A Method for Aspect-Oriented Meta-Model Evolution

VAO 2014 – York

Reiner Jung Robert Heinrich Eric Schmieders
Misha Strittmatter Wilhelm Hasselbring

22nd July 2014

iObserve

This work was partially supported by the DFG (German Research Foundation) under the Priority Program SPP1593: Design For Future – Managed Software Evolution (grant HA 2038/4-1, RE 1674/6-1, PO 607/3-1) and the Helmholtz Association of German Research Centers

Motivation

Driver of Meta-Model Evolution

1. Long-living software systems
2. Convergence of design-time and run-time models
   e.g. MAPE scenarios [Ghezzi, 2011]

Aspect-Oriented Modeling (Concern-Driven Development)

- Separate concerns
- Reduce meta-model complexity
- Ease meta-model evolution and reuse

Example: Palladio Component Meta-Model

[Becker, Koziolek, & Reussner, 2009]

Palladio

- **Domain** Performance prediction
- **Meta-Model** 188 classes (PCM)
- **Concerns** Components, deployment, workloads, ...

Scenario

- Reuse PCM as run-time meta-model for forecasting

Extensions

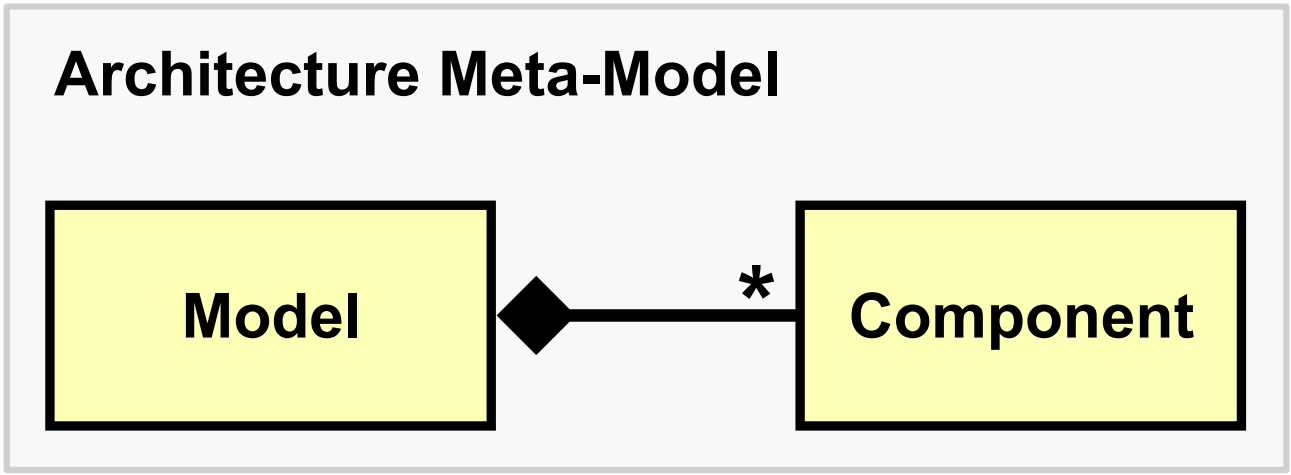
- Cloud
- Monitoring

Related Meta-Modeling Approaches

- Meta-Model Extension by Subclassing [Steinberg et al., 2009]
- Decorator Pattern Approach
- EMF Profiles [Langer et al., 2012]

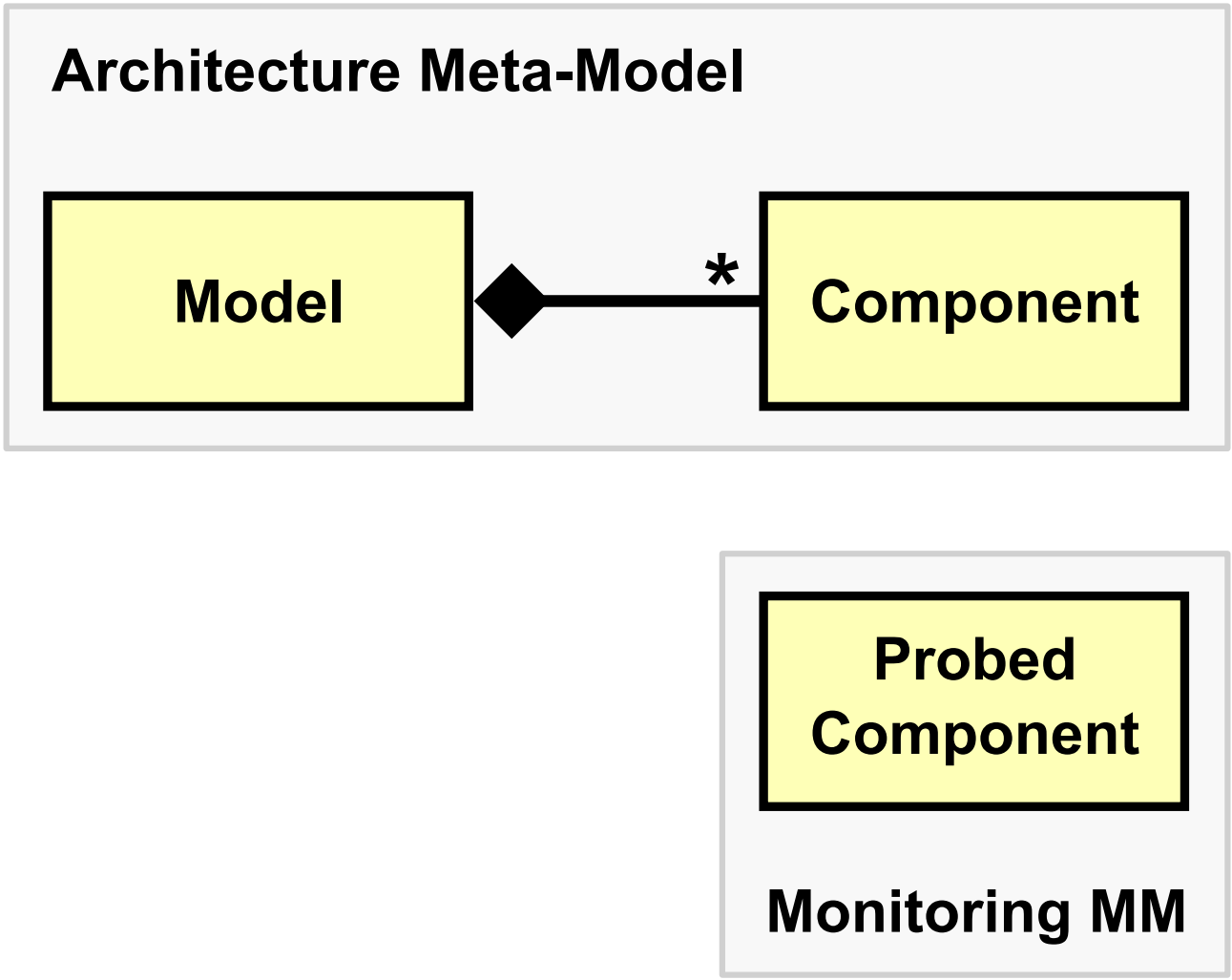
Meta-Model Extension by Subclassing

[Steinberg et al., 2009]



