Database Design and Creation

for The Journal of Computing Professionals

By:

Evan Bonsignori

Design Procedure

- Derive semantic rules from provided journal issues
- Derive FDs and MVDs from semantic rules
- Create initial FD diagram (universal relation)
- Normalize universal relation to 1NF, 2NF, 3NF, BCNF, and then 4NF
- Create MySQL schema using relations found from 4NF
- Populate MySQL database using data from provided journal issues
- Run a series of test queries to test integrity of database design
- Create views for user groups to access appropriate data in database

Scientific Article Excerpt

Prediction Capability of Neural Networks Trained in Monte-Carlo Paradigm

Ray Hashemi, Ph.D

Department of Computer and Information Science
University of Arkansas at Little Rock
Little Rock, AR. 72204
Affiliation: ACM (SIGAPP)

ABSTRACT

The Monte-Carlo training paradigm for Artificial Neural Networks has been studied, the training short cut to reduce the training time has been discussed, and the prediction capability of such trained neural network has been compared by prediction capability of the statistical approach of the Discriminant Analysis. The Artificial Neural Network trained in Monte-Carlo method proves itself as a reliable prediction tool which is superior to Discriminant Analysis.

Keywords: Prediction Power, Monte-Carlo paradim, Machine Learning, Intillegent Systems, and Neural Networks

INTRODUCTION

A neural network is composed of an input layer, one or more than one hidden layers, and an output layer. Each layer is composed of a set of nodes

REFERENCES

1. Hertz J., Krogh A., Palmer R. G., "Introduction to the Theory of Neural Computation", Addison-Wesley, Redwood City, Ca., 1991, pp. 145-156.

2. Barinaga, M. "Neuroscience Models the Brain", Science 247, February 2, 1990, pp 524-527.

3. Hashemi R, Jelovsek F.R., Razzaghi M., "Developmental Toxicity Risk Assessment: A Rough Sets Approach", The International Journal of Methods of Information in Medicine (in press).

4. Wasserman P. D., "Neural Computing", ANZA Research, Inc., Van Nostrand Reinhold, New York, 1989, pp. 77-87.

5. Hinton, G., and Sejnowski, T. "Learning and relearning in Boltzmann machines", Parallel Distributed Processing, Vol. 1, Cambridge, MA, MIT Press, 1986, pp. 282-317.

6. Kirkpatrick S., Gelatt C. Jr., and Vecchi M., "Optimization by simulated annealing", Journal of Science 220, 1983, pp. 671-680.

7. Hashemi R., Razzaghi M., Jelovsek F., and Talburt R., "Conflict Resolution in Learning Through Examples", Proceedings of the 1992 ACM/IEE International Symposium on Applied Computing, Kansas City, Missouri, March, 1992, pp. 509 - 602.

Semantic Rules

1. Each journal has only one title, and publication frequency.
2. Each journal has multiple issues.
3. Each issue has only one volume number, issue number, month, day, and year.
4. Each issue has multiple scientific articles, call for papers, events, jobs, and interns.
5. Each conference has only one title, start day, end day, start month, end month, start year, end year, location name, country, state, and city.
6. Each conference has multiple chairs and conference organizations.
7. Each conference organization has only one name, relationship, country, state, city, street, and zip code.
8. Each chair has only one name, type, and organization name.
9. Each call for papers has only one conference, secretariat, and call for papers body.
10. Each call for papers has multiple topics and deadlines.
11. Each deadline has only one deadline label, year, month, and day.
12. Each secretariat has only one title, email, fax, phone number, address heading, country, state, city, street, and postal code.
13. Each scientific article has only one title, paper body, and page number.
14. Each scientific article has multiple authors, topics, keywords, and references.
15. Each reference has only one title, type, and reference body.
16. Each reference has multiple authors.
17. Each event has only one start year, start month, start day, end year, end month, end day, title, event country, event location body, event body, contact title, and contact body.
18. Each event has multiple sponsoring organizations.
19. Each job has only one position, job country, job state, job city, number of references required, organization, application type, job status, job body, contact title, contact body, and salary.
20. Each job has multiple job qualifications and job deadlines.
21. Each job qualification has only one qualification label and qualification value.
22. Each job deadline has only one deadline label, year, month, and day.
23. Each intern has only one name, graduating year, country from, state from, city from, degree type, degree title, and intern description body.

Conversions into Dependencies

13. Each scientific article has only one title, paper body, and page number.
14. Each scientific article has multiple authors, topics, keywords, and references.

Semantic Rule 13

Each scientific article has only one title, paper body, and paper number.

1. Entity(s): scientific article
2. Attributes:
   a. title = ARTTITLE
   b. paper body = ARTBODY
   c. paper number = ARTPAGE#
3. Entity Identifier:
   a. scientific article = ART#, artificial identifier created because no provided attributes sufficiently identify the entity scientific article.
4. FD: ART# -> ARTTITLE, ARTBODY, ARTPAGE#

Semantic Rule 14

Each scientific article has multiple authors, topics, keywords, and references.

