Learning from Data Streams

Einführung

Christian Zimmer

Definition Data Streams

- Allgemeine Definition

  "… sequence of digitally encoded signals used to represent information in transmission"

  http://www.its.bldrdoc.gov/projects/devglossary/data_stream.html

Puzzle – ein einfaches Beispiel

- „Finde die fehlende Zahl!“
  - Paul: Zeigt Zahlen einer Menge *n,* lässt ein Element aus
  - Carole: Muss ausgelassenes Element bestimmen
    - trivial für kleines *n* (Gedächtnis)
    - Für großes *n* (beschränktes Gedächtnis) nicht praktikabel
- Lösungsalgorithmus:

$$s = \frac{n(n+1)}{2} - \sum_{j \le i} \pi_{-1}[j]$$

$$\pi : Permutation\ von\ \{1, ..., n\}$$

$$\pi_{-1} : \pi, wobei\ 1\ Element\ \ fehlt$$

Puzzle – ein einfaches Beispiel (2)

Paul

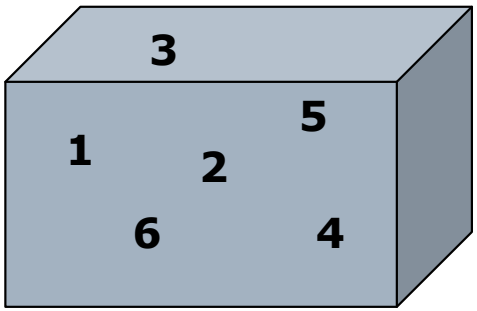
$\pi_{-1}[1] = 2$

$\pi_{-1}[2] = 1$

$\pi_{-1}[3] = 4$

$\pi_{-1}[4] = 6$

$\pi_{-1}[5] = 3$



Carole

$$s = \frac{n(n+1)}{2} = \frac{6 \cdot 7}{2} = 21$$

$$\sum_{j \le i} \pi_{-1}[j]$$

! **?**

2
3
7
13
16

$21 - 16 = \mathbf{5}$

Data Streams

- Im Puzzle: Lösungsalgorithmus ist deterministisch und exakt, *O(log n)* bits

  ➔ Untypisch für Data Streaming

- Charakteristika für Data Stream Algorithmen

  - Nicht deterministisch
  - Approximierte Lösung
  - Hohe Güte nur bei großer Anzahl von Elementen

Inhalt

- Definition Data Streams
- Einleitung
- Data Stream Phänomen
- Data Streaming: Formale Aspekte
- Grundlagen
- Streaming Systeme
- Zusammenfassung
- Diskussion

Data Stream Phänomen

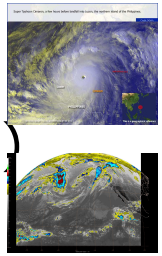
- Erweiterung der Definition
  - "… sequence of digitally encoded signals used to represent information in transmission"
  - "…input data that comes at a very high rate"

➔kommunikations- und rechenintensiv

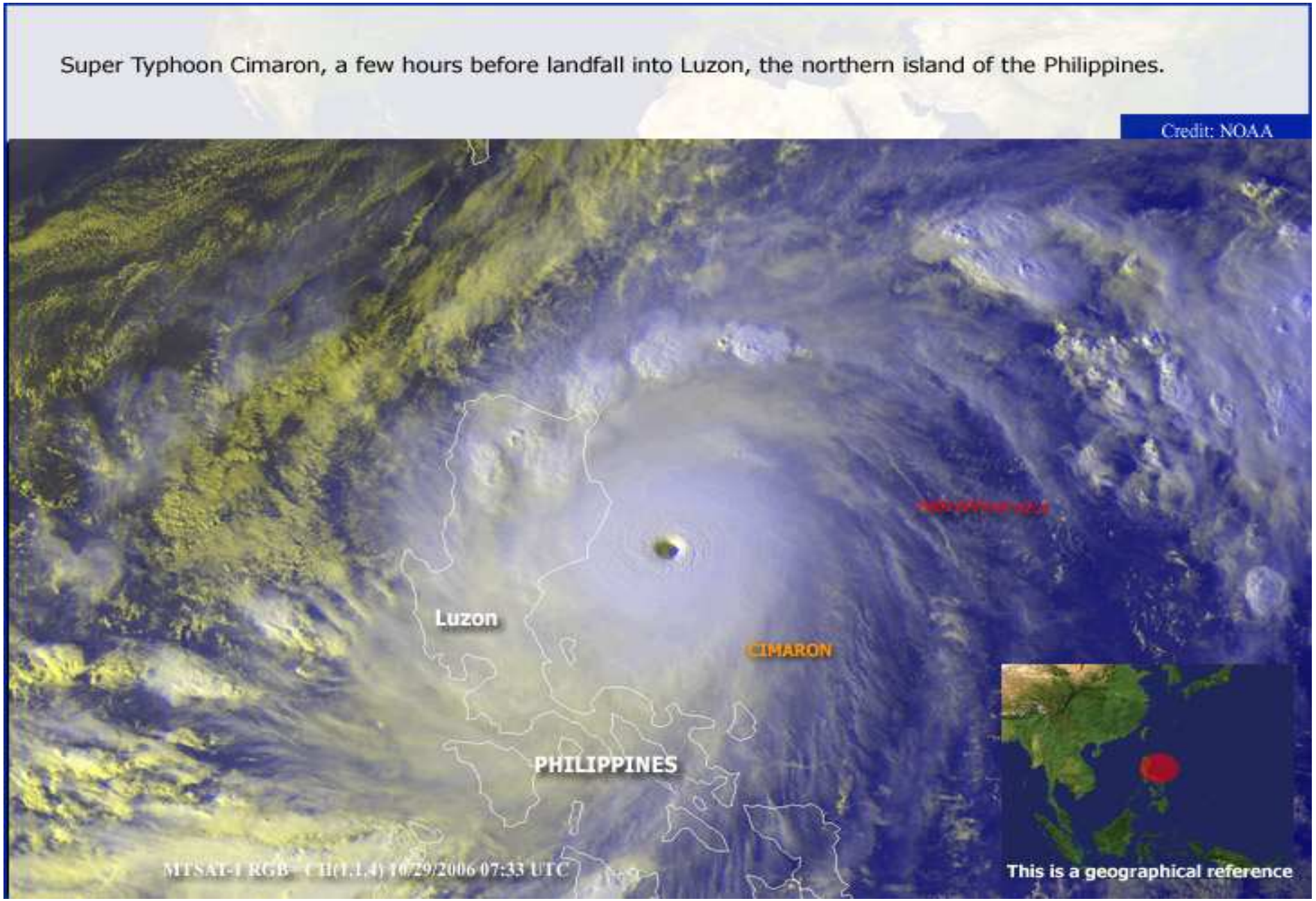
- **T** transmit
- **C** compute
- **S** store

