

# Extending an ORDBMS: The StateMachine Module

Wolfgang Mahnke<sup>1</sup>

Christian Mathis<sup>1</sup>

Hans-Peter Steiert<sup>2</sup>

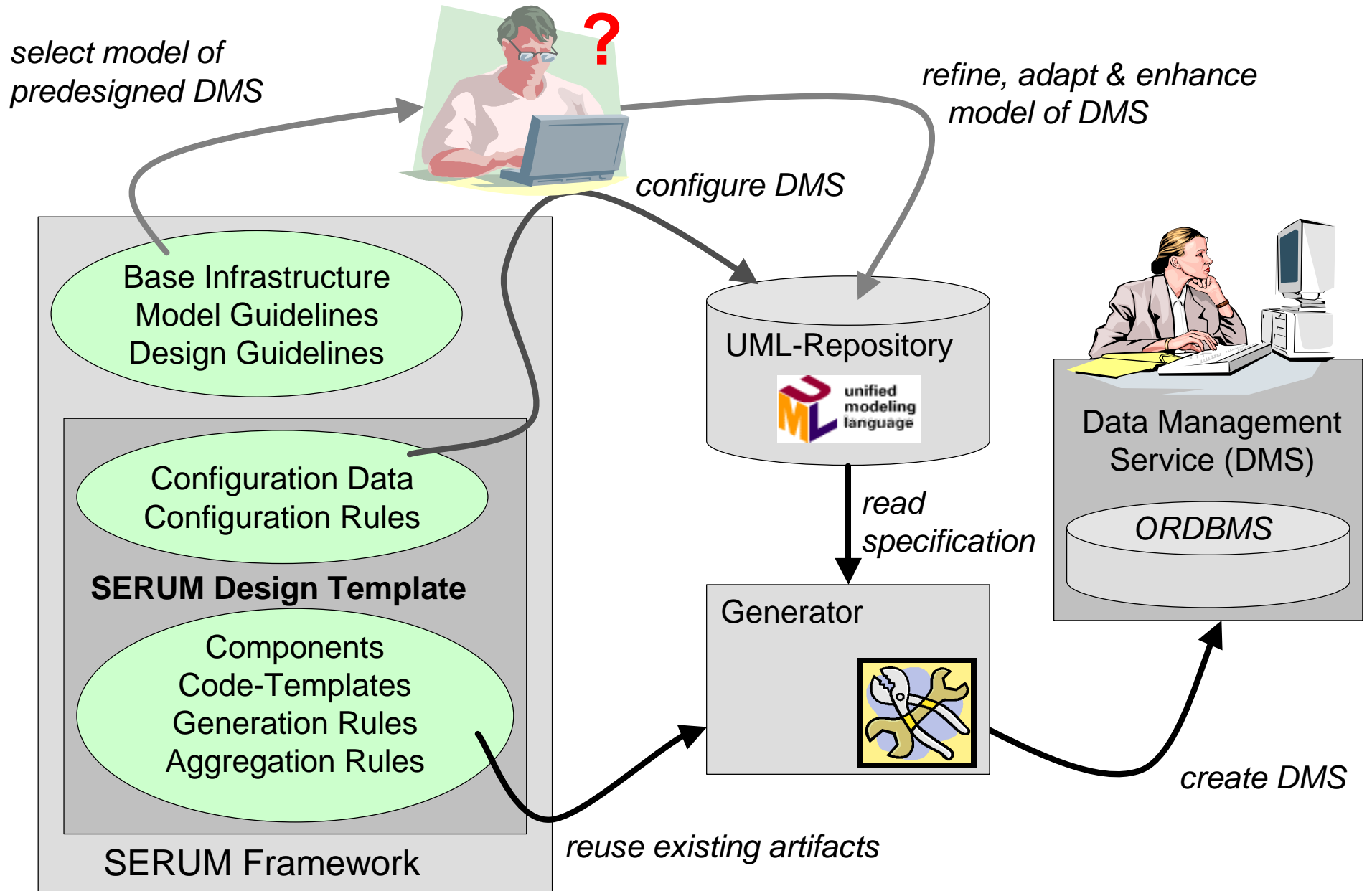
<sup>1</sup>University of Kaiserslautern  
Kaiserslautern, Germany

