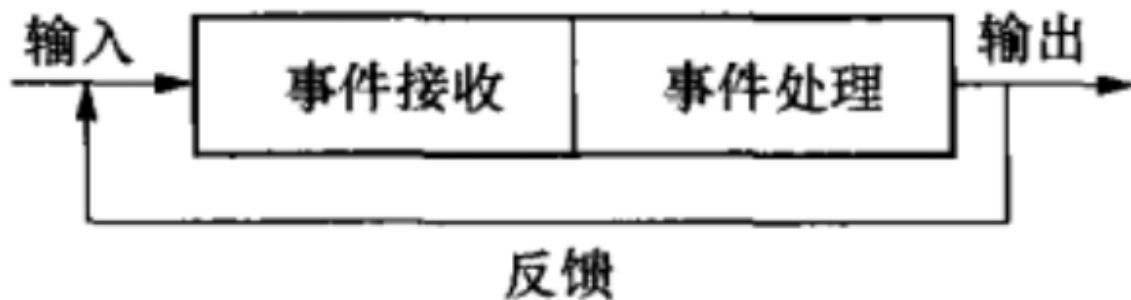


刘鑫宇

学习笔记：事件驱动 体系结构风格

1.1.1主要原理图



1.1.2原理图描述：

事件驱动就是在当前系统的基础之上，根据事件声明和发展状况来驱动整个应用程序运行。

事件驱动体系结构的基本思想是：系统对外部的行为表现可以通过它对事件的处理来实现。在这种体系结构中，构件不再直接调用过程，而是声明事件。系统其他构件的过程可以在这些事件中进行注册。当触发一个事件的时候，系统会自动调用这个事件中注册的所有过程。因此，触发一个事件会引起其他构件的过程调用。

1.1.3优缺点：

优点：

1. 事件声明者不需要知道哪些构件会响应事
2. 提高了软件复用能力
3. 便于系统升级

缺点：

1. 构件放弃了对计算的控制权，完全由系统来决定
2. 存在数据·传输问题

1.1.4历史背景和特点：

事件驱动体系结构风格的历史背景可以追溯到计算机科学的早期。该风格的核心思想是将系统的功能划分为一系列相互独立的事件，系统中的组件可以通过发布和订阅事件的方式进行通信和协作。

事件驱动体系结构的历史发展与许多相关技术和概念的出现密切相关，包括发布/订阅模式、观察者模式、消息队列和消息传递等。以下是事件驱动体系结构的历史背景和由来的一些重要里程碑：

1. 发布/订阅模式：该模式最早出现在1960年代，它将事件的生成者（发布者）与事件的消费者（订阅者）解耦，使得系统的各个组件可以独立地发送和接收事件。
2. 观察者模式：观察者模式是一种经典的设计模式，它于1994年由GoF（Gang of Four）提出。该模式通过定义一对多的依赖关系，使得当一个对象的状态发生变化时，所有依赖于它的对象都会收到通知。
3. 消息队列：消息队列是一种在系统组件之间传递消息的机制。它可以提供异步通信、解耦和缓冲的特性，使得事件的发送者和接收者可以独立运行，并且不需要直接的依赖关系。
4. 事件驱动架构（EDA）：事件驱动架构的概念最早于1992年由Gartner提出。EDA强调将企业应用系统建模为一系列的业务事件和事件处理器，以实现系统的松耦合、可扩展和灵活性。

1.1.5实际应用场景

分布式系统：事件驱动体系结构适用于构建分布式系统，其中各个组件可以通过发布和订阅事件的方式进行通信，实现松耦合和可扩展性。

企业应用：事件驱动体系结构可以应用于企业应用系统中，例如基于事件的业务流程管理、实时数据分析和响应等场景

1.1.6相关技术和工具

- 消息队列：用于事件的存储和传递，常见的消息队列包括Kafka、RabbitMQ等。
- 事件驱动框架：提供事件处理、订阅和发布等功能的框架，例如Spring Cloud Stream、Apache Camel等。
- 分布式系统技术：事件驱动体系结构可以与分布式系统技术相结合，例如微服务架构、容器等。

1.1.7参考文献:

1.Hohpe, G., & Woolf, B. (2003). Enterprise integration patterns: designing, building, and deploying messaging solutions. Addison-Wesley.