

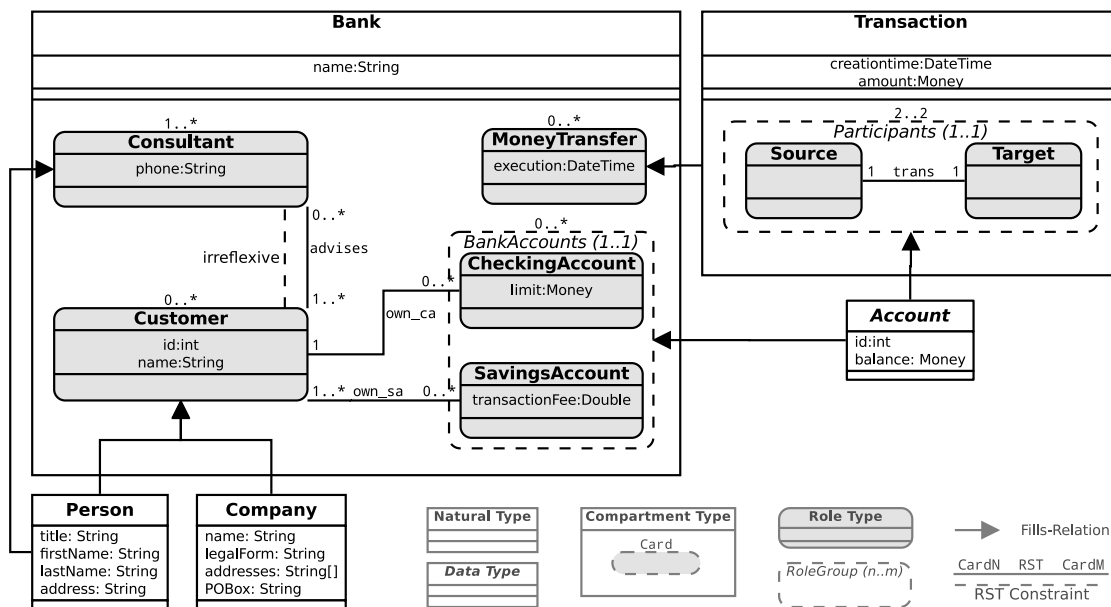
WS2018/19 – Design Patterns and Frameworks

Formal Model of Design Patterns

Professor: Prof. Dr. Uwe Aßmann
 Lectuer: Dr.-Ing. Sebastian Götz
 Tutor: Dr.-Ing. Thomas Kühn

Task 1 Role-based Modeling

This exercise focuses on a *Role-based Modeling Language*, because they are better suited to formalize and compose design patterns than plain UML [2]. In particular, this exercise introduces the *Compartment Role Object Model* (CROM) [1] as a graphical modeling language with a strong formal foundation. To illustrate CROM, the following figure showcases the role model of a banking application, extracted from [1, Fig. 2a].



- Read and understand Section 4 introducing CROM [1, Sect. 4].
- Give an overview of the notation of CROM highlighting all model elements, model relations, and model constraints.
- In preparation of the next task, use CROM *role groups* to formalize the role constraints introduced by Riehle and Gross [2], i.e., *role-implication* (\rightarrow), *role-prohibition* (\vdash), and *role-equivalence* (\leftrightarrow).

Task 2 Role-based Horse Shows

To learn using the CROM language, the next task is to use roles to model *horse shows*.

In the world of horse shows, there are horses, persons, teams, and shows. A team encompasses exactly one *rider* (a person) and one *ridden* horse. Moreover, teams can enter a horse show as a *participant*, where they get a starting number. Furthermore, a horse show enrolls several *referees* (persons) to ensure that each team is examined by at least one referee. In fact, as the average horse show takes about two to three days, both horses and persons needed a place to stay. Consequently, the organization team is additionally tasked to provide *accommodations* for both horses and persons.

- a) Classify the various concepts in the *horse show* domain in accordance to the ontological foundation [1, Sect. 4.1] as either compartment, role, or natural type.
- b) Design a role model for the *horse show* including persons, horses, teams, and shows.
- c) Extend the previous role model to additionally model accommodations for horses and persons, such as hotels, tents, and horse stables.
- d) Finally, revise the role model and include the various constraints declared in the *horse show* domain.

Task 3 Homework for Next Exercise

The homework assigns you to further familiarize yourself with the CROM language [1].

- a) The CROM language not only supports modeling a domain, but also its instances with the *Compartment Role Object Instance* (CROI). Following the role model from Task 2, draw a **valid** instance model of the horse show model.
- b) Afterwards, draw another instance model that **violates all** model constraint of the role-based horse show model.
- c) As a formal modeling language, CROM supports the formal representation of role models. Using this formal notation, specify the role model designed in Task 2 by defining a corresponding CROM model \mathcal{H} and constraint model $\mathcal{C}_{\mathcal{H}}$.

References

- [1] Thomas Kühn, Böhme Stephan, Sebastian Götz, and Uwe Aßmann. A combined formal model for relational context-dependent roles. In *Proceedings of the 2015 ACM SIGPLAN International Conference on Software Language Engineering*, pages 113–124. ACM, 2015. doi: 10.1145/2814251.2814255. URL <http://dl.acm.org/citation.cfm?id=2814255>.
- [2] Dirk Riehle and Thomas Gross. Role model based framework design and integration. In *ACM SIGPLAN Notices*, volume 33, pages 117–133. ACM, 1998.