

@author Valerio Del Grande

@title IMDB Mappings (GAV)

@version 22/10/2008

Versione Preliminare

Tutte le query testate su mySQL 5.0

1) ACTORS

Actor(x) \wedge name(x,y) \wedge surname(x,z) \wedge sex(x,w)

-

```
SELECT SUBSTRING_INDEX(a.name, ' ', -1) as name,
SUBSTRING_INDEX(a.name, ' ', 1) as surname, sex
FROM actors a
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

w \rightarrow sex xsd:Char

2) CINEMATGRS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,
SUBSTRING_INDEX(c.name, ' ', 1) as surname
FROM cinematgrs c
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

3) COMPOSERS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,
SUBSTRING_INDEX(c.name, ' ', 1) as surname
FROM composers c
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

4) COSTDESIGNERS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,  
SUBSTRING_INDEX(c.name, ' ', 1) as surname  
FROM costdesigners c
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

5) DIRECTORS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(d.name, ' ', -1) as name,  
SUBSTRING_INDEX(d.name, ' ', 1) as surname  
FROM directors d
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

5) DISTRIBUTORS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(d.name, ' ', -1) as name,  
SUBSTRING_INDEX(d.name, ' ', 1) as surname  
FROM distributors d
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

6) EDITORS

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(e.name, ' ', -1) as name,
```

```
SUBSTRING_INDEX(e.name, ',', 1) as surname  
FROM editors e
```

x → person(name, surname)

y → name xsd:String

z → surname xsd:String

7) PRODDESIGNERS

Person(x) ∧ name(x,y) ∧ surname(x,z)

-

```
SELECT SUBSTRING_INDEX(p.name, ',', -1) as name,  
SUBSTRING_INDEX(p.name, ',', 1) as surname  
FROM proddesigners p
```

x → person(name, surname)

y → name xsd:String

z → surname xsd:String

8) PRODUCERS

Person(x) ∧ name(x,y) ∧ surname(x,z)

-

```
SELECT SUBSTRING_INDEX(p.name, ',', -1) as name,  
SUBSTRING_INDEX(p.name, ',', 1) as surname  
FROM producers p
```

x → person(name, surname)

y → name xsd:String

z → surname xsd:String

9) WRITERS

Person(x) ∧ name(x,y) ∧ surname(x,z)

-

```
SELECT SUBSTRING_INDEX(w.name, ',', -1) as name,  
SUBSTRING_INDEX(w.name, ',', 1) as surname  
FROM writers w
```

x → person(name, surname)

y → name xsd:String

z → surname xsd:String

10) MISC

Person(x) \wedge name(x,y) \wedge surname(x,z)

-

```
SELECT SUBSTRING_INDEX(m.name, ', ', -1) as name,  
SUBSTRING_INDEX(m.name, ', ', 1) as surname  
FROM misc m
```

x \rightarrow person(name, surname)

y \rightarrow name xsd:String

z \rightarrow surname xsd:String

11) AKANAMES

Person(x) \wedge akaNames(x,y)

-

```
SELECT SUBSTRING_INDEX(a.name, ', ', -1) as name,  
SUBSTRING_INDEX(a.name, ', ', 1) as surname, akaname  
FROM akanames a
```

x \rightarrow person(name, surname)

y \rightarrow akaname xsd:String

12) BIOGRAPHIES

Person(x) \wedge biography(x,y)

-

```
SELECT SUBSTRING_INDEX(b.name, ', ', -1) as name,  
SUBSTRING_INDEX(b.name, ', ', 1) as surname, biotext  
FROM biographies b
```

x \rightarrow person(name, surname)

y \rightarrow biotext xsd:String

13) MOVIES

Movie(x) \wedge movieTitle(x,y) \wedge year(x,z)

-

```
SELECT movieid, title, year  
FROM movies WHERE title not like '%'
```

x → movie(movieid)
y → title xsd:String
z → year xsd:String

14) MOVIES (parsing TvSeries)

