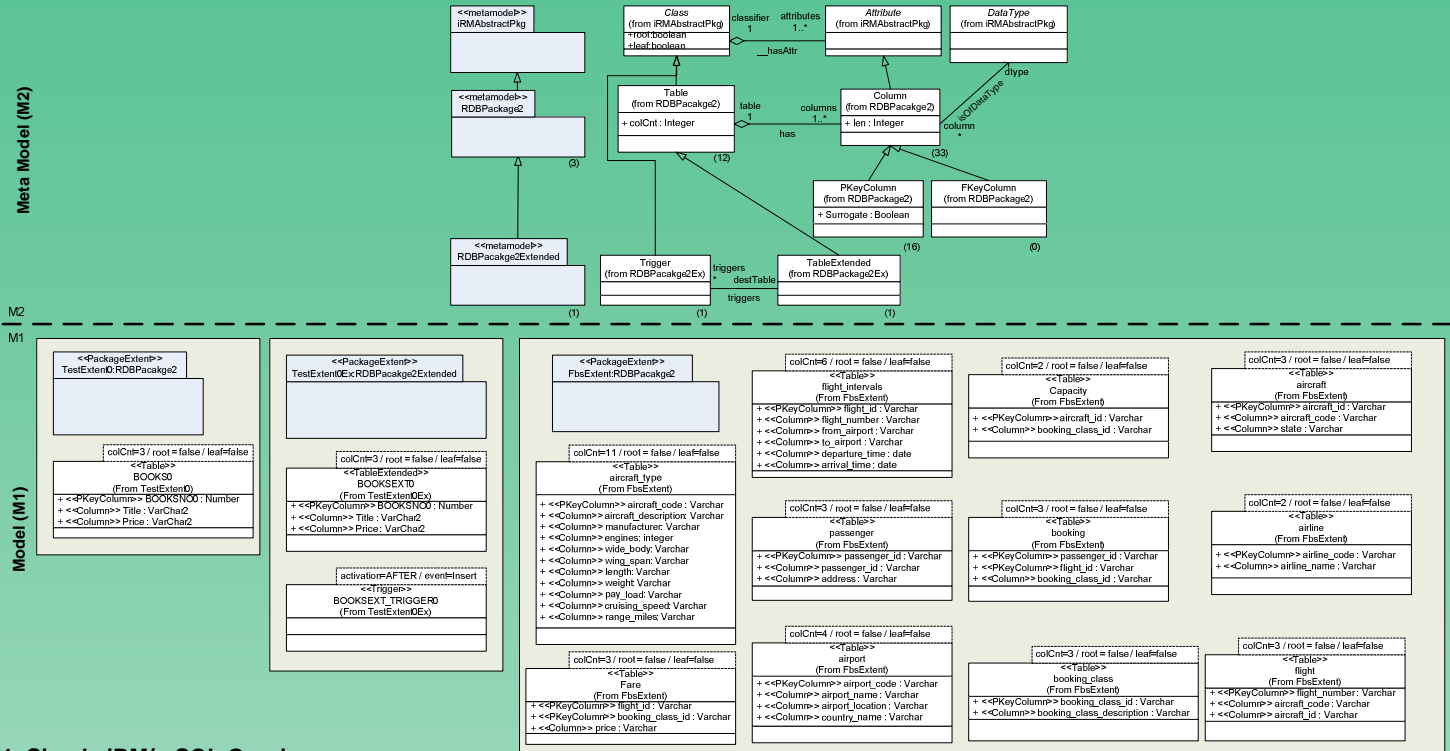


iRM: A MOF-based Repository System with Querying Capabilities

Ilia Petrov, Marc Holze, Gabor Nemes, Markus Schneider, Stefan Jablonski

ilia.petrov@informatik.uni-erlangen.de



1. Simple iRM/mSQL Queries

- ❑ Select a normal attribute
 1. `SELECT t.name, t.colCnt FROM M2::RDBPackage2:Table t;`
- ❑ Select pseudo attributes
 2. `SELECT t.name, t.level, t.mofid FROM M2::RDBPackage2:Table t;`
 3. `SELECT t.name, t.level, t.mofid FROM M2::RDBPackage2Extended:TableExtended t;`
- ❑ Select instances of a certain class, ignoring its sub-classes – Only()
 4. `SELECT t.name, t.level FROM ONLY(M2::RDBPackage2:Table) t;`
- ❑ Sub Classes and Super classes. Different variable type – class variable
 5. `SELECT t.name, t.level FROM SUBM(M2::RDBPackage2:Table) t;`
 6. `SELECT t.name, t.level, t.mofid FROM SUPERM(M2::RDBPackage2Extended:TableExtended) t;`

