

# Model-View-Controller-Paradigma/Model

Wechseln zu: [Navigation](#), [Suche](#)

Dieser Artikel erfüllt die [GlossarWiki-Qualitätsanforderungen](#) **nur teilweise**:

<b>Korrektheit:</b> 4 (größtenteils überprüft)	<b>Umfang:</b> 3 (einige wichtige Fakten fehlen)	<b>Quellenangaben:</b> 4 (fast vollständig vorhanden)	<b>Quellenarten:</b> 4 (sehr gut)	<b>Konformität:</b> 4 (sehr gut)
---	---	--	--------------------------------------	-------------------------------------

Diese Bewertungen beziehen sich auf alle im nachfolgenden Menü genannten Artikel gleichermaßen.

MVC-Paradigma:	Model (Data)   View   Controller	MVC-Pattern:	Singletons   Dependency Injection   Observers
MVCS-Paradigma:	Service	MVCS-Pattern:	Singletons   Dependency Injection   Observers
LDVCS-Paradigma:	Logik	VCLSD-Pattern:	Singletons   Dependency Injection   Observers

## Inhaltsverzeichnis

- [1 Definition \(Kowarschick \(MMProg\)\)](#)
- [2 Anmerkungen](#)
- [3 Quellen](#)
- [4 Siehe auch](#)

## 1 Definition (Kowarschick (MMProg))

Ein Modell (engl. [model](#)) einer [MVC](#)- oder [MVCS-Anwendung](#) bzw. ein Daten-Modul (engl. [data module](#)) einer [LDVCS-Anwendung](#) ist ein Modul zur Speicherung bestimmter Daten, d. h. zur Speicherung von Teilen des aktuellen Zustands der Anwendung.

Ein Modell bzw. ein Datenmodul kann weitere Aufgaben übernehmen:

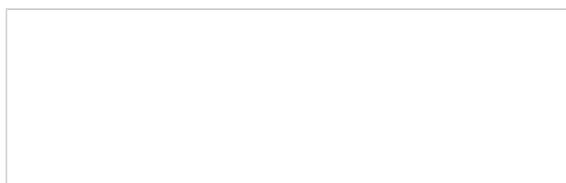
anderen Modulen Zugriff auf die Zustandsdaten gewähren  
andere Module über Änderungen informieren (meist mittels des [Observer-Patterns](#))

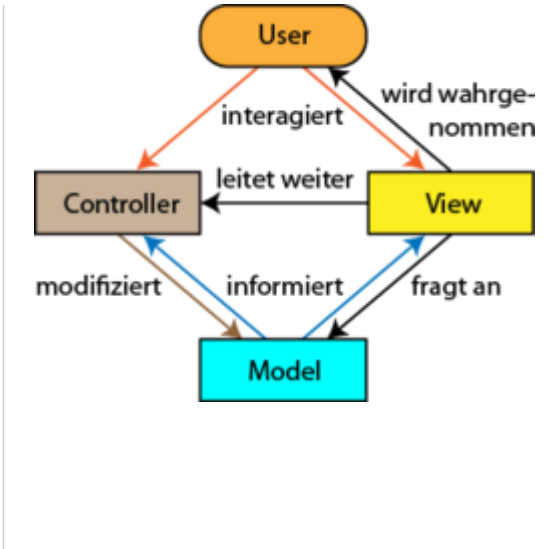
MVC- und MVCS-Modelle können darüber hinaus noch folgende Aufgabe übernehmen:

Umsetzung der Komponentenlogik

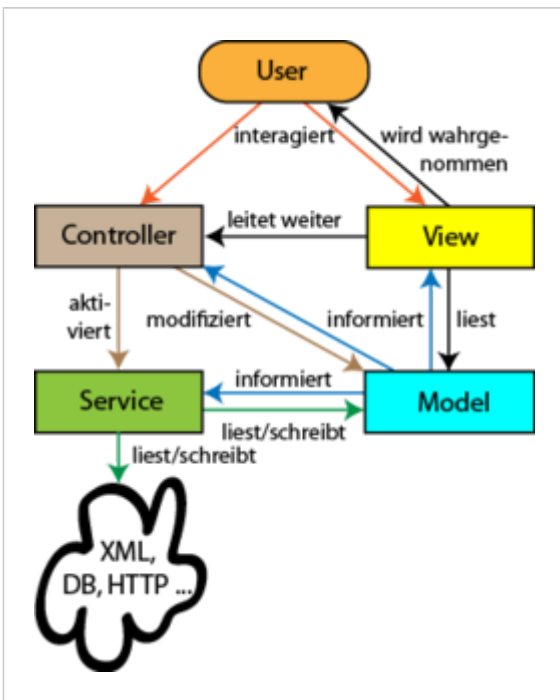
MVC-Modulen obliegt manchmal auch noch eine vierte Aufgabe:

Kommunikation mit externen Datenquellen (zum Zweck der Datensynchronisation)



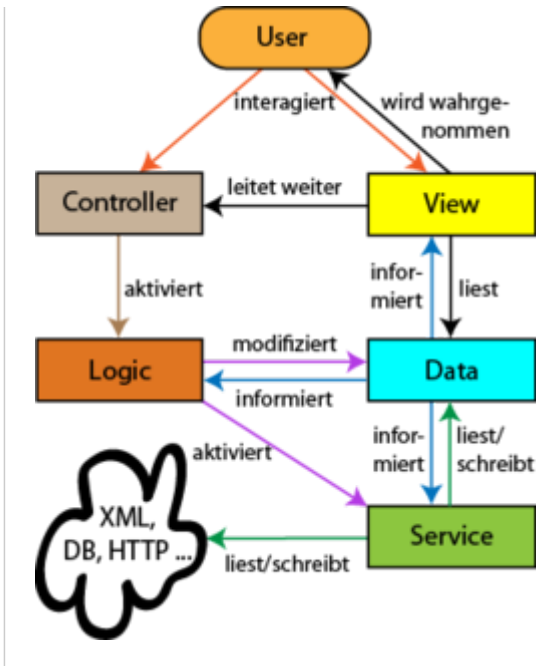


Der MVC-Prozess



Der MVCS-Prozess





Der LDVCS-Prozess

## 2 Anmerkungen

Modelle von MVC-Anwendungen, [Modelle von MVCS-Anwendungen](#) und [Daten-Module von LDVCS-Anwendungen](#) unterscheiden sich hinsichtlich der optionalen Aufgaben die sie wahrnehmen können.

MVC-Modelle können die Komponentenlogik umsetzen sowie mit externen Datenquellen kommunizieren.

MVCS-Modelle kommunizieren dagegen nie mit externen Datenquellen, dafür sind [MVCS-Services](#) zuständig. Sie können aber immer noch die Komponentenlogik realisieren.

VCLSD-Modelle gibt es nicht, sondern nur [LDVCS-Daten-Module](#). Daten-Module kommunizieren weder mit externen Datenquellen, noch realisieren sie die Komponentenlogik. Für diese Aufgaben gibt es zwei andere Module: [LDVCS-Logik-Module](#) und [LDVCS-Service-Module](#).

## 3 Quellen

1. **Kowarschick (MMProg)**: Wolfgang Kowarschick; Vorlesung „Multimedia-Programmierung“; Hochschule: [Hochschule Augsburg](#); Adresse: [Augsburg](#); [Web-Link](#); 2018; [Quellengüte](#): 3 (Vorlesung)

## 4 Siehe auch

[Service](#)

[Logic](#)

[Model-View-Controller-Paradigma](#) (insb. die dort angegebenen Quellen)

Kategorien:

[MVC](#)

Diese Seite wurde zuletzt am 22. September 2017 um 16:50 Uhr bearbeitet.

Inhalt verfügbar unter [CC BY-NC-SA 4.0](#), falls Dokument nach dem 5. 3. 2011 erstellt wurde, sonst [CC BY-SA DE 3.0](#).

