



# Introduction to Data Modeling

## Using the IDEF1X Notation

Scott Lurton  
Senior Database Architect  
[slurton@data-tactics.com](mailto:slurton@data-tactics.com)



## Current Topic

### IDEF1X

#### Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

## What will IDEF1X talk about, what is it?

- Is an industry standard notation published on December 21<sup>st</sup>, 1993 by the National Institute of Standards and Technology.
  - The IDEF1X notation, what is it?
    - Data Requirements
    - Entities
- Standard for Integration Definition For Information Modeling
  - Relationships/Business Rules
- Google it as "idef1x.pdf".
  - Rules of Normalization
- Is available as a notation with several data modelling tools.
  - Transforming from Logical To Physical
  - Reverse Engineering
  - Available Tools on the Market

## Alerts

IDEF1X

### Message Text

Many people think IDEF1X is a notation for data modeling and not for integration. IDEF1X is a notation for integration of my knowledge.

## Work Area

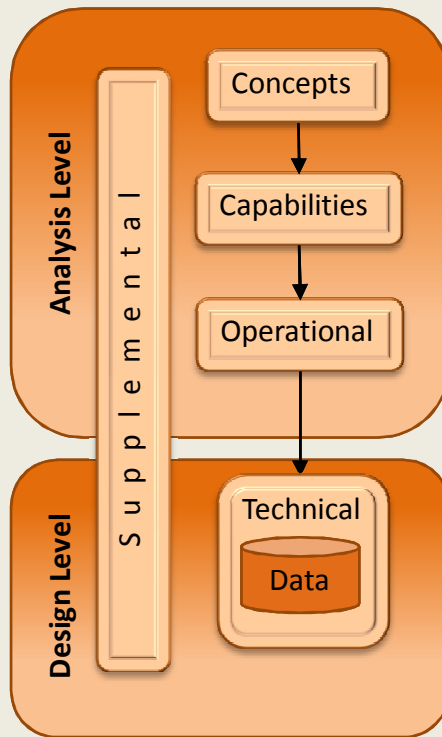
## Current Topic

Requirements

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Required IDEF1X related requirements?



Operational

**The system shall capture data about people and organizations.**



Technical Data

**The system requires an entity named Person.  
The system requires an entity named Organization.**

## Alerts

IDEF1X

Logical

## Message Text

For each level of Operational requirements, data requirements. Extrapolate data requirements are another form of what is needed to require the data with a requirement back up to the operational requirements.

## Work Area

## Current Topic

### Requirements

### Agenda

IDEF1X

Requirements

Entities

Attributes

Primary Keys

Relationships

Normalization

Transformation

Rev. Engineering

Tools

## Requirements: What is an entity?

- NOUN: 1. Something that exists as a particular and discrete unit:  
*Persons and corporations are equivalent entities under the law.*
- Person, partnership, organization, or business unit which has a legal existence, for which accounting records are kept, and about which financial statements are prepared.
- **entitydefinition** database In an **entity**-relationship model, an **entity** is a type of thing being modeled such as "person" or "product". Different **entities** have different sets ...
- **As it applies to data modeling, an Entity is anything of significance about which you want to capture data.**

## Alerts

Logical

## Message Text

## Work Area



Data Requirements

## Current Topic

### Entities

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# What is an entity?

## How do I know I have an entity? (Some rules of thumb.)

- ✓ Is the entity significant to the application system?
- ✓ Do you want to know many things about the entity?

Person



Is this an entity?

Race Car



Is this an entity?

Organization



Is this an entity?

Color



Is this an entity?

## Alerts

Logical

## Message Text

According to our  
definition of an  
entity, it is a  
requirement, but  
with the following  
requirements:  
- It is a noun  
- It is a thing  
- It is a thing that  
we want to know  
about an  
Organization.  
Also, a color is made  
up of a formula such  
as the amount of  
Red, Green and Blue  
to place in the color.  
  
Wow many things I  
want to know about  
the entity.