1. Entity(s): scientific article, reference
2. Attributes:
   a. author = ARTAUTH
   b. topics = ARTTOPIC
   c. keyword = ARTKEY
3. Entity Identifier:
   a. scientific article: ART#, chosen in semantic rule 13.
   b. reference: REF#, chosen in semantic rule 15.
4. MVD: ART# ->-> ARTAUTH, ARTTOPIC, ARTKEY, REF#

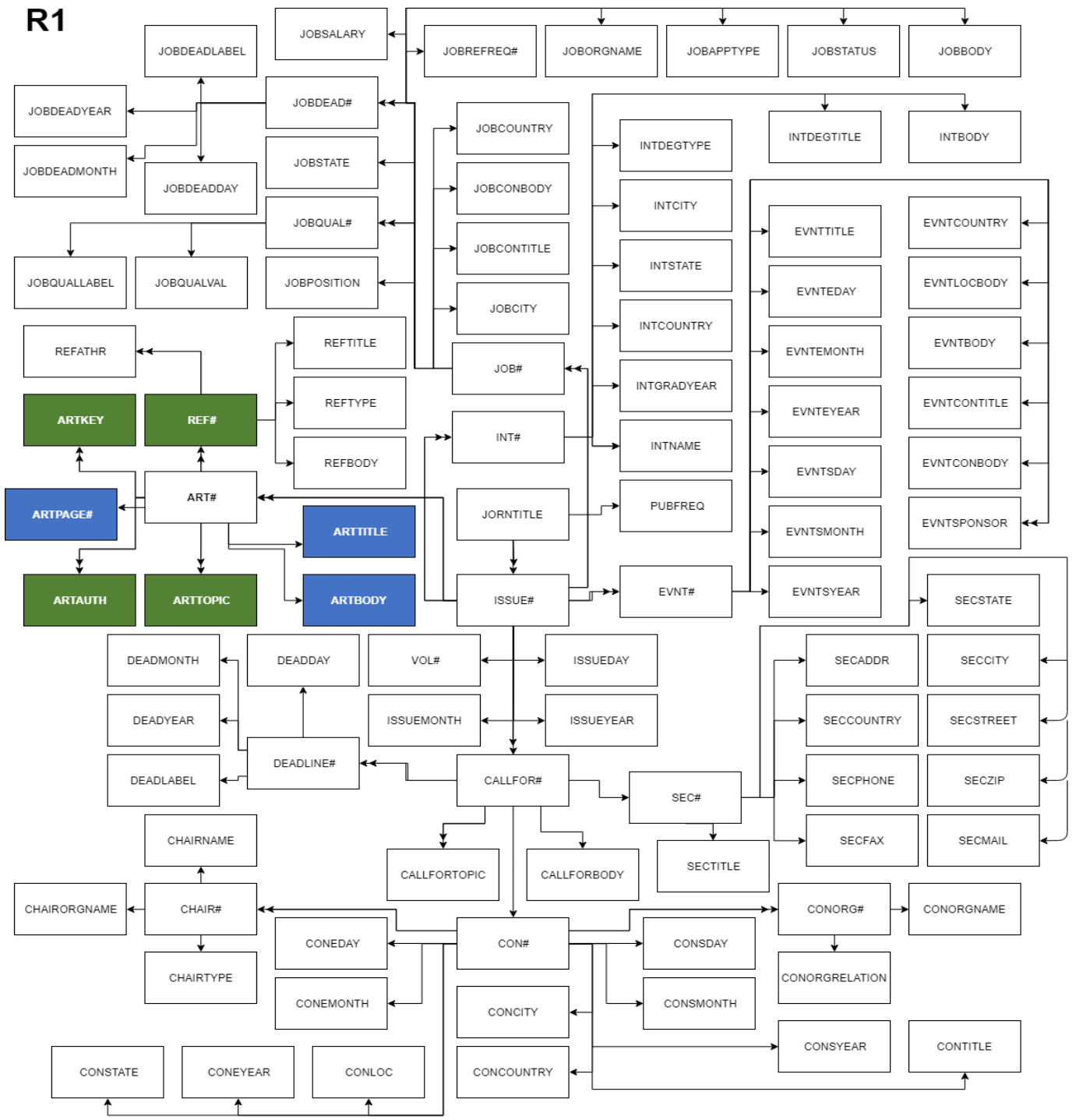
List of Dependencies

1. JORNTITLE -> PUBFREQ (SR1)
2. JORNTITLE ->-> ISSUE# (SR2)
3. ISSUE # -> VOL#, ISSUEMONTH, ISSUEDAY, ISSUEYEAR (SR3)
4. ISSUE# ->-> ART#, CALLFOR#, EVNT#, JOB#, INT# (SR4)
5. CON# -> CONTITLE, CONSDAY, CONEDAY, CONSMONTH, CONEMONTH, CONSYEAR, CONEYEAR, CONLOC, CONCOUNTRY, CONSTATE, CONCITY (SR5)
6. CON# ->-> CHAIR#, CONORG# (SR6)
7. CONORG# -> CONORGNAME, CONORGRELATION (SR7)
8. CHAIR# -> CHAIRNAME, CHAIRTYPE, CHAIRORGNAME (SR8)
9. CALLFOR# -> CON#, SEC#, CALLFORBODY (SR9)
10. CALLFOR# ->-> CALLFORTOPIC, DEADLINE# (SR10)
11. DEADLINE# -> DEADLABEL, DEADYEAR, DEADMONTH, DEADDAY (SR11)
12. SEC# -> SECTITLE, SECEMAIL, SECFAX, SECPHONE, SECADDR, SECCOUNTRY, SECSTATE, SECCITY, SECSTREET, SECZIP (SR12)
13. ART# -> ARTTITLE, ARTBODY, ARTPAGE# (SR13)
14. ART# ->-> ARTAUTH, ARTTOPIC, ARTKEY, REF# (SR14)
15. REF# -> REFTITLE, REFTYPE, REFBODY (SR15)
16. REF# ->-> REFATHR (SR16)
17. EVNT# -> EVNTSYEAR, EVNTSMONTH, EVNTSDAY, EVNTEYEAR, EVNTEMONTH, EVNTEDAY, EVNTTITLE, EVNTCOUNTRY, EVNTLOCBODY, EVNTBODY, EVNTCONTITLE, EVNTCONBODY (SR17)
18. EVNT# ->-> EVNTSPONSOR (SR18)
19. JOB# -> JOBPOSITION, JOBCOUNTRY, JOBSTATE, JOBCITY, JOBREFREQ#, JOBORGNAME, JOBAPPTYPE, JOBSTATUS, JOBBODY, JOBCONTITLE, JOBCONBODY, JOBSALARY (SR19)
20. JOB# ->-> JOBQUAL#, JOBDEAD# (SR20)
21. JOBQUAL# -> JOBQUALLABEL, JOBQUALVAL (SR21)
22. JOBDEAD# -> JOBDEADLABEL, JOBDEADYEAR, JOBDEADMONTH, JOBDEADDAY (SR22)
23. INT# -> INTNAME, INTGRADYEAR, INTCOUNTRY, INTSTATE, INTCITY, INTDEGTYPE, INTDEGTITLE, INTBODY (SR23)

