

# Subset, Equality, and Exclusion Rules in ORM

(Chapter 6)

**Mustafa Jarrar**

**Birzeit University**  
[mjarrar@birzeit.edu](mailto:mjarrar@birzeit.edu)  
[www.jarrar.info](http://www.jarrar.info)



# Watch this lecture and download the slides



Download: <http://www.jarrar.info/courses/ORM/Jarrar.LectureNotes.SubsetEqualityExclusionRules.pdf>

Online Courses: <http://www.jarrar.info/courses/>

Some diagrams in this lecture are based on [1]

**Keywords:** Subset, Equality, Exclusion, set-comparison constraints, Set Constraints, Rules, Business Rules, Business logic derivation rules, integrity constraints

# Conceptual Schema Design Steps

1. From examples to elementary facts



2. Draw fact types and apply population check



3. Combine entity types



4. Add uniqueness constraints



5. Add mandatory constraints



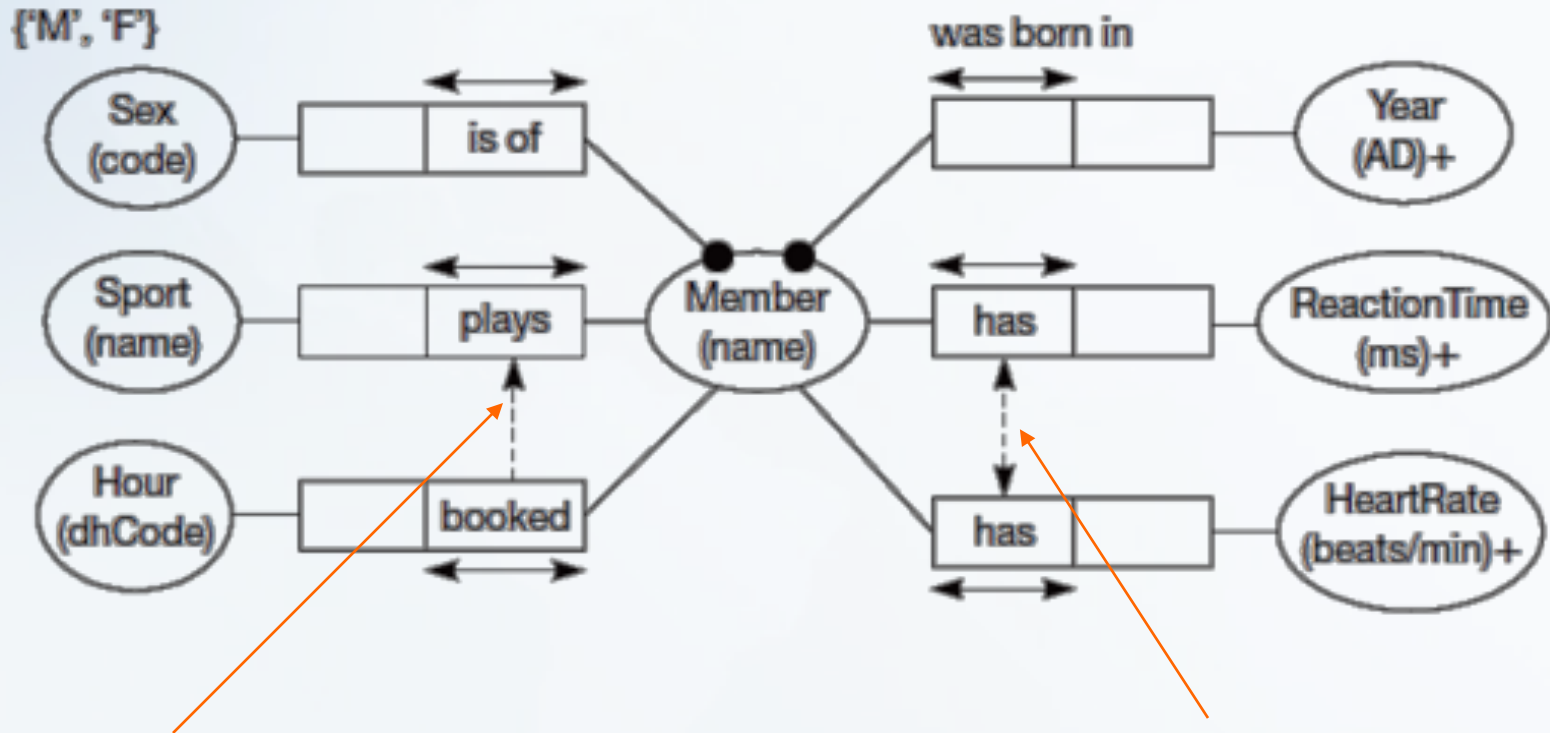
6. Add subtype relations and other constraints



7. Final checks, & schema engineering issues



# Role subset/equality constraint



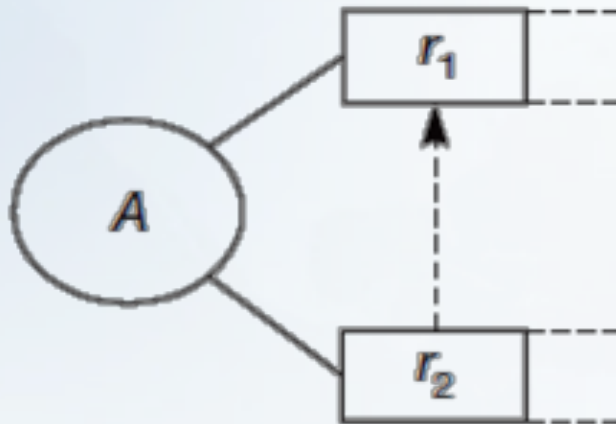
## Subset constraint:

Every Member booked an Hour should play sport.

