Mining for Meaning

Knowledge Representation Methods in Process Mining for Transparency, Verifiability, and Replicability.

Riley Moher
23.07.2025
CPMC Annual Meeting

UNIVERSITY OF TORONTO

Semantic Technologies Lab

What’s wrong here?

Timestamp	Event	Patient
12:02	Patient Intake	John Smith
12:05	Patient Intake	John Smith
12:06	Diagnostic	John Smith

What's wrong here?

- Issue: Patient Intake should only occur once for the same patient.

Timestamp	Event	Patient
12:02	Patient Intake	John Smith
12:05	Patient Intake	John Smith
12:06	Diagnostic	John Smith

What's wrong here?

- Issue: Patient Intake should only occur once for the same patient.
- Solution: The event label was wrong

Timestamp	Event	Patient
12:02	Patient Intake	John Smith
12:05	Additional Screening	John Smith
12:06	Diagnostic	John Smith

What's wrong here?

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

What's wrong here?

- Issue: Activities cannot end before they begin

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

What's wrong here?

- Issue: Activities cannot end before they begin
- Solution: Fix the timestamp with the correct ordering

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
16:30	Credit Check	Complete
15:38	Credit Check	Start

- Process data is given meaning once interpreted

- Process data is given meaning once interpreted
- Interpretations rely on process knowledge

- Process data is given meaning once interpreted
- Interpretations rely on process knowledge
- Process knowledge comes in many different forms

- Process data is given meaning once **interpreted**
- Interpretations rely on process **knowledge**
- Process knowledge comes in many **different forms**

Idea: provide the means to capture these interpretations & knowledge

Agenda

- What does it mean to represent knowledge?
- What's so special about process knowledge?
- How do we apply this to process mining?

Agenda

- What does it mean to represent knowledge?
- What’s so special about process knowledge?
- How do we apply this to process mining?

Knowledge Representation (KR)

- As humans, we can easily understand, interpret, and reason with knowledge
- We leverage that knowledge every day to achieve goals in the real world
- How can machines do the same?

Knowledge versus Data

- Storage =/= Representation

Knowledge versus Data

- Storage =/= Representation
- KR enables inference and verification

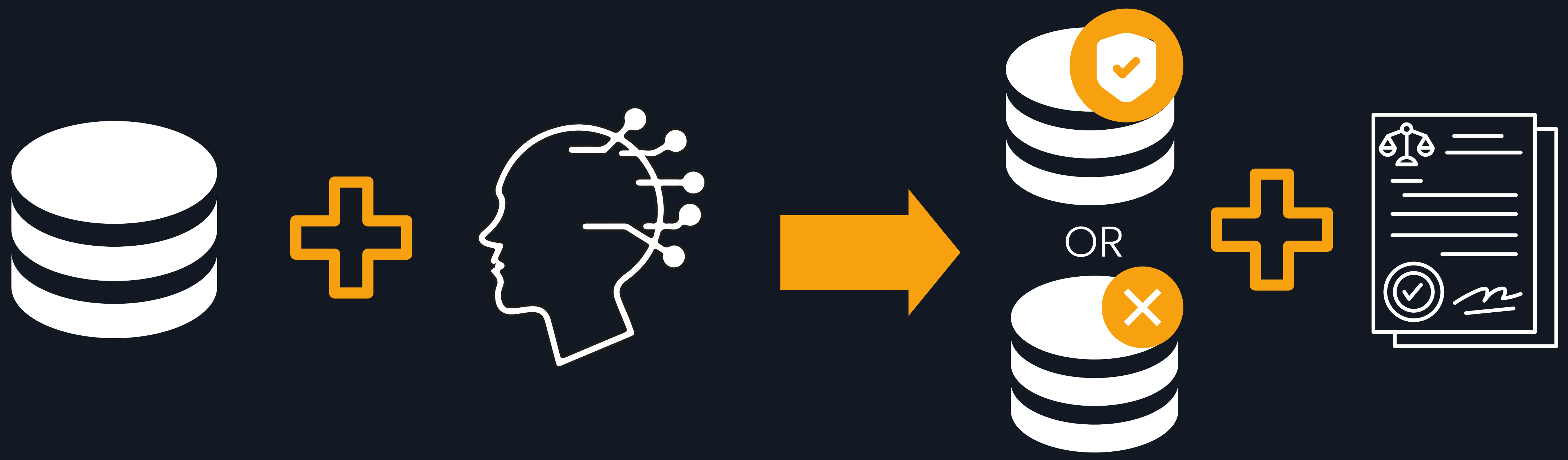
Knowledge versus Data

Inference

Facts and rules produce
new facts

Verification

Facts and rules produce
proofs of compliance or
non-compliance.



Knowledge versus Data

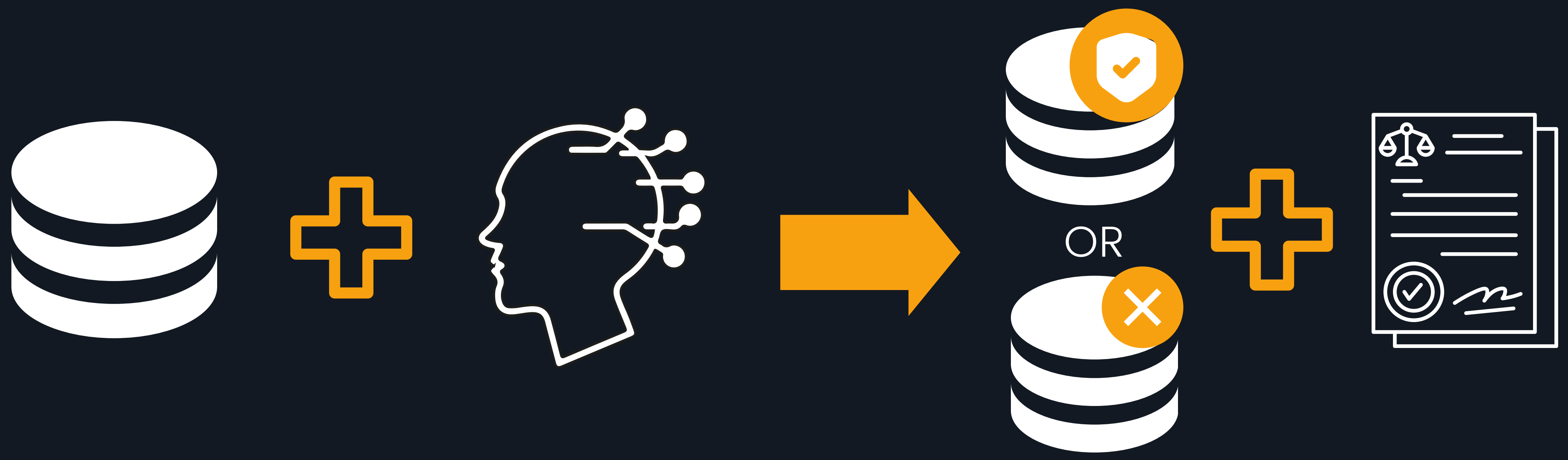
But what does this actually look like?

Inference

Facts and rules produce new facts

Verification

Facts and rules produce proofs of compliance or non-compliance.



KR in practice

- Data is translated into a set of logically represented facts

- Rules about a domain are given by an ontology

KR in practice

- Data is translated into a set of logically represented facts
- Rules about a domain are given by an ontology
- What is an ontology?

KR in practice

- Data is translated into a set of logically represented facts
- Rules about a domain are given by an ontology
- What is an ontology?
  - A structured vocabulary for a domain
  - An unambigious model of knowledge
  - Rules for entities and their relationships

Agenda

- What does it mean to represent knowledge?
- What's so special about process knowledge?
- How do we apply this to process mining?

