VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

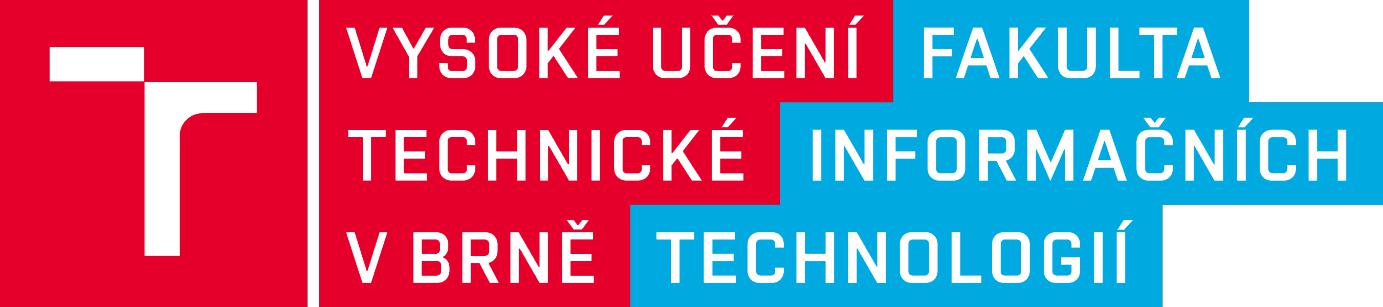
Fakulta informačních technologií

Databázové systémy

2017/2018

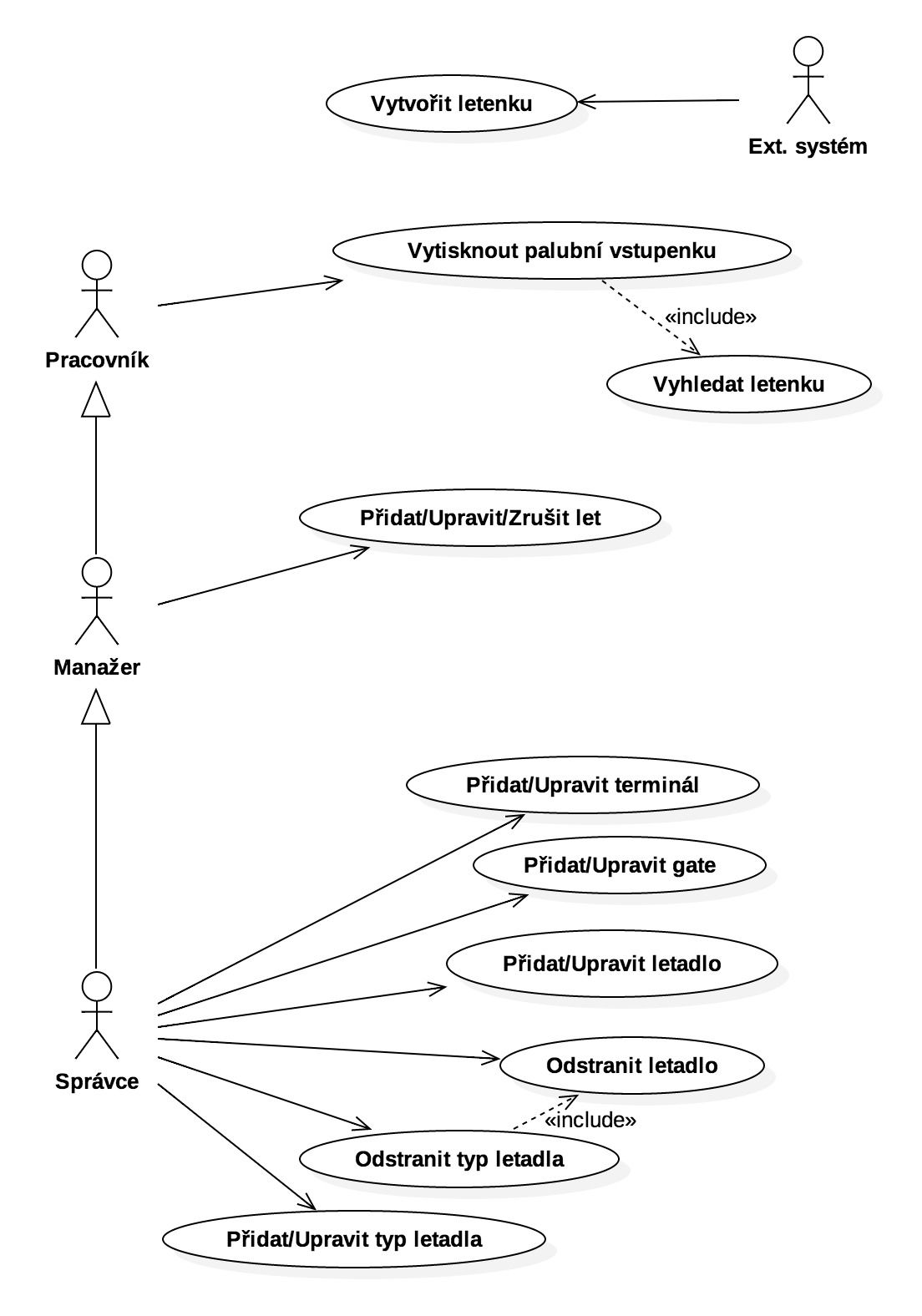
