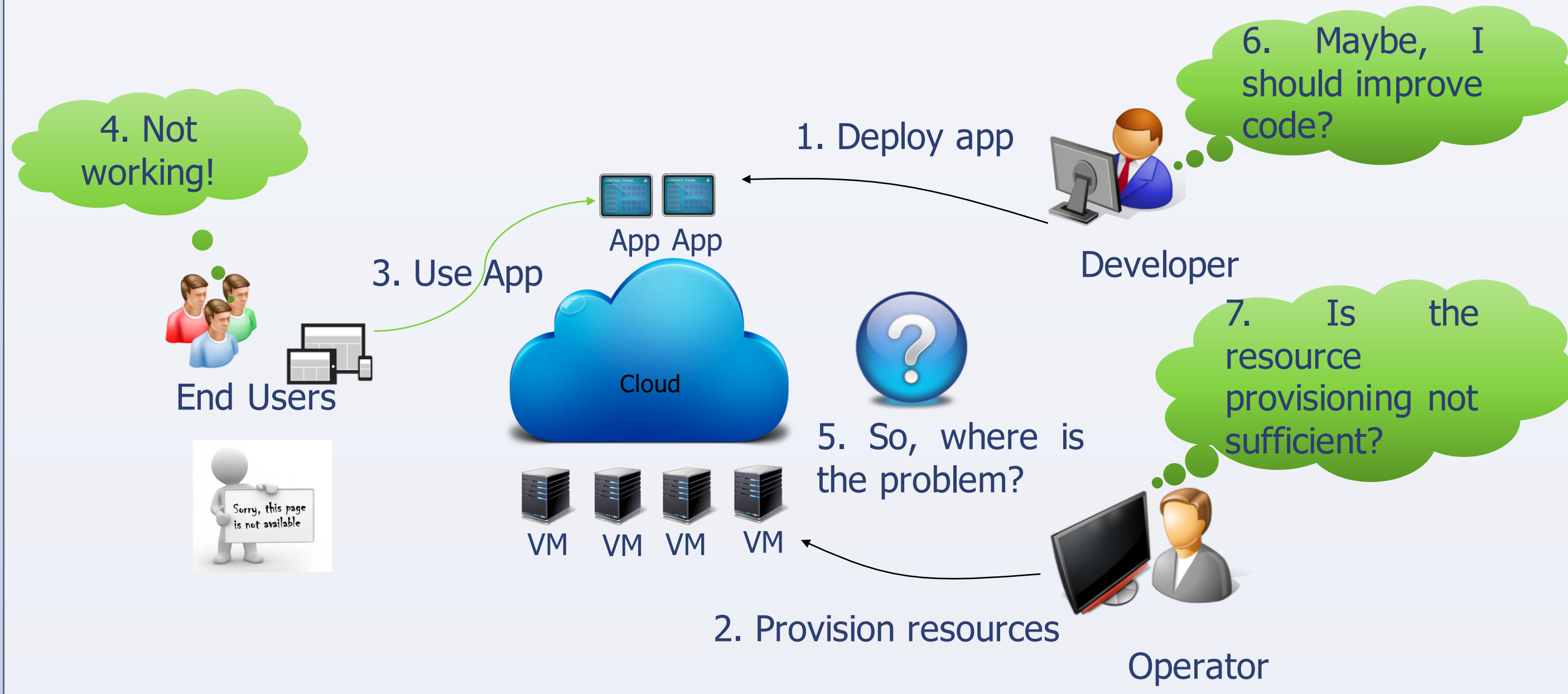
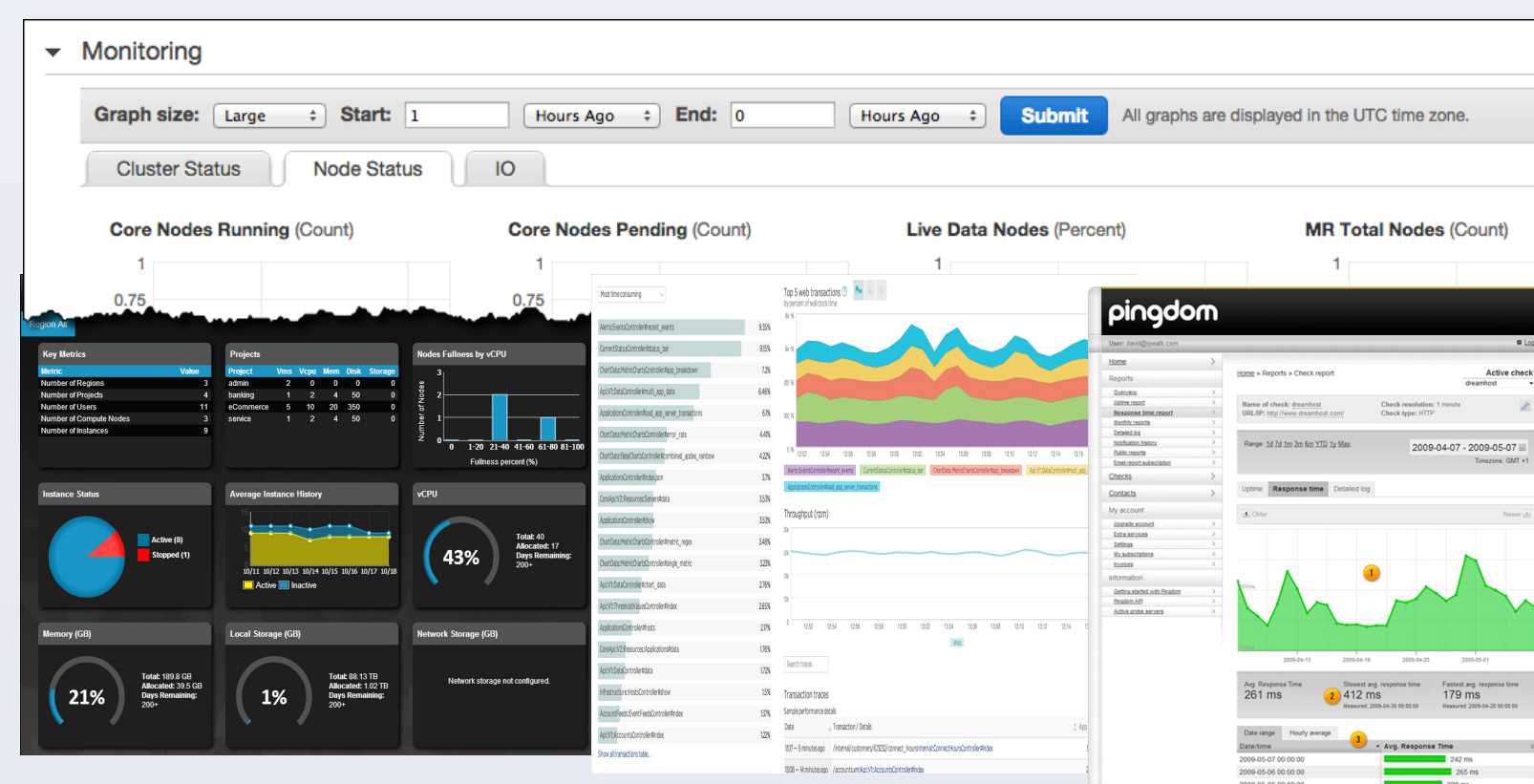


