Database / Data Mining Visualization

DataJewel: Tightly Integrating Visualization with Temporal Data Mining.

Mihael Ankerst, David H. Jones, Anne Kao, Changzhou Wang. ICDM Workshop on Visual Data Mining, Melbourne, FL, 2003

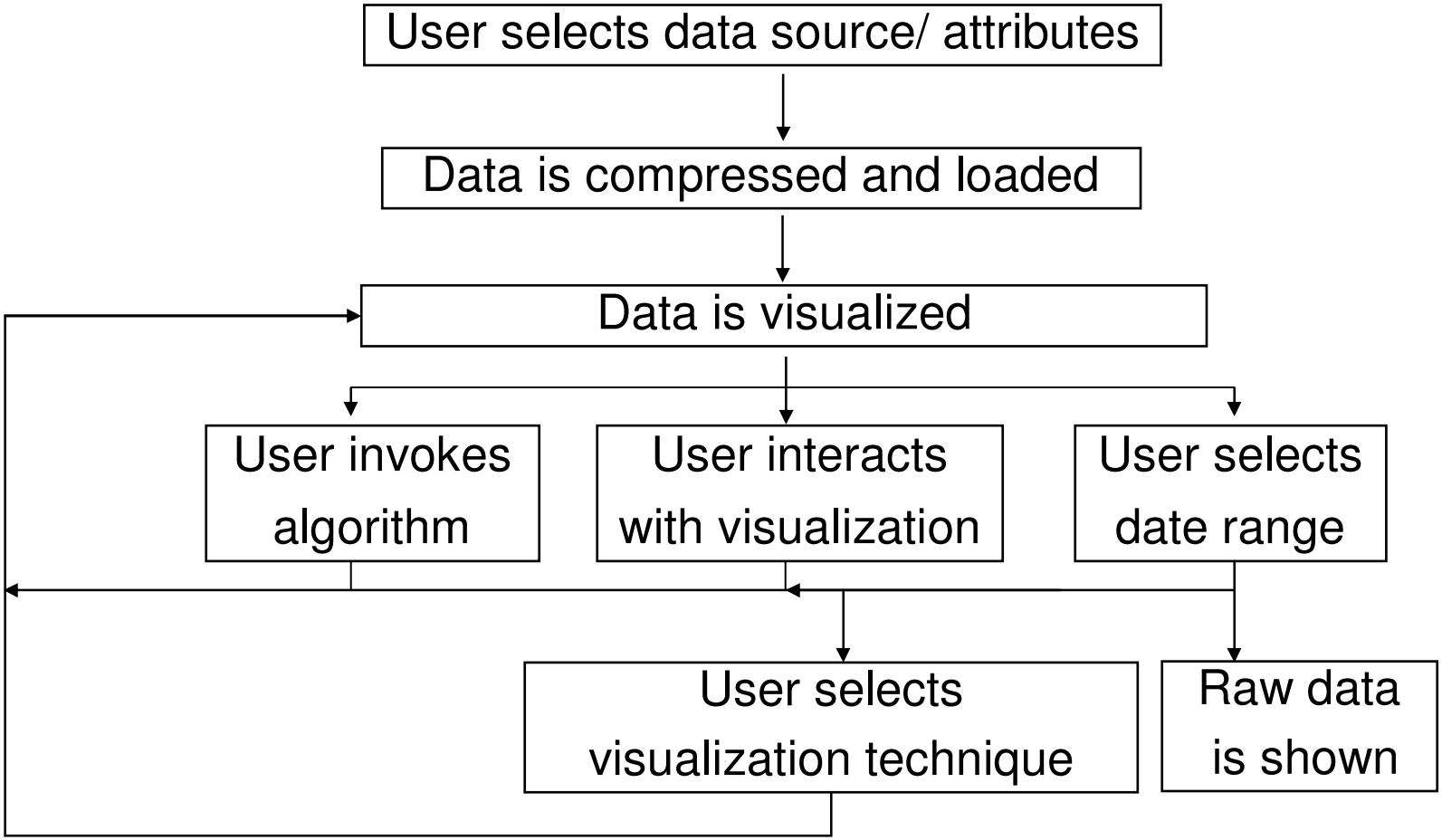
What is Data Mining ?

- **Data mining**, also known as **knowledge-discovery in databases (KDD)**, is the practice of automatically searching large stores of data for patterns.
- data mining uses computational techniques from statistics and pattern recognition.

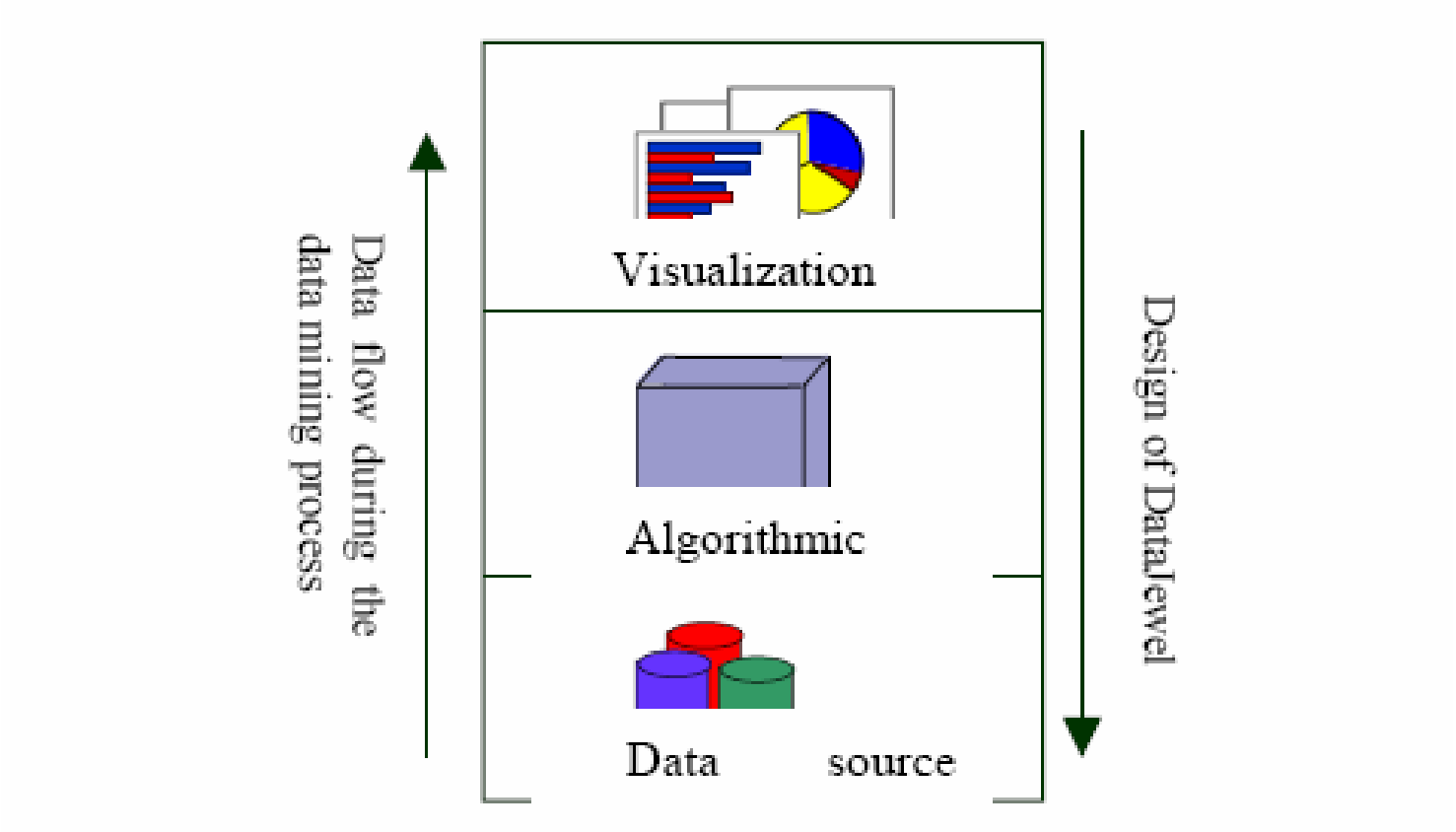
Temporal Data Mining

- Each record has a timestamp
- Databases evolve as a consequence of organizational need
- linking together two databases with respect to time can give us a powerful tool to explore the union of attributes

