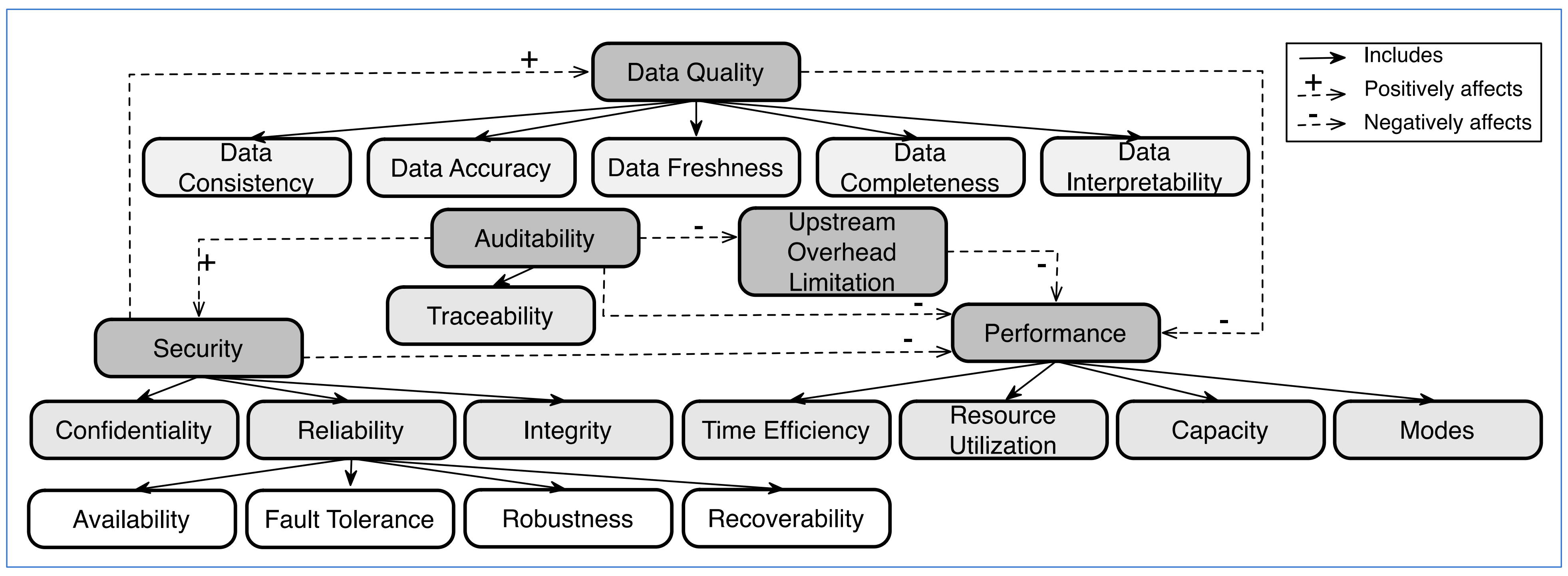
Automating User-Centered Design of Data-IntensiveProcesses

