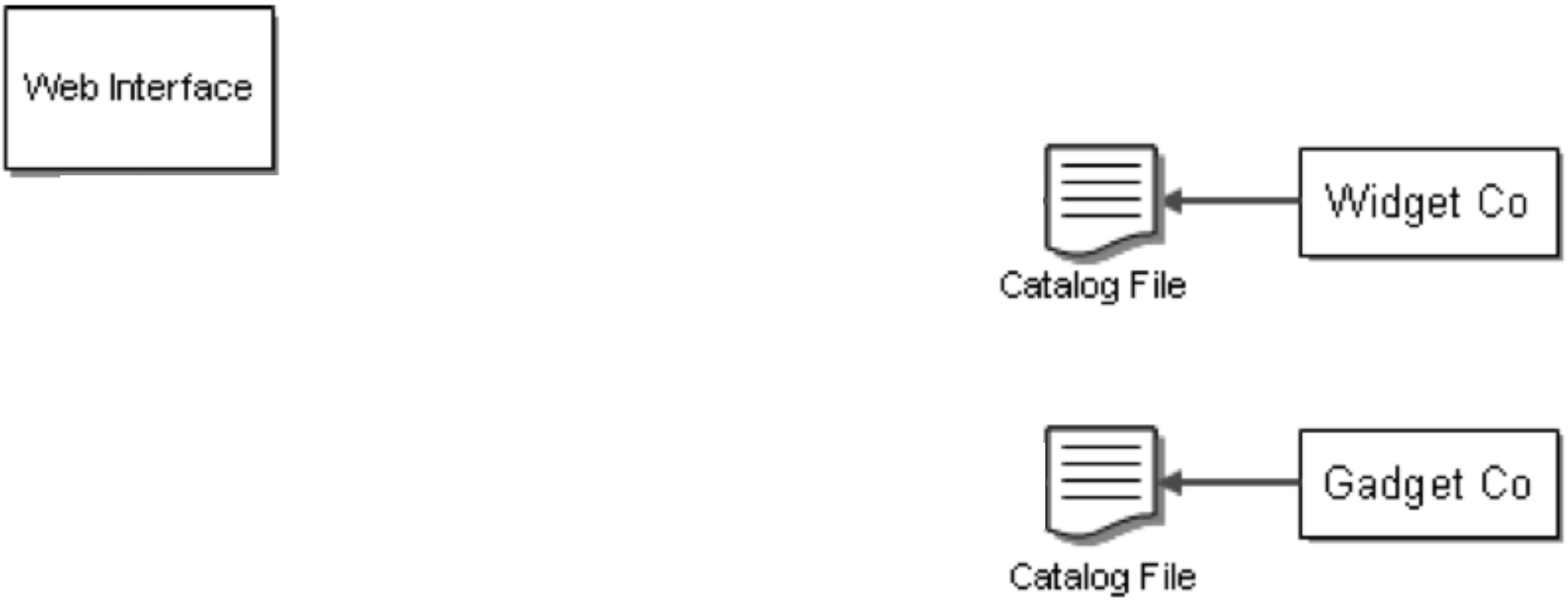
[Enterprise Integration Patterns]




*WGRUS IT Infrastructure*

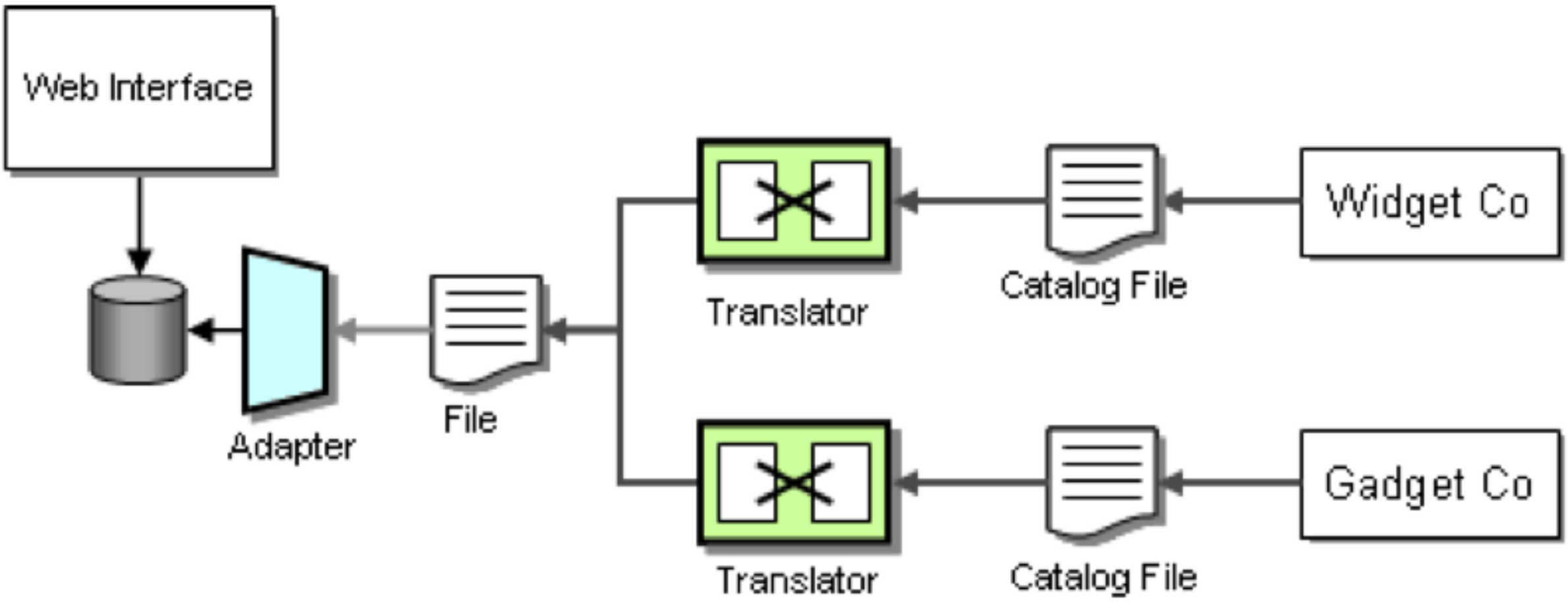
[Enterprise Integration Patterns]

