

Softvérové architektúry & komponentové programovanie

Sergej Chodarev, Michaela Bačíková · 2016

Softvérové inžinierstvo

Vývoj softvéru

- Nielen programovanie
- Pred programovaním
 - analýza, návrh
- ~~Po programovaní~~
 - vývoj prebieha počas celého života systému
- Súčasne s programovaním
 - testovanie
 - nasadenie

Ako zvládnuť zložitost
a zmeny?

Organizácia procesu

- Agilný vývoj softvéru
- Správa verzií
- Správa požiadaviek na zmeny a opravy (issue tracking)
- Automatizácia
 - testovania
 - nasadenia
 - infraštruktúry → cloud

Organizácia kódu

- Modularita
 - rozdelenie na časti (rekurzívne)
 - zjednodušenie zmien a vývoja
- Návrhové vzory
 - osvedčené postupy
- Komponenty
 - časti, ktoré je možné vyvíjať a nahradiť nezávisle od ostatných

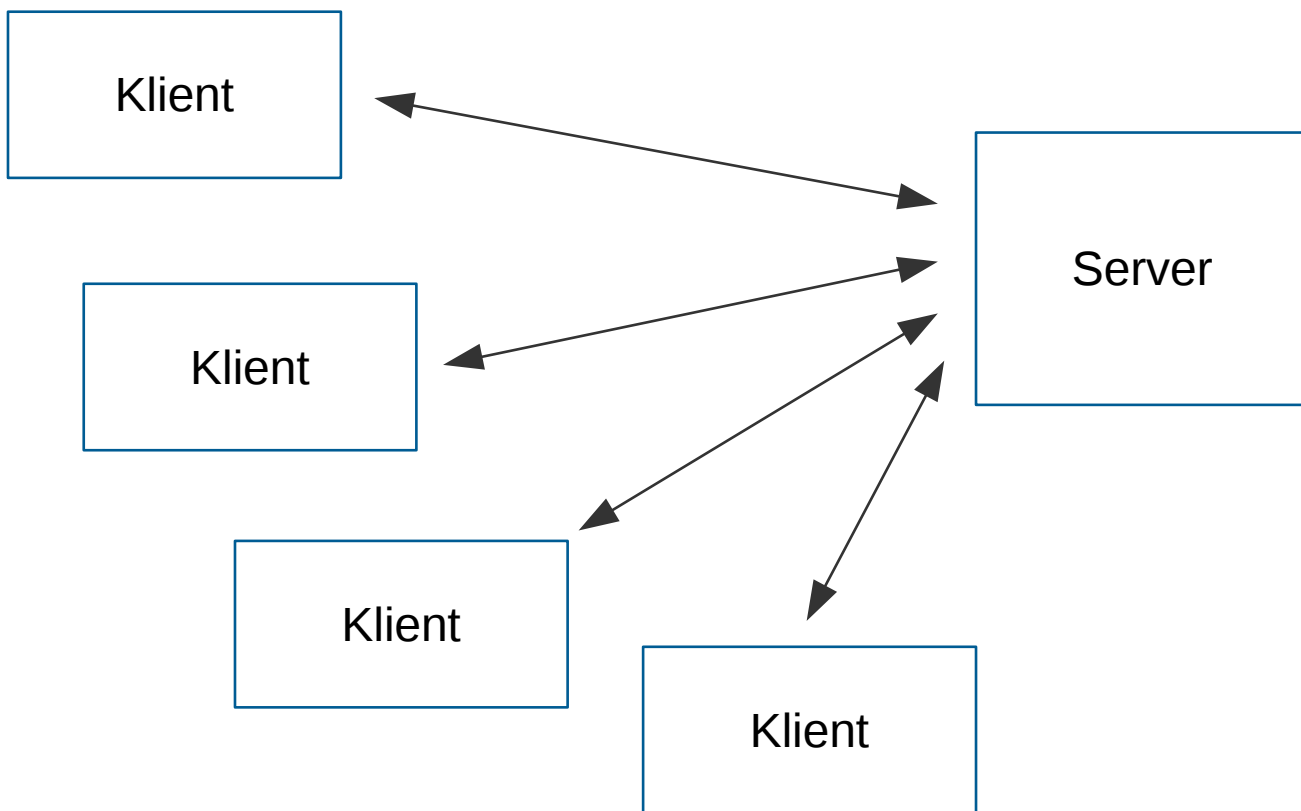
Softvérová architektúra

- Štruktúra systému
- Rozhodnutia, ktoré je zložité zmeniť
- Architektonické vzory a štýly

Príklady vzorov

Klient–server

- Architektúra distribuovaných aplikácií



Trojvrstvová architektúra

1. Prezentačná vrstva — používateľské rozhranie
 2. Logická vrstva — koordinácia a logika aplikácie
 3. Dátová vrstva — databáza
- Vzťah klient—server medzi všetkými vrstvami

Príklad: Obchod s knihami

1. Prezentačná vrstva

- webové rozhranie
- mobilná aplikácia
- aplikácia v predajniach

