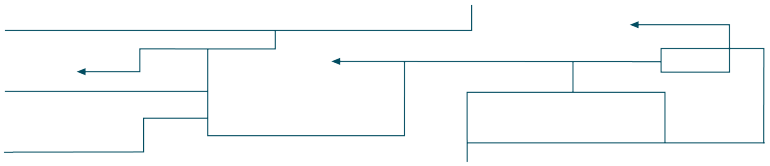
DAI-Labor
TU Berlin



UI Models at Runtime

Grzegorz Lehmann

DAI-Labor
Fakultät IV – Elektrotechnik und Informatik
Technische Universität Berlin



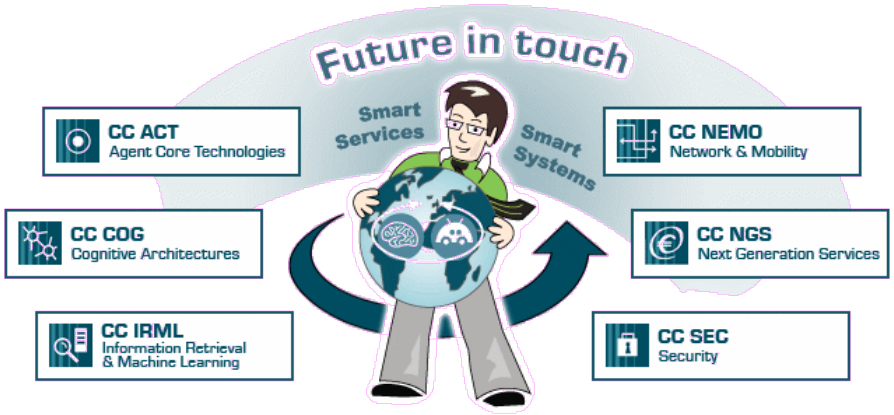
CC NGS
Next Generation Services



DAI-Labor Details

- DAI = Distributed Artificial Intelligence Laboratory
- Head: Prof. Dr. Sahin Albayrak
- ~100 researchers (postdocs, ph.d. & student assistants)
- Bridging industry and research

- 6 Competence Centers (CC)
  Agent Core Technologies,
  Security,
  Information Retrieval and Machine Learning,
  Networks and Mobility,
  Cognitive Architectures,
  Next Generation Services (NGS)



- NGS works with
  - Ambient Assisted Living
  - Smart Environments / Smart Homes
  - User Centric Systems
  - Engineering of Interactive Systems