Motivation

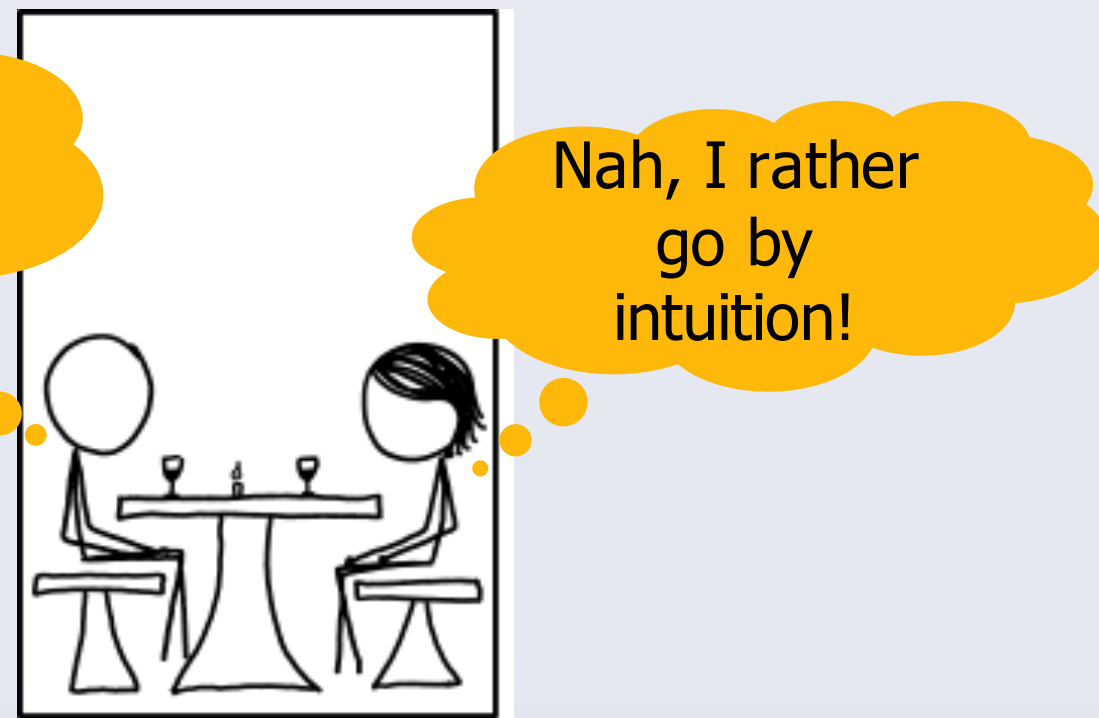


Problem Statement

There are several Cloud Application Monitoring tools such as Amazon cloudwatch, new relic etc, providing massive log information as graphs and dashboards. But do the DevOps really use it?

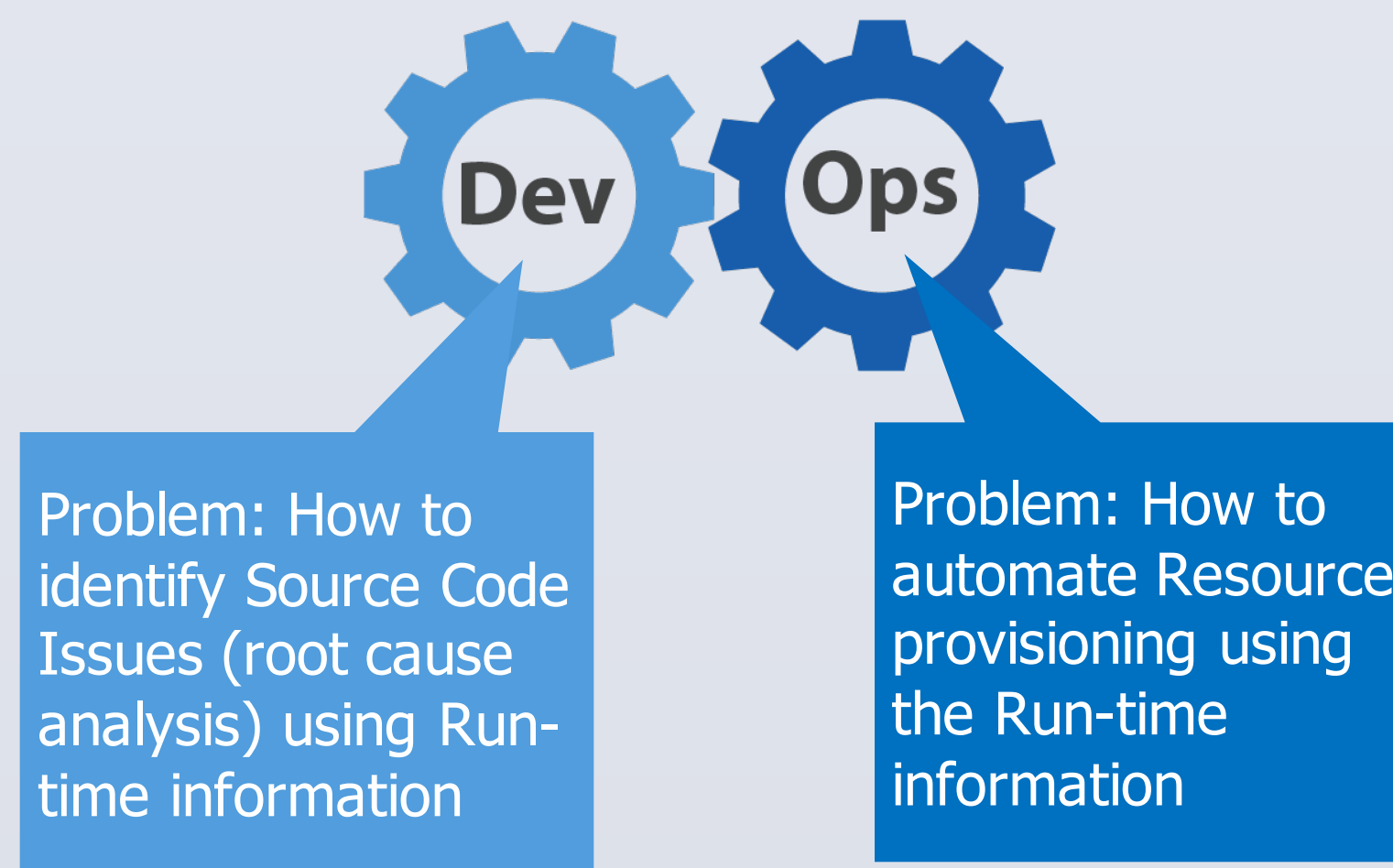


Do you really look into these Metrics to solve production issues?

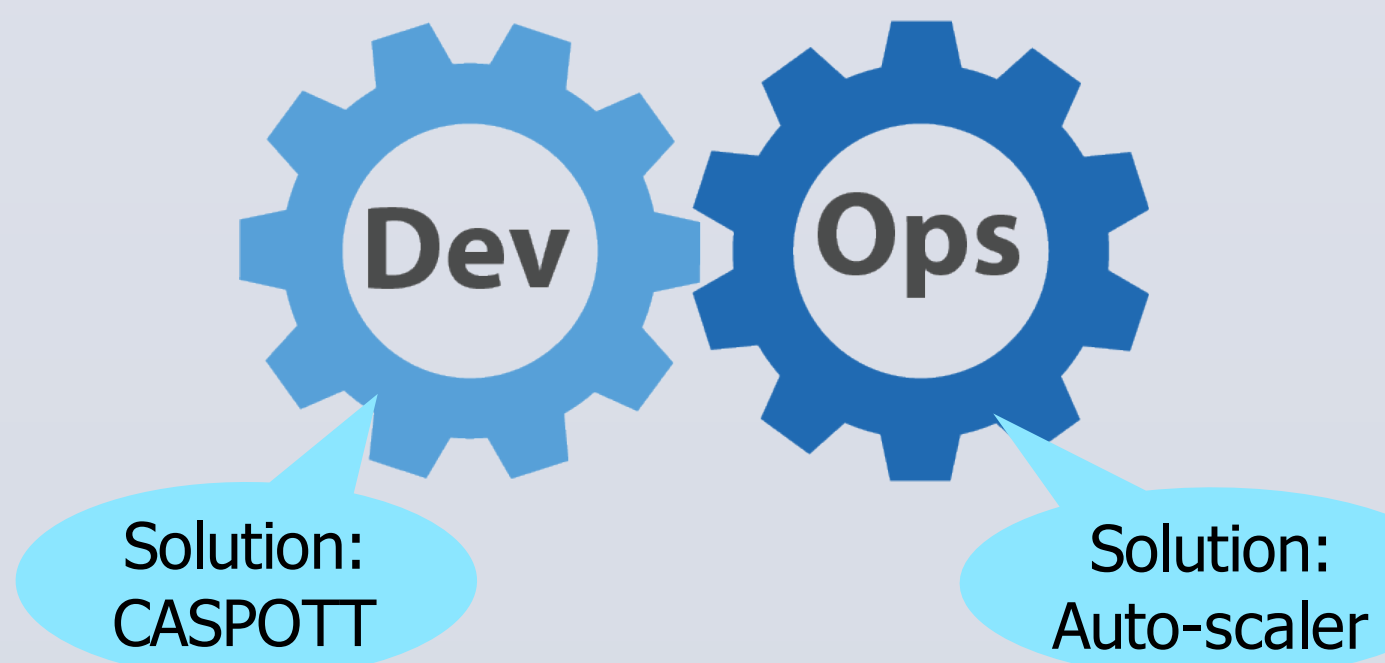


What is Feedback Driven Development?

