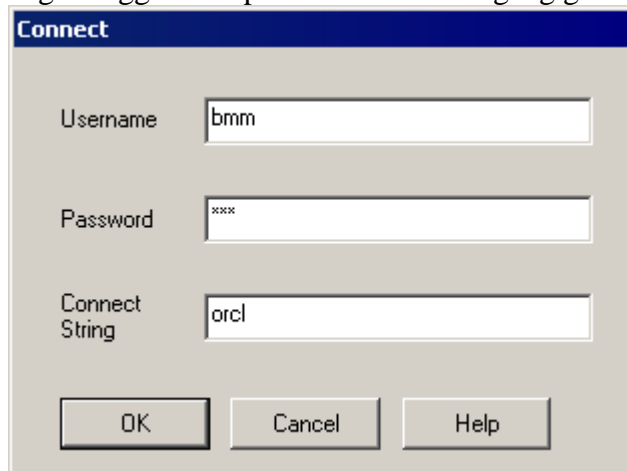
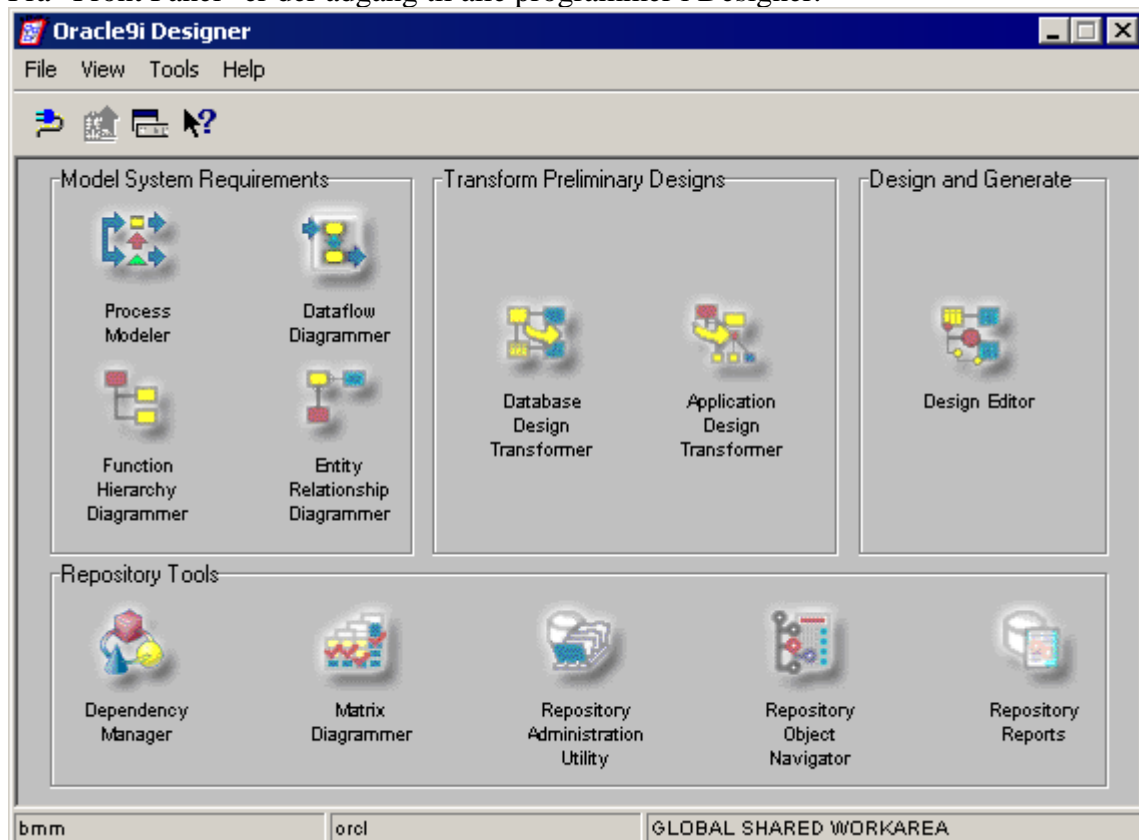


