ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Algorithms and Architecture for Managing Evolving ETL Workflows in a Big Data Environment

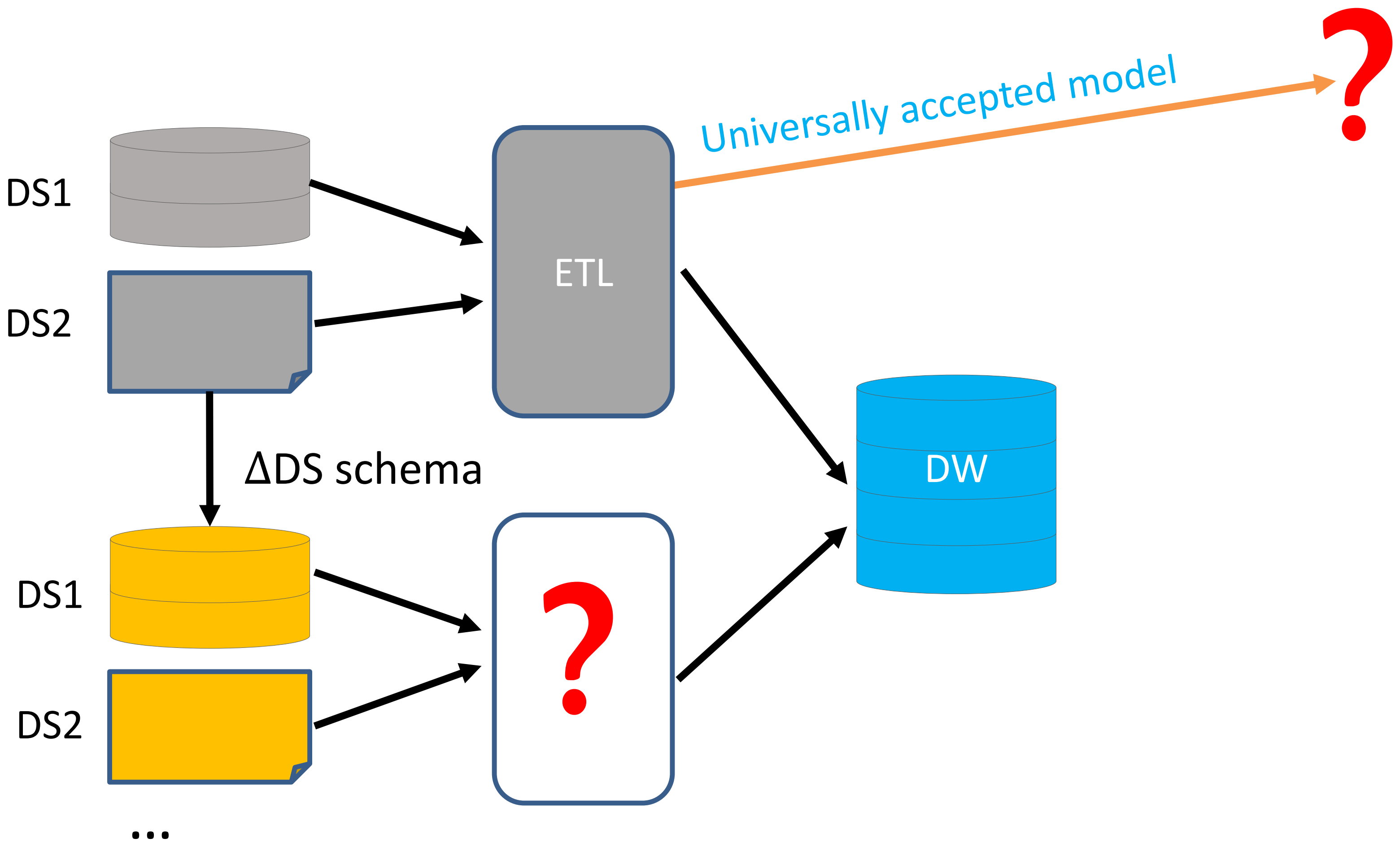
IT4BI DC

Judith Awiti

Esteban Zimányi (Home Supervisor) : Université Libre de Bruxelles

Robert Wrembel (Host Supervisor) : Poznań University of Technology

Problem Statement



Objectives

1. To propose a methodology for designing ETL processes that will facilitate a smooth transition from gathering user requirements to the actual implementation. This methodology will include all aspects of ETL design, from conceptual modelling to physical implementation
2. To develop a framework to (semi-) automatically repair ETL workflows upon data source changes