Process Knowledge

- Not only do we need to represent the world, but how it changes

Process Knowledge

- Not only do we need to represent the world, but how it changes
- Levels of knowledge - both domain-level and fundamental

Process Knowledge

- Not only do we need to represent the world, but how it changes
- Levels of knowledge - both domain-level and fundamental
- Overloaded concepts - the event

What makes an event?

- “An atomic granule of an activity that has been observed” - XES definition of an event

What makes an event?

- “An atomic granule of an activity that has been observed” - XES definition of an event
- “Awaiting Assignment”, “Loan Approval”, “Create Purchase Order Item“

What makes an event?

- “An atomic granule of an activity that has been observed” - XES definition of an event
- “Awaiting Assignment”, “Loan Approval”, “Create Purchase Order Item“
- How do events relate to tasks, resources, processes, occurrences?

Agenda

- What does it mean to represent knowledge?
- What's so special about process knowledge?
- How do we apply this to process mining?

Ontology-Aware Process Mining

- Make process knowledge used throughout the process mining lifecycle explicit

Ontology-Aware Process Mining

- Make process knowledge used throughout the process mining lifecycle explicit
- Enable new kinds of reasoning over event log data

Ontology-Aware Process Mining

- Make process knowledge used throughout the process mining lifecycle explicit
- Enable new kinds of reasoning over event log data
- Done by mapping data to a knowledge base, and augmenting it through knowledge patterns

Mapping and Formalization

Raw Event Data	Knowledge Base
Ad-Hoc Interpretations	Data Theory
Business Questions	Logical Queries
Business Validations	Logical Proofs