User-centric data mining



Architecture

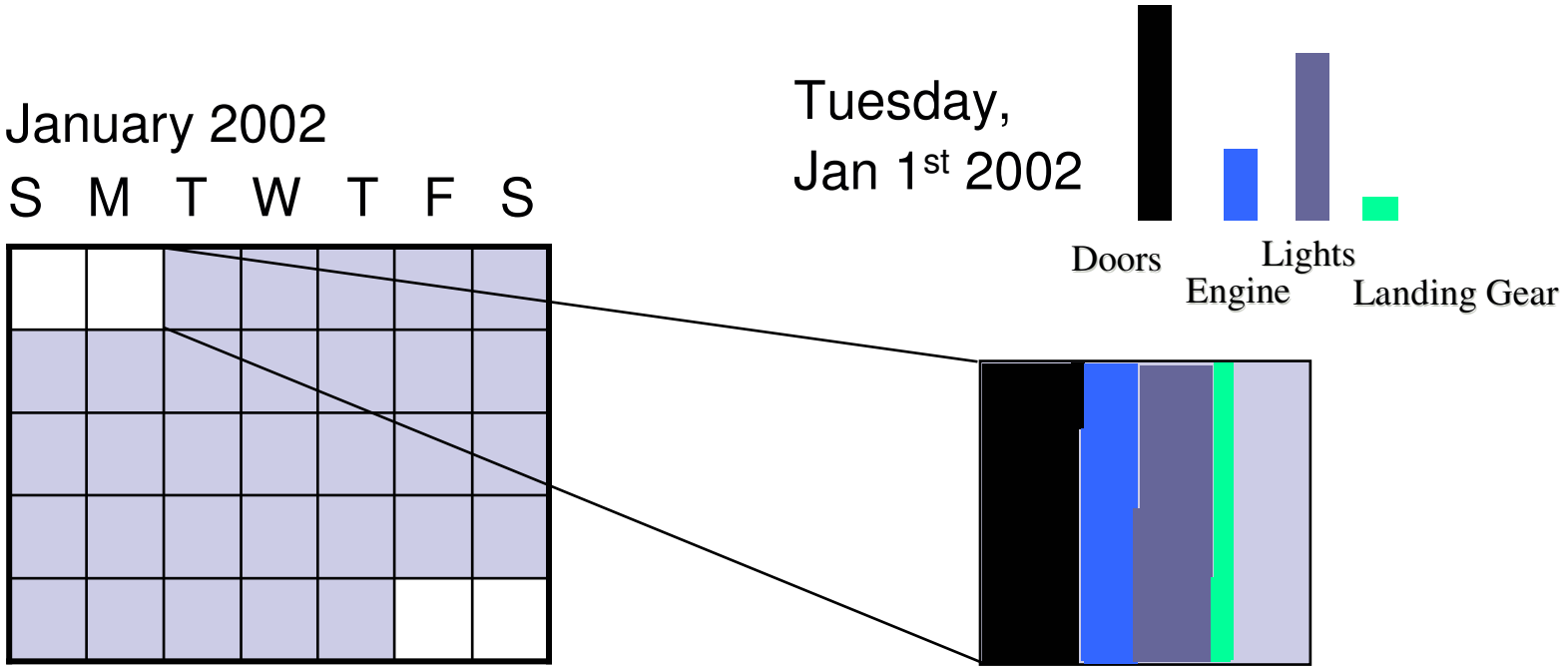


The Visualization Component

- Calendar View
  - Visual metaphor: Calendar.
  - Structure of data is represented along the event dates is the frequency of events.
  - Designed for domain experts – intuitive and versatile design
- If there are few events the visualization is powerful since human's pre-attentive perception is very efficient in looking for variety of patterns

The Visualization Component

Time	Event type	Location	…
09/11/2001	Door broken	Seattle	…
09/12/2001	…	…	…



The Visualization Component - interaction

- Selection – subset of dates
- Ascending/descending order frequency
- Interactive color assignment
- Zooming
- Detail on demand

The Temporal Mining Component

- Have algorithms that discover patterns
- Determine which events are involved in the patterns
- Automatically select colors based on the patterns

- Visualize not just data but also patterns
- Use of the same color assignment interface by user and algorithm.

The Temporal Mining Component

- Discover one event of one event attribute
  - For example - highest variance, most interesting trend - give the event a unique color
- Discover multiple events of one event attribute
  - Set of events that together represent a pattern (for example - discovery of similar events) - each event that is part of the pattern receives a distinct color
- Discover one event for each event attribute
  - Look for patterns relating event attributes to each other instead of analyzing them separately. (for example – finding similar events across different event attributes) – update the color assignments of each event attribute accordingly.

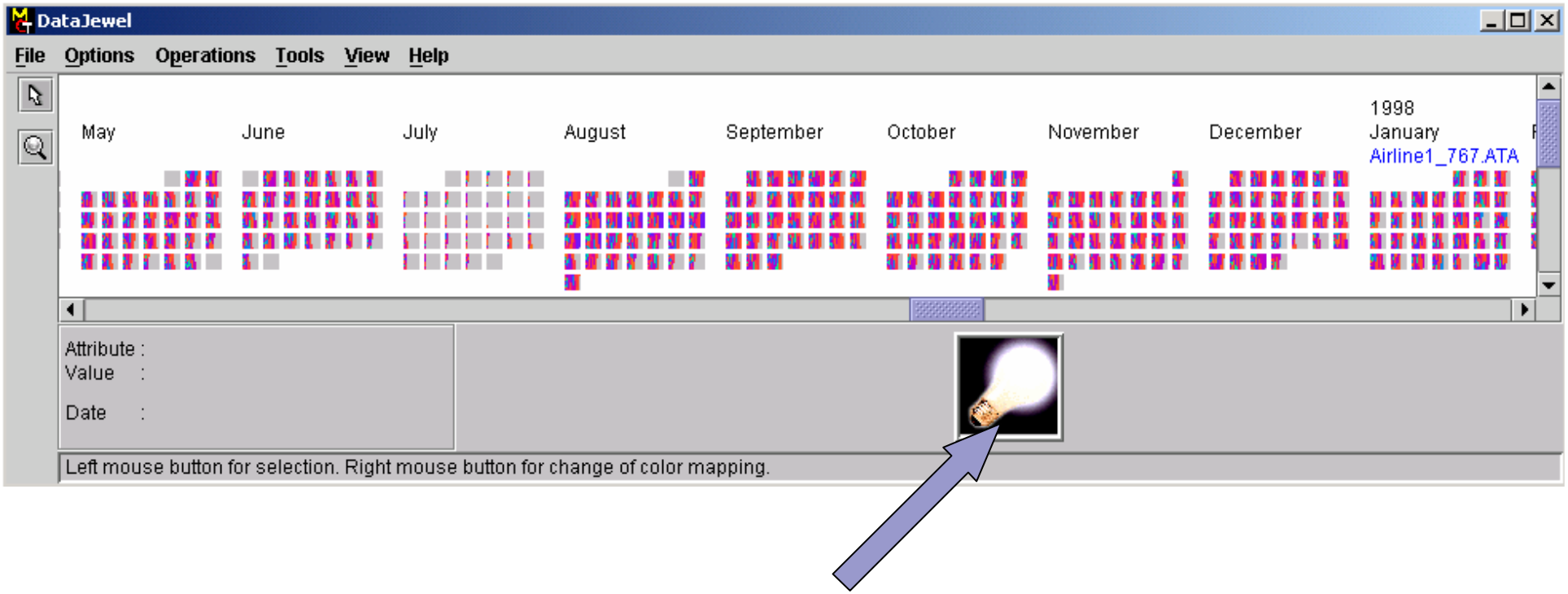
The Database component

- Each event is stored in one record
- Data resides in tables in one or more relational databases
- Aggregate database events according to event date (using select count(*) … group by …)
- Access the raw data of all attributes