# Catalog Update



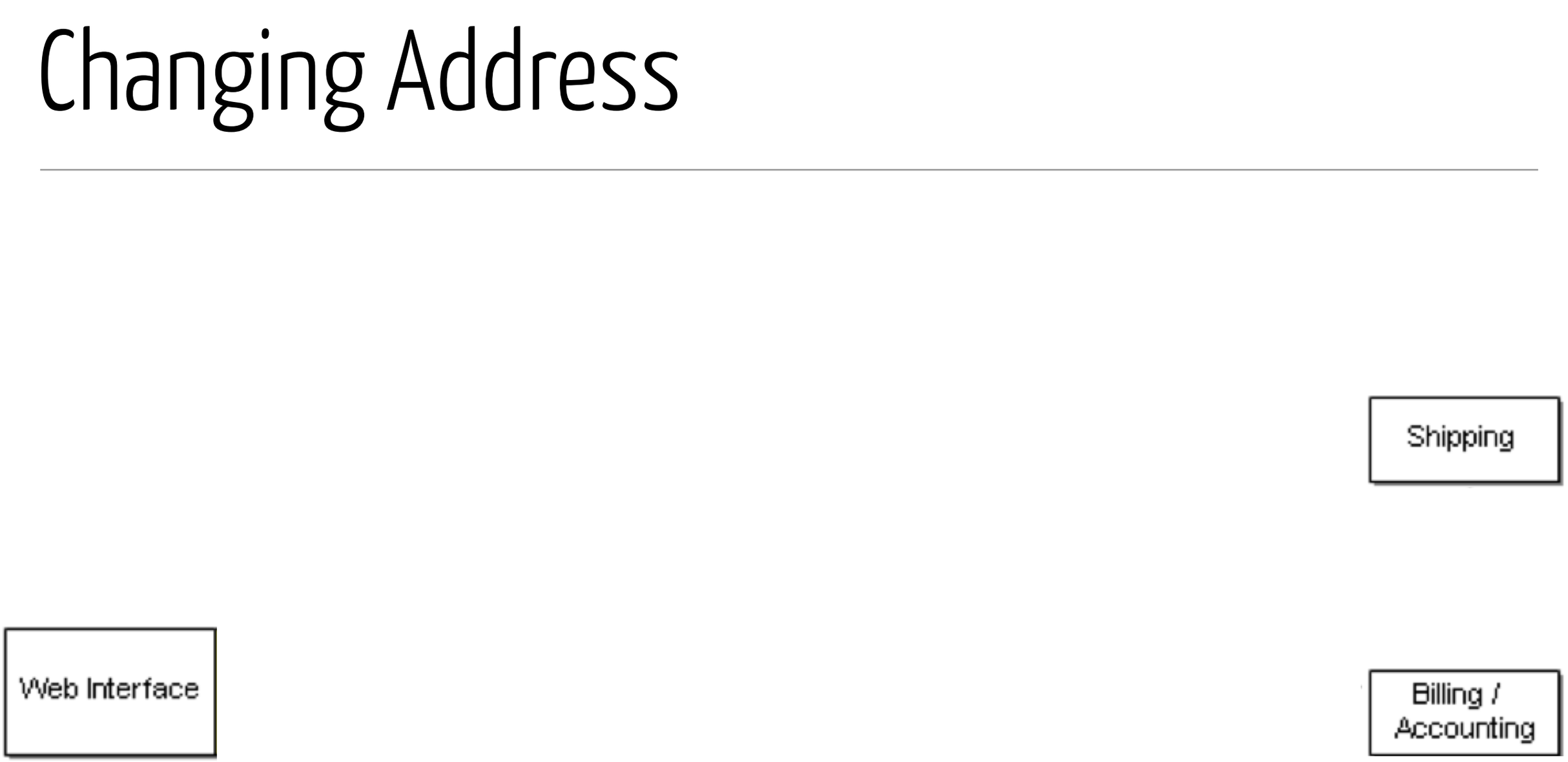

[Enterprise Integration Patterns]

# Catalog Update



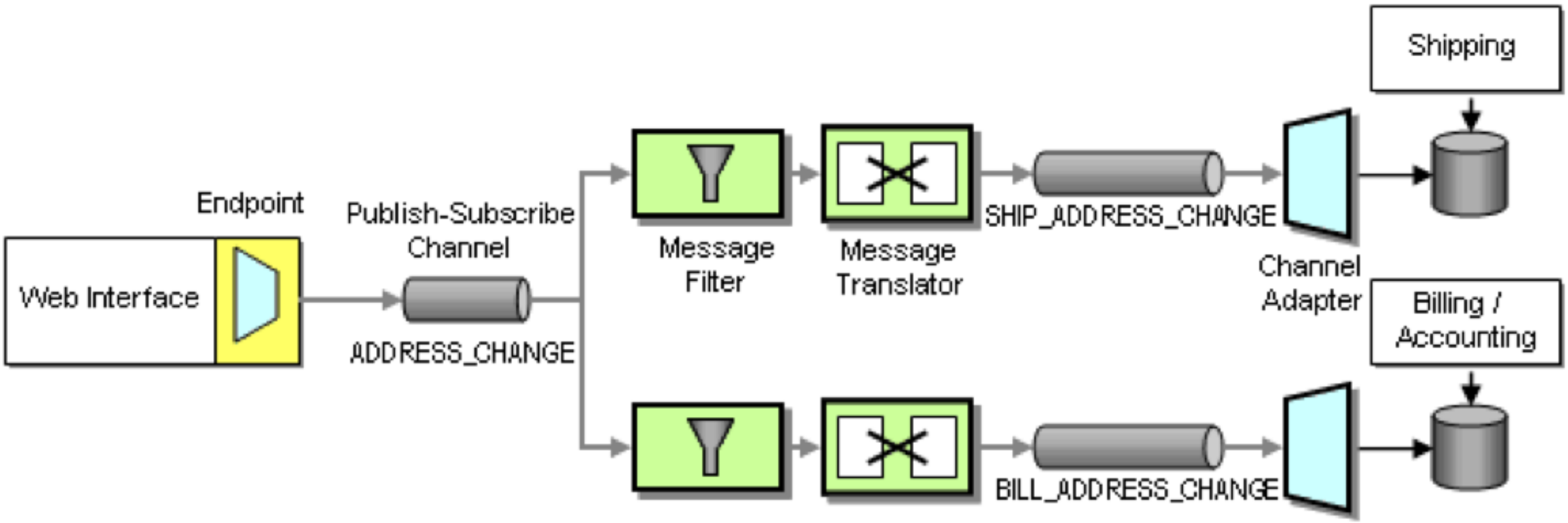

[Enterprise Integration Patterns]

# Changing Address



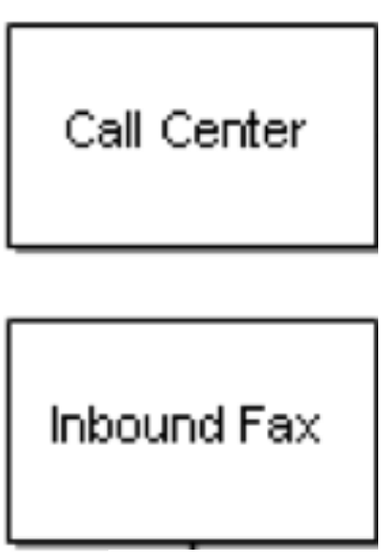

[Enterprise Integration Patterns]

# Changing Address




[Enterprise Integration Patterns]

# Taking Orders

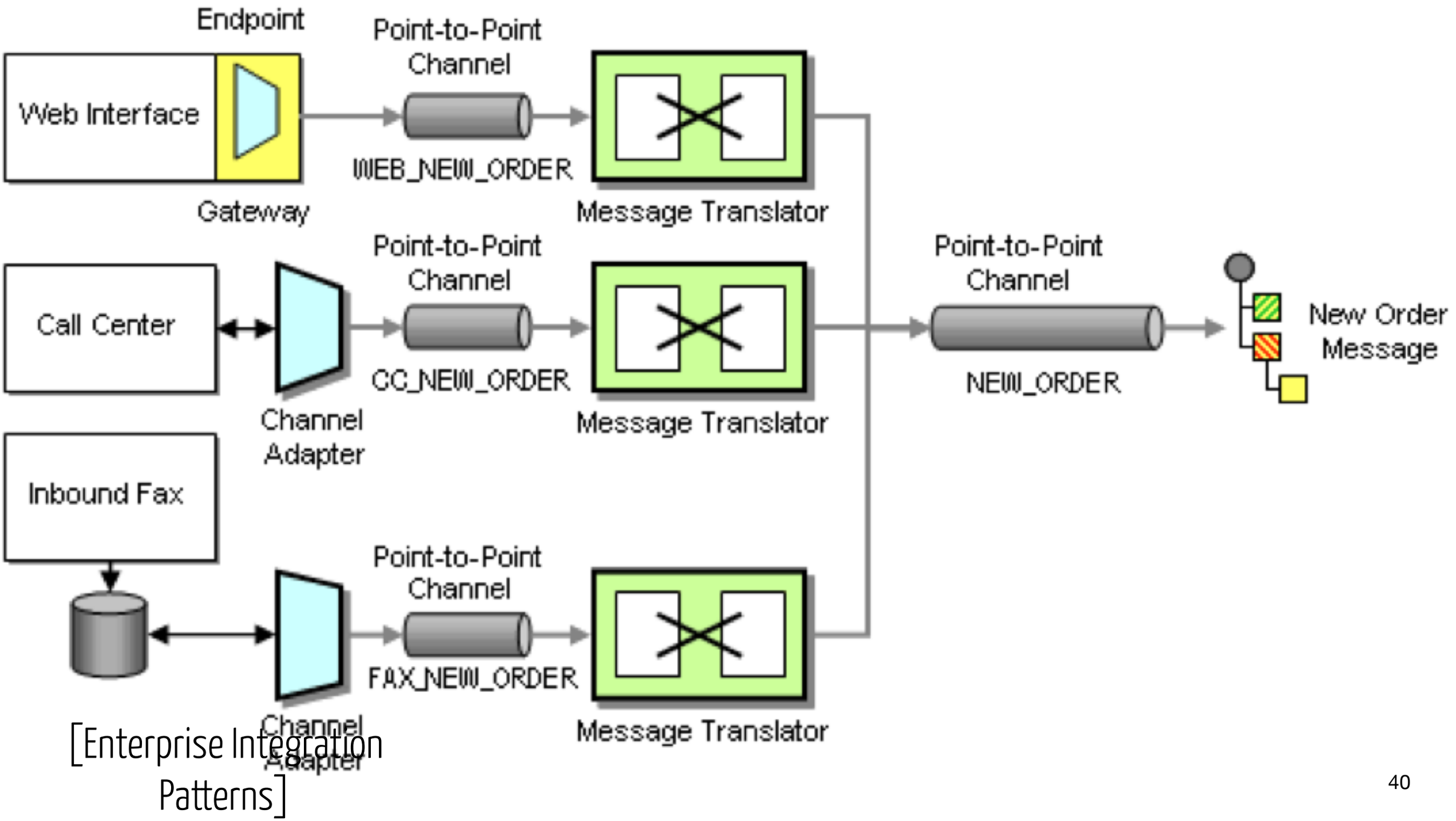






[Enterprise Integration Patterns]

