Disgaea 2 Database

Table of contents

Executive Summary

Disgaea 2 is a strategy role playing game that makes use of a lot of different game mechanics and a lot of data. The vast amount of information regarding Classes, Subclasses, base stats of these classes, and so on, may be overwhelming for players who are just beginning and starting out. For this reason, There should be an way to manage all the details and Data, so that nothing gets lost.

This database is a response to that, having been built to create and manage the vast majority of information and data that the game presents its players. By doing this, it will be far less likely that players will lose track of any sort of data, and they will have an easier time playing as a result of the data's organization and context.

ER-Diagram
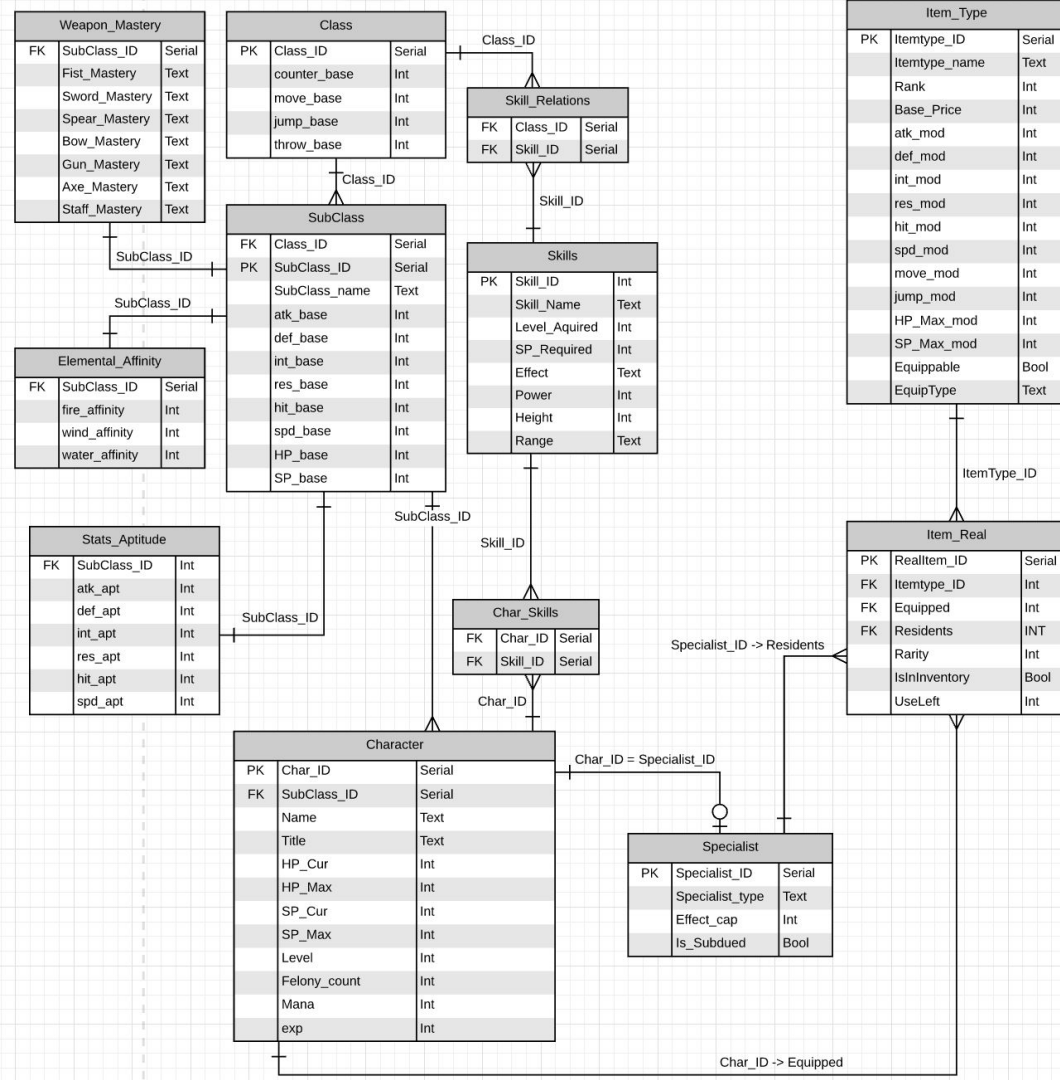
Tables

Class: This table holds the main information regarding Classes in Disgaea 2

```
Create Table If Not Exists "Class" (
  "Class_ID" Serial Primary Key,
  "counter_base" Int Not Null,
  "move_base" Int Not Null,
  "jump_base" Int Not Null,
  "throw_base" Int Not Null
);
```

Tables

SubClass: This table holds the main information regarding Sub-Classes in Disgaea 2

```
create table if not exists "SubClass" (
  "Class_ID" Serial references "Class"("Class_ID"),
  "SubClass_ID" Serial primary key,
  "SubClass_name" Text Not Null,
  "atk_base" Int Not Null,
  "def_base" Int Not Null,
  "int_base" Int Not Null,
  "res_base" Int Not Null,
  "hit_base" Int Not Null,
  "spd_base" Int Not Null,
  "HP_base" Int Not Null,
  "SP_base" Int Not Null
);
```

Tables

Weapon_Mastery: This table holds Weapon Mastery Details for Sub-Classes in Disgaea 2

```
create table if not exists "Weapon_Mastery" (
  "SubClass_ID" serial references "SubClass"("SubClass_ID"),
  "Fist_Mastery" Text,
  "Sword_Mastery" Text,
  "Spear_Mastery" Text,
  "Bow_Mastery" Text,
  "Gun_Mastery" Text,
  "Axe_Mastery" Text,
  "Staff_Mastery" Text
);
```