Data Stream Phänomen (2)

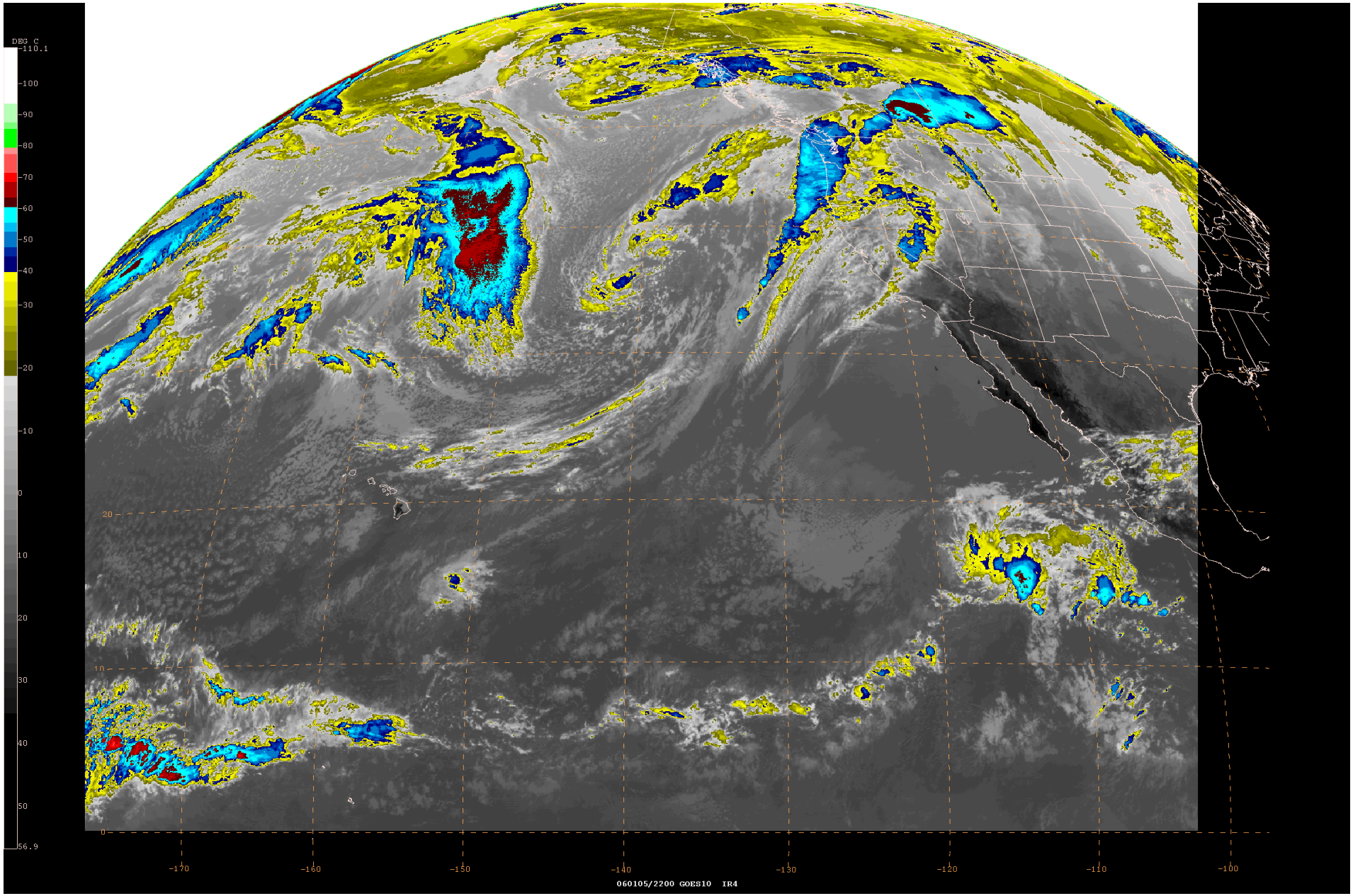
- Herausforderungen an die TCS Infrastruktur
  - Hoch-detaillierte Dateneingaben einschließlich fortlaufender Updates
    - Hochauflösende Abbildung der Erdoberfläche (Geodäsie)
    - Wetterdaten
    - Internet: Clicks, Queries, IP Traffic Logs



