

Canopy: An End-to-End Performance Tracing And Analysis System





Performance Got Worse – Why?



Performance Got Worse – Why?

• •

Performance Got Worse – Why?

Performance Got Worse – Why?

Performance Got Worse – Why?

Server?

Did the quantity of resources change, or our packaging?





Query



- Facebook's end-to-end tracing and analysis system

Canopy in use at Facebook

- **Usage:** 2 years, ~1B traces/day, from production systems
- **Domains:** Browsers, mobile applications, services
- **Tracing:** requests, user interactions, stutters...
- **Collected data**

Prior Approaches

- Distributed tracing [Xtrace, Dapper, ...]
- Black Box Inference [Mystery Machine, ...]
-

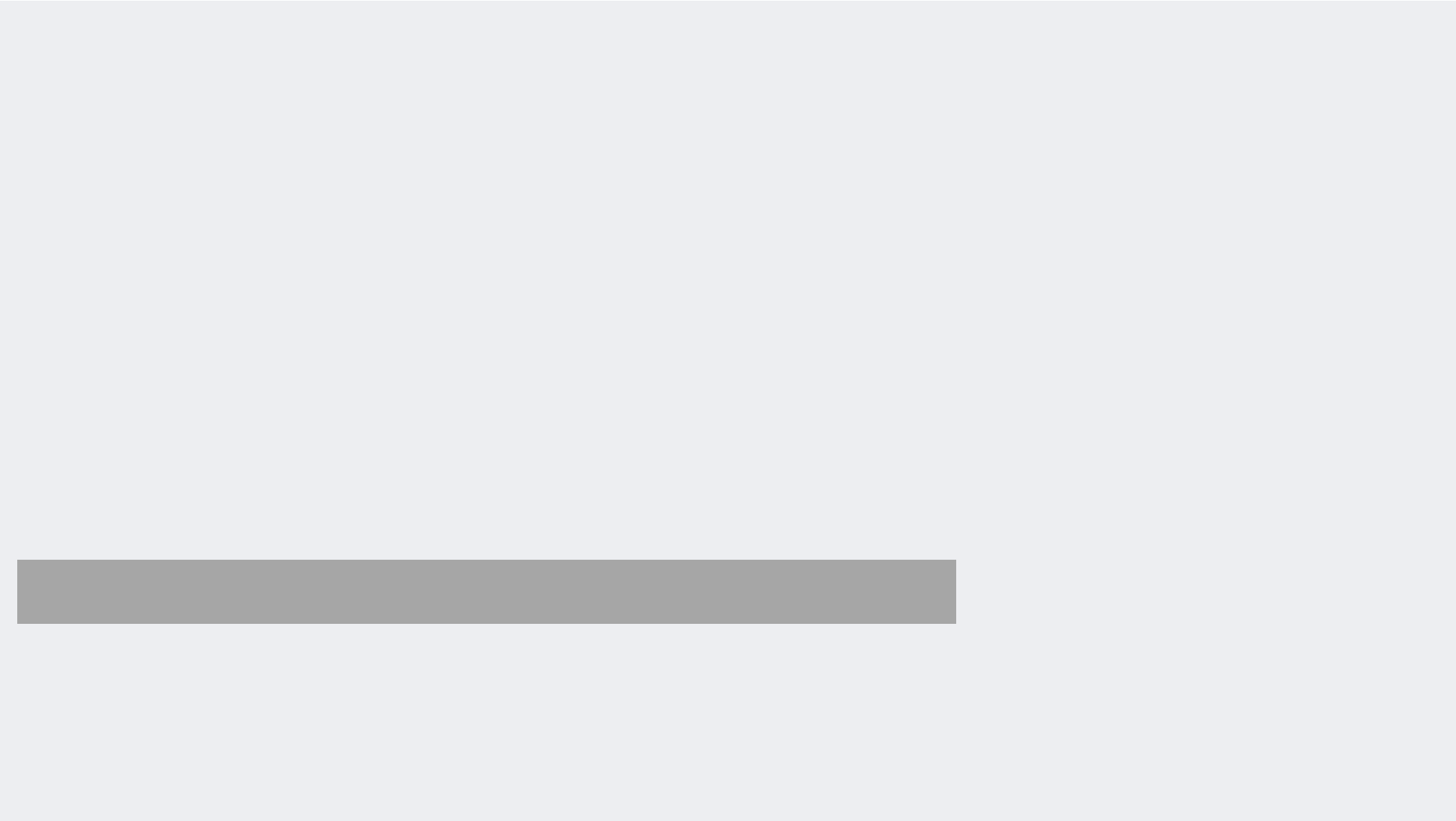
Instrumentation

Sampling

Instrumentation



Canopy Processing Pipeline



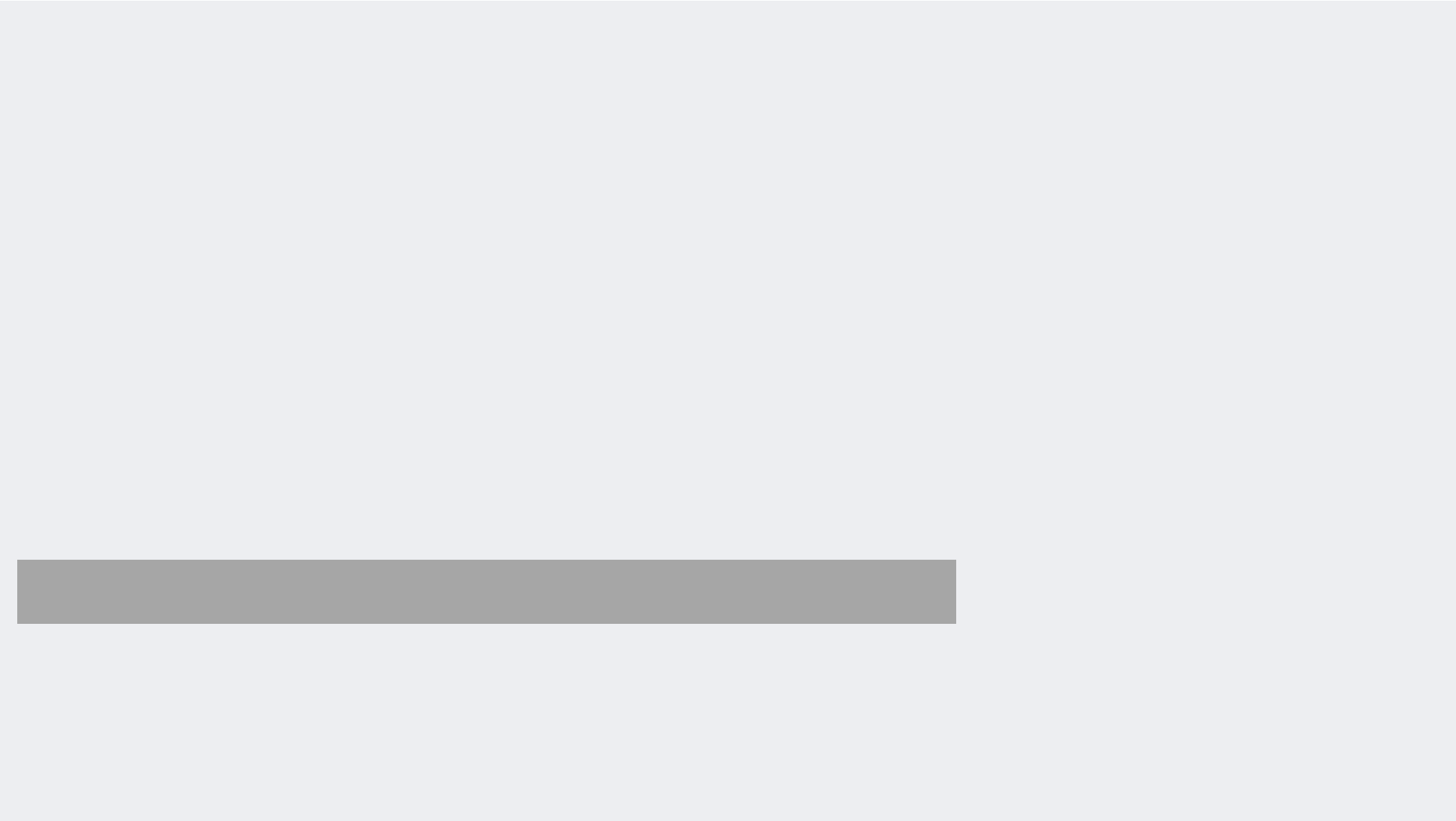
Feature Extraction Examples

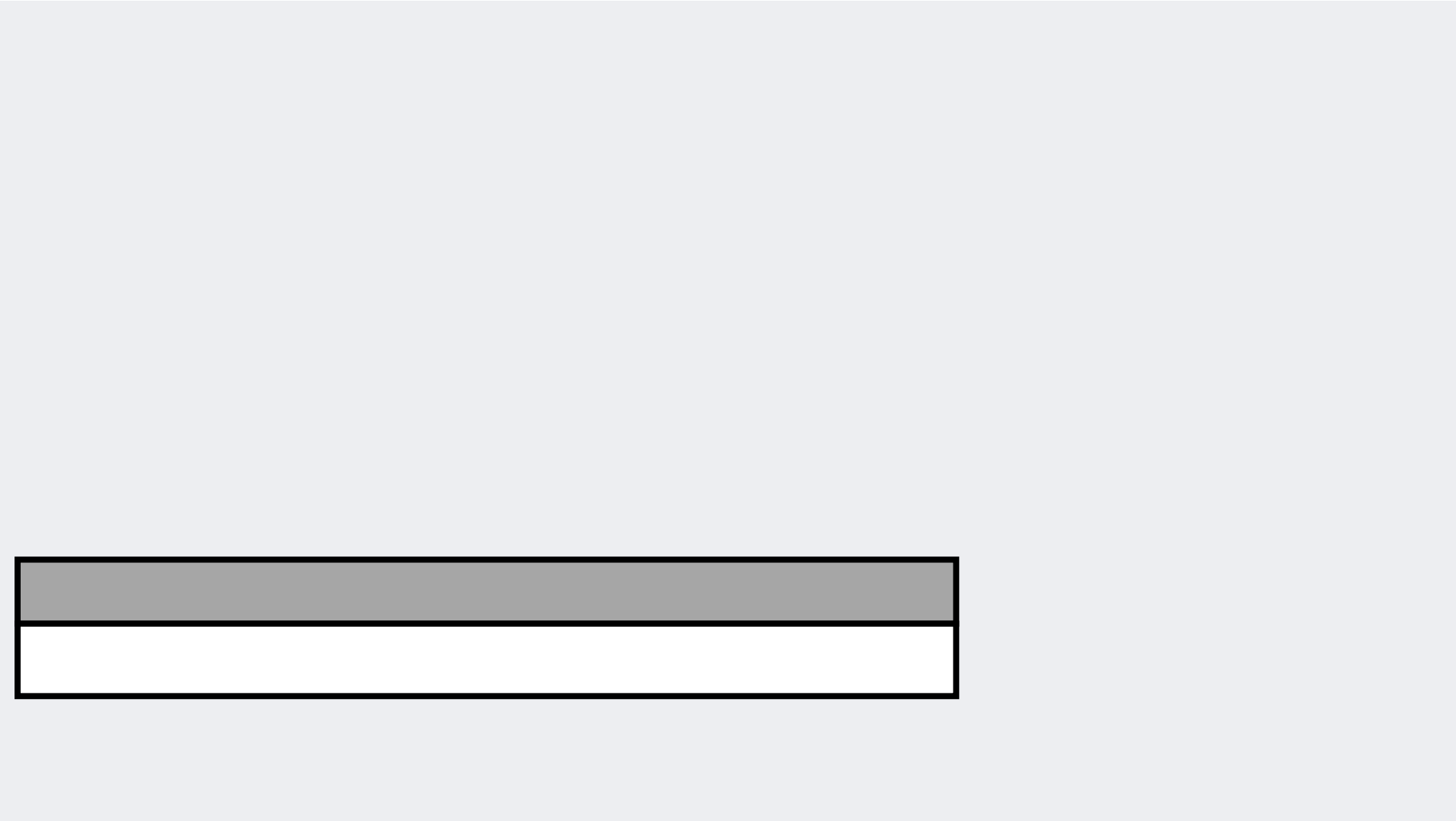


| trace_id | |
|----------|--|
