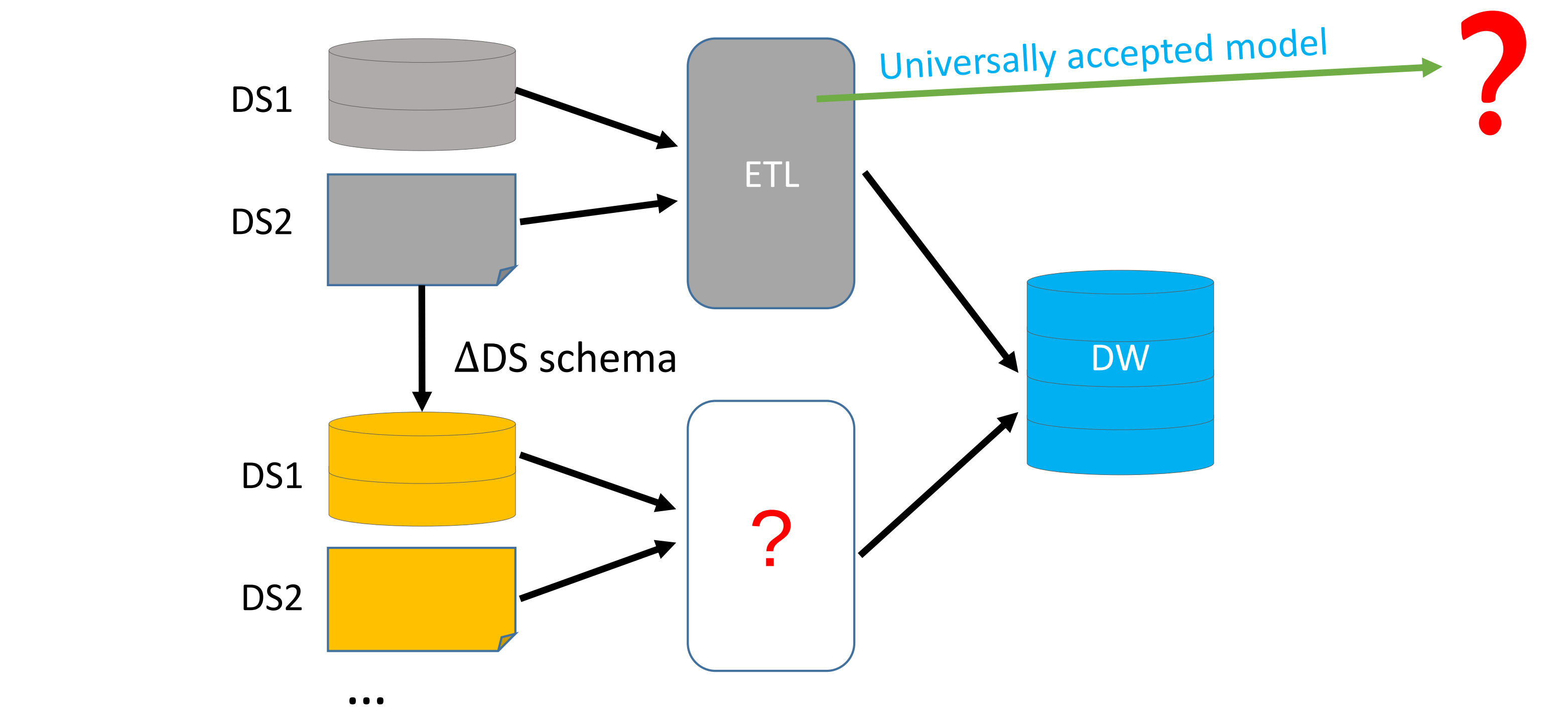
Algorithms and Architecture for Managing Evolving ETL Workflows in a Big Data Environment

