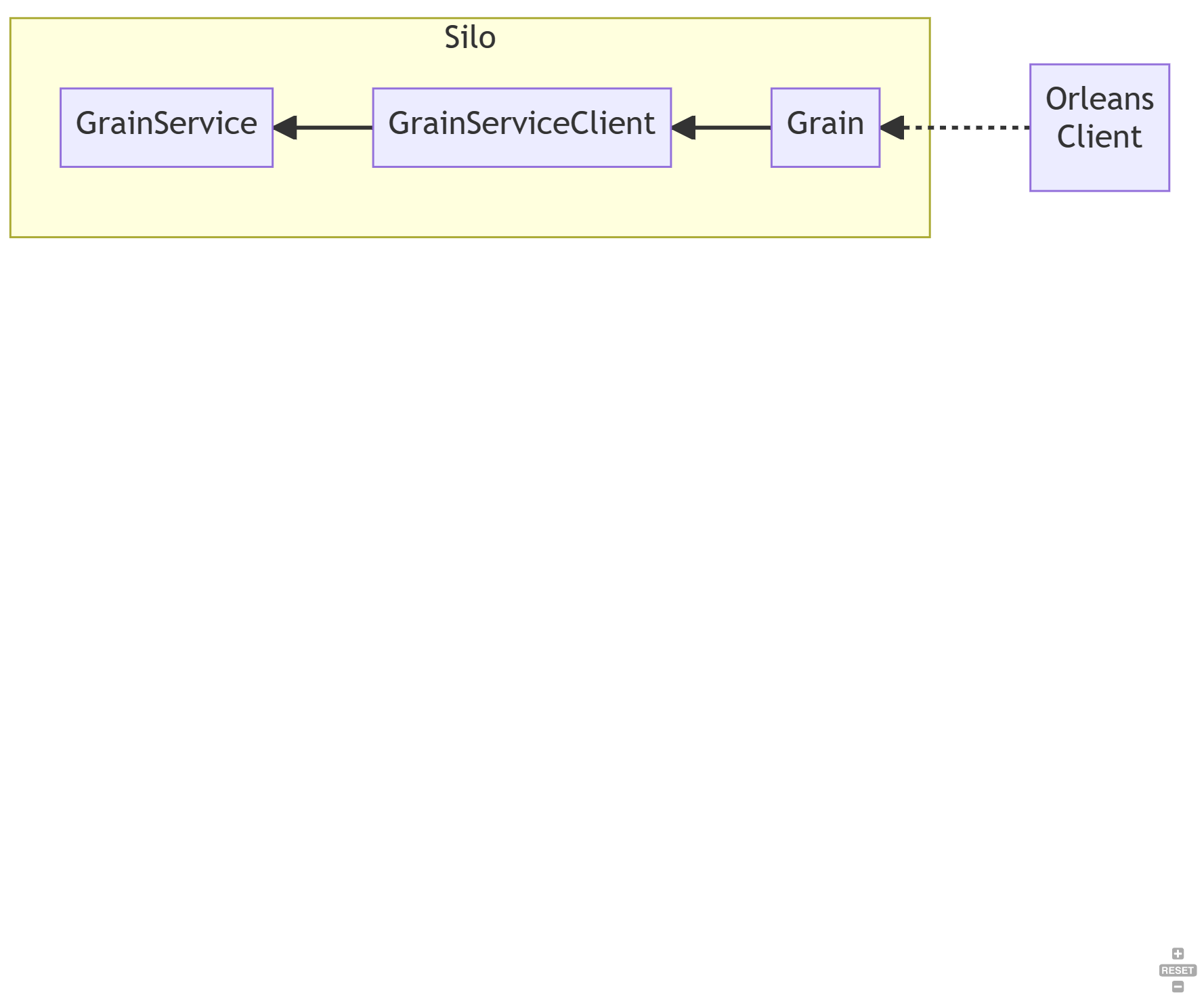
# Orleans的常駐服務：GrainService

Orleans的GrainService是不同於Actor model運算模式的特殊System daemon，它可以用來在每個Silo提供類似系統服務的常駐程式。

## GrainService介紹

* GrainService不需要定義Grain識別子(Identifier)，因為它不是一般Grain的物件實例，所以其RPC介面宣告需繼承自 [IGrainService](https://learn.microsoft.com/dotnet/api/orleans.services.igrainservice) 介面，而非原本Grain的那五種： IGrainWithIntegerKey 、 IGrainWithStringKey 、 IGrainWithGuidKey IGrainWithIntegerCompoundKey 、 IGrainWithGuidCompoundKey RPC介面的基礎介面型別。
* GrainService的RPC實作不繼承自Grain，而是繼承自 [GrainService](https://learn.microsoft.com/dotnet/api/orleans.runtime.grainservice)。
* GrainService的生命週期由Orleans管理，不需要開發者自行管理。要使GrainService能作用，得在Silo配置程式碼中使用：
  1. 將GrainService的類別使用DI依賴注入註冊為Singleton。
  2. 呼叫SiloBuilder的 [AddGrainService()](https://learn.microsoft.com/dotnet/api/orleans.hosting.grainservicessilobuilderextensions.addgrainservice) 擴充方法來註冊GrainService。
* GrainService本身可使用DI依賴注入的方式取得 IGrainFactory 物件，以便取得其他Grain的RPC參考來呼叫其方法。
* GrainService無法直接被其他一般Grain呼叫，但可以透過撰寫繼承於GrainServiceClient的子類別Proxy物件去呼叫GrainService；然後把此呼叫GrainService的Proxy物件，使用Grain的建構子DI依賴注入的方式，讓Grain呼叫Proxy物件開放出來的方法，來呼叫GrainService。
* GrainService無法被Orleans Client端直接呼叫，如果需要的話，可以藉由定義一個中介Grain去呼叫上述步驟設計的DI依賴注入的GrainServiceClient物件的方式來呼叫：   
  這樣的方式來達成呼叫GrainService的目的。

官方對於GrainService的應用，就是拿來實現Grain Reminder的機制，IReminderService 介面就是繼承自 IGrainService 介面，而 LocalReminderService 就是繼承自 GrainService 類別：  
https://github.com/dotnet/orleans/blob/main/src/Orleans.Reminders/ReminderService/LocalReminderService.cs

## GrainService實作範例

可使用GrainService來記特定Grain RPC方法呼叫次數，步驟如下：