Towards Pro-active Adaptation with High Confidence

Augmenting Monitoring with Online Testing

Andreas Metzger, Osama Sammodi, Klaus Pohl, Mark Rzepka

Paluno (The Ruhr Institute for Software Technology)
University of Duisburg-Essen
Essen, Germany
www.paluno.de

Funded by the European Community's 7th Framework Programme FP7/2007-2013 / Objective 1.2

www.s-cube-network.eu

Agenda

- Motivation and Problem Statement
- State of the Art
- Solution Idea & Technical Challenges
- Conclusion & Future Work

Need for Adaptation

- **Highly dynamic SBA contexts:** changes of
  - requirements
  - user types
  - 3rd party services
  - service providers
  - …
- Need to respond to **deviations during run-time**
  - no guarantees that 3rd party service fulfils its contract (SLA)
  - hard to assess behavior of infrastructure (e.g., the Internet) during design time
  - …

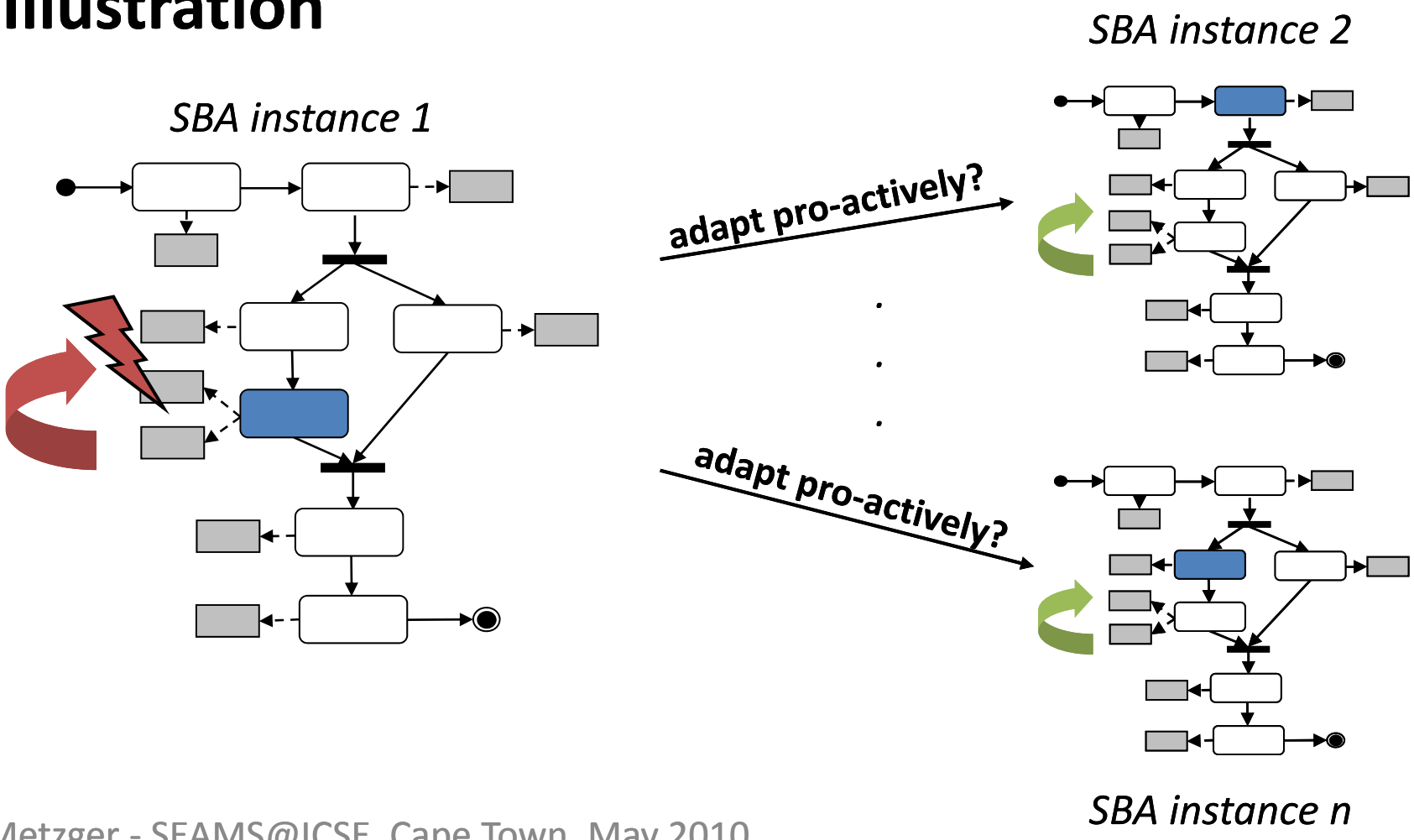
Reactive vs. Pro-active Adaptation

- **Reactive adaptation**
  - observe running SBA instances ("monitoring")
  - trigger adaptation in case of failures, deviations, changes, ...
  - drawbacks: execution of faulty services, reduction of performance, inconsistent end-states, ...
- **Pro-active adaptation**
  - adapt SBA instances to prevent them from failing

Our focus:
- Adaptation of SBA instances
- Adaptation by replacing services

Pro-active Adaptation

- **Illustration**



Avoiding Unnecessary Adaptations

