

Online Monitoring & Testing of Web Services



UNIVERSITÀ
DEGLI STUDI
FIRENZE

Tommaso Zoppi,
tommaso.zoppi@unifi.it

Andrea Bondavalli,
andrea.bondavalli@unifi.it

Lucas Carvalho Leal,
leal.lucas2@gmail.com

Eliane Martins,
eliane@ic.unicamp.br



A dark grey arrow points to the right from the left edge of the slide. Several thin, curved lines in shades of blue and grey originate from the left side and sweep across the slide towards the right, framing the content area.

Outline

- ▶ The Devasses Project
- ▶ Context
- ▶ Testing
 - ▶ Model-Based Testing
 - ▶ Offline testing
 - ▶ Online Testing
- ▶ Online Monitoring
- ▶ Our Goals
- ▶ Some research questions
- ▶ Next steps

DEVMSSES



DEsign, Verification and VAlidation of large scale, dynamic Service SystEmS



Centre for Informatics
and Systems of the
University of Coimbra



Department of
Mathematics and
Informatics
University of Florence



IC - Institute of Computing &
FT - School of Technology
State University of Campinas



IC - Institute of Computing
Federal University of Alagoas



Context

- ▶ Service-oriented Applications rely on third-parties services to implement their business
 - ▶ They usually require contractual guarantees of service quality
- ▶ A possible way to assess service quality is:
 - ▶ run offline tests,
 - ▶ monitoring execution at runtime
 - ▶ **Service user**: is the service quality provided as required?
 - ▶ **Service provider**: does service usage exceeds the contract? Is the service quality drops below the expectations?

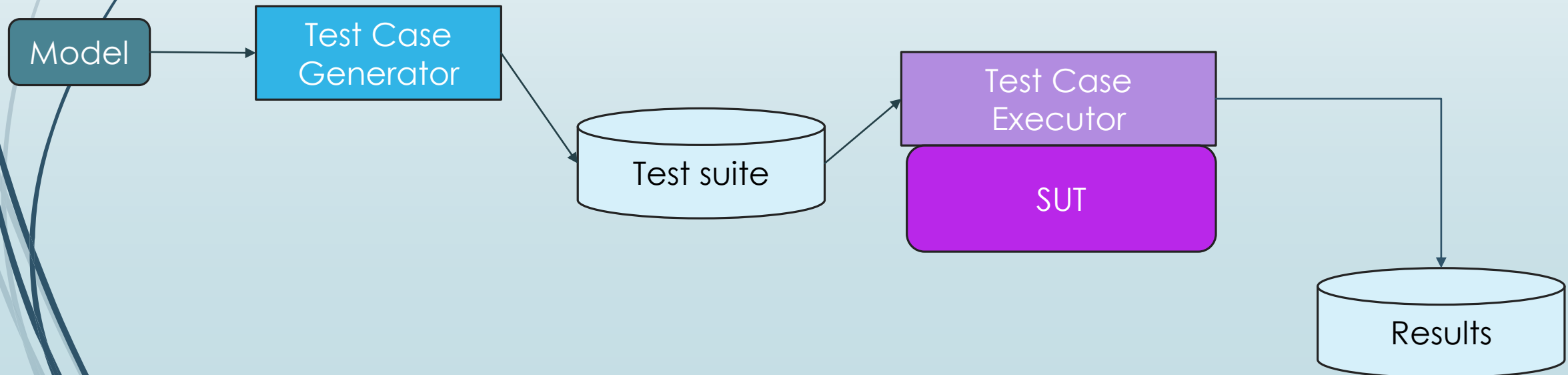


Testing

- ▶ Testing is the process of executing a system in presence of specific inputs with the intent of revealing faults
- ▶ Steps:
 - ▶ Generation of specific inputs and expected outcomes (test cases)
 - ▶ Test execution
 - ▶ Results analysis: observed x expected outcomes
- ▶ Model-based testing:
 - ▶ Systematic method to generate test cases from models
 - ▶ Two main approaches:
 - ▶ Offline testing
 - ▶ Online testing

