BIG DATA 6

Technológie spracovania veľkých dát

Peter Bednár, Martin Sarnovský

Integrovanie dát a distribuovanie správ

- Integrácia dát
  - Heterogenita dát
  - Prístup k dátam, harmonizácia dát
- Distribuovanie správ
  - Dátové kanály a zbernice
  - Fronty správ

Integrácia dát

- Aby sme mohli získať nové komplexné znalosti o danom fenoméne, je potrebné integrovať dáta z rôznych heterogénnych zdrojov
- Pri integrácií dát je potrebné prekonať dátovú heterogenitu a poskytnúť pre analýzu a spracovanie dát jednotné rozhranie

1. Syntaktická heterogenita
   - Rôzne komunikačné protokoly pre prístup k dátam, rôzne dopytovacie jazyky, rôzne dátové formáty, rôzne kódovanie údajov
2. Sémantická heterogenita
   - Rôzny model štruktúrovania dát, doménová nejednoznačnosť, dátová nejednoznačnosť

Syntaktická heterogenita - dátové zdroje

- Uložené priamo v súboroch na disku
- Uložené v relačných, alebo NoSQL databázach spravovaných rôznymi typmi systémov (ERP, CRM, QMS, SCM, ...)
- Poskytované cez webové služby (SOAP XML, REST)
- Na sieti poskytované cez internetové protokoly (HTTP, FTP, MAIL, ...) v rôznych formátoch
- Generované automaticky senzormi a poskytované cez rôzne komunikačné protokoly (Bluetooth Low Energy, MQTT, ...)

Syntaktická heterogenita - dátové formáty (1)

- Proprietárne formáty - poskytovateľ softvéru nemusí zverejniť špecifikáciu formátu
  - Dáta môžu byť len čiastočne čitateľné, problémy s verziami
  - Napr. staršie MS Office formáty .doc, .xls, .ppt
- Otvorené štandardy - existuje voľne dostupná špecifikácia
  - HTML, XML, JSON
  - Formát elektronickej pošty
  - Súborové formáty .docx (novší MS Office formát), .odt (OpenOffice formát), .pdf, .rtf
  - Často sú štandardizované medzinárodnými organizáciami RFC, ISO/IEC, OASIS

Syntaktická heterogenita - dátové formáty (2)

- Dáta môžu byť komprimované, alebo spojené do archívu - .zip, .gz, .bz2, .rar, .tar
  - znova môžu byť problematické proprietárne formáty
  - súčasť špecifikácie pre multimediálne súbory
- Na webe sa formáty rozlišujú podľa MIME (*Multipurpose Internet Mail Extensions*) štandardu
  - MIME spravuje organizácia IANA (*Internet Assigned Numbers Authority*)
  - Popis formátov zahŕňa odporúčanú súborovú príponu a jedinečné označenie typu, ktoré sa uvádza v hlavičke HTTP protokolu, alebo v prílohe pošty
  - Aktuálny zoznam formátov: http://www.iana.org/assignments/media-types/

Sémantická heterogenita (1)

- Údajová heterogenita
  - Napr. používanie rôznych merných jednotiek
- Doménová heterogenita
  - Významovo ten istý typ, dátový atribút alebo hodnota môže byť rôzne pomenovaná (napr. použitie synoným, skratiek)
  - Typy, dátové atribúty, alebo hodnoty môžu byť rovnako pomenované, ale majú pre rôzny zdroj rôzny význam
    - Kontextová nejednoznačnosť – rovnako označený atribút pre rôzne typy, alebo rovnaká hodnota pre rôzne atribúty

Sémantická heterogenita (2)

- Štrukturálna heterogenita
  - Rôzny spôsob štruktúrovania dát (napr. rôzna úroveň normalizácie do relačnej tabuľky, vnorenenie elementov v JSON/XML)
  - Jedna entita v jednej dátovej schéme môže byť reprezentovaná viacerými entitami v druhej schéme (napr. v jednej databáze máme tel. číslo a v druhej domáce, pracovné, mobilné číslo)
  - Záznamy o danej entite majú rôzne ID v rôznych databázach – mapovanie identity

Sémantická heterogenita – príklad (1)

- Chceme vyjadriť v relačnej databáze fakt, že predajca má na starosti danú oblasť

1. Ak má predajca len jednu oblasť:

Predajca		
ID	meno	oblasť

- Alebo môže byť daný fakt uložený v inom zázname, napr. v objednávke (nenormalizovaná schéma)

Objednávka			
ID	cena	predajca	oblasť

Sémantická heterogenita – príklad (2)

- Alebo môže byť oblasť zakódovaná do spoločnej hodnoty s ďalším atribútom, napr. priradenie = (oblasť, typ produktu):

Predajca		
ID	meno	priradenie

- Alebo môže byť naopak zakódovaná viacerými atribútmi:

Objednávka			
ID	meno	krajina	región

Sémantická heterogenita – príklad (3)

- Alebo môže byť tento fakt odvodený implicitne z iného atribútu (napr. ak predajcovia z jedného oddelenia majú pevne priradenú oblasť):

Predajca		
ID	meno	oddelenie

- Alebo môže byť dáta rozdelené do dvoch tabuliek podľa oblastí:

Predajca : Oblasť 1	
ID	meno

Predajca : Oblasť 2	
ID	meno

Sémantická heterogenita – príklad (4)

- Alebo môže byť hodnoty uložené parametricky:

Predajca		
ID	atribút	hodnota
1	meno	Smith
1	oblasť	NY

2. Ak má naopak každá oblasť len jedného predajcu:

Oblasť		
ID	názov	predajca

Sémantická heterogenita – príklad (5)

3. Ak je relácia N:M:

Oblasť	
predajca	oblasť

- Ak je len niekoľko oblastí:

Predajca				
ID	meno	oblasť 1	oblasť 2	...

