run_1

bauplan

together.ai

Trustworthy AI in the Agentic Lakehouse: from Concurrency to Governance

Jacopo Tagliabue, Federico Bianchi, Ciro Greco

Q1: How can we trust agents *in the lakehouse*?
We shouldn't need trust at all.

Q2: How do we build one that doesn't require trust?
Two lessons from the databases literature.

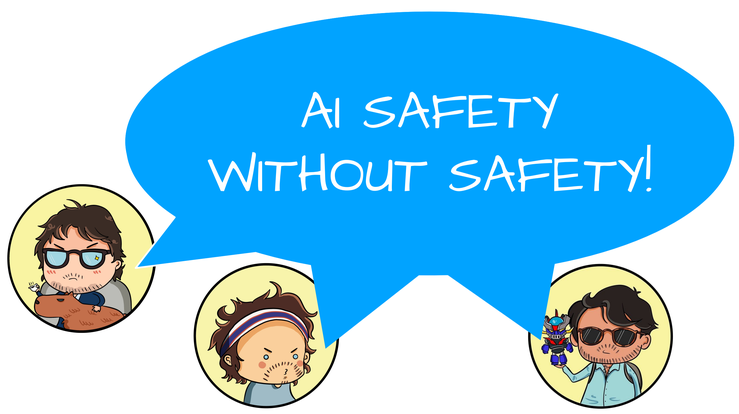
abstractions

FaaS

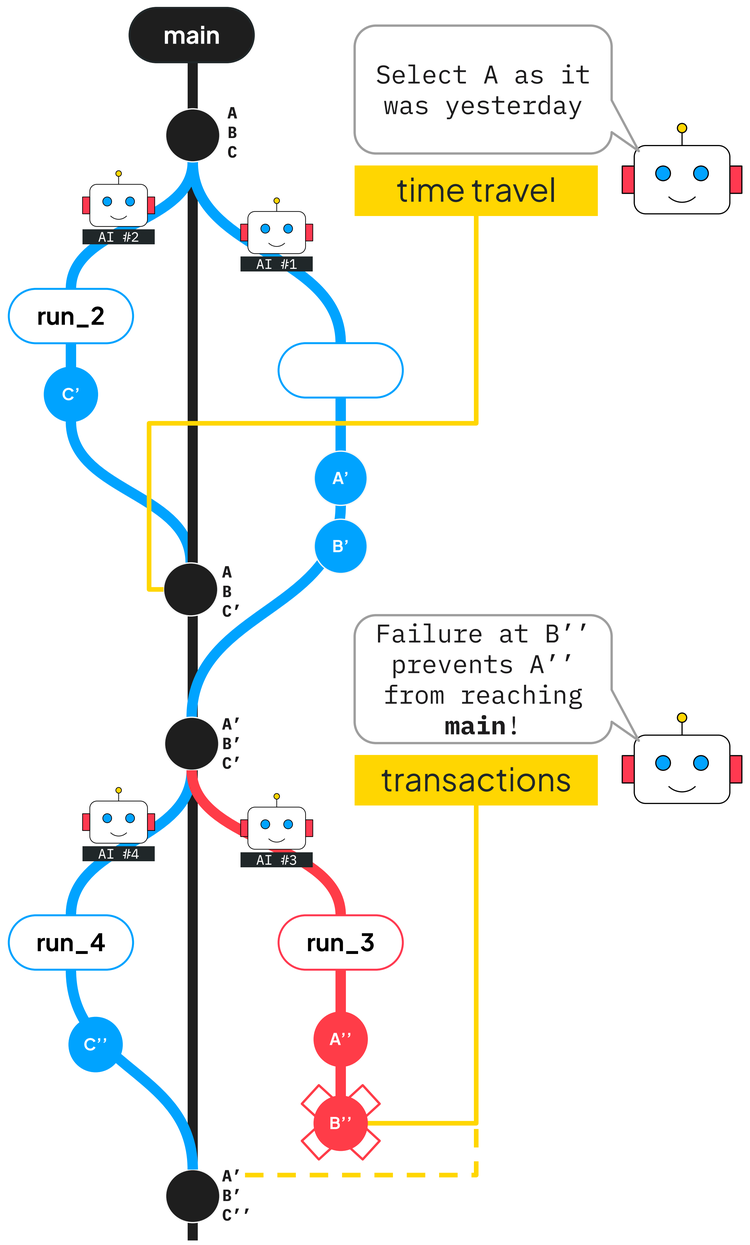
Declarative I/O

```python
@bauplan.model(materialize=True)
@bauplan.python(pip={'polars': '0.8.0'})
def parent(
    input='nyc_taxi',
    filter="date='2022-12-15'"
):
    # some code here
    return parent_table

@bauplan.model(materialize=True)
@bauplan.python(pip={'polars': '1.3.0'})
def child(input='parent'):
    # some code here
    return child_table
```



data versioning



- Correctness under concurrency is the hard problem for lakehouse agents.
- Good APIs encourage correct behavior and prevent unsafe actions.
- Once correctness is enforced, governance reduces to API access control.

CONCERN	MODE	ABSTRACTION
Trust	Data	Declarative I/O
Trust	Code	FaaS
Correctness	Data	Transactions
Correctness	Code	PRs

Read the paper