DataJe gorithm

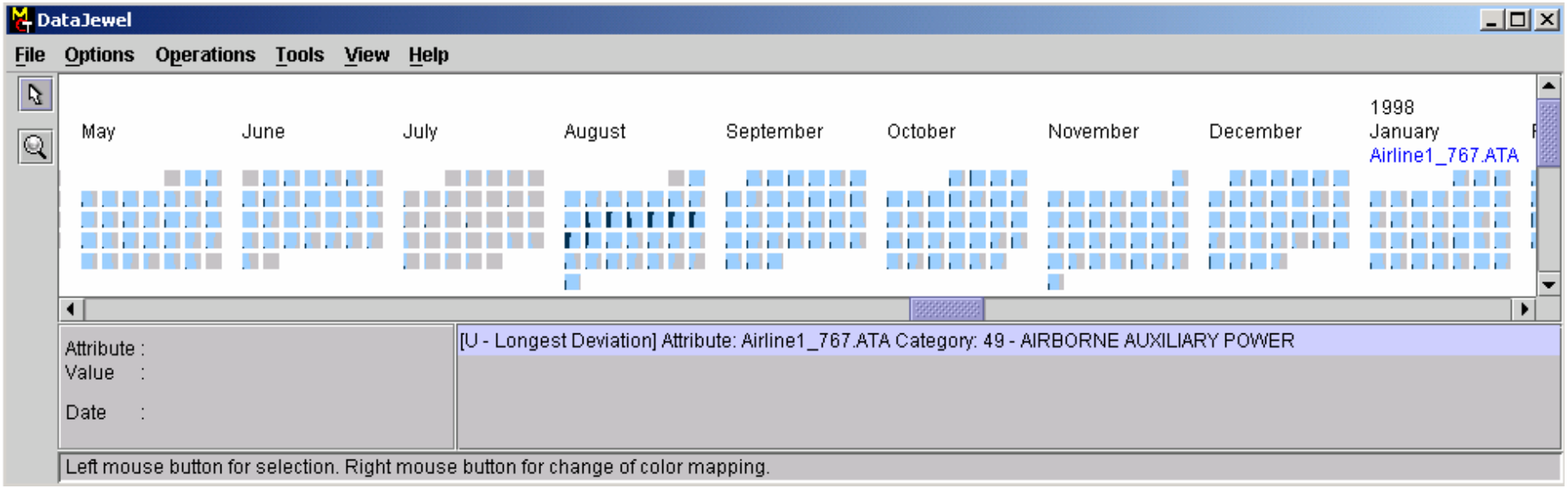
Category name	Frequency	Category color
05 - MAINTENANCE CHECKS	374	
21 - AIR CONDITIONING/PRESSURIZATION	10381	
22 - AUTO FLIGHT	9161	
23 - COMMUNICATIONS	50226	
24 - ELECTRICAL POWER	8160	
25 - EQUIPMENT/FURNISHINGS	82968	
26 - FIRE PROTECTION	4685	
27 - FLIGHT CONTROLS	7725	
28 - FUEL	9382	
29 - HYDRALIC POWER	3959	
30 - ICE AND RAIN PROTECTION	1845	
31 - INDICATING/RECORDING SYSTEM	6037	
32 - LANDING GEAR	30678	
33 - LIGHTS	29777	
34 - NAVIGATION	21238	
35 - OXYGEN	8087	
36 - PNEUMATICS	5327	
38 - WATER/WASTE	10649	
49 - AIRBORNE AUXILIARY POWER	10737	
51 - STRUCTURES	800	
52 - DOORS	6189	
53 - FUSELAGE	1484	
54 - NACELLES/PYLONS	1016	
55 - STABILIZERS	171	
56 - WINDOWS	2496	
57 - WINGS	1153	
71 - POWER PLANT	3398	
72 - ENGINE GENERAL	12929	
73 - ENGINE FUEL/CONTROL	4568	
74 - ENGINE IGNITION	890	
75 - ENGINE AIR	916	
76 - ENGINE CONTROL	925	
77 - ENGINE INDICATING	4347	
78 - ENGINE EXHAUST	4979	
79 - ENGINE OIL	5633	
80 - ENGINE STARTING	1731	