  - Differenzierte Analyse von Update Streams
    - Monitoring Anwendungen
      - atmosphärisch, astronomisch, erzeugt durch Netzwerke oder Sensoren
      - Erkennen von Ausreißern, Sonderfällen; Betrug, Eingriffe
      - zeitkritisch: Schritt halten mit der Update Rate



National Oceanic & Athmospheric Administration
Image of the day, 30th November 2006, http://www.noaa.gov/



Unidata – Enhanced Earth-system education and research

Real-time Remote Sensing Image Data , http://www.unidata.ucar.edu/data/

Data Stream Phänomen (3)

- Klassischer Blickwinkel der Informatik
  - Effizientes Manipulieren speichergebundener Daten
  - Verwalten von Datenbanken in Größenordnung von PB's
    → BaBar (Objectivity/DB) , Nov. 2004: 895 TB
  - Erzeugen synthetischer Datenbanken: Design Vergleich
- Wie den Herausforderungen begegnen?
  - Parallelisierung (CS!, T?)
  - Datenrate kontrollieren (Sampling vs. Shedding Updates)
    → CERN: 40TB/s reduziert auf 800GB/s (Quantenphysik)
  - Hierarchisch gegliederte Analyse (vgl. Speicherhierarchie)
  - "Kreativität"
- Logische Schlussfolgerung
  - Sammeln gewaltiger Datenmengen ist möglich
  - Aber: Verwerfen von Informationen aufgrund zeitkritischer Scans

Data Streaming: formale Aspekte

- Modelle
  - Sequentieller Datenstrom, elementweise
  - Beschrieben wird ein zu Grunde liegendes Signal *A*

$$a_1, a_2, \ldots$$

$$A : [1 \ldots N] \rightarrow R$$

- 'Time Series Modell': Signal entspricht dem aktuell gesehenen Element (**=**)

$$A[i] = a_i$$

- 'Cash Register Modell': gesehenes Element aktualisiert Signal (**+**)

$$a_i = (j, I_i), I_i \geq 0$$

$$A_i[j] = A_{i-1}[j] + I_i$$

- 'Turnstile Modell': gesehenes Element aktualisiert Signal (**+/-**)

$$a_i = (j, U_i), U_i \; positiv \; oder \; negativ$$

$$A_i[j] = A_{i-1}[j] + U_i$$

  - Strict turnstile

$$A_i[j] \geq 0 \; \forall i$$

Data Streaming: formale Aspekte (2)

- Grad der Allgemeinheit
  - theoretisch: Design gemäß des 'Turnstile' Modells wünschenswert
  - praktisch: schwächere Modelle können geeigneter, einfacher zu realisieren sein
- Berechnungen auf dem Signal **A**
  - Pro Element Bearbeitungszeit (processing time)
  - Platzbedarf (storage)
  - Berechnungszeit von Funktionen (compute time)
- Berechnungen sollten mit logarithmischem Aufwand durchführbar sein
  - Vgl. mit Baumstrukturen
  - Compute time kann höhere Komplexität

Data Streaming: formale Aspekte (3)

- Motivierendes (häufiges) Szenario: Traffic Data beim IP Packet Forwarding am Router
  - Verschiedene Ebenen vorstellbar
    - ***Packet log:*** Header mit Ausgangs- und Ziel IP, Ports, …
    - ***Flow log:*** Sammlung von Paketen mit gleichen Werten für Schlüsselattribute, kumulierte Informationen
    - ***SNMP log*** *(application layer)****:*** aggregierte Daten
  - Flow und Packet logs sind wesentlich voluminöser als SNMP log ➔ Data Streaming Argumente greifen hier
  - Mögliche Abfragen
    - HTTP Traffic über einen bestimmten Link initiiert durch einem gewissen IP Bereich?
    - Unterschiedliche IP Adressen, die gewissen Link zum Senden von Daten benutzen?
    - Top k der am meisten beanspruchten Flows?

Data Streaming: formale Aspekte (4)

- Im Detail: Wie viele verschiedene IP Adressen…
  - …haben einen bestimmten Link verwendet um Daten zu senden (seit Begin des Tages)?

$$input\ stream\ a_1, a_2, \ldots\ sequence\ of\ IP\ packets$$
$$a_i\ has\ source\ IP\ address\ s_i$$
$$A[0 \ldots N-1]\ number\ of\ packets\ sent\ by\ s_i, 0 \le i \le N-1$$
$$\text{Solution}: calculate\ nonzero\ A[i]'s$$

Cash Register

  - …nutzen zur Zeit einen gegebenen Link (d.h. sind momentan Bestandteil eines Flows über diesen Link)?