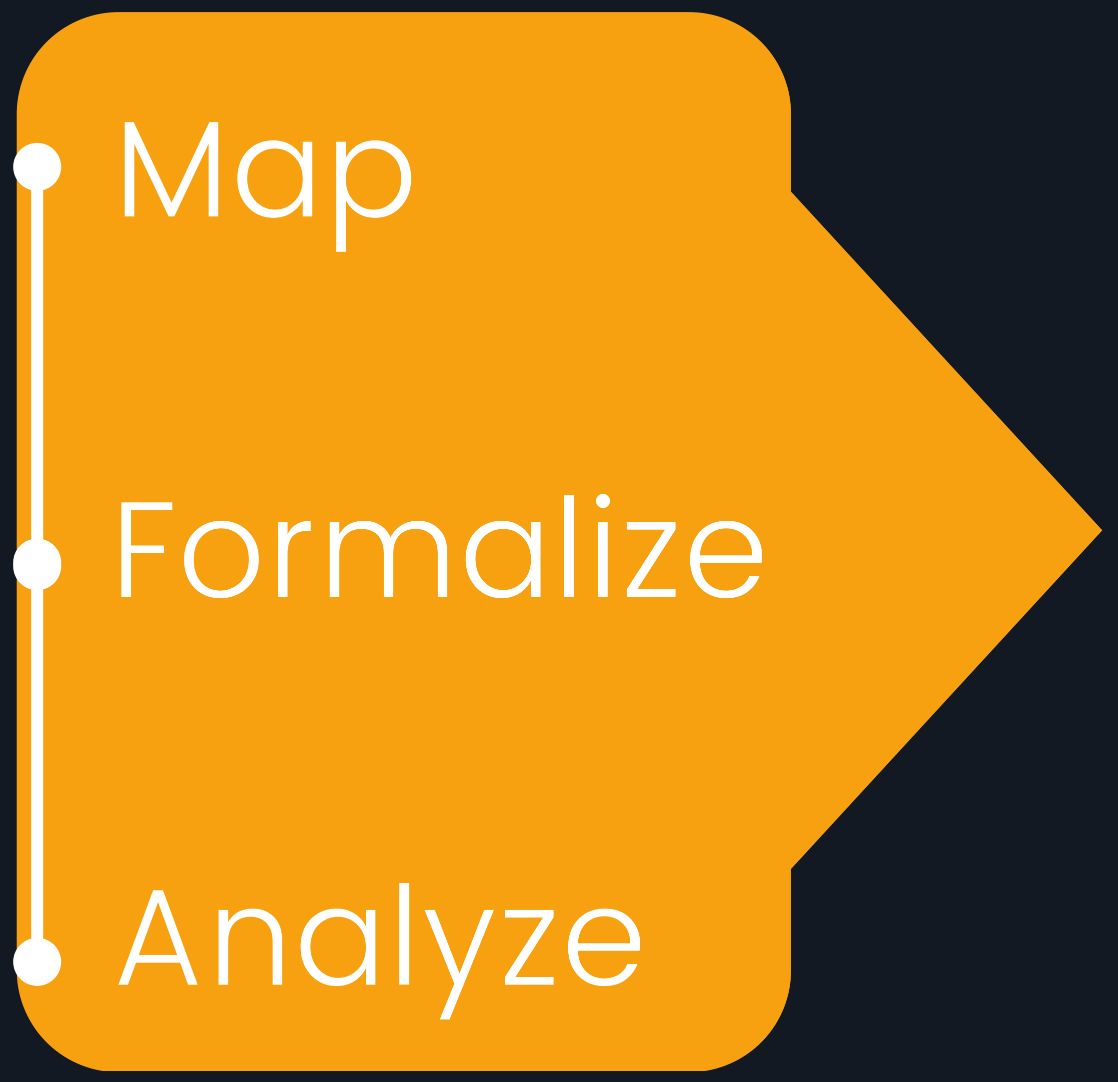
Mapping and Formalization

Raw Event Data

Ad-Hoc Interpretations

Business Questions

Business Validations

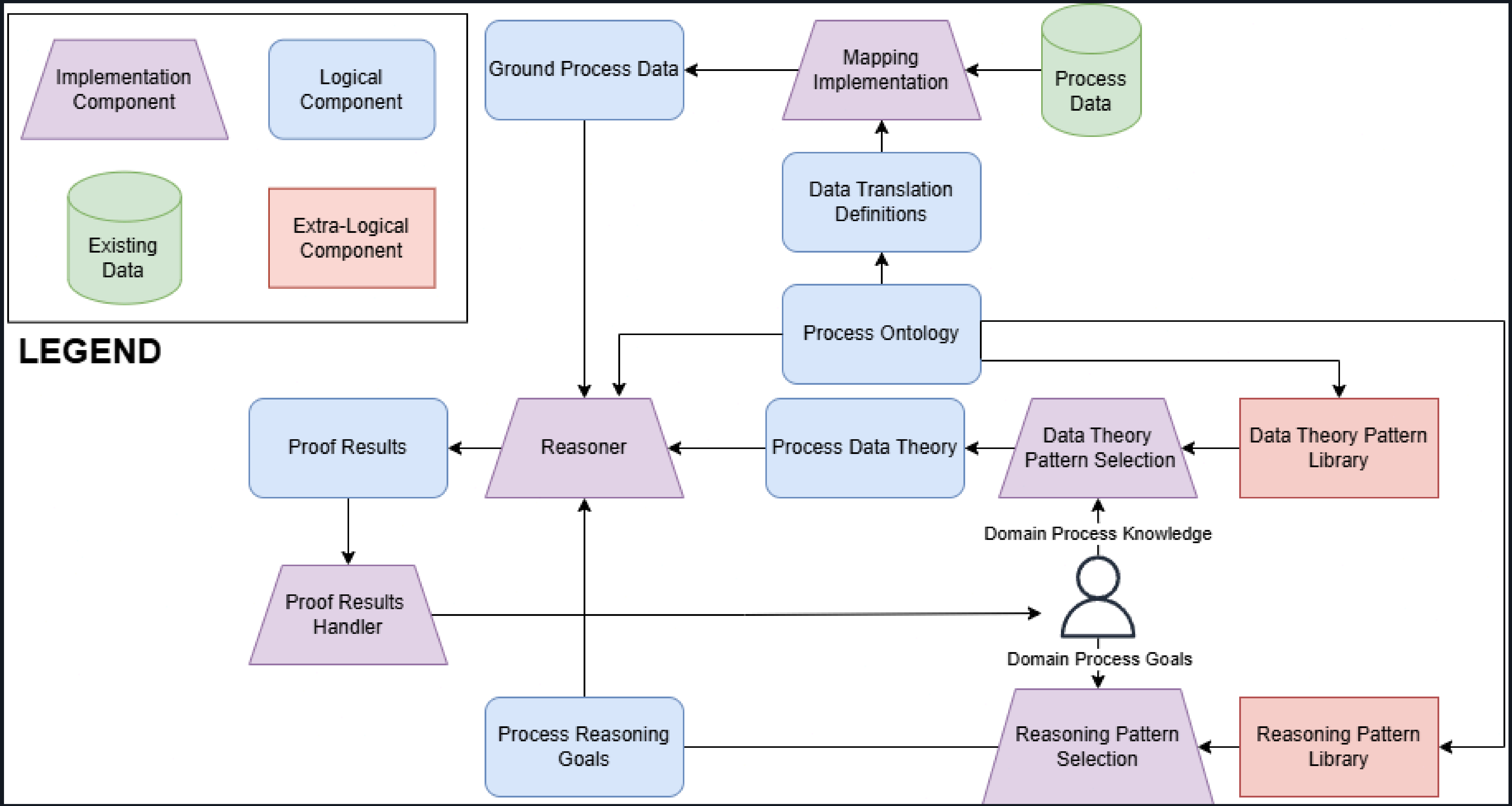


Knowledge Base

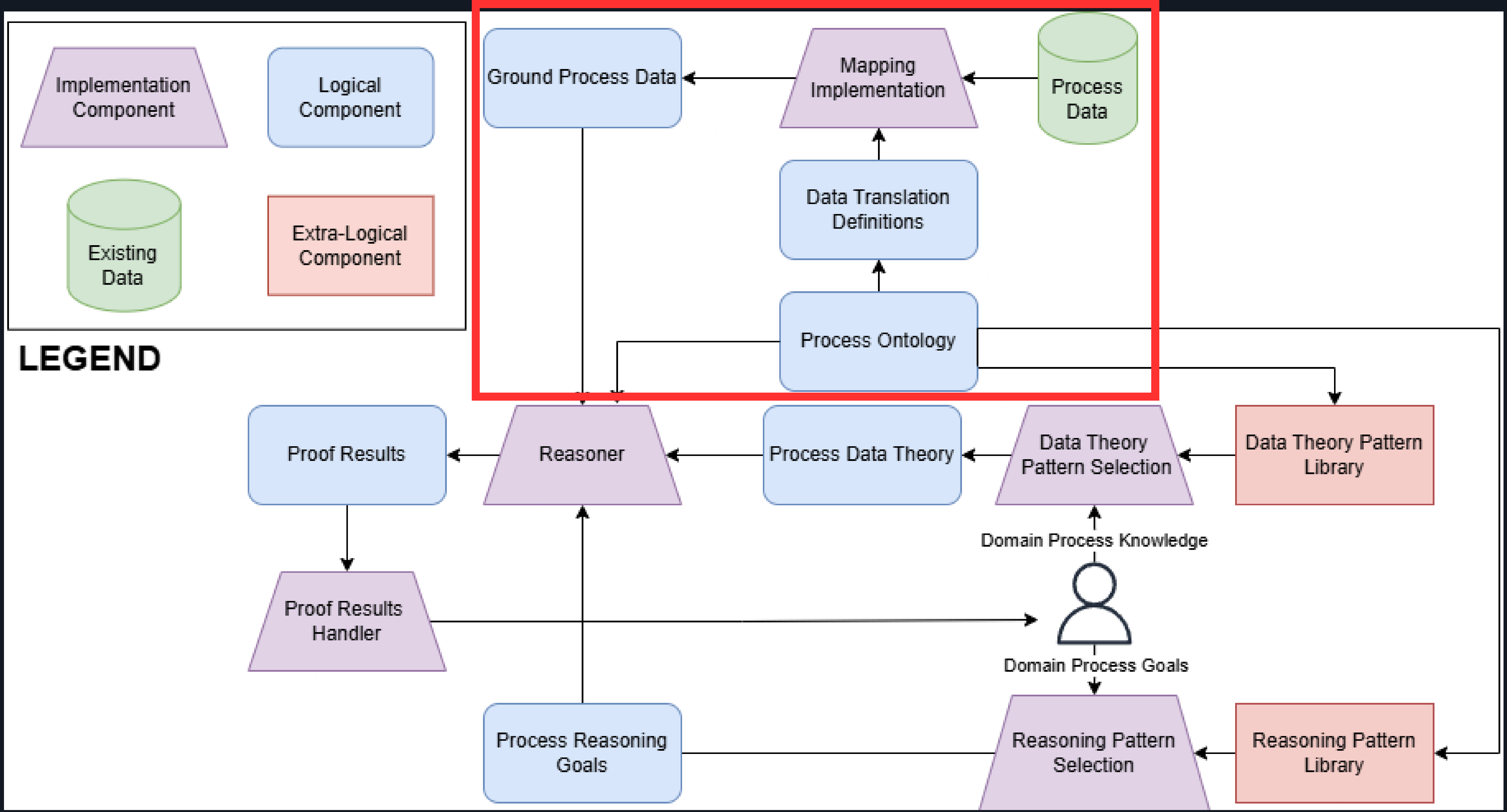
Data Theory

Logical Queries

Logical Proofs



Implementation Component
Logical Component
Existing Data
Extra-Logical Component
LEGEND
Ground Process Data
Mapping Implementation
Process Data
Data Translation Definitions
Process Ontology
Proof Results
Reasoner
Process Data Theory
Data Theory Pattern Selection
Data Theory Pattern Library
Domain Process Knowledge
Proof Results Handler
Domain Process Goals
Process Reasoning Goals
Reasoning Pattern Selection
Reasoning Pattern Library



Implementation Component
Logical Component
Existing Data
Extra-Logical Component
LEGEND
Ground Process Data
Mapping Implementation
Process Data
Data Translation Definitions
Process Ontology
Proof Results
Reasoner
Process Data Theory
Data Theory Pattern Selection
Data Theory Pattern Library
Domain Process Knowledge
Proof Results Handler
Domain Process Goals
Process Reasoning Goals
Reasoning Pattern Selection
Reasoning Pattern Library

