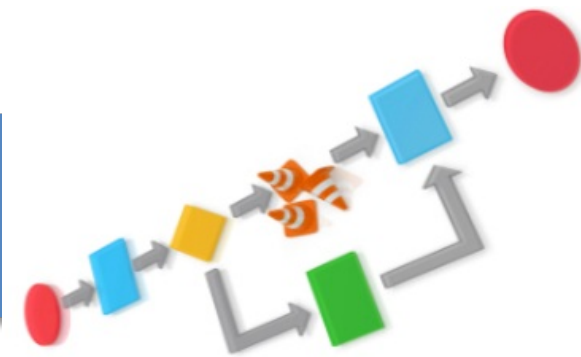


# Integration of applications

Jiří Šebek

*b6b36nss*



```
public final void onSensorChanged(SensorEvent event)
{
    m_flightIntensity = event.values[0];
    m_etAmblight.setText("" + m_flightIntensity + " lx");
}
...
resume()
...
light, ... (NORMAL);
```

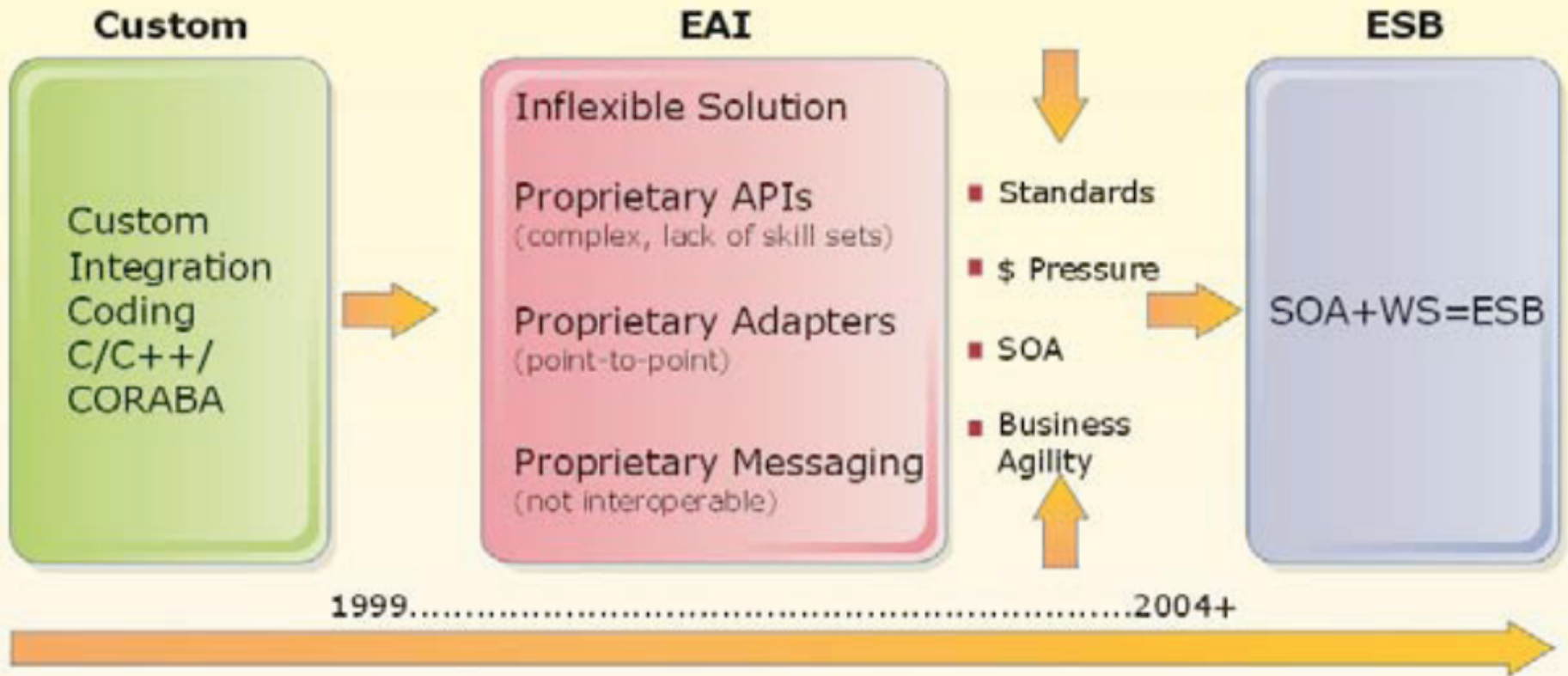
# Obsah

- Definice
  - Co je integration schema/model?– Jak budeme mergovat ruzne modely?
- Problem
  - Různé metody reprezentování stejných konceptů, například pojmenování
- Strategie
  - merge v 1 průchodu – all in one
  - Merge po krocích - stages

# Obsah

- Proces
  - Tři fáze:
    - i) identifikovat problémy,
    - (ii) vyřešit problémy,
    - (iii) slučovací schémata.
- Řešení konfliktů
  - Změna jmen a struktura entit

