Design Your Databases Straight to the Cloud

Heli Helskyaho Nordic ACE Tour 2017



Introduction, Heli

- Graduated from University of Helsinki (Master of Science, computer science), currently a doctoral student, researcher and lecturer (databases, Big Data, Multi-model Databases, methods and tools for utilizing semistructured data for decision making) at University of Helsinki
- * Worked with Oracle products since 1993, worked for IT since 1990
- * Data and Database!
- * CEO for Miracle Finland Oy
- * Oracle ACE Director
- Ambassador/EOUC (EMEA Oracle Users Group Community)
- Public speaker and an author
- Winner of Devvy for Database Design Category, 2015
- Author of the book Oracle SQL Developer Data Modeler for Database Design Mastery (Oracle Press, 2015), co-author for Real World SQL and PL/SQL: Advice from the Experts (Oracle Press, 2016)





Oracle SQL Developer Data Modeler for Database Design Mastery

Design, Deploy, and Maintain World-Class Databases on Any Platform

Heli Helskyaho Oracle ACE Director

Forewords by C.J. Date and Tom Kyte





Real World SQL & PL/SQL

Advice from the Experts

Arup Nanda Brendan Tierney Heli Helskyaho Martin Widlake Alex Nuijten





Helskyaho Heli, Valin-Raki Elise

Introduction, Heli

- Been an Oracle Designer user since 1996 and Data Modeler user since 2010
- * Been solving performance problems since 1993
 - * Absolutely convinced that good database design and documentation will reduce this work.
 - Absolutely convinced that it is easier to solve problems on database that has been documented



500+ Technical Experts Helping Peers Globally





3 Membership Tiers

- Oracle ACE Director
- Oracle ACE
- Oracle ACE Associate

bit.ly/OracleACEProgram





Nominate yourself or someone you know: acenomination.oracle.com

Why to design?

- * "Data is the most valuable property in our company"
- * "Why do we need to design the database? We already design the application!"



Why is designing the application not enough?

- * Point of view (saving and retrieving data vs. UI)
- First increment vs. 20 years from now
- * "the whole picture" vs. increments
- * Different goals/targets:
 - * Code tables vs. Code files (how about the data integrity?)
 - How about analysis, reports, ... everything else but the UI that the data is used for
- Same terminology, different meaning -> misunderstandings

* ...



Why to model the data?

- * To facilitate communication about the requirements
- * To find the questions that should be asked
- * To understand the requirements



How is Cloud different?

- It has a bigger chance to have latency (usually a hybrid solution)
 - Good reason to add as much business logic to the database as possible!
 - ∗ But you would do that anyway ☺
- Everything you have learned about database designing still applies....



What is database designing?

- * 4 (5) phases, over and over again
 - Requirement analysis (DM: logical)
 - Conceptual design (DM: logical)
 - * Logical design (DM: relational)
 - * Physical design (DM: physical)
 - * (Transaction design) (DM: process)



It would be crazy...

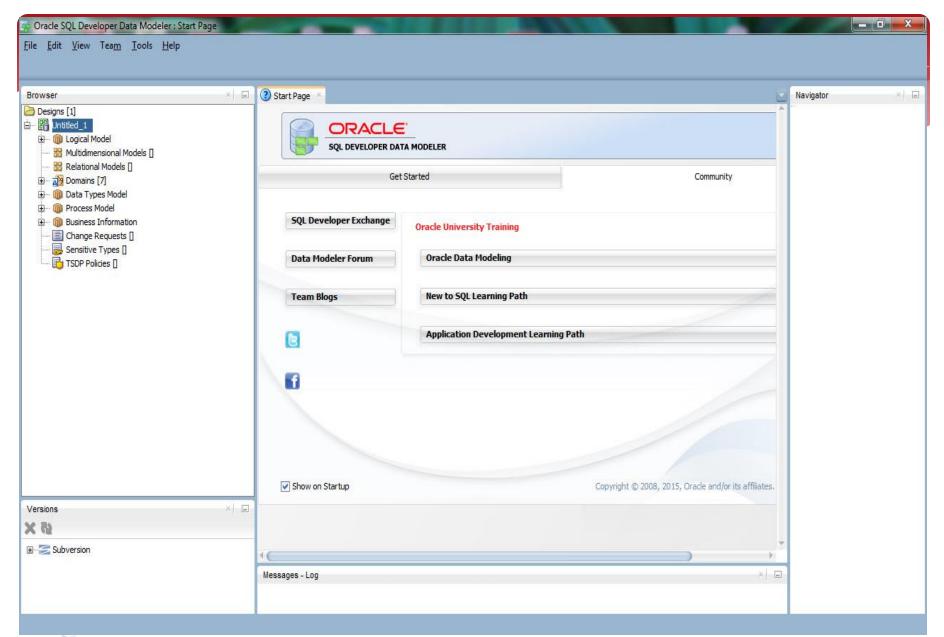
* ... to design databases without a tool!



Oracle SQL Developer Data Modeler

- To be efficient in designing you need a tool: my recommendation is Data Modeler
 - Free of charge
 - Support for many different databases (Oracle, MS SQL Server, DB2,...)
 - * Support for both documenting the existing databases and designing a new one (and mainintaining that)
 - Support for reporting, naming standards, glossaries, design rules, ...
 - * Support for version control and multiuser environment
 - Support for everything you need for database design plus more







Designing the database

1. **Requirement analysis:** finding and analysing the requirements the future end users have

Result: specification of user requirements

- data requirements
- functional requirements

Also requirements for security, performance, ...



Designing the database

2. **Conceptual design.** "Interpretation" of all the requirements to a formal presentation (conceptual model).

Result: conceptual schema, also textual documentation is possible/recommended (to make sure all the knowledge is documented)

This is a tool for communication with end users.



Requirement analysis and Conceptual design

- * Collecting requirements and analyzing them
- Fact-finding: interviews, questionnaires, existing documentation,... (recordings)
- Requirements specifications
 - * data requirements
 - functional requirements (performance, security, backup/recovery,..)
- * Completely neutral to any technology



Requirement analysis and Conceptual design

* Why entity-relationship model (ER)

- Defining the tables directly based on requirements can be too difficult and lead to a wrong db schema.
- Based on a good ER it is easy to generate the relational model (which is at least on 3NF)
- * Data Flow Diagrams (DFD)



Conceptual design

- Use right terminology and clear names, much easier to communicate with the end users (one of the reasons to model!)
- Try to find and understand the main concepts and their relationships (these are the most difficult to change during the iterations)



Conceptual design

* Modeling is **difficult** because

- * Spoken/written language is not exact
- Usually all the "important" things are those that "everybody knows" so they are not told.
- * At this stage we do not know one important thing: how the data will be **retrieved**? That will be on iteration 9...
- * Modeling is mandatory because when modeling the database designer realizes what must be asked!