Universal Relation

R1

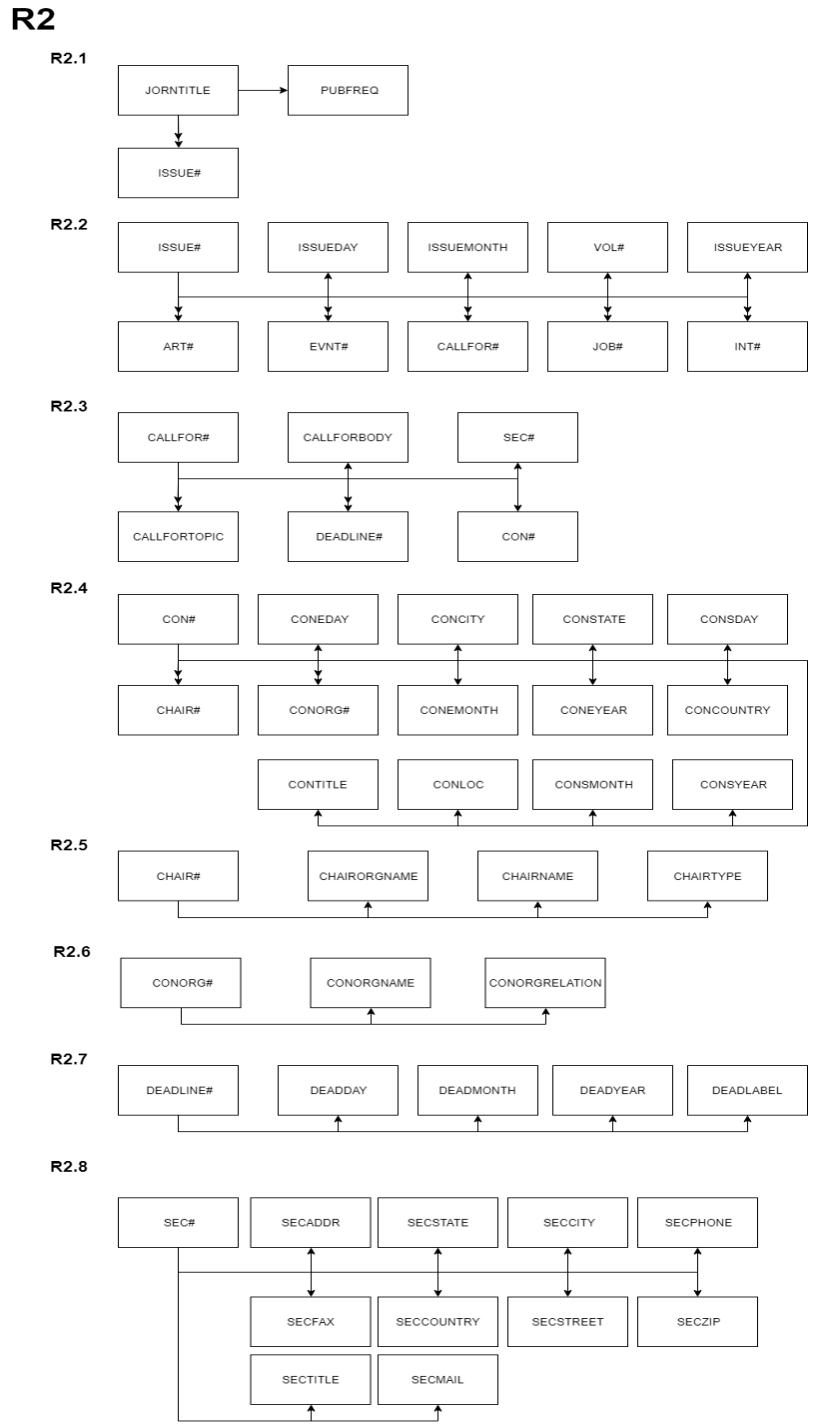
- Primary key is JORNTITLE
- No Reduction Required
- In First Normal Form (1NF)
- In Second Normal Form (2NF)



