



데이터 관리와 분석
2조 09학번 이용은
10학번 황영석
13학번 강영은
13학번 이정원

CONCEPTUAL DB DESIGN
주제 : 인천 아시안 게임
2014년 10월 1일

목차

- I. 문제 정의 (Problem Description)
- II. 아시안 게임 DB의 구성 요소
- III. ER 다이어그램 (ER Diagram)
- IV. 관계 스키마 (Relation Schema)
- V. 관계 스키마와 Constraints
- VI. Redundancy Issue들
- VII. 데이터 베이스 서치 & 통계 시나리오

문제 정의(Problem Description)

본 프로젝트 #1: Conceptual DB design에 우리 2조가 선정한 주제는 요즘 한창 진행되고 있는 아시안 게임이다. 2014년 인천에서 열리는 아시안 게임에 국내외 아시아 대륙의 관심이 집중되고 있다. 언론과 외신 뿐만 아니라 게임을 관람하는 일반 대중들도 아시안 게임을 즐기며 40억 아시아인의 축제에 참여하고 있다. 일반 대중들은 국내에선 경기장을 찾아가 직접 관람하기도 하지만 주로 인터넷 등을 이용하여 각종 뉴스와 데이터들을 접하고 있다. 스포츠에 조금이라도 관심이 있는 사람이라면 각 종목 선수들의 기록 소식, 금메달 소식 등에 관한 데이터를 찾아 보았기 마련이다. 또한 국가 별 메달 집계 순위도 꼼꼼히 검색하며 현재 어느 나라가 종합 메달 순위 상위권에 있는지 찾아본 적도 아마 있을지 모른다.

이러한 사용자들의 요구를 파악하여 우리는 일반 대중들이 관심 있을 만한 정보를 제공하는 데이터 베이스를 만들기로 하였다. (본 프로젝트는 Conceptual design에 한하기 때문에 개념적인 설계를 다루는 것으로 하자.) 아시안 게임에 관련한 데이터에 관심을 두는 일반 대중들과 조금 더 나아가 관련 뉴스를 양산할 수 있는 기자들 정도가 우리의 데이터 베이스 시스템의 End-User이다.

아시안 게임 데이터 베이스를 설계하기 위해 우리는 사용자들의 요구 사항을 파악하였으며 그에 따라 Entity들과 Attribute들을 설정하고, 관계를 만들어 ER 다이어그램을 만들었다. 또한 ER 다이어그램을 사상하여 관계 스키마를 만들었고 다양한 무결성 제약 조건 역시 설정했다.

아시안게임 DB의 구성 요소

아시안게임 DB 시스템은 Entity Type과 Relationship Type, 그리고 Attributes의 기본 요소로 구성된다. 여기서 아시안게임은 실재하는 대상이며, 앞으로의 시나리오 분석 등의 과정에서 통계, 분석이 수행 가능하므로 (R1)의 조건을 만족한다.

여기에 '아시안게임에 관심이 있는 평범한 관람객' 이라는 View를 설정하고 이 아시안게임 DB 시스템의 일부를 구성해보았다.

각 구성 요소의 개수의 통계를 내보면, Entity Type은 6개, Relationship Type은 7개, Attributes는 32개이다. 따라서 (R2) 의 조건을 만족한다.

1. Entity Type

1.1. PLAYER

아시안 게임에 참가하는 각 개인의 선수를 의미한다. 아시아 각국에서 모두 13,000여 명의 선수들이 참가했다.

다음은 attributes의 구성이다.

Player_code(Primary Key) : 선수에게 부여되어 고유 식별자로 쓰이는 코드이다.

Player_name : 선수의 본명이다.

Date_of_birth : 선수의 생년월일이다.

Sex : 선수의 성별이다.

Age(Derived attributes) : 선수의 나이이다. Date_of_birth로부터 유도된다.

Weight : 선수의 체중(Kg)이다.

Height : 선수의 신장(Cm)이다.

1.2. SPORTS

아시안 게임의 경기 종목을 의미한다. 예를 들어 수영 자유형 100m, 축구, 배드민턴 단식 등이 있다.

다음은 attributes의 구성이다.

Sports_code(Primary Key) : 종목에게 부여되어 고유 식별자로 쓰이는 코드이다.

Sports_name(Composite attributes) : 종목의 명칭이다.