# Taking Orders




[Enterprise Integration Patterns]

# Processing Orders [1/3]

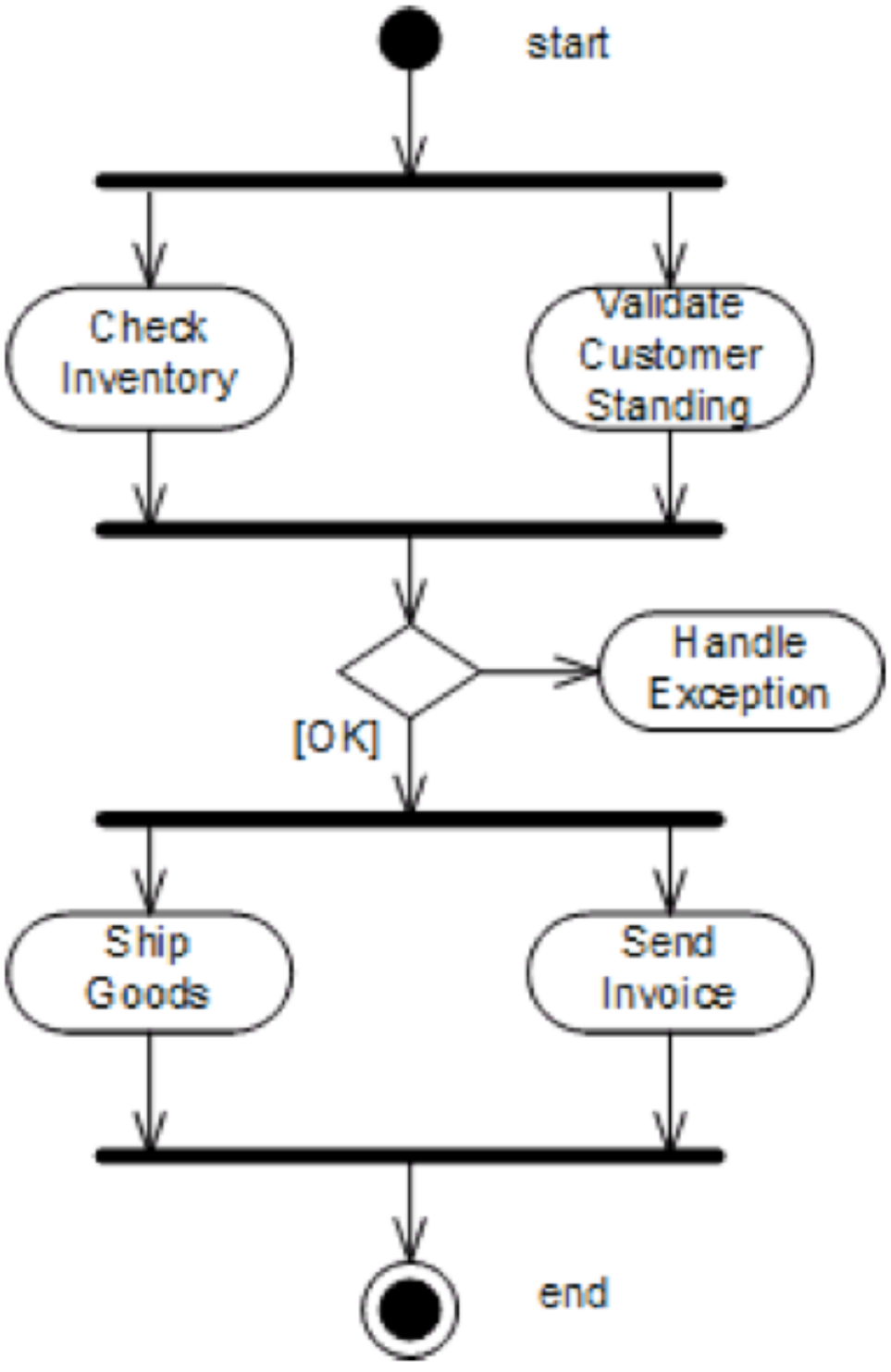
- Pre-check:
  - Verify Customer Credit
  - Verify the Inventory
- Processing:
  - Billing the customer
  - Sending the goods

[Enterprise Integration Patterns]

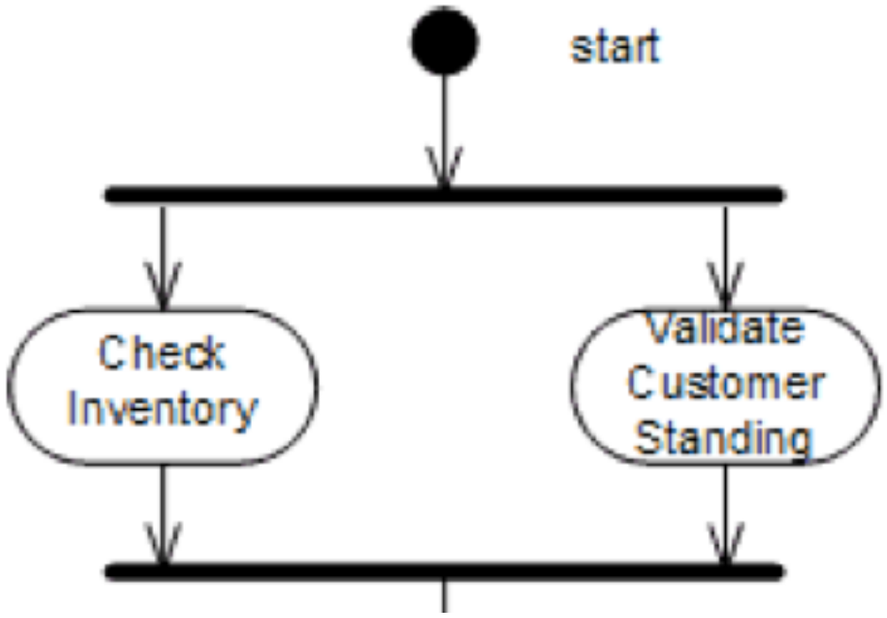
# Processing Orders [1/3]

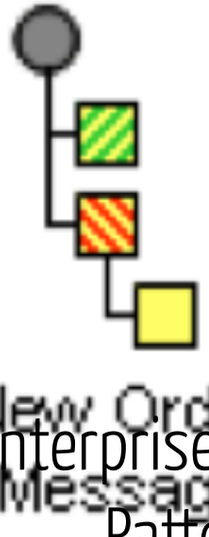
- Pre-check:
  - Verify Customer Credit
  - Verify the Inventory
- Processing:
  - Billing the customer
  - Sending the goods

[Enterprise Integration Patterns]