DataJewel – Scenario: Mining Algorithm

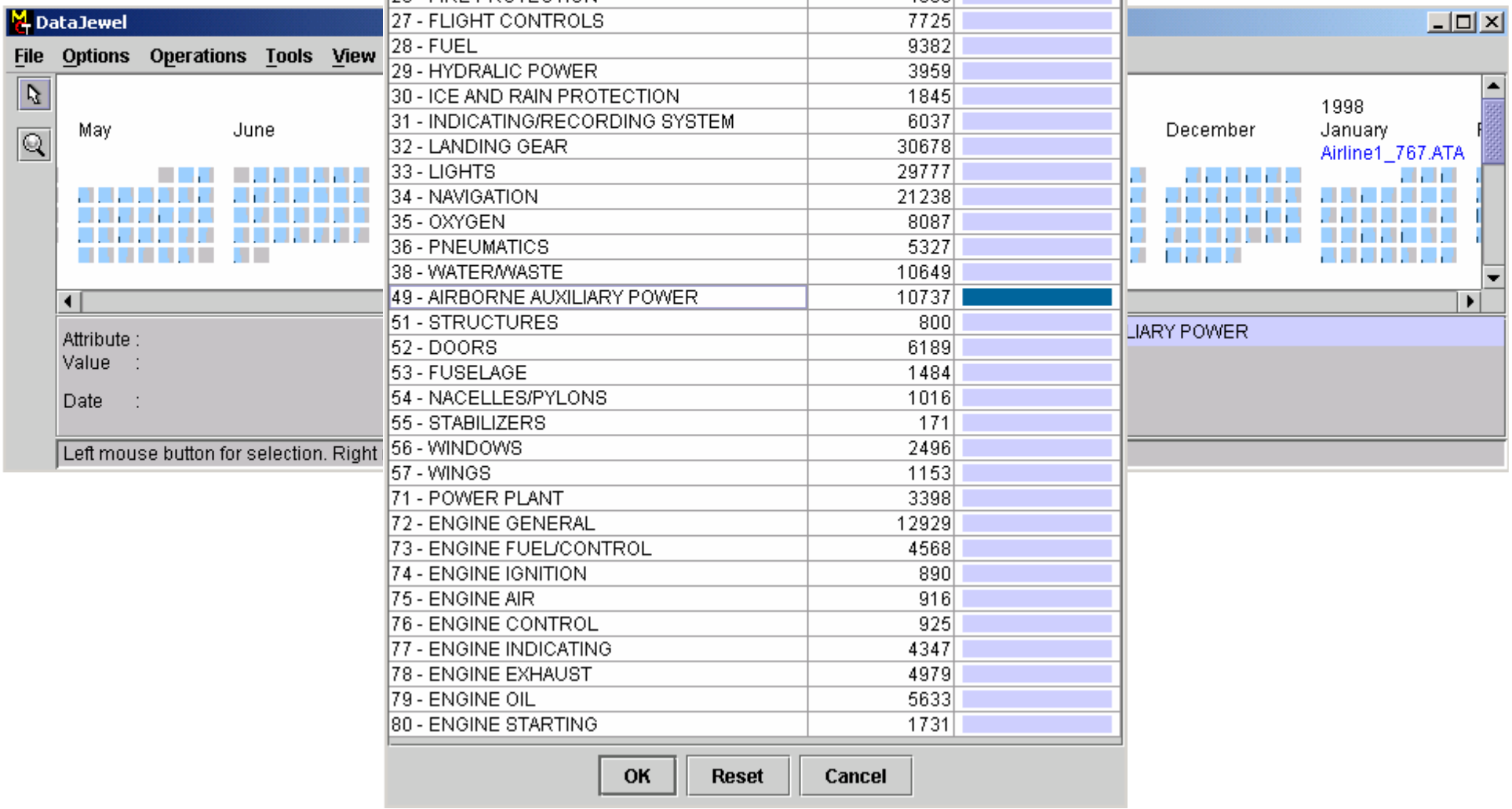


Press here for running mining algorithm

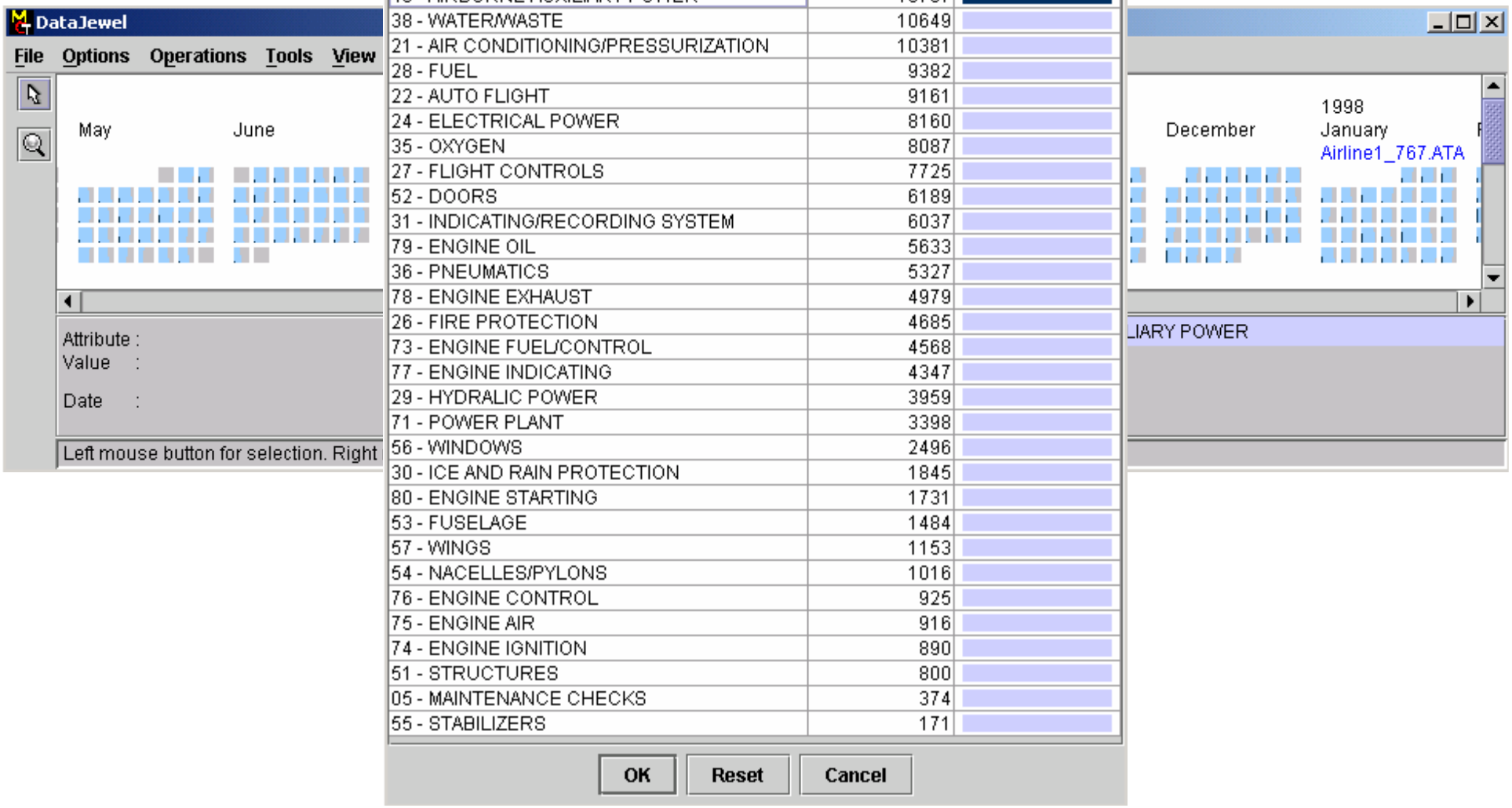
DataJewel – Scenario: Mining Algorithm



DataJe gorithm

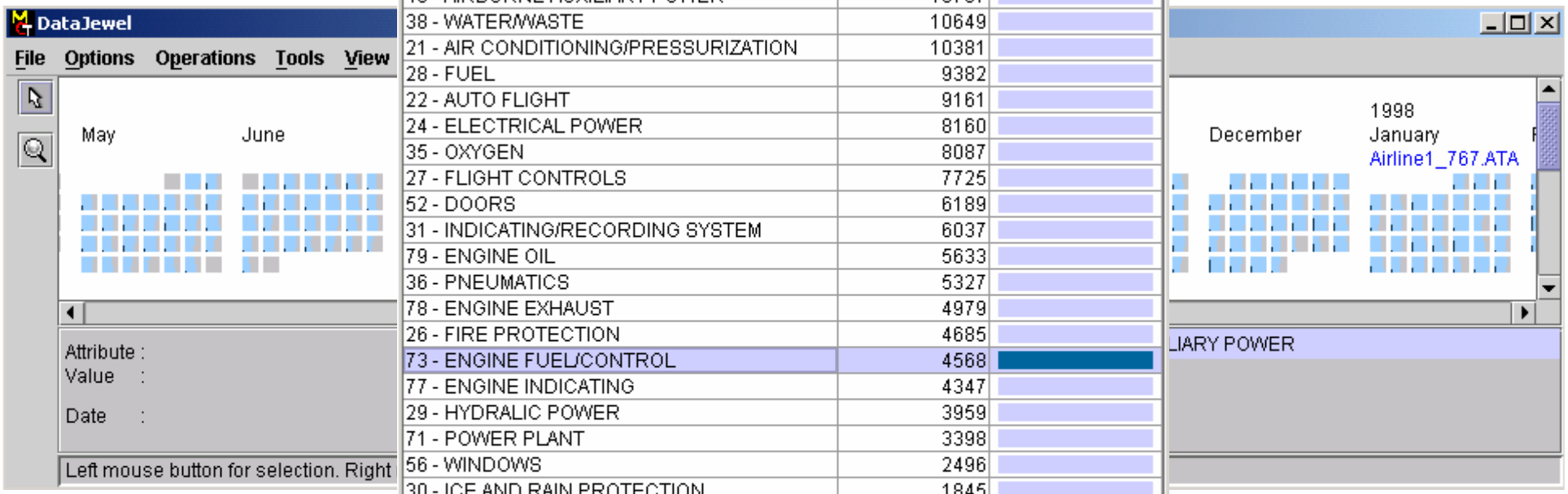


DataJe gorithm



Category name	Frequency	Category color
25 - EQUIPMENT/FURNISHINGS	82968	
23 - COMMUNICATIONS	50226	
32 - LANDING GEAR	30678	
33 - LIGHTS	29777	
34 - NAVIGATION	21238	
72 - ENGINE GENERAL	12929	
49 - AIRBORNE AUXILIARY POWER	10737	
38 - WATER/WASTE	10649	
21 - AIR CONDITIONING/PRESSURIZATION	10381	
28 - FUEL	9382	
22 - AUTO FLIGHT	9161	
24 - ELECTRICAL POWER	8160	
35 - OXYGEN	8087	
27 - FLIGHT CONTROLS	7725	
52 - DOORS	6189	
31 - INDICATING/RECORDING SYSTEM	6037	
79 - ENGINE OIL	5633	
36 - PNEUMATICS	5327	
78 - ENGINE EXHAUST	4979	
26 - FIRE PROTECTION	4685	
73 - ENGINE FUEL/CONTROL	4568	
77 - ENGINE INDICATING	4347	
29 - HYDRALIC POWER	3959	
71 - POWER PLANT	3398	
56 - WINDOWS	2496	
30 - ICE AND RAIN PROTECTION	1845	
80 - ENGINE STARTING	1731	
53 - FUSELAGE	1484	
57 - WINGS	1153	
54 - NACELLES/PYLONS	1016	
76 - ENGINE CONTROL	925	
75 - ENGINE AIR	916	
74 - ENGINE IGNITION	890	
