Lasse Berkensträter
Xi Wang
Alois Hannen

OC-DCR Graph Discovery in OCPA

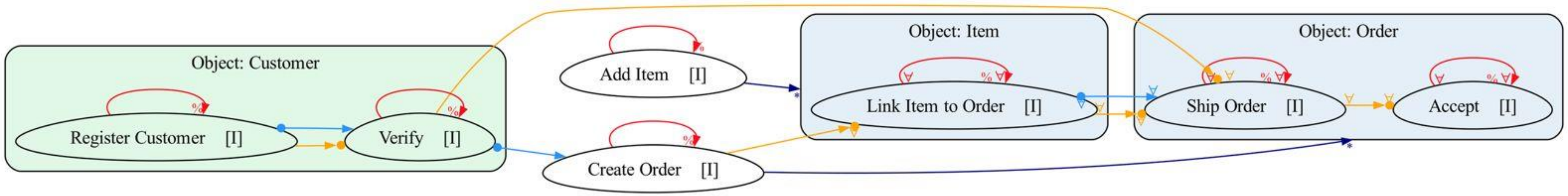
FINAL DEMO

Goal of this project

- Discovering OCDCR Graphs from Object Centric Event Logs

- Core features:
    - Data structures and visualization for (Object-Centric) DCR Graphs
    - OCDisCoveR Algorithm
    - Export to XML

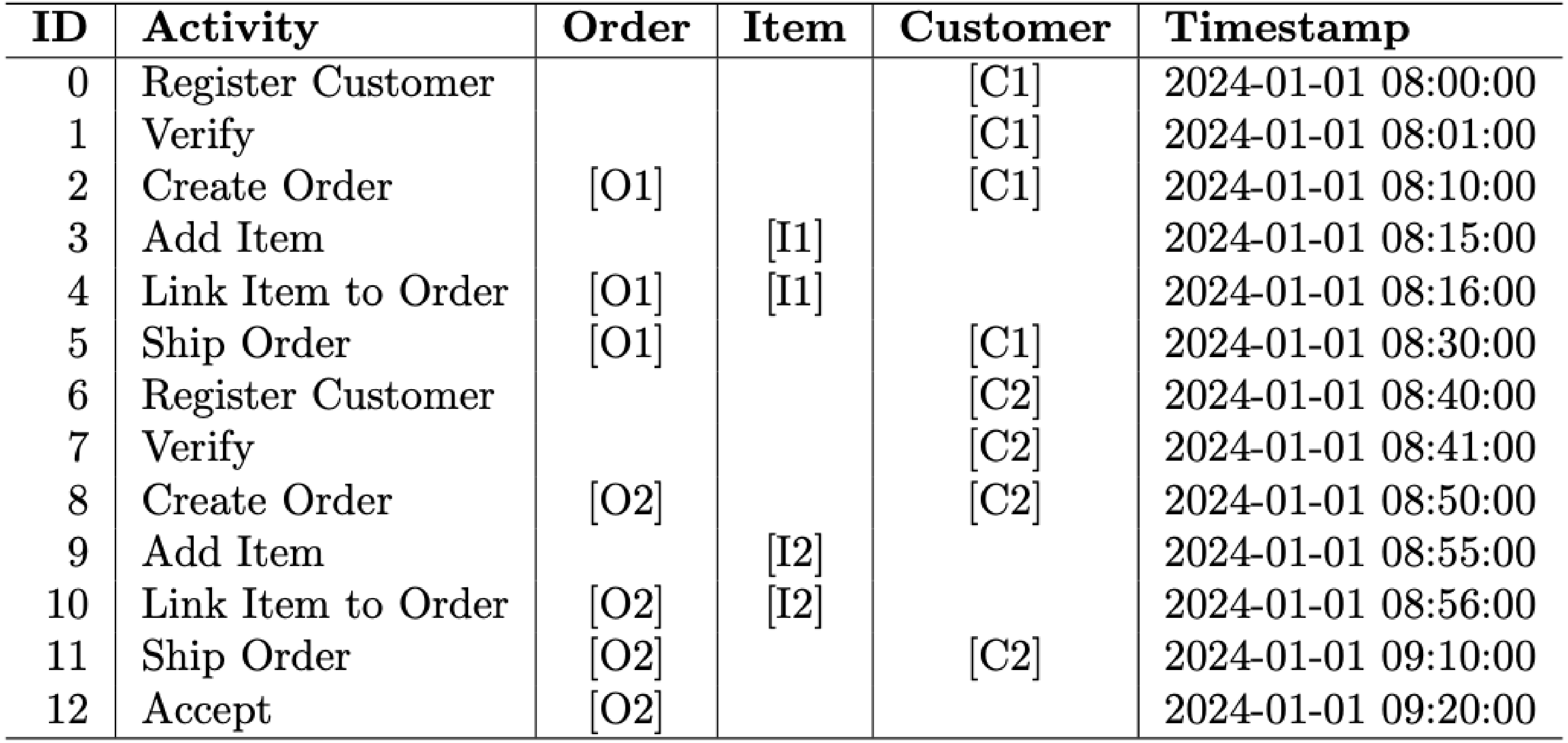
What are (OC) DCR Graphs?



