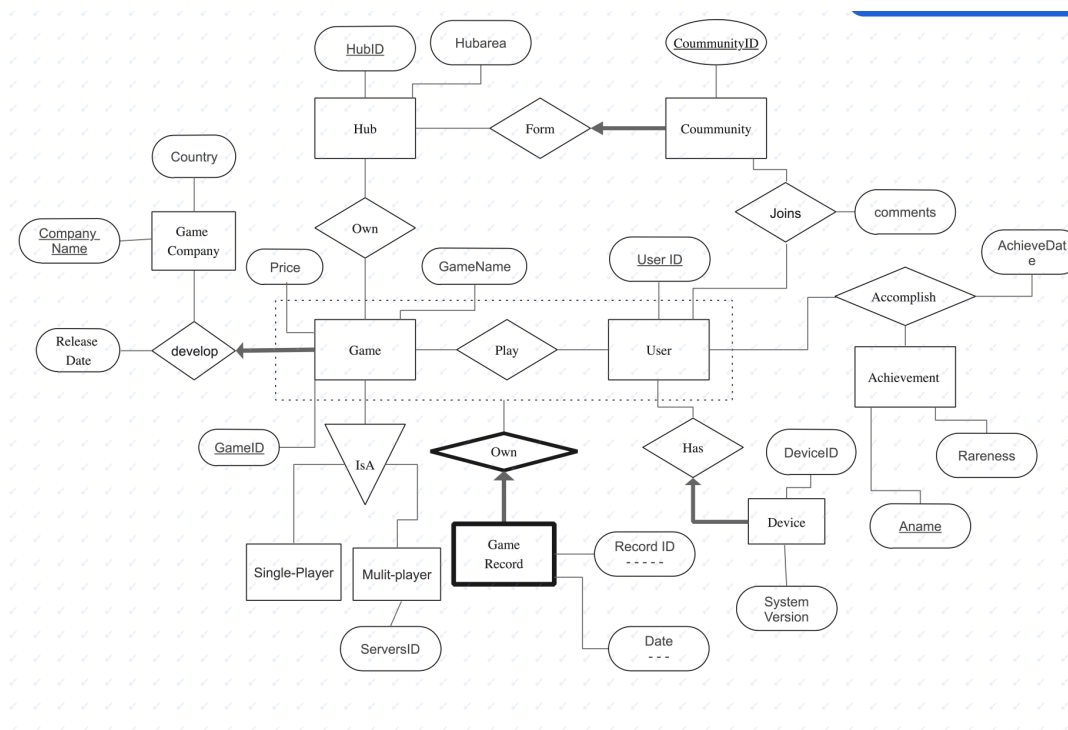


CPSC304 Project

ER diagram



Modification:

Change Platform attribute to Device and move the relation to User.

Add Game company attribute.

Change some cardinalities of relations(e.g. many Hubs can owns many games)

Add some participation constraints. (release date, AchieveDate)

Relational Schema

PKs, CKs, and **foreign keys**

GameCompany(Company Name, country)

Game(GameID, price, releaseDate, gameName, **Company Name**), companyname cannot be NULL

SinglePlayerGame(GameID)

MultitplayerGame(GameID, serverID)

Own(GameID, HubID)

Hub(HubID, hubarea)

Community(CommunityID, **HubID**) hubID cannot be NULL

User(UserID)

Join(UserID, CommunityID, comments)

Acheivement(AchievementName, rareness)

Acomplishes(UserID, AchievementName, AchieveDate)

Device(DeviceID, systemVersion, UserID), userID cannot be NULL

GameRecord(GameID, UserID, RecordID, Date)

Functional Dependency:

CompanyName -> country

GameID -> price, releaseDate, gameName, CompanyName

GameID -> serverID

HubID -> hubarea

CommunityID -> HubID

UserID, CommunityID -> comments

AchievementName -> rareness

UserID, AchievementName -> AchieveDate

DeviceID -> systemVersion, UserID

GameID, UserID -> RecordID, Date

Normalize:

CompanyName -> country

GameID -> price, releaseDate, gameName, CompanyName

DELETE GameID -> serverID because there maybe many servers for one game

HubID -> hubarea

CommunityID -> HubID

UserID, CommunityID, **commentid** -> comments **add comment id as primary key**

AchievementName -> rareness

UserID, AchievementName -> AchieveDate

DeviceID -> systemVersion, UserID

GameID, UserID, RecordID -> Date add recordid to primary key

List Of Table

PKs, CKs and **foreign keys**

GameCompany(CompanyName, country)

Game(GameID, price, releaseDate, gameName, **CompanyName**), companyname cannot be NULL

MultoplayerGame(GameID, serverID)

Own(GameID, HubID)

Hub(HubID, hubarea)

Community(CommunityID, **HubID**) hubID cannot be NULL

User(UserID)

User-Community(Commentid, UserID, CommunityID, comments)

Acheivement(AchievementName, rareness)

User-Acheivement(UserID, AchievementName, AchieveDate)

Device(DeviceID, systemVersion, UserID), userID cannot be NULL

GameRecord(GameID, UserID, RecordID, Date)

GameCompany	
Attribute	Description

CName(Primary Key)	Unique name of game company
Country	Country of game company

Game	
Attribute	Description
GameID(Primary Key)	Unique ID of game
gameName	name of game
price	price of name
releaseDate	releaseDate of game
Muti-player	ISA relation of game
serverID(Primary Key)	servers involved in the muti-player games
CompanyName(foreign key)	name of game company that develops the game

Own	
Hubid(Primary Key, foreign key)	id of hub that sells the game
GameID(Primary Key, foreign key)	ID of game

Hub	
HubID(Primary Key)	Unique id of hub that sells the game
HubArea	area of hub

Community	
Attributes	Description
CommunityID(Primary Key)	unique id of communities
HubID(foreign key)	Unique id of hub that creates the community(cannot be null)

User	
UserID(Primary Key)	Unique id of users

User-Community	
Attributes	Description

CommunityID(PK, foreign key)	id of communities
UserID(Primary Key, foreign key)	id of users
CommentID(Primary Key)	id of comments
Comments	content of comments

Acheivement	
Attributes	Description
AchievementName(Primary Key)	user has some achievements during playing games
rareness	rareness of achievements

User-Acheivement	
Attributes	Description
UserID(PK, foreign key)	the unique ID for each user (cannot be null)
AchievementName(PK, foreign key)	user has some achievements during playing games
AchieveDate	the date the user complete the achievement

Device	
Attribute	Description
DeviceID (Primay Key)	the ID of different devices
systemVersion	different versions of different systems
UserID(foreign key)	the unique ID for each user (cannot be null)

GameRecord	
Attribute	Description
RecordID(Partial Primary Key)	ID of each record
RecordDate	Date of each record
GameID(Partial Primary Key, foreignkey)	Game associated with this record
UserID(Partial Primary Key, foreign key)	User associated with this record

SQL DDL & CreateTable:

```
CREATE TABLE GameCompany {

    companyname    CHAR(50),
    country        CHAR(50),
    PRIMARY KEY (companyname )
}
```

CompanyName	Country
EA	U.S
Ubisoft	Canada
CD project	Poland
Nintendo	Japan
Sony	Japan

```
CREATE TABLE Game {
    gameId    INT,
    price     FLOAT,
    releaseDate DATE,
    gameName CHAR(50),
    companyname CHAR(50),
    PRIMARY KEY (gameID)
    FOREIGN KEY(companyName) REFERENCES GameCompany(companyname)
    ON DELETE SET NULL ON UPDATE CASCADE,
}
```

GameID	price	releaseDate	company	GameName
01	39.99	01/01/2014	CD project	The Witcher 3
02	49.99	01/01/2014	EA	The sims
03	59.99	01/01/2014	Ubisoft	Watch dogs
04	69.99	01/01/2014	Nintendo	Animal crossing
05	10.99	01/01/2014	Sony	Bloodborne

```
CREATE TABLE MultiPlayerGame {
GameID, serverID
    gameId    INT,
    serverID  INT,
    PRIMARY KEY (gameID, serverID)
    FOREIGN KEY(gameID) REFERENCES Game(gameID)
    ON DELETE SET NULL ON UPDATE CASCADE,
}
```