Tables

Elemental_Affinity: This table holds Elemental Affinity Details for Sub-Classes in Disgaea 2

```
create table if not exists "Elemental_Affinity" (
  "SubClass_ID" Serial references "SubClass"("SubClass_ID"),
  "fire_affinity" Int,
  "wind_affinity" Int,
  "water_affinity" Int
);
```

Tables

Stats_Aptitude: This table holds Stat Aptitude
Details for Sub-Classes in Disgaea 2

```
create table if not exists "Stats_Aptitude" (
  "SubClass_ID" Serial references "SubClass"("SubClass_ID"),
  "atk_apt" Int Not Null,
  "def_apt" Int Not Null,
  "int_apt" Int Not Null,
  "res_apt" Int Not Null,
  "hit_apt" Int Not Null,
  "spd_apt" Int Not Null
);
```

Tables

Skills: This table holds Information Regarding
All of the Skills in Disgaea 2

```
create table if not exists "Skills" (
  "Skill_ID" Serial primary key,
  "Skill_Name" Text Not Null,
  "Level_Aquired" Int Not Null,
  "SP_Required" Int Not Null,
  "Effect" Text Not Null,
  "Power" Int Not Null,
  "Height" Int Not Null,
  "Range" Text Not Null
);
```

Tables

Skills: This table Shows the Relations of Which Skills
Are attainable by Which Classes

```
create table if not exists "Skill_Relations" (
  "Class_ID" Serial references "Class"("Class_ID"),
  "Skill_ID" Serial references "Skills"("Skill_ID")
);
```

Tables

Character: This table details of all
Characters that Exist

```
create table if not exists "Character" (
  "Char_ID" Serial primary key,
  "SubClass_ID" Serial references "SubClass"("SubClass_ID"),
  "Name" Text,
  "Title" Text,
  "HP_Cur" Int,
  "HP_Max" Int,
  "SP_Cur" Int,
  "SP_Max" Int,
  "Level" Int,
  "Felony_count" Int,
  "Mana" Int,
  "exp" Int
);
```

Tables

Character: This table details of all Characters that Exist

```
create table if not exists "Character" (
  "Char_ID" Serial primary key,
  "SubClass_ID" Serial references "SubClass"("SubClass_ID"),
  "Name" Text,
  "Title" Text,
  "HP_Cur" Int,
  "HP_Max" Int,
  "SP_Cur" Int,
  "SP_Max" Int,
  "Level" Int,
  "Felony_count" Int,
  "Mana" Int,
  "exp" Int
);
```

Tables

Char_Skills: This table relates All existing Characters to Whatever Skills they have learned and achieved

```
create table if not exists "Char_Skills" (
  "Char_ID" Serial references "Character"("Char_ID"),
  "Skill_ID" Serial references "Skills"("Skill_ID")
);
```

Tables

Specialist: A Specialist has inheritance of the Character Table; Specialists can Reside within Items and affect them

```
create table if not exists "Specialist" (
-- Inherits all columns from Char Table
  "Spec_ID" Serial primary key Check ("Spec_ID" = "Char_ID"),
  "Specialist_type" Text,
  "Effect_cap" Int,
  "Is_Subdued" Bool
) Inherits ("Character");
```

Tables

Item_Type: This Table Contains Base information
For all item Types

```
create table if not exists "Item_Type" (
  "Itemtype_ID" Serial primary key,
  "Itemtype_name" Text,
  "Rank" Int,
  "Base_Price" Int,
  "atk_mod" Int,
  "def_mod" Int,
  "int_mod" Int,
  "res_mod" Int,
  "hit_mod" Int,
  "spd_mod" Int,
  "move_mod" Int,
  "jump_mod" Int,
  "HP_Max_mod" Int,
  "SP_Max_mod" Int,
  "Equippable" Bool,
  "EquipType" Text
);
```

Tables

Item_Real: This Table Contains Items that are Real and Exist in game

```
create table if not exists "Item_Real" (
  "Itemtype_ID" Int references "Item_Type"("Itemtype_ID"),
  "RealItem_ID" Serial primary key,
  "Equipped" Int references "Character"("Char_ID") Default Null,
  "Resident" Int references "Specialist"("Spec_ID"),
  "Rarity" Int,
  "IsInInventory" Bool,
  Constraint Resident_Equip check ("Resident"<>"Equipped")
);
```

Views

CharacterClassMastery: This view will return all Characters,
along with their subclass name, and that subclass' Weapon Mastery

```
Create view CharacterClassMastery As
Select "Character"."Name", "SubClass"."SubClass_name"
from "Character"
Right outer join "SubClass" On
"Character"."SubClass_ID" = "SubClass"."SubClass_ID"
inner join "Weapon_Mastery" on
"SubClass"."SubClass_ID" = "Weapon_Mastery"."SubClass_ID";
```

Views

emptyItems: This view will return the name, rarity, and price of all
Items in your inventory that are not equipped to any character
And that have no residents

```
Create view emptyItems As
Select "Item_Type"."Itemtype_name", "Item_Real"."Rarity", "Item_Type"."Base_Price"
from "Item_Type", "Item_Real"
where "Item_Real"."Resident" = null
AND "Item_Real"."Equipped" = null
AND "Item_Real"."IsInInventory" = true;
```

Reports

Stored procedures

Triggers

Security

Implementation notes

Known Problems

Future enhancements