## Equality constraint:

Every Member 'has' ReactionTime should 'has' HeartRate, and every Member 'has' HeartRate should 'has' ReactionTime.

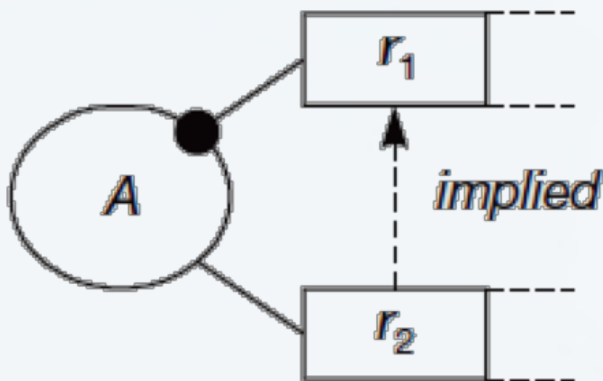
# Role subset constraint



For each database state:

$$\text{pop}(r_2) \subseteq \text{pop}(r_1)$$

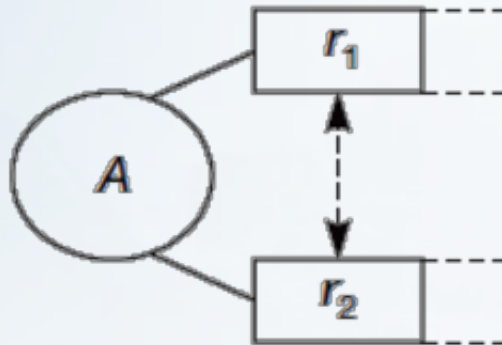
if a plays  $r_2$  then a plays  $r_1$



**Notice that this subset constraint is implied, and should be removed.**

That is, there is no need to say that every  $A$  playing  $r_2$  must also play  $r_1$  (subset), because the mandatory constraint here means that every  $A$  must play  $r_1$  (the Mandatory implies the subset).

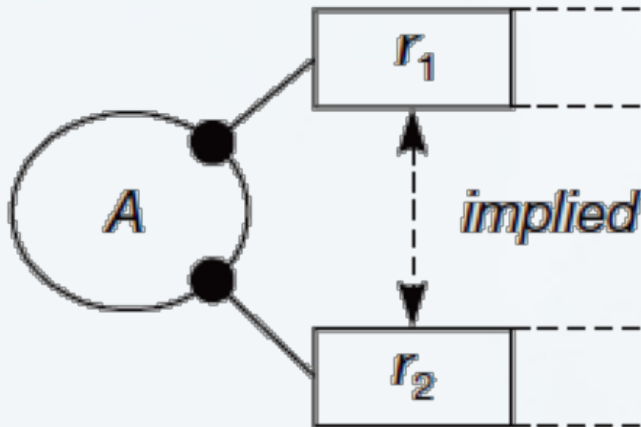
# Role equality constraint



For each database state:

$$\text{pop}(r_1) = \text{pop}(r_2)$$

*a* plays  $r_1$  if *a* plays  $r_2$



**Also this quality constraint is implied, and should be removed.**

# Implication

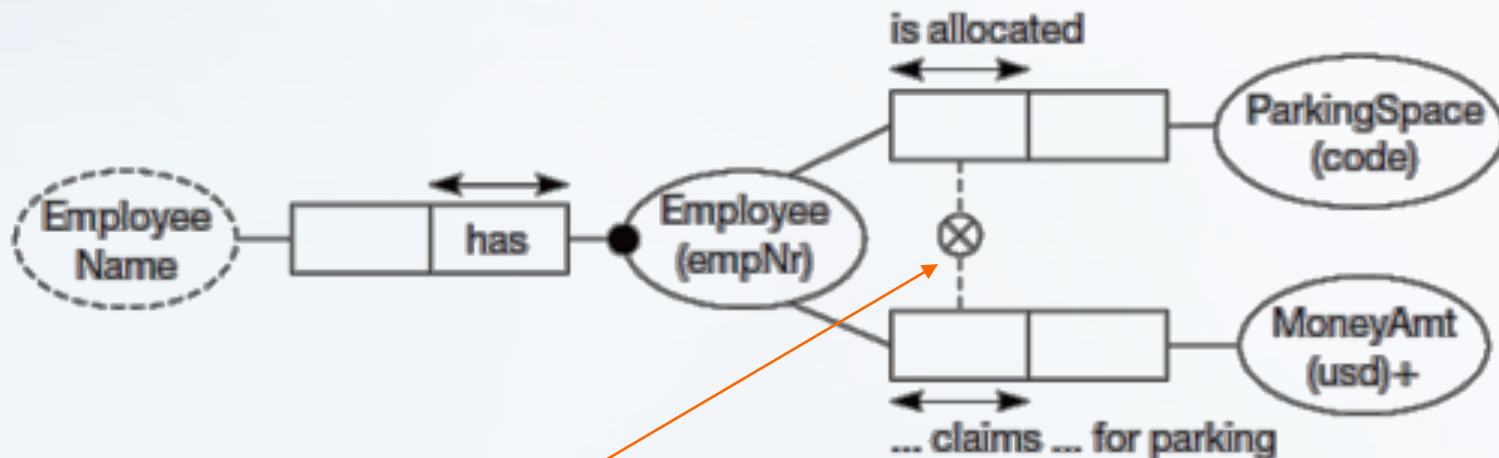
Who can explain the difference?



