https://www.halvorsen.blog

erwin Data Modeler

Academic Edition

Hans-Petter Halvorsen

erwin Data Modeler

- In this Tutorial we will use the erwin Data Modeler software.
- erwin Data Modeler is used to modeling, design and create ER (Entity Relationship) diagram for your Database.
- erwin Data Modeler comes in different paid versions, but you can request a free trial.
- If you are a student, you can also apply for the "Academic Edition".
- https://www.erwin.com

https://www.halvorsen.blog

Database Modeling and Design

Hans-Petter Halvorsen

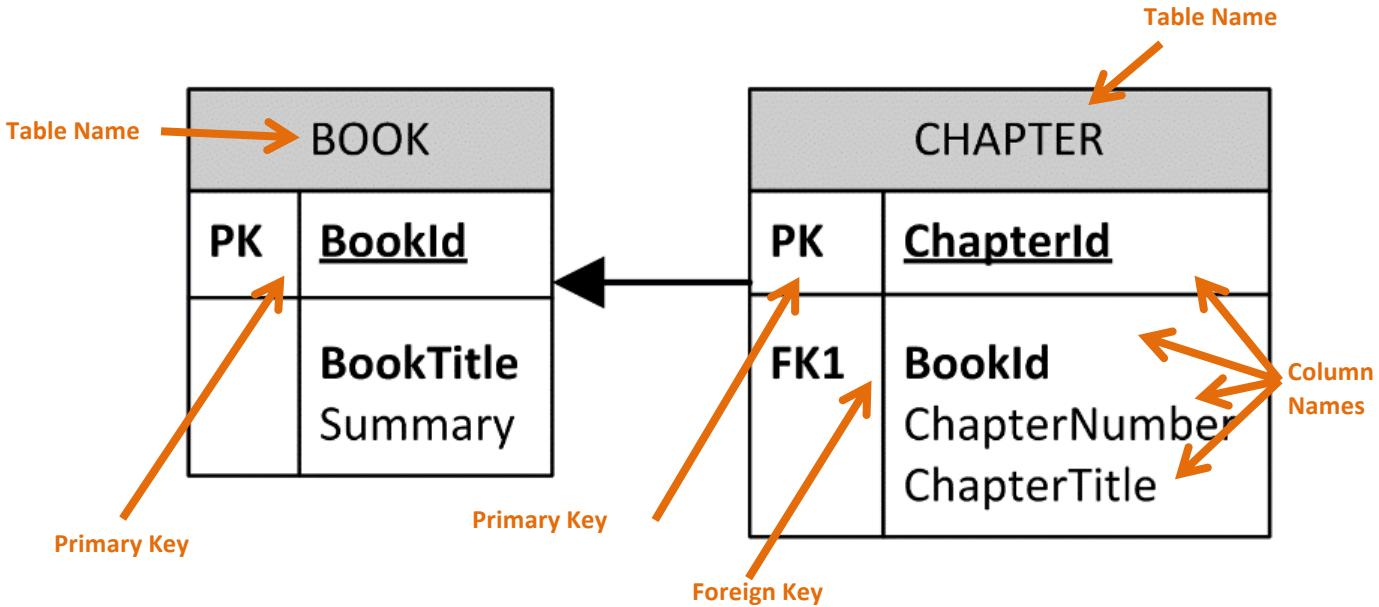
Table of Contents

Entity Relationship (ER) Diagram

ER Diagram (Entity-Relationship Diagram)

- Used for Design and Modeling of Databases.
- Specify Tables and **relationship** between them (**Primary Keys** and **Foreign Keys**)

Example:



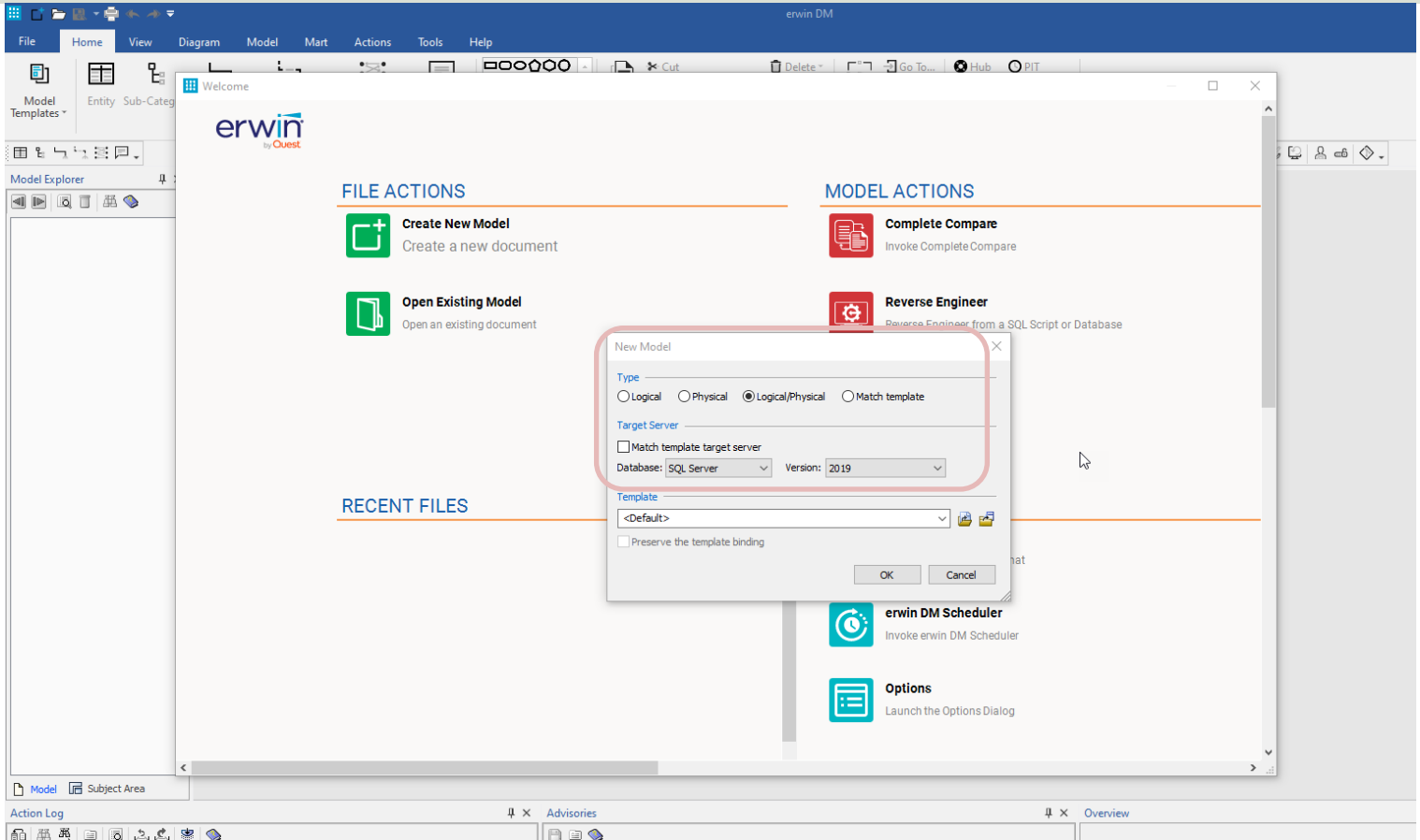
Relational Database. In a relational database all the tables have one or more relation with each other using Primary Keys (PK) and Foreign Keys (FK). Note! You can only have one PK in a table, but you may have several FK's.