## Work Area



Data Requirements

Person



Organization



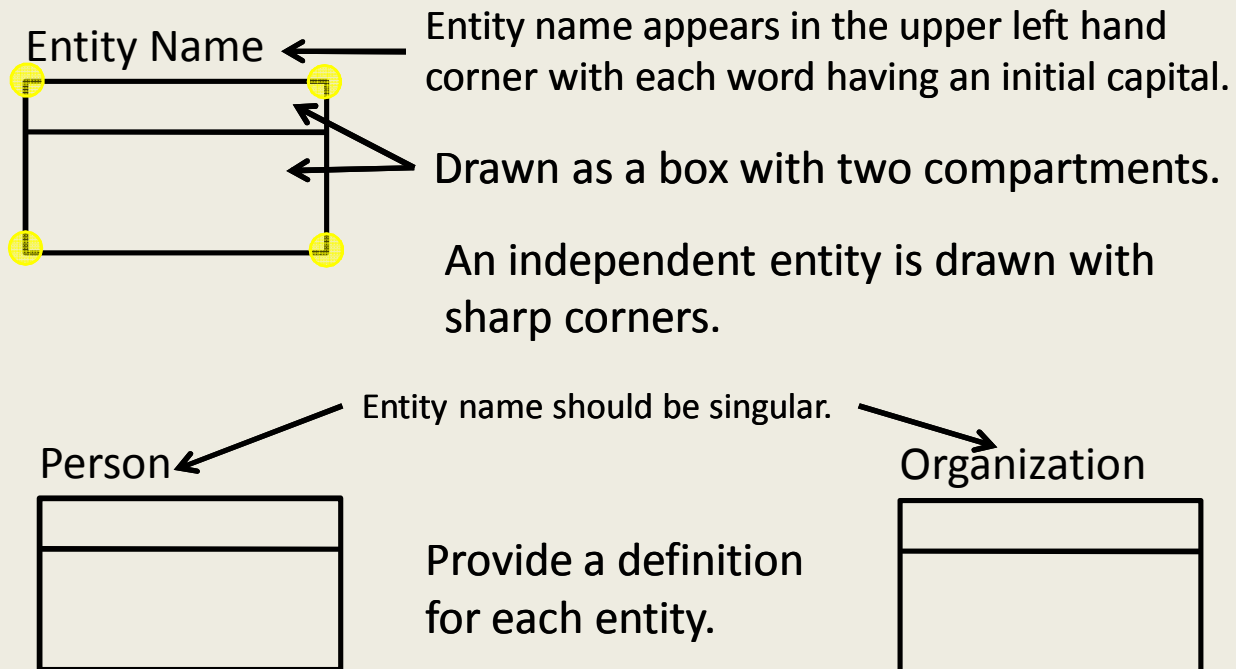
## Current Topic

### Drawing Entities

## Agenda

- IDEF1X
- Requirements
- Entities
- Attributes
- Primary Keys
- Relationships
- Normalization
- Transformation
- Rev. Engineering
- Tools

## How is an entity drawn?



## Alerts

IDEF1X

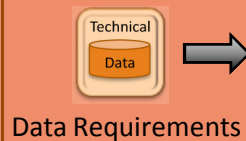
Logical

## Message Text

The IDEF1X standard does not provide for the drawing of the entity name in the place of a space.

The standard does not specifically call out how the entity name should be represented, however this is usually the standard in other notations.

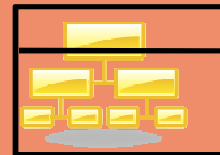
## Work Area



Person



Organization



Dependant Entities

## Current Topic

Drawing  
Attributes  
an Entity

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# What is an attribute?

“An abstraction belonging to or characteristic of an entity.”

If an entity is a Person, the attributes would be what you want to know about that Person.

## Keep this in mind when creating attributes.

- ✓ An attribute should not have internal components.
- ✓ An attribute should not hold a derived value.
- ✓ An attribute name should not exceed the limit of the DB engine.
- ✓ An attribute name must start with a letter and not a number.
- ✓ The name of the attribute should reflect the content of the attribute.

## Alerts

Logical

## Message Text

When attributes  
have internal  
components  
they are not  
logical. Only  
logical attributes  
are allowed.  
Derived values  
should not be  
queried or  
modified.  
A new type  
of column in  
11gR2 called  
an attribute  
column which  
will internally  
derive its  
values for you.  
discovered another  
Entity.

## Work Area



Data Requirements

Person



Organization



Dependant Entities

## Current Topic

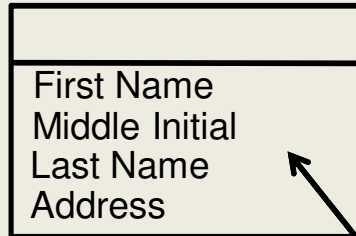
### Drawing an Attribute

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

## How is an attribute drawn?

### Person

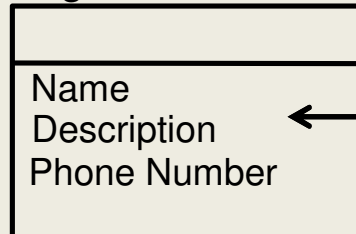


Each word in the attribute is separated by a space.

Each word in the attribute should have an initial capital.

Initially the attributes are placed in the bottom compartment.

### Organization



## Alerts

IDEF1X

Logical

## Message Text

Oops is actually another entity. Did it again.

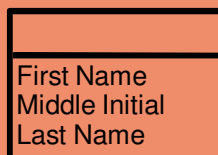
A phone number also has internal components.

## Work Area

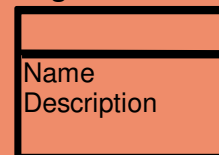


Data Requirements

### Person



### Organization



Dependant Entities

## Current Topic

Attributes  
Definitions

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# How Design Attributes?

Now that we have decided on our attributes we now need to complete their definition.

Step 1: Provide a Definition

Step 2: Decide on a data type and length.

Step 3: Is it mandatory or optional?

First Name	A name that is given to a person at birth. varchar2(100) Mandatory
Middle Initial	The first character of a person's middle name. varchar2(1) Optional
Last Name	A person's family name. varchar2(100) Mandatory

## Alerts

IDEF1X

Logical

## Message Text

## Work Area



Data Requirements

Person

First Name Middle Initial Last Name

Organization

Name Description

Dependant Entities

## Current Topic

Attribute  
Primary Keys  
Definition

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Defining an Attribute

Now that our attributes have good definitions we need to decide on a Primary Key

## Rules that apply to Primary Keys.

- Rule 1: One or more attributes can participate in a Primary Key.
- Rule 2: Any attribute that participates in a Primary Key **MUST** be mandatory.
- Rule 3: Primary Key query returns one record.

### Mandatory attributes from Person and Organization.

Person	<u>Person PK</u>	Organization PK	Organization					
<table border="1"><tr><td> </td></tr><tr><td>First Name</td></tr><tr><td>Last Name</td></tr></table>		First Name	Last Name	First Name Scott Lurton Last Name Adam Smith Last Name Sally May Last Name Smith	Name	<table border="1"><tr><td> </td></tr><tr><td>Name</td></tr></table>		Name
First Name								
Last Name								
Name								
	<b>We have a Problem</b>	<b>No problem We have a Natural Key</b>						

## Alerts

Logical

## Message Text

A Natural Key is a Primary Key which is made up of attributes that are unique across every record available in the Primary Key.

If you query a record using the value(s) in the Primary Key, only one record should ever come back from the database.

