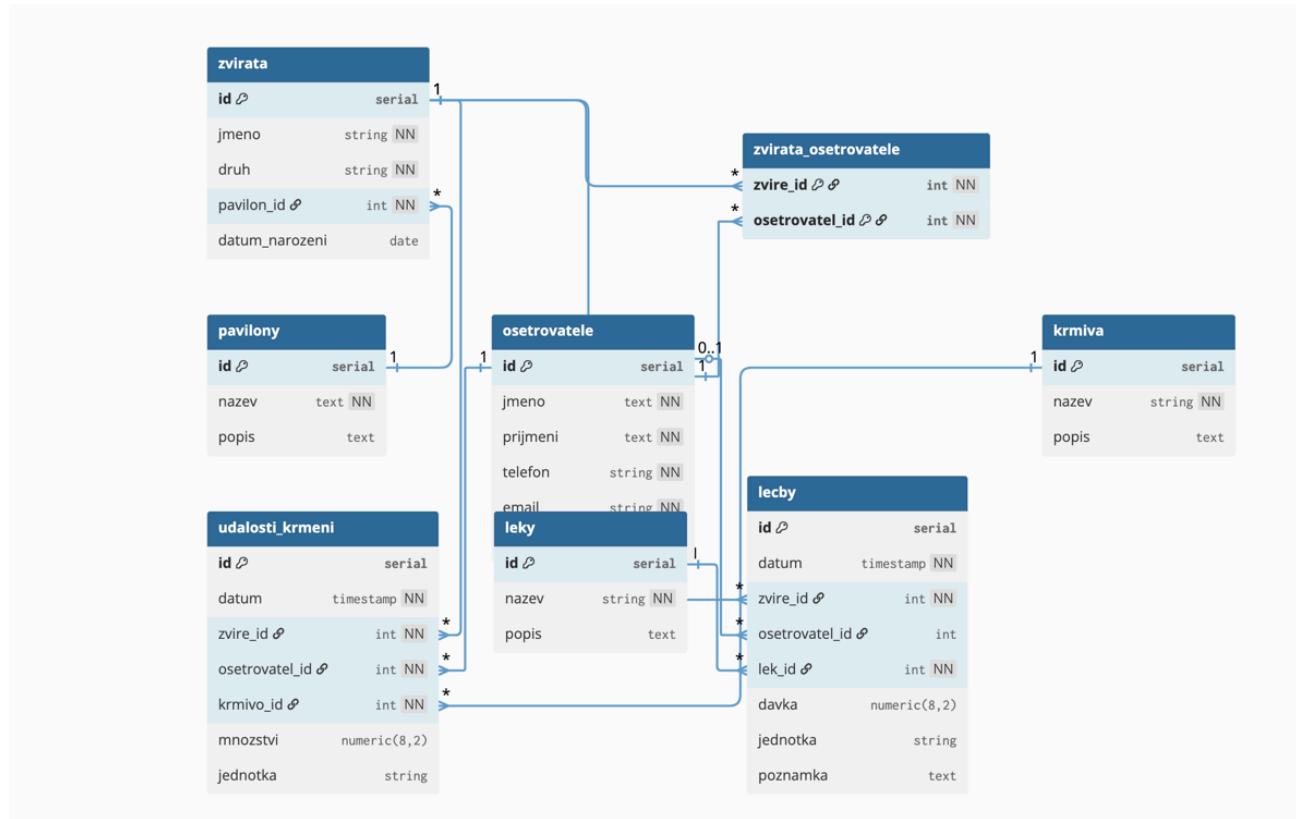
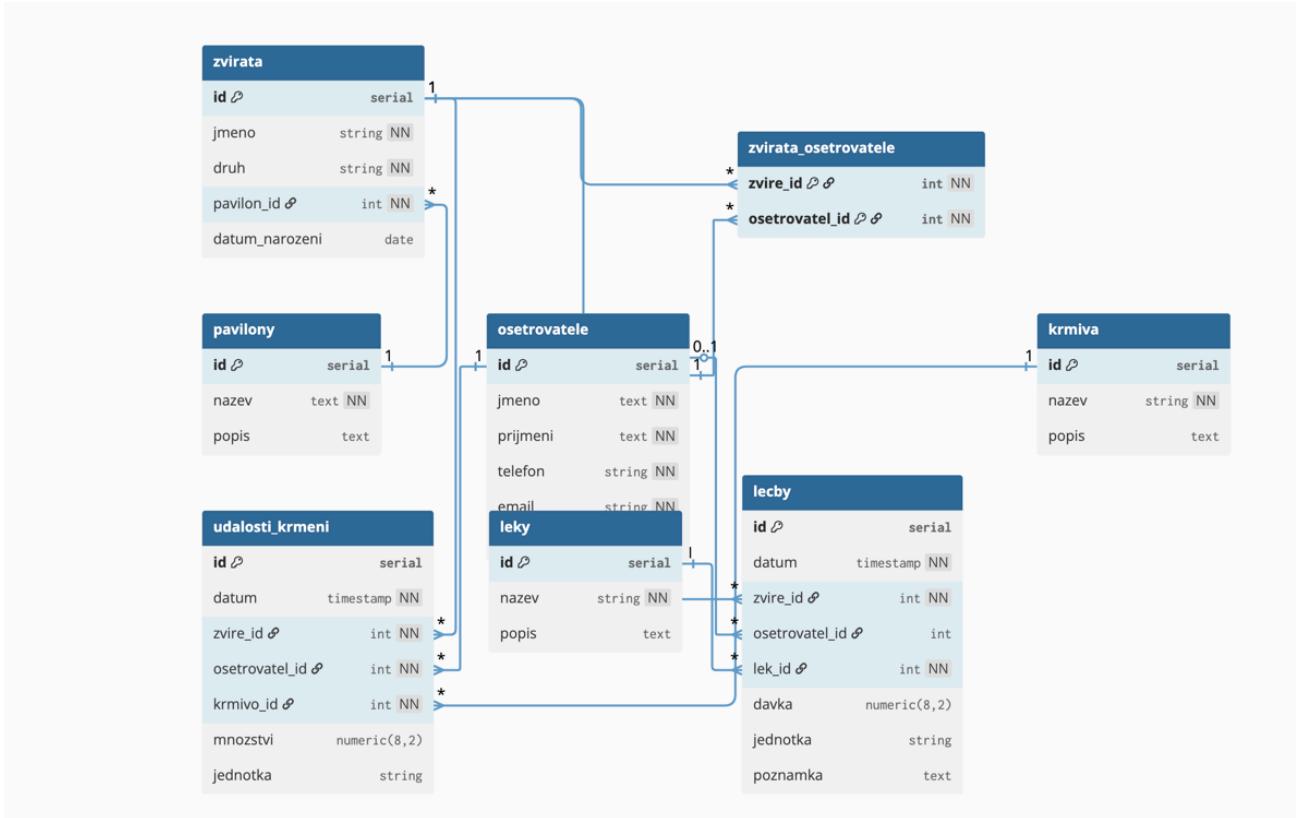