**Projekt IDS – 5. část**

**Zadání č. 28 – Letiště (IUS)**

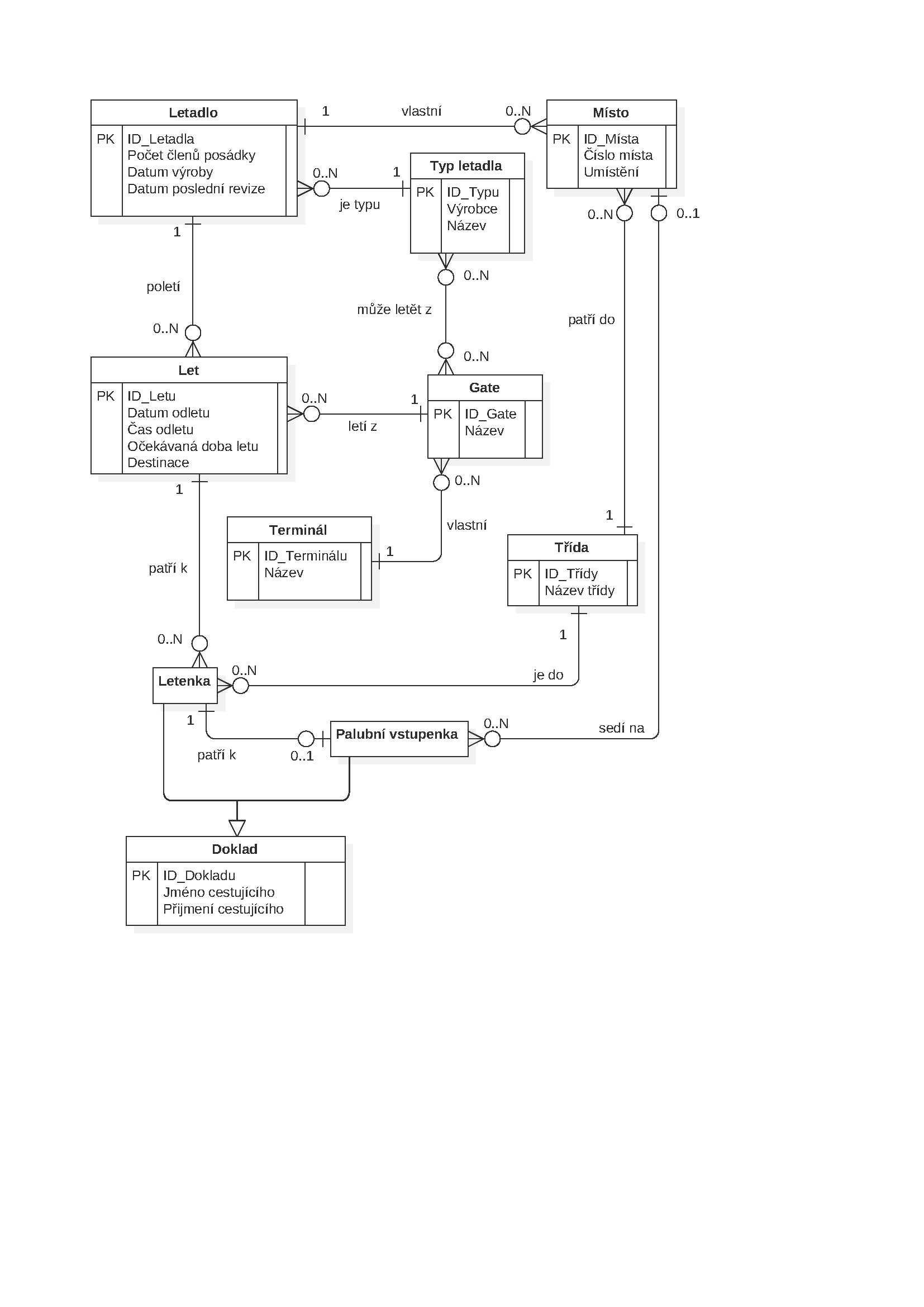
****

Autoři: Pavel Witassek (xwitas00), Jozef Vanický (xvanic09)  
Fakulta Informačních technologií  
Vysoké učení technické v Brně Brno 28.4.2018

