

Building Better Databases





Who am I

Name: Scott Heffron

Title: Sr. Software Engineer

Company: Access Development

Software:

- ColdFusion
- C++
- C# Compact Framework
- ASP.NET

DataBases:

- MS SQL Server
- Sybase
- Oracle
- SQL Anywhere



Agenda

- Definition of Database System
- Database Normalization
- Normalization Rankings
- Types of Normalization
- Case Study
- Where to go from here
- Questions



Definition of Database System

Often abbreviated DB. A collection of information organized in such a way that a computer program can quickly select desired pieces of data. You can think of a database as an electronic filing system.

Traditional databases are organized by fields, records, and files. A field is a single piece of information; a record is one complete set of fields; and a file is a collection of records. For example, a telephone book is analogous to a file. It contains a list of records, each of which consists of three fields: name, address, and telephone number.



Database Normalization

It is the process of efficiently organizing data in a database. There are two goals of the normalization process: **eliminate redundant data** (for example, storing the same data in more than one table) and **ensure data dependencies** make sense (only storing related data in a table).

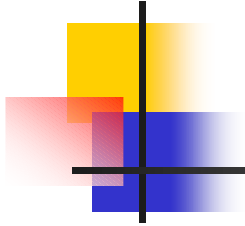
Both of these are worthy goals as they reduce the amount of space a database consumes and ensure that data is logically stored.



Normalization Rankings

The database community has developed a series of guidelines for ensuring normalization. These are referred to as normal forms and are numbered from one (first normal form or 1NF) through five (fifth normal form or 5NF).

In practical applications, you'll often see 1NF, 2NF, and 3NF along with the occasional 4NF. Fifth normal form is very rarely seen and won't be discussed at this time.



Normalization

Type

Definitions



1st Normal Form

First normal form (1NF) sets the very basic rules for an organized database and is the lowest form of normalization



1st NF Guidelines

- Eliminate duplicative columns from the same table.
- Create separate tables for each group of related data
- Identify each row with a unique column or set of columns (the primary key).



Primary Key

Uniquely identifies each record in a table. It can either be a normal attribute that is guaranteed to be unique or it can be generated by the DBMS. It may consist of a single attribute or multiple attributes in combination.



2nd Normal Form

Second normal form (2NF) further addresses the concept of removing duplicative data



2nd NF Guidelines

- Meet all the requirements of the 1NF
- Remove subsets of data that apply to multiple rows of a table and place them in separate tables.
- Create relationships between these new tables and their predecessors through the use of foreign keys.



Foreign Key

A **foreign key** is a relationship or link between two tables which ensures that the data stored in a database is consistent.

The foreign key link is set up by matching columns in one table (the **child**) to the primary key columns in another table (the **parent**).



Relationship Symbols





3rd Normal Form

Third normal form (3NF) goes
one large step further



3rd NF GuideLines

- Meet all the requirements of the 2NF.
- Remove columns that are not dependent upon the primary key



4th Normal Form

Finally, fourth normal form (4NF) has one additional requirement.



4th NF GuideLines

- Meet all the requirements of the 3NF.
- A relation is in 4NF if it has no multi-valued dependencies.



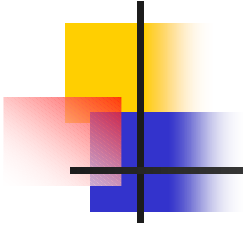
GuideLines Not Rules

It is important to point out that these are guidelines only. Occasionally, it becomes necessary to stray from them to meet business requirements.



GuideLines Not Rules contd.

When variations take place, it's extremely important to evaluate any possible problems they could have on your system and account for possible inconsistencies.

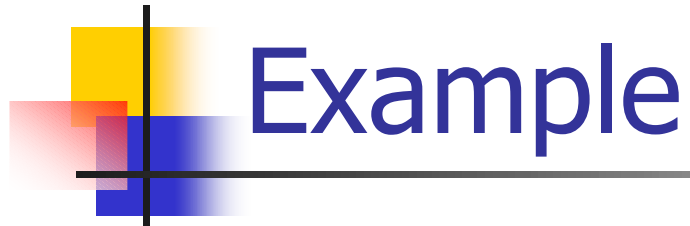


Case Study



User Story

We need to be create an address book. The system has to have the ability to contain multiple addresses, phone, web contact information.



