# Automating User-Centered Design of Data-Intensive Processes

## Research Project Report (RPR)

Vasileios Theodorou

26-05-2015

Home University Supervisor:
Prof. Alberto Abelló

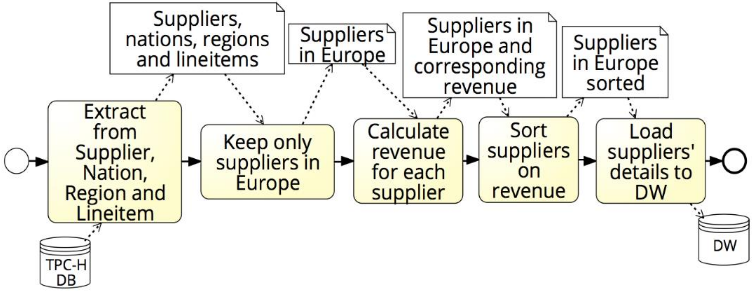
Host University Supervisor:
Prof. Wolfgang Lehner

Coadvisor:
Dr. Maik Thiele

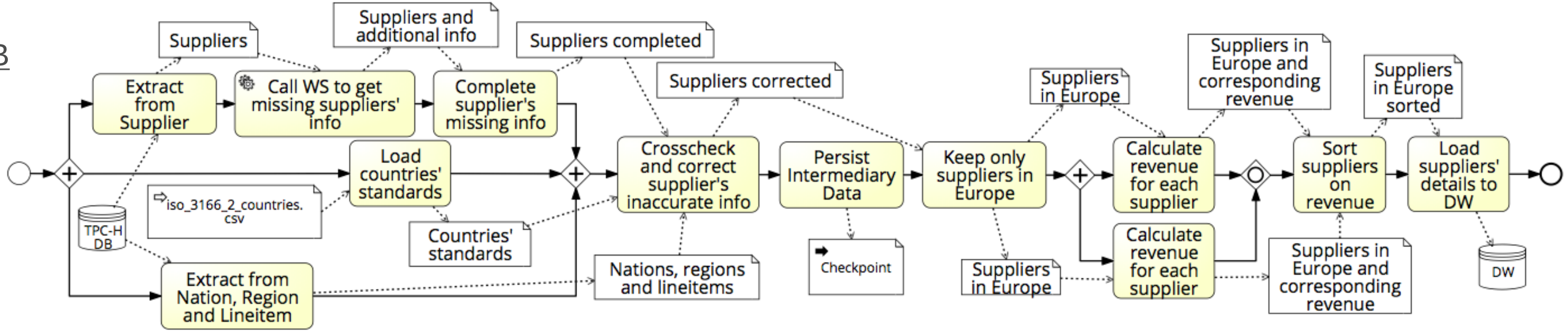
# Example - Two Alternative Flows

Conceptual model of flow: *“Details about suppliers in Europe sorted on revenue”*

- ETL Flow A


Suppliers, nations, regions and lineitems
Suppliers in Europe
Suppliers in Europe and corresponding revenue
Suppliers in Europe sorted
Extract from Supplier, Nation, Region and Lineitem
Keep only suppliers in Europe
Calculate revenue for each supplier
Sort suppliers on revenue
Load suppliers' details to DW
TPC-H DB
DW

- ETL Flow B


Suppliers
Suppliers and additional info
Suppliers completed
Suppliers corrected
Suppliers in Europe
Suppliers in Europe and corresponding revenue
Suppliers in Europe sorted
Extract from Supplier
Call WS to get missing suppliers' info
Complete supplier's missing info
Load countries' standards
iso_3166_2_countries.csv
Countries' standards
TPC-H DB
Extract from Nation, Region and Lineitem
Nations, regions and lineitems
Crosscheck and correct supplier's inaccurate info
Persist Intermediary Data
Checkpoint
Keep only suppliers in Europe
Suppliers in Europe
Calculate revenue for each supplier
Calculate revenue for each supplier
Suppliers in Europe and corresponding revenue
Sort suppliers on revenue
Load suppliers' details to DW
DW

# Measures from experiments

| | | ETL Flow A | ETL Flow B |
|---|---|---|---|
| Performance | Process cycle time | 10.4 sec | 18.9 sec |
| | Throughput | 52,906 tuples/sec | 29,179 tuples/sec |
| Data quality | % of correct tuples | 91.5% | 100% |
| | % of non-null tuples | 90.3% | 95.2% |
| Understandability | # of precedence dependencies | 20 | 40 |
| Manageability | Length of longest path | 9 steps | 23 steps |

## Execution

- TPC-H with s.f.=1
- Executed on Pentaho Data Integration (Kettle)
- Data quality improved – Performance, Understandability and Manageability reduced