The two constraints in the first model says: each  $A$  must play  $r_1$  or  $r_2$  (or both), and that if  $A$  plays  $r_2$  then it must play  $r_1$ . This means that  $r_1$  must be always played (which is the second model)

# Role Exclusion Constraint

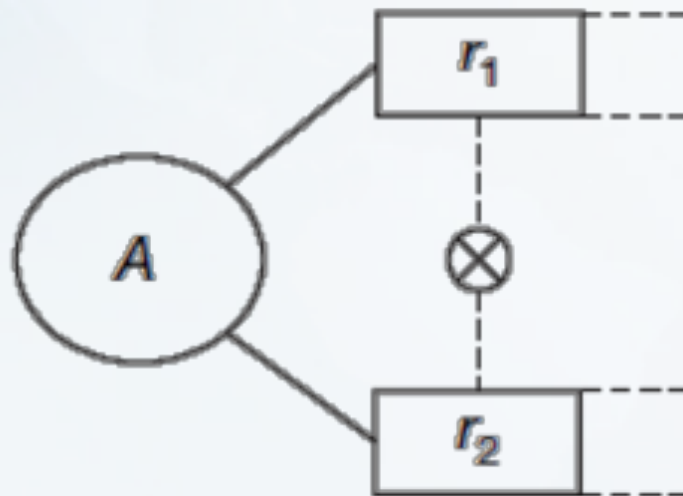
<i>EmpNr</i>	<i>Employee name</i>	<i>Parking space</i>	<i>Parking claim (\$)</i>
001	Adams B	C01	–
002	Bloggs F	–	200
003	Collins T	B05	–
004	Dancer F	–	250
005	Eisai Z	?	?



**Exclusion constraint:**  
**Every Employee is allocated a ParkingSpace should not claim MoneyAmt.**



# Role Exclusion Constraint

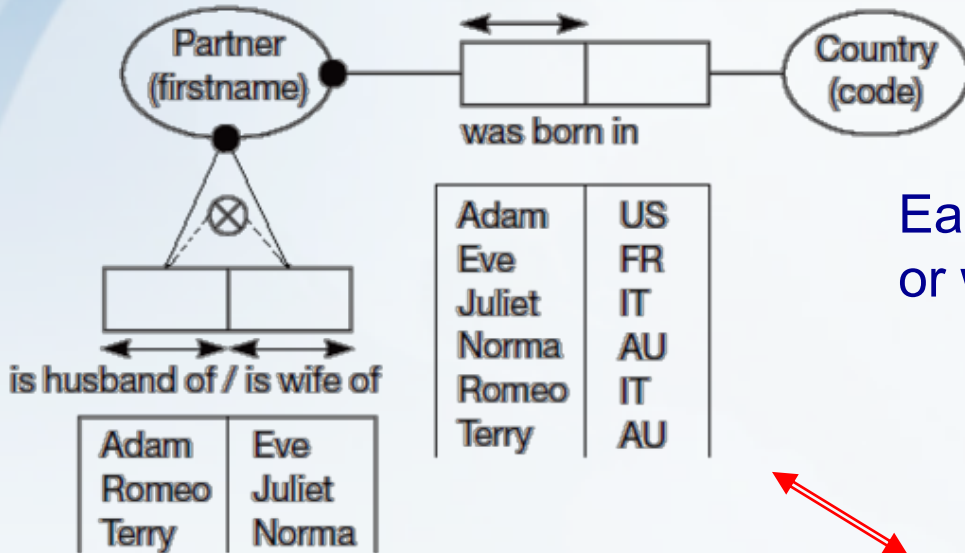


For each database state:

$$\text{pop}(r_1) \cap \text{pop}(r_2) = \{ \}$$

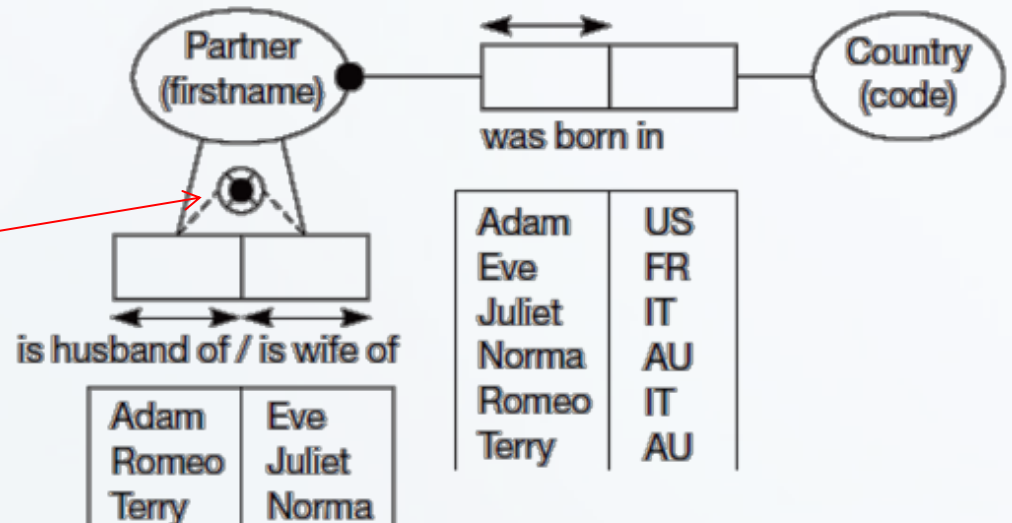
no  $a$  plays both  $r_1$  and  $r_2$

# Role Exclusion Constraint

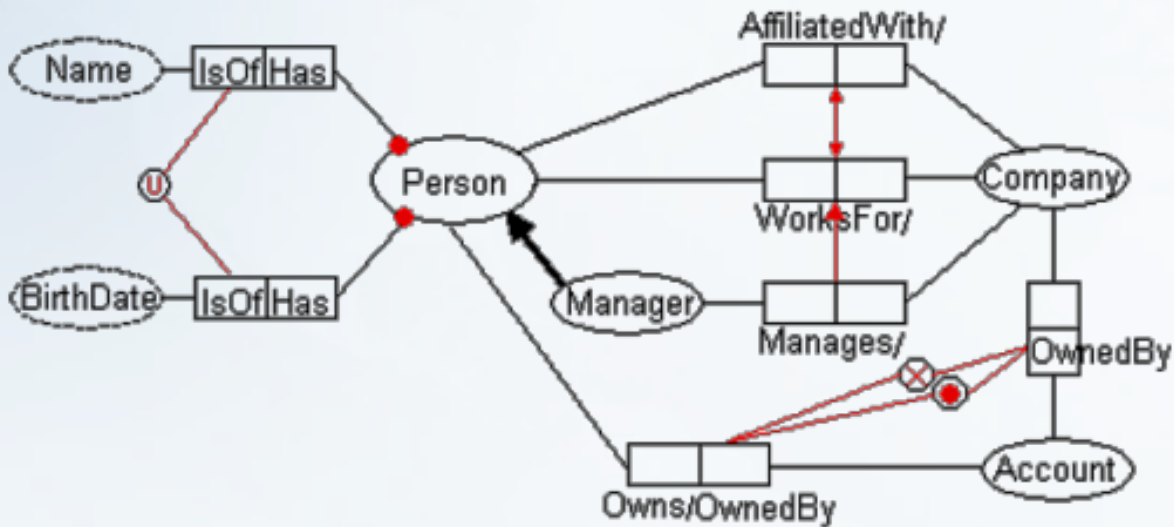


Each partner *must* be either a husband or wife (but not both at the same time).

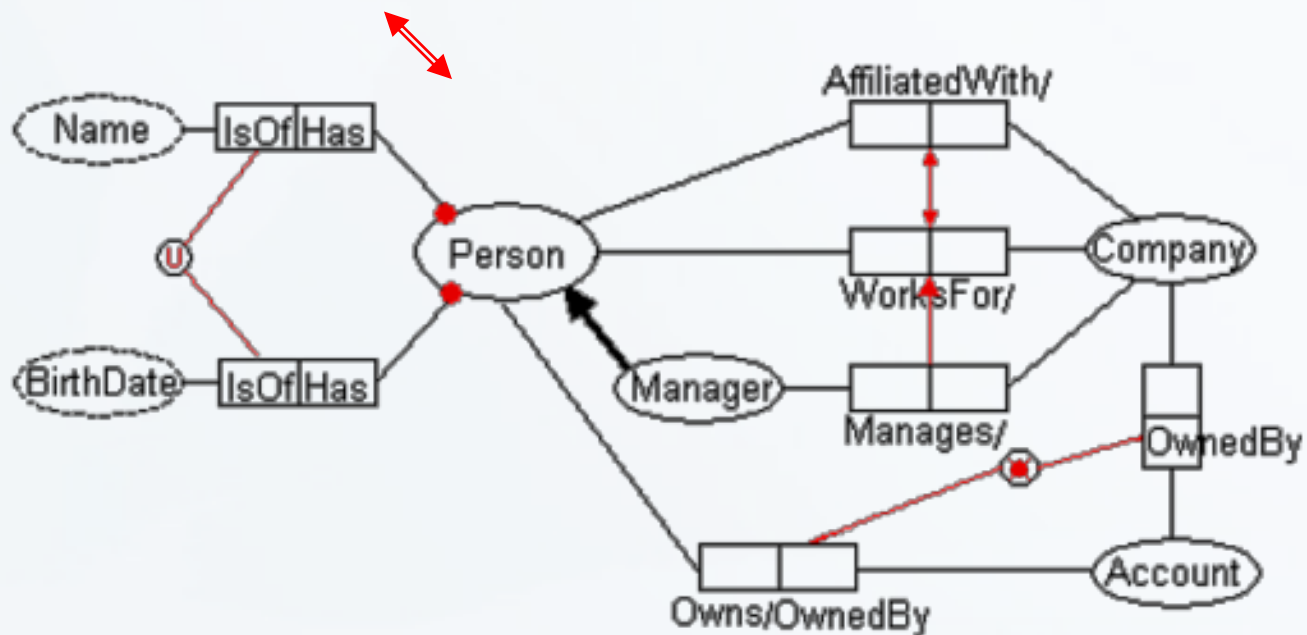
Called "*Exclusive-or*"