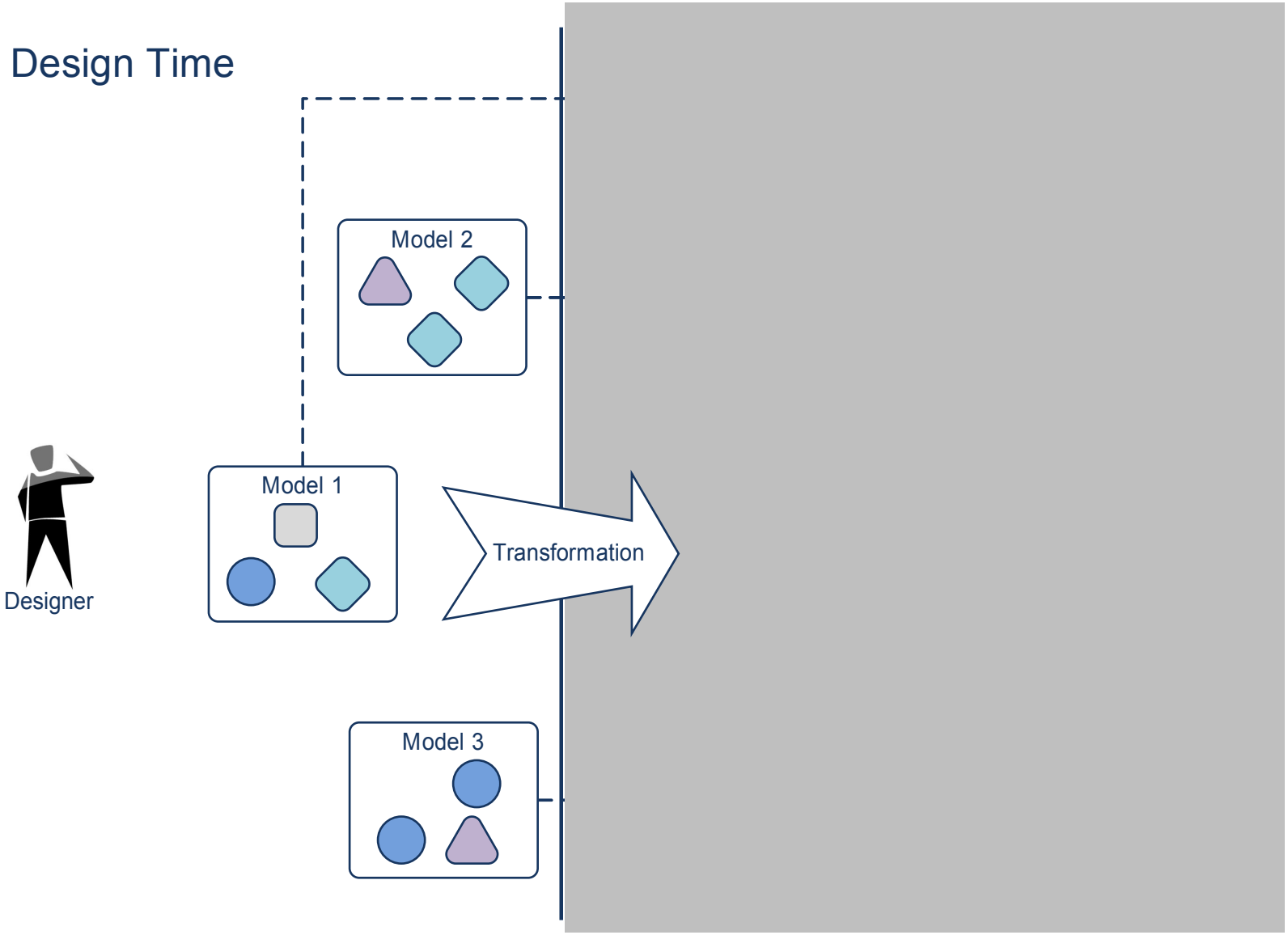
- See www.dai-labor.de for running projects, labs and testbeds

