

## Struktura databáze - RavenDb

COLLECTIONS (3)		
Movies	62,423	
Ratings	25	M
Tags	1.09	M

### MOVIES

```
{
  "Title": "Toy Story (1995)",
  "Genres": "Adventure|Animation|Children|Comedy|Fantasy",
  "Links": [
    {
      "ImdbId": 114709,
      "TmdbId": 862
    }
  ],
  "AverageRating": 3.893727098237655,
  "@metadata": {
    "@collection": "Movies"
  }
}
```

### RATINGS

```
{
  "Rating": "3.0",
  "Timestamp": 1573946191,
  "MovieId": "Movies/180989",
  "UserId": "Users/4",
  "@metadata": {
    "@collection": "Ratings"
  }
}
```

### TAGS

```
{
  "Tag": "robbery",
  "Timestamp": 1527518193,
  "MovieId": "Movies/189169",
  "UserId": "Users/162534",
  "@metadata": {
    "@collection": "Tags"
  }
}
```

## Zdůvodnění volby dokumentové databáze

Při pohledu na strukturu zvolených dat a skutečnost, že data nebudou v našem případě dělena na více serverů (nodů), připadá v úvahu užití jak relačních databází, tak relativně velkého množství databází nerelačních. Podíváme-li se na jednotlivé nabízející se nerelační databáze, tak jako nejvhodnější se jeví dokumentový typ databází. Důvodem je, že nám umožňují přistupovat k jednotlivým částem dokumentu a neomezují nás, jako např. key-value databáze pouze na znalost vyhledávacího klíče (neboť samotná values bývají serializovaná). Výhodou dokumentových ve vztahu k column-oriented databázím je poté skutečnost, že pro náš účel bychom primárně načítali data napříč jednotlivými sloupci, a tudíž byli nezbytně nuceni přistupovat k různým částem paměti pro každý záznam. Byť by se dalo tato skutečnost řešit např. implementací komplexnější struktury např. skrze tzv. super-columns umožňující implementovat další podsloupce, a tedy i nahlížet hlouběji do interní struktury jeví se daný postup jako zbytečně komplikovaný.

Shodneme-li se tedy na tom, že volba leží mezi některou relační databází a dokumentovou databází můžeme nyní vnést do našeho rozhodování tzv. CAP teorém. Zjednoduše řečeno... je pro nás důležitější konzistence dat či jejich dostupnost? Pro náš případ (databáze filmů – pouze prohlížení) asi není zcela klíčové, zda se potenciálnímu uživateli v některých situacích zobrazí o pár tagů méně, příp. že průměrné hodnocení nebude vždy maximálně přesné. Důležitější pro nás bude skutečnost, že uživatel obdrží nějaký výsledek na svůj dotaz v každé situaci. Tímto upřednostňujeme availability na úkor konzistence, a tedy ve své podstatě upřednostňujeme databáze nerelační. Tato volba by se nám navíc mohla vyplatit při dělení dat na více serverů, kdy by absence vazeb mezi daty učinila tento krok výrazně jednodušším.