# Agenda

## Approach

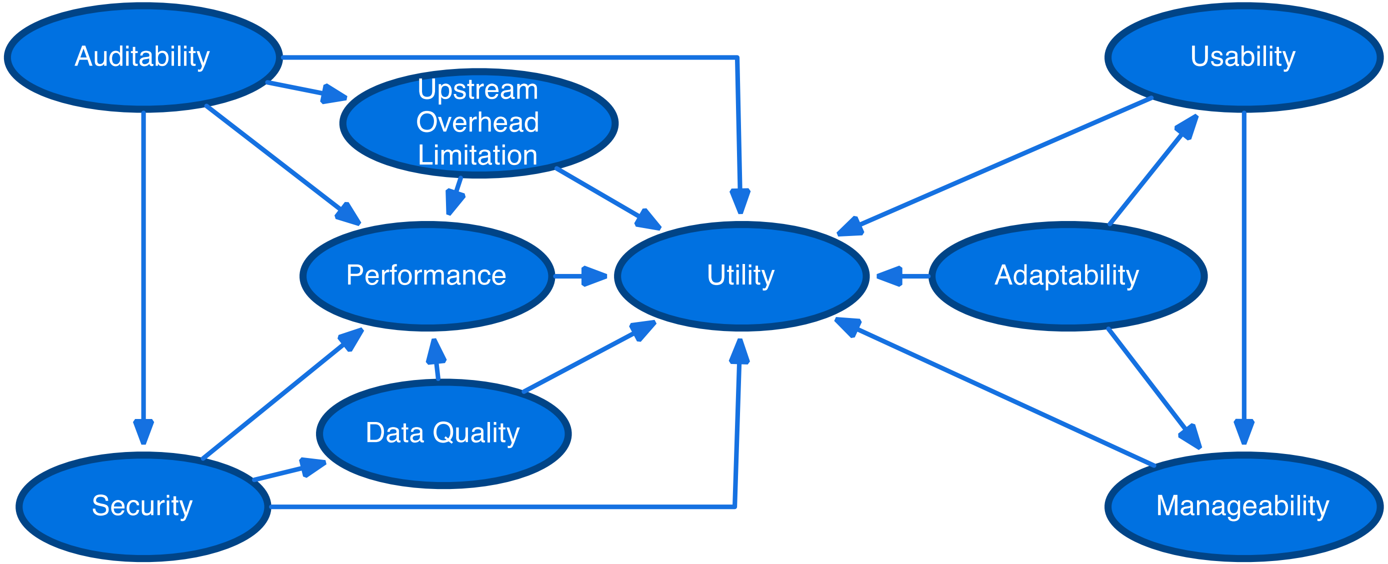
- Conceptual model reflecting user requirements
- User requirements-driven flow redesign
- Automatic “quality” pattern integration
- Configurable testing

## Challenges and Discussion

- Relate patterns to utility
- Assess pattern significance, model accuracy & completeness
- Future plan

# ETL Quality Attributes

Paper: *Quality Measures for ETL Processes (DaWaK ’14)*



## TRADE-OFFS

- It’s not only about performance!
- Improving some quality attributes can affect others positively or negatively

# ETL Quality Attributes

Paper: *Quality Measures for ETL Processes (DaWaK ’14)*

## Contribution

- Define a set of ETL **process** quality characteristics AND the relationships between them
- Provide quantitative measures for each characteristic, backed by literature!

## Methodology

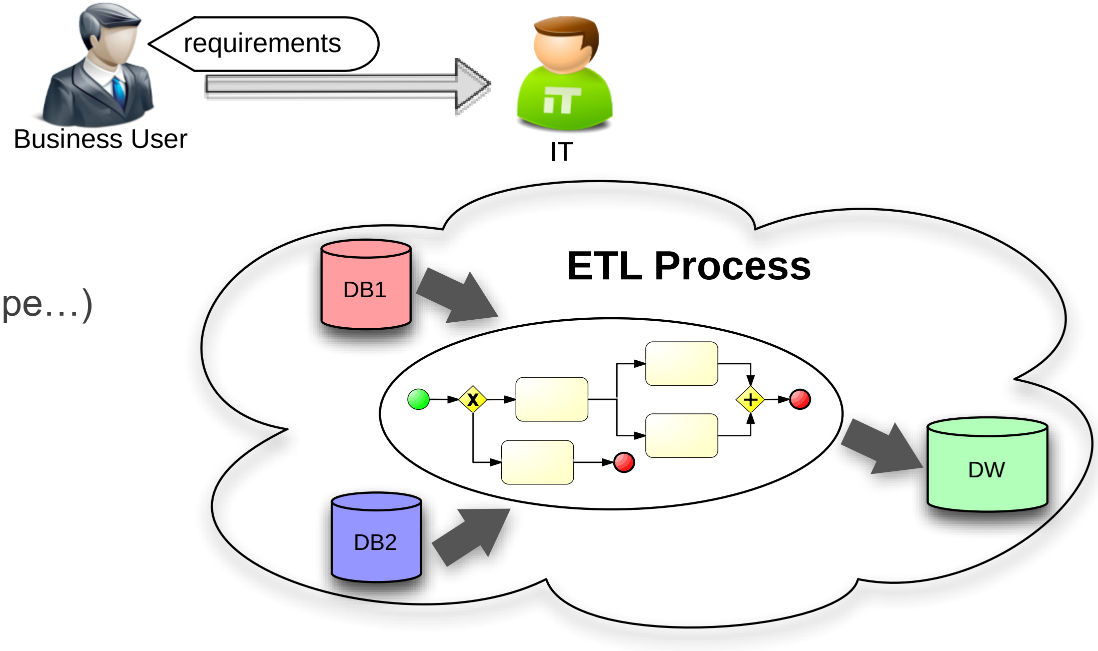
- SLR for quality attributes specific to data intensive processes
- Collection from literature of (proven) metrics for monitoring and quantitatively evaluating ETL processes