Example

Name: Joe Smith

Addresses:

House: 1234 Some St., West Jordan, UT, 84088

Work: 5555 Where St. Salt Lake City, UT, 99999

Mailing: PO Box 333, West Jordan, UT 84088



Example cont'd

Phone:

Work: 801-123-5555

Cell: 801-124-5555

Home: 801-152-5555

Pager: 801-222-5555

Fax: 800-555-1212

Web:

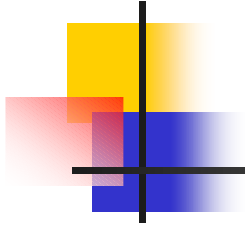
Work: JSmith@Bob.com

Personal: JS@yahoo.com

FTP: ftp:\\joebob.com

ICQ: 1234456

MSN: js@msn.com



Start Designing



Initial Data Structure

AddressBook	
	LastName
	FirstName
	AddressHomeStreet
	AddressHomeStreet2
	AddressHomeCity
	AddressHomeState
	AddressHomeZip
	AddressWorkStreet
	AddressWorkStreet2
	AddressWorkCity
	AddressWorkState
	AddressWorkZip
	AddressMailStreet
	AddressMailStreet2
	AddressMailCity
	AddressMailState
	AddressMailZip
	PhoneWork
	PhoneCell
	PhoneHome
	PhonePager
	PhoneFax
	WebWorkEmail
	WebPersonalEmail
	WebFTP
	WebPersonalURL
	WebWorkURL
	WebICQ
	WebMSN



Migrate from Initial to 1NF

- Check to see if there are duplicate columns
- Create separate tables for each different group
- Create a primary key for each group



#1: Check Duplicate Columns

AddressBook	
	LastName
	FirstName
	AddressHomeStreet
	AddressHomeStreet2
	AddressHomeCity
	AddressHomeState
	AddressHomeZip
	AddressWorkStreet
	AddressWorkStreet2
	AddressWorkCity
	AddressWorkState
	AddressWorkZip
	AddressMailStreet
	AddressMailStreet2
	AddressMailCity
	AddressMailState
	AddressMailZip
	PhoneWork
	PhoneCell
	PhoneHome
	PhonePager
	PhoneFax
	WebWorkEmail
	WebPersonalEmail
	WebFTP
	WebPersonalURL
	WebWorkURL
	WebICQ
	WebMSN

**None
Found**



#2: Create Separate Tables

Persons	
	FirstName LastName

Address	
	HomeStreet HomeStreet2 HomeCity HomeState HomeZip WorkStreet WorkStreet2 WorkCity WorkState WorkZip MailStreet MailStreet2 MailCity MailState MailZip

Phones	
	Work Cell Home Pager Fax

Web	
	WorkEmail PersonalEmail FTP PersonalURL WorkURL ICQ MSN



#3: Primary Key

Persons	
PK	ID
	FirstName LastName

Address	
PK	ID
	HomeStreet HomeStreet2 HomeCity HomeState HomeZip WorkStreet WorkStreet2 WorkCity WorkState WorkZip MailStreet MailStreet2 MailCity MailState MailZip

Phones	
PK	ID
	Work Cell Home Pager Fax

Web	
PK	ID
	WorkEmail PersonalEmail FTP PersonalURL WorkURL ICQ MSN



Migrate from 1NF to 2NF

- Remove subsets of data that apply to multiple rows of a table and place them in separate tables.
- Create relationships between these new tables and their predecessors through the use of foreign keys.