Mapping Data

- Tabular data from event logs becomes enriched relational data

Mapping Data

- Tabular data from event logs becomes enriched relational data

Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2

Mapping Data

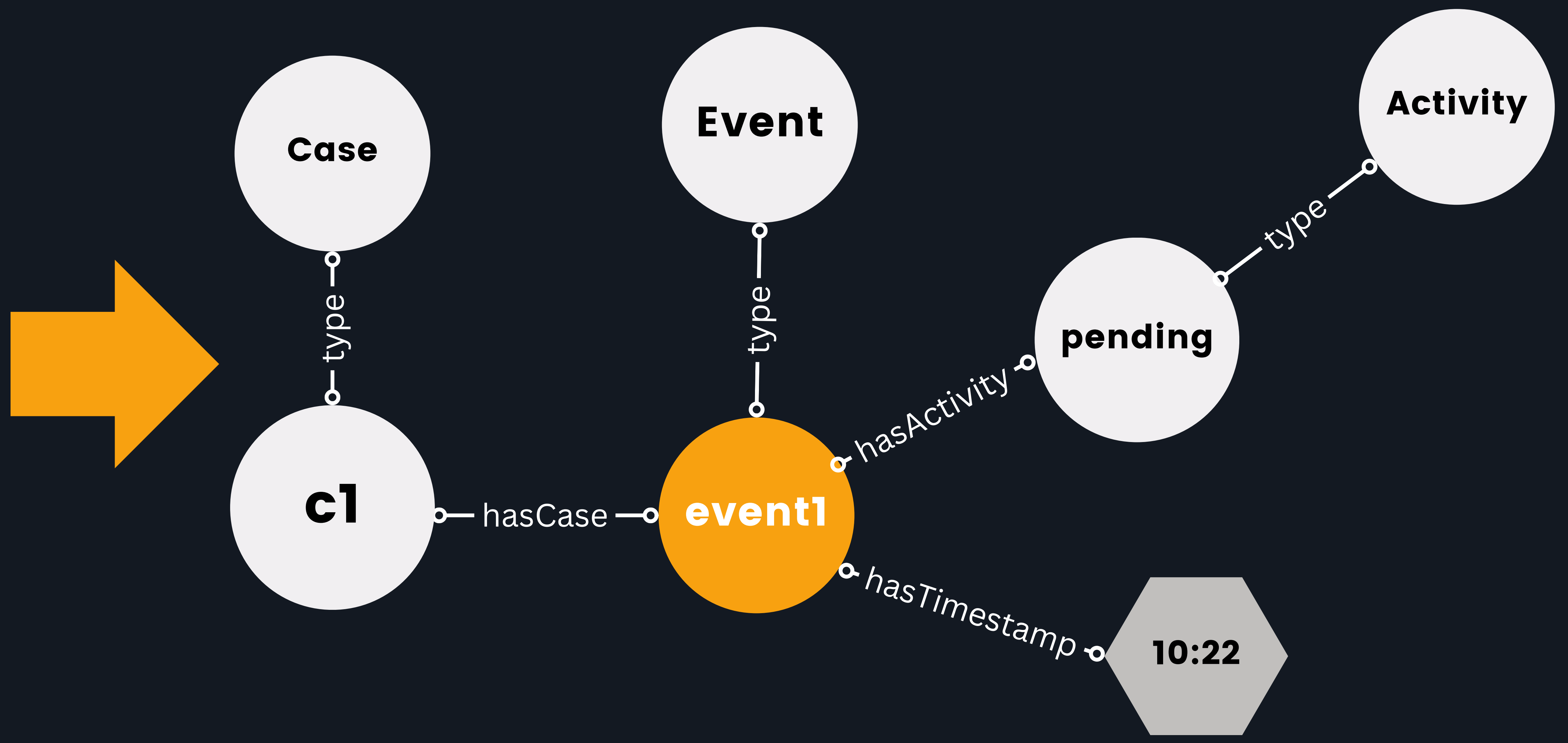
- Tabular data from event logs becomes enriched relational data

Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2

Mapping Data

- Tabular data from event logs becomes enriched relational data

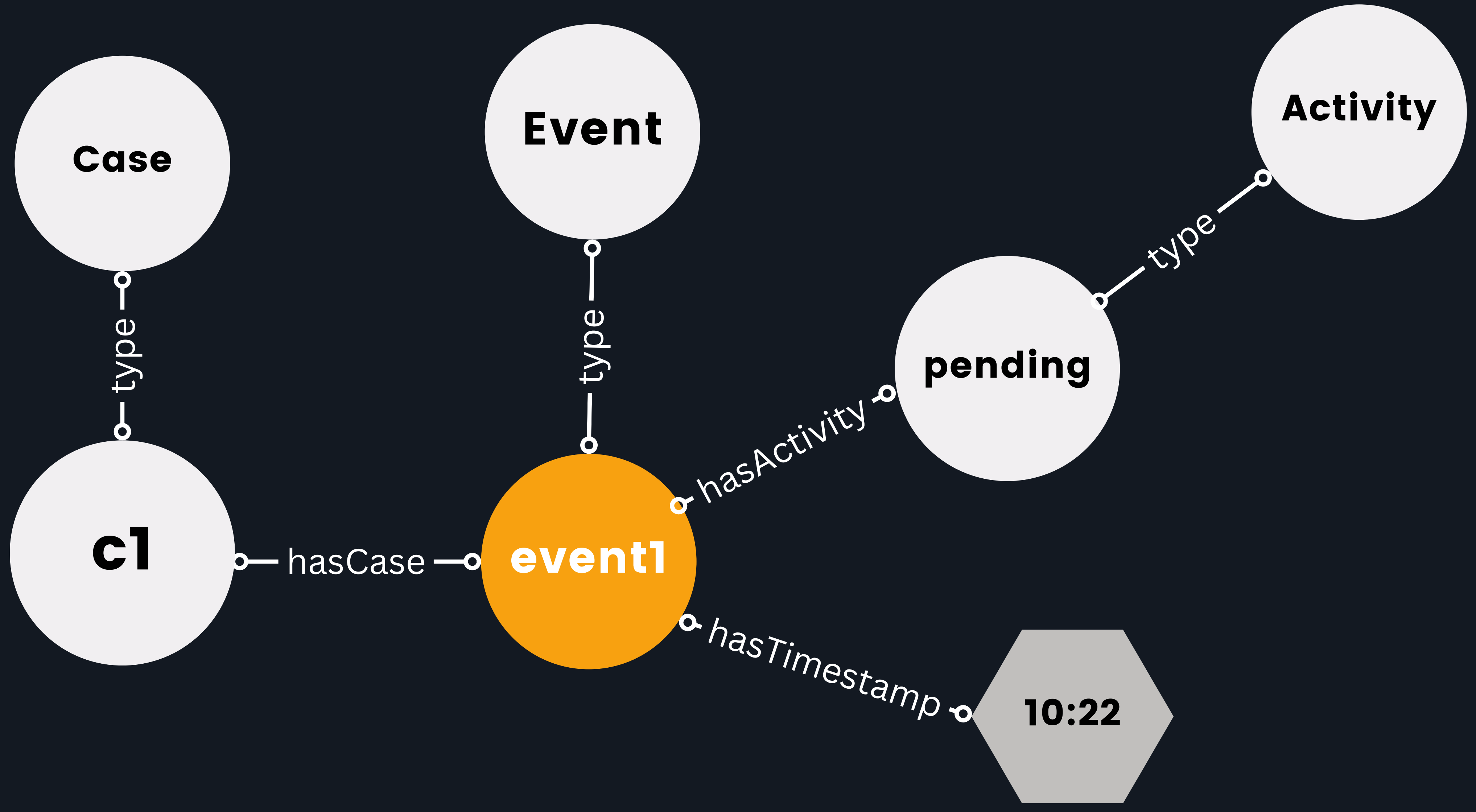
Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2