Offline Testing

- ▶ A finite set of test cases are generated earlier and executed later
- ▶ Test cases can be executed in a third party execution platform
 - ▶ Test cases can be platform independent



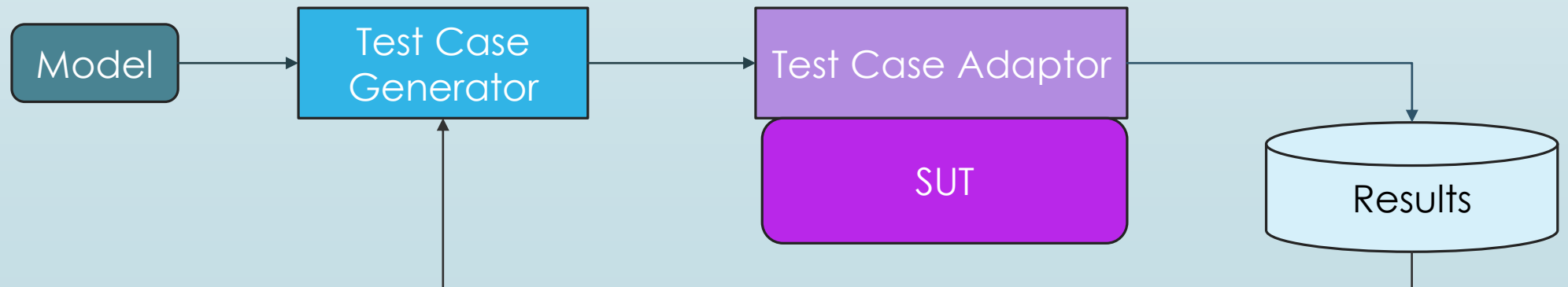
A decorative graphic on the left side of the slide. It features a dark grey arrow pointing to the right at the top. Below the arrow, several thin, curved lines in shades of blue and grey sweep downwards and to the right, creating a dynamic, abstract background element.

Limits of Offline Testing

- ▶ More adequate to the development cycle
 - ▶ The user has to wait until the testing phase is ended
- ▶ The testing is targeted to validate the outcomes of a wide range of behaviours, resulting in a very heavy process with a lot of related issues
 - ▶ Combinatorial explosion
 - ▶ Some tested functionalities might not be used
- ▶ Difficult to control the non determinism of the models

Online Testing

- ▶ Test case generation and execution together
 - ▶ Can be used during runtime
 - ▶ Next step is generated after the observed output is received
 - ▶ It is possible to test nondeterministic systems
 - ▶ Test cases not known in advance





Online Monitoring

- ▶ Data about service delivery is collected and analyzed while the service is being provided
- ▶ Typical steps:
 - ▶ Data collection by observing service execution
 - ▶ Alert generation when service quality violation is detected
 - ▶ As a consequence → corrective actions are taken

A dark grey arrow points to the right from the left edge of the slide. Below it, several thin, curved lines in shades of blue and grey sweep across the left side of the slide.

Our goals

- Combine model-based online testing and online monitoring
 - Monitoring can discover when to reapply the tests
 - Test results can provide data for monitoring analyses



Some questions (1)

► **When to trigger a service testing?**

- Observing changes of the service (system model, version, ...)
- Service provider warnings (maintenance, issues, ...)

► **How to have useful tests?**

- Need to avoid useless tests ⇒ Waste of time or resources!



Some questions (2)

- ▶ **How to test without interfering with a running application?**
- ▶ **How tests can help online monitoring?**
 - ▶ Could tests provide data to help the monitoring mechanism take a good decision?
 - ▶ Could monitoring focus on a subset of variables instead of monitoring everything?
 - ▶ Is it necessary to monitor other variables than the ones already considered?



Next steps

- Study of existing approaches
- Selection or definition of an approach
- Application of the selected/defined approach in a case study
 - Performance overhead
 - Scalability of the approach

Questions?



My contact: eliane@ic.unicamp.br