# Exclusive-or (another example)

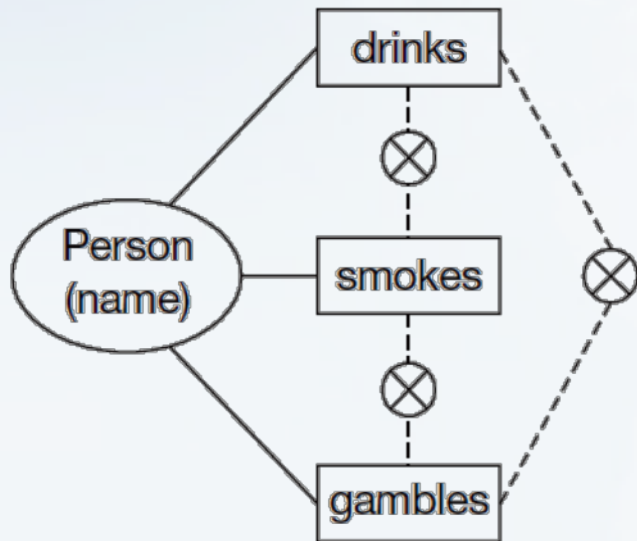


Each Account must be OwnedBy a Person or a Company, but not both.

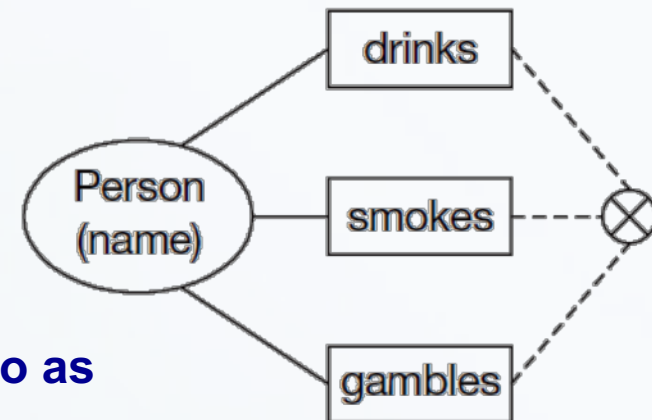


# Role Exclusion Constraint

Each person has at most one of three vices.  
i.e., from 0 to 3 vices.