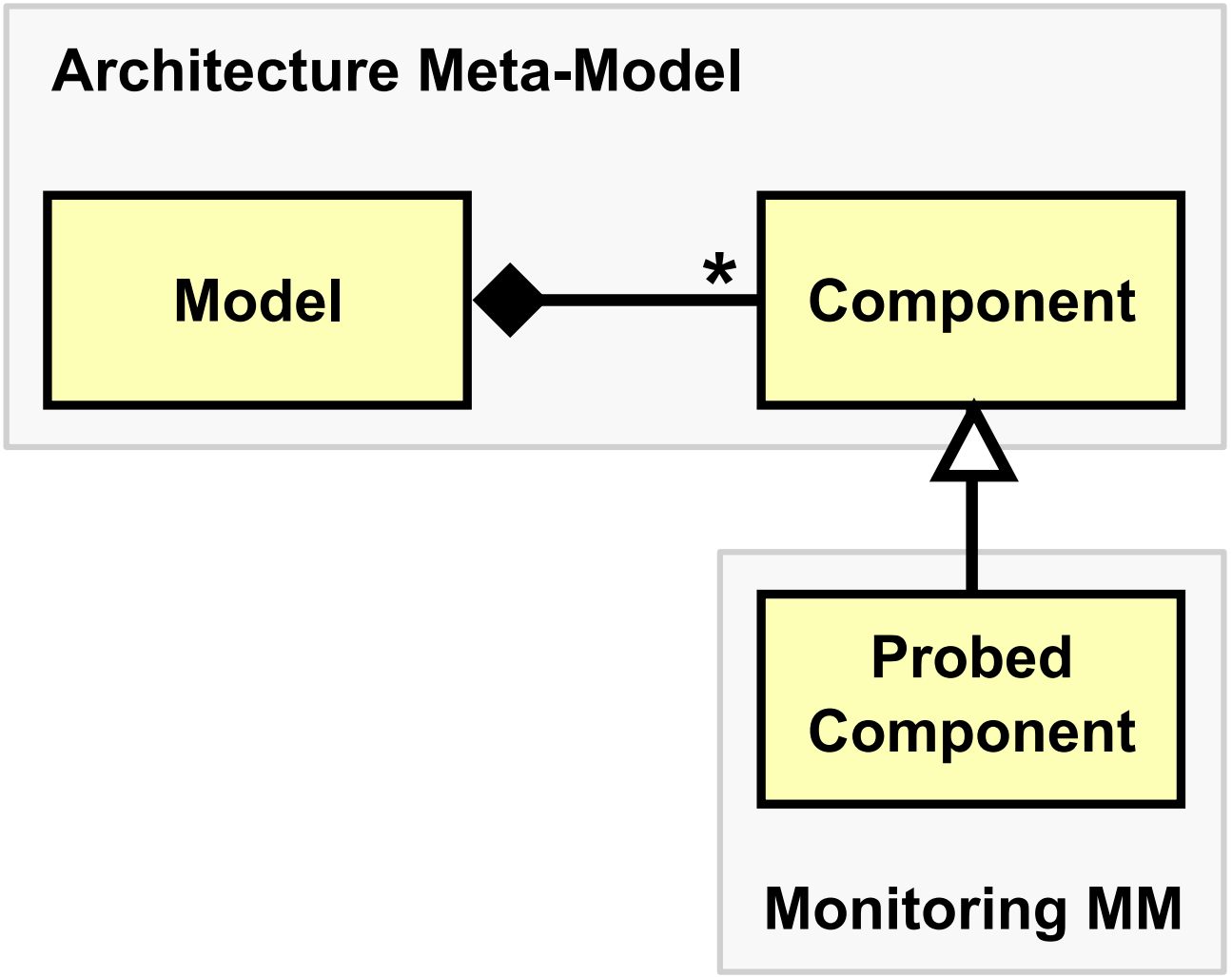
Meta-Model Extension by Subclassing

[Steinberg et al., 2009]



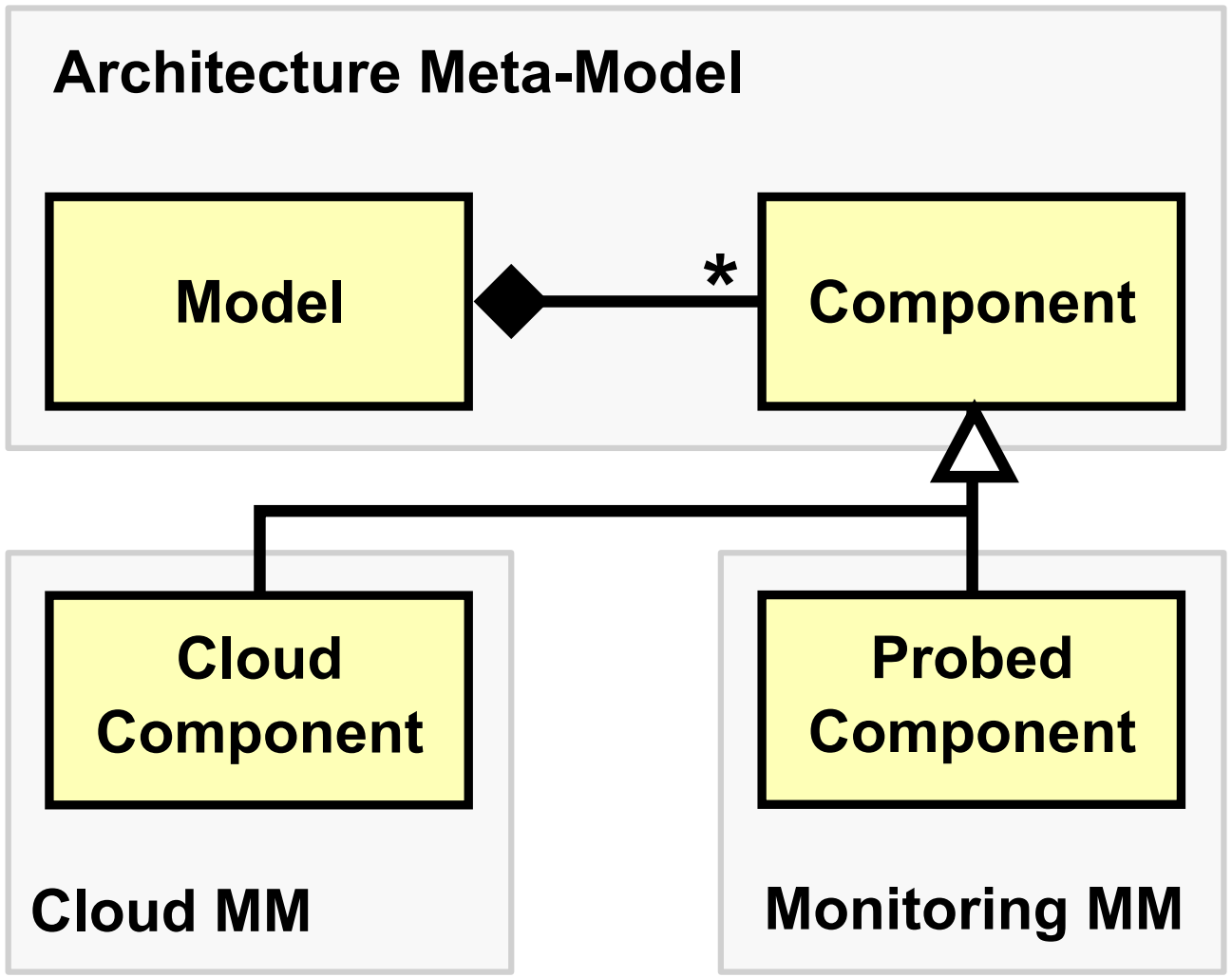
Meta-Model Extension by Subclassing

[Steinberg et al., 2009]