Currently focusing on relational data

ETL Modelling

Scenario: The part supplier DW project

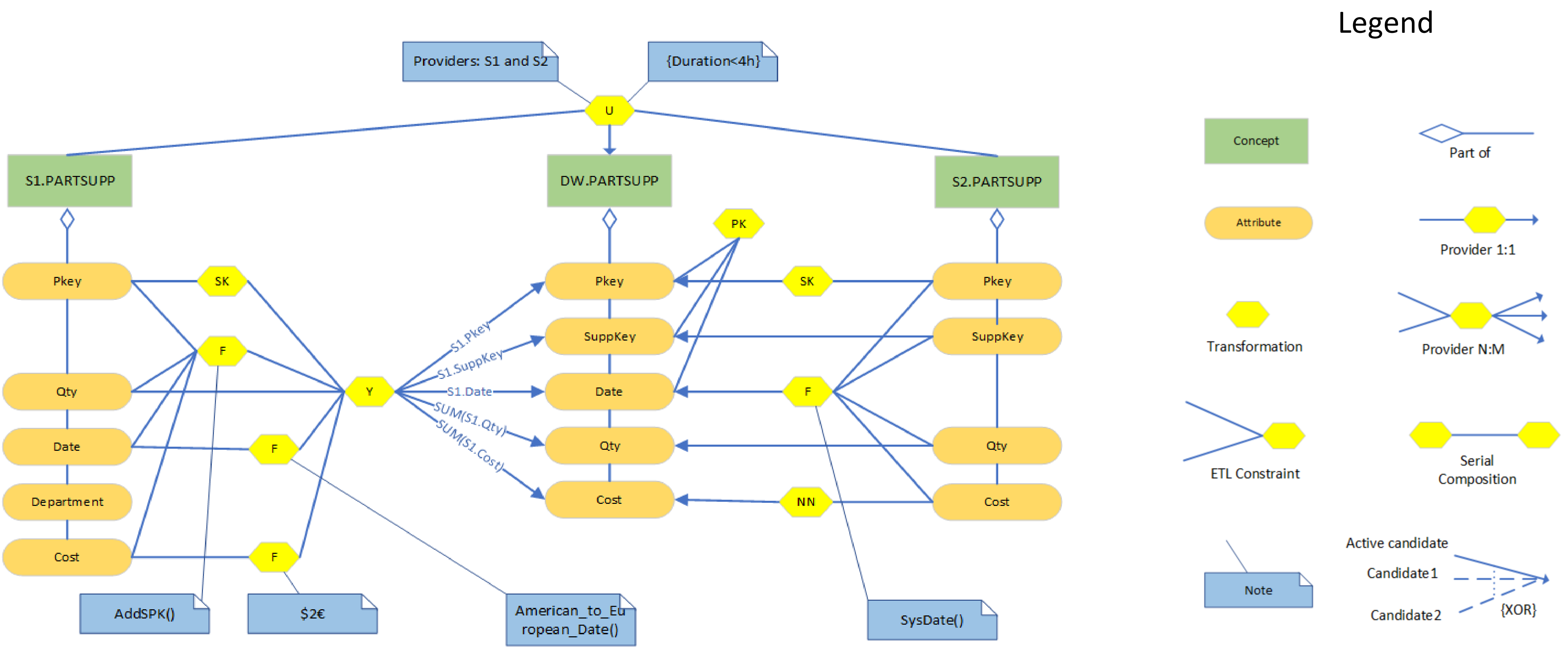
	ATTRIBUTES
S1.PARTSUPP	Pkey, Qty, Date, Department, Cost
S2.PARTSUPP	Pkey, SuppKey, Qty, Cost
DW.PARTSUPP	Pkey, SuppKey, Date, Qty, Cost

Transformations:

- Surrogate Key assignment
- Convert **Date** and **Cost** to European formats (**S1** is an American source whereas **S2** is an European source)
- Add **SuppKey** for **S1** (which is a constant value of 1 or 2)
- Aggregate sum of **Qty** and **Cost** in S1 (S1 stores department data, S2 ignores department detail)
- Not null check for **Cost** in **S2**
- Get System date for **S2**

ETL Modelling (Current Approaches)

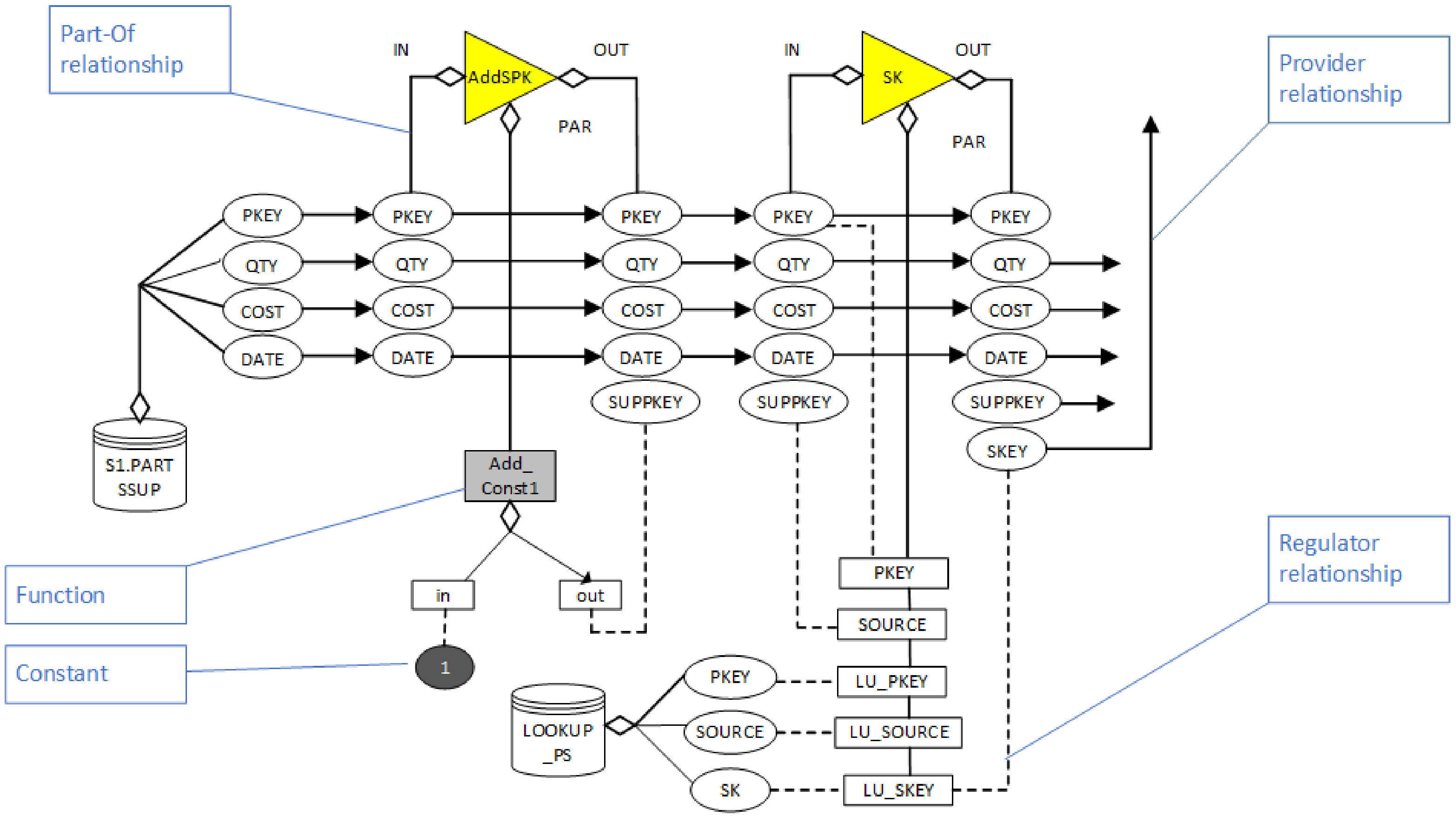
Graph: Conceptual modelling



- Vassiliadis, P., Simitsis, A., Skiadopoulos, S.: Conceptual modeling for ETL processes. In: Proc. of the 5th *ACM International workshop on Data Warehousing and OLAP*, DOLAP 2002. pp. 14–21. ACM, McLean, Virginia, USA (2002)