#1: Remove Subsets of Data

Persons	
PK	ID
	FirstName LastName

Phones	
PK	ID
	Number Ext

URLs	
PK	ID
	URLAddress

Address	
PK	ID
	Street1 Street2 City State Zip

PhoneTypes	
PK	CD
	Name

URLTypes	
PK	CD
	Name

AddressTypes	
PK	CD
	Name

#2: Create Relationships

Persons	
PK	ID
	FirstName LastName

Phones	
PK	ID
FK1	PhonetypeCD Number Ext

URLs	
PK	ID
FK1	WebTypeCD URLAddress

Address	
PK	ID
FK1	AddressTypeCD Street1 Street2 City State Zip

PhoneTypes	
PK	CD
	Name

URLTypes	
PK	CD
	Name

AddressTypes	
PK	CD
	Name

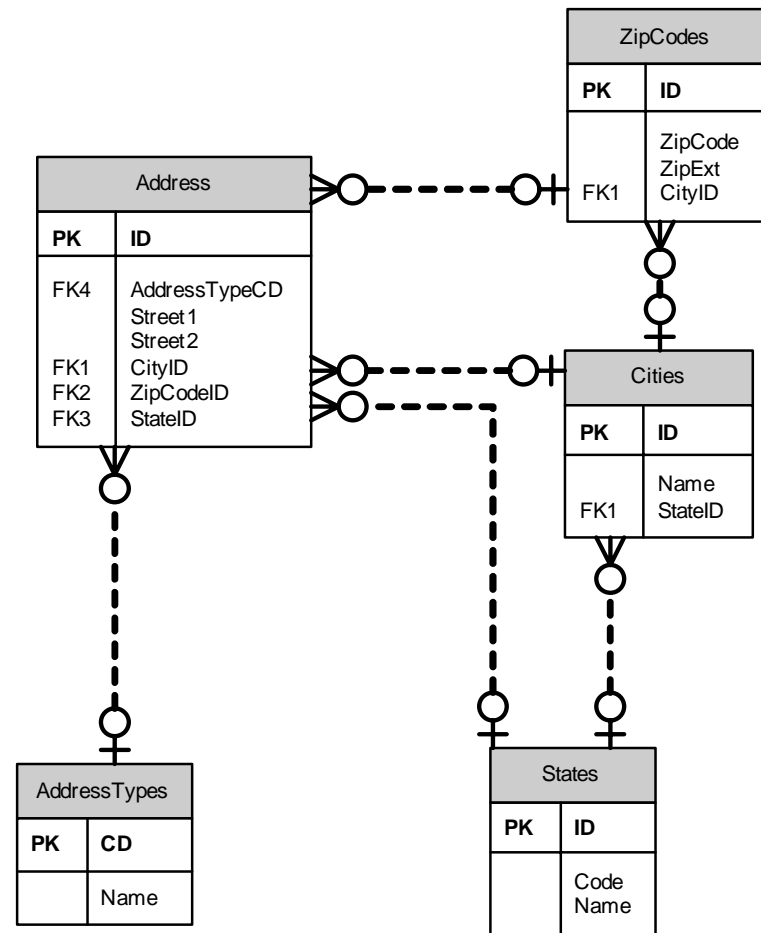
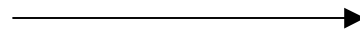
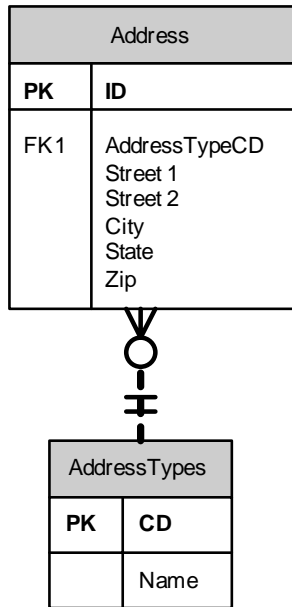