It can be written also as

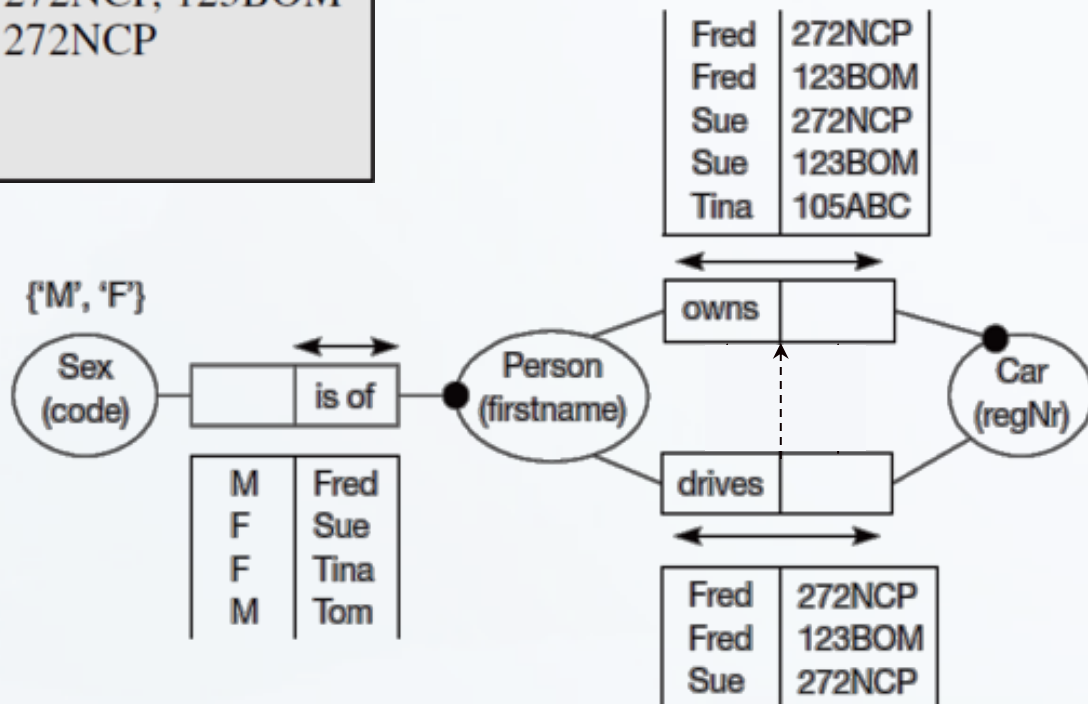


# Pair Exclusion Constraint

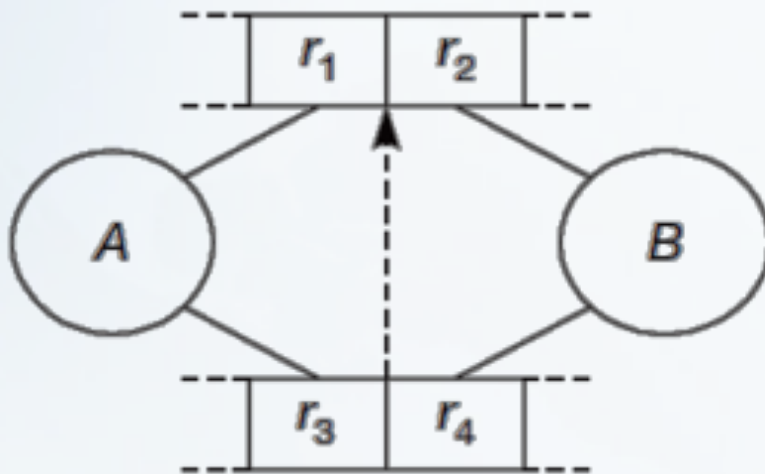


Person	Sex	Cars owned	Cars driven
Fred	M	272NCP, 123BOM	272NCP, 123BOM
Sue	F	272NCP, 123BOM	272NCP
Tina	F	105ABC	
Tom	M		

How can we restrict that a person can drive a car only if he owns *that* car.



# Pair-subset constraint

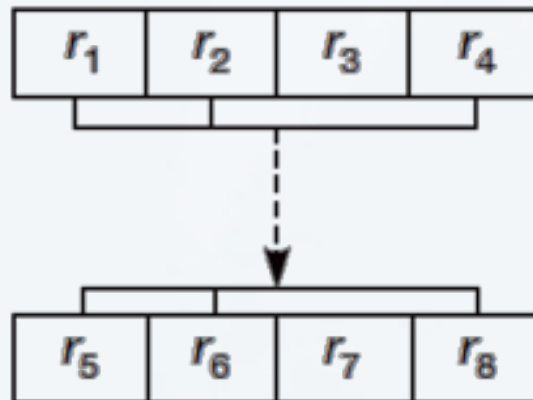


For each database state:

$$\text{pop}(r_3, r_4) \subseteq \text{pop}(r_1, r_2)$$

**each  $ab$  pair in  $\text{pop}(r_3, r_4)$  is also in  $\text{pop}(r_1, r_2)$**

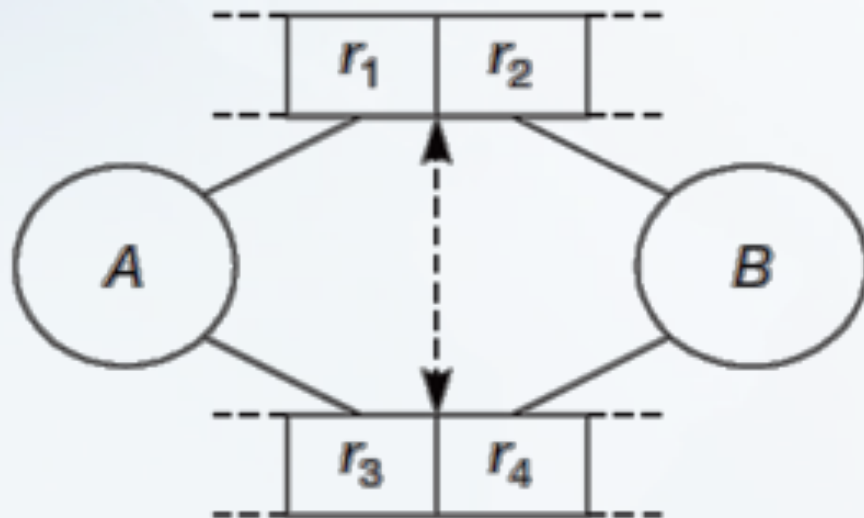
An example of a tuple-subset constraint between sequences of three roles.



For each database state:

$$\text{pop}(r_1, r_2, r_4) \subseteq \text{pop}(r_5, r_6, r_8)$$

# Equality Constraint

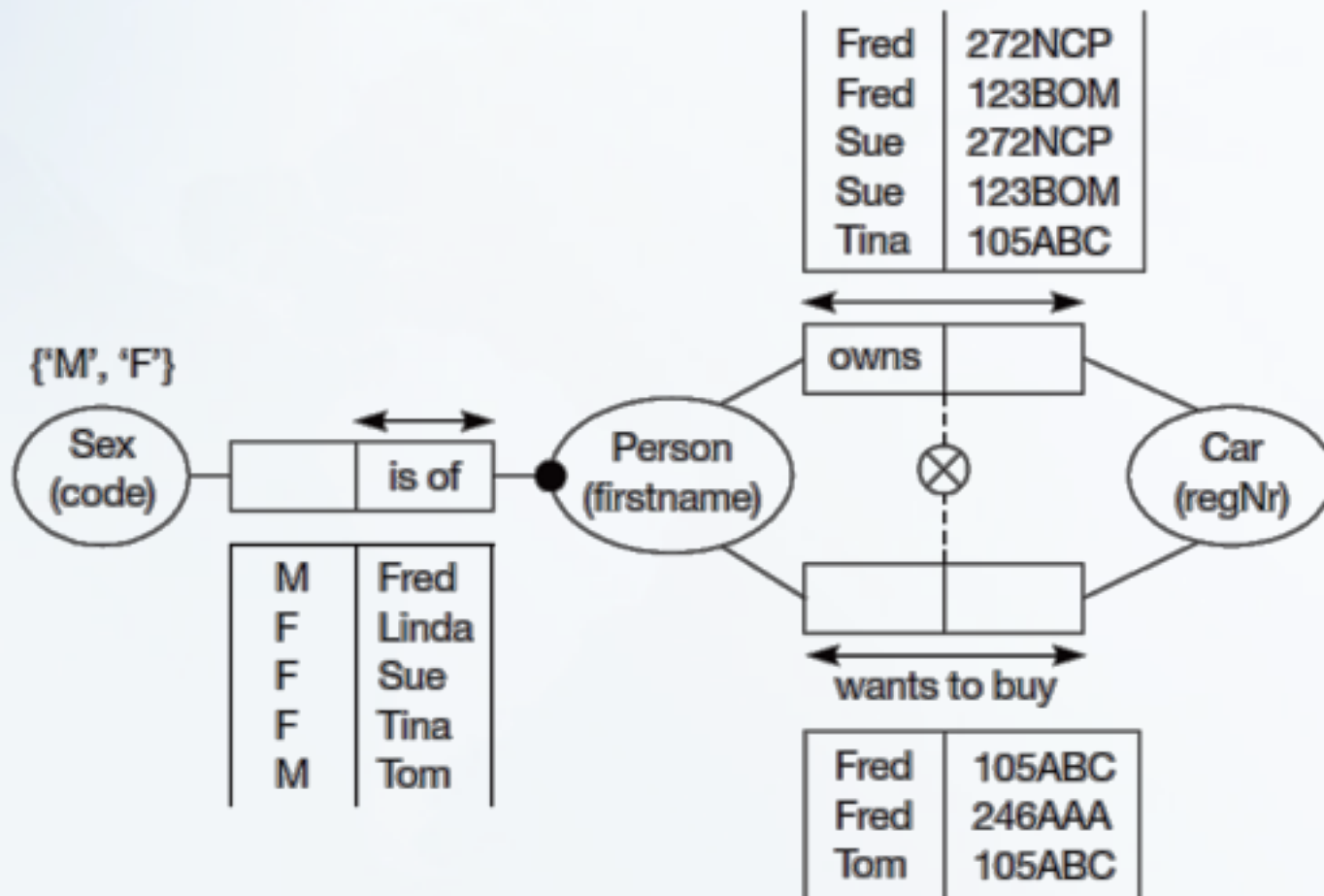


For each database state:

$$\text{pop}(r_3, r_4) = \text{pop}(r_1, r_2)$$

**each  $ab$  pair in  $\text{pop}(r_3, r_4)$   
is also in  $\text{pop}(r_1, r_2)$   
and vice versa**

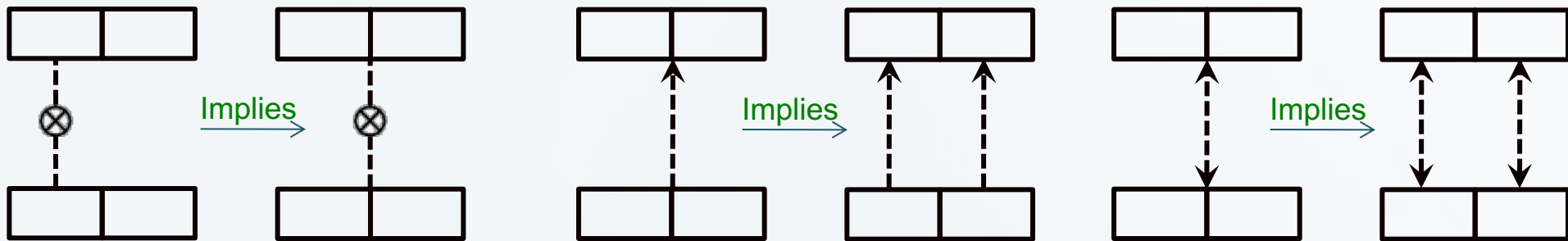
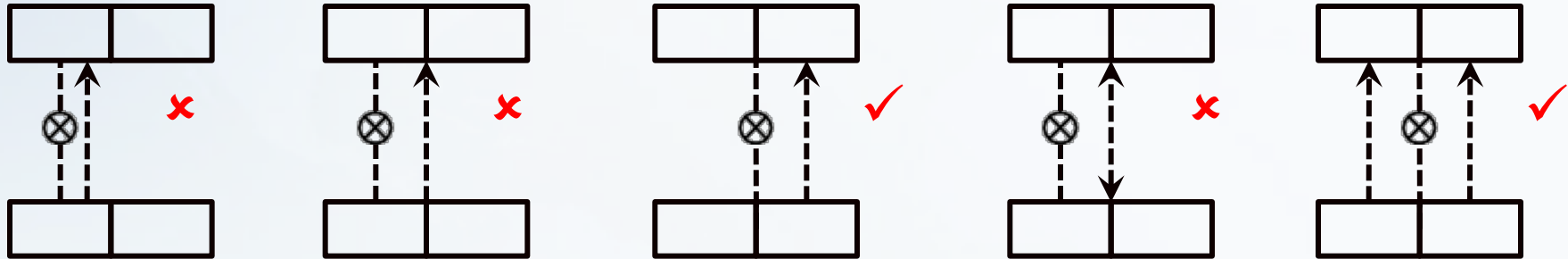
# Pair Exclusion Constraint