# Processing Orders [2/3]

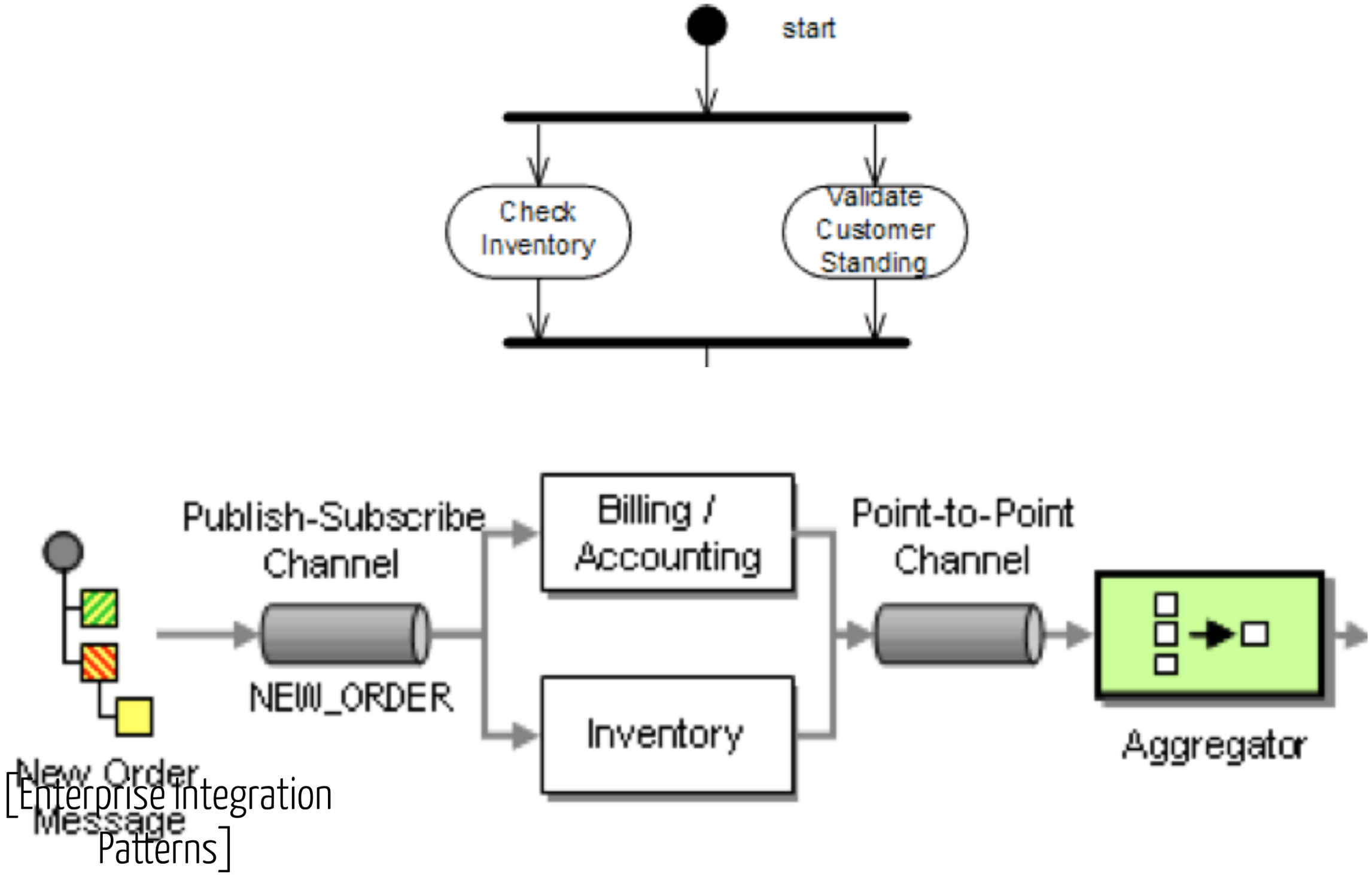



[Enterprise Integration Patterns]

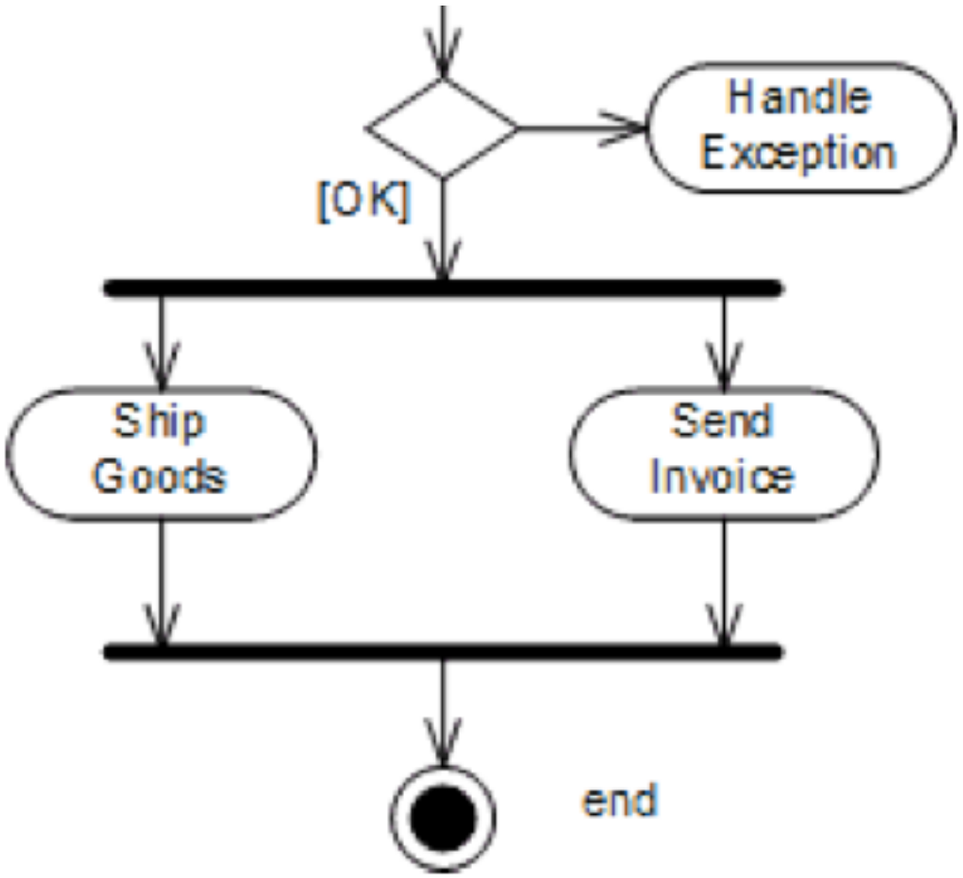
# Processing Orders

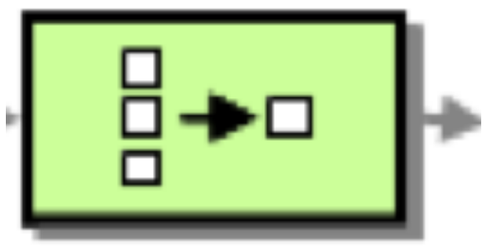





[Enterprise Integration Patterns]

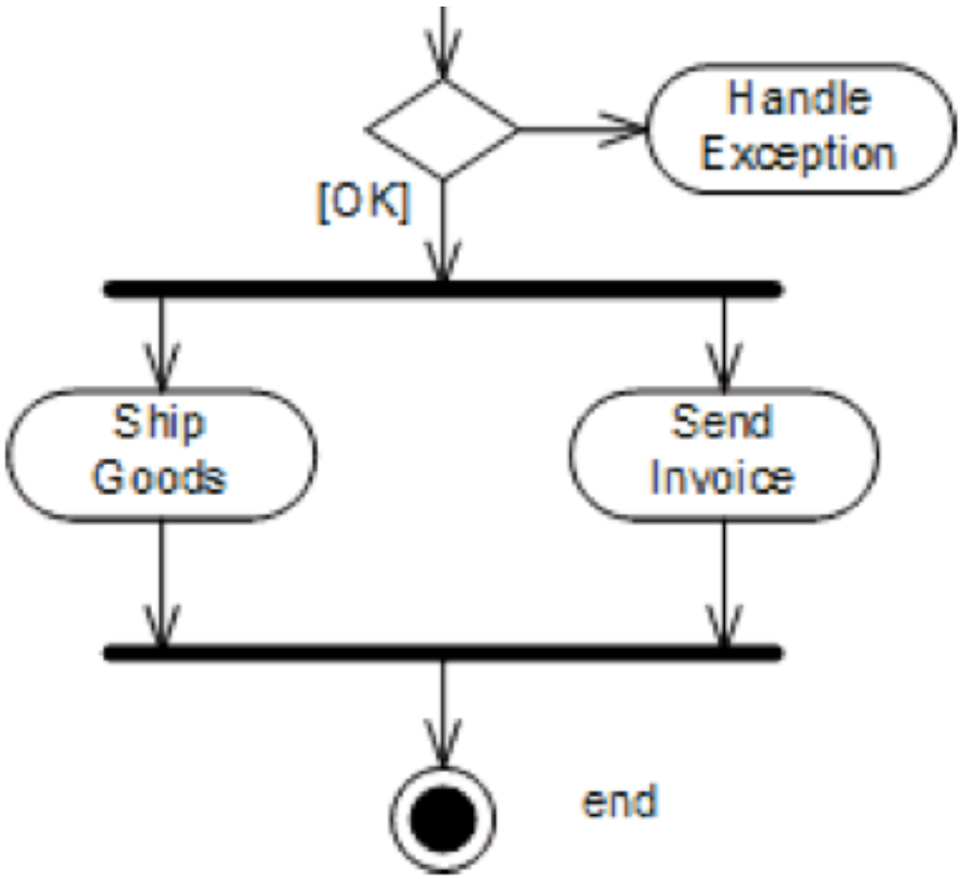
# Processing Orders [3/3]