Mapping Data

- Tabular data from event logs becomes enriched relational data

Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2



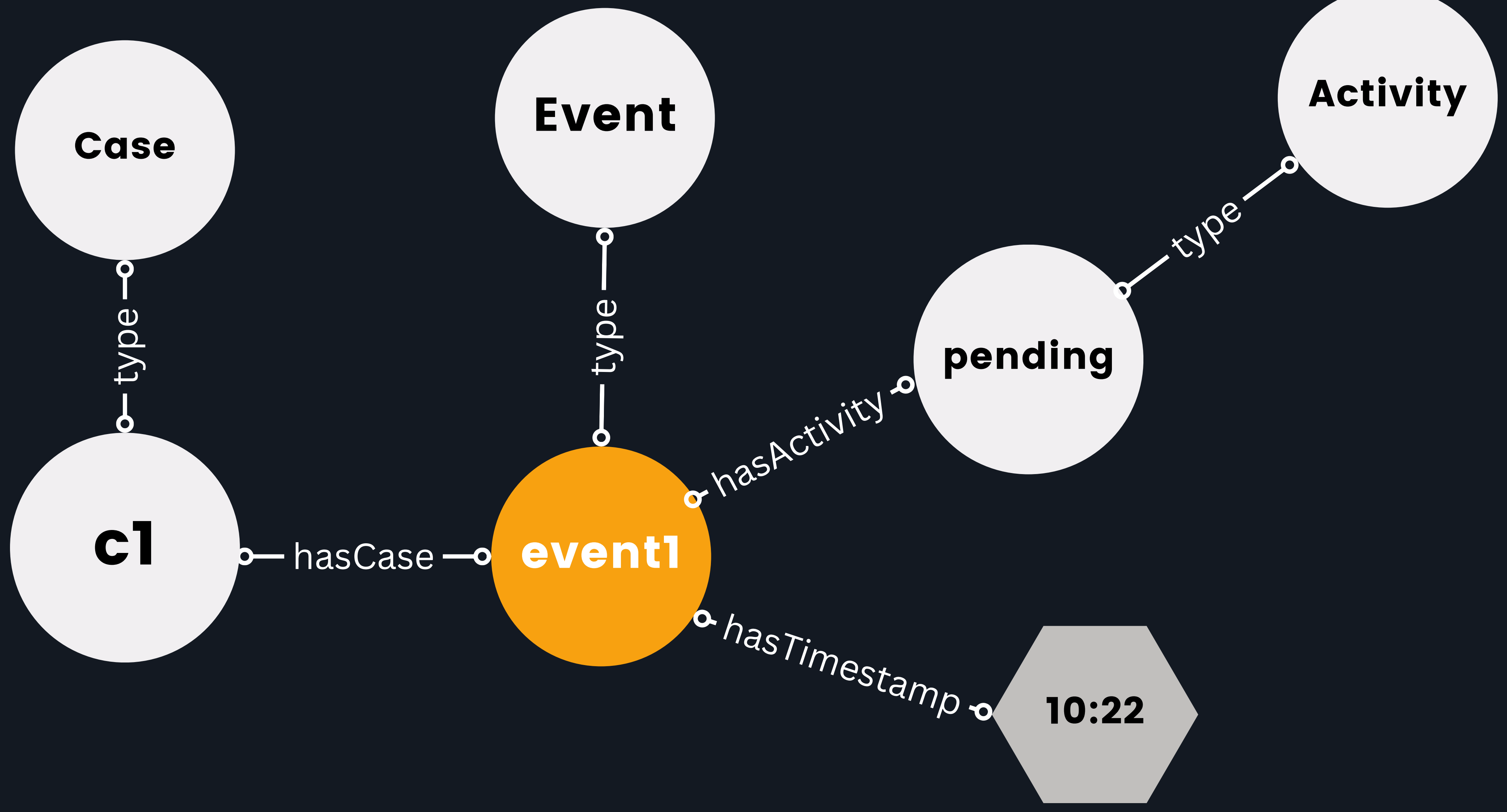
This forms our ground process data

Mapping Data

- Map
- Formalize
- Analyze

- Tabular data from event logs becomes enriched relational data

Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2



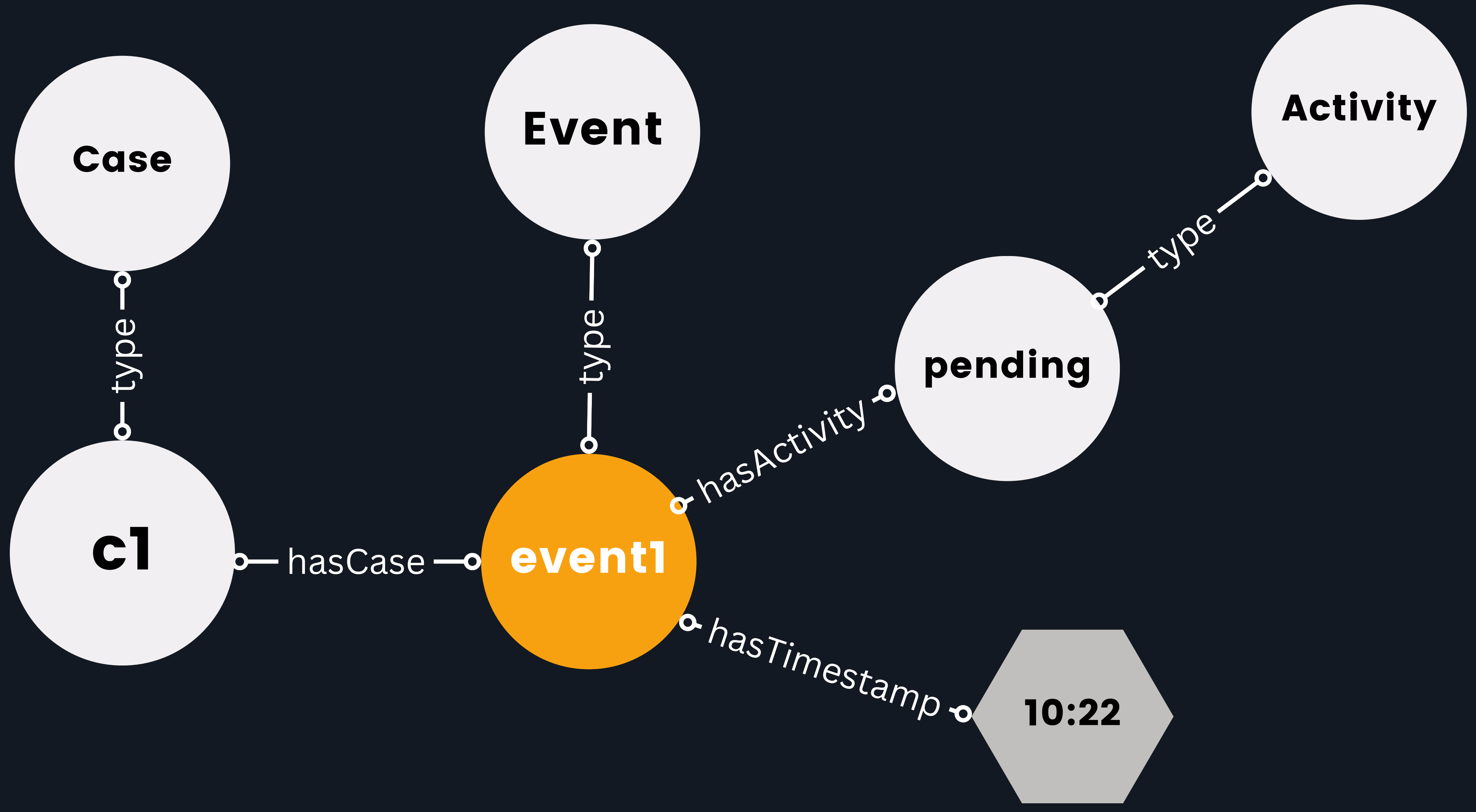
This forms our ground process data

Mapping Data

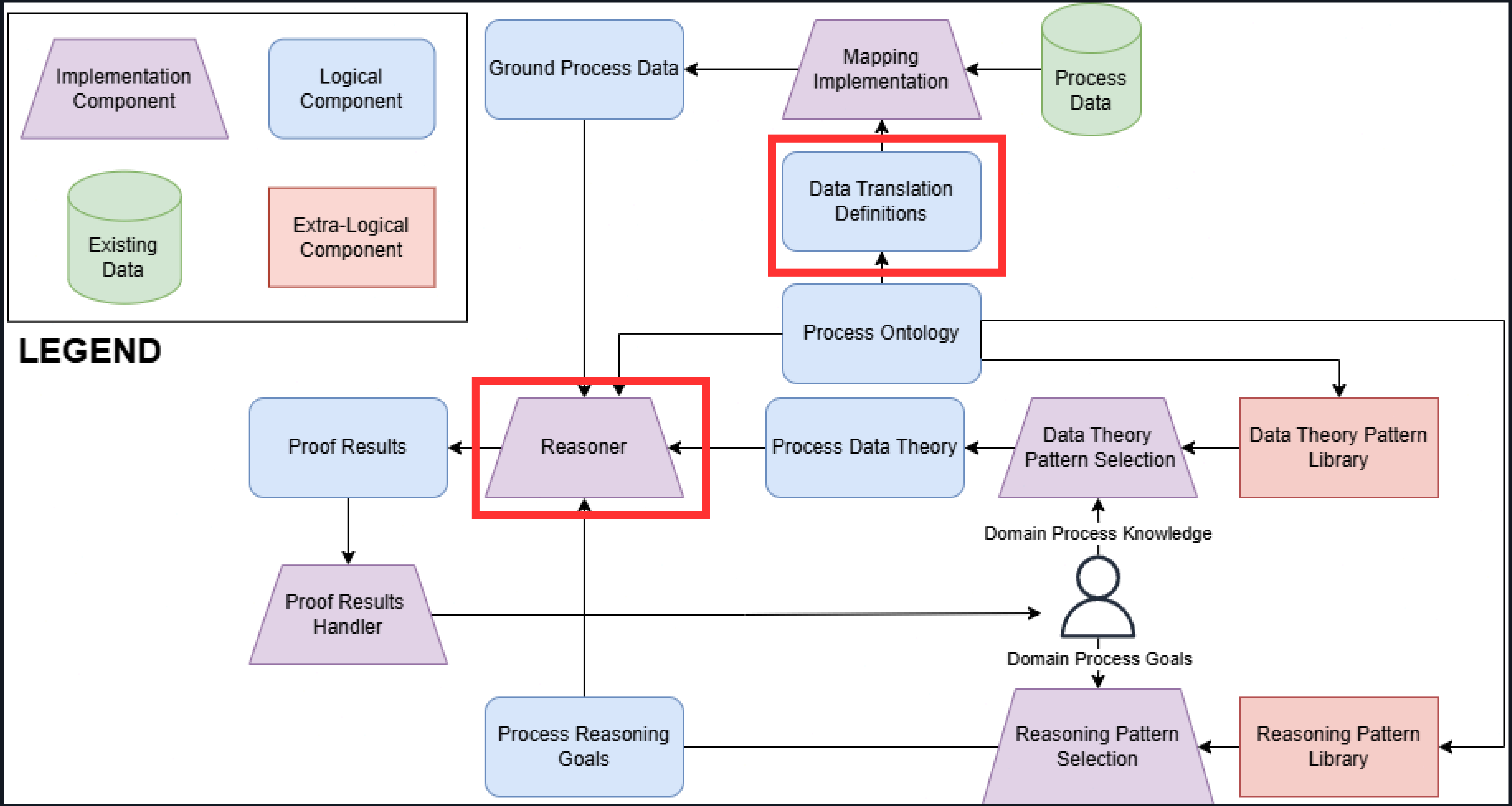
Map
Formalize
Analyze

- Tabular data from event logs becomes enriched relational data

Activity	Timestamp	Case
pending	10:22	c1
approval	10:23	c1
submission	12:10	c2



This forms our ground process data



Implementation Component
Logical Component
Existing Data
Extra-Logical Component
LEGEND
Ground Process Data
Mapping Implementation
Process Data
Data Translation Definitions
Process Ontology
Proof Results
Reasoner
Process Data Theory
Data Theory Pattern Selection
Data Theory Pattern Library
Domain Process Knowledge
Proof Results Handler
Domain Process Goals
Process Reasoning Goals
Reasoning Pattern Selection
Reasoning Pattern Library

Reasoning with Event Data

- How can we formally (and automatically) identify ordering errors?

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

Map
Formalize
Analyze

Reasoning with Event Data

- How can we formally (and automatically) identify ordering errors?

Map
Formalize
Analyze

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ground Data

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Timestamp	Event	Lifecycle Transition
14:20	Process Application	Start
15:30	Credit Check	Complete
15:38	Credit Check	Start

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ground Data