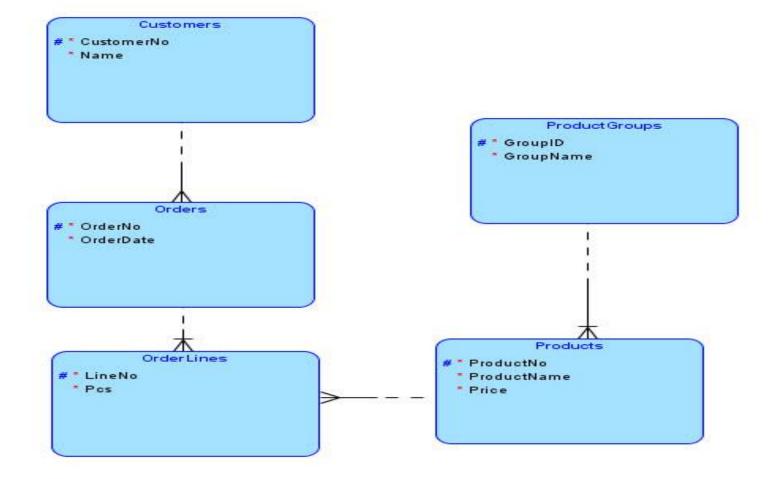
Logical model



Demo (Logical)









Notation

e <u>E</u> dit	2L Developer Data Modeler : Lo View Team Tools Help		
k}		<u>۲</u>	> 🔍 🔍 🕀 🖓 🍪
Browser	✓ Show Status Bar		
	Browser	Alt-B Alt-N	Croter erition er Coate
	E Log	Alt-L Ctrl-E	
	Show <u>O</u> nly Editor <u>Eiles</u>	Ctrl+Shift-Enter	Customer None Discription
	Logical Diagram Notation Vie <u>w</u> Details	•	Barker Notation Bachman Notation
	DDL Preview DDL File Edito <u>r</u>	Alt+Shift-I Alt+Shift-L	Information Engineering Notation
	Soom In Zoom Out	F7	
	Fit Screen	F8	
····· 📴 '	Default Size	F5	
	<u>මී M</u> odel Search <u>මී G</u> lobal Search	Ctrl-F	
	Refresh	Ctrl-R	
	Eull Screen		



Domains administration

ta Mod	deler : Logical (OTNEMEA)	
Tools	<u>H</u> elp	
De Da	mains Administration	Alt+Shift-D
🕹 Ту	pes Administration	Alt+Shift-T
🔊 RD	BMS Site Administration	Alt+Shift-B
19 M	ask Templates Administration	Alt+Shift-S
	ble To View Wizard	Alt+Shift-A
Vie Vie	ew To Table Wizard	Alt+Shift-V
AP9 Ту	p <u>e</u> s To Domains Wizard	Alt+Shift-P
Na	ame Abbreviations	Alt+Shift-N
GI GI	ossary Editor	Alt+Shift-Y
<u>o</u>	oject Names Administration	
De	esign <u>Rules And Transformations</u>	
Co	mpare/Merge Models	Alt+Shift-M
Ee	atures	
Pr	eferences	



Domains Administration

\sim
-

	Select Logical type	Comments
Available Domains	unknown	
Alg Unknown	Size	Unit of Measure
	Units	Default Value
	Precision	▼
		Check Constrain
	Scale	
		Ra <u>n</u> ges
	Domains File	
		▼ <u>V</u> alue List
<u>A</u> dd <u>R</u> emove	Modify	
	ı	Save Close

Domains Administration

Choose domain	Domain Properties	
Domains File	Name	Synonym
Select	Money	
	Logical type	Comments
Available Domains	NUMERIC	▼]
Money	Size	Unit of Measure
A Unknown	15	
	Units	Default Value
	CHAR	▼
	Precision	
	15	Check Constraint
	Scale	
	2	Ranges
	Domains File	
	defaultdomains	Value List
Add <u>R</u> emove <u>M</u> odify		
	Save	<u>C</u> lose <u>H</u> elp



Entity Properties - OrderLine						x
General						
Attributes			Attribut	es		
Unique Identifiers						
Relationships	Details Overview					
·····Volume Properties	Attributes:		Attribute Propertie	S		
Engineer To						
·····Comments	🔛 l 🕂 💥 🖓 🍕) I 🗈 💼 👘	Name:	Price		
Comments in RDBMS		2.1.1	Detet	O Denvia		
Overlapping Attributes	Name	Data type	Datatype:	<u> </u>	Logical Distinct	
Notes	1 OrderlineNo 2 Pcs	NUMERIC (10) NUMERIC		O Structured (Collection	
Impact Analysis	3 Price	Money	Type:	Money	✓ Preferr	
Measurements	4 OrderNo	NUMERIC	Type.	Money	Ficien	
Change Requests	5 ProductNo	NUMERIC (10)				
Responsible Parties						
Documents						
Dynamic Properties						
Summary			Pri	imary UID 📃 Relati	ion UID Mandatory	
			Comments Co	omments in RDBMS Note	es	
			/ Na	aming Rules	Cancel H	elp
						//



Column Properties - Price		×
General Default and Constraint Auto Increment		General
·····Security ·····UI Defaults	Name:	Price
·····Permitted Subtypes ·····Comments	Table:	OrderLine
·····Comments in RDBMS	Abbreviation	
····Impact Analysis	Engineer:	
·····Measurements ·····Change Requests	Allow Nulls:	
·····Responsible Parties ·····Documents	Datatype:	NUMBER (15,2) {Money}
Dynamic Properties Summary	Type:	•
	Computed:	
	Column Expression:	
	Auto Increment	Identity Column
	Scope:	· · · · · · · · · · · · · · · · · · ·
	Type Substitution:	ALL
		pply <u>C</u> ancel <u>H</u> elp



Design Properties



😼 Design Properties - Riga					×
General G-Settings		Naming	Standard		
Compare Mappings Compare Mappings Compare Mappings Compare Mappings Compare Mappings Column C	Logical Model Separator Space Relational Model Separator Abbreviated Domain Separator Glossary X	Only			
User Defined Propertie Comments Notes Summary	Name	Description		File	
<u>∢(</u>			QK Ar	pply <u>C</u> ancel	Help