## Diagramy

## Schválený diagram



## Finální diagram



## **Popis entit**

Tento dokument popisuje jednotlivé entity (tabulky) databázového modelu systému pro správu zoologické zahrady.

### **caretakers (Ošetřovatelé)**

Tabulka uchovává informace o zaměstnancích zoo, kteří se starají o zvířata. Každý ošetřovatel má evidováno jméno, příjmení, e-mail a telefonní číslo. E-mail i telefon musí být unikátní a jsou validovány pomocí vlastních domén - e-mail musí obsahovat znak @ a telefon musí odpovídat českému formátu. Volitelně lze ke každému ošetřovateli přidat poznámky, například o jeho specializaci či kvalifikaci.

### **habitats (Výběhy)**

Tabulka reprezentuje jednotlivé výběhy či prostory, ve kterých jsou zvířata umístěna. Každý výběh má unikátní název a volitelný popis. Výběhy slouží k logickému seskupení zvířat podle jejich umístění v zoo.

### **animals (Zvířata)**

Hlavní tabulka obsahující informace o jednotlivých zvířatech v zoo. U každého zvířete je evidováno jeho jméno, druh a datum narození. Každé zvíře musí být přiřazeno k jednomu výběhu prostřednictvím cizího klíče na tabulku habitats.

### **feed\_items (Krmivo)**

Tabulka eviduje druhy krmiva dostupné v zoo. Kromě unikátního názvu a volitelného popisu sleduje i aktuální množství na skladě. Množství je vyjádřeno v kusech a nesmí klesnout pod nulu.