Reasoning with Event Data

- How can we formally (and automatically) identify ordering errors?

Map
Formalize
Analyze

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

+

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Reasoning with Event Data

- How can we formally (and automatically) identify ordering errors?

Map
Formalize
Analyze

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

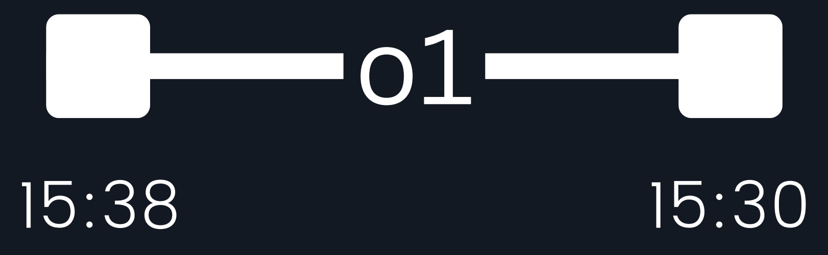
Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Translated Data

occurrence(o1)
beginOf(o1, 15:38)
endOf(o1, 15:30)



Reasoning with Event Data

Map
Formalize
Analyze

- How can we formally (and automatically) identify ordering errors?

Ground Data

Event(e2)
hasActivity(e0, creditCheck)
hasTransition(e2, complete)

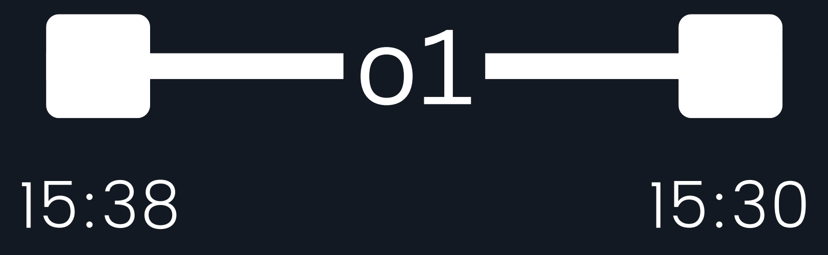
Event(e3)
hasActivity(e3, creditCheck)
hasTransition(e3, start)

Ontology Translations

"Two events sharing an activity each with a start and end transition indicates an acitivty occurrence"

Translated Data

occurrence(o1)
beginOf(o1, 15:38)
endOf(o1, 15:30)



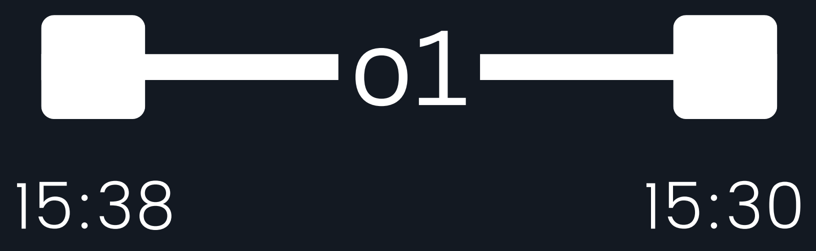
Reasoning with Event Data

Map
Formalize
Analyze

- How can we formally (and automatically) identify ordering errors?

Translated Data

occurrence(o1)
beginOf(o1, 15:38)
endOf(o1, 15:30)



Reasoning with Event Data

- How can we formally (and automatically) identify ordering errors?

Translated Data

occurrence(o1)
beginOf(o1, 15:38)
endOf(o1, 15:30)

Ontology

"Activity occurrences start points are less than or equal to their end points"

$$t_1 \leq t_2$$

Reasoning with Event Data

- Map
- Formalize
- Analyze

- How can we formally (and automatically) identify ordering errors?