Applications running on Cloud environments generate run-time production data. Making this run-time feedback available to DevOps in an accessible format to adapt accordingly, is known as Feedback Driven Development (FDD).

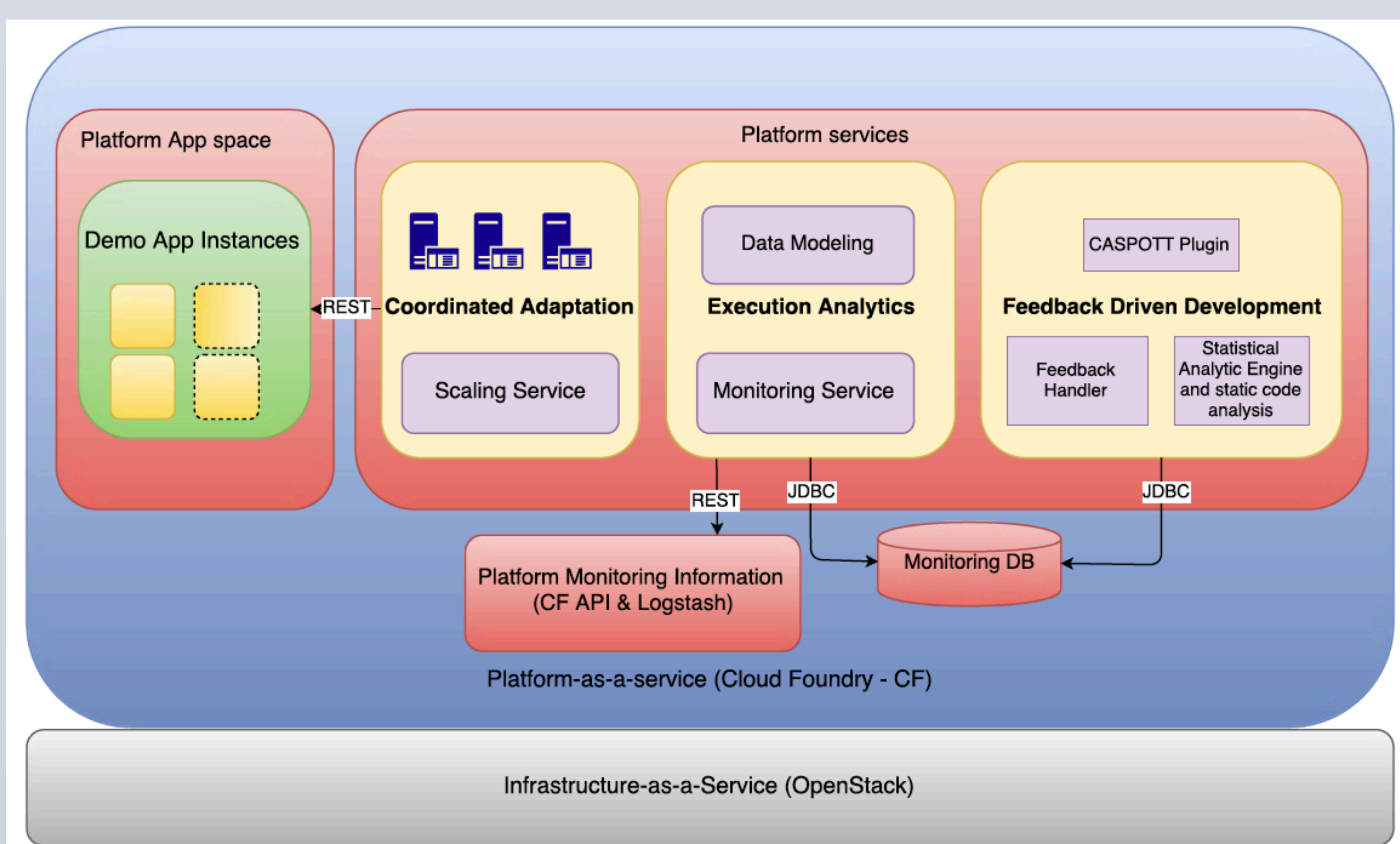


Proposed Solution: DO-AS-CASPOTT

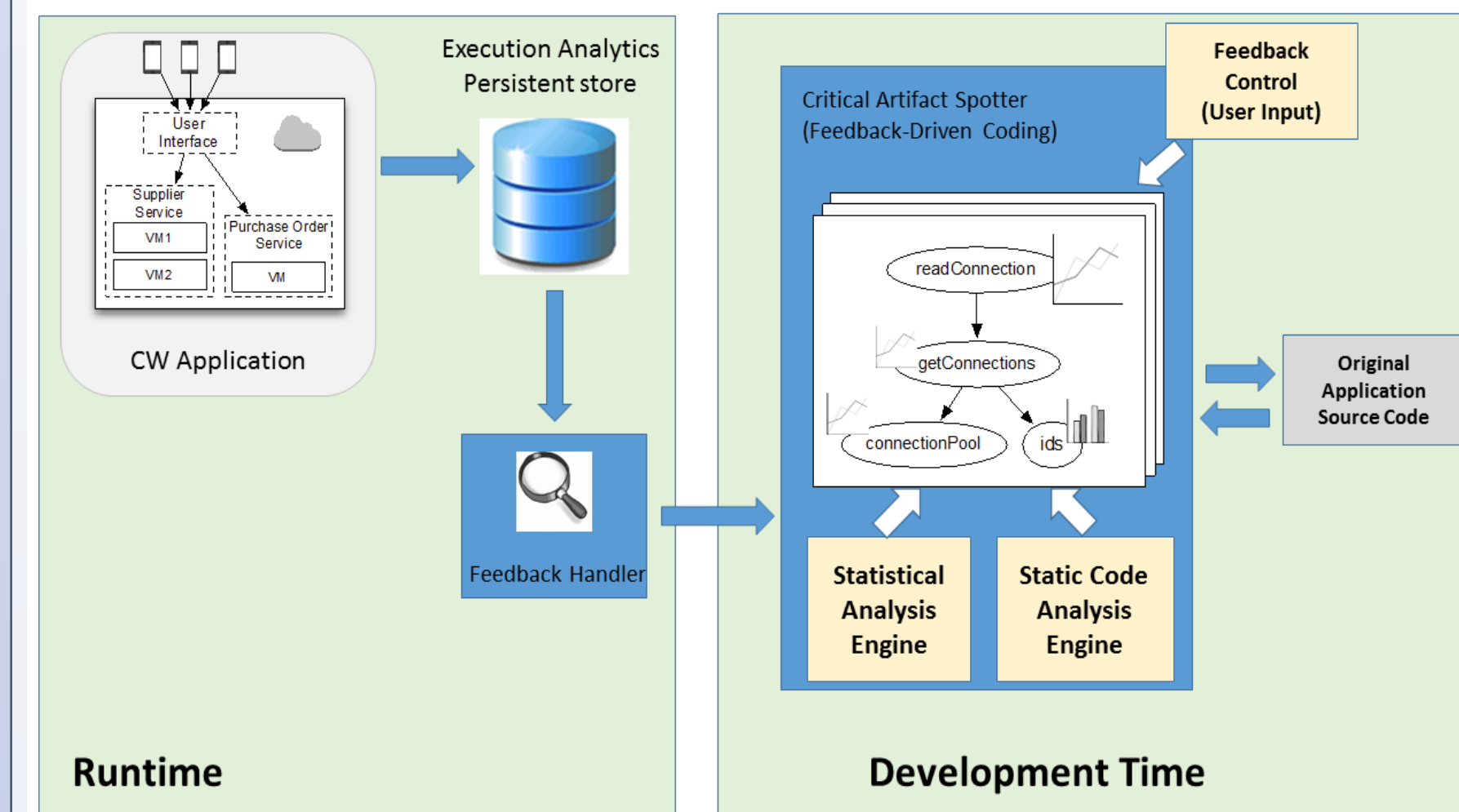


Solution: Combination of plug-ins that support the Developer (CASPOTT) and Operator (Auto-scaler)