3NF and BCNF

R2

sub-relations 1-8

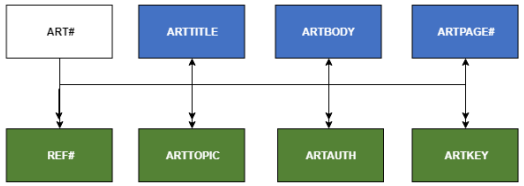


3NF and BCNF

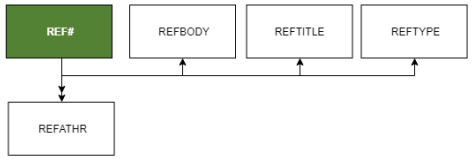
R2

sub-relations 9-15

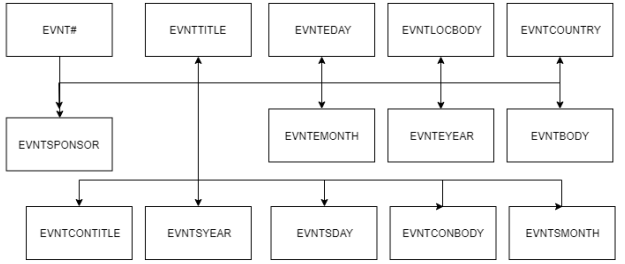
R2.9



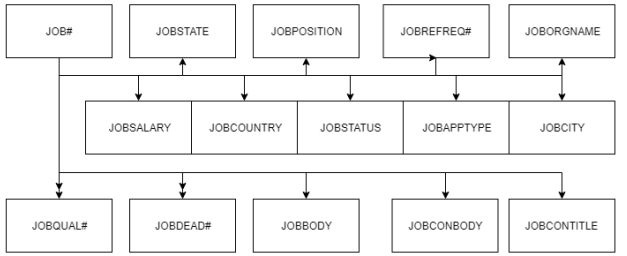
R2.10



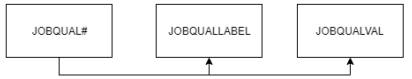
R2.11



R2.12

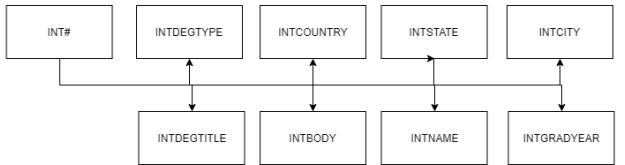


R2.13



R2.14

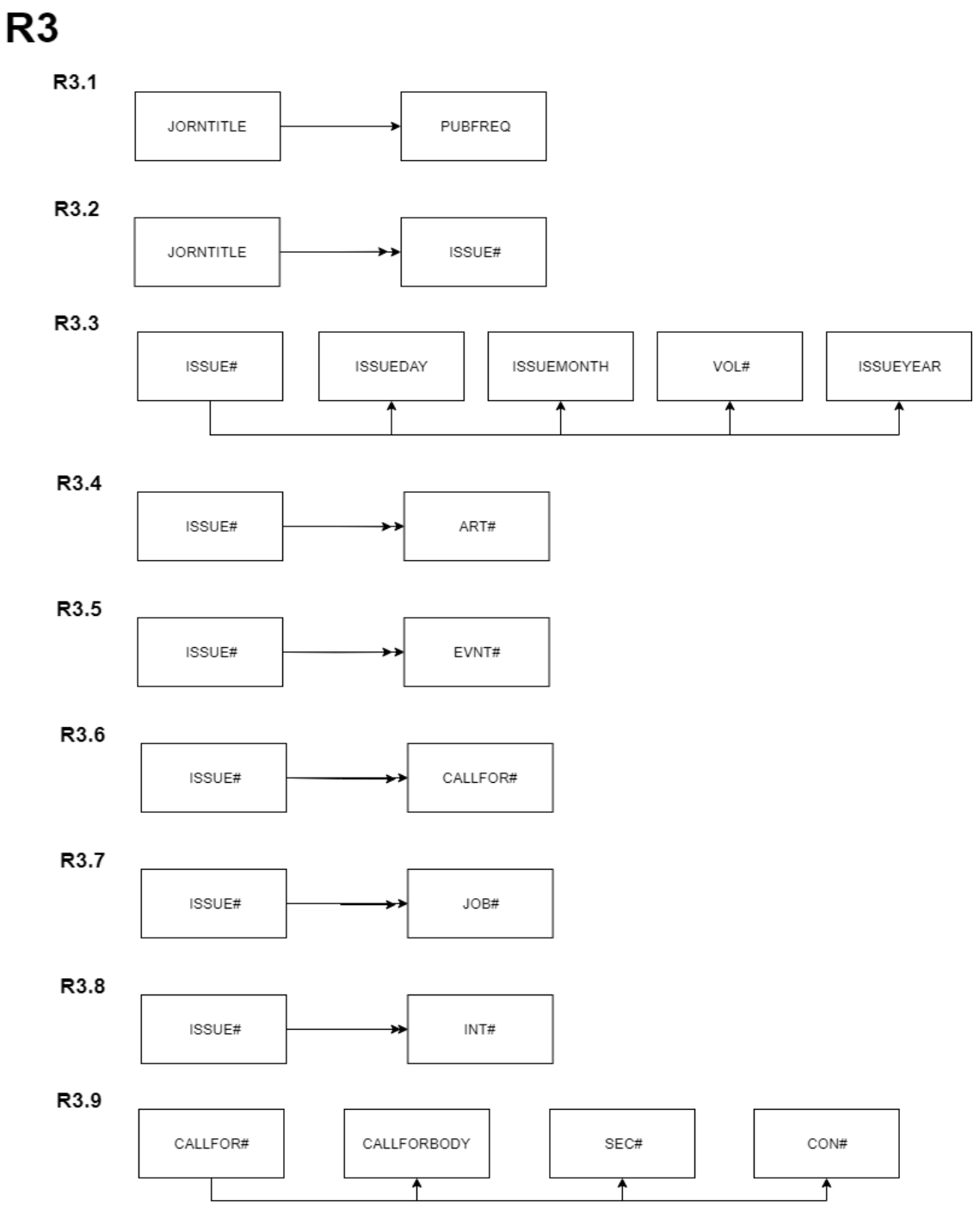
R2.15



4NF

R3

sub-relations 1-9

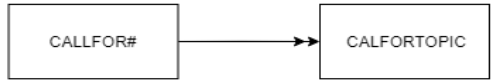


4NF

R3

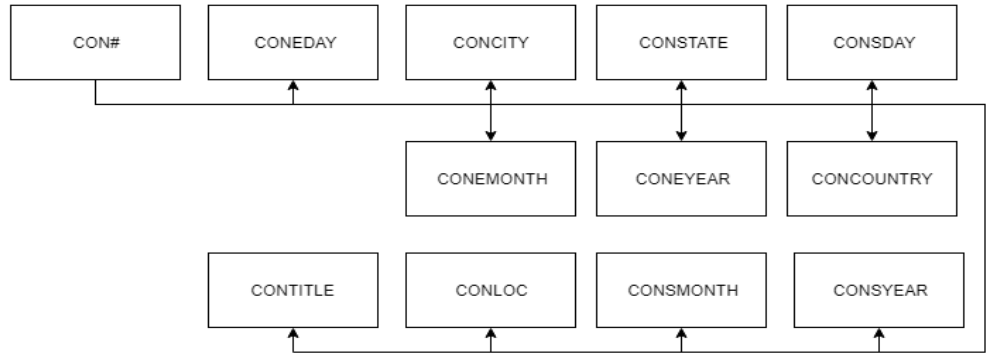
sub-relations 10-17

R3.10

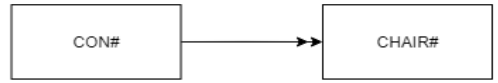


R3.11

R3.12



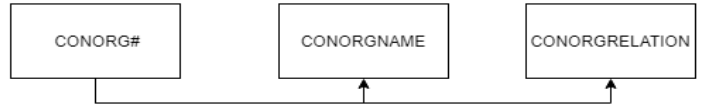
R3.13



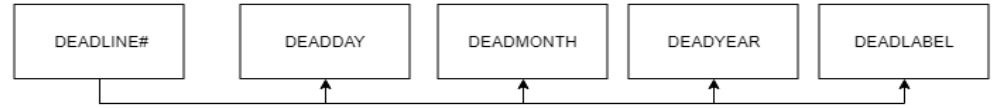
R3.14

R3.15

R3.16



R3.17

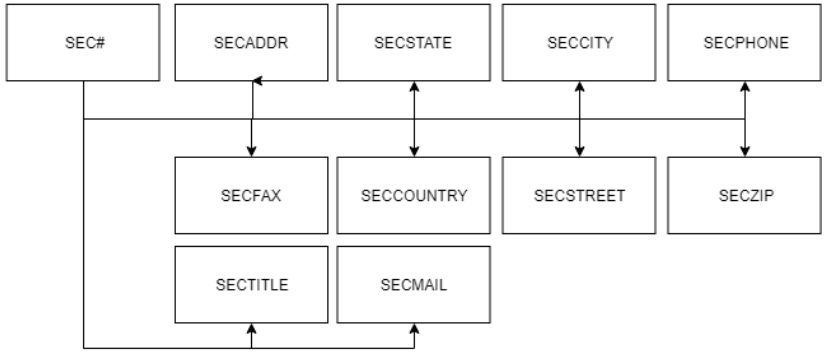


4NF

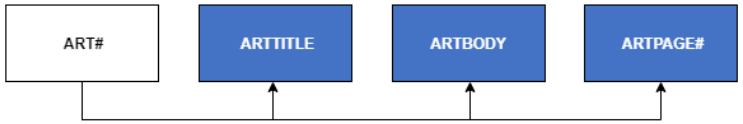
R3

sub-relations 18-25

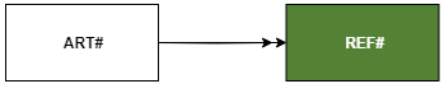
R3.18



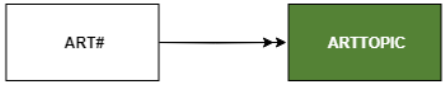
R3.19



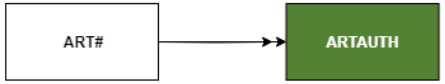
R3.20



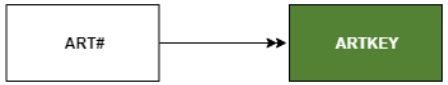
R3.21



R3.22

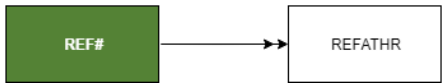


R3.23



R3.24

R3.25

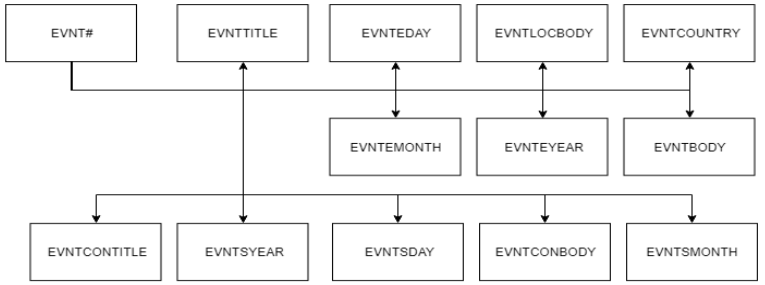


4NF

R3

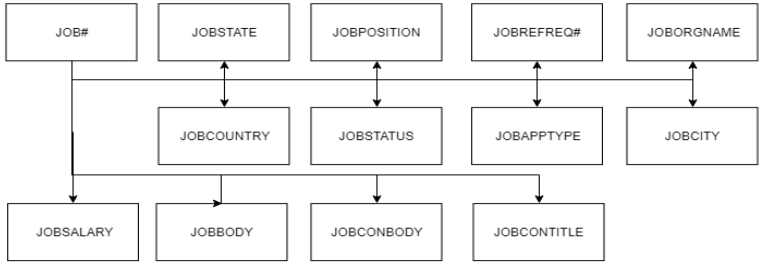
sub-relations 26-33

R3.26

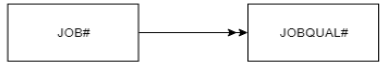


R3.27

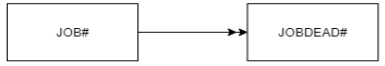
R3.28



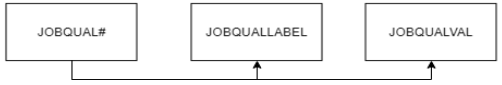