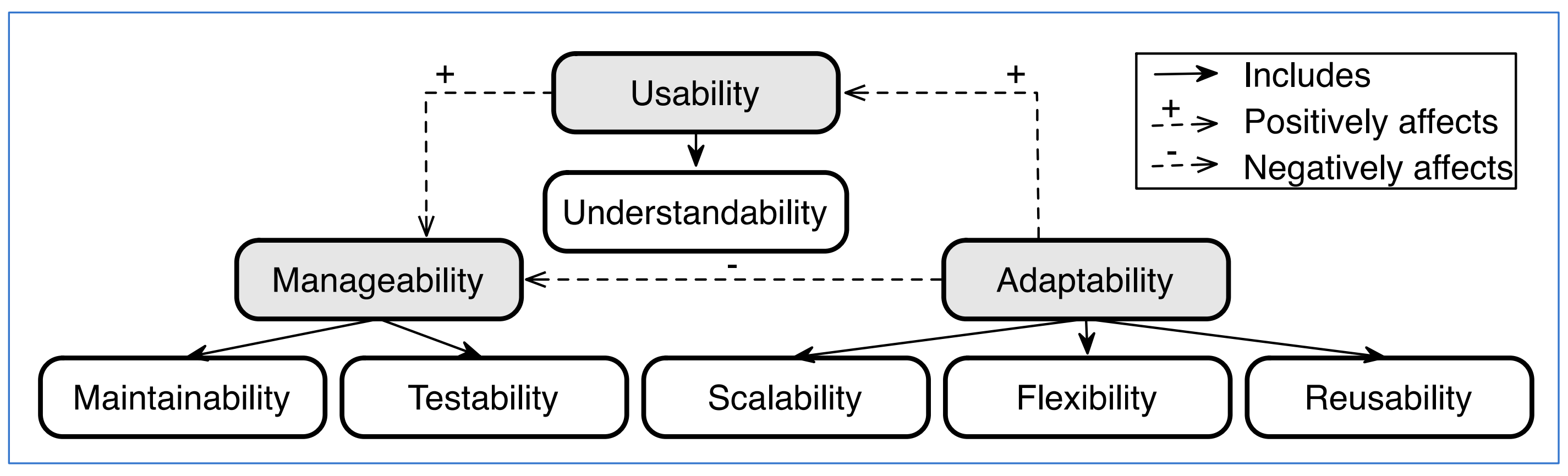
Vasileios Theodorou, Alberto Abelló
Universitat Politècnica de Catalunya, BarcelonaTech
[vasileios | aabello]@essi.upc.edu

Wolfgang Lehner, Maik Thiele
Technische Universität Dresden
[wolfgang.lehner | maik.thiele]@tu-dresden.de

Dependencies among process characteristics with construct implications



Dependencies among characteristics for design evaluation



POIESIS: A tool to automatically generate quality patterns over existing ETL processes in an iterative, dynamic fashion, with high-level user interaction and based on pursued goals.

ETL Generation and Improvement

- ***Functionality-Based Design***

ETL Process Designer: Semi-automatically designs ETL process model that implements basic ETL functionality. Input: domain metadata & business requirements.

- ***Quality Enhancement***

- iterative, incremental, user-centered
- ETL flow is represented as logical model that can be visualized for user as a BPMN process
- iterations are terminated at any point once user approves the model as adequate

Process Simulator : Simulates ETL processes and produces meaningful simple and aggregate analytics according to user's interest.