Technical Stack

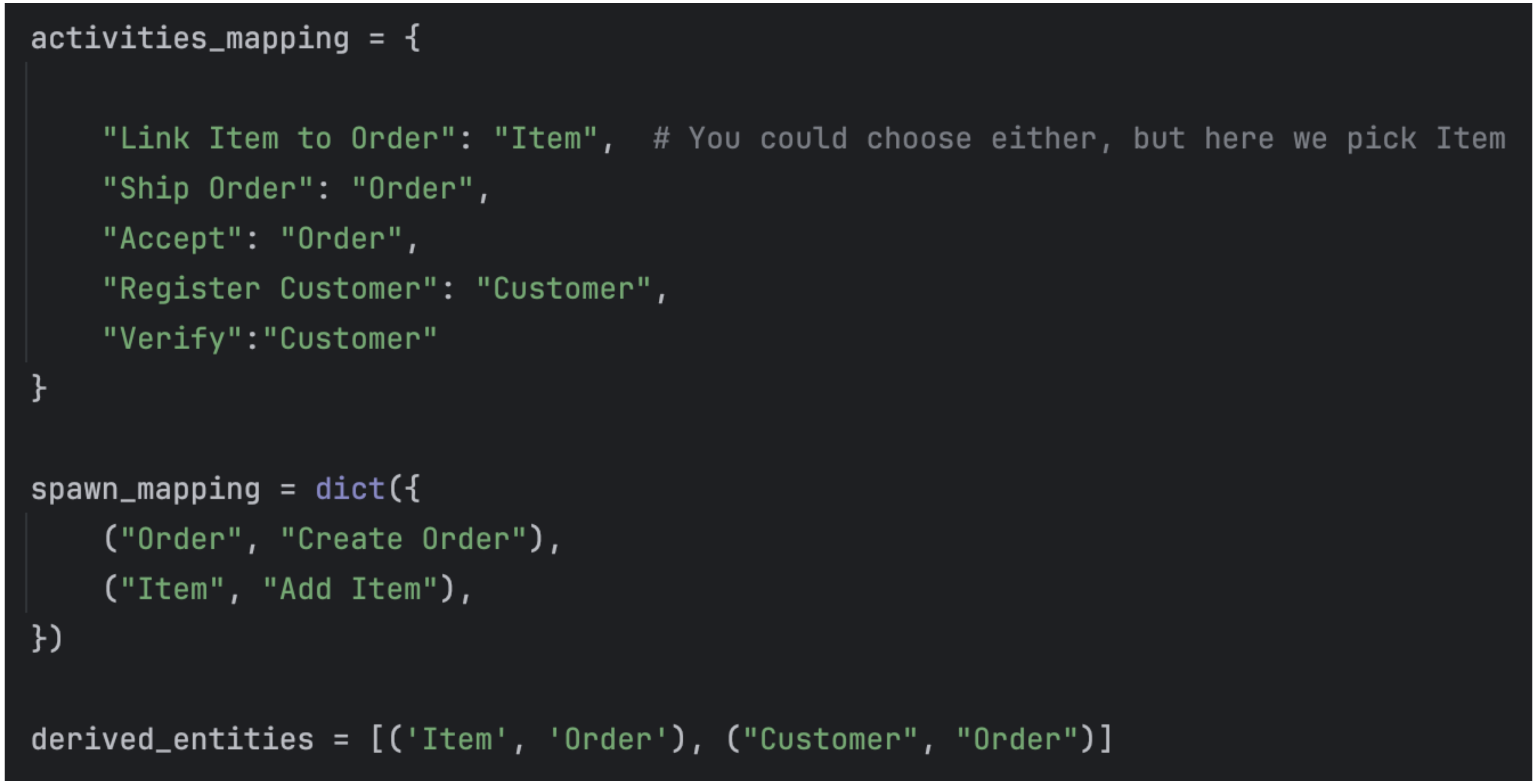
Library	Use Case
Polars Dataframes	Eventlogs
NetworkX	Heavy graph computations
lxml	XML handling
GraphViz	Visualisation capabilities
DCR4PY	PM4PY library extention for DCR Graphs