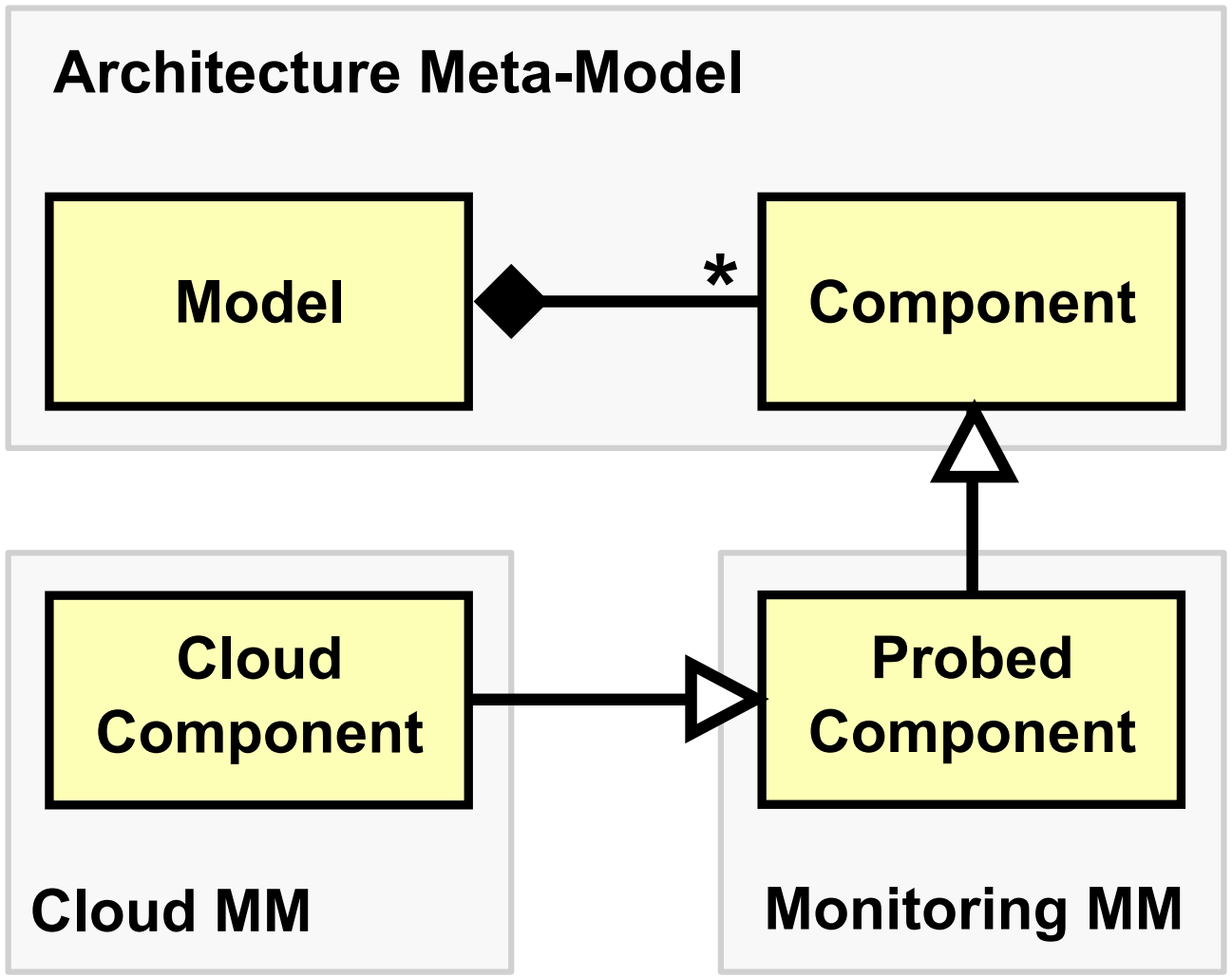
Meta-Model Extension by Subclassing

[Steinberg et al., 2009]



Meta-Model Extension by Subclassing

[Steinberg et al., 2009]



Solution Requirements

cf. [Langer et al., 2012]

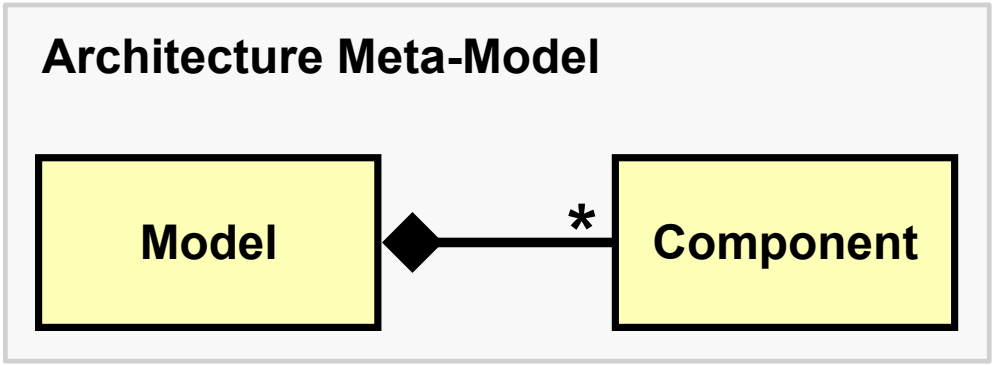
1. **Non-invasive** meta-model annotation mechanism
2. **Separation of concerns** from different modeling domains
3. Honor different **types of relationships and roles** of meta-models
4. Should support a **formal specification** for extensions/annotations
5. Must support **existing technology** to ease integration
6. Easy application and **compatible with existing tooling**

Overview

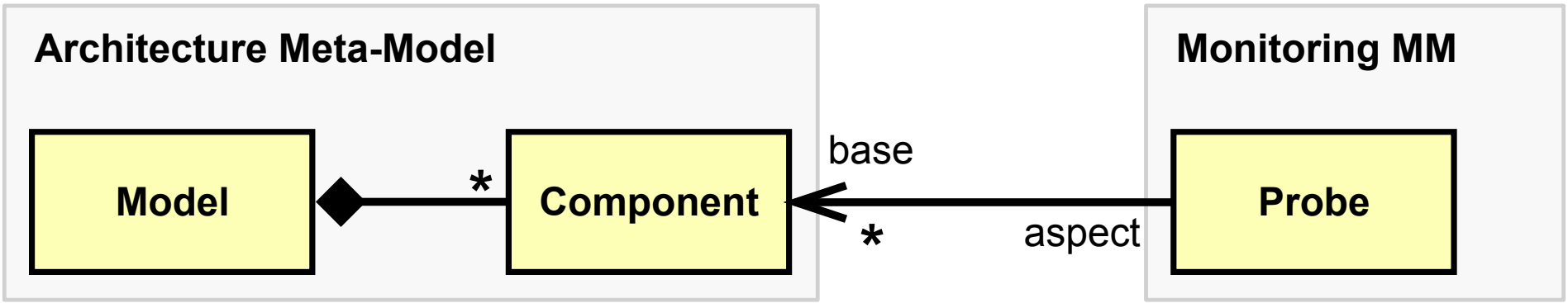
Central ideas

- Aspect-oriented meta-models
- Semantic of references
- Contextual meta-model patterns