# Historický přehled integrace

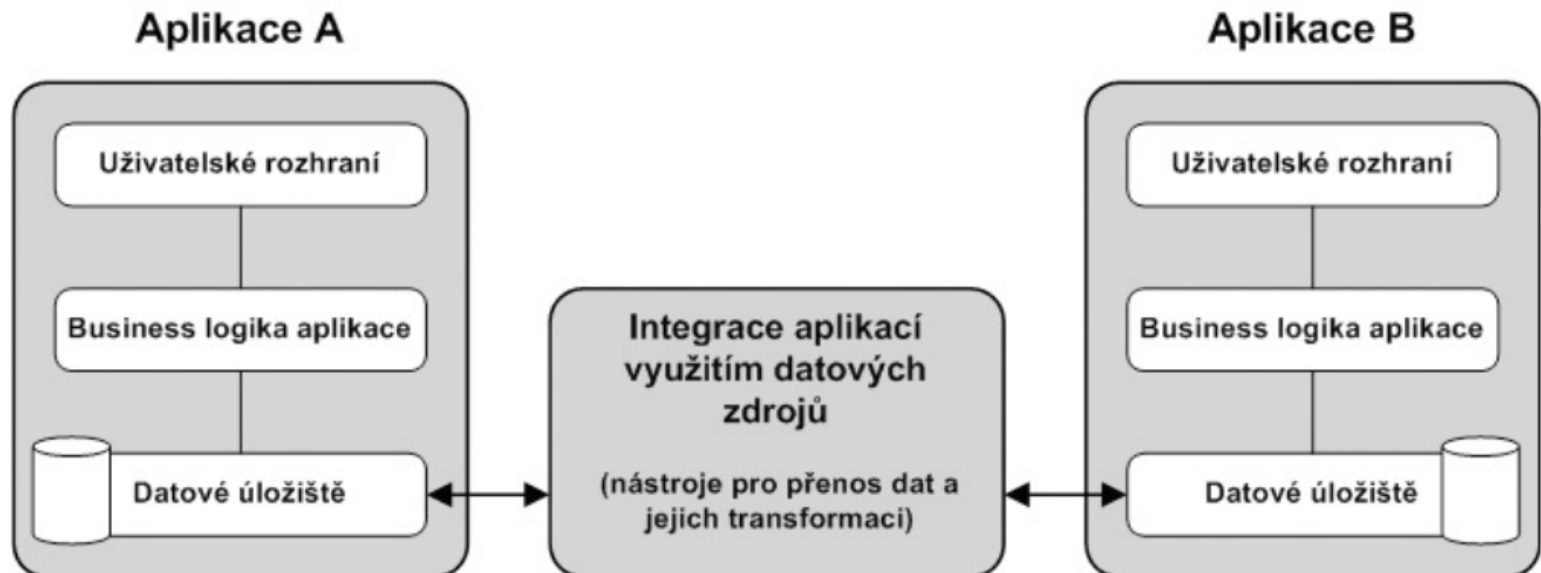


# Přístupy

- datová úroveň
- úroveň obchodní logiky
- úroveň uživatelského rozhraní

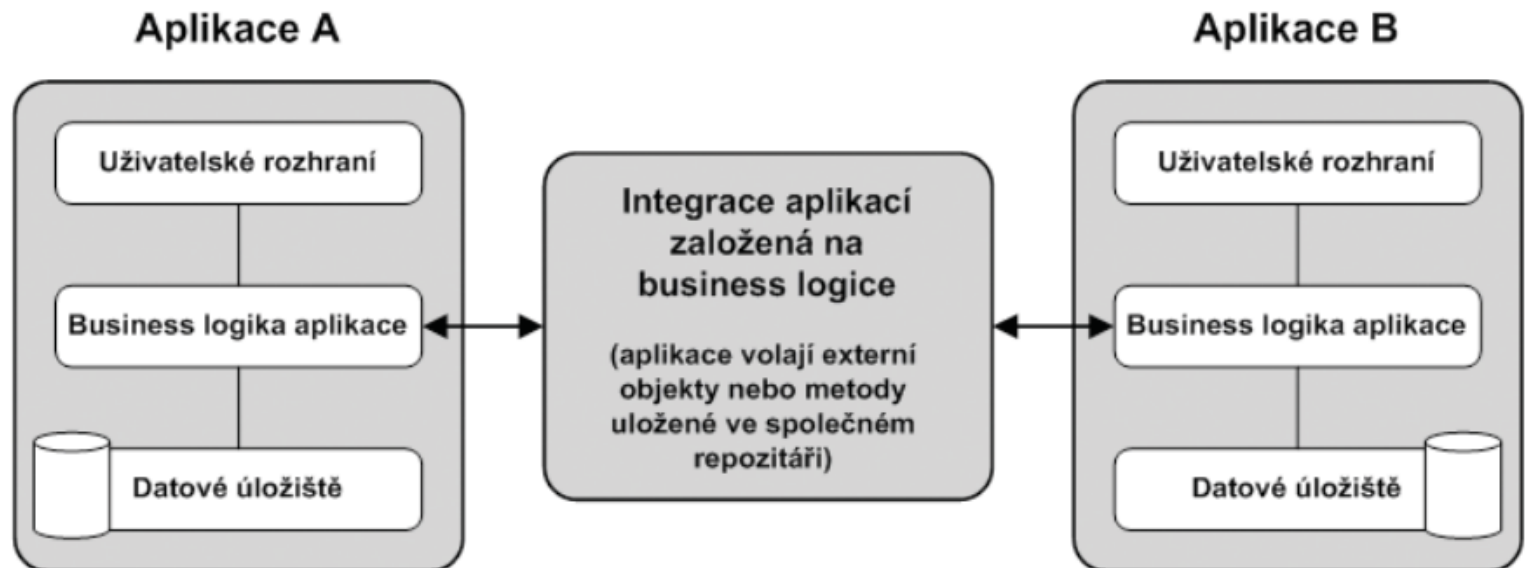
# Přístupy

- datová úroveň
- úroveň obchodní logiky
- úroveň uživatelského rozhraní



# Přístupy

- datová úroveň
- úroveň obchodní logiky
- úroveň uživatelského rozhraní



# Přístupy

- datová úroveň
- úroveň obchodní logiky
- úroveň uživatelského rozhraní

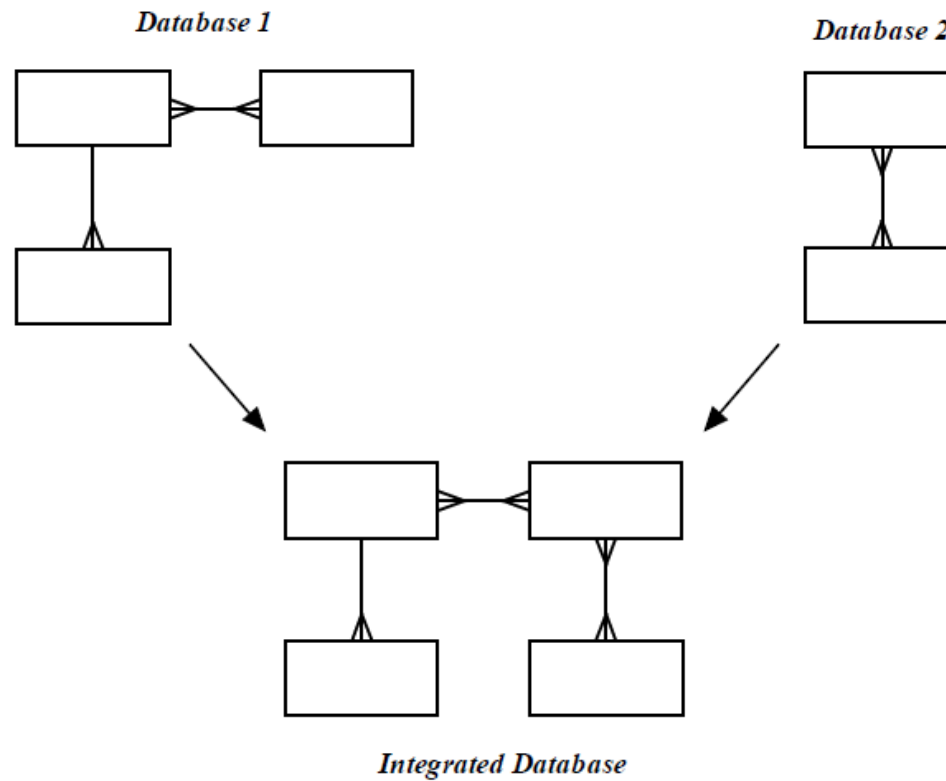




