Food Network Recipes Database

Database Design Specifications

By: Jessica Rieger

Table of Contents

Executive Summary

Overview

The world of food is expanding in every direction. Each day thousands of new ingredients, utensils, flavor combos, substitutions, chefs, and recipes are emerging. Recording and managing this huge repository of information is a challenge that many companies in the food industry are facing.

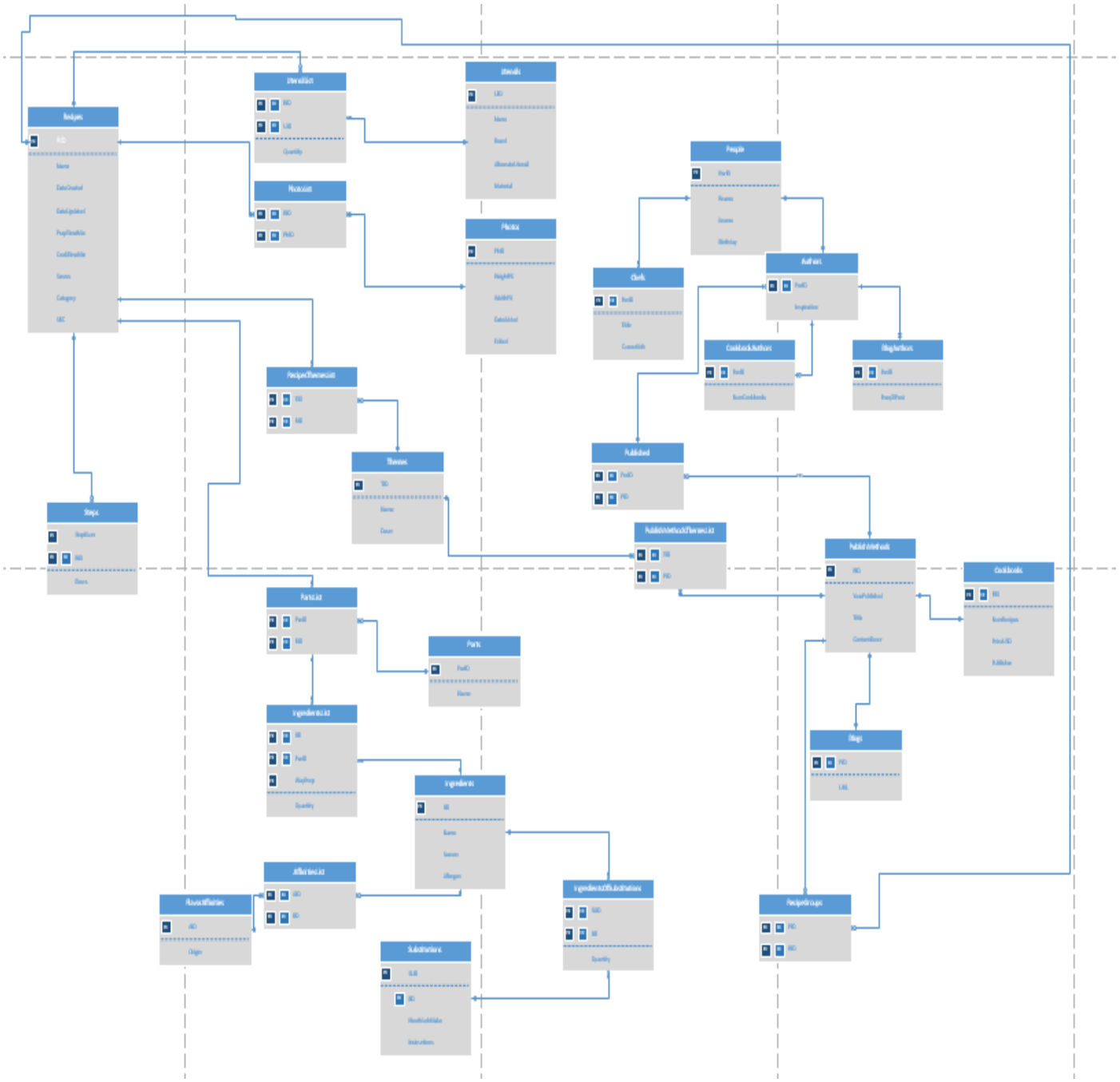
In order to address this problem, companies must find a way to organize the information in storage so that it can be accessed quickly to suit their needs. However, the complexities of the relationships between recipes, their source of publications, and all the different flavor affinities can be daunting. To stay on top of all the new information, a company must have a solid foundation for storage that works no matter what new flavor combo or publication method the chefs and authors of the world decide to invent.

Objectives

The purpose of this document is to outline a database system to manage Food Network's records about their recipes. This includes their authors and chefs, their source of publication, themes associated with the recipe and publication source, substitutions for different ingredients in the recipe, possible ways to tweak the recipes based on their ingredients' membership in different flavor affinities, and information about the recipes' ingredients, utensils required, and photos. This database is intended to provide a way for the company to manage the huge amounts of data that are associated with recipe storing.

This document will provide an overview of the database and its technical and implementation details. It will outline the tables and their functional dependencies, views, reports, stored procedures, triggers, and security features. It will also explore the databases potential for expansion and enhancements.

Entity-Relationship Diagram



Tables

People

Purpose

This table is used to store the ID, name, and date of birth of the people associated with the database including chefs, cookbook authors, and blog authors.

Create Statement

```
CREATE TABLE People (
   PerID              text not null,
   Fname              text not null,
   Lname              text not null,
   Birthday           date not null,
  PRIMARY KEY (PerID)
);
```

Functional Dependencies

PerID -> Fname, Lname, Birthday

Sample Data

	perid text	fname text	lname text	birthday date
1	PER09	Jessica	Rieger	1997-03-23
2	PER01	Kathryne	Smith	1990-01-15
3	PER02	Ina	Garten	1948-02-02
4	PER03	Andie	Mitchell	1985-03-20
5	PER04	Tessa	Bramley	1985-01-01
6	PER05	Rick	Bayless	1953-11-23
7	PER06	Deann	Groen Bayless	1948-10-30
8	PER07	Natalie	Smith	1953-11-23
9	PER08	Jane	Moorhead	1895-11-23
10	PER10	Peggy	Wilson	1896-11-23
11	PER11	Lidia	Bastianich	1947-02-21
12	PER12	Cassidy	Mazelin	1997-05-21

Chefs

Purpose

This table is used to store the information about a person in regards to their status as a chef, for example their title and current job.

