The Skyrim Database

Steven Bouwkamp, Joshua Crum,
Caitlin Crowe, and Bryce Hutton

CIS 353 (01)
Dr. Jamal Alsabbagh

Winter 2017

Database Requirements

Our database will be on the role playing video game Skyrim. As with many role playing games, there exists a large amount of items, actors, locations, and quests. Many of these, by the nature of the type of game, are related to each other and can be modeled nicely by a database. For example, a monster can be of a certain type and multiples of those monsters can be found in a certain location. There also exists the notion of ownership within the game to be able to differentiate two of the exact same entities apart. An example of this ownership would be in the case of summoned monsters, which is directly tied to the actor that summoned it.

The goal is to model these types of relationships that are found within the game and implement them into our database. The main entities that will be within the database are as follow and are described further below: Characters, Weapons, Weapon-Type, Monsters, Monster-Type, Locations, Summoned Monsters, and Quests.

Entities

Character: All characters can be one of two types: NPC or player character. All characters have a name and gender. Every character has a base ID, and every instance of every character has a unique reference ID. Characters must be of a single race from the ten possible races that are found within the game's characters. All NPC characters also have a class, such as assassin, warrior, child, or pawnbroker. Some characters provide a service, such as blacksmith, vendor, or follower. All characters have a level, and only some characters have health, magicka, stamina, skills, and abilities. More important characters are essential and cannot be killed, while other characters are not. Some characters are a part of one or more of the many factions that exist. Some characters are friendly to the player character while others will automatically be aggressive. The player character can start a quest at a point in time within the game. Some characters can carry weapons in their inventory.

Quests: In the game, each quest has a QuestName, which identifies the quest. Each quest can be inside multiple different locations, and each location can contain multiple quests. Each Quest

can be started by NPCs, and timeGiven tells the time that the NPC gives the character the quest. Multiple quests can be given by one NPC and multiple NPCs can give one quest.

Weapons: In skyrim there are many different weapons. Each weapon needs to be of one WeaponType, and each WeaponType can have multiple weapons. Each Weapon has a wID, which identifies that specific weapon. A weapon can be in one location, or be in one character. Each character and location can contain more than one weapon. Weapons have attack, which is a value that shows how much damage they can do a target. Value shows how much money a weapon is worth. soulCharge shows how many times you can use that weapon before needing to refill its magic. Weight shows how much the weapon weighs. Enchantment is a multi-valued attribute that shows any current enchantments that the weapon has.