51 - STRUCTURES	800	
05 - MAINTENANCE CHECKS	374	
55 - STABILIZERS	171	

DataJe raction

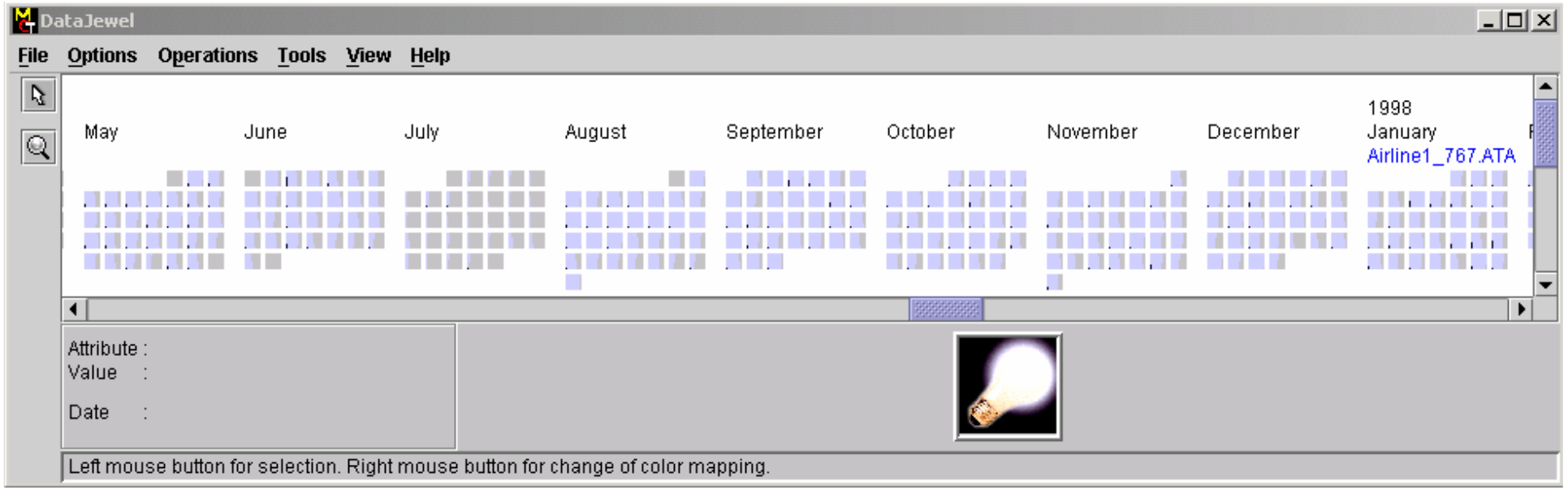


Categories for Airline1_767.ATA

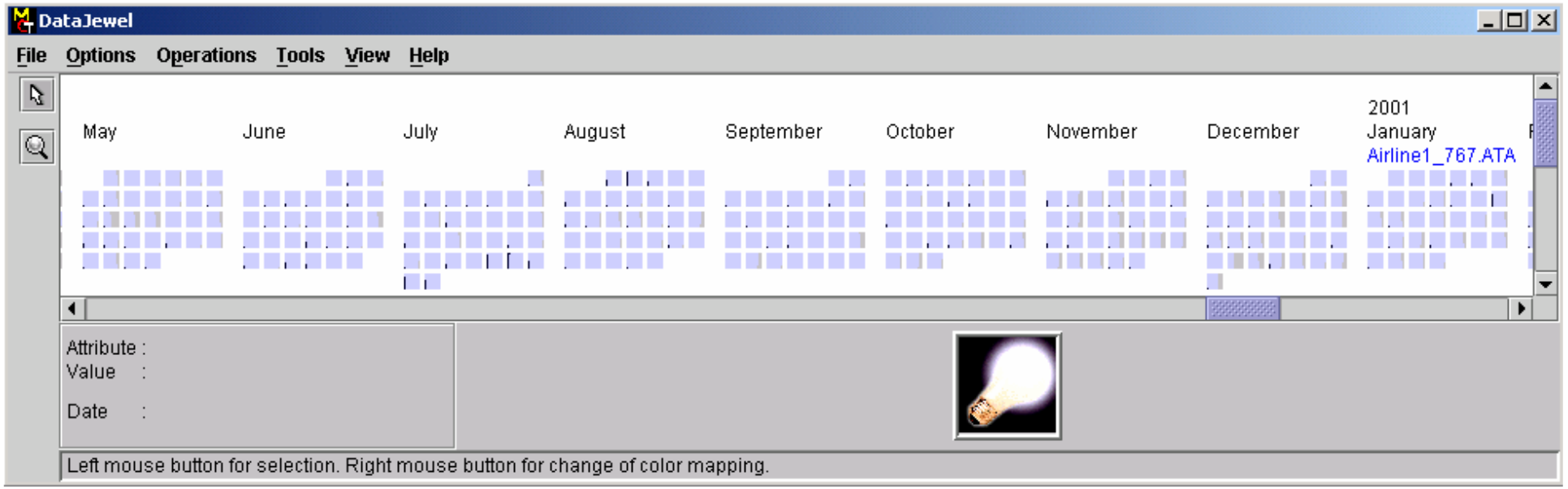
Category name	Frequency	Category color
25 - EQUIPMENT/FURNISHINGS	82968	
23 - COMMUNICATIONS	50226	
32 - LANDING GEAR	30678	
33 - LIGHTS	29777	
34 - NAVIGATION	21238	
72 - ENGINE GENERAL	12929	
49 - AIRBORNE AUXILIARY POWER	10737	
38 - WATER/WASTE	10649	
21 - AIR CONDITIONING/PRESSURIZATION	10381	
28 - FUEL	9382	
22 - AUTO FLIGHT	9161	
24 - ELECTRICAL POWER	8160	
35 - OXYGEN	8087	
27 - FLIGHT CONTROLS	7725	
52 - DOORS	6189	
31 - INDICATING/RECORDING SYSTEM	6037	
79 - ENGINE OIL	5633	
36 - PNEUMATICS	5327	
78 - ENGINE EXHAUST	4979	
26 - FIRE PROTECTION	4685	
73 - ENGINE FUEL/CONTROL	4568	
77 - ENGINE INDICATING	4347	
29 - HYDRALIC POWER	3959	
71 - POWER PLANT	3398	
56 - WINDOWS	2496	
30 - ICE AND RAIN PROTECTION	1845	
80 - ENGINE STARTING	1731	
53 - FUSELAGE	1484	
57 - WINGS	1153	
54 - NACELLES/PYLONS	1016	
76 - ENGINE CONTROL	925	
75 - ENGINE AIR	916	
74 - ENGINE IGNITION	890	
51 - STRUCTURES	800	
05 - MAINTENANCE CHECKS	374	
55 - STABILIZERS	171	