## Work Area



Data Requirements

First Name
Middle Initial
Last Name

Name
Description

Dependant Entities

## Current Topic

Define  
Primary Keys  
Surrogate Key

### Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Surrogate Keys

When we have a problem establishing a Primary Key we may need to create an artificial key instead.

Id

Step 1: Create another attribute call it **Id**.

Step 2: Data type of new attribute would be numeric.

Step 3: Attribute must be mandatory.

Step 4: Create a sequence generator.

**Primary key  
(Surrogate Key)**

**Sequence Generator**

Person

First Name
Last Name

**Person PK**

First Name

Last Name

**We have a Problem.**

**Id**



The sequence generator provides a surrogate value for the Primary Key .

## Alerts

Logical

### Message Text

Only mandatory attributes can be included in a Primary Key.

## Work Area



Data Requirements



Person

First Name
Middle Initial
Last Name

Organization

Name
Description

Dependant Entities

## Current Topic

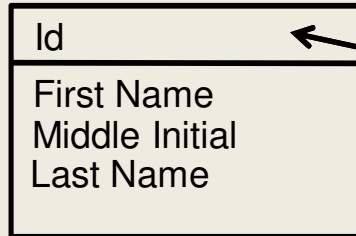
Define  
Primary Keys  
Surrogate Key

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Drawing a Primary Key

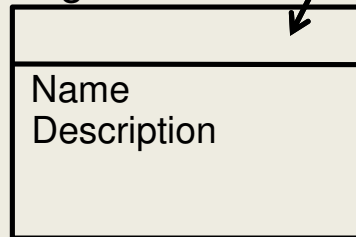
Person



**The Primary Key is placed in the top compartment.**

**If needed many attributes can be placed in the top compartment.**

Organization



## Alerts

IDEF1X

Logical

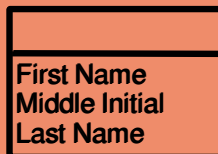
## Message Text

## Work Area

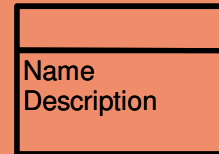


Data Requirements

Person



Organization



Dependant Entities

## Current Topic

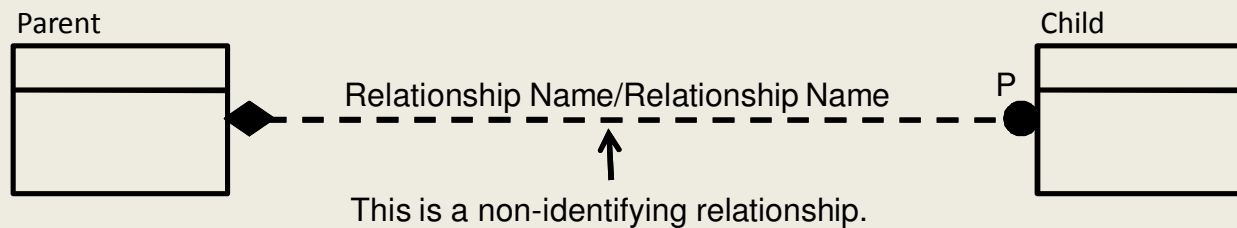
### Anatomy of Relationships

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The Anatomy of a Relationship

A relationship consists of the follow parts.



A line between the parent entity and the child entity.

A dot is placed on the child side of the relationship.

Additional annotations are used to represent the type of relationship.

A Diamond symbol is sometimes used to indicate optionality.

A relationship name is placed in between the Parent and Child entities.

## Alerts

IDEF1X

Logical

## Message Text

A diamond on the parent side of a relationship indicates that the foreign key on the child side of the relationship should be optional.

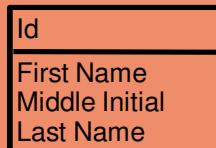
Separated by a /.

## Work Area

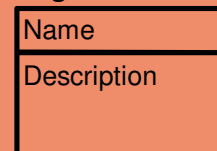


Data Requirements

Person



Organization



Dependant Entities

Identifying Relationship





## Current Topic

Validation  
One-To-Many  
Sentence

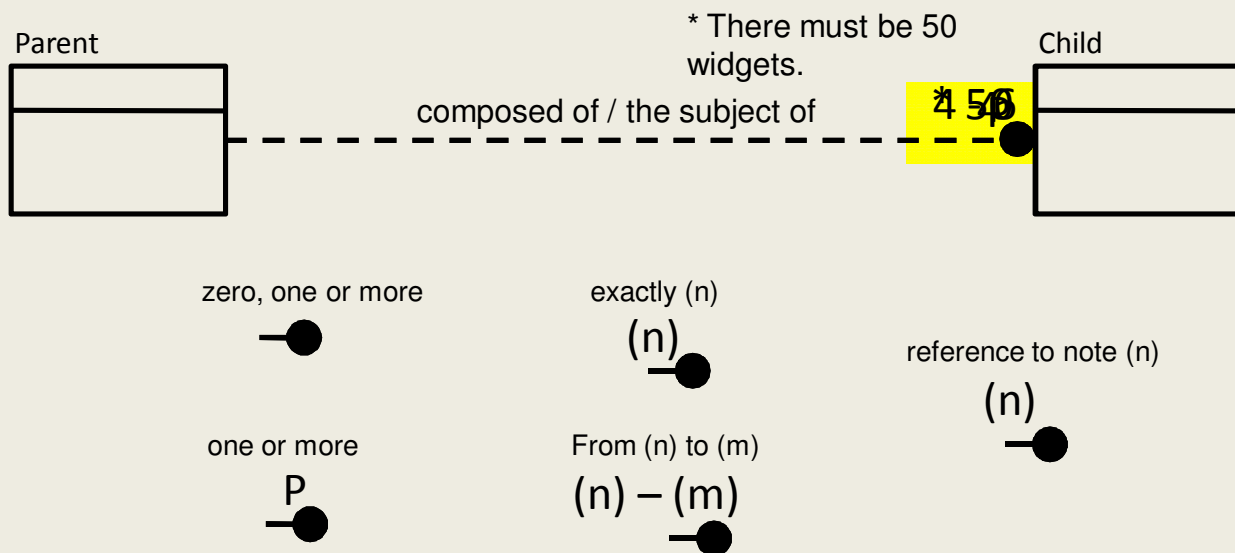
## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The One-To-Many Relationship

The One-To-Many relationship is the most common.

