**Use cases of Publisher Subscriber Model**

1. **Async Workflow**

Suppose we have an e-commerce website.

When a customer places an order, an event is published “orderPlaced”, to which multiple subscribers might be subscribed to, like:  
a. Generate Invoice

b. Send order confirmation

c. Initiate seller workflow

d. Initiate packaging

This tasks would take place simultaneously in background without keeping user in waiting state, hence this workflow is asynchronous.

In such scenarios, Pub Sub pattern is used.

1. **Decoupling**

Suppose, a company wants to perform analytics based on user’s interaction on Ecommerce website, items he liked, added to wish list, ignored items, etc.

These actions will generate multiple events.  
These events are then stored on DB and Data Analytics team builds insights from it.

Now we can’t keep user waiting, till whole analysis is done.

So, all events triggered by user action are queued in message queue and consumed in backend one by one, not holding the UI, doing tasks asynchronously.

In such cases, Pub Sub can be used.

1. **Load Balancing**

When there are multiple actions performed by user and put in message queue, no. of subscribers can also be increased to cater all requirements.

If subscribers are limited, the events will still remain in queue, until a subscriber gets free.

1. **Deferred Processing**

Suppose we need to perform some action like shifting data from local DB to S3, this is usually done in non-peak hours, when there are very few users. In that case, subscribers can be programmed to pick up an item from message queue at scheduled time.

1. **Data Streaming**

Pub sub pattern proved scaling of publishers as well as subscribers, hence they can be used in smooth Data Streaming.

**When Not To Use Pub Sub Pattern**

1. When user base is small, then don’t implement such architecture as it will unnecessarily consume resources to maintain this infrastructure hence might turn costly for small scale systems.
2. Do not use in case of Real Time systems, as most of the tasks are done asynchronously hence, it might give a lag.