## Invited Journal Extension

- Special Issue of Journal CCPE 2015 (under minor revision)
- Introduce and apply goal modeling “stepping” on defined models
- Showcase evaluation of use case ETLs using proposed measures

# User requirements driving flow redesign

Paper: *A Framework for User-Centered Declarative ETL (DOLAP ’14)*



## Traditional Approach Problems

- Expensive process
- Hard to map requirements-implementation
- IT optimize only for performance
- Need more dynamicity (Big Data, data scope…)

## Inspiration

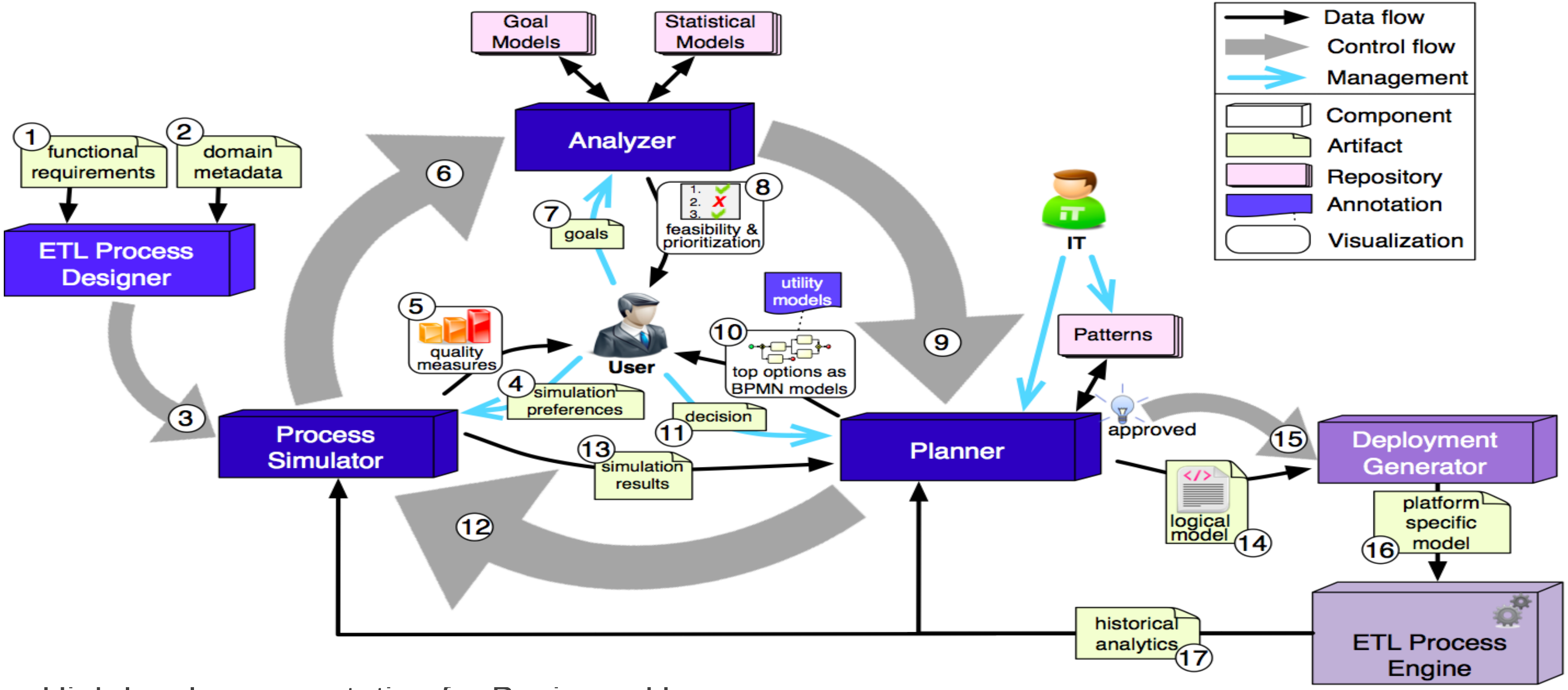
- Model-driven approach
- ETL process as a business process
- Agile BI, Self-service BI

