

# Zpracování proudů dat

Jaroslav POKORNÝ<sup>1</sup>, Václav SNÁŠEL<sup>2</sup>

<sup>1</sup>*Katedra softwarového inženýrství, MFF UK Praha  
Malostranské nám. 25, 118 00 Praha  
pokorny@ksi.ms.mff.cuni.cz*

<sup>2</sup>*Katedra informatiky, VŠB-TU Ostrava  
Třída 17. listopadu, 708 33 Ostrava  
vaclav.snasel@vsb.cz*

**Abstrakt.** Proudů dat se objevují v nových aplikačních oblastech jako jsou webové aplikace, finanční data, environmentální data, data pro zajištění bezpečnosti objektů apod. Jejich zpracování je často nepřetržité, ať už při vzniku dat nebo při dotazování. Procesy, které při těchto činnostech vznikají vedou v monitorovacích systémech ke vzniku událostí a spouštění dalších procesů. V pozadí dat jsou často senzory a senzorové sítě. Z hlediska databází přináší zpracování proudů dat nové podněty. Data nelze ukládat všechna na vnější paměti, dotazy mohou být nekonečné, uplatňují se zde aproximační metody a nové přístupy k optimalizaci. Cílem článku je ukázat tyto nové směry, sumarizovat dosažené výsledky a formulovat problémy, které jsou výzvou pro informatiku zejména v oblasti databází.

**Klíčová slova:** proud dat, senzor, senzorová síť, nanotechnologie, dotazování

## 1 Úvod

Vývoj hardwarové i softwarové složky informačních systémů, kvalitativní i kvantitativní rozšiřování jejich databází a zvyšující se požadavky a nároky uživatelů představují trvalý tlak na vývoj nových modelů vyhledávání a zpracování informací. Objevují se i zcela nové výpočetní a datové modely, které velmi těsně svazují softwarovou a hardwarovou složku zpracování dat [8], [31]. Nové modely musí reagovat mimo jiné na heterogenost dat, která se projevuje různým stupněm jejich strukturalizace a formalizace. Dalším důležitým aspektem je dynamičnost některých nových zdrojů dat. Tato dynamičnost hraje výraznou úlohu v oblasti senzorových sítí a projevuje se nejenom změnou hodnot, ale i změnou topologie.

I zdánlivě jednoduché úlohy se však stávají obtížnými, pokud jsou zpracovávána data velmi rozsáhlá. Nároky uživatelů jsou akcelerovány nejen rostoucím rozsahem dat a technickými parametry počítačů, ale i potřebou rychle proniknout do podstaty nalezených informací a znalostí. Důležitým dílčím úkolem je tedy uživatelsky pohodlná prezentace nalezených informací a znalostí, stále častějším požadavkem je poskytnutí přehledné a vhodným způsobem strukturované zprávy.

## *Zpracování proudů dat*

Je zřejmé, že je stále aktuálnější zabývat se otázkami efektivního zpřístupnění a zpracování rozsáhlých dynamických dat s cílem ulehčit rutinní práci odborníkům a vytvořit pro ně efektivní nástroje pro podporu rozhodování.

Rozsah a dynamičnost dat se v poslední době zvětšily natolik, že se začaly negativně projevovat statistické vlastnosti těchto dat vycházející z tradičních metod uložení a zpracování. Tato skutečnost vedla k tomu, že bylo možné použít nové metody vycházející ze statistiky, lineární algebry, neuronových sítí a metod řazených mezi bio-inspirované metody [26].

Mezi hlavní reprezentanty dynamických dat patří proudy dat a proudy událostí. Proudů dat [33] představují zcela nový typ dat, lišící se podstatným způsobem od dosud běžných pohledů na data. Dosud uvažované modely dat předpokládají víceméně pevnou strukturu dat uloženou persistentně na vnějších pamětech počítače. Tato data je možno zpracovávat v off-line režimu, náhodný přístup je omezen pouze technickými možnostmi vnějších pamětí. Naproti tomu data v proudě mají následující vlastnosti:

- přicházejí neustále (on-line),
- nelze předpokládat nic o pořadí dat, vstupujících do zpracování ať už v jednom proudě dat nebo v rámci více proudů,
- proudy dat mají obecně neomezenou velikost,
- jakmile jsou data z proudě zpracována, jsou smazána nebo archivována – nelze je jednoduchým způsobem opětovně získat, jako je tomu v případě uložení v klasických databázích,
- může se měnit jejich struktura (topologie).

Proudů dat se objevují v mnoha aplikačních oblastech, jako např. provoz Internetu, monitorování a analýza proudů dat získaných pomocí senzorů [10], [24] (data o pohybujících se objektech z provozu na dálnicích, turistů v krajině, geografická data, medicínská data), zpracování záznamů transakcí (Web logy, záznamy telefonních hovorů a použití kreditních karet), vývoj cen na burze, bezpečnostní aplikace. Jako příklady již existujících aplikací lze uvést:

- Traderbot [36] je webové aplikace umožňující vyhledávání v reálném čase nad proudy dat v oblasti finanční.
- Moderní bezpečnostní aplikací používající sofistikovaná pravidla pro proudy paketů je iPolicy Networks [26].
- Monitorování chování uživatelů v rozsáhlých webech [36], jako jsou portály a vyhledávače. Uživatelé svým pohybem po webu vytváří proud dat přechodů mezi stránkami tzv. clickstream. Analýzou tohoto proudě dat je možné web aktivně adaptovat pro potřeby konkrétního uživatele, případně sledovat obecné mechanismy chování uživatelů.
- Monitorování bojových akcí v moderních armádách. Tzv. „smart uniform“ umožňuje monitorovat zdravotní stav vojáka [3], [31].

Dalšími příklady jsou data systémů monitorujících provoz počítačových sítí či vyhodnocující digitální videosignál z bezpečnostních kamer apod. V průmyslu se jedná typicky o data vznikající při výrobě, v nepřetržitých provozech atd. V současných aplikacích proudě dat lze nalézt i proudy XML dat obsahující do sebe zahrnuté elementy odpovídající např. procesům monitorovaným v prostředí UNIX [9].

