Database Management
Final Project

Michael Cummins
CMPT308N112p
Dr. Donald Schwartz
December 5, 2016

Table of Contents
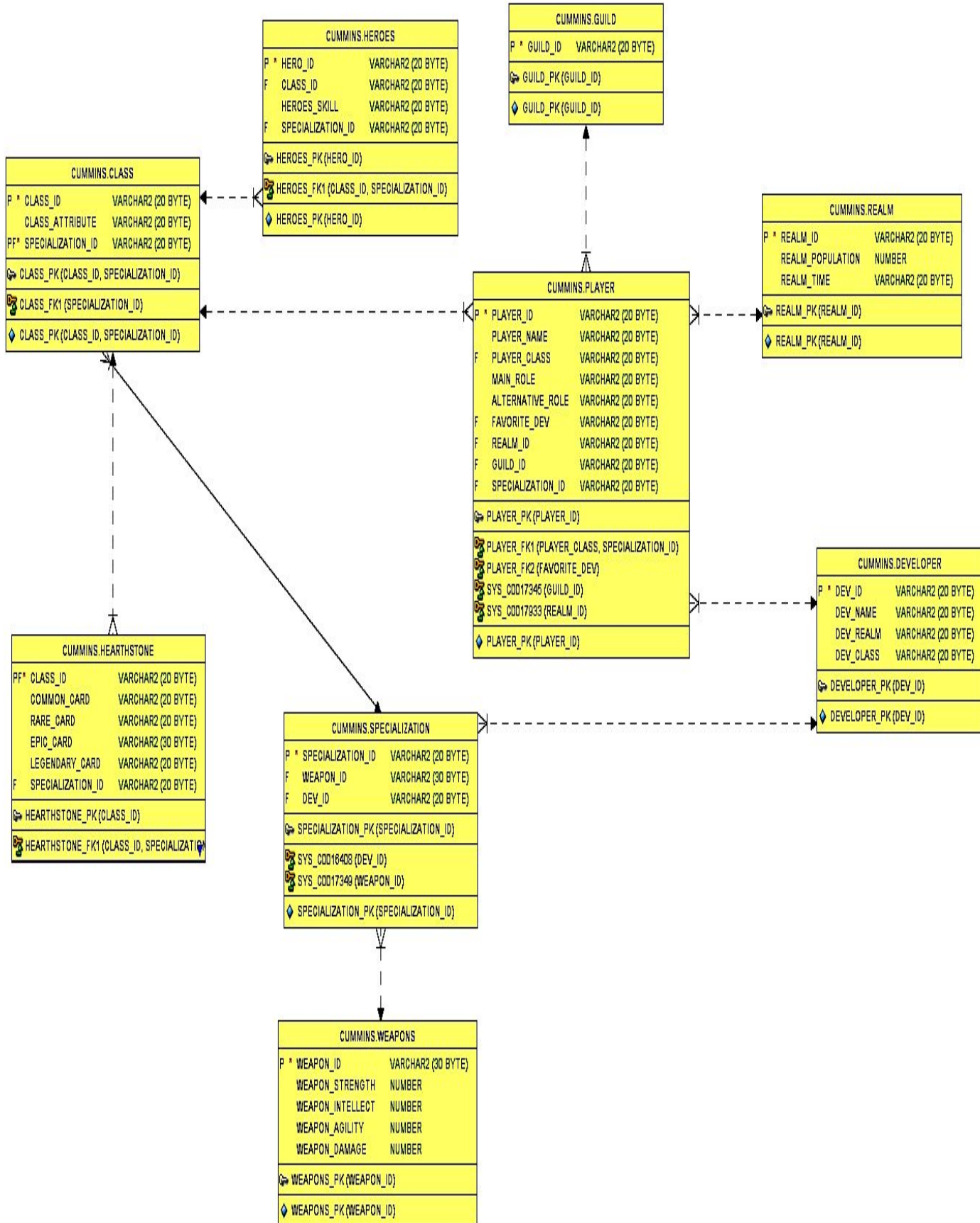
Blizzard Entertainment is in need of a system that tracks player info in many of their games including World of Warcraft, Hearthstone, and Heroes of the Storm. The system must be built to satisfy the following needs:

1. The company must track its players, including their in-game name, real life name, class, prefered main role, alternative role, guild ID, favorite developer, and realm.
2. Guilds are detailed by only their guild ID and their progression
3. Classes are identified by a class ID, specialization ID, and a main attribute.
4. Specializations are subclasses of classes that allow players to customize their character according to their playstyle. I.E three specializations of the Mage class are Arcane, Fire, and Frost. Specializations are defined by a specialization ID, main attribute, and a weapon ID.
5. Realms are defined by their name, population, and time zone.
6. Weapons are identifiable by their weapon ID, weapon damage, strength, intellect, and agility.
7. Blizzard also has an online collectible card game called Hearthstone where each playable character is identified by the class ID. The cards of the classes are identified by having one common card, one rare card, one epic card, one legendary card, and a specialization ID.
8. Blizzard has a fighting arena type game called Heroes of The Storm where heroes from other blizzard games fight to the death. Heroes are identified by their Hero ID, their class, their skill, and their specialization.
9. Developers of Blizzard's games are identified by their in game name, their real life name, what realm they play on, and the class that they work on.