The One-To-Many comes in many forms.



## Alerts

IDEF1X

Logical

## Message Text

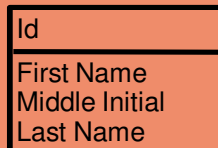
**For every record in the Parent entity there must be exactly 50 records associated with the Child entity.**

## Work Area

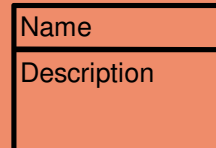


Data Requirements

Person



Organization



Dependant Entities

Identifying Relationship



## Current Topic

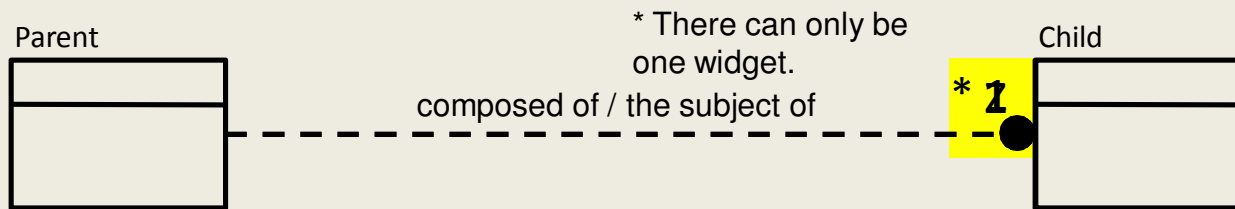
One-To-Many

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The One-To-One Relationship

The One-To-One relationship is not so common.  
The One-To-One is used for special situations.



Zero or one



exactly (n)



Reference to note \*.



## Alerts

IDEF1X

Logical

## Message Text

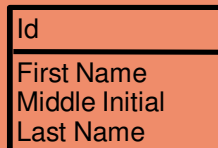
For every record in the Parent entity there Must be exactly one associated records in the Child entity.

## Work Area

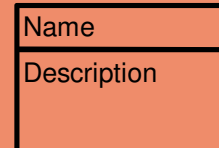


Data Requirements

Person



Organization



Dependant Entities

Identifying Relationship



## Current Topic

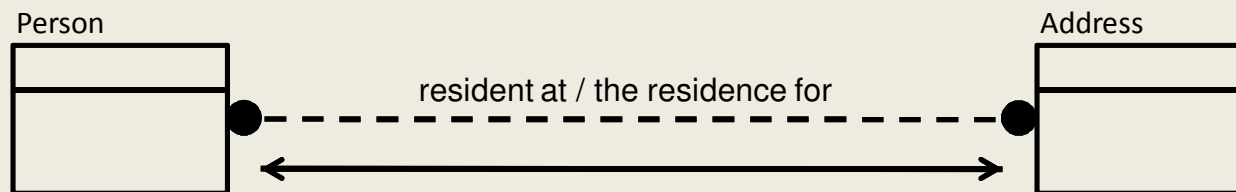
### Many-To-Many

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The Many-To-Many Relationship

The Many-To-Many relationship is much more common than people think.  
Initially the validation sentence is used to discover this relationship.



Each and every Person may be resident at zero, one or more addresses.

TRUE?

Each and every Address may be the residence for one and only one Person.

FALSE?

Each and every Address may be the residence for zero, one or more People.

TRUE

## Alerts

IDEF1X

Logical

## Message Text

Most people start out with a One-To-Many relationship between Parent and Child and then discover that the relationship between Child and Parent is a One-To-Many.

## Work Area



Data Requirements

Person

Id  
First Name  
Middle Initial  
Last Name

Organization

Name  
Description

Dependant Entities

Identifying Relationship



## Current Topic

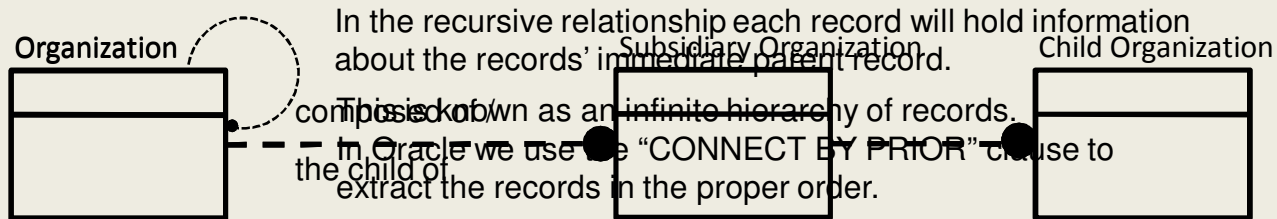
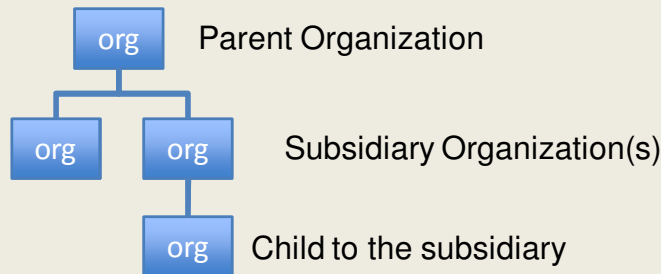
Many To Many

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The Recursive Many Relationship

The recursive relationship is addressing the following situation.



