# Monolitická architektura (Monolith)

**Monolit** je architektura, kde jsou všechny funkční moduly aplikace pevně propojeny. I když může být kód uvnitř logicky rozdělen do tříd nebo balíčků, výsledkem kompilace je jeden velký spustitelný soubor (např. .exe, .war nebo .jar).

## 1. Charakteristické znaky

- **Jedna databáze:** Celá aplikace obvykle používá jedno centrální datové úložiště.
- **Sdílená paměť:** Různé části aplikace spolu komunikují voláním funkcí přímo v paměti, nikoliv přes síť.
- **Jednotná technologie:** Celý monolit je zpravidla napsán v jednom programovacím jazyce a používá jeden framework.
- **Společné nasazení:** Jakákoliv změna (i oprava jedné ikony) vyžaduje zastavení a znovunasažení celé aplikace.

## 2. Výhody monolitu

I přes popularitu mikroslužeb zůstává monolit v mnoha případech lepší volbou:

- **Jednoduchost vývoje:** Na začátku projektu je mnohem snazší postavit monolit. Nemusíte řešit síťovou komunikaci mezi službami.
- **Snadné testování:** Testování „end-to-end“ je přímočaré, protože vše běží na jednom místě.
- **Výkon:** Volání funkcí uvnitř jednoho procesu je řádově rychlejší než posílání dat přes síť (API).
- **Snadný monitoring:** Sledujete pouze jednu běžící aplikaci a jeden soubor s logy.

## 3. Nevýhody a úskalí

Problémy nastávají, když aplikace vyroste do obřích rozměrů:

- **Bariéra pro inovace:** Pokud chcete přejít na novější verzi jazyka nebo jinou technologii, musíte přepsat celý monolit najednou.
- **Pomalé nasazování:** Sestavení (build) obřího projektu může trvat desítky minut i hodiny.
- **Nízká odolnost:** Pokud dojde k úniku paměti v modulu pro tisk faktur, spadne celý e-shop včetně košíku a plateb.
- **Škálování:** Nemůžete posílit pouze nejvytíženější část aplikace. Musíte duplikovat celý monolit, což zbytečně plýtvá zdroji.

## 4. Kdy zvolit monolit?

- **Startupy:** V rané fázi (MVP), kdy se požadavky rychle mění a potřebujete co nejrychleji uvést produkt na trh.
- **Malé týmy:** Tým o 3 lidech by nezvládl režii spojenou se správou desítek mikroslužeb.

- **Jednoduché aplikace:** Pokud aplikace nemá extrémní nároky na škálování nebo komplexitu.

## 5. Cesta k rozkladu (Refaktoring)

Mnoho firem začíná s monolitem a jakmile narostou, začnou z něj postupně „vykusovat“ jednotlivé moduly a převádět je na mikroslužby. Tomuto přístupu se říká **Modular Monolith** – mezistupeň, kde je kód čistě oddělen, ale stále běží jako jeden celek.

> **Zajímavost:** Mnoho úspěšných firem (včetně Stack Overflow nebo Basecamp) stále úspěšně provozuje obří monolity. Dokazují tím, že mikroslužby nejsou jedinou cestou k úspěchu a že dobře udržovaný monolit může být efektivnější než špatně navržené mikroslužby (tzv. „distribuovaný monolit“).

Zpět na Rozcestník

From:
https://serviceit.cz/ - **IT ENCYKLOPEDIE**

Permanent link:
**https://serviceit.cz/doku.php?id=monolit**

Last update: **2025/12/31 17:19**