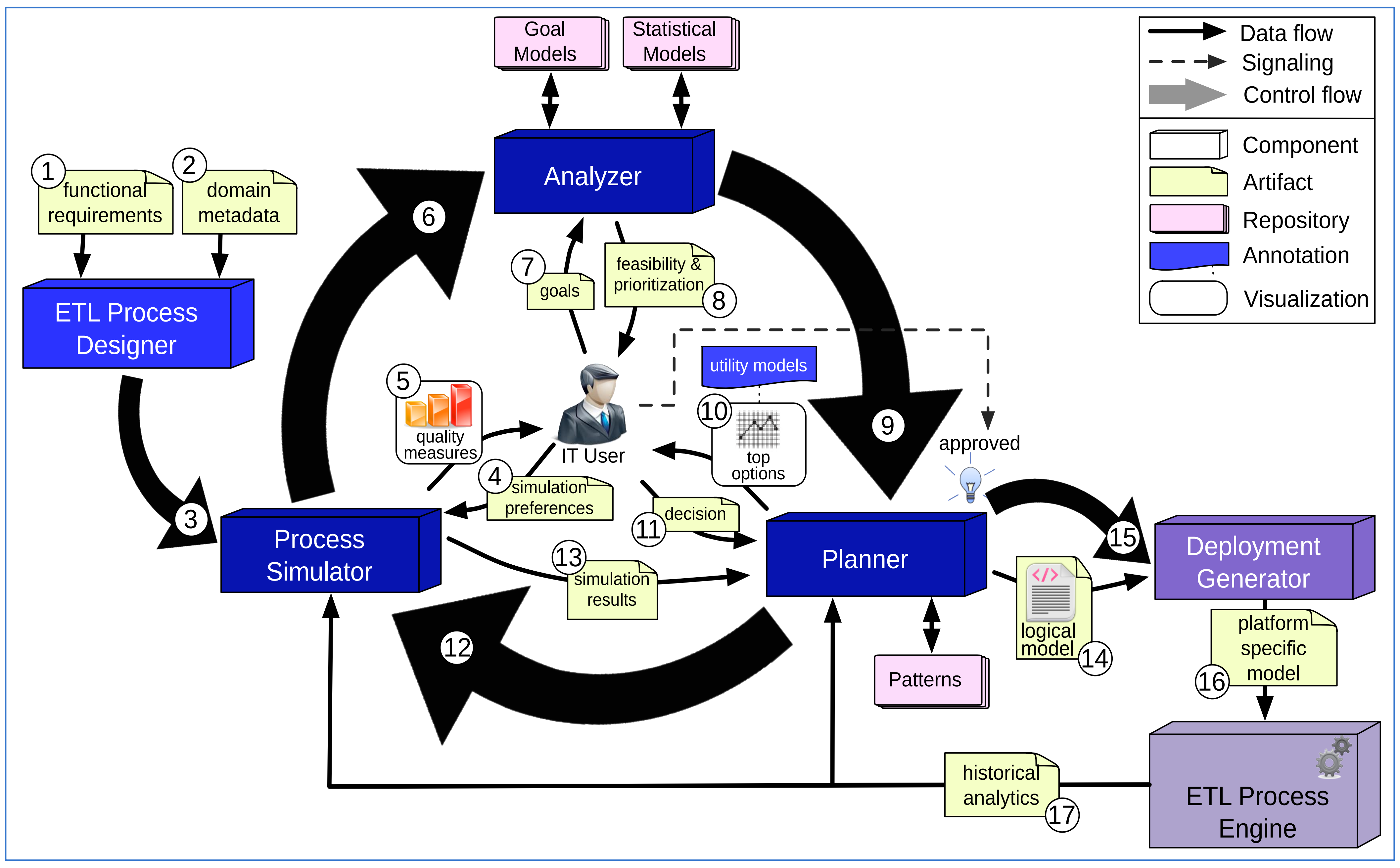
Analyzer: Performs feasibility analysis and prioritization of tasks about the quality patterns that can be integrated on the ETL process, using as input user-defined goals.

Planner: Using a set of available patters, it conducts a pre-selection of highest ranked pattern combinations, based on heuristics and cost models, as adjusted from real execution and simulation. The user selects one combination.