Start using erwin Data Modeler

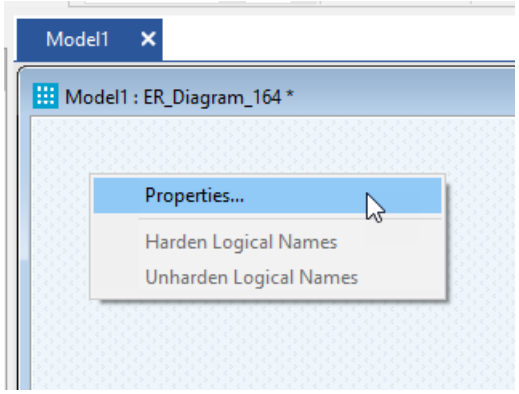
Hans-Petter Halvorsen

Table of Contents

New Model


erwin DM
File
Home
View
Diagram
Model
Mart
Actions
Tools
Help
Model Templates
Entity
Sub-Categ
Cut
Delete
Go To...
Hub
PIT
Model Explorer
Welcome
erwin by Quest
FILE ACTIONS
Create New Model
Create a new document
Open Existing Model
Open an existing document
RECENT FILES
MODEL ACTIONS
Complete Compare
Invoke Complete Compare
Reverse Engineer
Reverse Engineer from a SQL Script or Database
erwin DM Scheduler
Invoke erwin DM Scheduler
Options
Launch the Options Dialog
New Model
Type
Logical
Physical
Logical/Physical
Match template
Target Server
Match template target server
Database:
SQL Server
Version:
2019
Template
<Default>
Preserve the template binding
OK
Cancel
Model
Subject Area
Action Log
Advisories
Overview

Page Setup



ER Diagram 'ER_Diagram_164' Editor

Edit ER Diagrams in: Model_1

Enter filter text

Name	View Mode	Auto-populate
ER_Diagram_164	Logical	☐

General | Members | Relationships | Layout | Display | Entity | Table | View | Relationshi | Page Setup | Page Setup Margins | Definition | Where Used | UDP

Page Setup

Page Size	A4
Width(inches):	11.69
Height(inches):	8.27
Orientation	Landscape
Zoom Level	100