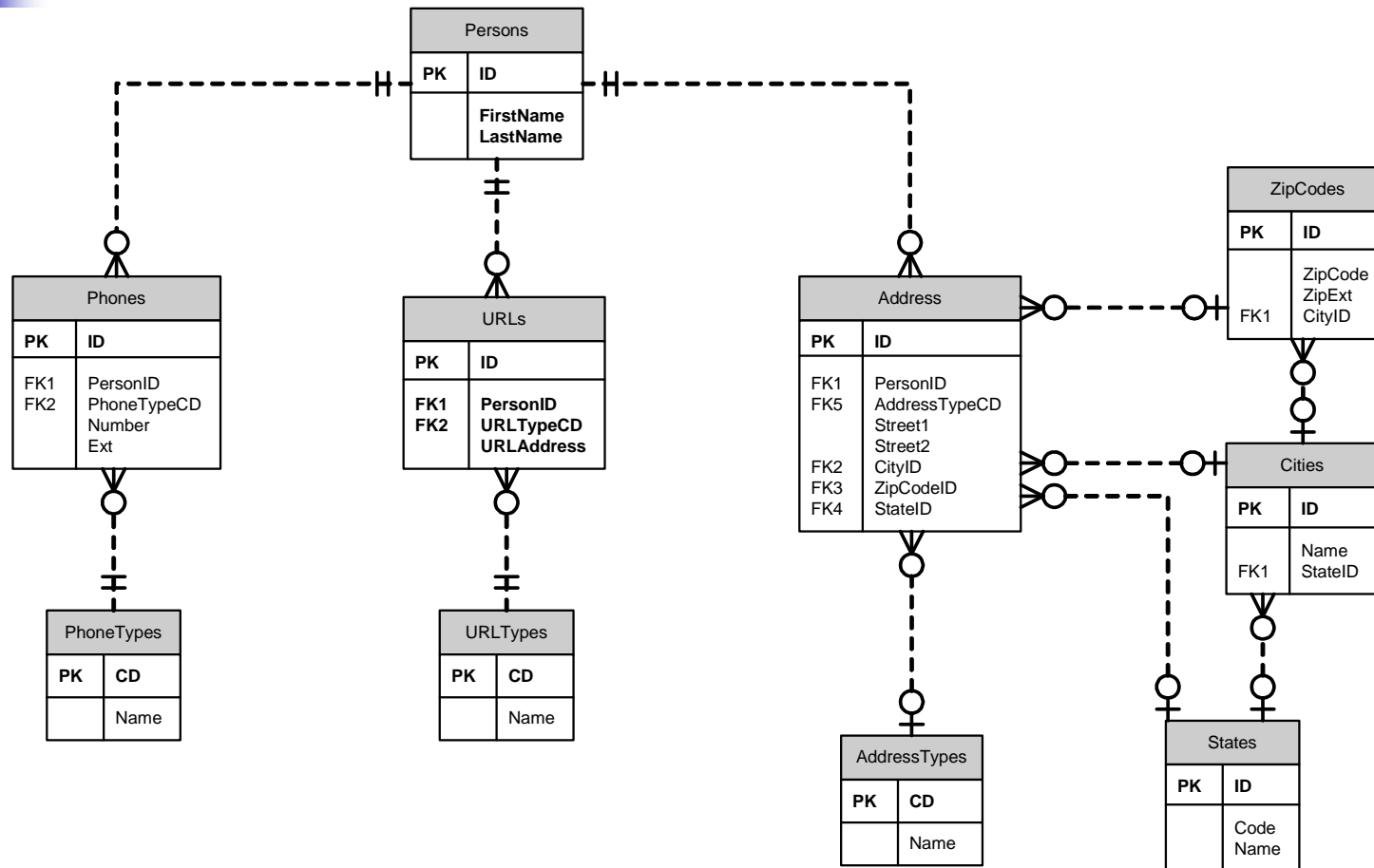
Migrate from 2NF to 3NF

- Remove columns that are not dependent upon the primary key

#1: Put Non-Dependent data in Different Table