General Settings	Templates			
Compare Mappings	Table Constraints			
⊕…Diagram ⊡…DDL	Primary Key	{table}_PK	Add Variable	
Migration	Foreign Key	{child}_{parent}_FK	Add Variable	
Naming Standard	Check Constraint	{table}_CK	Add Variable	
Attribute	Unique Constraint	{table}_{column}_UN	Add Variable	
Column Domain	Index	{table}_{column}_IDX	Add Variable	
Entity	Automatic Index	{table}_{column}_IDX	Add Variable	
Table	Column Check Constraint	CK_{table}_{column}	Add Variable	
Templates	Not Null Constraint NNC_{table abbr}_{column}			
Dynamic Properties User Defined Propertie	Column Foreign Key	{ref table}_{ref column}	Add Variable	
Comments	Surrogate Key	{table abbr}_PK	Add Variable	
Notes	Surrogate Key Column	{table abbr}_ID	Add Variable	
Summary	Discriminator Column	{table abbr}_TYPE	Add Variable	
	Entity identifier			
	Primary Identifier {entity	PK	Add Variable	
	Attribute Relation {ref en	iity}_{ref attribute}	Add Variable	
	Example			
	Example		Primary Key	
) - + (



Preferences

Preferences	
Q Search	Model: Logical
 Environment Data Modeler DDL DDL/Comparison DDL/Storage Diagram Model Physical Relational Synchronization Physic Reports Search SQL Formatter Third Party JDBC Drivers Global Ignore List Mouse Actions Shortcut Keys SSH Usage Reporting Versioning Web Browser and Proxy 	Relation Cardinality Source Optional Target Optional Primary Key option for Identifying Relationships Use And Set First Unique Key As Primary Key FK Attribute synchronization V Name - Keep as the name of the Originating attribute Comments, Notes - Automatically propagate from PK attribute Default Surrogate Key Settings Entity Create Surrogate Key Relationship Use Surrogate Key
Help	OK Cancel



Designing the database

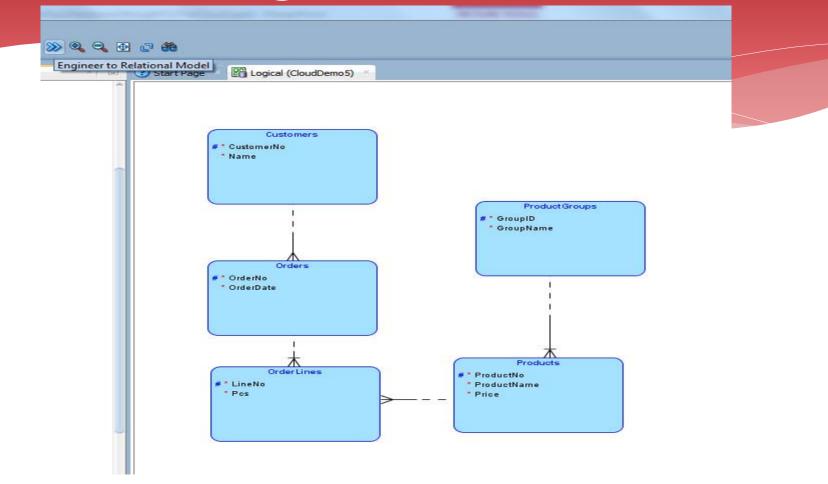
3. Logical design: transforming the conceptual model into a logical data model and a logical schema that the RDMS understands

Result: relational-database schema

(relational schemas and constraints)

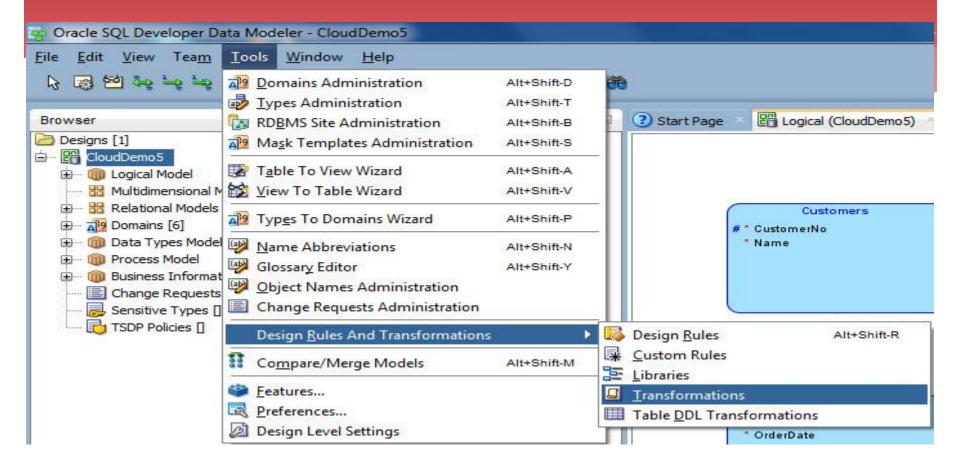


From logical to relational





Transformations





Adding technical columns

Engineer to Relational Model		
Tree View Tabular View		
Logical	Eilter	Relational_1
Logical Logical Entities Relations V Relations V V V Subviews Subviews		Relational_1 Tables Tables mapped to Hierarchies Objects mapped to relations Views Subviews Subviews
Details General Options Compare/Copy ✓ Engineer Coordinates Engineer only objects created in "Logic ✓ Apply name translation ✓ Use preferred abbreviations Use Totata Type Kind" Property in Com ✓ Prefix for Columns fr	cal* model	ion of deleted objects Overlapping and folding keys
Apply template for FlOrderLines Orders ProductGroups Products		



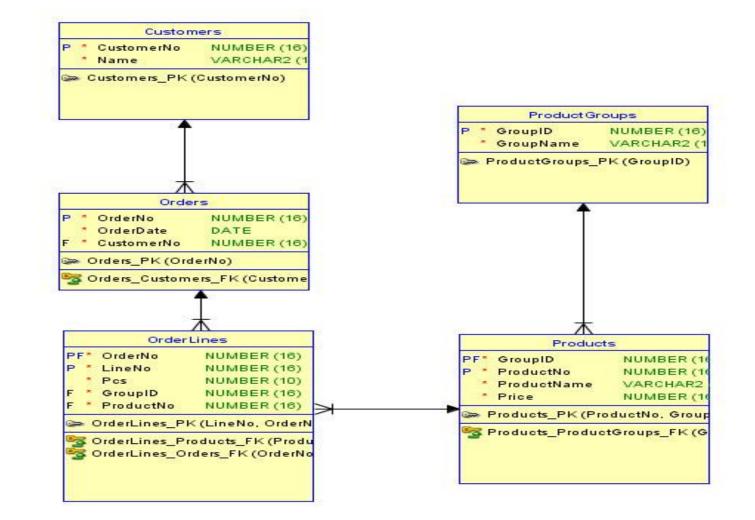
Demo (Relational)





* Check that you got what you wanted...