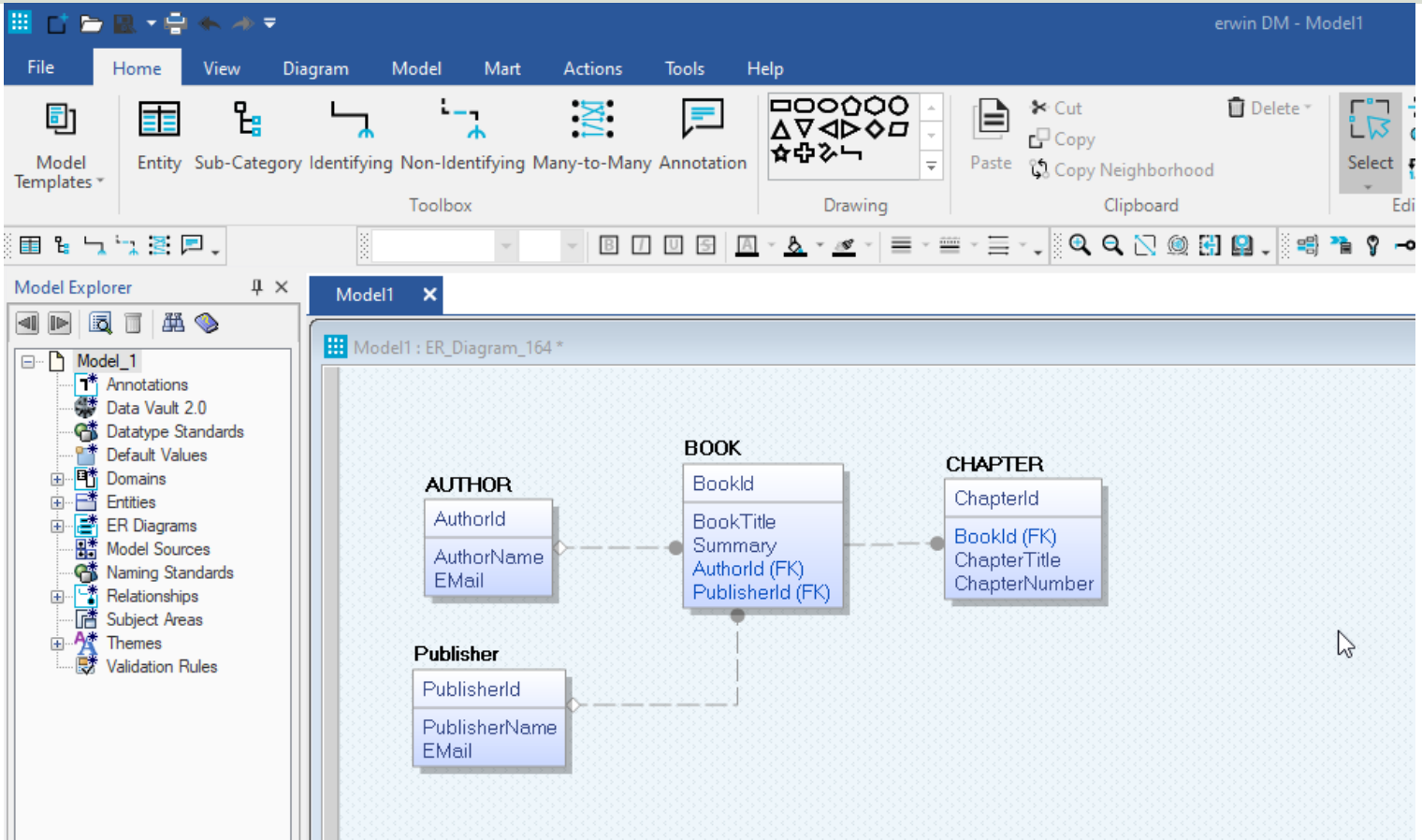
Miscellaneous

Print Border	☑
Print In Color	☑

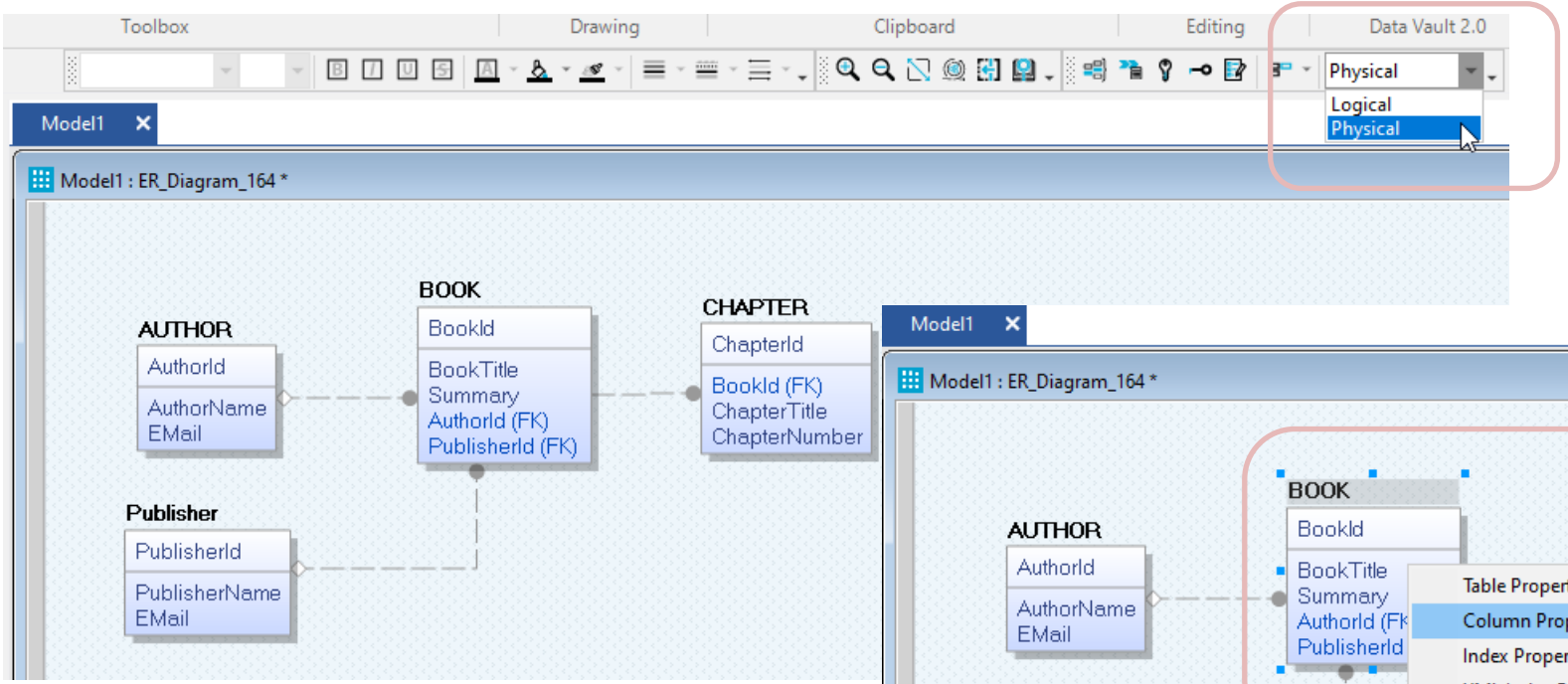
Close | Cancel

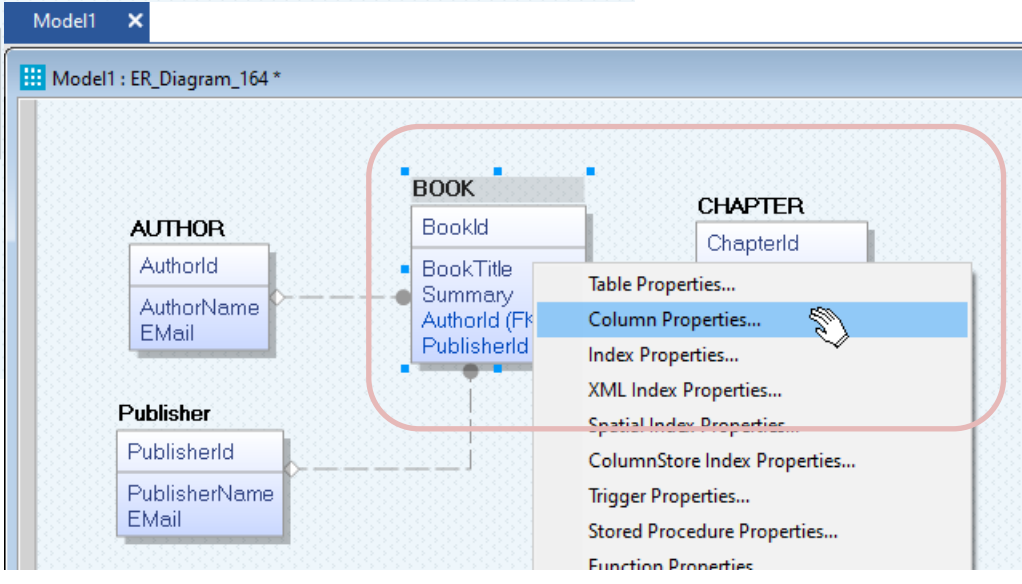
Details...

Example

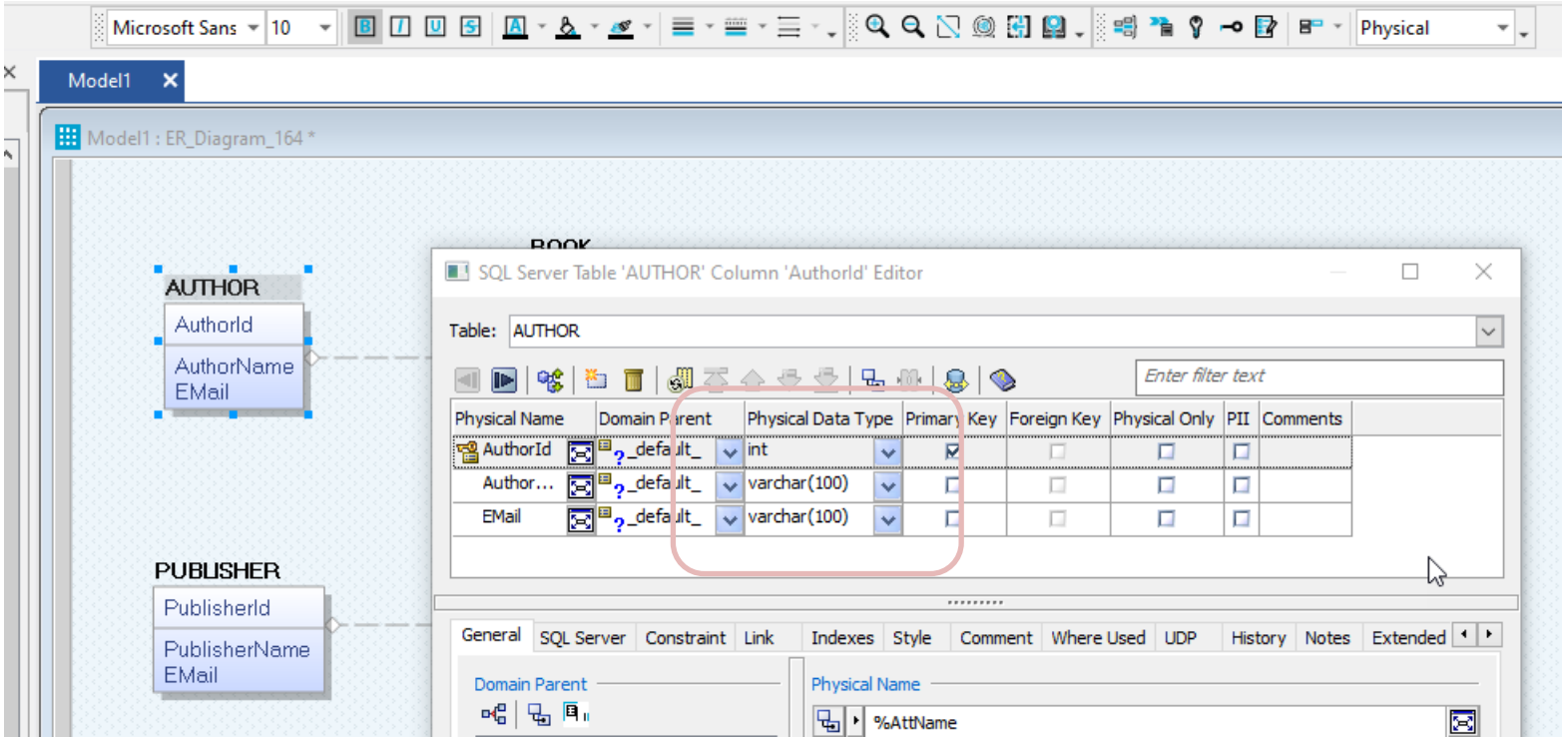

erwin DM - Model1
File
Home
View
Diagram
Model
Mart
Actions
Tools
Help
Model Templates
Entity
Sub-Category
Identifying
Non-Identifying
Many-to-Many
Annotation
Toolbox
Drawing
Paste
Cut
Copy
Copy Neighborhood
Delete
Clipboard
Select
Model Explorer
Model_1
Annotations
Data Vault 2.0
Datatype Standards
Default Values
Domains
Entities
ER Diagrams
Model Sources
Naming Standards
Relationships
Subject Areas
Themes
Validation Rules
Model1
Model1 : ER_Diagram_164 *
AUTHOR
AuthorId
AuthorName
EMail
BOOK
BookId
BookTitle
Summary
AuthorId (FK)
PublisherId (FK)
CHAPTER
ChapterId
BookId (FK)
ChapterTitle
ChapterNumber
Publisher
PublisherId
PublisherName
EMail