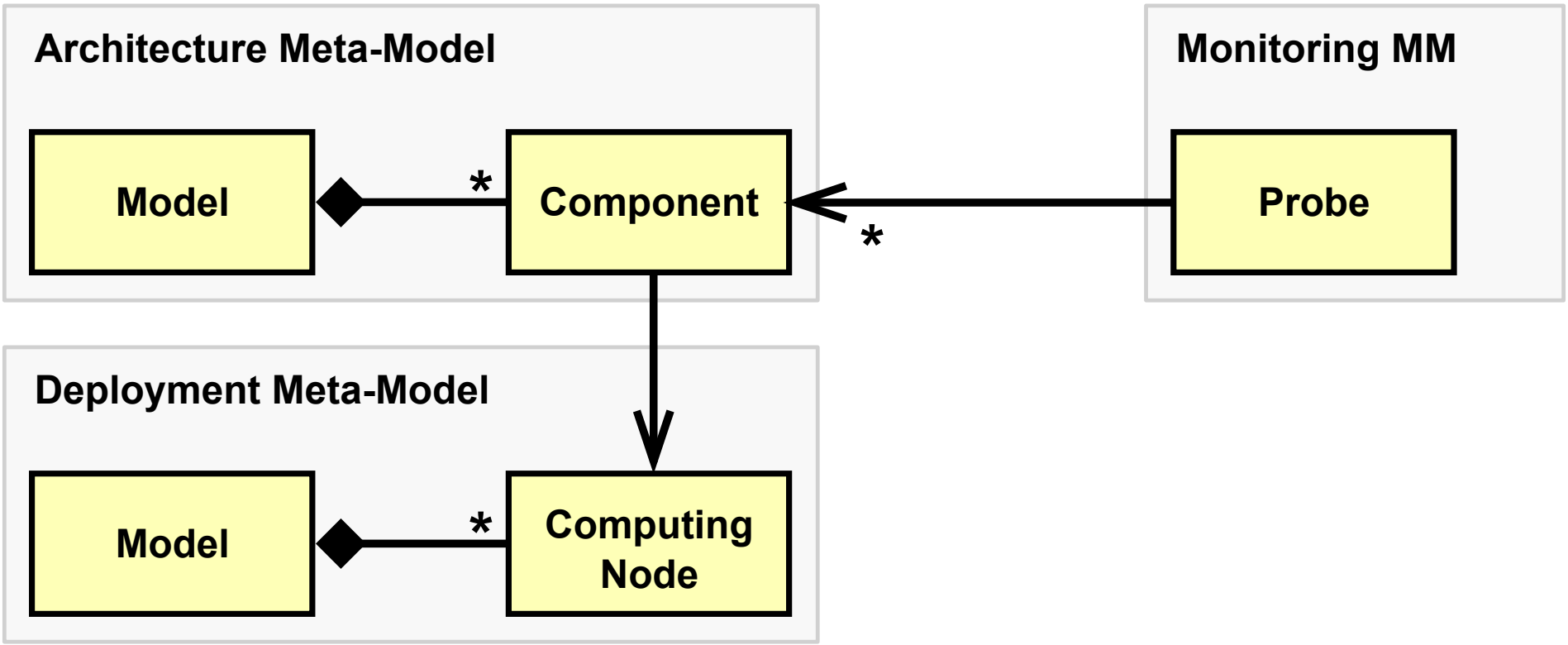
Aspect-oriented Modeling Roles



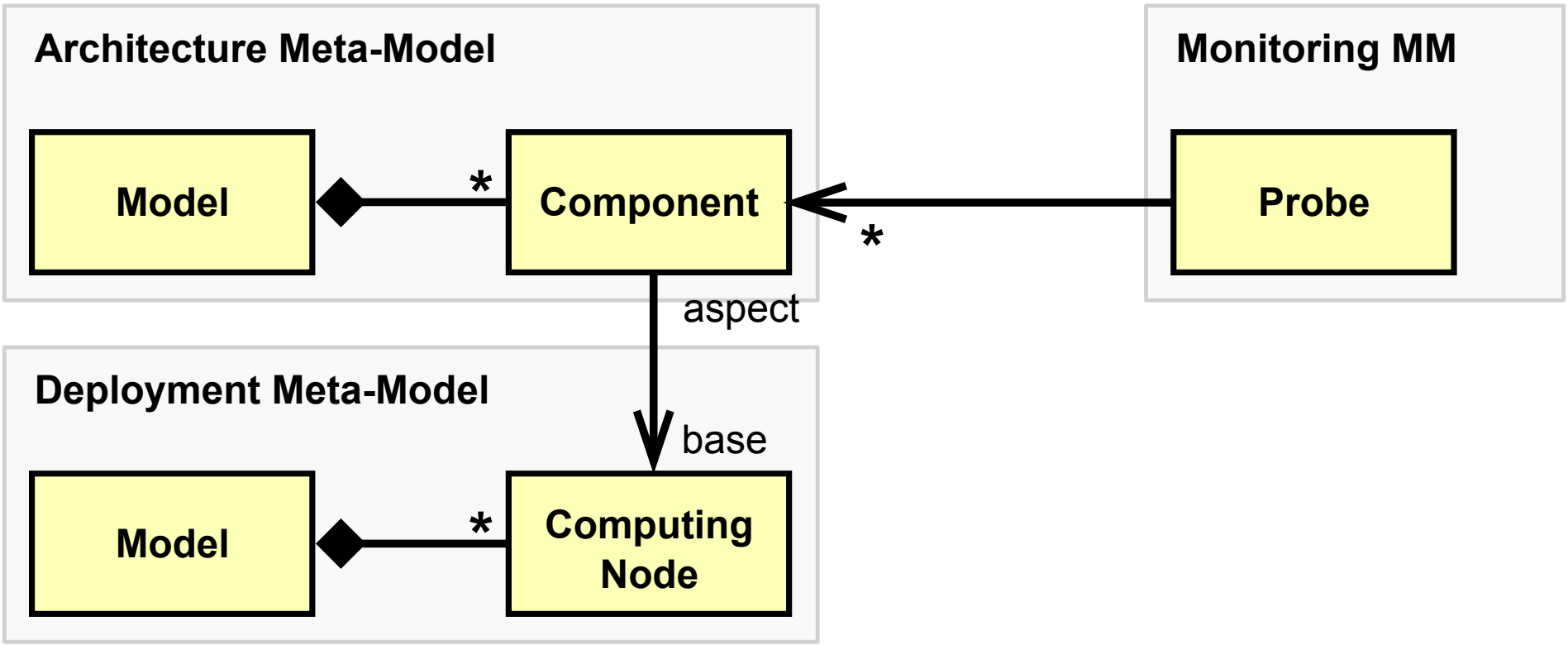
Aspect-oriented Modeling Roles



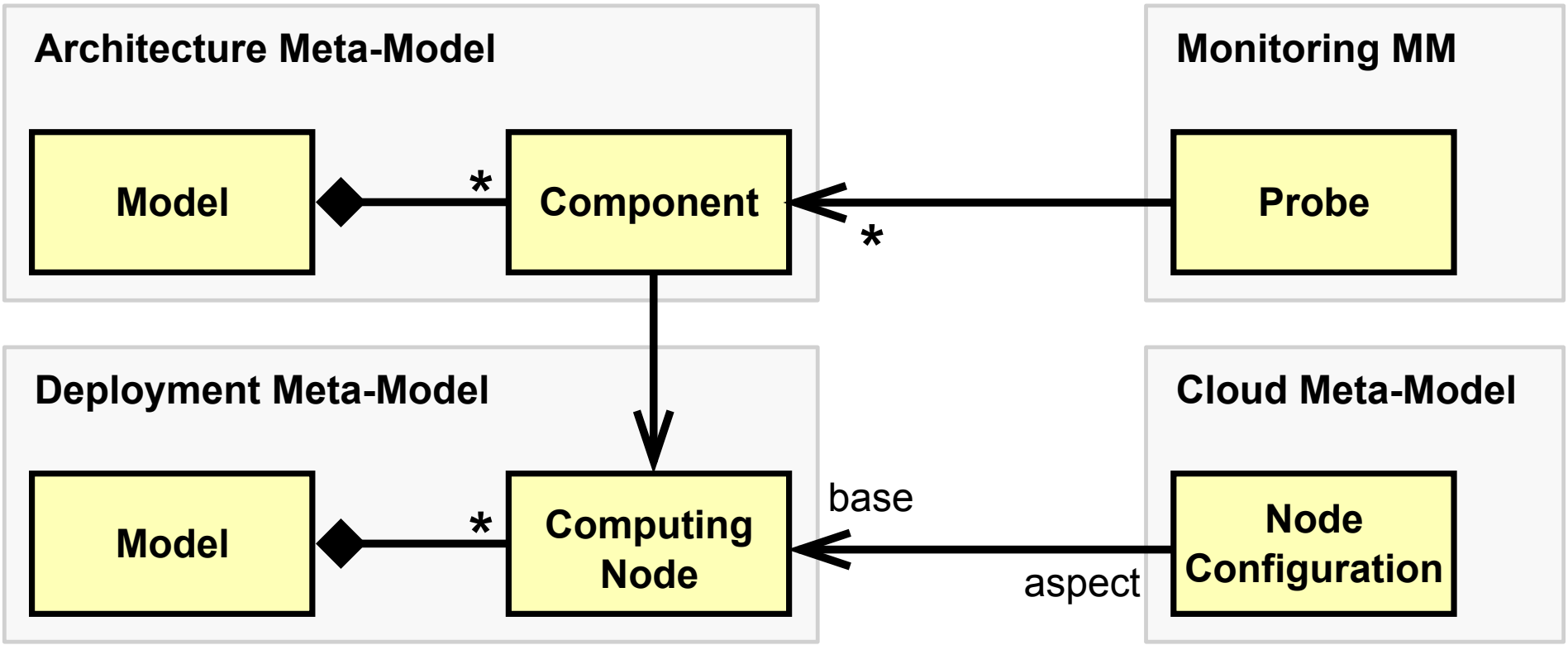
Aspect-oriented Modeling Roles



Aspect-oriented Modeling Roles

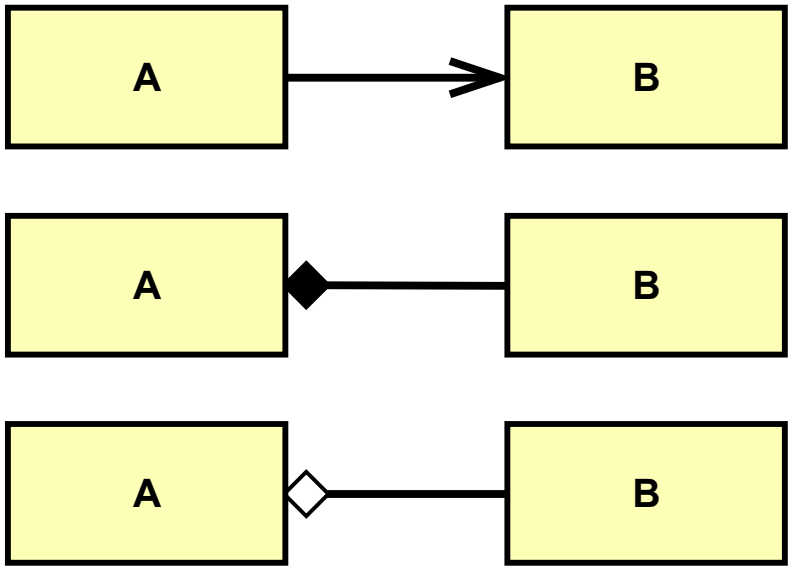