## Approach

- User at the center of the iterative process
- Functional and non-functional requirements are analyzed at the same time using automatic Pattern management

# User requirements driving flow redesign

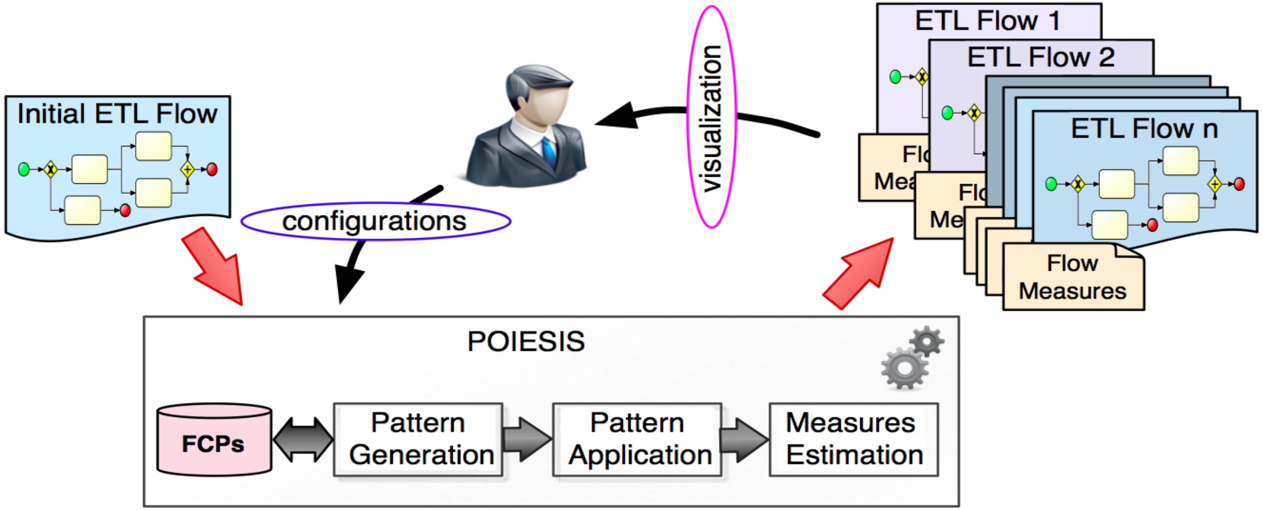
Paper: *A Framework for User-Centered Declarative ETL (DOLAP '14)*



- High level representation for Business Users
- Translation to low level models for IT and vice versa

# Automated Process Redesign (POIESIS)

*Demo Paper: POIESIS: a Tool for Quality-aware ETL Process Redesign (EDBT ’15)*

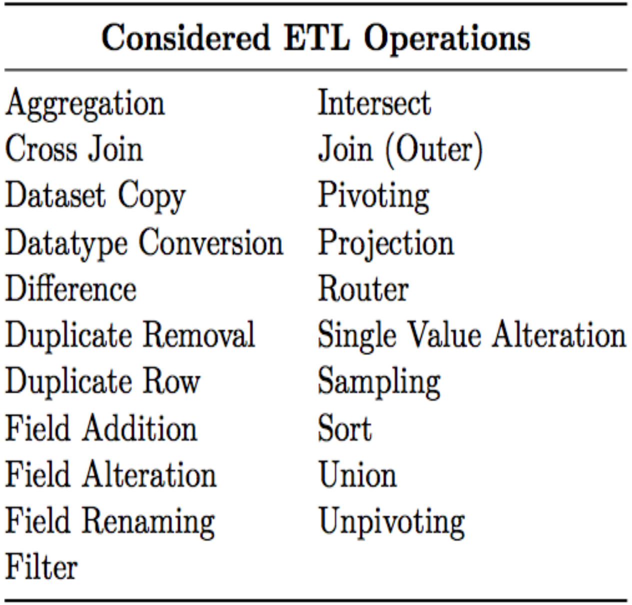


## AUTOMATIC GENERATION OF ALTERNATIVE PHYSICAL ETL FLOWS

- Alternative designs: Same functionality (constant data schemata), different flow components-permutations
- Policies and patterns
- Measures estimation for evaluation