- a veľa ďalších spôsobov!

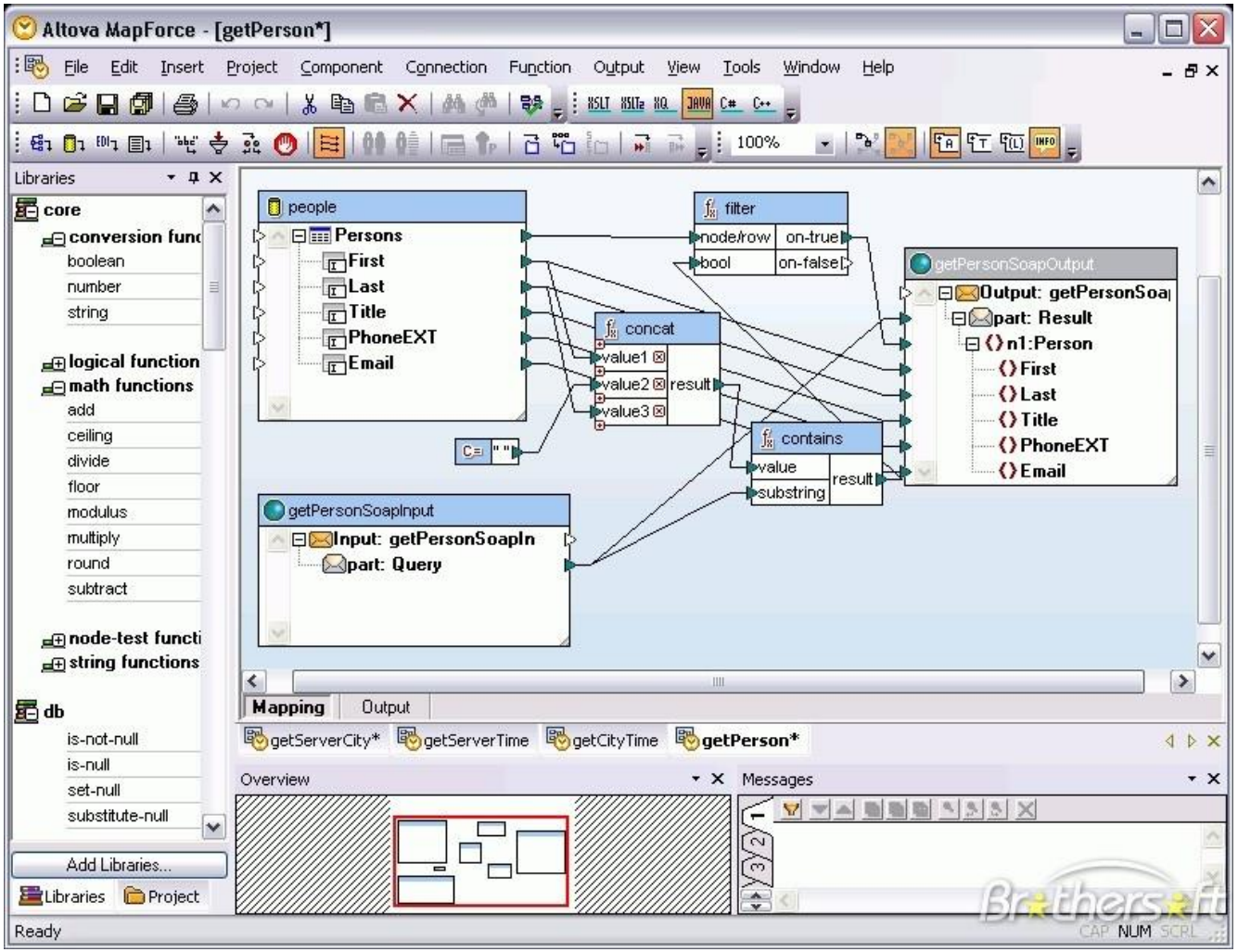
Dátové mapovanie

- Dáta z rôznych dátových zdrojov sa namapujú na jednotnú dátovú schému ku ktorej je možné jednotne pristupovať
- Dátové mapovanie je lokálne pre jednotlivé zdroje:
  - Pre nový zdroj sa navrhne samostatné mapovanie
  - Ak sa zmení schéma jedného zdroja, neovplyvní to mapovanie ostatných zdrojov
- Mapovanie schémy
  - Zjednotená schéma, pravidlá pre transformovanie dát
- Mapovanie dát
  - Zjednotený dátový model

Nástroje na mapovanie dát (1)

- Manuálne mapovanie
  - Vizuálne nástroje, kde môže ručne vývojár navrhnúť mapovanie z jednej schémy/formátu do inej, výstup je v podobe vygenerovaných transformačných programov alebo pravidiel (napr. v XSLT)
- Dátovo-orientované mapovanie
  - Na dáta sa priamo aplikujú rôzne heuristiky a automatické metódy, ktoré automaticky vygenerujú transformačné pravidlá. Využívajú sa aj techniky data/text miningu
- Sémantické mapovanie
  - Využívajú sa ontológie, ktoré definujú pojmy z danej domény, ich synonymné označenie a vzťahy medzi pojmami (napr. last name = surname *part of* person name)