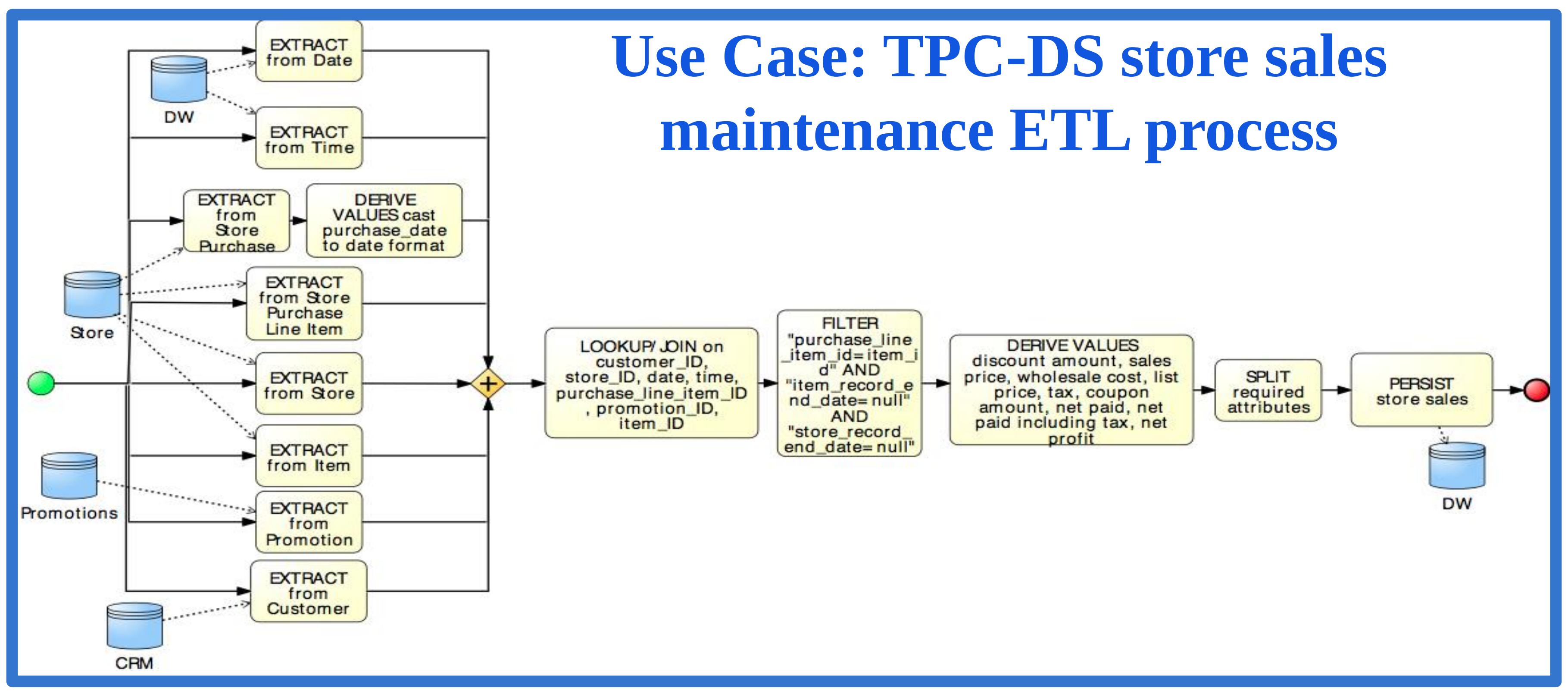
- ***Deployment & Execution***

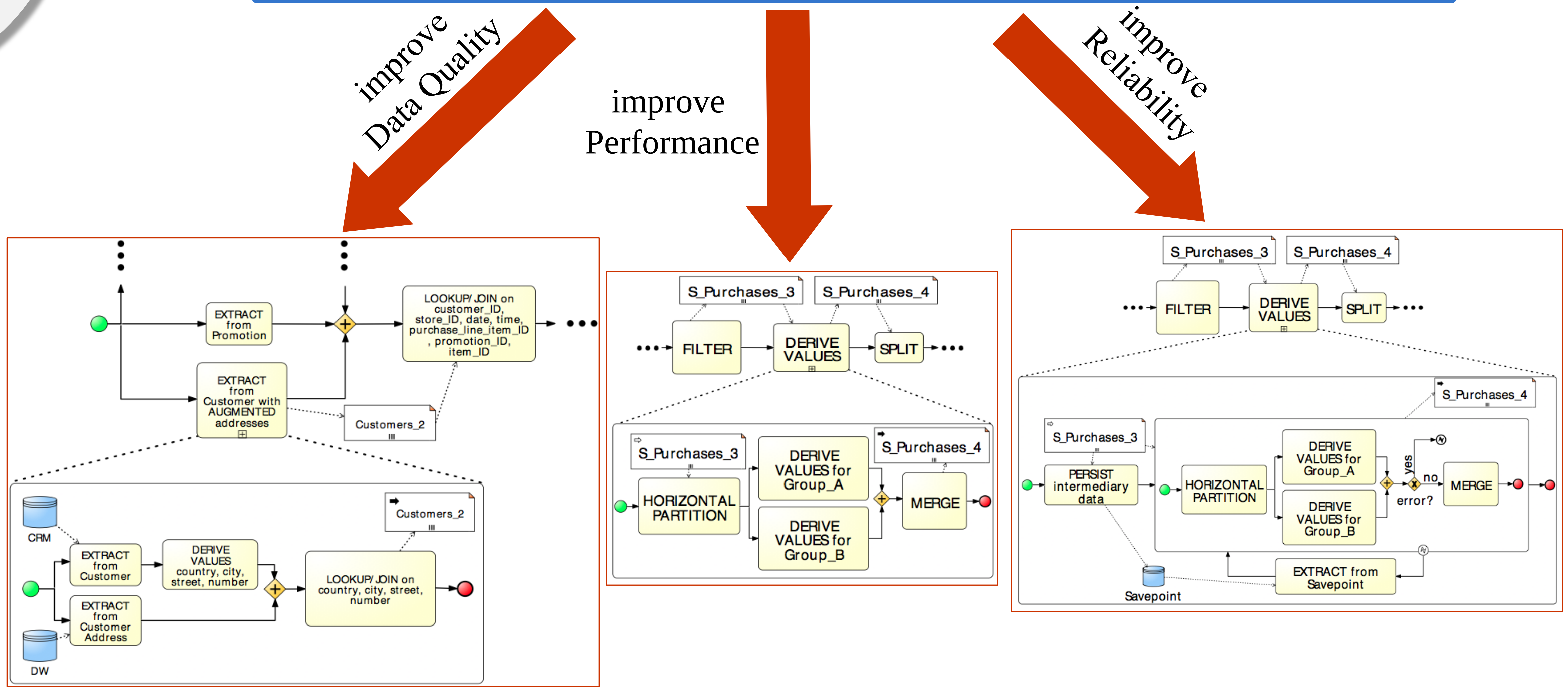
Deployment Generator : Translates logical model to platform specific model.

