

# Database Management System

October 2014,  
Reviewed: July 2015

# Agenda

- *Definitions*
  - ✓ *What is Database*
  - ✓ *What is DBMS*
- *Database Models*
- *Entity Relationship Diagram*
- *Database Objects*
- *General Data Types*



# Definitions

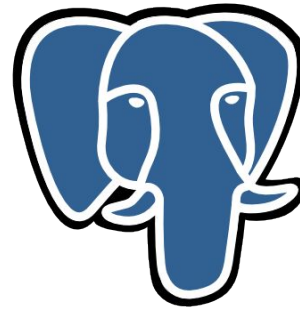
# What is database

- A database is an organized collection of data.
- A database is a collection of information that is organized so that it can easily be accessed, managed, and updated.

# What is DBMS

- Database management systems (DBMSs) are specially designed software applications that interact with the user, other applications, and the database itself to capture and analyze data.
- E.g.: MySQL, PostgreSQL, Microsoft SQL Server, Oracle etc.

# Logos

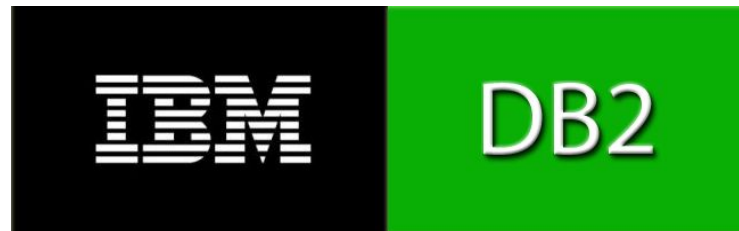


PostgreSQL



Microsoft®  
SQL Server®

**ORACLE®**  
DATABASE



# DBMS purpose

- To store data properly
- To provide simultaneous access to the data for many users
- To delimit the access to the data for different users
- To prevent data from loss