Set Data Types

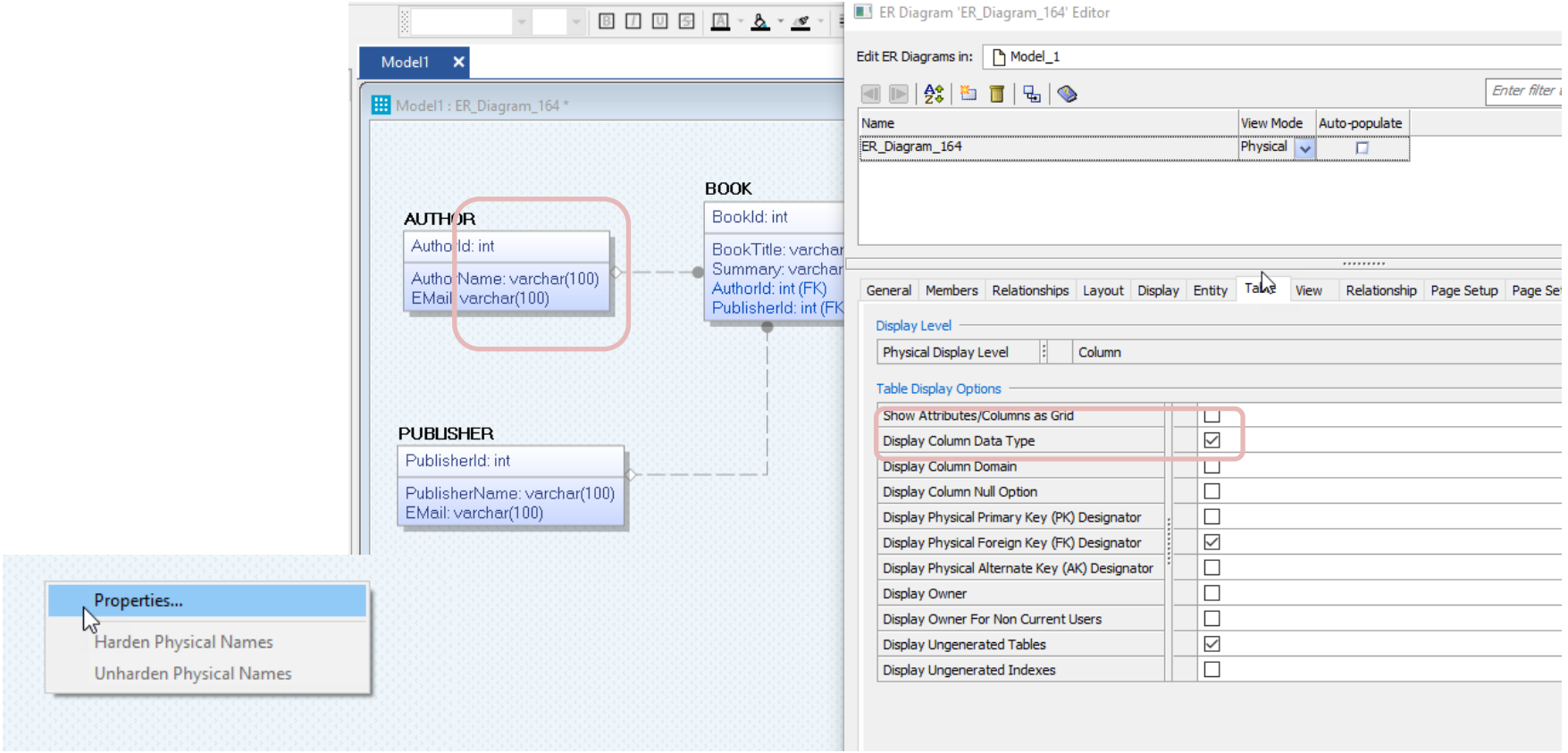




Set Data Types



Show Data Types in Diagram

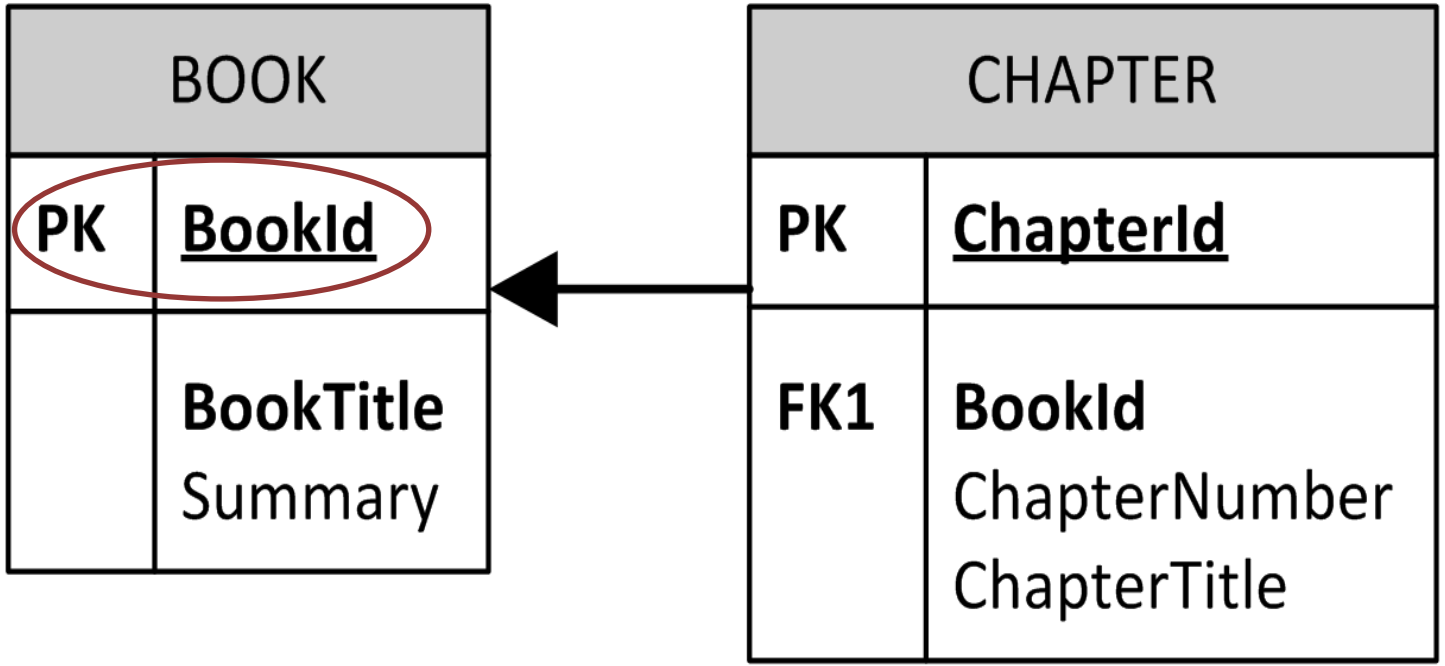


Best Practice

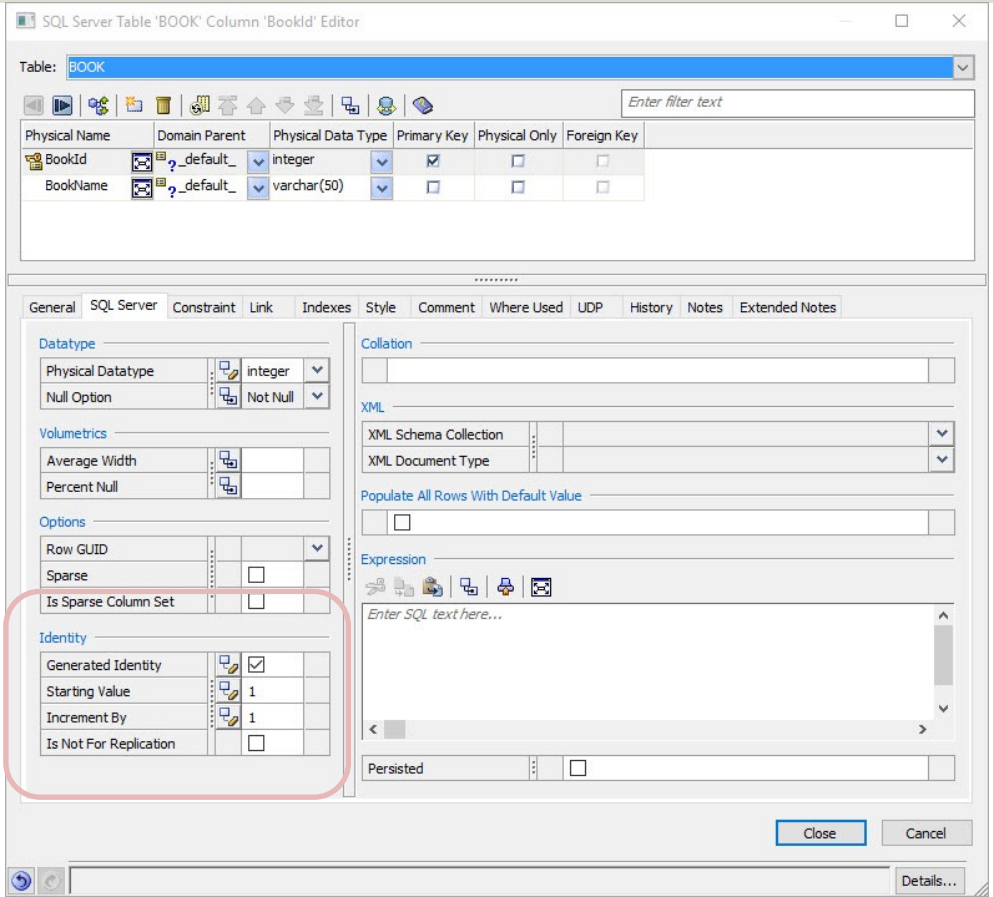
- **Tables**: Use upper case and singular form in table names – not plural, e.g., "STUDENT" (not "students")
- **Columns**: Use Pascal notation, e.g., "StudentId"
- **Primary Key**:
  - If the table name is "COURSE", name the Primary Key column "CourseId", etc.
  - "Always" use Integer and Identity(1,1) for Primary Keys. Use UNIQUE constraint for other columns that needs to be unique, e.g., "RoomNumber"