Category_name(Simple attributes) : 큰 카테고리의 구분이다. ex) 양궁, 수영

Sub_theme(Simple attributes) : 카테고리의 세부 종목의 구분이다. ex) 100m 남자 자유형

Gender_type : 여성/남성/혼성으로 종목을 구분한다. 조회나 통계를 위해 별도로 저장한다.

1.3. GAME

종목에 포함된 전체 경기의 각각의 특성을 나타낸다. 예를 들어 수영 자유형 100m 예선전 1라운드, 축구 16강 등의 요소들을 명시한다.

다음은 attributes의 구성이다.

Game_code(Primary Key) : 각 경기에게 부여되어 고유 식별자로 쓰이는 코드이다.

Round : 예선 1라운드/예선 2라운드/본선 등으로 종목을 구분한다.

Date : 구체적인 경기 일자를 나타낸다. 예를 들어 2014년 9월 29일 오후 3시 20분

Ticketprice(Multivalued attributes) : 1등석부터의 티켓 가격을 티켓의 종류와 상관없이 구분하도록 Multi Valued 형태로 나타낸다.

1.4. COUNTRY

아시안 게임에 참가한 국가를 나타낸다.

다음은 attributes의 구성이다.

Country_code(Primary Key) : 국가별로 부여되며 고유 식별자로 쓰이는 코드이다.

Countryname : 국가의 명칭(Full name)을 나타낸다.

Of Gold : 요번 아시안 게임에서 현재까지 받은 금메달 수를 나타낸다.

Of Silver : 요번 아시안 게임에서 현재까지 받은 은메달 수를 나타낸다.

Of Bronze : 요번 아시안 게임에서 현재까지 받은 동메달 수를 나타낸다.

1.5. STADIUM

아시안게임 주최에 사용된 경기장을 나타낸다. 인천시의 경기장 데이터를 사용한다.

다음은 attributes의 구성이다.

Stadium_No(Primary Key) : 경기장 번호를 나타낸다.

Stadium_name : 경기장 명칭을 나타낸다.

Capability : 경기장의 수용 인원을 나타낸다.

Address : 경기장의 정확한 주소를 나타낸다.

1.6. BROADCAST

방송사와 시간대를 고려한 코드 별로 방송 파일을 전송하는 방송 기지국의 DB의 일부로, 코드 별 정보를 나타낸다. 방송 기지국에서는 해당 시간대에 Schedule-Code를 부여하여 전파로 방송에 내보낸다.

다음은 attributes의 구성이다.

Schedule_code(Primary Key) : 방송마다 부여되어 고유 식별자로 쓰이는 코드이다.

Channel : KBS2/MBC/SBS 등의 방송국의 채널을 의미한다.

Time : 시간대를 나타낸다. 예를 들어 14:00-14:30, 15:30-16:00 등이 있다.

Original : 해당 시간대에 원래 방송되던 프로그램을 나타낸다. 예를 들어 21:00-21:30, 21:30-22:00 KBS2엔 원래 <왔다! 장보리>가 방송되므로 왔다! 장보리 프로그램의 식별코드를 입력한다.

2. Relationship

2.1. ROOMATE(PLATER:PLAYER=1:1, RECURSIVE)

Role Name은 A와 B이며 선수들끼리 룸메이트로 지정되는 관계를 나타낸다. 아시안 게임 선수촌은 2인 1실로 구성되어 있다. 카디날리티 비율은 1:1이며 Recursive relationship이다. 둘 다 Partial Participation이다. (굳이 선수촌에 머무르지 않을 수도, 일찍 떠날 수도, 2인실을 혼자 쓰게 되어 룸메이트가 없을 수도 있다.)

2.2. IS FROM(PLAYER:COUNTRY=N:1)

선수들의 국적이 어디냐를 나타내는 관계를 나타낸다. 카디날리티 비율은 N:1이다. 즉, 한 국

가에 N명의 선수가 포함되어 있다. 둘 다 Total Participation이다.

2.3. SG(SPORTS:GAME=1:N)

한 종목의 세부 경기 라운드에 대한 관계를 나타낸다. 카디날리티 비율은 1:N이다. 즉, 한 종목에 여러 경기 라운드가 포함되어 있다. 둘 다 Total Participation이다.

2.4. CAN PLAY(PLAYER:SPORTS=M:N)

선수들의 종목에 참가했는지에 대한 관계를 나타낸다. 카디날리티 비율은 M:N이다. 즉, 한 종목에 여러 명의 선수가 참가할 수 있고, 또 한 선수가 여러 종목에 참가할 수 있다. 둘 다 Total Participation이다.