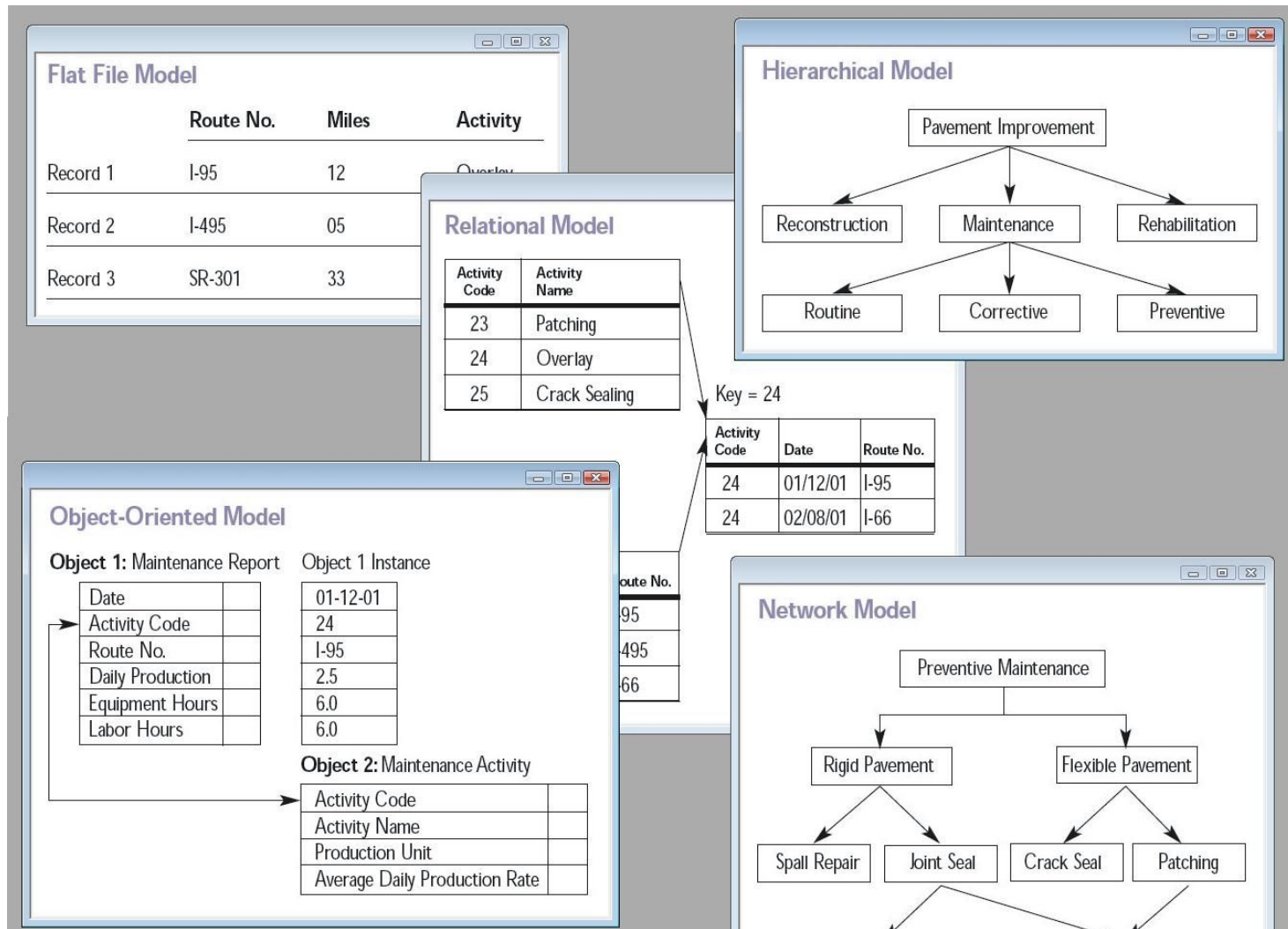
# Database Models



# Relational and non-relational DBs

- A database model is a type of data model that determines the logical structure of a database and fundamentally determines in which manner data can be stored, organized, and manipulated.
- ✓ **Relational model** (the eldest and the most popular),
- ✓ **object-oriented**,
- ✓ **document**,
- ✓ **hierarchical**,
- ✓ **network model** etc.
- Basing on those models, there are relational and non-relational DBs and DBMSs

# Database models



# Relational model

- Data is stored in tables called relations.
- Relations can be normalized.
- In normalized relations, values saved are atomic values.
- Each row in relation contains unique value
- Each column in relation contains values from a same domain

# RDBMS

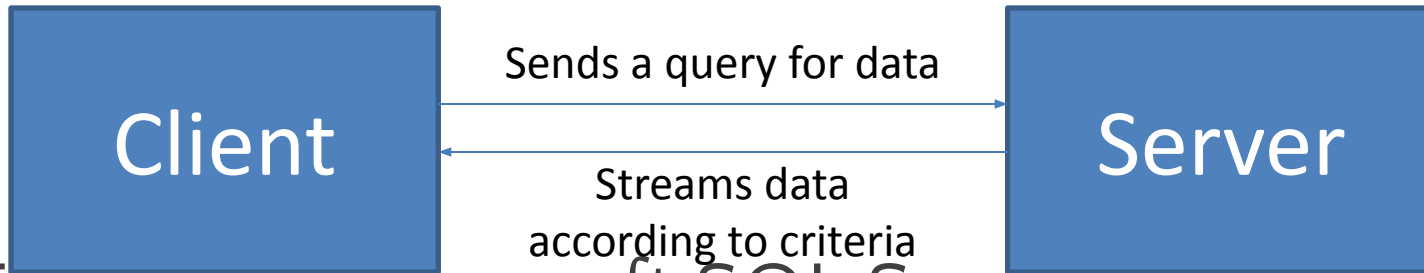
- Implement a relational data model
- Are used in most commercial projects
- Have been used for almost 40 years

# Desktop RDBMS

- Store data on local PC or network file storage
- Data processing is performed on the local PC
- Are used for small applications, mostly single-user, without strong security requirements
- Example: Microsoft Access