ETL Modelling (Current Approaches)

Graph: Logical modelling (Architecture Graph)
Part of the scenario for S1.PARTSUPP:
Add Supplier Key (AddSPK), and Surrogate key (SK)



- Vassiliadis, P., Simitsis, A., Skiadopoulos, S.: Modeling ETL activities as graphs. In: Proc. of the 4th *International Workshop on Design and Management of Data Warehouses*, DMDW 2002. pp. 52–61. CEUR-WS.org, Toronto, Canada (2002)

ETL Modelling (Current Approaches)

Graph

Pros

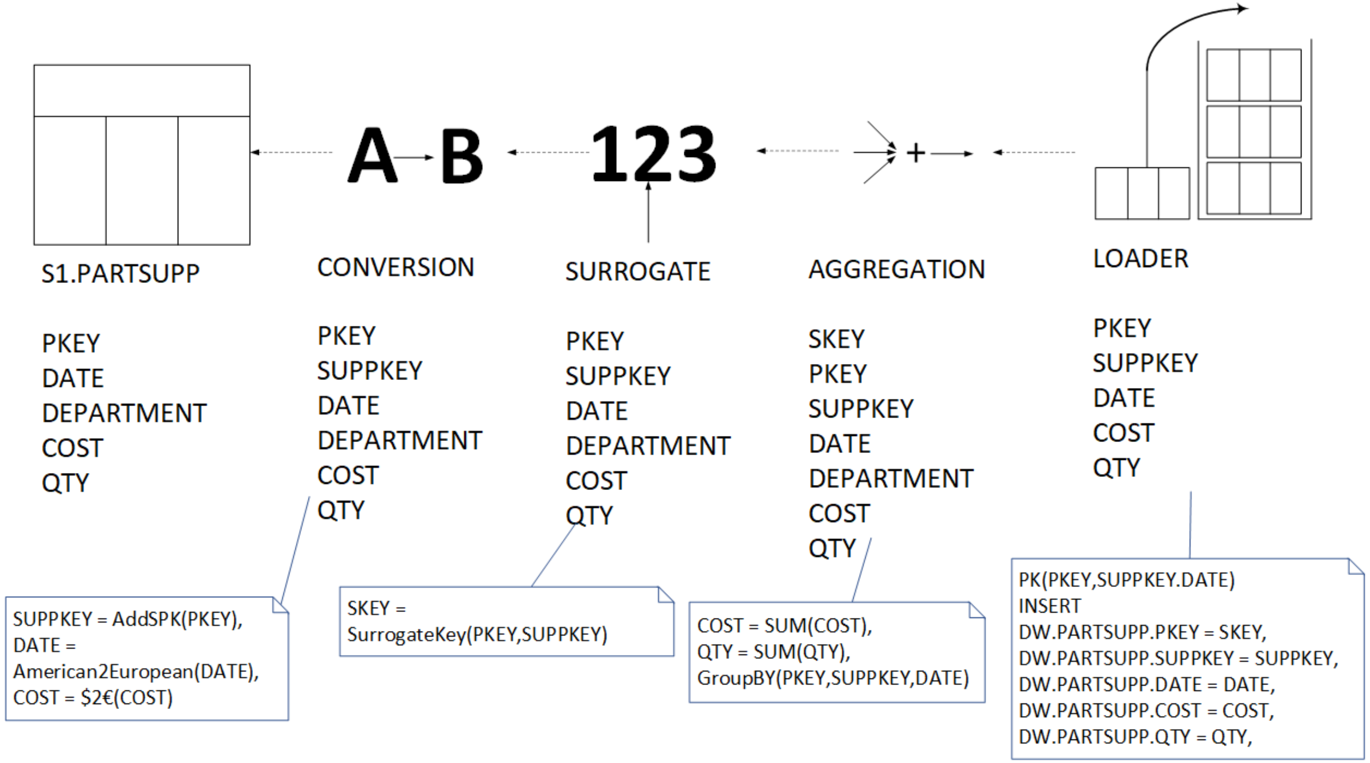
- Provides first steps toward translating a conceptual to a corresponding logical model
- Graph-based models are used (standard and widely accepted)

Cons

- Complex conceptual and logical Design
- Difficult to transform from conceptual model to logical model. Require excessive training even for technical experts to model complex scenarios
- Architecture graph is only implemented using a custom-built tool (ARKTOS)
- Not suitable for ETL workflow reparation due to complexity

ETL Modelling (Current Approaches)

UML: Conceptual modelling



ETL Mechanism (Stereotype)	Description	Icon
Aggregation	Aggregates data based on some criteria	+
Conversion	Changes data type and format or derives new data from existing data	A→B
Filter	Filters and verifies data	
Incorrect	Reroutes incorrect data	!!!
Join	Joins two data sources related to each other with some attributes	
Loader	Loads data into the target of an ETL process	
Log	Logs activity of an ETL mechanism	LOG
Merge	Integrates two or more data sources with compatible attributes	
Surrogate	Generates unique surrogate keys	123→
Wrapper	Transforms a native data source into a record based data source	

Trujillo, J., Luján-Mora, S.: A UML based approach for modeling ETL processes in data warehouses. In: Proc. of the 22 nd International Conference on Conceptual Modeling, ER 2003. pp. 307–320. Springer, Chicago, Illinois, USA (2003)