# SOA



# Definice



# Přístupy

Existují dva hlavní typy integrace schématu:

- Integrace pohledu

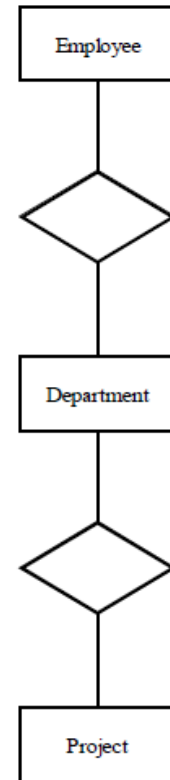
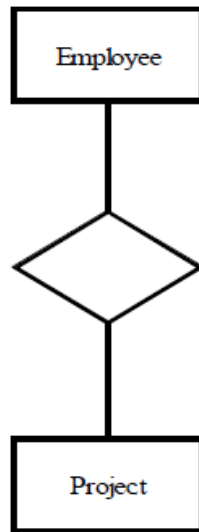
- probíhá během návrhu nové databáze, pokud se liší uživatelské požadavky lišit ( různé skupiny uživatelů)
- používá se ke sloučení různých pohledů do jediného datového modelu

- Integrace databáze

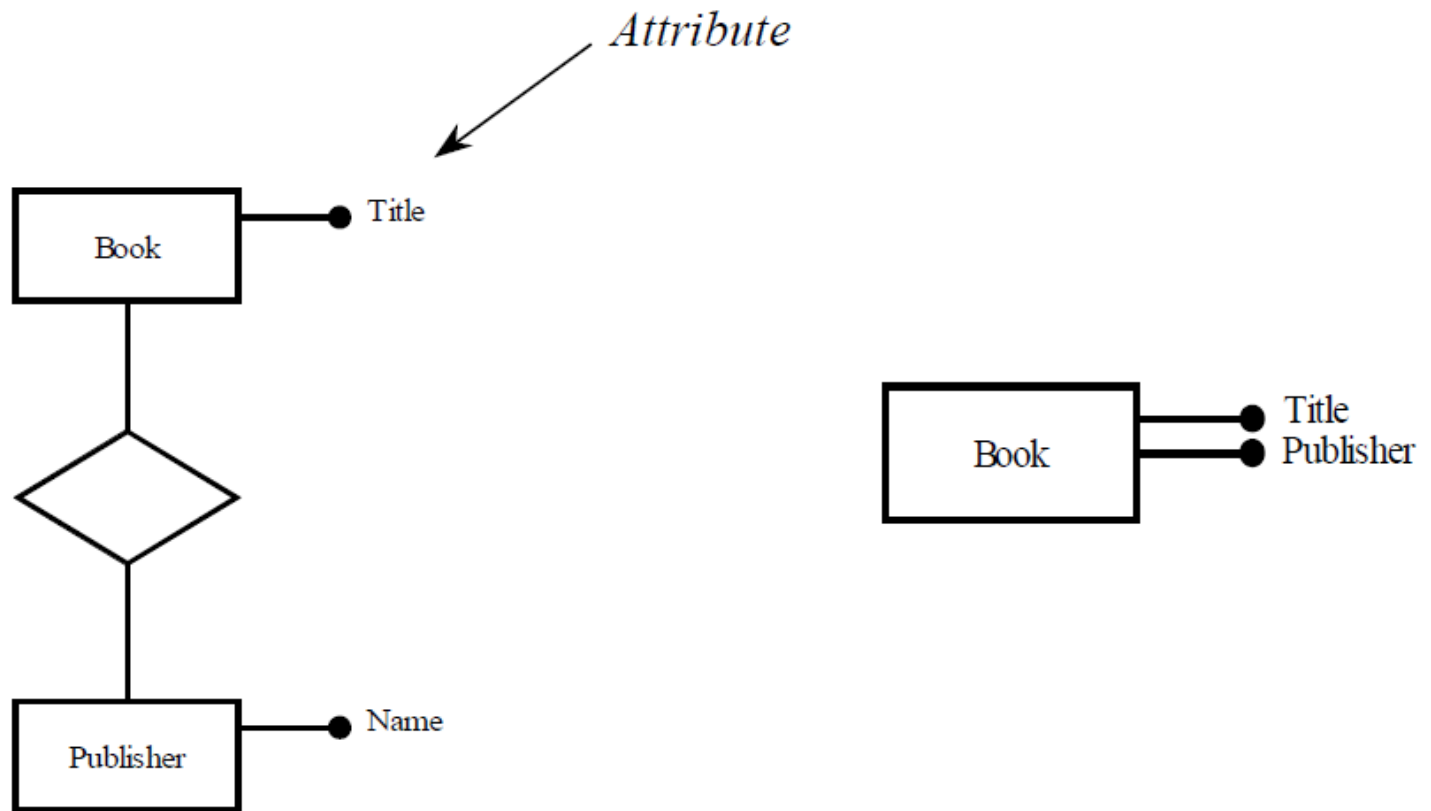
- používá se, pokud je třeba kombinovat více databází, a mají se použít pro vytvoření jediného schématu
  - Výsledné schéma se nazývá ***globální schéma***.