Discovery of an OCDCR Graph - Example



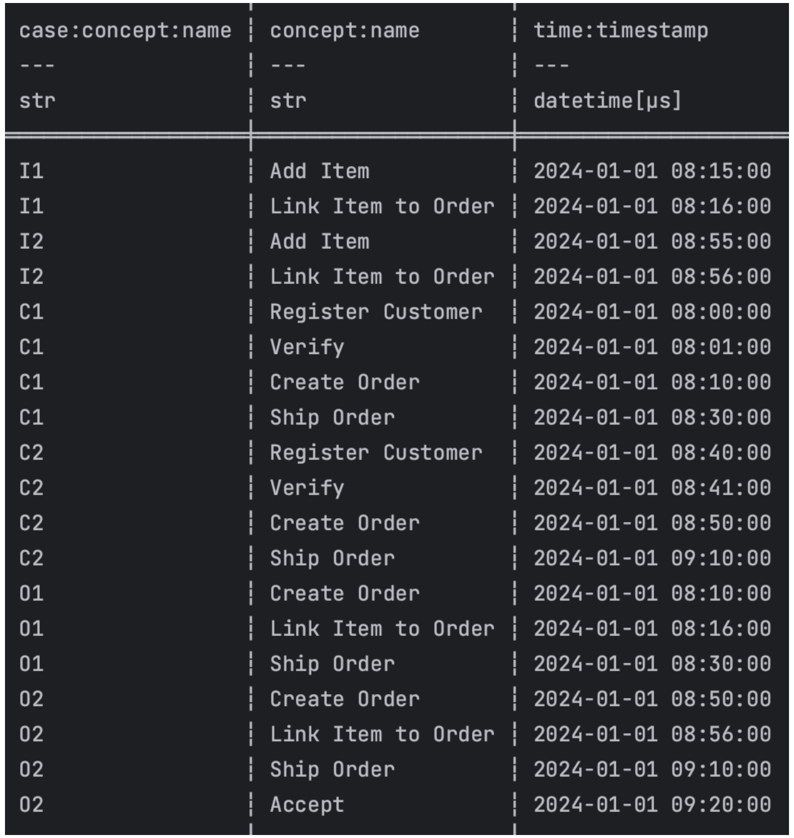
ID	Activity	Order	Item	Customer	Timestamp
0	Register Customer			[C1]	2024-01-01 08:00:00
1	Verify			[C1]	2024-01-01 08:01:00
2	Create Order	[O1]		[C1]	2024-01-01 08:10:00
3	Add Item		[I1]		2024-01-01 08:15:00
4	Link Item to Order	[O1]	[I1]		2024-01-01 08:16:00
5	Ship Order	[O1]		[C1]	2024-01-01 08:30:00
6	Register Customer			[C2]	2024-01-01 08:40:00
7	Verify			[C2]	2024-01-01 08:41:00
8	Create Order	[O2]		[C2]	2024-01-01 08:50:00
9	Add Item		[I2]		2024-01-01 08:55:00
10	Link Item to Order	[O2]	[I2]		2024-01-01 08:56:00
11	Ship Order	[O2]		[C2]	2024-01-01 09:10:00
12	Accept	[O2]			2024-01-01 09:20:00