다음은 attributes의 구성이다.

Medal_type : 금/은/동/Null 로 구분되며, 선수가 해당 종목에서 어떤 메달을 받았는가, 혹은 받지 못 하였는가를 알려준다.

2.5. PARTICIPATE IN(PLAYER:GAME=M:N)

선수들이 각 경기 라운드에 출전했는지에 대한 관계를 나타낸다. 카디날리티 비율은 M:N이다. 즉, 한 선수가 여러 경기 라운드에 출전할 수 있고 한 경기 라운드에 여러 선수가 출전한다. 둘 다 Total Participation이다.

다음은 attributes의 구성이다.

Rank : 각 경기 라운드에서 선수가 기록한 순위를 나타낸다.

2.6. HELD IN(STADIUM:GAME=1:N)

한 게임 라운드가 어떤 경기장에서 개최되느냐에 대한 관계를 나타낸다. 카디날리티 비율은 1:N이다. 즉, 한 경기장에서 여러 게임 라운드가 개최될 수 있다. Stadium은 Partial Participation이며 Game은 Total Participation이다. Stadium은 경기장 사정에 따라 임시 폐쇄 혹은 사용 불가 상태가 될 수도 있기 때문이다.

2.7. ON-AIR(BROADCAST:GAME=M:N)

한 방송 코드에서 어떤 경기 라운드가 중계되는지에 대한 관계를 나타낸다. 카디날리티 비율은 M:N이다. 즉, 한 방송 코드에서 여러 경기 라운드가 중계될 수 있고, 한 경기가 여러 방송 코드에서 중계될 수도 있다. 둘 다 Partial Participation인데, 방송 코드가 사용이 안될 수도 있고, 또 중계 방송을 하지 않는 경기도 있기 때문이다.

다음은 attributes의 구성이다.

Ratings : 시청률을 나타낸다.

이상의 설명을 통해 (R3)의 조건을 만족한다.

2.1.의 ROOMATE relationship은 recursive relationship으로써 카디날리티 비율이 1:1이다. 또한 2.4.,2.5.,2.7.의 relationship의 카디날리티 비율이 M:N이고, 나머지는 카디날리티 비율이 1:N이므로 이진 관계의 카디날리티 비율의 모든 종류가 나타난다. 따라서 (R4)의 조건 또한 만족한다.

Ticketprice는 1등석 등 모든 종류의 좌석의 가격을 나타내는 Multivalued attributes이므로 (R5)의 조건도 역시 만족한다.

마지막으로 2.2.,2.3.,2.4.,2.5.,2.6.에서 total participation이 나타나므로 (R6)의 조건도 만족한다.

ER 다이어그램(ER Diagram)