# Problém - Různé perspektivy

*Relationship*



# Problém – Stejný koncept

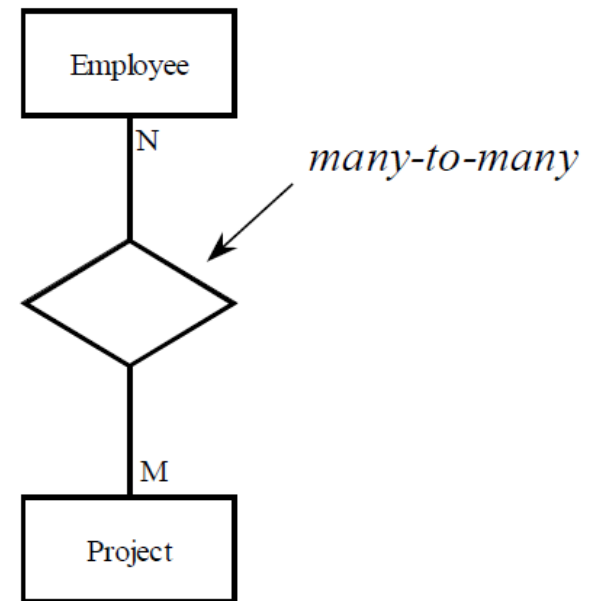
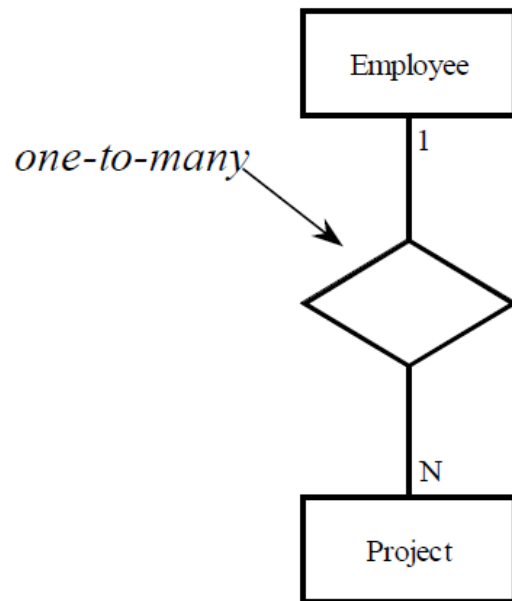


# Problémy

2 problematicke situace :

- různé koncepty modelovány stejným způsobem
- stejné koncepty modelovány odlišným způsobem