Indexes



Manually

Table Properties - Custome General	ers						×
Columns			Index	es			
Primary Key							
Unique Constraints <mark>Indexes</mark>	👳 💠 💥 🔂 🕹						
Table Level Constraints	Name	Unique	Generate Engir	eer Spatial	Deprecated	Supported Constraint	1
Existence Dependencies							
Foreign Keys							
Nested Columns Valid Time Dimensions							
Materialized Query							
Volume Properties							
Spatial Properties	Index Expression						
Column Groups	Columns			Selected			
Comments	CustomerNo						Desc
Comments in RDBMS	Name		~	Column			Desc
Notes			\$\$				
Impact Analysis			· · · · · · · · · · · · · · · · · · ·				
Measurements			1				
Change Requests			-n	1			
Responsible Parties			3				
Documents							
Scripts Dynamic Properties	Comments in RDBMS Notes	Comments					
User Defined Properties							
Redaction Policy							
Classification Types							
Summary							
		1		Namine Dr	dee	Canaal	Hala
	QK	Appl	У	Naming Ru	lies	Cancel	Help



Automatically

General	1				
Settings	DDL				
Compare Mappings Diagram DDL Compare Mappings Compare Mappings Compare Mappings Compare Mapping Comp	Automatic Index Generation Primary Key Constraint Unique Key Constraint	i f			
Attribute	Foreign Key Constraint				
Column Domain Entity	Preserve Current DDL Generation Options Selection Preserve DDL Generation Options				
Table Templates User Defined Propertie Comments Notes Summary					
			Cancel		



Designing the database

4. Physical design: instances, tablespaces, indexes, disks ...

And all of these phases over and over again...(this is different, we have always done that but not so many times and in such a short cycles)

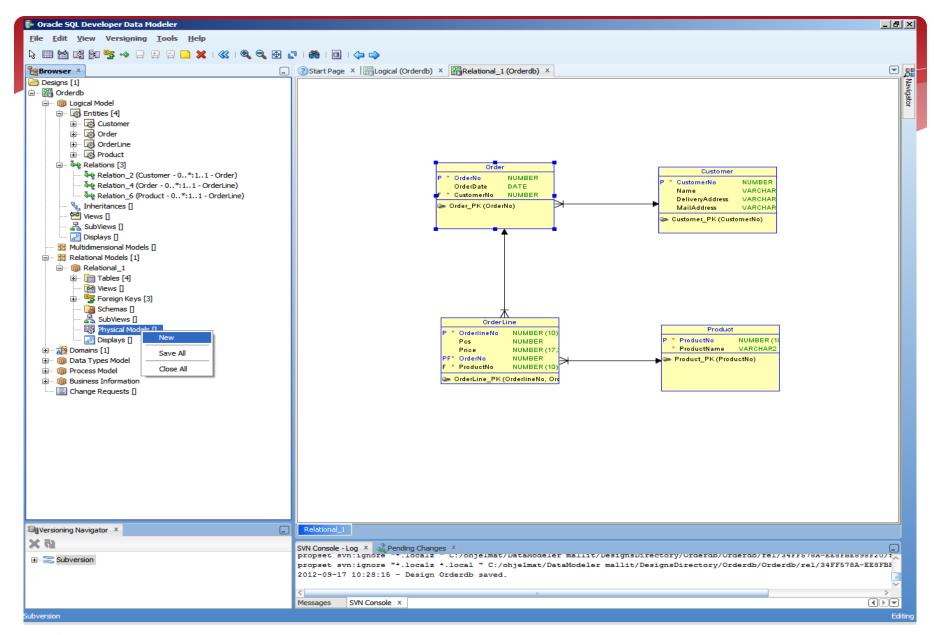


From relational to physical



Demo (Physical)







🚁 Database Sites	
Select Site:	
Oracle Database 12cR2	
Oracle Database 12c	
Oracle Database 11g	
Oracle Database 10g	
Oracle9i	
SQL Server 2012	
SQL Server 2008	
SQL Server 2005	
SQL Server 2000	
DB2/390 8	
DB2/390 7	
DB2/UDB 9	-
ок	Cancel





Physical Design

- * Creating physical elements (tablespaces, users, ...)
- * Privileges...



Setting up the Properties



🕞 TableSpace Prope	rties - TableSpace	1	×
Indexes Storage	Materialized Views	Comments	
General	Default Storage	Clusters	Tables
Name:		TableSpace 1	
Datafiles:		NO	
File Type:			-
Min. Ext Len:		0	
Block size:		0	
Logging:		LOGGING	
Force Logging:		NO	-
Status:		ONLINE	
Contents:		PERMANENT	_
Default Compression:			_
Ext. Management:		LOCAL	_
EM Local SubType:		AUTOALLOCATE	
EM Uniform Size:		0	
Seg. Management:			
Flashback Mode:		ON	
Propagate Prop	perties		
<u>0</u>		oply <u>C</u> ancel	



Setting the Defaults

Preferences

Environment
Data Modeler DDL DDL/Comparison DDL/Storage Diagram Dot/Storage Diagram Model Cogical Physical Relational Synchronization Physic Reports Search SQL Formatter Third Party JDBC Drivers Global Ignore List

Model: Physical			
	leli		
Default Tablespace:	March America		
Use Table Template	View/Edit View/Edit		
Auto Increment Column Te	emplates		
Default Trigger Name:	{table}_TRG		Add Variable
Default Sequence Name:	{table}_{column}_SEQ		Add Variable
Default Auto Increment DD	DL: Trigger	-	
Default Identity DDL:	IDENTITY dause	· •]	