WeaponType: Each weapon is of a WeaponType, each WeaponType can have many Weapons, but each Weapon needs to have only one type. WeaponType have Wield, which tells if the weapon requires two hands or one hand. They have DamageType, which tells what time of damage it, such as sharp or blunt. It has wName, which identifies what type of weapon it is, such as sword or mace. It has speed, which controls how fast weapons of that type swing, stagger, which controls how much it causes a target to pause before continuing action, and stamina, which details how many times it can be used in a row without stopping. Material shows what type of material the weapontype is, such as steel, iron, or ebony.
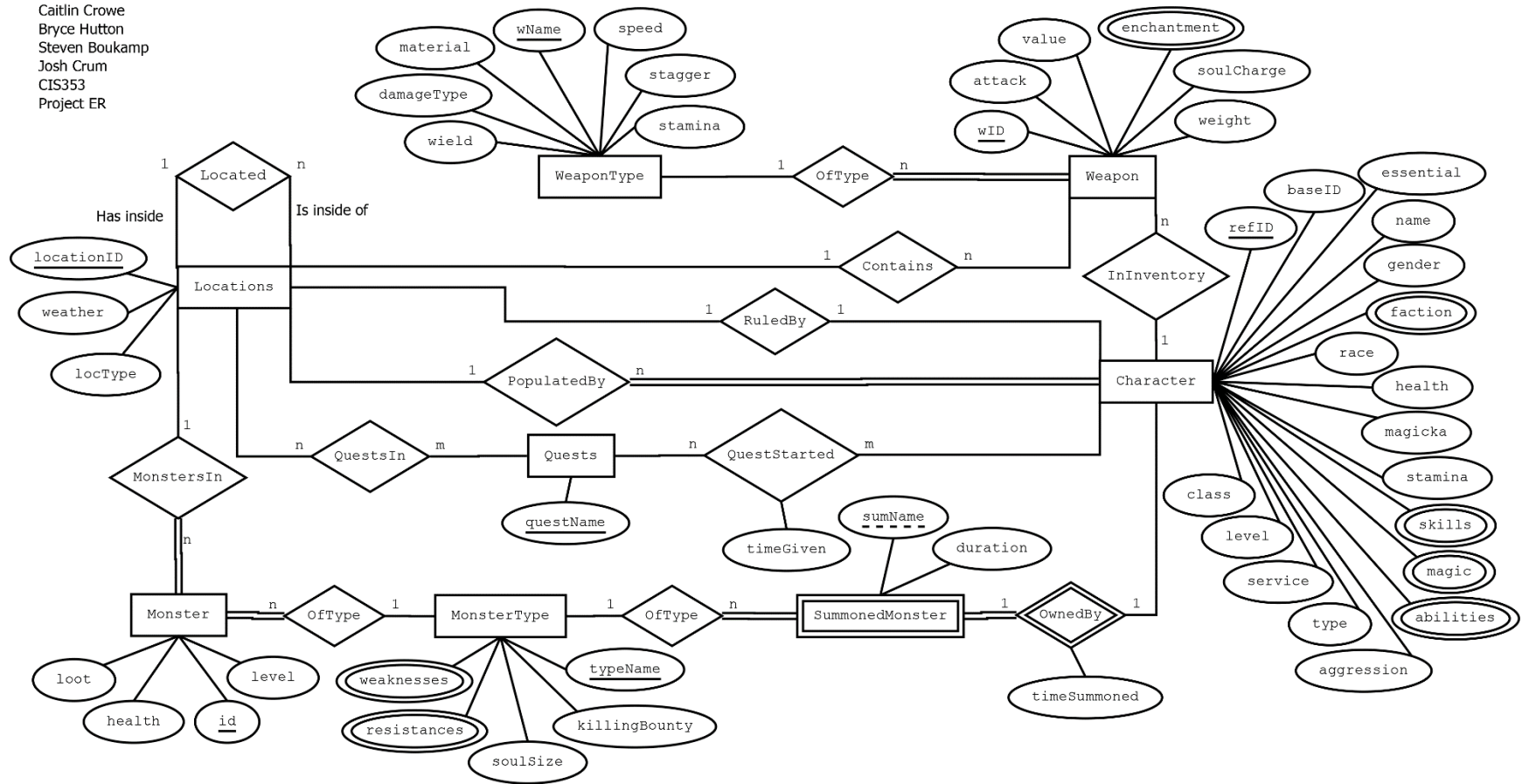
Monsters: There are many different kinds of monsters that can be found within Skyrim, some examples are bears, wolves, and dragons. All monsters have an id that is unique to that monster, a level that determines how strong the monster is, health that says how much a monster needs to be hit before it is killed, and loot, which determines the value of the loot that it drops. Monsters can be found in different locations, and need to be in a location. Each monster can only be in one location, but each location can have multiple monsters. Monsters also need to be of a type of monster, each monster can be of one MonsterType, and each MonsterType can have multiple monsters in it.

MonsterType: There are several types of monsters, such as undead, animals, or vampires, and each has a unique name to identify its type. Each type of monster has a certain soul size, potentially many resistances and weaknesses to damage, and potentially a bounty placed on the player for its death. Each monster needs to be one type of monster, and monster type can have multiple monsters in it. Each summoned monster needs to be of a monster type, and each monster type can have multiple summoned monsters in it.