# Problém – nekompatibilní design

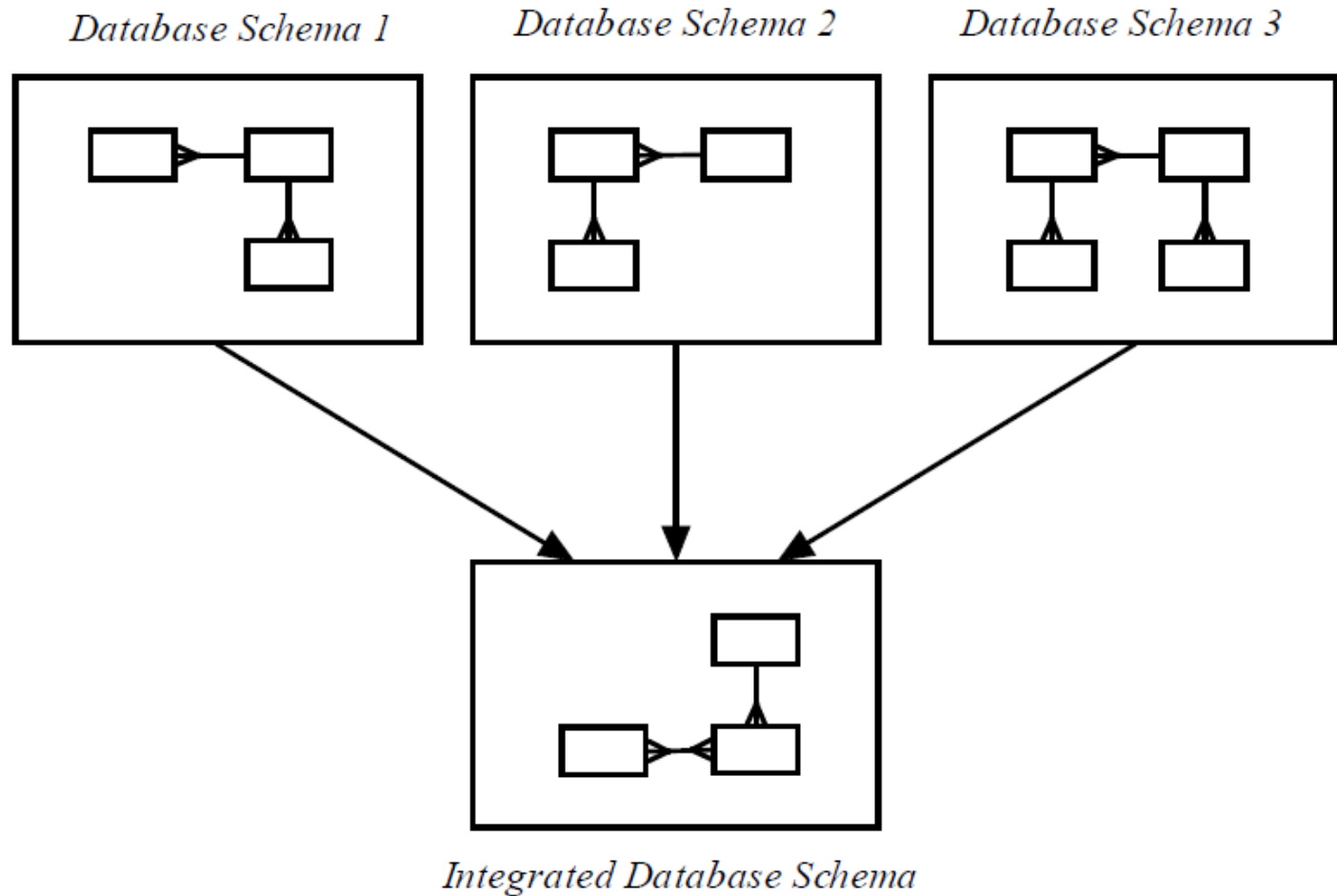


# Strategie

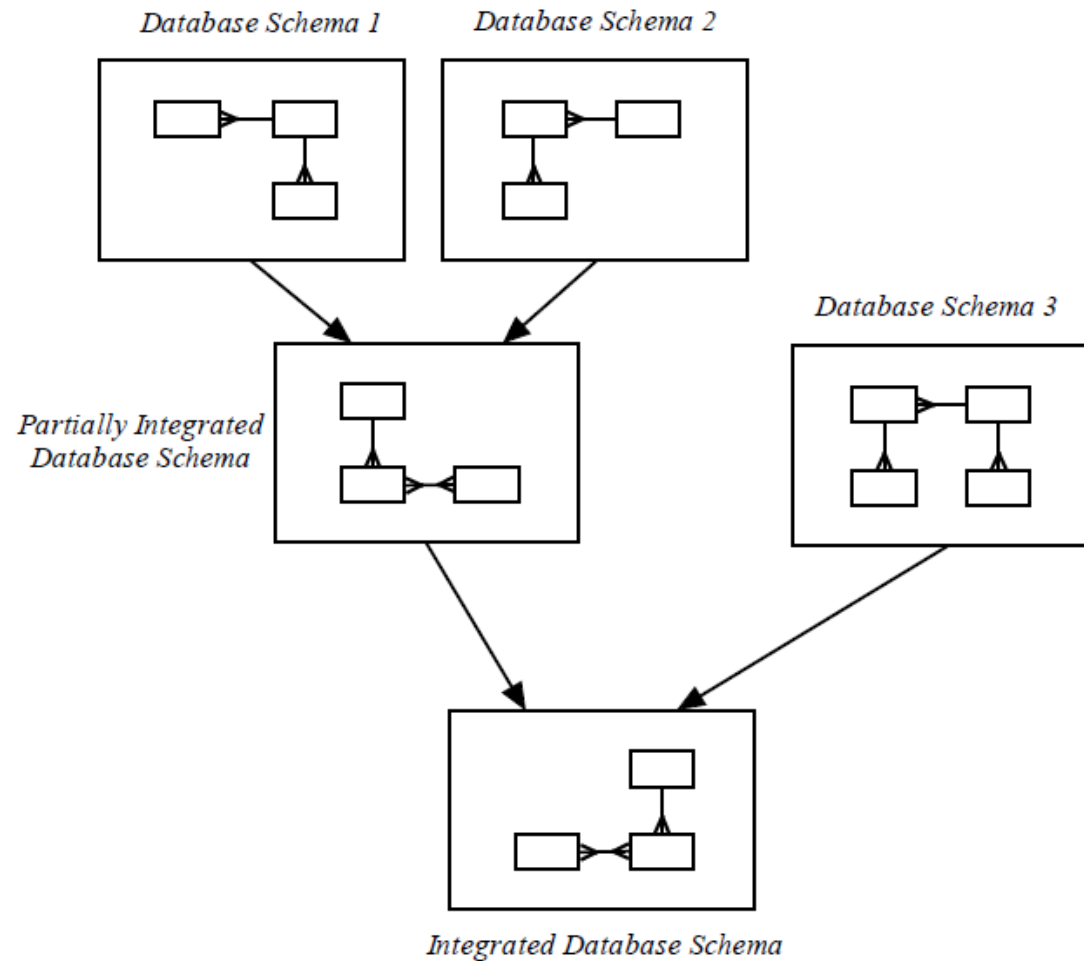
- All in one
- stejné koncepty modelovány odlišným způsobem



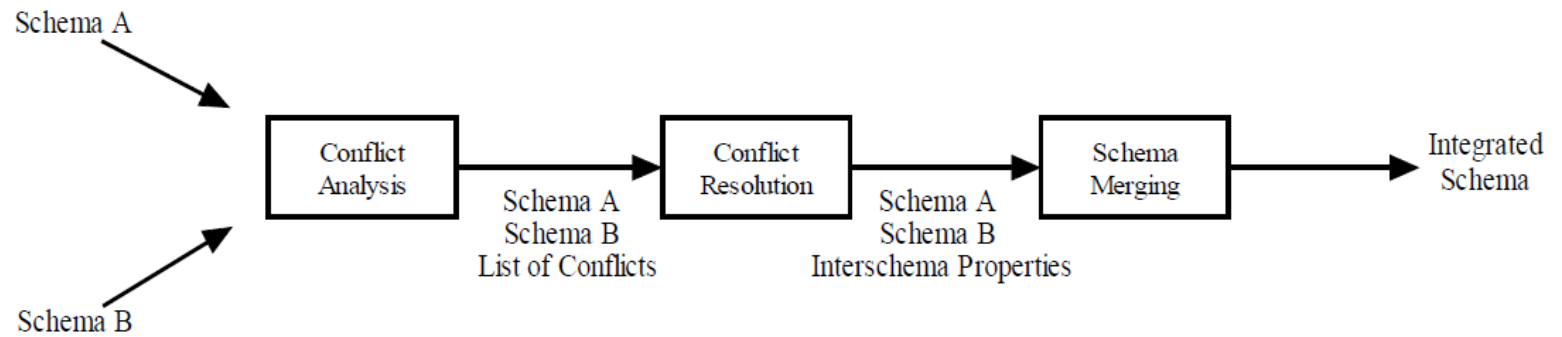
# All in one



# Stages



# Proces



# Řešení konfliktů - názvy

- Synonyma
  - Objekty, které jsou stejné, ale mají odlišné názvy
  - Například cestující a zákazník
- Homonymy
  - Objekty, které jsou různé, ale mají stejné názvy
    - Například publikace (= kniha) a publikace (= deník)

# Řešení konfliktů - názvy

Konflikty jmen způsobují problém, protože v systému mohou být duplikovány informace

integrovaná databáze. Je důležité identifikovat tyto datové položky v každé z nich

schéma, které ve skutečnosti představuje stejný koncept nebo by mělo být

znázorněné pomocí různých struktur v integrovaném schématu.

- Synonyma mohou být z databáze odstraněna přejmenováním konceptů

takže mají stejné jméno.