TvSerie(x) ∧ movieTitle(x,y) ∧ year(x,z) ∧ episodeTitle(x,w) ∧ seasonNo(x,a)
∧ episodeNo(x,b)

-

```
SELECT movieid,  
SUBSTRING(SUBSTRING_INDEX(title, '', 2), 2) as serie_title,  
SUBSTRING_INDEX(SUBSTRING_INDEX(title, ')', 1), '(', -1) as year,  
SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(title, '}', 1), '{', -1),  
'(', 1) as episode_title,  
SUBSTRING(SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(title, ')', 2),  
'(', -1), '.', 1), 2) as season,  
SUBSTRING_INDEX(SUBSTRING_INDEX(SUBSTRING_INDEX(title, ')', 2), '(', -1), '.',  
-1) as episode  
FROM movies WHERE title like '%"'
```

x → tvserie(movieid)
y → serie_title xsd:String
z → year xsd:String
w → episode_title xsd:String
a → season xsd:String
b → episode xsd:String

15) COLORINFO

Movie(x) ∧ color(x,y)

-

```
SELECT movieid, color FROM colorinfo
```

x → movie(movieid)
y → color xsd:String

16) COUNTRIES

Country(x) ∧ countryName(x,y)

-

```
SELECT DISTINCT country FROM countries
```

x → country(country)
y → country xsd:String

17) COUNTRIES (producedIn)

$\text{producedIn}(x,y) \wedge \text{Movie}(x) \wedge \text{Country}(y)$

-

SELECT movieid, country FROM countries

$x \rightarrow \text{movie}(\text{movieid})$

$y \rightarrow \text{country}(\text{country})$

18) GENRES

$\text{Movie}(x) \wedge \text{genre}(x,y)$

-

SELECT movieid, genre FROM genres

$x \rightarrow \text{movie}(\text{movieid})$

$y \rightarrow \text{genre } \text{xsd:String}$

19) GOOFS

$\text{Movie}(x) \wedge \text{goofs}(x,y)$

-

SELECT movieid, gooftext FROM goofs

$x \rightarrow \text{movie}(\text{movieid})$

$y \rightarrow \text{gooftext } \text{xsd:String}$

20) KEYWORDS

$\text{Movie}(x) \wedge \text{keyword}(x,y)$

-

SELECT movieid, keyword FROM keywords

$x \rightarrow \text{movie}(\text{movieid})$

$y \rightarrow \text{keyword } \text{xsd:String}$

21) LANGUAGE

$\text{Movie}(x) \wedge \text{language}(x,y)$

-

SELECT movieid, language FROM language

$x \rightarrow \text{movie}(\text{movieid})$

y → language xsd:String

22) MPAARATINGS

Movie(x) ∧ mpaa(x,y)

-

SELECT movieid, reasontext FROM mpaaratings

x → movie(movieid)

y → reasontext xsd:String

23) PLOTS

Movie(x) ∧ plotSynopsis(x,y)

-

SELECT movieid, plottext FROM plots

x → movie(movieid)

y → plottext xsd:String

24) QUOTES

Movie(x) ∧ quotes(x,y)

-

SELECT movieid, quotetext FROM quotes

x → movie(movieid)

y → quotetext xsd:String

25) RATINGS

Movie(x) ∧ rating(x,y) ∧ votes(x,z)

-

SELECT movieid, rank, votes FROM ratings

x → movie(movieid)

y → rank xsd:String

z → votes xsd:String

26) RUNNINGTIMES

Movie(x) ∧ runTime(x,y)

-

SELECT movieid, time FROM runningtimes

x → movie(movieid)
y → time xsd:String

27) TAGLINES

Movie(x) ∧ tagline(x,y)

-

SELECT movieid, taglinetext FROM taglines

x → movie(movieid)
y → taglinetext xsd:String

28) TRIVIA

Movie(x) ∧ movieTrivia(x,y)

-

SELECT movieid, triviastext FROM trivia

x → movie(movieid)
y → triviastext xsd:String

29) PRODCOMPANIES

Company(x) ∧ companyName(x,y)

-

SELECT DISTINCT SUBSTRING_INDEX(name, '-', 1) as name FROM prodcompanies

x → company(name)
y → name xsd:String

30) PRODCOMPANIES (productionCompany)

productionCompany(x,y) ∧ Movie(x) ∧ Company(y) ∧ role30(x,y,z)

-

SELECT movieid,
SUBSTRING_INDEX(name, '-', 1) as name,
IF(name LIKE '% - (%)', TRIM(TRAILING ')') FROM SUBSTRING_INDEX(name, '(',
-1)), NULL)as addition
FROM prodcompanies

x → movie(movieid)
y → company(name)
z → addition xsd:String

31) RELEASEDATES

releasedIn(x,y) \wedge Movie(x) \wedge Country(y) \wedge releaseDate(x,y,z)