Create Statement

```
CREATE TABLE Chefs (
   PerID              text not null references People(PerID),
   Title              text,
   CurrentJob         text,
  PRIMARY KEY (PerID)
);
```

Functional Dependencies

PerID -> Title, CurrentJob

Sample Data

	perid text	title text	currentjob text
1	PER09	Chef (The Gourmet College Chef)	Home Chef
2	PER12	Sous Chef	The Gourmet College Chefs Sous Chef
3	PER02	Chef	TV Chef
4	PER03	Chef in training	Home Chef
5	PER04	Sous Chef	Pastabilities
6	PER11	Chef	TV Chef

Authors

Purpose

This table is used to store the information about a person in regards to their status as an author, for example where their inspiration comes from.

Create Statement

```
CREATE TABLE Authors (
        PerID           text not null references People(PerID),
        Inspiration     text,
 PRIMARY KEY (PerID)
);
```

Functional Dependencies

PerID -> Inspiration

Sample Data

	perid text	inspiration text
1	PER09	All different types of cuisine!
2	PER01	Healthy
3	PER02	Home Cooking
4	PER03	Healthy and Quick
5	PER04	Vegetarian
6	PER05	Mexican Cuisine
7	PER06	Mexican Cuisine
8	PER07	Favorite Foods
9	PER08	Her Grandmas Recipes
10	PER10	Her Grandmas Recipes
11	PER11	Italian Cuisine

CookbookAuthors

Purpose

This table is used to store the information about a person in regards to their status as cookbook author, for example the number of cookbooks they have authored.

Create Statement

```
CREATE TABLE CookbookAuthors (
   PerID              text not null references Authors(PerID),
   NumCookbooks       int,
 PRIMARY KEY (PerID)
);
```

Functional Dependencies

PerID -> Inspiration

Sample Data

	perid text	numcookbooks integer
1	PER02	4
2	PER04	2
3	PER05	1
4	PER06	7
5	PER08	1
6	PER10	1
7	PER11	30

BlogAuthors

Purpose

This table is used to store the information about a person in regards to their status as a blog author, for example how frequently they post.

Create Statement

```
CREATE TABLE BlogAuthors (
   PerID              text not null references Authors(PerID),
   FreqOfPost         text,
 PRIMARY KEY (PerID)
);
```

Functional Dependencies

PerID -> FreqOfPost

Sample Data

	perid text	freqofpost text
1	PER09	Once a Week
2	PER01	Everyday
3	PER03	Once a Week
4	PER07	Once a week

PublishMethods

Purpose

This table is used to store the information about the publication, in particular the cookbook or blog. It stores information such as the year it was published/started, the title, and a short description of the content.

Create Statement

```
CREATE TABLE PublishMethods (
   PID                text not null,
   YearPublished      date not null,
   Title              text not null,
   ContentDescr       text,
  PRIMARY KEY (PID)
);
```

Functional Dependencies

PID -> YearPublished, Title, ContentDescr

Sample Data

	pid text	yearpublished text	title text	contentdescr text
1	P01	2010	COOKIE & kate	Healthy, simple cooking that has recipes for any time of day.
2	P08	2016	The Gourmet College Chef	Whatever I feel like cooking.
3	P02	2003	Barefoot Contessa at Home	Easy, simple, home cooking.
4	P03	2010	Andie Mitchell	Recipes * Inspiration * Life
5	P04	2007	Easy Vegetarian	Simple recipes for lunch, brunch, and dinner.
6	P05	1987	Authentic Mexican: Regional Cooking from the Heart of Mexico	Classic Mexican Cuisine.
7	P06	2014	Ce que Jaime	Indulgence Foods
8	P07	1915	Peoria Womens Cook Book	Your Grandmas Recipes
9	P09	2013-10-15	Lidias Commonsense Italian Cooking	Easy, simple, Italian cooking.
10	P10	2013	Lidia Recipe Archives	Italian cooking.

Published

Purpose

This table is used to store information about who the authors are of the cookbooks and blogs. In order to have comprehensive information about a publication, it is necessary to know who authored it, thus there is a clear need for this table.

Create Statement

```
CREATE TABLE Published (
   PID                  text not null references PublishMethods(PID),
   PerID                text not null references Authors(PerID),
  PRIMARY KEY (PerID, PID)
);
```

Functional Dependencies

PID, PerID ->

Sample Data

	pid text	perid text
1	P01	PER01
2	P08	PER09
3	P02	PER02
4	P03	PER03
5	P04	PER04
6	P05	PER05
7	P05	PER06
8	P06	PER07
9	P07	PER06
10	P07	PER10

Blogs

Purpose

This table is used to store information about a publication in regards to its status as a blog, for example its URL so that it can be visited if necessary.

Create Statement

```
CREATE TABLE Blogs (
   PID                  text not null references PublishMethods(PID),
   URL                  text not null,
 PRIMARY KEY (PID)
);
```

Functional Dependencies

PID, PerID ->

Sample Data

	pid text	url text
1	P01	http://cookieandkate.com
2	P08	http://thegourmetcollegechef.weebly.com/
3	P03	http://andiemitchell.com
4	P06	http://natandmac.tumblr.com/post/97603276107/bacon-brie-avocado-foodgasm?crlt.pid=camp.TAJXivS9eaDm

Cookbooks

Purpose

This table is used to store more detailed information about a publication in regards to its status as a cookbook, for example the number of recipes it contains, the its price in US dollars, and the publishing company.