- Homonymy mohou být z databáze odstraněny přejmenováním konceptů

# Řešení konfliktů - strukturální

- Identické koncepty
  - Sloučeno
- Kompatibilní koncepty
  - Reprezentace jsou upraveny a sloučeny
- Nekompatibilní koncepty
  - Různé kardinality
  - Různé identifikátory
  - Reverzní vztahy podmnožiny

# Řešení konfliktů - strukturální

- Identické koncepty
  - Pokud se koncept v obou databázích neliší (shoduje) mohou se modely sjednotit (mergnout)
    - Například pokud má entita publikace stejnou strukturu v obou ER modelech, mohou se sjednotit

# Řešení konfliktů - strukturální

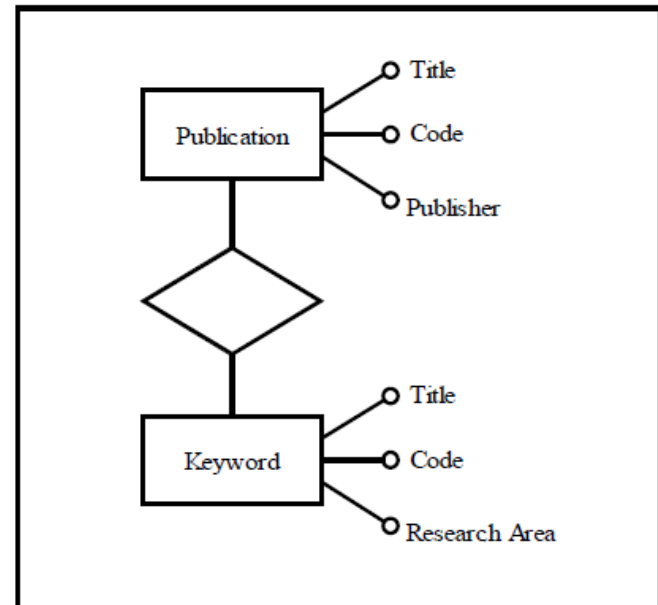
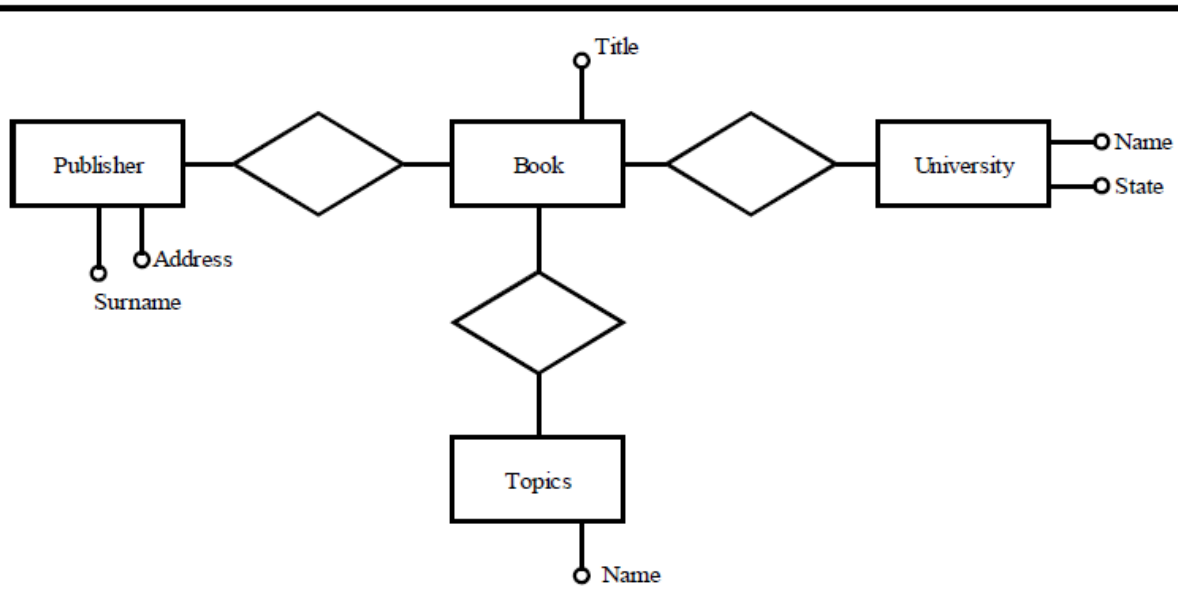
- Kompatibilní koncepty
  - Když se stejný koncept v odlišných databázích reprezentuje kompatibilními způsoby, tak je lze sloučit.
    - Například když je publikace entity reprezentována jako **atribut** v jedné databázi a jako **entita** v jiné databázi
      - mohou být sloučeny převedením atributu na entitu.



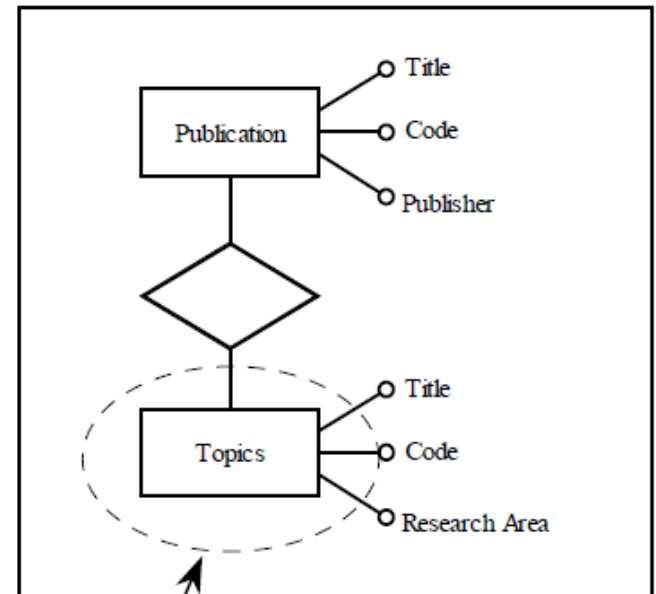
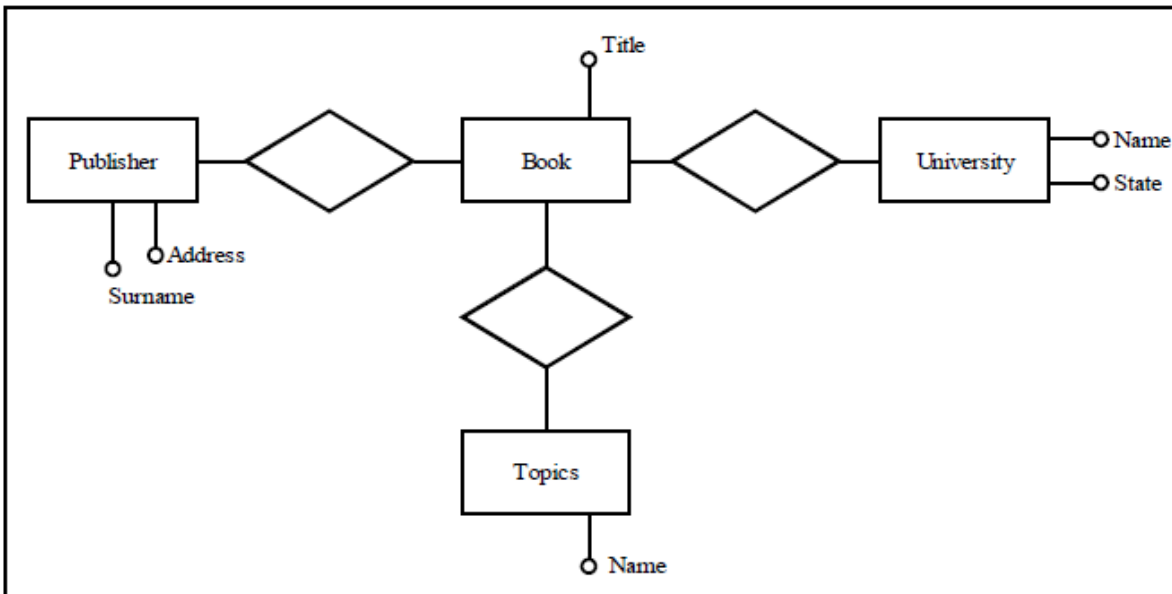
# Řešení konfliktů - strukturální