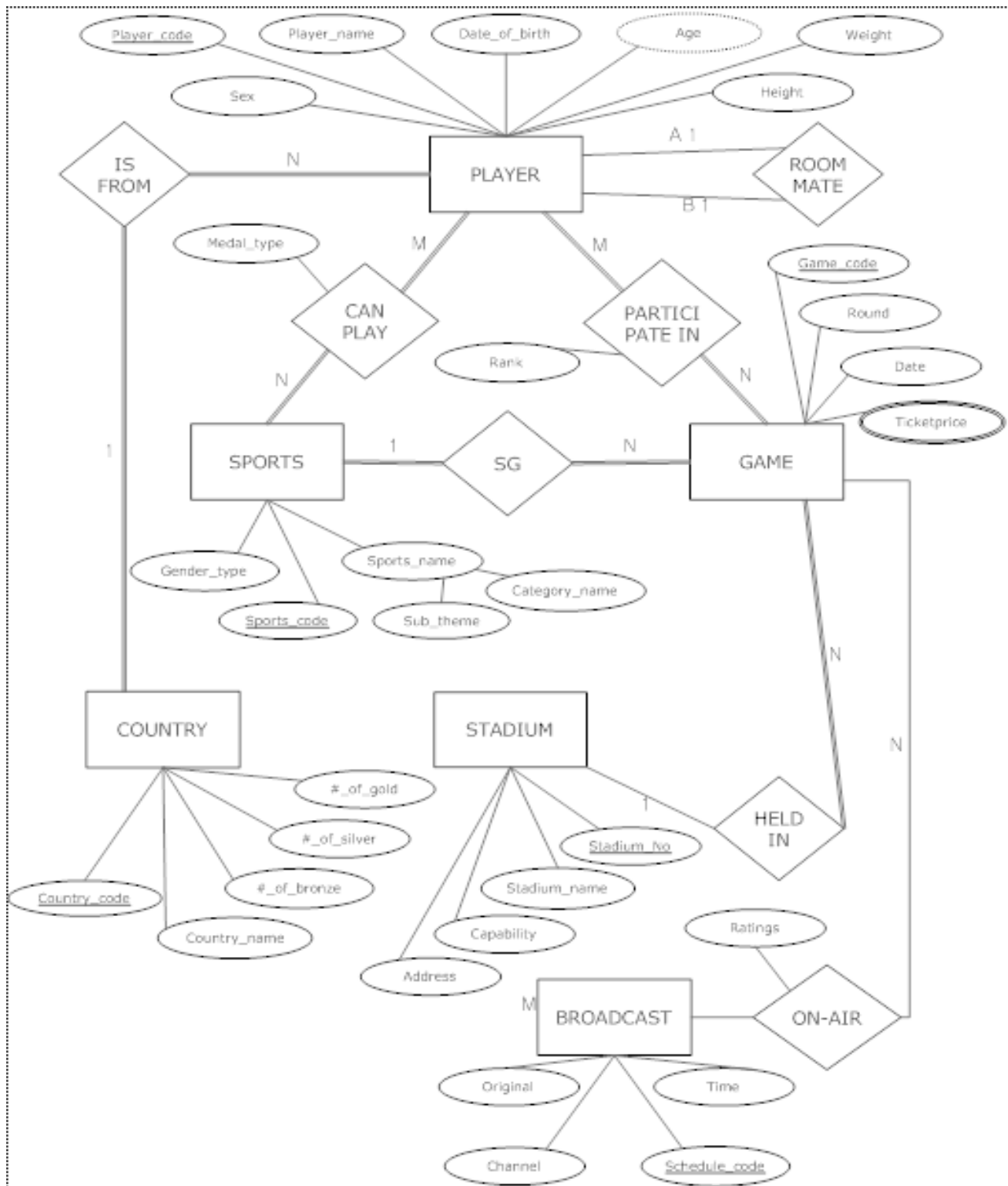


그림 1 ER Diagram

관계 스키마(Relation Schema)

ER Diagram을 관계 스키마로 Mapping한 것이다.