# Client-server RDBMS

Consist of two components: **client** and **server**

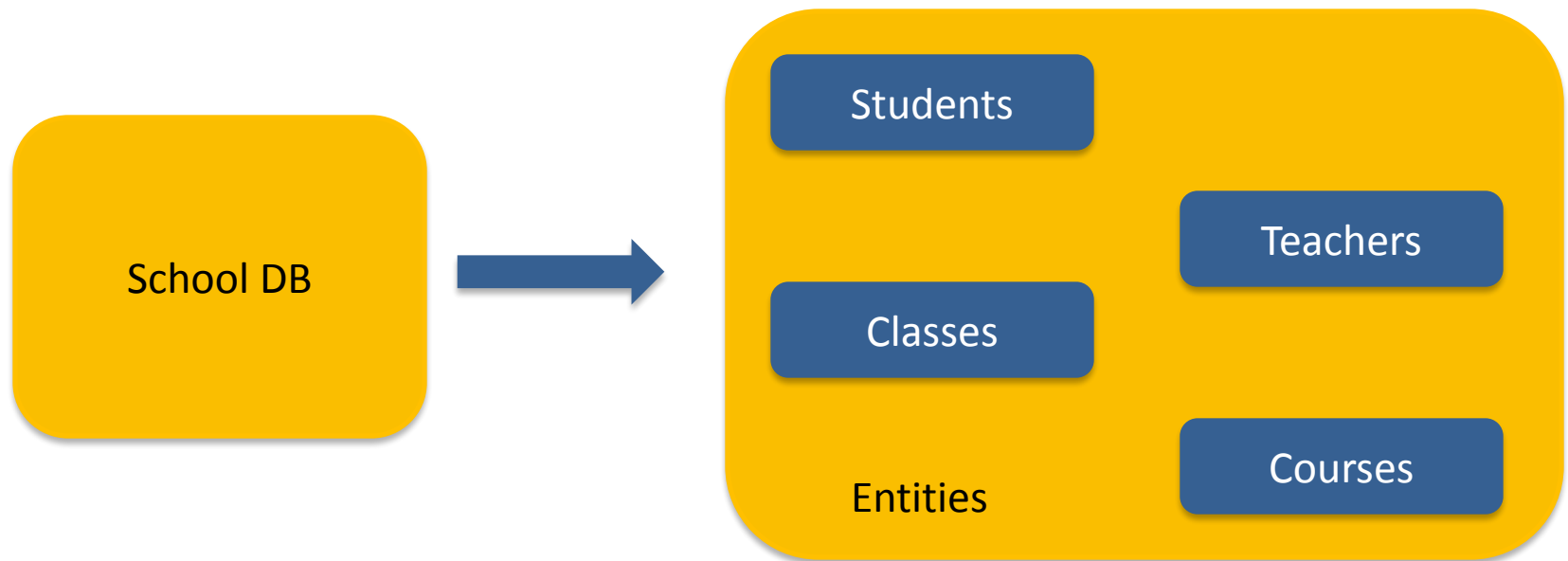


Example: Microsoft SQL Server, Oracle, MySql, Firebird, PostgreSQL

# Entity Relationship Diagram

# What is Entity

Entity it's a real-world thing either animate or inanimate that can be easily identifiable and distinguishable.

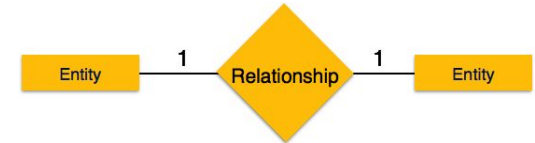
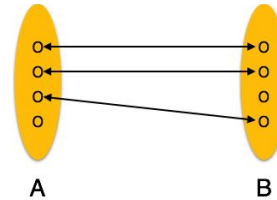




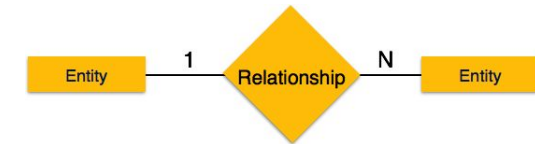
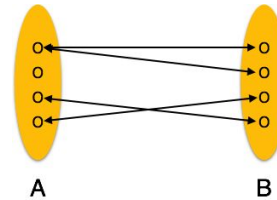
# What is Relationship

The association among entities is called relationship

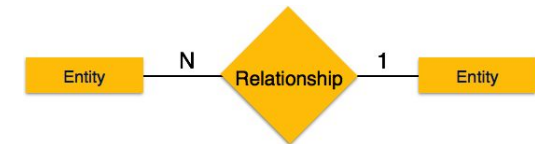
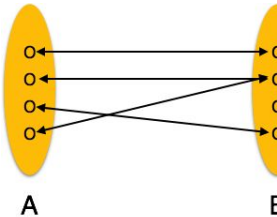
**One-to-one:** one entity from entity set A can be associated with at most one entity of entity set B and vice versa.