User Interfaces for Smart Environments

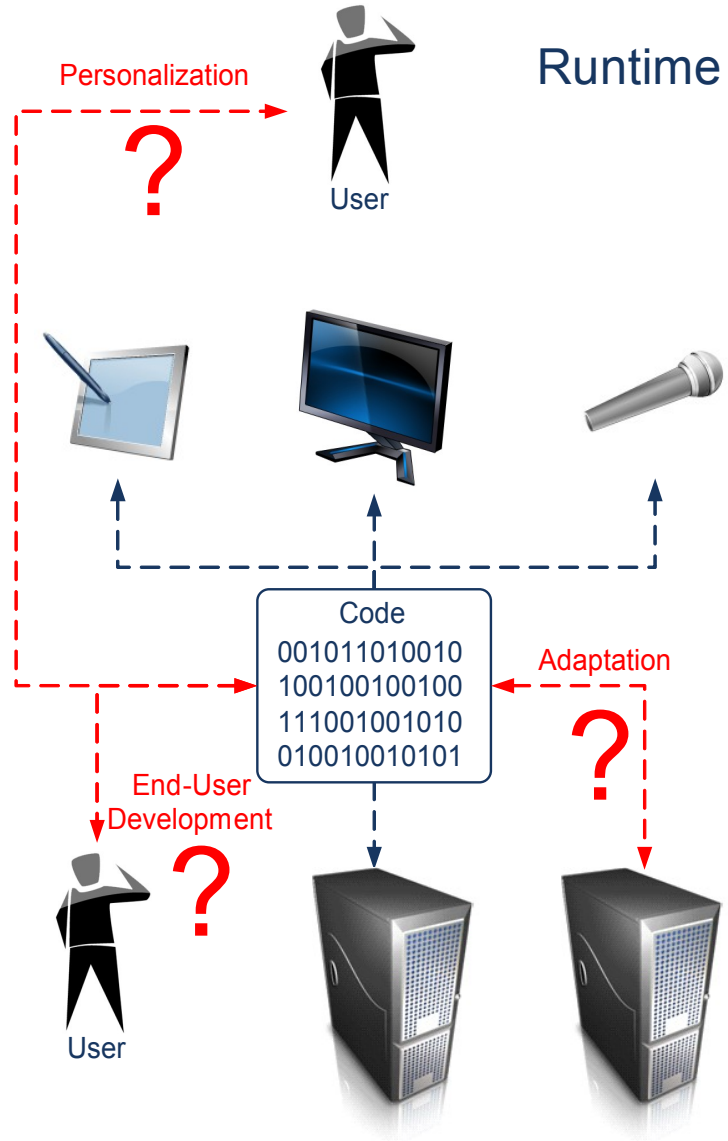
DAI-Labor
TU Berlin

- High heterogeneity and dynamics:
  - Interaction devices are unknown at design time
  - Users are unknown
  - Environment is unknown
- Requirements:
  - Personalization
  - Adaptation
  - End-User Development