R3.29



R3.30

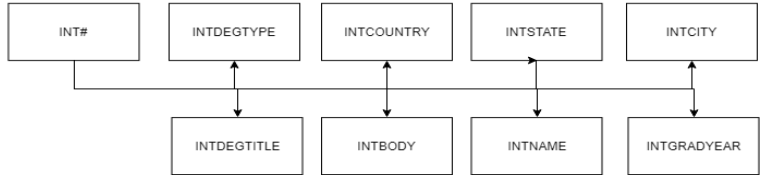


R3.31



R3.32

R3.33



Final Set of Relations

Derived from 4NF (R3) relations

Relation	Dependency
R3.1(JORNTITLE, PUBFREQ)	JORNTITLE -> PUBFREQ
R3.2(JORNTITLE, ISSUE#)	JORNTITLE ->-> ISSUE#
R3.3(ISSUE#, ISSUEDAY, ISSUEMONTH, VOL#, ISSUEYEAR)	ISSUE# -> ISSUEDAY, ISSUEMONTH, VOL#, ISSUEYEAR
R3.4(ISSUE#, ART#)	ISSUE ->-> ART#
R3.5(ISSUE#, EVNT#)	ISSUE# ->-> EVNT#
R3.6(ISSUE#, CALLFOR#)	ISSUE# ->-> CALLFOR#
R3.7(ISSUE#, JOB#)	ISSUE# ->-> JOB#
R3.8(ISSUE#, INT#)	ISSUE# ->-> INT#
R3.9(CALLFOR#, CALLFORBODY, SEC#, CON#)	CALLFOR# -> CALLFORBODY, SEC#, CON#
R3.10(CALLFOR#, CALFORTOPIC)	CALLFOR# ->-> CALFORTOPIC
R3.11(CALLFOR#, DEADLINE#)	CALLFOR# ->-> DEADLINE#
R3.12(CON#, CONEDAY, CONCITY, CONSTATE, CONSDAY, CONEMONTH, CONEYEAR, CONCOUNTRY, CONTITLE, CONLOC, CONSMONTH, CONSYEAR)	CON# -> CONEDAY, CONCITY, CONSTATE, CONSDAY, CONEMONTH, CONEYEAR, CONCOUNTRY, CONTITLE, CONLOC, CONSMONTH, CONSYEAR
R3.13(CON#, CHAIR#)	CON# ->-> CHAIR#
R3.14(CON#, CONORG#)	CON# ->-> CONORG#
R3.15(CHAIR#, CHAIRORGNAME, CHAIRNAME, CHAIRTYPE)	CHAIR# -> CHAIRORGNAME, CHAIRNAME, CHAIRTYPE
R3.16 (CONORG#, CONORGNAME, CONORGRELATION)	CONORG# -> CONORGNAME, CONORGRELATION
R3.17 (DEADLINE#, DEADDAY, DEADMONTH, DEADYEAR, DEADLABEL)	DEADLINE# -> DEADDAY, DEADMONTH, DEADYEAR, DEADLABEL