ETL Modelling (Current Approaches)

UML

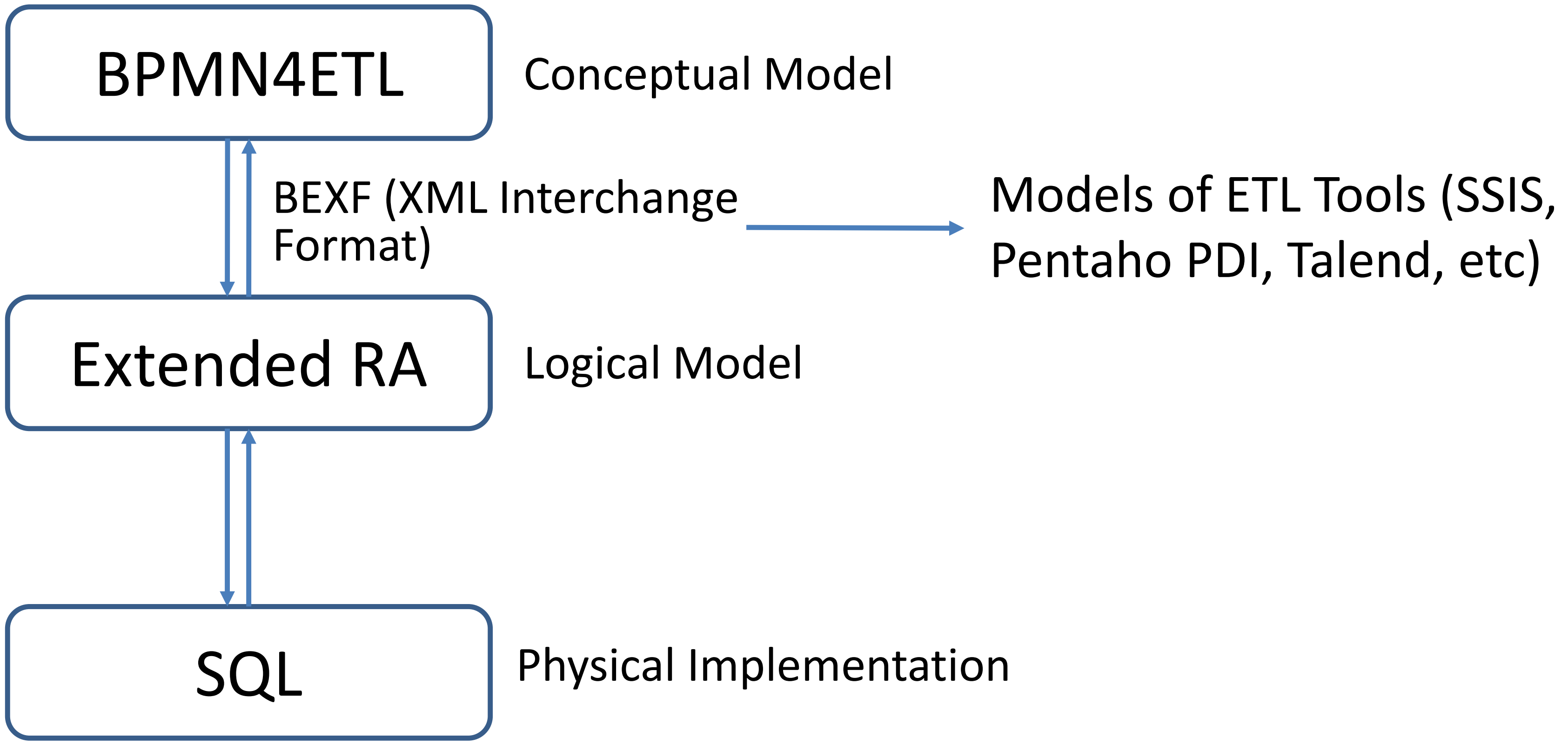
Pros

- UML is a standard modelling language
- ETL activities (e.g., aggregations, conversions, etc) can be plugged in easily
- Less complex because user is not overwhelmed with inter-attribute mappings