- Nekompatibilní koncepty Když se stejný koncept liší
  - databáze jsou reprezentovány pomocí různých struktur, je problém je spojit. Například :
    - Vztahy mohou mít různé kardinality (1:N a M:N)
    - Primární klíče se mohou lišit.
    - Nastavené vztahy se mohou obrátit (např. Projekty obsahují programy a programy obsahují projekty).
- Nekompatibilní návrhy musí být vyřešeny opětovnou analýzou dat a přizpůsobení jednoho nebo více schémat nebo vytvořením nového společného schématu.

# Example

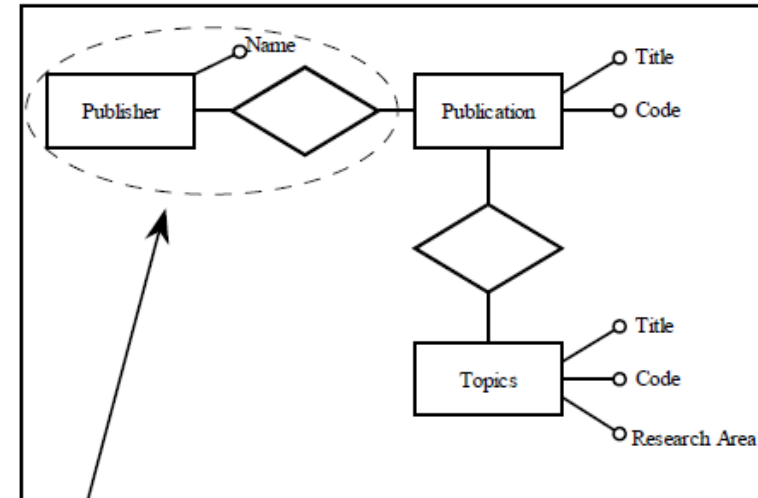
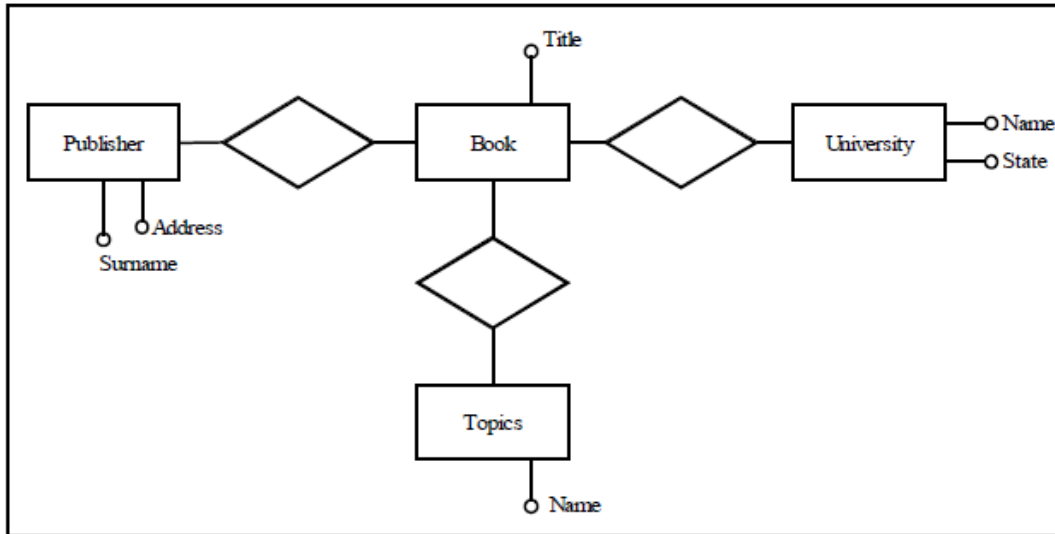