Create Statement

```
CREATE TABLE Cookbooks (
   PID              text not null references PublishMethods(PID),
   NumRecipes       integer,
   PriceUSD         decimal,
   Publisher        text,
  PRIMARY KEY (PID)
);
```

Functional Dependencies

PID, PerID ->

Sample Data

	pid text	numrecipes integer	priceusd numeric	publisher text
1	P02	150	18.99	Clarkson Potter/Publishers
2	P04	75	13.19	Ryland Peters & Small, Inc.
3	P05	200	26.99	Clarkson Potter/Publishers
4	P07	170	10.00	J.W. Franks and Sons Printers

Recipes

Purpose

This table is used to store the general information about a recipe, for example its name, creation date, the date it was updated, its prep time, cook time, how many it serves, the category that it belongs to, and its source. It also contains the ID of the recipe that can be used to retrieve the ingredients list, steps, utensils list, and photographs associated with the recipe. Its source indicates whether it is original, or it is inspired by another recipe.

Create Statement

```
CREATE TABLE Recipes (
   RID              text not null,
   Name             text not null,
   DateCreated      date not null,
   DateUpdated      date not null,
   PrepTimeMin      integer,
   CookTimeMin      integer,
   Serves           integer,
   Category         text,
   SRC              text not null,
   PRIMARY KEY (RID)
);
```

Functional Dependencies

RID -> Name, DateCreated, DateUpdated, PrepTimeMin, CookTimeMin, Serves, Category, SRC

Sample Data

	rid text	name text	datecreated date	dateupdated date	preptimemin integer	cooktimemin integer	totaltimemin integer	serves integer	category text	src text
1	R01	Butternut Squash Soup	2015-11-11	2015-11-11	10	55	65	4	Soup	Original
2	R02	Lemon Cake	2003-01-25	2016-11-11	30	60	90	12	Cake	Original
3	R03	Slow Cooker Mexican Pulled Pork Tacos	2011-03-08	2016-11-24	15	480	495	4	Tacos	Original
4	R04	Fiorentina	2007-01-01	2016-11-20	30	20	50	4	Pizza	Original
5	R05	Cold Chicken and Avocado with Chile Chipotle	1987-01-01	2016-11-11	30	85	105	4	Chicken	Original
6	R06	Bacon, Brie, and Avocado Sandwich	2014-09-15	2016-11-11	15	10	25	2	Sandwich	Original
7	R07	Steak A La Creole	1915-09-15	2016-11-11	25	120	145	4	Steak	Original
8	R08	Mexican Chili	1915-09-15	2016-11-12	20	30	50	4	Chili	Original
9	R09	Chocolate Chip Cookies	1915-09-15	2016-11-12	140	12	152	20	Cookies	My Mother
10	R10	Almond and Coffee Cream Mini-Tarts	2013-10-15	2016-11-12	75	25	100	8	Tarts	Original

RecipeGroups

Purpose

This table allows us to know which recipes are in each publication method.

Create Statement

```
CREATE TABLE RecipeGroups (
   PID                  text not null references PublishMethods(PID),
   RID                  text not null references Recipes(RID),
PRIMARY KEY (PID, RID, PRoID)
);
```

Functional Dependencies

PID, RID ->

Sample Data

	pid text	rid text
1	P01	R01
2	P08	R01
3	P02	R02
4	P08	R02
5	P03	R03
6	P08	R03
7	P04	R04
8	P08	R04
9	P05	R05
10	P06	R06
11	P08	R06
12	P07	R07
13	P07	R08
14	P08	R09
15	P09	R10
16	P10	R10

Utensils

Purpose

This table is used to store information about utensils, such as their name, the recommended brand, and the material that they are made of. It also stores an alternate utensil that can be used if the suggested utensil is unavailable. This is useful as the perfect utensil is not always readily available in the chef's kitchen.

Create Statement

```
CREATE TABLE Utensils (
   UID                  text not null,
   Name                 text not null,
   Brand                text not null,
   AlternateUtensil     text,
   Material             text,
  PRIMARY KEY (UID)
);
```

Functional Dependencies

UID -> Name, Brand, AlternateUtensil, Material

Sample Data

	uid text	name text	brand text	alternateutensil text	material text
1	U01	High-Performance Blender	Vitamix	Immersion Blender	n/a
2	U02	8 1/2 by 4 1/4 by 2 1/2 inch loaf pan	William Sonoma	alternate size loaf pans	aluminum
3	U03	Crockpot	William Sonoma	None	Clay
4	U04	Large Saucepan	All-Clad	Pasta Pot	Calphalon
5	U05	Strainer	None	None	Any
6	U06	Pizza Stone	Old Stone	Baking Sheet	Stone
7	U07	Medium Saucepan	William Sonoma	Pasta Pot	Calphalon
8	U08	Baking Sheet	Nordic Ware	aluminum foil	aluminum
9	U09	Large Knife	William Sonoma	Medium Knife	Stainless Steel
10	U10	Medium skillet	Cuisinart	Large skillet	Stainless Steel

UtensilsList

Purpose

This table is used to store information about which utensils are needed for a particular recipe as well as the number of each utensil required.

Create Statement

```
CREATE TABLE UtensilsList (
   RID              text not null references Recipes(RID),
   UID              text not null references Utensils(UID),
   Quantity         integer,
  PRIMARY KEY (UID, RID)
);
```

Functional Dependencies

RID, UID -> Quantity

Sample Data

	rid text	uid text	quantity integer
1	R01	U01	1
2	R02	U02	2
3	R03	U03	1
4	R04	U04	1
5	R04	U05	1
6	R04	U06	1
7	R05	U07	1
8	R06	U08	1
9	R06	U09	1
10	R06	U10	1
11	R06	U11	1

Photos

Purpose

This table is used to store information about photos for a recipe, such as the height and width in pixels, the date it was added, and whether or not it has been edited.

Create Statement

```
CREATE TABLE Photos (
   PhID              text not null,
   HeightPX          integer not null,
   WidthPX           integer not null,
   DateAdded         date,
   Edited            boolean,
  PRIMARY KEY (PhID)
);
```

Functional Dependencies

PhID -> HeightPX, WidthPX, DateAdded, Edited

Sample Data

	phid text	heightpx integer	widthpx integer	dateadded date	edited boolean
1	PH01	640	800	2016-11-23	t
2	PH02	406	305	2016-11-24	t
3	PH03	406	305	2016-11-24	t
4	PH04	440	400	2016-11-26	t
5	PH05	206	605	2016-11-30	t
6	PH06	606	405	2016-11-24	t
7	PH07	606	405	2016-11-24	t
8	PH08	206	315	2016-11-24	t
9	PH09	706	915	2016-11-24	t
10	PH10	406	315	2016-11-24	t
11	PH11	406	315	2016-11-24	t
12	PH12	404	315	2016-11-24	t

PhotosList

Purpose

This table is used to store information about which photos are associated with a particular recipe.