Nástroje na mapovanie dát (2)



Zjednotený dátový model

- Jednotná reprezentácia dát v pamäti, jednotný formát pre ukladanie dát do súboru a pre prenos cez sieť
- Dátové hodnoty:
  - Atomické hodnoty: čísla (celé, desatinné), reťazce, znaky, Boolovské hodnoty, dátum a čas, časové intervaly
  - N-tice
  - Mapy kľúč:hodnota
  - Polia/zoznamy (indexované)
  - Množiny (prvky sa nemôžu opakovať)
  - Kolekcie (prvky sa môžu opakovať)

Dátové formáty pre Veľké dáta (1)

- JSON/XML
  - Čitateľné (textový formát), dobrá podpora existujúcich nástrojov
  - Neefektívne pre veľké dáta, nepodporujú priame dopytovanie
- Formáty navrhnuté pre Veľké dáta:
- Avro
  - Binárny formát
  - Dáta sa ukladajú po riadkoch
  - Dátová schéma je uložená ako súčasť dát – podporujú sa verzie schémy
  - Podporuje kompresiu dát po blokoch
  - Dáta sa dajú rozdeliť na bloky po riadkoch, ktoré je možné distribuovať samostatne (dôležité pre MapReduce)

Dátové formáty pre Veľké dáta (2)

- Parquet
  - Binárny formát
  - Dáta sa ukladajú po stĺpcoch – efektívne je ich možné komprimovať:
    - Číselné dáta sú usporiadané, zakódujú sa rozdiely medzi nasledujúcimi hodnotami s minimálnym počtom bitov, napr.:

      6, 8, 10, 12, 12 -> 6, 2, 2, 2, 0
    - Pre reťazce sa používajú slovníky ktoré mapujú textové hodnoty na číslo
  - Čiastočne je možné rozšíriť schému dát (pridať na koniec jeden stĺpec)

Dátové formáty pre Veľké dáta (3)

- Optimized Row Columnar
  - Binárny formát
  - Dáta sa ukladajú po blokoch riadkoch a v každom bloku sú dáta uložené po stĺpcoch
  - Blok uchováva aj základné indexovanie hodnôt, takže je možné pri čítaní efektívne preskočiť nepotrebné bloky riadkov
  - Ako súčasť dát sa ukladajú aj základné štatistiky (min, max, suma, počet)
  - Dáta je možné rozdeliť po blokoch