Aspect-oriented Modeling Roles



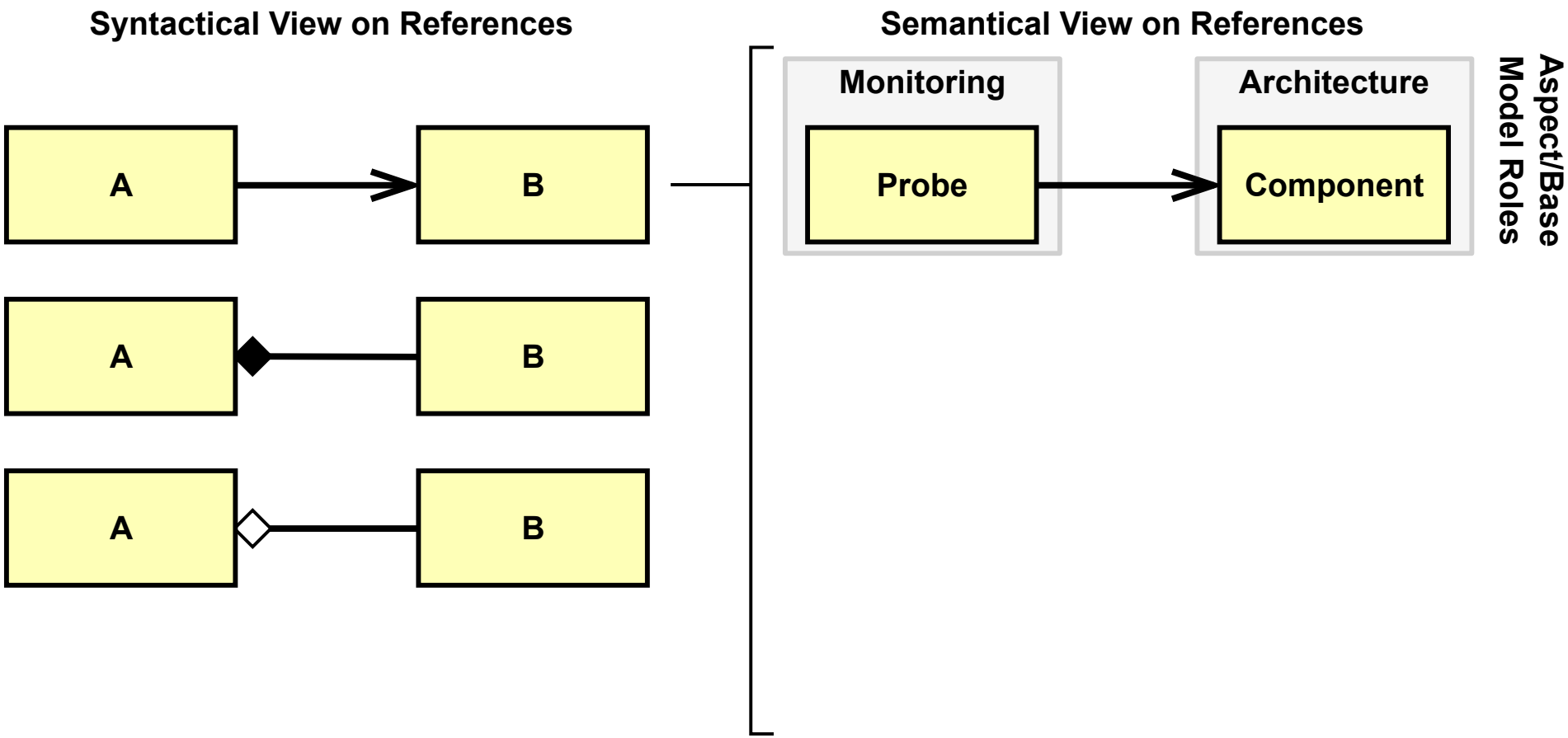
Semantic of References

Syntactical View on References

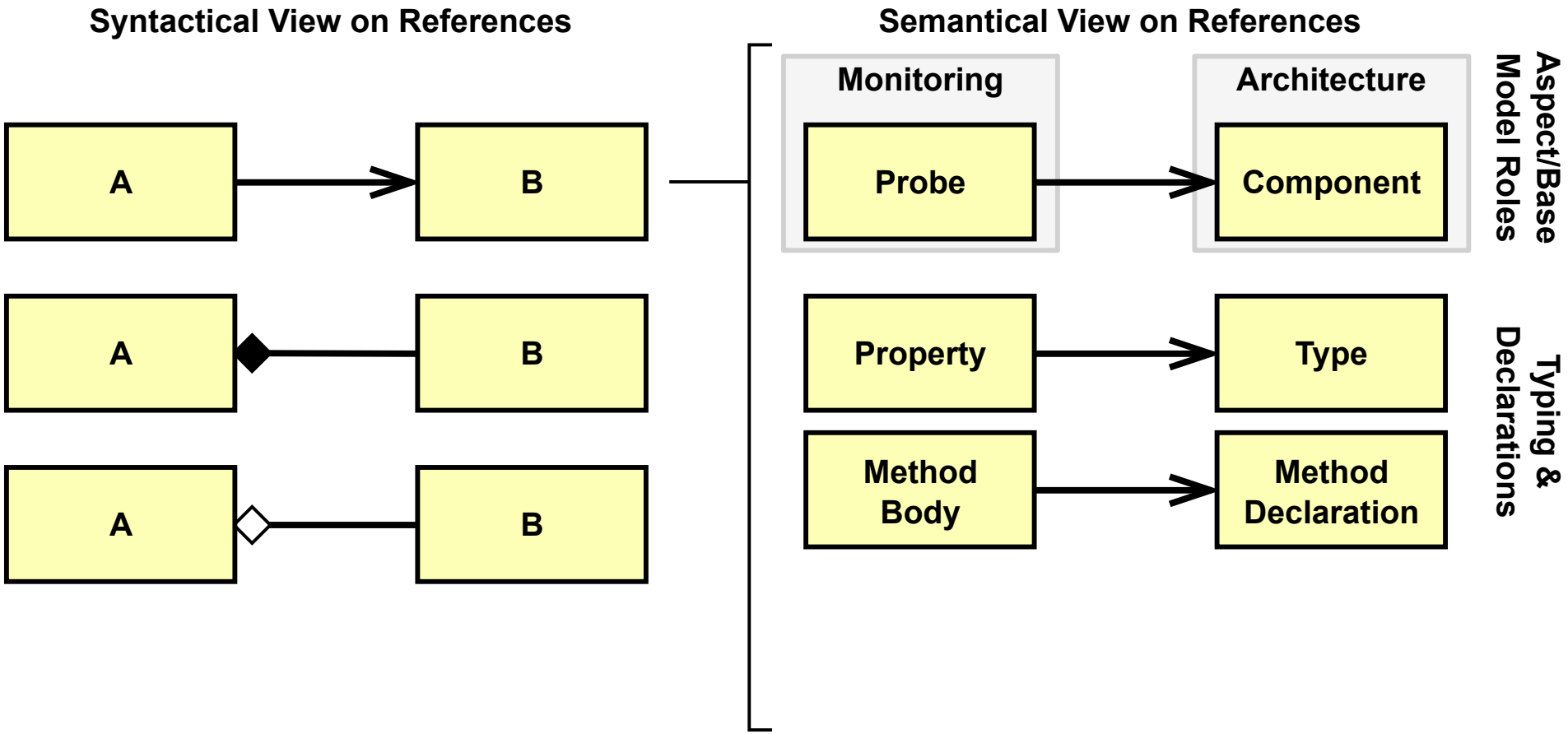


Semantical View on References

Semantic of References

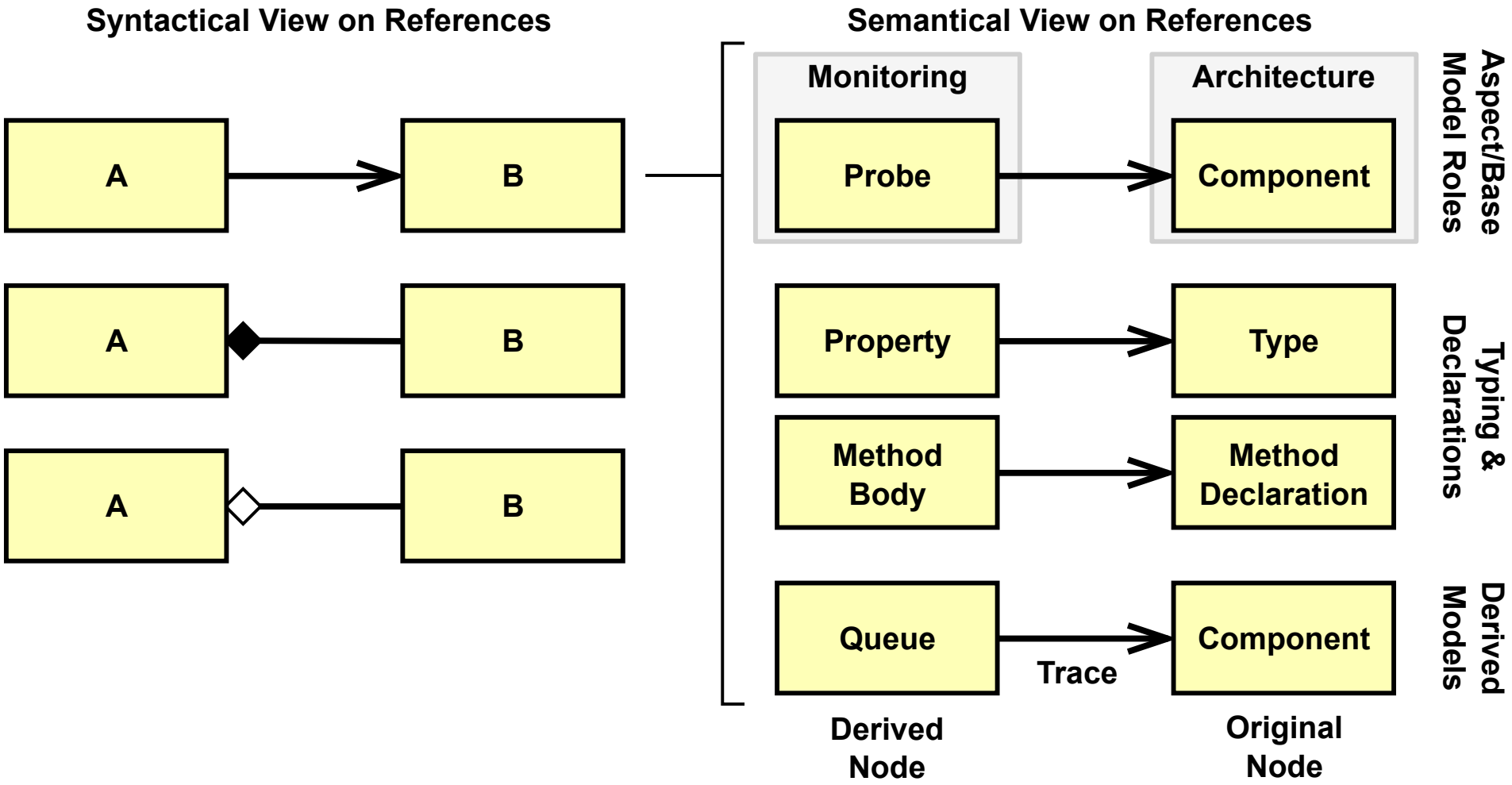


Semantic of References