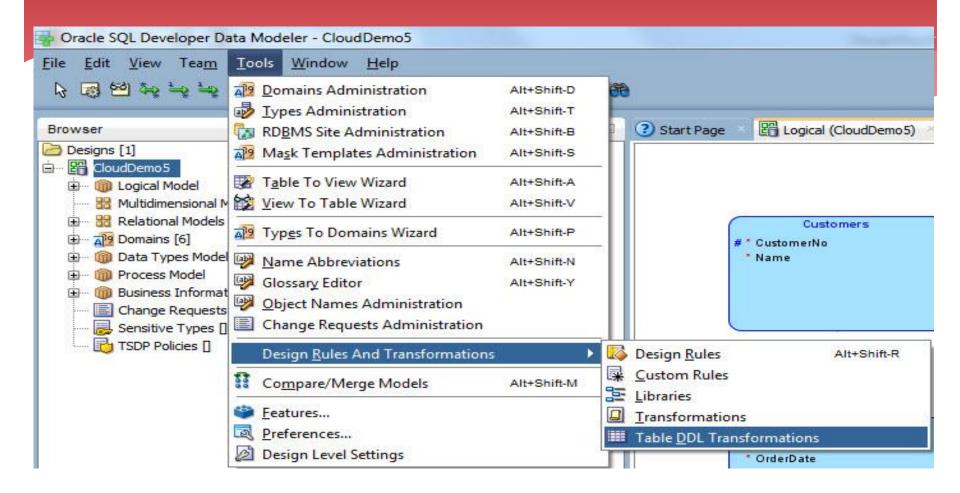


Copyright © Miracle Finland Oy

Physical to DDLs



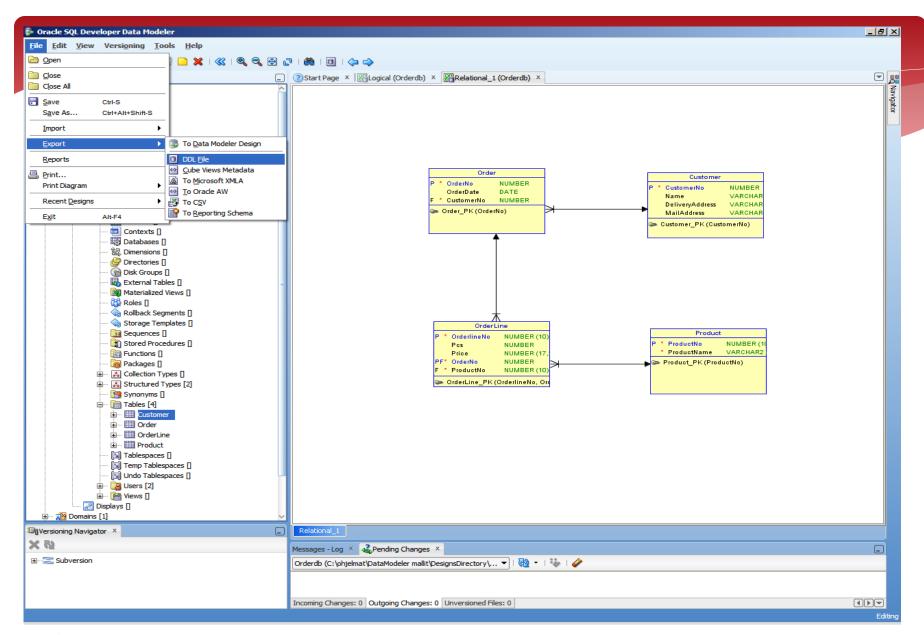
Table DDL Transformations





Demo (DDL)







DDL Generation	n Options	- Dark Testing	-					×
'Create' Selection	'Drop' Selection	Name Substitution	Include Table Scripts	Include Table DDL Scripts	Oracle Errors to I	Mask		
🛃 🧮 Filter			Active Script Set:	Journal tables	-			
Insert Scripts				Name				
1				Customers				
				OrderLines				
				Orders				
				ProductGroups				
				Products				
📓 Desig	n <u>R</u> ules [Include Comments	Apply Name Substitu	ution Regular DDL	- Generate	DDL in Separate File	es 🔄 Generate REST	Services
			Sa	ve Configuration	d Configuration	<u>0</u> K	<u>C</u> ancel	Help

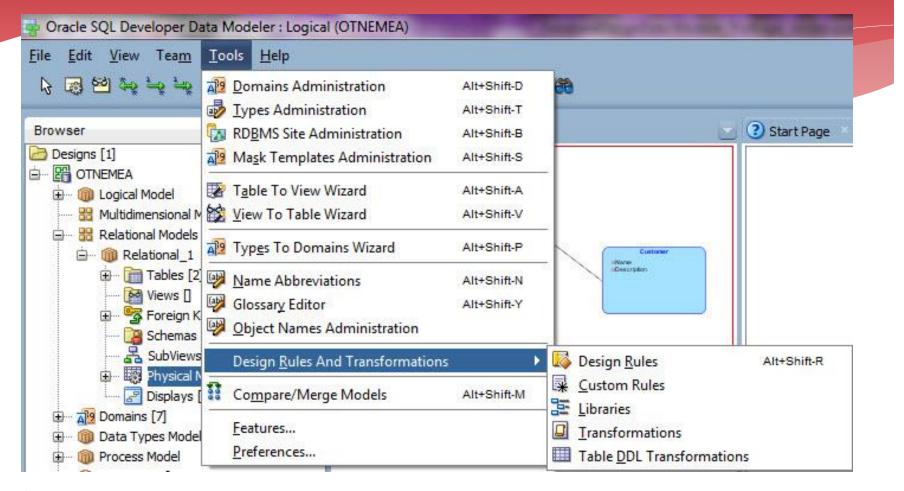


oracle Database 12cR2	 Relational_1 	-	<u>G</u> enerate	Clear		
21 CREATE TABLE Customer	rs					
22 (
23 CustomerNo NUMBE	ER (16) NOT NULL ,					
24 Name VARCH	HAR2 (100 CHAR) NOT	NULL				
25)						
LOGGING						
7 ;						
8 CREATE UNIQUE INDEX C	Customers_CustomerNo_	IDX ON Customer	rs			
9 (
0 CustomerNo ASC						
1)						
2 TABLESPACE Index1	Ibl					
3						
4 ALTER TABLE customers	B ADD CONSTRAINT cust	omers_pk PRIMA	RY KEY (customern	10);		
5						
6 🖾 CREATE TABLE OrderLin	nes					
7 (
	R (16) NOT NULL,					
	R (16) NOT NULL ,					
	R (10) NOT NULL ,					
	R (16) NOT NULL,					
2 ProductNo NUMBER	R (16) NOT NULL					
4 LOGGING						
15 ;						
6 CREATE UNIQUE INDEX C	Mines LineNo OrderNo	IDX ON OrderLi	ines			
7 (James_bineno_orderne	_IDA ON VIUCIDI	inco -			
8 LineNo ASC ,						
9 OrderNo ASC						