- Specify **Required** Columns (NOT NULL) – i.e., which columns that need to have data or not
- Standardize on few/these **Data Types**: *int*, *float*, *varchar(x), datetime*, *bit*
- Use English for table and column names
- Avoid abbreviations! (Use "RoomNumber" – not "RoomNo", "RoomNr", ...)

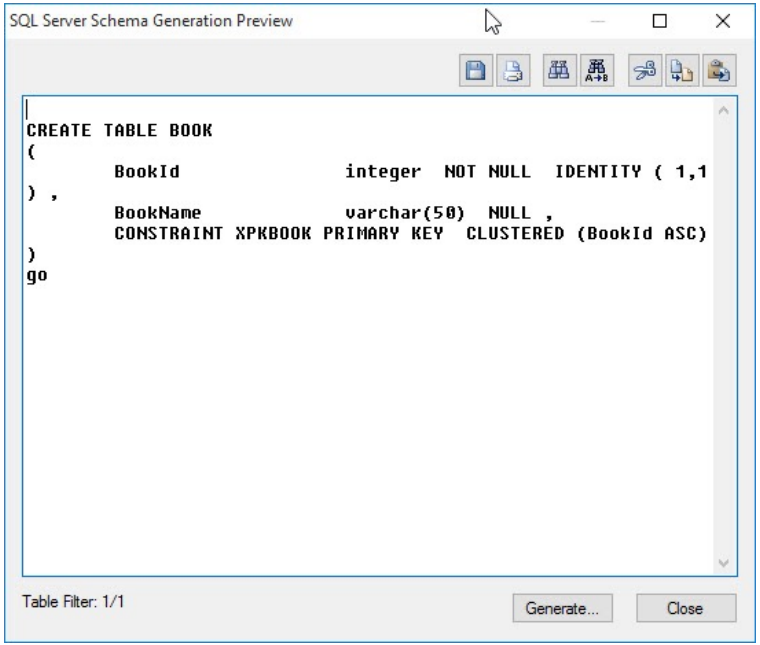
Identity(1,1)

Typically, you want to use Integer and Identity(1,1) on the Primary Keys



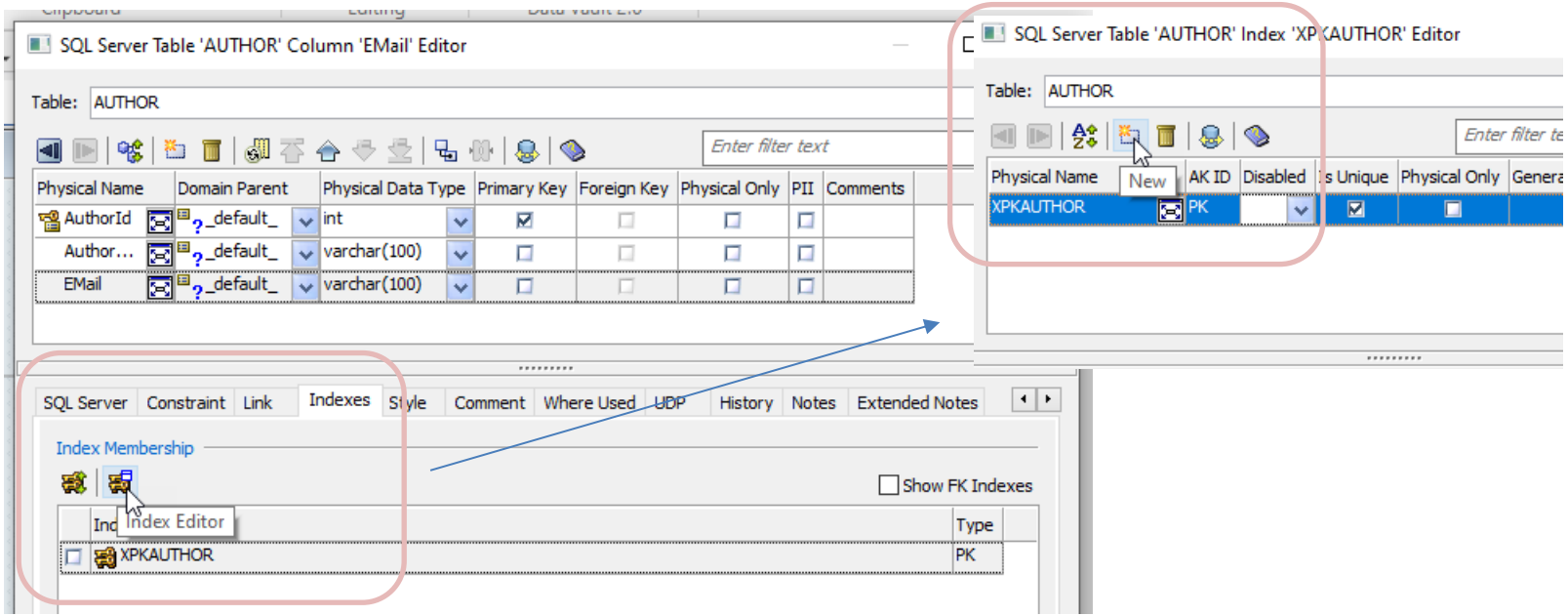
Identity(1,1)