R3.18(SEC#, SECADDR, SECSTATE, SECCITY, SECPHONE, SECFAX, SECCOUNTRY, SECSTREET, SECZIP, SECTITLE, SECMAIL)	SEC# -> SECADDR, SECSTATE, SECCITY, SECPHONE, SECFAX, SECCOUNTRY, SECSTREET, SECZIP, SECTITLE, SECMAIL
R3.19 (ART#, ARTTITLE, ARTBODY, ARTPAGE#)	ART# -> ARTTITLE, ARTBODY, ARTPAGE#
R3.20(ART#, REF#)	ART# ->-> REF#
R3.21(ART#, ARTTOPIC)	ART# ->-> ARTTOPIC
R3.22(ART#, ARTAUTH)	ART# ->-> ARTAUTH
R3.23(ART#, ARTKEY)	ART# ->-> ARTKEY
R3.24(REF#, REFBODY, REFTITLE, REFTYPE)	REF# -> REFBODY, REFTITLE, REFTYPE
R3.25(REF#, REFATHR)	REF# ->-> REFATHR
R3.26(EVNT#, EVNTTITLE, EVNTEDAY, EVNTLOCBODY, EVNTCOUNTRY, EVNTEMONTH, EVNTEYEAR, EVNTBODY, EVNTCONTITLE, EVNTSYEAR, EVNTSDAY, EVNTCONBODY, EVNTSMONTH)	EVNT# -> EVNTTITLE, EVNTEDAY, EVNTLOCBODY, EVNTCOUNTRY, EVNTEMONTH, EVNTEYEAR, EVNTBODY, EVNTCONTITLE, EVNTSYEAR, EVNTSDAY, EVNTCONBODY, EVNTSMONTH
R3.27(EVNT#, EVNTSPONSOR)	EVNT# ->-> EVNTSPONSOR
R3.28(JOB#, JOBSTATE, JOBPOSITION, JOBREFREQ#, JOBORGNAME, JOBCOUNTRY, JOBSTATUS, JOBAPPTYPE, JOBCITY, JOBSALARY, JOBBODY, JOBCONBODY, JOBCONTITLE)	JOB# -> JOBSTATE, JOBPOSITION, JOBREFREQ#, JOBORGNAME, JOBCOUNTRY, JOBSTATUS, JOBAPPTYPE, JOBCITY, JOBSALARY, JOBBODY, JOBCONBODY, JOBCONTITLE
R3.29(JOB#, JOBQUAL#)	JOB# ->-> JOBQUAL#
R3.30(JOB#, JOBDEAD#)	JOB# ->-> JOBDEAD#
R3.31(JOBQUAL#, JOBQUALLABEL, JOBQUALVAL)	JOBQUAL# -> JOBQUALLABEL, JOBQUALVAL
R3.32(JOBDEAD#, JOBDEADYEAR, JOBDEADMONTH, JOBDEADDAY, JOBDEADLABEL)	JOBDEAD# -> JOBDEADYEAR, JOBDEADMONTH, JOBDEADDAY, JOBDEADLABEL