Create Statement

```
CREATE TABLE PhotosList (
   PhID                    text not null references Photos(PhID),
   RID                     text not null references Recipes(RID),
 PRIMARY KEY (PhID, RID)
);
```

Functional Dependencies

PhID, RID ->

Sample Data

	phid text	rid text
1	PH01	R01
2	PH02	R02
3	PH03	R03
4	PH04	R03
5	PH05	R04
6	PH06	R05
7	PH07	R06
8	PH08	R07
9	PH09	R07
10	PH10	R08
11	PH11	R09
12	PH12	R10

Steps

Purpose

This table is used to store the information of all the steps for each recipe, such as the recipe each are associated with, its step number, and the text that makes up the instructions for the step.

Create Statement

```
CREATE TABLE Steps (
   StepNum          text not null,
   RID              text not null references Recipes(RID),
   Descr            text not null,
  PRIMARY KEY (StepNum, RID)
);
```

Functional Dependencies

StepNum, RID ->

Sample Data

	stepnum text	rid text	descr text
1	S1	R01	Preheat the oven to 425 degrees Fahrenheit and line a rimmed baking sheet with parchment paper. Place the butternut squash on th
2	S2	R01	Turn the squash face down and roast until it is tender and completely cooked through, about 45 to 50 minutes. Set the squash asi
3	S3	R01	Meanwhile, in a medium skillet (or large soup pot, if you'll be serving soup from that pot), warm 1 tablespoon olive oil over me
4	S4	R01	If you have a high performance blender like a Vitamix (see notes if you are using an immersion blender instead), transfer the co
5	S5	R01	If you would like to thin out your soup a bit more, add the remaining cup of broth (I used the full 4 cups, but if you used a sm
6	S6	R01	Serve immediately (I like to top each bowl with a little more black pepper). Let leftover soup cool completely before transferri
7	S1	R02	Preheat the oven to 350 degrees F. Grease and flour 2 (8 1/2 by 4 1/4 by 2 1/2-inch) loaf pans. You may also line the bottom wit
8	S2	R02	Cream the butter and 2 cups granulated sugar in the bowl of an electric mixer fitted with the paddle attachment, until light and
9	S3	R02	Sift together the flour, baking powder, baking soda, and salt in a bowl. In another bowl, combine 1/4 cup lemon juice, the butte
10	S4	R02	Combine 1/2 cup granulated sugar with 1/2 cup lemon juice in a small saucepan and cook over low heat until the sugar dissolves.
11	S5	R02	For the glaze, combine the confectioners sugar and the lemon juice in a bowl, mixing with a wire whisk until smooth. Pour over t
12	S1	R03	Place the pork and all the ingredients in part 1 into the crockpot and cook on low for 8 hours.
13	S2	R03	Pull the pork apart with a fork.
14	S3	R03	Assemble the taco using all the ingredients in part 2.
15	S1	R04	Put stone in oven and preheat it to 425 degrees F.
16	S2	R04	Wash Spinach, put in saucepan, put top on and cook until wilted (2-3 min.) Drain and squeeze out excess water.

Ingredients

Purpose

This table is used to store information about the ingredients, such as their name, their season that they are best, and whether or not they are an allergen.

Create Statement

```
CREATE TABLE Ingredients (
   IID            text not null,
   Name           text not null,
   Season          text not null,
   Allergen       boolean,
  PRIMARY KEY (IID)
);
```

Functional Dependencies

IID -> Name, Season, Allergen

Sample Data

	iid text	name text	season text	allergen boolean
1	I01	Butternut Squash	Fall	f
2	I02	Olive Oil	n/a	f
3	I03	Shallot	Fall	f
4	I04	Salt	n/a	f
5	I05	Garlic	Summer	f
6	I06	Maple Syrup	Winter	f
7	I07	Nutmeg	Fall	f
8	I08	Black Pepper	n/a	f
9	I09	Vegetable Broth	n/a	f
10	I10	Butter	n/a	f
11	I49	Anchovies	n/a	f
12	I50	Rosemary	Fall/Winter	f
13	I51	Apples	Fall	f
14	I52	Caramel	n/a	f

IngredientsList

Purpose

This table is used to store the information about which ingredients go with each part of a recipe, the way they should be prepared, and the quantity needed. This table basically represents the ingredients list that is typically seen on recipes.

Create Statement

```
CREATE TABLE IngredientsList (
   IID              text not null references Ingredients(IID),
   ParID            text not null references Parts(ParID),
   WayPrep          text not null,
   Quantity         text not null,
  PRIMARY KEY (IID, ParID, WayPrep)
);
```

Functional Dependencies

IID, ParID, WayPrep -> Quantity

Sample Data

	iid text	parid text	wayprep text	quantity text
1	I01	PAR01	Chop in half and seed	1 Large
2	I02	PAR01	None	1 Tablespoon
3	I03	PAR01	Chopped	1/2 cup
4	I04	PAR01	None	1 Teaspoon
5	I05	PAR01	Pressed or Minced	4 or 5
6	I06	PAR01	None	1 Teaspoon
7	I07	PAR01	None	1/8 Teaspoon
8	I08	PAR01	Grind	To Taste
9	I09	PAR01	None	Vegetable Broth
10	I10	PAR01	None	1 or 2 Tablespoons
11	I11	PAR02	Divided	2 1/2 Cups
12	I12	PAR02	at room temperature.	4 Large
13	I13	PAR02	Zest	1/3 cup
14	I13	PAR02	Juice	3/4 Cup
15	I14	PAR02	None	3 Cups

Parts

Purpose

This table is used to store the important information associated with each part of a recipe. When looking at the ingredients and instructions for a recipe, it is critical to know the part that they are associated so that the correct quantities of each ingredient and the proper techniques or preparatory methods can be used.