SQL Server Table 'BOOK' Column 'BookId' Editor
Table: BOOK
Enter filter text
Physical Name
Domain Parent
Physical Data Type
Primary Key
Physical Only
Foreign Key
BookId
default
integer
BookName
default
varchar(50)
General
SQL Server
Constraint
Link
Indexes
Style
Comment
Where Used
UDP
History
Notes
Extended Notes
Datatype
Physical Datatype
integer
Null Option
Not Null
Volumetrics
Average Width
Percent Null
Options
Row GUID
Sparse
Is Sparse Column Set
Identity
Generated Identity
Starting Value
1
Increment By
1
Is Not For Replication
Collation
XML
XML Schema Collection
XML Document Type
Populate All Rows With Default Value
Expression
Enter SQL text here...
Persisted
Close
Cancel
Details...


SQL Server Schema Generation Preview
CREATE TABLE BOOK
(
BookId integer NOT NULL IDENTITY (1,1
) ,
BookName varchar(50) NULL ,
CONSTRAINT XPKBOOK PRIMARY KEY CLUSTERED (BookId ASC)
)
go
Table Filter: 1/1
Generate...
Close

Unique Constraint

Typically, you want to use a UNIQUE constraint for other columns that needs to be unique, e.g., “RoomNumber”



Create Table Script

Hans-Petter Halvorsen

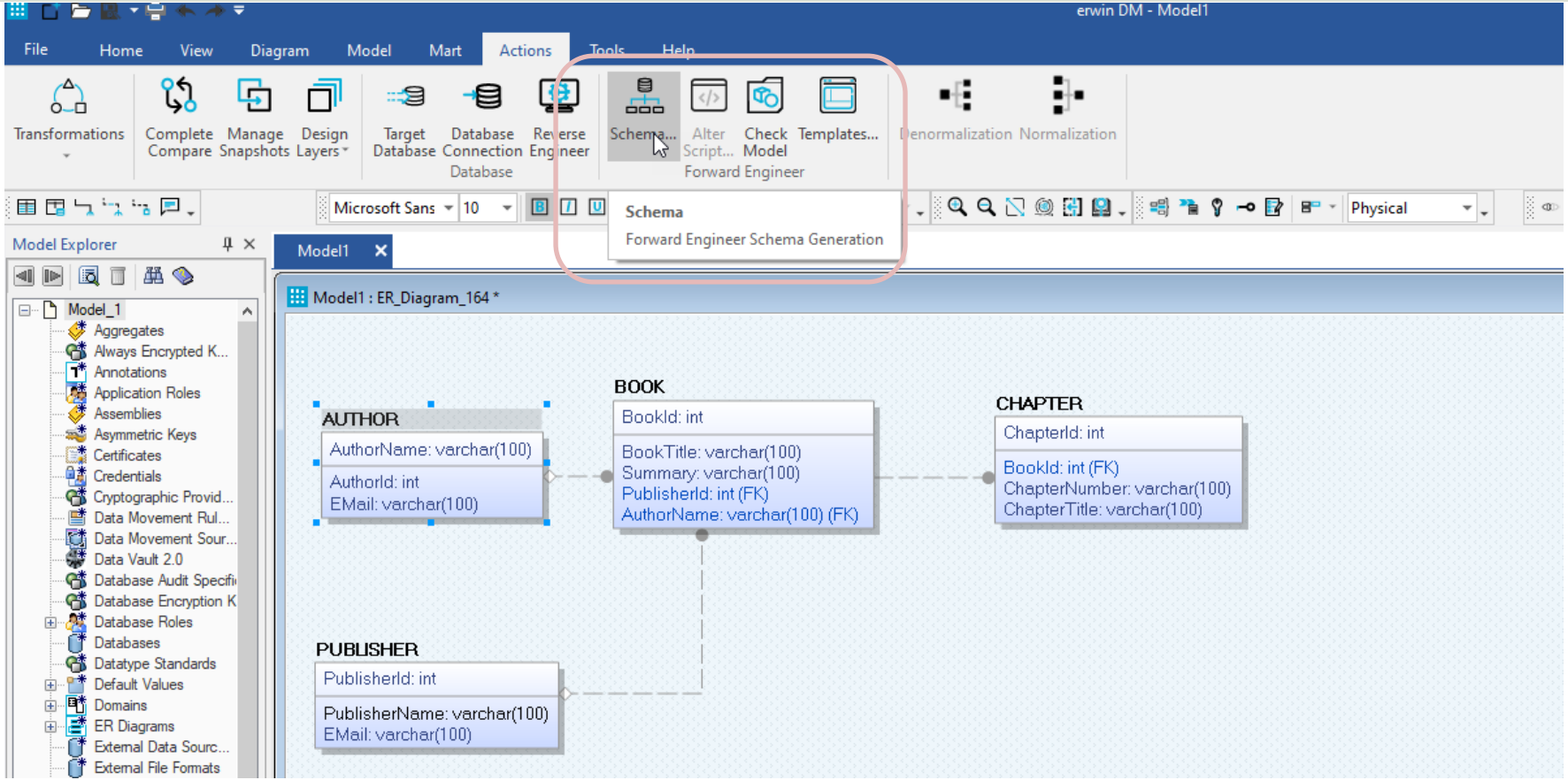
Table of Contents

Database Design & Implementation

Recommended Steps:

1. Database Modelling/Design using erwin Data Modeler
2. Generate SQL Table Script using erwin Data Modeler (you might need to adjust/improve it in order to make it more robust)
3. Create Tables in SQL Server using the SQL Script generated by erwin Data Modeler
4. Create Stored Procedures, View, Triggers, etc. inside SQL Server if needed.