OK Reset Cancel

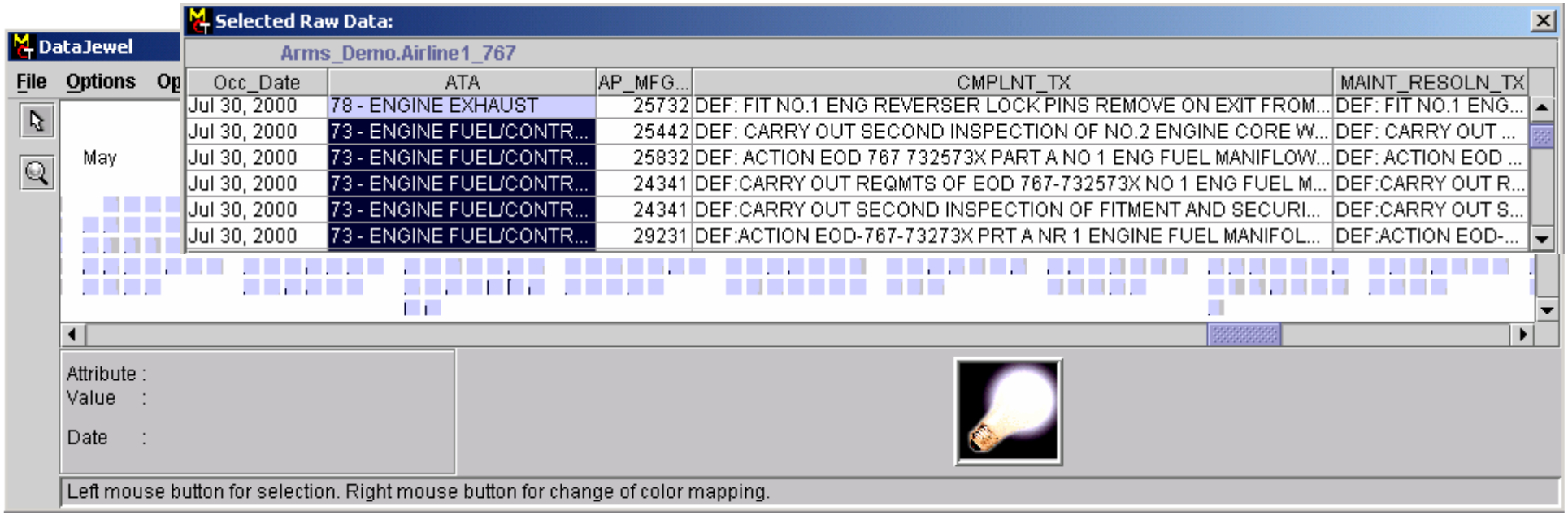
DataJewel – Scenario: User Interaction



DataJewel – Scenario: User Interaction



DataJewel – Scenario: User Interaction



Critique (+)

- Combine data mining algorithms with visualization
- Can work with several databases
- Scalable – handles large databases
- Intuitive and easy to use – don’t need a data mining expert

Critique (-)

- Hard to see patterns over weeks or months or within a single day
- Only one event attribute for each calendar presentation
- Not easily transferable to other domains like author claims.
- Only for categorical attributes
- Does not handle other types of databases other than relational
- No user studies

DE**Vise: Integrated** Q**uerying and Visual** E**xploration of** L**arge** D**atasets**

Miron Livny, Raghu Ramakrishnan, Kevin Beyer, Guangshun Chen, Donko Donjerkovic, Shilpa Lawande, Jussi Myllymaki, and Kent Wenger. Proc. .SIGMOD 1997

What is DEVise?

- A data exploration system that allows users to develop, browse, and share visual representations of datasets from several sources.
- A framework which describes a set of querying and visualization primitives that is combined to develop a visual presentation.

Basic concepts

- Mapping each source data record to a visual symbol on screen

TData (Textual Data) – a collection of records with one or more attributes (along with a schema).

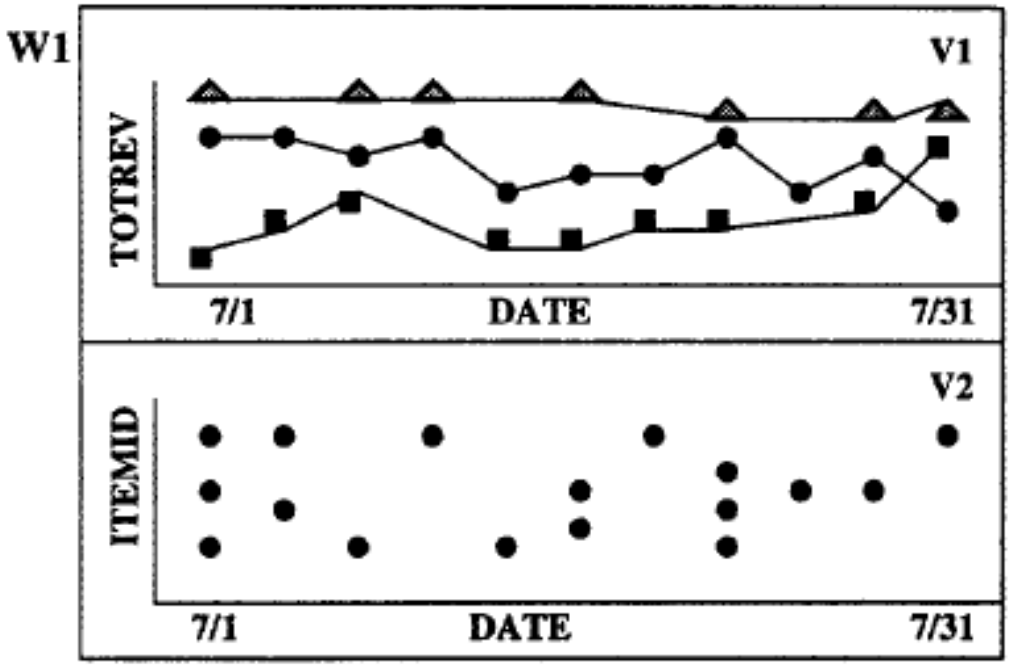
GData (Graphical Data) – high level representation of the screen (x, y, size, color, pattern, orientation, shape

Mapping – a function that is applied to the TData record to produce a GData record.

Basic concepts - presentation

- **View** – basic display unit
  - TData
  - mapping
  - Background (title, axes)
  - data display
  - cursor display – additional data independent information
  - visual filter - set of selection (a query) on the GData of a view
- **Window** – collection of views
- **Visual presentation** – collection of windows

