

Facade Pattern (auf Deutsch: **Fassaden-Entwurfsmuster**): bereitstellt eine **einfache Schnittstelle** zu einem komplexen System.

Es vereinfacht die Interaktion mit einem System, indem es die Komplexität hinter einer **Fassade** versteckt. Der Client muss sich nicht mehr um die Details kümmern, sondern verwendet nur die vereinfachte Schnittstelle.

Grundidee des Facade Patterns

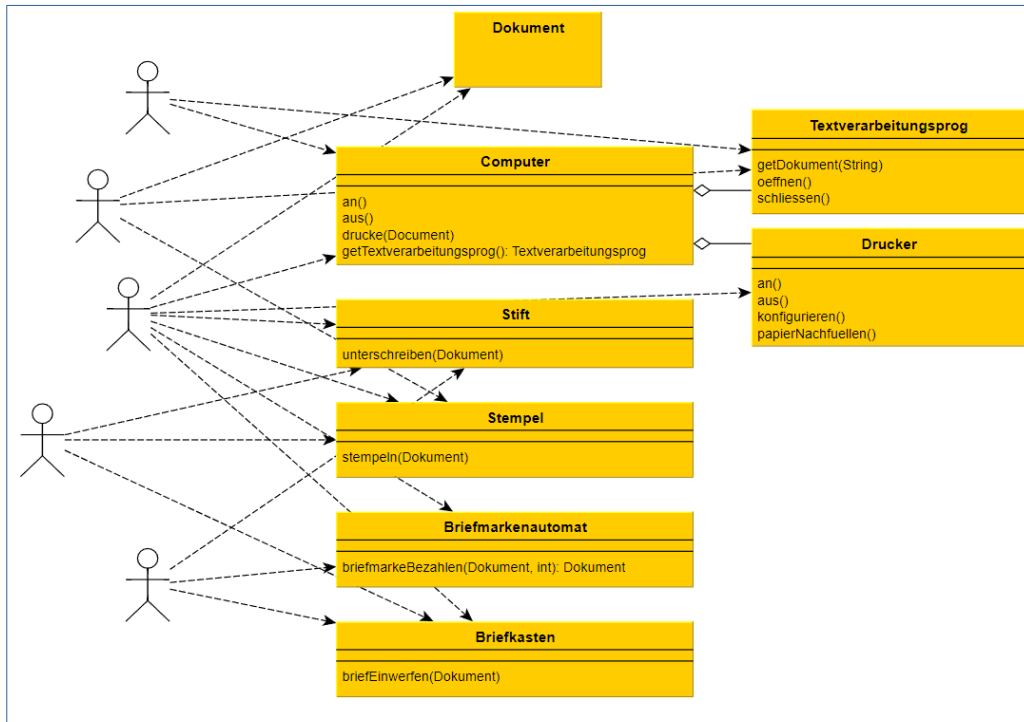
Das **Facade Pattern** stellt eine **vereinfachte Schnittstelle** zu einem komplexen System bereit. Es kapselt die Details und die Komplexität eines Systems und bietet eine **einfache Schnittstelle**, um häufig verwendete Funktionen auszuführen.

Wenn ein System viele Klassen und komplexe Abhängigkeiten hat, kann eine Fassade helfen, das System einfacher nutzbar zu machen, indem es **Hochabstraktionsmethoden** bereitstellt, die auf diese komplexen Klassen zugreifen und sie orchestrieren.