Semantic of References

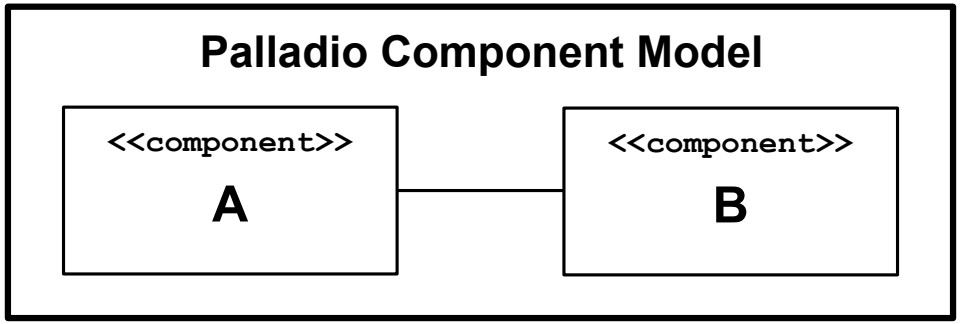
Method



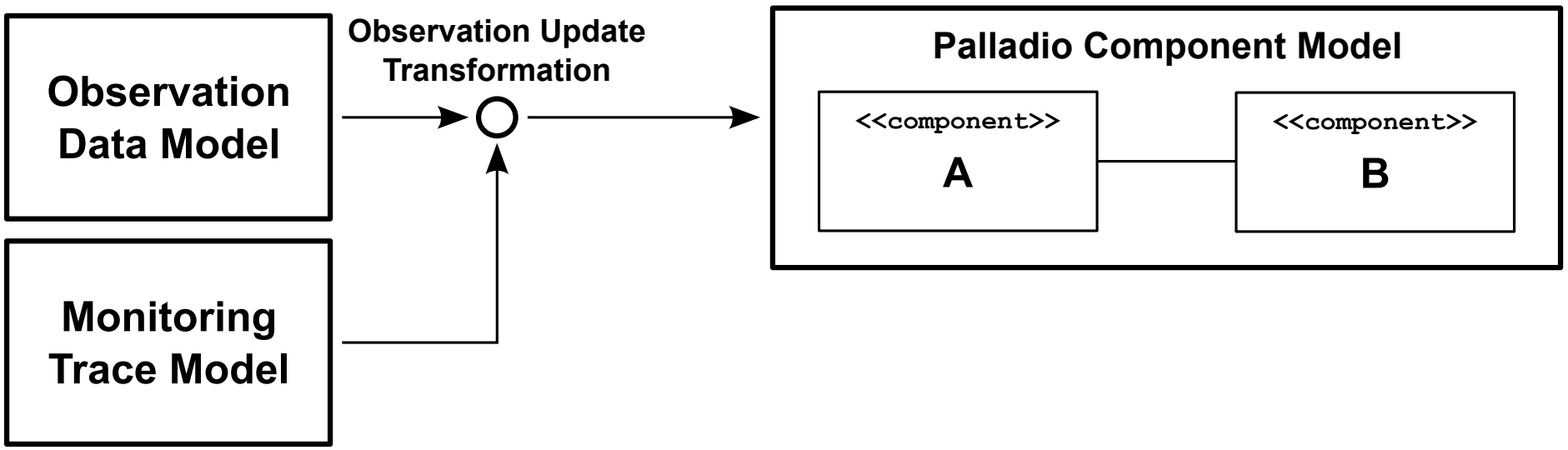
Contextual Meta-Model Pattern

Use Cases	**Pattern**
Models@Runtime	Tracability
Generators	Navigation
Editors	Queries
Simulation and Evaluation	Execution / Behavior
	Data
	State

