TECHNISCHE UNIVERSITÄT DRESDEN

DRESDEN concept

Models@run.time for Object-Relational Mapping Supporting Schema Evolution

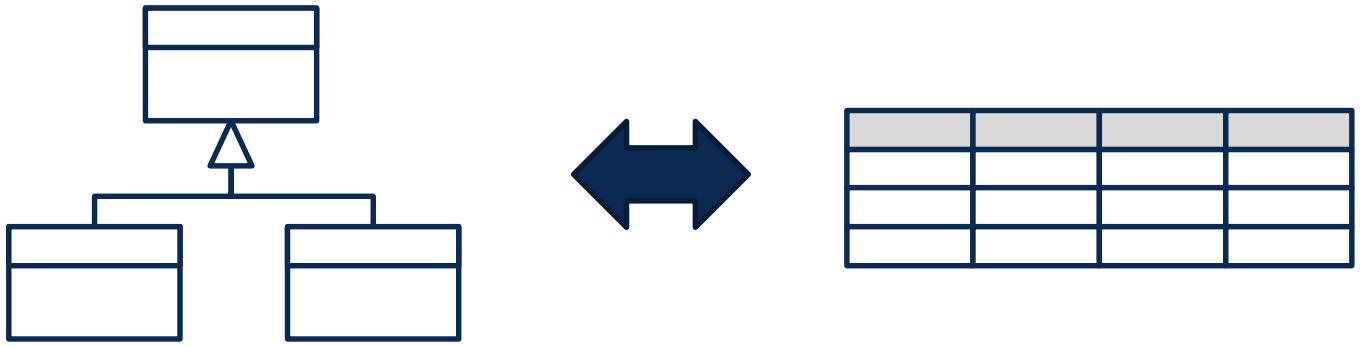
Sebastian Götz and Thomas Kühn

10th International Workshop on Models@run.time.

29th September 2015, Ottawa, Canada

Motivation

- **Context**: Enterprise Software following the three-tier architecture (Presentation, Business Logic, Data)
- In 2013, the worldwide ERP software market was $25.4B [1]
- A key time-consuming task in developing ERP software is the mapping between business logic and data management.
- To partially automate this translation, object-relational mappers (ORM) have been introduced (e.g., Hibernate)
- ORMs translate between the object-oriented and the relational paradigm, which are the most common paradigms in use for Enterprise Software.