Judith Awiti, Esteban Zimanyi, Robert Wrembel

Problem



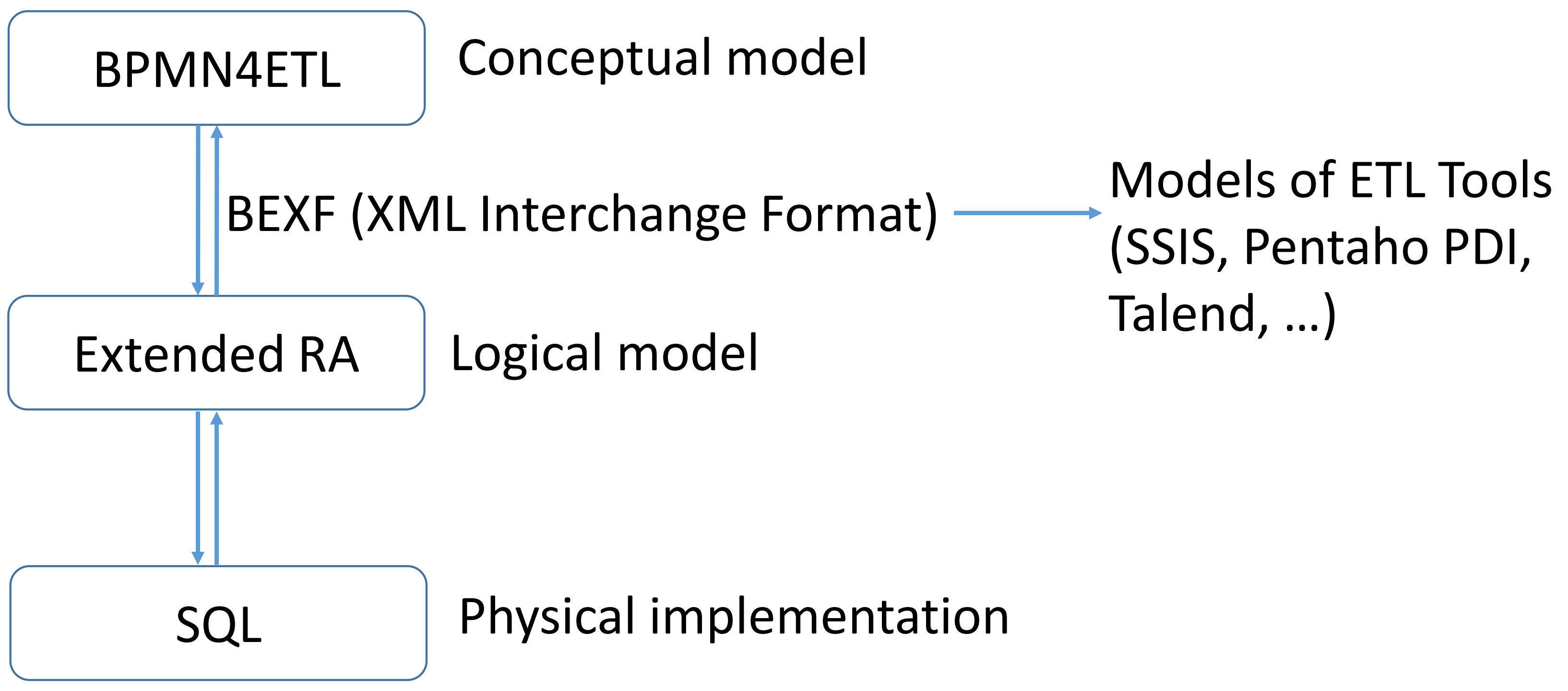
- No agreed-upon model for specifying ETL processes .
- Expensive to manually repair ETL workflows (time, expertise, money). Existing ETL tools tacitly assume DSs have static structure – not true (Wikipedia had 171 schema versions from Apr 2003 and Nov 2007 [1]).

Objectives

1. To propose a methodology for designing ETL processes that will facilitate a smooth transition from gathering user requirements to the actual implementation.
2. To develop an Extended Evolving ETL (E3TL) framework to (semi-) automatically repair ETL workflows upon data source changes.

