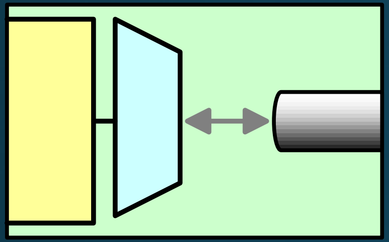


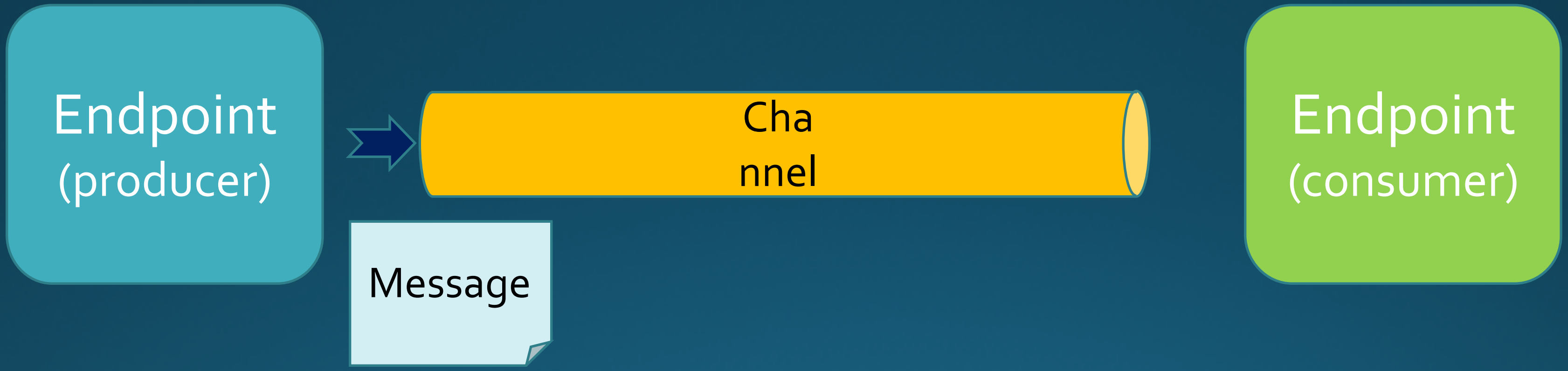
Adapters

Express Spring Integration

The labs and additional learning tools for this tutorial can be found on:
Intertech's Blog

Spring Integration Review

The main components



Endpoint (producer)

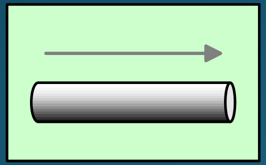
Cha nnel

Message

Endpoint (consumer)

Message Channel Review

- Two general classifications of message channels
  - **Pollable Channel**
    - May buffer messages
    - Consumers actively poll to receive messages
    - Only one receiver of a message in the channel
  - **Subscribable Channel**
    - Messages are delivered to all registered subscribers on message arrival
    - Doesn't buffer its messages
- Message Channels are represented by this icon in EIP diagrams

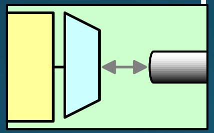


Message Endpoints

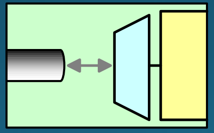
- Adapters (connect your channel to some other system)
- Filter (remove some messages from channels based on header, content, etc.)
- Transformer (convert a message content or structure)
- Enricher (add content to the message header or payload)
- Service activator (invoke service operations based on the arrival of a message)
- Gateway (connect your channels without SI coupling)

Adapters (aka Channel Adapters)

- An endpoint that connects a channel to an external system.
  - It “provide the bridge between integration framework and the external systems and services”[1]
  - Providing separation of the messaging concerns from the transports and protocols used.
- Adapters are inbound or outbound.
  - Those that bring messages into the SI channels.
  - Those that get messages from SI channels to the outside applications, databases, etc.
- Adapters are represented by the following icons in EIP diagrams



Inbound



Outbound

[1] – Pro Spring Integration, Lui et. al, Apress 2011.

Spring Integration Adapters

- Spring Integration comes with a number of built-in adapters
  - Stream adapters (like Standard Input and Output stream adapters)
  - File Adapters
  - JMS Adapters
  - JDBC & JPA Adapters
  - FTP and Secure FTP (SFTP) Adapters
  - Feed (RSS, Atom, etc.) Adapters
  - Mail Adapters
  - MongoDB Adapters
  - UDP Adapters
  - Twitter Adapter
- Use of the Adapter may (usually does) require the addition of another SI module

Example Adapter – JMS Adapters

- A JMS Inbound Adapter
  - Takes messages from a message Queue (via JMS under the covers) and gets it to a SI channel
  - Needs a JMS connection factory and queue (configuration not shown)
  - JMS channel adapters **are part of SI's JMS module** (int-jms namespace)
  - Note that the adapter pulls the messages into the channel at the poll rate

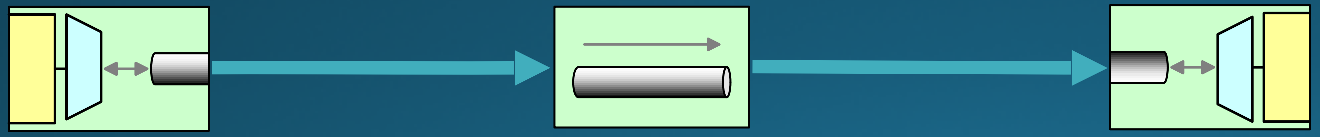
```
<int-jms:inbound-channel-adapter id="my-inbound-jms-adapter"
      connection-factory="jmsQueueConnectionFactorySecured" destination="in.message.queue.name"
      channel="my-message-channel">
      <int:poller fixed-rate="3000" />  <!-- time in milliseconds -->
</int-jms:inbound-channel-adapter>
```

Example Adapter – JMS Adapters

- A JMS Outbound Adapter
  - Takes messages from a message channel and delivers it to a message Queue (via JMS under the covers)
  - Also, needs a JMS queue

```
<int-jms:outbound-channel-adapter id="my-outbound-jms-adapter"
          destination="out.message.queue.name" channel="my-message-channel" />
```

- An EIP diagram representing the components in this example



my-inbound-jms-adapter

my-message-channel

my-outbound-jms-adapter

You are ready to tackle Lab 2

- Lab 2 is about getting to know and understand Spring Integration File adapters
- You also learn how to add SI modules to your Eclipse project, and making the module’s components available to your Spring XML configuration via namespace addition.

If you’re viewing this presentation on SlideShare you can click the link below to go to the page with links to the second lab:

Lab 2 - Adapters