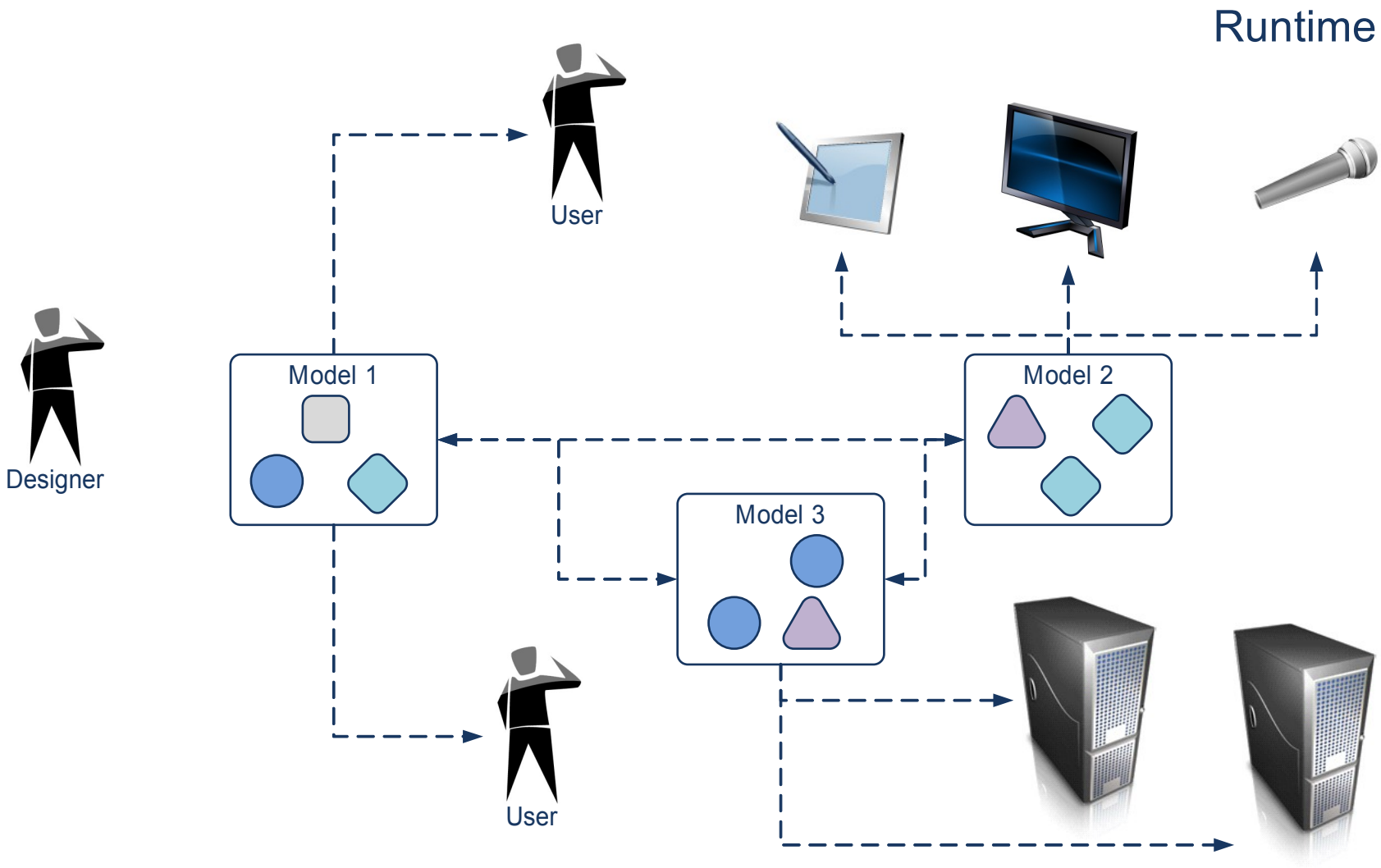
UI Modeling at Design Time



Design Rationale is Missing at Runtime