Design of DO-AS-CASPOTT



CASPOTT

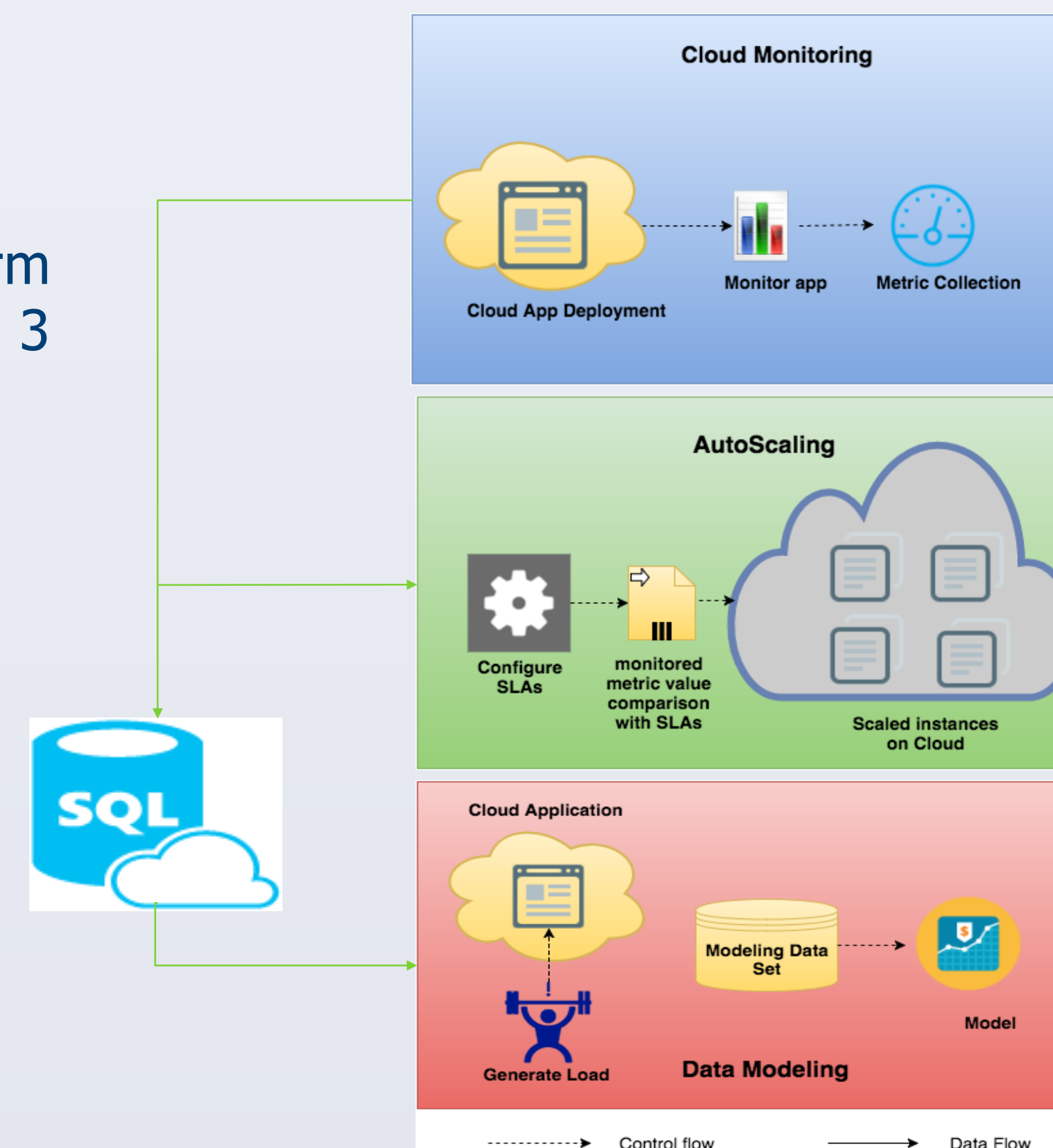


Feedback Handler
- Prepares the raw logs for further processing
Statistical Analysis Engine
- Analyzes and aggregates the collected information.
Static Code Analysis
- Mapping between code and feedback

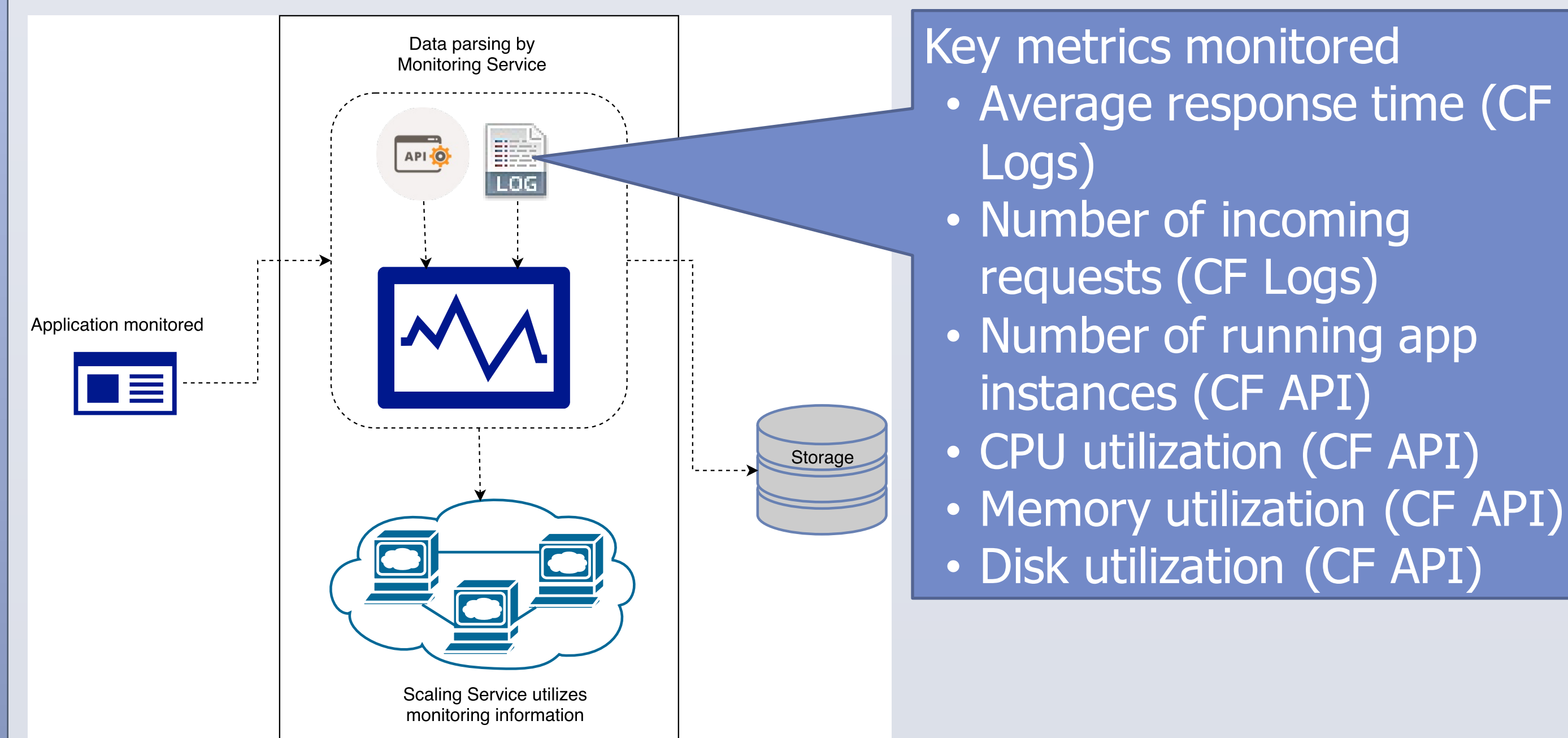
Auto-Scaler

Auto-scaling at platform level that consists of 3 components:

- Monitoring service
- Scaling service
- Data modeling

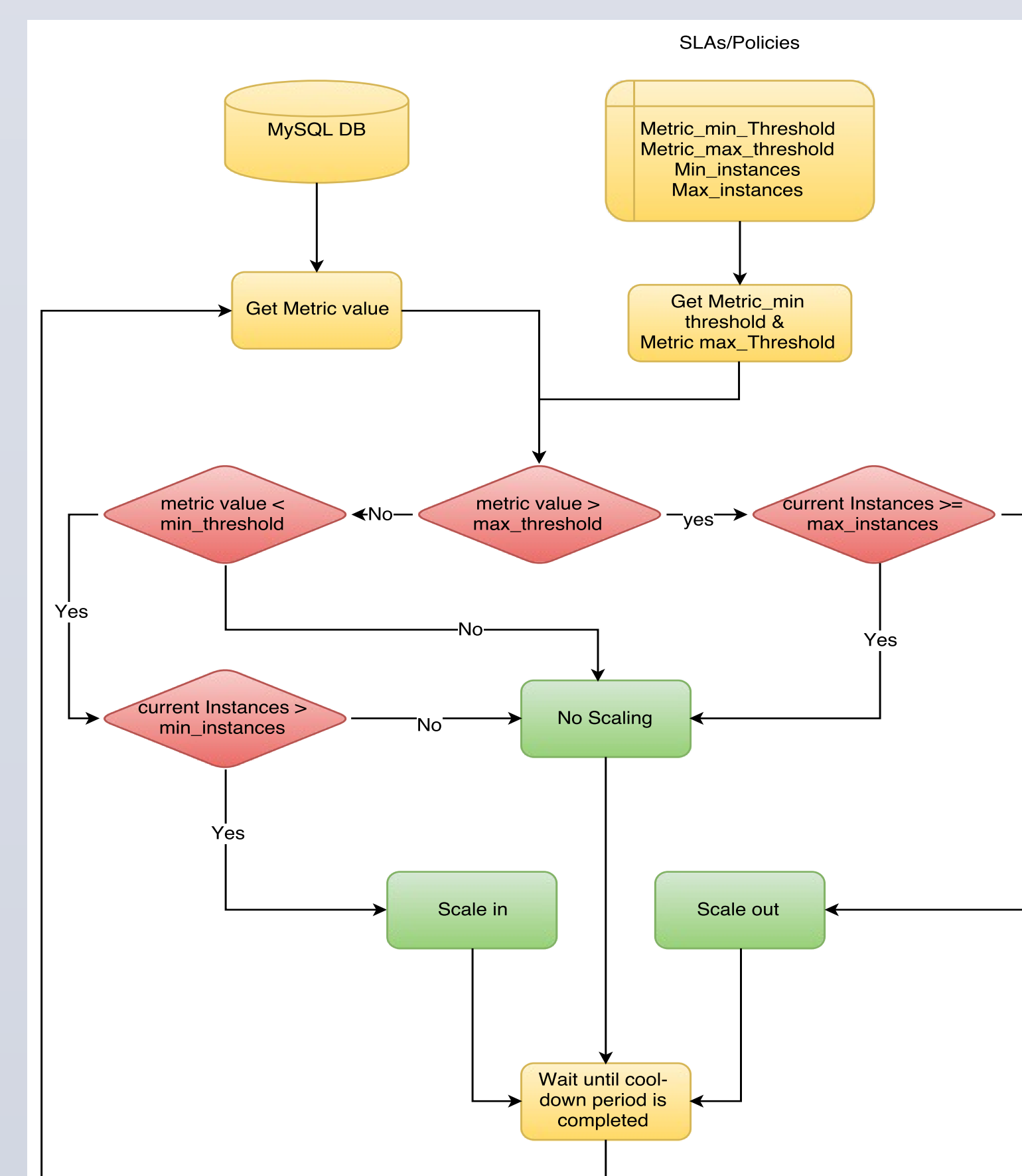


a. Monitoring Service



b. Scaling Service

- Get monitoring data for every cool-down period.
- Set metric thresholds
- Aggregate monitoring data Metrics
- Compare aggregated value of the metrics with the thresholds.
- If aggregated value > maximum metric threshold, then scale out.
- Else if aggregated value < minimum metric threshold, then scale in.

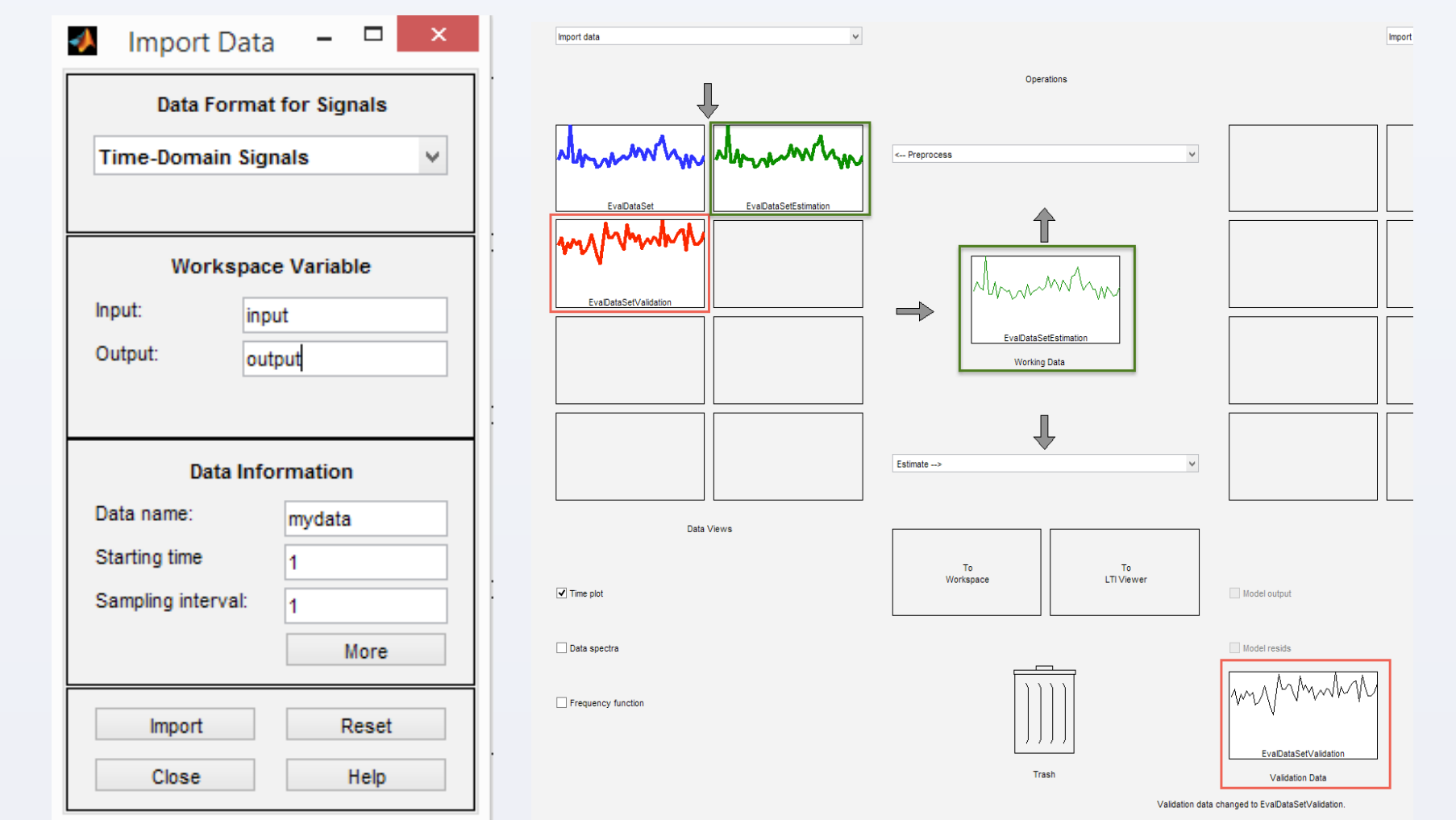


c. Data Modeling

Estimate the following models for the data collected using the MATLAB System Identification Toolbox:

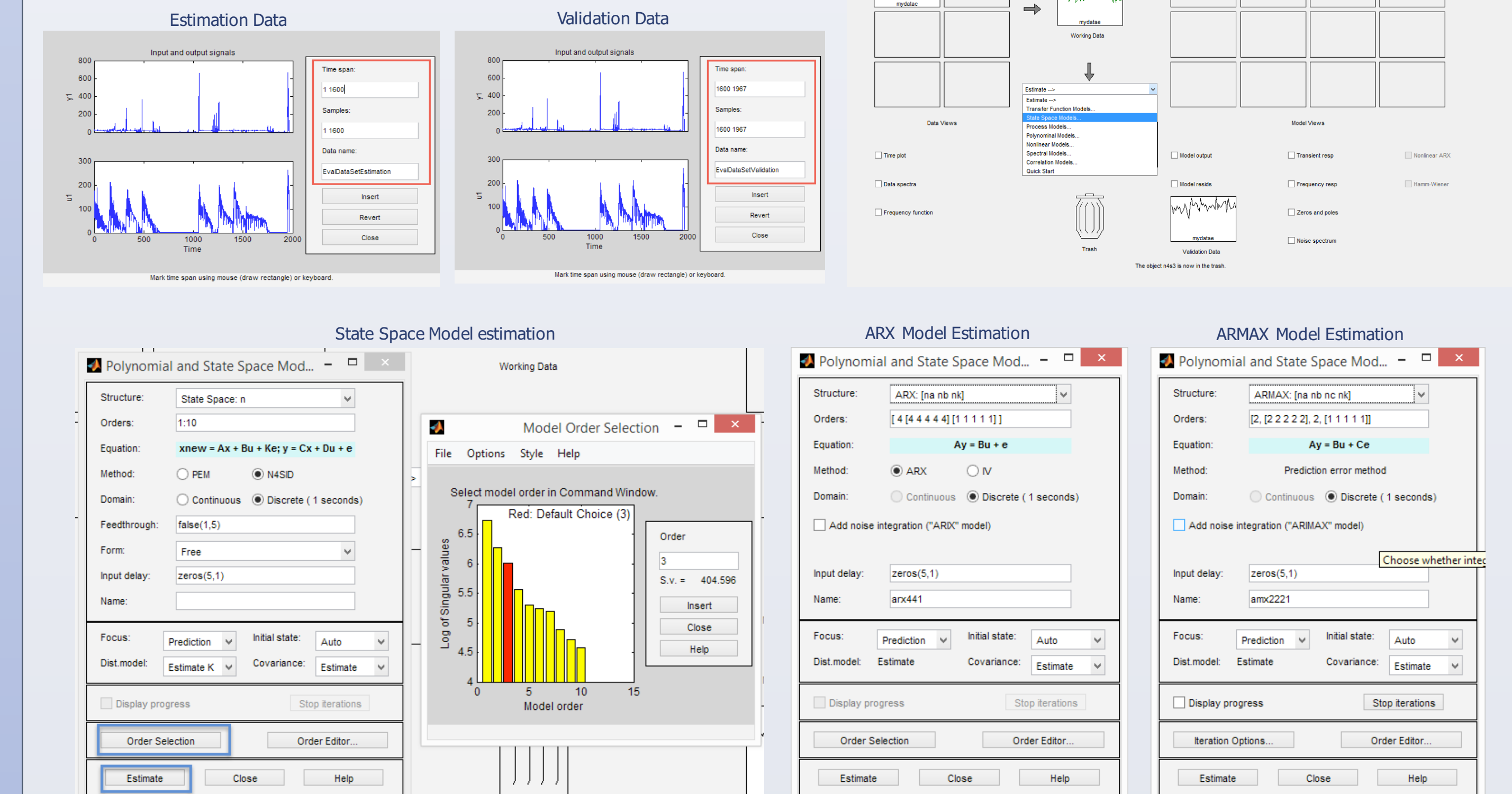
- State-space Model
- Polynomial Models: ARX and ARMAX

Step1: Import data



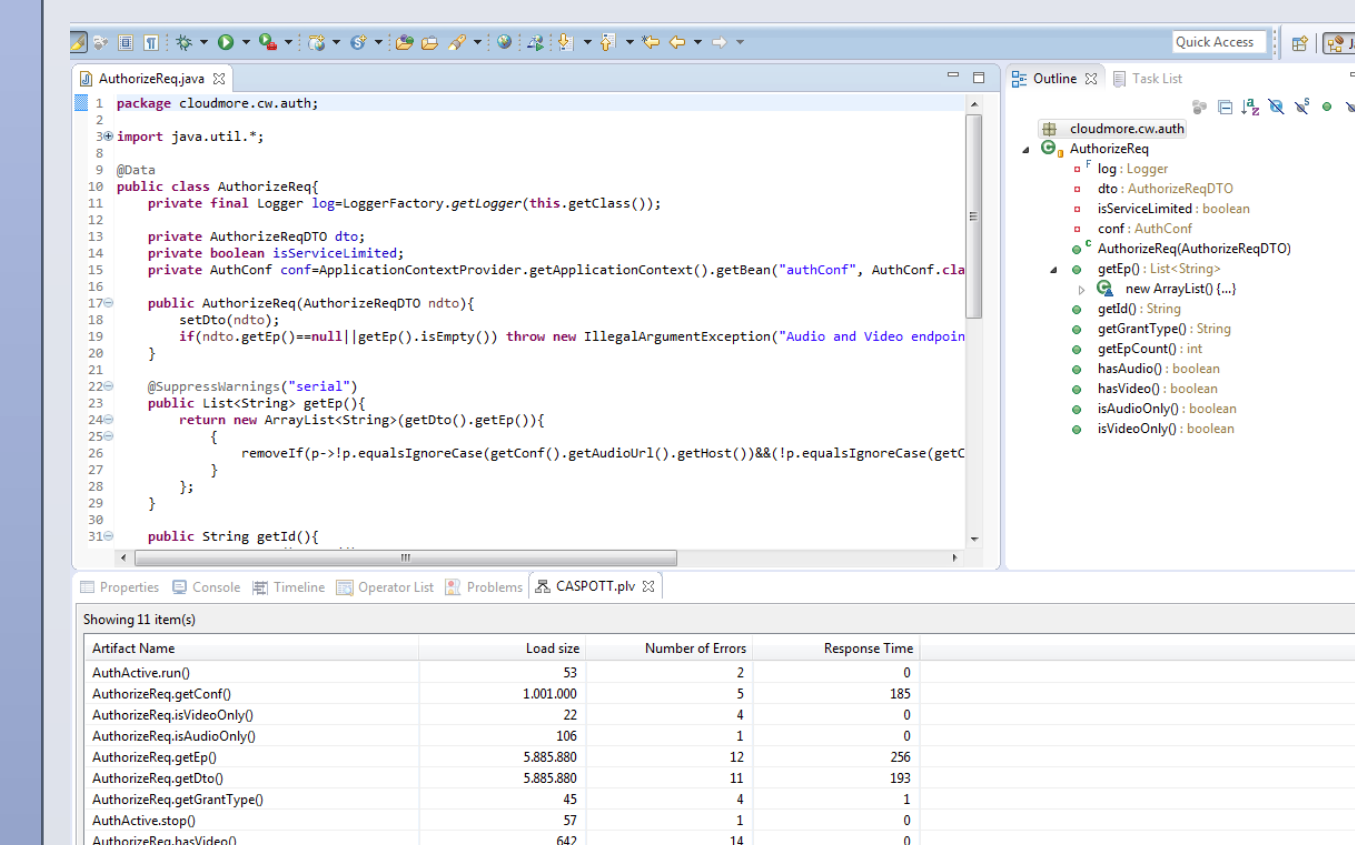
Step2: Split data for estimation and validation

Step3: Model estimation using the estimation dataset for
a. State space models
b. Polynomial Models: ARX and ARMAX

