

Análisis de Paper AJAX

“Declarative cleaning your data using AJAX”
Tratando de enfocarlo en el modelado del proceso como Grafo

Introducción

- ♦ Problema: limpieza de información.
- ♦ Contexto:
 - Estructuración de información poco estructurada
 - Number of Sources:
 - Single data source
 - Multiple sources
- ♦ Modelado como: grafo de transformación de información.

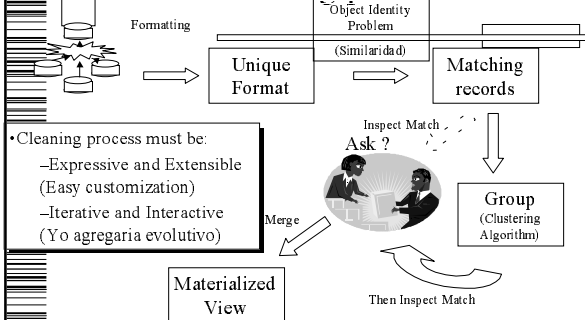
Quality & Anomalies

- ♦ Quality: “Data quality concerns arise when correcting anomalies ...”
- ♦ Data Cleaning: “The main goal of a data cleaning process is to eliminate anomalous data ...”
- ♦ Anomalies:
 - The absence of universal keys = Object Identity Problem ?
 - Missing Values
 - The use of different data formats

Quality & Anomalies (II)

- ♦ Anomalies:
 - The existence of data entry errors (“John Smith”, “Joh Smith”)
 - The presence of inconsistencies in data
- ♦ Data Cleaning (II): “Dealing with these problems is globally called the data cleaning process”

Ex. of Cleaning process in DW



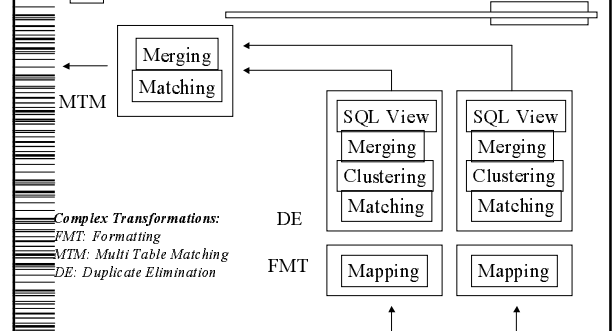
AJAX

- ♦ Is a **framework** (librería de funciones)
 - Extensiones:
 - Mas funciones en la libreria
 - Mas algoritmos de clustering
- ♦ Provides a **declarative and extensible language** (SQL + algunas primitivas)
- ♦ **Optimizations** (SQL Query Engine + external home-grown execution engine) (Producto Cartesiano)
- ♦ **Data Lineage** (debugger)

Cleaning as a Directed Acyclic Graph

- Directed Acyclic Graph of Atomic Transformations (**Process**)
 - Each one accepts as input the data flow(s) from previous transformation(s)
 - And returns one or more data flows
 - Tienen restricciones sobre la salida.
- Five types of atomic Transformations (**Process Types**):
 - Mapping: Standardizes data formats
 - Matching: Pairs of records that refer to the same object
 - Clustering: Group together matching pairs
 - Merging: Collapses each individual cluster
 - SQL View: An SQL join or union
- Complex Transformations (**Sub-Process**): Is a sequence of one or more atomic transformations

Example of Graph



Data Cleaning Program

```
<data cleaning program>::
  (<declaration of complex type>)*
  (<registration of functions and algorithms>)*
  (<declaration of input data flow>)+
  (<transformation definition>)+

<transformation definition>::
<CREATE>(<MAPPING>|<MERGING>|<CLUSTERING>|<MERGING>|<SQLVIEW>) <output data flow>
(<FROM> <input data flow>)+
....
  (<CONSTRAINT>(<expresion><DEP><expresion> ....
```