### **medications (Léky)**

Tabulka eviduje léky a léčiva používaná pro ošetření zvířat. Každý lék má unikátní název, povinné pokyny pro podání a volitelný popis. Podobně jako u krmiva se sleduje dostupné množství na skladě v kusech.

### **feeding\_events (Události krmení)**

Tabulka zaznamenává jednotlivé události krmení zvířat. Každý záznam propojuje konkrétní zvíře, ošetřovatele a druh krmiva pomocí cizích klíčů. Dále obsahuje časové razítko krmení, spotřebované množství krmiva a volitelné poznámky. Množství použitého krmiva musí být vždy kladné.

### **treatments (Ošetření)**

Tabulka zaznamenává lékařská ošetření a podání léků zvířatům. Struktura je obdobná jako u událostí krmení - každý záznam propojuje zvíře, ošetřovatele a použitý lék. Obsahuje časové razítko podání, spotřebované množství léku a volitelné poznámky k průběhu ošetření.

### **animal\_caretaker (Přiřazení ošetřovatelů ke zvířatům)**

Vazební tabulka realizující vztah M:N mezi zvířaty a ošetřovateli. Určuje, kteří ošetřovateli jsou zodpovědní za která zvířata. Primární klíč je složený z cizích klíčů na obě propojované tabulky. Navíc se u každého přiřazení eviduje datum a čas, kdy bylo vytvořeno.

## Databázové pohledy

Tento dokument popisuje databázové pohledy vytvořené pro zjednodušení práce s daty v systému správy zoologické zahrady.

### v\_caretakers\_full\_name

```
CREATE VIEW v_caretakers_full_name AS
SELECT
  *,
  concat(first_name, ' ', last_name) AS full_name
FROM caretakers;
```

Pomocný pohled, který rozšiřuje tabulku ošetřovatelů o sloupec s celým jménem. Spojení křestního jména a příjmení je v aplikaci často potřeba, a proto je výhodné mít tento výpočet na jednom místě. Pohled je využíván jako základ pro další složitější pohledy a dotazy, například při hledání nejzkušenějšího veterináře podle počtu provedených ošetření.

### v\_animals\_with\_habitat

```
CREATE VIEW v_animals_with_habitat AS
SELECT
  a.id,
  a.name,
  a.species,
  a.born_at,
  h.name as habitat_name,
  h.description as habitat_description
FROM animals a
LEFT JOIN habitats h
  ON a.habitat_id = h.id;
```

Pohled zobrazující zvířata společně s informacemi o jejich výběhu. Díky použití LEFT JOIN jsou zahrnuty i případná zvířata bez přiřazeného výběhu. Pohled usnadňuje získání kompletního přehledu o umístění zvířat v zoo bez nutnosti opakovaného psaní JOIN klauzule.

### v\_caretakers\_with\_animals

```
CREATE VIEW v_caretakers_with_animals AS
SELECT
  c.id as caretaker_id,
  c.full_name as caretaker_name,
  c.info as caretaker_info,
  c.email,
  c.phone,
  a.name as managed_animal_name,
  a.species as managed_animal_species,
  h.name as animal_habitat_name
FROM v_caretakers_full_name c
LEFT JOIN animal_caretaker ac
  ON c.id = ac.caretaker_id
LEFT JOIN animals a
  ON a.id = ac.animal_id
```

```
LEFT JOIN habitats h
    ON h.id = a.habitat_id;
```

Komplexní pohled propojující ošetřovatele s jejich přiřazenými zvířaty včetně informací o výběhu. Díky LEFT JOIN jsou zahrnuti i ošetřovatelé, kteří aktuálně nemají přiřazené žádné zvíře. Pohled je využíván při ověřování přiřazení nového zvířete k ošetřovateli, jak ukazuje první scénář při kontrole, zda byl Pavel úspěšně přiřazen k zebře Mártě.

### v\_caretakers\_without\_animals

```
CREATE VIEW v_caretakers_without_animals AS
SELECT
    caretaker_id as id,
    caretaker_name as full_name,
    email,
    phone,
    caretaker_info as info
FROM v_caretakers_with_animals
WHERE managed_animal_name IS NULL;
```

Pohled filtroující ošetřovatele, kteří momentálně nemají přiřazené žádné zvíře. Slouží k rychlému zjištění dostupných pracovníků, kteří mohou převzít péči o nové zvíře nebo provést ad-hoc úkoly jako krmení či ošetření. V prvním scénáři je použit při hledání vhodného ošetřovatele pro nově příchozí zebru, v druhém scénáři pro nalezení volného ošetřovatele pro krmení Iva Simby.