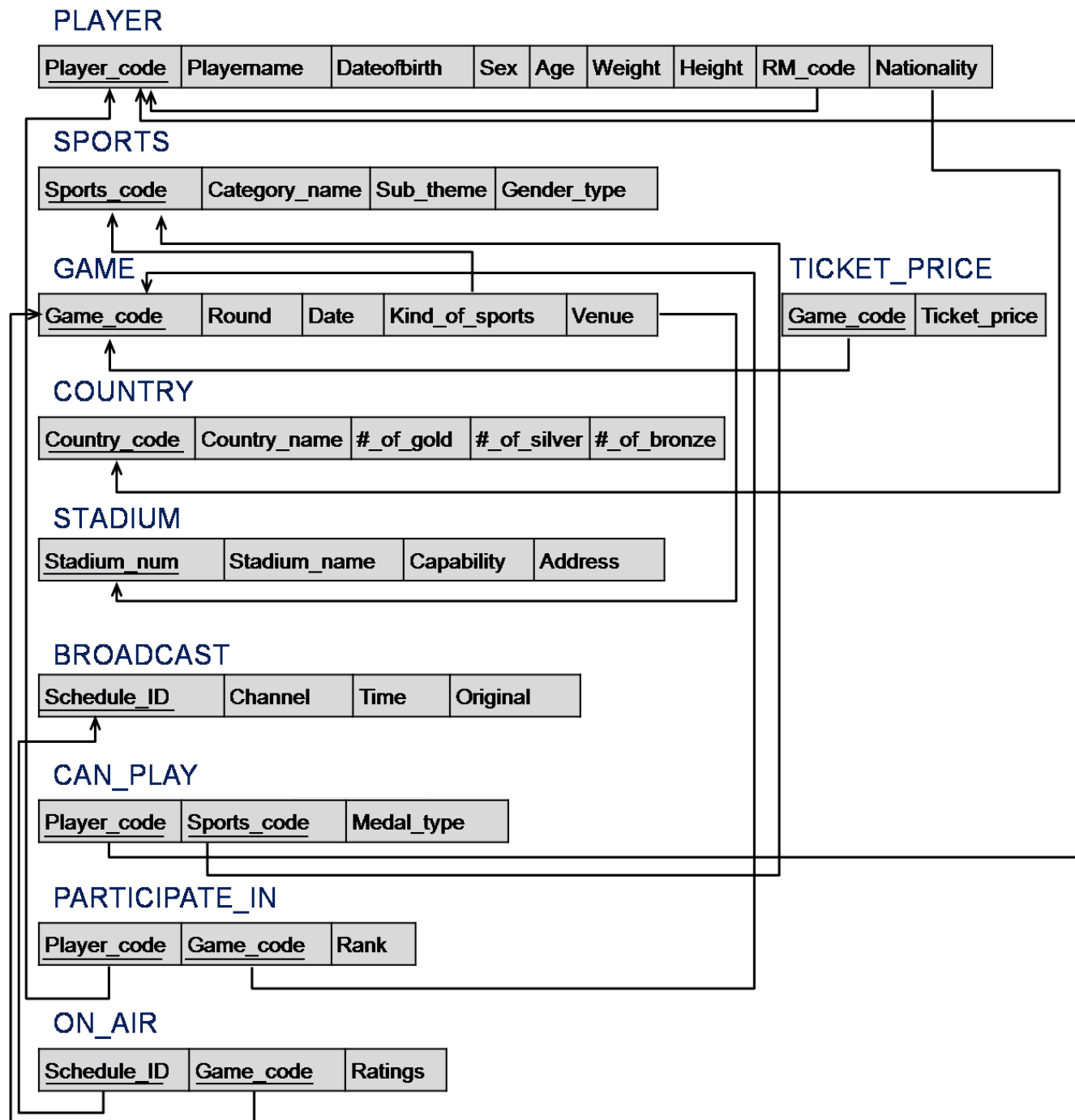


그림 2 Relation Schema

관계 스키마와 Constraints

-Domain Constraints, PK, FK, Integrity Constraints(Entity, Referential)

PK라 표시되어 있는 애트리뷰트는 각 릴레이션의 Primary Key이다. 어떠한 PK 값도 NULL이 될 수 없다. (Entity Integrity Constraints)

FK라 표시되어 있는 애트리뷰트들은 각 릴레이션의 Foreign key이며 다른 릴레이션의 Primary Key를 참조한다. 이 때 위 관계 스키마에서 화살표로 표현되는 부분이 참조되는 PK 애트리뷰트이며, FK의 튜플이 PK의 튜플을 참조할 때 반드시 참조되는 튜플이 그 릴레이션 내에 존재해야 한다.(Referential Integrity Constraints) 물론 참조하는 FK와 참조되는 PK의 도메인은 같아야 한다.

DB 전체적으로는 특정 애트리뷰트(Primary Keys)를 제외하고는 NULL값이 허용된다. (Null Constraints)

아래의 관계 스키마 설명에는 추가 설명과 예시를 통해 Domain Constraints와 Primary Key, Foreign Key 등을 나타내었다.

1.Entity types Relation Mapping

1. PLAYER

Player_code(PK) : 8자리의 정수

Player_name : 64자 미만 글자 (띄어쓰기 포함)

Date_of_birth : YY/MM/DD

Sex : M or F

Age(Derived) : 1~100

Weight : 1~255. 단위 kg

Height : 1~255. 단위 cm

RM_Code(FK) : 룸메이트의 Player_Code. 다른 Player의 Player-Code를 참조한다. 참조되는 어트리뷰트와 같은 도메인을 가진다.(이하 FK-PK관계에서 동일) Null값을 가질 수 있다.

Nationality(FK) : 선수가 소속한 국가. Country의 Country_code을 참조한다.

(Player_code(PK), Player_name, Date_of_birth, Sex, Age(Derived), Weight, Height, RM_Code, Nationality)

(00320124, Park Tae Hwan, 89/09/27, M, 24, 78, 183, 00320789, ROK)

2. SPORTS

Sports_code(PK) : 4자리 정수

Category_name : 32자 미만 글자 ex) Swimming, Archery

Sub_theme : 32자 미만 글자 ex) 100m Male Freestyle

Gender_type : Male / Female / Mix

(Sports_Code, Category_name, Sub_theme, Gender_type)

(2314, Swimming, 100m Male Freestyle, Male)

3. GAME

Game_code(PK) : 8자리 정수

Round : 2자리 정수

Date : TIME (현재 시간, 날짜 표기)

Kind_of_sports(FK) : 게임의 종목으로 SPORTS의 Sports_code를 참조한다.