Summoned Monster: Characters in the game can summon a monster in order to help them in combat. The summoned monster has sumName, which partially identifies it, and also has the character's (who summoned it), ID. Each summon needs to be owned by one character, and each character can only have one summoned monster. Time Summoned is the time that the character summoned the monster. Duration is how long the monster stays alive after being summoned. Each summoned monster needs to be of a monster type, and each monster type can have multiple summoned monsters.

Locations: Locations can be of different types, such as regions, cities, shops, and rooms. Some locations will have multiple locations inside of them, and some locations will be inside of other locations, but there can be locations that have nothing in them and aren't part of other locations. Locations have LocationID, which identifies them, the weather of the location, and the type of location. Each location can have monsters in it, and each monster needs to be in one location. Each location can have NPCs in it, and each NPC needs to be in one location. Some locations are ruled by certain npcs, but not every location has a ruler and not every npc is a ruler. Locations also have weapons inside of them, and there can be many weapons for a location but a weapon can only be in one location at once. Locations can also have many different quests in them, and quests can also be in many different locations.

Caitlin Crowe
Bryce Hutton
Steven Boukamp
Josh Crum
CIS353
Project ER

Entities, attributes, and relationships in the ER diagram:

- Located (1 ... n) — Has inside / Is inside of
- Locations: locationID, weather, locType
- Contains (1 ... n)
- WeaponType: material, wName, speed, stagger, stamina, damageType, wield
- OfType (1 ... n)
- Weapon: value, attack, enchantment, soulCharge, wID, weight
- InInventory (n ... 1)
- Character: baseID, essential, refID, name, gender, faction, race, health, magicka, stamina, skills, magic, abilities, class, level, service, type, aggression
- RuledBy (1 ... 1)
- PopulatedBy (1 ... n)
- QuestsIn (n ... m)
- Quests: questName
- QuestStarted (n ... m): timeGiven, duration
- SummonedMonster: sumName
- OwnedBy (1 ... 1): timeSummoned
- MonstersIn (1 ... n)
- Monster: loot, level, health, id
- OfType (n ... 1)
- MonsterType: weaknesses, typeName, resistances, killingBounty, soulSize
- OfType (1 ... n)

Relational Schema

WeaponType(<u>wName</u>, speed, stagger, stamina, damageType, wield, material)

Weapon(<u>wID</u>, attack, value, soulCharge, weight, locationID, refID)

Character(<u>refID</u>, baseID, essential, name, gender, race, health, magicka, stamina, aggression, type, service, level, class, locationID)

Quests(<u>questName)</u>

MonsterType(<u>typeName</u>, killingBounty, soulSize)

Monster(<u>id</u>, health, level, loot, locationID, typeName)

Locations(<u>locationID</u>, weather, locType, refID, locatedIn)

SummonedMonster(<u>refID, sumName</u>, duration, timeSummoned, typeName)

MonsterWeaknesses(<u>name, weakness</u>)

MonsterResistances(<u>name, resistance</u>)

WeaponEnchantment(<u>wID, enchantment</u>)

CharacterSkills(<u>refID, skill</u>)

CharacterMagic(<u>refID, magic</u>)

CharacterAbility(<u>refID, ability</u>)

CharacterFaction(<u>refID, faction</u>)

QuestStarted(<u>questName, refID</u>, timeStarted)

QuestsIn(<u>locationID, questName</u>)

Integrity Constraints

IC name & table(s)	IC type	English statement
ICChar_PK Character Table	Key	The primary key of the entity Character is refID.
ICChar_Loc_FK Character Table	Foreign Key	Every location that a Character can be in must be a valid location.
ICChar_1att Character Table	1-attribute	Levels of a character must be between 1 and 100.
ICMon_2att Monster Table	2-attribute, 1-row	Check that low level monsters do not drop a large amount of loot.