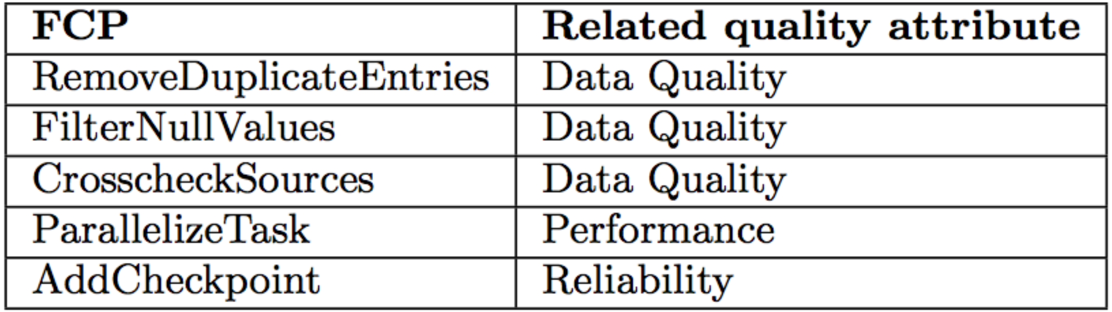
# Logical Modeling & FCPs

*Demo Paper: POIESIS: a Tool for Quality-aware ETL Process Redesign (EDBT ’15)*



**Considered ETL Operations**

Aggregation
Cross Join
Dataset Copy
Datatype Conversion
Difference
Duplicate Removal
Duplicate Row
Field Addition
Field Alteration
Field Renaming
Filter
Intersect
Join (Outer)
Pivoting
Projection
Router
Single Value Alteration
Sampling
Sort
Union
Unpivoting



| FCP | Related quality attribute |
|---|---|
| RemoveDuplicateEntries | Data Quality |
| FilterNullValues | Data Quality |
| CrosscheckSources | Data Quality |
| ParallelizeTask | Performance |
| AddCheckpoint | Reliability |

## Logical Modelling of ETL Flows

- Each operator is a node in a DAG structure
- Flow Component Patterns represented in the same logical model
- Each (combination of) pattern application(s) produces a new ETL flow

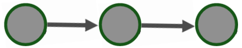
# Flow Component Patterns (FCPs)