Link Person to Contact information

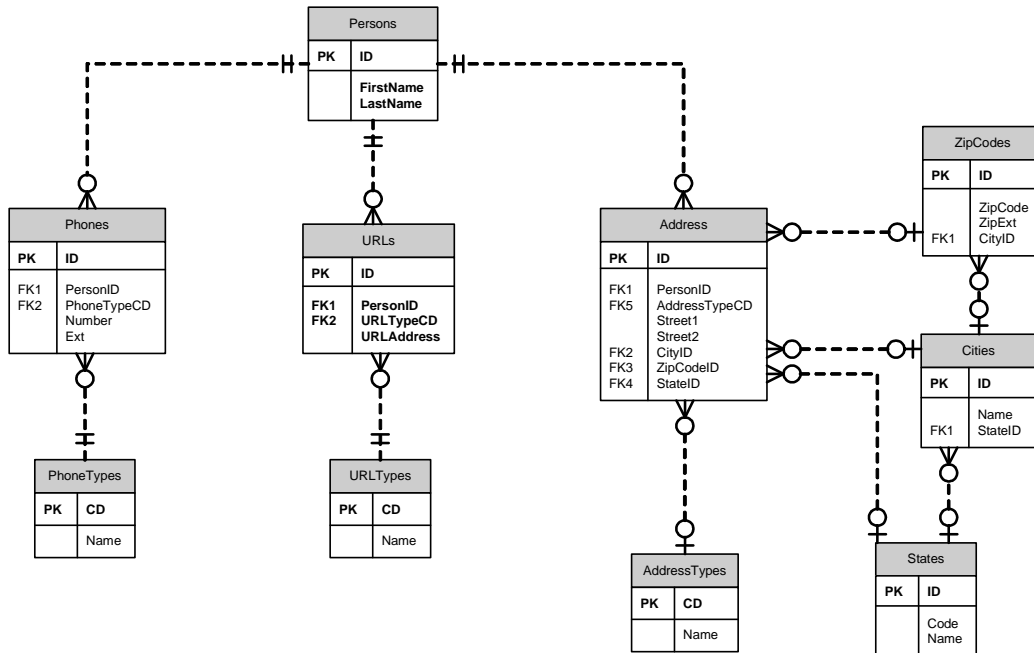




Migrate from 3NF to 4NF

- A relation is in 4NF if it has no multi-valued dependencies.