Table – Intension of Relations and Relation Dependencies

Create Database From 4NF Relations

Using MySQL

Step 1 : Create Schema

```
-- -----------------------------------------------------
-
-- Remove any previous relations
-- -----------------------------------------------------
-
DROP SCHEMA IF EXISTS `r3` ;

-- -----------------------------------------------------
-
-- Create new schema, r3
-- -----------------------------------------------------
-
CREATE SCHEMA IF NOT EXISTS `r3` DEFAULT CHARACTER SET
utf8 ;
USE `r3` ;
```

Create Database From 4NF Relations

Using MySQL

Step 2 : Create Relations (MySQL Tables)

```
-- -----------------------------------------------------
-- Create the relations of r3 in order of foreign key dependencies
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Create relation `r3`.`1`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`1` ;

CREATE TABLE IF NOT EXISTS `r3`.`1` (
  `JORNTITLE` VARCHAR(255) NOT NULL,
  `PUBFREQ` VARCHAR(45) NULL DEFAULT NULL,
  PRIMARY KEY (`JORNTITLE`),
  UNIQUE INDEX `JORNTITLE_UNIQUE` (`JORNTITLE` ASC))
DEFAULT CHARACTER SET = utf8;
```

Create Database From 4NF Relations

Using MySQL

Step 3 :

Create MySQL tables for all relations, but in order of their foreign key dependencies.

For Example

- Relations *R3.19* and *R3.24* must be created before *R3.20*
- Relation *R3.19* must be created before *R3.21*, *R3.22*, and *R3.23*

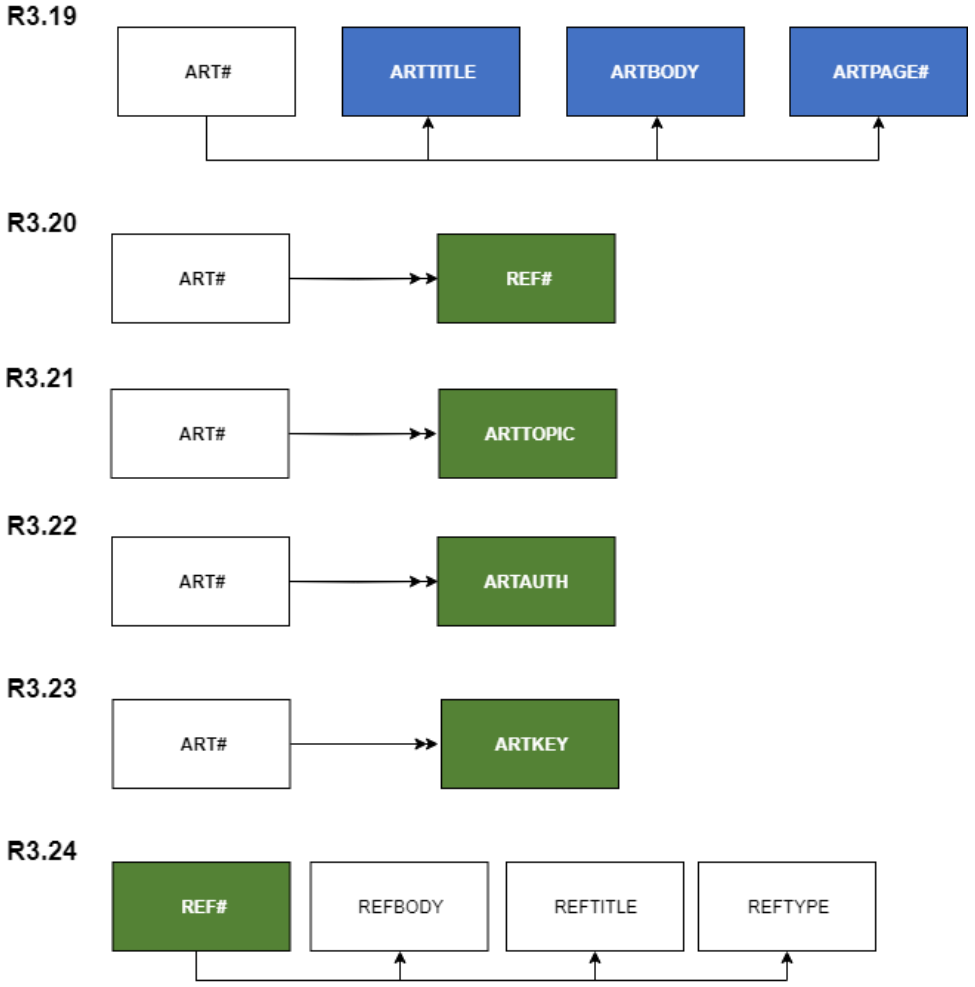


Table Creation Excerpts

In order of foreign key dependencies

First

```
-- -----------------------------------------------------
-- Create relation `r3`.`19`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`19` ;

CREATE TABLE IF NOT EXISTS `r3`.`19` (
  `ART#` INT(11) NOT NULL AUTO_INCREMENT,
  `ARTTITLE` VARCHAR(255) NULL DEFAULT NULL,
  `ARTBODY` TEXT NULL DEFAULT NULL,
  `ARTPAGE#` INT(11) NULL DEFAULT NULL,
  PRIMARY KEY (`ART#`))
DEFAULT CHARACTER SET = utf8;
```

Second

```
-- -----------------------------------------------------
-- Create relation `r3`.`24`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`24` ;

CREATE TABLE IF NOT EXISTS `r3`.`24` (
  `REF#` INT(11) NOT NULL AUTO_INCREMENT,
  `REFBODY` VARCHAR(4000) NULL DEFAULT NULL,
  `REFTITLE` VARCHAR(255) NULL DEFAULT NULL,
  `REFTYPE` VARCHAR(12) NULL DEFAULT NULL,
  PRIMARY KEY (`REF#`))
DEFAULT CHARACTER SET = utf8;
```