*Demo Paper: POIESIS: a Tool for Quality-aware ETL Process Redesign (EDBT ’15)*

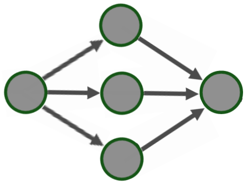
**Component Types**

*Atomic ETL Step*

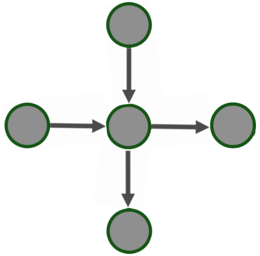
*Sequence Component*



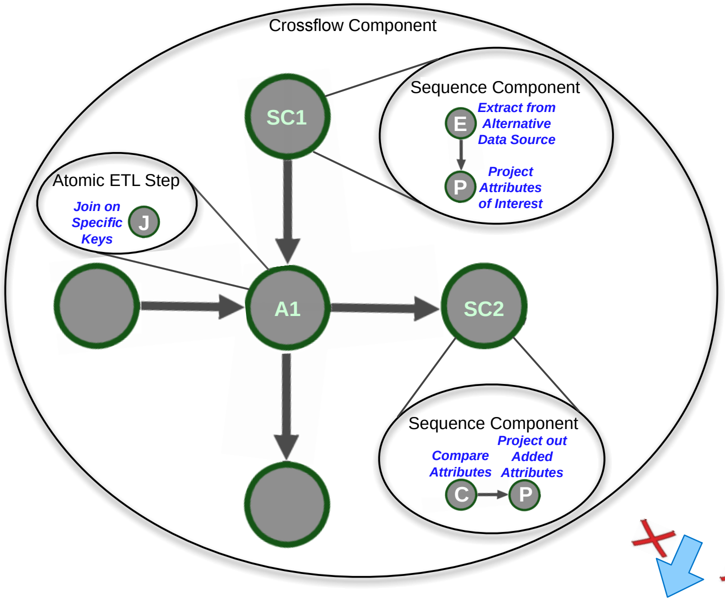
*Flow Component*



*Crossflow Component*



**FCP example: Crosscheck Data Sources**

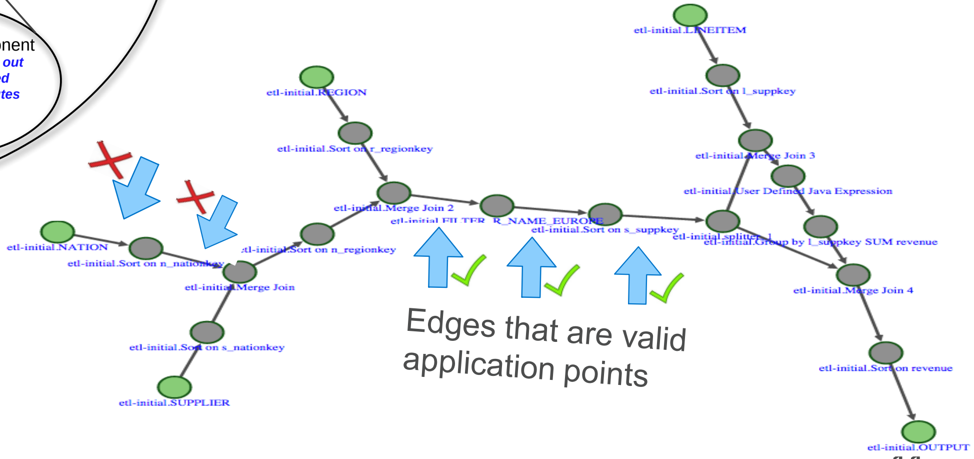


Application Point:

- Edge
- Node
- Complete Graph

Application Properties:

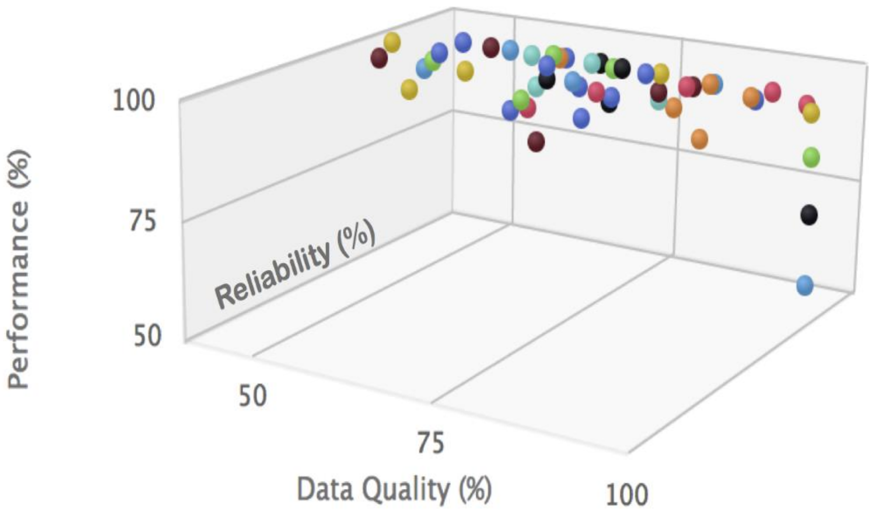
- *Applicability* based on rules →Pruning
- *Fitness* based on heuristics→Optimization

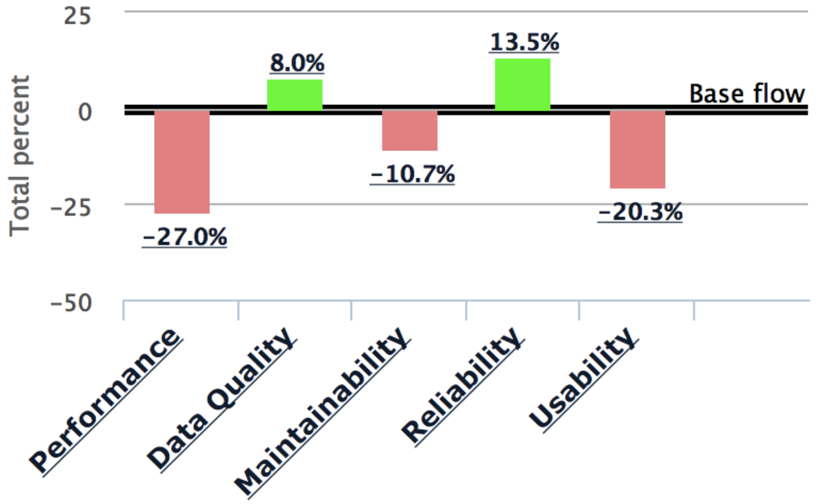

etl-initial.LINEITEM
etl-initial.Sort on l_suppkey
etl-initial.REGION
etl-initial.Sort on r_regionkey
etl-initial.Merge Join 3
etl-initial.User Defined Java Expression
etl-initial.Merge Join 2
etl-initial.FILTER_R_NAME_EUROPE
etl-initial.Sort on s_suppkey
etl-initial.splitter_1
etl-initial.Group by l_suppkey SUM revenue
etl-initial.NATION
etl-initial.Sort on n_nationkey
etl-initial.Sort on n_regionkey
etl-initial.Merge Join
etl-initial.Merge Join 4
etl-initial.Sort on s_nationkey
etl-initial.Sort on revenue
etl-initial.SUPPLIER
etl-initial.OUTPUT