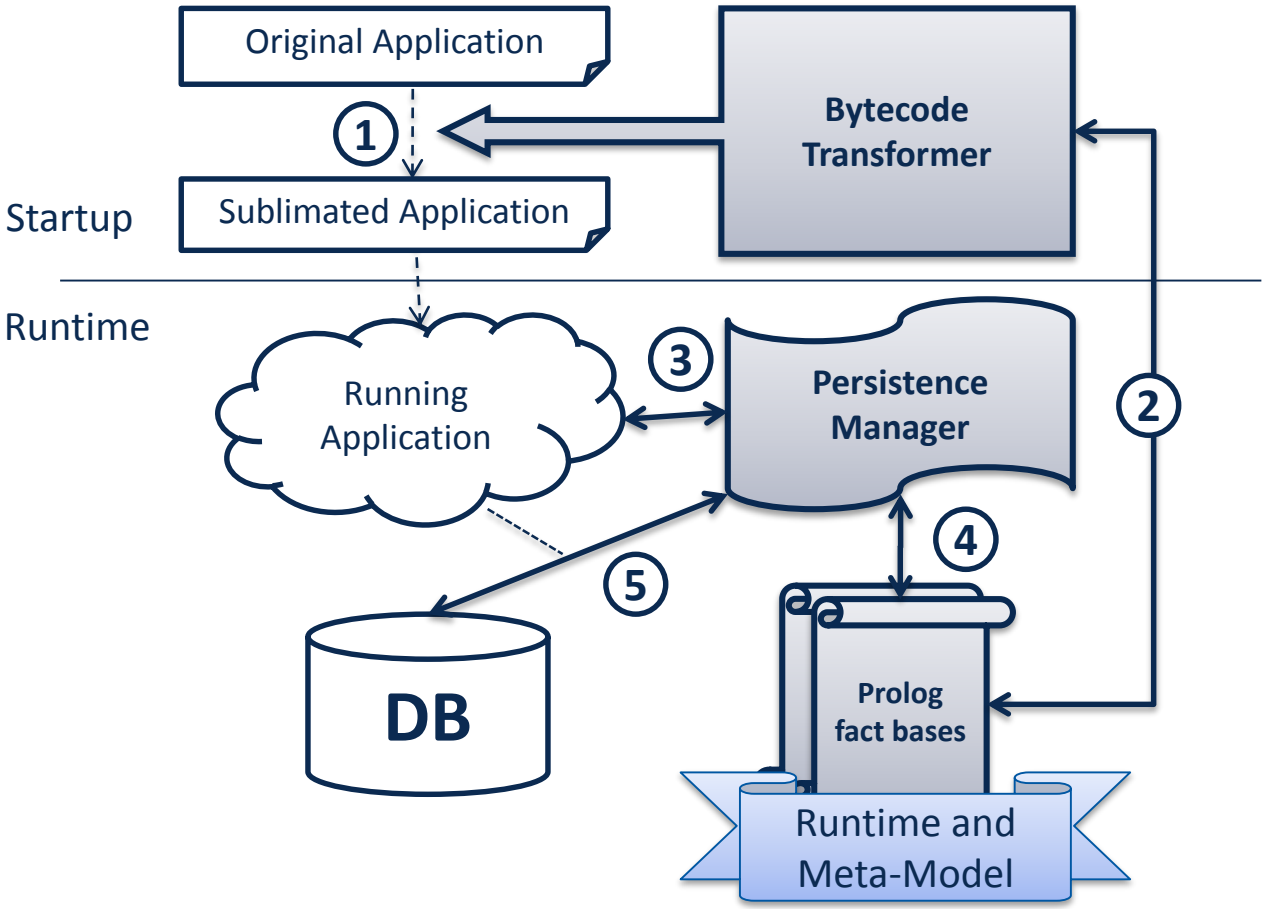


[1] Forbes: Gartner's ERP Market Share Update Shows The Future Of Cloud ERP Is Now. 12th May 2014.