Venue(FK) : 특정 경기가 열리는 장소로 STADIUM의 Stadium_num를 참조한다.

(Game_code, Round, Date, Kind_of_sports, Venue)

(40012792, 02, 14/09/29 1830EST 9/24/2014, 0014, 01)

4. COUNTRY

Country_code(PK) : 4자리 문자

Country_name : 64자 미만 글자

#_of_gold : 4자리 정수

#_of_silver : 4자리 정수

#_of_bronze : 4자리 정수

(Country_code, Country_name, #_of_gold, #_of_silver, #_of_bronze)

(ROK, Republic of Korea, 47, 53, 63)

Country_code는 4자리 문자로 각 나라의 고유한 Abbreviation을 나타낸다. (e.g. 한국 : ROK, 북한 : PRK) Country_name이 직접 키가 될 수 있지만, Primary Key로 Country_code를 사용한 이유는 자주 참조되는 PK가 가벼워야 더 효율적이기 때문이다. PK로 최대 몇 십 자가 될 수도 있는 국가 이름을 넣으면 redundancy가 심할 것이다.

5. STADIUM

Stadium_num(PK) : 4자리 정수

Stadium_name : 32자 미만 글자

Capability : 8자리 정수

Address : 255자 미만 글자

(Stadium_num, Stadium_name, Capability, Address)

(01, ParkTaeHwan Swim Stadium, 3000, 300 Asiagame-ro Song-do Incheon)

6. BROADCAST

Schedule_ID(PK) : -(hyphen)으로 이어진 16자리 정수

Channel : 8자 미만 글자

Time : TIME (현재 시간, 날짜 표기)

Original : -(hyphen)으로 이어진 16자리 정수

(Schedule-ID, Channel, Time, Original)

(00238756-2837592, KBS2, 2000 EST 9/26/2014, 00238756-2836432)

2. Relationship types Relation Mapping

1. CAN PLAY(PLOYER:SPORTS=M:N)

Player_code(FK)(PK의 한 요소) : 4자리 정수(참조되는 도메인과 같음)

Sports_code(FK)(PK의 한 요소) : 8자리 정수(참조되는 도메인과 같음. 이하 동문)

Medal-type : Gold, Silver, Bronze, Null

(Player_code, Sports_code, Medal_type)

(00320124, SW, 0014, , Bronze)

2. PARTICIPATE IN(PLOYER:GAME=M:N)

Player_code(FK)(PK의 한 요소)

Game_code(FK)(PK의 한 요소)

Rank : 2자리 정수

(Player_code, Game_code, Rank)

(00320124, 40012792, 1)

3. ON-AIR(BROADCAST:GAME=M:N)

Schedule_code(FK)(PK의 한 요소)

Game_code(FK)(PK의 한 요소)

Ratings : 소수점 한 자리까지 받는 실수 (단위는 %)

(Schedule_code, Game_code, Ratings)

(00238756-2837592, 40012792, 10.9)

3. Multi-valued Attribute Relation Mapping

1. TICKETPRICE

Game_code(FK)(PK의 한 요소)

Ticket_price(PK의 한 요소) : 4자리 정수 (단위 천 원)

다치 애트리뷰트 릴레이션에선 {Game_code, Ticket_price}의 조합이 이 릴레이션의 PK가 된다. (관계 타입 릴레이션도 같은 방식으로 작용한다.)

(Game_code, Ticket_price)

(40012792, 50)

이 때 잡시, Redundancy Issue들

1) 삼진 하나로 하지 않고 이진 관계 3개로 만든 것

GAME, SPORTS, PLAYER 엔티티 타입을 삼진 관계 하나로 할 법도 한데 우리는 이진 관계 세 개를 사용하였다. 이 세 엔티티 타입을 어떻게 관계로 연결할 지 고민할 때 가장 큰 주안점은 수업시간에 이진관계를 중점적으로 다뤘기 때문에 이를 최대한 활용하는 것이었다. '(R4)이진 관계 Cardinality (1:1/1:N/M:N)이 모두 나타나야 한다' 는 Requirement가 있는 것도 중요했다. 그런데 이 때 (R4)를 만족시키면 Redundancy에 대한 Requirement와 충돌할 수도 있다는 생각이 들었다.

그러나 Redundancy에 치명적인 문제점까진 아니고, 또한 다른 엔티티들과의 관계 등을 생각했을 때, 그리고 데이터의 조회 및 통계 시나리오를 생각했을 때 이진 관계 3개로 해도 충분하다고 결론내렸다.