Atomic Transformations (Primitivas) & User Interaction

- Biblioteca de Transformaciones Atómicas
 - Data Mapping, Data Matching, Matching for Duplicate Elimination, Matching for Multi-Table Matching, etc.
- Interactúa con el usuario
 - Mecanismo excepciones (Java like).
 - Al final de cada transformación manejo excepciones.
 - Tipos de excepciones: Anticipadas / No Anticipadas.
 - Handle:
 - insert/delete/modify la tupla de la excepción
 - modificar la definición de la transformación

Comparacion de AJAX vs. SQL

- Clustering & Merging no tienen equivalente (Dice que lógicamente en SQL no se pueden llamar algoritmos externos de clustering y tampoco está la idea de consolidar tuplas en una sola).
- SQL no tiene excepciones (si hay error stop).
- Optimización (SQL99 soporta lenguaje procedural pero la optimización no explota esto).

AJAX Architecture

- La divide en:
 - Repositorios:
 - "Data Repository" (store input/output data),
 - "Library of Functions" (para transformaciones),
 - "Library of Algorithms" (para clustering)
 - Unidades operacionales:
 - "Analyzer" (parse program),
 - "Optimizer" (choose plan),
 - "Execution Engine" (execute transformation),
 - "Explainer" (audit, continua ...)

AJAX Architecture (II)

- ♦ Explainer (saber que pasa con los datos)
 - Debug o
 - Refinar el proceso
 - Acciones:
 - Cambio (errores en la construccion del proceso)
 - Modificaciones locales (toco los datos origen)
 - “Incremental Cleaning” (solo lo menciona)

Optimizations

- ♦ Mixed Evaluation (parte dentro / parte fuera del DBMS)
- ♦ Operation re-ordering (subir .)
- ♦ Neighborhood Hash Join (similarity, distancia con posibles tuplas de Join)
- ♦ Short-Circuited Computation (lazy evaluation)
- ♦ Cached Computation (Cache para funciones externas)
- ♦ Parallel Computation (Uso de paralelismo y replicacion)

Related Work

- ♦ Main Approaches for Cleaning Data
 - Dedicated Application Program
 - Solucion no flexible (refinamiento = reprogramacion).
 - No aprovecha capacidades de las BD (p.e. Join).
 - Optimizaciones son dejadas al programador.
 - Normalization functions (se pueden integrar)
 - Solo para campos estandarizables (p.e. US address).
 - Diccionarios pre-construidos.
 - Multi-Pass Neighborhood Method (Se puede hacer)
 - Repetir 2 pasos (elegir clave, comparar con los cercanos)
 - Interleave discovery of data with their correction
 - Tradicionalmente 2 pasos Analisis (data mining) y Limpieza.
 - AJAX se concentra en limpieza.

Productos Comerciales

- ♦ Fixed key based matching strategy, usually variants of the multi-pass neighborhood method.
- ♦ Matching/Clustering/Consolidation are not all supported by every tool.
- ♦ Criterios limitados por lo que el producto soporta.
- ♦ Do not enterely support the user interaction during the cleaning process.
- ♦ En general no soportan languages de especificacion.
- ♦ Backtracking no es soportado.

Ideas Interesantes

- ♦ Transformaciones complejas . Primitivas Complejas
- ♦ Lenguaje de definicion de primitivas
- ♦ Excepciones para manejo de errores e imprevistos.
- ♦ Refinamiento de proceso mediante trace (componente Explainer)

Conclusiones

- ♦ Hay ideas interesantes que se pueden extrapolar a las primitivas y los procesos de carga.
- ♦ El modelado del proceso como grafo es mas bien título (porque en el abstract ?).
- ♦ Se focalizan en describir el framework y el lenguaje de consulta, tambien dan un overview de las principales tecnicas de optimizacion.
- ♦ A corto plazo se proponen:
 - Estudiar la creacion de diccionarios durante el proceso.
 - Estudiar refinamientos del proceso de limpieza (incremental cleaning)

Conclusiones (II)

- ♦ A largo plazo se proponen:
 - Un nuevo conjunto de transformaciones que permitan integrar informacion estadistica y metodos de aprendizaje para analizar la informacion y ayudar al usuario.

Referencias interesantes

- ♦ W, Cohen. Some practical observations on integration of Web information. In WebDB'99 Workshop in conj. With ACM SIGMOD, 1999.