Same person cannot 'own' and 'wants to buy' the same car?



# What is Wrong?



# References

- [1] Terry Halpin, Tony Morgan: Information Modeling and Relational Databases, Second Edition. Second Edition. The Morgan Kaufmann Series in Data Management Systems. ISBN: 0123735688
- [2] Mustafa Jarrar and Robert Meersman: Ontology Engineering -The DOGMA Approach. Book Chapter in "Advances in Web Semantics I". Chapter 3. Pages 7-34. LNCS 4891, Springer.ISBN:978-3540897835. (2008).
- [3] Mustafa Jarrar, Anton Deik, Bilal Faraj: Ontology-Based Data And Process Governance Framework -The Case Of E-Government Interoperability In Palestine . In pre-proceedings of the IFIP International Symposium on Data-Driven Process Discovery and Analysis (SIMPDA'11). Pages(83-98). ISBN 978-88-903120-2-1. Campione, Italy. June 30, 2011.
- [4] Mustafa Jarrar: Mapping ORM Into The SHOIN/OWL Description Logic- Towards A Methodological And Expressive Graphical Notation For Ontology Engineering . In OTM 2007 workshops: Proceedings of the International Workshop on Object-Role Modeling (ORM'07). Pages (729-741), LNCS 4805, Springer. ISBN: 9783540768890. Portugal. November, 2007
- [5] Mustafa Jarrar: Towards Automated Reasoning On ORM Schemes. -Mapping ORM Into The DLR\_idf Description Logic. In proceedings of the 26th International Conference on Conceptual Modeling (ER 2007). Pages (181-197). LNCS 4801, Springer. Auckland, New Zealand. ISBN 9783540755623. November 2007
- [6] Mustafa Jarrar and Stijn Heymans: Unsatisfiability Reasoning In ORM Conceptual Schemes. In Current Trends in Database Technology - EDBT 2006: Proceeding of the IFIP-2.6 International Conference on Semantics of a Networked. Pages (517-534). LNCS 4254, Springer. Munich, Germany. ISBN: 3540467882. March 2006.
- [7] Mustafa Jarrar and Stijn Heymans: [Towards Pattern-Based Reasoning For Friendly Ontology Debugging](#) . Journal of Artificial Intelligence Tools. Volume 17. No.4. World Scientific Publishing. August 2008.
- [8] Mustafa Jarrar, Maria Keet, and Paolo Dongilli: Multilingual Verbalization Of ORM Conceptual Models And Axiomatized Ontologies. Technical report. STARLab, Vrije Universiteit Brussel, February 2006.
- [9] Sergey Lukichev and Mustafa Jarrar: Graphical Notations For Rule Modeling . Book chapter in "Handbook of Research on Emerging Rule-Based Languages and Technologies". IGI Global. ISBN:1-60566-402-2. (2009)
- [10] Mustafa Jarrar: Modularization And Automatic Composition Of Object-Role Modeling (ORM) Schemes .OTM 2005 Workshops: Proceedings of the Object-Role Modeling (ORM'05). Pages (613-625). LNCS 3762, Springer. ISBN: 3540297391. 2005.
- [11] Mustafa Jarrar: Towards Methodological Principles For Ontology Engineering. PhD Thesis. Vrije Universiteit Brussel. (May 2005)
- [12] Mustafa Jarrar, Jan Demey, and Robert Meersman: On Using Conceptual Data Modeling For Ontology Engineering . Journal on Data Semantics, Special issue on "Best papers from the ER/ODBASE/COOPIS 2002 Conferences". LNCS 2800. No 1. Springer. 2003.
- [13] Jan Demey, Mustafa Jarrar, and Robert Meersman: A Markup Language For ORM Business Rules . Proceedings of the International Workshop on Rule Markup Languages for Business Rules on the Semantic Web (RuleML 2002). Pages(107-128). Volume 60. CEUR Workshop Proceedings. ISSN 1613-0073. June 2002
- [14] Mustafa Jarrar: Towards Effectiveness And Transparency In E-Business Transactions, An Ontology For Customer Complaint Management . A book chapter in "Semantic Web Methodologies for E-Business Applications". chapter 7. IGI Global. (2008)
- [15] Mustafa Jarrar: ORM Markup Language, Version 3 . Technical Report. STAR Lab, Vrije Universiteit Brussel, Belgium. January 2007