Runtime UI Models

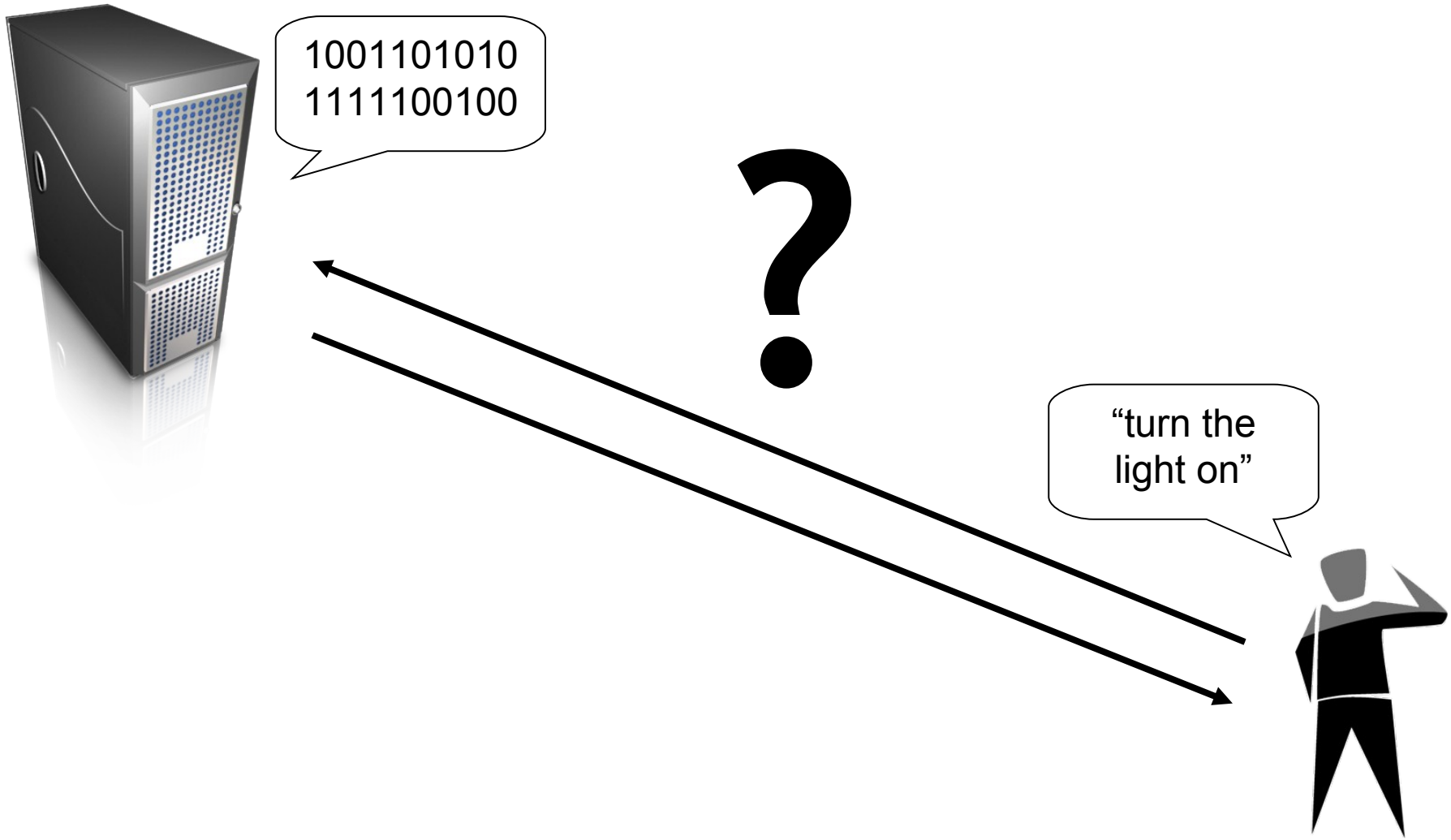


DAI-Labor
TU Berlin