Demo (executing the DDLs to the Cloud)



Design Rules





📮 Design Rules

Design Rules Rule Sets	Results					
Expand All Collapse All						
C Rules						
🖨 🗁 General						
🖮 🗁 Domain						
Identify domains with wrong naming standards						
🖶 🗁 Logical						
🖨 🔁 Entity						
Identify entities without attributes						
Identify entities without relationships						
Identify entities without unique identifiers						
▲ Identify entities without primary unique identifier						
▲ Identify entities with wrong naming standards						
▲ Identify primary identifier with wrong naming standa						
 Check for entity maximum name length 						
△ Check for entity name case type						
Check for entity name with invalid characters						
Identify attributes without datatype						
Identify attributes with wrong naming standards						
Check for attribute maximum name length						
Check for attribute name case type						
Check for attribute name with invalid characters						
Check for attribute default values compiled with per						
Relational						
i Ceautina						
Identify tables without columns						
Identify tables without countris						
Identify cables without primary key Identify correct creation of spatial tables.						
Identify confect dealer of spada tables. A Identify tables with wrong naming standards						
Check for table maximum name length						
Check for table name case type						
Check for table name with invalid characters						
Identify tables with columns with undefined security						
Complex rule - check comments demo						
Identify columns without datatype						
Identify countris without datatype Identify circular refering columns						
Identify circular referring countris A Identify columns with wrong paming standards			Clear	Save as CSV		
		Apply All		Apply Selected	Close	Help
				Apply Delected		

Miracle Finland Oy

© Miracle Finland Oy

×

🚰 Design Rules

Design Rules Rule Sets	Results
	ERRORS:4WARNINGS:13
Expand All Collapse All	Warning: Customer: Table has columns with undefined security properties
Rules	Error: Customer: no comments in RDBMS defined, no comments defined
🖶 🛅 General	Warning: Order: Table has columns with undefined security properties
🗄 🗝 Logical	Error: Order: no comments in RDBMS defined , no comments defined
🖶 🗝 Relational	Warning: OrderLine: Table has columns with undefined security properties
🖶 🗝 Process Model	Error: OrderLine: no comments in RDBMS defined , no comments defined
🗄 🗝 Physical	Warning: Product: Table has columns with undefined security properties
	Error: Product: no comments in RDBMS defined , no comments defined
	Warning: Customer_Customer_PK: Primary key with wrong naming standards is not recommended.
	Warning: Order.Order_PK: Primary key with wrong naming standards is not recommended.
	Warning: Order.Relation_2: Foreign keys with wrong naming standards are not recommended.
	Warning: OrderLine.OrderLine_PK: Primary key with wrong naming standards is not recommended. Warning: OrderLine.Relation_4: Foreign keys with wrong naming standards are not recommended.
	Warning: OrderLine.Relation_4: Foreign keys with wrong naming standards are not recommended. Warning: OrderLine.Relation_6: Foreign keys with wrong naming standards are not recommended.
	Warning: Order Line: Relation_0.1 oregin keys with wrong naming standards is not recommended. Warning: Product.Product_PK: Primary key with wrong naming standards is not recommended.
	Warning: Order: The name is a reserved word: Order
	Warning: Order: The name is a reserved word: Order
	Clear Save as CSV
	Apply All Apply Selected Close Help



Different Compares

- * Design vs Design
- Design vs Database
- Database vs Design



Different Compares, Designs

- * File | Import | Data Modeler Design
 - * Compares "everything"
- * Tools | Compare/Merge Models
 - * Compares only relational and physical models
 - * ALTER DDLs can be generated
- These can be used for instance to compare different versions of the design from version control

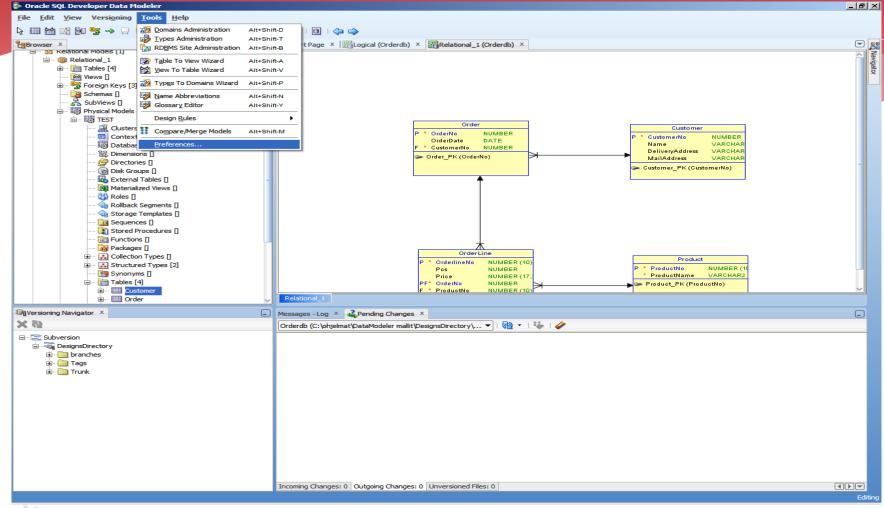


Different Compares, Database

- * Synchronize Model With Data Dictionary
 - * Target: model
- * Synchronize Data Dictionary With Model
 - * Target: database
- * File | Import | DDL File
- * File | Import | Data Dictionary



Preferences





© Miracle Finland Oy

Preferences			and the second second	×
Q	Environment			
 Environment Data Modeler Global Ignore List Mouse Actions Shortcut Keys SSH Usage Reporting Versioning Web Browser and Proxy 	Automatically R	Deactivating or Exiting Reload Externally Modified Files eload When File Is Unmodified rnally Modified Files on Startup 50 20 Oracle Platform Default Applies to new files only ISO-8859-2 Messages		
Help			ОК	Cancel