Z aplikací vyplývá, že proudy dat lze rozdělit do dvou základních kategorií: *proudy transakční dat* a *proudy naměřených dat*. Je zřejmé, že data těchto aplikací představují nepřetržitý, v čase se měnící proud dat, který je sice možné uložit v tradičních databázích, ovšem ne vždy uspokojivým způsobem. Vždy se setkáme s proudy dat, které buď není nutné ukládat do databází, nebo to dokonce není pro jejich rozsah ani fyzicky možné. Tradiční databáze nejsou stavěny na rychle se měnící proudy dat. Jejich úkolem je data shromažďovat a v daných časových okamžicích vyhodnocovat dotazy nad aktuálním stavem databáze. Získané odpovědi jsou úplné a přesné, nejsou zde dovoleny žádné aproximace, které jsou pro proudy dat často nutné, protože dotazy mohou být *dlouhotrvající* (vyhodnocují se ve velkém časovém úseku) nebo dokonce *nepřetržitě* (do dotazu vstupuje nepřetržitě proud dat).

Cílem článku je ukázat tyto nové směry, sumarizovat dosažené výsledky a formulovat problémy, které jsou výzvou pro informatiku zejména v oblasti databází. V sekci 2 vysvětlíme detailněji významný zdroj proudících dat, jako jsou senzory a sensorové sítě. Sekce 3 je věnována zpracování proudů dat a novým databázovým architekturám vhodným k jeho realizaci. V sekci 4 jsou zavedeny některé nové pojmy v souvislosti s dotazováním nad proudy dat. Diskutovány jsou též možnosti použití SQL pro tento účel. V sekci 5 se zaměříme na speciální typ proudů dat obsahující XML data. Sekce 6 uvádí do problematiky dolování proudů dat. Sekce 7 je věnována stručnému přehledu akademických a komerčních aktivit věnovaných dané problematice. Závěr nabízí pohled na budoucí možnosti zpracování proudů dat.

## **2 Senzory a sensorové sítě**

Mezi novými ITC technologiemi zaujímá v kontextu nových požadavků na databáze čelné místo levná mikrosenzorová technologie, která umožňuje většině objektů podávat ve formě proudu dat v reálném čase zprávy o hodnotách jejich atributů, což jsou např. fyzikální veličiny jako teplota, tlak, stav nebo umístění (např. pomocí globálního pozičního systému). Tyto informace mají podporovat aplikace, jejichž hlavním účelem je monitorovat tyto atributy. Senzory mohou zřejmě produkovat nepřetržitě, možná nekonečné proudy dat.

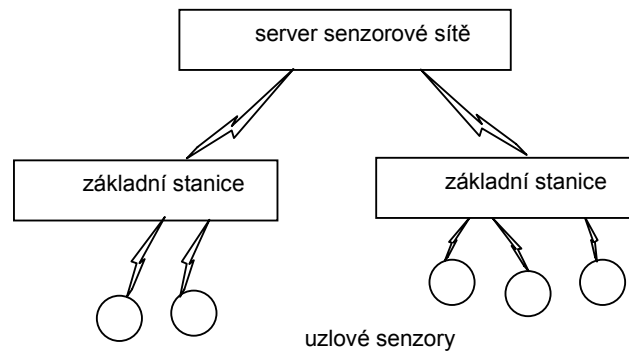
*Senzor* je bezdrátové zařízení s vlastním energetickým zdrojem. Složitějším zdrojem proudů dat jsou *sensorové sítě*. Tyto sítě jsou tvořeny množstvím malých, soběstačných počítačů, vyrobených na bázi nanotechnologií, vybavených senzory. Sensorové sítě vytvářejí nové požadavky na správu dat. Sensorová data se pomocí bezdrátového přenosu přenášejí do dalšího zpracování, protože sensorové počítače nemají pro zpracování dostatečný vhodný výpočetní výkon. Taková zařízení spotřebují více energie na komunikaci než na výpočty v jednotlivých uzlech. Ve skutečnosti se tak sensorová síť stává novým druhem databázového stroje, jehož optimální využití požaduje, aby operace byly tlačeny co možná nejbliže k datům. Senzory mohou být instruovány dělat periodická měření a provádět předzpracování dat. Mezi jejich typické úkoly patří operace

- pošli data nebo kombinaci dat (průměr, součet apod.) do nějakého vybraného uzlu,
- ulož data (která mohou být zpracována později v čase),
- pošli zprávu jako reakci na nějakou pozorovanou událost.

Ve složitějším případě mohou být senzory a/nebo uživatelé jejich mobilní.

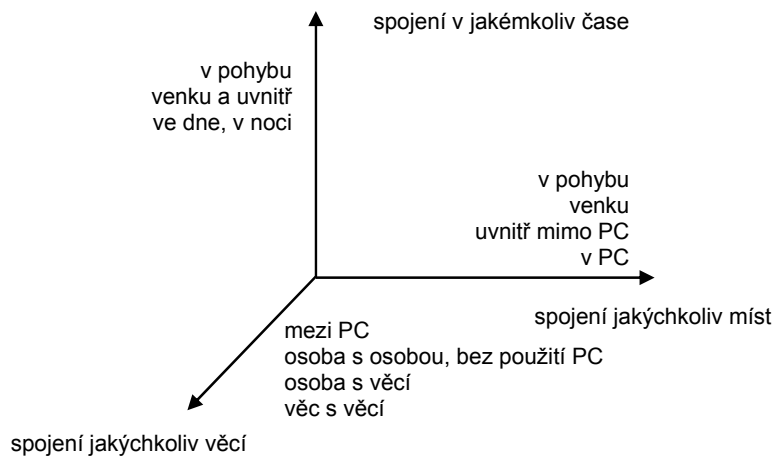
## 2.1 Od sensorových sítí k Internetu věcí

Obrázek 1 ukazuje příklad vícevrstvé architektury sensorové sítě. Senzory vyšší úrovně v sensorové síti mohou předzpracovat sensorová data nižší úrovně a vysílat pak získané odvozené informace do klientských zařízení. Aby byla úspěšná, mohou taková zpracování požadovat modifikované databázové techniky.



Obr. 1: Vícevrstvá architektura sensorové sítě

Senzorové sítě jsou na první pohled podobné distribuovaným databázím rozšířeným o vlastnosti související s využíváním reálného času. Důležitým rozdílem ale je, že míra vyhodnocování dat vytvořených v síti senzorů je vyšší, než se typicky uvažuje u distribuovaných SŘBD. To láme tradiční paradigma integrace informací, protože neexistuje žádný praktický způsob extrakce a natahování dat do společné databáze ke každému jejich výskytu. Rovněž musí být redefinovány strategie zpracování a optimalizace databázových dotazů.