Visualization model



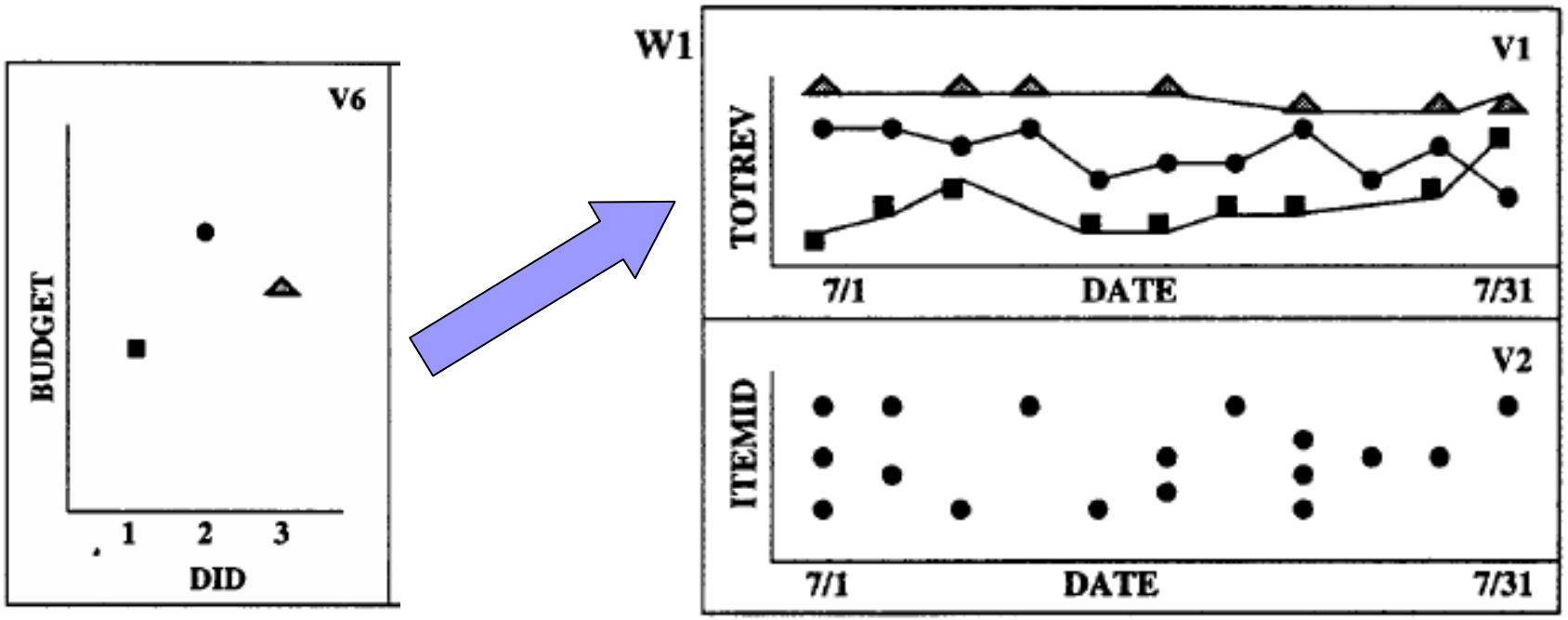
Overall_sales (date, Did, totRev)

Sales (date, itemid, custid, number)

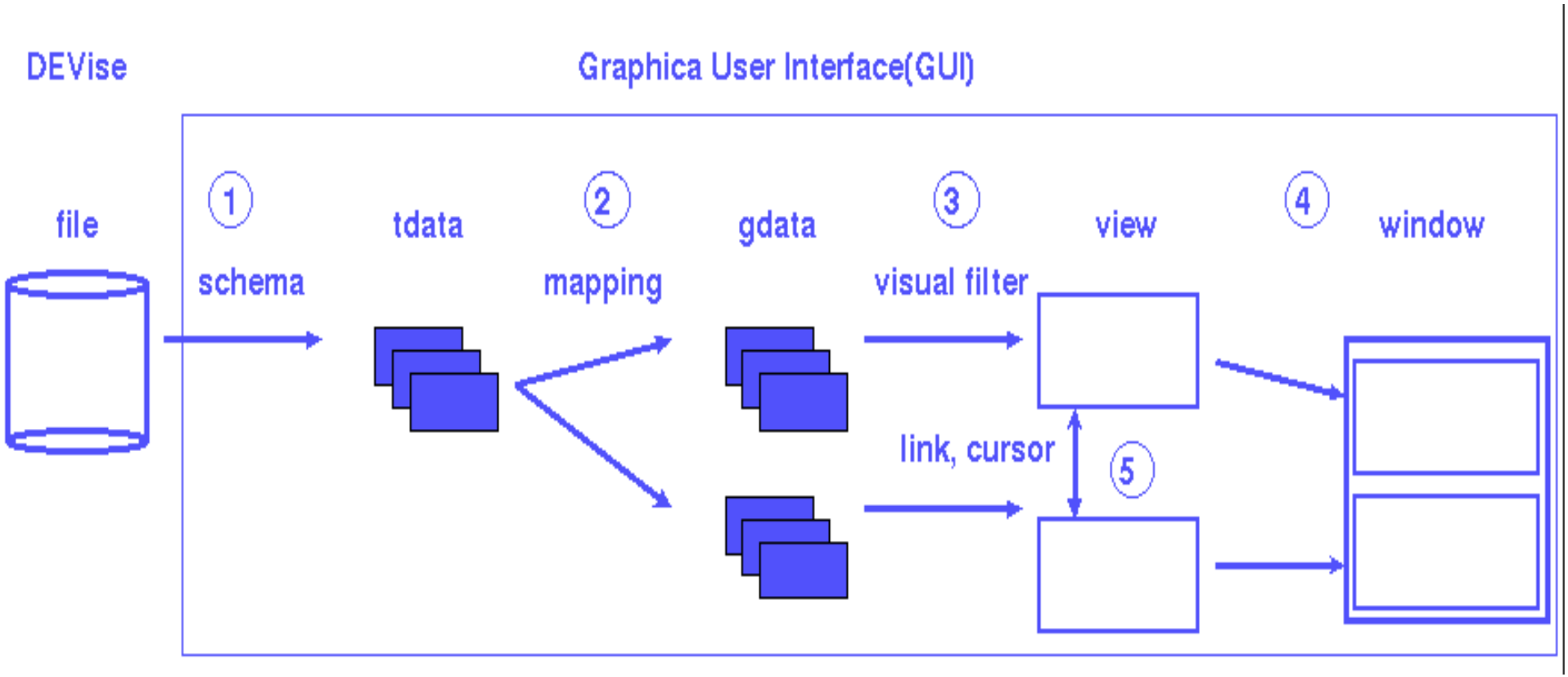
Some more concepts…

- Cursors – allows the visual filter of one view to be seen as a highlight in another view
- Links – constraints that allows the contents of two views to be coordinated.
  - Visual – associate visual filters of two views
  - Record – the projection of the data in one view (on the linked attributes) will act as a filter on the TData of the other view
  - Operator
  - aggregate

Record link example



DEVise Model



Semantics of a visual display

A mapping function is applied from the TData record to produce a Gdata record:

$$< t_1, t_2, \cdots, t_m > \xrightarrow{\sigma_1} g_1$$
$$< t_1, t_2, \cdots, t_m > \xrightarrow{\sigma_2} g_2$$
$$\vdots$$
$$< t_1, t_2, \cdots, t_m > \xrightarrow{\sigma_n} g_n$$

A view can then be represented as: (B, σ^G, μ, T, C)