The database must be able to satisfy the following queries:
1. Name Players that do not play the same class as any Hero
2. Give the classes and specializations developed by Watcher.
3. Name guilds that only have priest players
4. Name players who are on Stormrage and whose favorite developer plays a paladin or players that play the same class as a hero who uses a skill that is also an epic card in Hearthstone.
5. Name the identities of Michael Cummins and name the guild and favorite developer each identity is associated with. Cardinality
6. Identify heroes and their specialization
7. Name all players who do and do not have a favorite developer
8. Identify weapons that have more strength than intellect
9. Name realms that have a population greater than the agility of Felo'melorn
10. Name legendary cards that are heroes

CUMMINS.HEROES

P	* HERO_ID	VARCHAR2 (20 BYTE)
F	CLASS_ID	VARCHAR2 (20 BYTE)
	HEROES_SKILL	VARCHAR2 (20 BYTE)
F	SPECIALIZATION_ID	VARCHAR2 (20 BYTE)

HEROES_PK (HERO_ID)

HEROES_FK1 (CLASS_ID, SPECIALIZATION_ID)

HEROES_PK (HERO_ID)

CUMMINS.GUILD

P	* GUILD_ID	VARCHAR2 (20 BYTE)

GUILD_PK (GUILD_ID)

GUILD_PK (GUILD_ID)

CUMMINS.CLASS

P	* CLASS_ID	VARCHAR2 (20 BYTE)
	CLASS_ATTRIBUTE	VARCHAR2 (20 BYTE)
PF	* SPECIALIZATION_ID	VARCHAR2 (20 BYTE)

CLASS_PK (CLASS_ID, SPECIALIZATION_ID)

CLASS_FK1 (SPECIALIZATION_ID)

CLASS_PK (CLASS_ID, SPECIALIZATION_ID)

CUMMINS.REALM

P	* REALM_ID	VARCHAR2 (20 BYTE)
	REALM_POPULATION	NUMBER
	REALM_TIME	VARCHAR2 (20 BYTE)

REALM_PK (REALM_ID)

REALM_PK (REALM_ID)

CUMMINS.PLAYER

P	* PLAYER_ID	VARCHAR2 (20 BYTE)
	PLAYER_NAME	VARCHAR2 (20 BYTE)
F	PLAYER_CLASS	VARCHAR2 (20 BYTE)
	MAIN_ROLE	VARCHAR2 (20 BYTE)
	ALTERNATIVE_ROLE	VARCHAR2 (20 BYTE)
F	FAVORITE_DEV	VARCHAR2 (20 BYTE)
F	REALM_ID	VARCHAR2 (20 BYTE)
F	GUILD_ID	VARCHAR2 (20 BYTE)
F	SPECIALIZATION_ID	VARCHAR2 (20 BYTE)

PLAYER_PK (PLAYER_ID)

PLAYER_FK1 (PLAYER_CLASS, SPECIALIZATION_ID)
PLAYER_FK2 (FAVORITE_DEV)
SYS_C0017345 (GUILD_ID)
SYS_C0017933 (REALM_ID)

PLAYER_PK (PLAYER_ID)

CUMMINS.DEVELOPER

P	* DEV_ID	VARCHAR2 (20 BYTE)
	DEV_NAME	VARCHAR2 (20 BYTE)
	DEV_REALM	VARCHAR2 (20 BYTE)
	DEV_CLASS	VARCHAR2 (20 BYTE)

DEVELOPER_PK (DEV_ID)

DEVELOPER_PK (DEV_ID)

CUMMINS.HEARTHSTONE

PF	* CLASS_ID	VARCHAR2 (20 BYTE)
	COMMON_CARD	VARCHAR2 (20 BYTE)
	RARE_CARD	VARCHAR2 (20 BYTE)
	EPIC_CARD	VARCHAR2 (30 BYTE)
	LEGENDARY_CARD	VARCHAR2 (20 BYTE)
F	SPECIALIZATION_ID	VARCHAR2 (20 BYTE)

HEARTHSTONE_PK (CLASS_ID)

HEARTHSTONE_FK1 (CLASS_ID, SPECIALIZATION...)

CUMMINS.SPECIALIZATION

P	* SPECIALIZATION_ID	VARCHAR2 (20 BYTE)
F	WEAPON_ID	VARCHAR2 (30 BYTE)
F	DEV_ID	VARCHAR2 (20 BYTE)

SPECIALIZATION_PK (SPECIALIZATION_ID)

SYS_C0016408 (DEV_ID)
SYS_C0017349 (WEAPON_ID)

SPECIALIZATION_PK (SPECIALIZATION_ID)

CUMMINS.WEAPONS

P	* WEAPON_ID	VARCHAR2 (30 BYTE)
	WEAPON_STRENGTH	NUMBER
	WEAPON_INTELLECT	NUMBER
	WEAPON_AGILITY	NUMBER
	WEAPON_DAMAGE	NUMBER

WEAPONS_PK (WEAPON_ID)

WEAPONS_PK (WEAPON_ID)