Obr. 2: Nová dimenze v komunikaci (podle Nomura Research Institute)

## *Vybraný příspěvek*

Jako příklad rozsáhlejšího využití senzorových sítí je možné uvést výsledky výzkumu firmy Intel, zahájeném na jaře roku 2002, kdy je na Velkém kachním ostrově bez přítomnosti člověka sledováno klima a populace zde hnízdících ptáků. Dalším příkladem může být výzkum prováděný v US Army Medicine Research Center, kde je vyvíjen systém Warfighter Physiologic Status Monitoring, určený pro sledování zdravotního stavu vojáků v poli s využitím senzorů umístěných na a v lidském těle [3]. Existuje též vize Internetu věcí (Internet of Things [17]), který je založen na existenci velkého množství rozsáhlých heterogenních senzorových sítí. Již ne jenom spojení kdykoliv, kdekoliv, komukoliv, ale i cokoliv. „Cokoliv“ se stává novou dimenzí v komunikaci (obrázek 2).

### **2.2 Bezdrátové vysílání dat**

Použití senzorů se uplatňuje též v bezdrátovém vysílání dat. Sensory rozmístěné v prostředí mohou vysílat svá data periodicky nebo když nastane zajímavá událost. Na rozdíl od tradičního počítání nemohou klientská zařízení vytvářet požadavky na data od senzorů. Místo toho naslouchají klientská zařízení vysílacím kanálům pasivně. Sensory jsou tedy v komunikaci iniciativní. Sensory mohou vysílat data periodicky, jestliže měří nějaký nepřetržitý jev produkující data, nebo pouze když se vyskytne určitá událost.

Je jasné, že proud dat vytvářený senzory představuje nejen zajímavý zdroj aplikací, ale i výzkumný problém zejména z hlediska úlohy zpracování dat. Zpracování senzorových informací nastoluje nejzajímavější databázové problémy do nového prostředí, s novými omezeními a příležitostmi.

### **3 Zpracování proudů dat**

Správa dat přicházejících ze senzorů založená výlučně na tradičním modelu „ukládání a dotazování“ obvykle nemůže efektivně zacházet s objemem a rychlostí proudících dat, jejichž hodnoty mohou existovat jen na krátký okamžik. Tradiční SRBD jsou pro zacházení s proudy dat nevhodné z mnoha důvodů [2]:

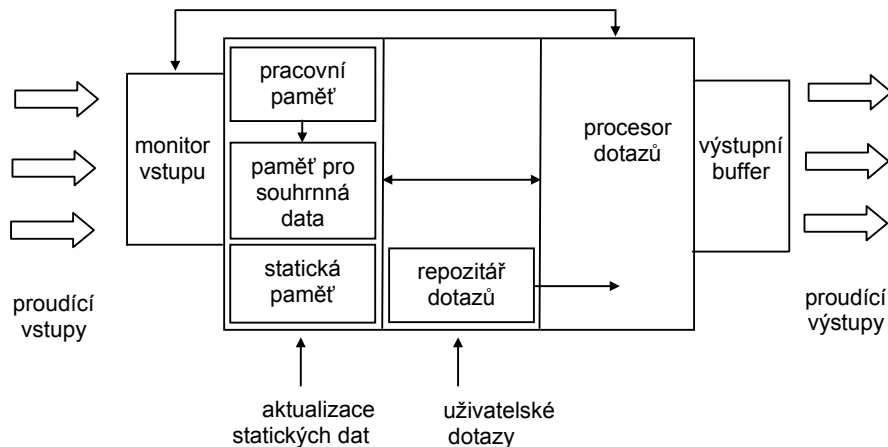
- senzory produkují a odesílají data nepřetržitě bez ohledu na to, existují-li přímé požadavky na tato data,
- dotazy nad příslušnými kolekcemi dat mohou být méně časté než vkládání dat do kolekcí,
- vytvořená data jsou často zpracovávána v reálném čase, protože mohou reprezentovat události, které potřebují okamžitou reakci,
- dotazy se zpracovávají nepřetržitě, protože proudy dat nikdy nekončí, takže mohou „vidět“, jak se mění podmínky systému během jejich volání,
- kvůli omezením na paměť, nemůže být proud dat uložen celý na vnější médium.

Naproti tomu z povahy proudu dat plyne, že dotazy je možné klást i průběžně [6]. Je také zřejmé, že vyhodnocení některých typů dotazů je jen aproximací skutečného výsledku dotazu – nemáme a nikdy nebudeme mít k dispozici všechna data. Dále je

zřejmé, že vyhodnocování dotazů musí být nutně adaptivní, na rozdíl od klasických databází, kde vyhodnocování probíhá podle předem připraveného, stabilního plánu.

### 3.1 Systémy řízení proudů dat

V důsledku těchto skutečností se objevily *systémy řízení proudů dat* (Stream Data Management System - SDMS), viz např. [10]. Příkladem budiž (work-flow) systém Aurora, společný projekt Brandeis University, Brown University, a M.I.T. [1]. *Processor (stroj) pro zpracování proudů dat* (Stream Processing Engine - SPE) je pak příkladem nové databázové architektury, která dovoluje volání dotazů, výpočtů a akcí na proudících datech v reálném čase. Processor by měl akceptovat proudově orientované dotazy zapsané v notaci SQL a spouštět je nad proudy událostí s výstupem v reálném čase. Zpracování je realizováno z větší části ve vnitřní paměti, operace čtení a zápisu na disk jsou volitelné a mohou být v mnoha případech zvládnuty asynchronně. Na logické úrovni jde o tři druhy paměti (viz obrázek 2): *pracovní* (pro dotazy typu okno), *paměť pro souhrnná data* o dotazech (sumarizace, synopse) a *statická paměť* pro metadata (např. fyzické umístění zdroje, rychlost proudu dat). Nepřetržitě či dlouhotrvající dotazy jsou registrovány v *rezpozitáři dotazů*, mohou mít zpracování sdílené ve skupině takových dotazů. Jednorázové dotazy, např. na stav proudu realizuje *procesor dotazů* přímo na vstupu. Tento procesor může také na základě zpráv z *monitoru vstupu* znovu optimalizovat plány nepřetržitých dotazů.