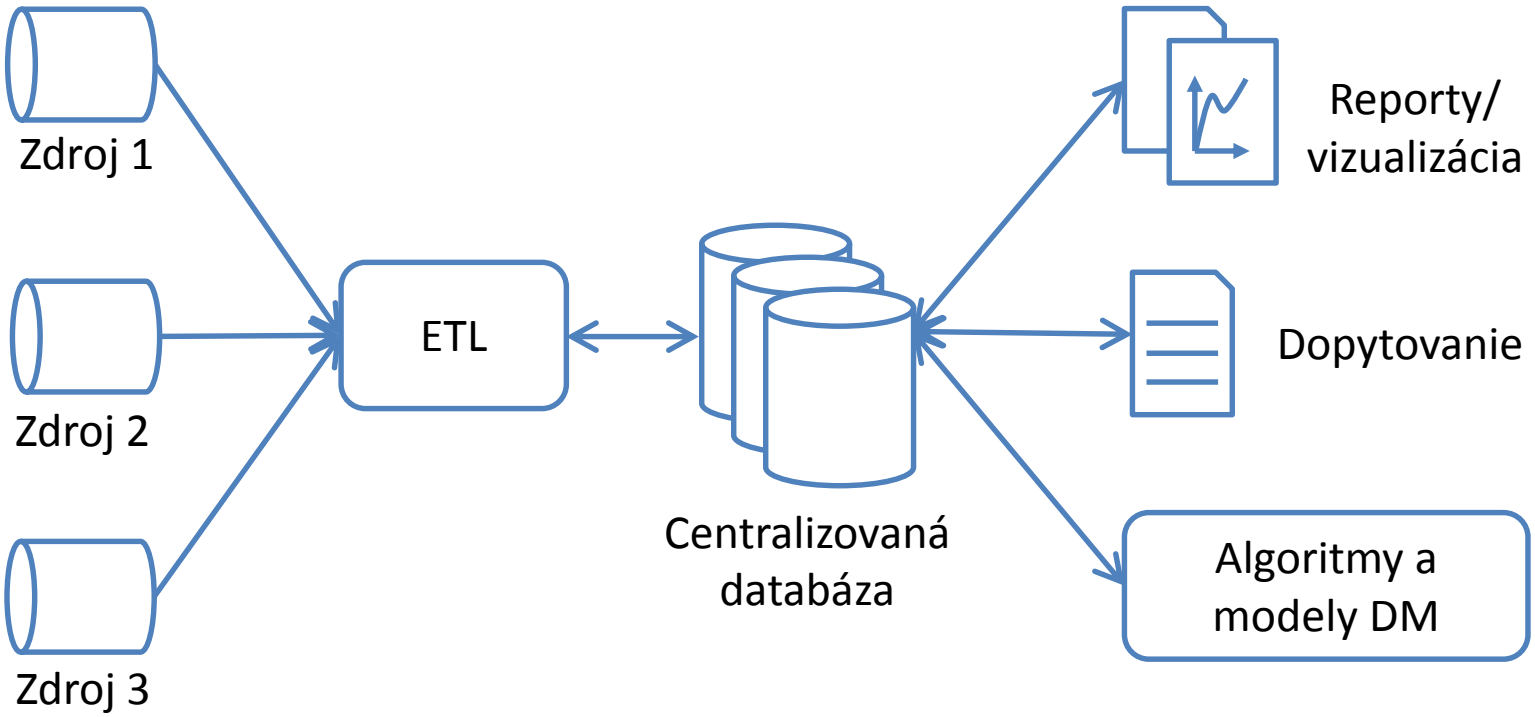
Architektúry pre integrovanie dát

- Požiadavky: je potrebné zintegrovať
  - Viacero rôznych dátových zdrojov s heterogénnymi dátami
  - Viacero rôznych klientov, ktorý majú rôzne požiadavky na prístup k dátam (metódy data miningu, interaktívne dopytovanie – „ad-hoc“, vizualizácia a reportovanie, ...)
- Základné rozdelenie:
  - Centralizovaný prístup k dátam
  - Federovaný prístup k dátam

Centralizovaný prístup k dátam (1)

- Dáta sú pomocou ETL operácií:
  1. Načítané z pôvodného zdroja
  2. Transformované do spoločnej schémy – mapovanie dát
  3. Uložené v centrálnej distribuovanej databáze, ktorá poskytuje jednotné rozhranie pre prístup k dátam
- Architektúra používaná pri dátových skladoch
- Dáta je potrebné synchronizovať – aktualizovať centrálnu databázu keď sa zmenia dáta na zdroji
- Je potrebné mať dostatočné zdroje pre centralizované spracovanie dát

Centralizovaný prístup k dátam (3)



Federovaný prístup k dátam (1)

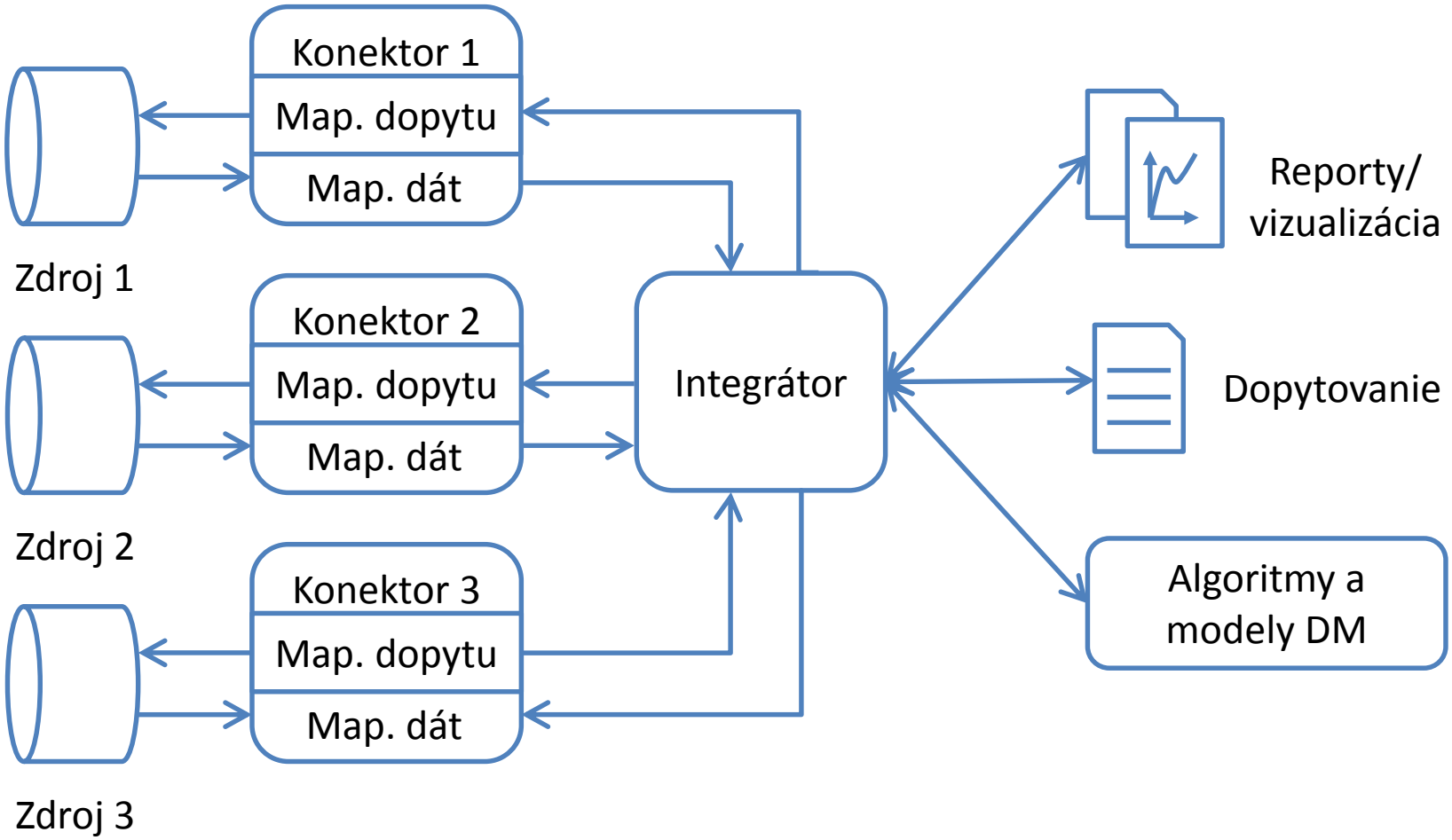
- Základné komponenty
  - Integrátor – poskytuje jednotné rozhranie pre klientov
  - Konektory pre dátové zdroje – mapujú dopyty/dáta do/z lokálnej schémy