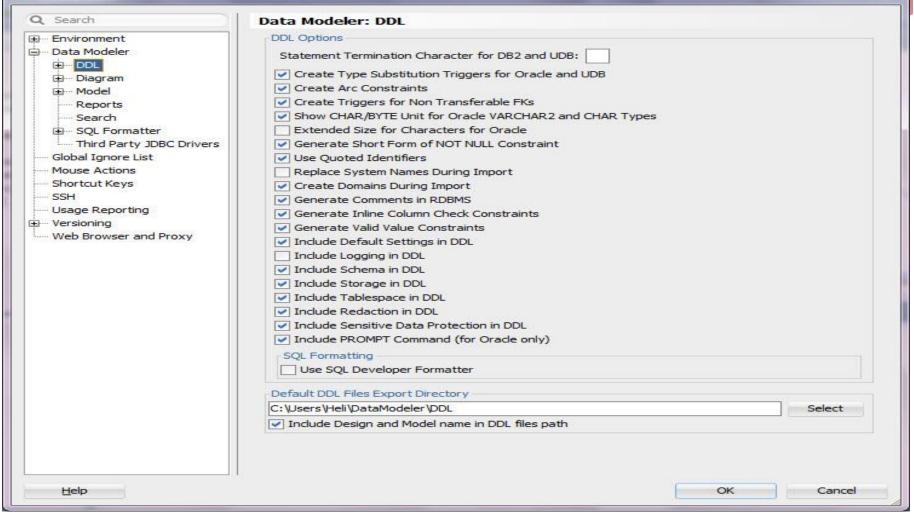


100	Dre	ofe	rei	ne	es
100	E14	e i e	10	IIC.	C 2

Preferences						
Q Search	Data Modeler					
± Environment	Default Designs Directory					
Data Modeler Global Ignore List Mouse Actions	C:\Users\helhel\DataModelerMallit	Select				
	Default Import Directory					
		Select				
Shortcut Keys SSH	Show Log After Import	Joceee				
	Default Save Directory					
·· Versioning	C:\Users\helhel\DataModelerMallit\Reports					
Web Browser and Proxy	C:\Users\helhel\DataModelerMallit\Reports Select					
		Select				
	- Miscellaneous					
	Show "Select Relational Models" Dialog					
	Show Properties Dialog On New Object					
	Reload Last State					
	Use Versioning Functionality					
	Use OCI/Thick driver					
	TNSnames Directory					
		Select				
	Import Export					
Help		OK Cancel				



Preferences



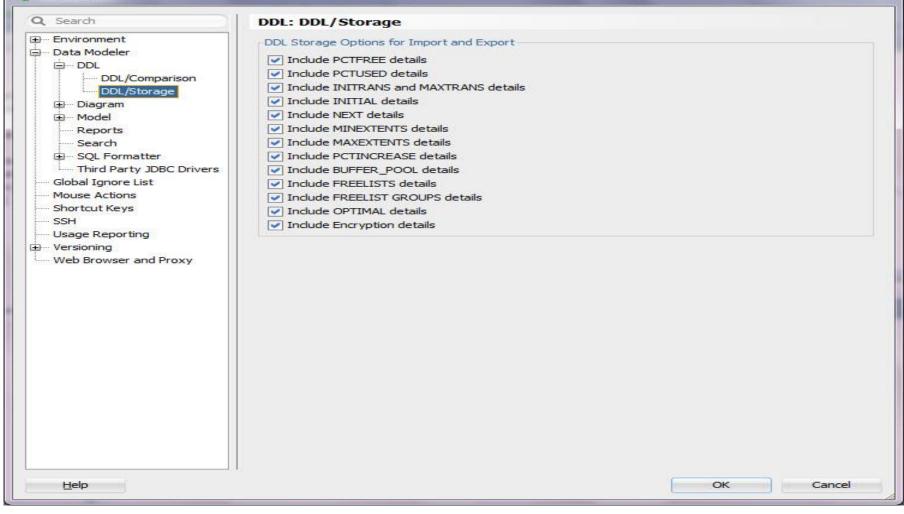
×



Preferences			×
p Preferences			
Q Search	DDL: DDL/Comparison DDL Comparison Options Include Physical Properties in Compare Functionality Include Storage Properties in Compare Functionality		
DDL/Storage □ Diagram □ Model ···· Reports ···· Search	 Case Sensitive Names in Compare Functionality Include System Names in Compare Functionality Use "Data Type Kind" Property in Compare Functionality Use "Schema" Property in Compare Functionality Use "Columns Order" Property in Compare Functionality 		
SQL Formatter Third Party JDBC Drivers Global Ignore List Mouse Actions Shortcut Keys SSH Usage Reporting Versioning Web Browser and Proxy			
Web browser and Proxy			
Help		ок	Cancel



Preferences



23



Preferences

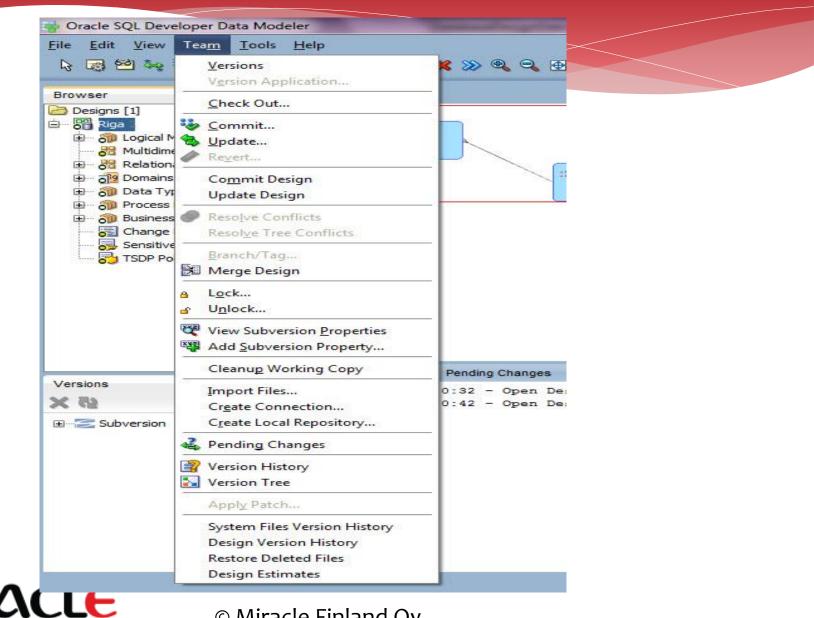


Q Search Data Modeler: Model . Environment **RDBMS Settings** 🖻 -- Data Modeler Oracle Database 12c 🔻 Default RDBMS Type DDL Default RDBMS Site Oracle Database 12c -DDL/Comparison Columns & Attributes Defaults DDL/Storage . Diagram Nulls Allowed Model Datatype: O Domain O Logical O Distinct O Structured O Collection - Reports FK Columns Search Allow Similar Data Types for Foreign Keys E- SOL Formatter ----- Third Party JDBC Drivers On New Foreign Key Existing By Template 🔻 Global Ignore List Preferred Domains & Logical Types Mouse Actions Preferred Logical Types Preferred Domains Shortcut Keys Preferred Logical Types SSH All Logical Types Usage Reporting unknown Date Versioning Audio Datetime BFile NUMERIC Web Browser and Proxy BIGINT VARCHAR BINARY BINARY DOUBLE BINARY FLOAT BIT 0 BLOB Boolean 0 CHAR CLOB DATALINK DATETIME2 DBURIType DECIMAL DECFLOAT DOUBLE Help OK Cancel