### v\_caretaker\_responsibilities

```
CREATE VIEW v_caretaker_responsibilities AS
SELECT
    c.id,
    c.full_name,
    c.info,
    COUNT(ac.caretaker_id) as managed_animals_count
FROM v_caretakers_full_name c
LEFT JOIN animal_caretaker ac
    ON ac.caretaker_id = c.id
GROUP BY
    c.id,
    c.full_name,
    c.info;
```

Pohled zobrazující vytíženosť jednotlivých ošetřovatelů podle počtu přiřazených zvířat. Umožňuje rovnoměrné rozdělování práce a identifikaci přetížených či naopak nevytížených zaměstnanců. V prvním scénáři je využit pro nalezení ošetřovatele s nízkou vytíženosťí poté, co volní ošetřovatelé nebyli vhodní pro zebru z důvodu jiné specializace.

### v\_treatments\_detail

```
CREATE VIEW v_treatments_detail AS
SELECT
    t.id as treatment_id,
    t.administered_at,
    t.medication_qty_used,
    m.name as medication_name,
    t.notes as procedure_notes,
    c.full_name as caretaker_name,
```

```

a.name as animal_name,
a.species as animal_species,
a.born_at as animal_born_at
FROM treatments t
INNER JOIN animals a
    ON t.animal_id = a.id
INNER JOIN v_caretakers_full_name c
    ON t.caretaker_id = c.id
INNER JOIN medications m
    ON t.medication_id = m.id;

```

Detailní pohled na provedená veterinární ošetření. Spoluji informace o ošetření s údaji o zvířeti, ošetřovateli a použitém léku do jednoho přehledného výstupu. Na rozdíl od předchozích pohledů používá INNER JOIN, protože ošetření bez platných referencí by nemělo existovat. Pohled je využíván pro ověření úspěšně provedených zákroků, jak demonstruje druhý scénář při kontrole uspání a dezinfekce rány Iva Simby.

### v\_feeding\_event\_detail

```

CREATE VIEW v_feeding_event_detail AS
SELECT
    f.id as feeding_event_id,
    f.fed_at,
    i.name as feed_type_name,
    f.feed_item_qty_used as qty_used,
    f.notes as feeding_notes,
    c.full_name as caretaker_name,
    a.name as animal_name,
    a.species as animal_species,
    a.born_at as animal_born_at
FROM feeding_events f
INNER JOIN animals a
    ON f.animal_id = a.id
INNER JOIN v_caretakers_full_name c
    ON f.caretaker_id = c.id
INNER JOIN feed_items i
    ON f.feed_item_id = i.id;

```

Detailní pohled na události krmení analogický k pohledu na ošetření. Poskytuje kompletní informace o každém krmení včetně údajů o zvířeti, ošetřovateli a použitém krmivu. Používá INNER JOIN ze stejného důvodu jako pohled na ošetření. Druhý scénář tento pohled využívá pro ověření, že krmení Iva Simby bylo správně zaznamenáno, včetně poznámky o nalezené ráně.

## Testovací scénáře

Tento dokument popisuje dva testovací scénáře demonstrující praktické použití databázového systému pro správu zoologické zahrady.

**Poznámka:** Výsledky dotazů v tabulkách mohou být upraveny pro lepší čitelnost - některé sloupce jsou vynechány a počet řádků může být omezen.

### Scénář 1: Příchod nového zvířete do ZOO

Tento scénář simuluje proces přijetí nového zvířete - zebry jménem Máry - do zoologické zahrady. Zahrnuje výběr vhodného výběhu, registraci zvířete a přiřazení ošetřovatele.

#### Krok 1: Výběr vhodného výběhu

Nejprve zjistíme, jaké výběhy jsou v zoo k dispozici.

```
SELECT * FROM habitats;
```

id	name
1	Africká savana
2	Tropický pavilon
3	Severský výběh
4	Ptačí dům
5	Noční pavilon
6	Akvárium
7	Pouštní expozice
8	Dětská zoo
9	Horský výběh
10	Zázemí

Pro zebru je nevhodnější **Africká savana** (id 1).