| FAC3B00C | |

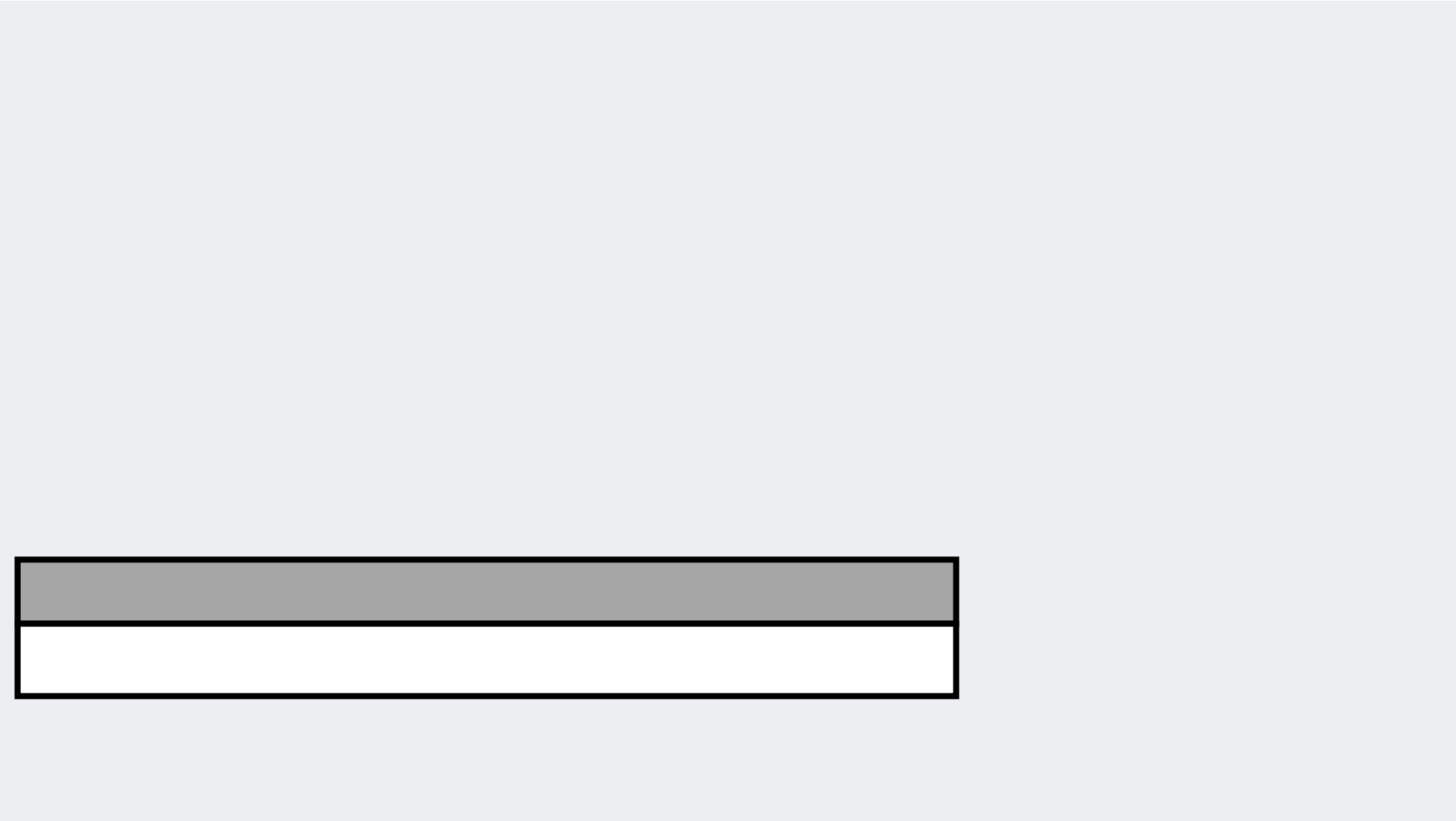
trace_id

FAC3B00C





| trace_id | duration | |
|----------|----------|--|
| FAC3B00C | 2500 | |



Feature Extraction Examples

| trace_id | duration | critpath_resource | ... |
|----------|----------|-------------------|-----|
| FAC3B00C | 2500 | 600 | ... |