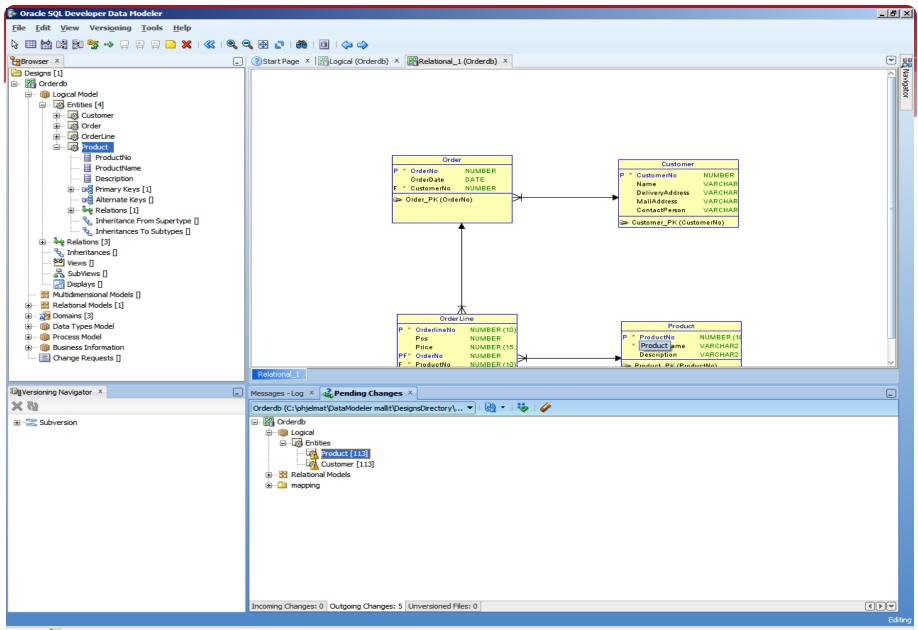
Miracle Finland Oy

Version control (Subversion)

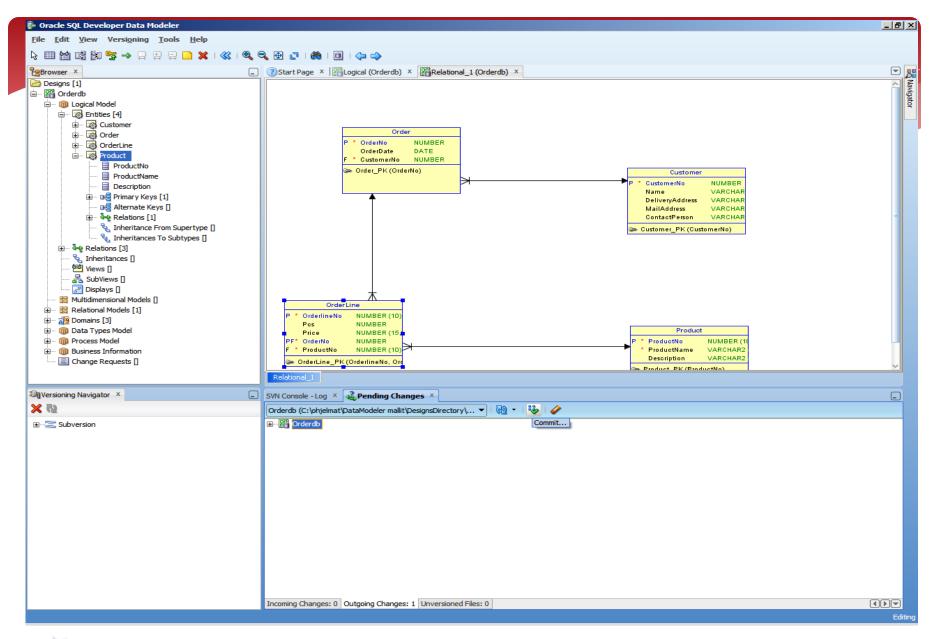




Finland Ov



Complex Properties			Product (Local Saved) Complex Properties attributes (0) ProductNo (1) ProductName Complex Properties (1) ProductName (1) ProductName (
Details					-
Property	Selected				
name	 Image: A start of the start of		Description		
id	Image: A start of the start			4-590E-3A31-BB36CAA16	
De l'en Marine	 Image: A start of the start of	6-las	Orderdb		
	~	false	true		
ownerDesignName nullsAllowed		true	false		
nullsAllowed useDomainConstraints	✓			1	
nullsAllowed useDomainConstraints Data Type Kind	Image: A start of the start	OCDT017	Logical Type	CDT024	
nullsAllowed useDomainConstraints Data Type Kind logicalDatatype	✓	LOGDT017	Logical Type VARCHAR / LO	GDT024	
nullsAllowed useDomainConstraints Data Type Kind logicalDatatype dataTypeSize	✓✓✓	LOGDT017	Logical Type VARCHAR / LO 200	GDT024	
nullsAllowed useDomainConstraints Data Type Kind	✓	LOGDT017	Logical Type VARCHAR / LO		





Commit Resources					
Name 🔺	Location		Status	Select	
🐉 Customer	Logical \Entities		М	✓	4
] Keep Locks					
omments:					
					Ī
emplate or Previous Comments:					
<select></select>					-
nis list is configured with <u>comment templates</u>					
Ohje		ОК		eruuta	_



Conclusions

- * To be able to connect to the database, to see its content online and to maintain the data: SQL
 Developer
- * To design the database and to maintain the data structures: **Data Modeler**
- * To design the data architecture: **Data Modeler**



Conclusions

- Data Modeler is a good tool; good support for iterative processes
- Enables documenting and versioning (and comparing the versions)
- Enables multiuser environment
- * Is free to use
- Support for other databases as well
- * I see no reason not to use it!



THANK YOU!

QUESTIONS?

Heli: Email: heli@miracleoy.fi Twitter: @HeliFromFinland Blog: Helifromfinland.com