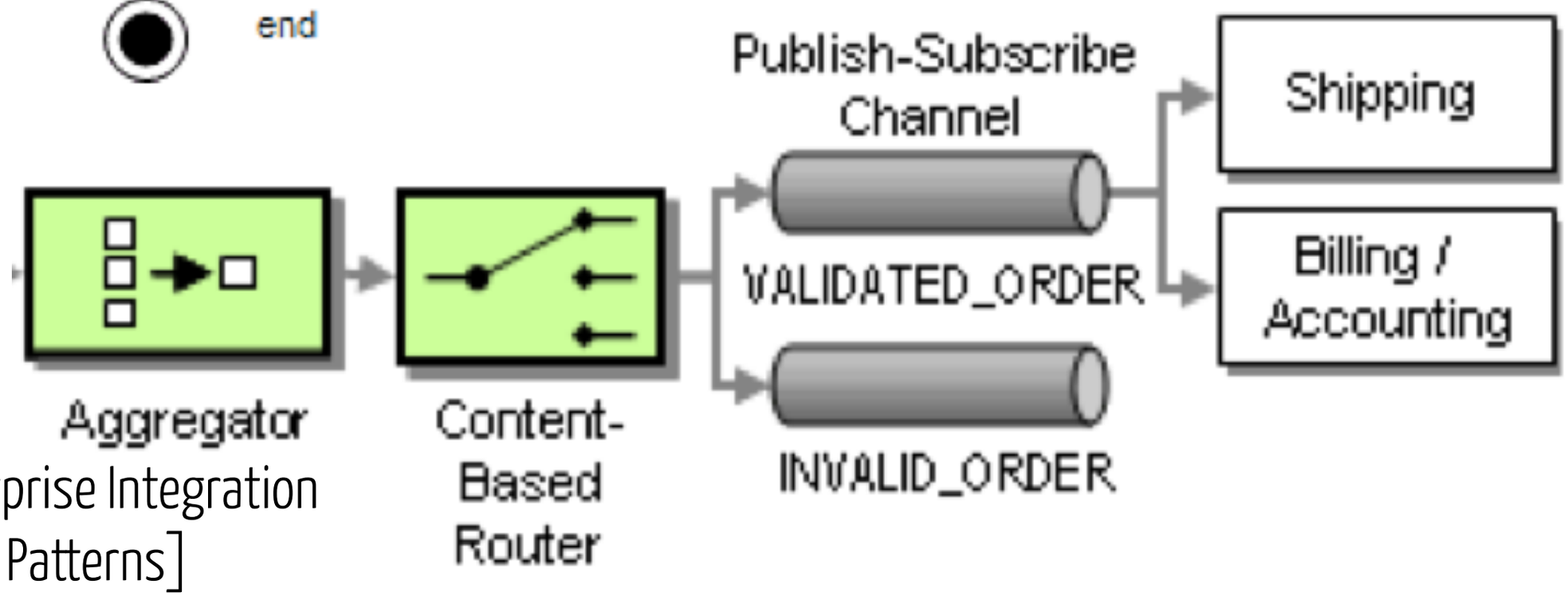



[Enterprise Integration Patterns]

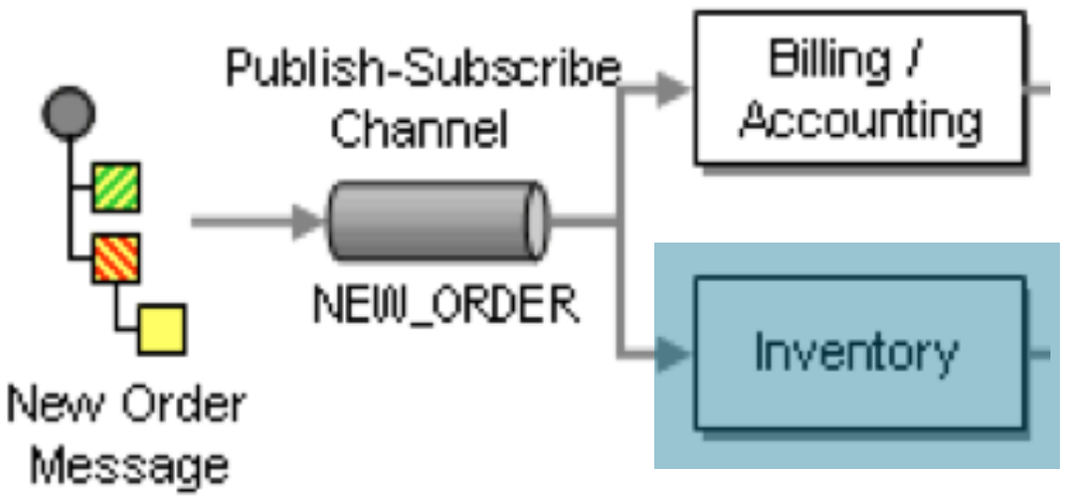
# Processing Orders [3/3]



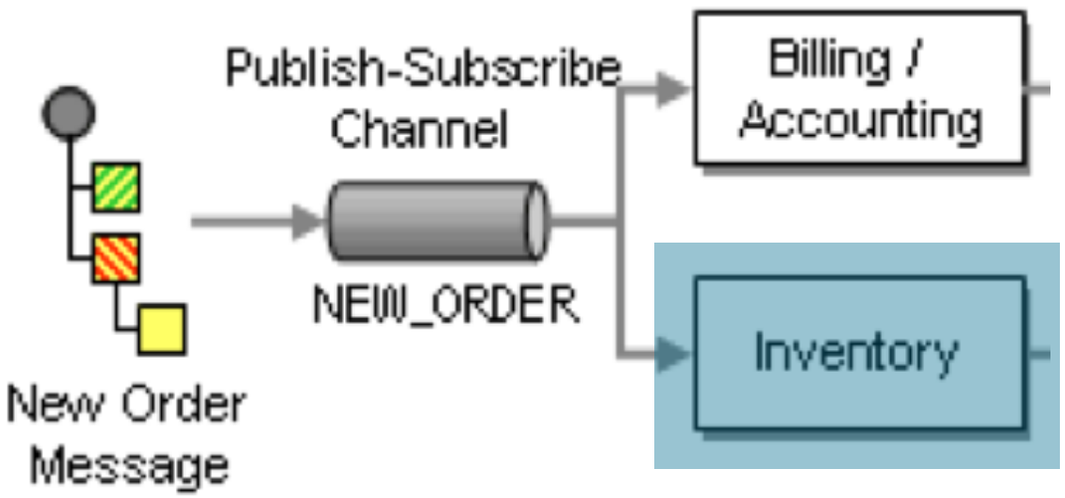

[Enterprise Integration Patterns]

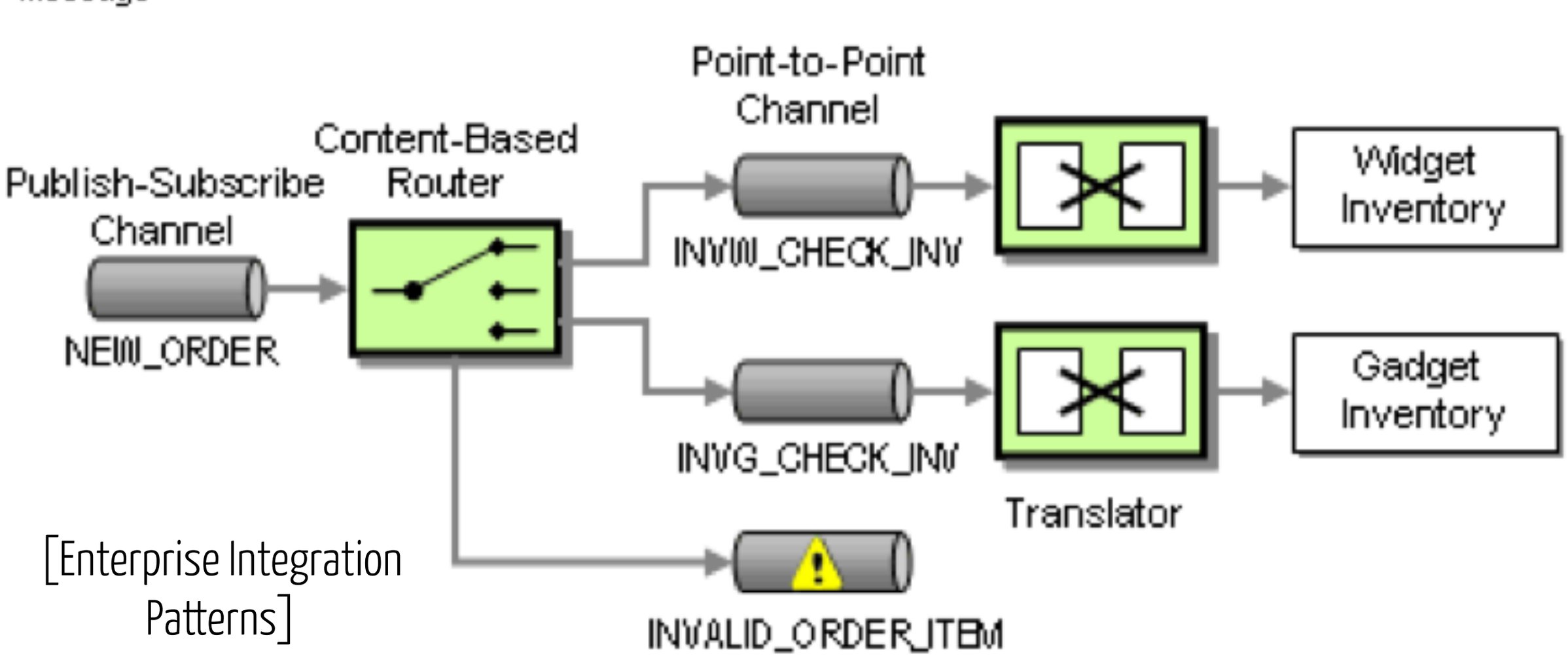
# Routing the **Inventory** Request



[Enterprise Integration Patterns]