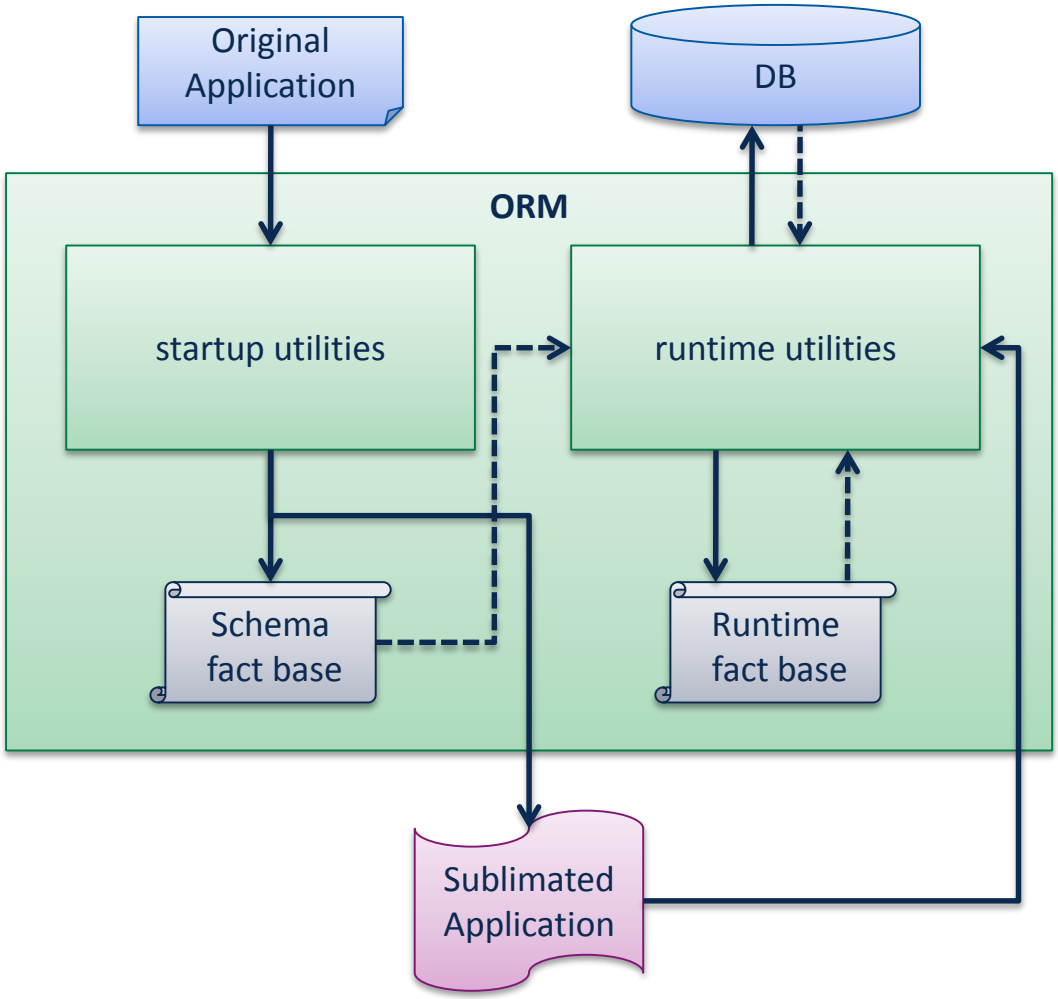
Problems

- The problems of current ORM solutions are:
  - High configuration effort (time intensive, prone to error)
    - Either in XML files
    - Or as annotations in code
    - Typically demands for manual tuning
  - Lacking support for continuous development
    - Data of previous versions easily gets lost or inaccessible due to schema changes
    - But, current software engineering processes (e.g., agile or lean SE) demand for small increments

Solution: Models@run.time for ORMs



Models@run.time for ORMs



Example:

Schema Fact Base

```
isClass ('Student').
hasAttribute ('Student','studentid','int',0).
hasAttribute ('Student','name','java.lang.String',1).
hasAttribute ('Student','__oid','int',2).
```

Runtime Fact Base

```
instanceof('Student',[ - , - ,1]).
instanceof('Student',[300 , - ,1]).
instanceof('Student',[300 , 'John' ,1])
```

The Runtime and Metamodel

Schema fact base (alias Metamodel):

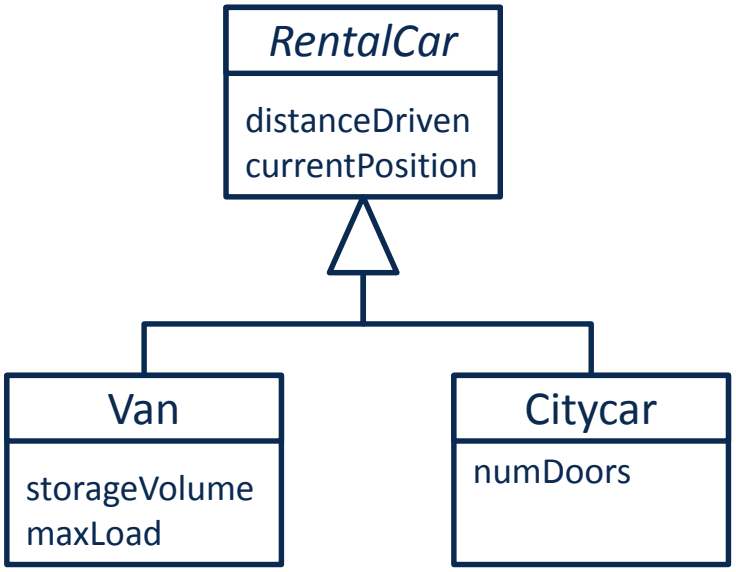
- isClass/1,
- hasAttribute/4, hasStaticAttribute/4
- subclasses/2, references/4
- Remaining fact types related to changes (e.g., addedAttribute/4)

Runtime fact base (alias Runtime Model):

- instanceof/2
- sameInstance/4

Low Configuration Effort due to Runtime Model:

- Types of Relationships can be inferred
- Best inheritance mapping can be inferred (and changed at runtime)
  - Imagine in the beginning only/mostly citycars are requested by customers
  - Then the DB only has to keep one table in memory



Citycar

id	dist	curPos	doors

Van

id	dist	curPos	storage	load

VS.

RentalCar

id	dist	curPos	storage	load	doors

Support for Continuous Development

- At each application startup, the approach compares the new application schema with the old and derives the changes
- These changes are applied to the runtime fact base, which keeps the data across restarts by default (can be deactivated for productive use)
- The old database will be replaced by a new one, generated from the new runtime and schema fact base

Conclusion and Future Work

- Models@run.time help to reduce software development time for Enterprise Software demanding object-relational mapping by:
  - Reducing the configuration effort
  - Supporting continuous development
- Future Work
  - The approach is to be evaluated using a real-world case study
  - Results from model co-evolution and database co-evolution should be integrated (to avoid regenerating the database)

Contact

http://st.inf.tu-dresden.de/~sgoetz

sebastian.goetz@acm.org

Software Technology Group

HAEC

ROSI

cfaed CENTER FOR ADVANCING ELECTRONICS DRESDEN