## Alerts

IDEF1X

Logical

## Message Text

A recursive relationship can be any type of standard relationship, however, the catch is that the Parent entity is also the Child entity.

## Work Area



Data Requirements

Person

Id  
First Name  
Middle Initial  
Last Name

Organization

Name  
Description

Dependant Entities

Identifying Relationship

## Current Topic

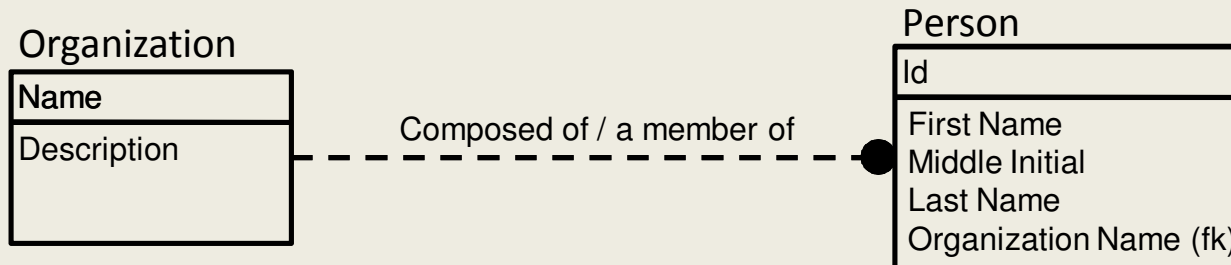
Putting it all  
Recursive  
together

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# The Primary Relationship

Here is how we draw a foreign key attribute in another entity.



Step 1: Take the Primary Key from the one side of the relationship.

Step 2: Make a copy of the Primary Key as a new attribute on the many or child side of the relationship with the dot.

Step 3: Name the foreign key attribute an appropriate name.

Step 4: Append (fk) to the end of the attribute.

## Alerts

IDEF1X

Logical

## Message Text

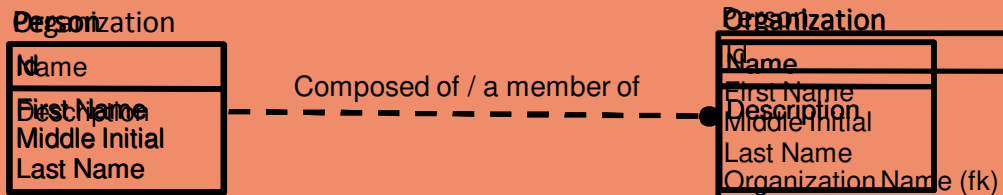
The Reason we kept the key from the side of the relationship was to get the Primary Key on the many side of the relationship line.

AWAYS!!!

## Work Area



Data Requirements



## Dependant Entities

Identifying Relationship



## Current Topic

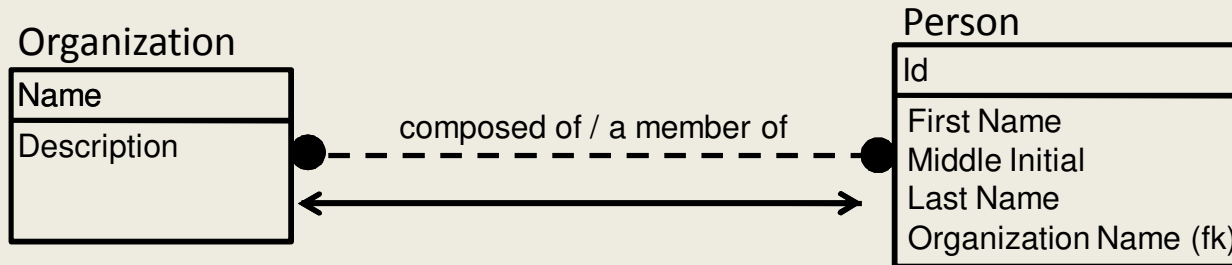
Putting it all  
continued...  
together

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Putting it all together continued...

Let's validate our Organization/Person relationship.



Each and every Organization may be composed of zero one or more People.

**Sounds reasonable**

Each and every person may be a member of one and only one Organization.

**Nope, I want to capture all of the Organizations a Person is a member of.**

Each and every person may be a member of zero, one or more Organizations.

**That's' better.**

## Alerts

IDEF1X

Logical

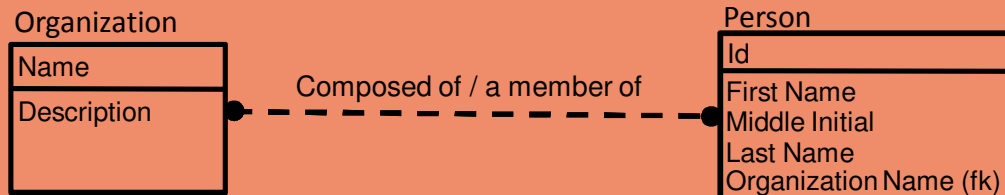
## Message Text

So we should  
now validate  
our new  
relationship.