Our approach

1. ETL Modelling



BPMN4ETL — Conceptual model

BEXF (XML Interchange Format) → Models of ETL Tools (SSIS, Pentaho PDI, Talend, …)

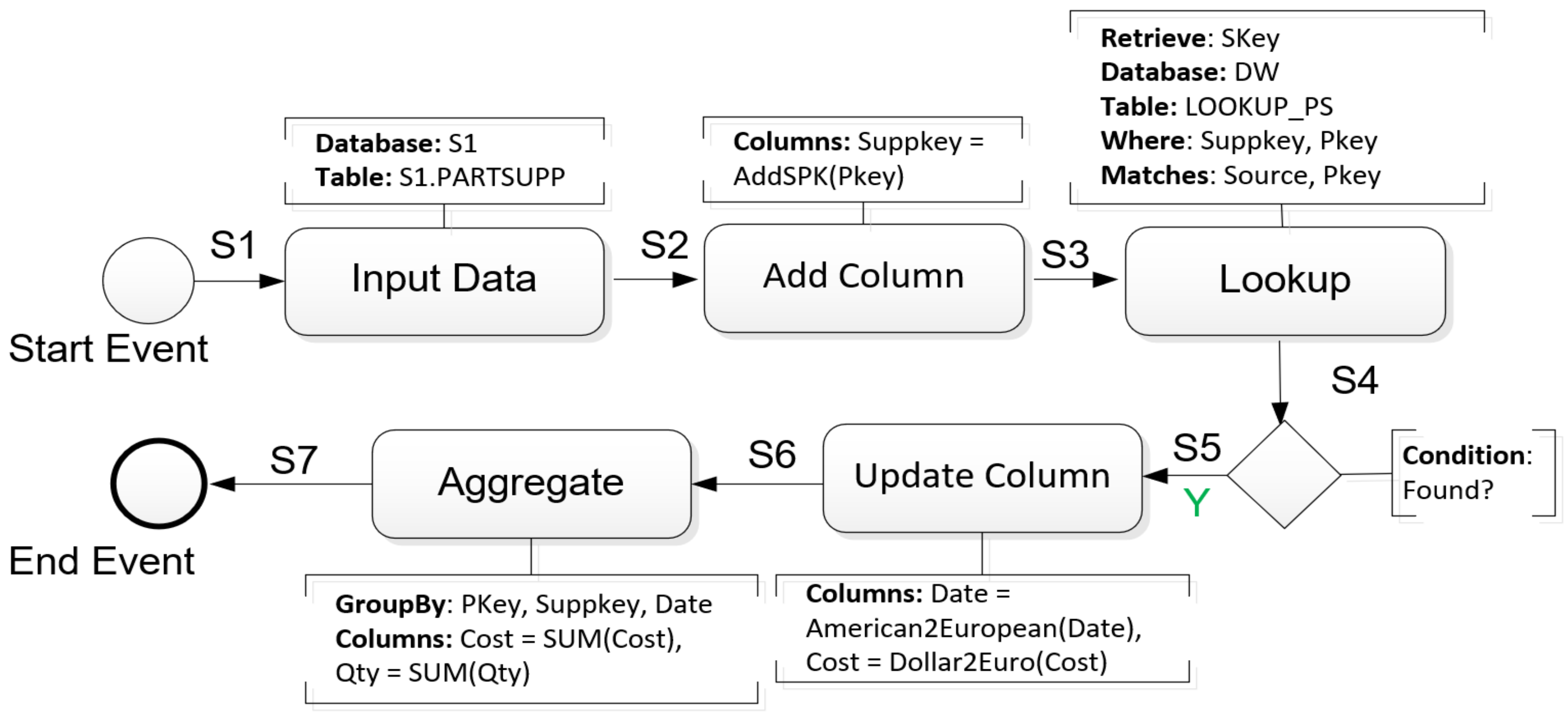
Extended RA — Logical model

SQL — Physical implementation

I. Scenario

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

II. BPMN4ETL



Database: S1
Table: S1.PARTSUPP

Columns: Suppkey = AddSPK(Pkey)

Retrieve: SKey
Database: DW
Table: LOOKUP_PS
Where: Suppkey, Pkey
Matches: Source, Pkey

Condition: Found?