$$time\ t, source\ IP\ address\ s_i,\ flow\ f_i\ began \prec t, ends \succ t$$
$$A[0 \ldots N-1]\ number\ of\ flows\ s_i\ is\ currently\ involved\ in$$
$$set\ i = 0\ \forall\ 0 \le i \le N-1$$

$$if\ a_i\ is\ beginning\ of\ flow \rightarrow +1\ for\ A[s_j]\ if\ s_j\ is\ source\ of\ a_i$$
$$if\ a_i\ is\ end\ of\ flow \rightarrow -1\ for\ A[s_j]\ if\ s_j\ is\ source\ of\ a_i$$
$$\text{Solution}: calculate\ nonzero\ A[i]'s$$

Turnstile

Data Streaming: formale Aspekte (5)

- Weitere Anwendungen
  - "One-pass", sequentielle E/A
    - Data Streams entstehen als Produkt beim Arbeiten mit großen Datenmengen
    - Einmalige (oder einige wenige) Abtastungen werden dabei bevorzugt (da 'kostspielig')
    - Daten entstehen inkrementell als Serie von Updates (z.B. disk, bus, tape transfers)
  - Monitoring von Datenbank Inhalten
    - Große Datenbanken mit regelmäßigen Transaktionen: Einfügen/Löschen/Abfragen
    - *Selectivity Estimation:* Abschätzen des Zeitbedarfs einfacher Abfragen um komplexe Abfragen effektiv zu gestalten
    - Data Stream Szenario (Turnstile): Einfügen und Löschen sind Updates, Signal wird durch Datenbank repräsentiert

Grundlagen

- Mathematische Konzepte
  - Sampling
    - Auswahl von Elementen aus dem Data Stream anhand festgelegter Kriterien
    - Domain sampling, universe sampling, distinct sampling, etc.
    - z.B. bestimme Anzahl verschiedener Elemente, finde häufige Elemente, …

    15. November — *Approximate frequency counts over data streams*; *Mining Top-K Frequent Itemsets from Data Streams*

  - Random Projections
    - Reduktion der Dimension durch Projektion anhand zufälliger Vektoren
    - Ansatz lässt sich auf Turnstile Modell anwenden (*sketches, synopses*)

Grundlagen (2)

- Grundlegende Algorithmen
  - Binäre Suche, Greedy-Algorithmen bzw. Dynamische Programmierung, " Teile-und-Herrsche" Ansätze
  - Group Testing
    - Beispiel: Durch $I \leq x?$ gesuchte Zahl erfragen
    - Bestimme *B* häufigste Elemente in Turnstile Data Streams
  - Tree Method
    - Datenstrom wird durch (balancierten) Baum repräsentiert
    - Update: Aufdecken der Blätter des Baums (z.B. Histogramme)

    22. November *Learning Decision Trees from Data Streams*
  - Exponential Histograms
    - Zerlegung einer Struktur in Regionen mit Abstand 2^i
    - z.B. nearest neighbour Probleme, Sliding Windows

    08. November *Aggregating Statistics: Stream Statistics over Sliding Windows*

Streaming Systeme

- Hands-on Systeme
  - Streams mit Unterstützung des BS und Standard Programmiersprachen aufzeichnen
  - Beispiele: AT&T Research – Call Detail Records
- Leistungsfähige Datenbank verarbeitet Updates
  - Standard Technologien (bulk loading, fast transaction support)
  - Applikationen setzen auf die Datenbank auf (IPSOFACTO: ***IP S****tream-****O****riented* ***Fa****st* ***C****orrelation* ***To****ol:* SNMP log updates)
- Datenbanksysteme, die auf die Verarbeitung von Data Streams ausgelegt sind
  - aktives Forschungsgebiet (new stream operators, sql extensions, scheduling methods)
  - Beispiele: Aurora, Telegraph, **Stanford Stream**

Zusammenfassung

- Was sind Data Streams?
  - Welche Eigenschaften haben sie?
  - In welchem Zusammenhang enstehen sie?
- Welche Herausforderungen bieten Data Streams?
- Welche Modelle werden unterschieden?
  - Was sind die Annahmen der Modelle?
  - Wo finden sich die Modelle in der Praxis wieder?
- Welchen mathematischen Konzepte und Algorithmen finden Anwendung?
- Welche Streaming Systeme werden unterschieden?

Literaturliste

- S. Muthukrishnan (2003) Data streams: Algorithms and Applications. Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms.
- Flip Korn, S. Muthukrishnan and Yunyue Zhu (2003) IPSOFACTO: A Visual Correlation Tool for Aggregate Network Traffic Data. SIGMOD 2003, June 9-12.
- A. Arasu, B. Babcock, J. Cieslewicz (2004) STREAM: The Stanford Data Stream Management System. Book chapter.
- J. Gray, P. Sundaresan, S. Englert, K. Baclawski, P. Weinberger (1994) Quickly Generating Billion-Record Synthetic Databases. Proc. ACM SIGMOD Conf. Minneapolis.
- M. Datar, A. Gionis, P. Indyk, and R. Motwani (2002) Maintaining Stream Statistics Over Sliding Windows. In SIAM Journal on Computing, Vol. 31 No. 6.
- S. Guha, K. Mungala, K. Shankar and S. Venkatasubramanian (2003) Application of the two-sided depth test to CSG rendering. I3d, ACM Interactive 3D graphics.