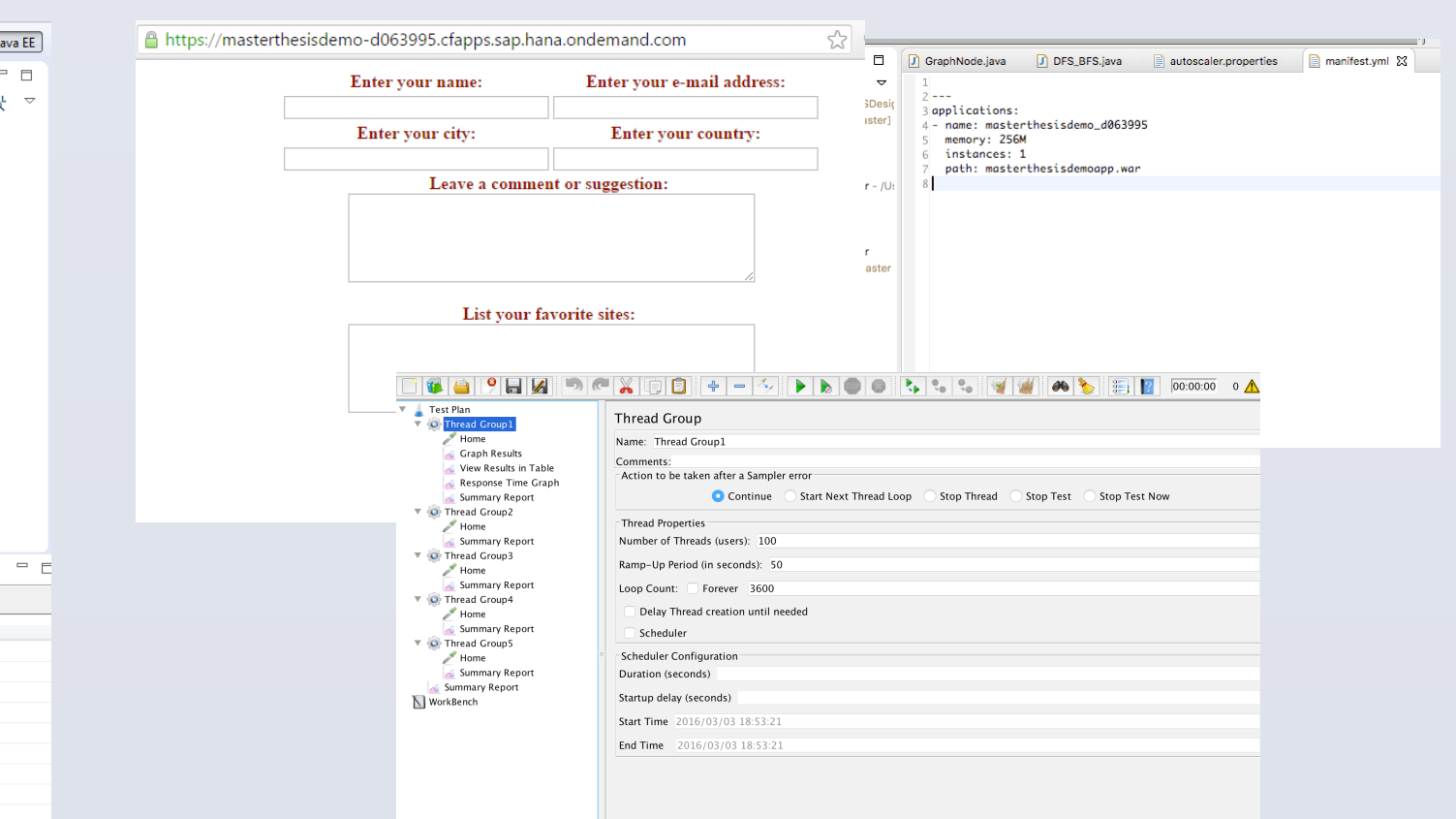


Evaluation and Results

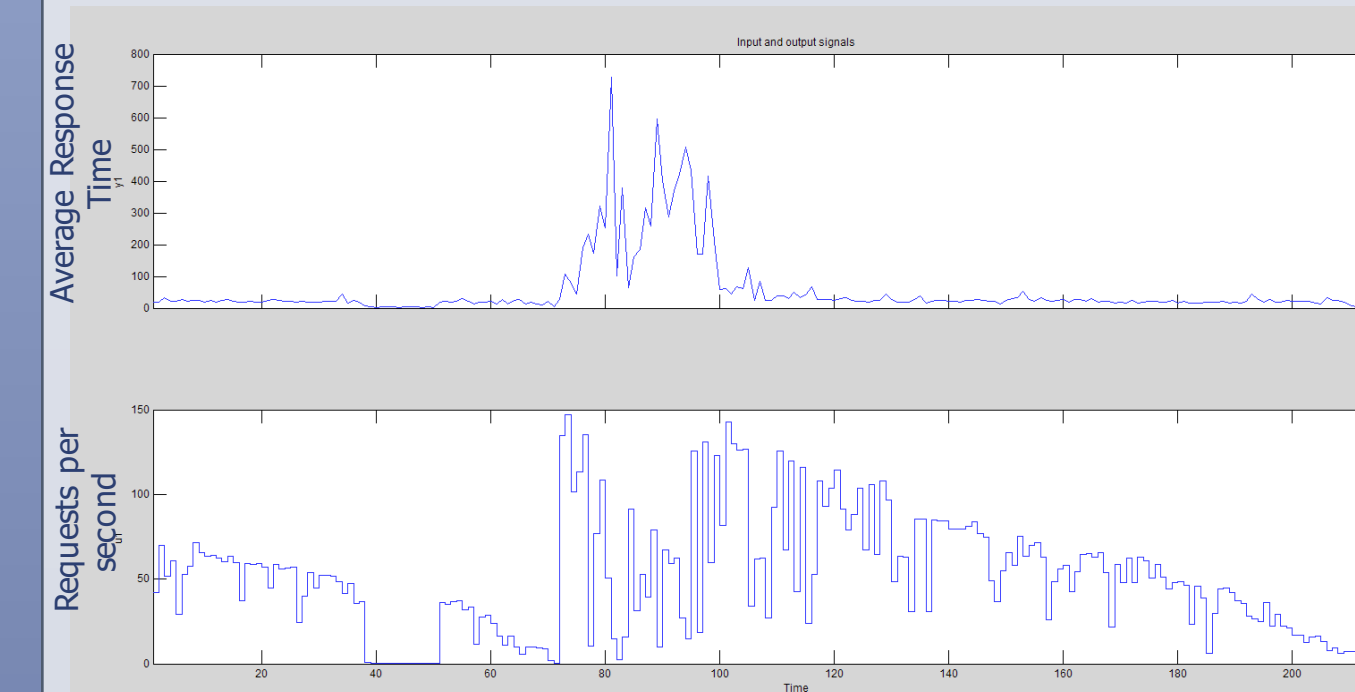
CASPOTT Plugin displaying the method calls with run-time info.



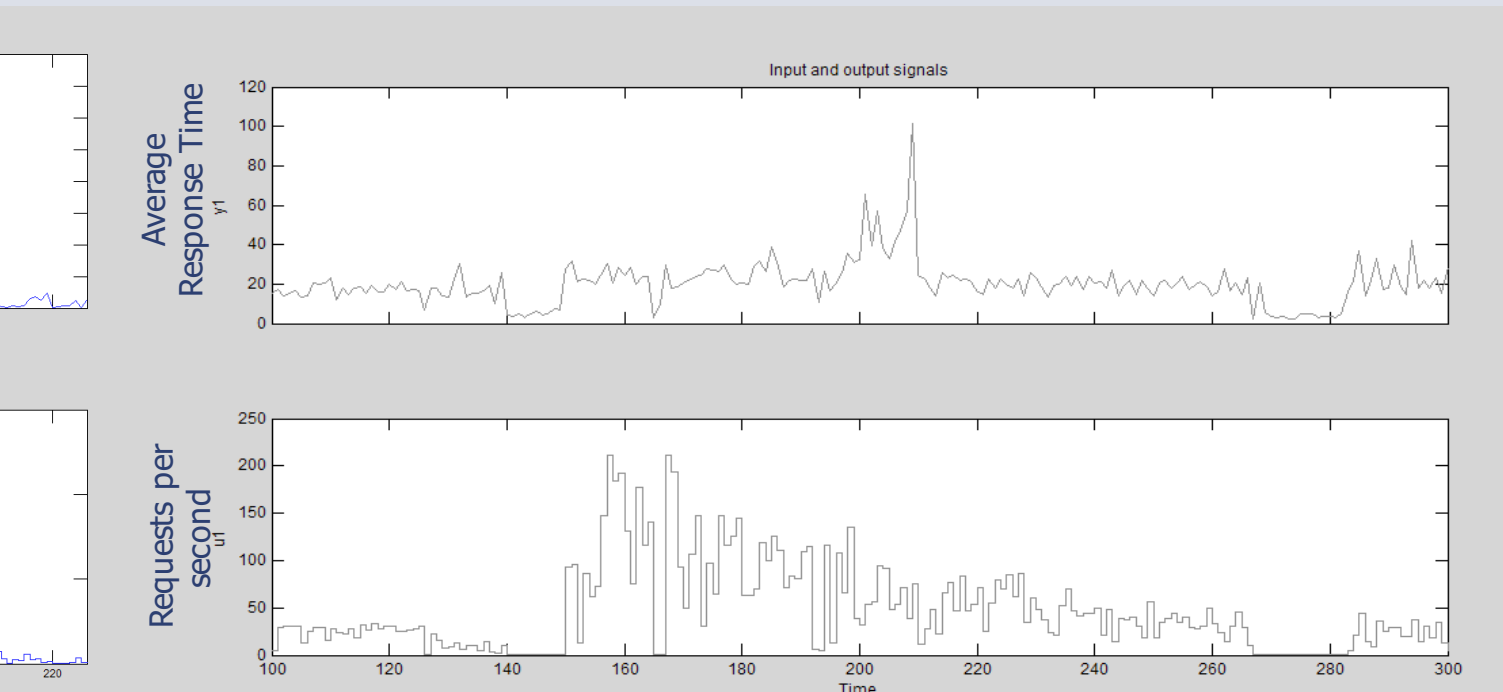
Auto-scaler evaluated on a Guestbook app. Load generated using Jmeter.