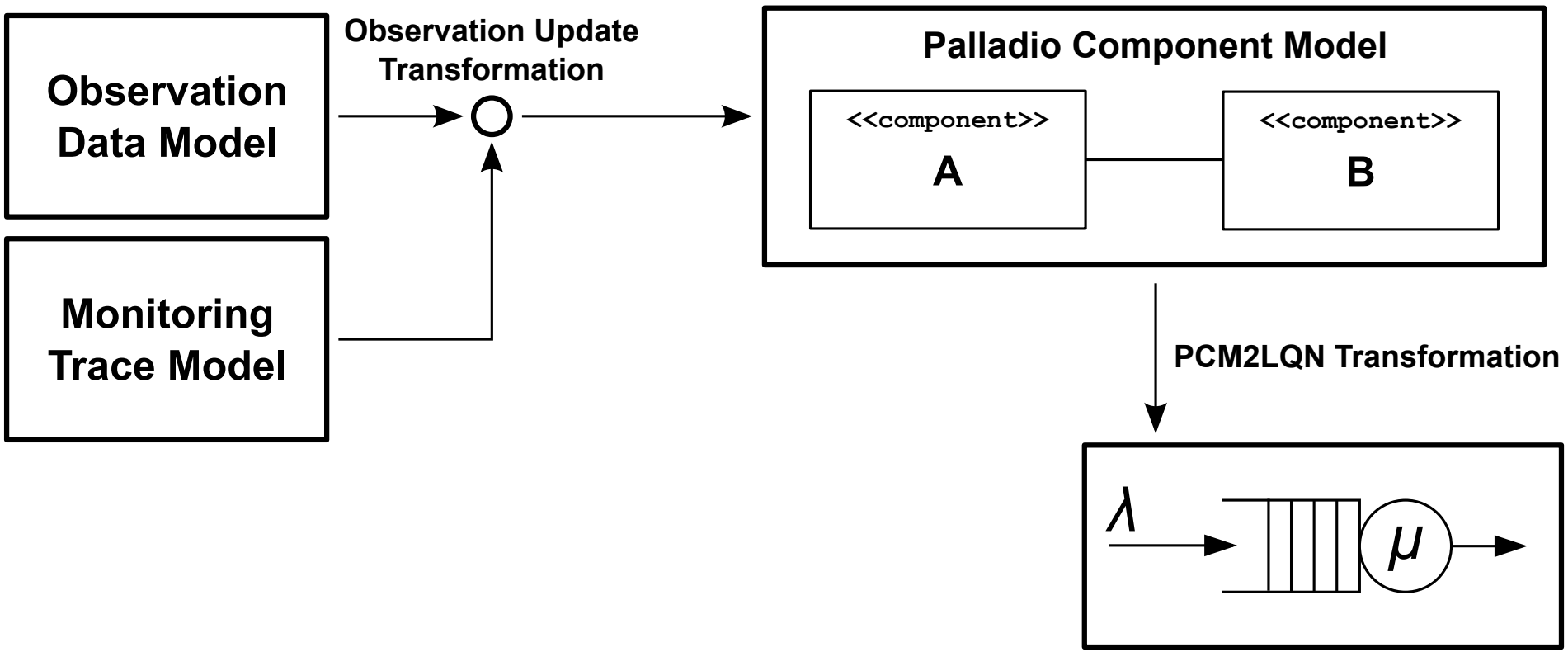
Use Case: Run-Time Model



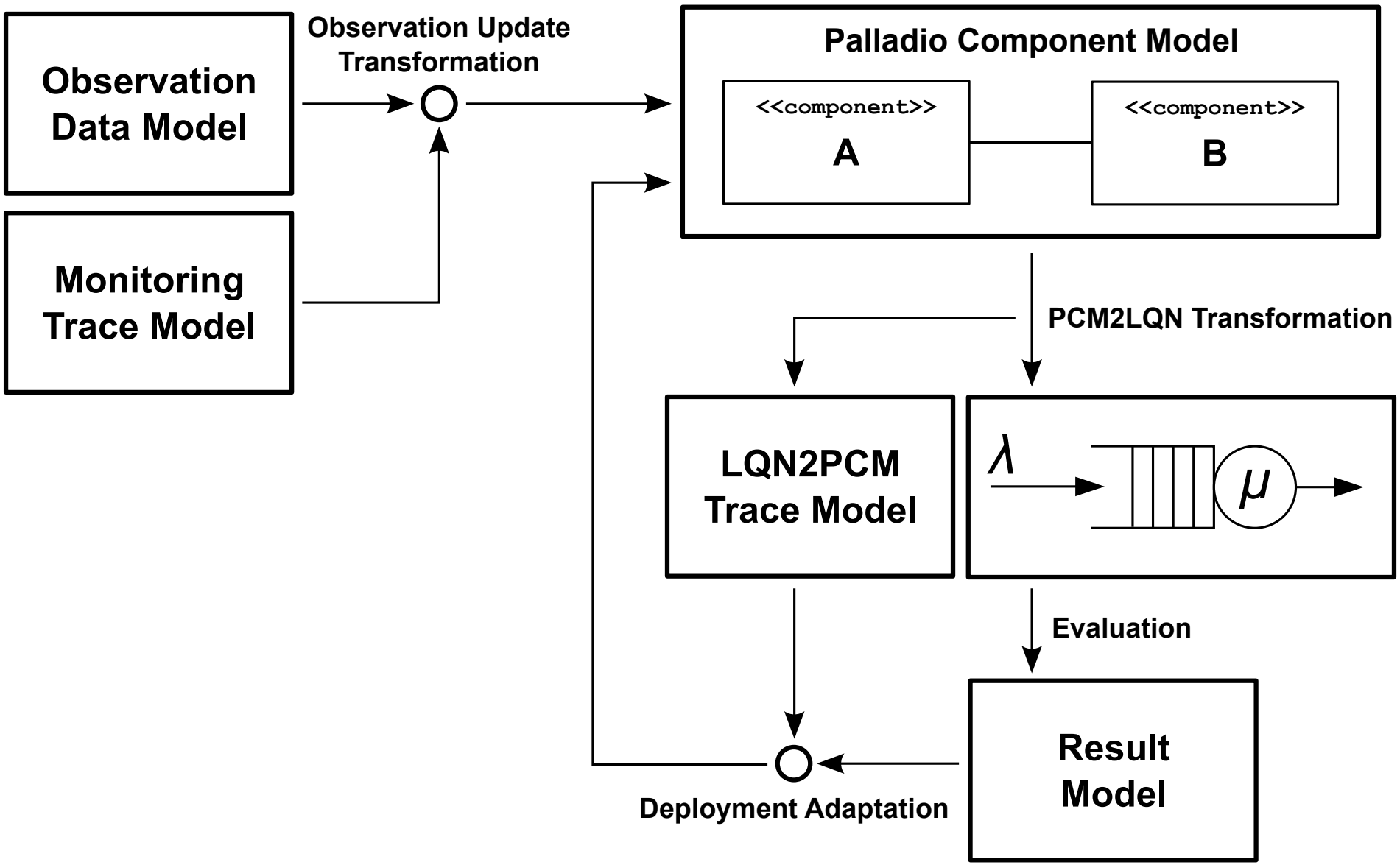
Use Case: Run-Time Model



Use Case: Run-Time Model



Use Case: Run-Time Model



Summary and Outlook

Summary

- **Non-invasive** annotation mechanism through AOM
- **Separate concerns** represented by separate meta-models
- Better understanding of **pattern and semantics** in meta-models
- Use EMF/Ecore as **formal notation** for meta-models
- No new meta-meta-model required
- **Compatible** with existing frameworks

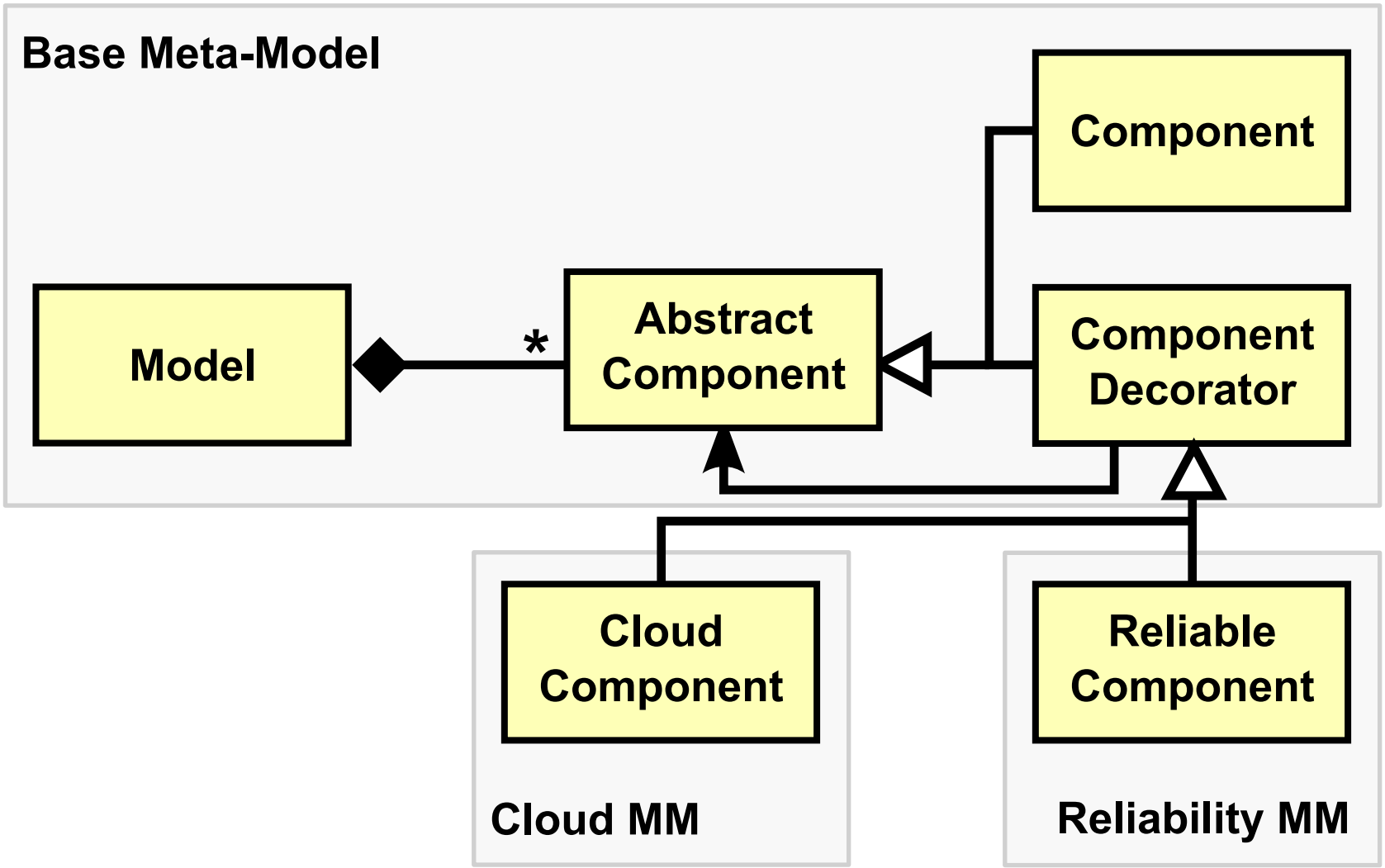
Outlook

- Detailed definition of meta-model pattern
  - Implications on tooling
- **Evaluation** of our method in a PCM modernization effort
  - Solid core PCM
  - Wide palette of extensions

Appendix

Decorator Pattern

[Gamma, Helm, Johnson, & Vlissides, 1994]



Features

- Provide a profile mechanism for EMF
- Multiple profiles can be annotated to one class
- Generic reusable profiles for reuse
- Comes with its own (meta-)meta-model

Downside No tool-integration

- Xtext DSL framework [XText, 2011]
- KLighD diagram and auto-layout framework [Schneider, Spönemann, & von Hanxleden, 2013]
- Genmodel facilities

References I

AG, I., & Foundation, E. [2011]. *Xtext.* http://www.eclipse.org/Xtext/.