#### Krok 2: Vložení nového zvířete

Zaregistrujeme zebru Máry do systému s přiřazením do vybraného výběhu.

```
INSERT INTO animals (habitat_id, name, species, born_at)
VALUES (1, 'Máry', 'Zebra Grévyho', '2018-10-02');
```

#### Krok 3: Ověření vložení záznamu

Zkontrolujeme, zda bylo zvíře správně přidáno.

```
SELECT id, name, lower(species) FROM animals
WHERE name LIKE '%Máry%';
```

id	name	species
1	Máry	zebra grévyho

11	Márty	zebra grévyho
----	-------	---------------

#### Krok 4: Hledání vhodného ošetřovatele

Nejprve zjistíme, zda jsou k dispozici ošetřovatelé bez přiřazených zvířat.

```
SELECT id, full_name, info FROM v_caretakers_without_animals;
```

id	full_name	info
2	Petra Svobodová	Specialistka na plazy
6	Eva Procházková	Krmení ptactva

Volní ošetřovatelé jsou specialisté na plazy a ptactvo, což není vhodné pro zebru. Rozšíříme proto vyhledávání na ošetřovatele s nízkou vytížeností.

```
SELECT id, full_name, info, managed_animals_count
FROM v_caretaker_responsibilities
WHERE managed_animals_count <= 2
ORDER BY managed_animals_count;
```

id	full_name	info	managed_animals_count
6	Eva Procházková	Krmení ptactva	0
2	Petra Svobodová	Specialistka na plazy	0
7	Pavel Beneš	Expert na africké savce a hmyz	2
8	Alena Horáková	Administrativa	2
5	Martin Černý		2

**Pavel Beneš** (id 7) je expert na africké savce s pouze dvěma přiřazenými zvířaty - ideální kandidát.

#### Krok 5: Přiřazení ošetřovatele ke zvířeti

```
INSERT INTO animal_caretaker (animal_id, caretaker_id)
VALUES (11, 7);
```

#### Krok 6: Kontrola výsledku operace

Ověříme, že přiřazení proběhlo správně.

```
SELECT caretaker_name, managed_animal_name, managed_animal_species
FROM v_caretakers_with_animals
WHERE managed_animal_name = 'Márty';
```

caretaker_name	managed_animal_name	managed_animal_species
Pavel Beneš	Márty	Zebra Grévyho

Zebra Márty byla úspěšně přidána do zoo a přiřazena ošetřovateli Pavlu Benešovi.

## Scénář 2: Krmení zvířete a následné veterinární ošetření

Tento scénář simuluje běžný pracovní den - krmení zvířete, při kterém je objevena zdravotní komplikace vyžadující veterinární zákrok.

### Krok 1: Výběr ošetřovatele pro krmení

Hledáme volného ošetřovatele, který může provést krmení.

```
SELECT id, full_name, info FROM v_caretakers_without_animals;
```

id	full_name	info
2	Petra Svobodová	Specialistka na plazy
6	Eva Procházková	Krmení ptactva

Nejzkušenější s krmením je **Eva Procházková** (id 6).

### Krok 2: Identifikace zvířat vyžadujících krmení

Vyhledáme zvířata, která nebyla krmena více než 20 dní nebo nemají žádný záznam o krmení.

```
SELECT a.id, a.name, a.species, f.fed_at
FROM animals a
LEFT JOIN feeding_events f ON f.animal_id = a.id
WHERE fed_at < current_date - interval '20 days' OR fed_at IS NULL
ORDER BY fed_at
LIMIT 5;
```

id	name	species	last_fed
1	Simba	Lev africký	2025-09-24
2	Nala	Lvice	2025-09-24
3	Nanuq	Lední medvěd	2025-09-29
4	Azura	Leguán zelený	2025-10-09
5	Kiki	Ara ararauna	2025-10-14

**Simba** (id 1) je nejdéle bez krmení.

### Krok 3: Výběr vhodného krmiva

Pro Iva hledáme masité krmivo s dostatečnou zásobou.

```
SELECT id, lower(name) AS name, qty_available
FROM feed_items
WHERE name LIKE '%maso%'
ORDER BY qty_available DESC;
```

id	name	qty_available
1	hovězí maso	500
2	kuřecí maso	300

**Hovězí maso** (id 1) má nejvyšší zásobu.

#### Krok 4: Provedení krmení a aktualizace skladu

Operace je provedena v transakci pro zajištění konzistence dat.

```
BEGIN;

INSERT INTO feeding_events (animal_id, caretaker_id, feed_item_id, fed_at, feed_item_qty_used,
notes)
VALUES (1, 6, 1, now(), 40, 'Při krmení nalezena zanícená rána na krku');

UPDATE feed_items
SET qty_available = qty_available - 40
WHERE id = 1;

COMMIT;
```