Cons

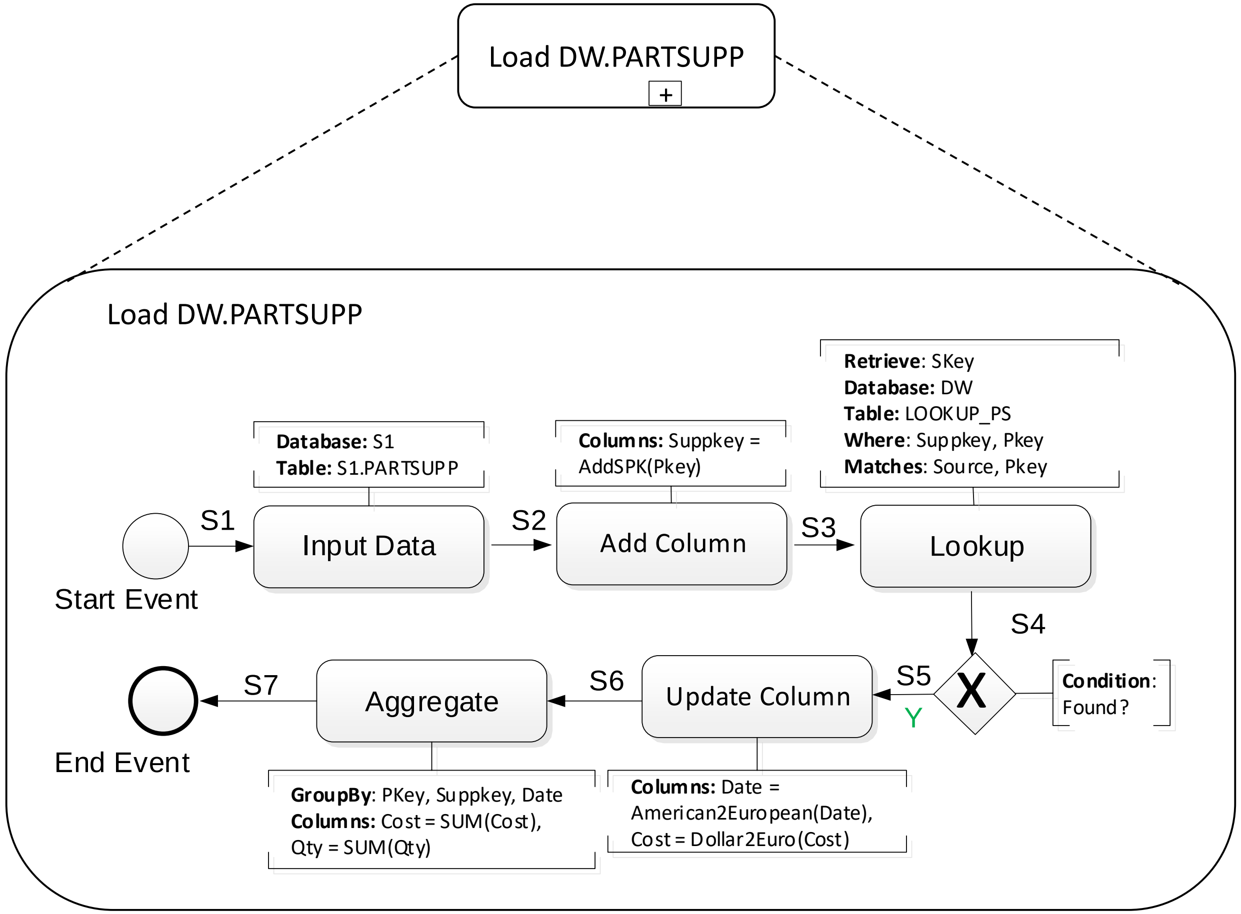
- No corresponding logical or implementation model
- Knowledge of UML needed
- Not suitable for showing the control flow or the run-time communication structure of these processes

ETL Modelling (Our approach)



ETL Modelling (Our Approach)

BPMN4ETL: Conceptual modelling



Pros

- BPMN is a standard modelling language
- Models both control and data flow
- ETL activities (e.g., aggregations, conversions, etc) can be plugged in easily
- Less complex because user is not overwhelmed with inter-attribute mappings
- Easy communication and validation between an operational database designer, an ETL designer and a business intelligence analyst
- Exposes the manipulation of data and their order from one ETL task to the other
- Can be used for documentation
- Can be translated directly to relational algebra, SQL, or an XML interchange format

Cons

- Knowledge of BPMN needed

ETL Modelling (Our Approach)

Extended Relational Algebra

Operator	Notation
Selection	$\sigma_C(R)$
Projection	$\pi_{A_1,\ldots,A_n}(R)$
Cartesian Product	$R_1 \times R_2$
Union	$R_1 \cup R_2$
Intersection	$R_1 \cap R_2$
Difference	$R_1 - R_2$
Join	$R_1 \bowtie_C R_2$
Natural Join	$R_1 * R_2$
Left Outer Join	$R_1 ⟕_C R_2$
Right Outer Join	$R_1 ⟖_C R_2$
Full Outer Join	$R_1 ⟗_C R_2$
Semijoin	$R_1 \ltimes_C R_2$
Division	$R_1 \div R_2$

Operator	Notation
Aggregate	$\mathcal{A}_{A_1,\ldots,A_m \mid C_1=\mathsf{F}_1(B_1),\ldots,C_n=\mathsf{F}_n(B_n)}(R)$
Delete	$R \leftarrow R - \sigma_C(R)$
Extend	$\mathcal{E}_{A_1=\mathsf{Expr}_1,\ldots,A_n=\mathsf{Expr}_n}(R)$
Input	$R \leftarrow \mathcal{I}_{A_1,\ldots,A_n}(F)$
Insert	$R \leftarrow R \cup S$ or $R \leftarrow S$
Lookup	$R \leftarrow \pi_{A_1,\ldots A_n}(R_1 \bowtie_C R_2)$
Remove duplicates	$\delta(R)$
Rename	$\rho_{A_1 \leftarrow B_1,\ldots,A_n \leftarrow B_n}(R)$ or $\rho_S(R)$
Sort	$\tau_A(R)$
Update	$\mathcal{U}_{A_1=\mathsf{Expr}_1,\ldots,A_n=\mathsf{Expr}_n \mid C}(R)$
Update Set	$R \leftarrow \mathcal{U}(R)_{A_1=\mathsf{Expr}_1,\ldots,A_n=\mathsf{Expr}_n \mid C}(S)$

ETL Modelling (Our Approach)

ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Relational Algebra: Logical modelling

$$\text{Temp1} \leftarrow \mathcal{I}_{\text{Pkey, Qty, Department, Cost}}(\text{SI.PARTSUPP}) \quad (1)$$

$$\text{Temp2} \leftarrow \mathcal{E}_{\text{Suppkey} = \text{AddSPK(Pkey)}}(\text{Temp1}) \quad (2)$$

$$\text{Temp3} \leftarrow \pi_{\text{Skey, Pkey, Suppkey, Qty, Department, Cost}}(\text{Temp2} \bowtie_{\text{Pkey} = \text{Pkey} \wedge \text{Suppkey} = \text{Source}} \text{LOOKUP\_PS}) \quad (3)$$

$$\text{Temp4} \leftarrow \mathcal{U}_{\text{Date} = \text{American2European(Date)}, \text{Cost} = \text{Dollar2Euro(Cost)}}(\text{Temp3}) \quad (4)$$

$$\text{Temp5} \leftarrow \mathcal{A}_{\text{Pkey, Suppkey, Date} | \text{Cost} = \text{SUM(Cost)}, \text{Qty} = \text{SUM(Qty)}}(\text{Temp4}) \quad (5)$$

Pros

- RA provides a set of operators that manipulates relations to ensure that there is no ambiguity
- Can also be directly translated into SQL to be executed in any Relational Database Management System (RDBMS). We avoid dealing with the peculiarities of a particular programming language
- When extended with update operations, the can provide a logical model of different ETL scenarios. E.g. Slowly changing dimension with dependencies found in the TPC-DI Benchmark

Limitations

Difficult to model certain complex tasks in relational algebra even though they can be done directly with SQLs. (E.g. window functions and loops)

ETL Modelling (Experiments)

Experimental Evaluation

Experiments implemented in two ways:

1. Using Pentaho PDI , translating the BPMN4ETL directly into Pentaho PDI
2. Using RA , translating BPMN4ETL into extended RA, and then implementing the RA operations using Postgres PLSQL.