## Work Area



Data Requirements



Dependant Entities

Identifying Relationship



## Current Topic

Many-To-Many  
Resolution

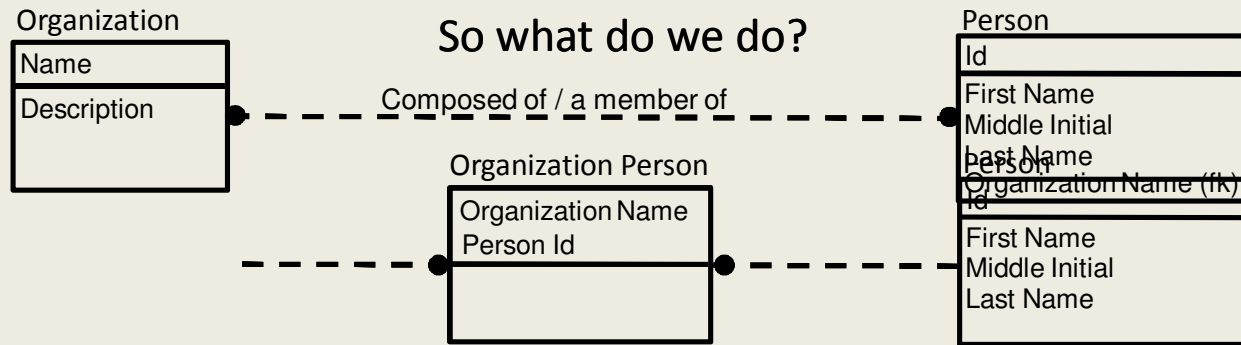
## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Resolving a Many-To-Many relationship.

A Many-To-Many cannot be physically implemented.

So what do we do?



- Step 1: Create another entity called **Organization Person**.
- Step 2: Copy the Primary Key from the Organization entity to the Primary Key for the Organization Person.
- Step 3: Copy the Primary Key from the person entity to the Primary Key for the Organization Person.
- Step 4: Remove the Many-To-Many relationship.
- Step 5: Turn the relationship from Organization around so that the many side is on Organization Person.
- Step 6: Turn the relationship from Person around so that the many side is on Organization Person.

## Alerts

IDEF1X

Logical

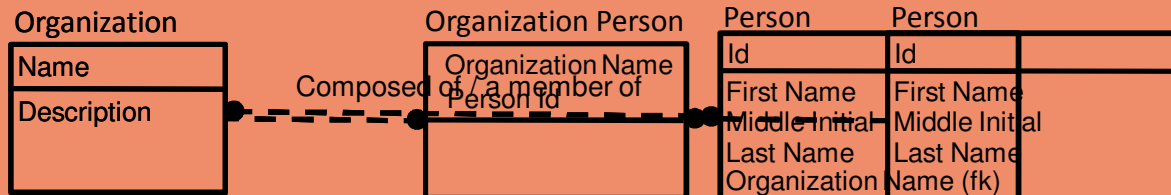
## Message Text

Database Foreign Key constraints will prevent the implementation of a Many-To-Many relationship.

## Work Area



Data Requirements



## Dependant Entities

Identifying Relationship



## Current Topic

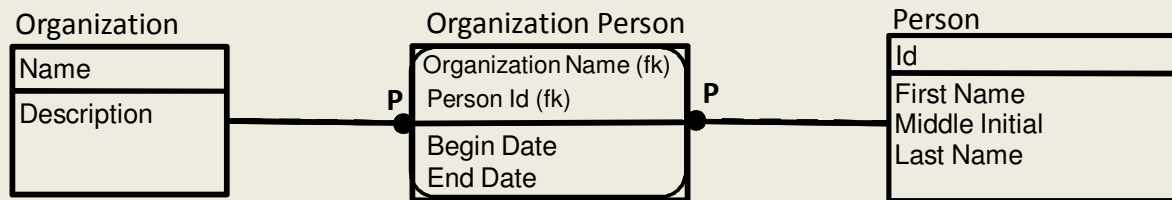
Many-To-Many  
continued...  
Resolution

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Resolving a Many-To-Many relationship.

To finish the resolution of the Many-To-Many we need to...



Take the non-identifying relationship and turn it into an identifying relationship.

By definition the Primary Key must be mandatory.

The contents of the new entity is now dependent on the other two entities.

Next ask the question: What do you want to know about the relationship between person and Organization?

## Alerts

IDEF1X

Logical

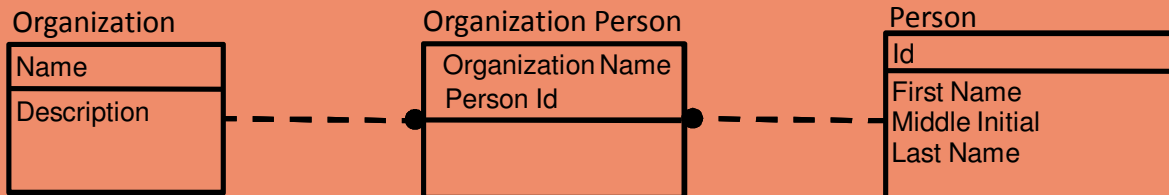
## Message Text

The current relationship is a **zero, one or more** it must be turned into a **one or more**.

## Work Area



Data Requirements



Dependant Entities

Identifying Relationship

## Current Topic

Normalization

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Rules of Normalization

[http://en.wikipedia.org/wiki/Database\\_normalization](http://en.wikipedia.org/wiki/Database_normalization)

The **normal forms** (abbrev. **NF**) of relational database theory provide criteria for determining a table's degree of vulnerability to logical inconsistencies and anomalies.

## First Normal Form (1NF)

Table faithfully represents a [relation](#) and has no *repeating groups*

## Second Normal Form (2NF)

No non-prime attribute in the table is [functionally dependent](#) on a [proper subset](#) of a [candidate key](#)

## Third Normal Form (3NF)

No non-prime attribute in the table is [functionally dependent](#) on a [proper subset](#) of a [candidate key](#)

## Alerts

IDEF1X

Logical

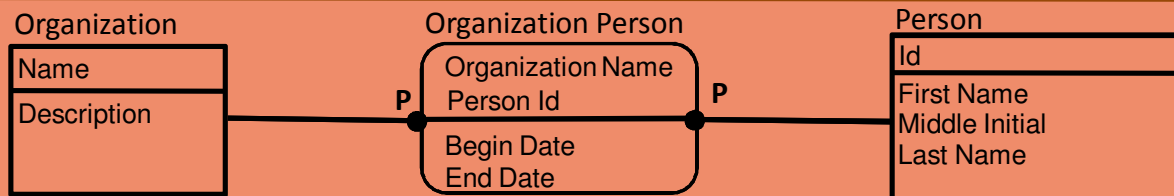
## Message Text

The rules of normalization are used to ensure that we don't mess up our database.