-

SELECT movieid, country, releasedate FROM releasedates

x \rightarrow movie(movieid)

y \rightarrow country(country)

z \rightarrow releasedate xsd:String

32) AKATITLES

releasedIn(x,y) \wedge Movie(x) \wedge Country(y) \wedge akaTitle(x,y,z)

-

SELECT movieid, title,
TRIM(LEADING '(' FROM SUBSTRING_INDEX(SUBSTRING_INDEX(addition, ')', 1), ':',
1)) as country
FROM akatitles

x \rightarrow movie(movieid)

y \rightarrow country(country)

z \rightarrow title xsd:String

33) CERTIFICATES

Movie(x) \wedge certificate(x,y)

-

SELECT movieid, CONCAT(country, ' : ', certification) as cert
FROM certificates

x \rightarrow movie(movieid)

y \rightarrow cert xsd:String

34) MOVIES2CINEMATGRS

cinematographyBy(x,y) \wedge Person(x) \wedge Movie(y) \wedge role6(x,y,z)

-

SELECT SUBSTRING_INDEX(c.name, ',', -1) as name,
SUBSTRING_INDEX(c.name, ',', 1) as surname, movieid, addition
FROM movies2cinematgrs m, cinematgrs c
WHERE m.cinematid = c.cinematid

x \rightarrow person(name, surname)

y \rightarrow movie(movieid)

z \rightarrow addition xsd:String

35) MOVIES2COMPOSERS

composedBy(x,y) \wedge Person(x) \wedge Movie(y) \wedge role5(x,y,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,  
SUBSTRING_INDEX(c.name, ' ', 1) as surname, movieid, addition  
FROM movies2composers m, composers c  
WHERE m.composerid = c.composerid
```

x \rightarrow person(name, surname)

y \rightarrow movie(movieid)

z \rightarrow addition xsd:String

36) MOVIES2COSTDES

costumeDesignBy(x,y) \wedge Person(x) \wedge Movie(y) \wedge role7(x,y,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,  
SUBSTRING_INDEX(c.name, ' ', 1) as surname, movieid, addition  
FROM movies2costdes m, costdesigners c  
WHERE m.costdesid = c.costdesid
```

x \rightarrow person(name, surname)

y \rightarrow movie(movieid)

z \rightarrow addition xsd:String

37) MOVIES2DIRECTORS

directedBy(x,y) \wedge Person(x) \wedge Movie(y) \wedge role1(x,y,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,  
SUBSTRING_INDEX(c.name, ' ', 1) as surname, movieid, addition  
FROM movies2directors m, directors c  
WHERE m.directorid = c.directorid
```

x \rightarrow person(name, surname)

y \rightarrow movie(movieid)

z \rightarrow addition xsd:String

38) MOVIES2EDITORS

editedBy(x,y) \wedge Person(x) \wedge Movie(y) \wedge role4(x,y,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ' ', -1) as name,
```

```
SUBSTRING_INDEX(c.name, ',', 1) as surname, movieid, addition
FROM movies2editors m, editors c
WHERE m.editorid = c.editorid
```

x → person(name, surname)
y → movie(movieid)
z → addition xsd:String

39) MOVIES2PRODDDES

```
productionDesignBy(x,y) ∧ Person(x) ∧ Movie(y) ∧ role8(x,y,z)
-
SELECT SUBSTRING_INDEX(c.name, ',', -1) as name,
SUBSTRING_INDEX(c.name, ',', 1) as surname, movieid, addition
FROM movies2prodddes m, prodddesigners c
WHERE m.prodddesid = c.prodddesid
```

x → person(name, surname)
y → movie(movieid)
z → addition xsd:String

40) MOVIES2PRODUCERS

```
producedBy(x,y) ∧ Person(x) ∧ Movie(y) ∧ role3(x,y,z)
-
SELECT SUBSTRING_INDEX(c.name, ',', -1) as name,
SUBSTRING_INDEX(c.name, ',', 1) as surname, movieid, addition
FROM movies2producers m, producers c
WHERE m.producerid = c.producerid
```