Translated Data

occurrence(o1)
beginOf(o1, 15:38)
endOf(o1, 15:30)

Ontology

"Activity occurrences start points are less than or equal to their end points"

$$t_1 \leq t_2$$

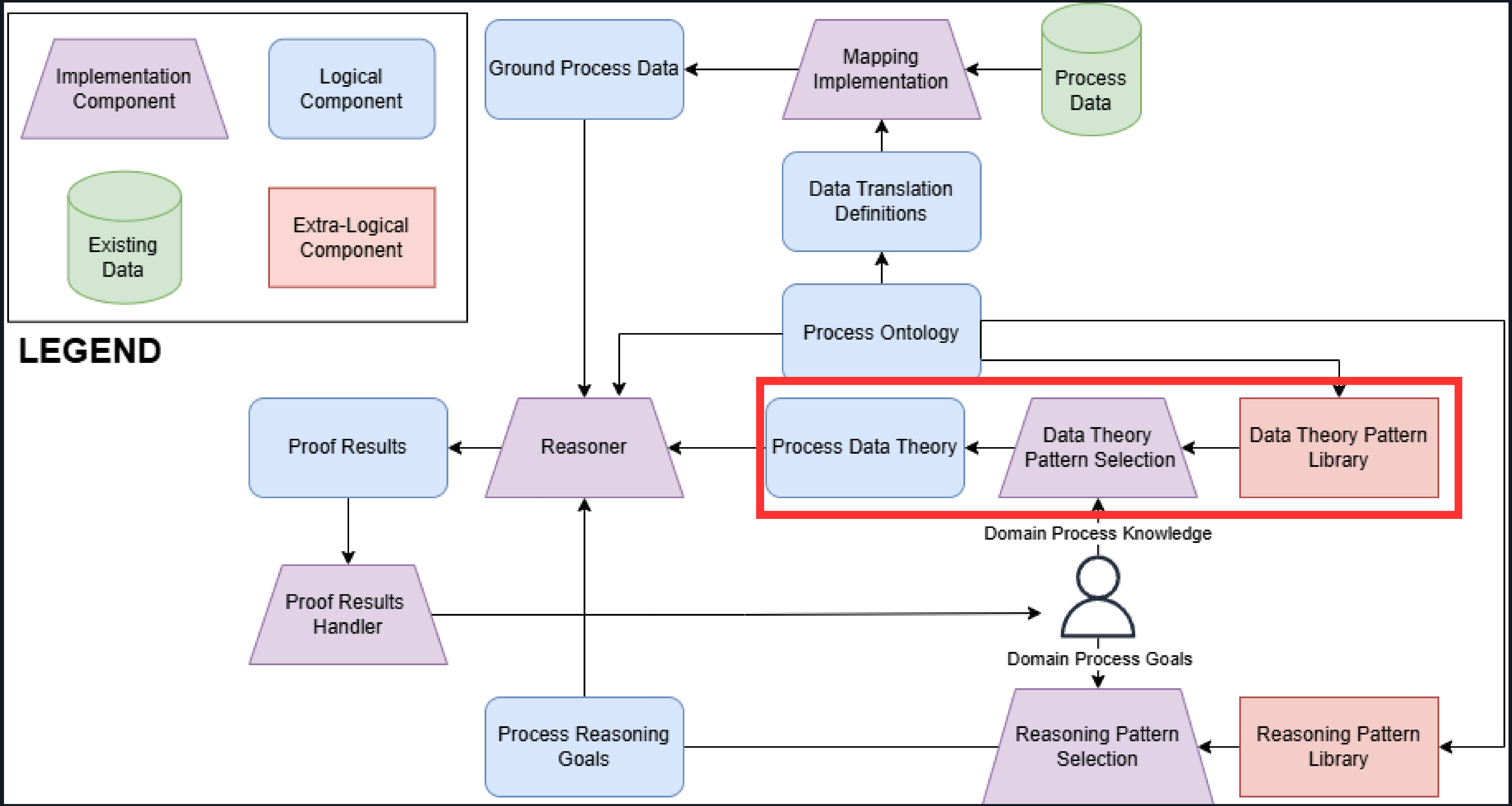
Proof of Inconsistency

$$t_1 = 15:38$$

$$t_2 = 15:30$$

$$\boxed{\begin{array}{c} t_1 > t_2 \\ t_1 \leq t_2 \end{array}}$$

What about domain-specific analysis?



Implementation Component
Logical Component
Existing Data
Extra-Logical Component
LEGEND
Ground Process Data
Mapping Implementation
Process Data
Data Translation Definitions
Process Ontology
Proof Results
Reasoner
Process Data Theory
Data Theory Pattern Selection
Data Theory Pattern Library
Domain Process Knowledge
Proof Results Handler
Domain Process Goals
Process Reasoning Goals
Reasoning Pattern Selection
Reasoning Pattern Library

Knowledge Patterns

- “When a fragile object is dropped, it breaks”
- “Patient Intake should only occur once for the same patient.”

Knowledge Patterns

- “When a fragile object is dropped, it breaks”
- “Patient Intake should only occur once for the same patient.”

Knowledge Patterns

- “When a fragile object is dropped, it breaks”

State Based Effect

- “Patient Intake should only occur once for the same patient.”

Occurrence Constraint

Knowledge Patterns

- "When a fragile object is dropped, it breaks"

  State Based Effect

- "Patient Intake should only occur once for the same patient."

  Occurrence Constraint

Knowledge Patterns

- State-Based Effects (SBE)
- “When a fragile object is dropped, it breaks”

Knowledge Patterns

- State-Based Effects (SBE)
- “When a fragile object is dropped, it breaks”
- While some initial condition holds, and an occurrence happens, some resulting condition holds afterwards

Knowledge Patterns

- State-Based Effects (SBE)
- “When a fragile object is dropped, it breaks”
- While some initial condition holds, and an occurrence happens, some resulting condition holds afterwards

Knowledge Patterns

- State-Based Effects (SBE)
- "When a fragile object is dropped, it breaks"
- While some initial condition holds, and an occurrence happens, some resulting condition holds afterwards
- Patterns abstract common process knowledge
- SBE(c1, a, c2)

Knowledge Patterns in Reasoning

- For domain specific reasoning problems
  - Use patterns to declare business rules, data interpretations (the data theory)
  - “Patient Intake occurs exactly once for one patient visit”

Knowledge Patterns in Reasoning

- For domain specific reasoning problems
  - Use patterns to declare business rules, data interpretations (the data theory)
  - "Patient Intake occurs exactly once for one patient visit"
- For general reasoning problems
  - The process ontology contains fundamental knowledge about how processes work
  - "Occurrences cannot end before they begin"