1. Klient odošle dopyt v zjednotenej schéme na rozhranie Integrátora
2. Integrátor rozošle dopyt na jednotlivé konektory
3. Konektor prevedie dopyt do lokálnej schémy a dopytovacieho jazyka a získa relevantné dáta zo zdroja
4. Konektor premapuje lokálne dáta do zjednotenej schémy
5. Integrátor spojí čiastkové výsledky z jednotlivých zdrojov a vráti výsledné dáta klientovi

Federovaný prístup k dátam (2)

- Dáta sú uložené a spravované v pôvodnom zdroji – nie je potrebná synchronizácia
- Zdroj nemusí úplne podporovať všetky možnosti zjednoteného dopytovacieho prostredia – obmedzený prístup k dátam
- Nie je možné použiť ak sú dáta na zdroji dostupné iba dočasne

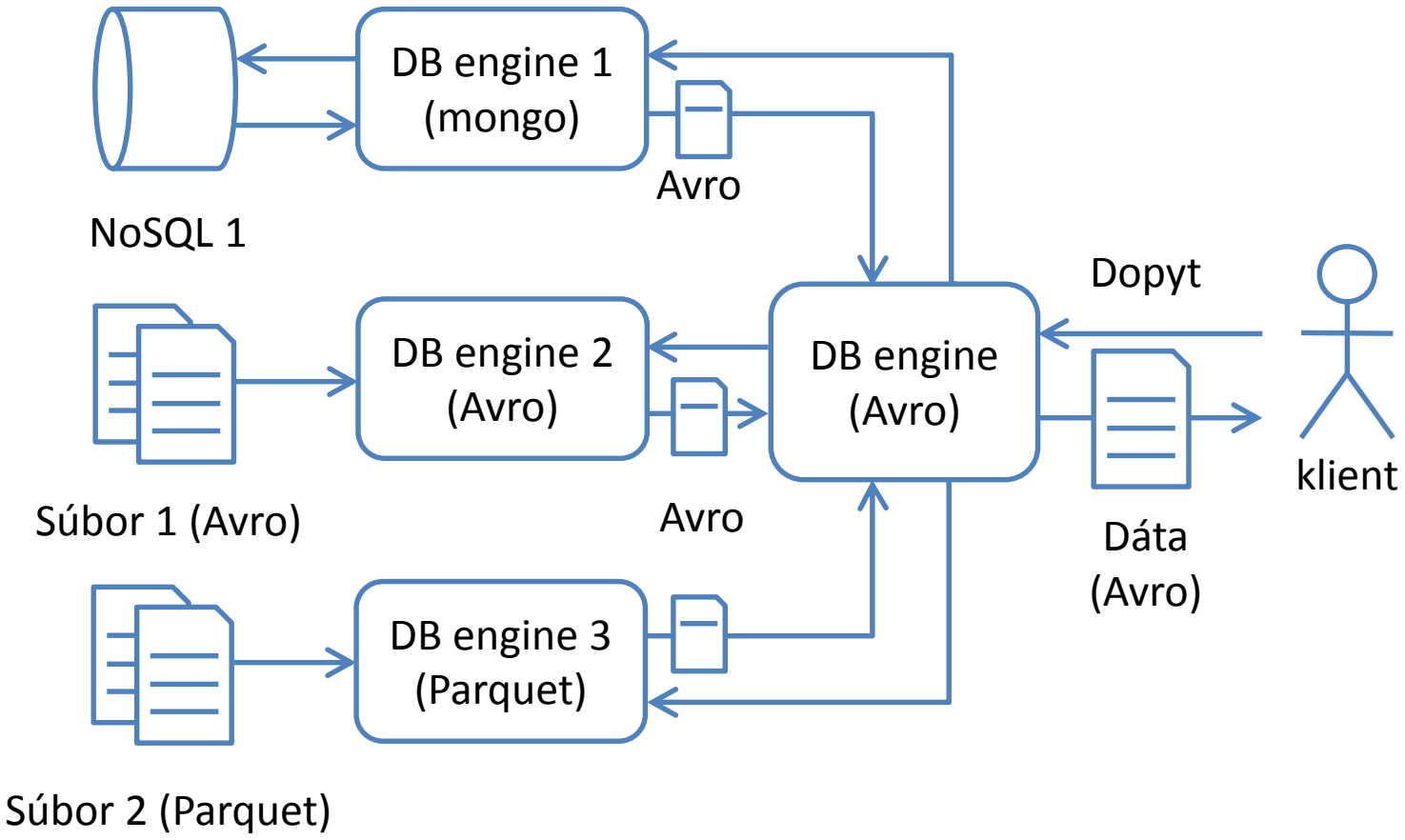
Federovaný prístup k dátam (3)