Obr. 2: Referenční architektura stroje pro zpracování proudů dat

Speciální implementace procesorů zpracovávající nové typy dat jsou dnes trendem ve vývoji nových databázových architektur [30]. Obvykle vedou k zvýšení výkonu. Např. v systému StreamBase vyvinutém Stonebrakerem [35] v r. 2005 je možné analyzovat 140000 zpráv za sekundu, zatímco běžný relační SRBD by zvládl v téže době pouze 900 zpráv. Důraz na zpracování ve vnitřní paměti pro některé dodavatele SDMS znamenal renesanci SRBD založených právě na tomto přístupu. Globální rozdíly mezi klasickými SRBD a SDMS ukazuje tabulka 1.

### Vybraný příspěvek

|                   | SDMS  | SŘBD                                       |
|-------------------|---|--|
| zdroje            | omezené (paměť, počítání po n-ticích)             | bohaté (paměť, disk, počítání po n-ticích) |
| zpracování dotazu | rozumně složité, blízké zpracování v reálném čase | velmi sofistikované, analýzy               |
| použití           | zjištění, jaká data ukládat do databáze SŘBD      | audit výsledků dotazu z SDMS               |

Tab.1. SDMS vs. SŘBD: globální rozdíly

Další variantou architektury je SDMS kombinovaný se SŘBD. V tomto případě jsou SDMS umístěny ve více bodech sítě v závislosti na topologii aplikace včetně klasického (relačního) SŘBD. Do SDMS vstupují objemné proudy dat a vycházejí z něho redukované. Výstup ze SDMS tvoří vstup do SŘBD pro další zpracování.

V současné době je možné pozorovat, že vývoj SDMS je řízen buď aplikacemi nebo technologiemi. V první případě je důležité dosáhnout sofistikovaného vyhodnocení dotazů resp. analýzy dat (bankovní podvody, počasí) ve skoro reálném čase. Druhý případ zahrnuje efektivní práci s masivními objemy dat z transakcí a měření (data generovaná ze senzorových sítí, satelitů). Pro tvorbu aplikací využívající proudy dat se objevují nové programovací jazyky, např. Hancock vyvinutý v AT&T [13].

### 3.2 Speciální architektury

Zpracování proudů dat může být realizováno v mobilním prostředí, kde proudící data jsou bezdrátově vysílána ke klientským zařízením. V budoucnosti se zřejmě uplatní technologie RFID (Radio-Frequency Identification). Připomeňme, že RFID neboli radiofrekvenční identifikace, je současná technologie založená na radiových vlnách pro přesnou identifikaci objektů pomocí značek EPC (Electronic Product Code). Jde o komunikaci vysílacího a přijímacího zařízení, jakési čtečky, s integrovaným obvodem vybaveným anténou (tagem). Událostí pak může být situace, když se nějaká RFID anténa dostane do dosahu senzoru. RFID technologie je použitelná např. pro kontrolu přístupu do objektu, vstupně-výstupní kontroly v knihovnách, evidence stopy pohybujícího se objektu (zboží, dokument, dítě ve škole), ve zdravotnictví (monitorování pacientů – e-health) apod.

RFID data jsou opět generována v proudech. Pro proudy RFID dat je typickou úlohou odhalování duplicit a chybných čtení obsažených v proudu [7]. RFID je klíčovou technologií pro realizaci Internetu věcí.

Speciálním případem diskutovaných procesorů jsou ty, od kterých se vyžaduje zpracování v reálném čase. Po vzoru dnes již klasických souborů pravidel pro relační databáze či OLAP od E.F. Coddů uvádějí autoři [34] 8 pravidel charakterizujících takové procesory:

1. Zpracovávat data proudů bez zpoždění, tj. bez požadavku na uložení dat.
2. Podpora neprocedurálního dotazovacího jazyka vyšší úrovně, jakéhosi „StreamSQL“.
3. Umět ošetřit situace poruch v proudu dat (např. chybějící prvky).

## Zpracování proudů dat

4. Garantovat predikovatelné a opakovatelné výsledky.
5. Umět efektivně ukládat, přistupovat a modifikovat informace o stavu zpracování, když je to nutné.
6. Vysoká dostupnost v případě chyb v systému (rychlé zotavení z chyb).
7. Možnost distribuce zpracování a tím škálovatelnosti systému. Distribuce by měla být automatická a transparentní.
8. Zajistit odezvu systému v reálném čase i pro proudy dat vysokých objemů.

### 4 Dotazování nad proudícími daty

Pro formalizaci pojmu proudů dat je vhodné začít posloupností. *Posloupnost* je uspořádaná množina prvků (n-tic, zpráv). Zdůrazněme, že posloupnost může obsahovat duplicitní n-tice. Využijeme-li pro posloupnosti relační model dat, pořadí n-tic v relacích není podstatné. *Proud dat* je posloupnost prvků, jejíž délka je neohraničená, přičemž prvky jsou uspořádány implicitně podle časového razítka odpovídajícího vstupu prvku do proudu. V rámci relačního modelu dat je možné vidět proudy dat jako neomezeně velké relace, speciálně uspořádané, definované nad relačními schématy.

Pro dotazování nad proudy dat je tedy podstatný pojem *modelu proudů dat*. V závislosti na něm lze SDMS vyzorovat dva typy zpracování:

- streams-in, stream-out (např. v systémech Aurora [1], Gigascope [14], STREAM [6]),
- stream-in, relation-out (v jazyku Hancock [13]),
- kombinace obojího.

Hlavní rozdíly mezi v modelování proudů dat a dotazování nad nimi v SDMS a klasických relačních SŘBD ukazuje tabulka 2.

| Modelování a dotazování | SDMS               | SŘBD                       |
|-------------------------|--------------------|----------------------------|
| model                   | transientní relace | persistentní relace        |
| relace                  | posloupnost n-tic  | množina/multimnožina n-tic |
| uspořádání n-tic        | podle vstupu       | neuspořádané               |
| aktualizace dat         | přidávání          | modifikace                 |
| dotaz                   | persistentní       | transientní                |
| odpověď na dotaz        | aproximovaná       | přesná                     |
| vyhodnocení dotazu      | jeden průchod      | libovolné                  |
| plán dotazu             | adaptivní          | pevný                      |

Tab.2. SDMS vs. SŘBD: modelování a dotazování

Dotazy nad proudy dat lze kategorizovat podle různých kritérií, např. *dotazy v daném čase* (one-time) využívající data proudů s daným společným časovým razítkem a *nepřetržitě (kontinuální) dotazy* tvořící zřejmě nejzajímavější třídu dotazů. Dotazy mohou být *předdefinované* a *ad hoc*. Ad hoc dotazy představují problém pro stavbu optimalizátoru SDMS.