#1: No Multi-Value Dependencies

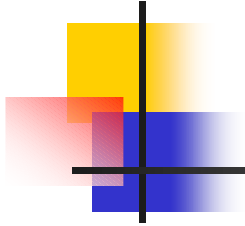


**None
Found**



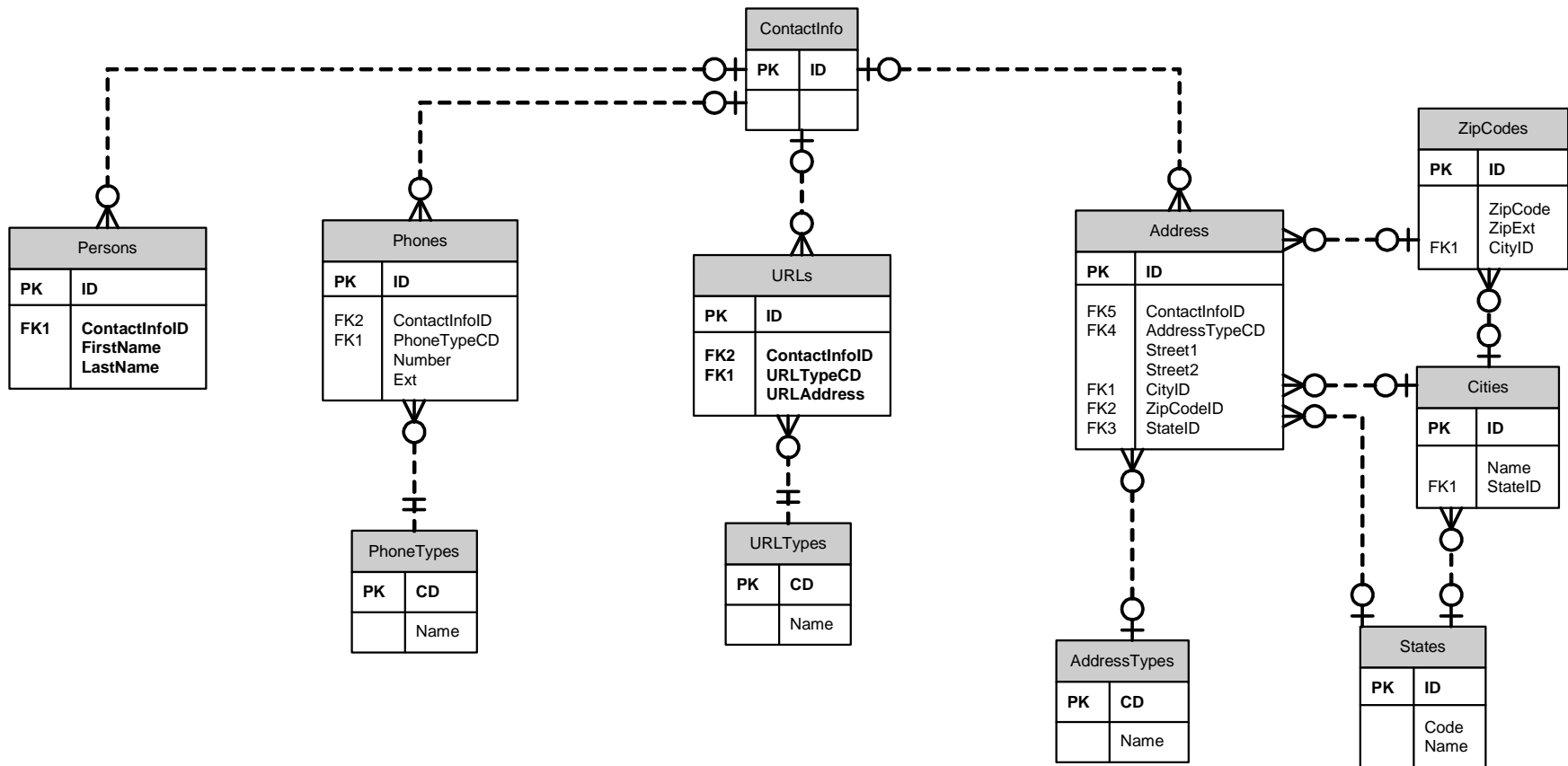
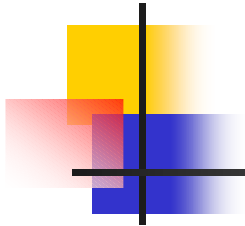
Which area is main focus

- Person
- Phone
- Email
- Address
- All the above



All of the Above

You may want to search from different view points.