Literaturliste (2)

- Sudipto Guha and Nick Koudas (2001) Data-Streams and Histograms. In Proc. STOC
- Y. Chen, G. Dong, J. Han, B. W. Wah, and J. Wang (2002) Multidimensional regression analysis of time-series data streams. In VLDB Conference.
- G. Wrochna Soltan (1996) Control Data Rate by Sampling or Shedding. Institute for Nuclear Studies, Warsaw.
- N. Koudas, D. Srivastava (2003) Data Stream Query Processing: A Tutorial. Proceedings of the 29th VLDB Conference, Berlin.
- Lukasz Golab and M. Tamer Ozsu (2003) Issues in Data Stream Management. In SIGMOD Record, Volume 32, Number 2, pp. 5--14.
- M. Garofalakis, Johannes Gehrke, Rajeev Rastogi (2002) Querying and mining data streams: you only get one look a tutorial. SIGMOD Conference 635.
- B. Babcock, S.Babu, M.Datar, R.Motwaniand J.Widom (2002) Models and issues in data stream systems. ACMPODS, 1–16.

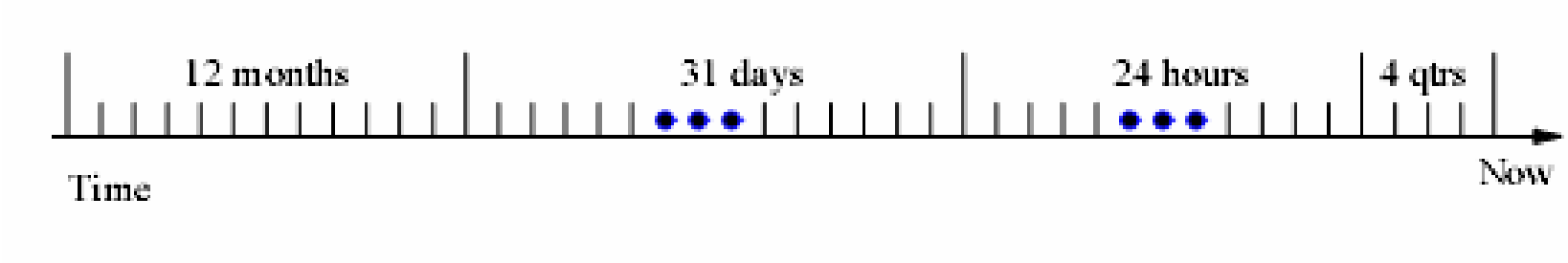
Diskussion

- Vielen Dank für die Aufmerksamkeit

ChristianZimmer@freenet.de

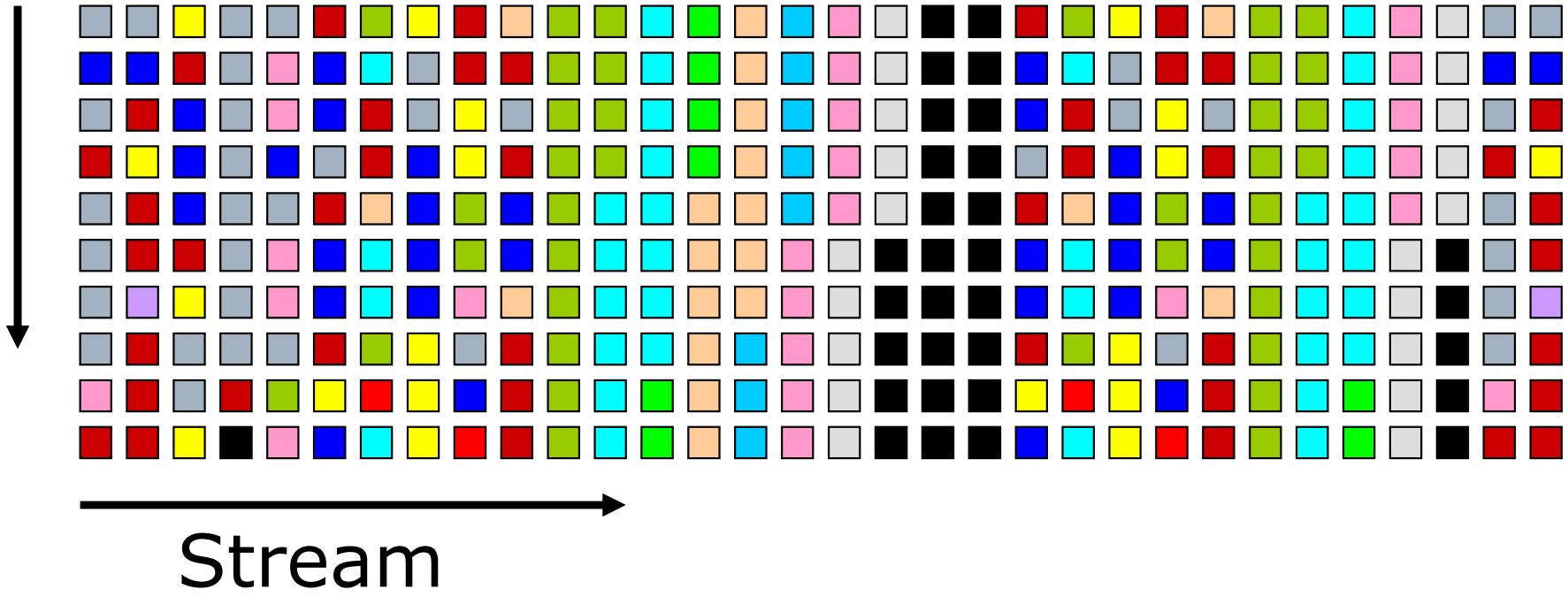
Zusatz

- Der Mensch als “Data Stream Verarbeitungsmaschine“
- Streaming Models (Window Streaming, Permutation Streaming)