x → person(name, surname)
y → movie(movieid)
z → addition xsd:String

41) MOVIES2WRITERS

```
writtenBy(x,y) ∧ Person(x) ∧ Movie(y) ∧ role2(x,y,z)
-
SELECT SUBSTRING_INDEX(c.name, ',', -1) as name,
SUBSTRING_INDEX(c.name, ',', 1) as surname, movieid, addition
FROM movies2writers m, writers c
WHERE m.writerid = c.writerid
```

x → person(name, surname)
y → movie(movieid)

z → addition xsd:String

42) MOVIES2MISC

otherCrew(x,y) ∧ Person(x) ∧ Movie(y) ∧ role28(x,y,z)

-

```
SELECT SUBSTRING_INDEX(c.name, ', ', -1) as name,  
SUBSTRING_INDEX(c.name, ', ', 1) as surname, movieid, addition  
FROM movies2misc m, misc c  
WHERE m.miscid = c.miscid
```

x → person(name, surname)

y → movie(movieid)

z → addition xsd:String

43) MOVIE2ACTORS

cast(x,y) ∧ Actor(x) ∧ Movie(y) ∧ character(x,y,z) ∧ sex(x,w)

-

```
SELECT SUBSTRING_INDEX(a.name, ', ', -1) as name,  
SUBSTRING_INDEX(a.name, ', ', 1) as surname, sex, movieid,  
SUBSTRING_INDEX(SUBSTRING_INDEX(as_character, ']', 1), '[', -1) as  
charactername  
FROM movies2actors m, actors a  
WHERE m.actorid = a.actorid
```

x → person(name, surname)

y → movie(movieid)

z → charactername xsd:String

w → sex xsd:String

44) MOVIELINKS (referencedIn)

referencedIn(x,y) ∧ Movie(x) ∧ Movie(y)

-

```
SELECT movielinks.movieid as this, movies.movieid as that  
FROM movielinks, movies  
WHERE movielinkstext like 'referenced in%' and SUBSTRING(movielinkstext, 15)  
= movies.title
```

x → movie(this)

y → movie(that)

45) MOVIELINKS (references)

references(x,y) \wedge Movie(x) \wedge Movie(y)

-

```
SELECT movielinks.movieid as this, movies.movieid as that
FROM movielinks, movies
WHERE movielinkstext like 'references%' and SUBSTRING(movielinkstext, 12) =
movies.title
```

x \rightarrow movie(this)

y \rightarrow movie(that)

46) MOVIELINKS (featuredIn)

featuredIn(x,y) \wedge Movie(x) \wedge Movie(y)

-

```
SELECT movielinks.movieid as this, movies.movieid as that
FROM movielinks, movies
WHERE movielinkstext like 'featured in%' and SUBSTRING(movielinkstext, 13) =
movies.title
```

x \rightarrow movie(this)

y \rightarrow movie(that)

47) MOVIELINKS (features)

features(x,y) \wedge Movie(x) \wedge Movie(y)

-

```
SELECT movielinks.movieid as this, movies.movieid as that
FROM movielinks, movies
WHERE movielinkstext like 'features%' and SUBSTRING(movielinkstext, 10) =
movies.title
```

x \rightarrow movie(this)

y \rightarrow movie(that)

48) MOVIELINKS (followedBy)

followedBy(x,y) \wedge Movie(x) \wedge Movie(y)

-

```
SELECT movielinks.movieid as this, movies.movieid as that
FROM movielinks, movies
WHERE movielinkstext like 'followed by%' and SUBSTRING(movielinkstext, 13) =
movies.title
```

x \rightarrow movie(this)

y \rightarrow movie(that)

49) MOVIELINKS (remakeOf)

remakeOf(x,y) \wedge Movie(x) \wedge Movie(y)

-

SELECT movielinks.movieid as this, movies.movieid as that

FROM movielinks, movies

WHERE movielinkstext like 'remake of' and SUBSTRING(movielinkstext, 11) =
movies.title

x \rightarrow movie(this)

y \rightarrow movie(that)

50) MOVIELINKS (spinOffFrom)

spinOffFrom(x,y) \wedge Movie(x) \wedge Movie(y)

-

SELECT movielinks.movieid as this, movies.movieid as that

FROM movielinks, movies

WHERE movielinkstext like 'spin off from%' and SUBSTRING(movielinkstext, 15)
= movies.title

x \rightarrow movie(this)

y \rightarrow movie(that)