--	--

GameID	ServerID
01	01
01	02
01	03
01	04
02	01

```
CREATE TABLE Hub {
    hubid    INT,
    hubarea  CHAR(50),
    PRIMARY KEY (hubid),
}
```

hubid	hubarea
01	Canada
02	U.S
03	Germany
04	China
05	Japan

```
CREATE TABLE own{
    gameId    INT,
    hubID     INT,
    PRIMARY KEY (gameid,hubid),
    FOREIGN KEY(gameID) REFERENCES Game(gameID)
        ON DELETE SET NULL ON UPDATE CASCADE,
    FOREIGN KEY(hubID) REFERENCES Hub(hubID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}
```

GameID	hubiD
01	02
01	03
01	04
02	04
01	05

```
CREATE TABLE User {
    UseriD    INT,
    PRIMARY KEY (UseriD),
}
```

UserID
01

02
03
04
05

```
CREATE TABLE Community {
    communityID    INT,
    hubID          INT,
    PRIMARY KEY (communityID),
    FOREIGN KEY(hubID) REFERENCES Hub(hubID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}
```

CommunityID	HubID
01	01
02	01
03	01
06	02
05	01

```
CREATE TABLE User-Community {
    communityID    INT,
    UserID        INT,
    commentid     INT,
    comments      Char(1000),
    PRIMARY KEY (commentid, userid, communityID),
    FOREIGN KEY(UserID) REFERENCES User(UserID)
        ON DELETE SET NULL ON UPDATE CASCADE,
    FOREIGN KEY(communityID) REFERENCES Hub(communityID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}
```

communityID	UserID	commentid	comments
01	1	01	great
02	3	02	awesome
01	1	01	fine
04	4	01	bad
01	5	01	awful

```
CREATE TABLE Acheivement {
    Acheivename    CHAR(50),
```

```

    rareness      INT,
    PRIMARY KEY (Acheivename),
}

```

AchievementName	rareness
Never stop 1	1
Never stop 3	5
Never stop 4	3
Never stop 2	2
Never stop 5	10

```

CREATE TABLE User-Acheivement {
    Acheivename    CHAR(50),
    Userid         INT,
    AchieveDate    DATE,
    PRIMARY KEY (Acheivename,userid)
    FOREIGN KEY(Acheivename) REFERENCES User(UserID)
        ON DELETE SET NULL ON UPDATE CASCADE,
    FOREIGN KEY(UserID) REFERENCES User(UserID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}

```

Acheivename	UserID	AchieveDate
Never stop 2	01	2020/03/03
Never stop 1	01	2020/03/03
Never stop 4	01	2020/03/05
Never stop 1	02	2020/03/03
Never stop 1	03	2020/03/03

```

CREATE TABLE Device {
    deviceID      INT,
    systemVersion    FLOAT,
    UserID        INT,
    PRIMARY KEY (deviceID),
    FOREIGN KEY(UserID) REFERENCES User(UserID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}

```

deviceID	systemVersion	UserID
01	1.0	01
02	2.0	01
03	3.0	01

04	4.0	01
05	1.0	02

```
CREATE TABLE GameRecord{
    gameId    INT,
    userID    INT,
    recordID   INT,
    recorddate    DATE,
    PRIMARY KEY (gameID, userID, recordID)
    FOREIGN KEY(gameID) REFERENCES Game(gameID)
        ON DELETE SET NULL ON UPDATE CASCADE,
    FOREIGN KEY(userID) REFERENCES User(userID)
        ON DELETE SET NULL ON UPDATE CASCADE,
}
```

GameID	UserID	RecordID	RecordDate
01	01	01	2020/03/03
01	01	02	2020/03/03
01	01	03	2020/03/05
01	02	04	2020/03/03
01	03	02	2020/03/03

Queries

Insertion: Add a new game record

Delete: Delete one comment of user from community

Update: Update the price of a game

Selection: Select all the games released in 2014

Division: find all games developed by one gamecompany