<sup>2</sup>DaimlerChrysler AG  
Research & Technology  
Ulm, Germany

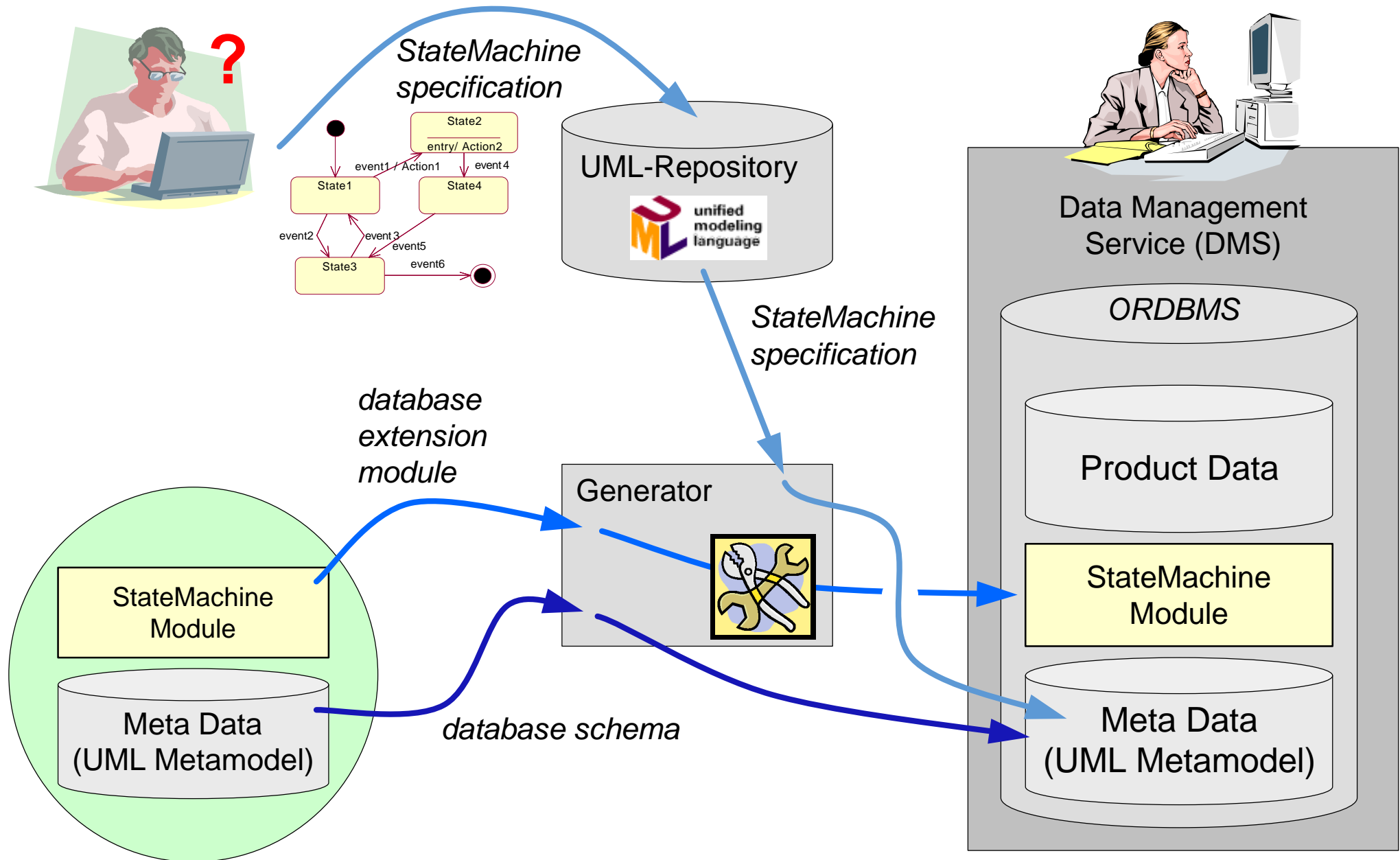


DAIMLERCHRYSLER

# Generating Software Engineering Repositories using UML



# SERUM: Mapping the Dynamic Model

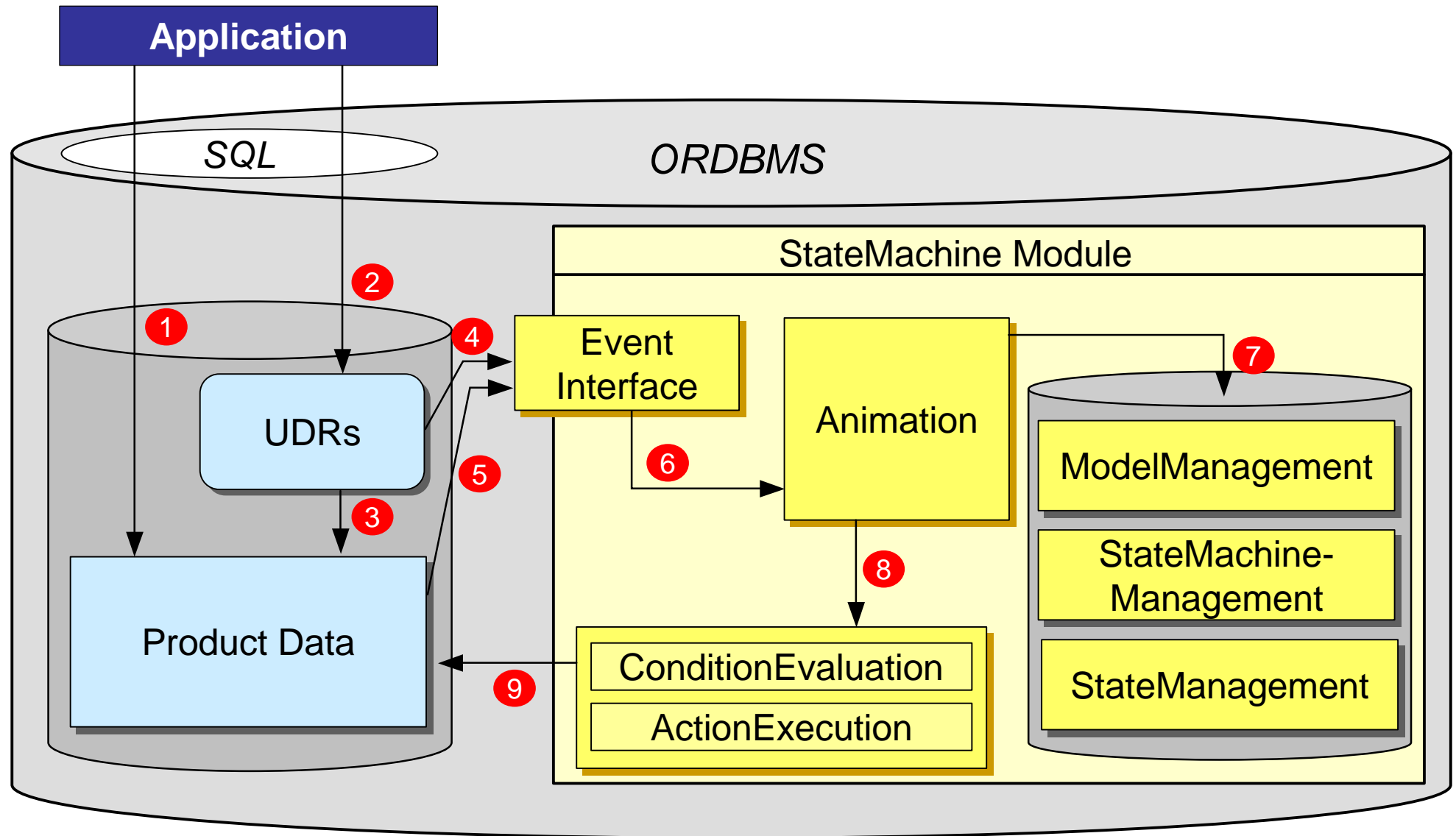


# Goals of the StateMachine Module

---

- Handling dynamic aspects inside an ORDBMS
  - Observing dynamic integrity constraints
  - Implementing active behavior
  - [Manage workflow activities (needs activity diagrams)]
- Using UML Statecharts as adequate specification for dynamic aspects
  - Provide high-level development environment for UDRs
- Using a generic module (instead of generated modules)
  - Allow dynamic changes of the specification
  - Speed up development cycle

# Architecture of the StateMachine Module



1 2 3 9 Accessing product data

4 5 6 Sending events

7 Accessing meta data and internal data

8 Evaluating conditions and executing actions

# The Product Data Schema

---

## Implementation rules:

- each UML class is represented by a ROW TYPE and a typed table
- each UML operation is represented by a user-defined routine (UDR)

## Events are generated

- by SQL-trigger for
  - insert events,
  - update events, and
  - delete eventson tables storing UML objects, or
- by code inside the UDR bodies, i.e.
  - a before event and
  - a after event.

```
CREATE FUNCTION check(...)
RETURNING boolean;
DEFINE ...;
BEGIN
EXECUTE PROCEDURE
  signal(call_event('before',...));
...; --function body
EXECUTE PROCEDURE
  signal(call_event('after',...));
END FUNCTION;
```

```
CREATE ROW TYPE aircraft_type
( name VARCHAR NOT NULL,
  ... );
CREATE TABLE aircraft OF TYPE aircraft_type;
CREATE TRIGGER insert_aircraft
INSERT ON aircraft
REFERENCING NEW AS new
FOR EACH ROW
( EXECUTE PROCEDURE
  signal(insert_event(LIST{new.name,...})));
```

# The StateMachine Module in Detail