GroupBy: PKey, Suppkey, Date
Columns: Cost = SUM(Cost), Qty = SUM(Qty)

Columns: Date = American2European(Date), Cost = Dollar2Euro(Cost)

III. Extended RA[2]

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

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

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

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

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

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

IV. BEXF[3]

```
<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>
```

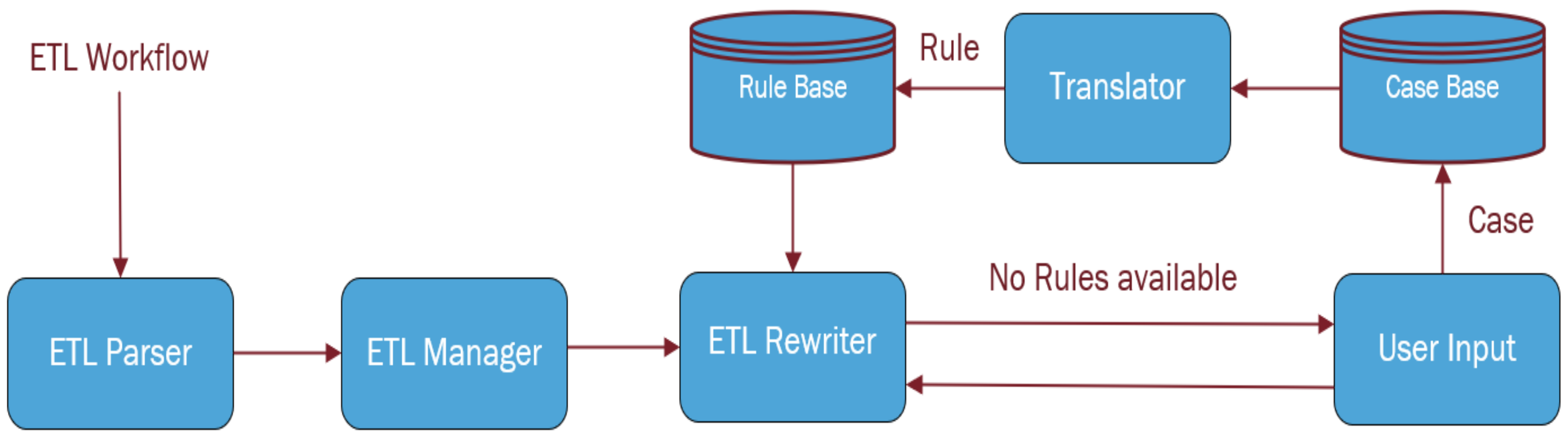
V. Performance Evaluation with TPC-DI benchmark [4].

PLSQL: Our approach (BPMN4ETL to Extended RA - SQL)
PDI: BPMN4ETL to Pentaho Data Integration tool
Execution 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

2. ETL Evolution

E3TL Framework



ETL Parser: Parses each command of the an ETL workflow. ETL workflow format (RA or SQLs).
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: Rewrites the commands in the ETL workflow by applying recommendations from the ETL manager.
Rule Base: Contains distinct rules based on conditions.
User Input: Request the user's input if no rule is available in the rule base to deal with the problem or several solutions are applicable to solve the problem.
Case Base: Repository to store cases.
Translator: Applies algorithms to develop rules from cases.

References

[1] Curino, C. A., Tanca, L., Moon, H. J., & Zaniolo, C. (2008). Schema Evolution in Wikipedia: Toward a Web Information System Benchmark. In: Proc. Of the 10th *International Conference on Enterprise Information Systems(ICEIS),* Barcelona, Spain.

[2] Awiti, J., Vaisman, A., Zimányi, E.: From Conceptual to Logical ETL Design Using BPMN and Relational Algebra. In: Proc. of the 21st *ACM International Conference on Big Data Analytics and Knowledge Discovery*, DAWAK 2019. Springer, Linz, Austria (2019), *forthcoming*

[3] Awiti, J., Zimányi, E.: An XML Interchange Format for ETL Models. In: Proc. of the 23rd *European Conference on Advances in Databases and Information Systems*, ser. Workshop on BI & Big Data Applications, ADBIS 2019, *forthcoming*.

[4] Poess, M., Rabl, T., Jacobsen, H. A., & Caufield, B. (2014). TPC-DI: The First Industry Benchmark for Data Integration. *Proceedings of the VLDB Endowment*, *7*(13), 1367-1378.

Judith Awiti
Judith.awiti@ulb.ac.be