2) 하나의 릴레이션십만 갖고 있는 엔티티

엔티티가 관계를 하나만 가지면(1:N) Redundancy 방지를 위해 다른 엔티티 타입의 어트리뷰트로 뺄 수도 있을지 모른다. (e.g. STADIUM을 Game의 애트리뷰트로 넣는 것) 하지만 이는 우리가 하려는 데이터 조회/분석 시나리오를 생각했을 때 여러모로 빼는 것이 나았고, 경기장 등의 엔티티들은 릴레이션을 따로 두어 관리하는 것이 아시안 게임 운영상 더 좋은 방안으로 생각하여 따로 엔티티 타입으로 선정했다.

3) 통계 분석이나 조회 할 때 반복을 막기 위한 애트리뷰트 추가

예를 들어 COUNTRY의 금메달수, 은메달수, 동메달수는 굳이 하자면 CAN PLAY 관계 타입의 Medal-Type 어트리뷰트를 이용해 선수들과 엮은 후 선수의 국적 별로 통계를 내는 쿼리를 만들어서 표현할 수 있다. 그러나 이런 메달 통계는 워낙 많이 쓰이고 위의 쿼리를 계속 반복할 바에야 COUNTRY에 메달 갯수를 저장해 두는 것이 나을 것이다. PLAYER의 Age 등도 마찬가지이다.

이상으로 구성 요소 설명부터 스키마 설명을 거쳐 그리고 Redundancy Issue까지 데이터 베이스의 Redundancy를 최소화하는 방향으로 설계하려는 요구 조건인 (R7)을 만족한다.

데이터 베이스 서치 & 통계 시나리오

I. SEARCH Scenario (사용하는 엔티티, 어트리뷰트 따라 가능한 조회 를 5가지)

1. 종목별로 우승한 선수들의 연령들을 각자 다 구해보자

1. 먼저 종목별로 메달현황이 나오는 CAN_PLAY 릴레이션에서 Medal_type이 Gold인 튜플들을 뽑아서 플레이어 코드와 종목 코드는 받아온다.

$$CHAMPION \leftarrow \pi_{Player_code, Sports_code}(\sigma_{(Medal_type=Gold)}(CAN_PLAY))$$

2. 그리고 선수들의 연령을 받아오기 위해 PLAYER 릴레이션에서 Player_code와 Age 어트리뷰트만 Select 해온다.

$$PLAYER_AGE \leftarrow \pi_{Player_code, Age}(PLAYER)$$

3. 생성된 두 릴레이션에서 Player_code를 이용해 natural join을 실시한 후 종목코드와 나이만 남긴다.

$$PLAYER_CHAM \leftarrow \pi_{Sports_code, Age}(PLAYER_AGE * CHAMPION)$$

4. 종목별 우승자의 나이가 나타난다.

2. 종목별 전체 참가선수들의 평균나이를 알고 싶다.

1. 종목별로 전체 참여 선수들의 코드를 받아온다.

$$LIST_SP_PL \leftarrow \pi_{Sports_code, Player_code}(CAN_PLAY)$$

2. 전체 선수 릴레이션에서 Player_code와 Age만 남긴다

$$PLAYER_AGE \leftarrow \pi_{Player_code, Age}(PLAYER)$$

3. 생성된 두 릴레이션에서 Player_code를 이용해 natural join을 실시한 후 종목코드를 기준으로 플레이어 수와 나이의 평균을 구한다.

$$SPORTS_AGE \leftarrow \rho_{(Scode, Participate, Average_age)}$$

$$(Sports_code \text{ } \mathfrak{Z} \text{ } COUNT \text{ } Player_code, AVERAGE \text{ } Age (LIST_SP_PL * PLAYER_AGE))$$

4. 마지막으로 생선 된 SPORTS_AGE 릴레이션에 종목별 참가자 수와 평균 나이가 나타나게 된다.

3. 특정 선수가 참여한 경기가 방송되는 방송의 시청률

1. 특정 선수가 출전한 경기들을 모두 뽑아낸다.(특정 선수의 Player_code가 28001501이라고 가정한다.)

$GAME_28001501 \leftarrow \pi_{Game_code} (\sigma_{(Player_code=28001501)}(PARTICIPATE_IN))$

2. ON_AIR 릴레이션과 위의 결과 릴레이션을 Game_code를 이용해 natural join을 실시한다.

$BROADCAST_28001501 \leftarrow \pi_{Schedule_id, Ratings} (ON_AIR * GAME_28001501)$

3. 결과로 나타난 릴레이션의 Ratings 어트리뷰트가 우리가 원하던 결과이다.