This table stores the data for each class combination

Create Table Class
(
Class_ID   VARCHAR2(20) NOT NULL,
Class_attribute   VARCHAR2(20),
Specialization_ID   VARCHAR2(20) NOT NULL ,
PRIMARY KEY (CLASS_ID, SPECIALIZATION_ID)
CONSTRAINT fk_Class FOREIGN KEY (SPECIALIZATION_ID)
REFERENCES SPECIALIZATION(SPECIALIZATION_ID)
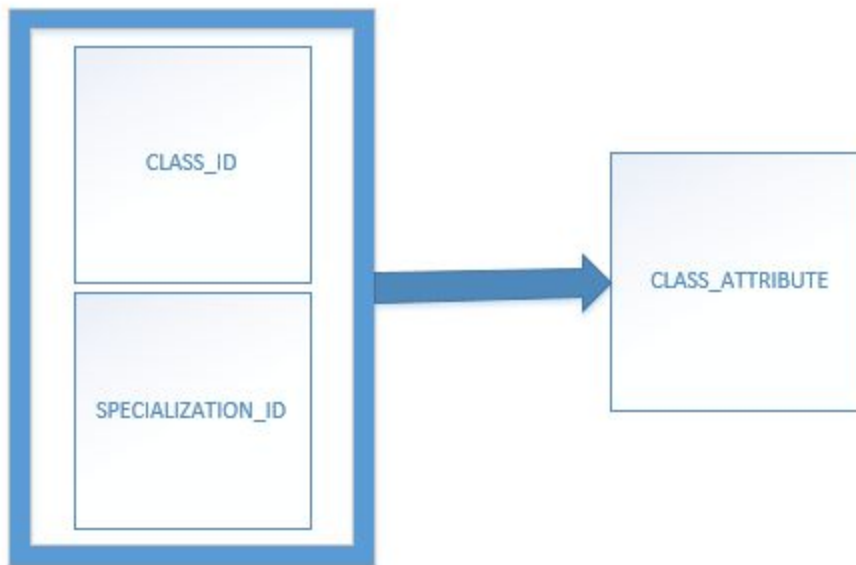);

1st NF - No repeating groups and has a key:
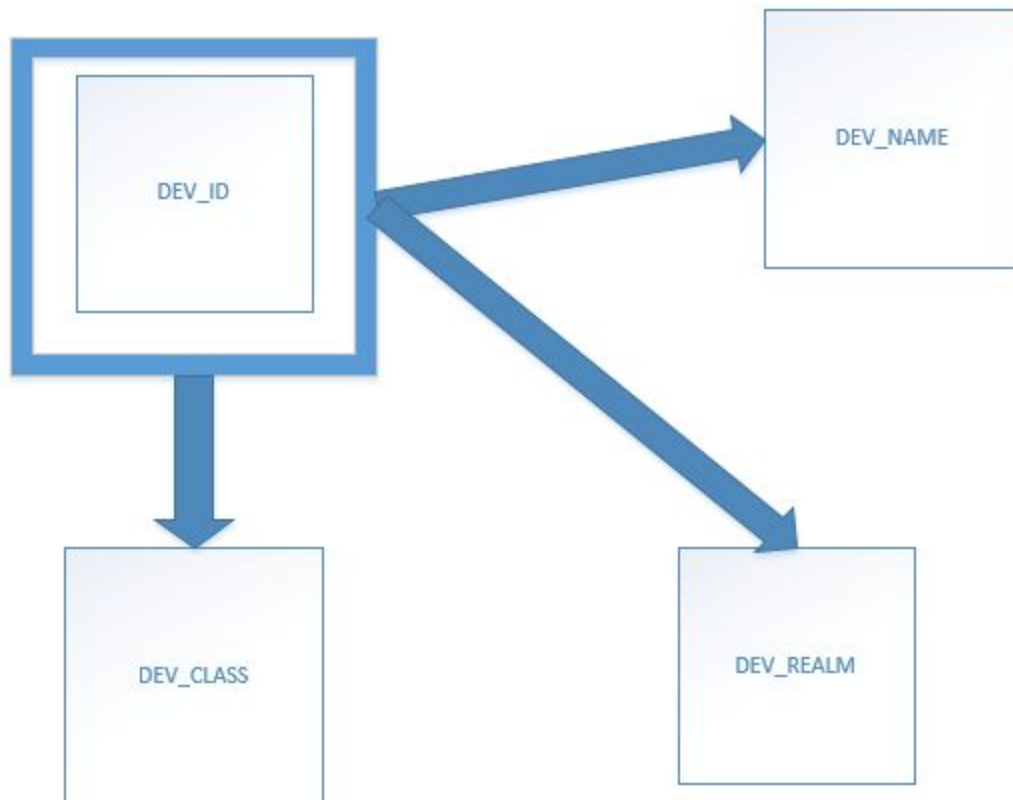- Repeating groups means that no single field can contain an array of repeating values
- The primary key is Class_ID, SPECIALIZATION_ID

2nd NF - Is in first NF and has no partial dependencies
- Both Class_ID and Specialization_ID determine Class_attribute