TPC-DI Benchmark

- Data sources are of different formats (xml, csv, txt, and so on)
- Source data model: Based on a fictitious retail brokerage firm and external sources
- Target data model: Has a snowstorm schema
- One historical load and two identical incremental loads
- Scale factor (number of records) - 3 (4.5 million), 5 (7.8 million), 10 (16.1 million)

Platform: Intel i7 computer, with a RAM of 16 GB, running the Windows 10 Enterprise operating system, using the Postgres SQL database as the DW storage

ETL Modelling (Experiments)

ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Performance:

Execution times to complete TPC-DI benchmark Load

Time = hours:minutes:seconds

		Historical	Incremental 1	Incremental 2
SF-3	PLSQL	00:12:50	00:00:09	00:00:07
	PDI	11:23:52	00:01:32	00:01:40
SF-5	PLSQL	00:22:31	00:00:15	00:00:14
	PDI	20:25:32	00:03:03	00:03:11
SF-10	PLSQL	02:11:15	00:00:39	00:00:36
	PDI	25:08:13	00:11:35	00:12:38

Pentaho PDI Optimization

- PDI memory limit was increased from 2G to 4G
- PDI performance tuning tips were applied [2]
    - [2] https://help.pentaho.com/Documentation/7.1/0P0/100/040/010

ETL Modelling (XML Interchange format)

UNIVERSITÉ LIBRE DE BRUXELLES

BPMN4ETL eXchange format (BEXF)

Load of DW.PARTSUPP Dimension

```xml
<ETLProcess id="_idProcess" name="Load of DW.PARTSUPP dimension table">
  <ETLTask id="_idInputData" name="Input Data" type="Input Data">
    <Database name="S1"/>
    <Table name="S1.PARTSUPP"/>
    <inputs>
    <inputColumn name="Pkey"/>
    <inputColumn name="Qty"/>
    <inputColumn name="Date"/>
    <inputColumn name="Department"/>
    <inputColumn name="Cost"/>
    </inputs>
    <inRefId>_idS1</inRefId>
    <outRefId>_idS2</outRefId>
  </ETLTask>
  ...
  <ETLTask id="_idAggregate" name="Aggregate" type="Aggregate">
    <AggColumn name="Pkey" order="1"/>
    <AggColumn name="Suppkey" order="2"/>
    <AggColumn name="Date" order="3"/>
    <NewColumn name="Cost" function="SUM(Cost)"/>
    <NewColumn name="Qty" function="SUM(Qty)"/>
    <inRefId>_idS6</inRefId>
    <outRefId>_idS7</outRefId>
  </ETLTask>
</ETLProcess>
```

ETL Evolution (Current Approaches)

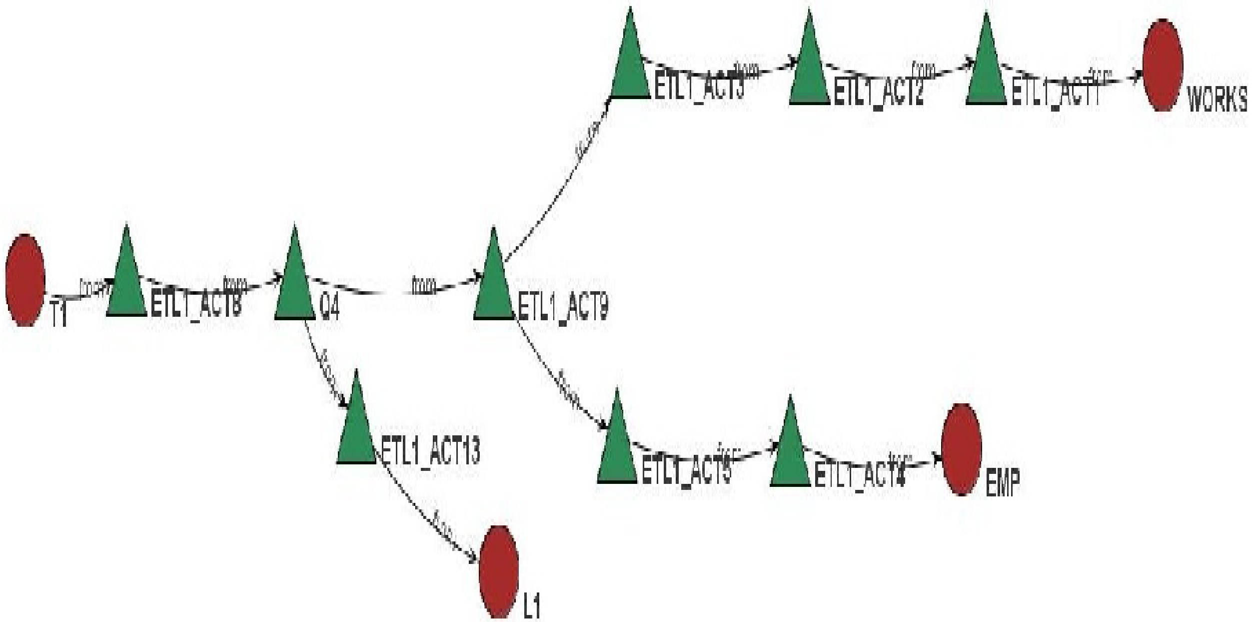
- *Hecataeus* – based on rules/policies
  - Papastefanatos, G., Vassiliadis, P., Simitsis, A., & Vassiliou, Y.: Policy-regulated management of ETL evolution. In Journal on Data Semantics XIII, 2009
- *E-ETL* – based on case-based reasoning
  - Wojciechowski, A.: ETL workflow reparation by means of case-based reasoning. Information Systems Frontiers 20(1): 21-43, 2018