Create Statement

```
CREATE TABLE Parts (
        ParID                   text not null,
        Name                    text not null,
 PRIMARY KEY (ParID)
);
```

Functional Dependencies

IID -> Name, Season, Allergen

Sample Data

	parid text	name text
1	PAR01	Soup
2	PAR02	Lemon Cake
3	PAR03	Lemon Glaze
4	PAR04	Pulled Pork
5	PAR05	Taco Assembly
6	PAR06	Entire Recipe
7	PAR07	The Chicken Mixture
8	PAR08	Finishing the dish
9	PAR09	Entire Recipe
10	PAR15	Entire Recipe
11	PAR11	Entire Recipe
12	PAR12	Entire Recipe
13	PAR13	Dough
14	PAR14	Coffee Cream

PartsList

Purpose

This table is used to store information about which parts go with each particular recipe. This is critical because a recipe is not complete without all of its parts, and the ingredients associated with each part must come together as a single ingredients list for the recipe.

Create Statement

```
CREATE TABLE PartsList (
   ParID              text not null references Parts(ParID),
   RID                text not null references Recipes(RID),
  PRIMARY KEY (ParID, RID)
);
```

Functional Dependencies

ParID, RID ->

Sample Data

	parid text	rid text
1	PAR01	R01
2	PAR02	R02
3	PAR03	R02
4	PAR04	R03
5	PAR05	R03
6	PAR06	R04
7	PAR07	R05
8	PAR08	R05
9	PAR09	R06
10	PAR15	R07
11	PAR11	R08
12	PAR12	R09
13	PAR13	R10
14	PAR14	R10

Substitutions

Purpose

This table is used to store information about the substitutions, such as the ingredient that is being substituted for, how much the substitution makes, and the instructions for doing the substitution. This information allows the chef to successfully replace one ingredient for another if they do not have the original ingredient called for.

Create Statement

```
CREATE TABLE Substitutions (
   SLID              text not null,
   IID               text not null references Ingredients(IID),
   HowMuchMake       text not null,
   Instructions      text not null,
  PRIMARY KEY (SLID)
);
```

Functional Dependencies

ParID, RID ->

Sample Data

	slid text	iid text	howmuchmake text	instructions text
1	SL01	I17	2 Cups	Put the lemon juice in the 1 Cup measuring cup then fill the rest with milk.
2	SL02	I37	Any	Do direct substitution.
3	SL03	I42	Any	Do direct substitution.
4	SL04	I33	Any	Do direct substitution.

IngredientsofSubstitution

Purpose

This table is used to store information about which ingredients are needed to complete a particular substitution, and the quantity of each.

Create Statement

```
CREATE TABLE IngredientsOfSubstitution (
   SLID              text not null references Substitutions(SLID),
   IID               text not null references Ingredients(IID),
   Quantity          text not null,
  PRIMARY KEY (SLID, IID)
);
```

Functional Dependencies

SLID, IID -> Quantity

Sample Data

	slid text	iid text	quantity text
1	SL01	I58	About 1 Cup
2	SL01	I13	1 Tablespoon
3	SL02	I37	As much as the recipe calls for.
4	SL03	I42	As much as the recipe calls for.
5	SL04	I61	As much as the recipe calls for.

FlavorAffinities

Purpose

This table is used to store information the Flavor Affinities, such as their origin. For example, a flavor affinity may be of Mexican origin, meaning that the flavor combos it creates reflect Mexican cuisine. This is useful for chefs, as it tells them which particular flavors to combine to imitate a particular cuisine.

Create Statement

```
CREATE TABLE FlavorAffinities (
   AID                          text not null,
   Origin                       text,
  PRIMARY KEY (AID)
);
```

Functional Dependencies

SLID, IID -> Quantity

Sample Data

	aid text	origin text
1	A01	African Cuisine West
2	A02	Spanish Cuisine
3	A03	American Cuisine
4	A04	Asian Cuisine
5	A05	French Cuisine
6	A06	Mexican Cuisine
7	A07	Mexican Cuisine

AffinitiesList

Purpose

This table is used to store the association of particular ingredients with a particular flavor affinity. All the ingredients that are members of the affinity can be combined by the chef to make their dish resemble the cuisine of origin, and to create harmonious flavors.

Create Statement

```
CREATE TABLE AffinitiesList (
   AID          text not null references FlavorAffinities(AID),
   IID          text not null references Ingredients(IID),
  PRIMARY KEY (AID, IID)
);
```

Functional Dependencies

AID, IID ->

Sample Data

	aid text	iid text
1	A01	I43
2	A01	I29
3	A01	I48
4	A02	I49
5	A02	I50
6	A02	I02
7	A02	I13
8	A03	I51
9	A03	I52
10	A03	I48
11	A04	I53
12	A04	I10
13	A04	I25

Themes

Purpose

This table allows us to identify different themes that could be associated with a publication method or recipe in particular. This allows us to categorize the recipes and publications based on their themes. It also gives us information about a particular theme, such as its name and description.

Create Statement

```
CREATE TABLE Themes (
   TID          text not null,
   Name         text not null,
   Descr        text,
  PRIMARY KEY (TID)
);
```

Functional Dependencies

TID -> Name, Descr

Sample Data

	tid text	name text	descr text
1	T01	Comfort	Food you want to eat on a cold day with family.
2	T02	Vegetarian	To cook without the use of any meat.
3	T03	Dessert	A nice way to finish a meal.
4	T04	Simple	Good for mid-week cooking/baking.
5	T05	Mexican	Spicy and sweet.
6	T06	Chipolte	Spicy.
7	T07	Indulgence	Foods you love but dont eat that often.
8	T08	Classics	The basic recipes.
9	T09	In the Family	Passed down recipes.
10	T10	Italian Dessert	Classic to the Italian cuisine and a good way to end a meal.

RecipeThemesList

Purpose

This table allows us to identify the particular themes that are associated with a particular recipe.