Quelle: *Randomization for Massive and Streaming Data Sets, http://theory.stanford.edu/~rajeev/forum-2003-talk.ppt*

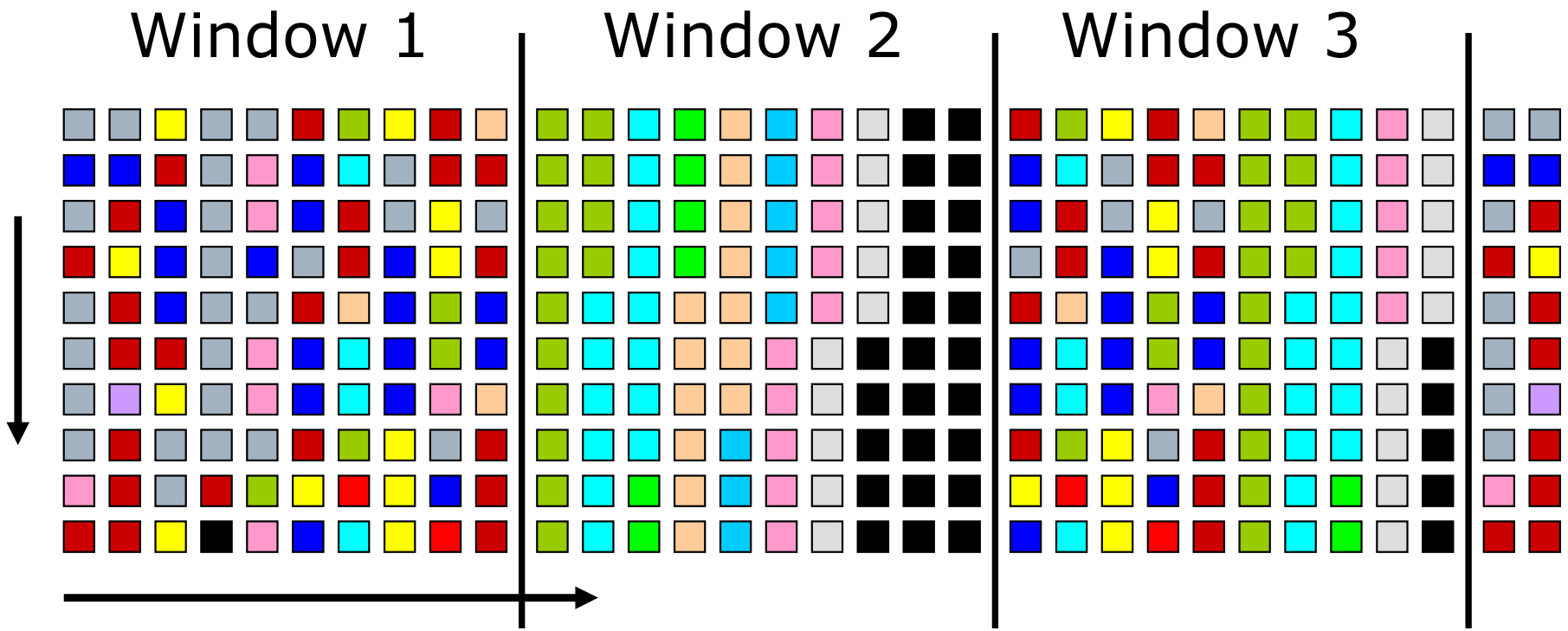
Window Streaming



Quelle: *Randomization for Massive and Streaming Data Sets, http://theory.stanford.edu/~rajeev/forum-2003-talk.ppt*