## Work Area



Data Requirements



## Current Topic

Normalization

## Agenda

IDEF1X

Requirements

Entities

Attributes

Primary Keys

Relationships

Normalization

Transformation

Rev. Engineering

Tools

# First Normal Form (1NF)

## First Normal Form (1NF)

Table faithfully represents a [relation](#) and has no *repeating groups*

**No repeating groups.**

So how do we fix it?

Organization

Name
Description
Address1
City 1
State 1
Zipcode 1
Address2
City 2
State 2
Zipcode 2

Address

Address id
Organization Name (fk)
Street Name
Location Number
Attention Line
City
State
Zipcode

What happens if there is a third address that you want to capture?

Do you recode the application system to account for the third address?

## Alerts

Logical

## Message Text

When you have several attributes which share the same name and definition but have a slight difference such as a number at the end, this is a repeating group.

## Work Area



Data Requirements

Organization

Name
Description

P

Organization Person

Organization Name
Person Id
Begin Date
End Date

P

Person

Id
First Name
Middle Initial
Last Name

## Current Topic

2NF

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Second Normal Form, Decomposed

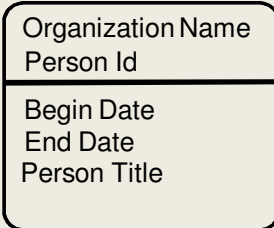
## Second Normal Form (2NF)

No non-prime attribute in the table is functionally dependent on a proper subset of a candidate key

Must be in 1NF before it can be in 2NF.

If the entity has two attributes in it's Primary Key then all non Primary Key attributes must be dependent on all attributes in the Primary Key.

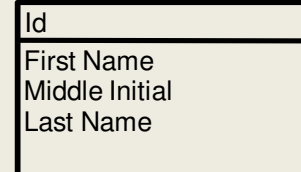
Organization Person



So how do we fix it?

Person Title is a misplaced attribute.

Person



~~The entity is not 2NF.~~

The attributes must be dependent on the key, the whole key....

## Alerts

Logical

## Message Text

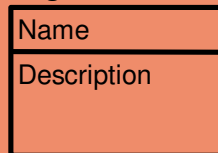
If there is only one attribute in the Primary Key then the entity is dependent on the Primary Key automatically in second normal form.  
Organization.

## Work Area



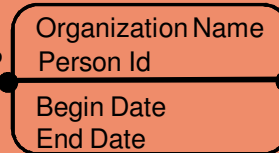
Data Requirements

Organization



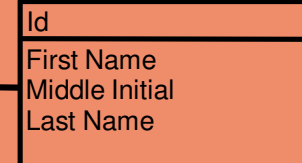
P

Organization Person



P

Person



## Current Topic

3NF

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

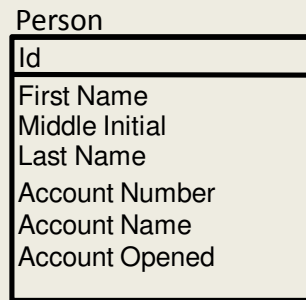
# Third Normal Form Demystified

## Third Normal Form (3NF)

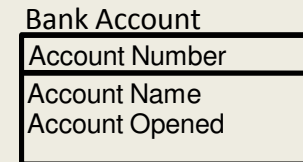
No non-prime attribute in the table is [functionally dependent](#) on a [proper subset](#) of a [candidate key](#)

Must be in 1NF and 2NF before it can be in 3NF.

No two attributes can be completely dependent on each other.



So how do we fix it?



The entity is not 3NF.

The attributes must be dependent on the key, the whole key and nothing but the key, so help me Cod.

## Alerts

Logical

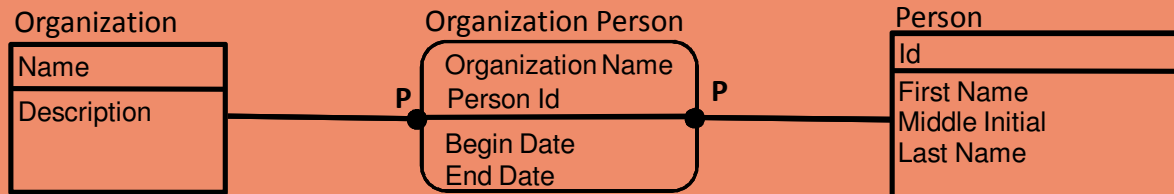
## Message Text

The attributes for a person's bank account are dependent on each other and should belong in another entity.