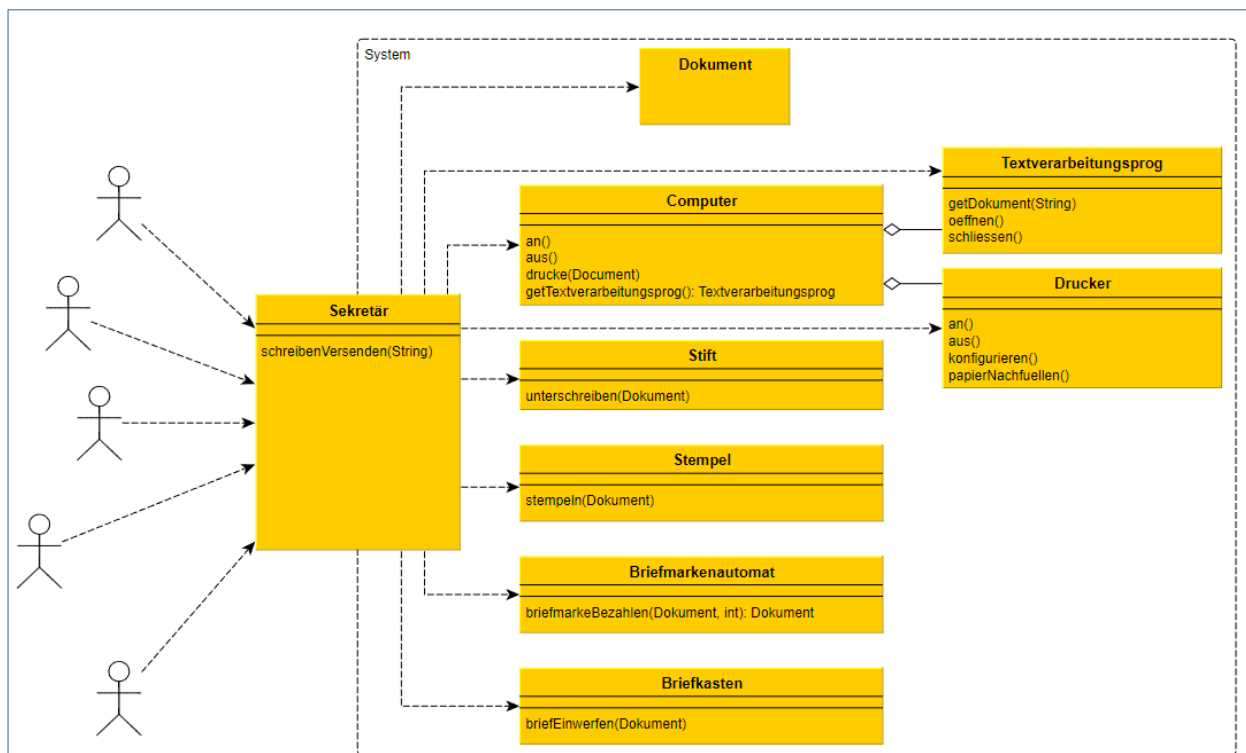
Wann verwendet man das Facade Pattern?

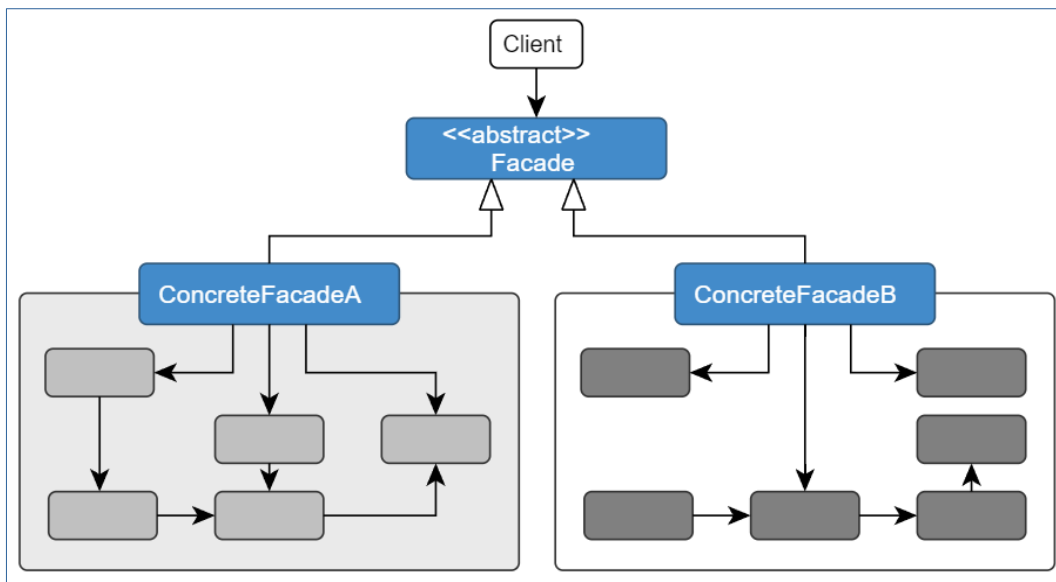
1. Wenn du den **Zugriff auf ein komplexes System** vereinfachen möchtest.
2. Wenn du eine **leichte Bedienung** für ein kompliziertes oder schwer zu verwendendes Subsystem bieten willst.
3. Wenn du den **Client vom komplexen System entkoppeln** möchtest, um den Code flexibler zu gestalten.
4. Wenn du eine **klare und standardisierte Schnittstelle** zu einer Bibliothek, einem Framework oder einem Subsystem bereitstellen willst.

Ohne Fassade:



Mit Fassaden:





Beispiel:

