Juggling with Data:
On the Lack of Database Monitoring in Long-Living Software Systems

EMLS 2017

4th Collaborative Workshop on Evolution and
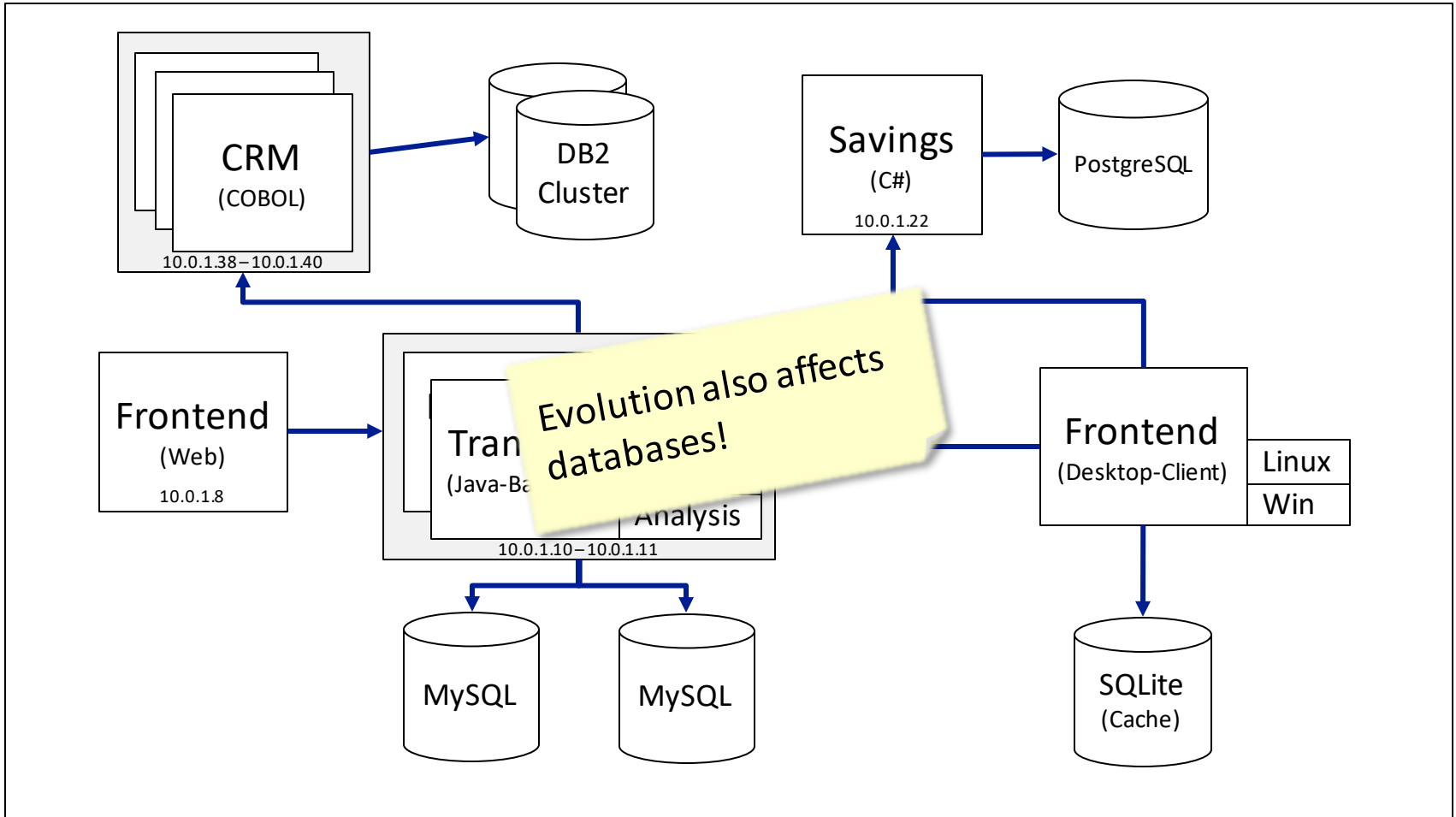
Maintenance of Long-Living Software System

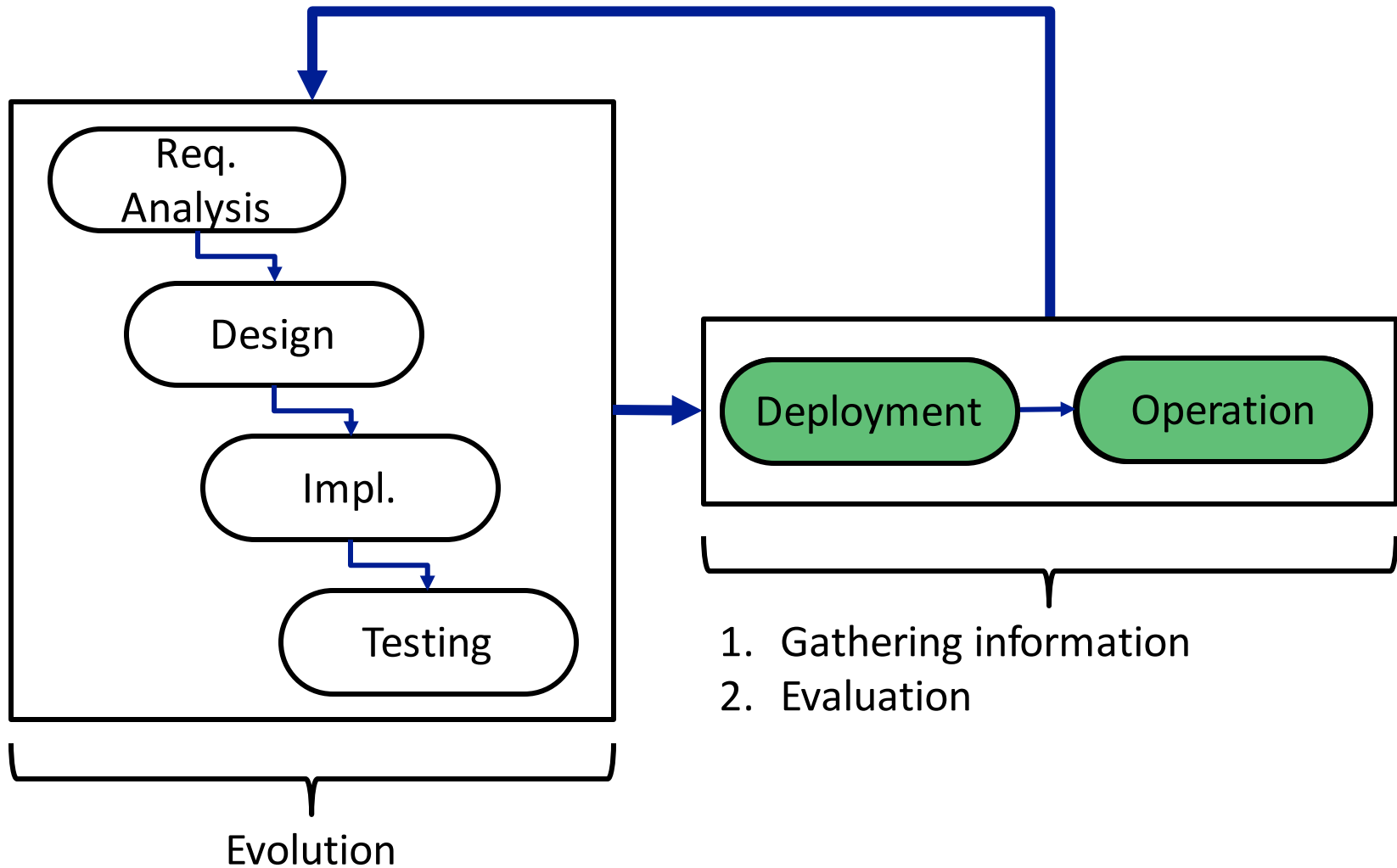Christian Zirkelbach, Kiel University

Feburary 21, 2017

Example Software Landscape (Banking Industry)

Systems, Applications, and Databases

Challenges

- Handling large-scale, enormous-in-size data repositories
  [Cuzzocrea et. al]

- Changing requirements or increasing workload

- Performance issues or customer requests   [Zirkelbach et. al]
  → ineveitable software updates or refactoring

- Legacy systems: often based on outdated technologies
  and poorly documented   [Godfrey and German]

- Insufficient knowledge of the (actual) systems hamper
  the process   [LaToza et. al]
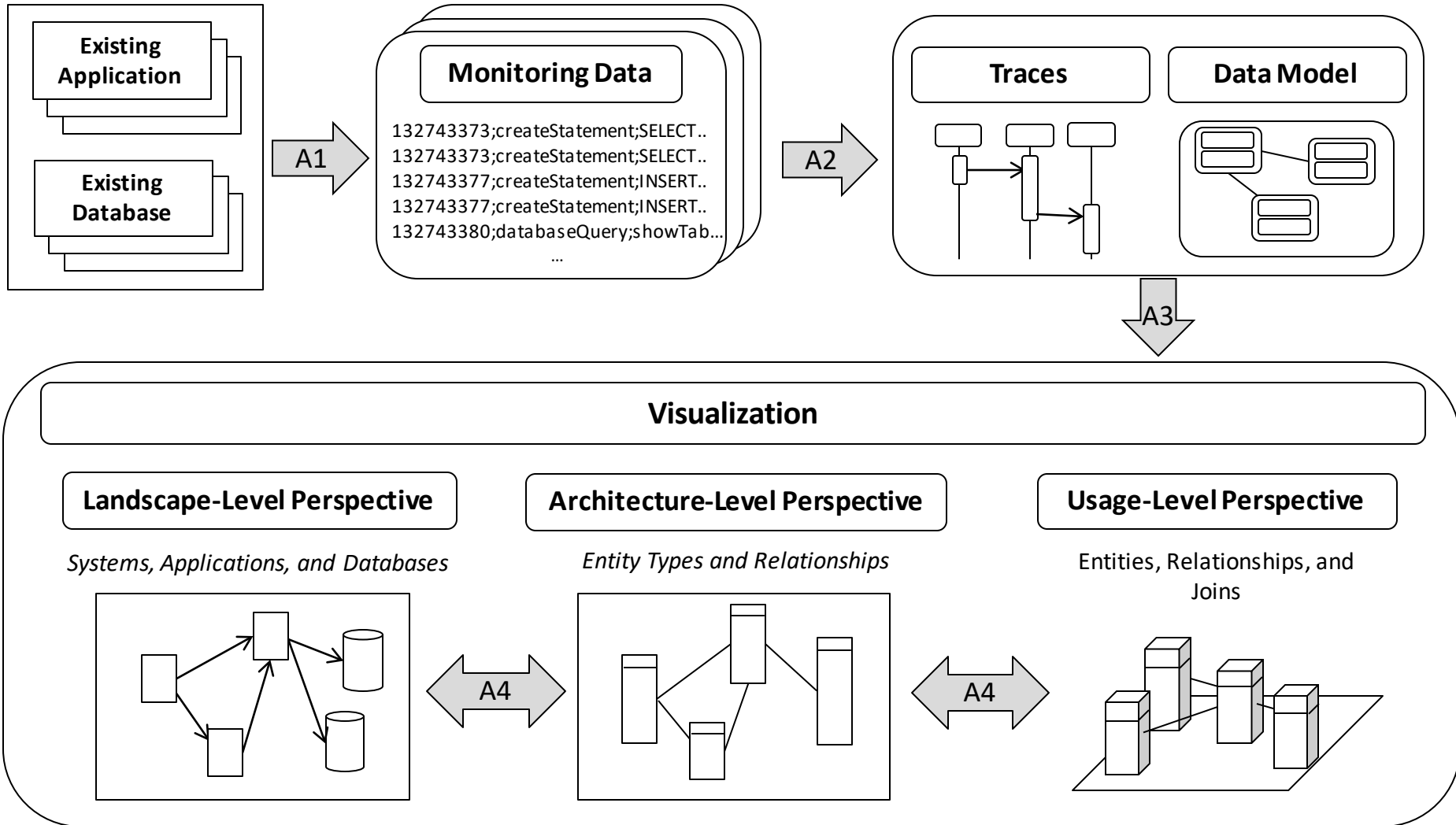
Where to start?

Req. Analysis

Design

Impl.

Testing

Evolution

Deployment → Operation

1. Gathering information
2. Evaluation

Extended system development life cycle (SDLC) based on [Avison and Fitzgerald]

Envisioned Approach

Envisioned Approach

Existing Application

Existing Database

A1

Monitoring Data

132743373;createStatement;SELECT..
132743373;createStatement;SELECT..
132743377;createStatement;INSERT..
132743377;createStatement;INSERT..
132743380;databaseQuery;showTab...
...

A2

Traces

Data Model

A3

Visualization

Landscape-Level Perspective

Systems, Applications, and Databases

A4

Architecture-Level Perspective

Entity Types and Relationships

A4

Usage-Level Perspective

Entities, Relationships, and Joins

Legend A1: *Monitoring* | A2: *Analysis* | A3: *Transformation* | A4: *Navigation*

Conclusions & Open Questions

Conclusions

- Lack on database monitoring in long-living systems
  - Based on…
    - obsolete technologies and platforms
    - poor documentation
    - insufficient knowledge

- Presented an approach as a solution
  - Live database trace visualization for large software landscapes
  - Adresses developers and operators
  - Only first draft – open for feedback and suggestions

- Which visualization and layout is suitable for our landscape-level perspective, that comprises the complete software landscape including the databases?

- Does our 3D visualization within our usage-level perspective offer an advantage over a traditional 2D visualization like the architecture-level perspective?

- Which related approaches or tools could be employed, when evaluating our approach within a controlled experiment?

Bibliography

[Chen] P. P.-S. Chen. "The Entity-Relationship Model – Toward a Unified View of Data." In: ACM Trans. Database Syst. 1.1 (Mar. 1976), pp. 9–36.

[Raijlich et. al] V. Raijlich et al. "Software cultures and evolution." In: Computer 34.9 (Sept. 2001), pp. 24–28.

[De Pauw et. al] W. De Pauw et al. "Visualizing the Execution of Java Programs." In: Software Visualization. Springer, 2002, pp. 151–162.

[Mens and Tourw´e] T. Mens and T. Tourw´e. "A Survey of Software Refactoring." In: IEEE Trans. Softw. Eng. 30.2 (Feb. 2004), pp. 126–139.

[Avison and Fitzgerald] D. Avison and G. Fitzgerald. Information Systems Development: Methodologies, Techniques and Tools. 4th. Information systems series. McGraw-Hill Higher Education, 2006.

[LaToza et. al] T. D. LaToza, G. Venolia, and R. DeLine. "Maintaining Mental Models: A Study of Developer Work Habits." In: Proceedings of the 28th International Conference on Software Engineering. ICSE '06. Shanghai, China: ACM, 2006, pp. 492–501.

[Wettel and Lanza] R. Wettel and M. Lanza. "Visualizing Software Systems as Cities." In: Proceedings of the 4th IEEE International Workshop on Visualizing Software for Understanding and Analysis, 2007, pp. 92–99.

[Godfrey and German] M. Godfrey and D. German. "The past, present, and future of software evolution." In: Frontiers of Software Maintenance, 2008. FoSM 2008. Sept. 2008, pp. 129–138.

[Cuzzocrea et. al] A. Cuzzocrea, I.-Y. Song, and K. C. Davis. "Analytics over Large-scale Multidimensional Data: The Big Data Revolution!" In: Proceedings of the ACM 14th International Workshop on Data Warehousing and OLAP. 2011, pp. 101–104.

Bibliography (cont'd)

[Ray et. al] S. Ray, B. Simion, and A. D. Brown. "Jackpine: A benchmark to evaluate spatial database performance." In: Proceedings of the 27th International Conference on Data Engineering. Apr. 2011, pp. 1139–1150.

[Durdik et. al] Z. Durdik et al. "Sustainability guidelines for long-living software systems." In: Proceedings of the 28th IEEE International Conference on Software Maintenance (ICSM), 2012, pp. 517–526.

[Meurice and Cleve 2014] L. Meurice and A. Cleve. "DAHLIA: A visual analyzer of database schema evolution." In: Proceedings of the IEEE Conference on Software Maintenance, Reengineering, and Reverse Engineering (CSMR-WCRE), 2014, pp. 464–468.

[Valacich et. al] J. S. Valacich, J. F. George, and J. A. Hover. Essentials of Systems Analysis and Design. 6th. Pearson Education, 2015.

[Zirkelbach et. al] C. Zirkelbach, W. Hasselbring, and L. Carr. "Combining Kieker with Gephi for Performance Analysis and Interactive Trace Visualization." In: Symposium on Software Performance 2015: Joint Developer and Community Meeting of Descartes/Kieker/Palladio. 2015.

[Chen et. al] T. H. Chen et al. "Finding and Evaluating the Performance Impact of Redundant Data Access for Applications that are Developed Using Object-Relational Mapping Frameworks." In: IEEE Transactions on Software Engineering 42.12 (Dec. 2016), pp. 1148–1161.

[Fittkau et. al] F. Fittkau, A. Krause, and W. Hasselbring. "Software landscape and application visualization for system comprehension with ExplorViz." In: Information and Software Technology (2016). http://dx.doi.org/10.1016/j.infsof.2016.07.004.

[Meurice and Cleve 2016] L. Meurice and A. Cleve. "DAHLIA 2.0: A Visual Analyzer of Database Usage in Dynamic and Heterogeneous Systems." In: Proceedings of the IEEE Working Conference on Software Visualization (VISSOFT), 2016, pp. 76–80.