Federovaný prístup k dátam (4)

- Databázový stroj (*engine*) pre dopytovanie
  - Konektor, ktorý poskytuje jednotný dopytovací jazyk a dátový formát výsledkov
  - Pri dopytovaní nie je harmonizovaná schéma dát
  - Ako dátový zdroj môže byť pripojený priamo súbor vo formáte pre Veľké dáta – stroj slúži na efektívne čítanie dát zo súboru a vyhodnocovanie dopytov
  - Ak sa rovnaký formát použije na prenos dát medzi Integrátorom a Konektormi, ten istý stroj je možné použiť aj v Integrátore pre agregovanie medzivýsledkov

Federované dopytovanie



Prepojenie komponentov (1)

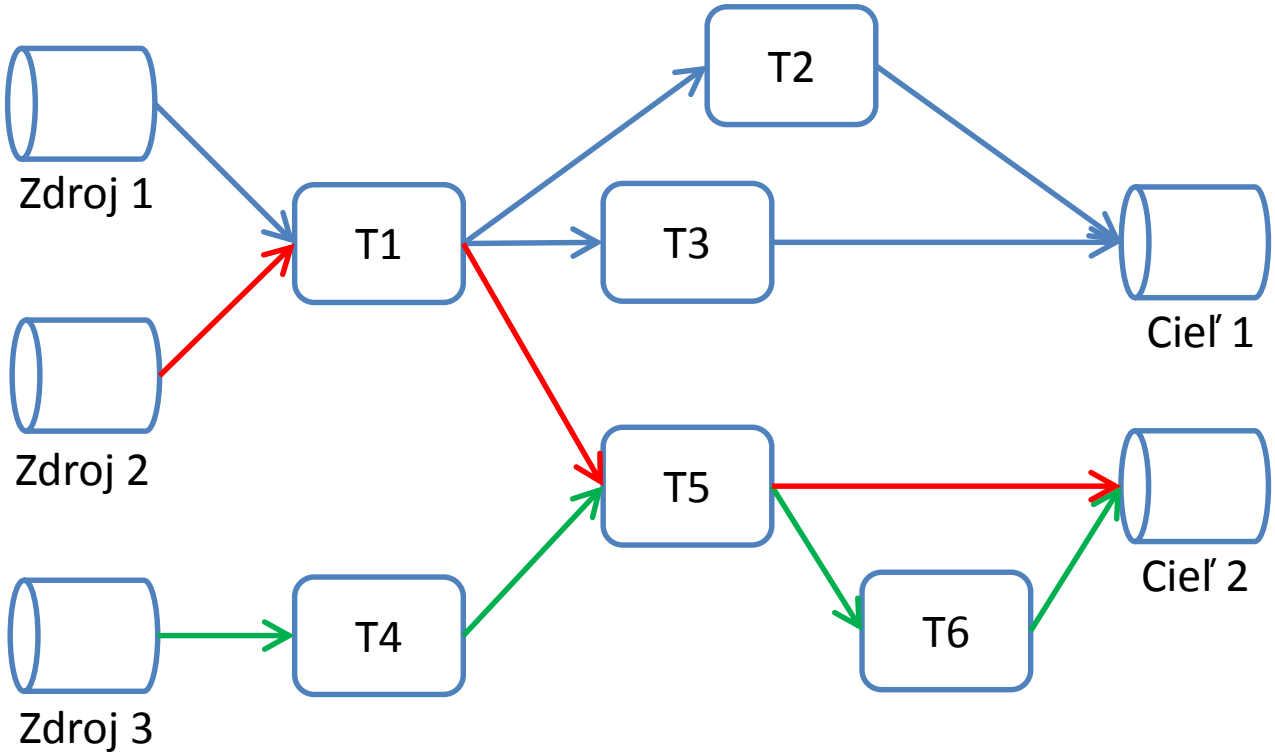
- Pri integrovaní dát je potrebné prepojiť veľký počet softvérových komponentov:
  - Konektory pre pripojenie zdrojov s rôznymi komunikačnými protokolmi
  - Transformácie – dátové mapovanie môže byť zložitý proces, ktorý je potrebné rozdeliť do viacerých krokov (dátových transformácií)
  - Transformácie môžu bežať v samostatných komponentoch distribuovane, dáta sú vymieňané pri spracovávaní v správach odosielaných medzi komponentami

Prepojenie komponentov (2)