4. 각 국가별로 출전한 선수들이 전체 경기에서 거둔 등수의 평균을 알아서 국가끼리 비교를 해보고 싶다.

1. 먼저 모든 경기에 참가한 모든 선수들 리스트에 국적 부분을 join 시켜준다.

$EVERY_GAME \leftarrow PARTICIPATE_IN * (\pi_{Player_code, Nationality}(PLAYER))$

2. EVERY_GAME에는 모든 선수가 참여한 모든 경기가 다 나타난다. 따라서 국가별로 총 참여횟수를 세고 Rank 어트리뷰트의 평균을 구한다.

$COUNTRY_EVERYRANK \leftarrow \rho_{(Country, Number_of_participate, Average_rank)}$

$(Nationality \Join COUNT Player_code, AVERAGE Rank(EVERY_GAME))$

3. 위의 COUNTRY_EVERYRANK 릴레이션에 나타나는 결과가 국가별 참가 선수들의 모든 참여에서 각 경기별 얻은 순위의 평균과 총 참가횟수이다.

5. 각 선수 별로 메달 획득 현황을 알고싶다.

1. 먼저 모든 종목별 현황에서 각 메달을 받은 선수의 튜플만 걸러낸다.

$$GOLD_MEDAL \leftarrow \sigma_{(Medal_type=Gold)}(CAN_PLAY)$$

$$SILVER_MEDAL \leftarrow \sigma_{(Medal_type=Silver)}(CAN_PLAY)$$

$$BRONZE_MEDAL \leftarrow \sigma_{(Medal_type=Bronze)}(CAN_PLAY)$$

2. 각 메달에서 선수를 기준으로 메달의 개수를 카운트한다.

$$NUM_GOLD \leftarrow \rho_{(Pcode,Gold)}(Player_code \Join COUNT Sports_code(GOLD_MEDAL))$$

$$NUM_SILVER \leftarrow \rho_{(Pcode,Silver)}$$

$$(Player_code \Join COUNT Sports_code(SILVER_MEDAL))$$

$$NUM_BRONZE \leftarrow \rho_{(Pcode,Bronze)}$$

$$(Player_code \Join COUNT Sports_code(BRONZE_MEDAL))$$

3. 위에서 나온 세 릴레이션을 Player_code를 이용하여 natural join 한다.

$$NUM_MEDAL \leftarrow$$

$$GOLD_MEDAL * SILVER_MEDAL * BRONZE_MEDAL$$

4. 마지막으로 나온 결과값이 선수 별 메달 획득 현황이 된다.

II. STATISTICS Scenario(사용하는 엔티티, 어트리뷰트 따라 가능한 분석 틀 3가지)

1. 우승자의 연령과 각 종목별 참가자 평균 연령을 비교해 연령과 우승 여부가 관계가 있는지 파악해 볼 수 있을 것이다. 위의 Search Scenario에서 1번과 2번의 결과를 이용하여 1번의 평균과 2번의 평균이 유의미한 차이를 가지는지 알아볼 수 있다.
2. 특정 선수(예: 박태환)이 참여하는 경기와 이외의 경기가 방송에 방영될 때 시청률의 유의미한 차이가 있는지 알아 볼 수 있다. 위의 Search Scenario 3번의 결과와 ON_AIR 릴레이션의 Rating 값을 이용하여 두 숫자 간의 유의미한 차이가 있는지 비교해 볼 수 있다.
3. 국가 별로 금메달의 개수와 국가에 속한 선수들의 개인 별 성적이 얼마나 상관이 있는지 알아볼 수 있을 것이다. 한 나라에서 금메달을 많이 받는 것이 단순히 선수를 많이 출전

시켜서 그런 것인지 혹은 개개인이 능력이 좋은 선수들이 출전했기에 많이 받을 수 있었던 것인지 알아보는 것이 이것의 목표이다. 위의 Search Scenario 4번의 결과와 국가별 금메달 획득 순위를 각 국가별로 순서쌍으로 주었을 때 두 값이 유의미한 상관관계를 가지는지 유의성 검정을 해보면 된다.

이상으로 데이터 베이스를 조회하고, 이를 이용해 어떤 통계를 구하고 분석하는 시나리오를 제시하였다. 그러므로 (R8)과 (R9)의 조건을 만족한다.