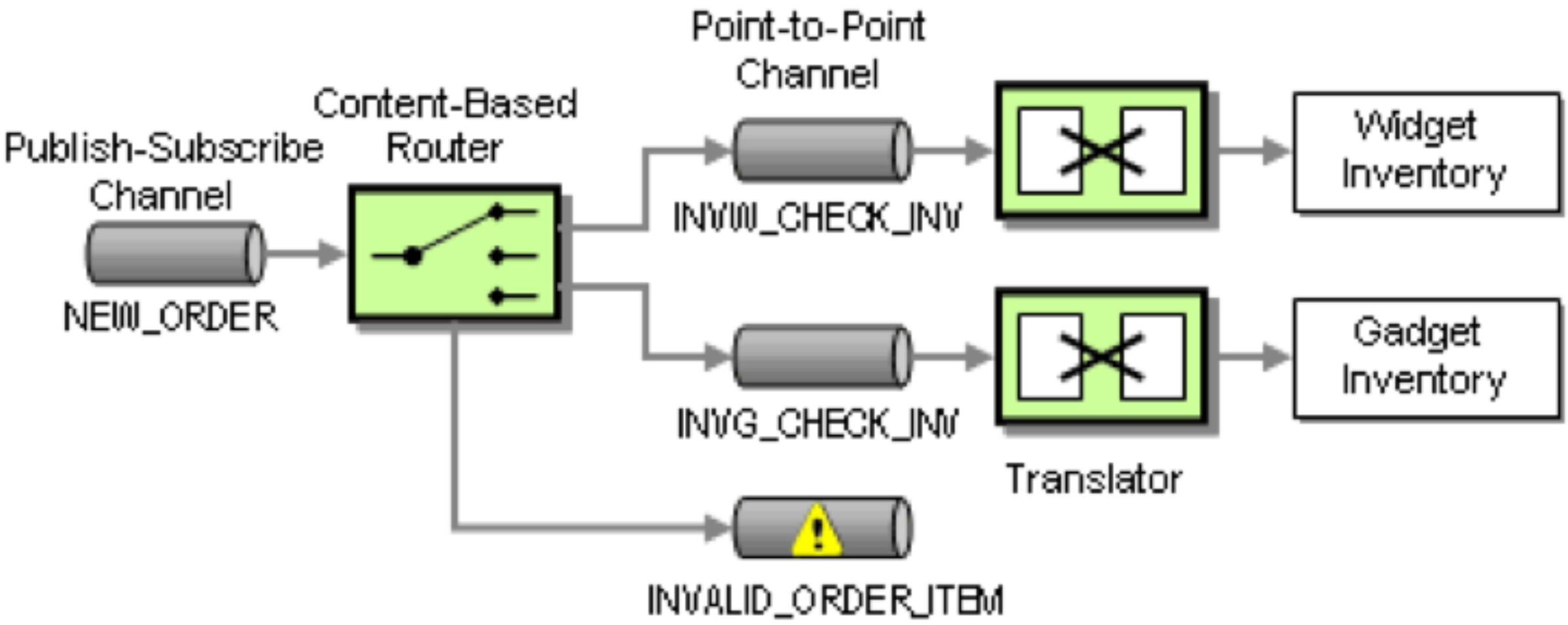
# Routing the **Inventory** Request



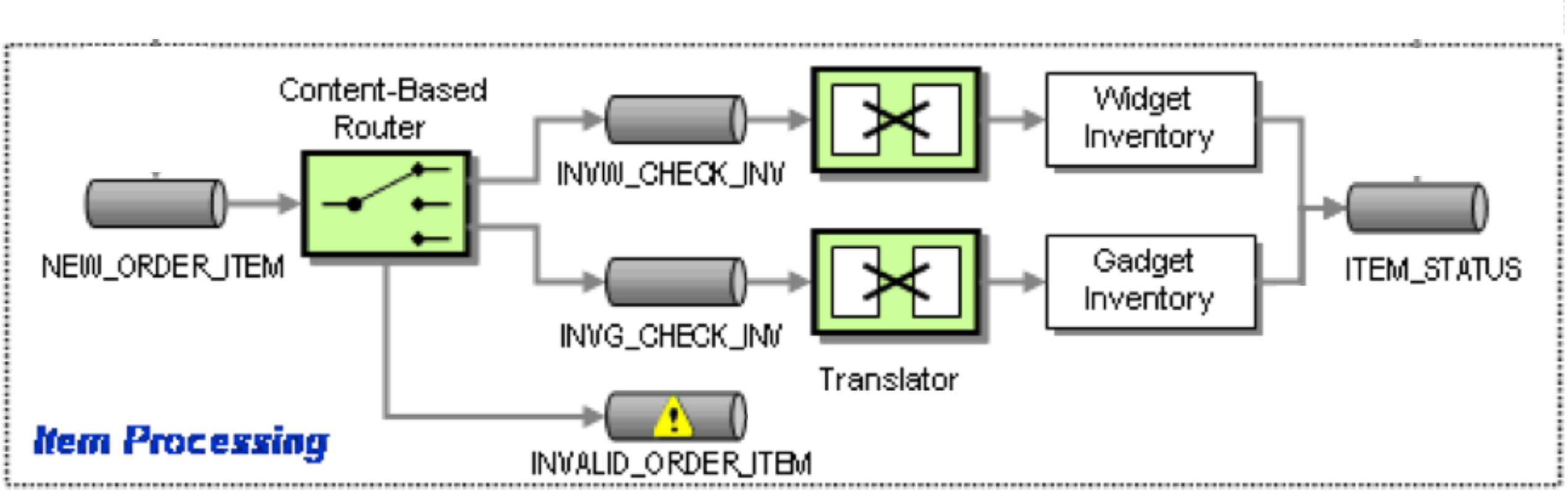

[Enterprise Integration Patterns]

# Problem: **multi-items** orders?



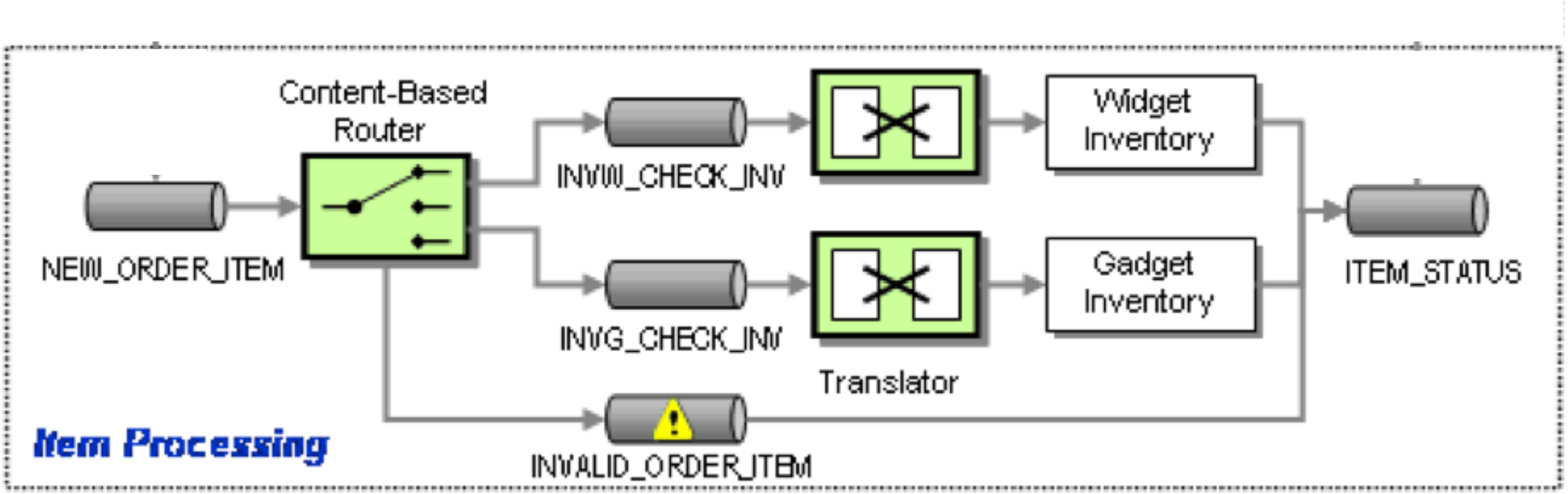

[Enterprise Integration Patterns]