Create Statement

```
CREATE TABLE RecipeThemesList (
   TID          text not null references Themes(TID),
   RID          text not null references Recipes(RID),
 PRIMARY KEY (TID, RID)
);
```

Functional Dependencies

TID -> Name, Descr

Sample Data

	tid text	rid text
1	T01	R01
2	T03	R02
3	T05	R03
4	T02	R04
5	T06	R05
6	T07	R06
7	T01	R06
8	T04	R06
9	T08	R07
10	T01	R07
11	T04	R07

PublishMethodThemesList

Purpose

This table is used to store the association of themes with each publication. Thus, it allows us to describe the style of the publication using particular themes.

Create Statement

```
CREATE TABLE PublishMethodsThemesList (
   TID          text not null references Themes(TID),
   PID          text not null references PublishMethods(PID),
 PRIMARY KEY (TID, PID)
);
```

Functional Dependencies

TID, PID ->

Sample Data

	tid text	pid text
1	T02	P01
2	T04	P02
3	T04	P03
4	T02	P04
5	T05	P05
6	T07	P06
7	T08	P07
8	T08	P08
9	T04	P09
10	T04	P10

Views

Entire Recipe

Purpose

This view shows the entire recipe with all of its associated information. Essentially, this view shows all the information you need to access the different parts of a recipe. Using this view, you can get the ingredients, recipe information, utensils, photos, steps, and parts associated with this recipe. For a chef this would be necessary so that they can make the recipe.

Create Statement

```
create view EntireRecipe AS
select  r.name,
        r.datecreated,
        r.preptimemin,
        r.cooktimemin,
        r.serves,
        r.src,
        ul.uid,
        pl.ParID,
        rg.pid,
        phl.phid,
        s.descr
from Recipes r inner join utensilslist ul ON r.rid = ul.rid
                inner join PartsList pl ON r.rid = pl.rid
                inner join recipegroups rg ON r.rid = rg.rid
                inner join photoslist phl ON r.rid = phl.rid
                inner join Steps s ON r.rid = s.rid
                inner join RecipeThemesList rtl ON r.rid = rtl.rid;
```

Ingredient Affinities

Purpose

This view shows information about all the ingredients involved in each flavor affinity as well as the information about the affinity itself. This would be necessary for a chef who wanted to see which ingredients combine well to make a particular flavor palette.

Create Statement

```
create view IngredientAffinities AS
Select  al.aid,
        i.name,
        i.allergen,
        fa.origin
from Affinitieslist al  inner join ingredients i        ON i.iid = al.iid
                        inner join flavoraffinities fa  ON fa.aid = al.aid;
```

Publication Information

Purpose

This view shows comprehensive information about the publications. It expands the general publication information to include the themes associated with the publication and its authors. This would be necessary for a chef who wished to learn more about a particular source or someone who wanted to find the publication.

Create Statement

```
create view publicationinformation AS
Select  p.pid,
        p.title,
        p.yearpublished,
        p.contentdescr,
        peo.perid,
        peo.fname,
        peo.lname
from    Published pd,
        publishmethods p,
        authors a,
        people peo
Where pd.pid = p.pid
   and pd.perid = a.perid
   and peo.perid = a.perid;
```

Reports

Number of Recipes Published by an Author in a Particular Year

Food Network is a company that hires many different chefs and authors. This report can be used to check the productivity of their chefs and authors to ensure that they are keeping an appropriate rate of publication by showing the number of publications that they authored in a particular year. The year that the person desires is indicated by YEAR.

Query

```
select peo.fname, peo.lname, count(*) as NumberOfPublications
from    publishmethods p,
        published pu,
        authors a,
        people peo
where peo.perid = a.perid
   and p.pid = pu.pid
   and pu.perid = a.perid
   and yearpublished = YEAR
group by peo.fname, peo.lname;
```

This report will work with user input to generate the appropriate report based on the year that is entered. In the query above, I have written YEAR where the user-entered year would go. For example, if the user wanted a report about 2016, then the query would look as follows.

Query:

```
select  peo.fname, peo.lname,
        count(*) as NumberOfPublications
from    publishmethods p,
        published pu,
        authors a,
        people peo
where peo.perid = a.perid
   and p.pid = pu.pid
   and pu.perid = a.perid
   and yearpublished = '2016'
group by peo.fname,
            peo.lname;
```

Complexity of Recipe

As a company, Food Network is concerned with the average complexity of the recipes that they are posting to ensure that they have a good balance of complex and simple recipes for their viewers to choose from. The complexity of a recipe is determined based on the number of utensils it uses, the number of parts and steps it has. This report shows the average number of utensils used, and the average number of parts and steps associated with recipes. If the average of any one of these things is too high, then Food Network knows it must add recipes that are simpler in that particular area to lower the average.

Query

```
select *
from (select avg(num) as averageUtensils
        from (Select count(ul.uid) as num
                from recipes r, utensilslist ul, utensils u
                where r.rid = ul.rid
                and ul.uid = u.uid
                group by r.rid
                ) as utensilcount
        ) as utensils,
        (select avg(num1) as averageSteps
        from (Select count(steps.stepnum) as num1
            from recipes r, steps
            where r.rid = steps.rid
            group by r.rid
           ) as stepscount
        ) as average2,
        (select avg(num2) as averageParts
        from (Select count(pl.parid) as num2
            from recipes r, parts p, partslist pl
            where r.rid = pl.rid
            and pl.parid = p.parid
            group by r.rid
           ) as partscount
        ) as average3;
```

Theme Usage

This report shows how many times each theme is being used. This data report can be used to figure out which themes are being underused and need to be developed further. Unfortunately, it will not show themes that are not used by both recipes and publications, so it is up to the admin to know a list of the themes they have.