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value

This table stores the information for each developer
CREATE TABLE DEVELOPER
(
DEV_ID   VARCHAR2(20) NOT NULL,
DEV_NAME   VARCHAR2(20),
DEV_REALM   VARCHAR2(20),
DEV_CLASS   VARCHAR2(20),
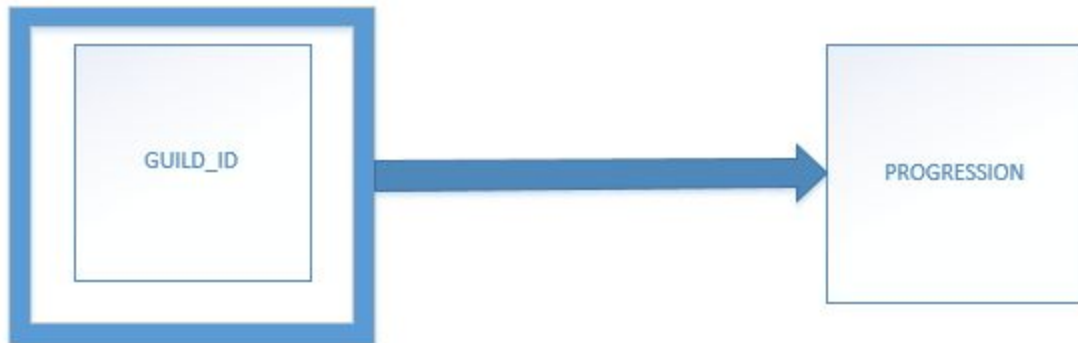PRIMARY KEY (DEV_ID)
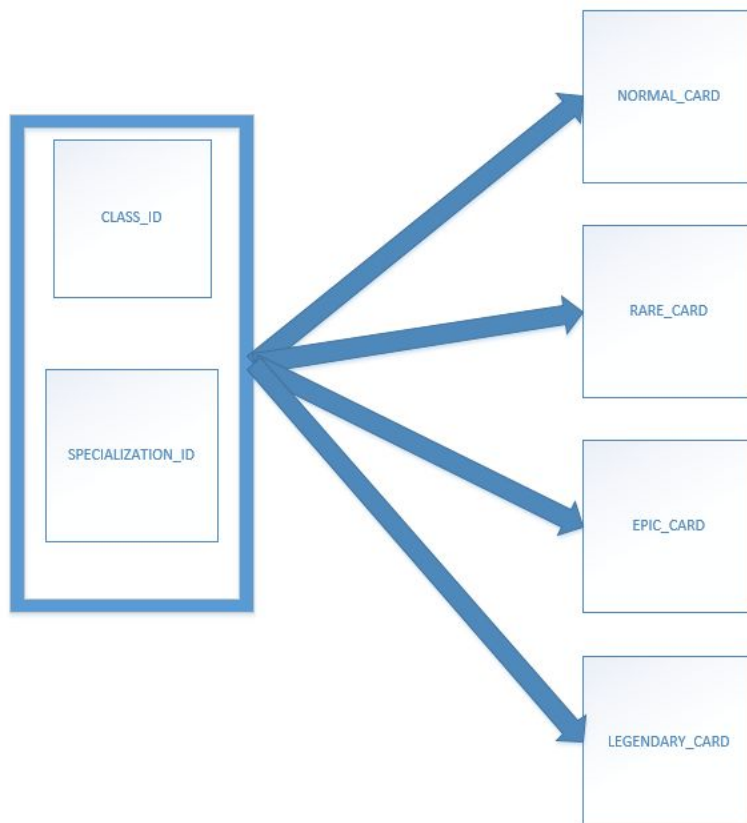);

1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is DEV_ID

2nd NF - Is in first NF and has no partial dependencies
- Both Class_ID and Specialization_ID determine Class_attribute

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value
This table stores the information for each developer

This table only stores guild names and their progression

```
CREATE TABLE GUILD
(
GUILD_ID            VARCHAR2(20) NOT NULL,
PROGRESSION         VARCHAR2(20),
PRIMARY KEY (GUILD_ID)
);
```

1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is GUILD_ID

2nd NF - Is in first NF and has no partial dependencies
- GUILD_ID determines PROGRESSION

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key values

This table tracks data for what cards are in Hearthstone
```
CREATE TABLE HEARTHSTONE
(
CLASS_ID                VARCHAR2(20) NOT NULL,
COMMON_CARD             VARCHAR2(20),
RARE_CARD               VARCHAR2(20),
EPIC_CARD               VARCHAR2(20),
LEGENDARY_CARD          VARCHAR2(20),
SPECIALIZATION_ID       VARCHAR2(20),
PRIMARY KEY (CLASS_ID)
CONSTRAINT fk_HEARTHSTONE FOREIGN KEY (CLASS_ID, SPECIALIZATION_ID)
REFERENCES CLASS (CLASS_ID,SPECIALIZATION_ID)
);
```
1st NF - No repeating groups and has a key:

- Repeating groups means that no single field can contain an array of repeating values
- The primary key is Class_ID