ETL Evolution (Current Approaches)

UNIVERSITÉ LIBRE DE BRUXELLES

HECATAEUS

- Abstract ETL activities as queries and sequence of views
- Transforms SQL queries to graph
- User annotate graph with rules/policies (Propagate, Block, Prompt)
- System detects parts of the graph affected by a change in data source and highlights the way they respond to it



ETL Evolution (Current Approaches)

ULB

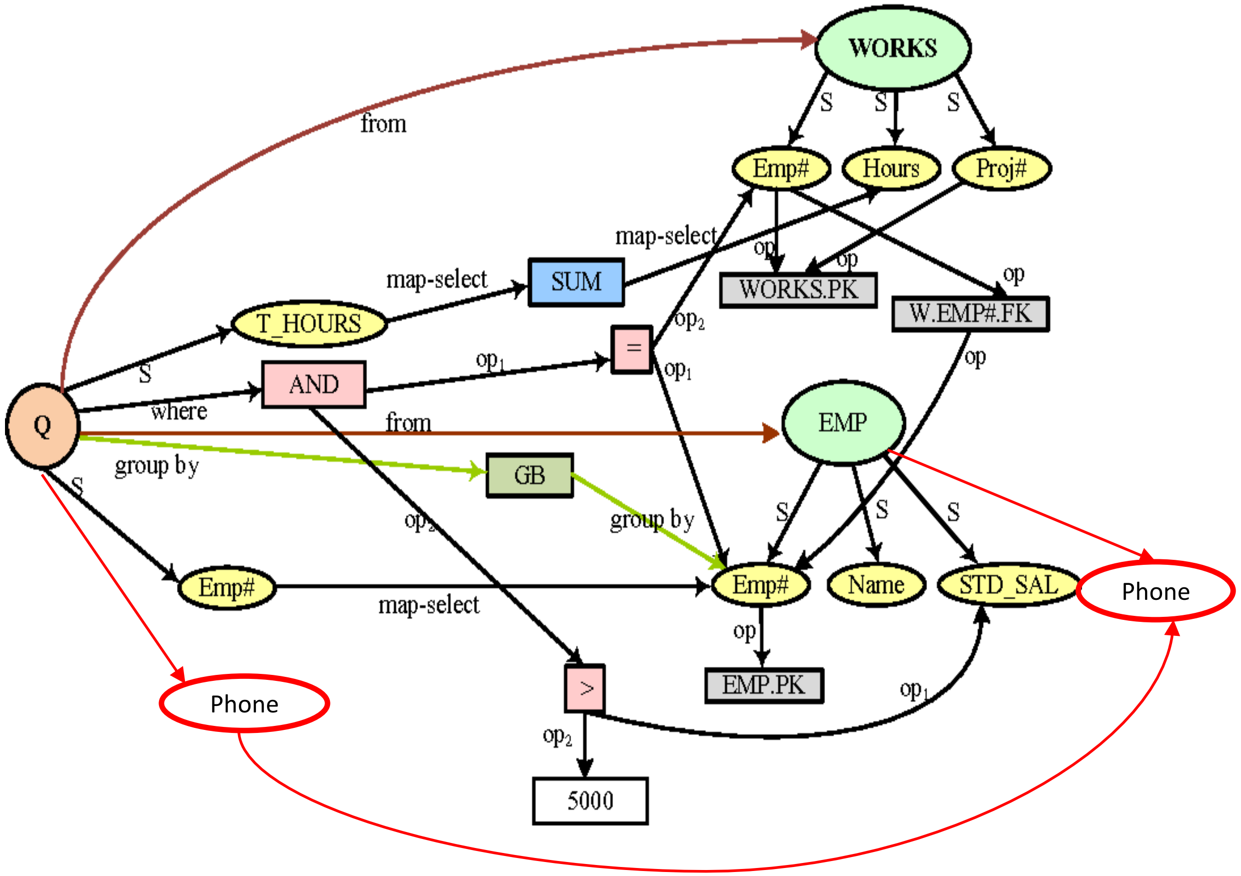
UNIVERSITÉ LIBRE DE BRUXELLES

HECATAEUS

DS change = Add *Phone* to *EMP*

Policy = *Propagate*

Detailed graph representation of ETL1_ACT9



Q: SELECT EMP.Emp# as Emp#, Sum(WORKS.Hours) as T_Hours FROM EMP, WORKS WHERE EMP.Emp# = WORKS.Emp# AND EMP.STD_SAL >5000 GROUP BY EMP.Emp#

ETL Evolution (Current Approaches)

Concerns with Hecataeus

- Near manual – policies must be explicitly stated for each node
- User must determine policy in advance before evolution event occurs
- No explanation on how graph is constructed or how the evolution event discovered

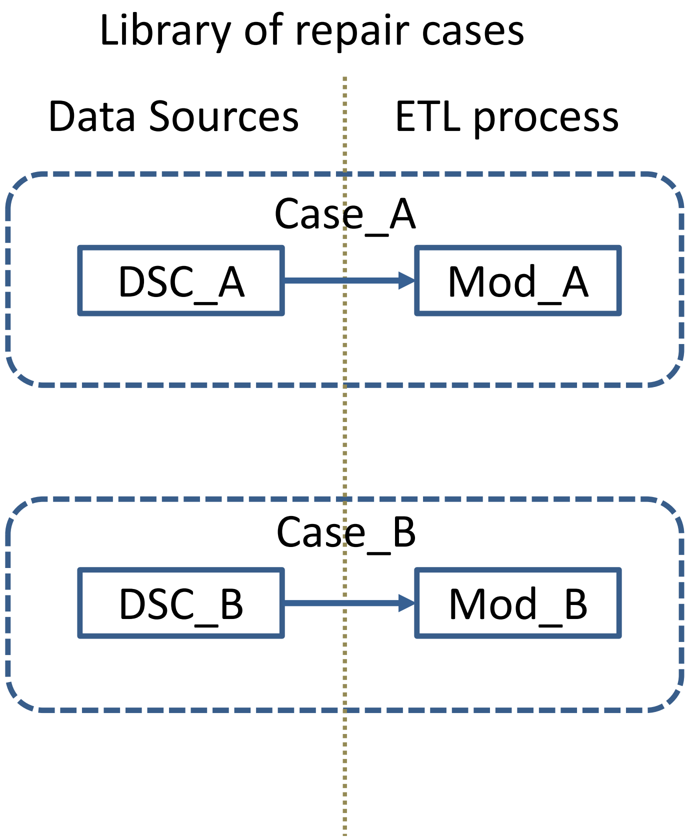
ETL Evolution (Current Approaches)