3. Uniform treatment of data and metadata in iRM / mSQL

21. `SELECT c.name, ai.name, dt.name FROM M2::RDBPackage2->c, c->ai^dt;`
22. `SELECT c.name, ci.name, ai.name, dt.name FROM M2::RDBPackage2->c ci, ci->ai^dt;`

4. iRM / mSQL Sub-Queries

- ❑ Not correlated iRM/mSQL sub-queries
 23. `SELECT DISTINCT a.name FROM M2::RDBPackage2:Table->a^da, M2::RDBPackage2->c WHERE da IN (SELECT * FROM M2::RDBPackage2:Table->a1^dt);`
- ❑ Correlated iRM/mSQL sub-queries
 24. `SELECT DISTINCT a.name FROM M2::RDBPackage2:Table->a^da, M2::RDBPackage2->c WHERE da IN (SELECT * FROM c->a1^dt);`
- ❑ iRM/mSQL sub-queries in the from clause
 25. `SELECT c.name FROM (SELECT * FROM M2::RDBPackage2:Table->t WHERE TYPEM(t) != M3::Model:Reference) c WHERE TYPEM(c) IN SUB(M3::Model:ModelElement);`

2. iRM/mSQL Variable Declaration

- ❑ Select all M2 Packages
 7. `SELECT p.name FROM M2::>p;`
- ❑ Select all M2 Packages and their M1 instances
 8. `SELECT pi.name, pi.level, p.name, p.level FROM M2::>p pi;`
- ❑ Eliminate the duplicate entries
 9. `SELECT DISTINCT pi.name FROM M2::>p pi;`
- ❑ Select all classes defined in a package
 10. `SELECT p.name, p.level, c.name, c.level FROM M2::>p, p->c;`
- ❑ Get also the instances of these classes
 11. `SELECT p.name, c.name, ci.name, ci.level FROM M2::>p, p->c, c ci;`
- ❑ Retrieve the attributes of all classes in M2::RDBPackage2 and of the class M2::RDBPackage2:Table. How are attributes handled?
 12. `SELECT c.name, a.name FROM M2::RDBPackage2->c, c->a;`
 13. `SELECT a.name FROM M2::RDBPackage2:Table->a;`
- ❑ Select the type of an attribute
 14. `SELECT Y attr_t.name FROM TYPEM(M1::TestExtent0:BOOKS0:BOOKNO0) attr_t;`
- 15. `SELECT a.name, a.level, ta.name, ta.level, ta.mofid FROM M1::TestExtent0:BOOKS0->a, TYPEM(a) ta;`
- 16. `SELECT a.name, ta.name AS TypeA, tta.name AS TypeTypeA FROM M1::TestExtent0:BOOKS0->a, TYPEM(a) ta, TYPEM(ta) tta;`
- 17. `SELECT a.name, tta.name AS TypeOf_TypeOf_A FROM M1::TestExtent0:BOOKS0->a, TYPEM(TYPEM(a)) tta;`
- ❑ Get the data type of an attribute of a class
 18. `SELECT a.name, dt.name FROM M2::RDBPackage2:Table->a^dt;`
- ❑ Compatibility of attributes' data types - Similar
 19. `SELECT t.name, x.name, x.level FROM M2::RDBPackage2:Table t, TYPEM(t) x WHERE t.name SIMILAR x.mofid;`
- ❑ Aggregate Functions
 20. `SELECT COUNT(tbl.colCnt), SUM(tbl.colCnt) FROM M2::RDBPackage2Extended:TableExtended tbl WHERE tbl.colCnt + LEVEL(tbl) <= 7;`