## Anvendelse af Oracle Designer 9i

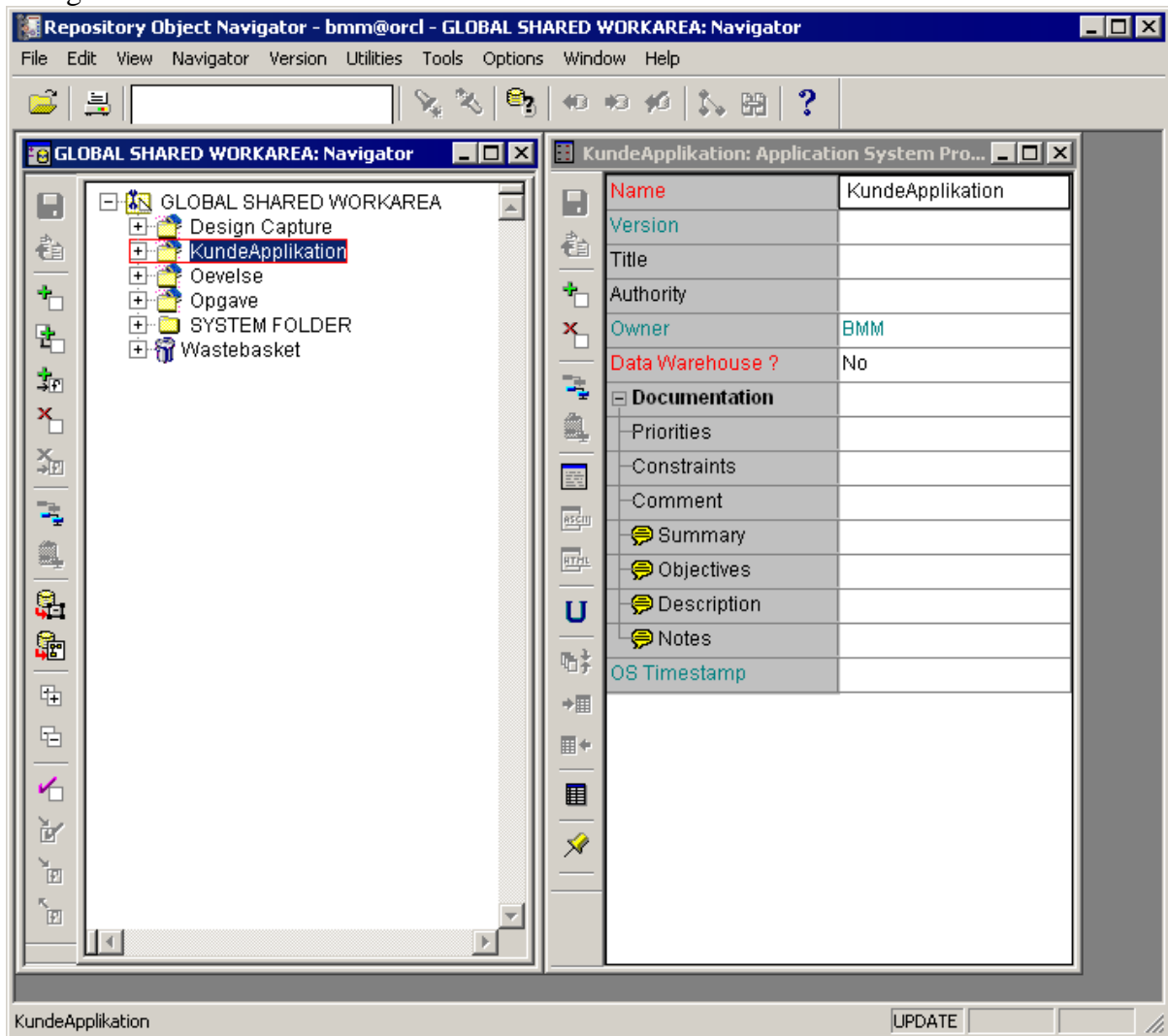
Brugeren man anvender til at logge på Designer med skal I forvejen have adgang til det bagvedliggende repository. Denne adgang gives fra Repository Administration Utility.