Query

```
select  COALESCE(recipetheme.rtheme,'Not Used') AS RThemes,
        COALESCE(recipetheme.ruse,'0') AS RNumTimesUsed,
        COALESCE(publishtheme.ptheme,'Not Used') AS PThemes,
        COALESCE(publishtheme.puse,'0') AS PNumTimesUsed
from (select   count(rtl.rid) as ruse,
                rtl.tid as rtheme
        from    recipes r,
                recipethemeslist rtl
        where r.rid = rtl.rid
        group by rtl.tid
       ) as recipetheme full outer join (select  count(ptl.tid) as puse,
                                                 ptl.tid as ptheme
                                          from        publishmethodsthemeslist ptl,
                                                      publishmethods pm
                                          where pm.pid = ptl.pid
                                          group by ptl.tid
                                         ) as publishtheme ON recipetheme.rtheme =
publishtheme.ptheme
Order by rthemes;
```

Potential Tweaks to a Recipe

Food network is always looking to tweak their recipes in new and interesting ways. This report shows which cuisine the ingredients in the recipe are associated with. By showing this, it tells the chefs which flavor palettes of different cuisines to draw from to enhance the recipe. As well, it also shows the flavor affinities associated with the ingredients in each recipe. This can be used to figure out which ingredients to add to the recipe to change the flavor in a way that is pleasant.

Query

```
select  r.rid,
        r.name,
        fa.origin,
        al.*,
        i.name
from    recipes r,
        partslist pl,
        ingredientslist il,
        ingredients i,
        flavoraffinities fa,
        affinitieslist al
where r.rid = pl.rid
   and pl.parid = il.parid
   and il.iid = i.iid
   and fa.aid = al.aid
   and i.iid = al.iid
order by r.rid
```

Stored Procedures

Get_Recipe_Ingredientslist_byName

Purpose:

This method allows database user to retrieve all ingredients associated with a recipe by entering the recipe ID. Clearly, this method is crucial to the database because it is necessary for a user to be able to access the ingredients associated with a recipe.

Query:

```
create or replace function get_recipe_ingredientslist_byName(text)
returns table (iid          text,
               recipename   text,
               parid        text,
               wayprep      text,
               quantity     text)
as
$$
declare
  recipeName text := $1;
begin
  return query
    select ingredients.iid,
           ingredients.name,
           ingredientslist.parid,
           ingredientslist.wayprep,
           ingredientslist.quantity
    from Ingredients,
           Ingredientslist,
           partslist,
           recipes
    where Ingredients.iid = Ingredientslist.iid
      and partslist.parid = ingredientslist.parid
      and partslist.rid = recipes.rid
      and recipes.name = recipeName;
end;
$$
language plpgsql;
```

Get_Recipe_Ingredientslist_byID

Purpose:

This method allows database user to retrieve all ingredients associated with a recipe by entering the recipe ID.

Create Statement:

```
create or replace function get_recipe_ingredientslist_byID(text)
returns table (iid          text,
               recipename   text,
               parid        text,
               wayprep      text,
               quantity     text)
as
$$
declare
  recipeID text := $1;
begin
  return query
    select ingredients.iid,
        ingredients.name,
        ingredientslist.parid,
        ingredientslist.wayprep,
        ingredientslist.quantity
    from Ingredients,
          Ingredientslist,
          partslist
    where Ingredients.iid = Ingredientslist.iid
      and partslist.parid = ingredientslist.parid
      and partslist.rid = recipeID;
end;
$$
language plpgsql;
```

Get_Recipe_Ingredientslist_byNameORID

Purpose:

The purpose of this function is to facilitate the use of the database. It allows users to either enter the ID or the name of a recipe that they want to find and it will return the correct record from the database.

Create Statement:

```
create or replace function get_recipe_ingredientslist_byNameOrId(text, text,
REFCURSOR) returns refcursor as
$$
declare
  recipeName text     := $1;
  recipeId   text     := $2;
  resultset  REFCURSOR := $3;
begin
  if (recipeId IS NOT NULL) then
    open resultset for
      select *
      FROM get_recipe_ingredientslist_byId(recipeId);
  else
                open resultset for
                        select *
                        From get_recipe_ingredientslist_byName(recipeName);
  end if;

  return resultset;
end;
$$
language plpgsql;
```

calculateTotalTimeMin

Purpose

This function is used to calculate or recalculate the total time for a recipe each time a recipe record is inserted or updated.

Query

```
create or replace function calculateTotalTimeMin() returns trigger as
$$
declare

  total integer := cast(new.preptimemin as Integer) + cast(new.cooktimemin as
integer);
begin

                new.totaltimemin = total;
                return NEW;
end;
$$
language plpgsql;
```

Triggers

TotalTimeCheck

Purpose

This trigger is called after a recipe record is updated or inserted into the recipes table. It calls the stored procedure calculateTotalTimeMin() to ensure that the total time is correct.

Query

```
create trigger totalTime
BEFORE INSERT OR UPDATE ON Recipes
        FOR EACH ROW EXECUTE PROCEDURE calculateTotalTimeMin();
```

Security

There are 6 primary users of this database: chefs, authors, administrators, photographers, affinity experts, and theme writers. For each role, the user is revoked of all privileges before being granted the appropriate privileges. To reduce the length of this section, the revoke statements have been excluded.

Administrators

Administrators are the employees at Food Network that manage the recipes database. They must be able to edit, select or delete any information that they desire in order to maintain the accuracy of the data and remove unneeded information. They are also in charge of matching recipe and publications with their appropriate themes.

GRANT SELECT, INSERT, UPDATE, DELETE ON entirerecipe TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Recipes TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON steps TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Ingredients TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON IngredientsList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON PhotosList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Photos TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Parts TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON PartsList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON UtensilsList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Utensils TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON AffinitiesList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON FlavorAffinities TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON ingredientaffinities TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON RecipeThemesList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON PublishMethodsThemesList TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Themes TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON IngredientsOfSubstitution TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Substitutions TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Pubished TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON publicationinformation TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Published TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON People TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Authors TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Chefs TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON CookbookAuthors TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON BlogAuthors TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON PublishMethods TO admins;

```
GRANT SELECT, INSERT, UPDATE, DELETE ON Blogs TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON Cookbooks TO admins;
GRANT SELECT, INSERT, UPDATE, DELETE ON RecipeGroups TO admins;
```