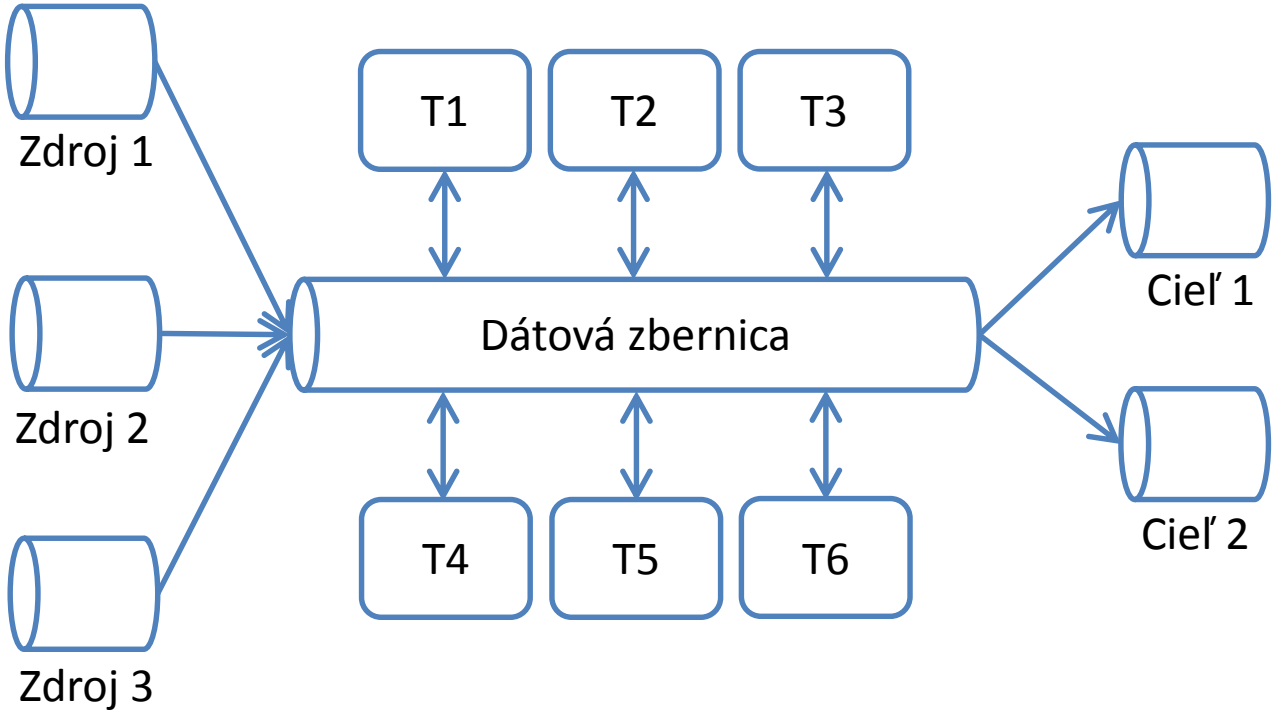
- Niektoré transformácie môžu byť zdieľané pri spracovávaní rôznych typov dát – zložitá štruktúra prepojenia medzi komponentami – komponenty musia smerovať správy podľa typu dát
- Ak sa zmení schéma niektorého zo zdrojov resp. ak sa pridá nový dátový zdroj, spracovanie je potrebné rozšíriť

- Komponenty implementujú iba požadovanú funkčnosť, o smerovanie dát sa stará dátová zbernica, ktorá udržiava konfiguráciu ako majú byť dáta smerované medzi komponentami
- Používa sa stratégia „publikovania-odoberania“ správ

Priame prepojenie komponentov



Prepojenie komponentov - dátová zbernica (1)



Základné komponenty dátovej zbernice (1)

- Zdroj správ
  - Napája sa na zdroj zvoleným komunikačným protokolom
  - Prevedie dáta to jednotného dátového formátu a odošle ich vo forme správy do zvoleného dátového kanála/kanálov
  - Zdroj správ harmonizuje dáta na syntaktickej úrovni – všetky ostatné komponenty už majú na vstupe jednotný dátový formát (schéma sa však môže líšiť podľa zdroja a typu dát)
- Cieľ správ
  - Prijíma dáta z dátového kanála
  - Zakóduje dáta z jednotného dátového formátu do požadovaného komunikačného protokolu
  - Odosiela dáta do cieľa dát (napr. databázu, službu, e-mailový server, a pod.)

Základné komponenty dátovej zbernice (2)

- Dátový kanál
  - Pomenované prepojenie medzi komponentami, kanál má zdrojové a cieľové pripojenie
  - Pri pripojení komponentu sa určí z ktorého kanála komponent číta zdrojové dáta a kam zapisuje výstupné dáta, to kam budú smerované závisí na konfigurácii cieľového pripojenia
  - Jeden kanál môže prepájať viacero zdrojových a cieľových komponentov (dáta z každého zdrojového komponentu sa odošlú každému cieľovému)