Demonstration

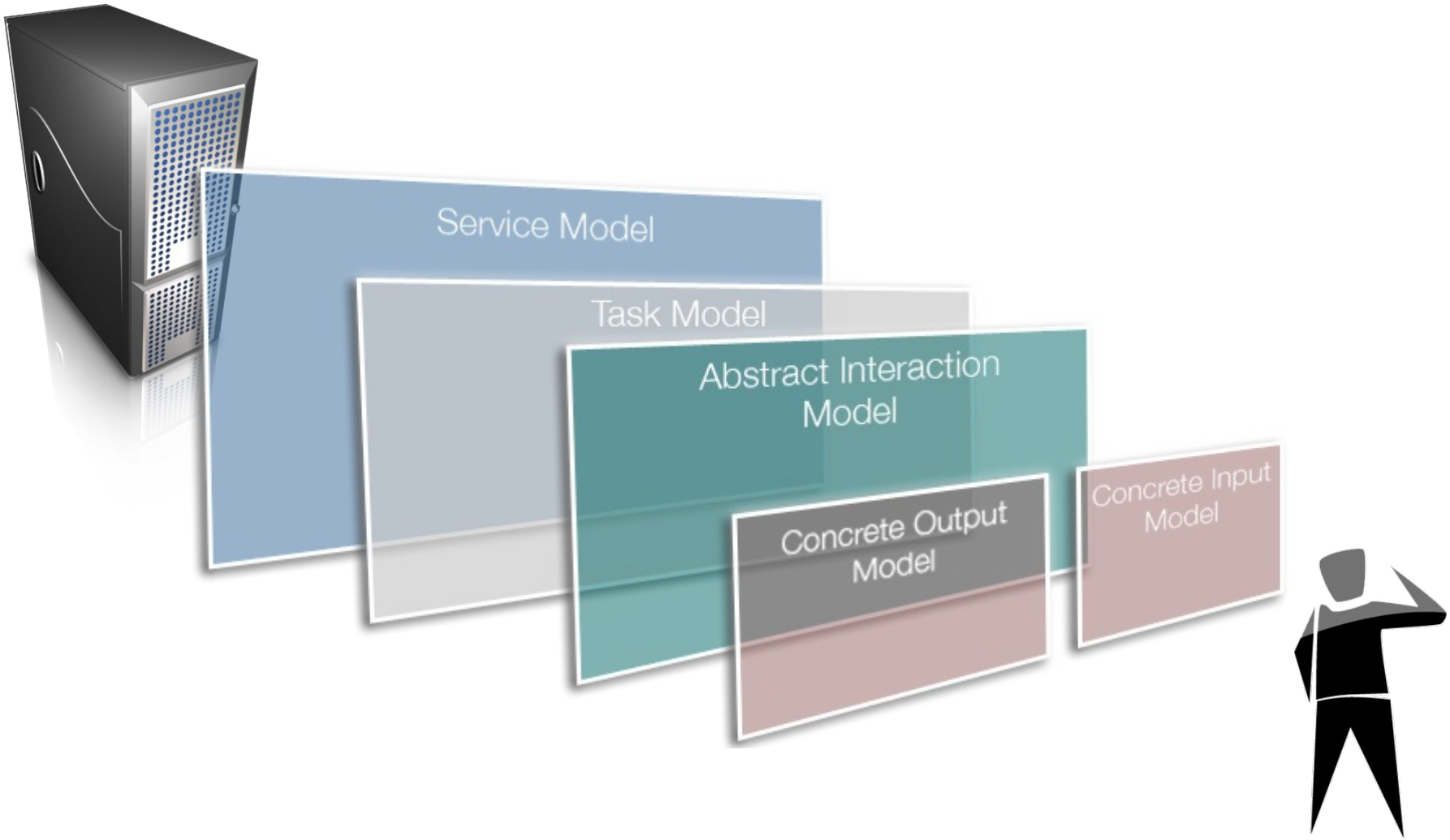
http://www.youtube.com/watch?v=HLHKTYniVDU