Edges that are valid application points

# Example Visualization

*Demo Paper: POIESIS: a Tool for Quality-aware ETL Process Redesign (EDBT ’15)*





## Multidimensional Analysis

- Pareto frontier
- Each point represents an ETL flow
- Metrics (compound and detailed) compared to initial flow

# Quality-aware testing

Paper: *Bijoux: Data Generator for Evaluating ETL Process Quality (DOLAP ’14)*

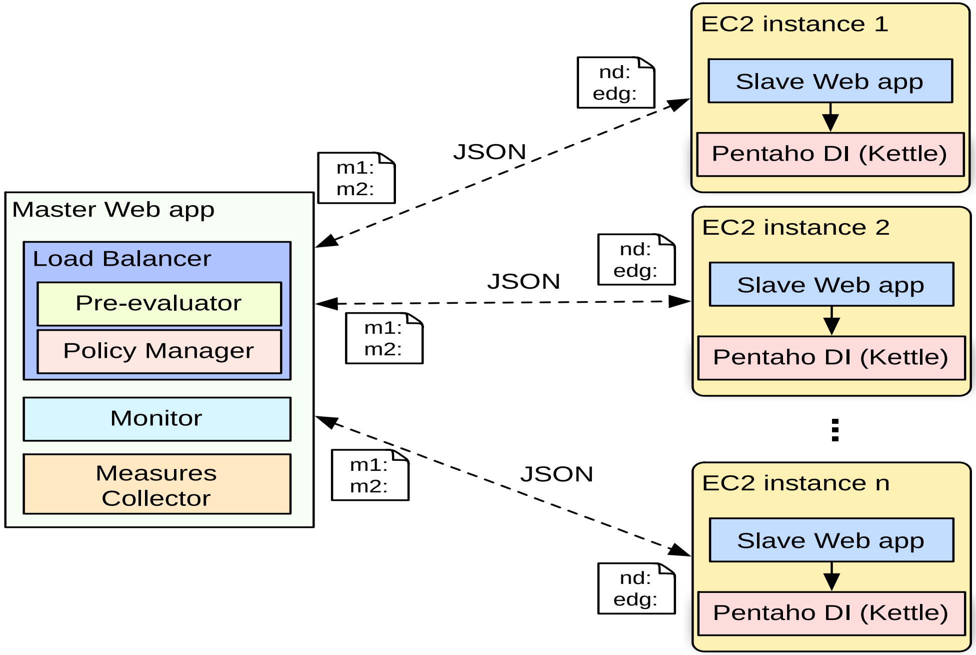
## APPROACH

- An automatic, semantic-aware framework for generating testing workloads for evaluating quality of ETL processes
- Using a taxonomy of ETL operations and their semantics, create synthetic datasets to test flows
- Configurable properties (e.g., selectivity, distribution) to emphasize specific flow parts characteristics

## INVITED JOURNAL EXTENSION

- Information Systems, Elsevier 2015 (under review)
- Highlight workflow perspective and analyze properties like flow coverage
- Propose architecture and showcase updated implementation that scales

# Execution on the Cloud



## Elasticity for responsiveness

- Hundreds of flows executed very fast
- Load balancing based on pre-evaluation

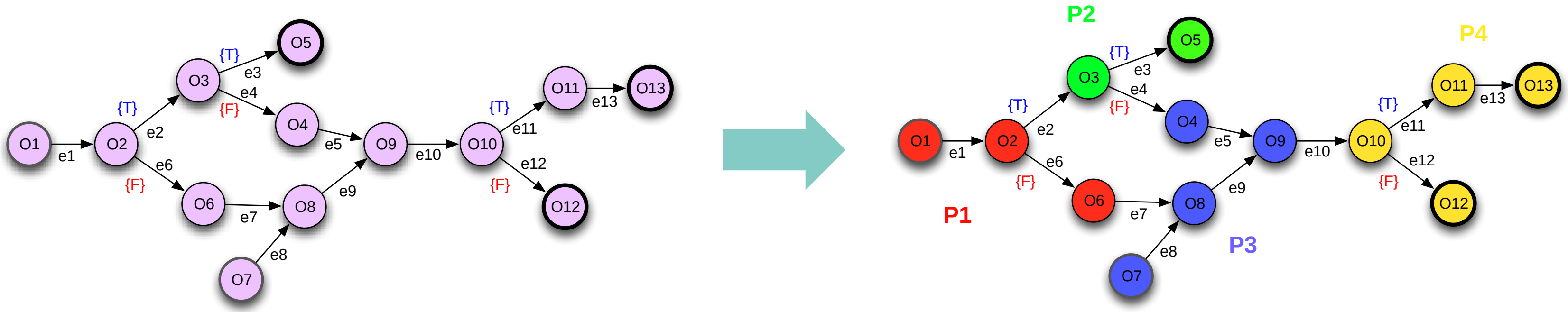
## Open research questions

- Do instances share state? Common input data?
- Can results be generalized for platform dependent executions?