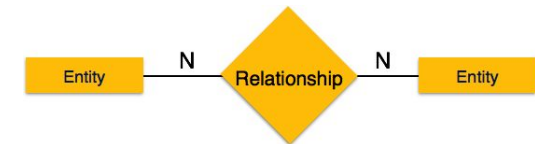
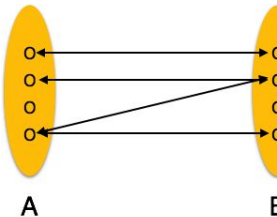
**One-to-many:** One entity from entity set A can be associated with more than one entities of entity set B but from entity set B one entity can be associated with at most one entity



**Many-to-one:** More than one entities from entity set A can be associated with at most one entity of entity set B but one entity from entity set B can be associated with more than one entity from entity set A.



**Many-to-many:** one entity from A can be associated with more than one entity from B and vice versa.

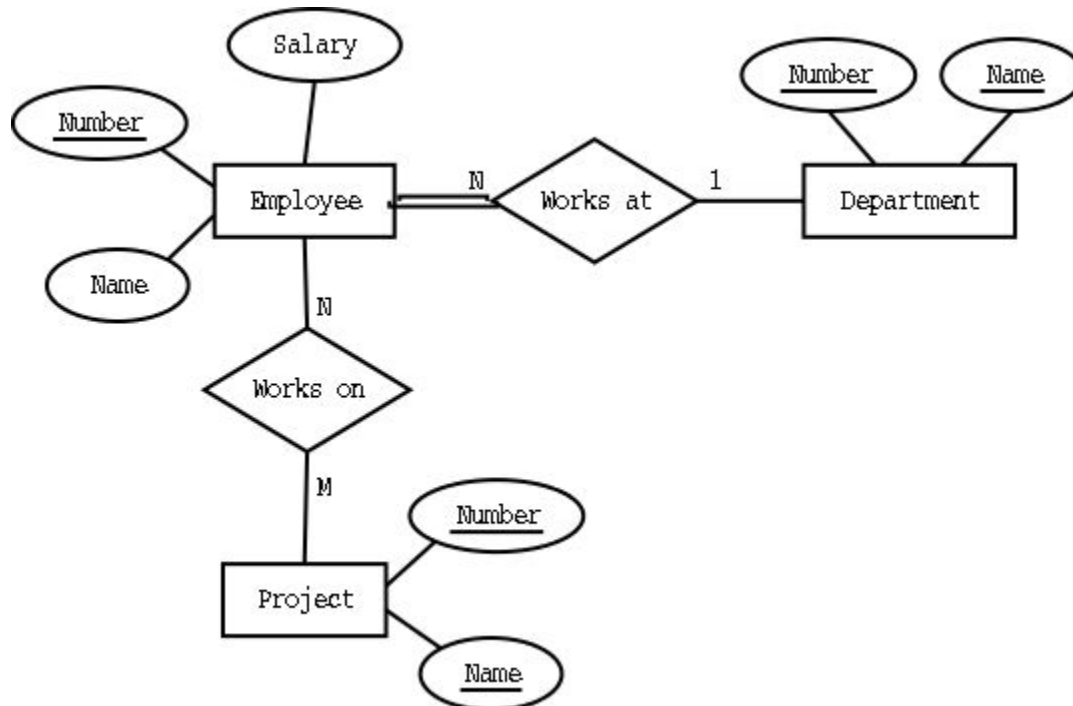


# Entity Relationship Diagram

ER Model when conceptualized into diagrams gives a good overview of entity-relationship, which is easier to understand.

ER Diagrams mainly comprised of:

- ✓ Entity and its attributes;
- ✓ Relationship, which is association among entities.



# Database Objects

# Main Database Objects

Typical relational database contains:

- Tables
- Views
- Stored procedures
- Triggers
- Keys
- Indexes
- etc.