#### 4.1 Operátory nad proudy dat

Díky požadavku na rozšíření SQL pro dotazování nad proudy dat, je zajímavým problémem, jak využít stávajících možností relačního modelu dat pro tento účel. Ukazuje se, že je možné využívat pouze tzv. neblokující operátory, jejichž charakteristikou je, že nevyžadují pro tvorbu výsledku znát celý vstup. To je dáno neohraničeností proudů dat. Z toho plyne, že operátory vedoucí k univerzální kvantifikaci (NOT EXISTS, EXCEPT, NOT IN apod.) nemohou být využity.

Podle [6] je *blokující operátor dotazu* takový operátor, který není schopen produkovat první  $n$ -tici výsledku aniž by „viděl“ celý vstup. Např. tradiční agregační funkce v SQL jsou blokující operátory. *Neblokující operátor dotazu* je definován jako operátor, které produkuje všechny  $n$ -tice výstupu před tím, než detekuje konec vstupu. Autoři [22] dokazují jednoduchou úvahou, že nějaká funkce  $F$  na konečných posloupnostech může být spočítána neblokujícím operátorem právě když  $F$  je monotónní vzhledem k částečnému uspořádání  $\sqsubseteq$  na posloupnostech, kde  $S_1 \sqsubseteq S_2$ , právě když  $S_1$  je počáteční podposloupností  $S_2$ . Protože proudy dat jsou obecně nekonečné posloupnosti, mohou být použity pro dotazy nad nimi pouze neblokující operátory. Dotazy tedy musí být monotónní. Připomeňme, že operátor  $F$  je *monotónní* jestliže pro jakékoli dvě posloupnosti  $S_1 \sqsubseteq S_2$

$$S_1 \sqsubseteq S_2 \Rightarrow f(S_1) \sqsubseteq f(S_2)$$

Monotónními operátory jsou např. relační selekce a projekce přenesené na proudy dat, ne však již tradiční agregační funkce. Na druhé straně nepřetržité COUNT a SUM jsou monotónní a tedy i neblokující.

Typy dotazů zahrnují upravené operace relační algebry a další typické operátory, včetně pokročilých modifikovaných metod dolování dat:

- selekce (obecněji konjunktivní dotaz nad jedním nebo více proudy dat)
- projekce,
- spojení (kartézský součin),
- agregační funkce (event. hnížděné),
- okno,
- nejčastější prvek,
- dolování nad proudy dat.

Plán dotazu obecně vyžaduje udržovat informace o bezprostředním stavu operátoru (běžně se nazývá *synopse* [4]).

Pojem *okna* je pro zpracování proudu dat důležitý a poskytuje řadu variant, např. okno pevné délky, které se posouvá v čase (*posuvné okno*), nebo má pohyblivý pouze jeden konec. Délka okna je měřena v čase (např. 5 minut) nebo v počtu příchozích prvků (každých dalších 100 prvků, posledních 100 prvků). Další možností je použití speciálních interpunkčních znamének. Okna představují jednu možnost pro aproximované odpovědi v dotazování nad proudy dat.

Typicky nemonotónním a tedy i neblokujícím operátorem je rozdíl dvou množin. *Rozdíl dvou proudů* dat má tedy také tuto vlastnost a není možné jej pro zpracování proudů dat využít.

*Selekce* (či *konjunktivní dotaz nad jedním proudem dat*) a *projekce* zřejmě nečiní žádné problémy, jde o monotónní neblokující operátory.

## Zpracování proudů dat

*Spojení* je monotónní, může mít blokující implementaci, např. spojení pomocí hnížděných cyklů, ale i neblokující, např. pomocí hašovaných spojení [29]. Ve druhém případě je ovšem nutné počítat s nároky na vnitřní paměť pro hašování prvků operandů spojení – dvou proudů dat v závislosti na doménách atributů, přes které se spojuje. Tedy i kartézský součin, jako speciální případ spojení, lze implementovat neblokujícím způsobem.

Jsou však operátory, které jsou definovány pomocí nemonotónních operátorů, nicméně k nim existuje neblokující implementace. Např. *průnik* definovaný pomocí rozdílu, je možné implementovat pomocí operace spojení.

*Konjunktivní dotaz nad více proudy dat* již vyžaduje hlubší analýzu pro zajištění jeho vyčísitelnosti v omezené paměti. Uvažujme proudy  $S(A, B, C)$  a  $T(C, D)$  s celočíselnými doménami pro všechny atributy a tři dotazy:

D1:  $S(A > 10)[A]$

D2:  $(S * T)(S.C > 10 \text{ AND } T.C < 20)[C]$

D3:  $(S[B < D]T)(A > 1 \text{ AND } A < 10)[A]$

Dotaz D1 lze zpracovat v konečné paměti pouze při zachování duplicit ve výsledku. U dotazu D2 je to možné též, dokonce s odstraněním duplicit. Stačí udržovat jistou informaci pro hodnoty atributu C z intervalu (10, 20). Pro dotaz D3 je ovšem nutné duplicit eliminovat. Pro každou hodnotu A z intervalu (1, 10) se udržují v synopsích stávající hodnoty  $\min(S.B)$  a  $\max(T.D)$  přes všechny dosud zpracované n-tice.

Blokující agregace, např. AVARAGE, lze „odblokovat“ např. tak, že uchováваме částečný součet a počet zpracovaných prvků. Samotné použití okna představuje také jisté odblokování vedoucí ovšem pouze k aproximovanému výsledku. Aproximace odpovědi se získávají zejména pomocí synopsí. Mezi metody pro vypočítávání synopsí patří zejména použití vzorků (samplování), histogramů a vlnek (wavelets).

Ve víceúrovňových sensorových sítích je možné modelovat proudy dat přímo jako *odvozené*, tj. předzpracované agregačními operacemi. Nad těmito proudy se pak odehrává vlastní dotazování.

V souvislosti s okny připomeňme konstrukt *window* specifikovaný ve standardu SQL:2003 OLAP funkcích. Pro proudy relačních dat lze tento konstrukt využít jako prostředek pro realizaci oken.