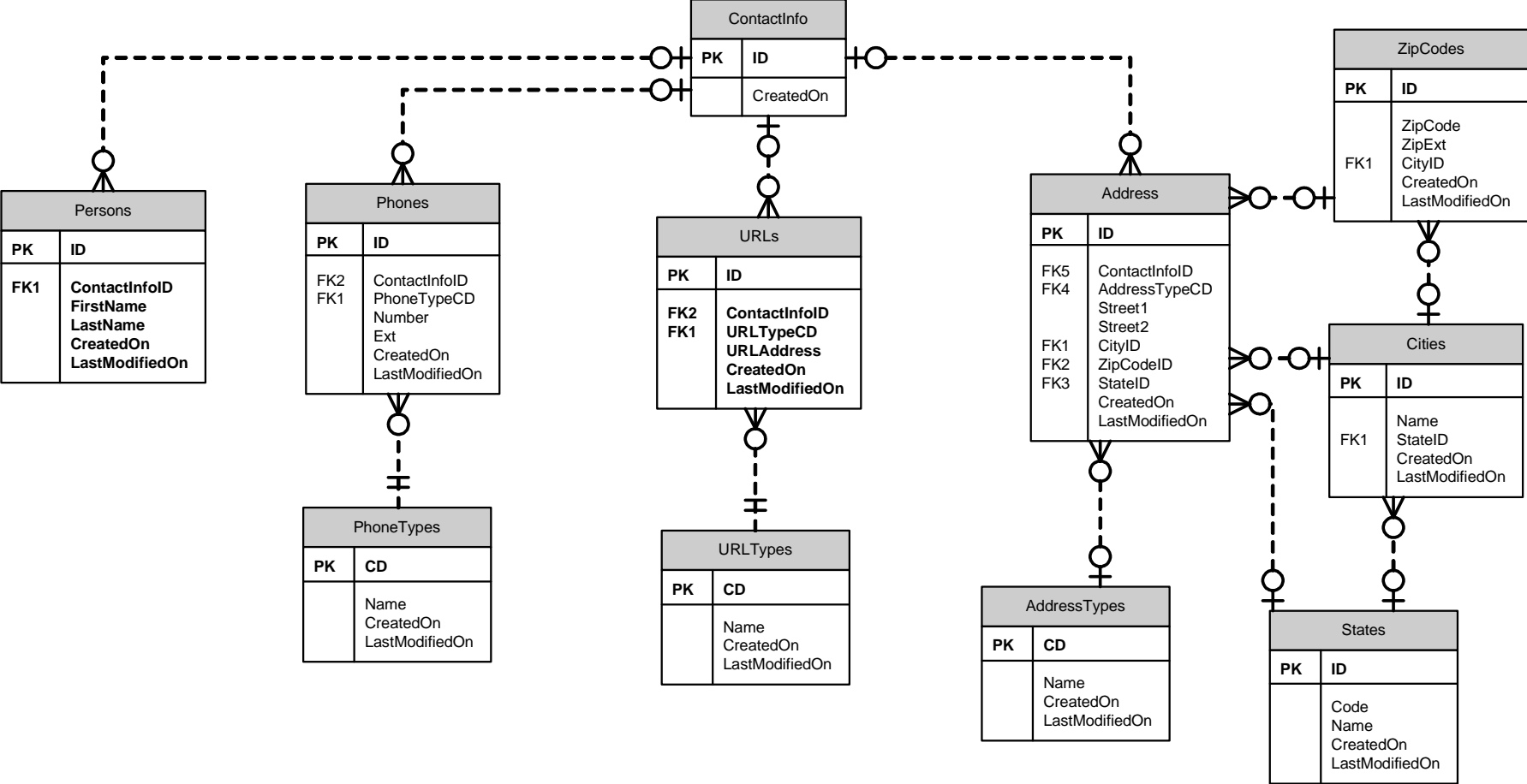
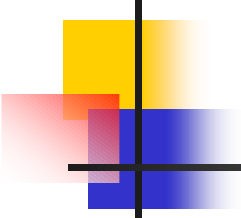
Reasons for ContactInfo Table

- All areas are collections of 1 or more items.
- You can add new items with little effect on system.
 - Company
 - Locations
 - Anything that may need an address, phone, url



Auditing Ability

- Check when the record was created.
- Check when the record was last modified.
- Conflict Resolution