# Tables

- A table is a collection of related data held in a structured format within a database. It consists of fields (columns), and rows

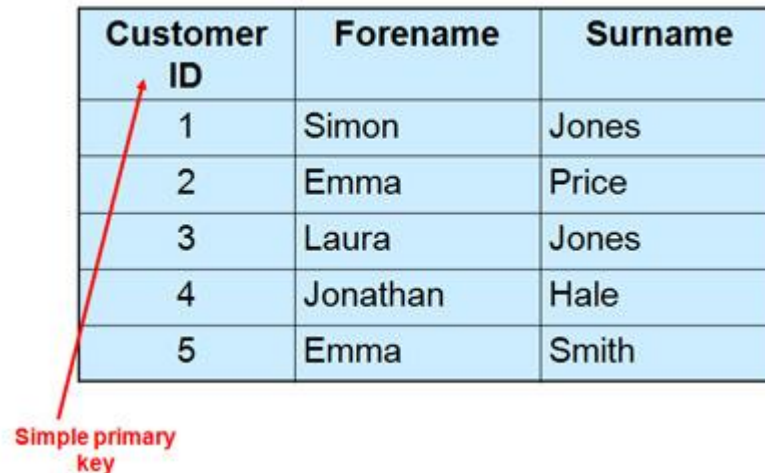
| First Name | Last Name | Address             | City     | Age |
|------------|-----------|---------------------|----------|-----|
| Mickey     | Mouse     | 123 Fantasy Way     | Anaheim  | 73  |
| Bat        | Man       | 321 Cavern Ave      | Gotham   | 54  |
| Wonder     | Woman     | 987 Truth Way       | Paradise | 39  |
| Donald     | Duck      | 555 Quack Street    | Mallard  | 65  |
| Bugs       | Bunny     | 567 Carrot Street   | Rascal   | 58  |
| Wiley      | Coyote    | 999 Acme Way        | Canyon   | 61  |
| Cat        | Woman     | 234 Purrfect Street | Hairball | 32  |
| Tweety     | Bird      | 543                 | Itottlaw | 28  |

# Keys

- Keys are, as their name suggests, a key part of a relational database and a vital part of the structure of a table. They ensure each record within a table can be uniquely identified by one or a combination of fields within the table. They help enforce integrity and help identify the relationship between tables.

# Primary Key

- Primary key is the field that uniquely identifies the table row. Traditionally, this field is named ID or <TableName>ID




| Customer ID | Forename | Surname |
|-------------|----------|---------|
| 1           | Simon    | Jones   |
| 2           | Emma     | Price   |
| 3           | Laura    | Jones   |
| 4           | Jonathan | Hale    |
| 5           | Emma     | Smith   |

Simple primary key

# Foreign Key

- The foreign key field is a field that references the primary key field of another table. It is used for creating relationships between tables

| customer_id | first_name | last_name | email          | phone        |
|-------------|------------|-----------|----------------|--------------|
| 1           | John       | Bonham    | bonham@example | 020394984    |
| 2           | Dave       | Grohl     | grohl@example. | 920930938    |
| 3           | Robert     | Smith     | smith@example. | 9873219847   |
| 4           | Frank      | Black     | black@example. | 892372039872 |



| order_id | order_date          | customer |
|----------|---------------------|----------|
| 1        | 2012-02-19 01:14:30 | 2        |
| 2        | 2012-03-11 16:14:59 | 2        |
| 3        | 2012-03-01 18:15:15 | 1        |
| 4        | 2012-03-11 01:00:26 | 4        |
| 5        | 2012-03-11 01:27:51 | 3        |



# Indexes

- A database index is a data structure that improves the speed of data retrieval operations on a DB table at the cost of additional writes and storage space to maintain the index data structure.
- Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed.

# Views

- A Database View is a subset of the database sorted and displayed in a particular way
- View is the result set of a stored query on the data, which the database users can query just as they would in a persistent database collection object

# Stored Procedures

- A stored procedure is a subroutine available to applications that access a relational database system
- Usually is written in special language which is the extension of SQL.
- For MS SQL Server this language is called Transact-SQL

# Stored Procedures Creation

```
USE AdventureWorks2012;
GO
CREATE PROCEDURE HumanResources.uspGetEmployeesTest2
    @LastName nvarchar(50),
    @FirstName nvarchar(50)
AS
    SET NOCOUNT ON;
    SELECT FirstName, LastName, Department
    FROM HumanResources.vEmployeeDepartmentHistory
    WHERE FirstName = @FirstName AND LastName = @LastName
    AND EndDate IS NULL;
GO
```

# Stored Procedures Execution

```
EXECUTE HumanResources.uspGetEmployeesTest2
    N'Ackerman', N'Pilar';
-- Or
EXEC HumanResources.uspGetEmployeesTest2
    @LastName = N'Ackerman', @FirstName = N'Pilar';
GO
-- Or
EXECUTE HumanResources.uspGetEmployeesTest2
    @FirstName = N'Pilar', @LastName = N'Ackerman';
GO
```

# Triggers

- A database trigger is procedural code that is automatically executed in response to certain events on a particular table or view in a database.
- The trigger is mostly used for maintaining the integrity of the information on the database.