# Example



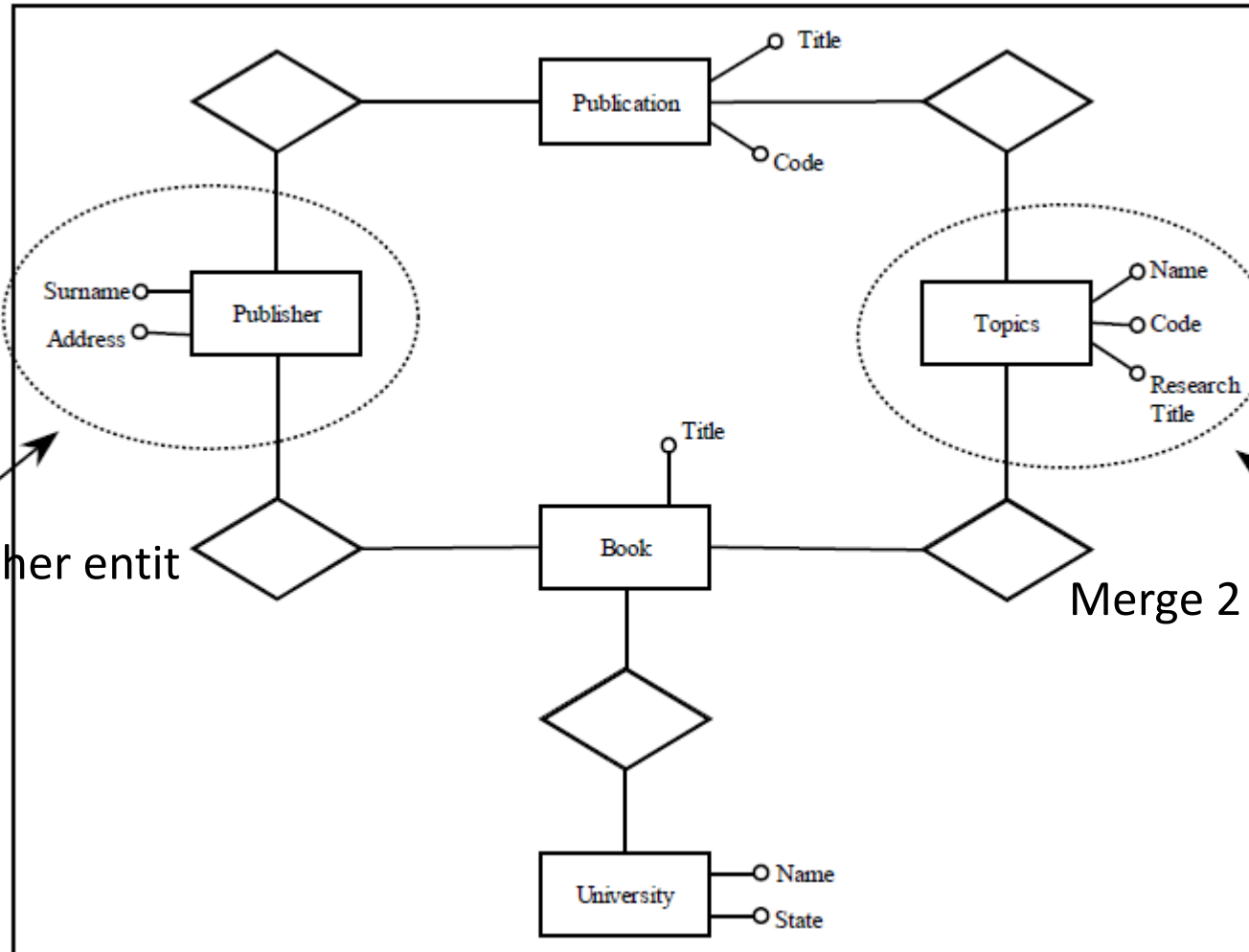
Přejmenovat entitu **Keywords** na Topics

# Example



Z **Publisher** atributu uděláme entitu

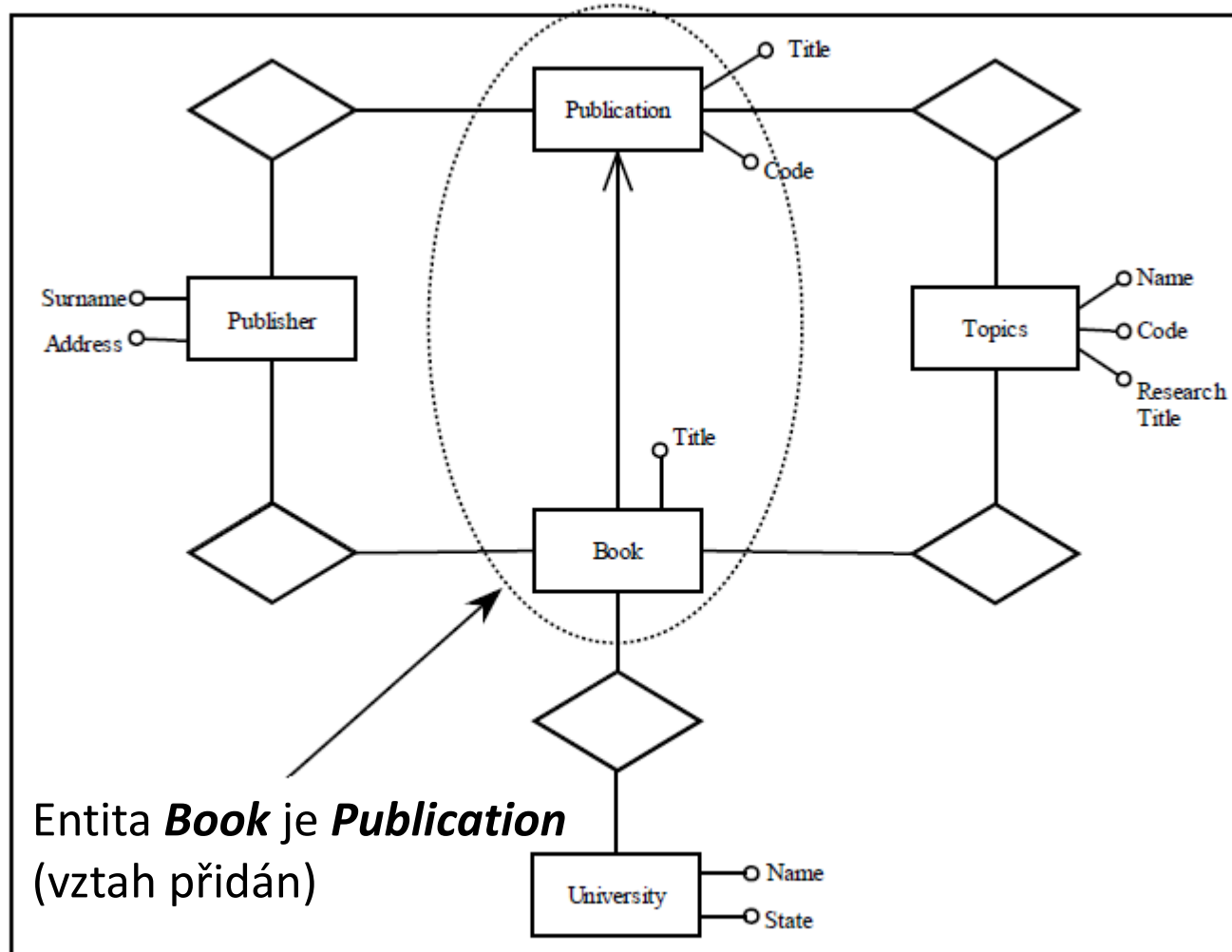
# Example



Merge 2 Publisher entit

Merge 2 Topics entit

# Example



# Example