Forward Engineering Shema



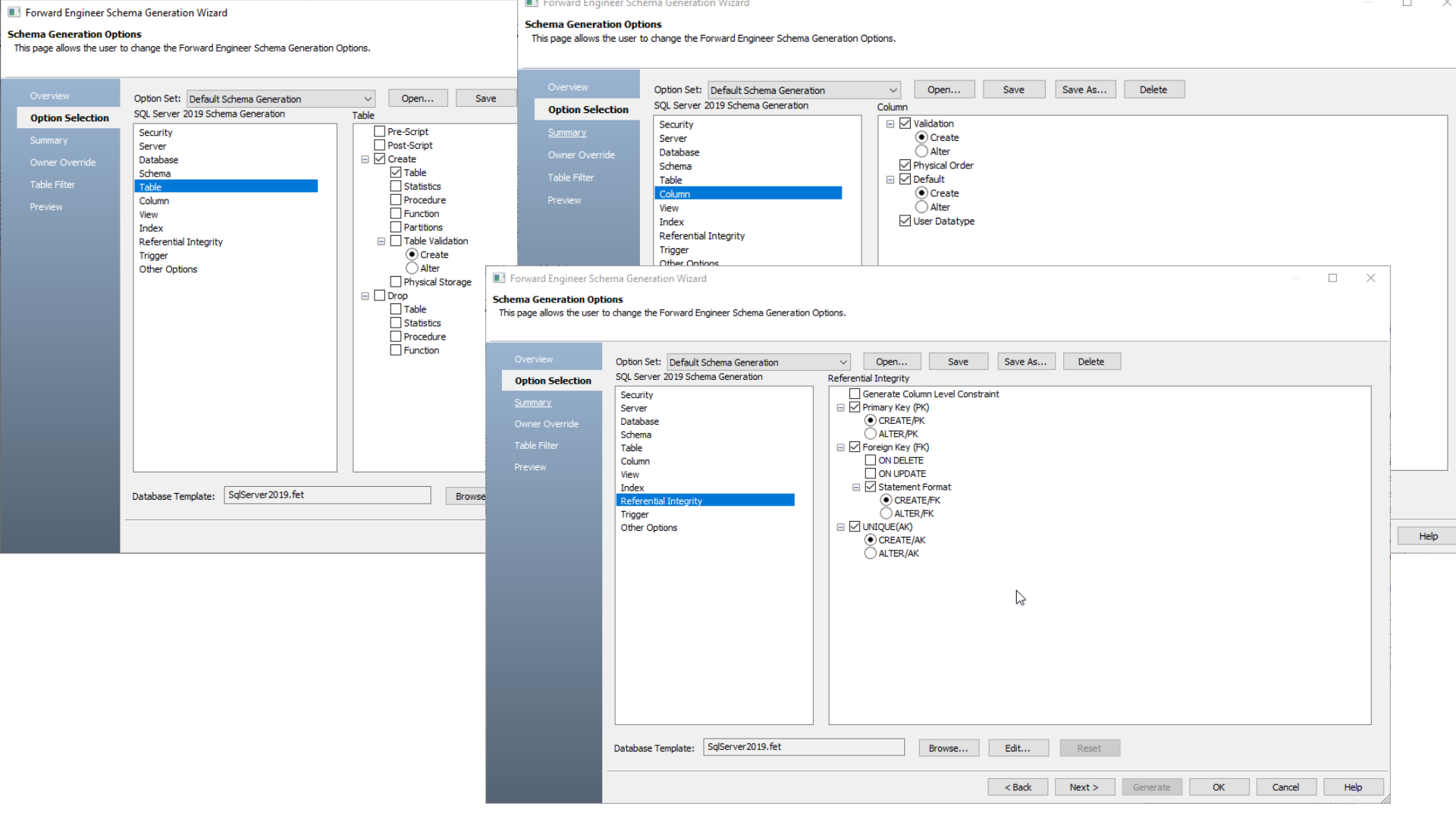
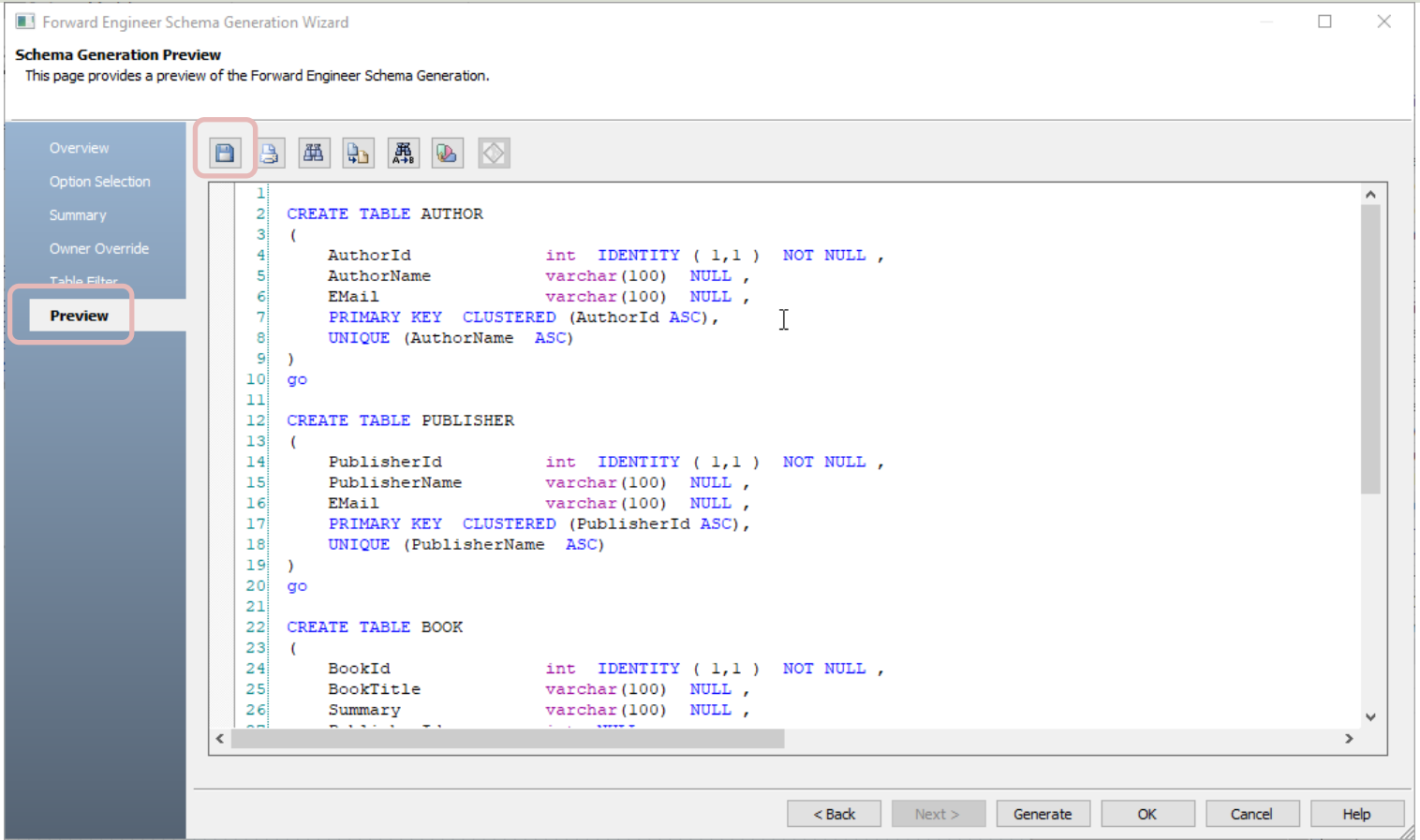

Forward Engineer Schema Generation Wizard
Schema Generation Options
This page allows the user to change the Forward Engineer Schema Generation Options.
Overview
Option Selection
Summary
Owner Override
Table Filter
Preview
Option Set: Default Schema Generation
Open...
Save
Save As...
Delete
SQL Server 2019 Schema Generation
Security
Server
Database
Schema
Table
Column
View
Index
Referential Integrity
Trigger
Other Options
Table
Pre-Script
Post-Script
Create
Table
Statistics
Procedure
Function
Partitions
Table Validation
Create
Alter
Physical Storage
Drop
Table
Statistics
Procedure
Function
Column
Validation
Create
Alter
Physical Order
Default
Create
Alter
User Datatype
Referential Integrity
Generate Column Level Constraint
Primary Key (PK)
CREATE/PK
ALTER/PK
Foreign Key (FK)
ON DELETE
ON UPDATE
Statement Format
CREATE/FK
ALTER/FK
UNIQUE(AK)
CREATE/AK
ALTER/AK
Database Template:
SqlServer2019.fet
Browse...
Edit...
Reset
< Back
Next >
Generate
OK
Cancel
Help