2nd NF - Is in first and has no partial dependencies

- Class_ID determines COMMON_CARD, RARE_CARD, EPIC_CARD, and LEGENDARY_CARD

3rd NF - Is in second and has no transitive dependencies

- No non primary key defines another non primary key value

This table is designed to store hero data from Heroes of The Storm
CREATE TABLE HEROES
(
HERO_ID   VARCHAR2(20) NOT NULL,
CLASS_ID   VARCHAR2(20),
HEROES_SKILL   VARCHAR2(20),
SPECIALIZATION_ID   VARCHAR2(20),
PRIMARY KEY (HERO_ID)
CONSTRAINT fk_heroes FOREIGN KEY (CLASS_ID, SPECIALIZATION_ID)
REFERENCES CLASS (CLASS_ID, SPECIALIZATION_ID)
);
1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is HERO_ID
2nd NF - Is in first NF and has no partial dependencies
- Both HERO_ID determines their skill
3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value
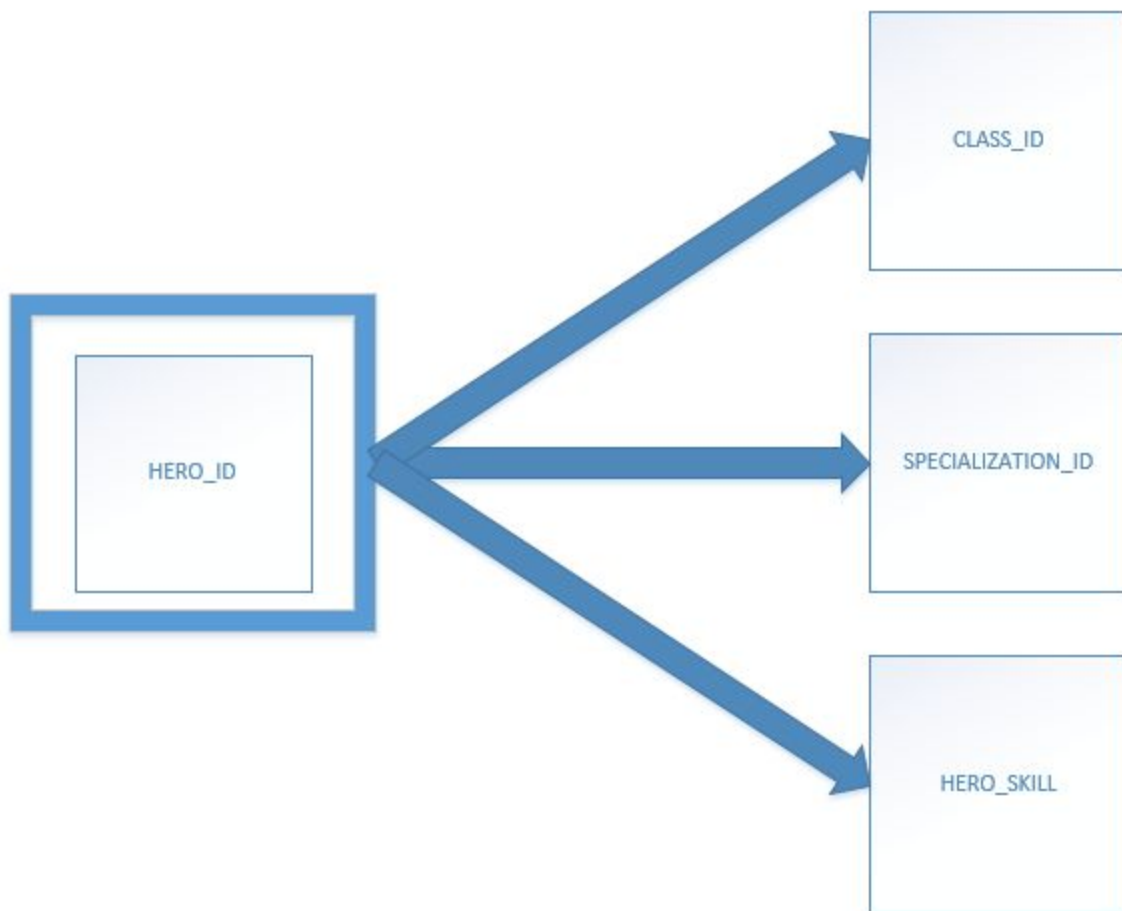
This table stores character data

```
CREATE TABLE PLAYER
(
PLAYER_ID               VARCHAR2(20) NOT NULL,
PLAYER_NAME             VARCHAR2(20),
PLAYER_CLASS            VARCHAR2(20),
MAIN_ROLE               VARCHAR2(20),
ALTERNATIVE_ROLE        VARCHAR2(20),
FAVORITE_DEV            VARCHAR2(20),
REALM_ID                VARCHAR2(20),
GUILD_ID                VARCHAR2(20),
SPECIALIZATION_ID       VARCHAR2(20),
PRIMARY KEY (PLAYER_ID)
CONSTRAINT fk_PLAYER FOREIGN KEY (PLAYER_CLASS, SPECIALIZATION_ID)
REFERENCES CLASS (PLAYER_CLASS, SPECIALIZATION_ID)
CONSTRAINT fk_PLAYER FOREIGN KEY (REALM_ID)
REFERENCES REALM (REALM_ID)
CONSTRAINT fk_PLAYER FOREIGN KEY (GUILD_ID)
REFERENCES GUILD(GUILD_ID)
CONSTRAINT fk_PLAYER FOREIGN KEY (FAVORITE_DEV)
REFERENCES DEVELOPER (DEV_ID)
)
```
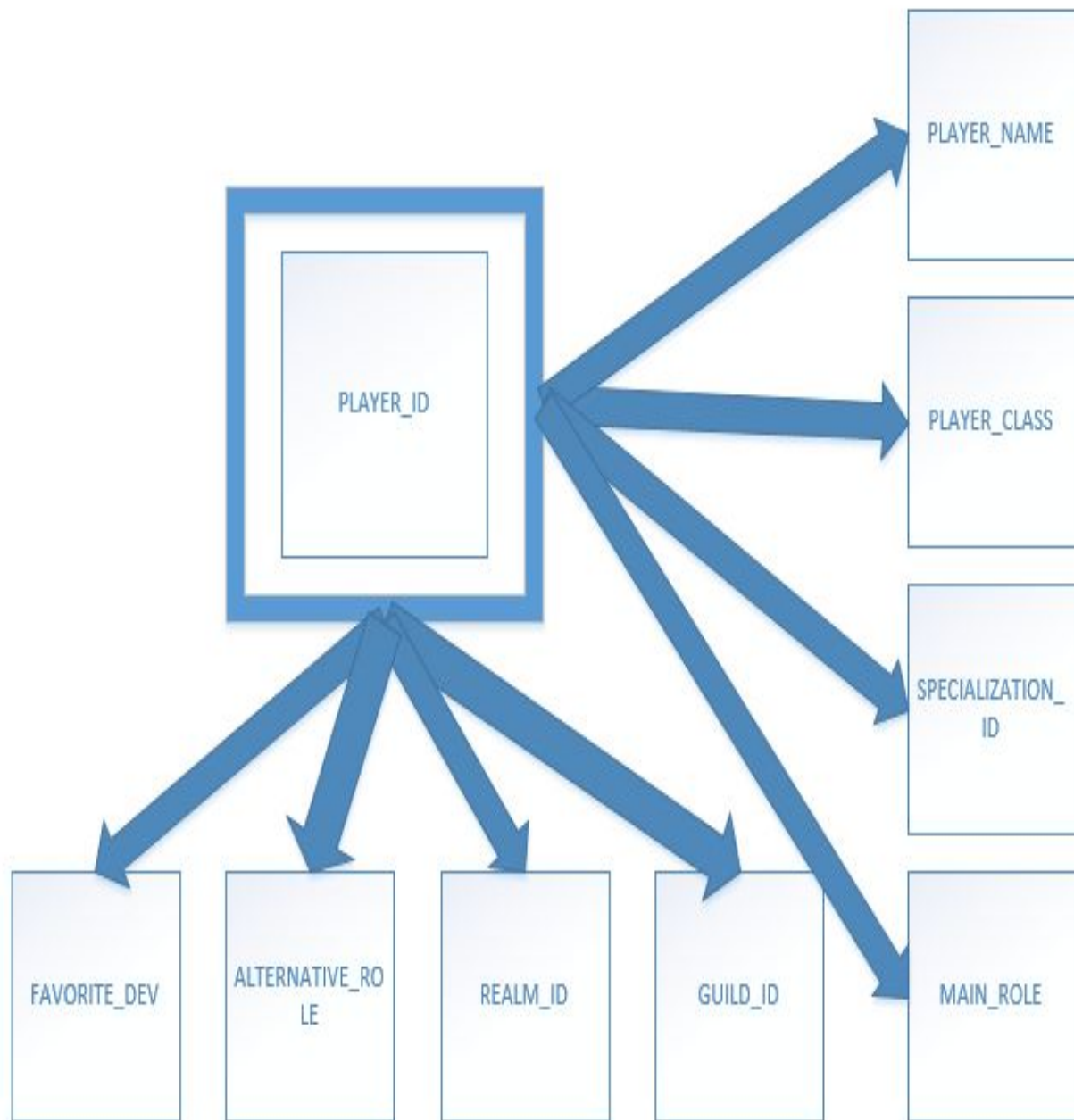
1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is PLAYER_ID