# **Dataflow**: Split / Aggregate, Invalid



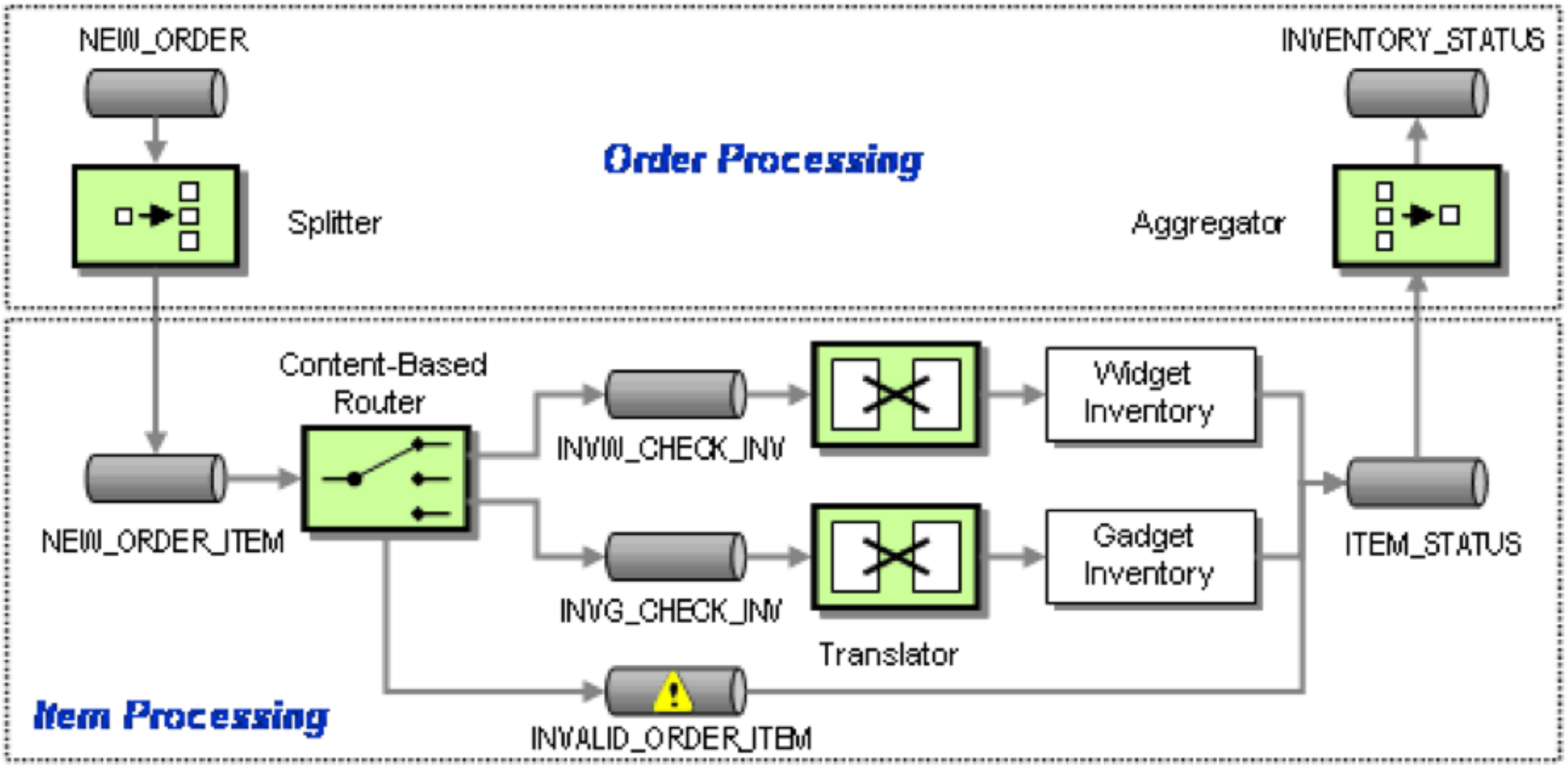

[Enterprise Integration Patterns]

# **Dataflow**: Split / Aggregate, Invalid



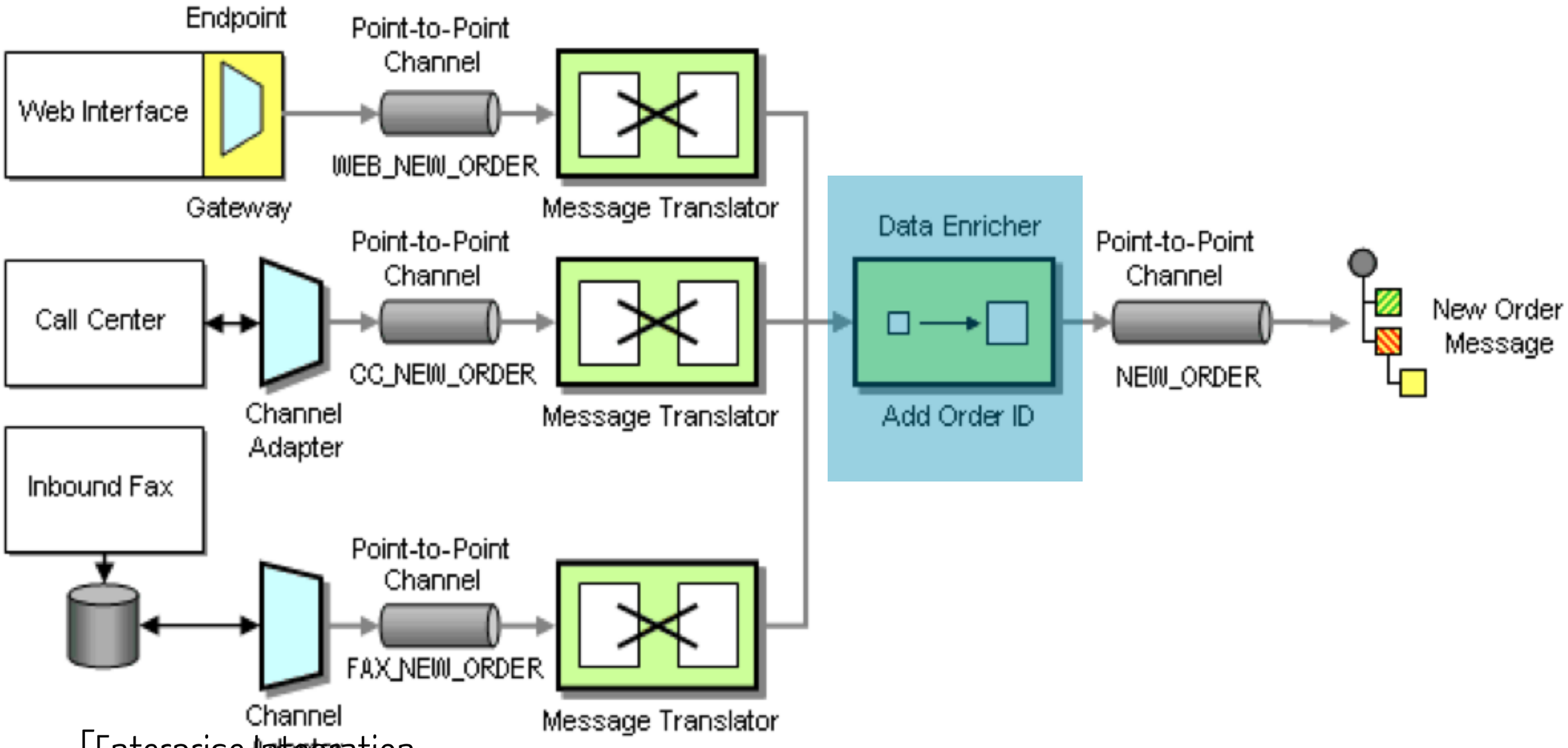

[Enterprise Integration Patterns]