Table Script

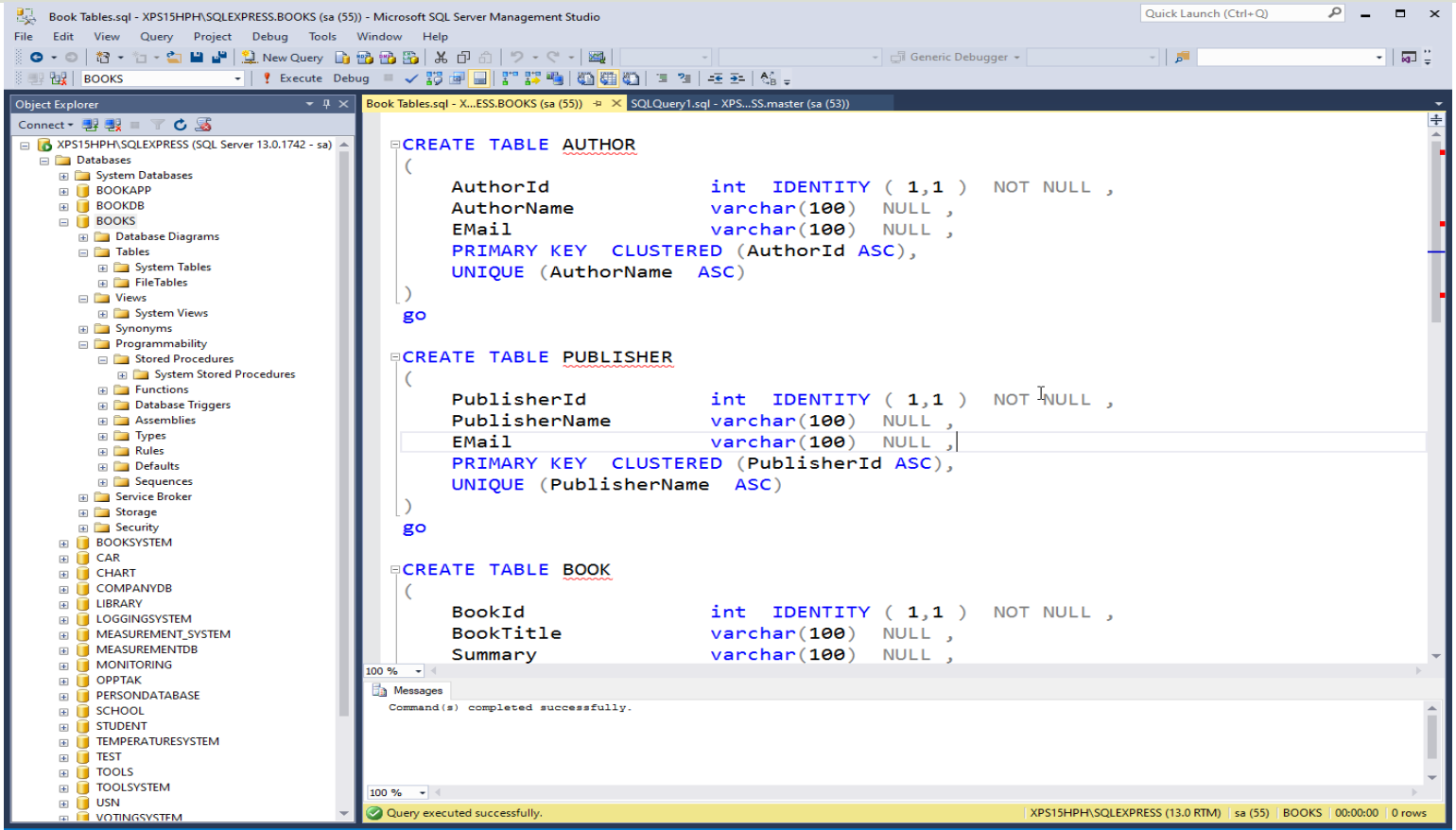


SQL Server

Hans-Petter Halvorsen

Table of Contents

Create Tables in SQL Server

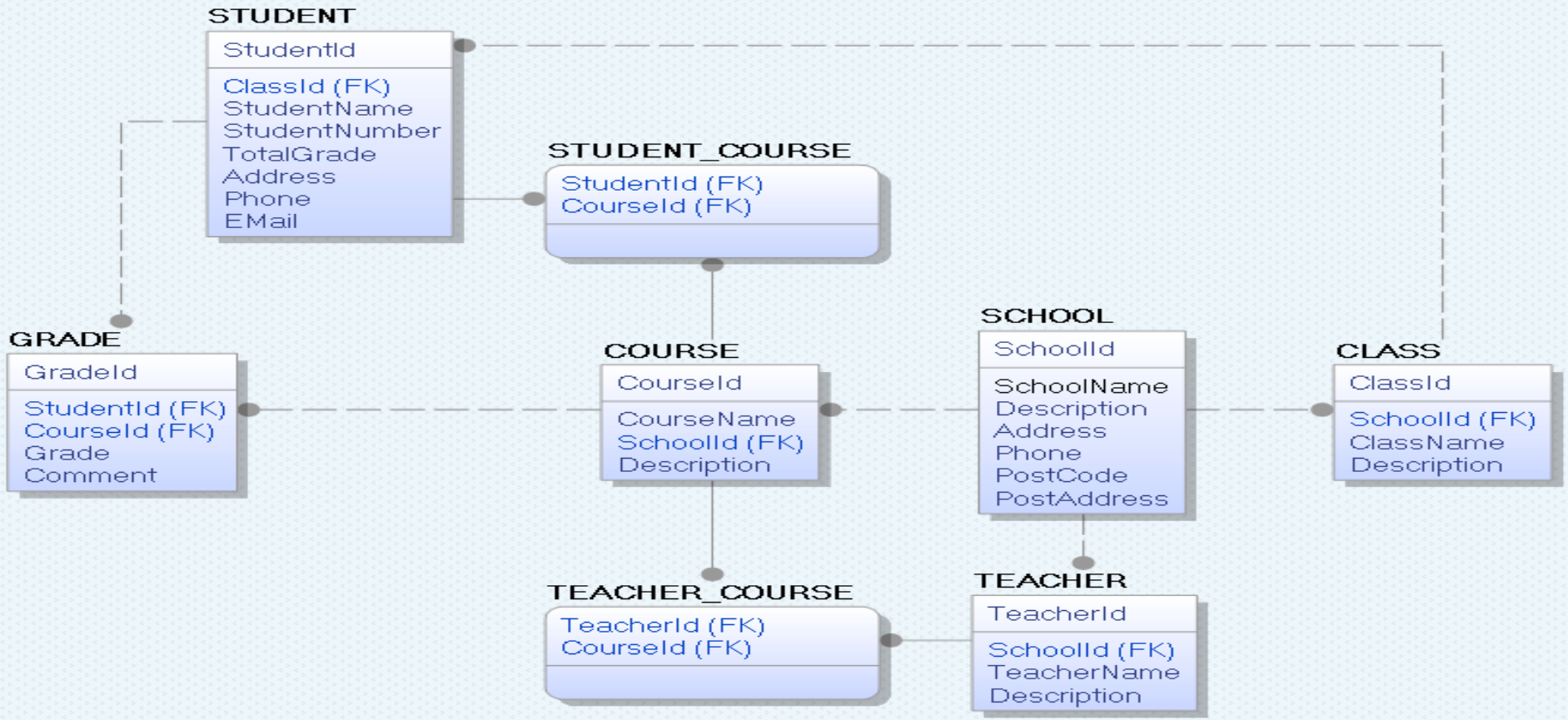


Summary

Summary

1. Database Modelling/Design using erwin Data Modeler
2. Generate SQL Table Script using erwin Data Modeler (you might need to adjust/improve it in order to make it more robust)
3. Create Tables in SQL Server using the SQL Script generated by erwin Data Modeler
4. Create Stored Procedures, View, Triggers, etc. inside SQL Server if needed.

Are you able to make this diagram?



Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: https://www.halvorsen.blog