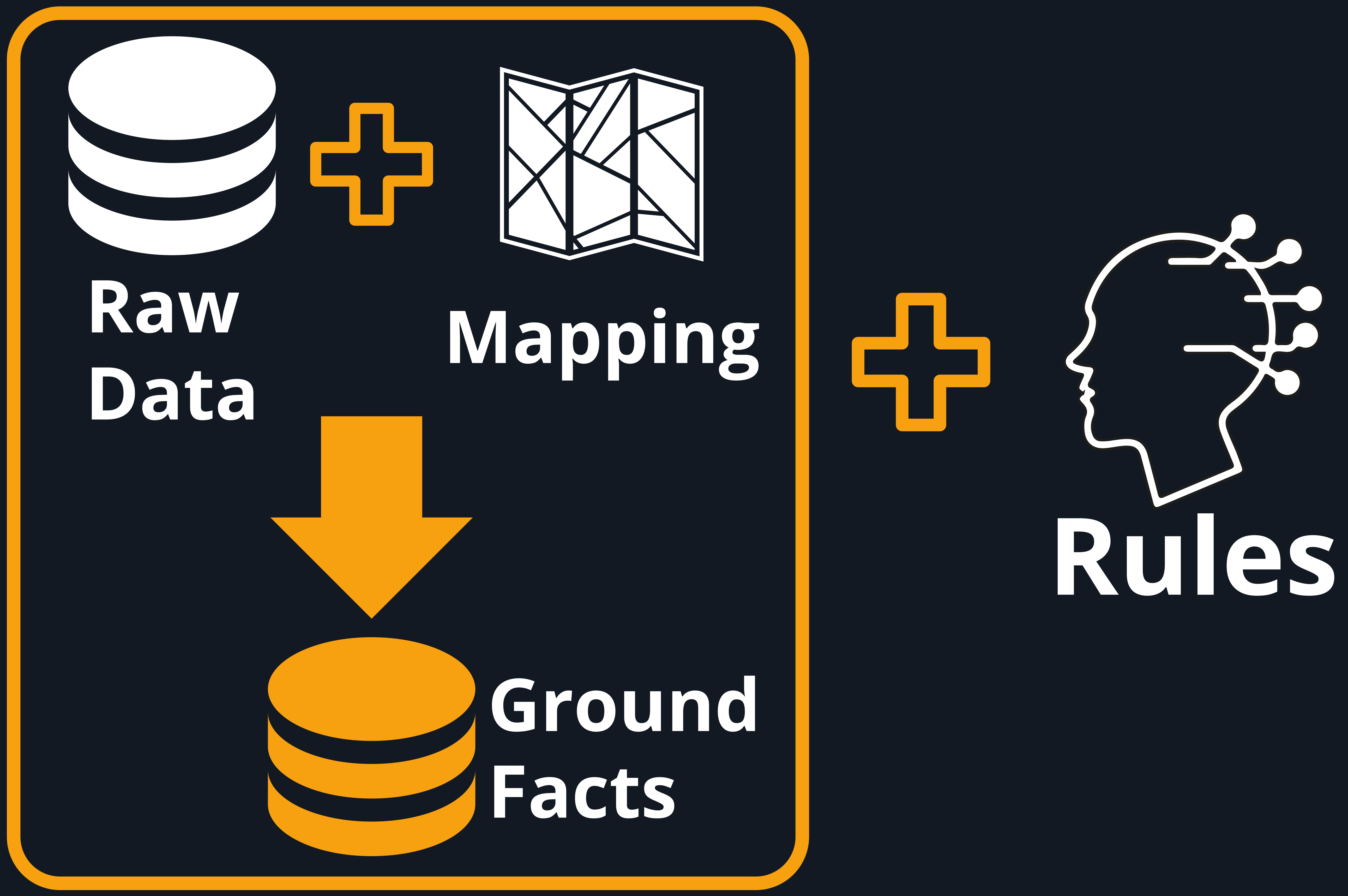
Ontology-Driven Process Mining

Ontology-Driven Process Mining

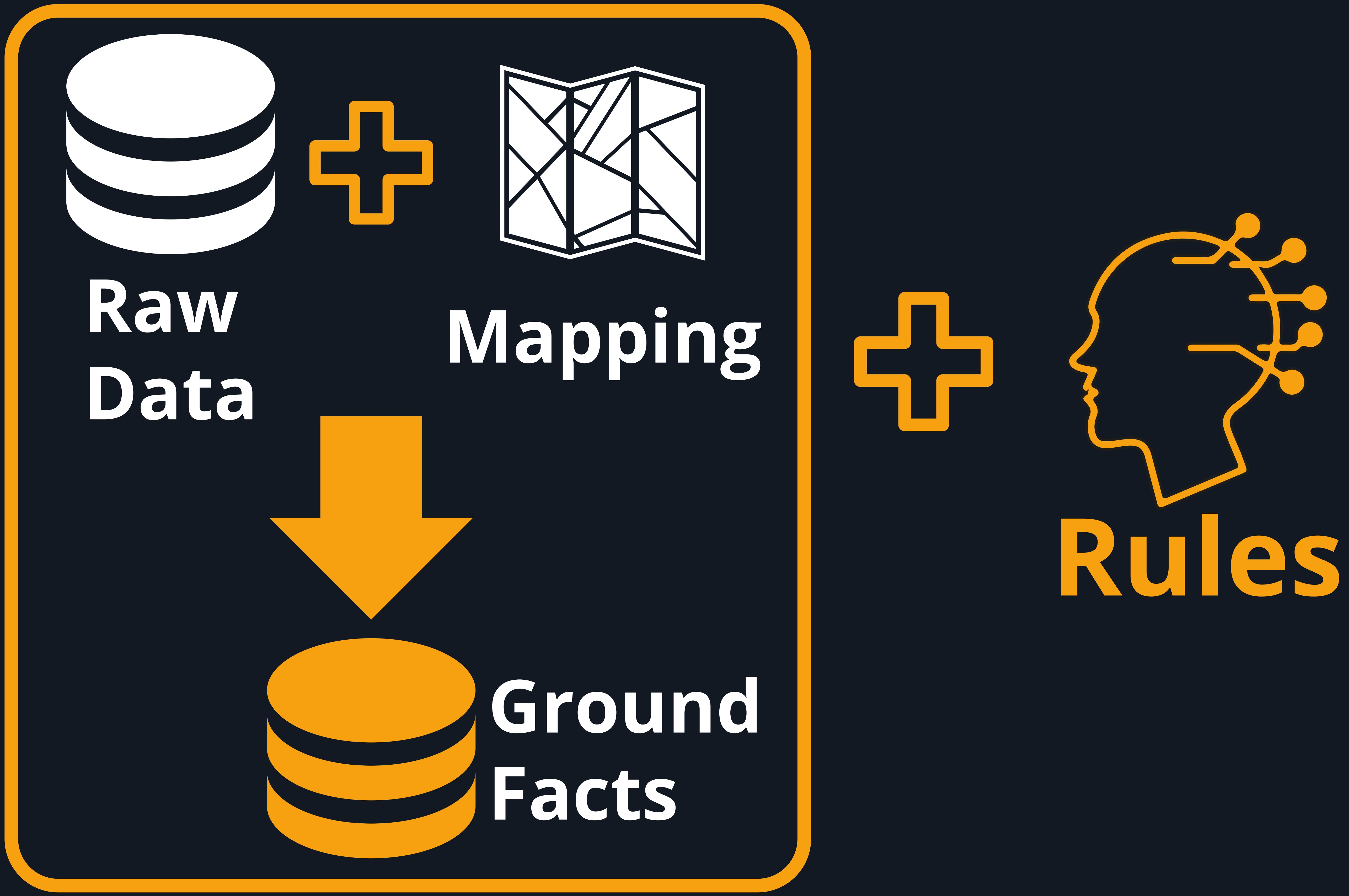
Data Rules

Ontology-Driven Process Mining

Ontology-Driven Process Mining



Ontology-Driven Process Mining



Ontology-Driven Process Mining

Ontology-Driven Process Mining

Making
Ontology-Driven Process Mining
a Reality

Making Ontology-Driven Process Mining a Reality

- We are developing a python library to implement our framework - OPAL (Ontology-driven Process Mining AnaLysis engine)

Making Ontology-Driven Process Mining a Reality

- We are developing a python library to implement our framework - OPAL (Ontology-driven Process Mining AnaLysis engine)

- Our work is benefitting the knowledge representation and process mining communities alike

Making Ontology-Driven Process Mining a Reality

- We are developing a python library to implement our framework - OPAL (Ontology-driven Process Mining AnaLysis engine)

- Our work is benefitting the knowledge representation and process mining communities alike

- We are interested in collaborating on new domains, datasets, and problems!

Making Ontology-Driven Process Mining a Reality

- We are developing a python library to implement our framework - OPAL (Ontology-driven Process Mining AnaLysis engine)

- Our work is benefitting the knowledge representation and process mining communities alike

- We are interested in collaborating on new domains, datasets, and problems!

Thank you!