### 4.2 Optimalizace dotazů

Až dosud jsme modelovali jeden dotaz nad jedním nebo dvěma proudy dat. V reálném nasazení SDMS je třeba počítat s více dotazy na proudem dat, resp. s více dotazy nad více proudy dat. V souvislosti s velikostí oken dotazů pak může nastat situace, že se do pracovní paměti nevejde sjednocení potřebných zpracovávaných oken nebo dokonce ani jedno velké okno. Tato omezení vedou k tomu, že některé prvky musí být odstraněny nebo vzorkovány.

Z podstaty proudů dat a požadavků na nepřetržité či dlouhotrvající dotazy plynou požadavky na procesor dotazů v SDMS [18]:

- Model dat a sémantika dotazů musí dovolovat operátory založené na uspořádání a čase.

- Nemožnost uložit celý proud dat vede k použití aproximativních agregačních struktur (synopse). Dotazy využívající agregace nemohou tedy vrátit přesné odpovědi.
- Plány dotazu nad proudy dat nemohou používat blokující operátory, které pro vyprodukování výsledku požadují celý vstup.
- Z důvodu paměťových omezení není možné používat víceprůchodové algoritmy. On-line algoritmy nad proudy dat jsou omezeny pouze na jeden průchod daty.
- Aplikace, které monitorují proudy dat, musí v reálném čase rychle reagovat na neobvyklé hodnoty dat.
- Dlouhotrvající dotazy mohou v průběhu svého života narazit na změny (např. rychlosti proudu), což může vést ke změně plánu vyhodnocení dotazu.
- Pro sdílené volání více dlouhotrvajících dotazů je třeba zajistit škálovatelnost dotazovacího systému.

Speciální problémy nastoluje zpracování dotazů v bezdrátové senzorové síti. Vzhledem k šetření energie hodnotných zdrojů vysílaných dat není vhodné přenášet všechna data do centrálního místa a tam provádět zpracování dotazu. Část zpracování se proto odehrává již v uzlech sítě.

## **5 Proudly XML dat**

Proudly XML dat jsou neanalyzované XML dokumenty, tj. dokumenty chápané jako posloupnost znaků, které jsou generovány např. editorem webové stránky nebo se objevují jako předmět výměny dat na Internetu. Dobře vytvořený proud XML dat odpovídá dobře vytvořenému XML dokumentu, jehož model je obvyklý strom s uzly reprezentujícími v nejjednodušších případech elementy, atributy a textová data. Tento strom je samozřejmě virtuální, tj. není spolu s XML daty materializován. Představíme-li si prudící XML data, objevuje se přirozeně úloha vyhodnocení dotazu resp. množiny dotazů nad tímto proudem.

Dobrým východiskem pro řešení takové úlohy je fakt, že sekvenční průchod proudem XML dat odpovídá průchodu virtuálním XML stromem do hloubky způsobem preorder. Dále je třeba charakterizovat dotazovací jazyk, jehož výrazy lze vyhodnotit jednorůchodovým přístupem k proudu XML dat. Autoři [28] k tomuto účelu našli vhodnou podmnožinu jazyka XPath - jazyk SXP (Simple XPath). Připomeňme, že tzv. osy v XPath (binární relace mezi uzly) zahrnují jednak dopředné osy (děti, potomci, následníci, následující sourozenci), jednak zpětné osy (rodič, předchůdci, předci). Jazyk SXP obsahuje kromě aplikace os ještě testy uzlů a množinové operace.

V případě zpracování proudu XML dat nahrává jeden průchod virtuálním XML stromem dopředným osám a nepřipouští efektivní zpracování zpětných os (v limitním případě je nutné si pamatovat celý XML strom). Existuje však možnost jak transformovat dotazy v SXP se zpětnými osami do ekvivalentních dotazů s dopřednými osami a predikáty. V [28] je popsán vyhodnocovač dotazů SXP, který je jednorůchodový a pracuje pro dotazy bez predikátů v čase blízcím se zpracování uvedených dotazů ve vnitřní paměti. Jedině taková podmnožina SXP je vhodná pro

## Zpracování proudů dat

zpracování nekonečných proudů XML dat. Pro silnější jazyky je nutné použít aproximační techniky využívající omezenou vnitřní paměť.

V [38] je ukázáno, jak zpracovávat proud relačních dat a proudy XML dat unifikovaným způsobem. Tato aktivita vychází z požadavku aplikací zpracovávat současně proudy relačních i XML dat. Další příklady proudů XML dat lze najít v [20], kde je popsán systém TurboXPath.

## 6 Dolování v proudech dat

Proudy dat přinášejí nové výzvy i v oblasti dolování viz [16]. Inteligentní analýza dat prochází mnoha etapami. Každá etapa je motivována novými problémy, které se v praxi objeví. Statistická analýza dat představuje první etapu. Jejím cílem bylo studium dat na základě statistických hypotéz. Rostoucí výpočetní výkon umožnil rozvoj metod strojového učení. Začaly se řešit problémy výpočetní efektivity problémů analýzy dat. V průběhu rozvoje metod strojového učení se objevily nové problémy analýzy dat. Rostoucí velikost databází podnítila vývoj škálovatelných algoritmů. Metody statistické analýzy dat a strojového učení byly modifikovány pro použití nad rozsáhlými databázemi. Dolování dat je interdisciplinární metoda, jenž se zabývá extrakcí modelů a vzorů z rozsáhlých dat viz [19].

Rozvoj paralelních výpočtů a sítí podnítil rozvoj paralelních a distribuovaných metod dolování dat. Nyní je hlavním problémem extrakce znalostí z heterogenních dat a jejich následná integrace. Generování dat se stává stále rychlejší než bylo dříve. Generování spojených proudů dat se tak stalo novou výzvou pro výzkum [25].

Problémy spojené s dolováním v proudech dat můžeme rozdělit do následujících oblastí: řešení *založená na datech* a *úkolově orientovaná*. Datové řešení je založeno na zpracování podmnožiny dat nebo na transformaci dat vertikální nebo horizontální a aproximaci menší datové reprezentace. Úkolově orientované řešení je založeno na technikách výpočetní teorie tak, abychom dosáhli lepších časových a prostorových parametrů použitých algoritmů.

### 6.1 Techniky založené na datech

Datové techniky využívají sumarizace dat nebo vybírají podmnožinu z daného proudu pro další analýzu. Mezi první techniky patřily vzorkování (sampling), skicování (sketching) a vypouštění (load shedding). Synopse a agregace se objevují až později. Jejich podstata je uvedena v odstavci 4.1.