# Decomposition to Structural Patterns

## Quality evaluation of ETL flows

- Different design choices → large number of alternative ETL flows
- Need for fine-grained cost models
- Repository of patterns to increase reusability of models



## Pattern-based decomposition of ETL flows

- Classify structural patterns & identify on each flow
- Derive utility as a function of the patterns that each flow contains
- Adaptive model: Knowledge Base enrichment ⟳ Flow evaluation improvement

# Challenges

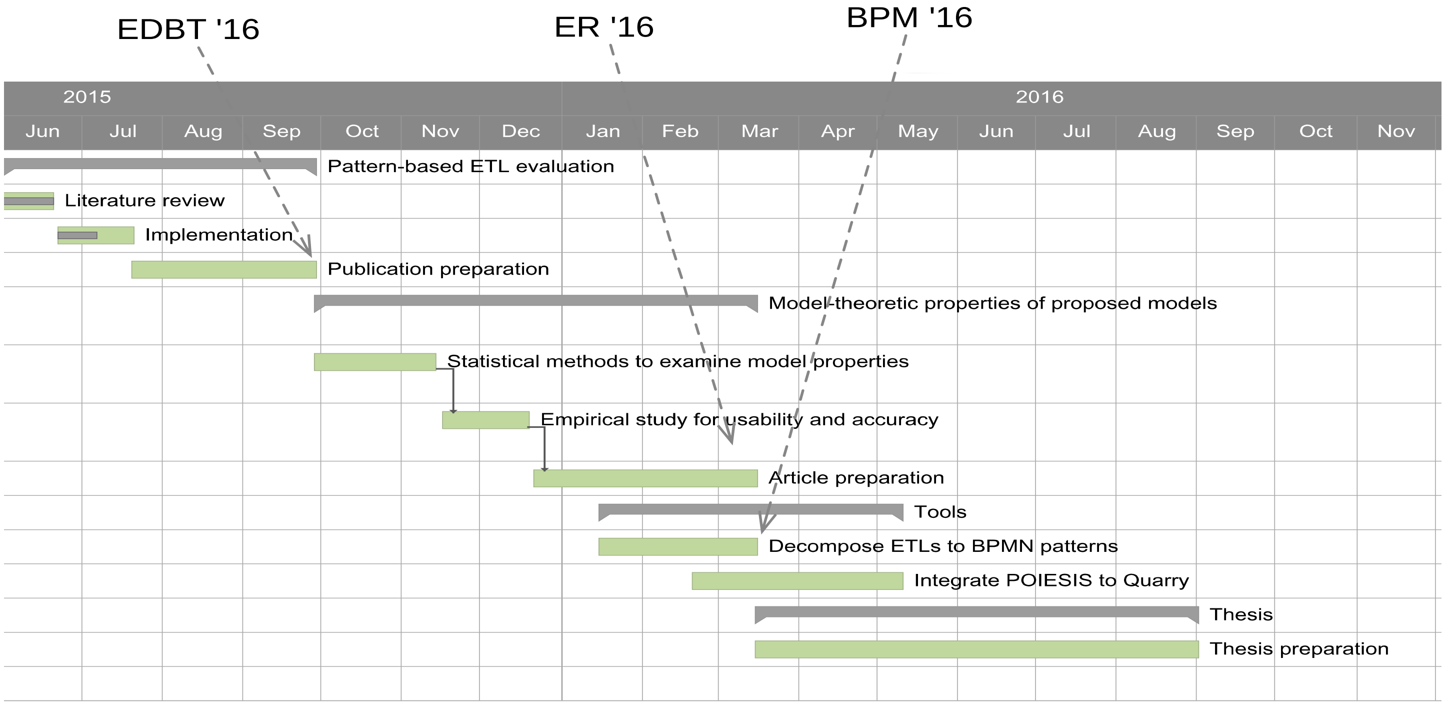
## Relate Structural Patterns to quality measures

- When and where is a quality pattern worth considering?
- Knowledge Base including pattern applications – detailed (measured) quality tradeoffs
- Also rules about pattern combinations

## Model-theoretic properties

- Accuracy, completeness
- How to evaluate significance of models?

# Future Plan



Pattern-based ETL evaluation

Literature review

Implementation

Publication preparation

Model-theoretic properties of proposed models

Statistical methods to examine model properties

Empirical study for usability and accuracy

Article preparation

Tools

Decompose ETLs to BPMN patterns

Integrate POIESIS to Quarry

Thesis

Thesis preparation

## JOURNALS

- DSS ’16: Using statistical methods to examine model-theoretic properties of ETL utility characteristics
- IJDWM ’16: ETL utility characteristics modelling and results from empirical study