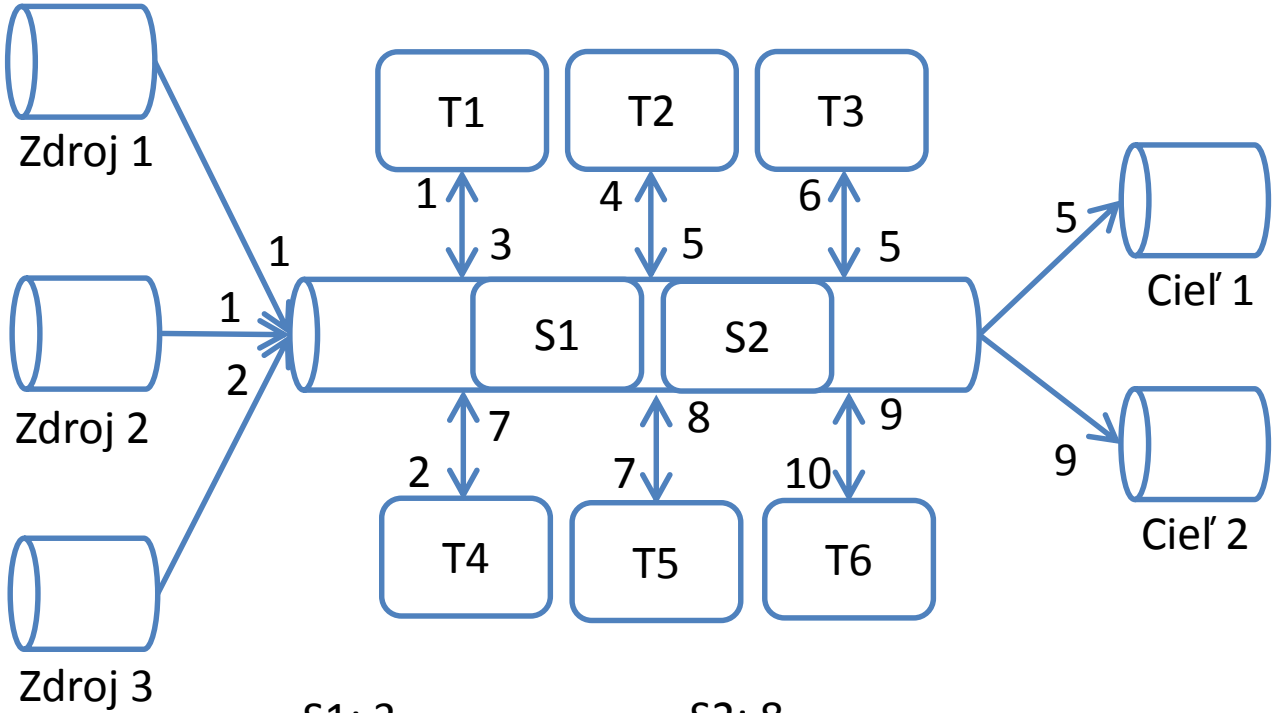
Základné komponenty dátovej zbernice (3)

- Smerovací selektor
  - Dynamicky smeruje dáta podľa ich typu, alebo podmienky ktorá testuje obsah dát zo zdrojového kanála do jedného, alebo viacerých cieľových kanálov
  - Môže slúžiť aj na filtrovanie dát
  - Môže slúžiť aj na zabezpečenie spoľahlivosti alebo škálovateľnosti spracovania, napr. ak zlyhá smerovanie do jednej komponenty, selektor ju môže presmerovať do ďalšej, ktorá vykonáva tú istú transformáciu

Základné komponenty dátovej zbernice (4)

- Externé komponenty – transformácie
  - Sú pripojené k jednému zdrojovému a jednému cieľovému kanálu
  - Z jednej správy môžu vygenerovať viac (všetky sa odošlú do cieľového kanála, o ich smerovaní rozhoduje konfigurácia ktorá je nezávislá na komponente)

Prepojenie komponentov - dátová zbernica (2)



S1: 3
podmienka 1 : 4
podmienka 2 : 6
podmienka 3 : 7

S2: 8
podmienka 1 : 9
podmienka 2 : 10

Komunikácia cez dátový kanál

- Z pohľadu cieľovej komponenty – klienta správ:
- Pull stratégia
  - Klient inicializuje komunikáciu a ak sú dostupné dáta, tak ich prevezme
  - Klient sa musí periodicky dopytovať na dostupnosť nových dát
  - Dátový zdroj musí dáta uchovávať pokiaľ si ich klient neprevezme
- Push stratégia
  - Ak sú dostupné dáta, zdroj dát inicializuje komunikáciu a pošle ich klientovi
  - Klient nemusí byť aktuálne pripravený dáta prijať, zdroj sa môže znovu pokúsiť dáta poslať (musí ich dočasne uchovať)

Apache Flume

- Distribuovaný systém (komponenty môžu byť spustené na rôznych počítačoch), ktorý zabezpečuje spoľahlivé doručovanie správ
- Source – zdroj správ, Sink – cieľ správ, Channel – dátový kanál
- Zdroje pre rôzne protokoly a formáty (napr. Avro, SYSLOG, REST/HTTP, Twitter, ...)
- Zápis dát do relačných alebo NoSQL databáz (JDBC, Elasticsearch, Hive), HDFS
- Programátorské rozhranie pre selektory

Fronty správ

- Fronta správ je dátový kanál, ktorý umožňuje dočasne uchovávať prijaté správy, kým si ich neprevezmú klienti
- Umožňujú asynchrónnu výmenu dát
- Správy sú radené a distribuované podľa poradia prijatia – First In First Out
- Ako dlho sú správy uchovávané je možné konfigurovať:
  - Podľa časového intervalu (time-out)
  - Podľa kapacity kanálu – napr. max. počet správ, max. celková veľkosť dát
- Správy môžu byť uchované v pamäti, perzistentne na disku, alebo kombinovane (napr. ak sa prekročí kapacita pamäte, tak sa staršie správy uložia na disk)

Apache Kafka

- Distribuovaný systém, ktorý udržuje fronty správ a umožňuje asynchrónnu komunikáciu medzi komponentmi
- Klienti publikujú alebo odoberajú správy z/do tzv. *topicu* (označenie fronty)
- Správy sú udržiavané stanovený časový interval (*retention time*)
- Správy sú perzistentne uložené na viacerých uzloch v transakčnom logu
  - Zabezpečuje spoľahlivosť