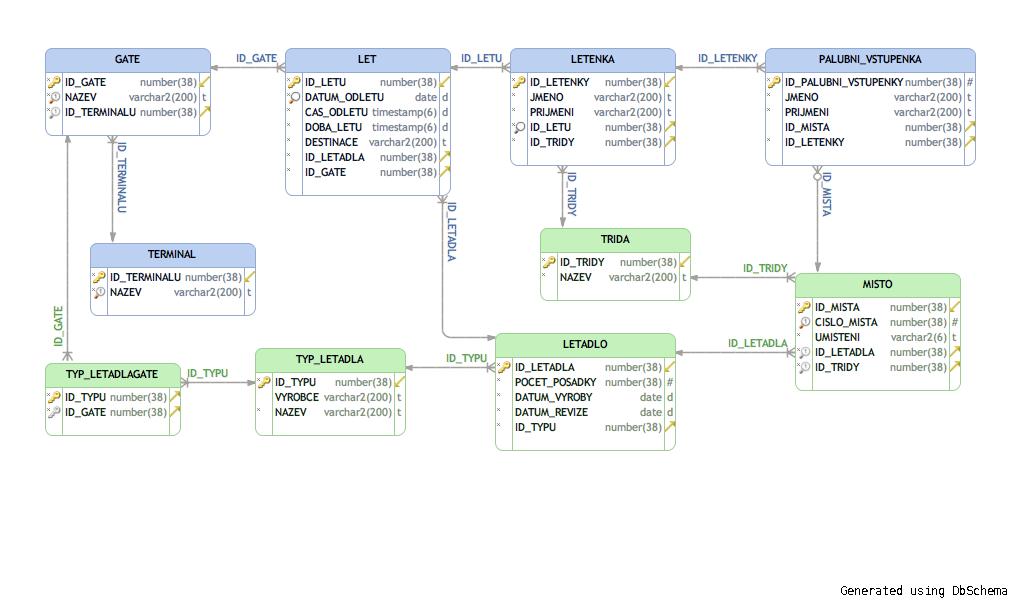
# Diagram případů užití



# ER diagram (Entity Relationship Diagram)

****

# Schéma tabulek relační databáze



## Převod generalizace

Generalizaci entity Doklad jsme provedli jako samostatné tabulky pro její specializace, protože specializace je totální a disjunktní. Nemůže existovat jeden doklad, který je zároveň letenka i palubní vstupenka a doklad musí být buď palubní vstupenka nebo letenka, dále je potřeba, aby palubní vstupenka byla vydána vždy k nějaké letence.

## Triggery

V projektu jsme vytvořili dva triggery, jeden pro automatické generováno hodnot primárního klíče v tabulce letenka a druhý pro ošetření správnosti dat v tabulce let, kdy se ověřuje, že zadané letadlo může z daného gatu letět na základě typu daného letadla, tedy dat v tabulce typ\_letadlagate. Oba triggery se spouštějí před vložením dat do tabulky a pro automatické generování byla využita sekvence.

## Procedury

TBD

## Indexy

Nejprve jsme naplnily tabulky let a letenka dalšími údaji, aby v nich nebyli pouze jednotky řádků, ale aspoň desítky. Následně jsme spustili vysvětlení dotazu bez indexů, jehož výsledek zobrazuje následující tabulka. V níž jde vidět, že tabulka let se prochází celá, ale v dotazu vybíráme pouze část záznamů dle data odletu a procházení celé tabulky je tedy neoptimální.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Operation | Name | Rows | Bytes | Cost (%CPU) | Time |
| 0 | SELECT STATEMENT |  | 2 | 74 | 7 (29) | 00:00:01 |
| 1 | SORT GROUP BY |  | 2 | 74 | 7 (29) | 00:00:01 |
| 2 | MERGE JOIN OUTER |  | 3 | 111 | 6 (17) | 00:00:01 |
| \*3 | TABLE ACCESS BY INDEX ROWID | LET | 3 | 33 | 2 (0) | 00:00:01 |
| 4 | INDEX FULL SCAN | PK\_LET | 5 |  | 1 (0) | 00:00:01 |
| \*5 | SORT JOIN |  | 1 | 26 | 4 (25) | 00:00:01 |
| 6 | TABLE ACCESS FULL | LETENKA | 1 | 26 | 3 (0) | 00:00:01 |

Pro omezení procházení celé tabulky je vhodné zavést index nad sloupci datum\_odletu a id\_letu, aby se neprocházely všechny záznamy v tabulce.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Operation | Name | Rows | Bytes | Cost (%CPU) | Time |
| 0 | SELECT STATEMENT |  | 2 | 74 | 1 (0) | 00:00:01 |
| 1 | SORT GROUP BY NOSORT |  | 2 | 74 | 1 (0) | 00:00:01 |
| 2 | NESTED LOOPS OUTER |  | 3 | 111 | 1 (0) | 00:00:01 |
| \*3 | INDEX RANGE SCAN | DATUM\_LET | 3 | 33 | 1 (0) | 00:00:01 |
| \*4 | INDEX FULL SCAN | LETENKA\_LET | 1 | 26 | 0 (0) | 00:00:01 |

Po zavedení indexů jde vidět, že se snížil počet procházených řádků.

## Materializovaný pohled a práva

TBD