E-ETL

- Applies case-based reasoning
- Keeps *library of repair cases (LRC)* as knowledge base

Concerns with E-ETL

- Developers cannot guarantee correctness
- It needs a case base in advance to work



ETL Evolution (Our approach)

ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Subgoals:

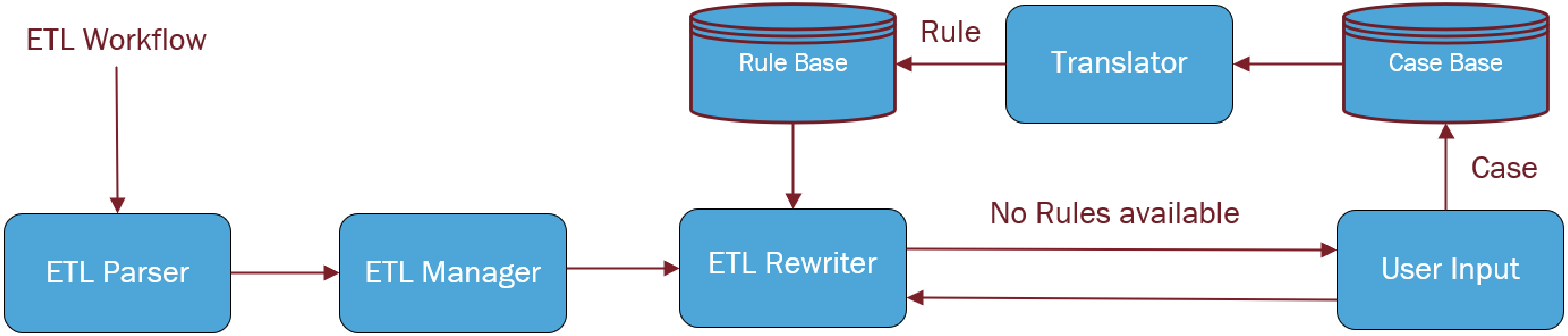
- Develop algorithms for (semi-)automatic repair of ETL workflows upon DS changes
  - Rules may be inferred from cases
  - Cases may be built from applying rules
  - Rule based + Case based (a quality measure for RB and CB)
- Develop an architecture for handling ETL evolution
- Implement a prototype
- Verify the applicability of the proposed solution with the TPC-DI benchmark
  - Poess, M., Rabl, T., Jacobsen, H. A., & Caufield, B.:. TPC-DI: the first industry benchmark for data integration. *Proceedings of the VLDB Endowment*, *7*(13), 2014

ETL Evolution (Our approach)

Extended Evolving ETL (E3TL) framework

RBR + CBR

- ETL workflows are rewritten by applying rules
- Rules are inferred from cases (By applying algorithms)
- Cases are built from user input



ETL Evolution (Our approach)

ULB

UNIVERSITÉ LIBRE DE BRUXELLES

Extended Evolving ETL (E3TL) framework – Learns rules from user input

Components:

ETL Parser: The ETL parser takes an entire ETL workflow in the form of RA or SQLs and parses the parts of each command of the workflow

ETL Manager: The ETL manager assesses the impact of the data source change on each command of the ETL workflow and takes these decisions by applying rules stored in a the rule base

ETL Rewriter: This component of the framework rewrites the commands in the ETL workflow by applying recommendations from the ETL manager

Rule Base: This contains distinct rules based on conditions

User Input: This part of the framework request the user's input if any of the following conditions is true:

- no rule is available in the rule base to deal with the problem
- several solutions are applicable to solve the problem

Case Base: This is a repository to store cases

Translator: This component applies algorithms to develop distinct rules from cases

Publication Strategy

ULB

UNIVERSITÉ LIBRE DE BRUXELLES

- Conference paper: **From Conceptual to Logical ETL Design using BPMN and Relational Algebra**
  Authors: Judith Awiti, Alejandro Vaisman, Esteban Zimányi
  Target venue: DAWAK 2019 - **Accepted**
- Workshop paper: **An XML Interchange Format for ETL Models**
  Authors: Judith Awiti, Esteban Zimányi
  Target venue: ADBIS 2019 - **Accepted**
- PhD Consortium paper: Algorithms and Architecture for Managing Evolving ETL Workflows
  Authors: Judith Awiti
  Target venue: ADBIS 2019 - **Accepted**
- Conference paper: **Rule-based management of evolving ETL workflows**
  Authors: Judith Awiti, Esteban Zimányi, Robert Wrembel
  Target venue: DOLAP 2020
  Deadline: November 2019
- Journal paper: **Management of evolving ETL workflows with a combination of RBR and CBR**
  Authors: Judith Awiti, Esteban Zimányi, Robert Wrembel
  Target Journal: ACM Transactions on information systems
  Deadline: March 2020
- Journal paper**: (E3TL) on (semi-)automatic repairs of ETL workflows**
  Authors: Judith Awiti, Esteban Zimányi, Robert Wrembel
  Target venue: Journal of the Association for Information Systems
  Deadline: To be decided
- Journal paper: **A review of methods of ETL workflow reparation**
  Authors: Judith Awiti, Esteban Zimányi, Robert Wrembel
  Target venue: Information Systems Management
  Deadline: To be decided