2. Logická vrstva

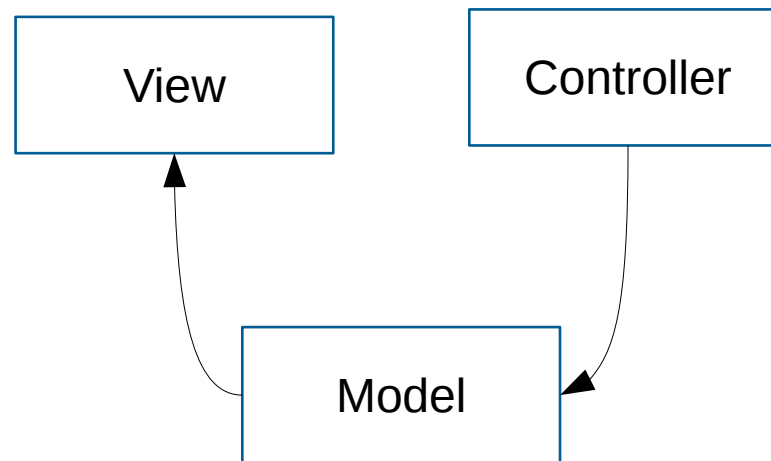
- poskytnutie informácií klientom
- autentifikácia a autorizácia
- spracovanie objednávok

3. Dátová vrstva

- databáza kníh, zákazníkov, skladových zásob

Model—view—controller (MVC)

- Architektonický vzor používateľského rozhrania
 1. Model — reprezentácia dát
 2. View — zobrazenie pre používateľa
 3. Controller — manipulácia
- Modifikácie a varianty
 - MVP, MVVM, MTV, ...



Komponenty

Dva typy komponentov

- Knižnice
 - v jednom procese
 - komunikácia — volania funkcií, spoločná pamäť
- Služby
 - samostatné procesy
 - medziprocesová komunikácia

microservices

<http://www.martinfowler.com/articles/microservices.html>

Rozdelenie dané doménou

- Horizontálne
- Namiesto rozdelenie podľa technológie
 - obmedzená škálovateľnosť
 - silné previazanie vrstiev

Príklad služieb

- Ohraničené (*bounded*) kontexty
 - Katalóg knih
 - Sklad
 - Spracovanie platieb
 - Doručovanie

„Smart endpoints & dumb pipes“

- Jednoduché komunikačné kanály
 - HTTP (REST)
 - Messaging (RabbitMQ, ZeroMQ, atď.)

Decentralizácia

- Samostatné tímy
- Rôzne technológie
- Samostatné dátové úložiská

Distribovaná architektúra

- Škálovateľnosť
- Automatizáciu infraštruktúry
- Nezávislý vývoj a aktualizácia
- Odolnosť voči zlyhaniu

Diskusia