User defined Input



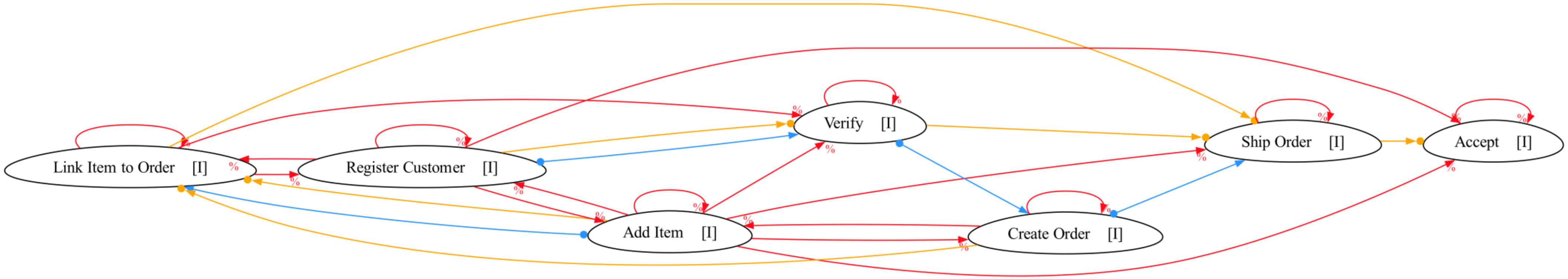
```
activities_mapping = {

    "Link Item to Order": "Item",  # You could choose either, but here we pick Item
    "Ship Order": "Order",
    "Accept": "Order",
    "Register Customer": "Customer",
    "Verify":"Customer"
}

spawn_mapping = dict({
    ("Order", "Create Order"),
    ("Item", "Add Item"),
})

derived_entities = [('Item', 'Order'), ("Customer", "Order")]
```

Abstraction of the log

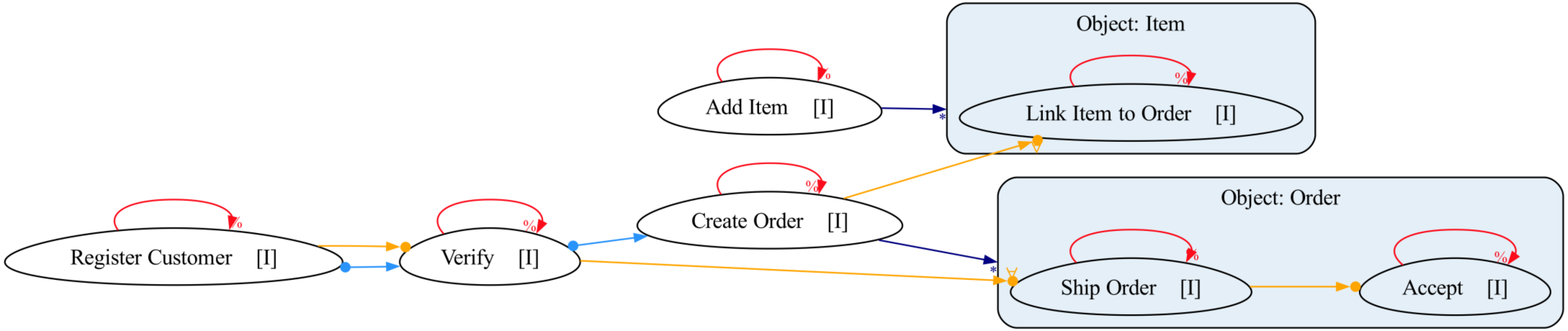


```
case:concept:name ┆ concept:name       ┆ time:timestamp
---               ┆ ---                ┆ ---
str               ┆ str                ┆ datetime[μs]
══════════════════╪════════════════════╪═════════════════════
I1                ┆ Add Item           ┆ 2024-01-01 08:15:00
I1                ┆ Link Item to Order ┆ 2024-01-01 08:16:00
I2                ┆ Add Item           ┆ 2024-01-01 08:55:00
I2                ┆ Link Item to Order ┆ 2024-01-01 08:56:00
C1                ┆ Register Customer  ┆ 2024-01-01 08:00:00
C1                ┆ Verify             ┆ 2024-01-01 08:01:00
C1                ┆ Create Order       ┆ 2024-01-01 08:10:00
C1                ┆ Ship Order         ┆ 2024-01-01 08:30:00
C2                ┆ Register Customer  ┆ 2024-01-01 08:40:00
C2                ┆ Verify             ┆ 2024-01-01 08:41:00
C2                ┆ Create Order       ┆ 2024-01-01 08:50:00
C2                ┆ Ship Order         ┆ 2024-01-01 09:10:00
O1                ┆ Create Order       ┆ 2024-01-01 08:10:00
O1                ┆ Link Item to Order ┆ 2024-01-01 08:16:00
O1                ┆ Ship Order         ┆ 2024-01-01 08:30:00
O2                ┆ Create Order       ┆ 2024-01-01 08:50:00
O2                ┆ Link Item to Order ┆ 2024-01-01 08:56:00
O2                ┆ Ship Order         ┆ 2024-01-01 09:10:00
O2                ┆ Accept             ┆ 2024-01-01 09:20:00
```