# **Dataflow**: Split / Aggregate, Invalid



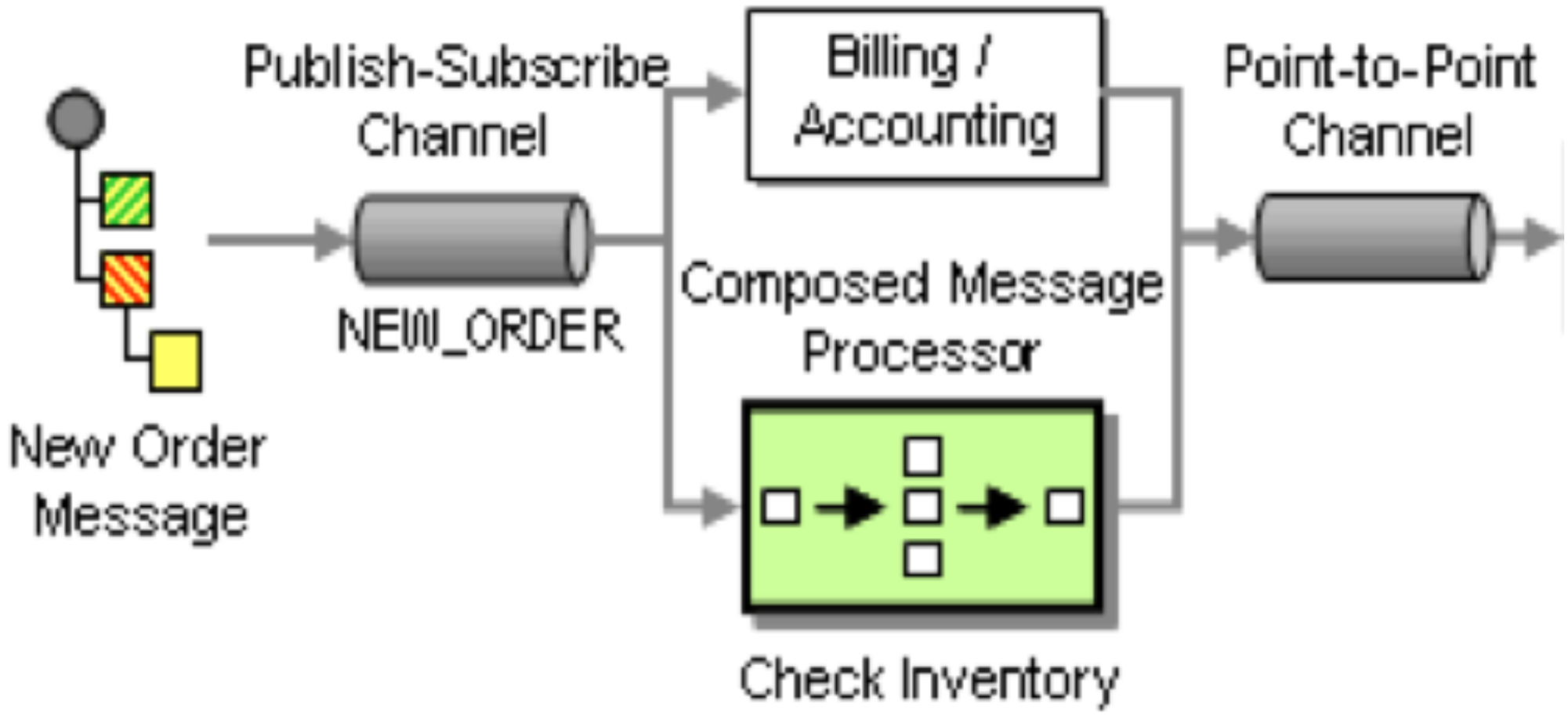

[Enterprise Integration Patterns]

# How to Aggregate?



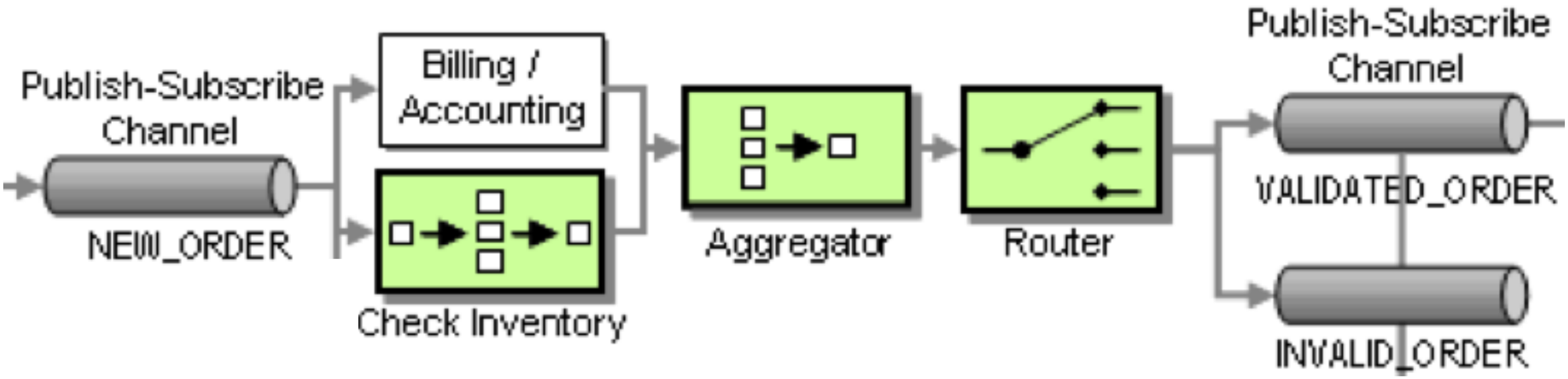

[Enterprise Integration Patterns]

# **Pattern**: Split + Process + Aggregate



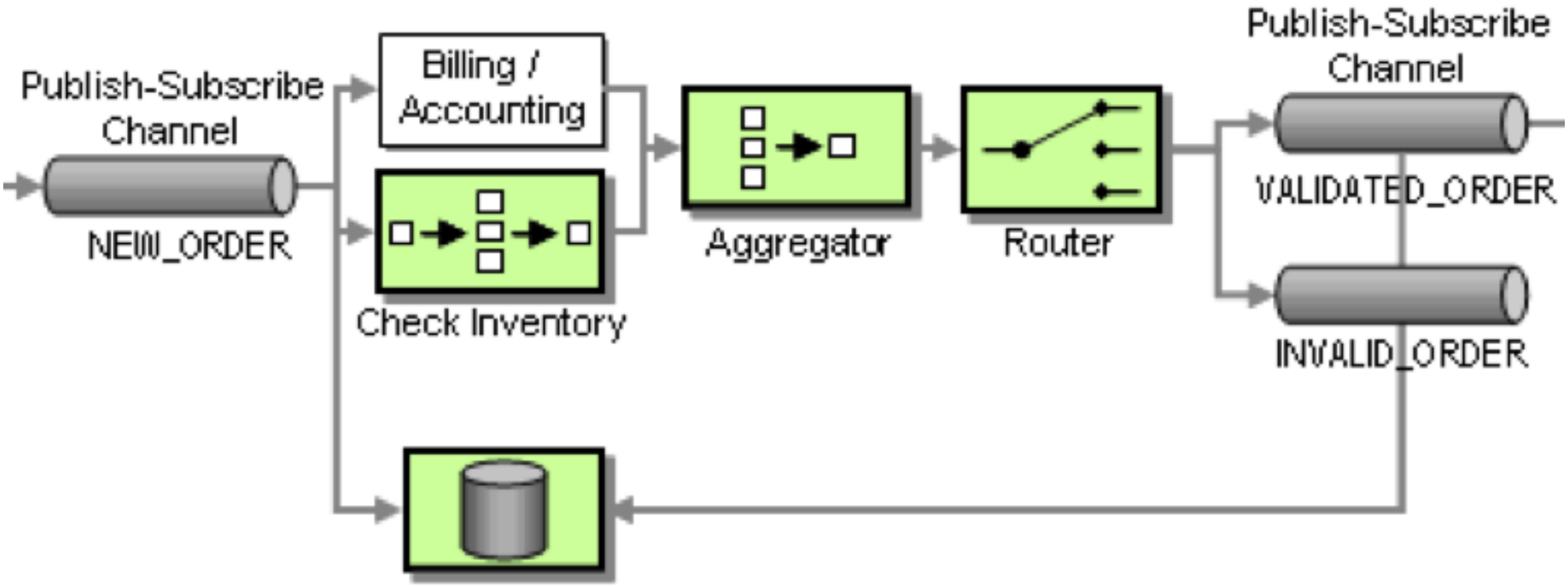

[Enterprise Integration Patterns]

# Checking Status




[Enterprise Integration Patterns]

# Checking Status




[Enterprise Integration Patterns]