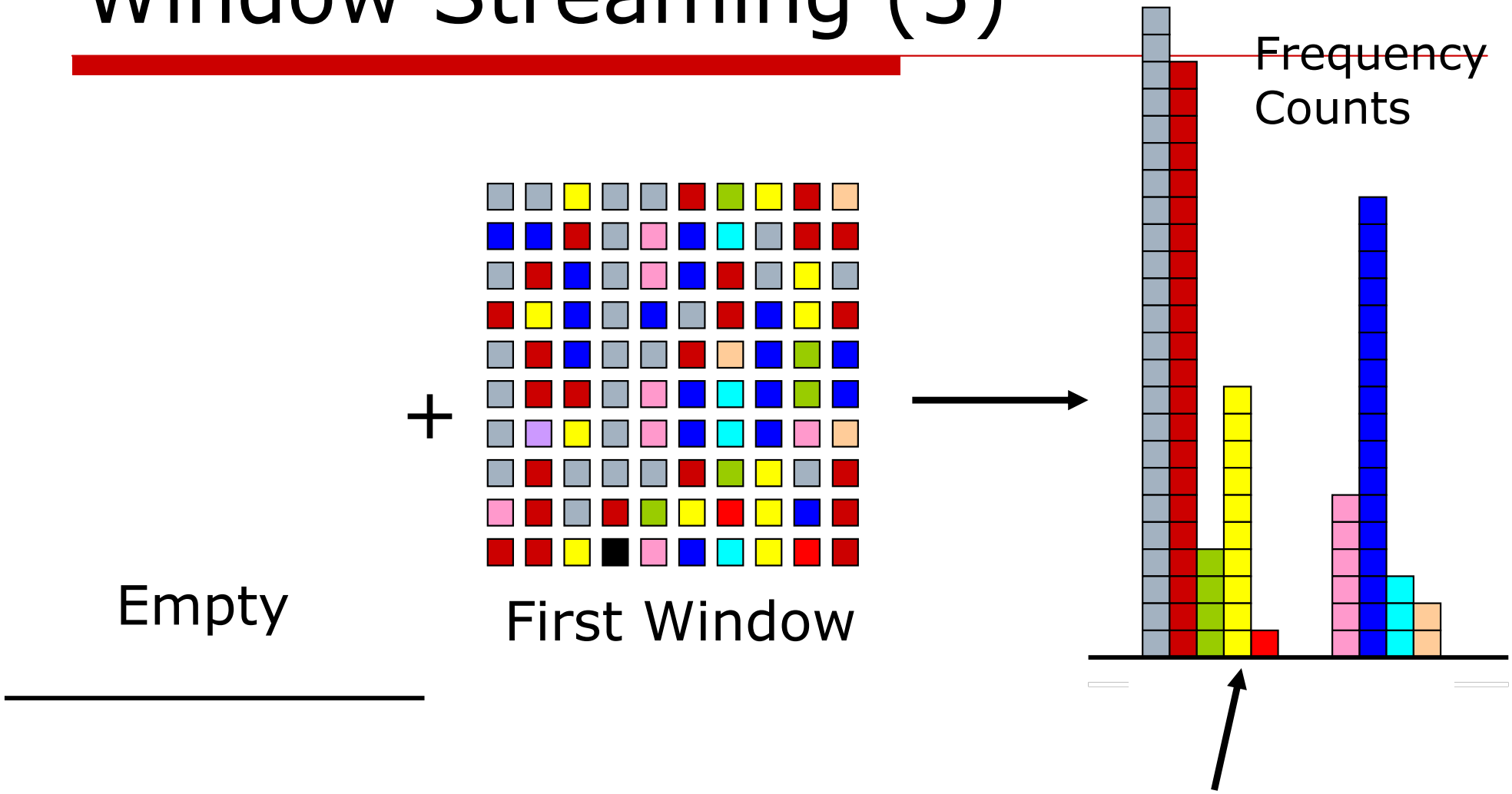
Window Streaming (2)

Step 1: Divide the stream into 'windows'



Quelle: *Randomization for Massive and Streaming Data Sets, http://theory.stanford.edu/~rajeev/forum-2003-talk.ppt*

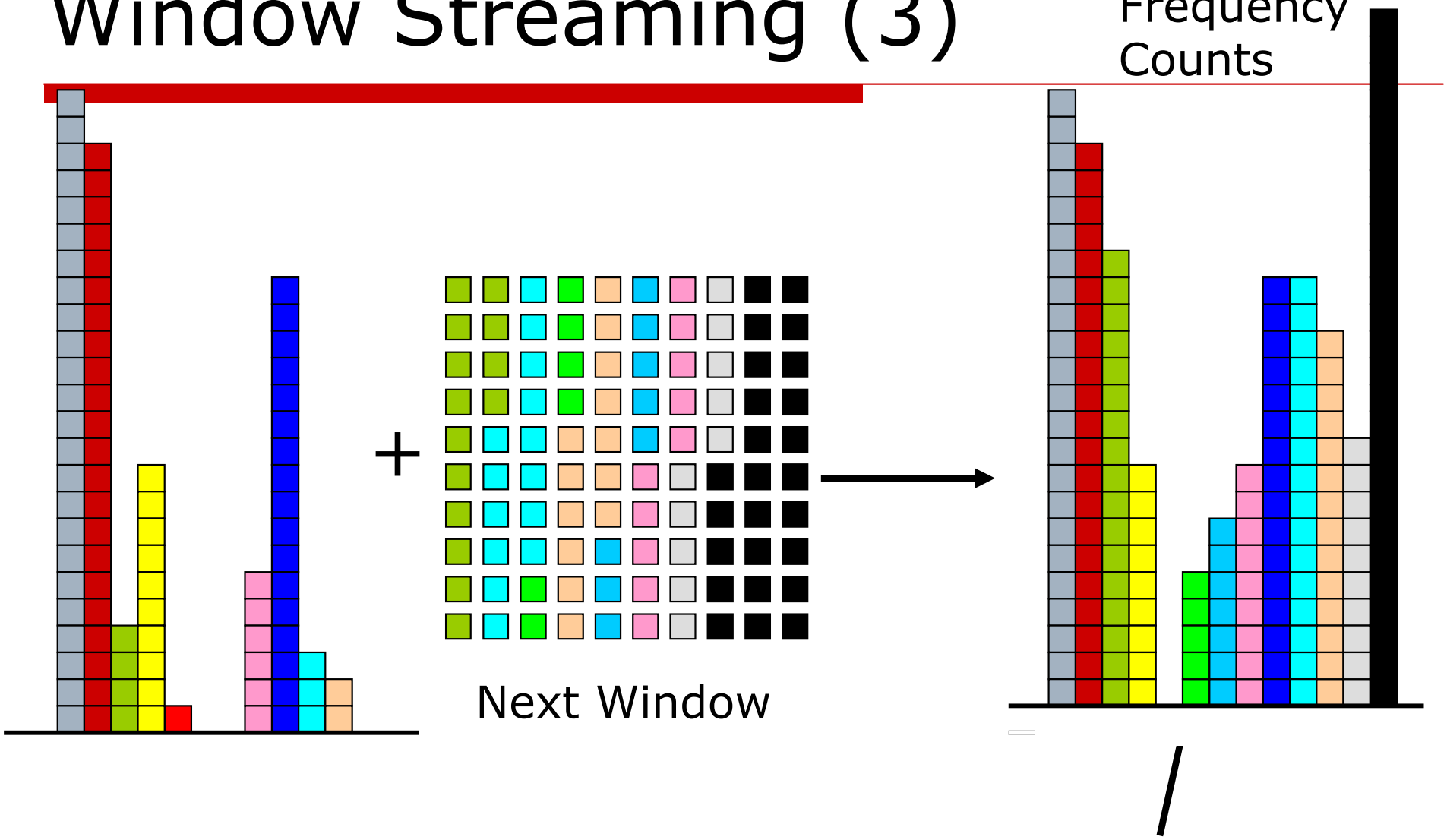
Window Streaming (3)