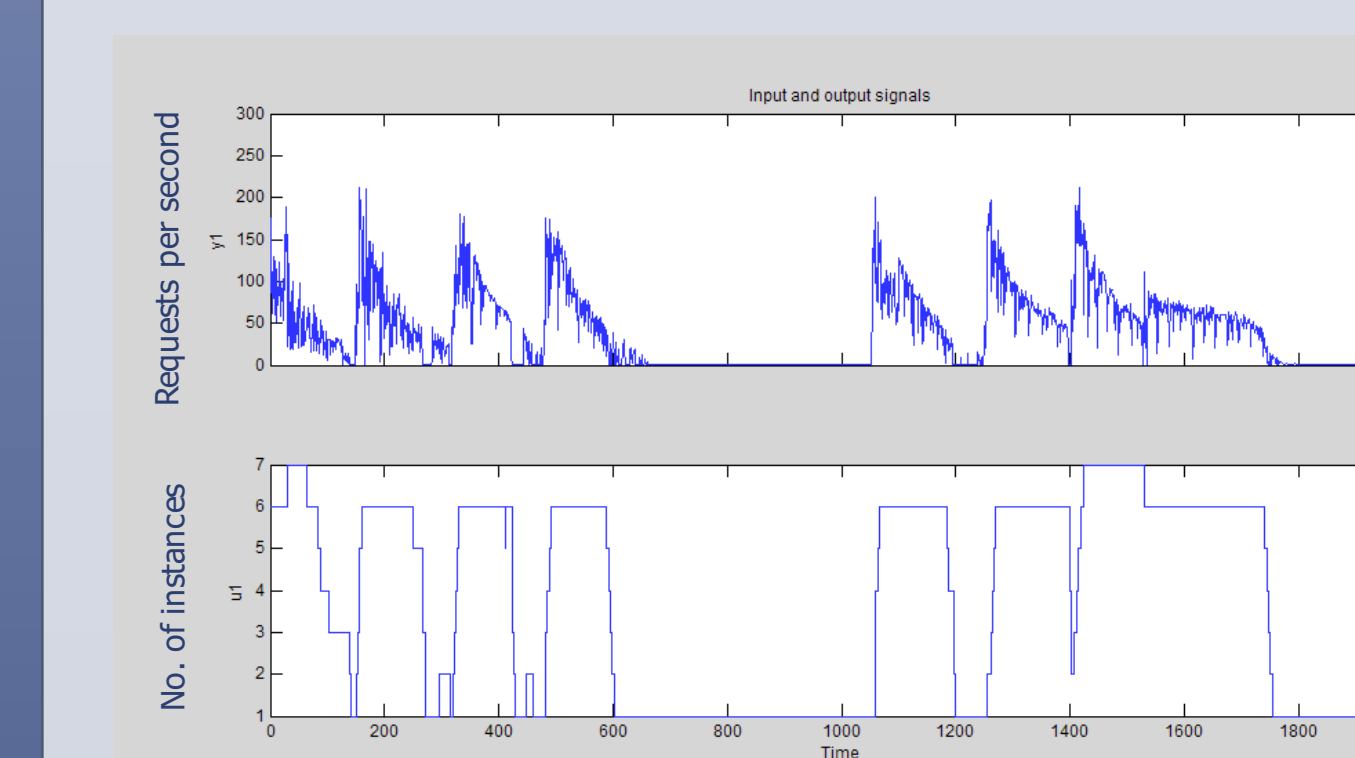
Without auto-scaler, Maximum response time is about 750 ms.



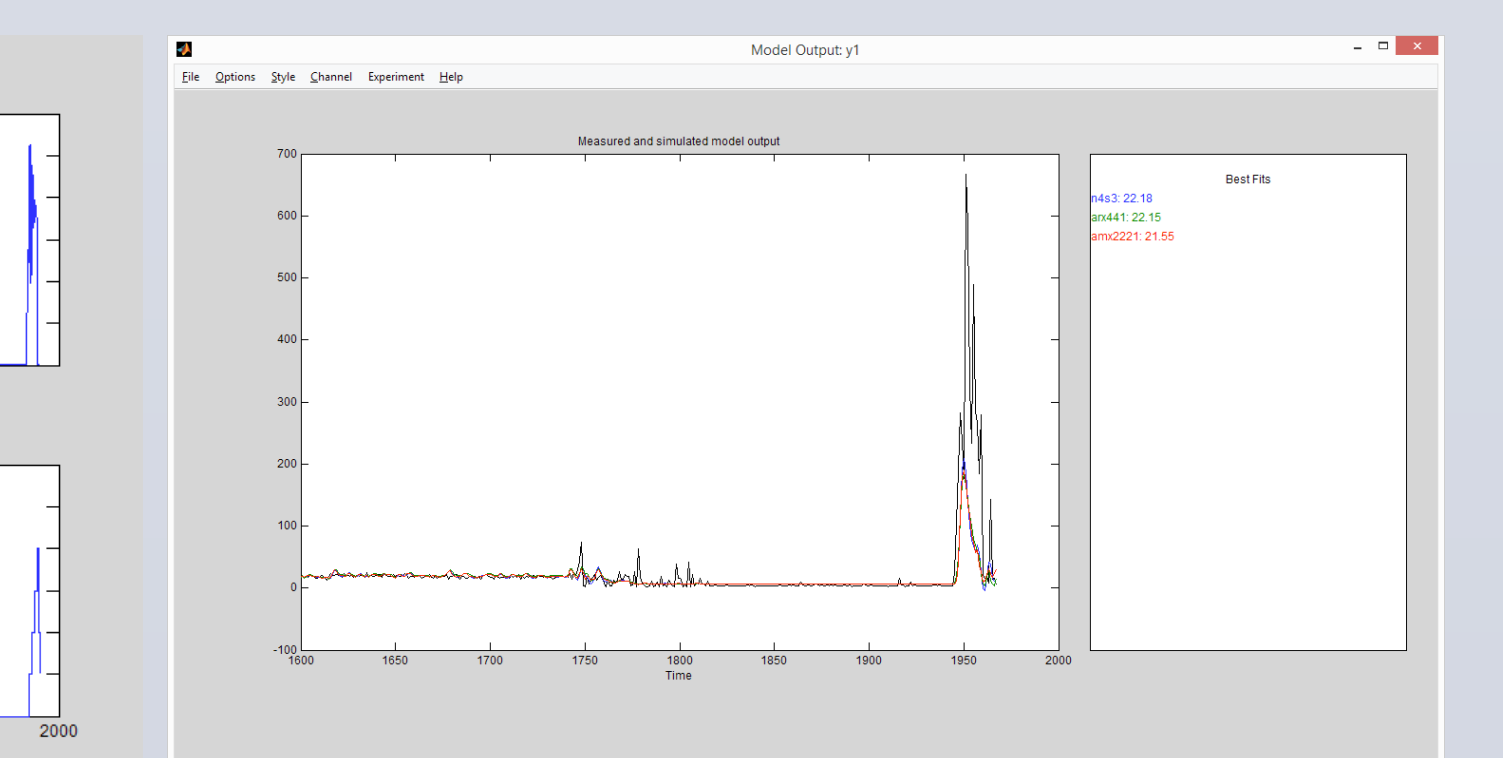
With auto-scaler, Maximum response time is about 100 ms.



Graph shows load versus number of application instances running.



Model estimation output for State space, ARX and ARMAX.



References

- Harini Gunabalan. 2016. *Feedback Driven Development of Cloud Applications*. Master's Thesis. Technical University of Darmstadt, Germany.
- Cito, Jürgen, et al. "Runtime metric meets developer: building better cloud applications using feedback." *2015 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software (Onward!)*. ACM, 2015.

Acknowledgements

This work has been developed partially in the context of the EU-funded CloudWave project (grant 610802), a research project that is enabling a next generation of cloud infrastructure operations and agile development for their hosted applications.

Contact

Harini Gunabalan, <https://harinigunabalan.github.io/>
Project webpage: <https://harinigunabalan.github.io/views/thesis.html>
Master Thesis Supervisors: Prof. Dr. -Ing. Mira Mezini, Dr. -Ing. Guido Salvaneschi