Chefs

Chefs need the privileges to access any information about recipes in the database so that they can make them and use them as inspiration for new recipes.

```
GRANT SELECT ON EntireRecipe TO chefs;
GRANT SELECT ON recipes TO chefs;
GRANT SELECT ON Parts TO chefs;
GRANT SELECT ON Partslist TO chefs;
GRANT SELECT ON Ingredientslist TO chefs;
GRANT SELECT ON Ingredients TO chefs;
GRANT SELECT ON IngredientsOfSubstitution TO chefs;
GRANT SELECT ON Substitutions TO chefs;
GRANT SELECT ON FlavorAffinities TO chefs;
GRANT SELECT ON AffinitiesList TO chefs;
GRANT SELECT ON Steps TO chefs;
GRANT SELECT ON IngredientAffinities TO chefs;
GRANT SELECT ON publicationinformation TO chefs;
```

Authors

This group of users needs to be able to see the information about the recipes that have been published, the other authors and all of the different publications. They are able to insert new publications, however they are not allowed to update or delete existing records.

```
GRANT SELECT, INSERT ON entirerecipe TO authors;
GRANT SELECT, INSERT ON publicationinformation TO authors;
GRANT SELECT, INSERT ON PublishMethods TO authors;
GRANT SELECT, INSERT ON Blogs TO authors;
GRANT SELECT, INSERT ON Cookbooks TO authors;
GRANT SELECT, INSERT ON RecipeGroups TO authors;
GRANT SELECT, INSERT ON steps TO authors;
GRANT SELECT, INSERT ON IngredientsList TO authors;
GRANT SELECT, INSERT ON Ingredients TO authors;
GRANT SELECT, INSERT ON PartsList TO authors;
GRANT SELECT, INSERT ON Parts TO authors;
GRANT SELECT, INSERT ON UtensilsList TO authors;
GRANT SELECT, INSERT ON Utensils TO authors;
GRANT SELECT, INSERT ON Recipes TO authors;
GRANT SELECT ON IngredientsOfSubstitution TO authors;
GRANT SELECT ON Substitutions TO authors;
```

Photographers

This group needs to be able to see the recipes associated with each publication and add photographs to be associated with the different recipes.

GRANT SELECT, INSERT, UPDATE, DELETE ON Photos TO photographers;
GRANT SELECT, INSERT, UPDATE, DELETE ON PhotosList TO photographers;
GRANT SELECT ON EntireRecipe TO photographers;
GRANT SELECT ON RecipeGroups TO photographers;
GRANT SELECT ON PublishMethods TO photographers;
GRANT SELECT ON Cookbooks TO photographers;
GRANT SELECT ON Blogs TO photographers;

Theme Writers

These users need to be able to update, insert and delete themes and their associations with particular recipes as well as access all information about recipes and their publications.

GRANT SELECT, INSERT, UPDATE, DELETE ON Themes TO themeWriters;
GRANT SELECT, INSERT, UPDATE, DELETE ON PublishMethodsThemesList TO themeWriters;
GRANT SELECT, INSERT, UPDATE, DELETE ON RecipeThemesList TO themeWriters;
GRANT SELECT ON publicationinformation TO themeWriters;
GRANT SELECT ON PublishMethods TO themeWriters;
GRANT SELECT ON Cookbooks TO themeWriters;
GRANT SELECT ON Blogs TO themeWriters;
GRANT SELECT ON Recipes TO themeWriters;
GRANT SELECT ON RecipeGroups TO themeWriters;

Affinity Experts

This group of users needs to be able to access the information about all the ingredients as well as modify the affinities and the list of ingredients associated with them.

GRANT SELECT, INSERT, UPDATE, DELETE ON ingredientaffinities TO affinityexperts;
GRANT SELECT, INSERT, UPDATE, DELETE ON AffinitiesList TO affinityexperts;
GRANT SELECT, INSERT, UPDATE, DELETE ON FlavorAffinities TO affinityexperts;
GRANT SELECT, INSERT ON Ingredients TO affinityexperts;

Implementation Notes

The following are suggestions for implementation:

1. When a recipe is entered into the database, it should be entered with all of its corresponding data at once in order to ensure that the complete recipe is present.
2. A large number of flavor affinities from different style cuisines should be entered for the first use of the database so that the ingredients of recipes looking for affinities can have a large repository to draw from.
3. If an author realizes that information about a publication they are associated with is incorrect, they should notify an admin to update the information accordingly.

Known Problems

The following are known problems with the database:

1. An author could in theory add false publications to the database that do not actually exist. The accuracy of the publications records are thus dependent on the honestly of the authors and the diligence of the admins in checking the database. The only prevention that is built in against this is that authors cannot add an author so to people or link an author to a publication through published. This means that for any authors to be associated with a publication, the publication must be reviewed by the admins first.
2. The reference for an alternate utensil is a text name. To make the database as useful as possible, this should actually be a Utensil ID so that the user can get the needed information about the alternate utensil.
3. The number of cookbooks published by an author is stored in the database, however this number will change as the database grows and the authors add new publications. Thus, this is definitely a source for inaccuracy and should be addressed in some way by the administration.
4. Additionally, there is no way to ensure that a recipe is associated with the correct publication method, meaning the administrators and the authors must check this.
5. Finally, there is no way to ensure that a publication is entered in only the cookbooks subtype or the blogs subtype. Obviously a publication can only be one at a time, so the users of the database must watch this.

Future Enhancements

The following are suggestions for future enhancements that may be desired:

1. First, the database has the potential to grow with the enterprise that it serves. For example, additional methods of publication and types of authors can be added by simply adding an additional table for each.
2. The amount of information that is stored about a recipe can be increased easily, so as the information about recipes that is important changes, the database will change as well.
3. There is a lot of potential for additional stored procedures to enhance the facility of use by the users. For example, you could implement a stored procedure that automatically makes the association of themes with recipes and publish methods based on the keywords in their descriptions.
4. It may be desired to add a table relating to the source of the recipe, such as a reference to the recipe it was inspired by and the author who originated the inspirational recipe.
5. Certainly an enhancement that would be helpful would be the addition of a way to ensure that a publication is only entered in one subtype of publishmethods. This would prevent the known problem above of falsely declaring a publishmethod as both a cookbook and a blog.
6. Finally, it may be practical to implement a method for determining which flavor affinity would be best to use for an addition to a particular recipe. For example, when making a soup it would be incorrect to add a flavor affinity meant for baking, thus this method would prevent this type of error.