2nd NF - Is in first NF and has no partial dependencies
- PLAYER_ID determines PLAYER_NAME, PLAYER_CLASS, MAIN_ROLE, ALTERNATIVE_ROLE, GUILD_ID, REALM_ID, FAVORITE_DEV, and SPECIALIZATION_ID

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value

This table stores data pertaining to population and the time zone

```
CREATE TABLE REALM
(
REALM_ID                VARCHAR2(20) NOT NULL,
REALM_POPULATION        NUMBER,
REALM_TIME              VARCHAR2(20),
PRIMARY KEY (REALM_ID)
)
```
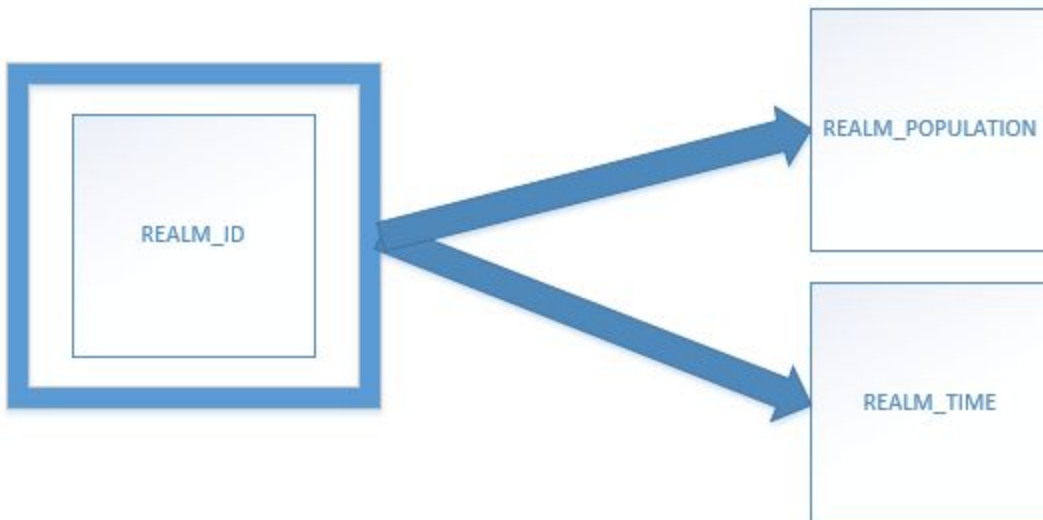
1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is REALM_ID

2nd NF - Is in first NF and has no partial dependencies
- REALM_ID determines REALM_POPULATION and REALM_TIME

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value

This table stores data for each specialization within the game

```
CREATE TABLE SPECIALIZATION
(
SPECIALIZATION_ID VARCHAR2(20) NOT NULL,
WEAPON_ID          VARCHAR2(20),
DEV_ID             VARCHAR2(20),
PRIMARY KEY (SPECIALIZATION_ID)
CONSTRAINT fk_SPECIALIZATION FOREIGN KEY (WEAPON_ID)
REFERENCES WEAPONS (WEAPON_ID)
CONSTRAINT fk_SPECIALIZATION FOREIGN KEY (DEV_ID)
REFERENCES DEVELOPER (DEV_ID)
)
```