At window boundary, decrement all counters by 1

Quelle: *Randomization for Massive and Streaming Data Sets, http://theory.stanford.edu/~rajeev/forum-2003-talk.ppt*

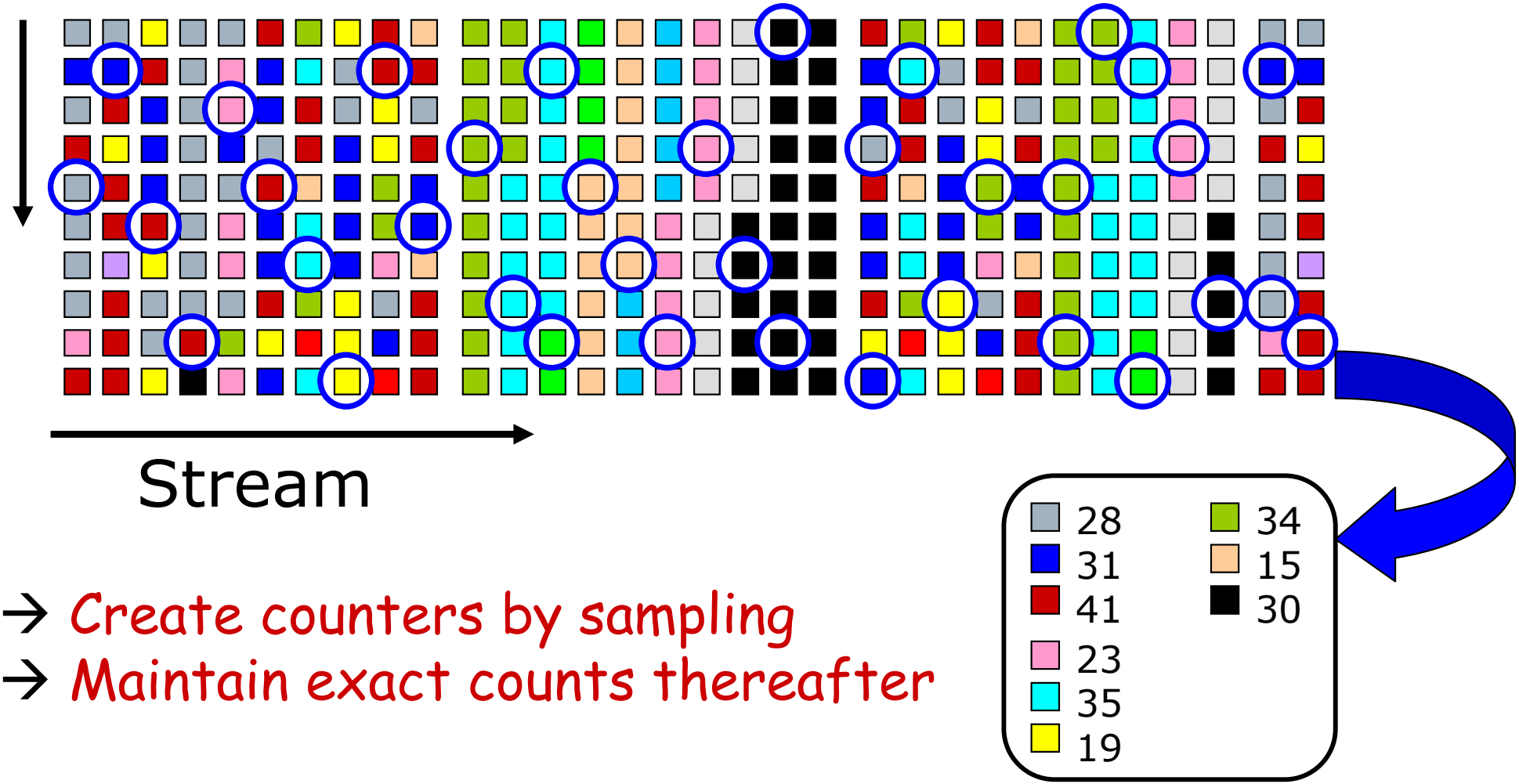
Window Streaming (3)



At window boundary, decrement all counters by 1

Quelle: *Randomization for Massive and Streaming Data Sets, http://theory.stanford.edu/~rajeev/forum-2003-talk.ppt*

Permutation Streaming



→ Create counters by sampling
→ Maintain exact counts thereafter