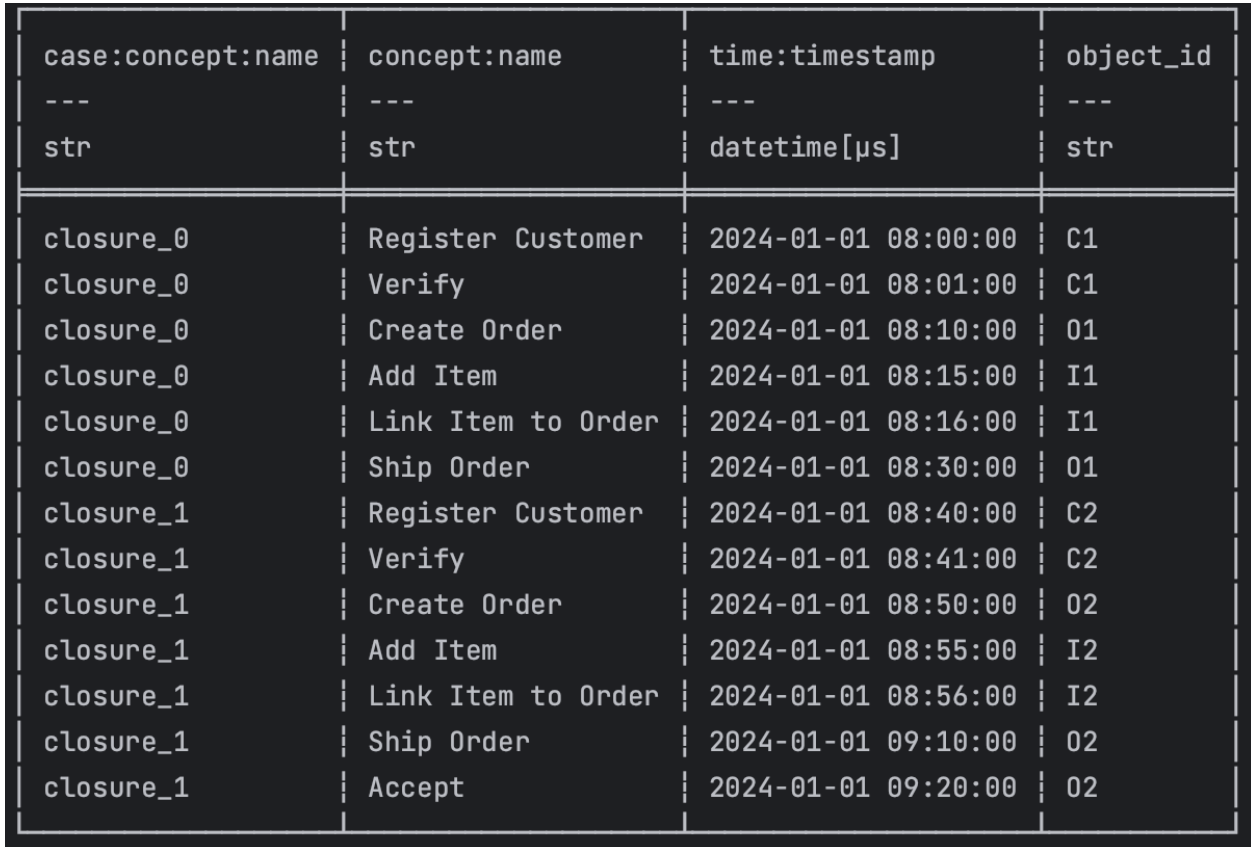
Initial Constraints Discovery



Translation to OC-DCR Structure

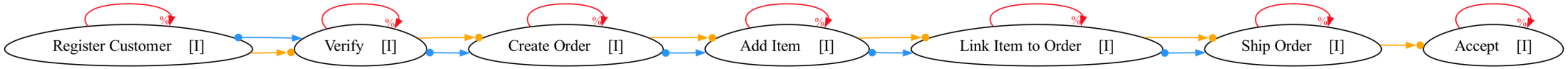


Computation of transitive closure

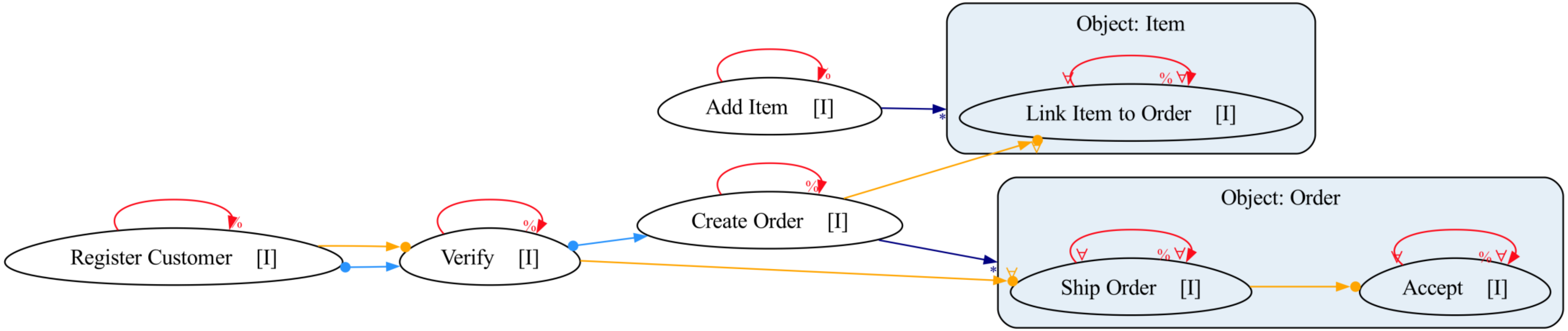


```
┌───────────────────┬────────────────────┬─────────────────────┬───────────┐
│ case:concept:name ┆ concept:name       ┆ time:timestamp      ┆ object_id │
│ ---               ┆ ---                ┆ ---                 ┆ ---       │
│ str               ┆ str                ┆ datetime[μs]        ┆ str       │
╞═══════════════════╪════════════════════╪═════════════════════╪═══════════╡
│ closure_0         ┆ Register Customer  ┆ 2024-01-01 08:00:00 ┆ C1        │
│ closure_0         ┆ Verify             ┆ 2024-01-01 08:01:00 ┆ C1        │
│ closure_0         ┆ Create Order       ┆ 2024-01-01 08:10:00 ┆ O1        │
│ closure_0         ┆ Add Item           ┆ 2024-01-01 08:15:00 ┆ I1        │
│ closure_0         ┆ Link Item to Order ┆ 2024-01-01 08:16:00 ┆ I1        │
│ closure_0         ┆ Ship Order         ┆ 2024-01-01 08:30:00 ┆ O1        │
│ closure_1         ┆ Register Customer  ┆ 2024-01-01 08:40:00 ┆ C2        │
│ closure_1         ┆ Verify             ┆ 2024-01-01 08:41:00 ┆ C2        │
│ closure_1         ┆ Create Order       ┆ 2024-01-01 08:50:00 ┆ O2        │
│ closure_1         ┆ Add Item           ┆ 2024-01-01 08:55:00 ┆ I2        │
│ closure_1         ┆ Link Item to Order ┆ 2024-01-01 08:56:00 ┆ I2        │
│ closure_1         ┆ Ship Order         ┆ 2024-01-01 09:10:00 ┆ O2        │
│ closure_1         ┆ Accept             ┆ 2024-01-01 09:20:00 ┆ O2        │
└───────────────────┴────────────────────┴─────────────────────┴───────────┘
```

