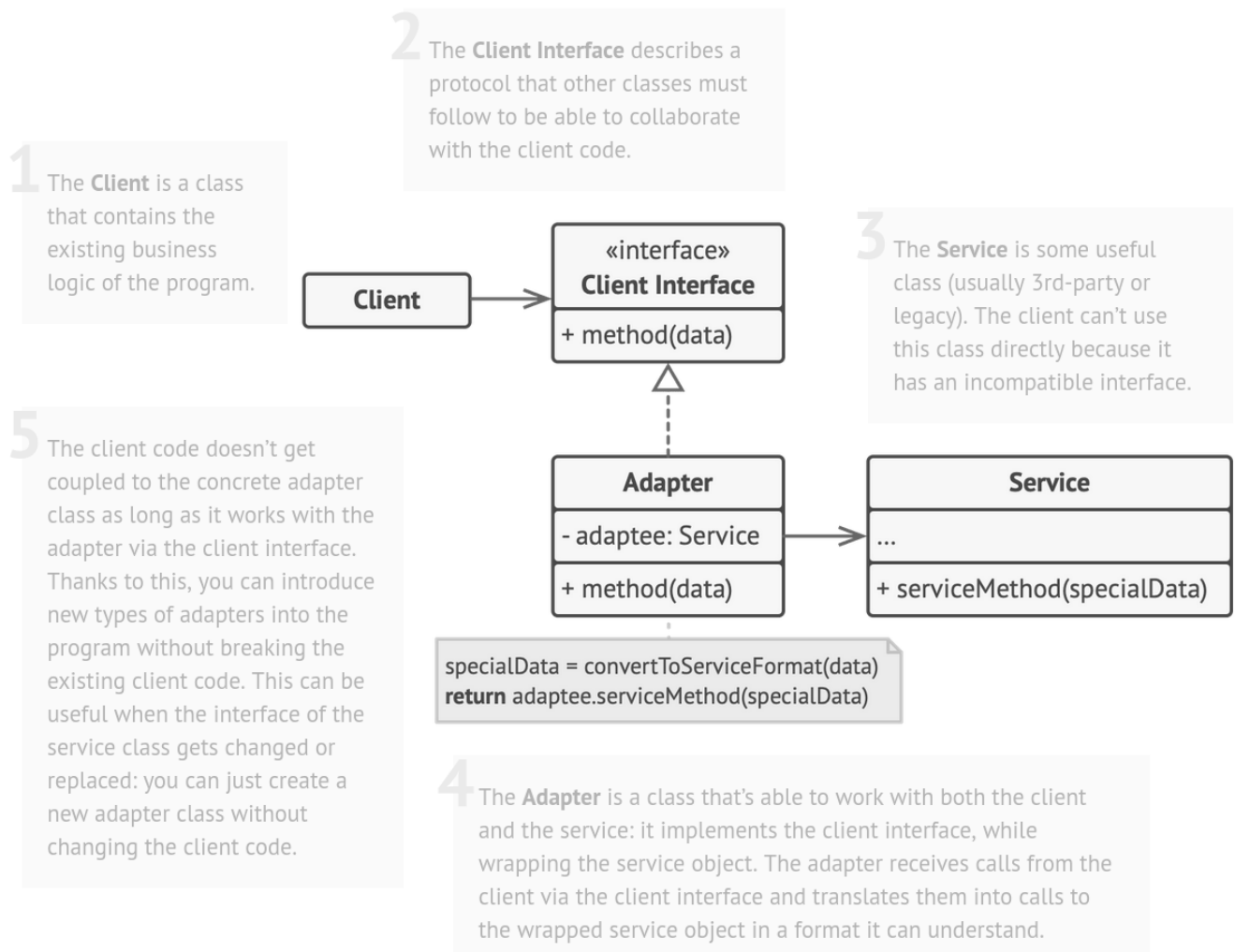


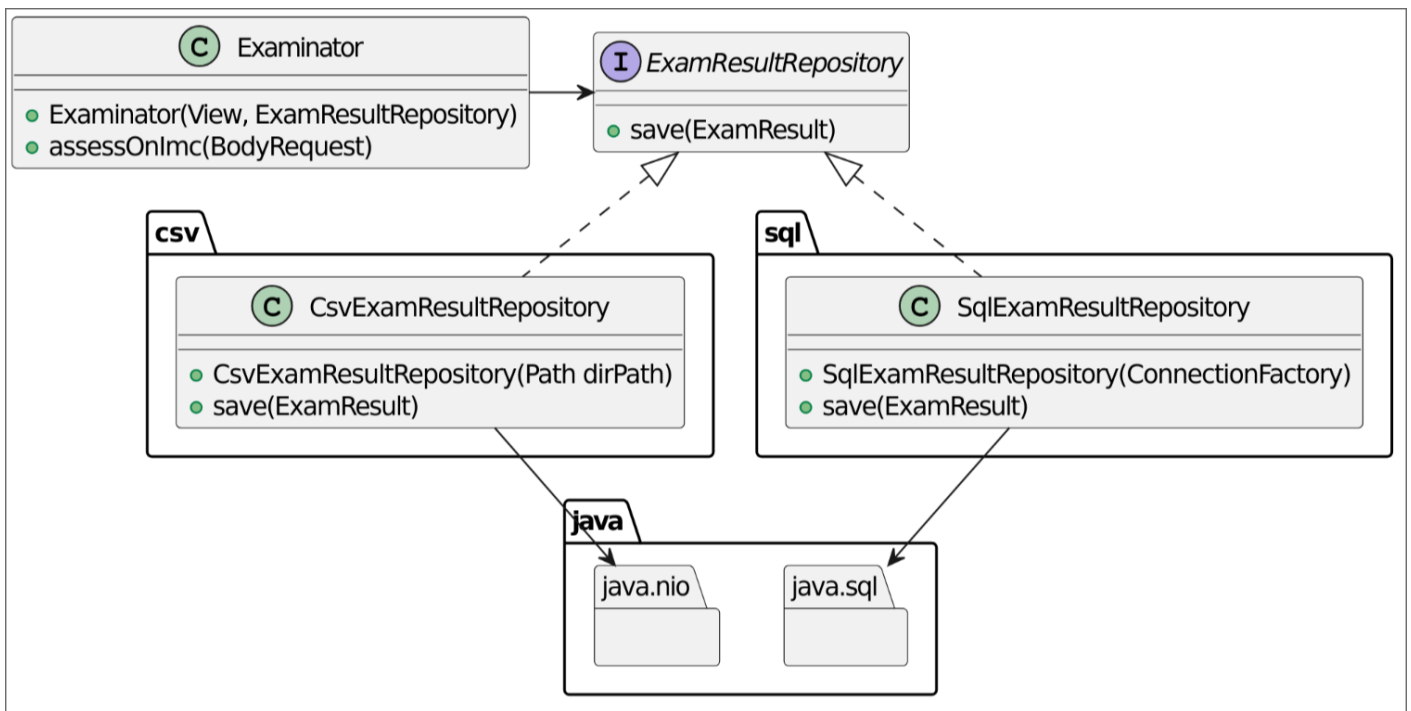
Adaptateur

Le patron de l'adaptateur sert à faire collaborer des objets qui ont une interface incompatible.



Le fonctionnement est que l'on crée des classes adaptateurs entre une interface et un service.

Voici un exemple :



Dans cet exemple, l'Examinator est notre client, il utilise l'interface "ExamResultRepository". Les classes `CsvExamResultRepository` et `SqlExamResultRepository` sont les adaptateurs qui implémentent l'interface.

Ces adaptateurs vont ensuite utiliser les services appropriés, dans le cas présent, les fonctions de `java.nio` et `java.sql`. Cela permet ainsi de séparer les préoccupations, de programmer avec des interfaces et d'éviter l'héritage.

Cependant ce patron a le désavantage que si il est utilisé de manière appropriée il peut complexifier le code, il faut donc d'abord se demander si il n'est pas plus simple de changer directement les services pour les faire utiliser une interface commune, plus-tôt que de créer des adaptateurs (voir [Conception et découverte du patron Strategie](#)). Dans le cas de l'exemple, les adaptateurs sont une bonne idée car on ne peut pas modifier les services qui font simplement partie du JDK.

Revision #1

Created 3 October 2023 21:03:12 by SnowCode

Updated 3 October 2023 21:07:06 by SnowCode