- *Vzorkování*. Vzorkování vybírá datové položky, jenž budou dále zpracovány na základě nějakého pravděpodobnostního modelu. Vzorkování patří k velmi starým statistickým metodám. Hlavní problémem použití vzorkování v kontextu analýzy proudů dat je to, že předem neznáme velikost dat.
- *Vypouštění*. Je založeno na vypuštění částí toku dat [5]. Je to metoda používaná pro dotazování. Má však stejné nedostatky jako vzorkování. Použití této metody je pro dolování dat problematické. Vypouštění většího množství dat může vést ke ztrátě informace o struktuře dat.
- *Skicování*. Skicování [25] je metoda založená na náhodném promítání podmnožiny vlastností. Je to proces vertikálního vzorkování datového

proudu. Skicování můžeme použít pro porovnání datových proudů a pro agregační dotazy. Hlavní nevýhodou skicování je nepřesnost. Metoda, jenž je velmi podobná, ale daleko přesnější je metoda hlavních komponent (PCA) nebo postupy založené na redukci dimenze jako SVD, a NMF.

## 6.2 Úkolově orientované techniky

- *Aproximační algoritmy.* Vychází z předpokladu, že algoritmy dolování dat jsou samy o sobě výpočetně složité a proto je nutné uvažovat aproximace.
- *Posuvná okna.* Vychází z předpokladu, že uživatelé zajímají nejnovější data v proudu. Technika oken je popsána opět v odstavci 4.1.

## 6.3 Techniky dolování

Pro vlastní dolování v proudu dat se modifikují známé techniky jako jsou shlukování, frekvenční metody, časové řady, rozhodovací stromy. Důležitým požadavkem je omezit počet průchoďů nejlépe na jeden či využívat např. dvě fáze, kdy první omezuje významně velikost proudu a druhá takový výsledek zpracovává.

## 7 Aplikace

Některé další systémy pro zpracování proudů dat:

- COUGAR [37]: senzorová database, jejímž výstupem jsou časové řady. <http://www.cs.cornell.edu/database/cougar>. (25.6.2006)
- Gigascope [14]: systém monitorující distribuované síť. <http://www.research.att.com/viewProject.cfm?projID=129> (25.6.2006)
- NiagaraCQ [11] je systém pro vyhledávání v XML dokumentech pomocí XML-QL dotazů. <http://www.cs.wisc.edu/niagara>. (25.6.2006)
- WebCQ [23]: systém monitorující změny na Internetu. <http://disl.cc.gatech.edu/CQ> (25.6.2006)
- StatStream [39]: systém věnovaný statistické analýze časových řad <http://cs.nyu.edu/cs/faculty/shasha/papers/statstream.html>. (25.6.2006)
- STREAM [6]: projekt standfordské university věnovaný problematice proudových dat. <http://www-db.stanford.edu/stream>. (25.6.2006)
- Stream Mill [38]: skupina zabývající se spojitými dotazy <http://wis.cs.ucla.edu/stream-mill/index.html> (25.6.2006)
- TelegraphCQ [21]: projekt zaměřený na adaptivní vyhodnocování dotazů a na adaptivní workflow systémy <http://telegraph.cs.berkeley.edu>. (25.6.2006)

## 8 Závěr

Z hlediska srovnání zpracování proudů dat a databází je zajímavé pozorovat, že to, co je typické pro SDMS, se v jednotlivých bodech (jistě v menší míře) řešilo v 90. letech v kontextu databází. Šlo o snahu rozšířit klasický SRBD o možnosti nových

## Zpracování proudů dat

aplikací. Největší změny databázového zpracování v případě proudů dat se ovšem týkají zpracování dotazů, v důsledku čehož se stává využití tradičního SŘBD nemožné. Zpracování proudů dat na druhé straně těží i z jiných vlivů, částečně např. ze zkušeností (zejména negativních) získaných v oblasti zpracování časových řad a posloupností databázovým způsobem.

V budoucnu lze očekávat rychlý rozvoj problematiky zpracování proudů dat, zvláště pak v sítích typu Internet věcí, kdy inteligence vnořená do věcí bude distribuovat zpracování dat v síti více než při pouhém propojování počítačů. Tento rozvoj bude akcelеровán rozvojem senzorových sítí [32] a uplatněním nanotechnologií [12]. Nanotechnologie umožní další zlevnění senzorů a tím otevrou spolu s miniaturizací cestu k jejich ještě většímu nasazení. Spolu s RFID a vnořením inteligence do předmětů denního života přispějí tyto technologie k realizaci aplikací zmiňovaných v úvodu a vize Internetu věcí.

### Poděkování

Práce byla částečně podporována Národním programem výzkumu – v rámci projektu Informační společnost IET100300419.

### Literatura

1. Abadi, D. J., Carney, D., Cetintemel, U., Cherniack, M., Convey, C., Lee, S., Stonebraker, M., Tatbul, N., Zdonik, S.: Aurora: a new model and architecture for data stream management. *The VLDB Journal* (2003) 12: 120–139.
2. Amato, G., Caruso, A., Chessa, S., Masi, V., Urpi, A.: State of the Art and Future Directions in Wireless Sensor Network's Data Management. Technical Report 004-TR-16, ISTI, 2004.
3. Army Medical Surveillance Activity. Dostupné na: <http://amsa.army.mil/>. (25.6.2006)
4. Babcock, B., Babu, S., Datar, M., Motwani, R., Widom, J.: Models and Issues in Data Stream Systems. In: *Proc. of the ACM SIGACT SIGMOD SIGART Symposium on Principles of Database Systems*, 2002, Vol. 21, 1-16.
5. Babcock, B., Datar, M., Motwani, R.: Load Shedding Techniques for Data Stream Systems. In *Proc. of the 2003 Workshop on Management and Processing of Data Streams*, June 2003.
6. Babu, S., Widom, J.: Continuous queries over data streams. *SIGMOD Record*, 30(3):109–120, Sept. 2001.
7. Bai, Y., Wang, F., Liu, P.: Efficiently Filtering RFID Data Streams. In: *Proc. of CleanDB, Seoul, Korea*, 2006.
8. Bonnet, P., Gehrke, J., Seshadri, P.: Towards Sensor Database Systems. *LNCS*, Vol. 1987, Jan 2001, p. 3.
9. Bry, F., Coskun, F., Durmaz, S., Furche, T., Olteanu, D., Spannagel, M.: The XML Stream Query Processor SPEX. In: *Proc. of 21st International Conference on Data Engineering (ICDE'2005)*, Tokyo, Japan, 2005.