# Visual tools for working with RDBMS

- There are some visual tools that allow to manipulate DB objects, administer the DB and execute SQL statements and program scripts.
- MS SQL Server Management Studio

# General Data Types



# Data Types

| Data types                         | Description  |
|------------------------------------|--|
| CHARACTER(n)                       | Character string. Fixed-length n   |
| VARCHAR(n) or CHARACTER VARYING(n) | Character string. Variable length. Maximum length n  |
| BOOLEAN                            | Stores TRUE or FALSE values  |
| INTEGER(p)                         | Integer numerical (no decimal). Precision p  |
| INTEGER                            | Integer numerical (no decimal). Precision 10   |
| BIGINT                             | Integer numerical (no decimal). Precision 19   |
| DECIMAL(p,s)<br>NUMERIC(p,s)       | Exact numerical, precision p, scale s. Example: decimal(5,2) is a number that has 3 digits before the decimal and 2 digits after the decimal   |
| FLOAT(p)                           | Approximate numerical, mantissa precision p. A floating number in base 10 exponential notation. The size argument for this type consists of a single number specifying the minimum precision |
| DATE                               | Stores year, month, and day values   |
| TIME                               | Stores hour, minute, and second values   |

# Data Type Conversion

|                  | To:    |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
|------------------|--------|-----------|------|---------|-------|----------|----------|---------------|------|------|----------------|-----------|---------|---------|-------|------|--------|-----------|----------------|---------------|-------|------------|-----|-----------|------------------|-------|-------|------|-------------|-----|---------|-------------|
| From:            | binary | varbinary | char | varchar | nchar | nvarchar | datetime | smalldatetime | date | time | datetimeoffset | datetime2 | decimal | numeric | float | real | bigint | int(INT4) | smallint(INT2) | tinyint(INT1) | money | smallmoney | bit | timestamp | uniqueidentifier | image | ntext | text | sql_variant | xml | CLR UDT | hierarchyid |
| binary           |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| varbinary        |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| char             |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| varchar          |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| nchar            |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| nvarchar         |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| datetime         |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| smalldatetime    |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| date             |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| time             |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| datetimeoffset   |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| datetime2        |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| decimal          |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| numeric          |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| float            |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| real             |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| bigint           |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| int(INT4)        |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| smallint(INT2)   |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| tinyint(INT1)    |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| money            |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| smallmoney       |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| bit              |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| timestamp        |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| uniqueidentifier |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| image            |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| ntext            |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| text             |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| sql_variant      |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| xml              |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| CLR UDT          |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |
| hierarchyid      |        |           |      |         |       |          |          |               |      |      |                |           |         |         |       |      |        |           |                |               |       |            |     |           |                  |       |       |      |             |     |         |             |

Explicit conversion

Implicit conversion

Conversion not allowed

\*

Requires explicit CAST to prevent the loss of precision or scale that might occur in an implicit conversion.

Implicit conversions between xml data types are supported only if the source or target is untyped xml. Otherwise, the conversion must be explicit.



*Empowering your Business  
through Software Development*

# Thank you

## US OFFICES

Austin, TX  
Fort Myers, FL  
Boston, MA  
Newport Beach, CA  
Salt Lake City, UT

## EUROPE OFFICES

United Kingdom  
Germany  
The Netherlands  
Ukraine  
Bulgaria

## EMAIL

[info@softserveinc.com](mailto:info@softserveinc.com)

## WEBSITE:

[www.softserveinc.com](http://www.softserveinc.com)

## USA TELEPHONE

Toll-Free: 866.687.3588  
Office: 239.690.3111

## UK TELEPHONE

Tel: 0207.544.8414

## GERMAN TELEPHONE

Tel: 0692.602.5857