B – Background

Sigma – visual filter

Mu – mapping

T – TData

C – cursor layer

Visual Queries and SQL

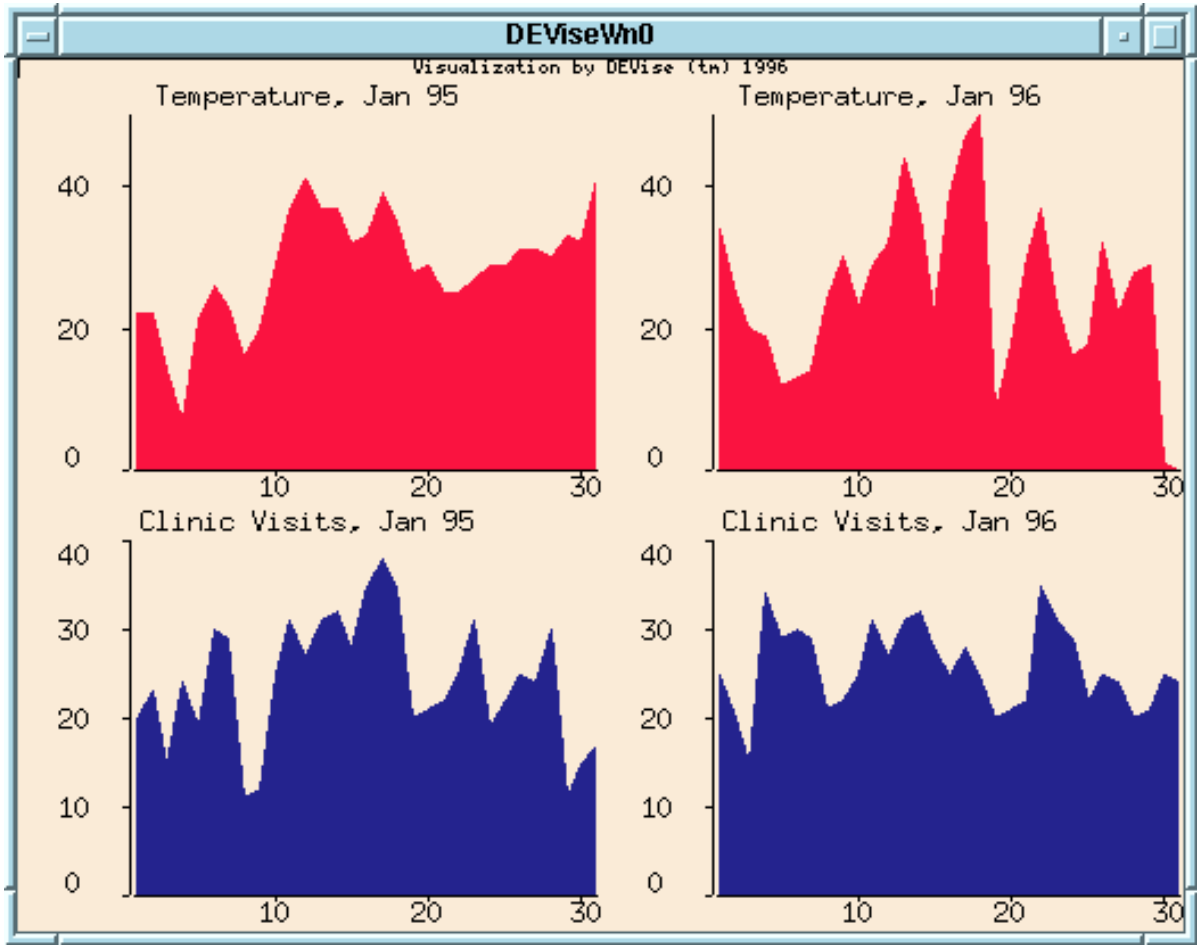
- Visual queries – user selection on visual attributes of a view. (zoom in/out, scroll, point selection)
- Can save and transfer a visual query
- Enables users to generate sophisticated SQL queries through intuitive graphical operations
- Can be used as an SQL front-end (but not only!)

Achievements

- Visual presentation capabilities – users can render their data. Simple mapping between data and presentation
- Ability to handle large distributed databases (not limited to available memory)
- Collaborative data analysis
- Support for interactively exploring the data visually at any level of detail

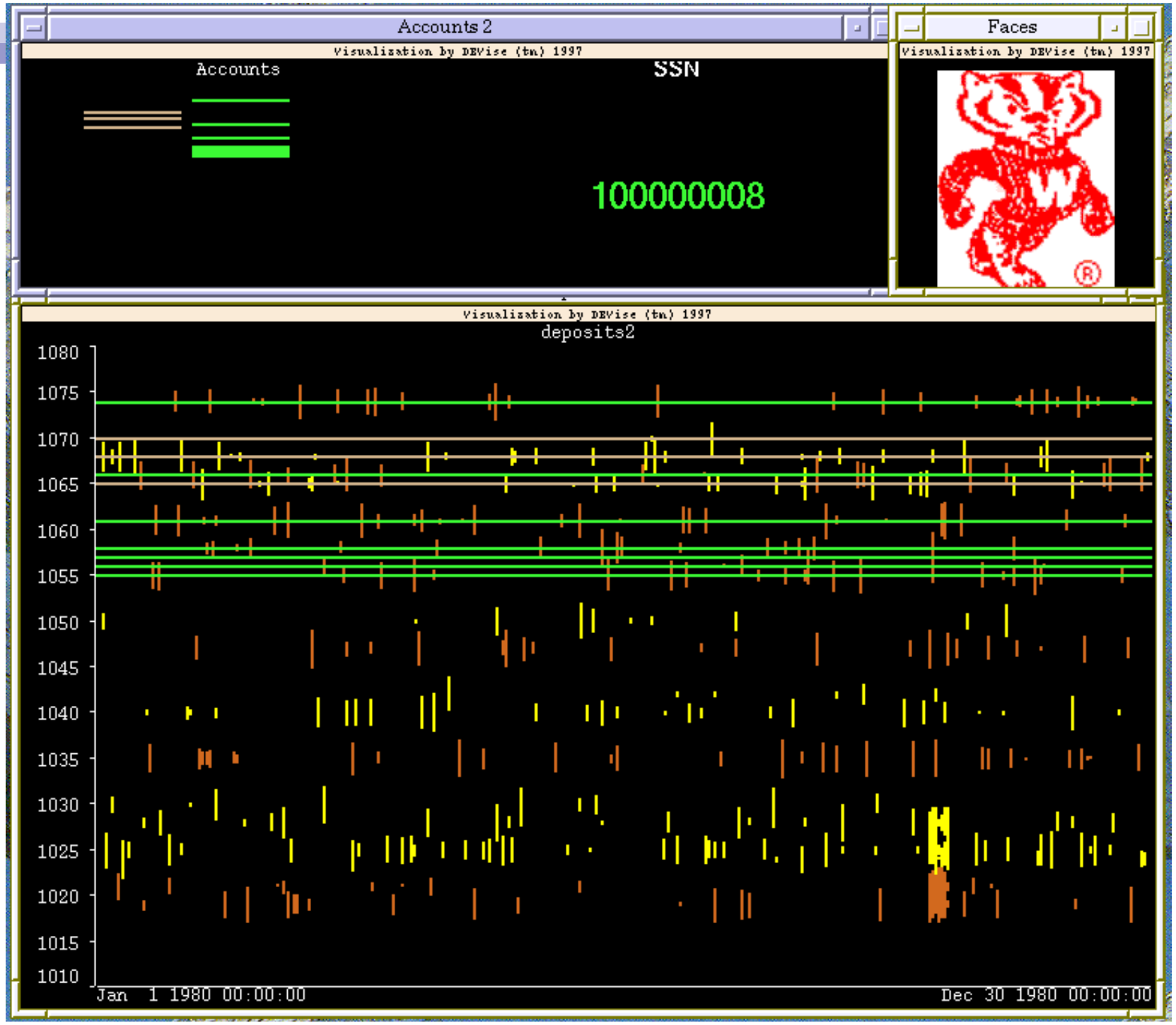
Example

Input two data sources: clinic information about number of visits, and information about temperature



Another Example:

- Input data: has information about deposits into various accounts at 2 different banks:
  - Account (bankNum, SSN, accNum, pic, …)
  - Deposit (accNum, date, amount)
- problem: We want to analyze the transactions to find out who has a suspiciously large number of transactions within a short period of time.


Accounts 2
Visualization by DEVise (tm) 1997
Accounts
SSN
100000008
Faces
Visualization by DEVise (tm) 1997
Visualization by DEVise (tm) 1997
deposits2
1080
1075
1070
1065
1060
1055
1050
1045
1040
1035
1030
1025
1020
1015
1010
Jan 1 1980 00:00:00
Dec 30 1980 00:00:00

critique

+

- Very thorough well-defined framework
- Many examples of implementations in real application

-

- Leaves the visualization decisions to the user (but that's the idea…)
- Some visualizations are very hard or impossible to do

Questions?