*Vybraný příspěvek*

10. Carney, D., Cetintemel, U., Cherniack, M., Convey, C., Lee, S., Seidman, G., Stonebraker, M., Tatbul, N., Zdonik, S.: Monitoring streams – a new class of DBMS applications. Technical Report CS-02-01, Department of Computer Science, Brown University, Feb. 2002.
11. Chen, J., DeWitt, D., Tian, F., Wang, Y.: NiagaraCQ: A Scalable Continuous Query System for Internet Databases. In: *Proc. ACM Int. Conf. on Management of Data*, 2000, 379-390.
12. Cooper, J. M., Johannessen, E. A., Cumming, D. R. S.: Bridging the Gap Between Micro and Nanotechnology: Using Lab-on-a-Chip to Enable Nanosensors for Genomics, Proteomics, and Diagnostic Screening. *LNCS*, Vol. 3222, Jan 2004, 517 – 521.
13. Cortes, C., Fisher, K., Pregibon, D., Rogers, A., Smith, F.: Hancock: a language for extracting signatures from data streams. *KDD 2000*: 9-17.
14. Cranor, C., Gao, Y., Johnson, T., Shkapenyuk, V., Spatscheck, O.: GigaScope: High Performance Network Monitoring with an SQL Interface. In: *Proc. ACM Int. Conf. on Management of Data*, 2002, p. 623.
15. Drineas, P., Frieze, A., Kannan, R., Vempala, S., Vinay, V.: Clustering Large Graphs via the Singular Value Decomposition. Kluwer Academic Publisher, *Machine Learning*, 56, 2004, 9–33.
16. Gaber, M. M., Zaslavsky A. B., Krishnaswamy, S.: Mining data streams: a review. *SIGMOD Record* 34(2): 18-26, 2005.
17. Geshenfeld, N., Krikoran, R., Cohen, D.: The Internet of Things. *Scientific American Magazine*. October 2004.
18. Golab, L., Özsu, N. T.: Data stream Management Issues – A Survey. University of Waterloo, TR CS-2003-08, April 2003.
19. Hand, D.J., Mannila, H., Smyth, P.: *Principles of data mining*. MIT Press. 2001.
20. Josifovski, V., Fontoura, M., Barta, A.: Querying XML streams. *The VLDB Journal* (2005) 14: 197–210.
21. Krishnamurthy, S., Chandrasekaran, S., Cooper, O., Deshpande, A., Franklin, M. J., Hellerstein, J. M., Hong, W., Madden, S. R., Raman, V., Reiss, F., Shah, M.A.: TelegraphCQ: An Architectural Status Report. *IEEE Data Engineering Bulletin*, Vol 26(1):11-18, March 2003.
22. Law, Y.-N., Wang, H., Zaniolo, C.: Query Languages and Data Models for Database Sequences and Data Streams. In: *Proc. of the 30th VLDB Conf.*, Toronto, Canada, 2004.
23. Liu, L., Pu, C., Tang, W.: Continual Queries for Internet-Scale Event-Driven Information Delivery. *IEEE Trans. Knowledge and Data Eng.*, 11(4): 610-628, 1999.
24. Madden S., Franklin, M. J.: Fjording the stream: An architecture for queries over streaming sensor data. In: *Proc. of the 2002 Intl. Conf. on Data Engineering*, Feb. 2002.
25. Muthukrishnan, S.: Data streams: algorithms and applications. In *Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms*. 2003.

## Zpracování proudů dat

26. iPolicy Networks home page. Dostupné na: <http://www.ipolicynetworks.com> (25.6.2006)
27. Iyengar S. S., Brooks, R. R. (eds.): *Distributed Sensor Networks*. Chapman & Hall/CRC, 2004.
28. Oltaneu, D., Furche, T., Bry, F.: An Efficient Single-Pass Query Evaluator for XML Data Streams. In: *Proc. of ACM Symposium on Applied Computing'04*, March 14-17, Nicosia, Cyprus, 2004, 627-631.
29. Pokorný, J.: *Dotazovací jazyky*. Skripta MFF UK, Univerzita Karlova, Nakladatelství Karolinum, 2002.
30. Pokorný, J.: Databázové architektury: současné trendy a jejich vztah k novým požadavkům praxe. In: *Sborník konference Moderní databáze*, Legner Hotel Zvánovice, Jevany, KOMIX, 5-14.
31. *Proceedings of the First Workshop on Data Management for Sensor Networks* (DMSN 2004), Toronto, Canada, August 30th, 2004. Dostupné na: <http://db.cs.pitt.edu/dmsn04/> (25.6.2006)
32. On-line Magazine 'Sensors & Transducers' (S&T e-Digest). Dostupné na: <http://www.sensorsportal.com/> (25.6.2006)
33. Stanford Stream Data Management (STREAM) Project. Dostupné na: <http://www.db.stanford.edu/stream>. (25.6.2006)
34. Stonebraker, M., Çetintemel, U., Zdonik, S.: The 8 Requirements of Real-Time Stream Processing. *SIGMOD Record*, 34(4):42-47, Dec. 2005.
35. StreamBase Systems, Inc.: StreamBase™ 2.0, 2005. Dostupné na: <http://www.streambase.com/index.html> (25.6.2006)
36. Traderbot home page. <http://www.traderbot.com>. (25.6.2006)
37. Yao, Y., Gehrke, J.: Query Processing for Sensor Networks. In *Proc. Conf. on Innovative Data Systems Research (CIDR)*, 2003, 233-244.
38. Zhou, X., Thakkar, H., Zaniolo, C.: Unifying the Processing of XML Streams and Relational Data Streams. In: *Proc. of ICDE*, 2006, p. 50.
39. Zhu, Y., Shasha, D.: StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time. In: *Proc. of the 28<sup>th</sup> VLDB Conf.*, 2002, 358-369.

### Annotation:

Data streams appear in new application areas such as web applications, financial data, environmental data, safety data, etc. Their processing is usually continual both during generating data and in querying the data. Processes arising in such activities cause an emergence of events and triggering other processes in monitoring systems. The sources of streams data are often sensors and sensor networks. For the database approach, data stream processing brings new challenges. Streams data can not be stored on external memory, queries can be unlimited, approximate methods and new approaches to query optimization are useful. The goal of paper is to introduce into these new directions, to summarize results achieved as well as to formulate problems that are challenging for computer science, particularly for the database area.