Mediating between human and computer



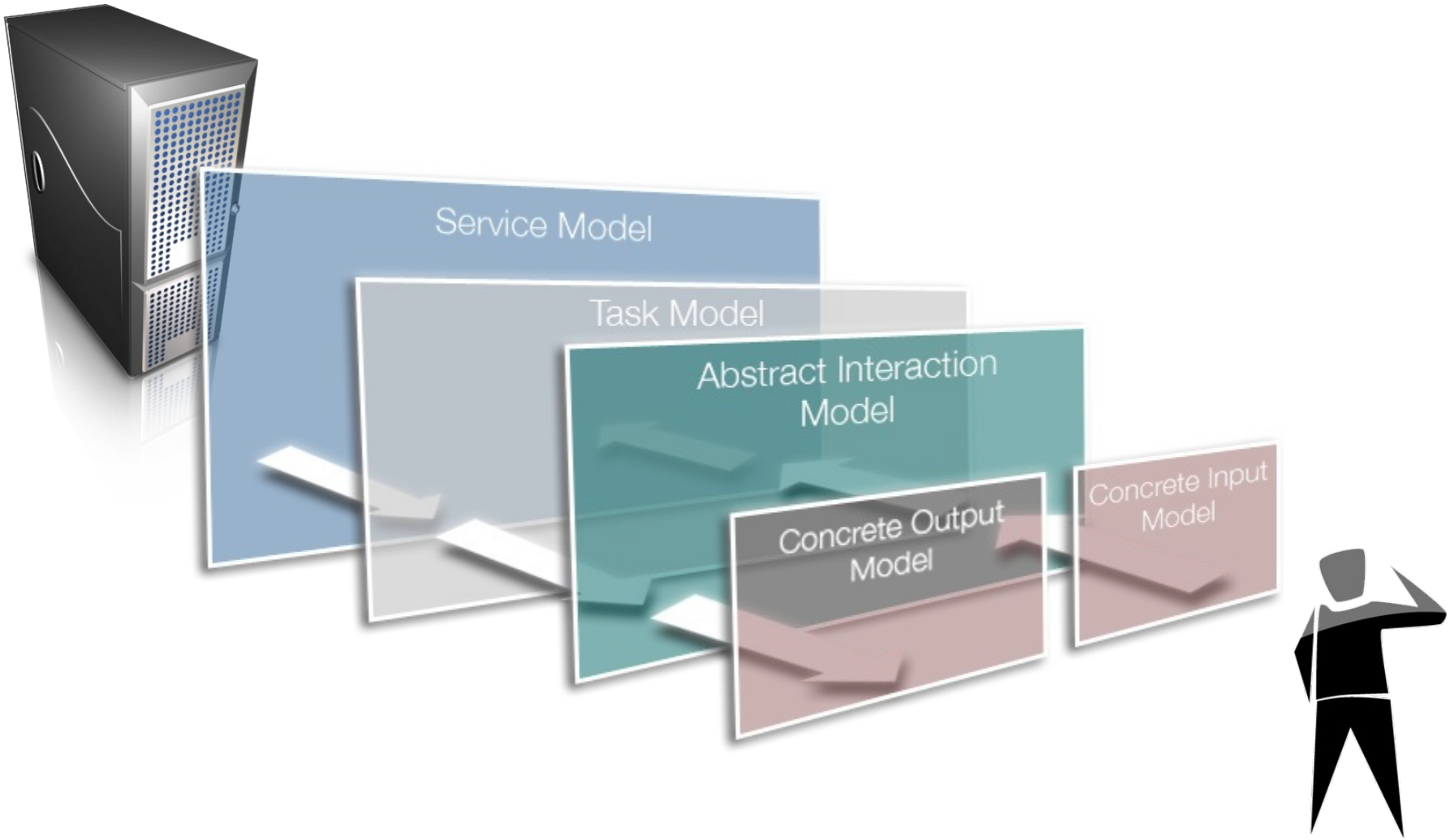
Network of UI Models at Runtime

DAI-Labor
TU Berlin

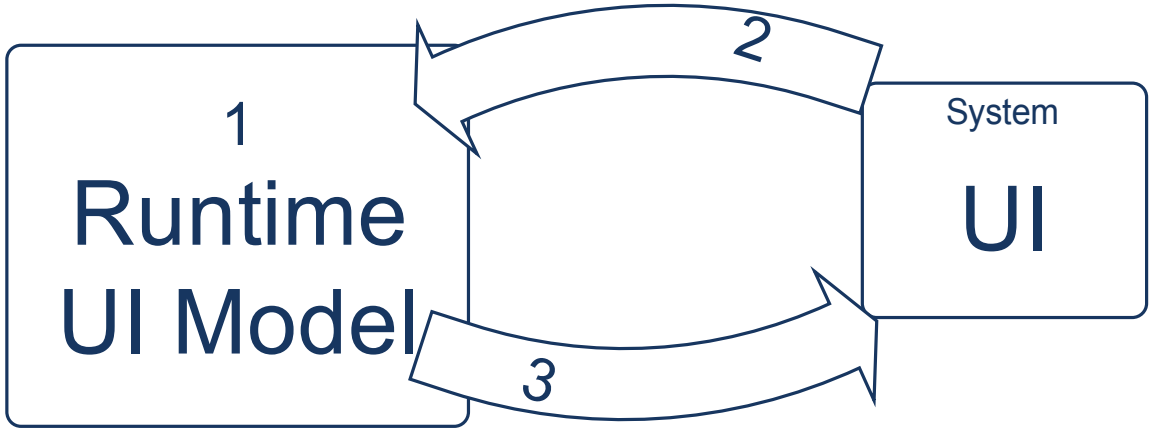


Abstraction of User Input and Reification of System Responses

DAI-Labor
TU Berlin



Roles of run-time UI models



1. Make the design rationale available at runtime
2. Provide an abstract view on the UI and its state at runtime
3. Provide means of influencing the UI at runtime

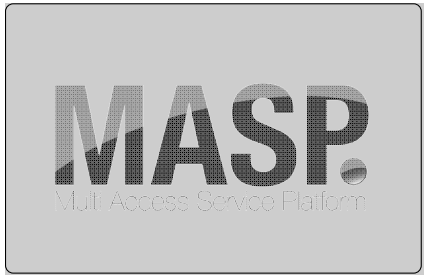
Some open issues

- How can we distinct runtime and design time information in runtime UI models?
- How can the runtime information flow into the models so they are up-to-date at runtime?
- What does a UI language contain if a ubiquitous UI has no fixed size, no fixed interaction device, no fixed user and is not executed in a fixed context?
- What parts of models can be adapted automatically at runtime?
- How can we evaluate the usability of the adapted applications?