Date Auditing Columns

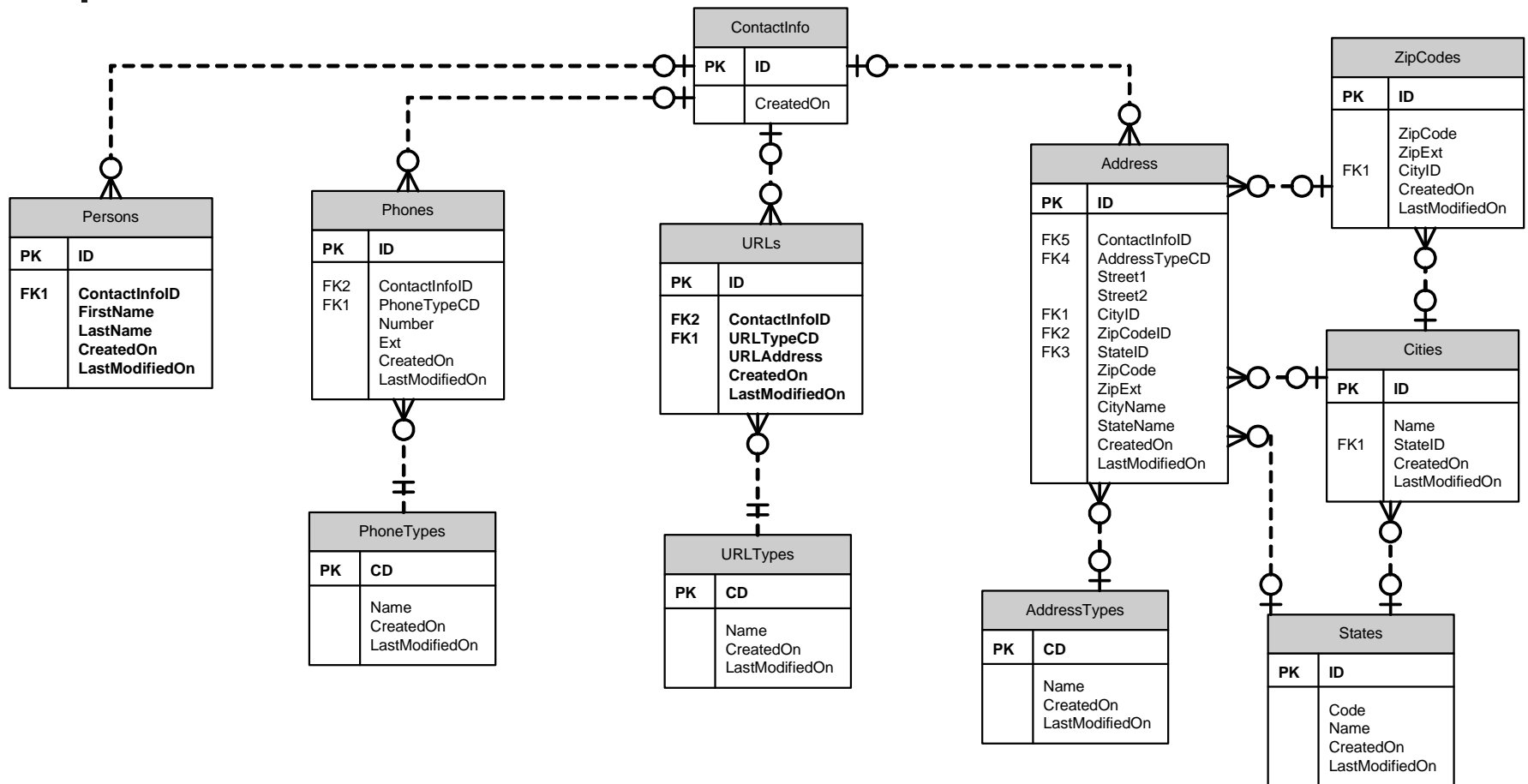
These attributes are updated by the database through the use of a trigger or default constraint.



Performance

- De-Normalization is sometimes needed
- Controlled Redundancy
 - Note: *** Be careful ***
 - Control through Triggers or database side tools
- Data Upload/Time SQL Queries

Final Rough Draft





Presentation Download

<http://www.SilverTigerGroup.com>



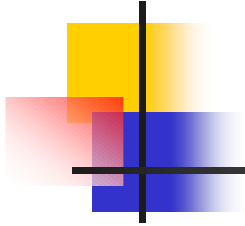
Where to go from here

- SQL
- Triggers
- Indexes
- Constraints
- Views
- Stored Procedures



Links to look at

- SQL Server Central
 - <http://www.sqlservercentral.com>
- Database Journal
 - <http://www.databasejournal.com>
- Blogs
 - <http://database.ittoolbox.com/blogs>

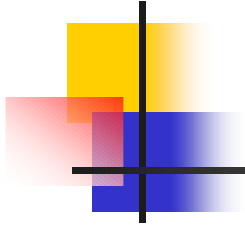


Questions & Answers



Contact Information

Email: Scott.Heffron@SilverTigerGroup.com



Thank You