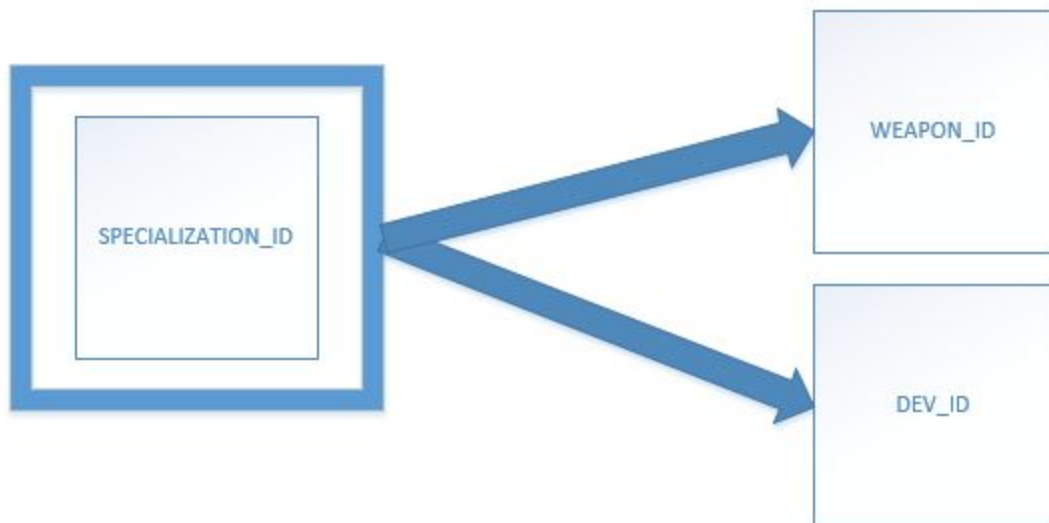1st NF - No repeating groups and has a key:
-   Repeating groups means that no single field can contain an array of repeating values
-   The primary key is SPECIALIZATION_ID

2nd NF - Is in first NF and has no partial dependencies
- SPECIALIZATION_ID determines WEAPON_ID and DEV_ID

3rd NF - Is in second NF and has no transitive dependencies
-   No non primary key defines another non primary key value

This table stores weapon data, specifically the amount of strength, intellect, agility, and damage a weapon has on it

CREATE TABLE WEAPONS
(
WEAPON_ID   VARCHAR2(30) NOT NULL,
WEAPON_STRENGTH   NUMBER,
WEAPON_INTELLECT   NUMBER,
WEAPON_AGILITY   NUMBER,
WEAPON_DAMAGE   NUMBER,
PRIMARY KEY (WEAPON_ID)
);

1st NF - No repeating groups and has a key:
- Repeating groups means that no single field can contain an array of repeating values
- The primary key is WEAPON_ID

2nd NF - Is in first NF and has no partial dependencies
- WEAPON_ID determines WEAPON_STRENGTH, WEAPON_INTELLECT, WEAPON_AGILITY, and WEAPON_DAMAGE

3rd NF - Is in second NF and has no transitive dependencies
- No non primary key defines another non primary key value
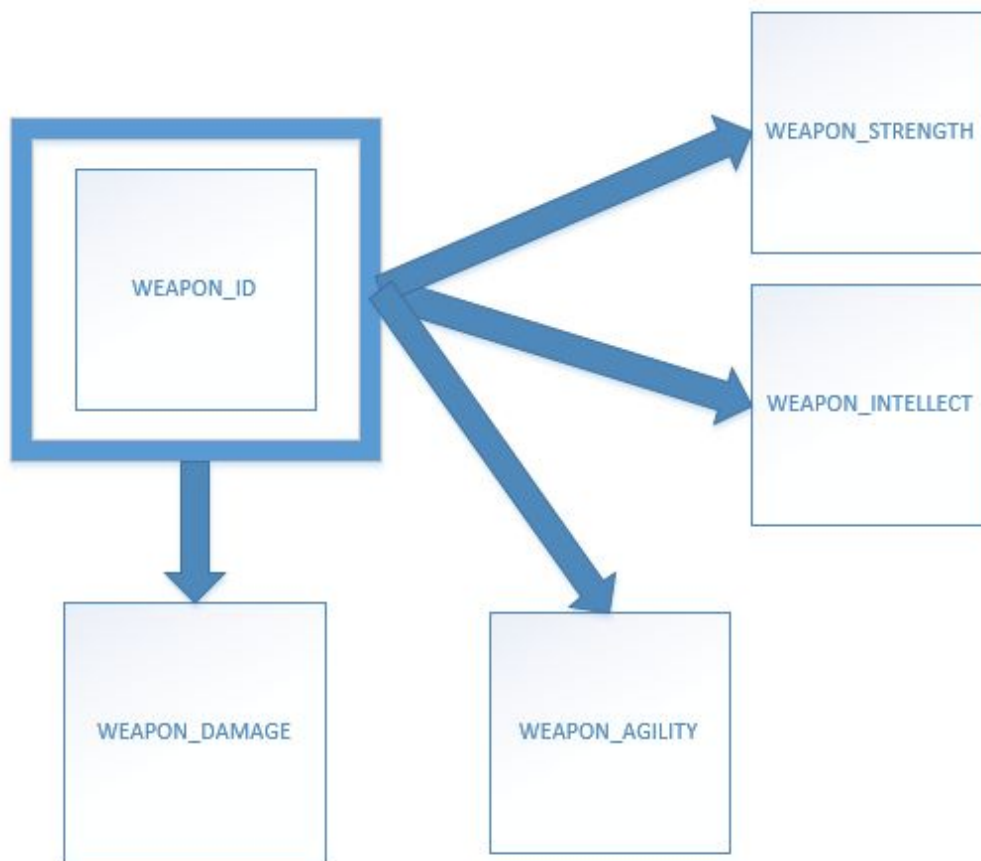
1) Give player IDs that do not play the same class as any Hero. Cardinality = 1