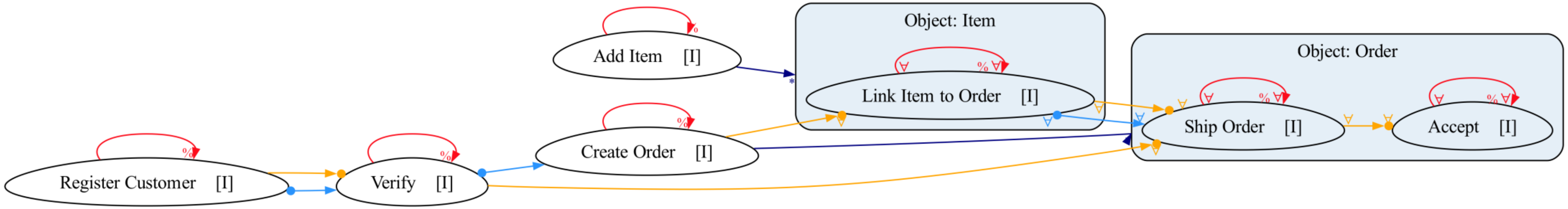
Discovery on transitive Closure



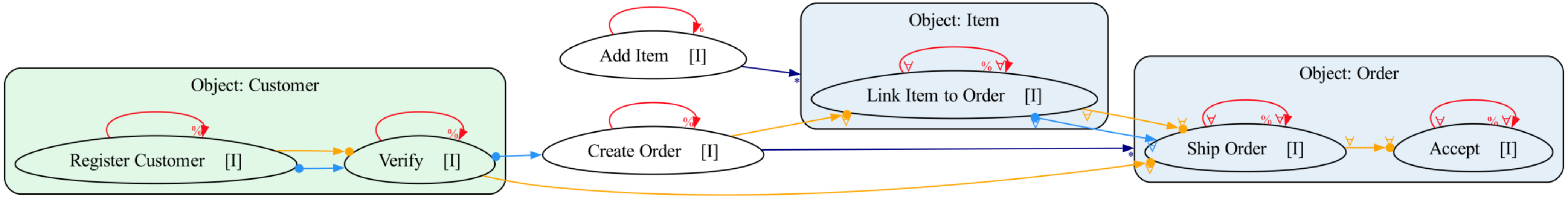
Many to Many Excludes



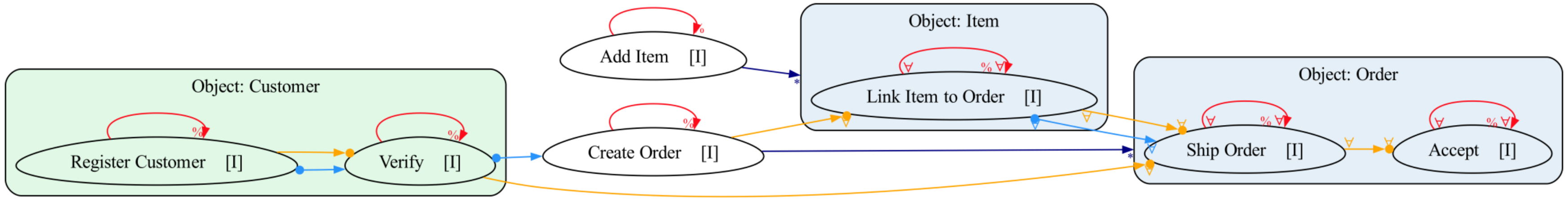
Many to Many Conditions and Responses

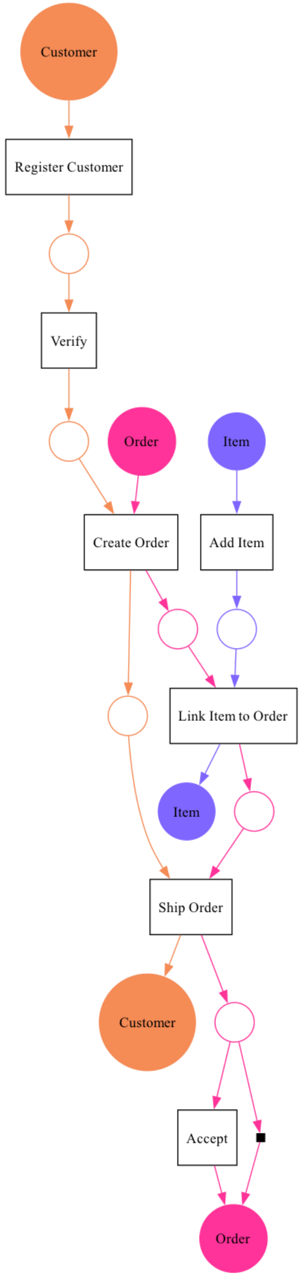


Final Results



Comparison to OC Petri Net

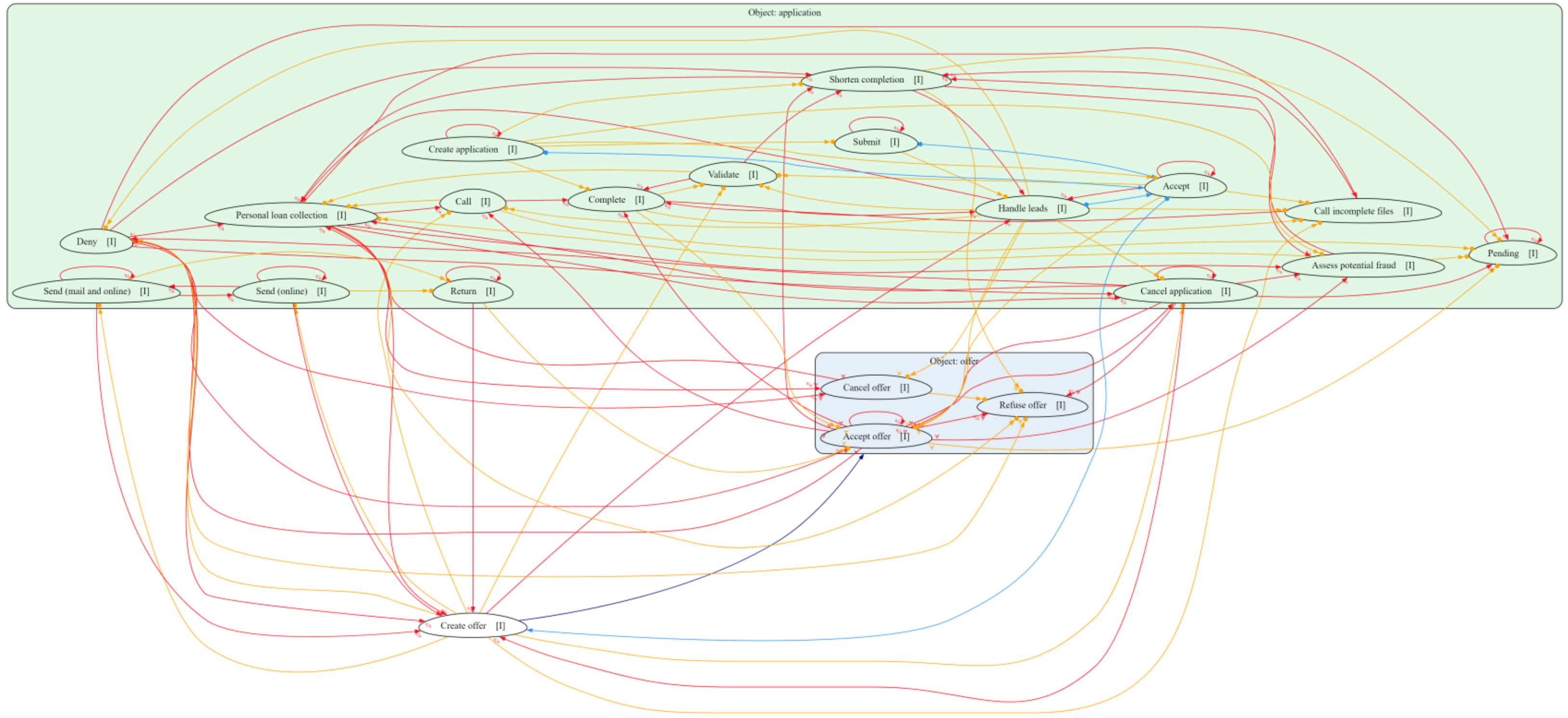




Advantages

- Extension of the OCPA library to discover models that capture object lifecycles and synchronization:
  - Differentiation between spawned and static objects
  - Adds lifecycle, one-to-many and many-to-many constraints

Application on real world dataset



Limitations

- Activities have to be mapped to at most one object type
- User still has to define the object types, activity mapping, etc. by hand
- Nested graphs