ETL Process Engine : Executes ETL and keeps traces for providing historical analytics.



Use Case: TPC-DS store sales maintenance ETL process


DW
EXTRACT from Date
EXTRACT from Time
EXTRACT from Store Purchase
DERIVE VALUES cast purchase_date to date format
Store
EXTRACT from Store Purchase Line Item
EXTRACT from Store
EXTRACT from Item
Promotions
EXTRACT from Promotion
EXTRACT from Customer
CRM
LOOKUP/ JOIN on customer_ID, store_ID, date, time, purchase_line_item_ID , promotion_ID, item_ID
FILTER "purchase_line_item_id=item_id" AND "item_record_end_date=null" AND "store_record_end_date=null"
DERIVE VALUES discount amount, sales price, wholesale cost, list price, tax, coupon amount, net paid, net paid including tax, net profit
SPLIT required attributes
PERSIST store sales
DW


improve Data Quality
improve Performance
improve Reliability
EXTRACT from Promotion
LOOKUP/ JOIN on customer_ID, store_ID, date, time, purchase_line_item_ID , promotion_ID, item_ID
EXTRACT from Customer with AUGMENTED addresses
Customers_2
CRM
EXTRACT from Customer
DERIVE VALUES country, city, street, number
EXTRACT from Customer Address
LOOKUP/ JOIN on country, city, street, number
DW
S_Purchases_3
S_Purchases_4
FILTER
DERIVE VALUES
SPLIT
HORIZONTAL PARTITION
DERIVE VALUES for Group_A
DERIVE VALUES for Group_B
MERGE
PERSIST intermediary data
yes
no
error?
EXTRACT from Savepoint
Savepoint

Characteristic	Sub-characteristic	Measure
performance	time efficiency	• Process cycle time • Average latency per tuple
	capacity	• Throughput of regular execution
data quality	data consistency	• % of tuples that violate business rules • % of duplicates
	data freshness	• Request time - Time of last update • 1 / (1 - age * Frequency of updates)
reliability	availability	• Mean Time Between Failures (MTBF) • Uptime of ETL process
	recoverability	• Number of recovery points used • % of successfully resumed workflow executions • Mean time to repair (MTTR)
manageability	maintainability	• Length of process workflow's longest path • Coupling of process workflow • # of merge elements in the process model
	testability	• Cyclomatic Complexity of the ETL process workflow

Fourth European Business Intelligence Summer School (eBISS 2014) – Berlin, Germany