The End …

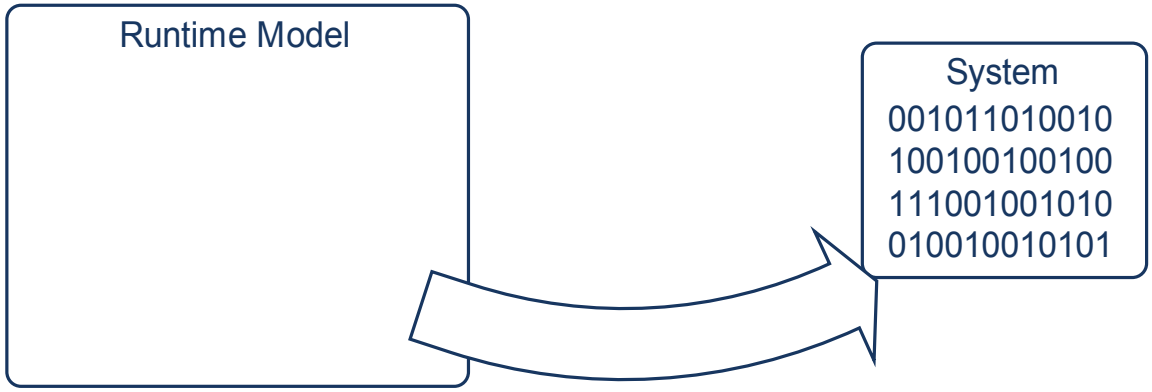
Your questions please …

grzegorz.lehmann@dai-labor.de
http://masp.dai-labor.de



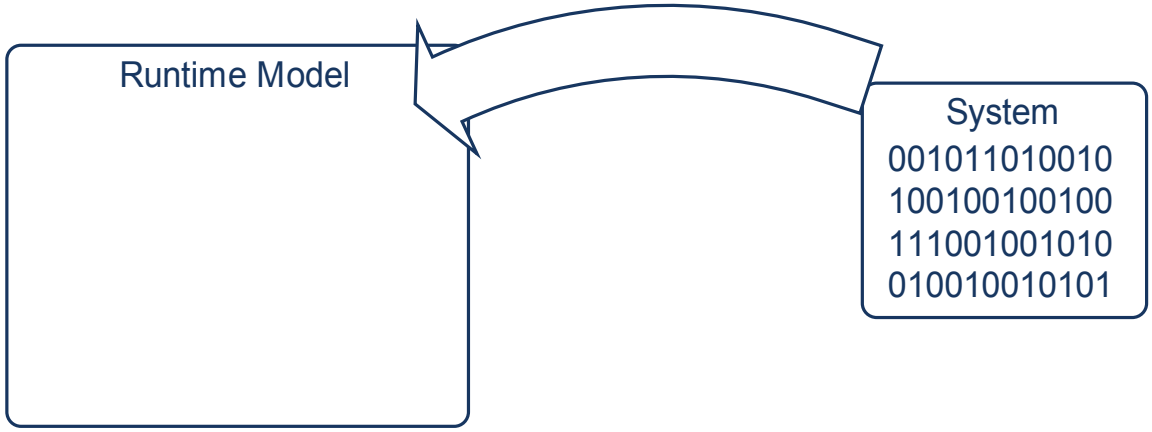
ACM SIGCHI Symposium on
Engineering Interactive Computing Systems
June 21-23, 2010
http://eics-conference.org

Model to System Connection at Runtime



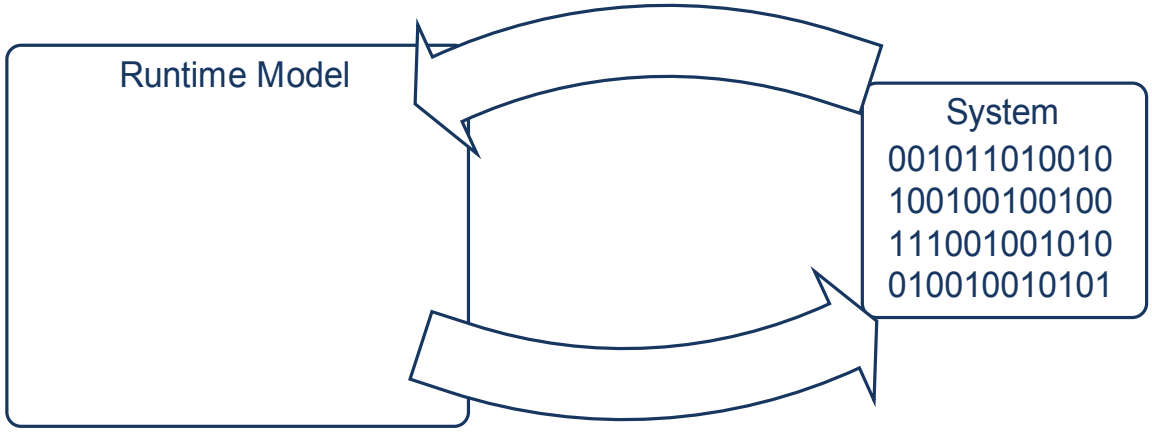
- Common in large, (self-) adaptive systems
- Staikopoulos et al., Mutual dynamic adaptation of models and service enactment in alive*, 2008:
  - Adaptations performed on the running system via transformations of the system model
- Kuhn and Verwaest, Fame, a polyglot library for meta-modeling at runtime, 2008
  - FAME (Polyglot Library)
  - Adaptation of software at runtime through modifications of models and meta-models

System to Model Connection at Runtime



- Many approaches based on state charts and stateful model elements
- Monitoring state machines enables debugging and tracing of occurrences in the system on model level
- Maoz, Model-Based Traces, 2008
  - Model is updated at runtime via traces
- Graf and Müller-Glaser, Gaining insight into executable models during runtime: Architecture and mappings, 2007
  - Driver Layer between the model and the system with a set of operations
  - Inspecting and debugging model-based embedded systems at runtime

Model-System Cycle at Runtime



- Cycle between the model and the system
  - Models reflect the state of the system
  - The system reacts to changes in the model
- Blair et al., Models@Run.time, 2009:
  - model@run.time is a causally connected self-representation of the associated system