SELECT PLAYER_ID
FROM PLAYER
WHERE NOT EXISTS
(SELECT *
FROM HEROES
WHERE EXISTS
(SELECT *
FROM CLASS
WHERE PLAYER.PLAYER_CLASS = CLASS.CLASS_ID
AND CLASS.CLASS_ID = HEROES.CLASS_ID));

PLAYER ID
Piffweaver

2) Give every class and specialization combination developed by Watcher. Cardinality = 6

SELECT CLASS.CLASS_ID, CLASS.SPECIALIZATION_ID
FROM CLASS
WHERE EXISTS
(SELECT *
FROM SPECIALIZATION
WHERE SPECIALIZATION.SPECIALIZATION_ID = CLASS.SPECIALIZATION_ID
AND SPECIALIZATION.DEV_ID = 'Watcher');

CLASS ID	SPECIALIZATION ID
Mage	Arcane
Hunter	Beast Mastery
Mage	Fire
Priest	HolyPriest
Rogue	Outlaw
Shaman	RestorationShaman

3) Name guilds that only have priest players. Cardinality = 1

SELECT GUILD.GUILD_ID
FROM GUILD
WHERE NOT EXISTS
(SELECT *
FROM PLAYER
WHERE GUILD.GUILD_ID = PLAYER.GUILD_ID
AND PLAYER.PLAYER_CLASS <>'Priest');

GUILD_ID
Death Jesters

4) Name players who are on Stormrage and whose favorite developer plays a paladin or players that play the same class as a hero who uses a skill that is also an epic card in Hearthstone. Cardinality = 7

SELECT DISTINCT PLAYER.PLAYER_NAME
FROM REALM, PLAYER, CLASS, HEROES, HEARTHSTONE, DEVELOPER
WHERE (PLAYER.REALM_ID = REALM.REALM_ID
AND REALM.REALM_ID = 'Stormrage'
AND PLAYER.FAVORITE_DEV = DEVELOPER.DEV_ID
AND DEVELOPER.DEV_CLASS = 'Paladin')
OR (PLAYER.PLAYER_CLASS = CLASS.CLASS_ID
AND CLASS.CLASS_ID = HEROES.CLASS_ID
AND CLASS.CLASS_ID = HEARTHSTONE.CLASS_ID
AND HEARTHSTONE.EPIC_CARD = HEROES.HEROES_SKILL);

PLAYER_NAME
Rock Hayden
Rock Jenkins
Timo Gerhardt
Frederick Nicholas
Michael Cummins
Sam Smith
Lindon Roydon

5)Name the identities of Michael Cummins and name the guild and favorite developer each identity is associated with. Cardinality = 3

SELECT DISTINCT PLAYER.FAVORITE_DEV, PLAYER.PLAYER_ID,GUILD.GUILD_ID
FROM PLAYER FULL OUTER JOIN GUILD ON PLAYER.GUILD_ID=GUILD.GUILD_ID
WHERE PLAYER.PLAYER_NAME='Michael Cummins';

FAVORITE DEV	PLAYER ID	GUILD ID
Pelinel	Luggindy	Tesseract
	Larownie	
Lore	Baroxine	

6) Identify heroes and their specialization. Cardinality = 16

SELECT HEROES.HERO_ID, HEROES.SPECIALIZATION_ID
FROM HEROES LEFT JOIN SPECIALIZATION
ON HEROES.HERO_ID = SPECIALIZATION.SPECIALIZATION_ID;

HERO ID	SPECIALIZATION ID
Arthas	FrostDK
Putricide	Unholy
Gul'dan	
Valeera	Outlaw
Uther	HolyPaladin
Thrall	
Rehgar	RestorationShaman
Morales	HolyPriest
Jaina	
Khadgar	Arcane
Rexxar	Beast Mastery
Varian	Arms
Anduin	Discipline
Malfurion	RestorationDruid
Tyrande	
Chromie	Fire

7)Name all players and their favorite developer. Cardinality = 27

SELECT DISTINCT PLAYER.PLAYER_ID, DEVELOPER.DEV_ID
FROM DEVELOPER RIGHT JOIN PLAYER
ON PLAYER.FAVORITE_DEV = DEVELOPER.DEV_ID
ORDER BY DEVELOPER.DEV_ID;

8)Identify weapons that have more strength than intellect. Cardinality = 13

SELECT WEAPONS.WEAPON_ID
FROM WEAPONS
WHERE WEAPONS.WEAPON_STRENGTH > WEAPONS.WEAPON_DAMAGE;

WEAPON_ID
Twinblades of the Deceiver
Fangs of Ashamane
Thas'dorah
Talonclaw
Felo'melorn
Sheilun, Staff of the Mists
Fists of the Heavens
Light's Wrath
The Kingslayers
The fist of Ra-den
Ulthalesh
Scepter of Sargeras
Warswords of the Valarjar

9)Name realms that have a population greater than the agility of Felo'melorn. Cardinality = 3

SELECT REALM.REALM_ID
FROM REALM
WHERE REALM.REALM_POPULATION >
(SELECT WEAPONS.WEAPON_AGILITY
FROM WEAPONS
WHERE WEAPONS.WEAPON_ID = 'Felo''melorn');

REALM ID
Stormrage
Ysera
Barthilas

10)Name legendary cards that are heroes . Cardinality = 2

SELECT DISTINCT HEARTHSTONE.LEGENDARY_CARD
FROM HEROES, CLASS, HEARTHSTONE
WHERE HEROES.CLASS_ID = CLASS.CLASS_ID
AND HEARTHSTONE.CLASS_ID = CLASS.CLASS_ID
AND HEARTHSTONE.LEGENDARY_CARD = HEROES.HERO_ID;

LEGENDARY CARD
Jaina
Anduin