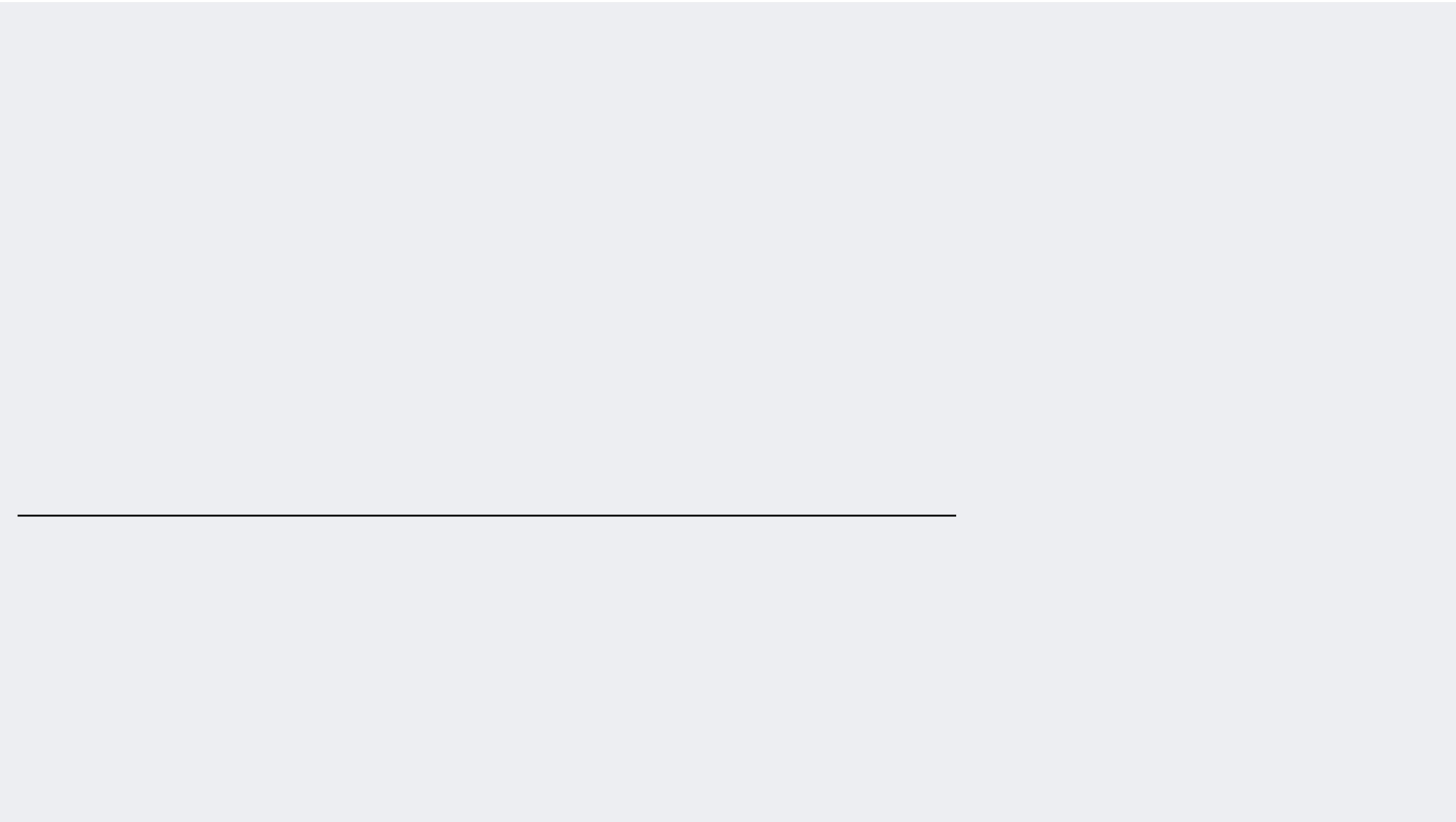
What was
resource time
along the critical
path for each

| trace_id | |
|----------|--|
| | |

was What

was What

| trace_id | |
|----------|--|
| | |











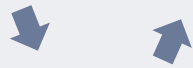
Features at Facebook

- Domain-specific, customized
- Currently: several thousand features across >100 datasets
- Currently: expressed as DSL, Python-based extraction expressions are in testing

Instrumenting Systems

- Developers can choose what / how to represent performance information and structure
- Instrumentation becomes events, which are later reconstructed into a model
- Tradeoff: Flexibility in system representation vs common concepts for analysis

Execution Models May Vary



Execution Models May Vary







Backend Considerations

Backend Considerations

Case Study: Causal Ordering





Case Study: Causal Ordering



In Summary