- Unnecessary adaptations can be
  - **costly** *(e.g., SLA negotiation; expensive alternative services)*
  - **faulty** *(e.g., binding of a "buggy" service)*

➔ Pro-active adaptation should be based on **confidence** that failure will occur in SBA instances

- **But: difference** from traditional SW systems:
  - **lack of control** on 3rd party services
  - **functionality and QoS can vary** between service invocations (even if input is identical)
  - Service implementation can **change** without notice

Agenda

- Motivation and Problem Statement
- State of the Art
- Solution Idea & Technical Challenges
- Conclusion & Future Work

State of the Art (1)

- **Statistical evaluation / data mining of monitoring data** [Liang et al. 2006, Fu et al. 2007, Sahoo et al. 2003]
  - requires large volume of monitoring data
  - adaptation may invalidate past monitoring data and thus prediction

- **Statistical testing** [Poore et al. 1998, Trammell et al. 1995]
  - prediction of *overall* system's reliability (and not individual failures)
  - based on usage profile of the system
  - requires large number of test cases (significant effort/costs due to invocation of external services)

- **Reliability modelling and analysis** [Goel et al. 1985, Zhang et al. 2008, Roshandel et al 2006]
  - prediction of *overall* system's reliability (and not individual failures)
  - static analysis technique (can complement monitoring & online testing)

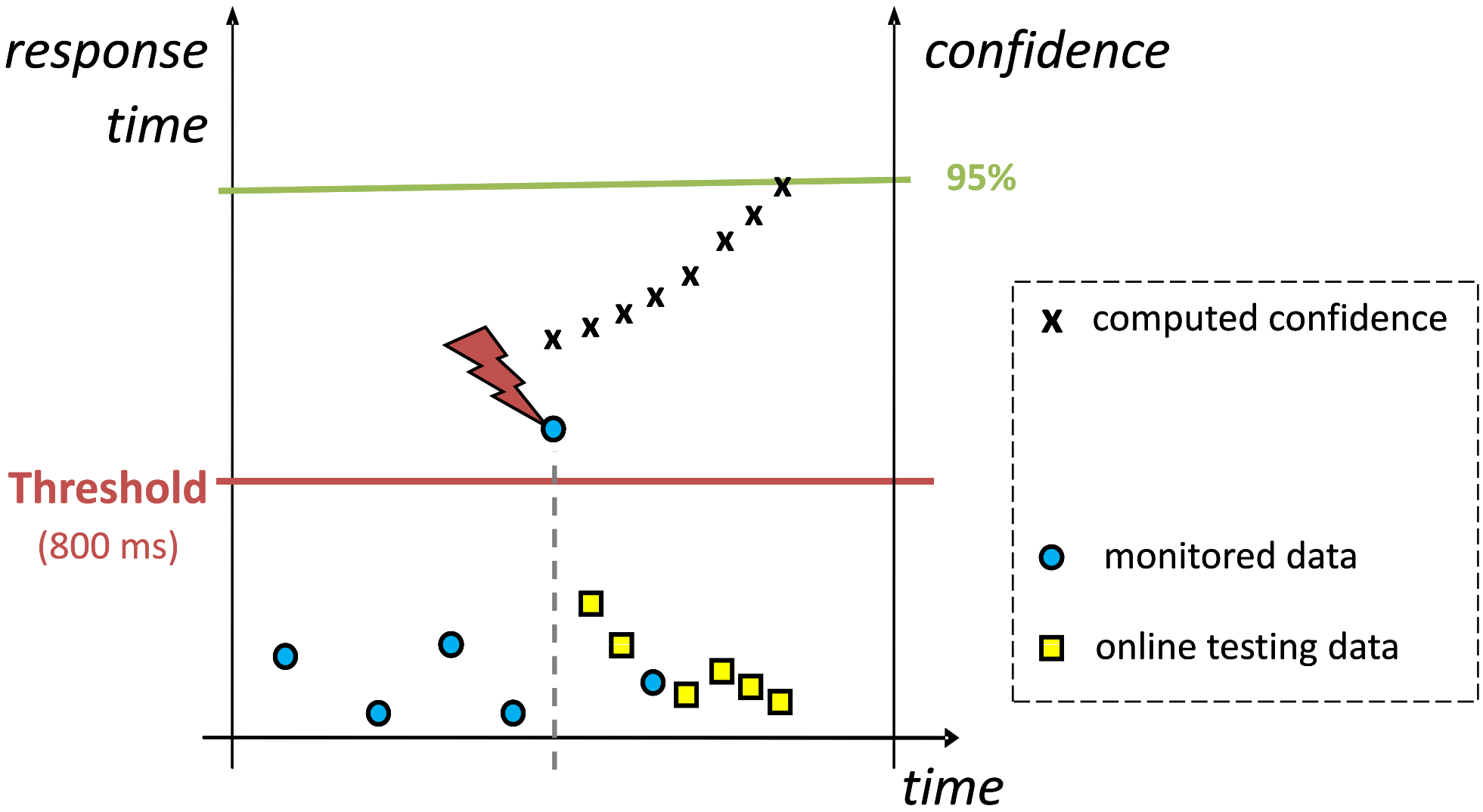
State of the Art (2)

- **Joint monitoring & testing efforts** [www.secse-project.eu]
  - Proposed applications:
    1. Use monitoring data to mimic service behaviour
    2. Generate test cases to violate the SLAs (search-based technique using monitoring data)
    3. Analyse the monitoring data to infer invariants
- **Online testing** [Wang et al. 2004, Deussen et al. 2003, Bai et al. 2007]
  - **"Definition":** SBA is fed with dedicated test input in parallel to its normal use and operation ("production system")
  - Current techniques not used for building confidence

Agenda

- Motivation and Problem Statement
- State of the Art
- Solution Idea & Technical Challenges
- Conclusion & Future Work

Solution Idea



Assumptions

1. **Failure** in constituent service
   → **requirements deviation** (end-to-end quality violation)
   - not necessarily the case in all situations
     - *e.g.; slower response of service B compensated by faster response of service A*

2. Test of services has **no side effects**
   - *e.g., no books are delivered if testing online book service*

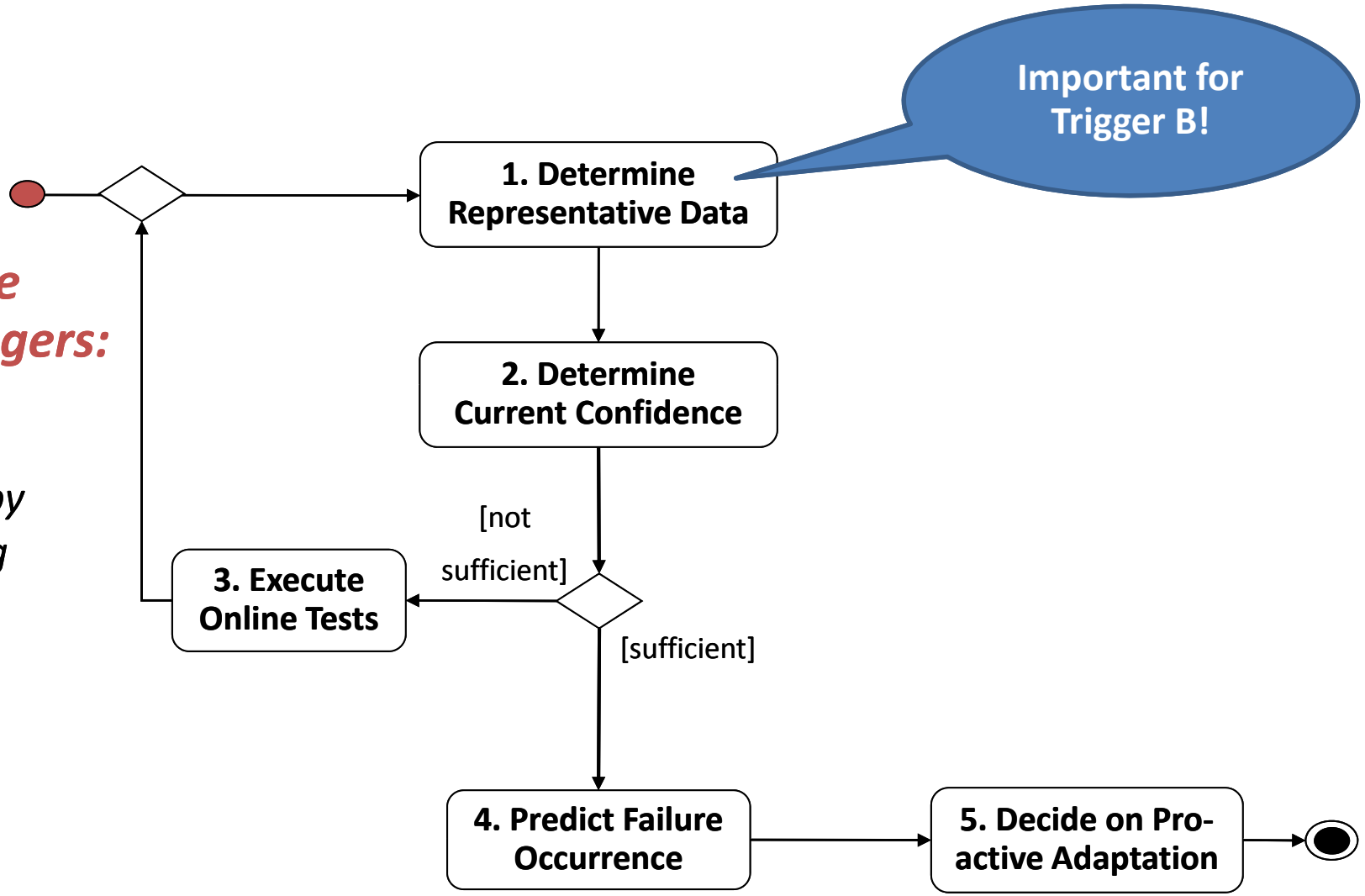
3. Services **notify about changes** that could invalidate monitoring/testing data

Steps of the Approach

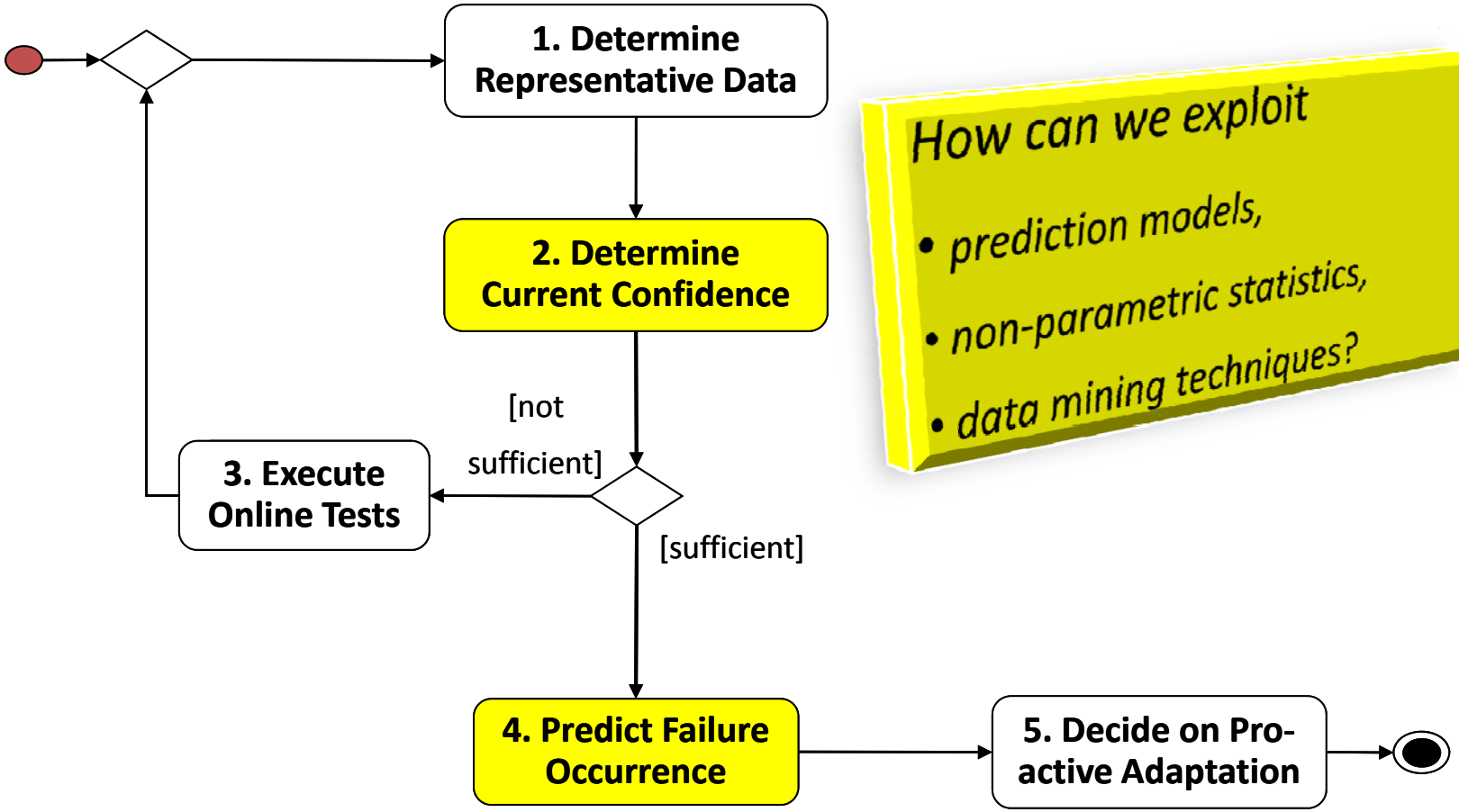
Two possible kinds of triggers:

A) *Failure observed by monitoring*

B) *Change of service*



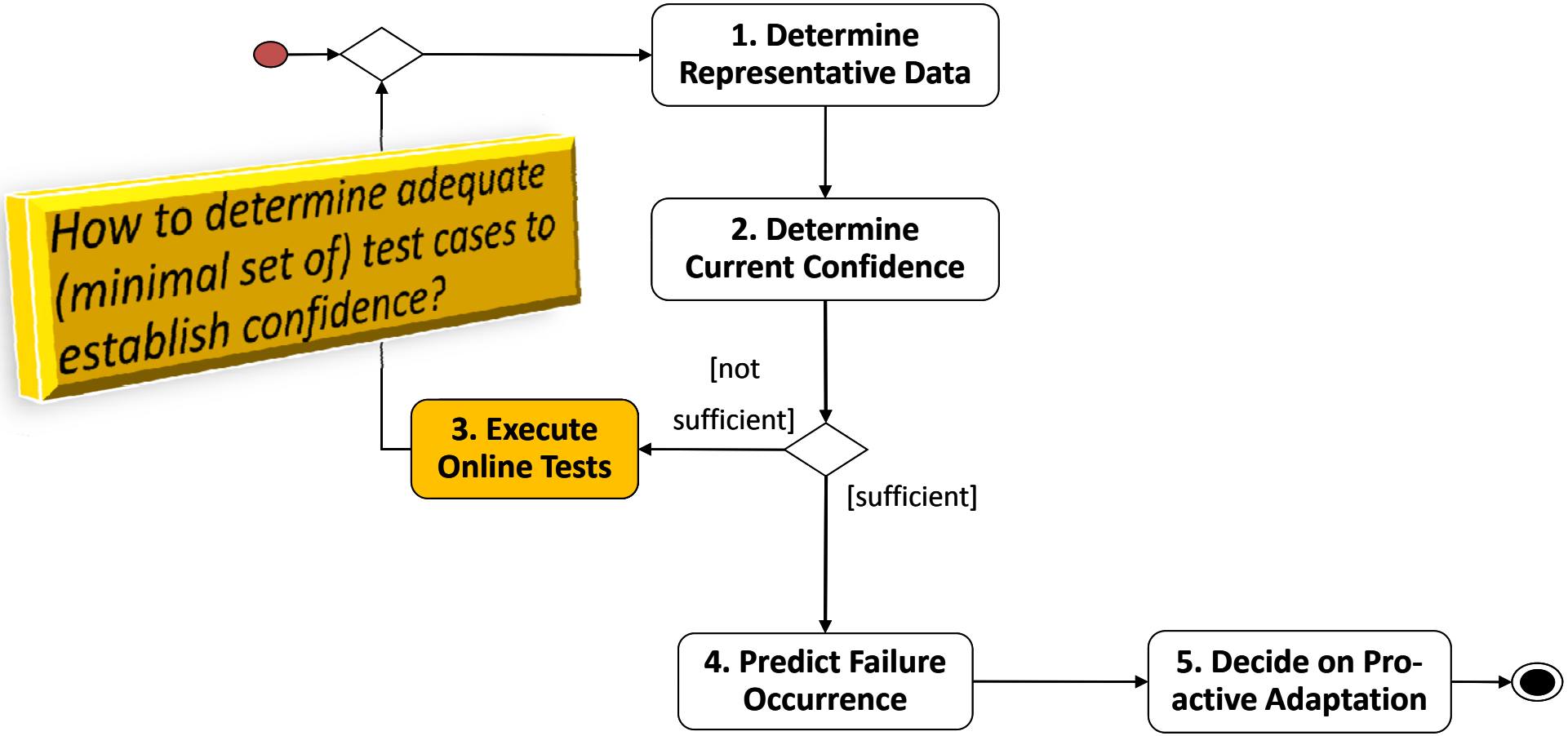
Technical Challenges



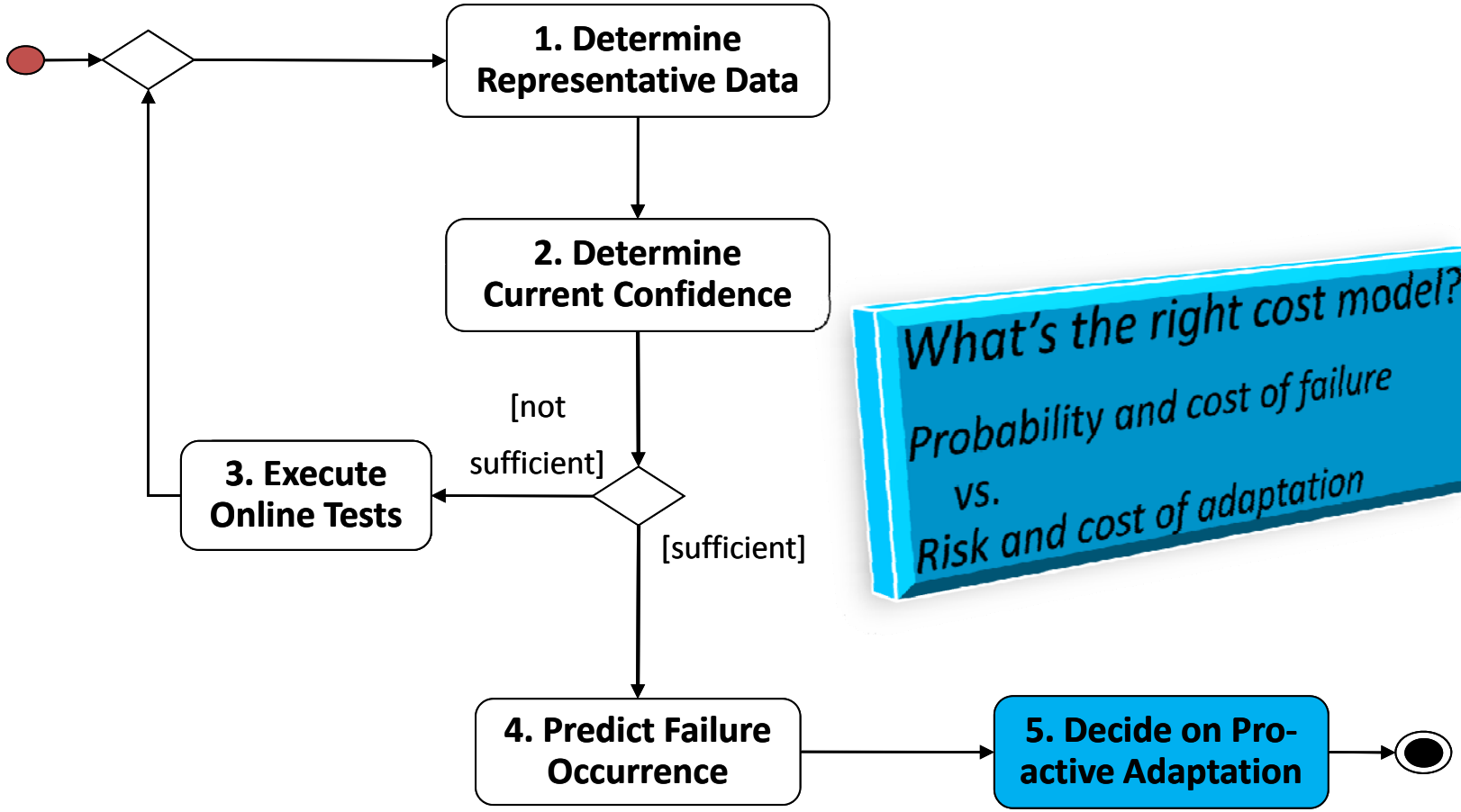
How can we exploit

- *prediction models,*
- *non-parametric statistics,*
- *data mining techniques?*

Technical Challenges



Technical Challenges



What's the right cost model?

Probability and cost of failure

vs.