#### Krok 5: Ověření výsledků krmení

```
SELECT fed_at, feed_type_name, qty_used, feeding_notes
FROM v_feeding_event_detail
WHERE animal_name = 'Simba'
ORDER BY fed_at DESC
LIMIT 1;
```

fed_at	feed_type_name	qty_used	feeding_notes
2025-12-18	Hovězí maso	40	Při krmení nalezena zanícená rána na krku

Při krmení byla nalezena zdravotní komplikace, kterou je třeba řešit.

#### Krok 6: Výběr veterináře

Hledáme nejzkušenějšího ošetřovatele podle počtu provedených ošetření.

```
SELECT c.id, c.full_name, COUNT(t.id) AS treatment_count
FROM v_caretakers_full_name c
LEFT JOIN treatments t ON t.caretaker_id = c.id
GROUP BY c.id, c.full_name
ORDER BY treatment_count DESC
LIMIT 1;
```

id	full_name	treatment_count
4	Lucie Králová	9

**Lucie Králová** (id 4) má nejvíce provedených ošetření.

#### Krok 7: Kontrola dostupnosti léčiv

Pro ošetření rány je zapotřebí sedativum a dezinfekce.

```
SELECT id, name, qty_available
FROM medications
```

```

WHERE lower(name) LIKE '%sedativ%' OR lower(name) LIKE '%dezinfek%'
ORDER BY name;

```

<b>id</b>	<b>name</b>	<b>qty_available</b>
6	Dezinfekce F	80
3	Sedativum C	20
10	Sedativum J	0

**Sedativum C** (id 3) a **Dezinfekce F** (id 6) jsou skladem.

#### Krok 8: Provedení veterinárního zákroku

Celý zákrok je proveden v jedné transakci. Bez uspání nelze provést dezinfekci - při selhání dojde k rollbacku celé operace.

```

BEGIN;

-- Uspání zvířete
INSERT INTO treatments (animal_id, caretaker_id, medication_id, administered_at,
medication_qty_used, notes)
VALUES (1, 4, 3, now(), 1, 'Uspání před ošetřením rány');

UPDATE medications
SET qty_available = qty_available - 1
WHERE id = 3;

-- Dezinfekce rány (10 minut po uspání)
INSERT INTO treatments (animal_id, caretaker_id, medication_id, administered_at,
medication_qty_used, notes)
VALUES (1, 4, 6, now() + interval '10 minutes', 2, 'Dezinfekce zanícené rány na krku');

UPDATE medications
SET qty_available = qty_available - 2
WHERE id = 6;

COMMIT;

```

#### Krok 9: Ověření provedených zákroků

```

SELECT administered_at, medication_name, procedure_notes
FROM v_treatments_detail
WHERE animal_name = 'Simba'
ORDER BY administered_at DESC
LIMIT 2;

```

<b>administered_at</b>	<b>medication_name</b>	<b>procedure_notes</b>
2025-12-18 14:10	Dezinfekce F	Dezinfekce zanícené rány na krku
2025-12-18 14:00	Sedativum C	Uspání před ošetřením rány

Lev Simba byl úspěšně nakrmen a ošetřen. Oba zákroky (uspání i dezinfekce) proběhly v rámci jedné transakce.

## Závěr

Semestrální práce realizovala relační databázi pro správu zoologické zahrady s osmi tabulkami v 3. normální formě. Systém pokrývá evidenci zvířat, ošetřovatelů, výběhů, krmiva, léků a zaznamenává události krmení i veterinárních ošetření.

Během návrhu byly provedeny zjednodušení pro udržení rozsahu projektu. Jednotky krmiva a léků jsou evidovány pouze v kusech, což umožnilo vyzkoušet pokročilejší integrativní omezení bez zbytečné komplexity. Skladové zásoby nejsou vedeny formou transakčního deníku - při každém použití je nutné manuálně aktualizovat dostupné množství.

Oproti původnímu návrhu byly názvy entit a atributů změněny do angličtiny pro lepší čitelnost kódu. Finální verze se zaměřuje na každodenní provoz zoo - správu zvířat, jejich krmení a zdravotní péči.