Third

```
-- -----------------------------------------------------
-- Create relation `r3`.`20`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`20` ;

CREATE TABLE IF NOT EXISTS `r3`.`20` (
  `ART#` INT(11) NOT NULL,
  `REF#` INT(11) NOT NULL,
  PRIMARY KEY (`ART#`, `REF#`),
  INDEX `REF#_Foreign_idx` (`REF#` ASC),
  CONSTRAINT `ART#_Foreign4`
    FOREIGN KEY (`ART#`)
    REFERENCES `r3`.`19` (`ART#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `REF#_Foreign`
    FOREIGN KEY (`REF#`)
    REFERENCES `r3`.`24` (`REF#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
DEFAULT CHARACTER SET = utf8;
```

Fourth

```
-- -----------------------------------------------------
-- Create relation `r3`.`21`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`21` ;

CREATE TABLE IF NOT EXISTS `r3`.`21` (
  `ART#` INT(11) NOT NULL,
  `ARTTOPIC` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`ART#`, `ARTTOPIC`),
  CONSTRAINT `ART#_Foreign3`
    FOREIGN KEY (`ART#`)
    REFERENCES `r3`.`19` (`ART#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
DEFAULT CHARACTER SET = utf8;
```

Fifth

```
-- -----------------------------------------------------
-- Create relation `r3`.`25`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`25` ;

CREATE TABLE IF NOT EXISTS `r3`.`25` (
  `REF#` INT(11) NOT NULL AUTO_INCREMENT,
  `REFATHR` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`REF#`, `REFATHR`),
  CONSTRAINT `REF#_Foreign2`
    FOREIGN KEY (`REF#`)
    REFERENCES `r3`.`24` (`REF#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
DEFAULT CHARACTER SET = utf8;
```

Sixth

```
-- -----------------------------------------------------
-- Create relation `r3`.`22`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`22` ;

CREATE TABLE IF NOT EXISTS `r3`.`22` (
  `ART#` INT(11) NOT NULL,
  `ARTAUTH` VARCHAR(255) NOT NULL,
  PRIMARY KEY (`ART#`, `ARTAUTH`),
  CONSTRAINT `ART#_Foreign2`
    FOREIGN KEY (`ART#`)
    REFERENCES `r3`.`19` (`ART#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
DEFAULT CHARACTER SET = utf8;
```

Seventh

```
-- -----------------------------------------------------
-- Create relation `r3`.`23`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `r3`.`23` ;

CREATE TABLE IF NOT EXISTS `r3`.`23` (
  `ART#` INT(11) NOT NULL,
  `ARTKEY` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`ART#`, `ARTKEY`),
  CONSTRAINT `ART#_Foreign1`
    FOREIGN KEY (`ART#`)
    REFERENCES `r3`.`19` (`ART#`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
DEFAULT CHARACTER SET = utf8;
```

Populate Database From Journal Excerpts

Using MySQL Insert Commands

Like table creation, tuples with foreign keys need to be inserted after their dependencies are inserted.

Step 1

```
INSERT INTO `r3`.`19` (`ART#`,`ARTTITLE`,`ARTBODY`,`ARTPAGE#`) VALUES (2,'Prediction Capability
of Neural Networks Trained in Monte-Carlo Paradigm','The Monte-Carlo training paradigm…', 3);
```

Step 2

```
(9,'Addison-Wesley, Redwood City….','Introduction to the Theory of Neural Computation','Book');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (10,'Science 247, February 2...','Neuroscience Models the
Brain','Article');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (11,'The International Journal of Methods..','Developmental
Toxicity Risk Assesment: A Rough Sets Approach','Article');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (12,'ANZA Research, Inc...','Neural Computing','Book');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (13,'Parallel Distributed Processing, ….','Learning and relearning
in Boltzmann machines','Book');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (14,'Journal of Science…','Optimization by simulated
annealing','Article');
INSERT INTO `r3`.`24` (`REF#`,`REFBODY`,`REFTITLE`,`REFTYPE`) VALUES (15,'Proceedings of the 1992 ACM/IEE ….','Conflict Resolution in
Learning Through Examples','Article');
```

Step 3

```
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,9);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,10);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,11);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,12);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,13);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,14);
INSERT INTO `r3`.`20` (`ART#`,`REF#`) VALUES (2,15);
```

Example SQL Query

Get the author name(s) of those articles that have more than six references and whose keywords don't contain "management"

Solution

```
SELECT A.ARTAUTH
FROM `r3`.`22` A
WHERE A.`ART#` IN (
  SELECT `ART#`
  FROM `r3`.`20`
  WHERE (
    SELECT COUNT(`REF#`)
    FROM `r3`.`20`
    WHERE `ART#` = A.`ART#`
  ) > 6
  AND `ART#` NOT IN (
    SELECT `ART#`
    FROM `r3`.`23`
    WHERE ARTKEY LIKE '%management%'
  )
);
```

Output

```
+-------------+
| ARTAUTH     |
+-------------+
| Ray Hashemi |
+-------------+
1 row in set (0.01 sec)
```

Reference

R3.19 (ART#, ARTTITLE, ARTBODY, ARTPAGE#)
R3.20(ART#, REF#)
R3.21(ART#, ARTTOPIC)
R3.22(ART#, ARTAUTH)
R3.23(ART#, ARTKEY)
R3.24(REF#, REFBODY, REFTITLE, REFTYPE)
R3.25(REF#, REFATHR)

References

Ray Hashemi, Lecture #2, January 10, 2018.
Ray Hashemi, Lecture #4, January 22, 2018.
Ray Hashemi, Lecture #5, February 27, 2018.
Ray Hashemi, Lecture #9, February 7, 2018.
Ray Hashemi, Lecture #10, February 12, 2018.
Ray Hashemi, Lecture #11, February 14, 2018.
Ray Hashemi, Lecture #12, February 19, 2018.