Becker, S., Koziolek, H., & Reussner, R. [2009]. The Palladio component model for model-driven performance prediction. *J. of Systems and Software*, *82*, 3–22. Retrieved from `http://dx.doi.org/10.1016/j.jss.2008.03.066` doi: 10.1016/j.jss.2008.03.066

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. [1994]. *Design patterns: elements of reusable object-oriented software*. Pearson Education.

Ghezzi, C. [2011, Sept]. The fading boundary between development time and run time. In *Web services (ecows), 2011 9th ieee european conf. on* (p. 11). doi: 10.1109/ECOWS.2011.33

Langer, P., Wieland, K., Wimmer, M., & Cabot, J. [2012]. EMF Profiles: A lightweight extension approach for emf models. *JOT*, *11*(1), 1-29. Retrieved from `http://dblp.uni-trier.de/db/journals/jot/jot11.html#LangerWWC12`

References II

Schneider, C., Spönemann, M., & von Hanxleden, R. [2013, 15–19 September]. Just model! – Putting automatic synthesis of node-link-diagrams into practice. In *Proceedings of vl/hcc.* San Jose, CA, USA.

Steinberg, D., Budinsky, F., Paternostro, M., & Merks, E. [2009]. *Emf: Eclipse modeling framework* (2. ed.). Boston, MA: Addison-Wesley. Retrieved from `http://my.safaribooksonline.com/9780321331885`