Risk and cost of adaptation

Relaxing the Assumptions

1. **Failure → requirements deviation**
   - S-Cube approach: **Run-time verification** [Bianculli et al. 2008 , Gehlert et al. 2010]
   - Reduces effort for online testing:
     - Initiate steps only if deviation (→ Strategy A)
2. Tests have **no side effects**
   - *Service invocation = query only*
   - *Dedicated test mode / interface*
3. **Notification about changes**
   - *Novel service registries*
   - *New SLA constituents*

Agenda

- Motivation and Problem Statement
- State of the Art
- Solution Idea & Technical Challenges
- Conclusion & Future Work

Conclusion and Future Work

- Avoid unnecessary pro-active adaptations
- Many data points needed for confident prediction
  - Thus: augment monitoring with online testing
- Interesting challenges remain to be addressed

Participate to the shaping of the Future Internet of Services

Apply as **S-Cube Associate Member**

S-CUBE

Application Deadline: May, 17

www.s-cube-network.eu

References (1)

- **[Bai et al. 2007]** Bai, X., Chen, Y., Shao, Z.: Adaptive web services testing. In: 31st Annual International Computer Software and Applications Conference (COMPSAC). (2007) 233–236
- **[Bianculli et al. 2008]** Bianculli, D. et al.. A Guided Tour through SAVVY-WS: a Methodology for Specifying and Validating Web Service Compositions. In Proceedings Advances in Software Engineering 2008
- **[Chan et al. 2007]** Chan, W., Cheung, S., Leung, K.: A metamorphic testing approach for online testing of service-oriented software applications. International Journal of Web Services Research 4 (2007) 61–81
- **[Deussen et al. 2003]** Deussen, P., Din, G., Schieferdecker, I.: A TTCN-3 based online test and validation platform for Internet services. In: Proceedings of the 6th International Symposium on Autonomous Decentralized Systems (ISADS). (2003) 177–184
- **[Fu el al. 2007]** Song Fu and Cheng-Zhong Xu. Exploring event correlation for failure prediction in coalitions of clusters. Proceedings of the 2007 ACM/IEEE conference on Supercomputing, November 10-16, 2007, Reno, Nevada
- **[Gehlert et al. 2010]** Gehlert, A. et al.: Exploiting Assumption-Based Verification for the Adaptation of Service-Based Applications. In Proceedings SOAP @ SAC 2010
- **[Goel et al. 1985]** Goel, A.L.: Software reliability models: Assumptions, limitations, and applicability. IEEE Trans. Softw. Eng. 11 (1985) 1411–1423

References (2)

PALUNO
The Ruhr Institute for Software Technology

- **[Liang et al. 2006]** Y. Liang, Y. Zhang, A. Sivasubramaniam, M. Jette, and R. K. Sahoo. BlueGene/L failure analysis and prediction models. In Proceeding of IEEE International Conference on Dependable Systems and Networks (DSN), 2006
- **[Poore et al. 1998]** Poore, J., Trammell, C.: Engineering practices for statistical testing. Crosstalk: The Journal of Defense Software Engineering 11 (1998) 24–28
- **[Roshandel et al 2006]** Roshandel, R., Banerjee, S., Cheung, L., Medvidovic, N., Golubchik, L.: Estimating software component reliability by leveraging architectural models. In: ICSE. (2006) 853–856
- **[Sahoo et al. 2003]** R. K. Sahoo, A. J. Oliner, I. Rish, and et al. Critical event prediction for proactive management in large-scale computer clusters. In Proceeding of ACM International Conference on Knowledge Discovery and Data Dining (SIGKDD), 2003
- **[Trammell wt al. 1995]** Trammell, C.: Quantifying the reliability of software: statistical testing based on a usage model. In: ISESS ’95: Proceedings of the 2nd IEEE Software Engineering Standards Symposium, Washington, DC, USA, IEEE Computer Society (1995) 208
- **[Wang et al. 2004]** Wang, Q., Quan, L., Ying, F.: Online testing of Web-based applications. In: Proceedings of the 28th Annual International Computer Software and Applications Conference (COMPSAC). (2004) 166–169
- **[Zhang et al. 2008]** Zhang, F., Zhou, X., Chen, J., Dong, Y.: A novel model for component-based software reliability analysis. In: HASE ’08: Proceedings of the 2008 11th IEEE High Assurance Systems Engineering Symposium, Washington, DC, USA, IEEE Computer Society (2008) 303–309

End of Slides