## Work Area



Data Requirements



## Current Topic

Transformation

## Agenda

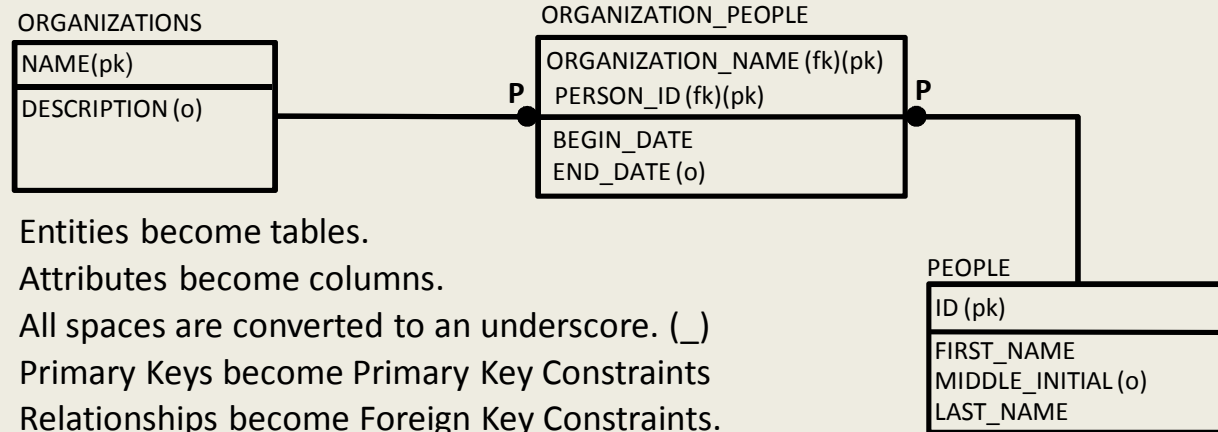
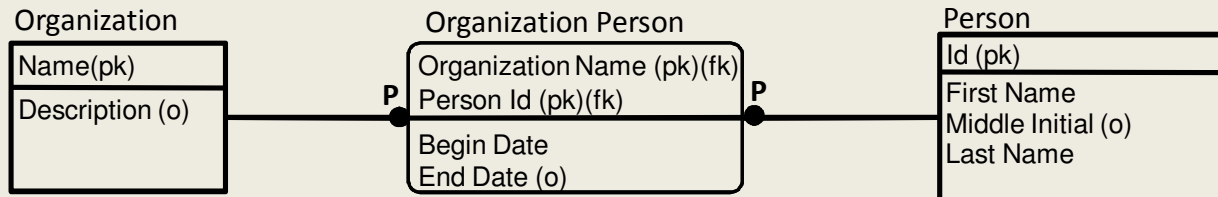
IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Transformation from Logical to Physical

## Alerts

Physical

## Message Text



Entities become tables.

Attributes become columns.

All spaces are converted to an underscore. ( \_ )

Primary Keys become Primary Key Constraints

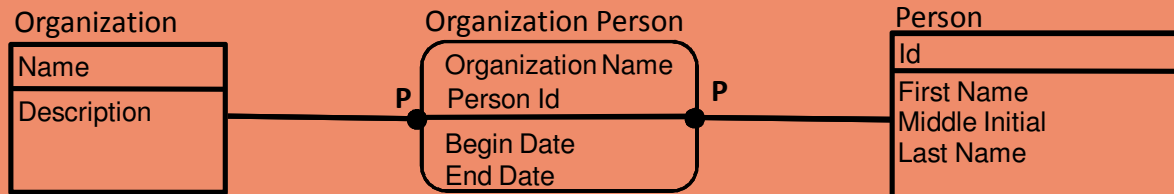
Relationships become Foreign Key Constraints.

Optional columns have an (o). Mandatory columns have nothing.

## Work Area



Data Requirements



## Current Topic

Reverse Transformation Engineering

## Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Reverse Engineering Physical

Many projects start up by reverse engineering an existing database.

Most if not all data modeling tools provide the ability to reverse engineer an existing database.

Some modeling tools will then allow a physical design to be retrofitted to a logical design.

## Alerts

Physical

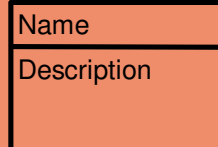
## Message Text

## Work Area

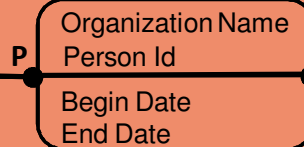


Data Requirements

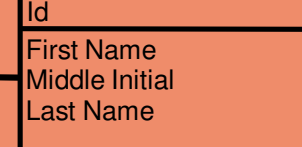
Organization



Organization Person



Person



P

P

## Current Topic

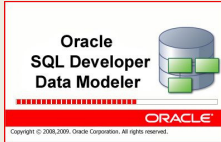
Reverse  
Engineering  
Tools

## Agenda

IDE1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Available Data Modeling Tools

ORACLE



[www.oracle.com/technetwork/developer-tools/datamodeler](http://www.oracle.com/technetwork/developer-tools/datamodeler)

embarcadero



[www.embarcadero.com/products/er-studio](http://www.embarcadero.com/products/er-studio)



[erwin.com/](http://erwin.com/)

SYBASE | An SAP Company

## PowerDesigner

[www.sybase.com/products/modeling-development/powerdesigner](http://www.sybase.com/products/modeling-development/powerdesigner)



[www.codebydesign.com/SystemArchitect/downloads/](http://www.codebydesign.com/SystemArchitect/downloads/)

## Alerts

Physical

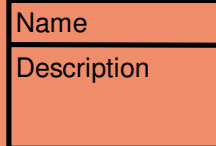
## Message Text

## Work Area



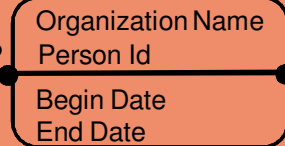
Data Requirements

Organization



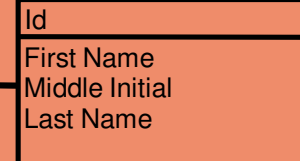
P

Organization Person



P

Person



## Current Topic

### Questions

### Agenda

IDEF1X  
Requirements  
Entities  
Attributes  
Primary Keys  
Relationships  
Normalization  
Transformation  
Rev. Engineering  
Tools

# Questions?

## Alerts

Physical

## Message Text

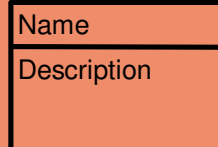


## Work Area



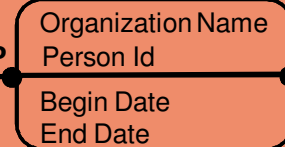
Data Requirements

### Organization



P

### Organization Person



P

### Person

