WETSUIT: An Efficient Mashup Tool for Searching and Fusing Web Entities

Stefan Endrullis, Andreas Thor, Erhard Rahm
Database Group, University of Leipzig
http://dbs.uni-leipzig.de

Motivation
What are the top cited papers of the VLDB 2002?
- Tasks involved:
  - Requires efficient entity search: e.g. start with one venue query, refine with a keyword query for each DBLP publication
  - Requires entity resolution: e.g. compare pub. titles and authors
  - Present intermediate results to reduce waiting time for user
  - May increase recall while reducing the number of search queries

WETSUIT
What is WETSUIT?
- WETSUIT: Web Entity Search and Fusion Tool
- Efficient and powerful mashup framework

Features
- Efficient execution of mashup workflows
  - Efficient and effective entity retrieval from entity search engines using advanced search strategies which exploit multiple query generators
- Fast presentation of first results
  - By streaming partial results immediately to the next operators
- Pipeline parallelism: all workflow operators work in parallel as long as they have input entities to process
- Data parallelism: each operator may process multiple entities simultaneously
- Entity resolution to deal with dirty data
- Well supported by development tools (e.g. Eclipse, IntelliJ IDEA, maven, ant)
- Automatic GUI generation based on workflow definition

Mashup Operators
- Selected mashup operators of WETSUIT
  - Operator: Definition
    - inputOne (label) asks the user for one/more input value(s); label denotes the caption of the GUI component
    - selectOne (label) displays intermediate results and asks the user to select one/more of them
    - outputOne (label) presents results
    - map (f) maps each input entity to a new entity using the mapping function f
    - flatMap (f) maps each input entity to a set of new entities using the mapping function f
    - filter (cond) filters the input entities based on the filter condition cond
    - groupBy (groupAttr) groups the input entities by groupAttr; groupBy has to be followed by filterTop or aggregateValue
    - filterTop (n) by (orderAttr) for each group/whole input set: filters the top n elements on orderAttr
    - aggregateValue (value) via (agg. [rev]) for each group/whole input set: aggregates values using aggregation function agg and its reverse operation rev (rev is needed if input entities can be revoked)
    - findAt (ese) searches for the input entities at entity search engine ese using an implicitly defined search strategy
    - matchWithSet using (matcher) matches the input entities against entities of set, by using the match strategy matcher
    - union, intersect, diff classical set operators
    - join set, on (theta) θ-join of two input sets using the binary function theta as join condition
    - buffer (minDelay, maxDelay) buffers new entities and entity revokes for at least minDelay and at most maxDelay milliseconds

Related Work