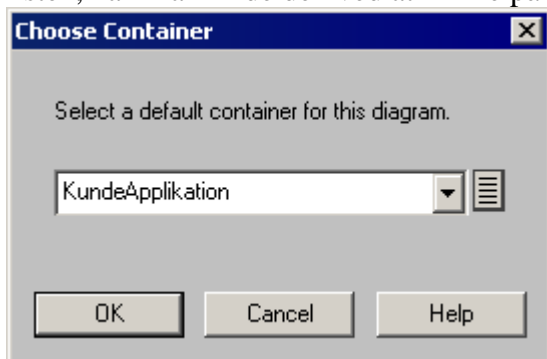
Fra "Front Panel" er der adgang til alle programmer i Designer.



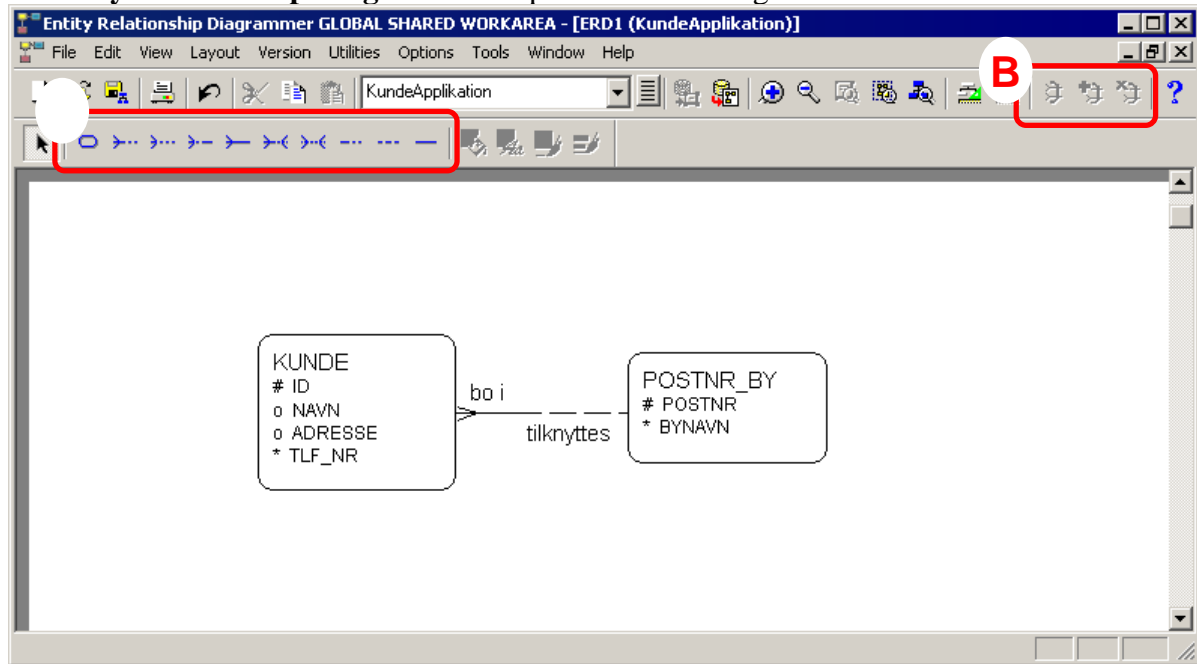
Den første opgave til et nyt projekt i Designer er at oprette en Applikation, som alle entiteter, processer, diagrammer m.m. oprettes i. Dette sker i Repository Object Navigator, hvor man markerer GLOBAL SHARED WORKAREA, højreklikker på denne og vælger "Create Child". I det nye vindue skal man vælge at oprette et Application System, sige OK og navngive dette.



Når man efterfølgende skal til at arbejde i en af Designers del-programmer, skal man altid som det første vælge at arbejde i den relevante Applikation/Container. Hvis den ikke lige er i listen, kan man finde den ved at klikke på den lille knap med de vandrette streger.



I "Entity Relationship Diagrammer" oprettes et E/R-diagram over databasen.

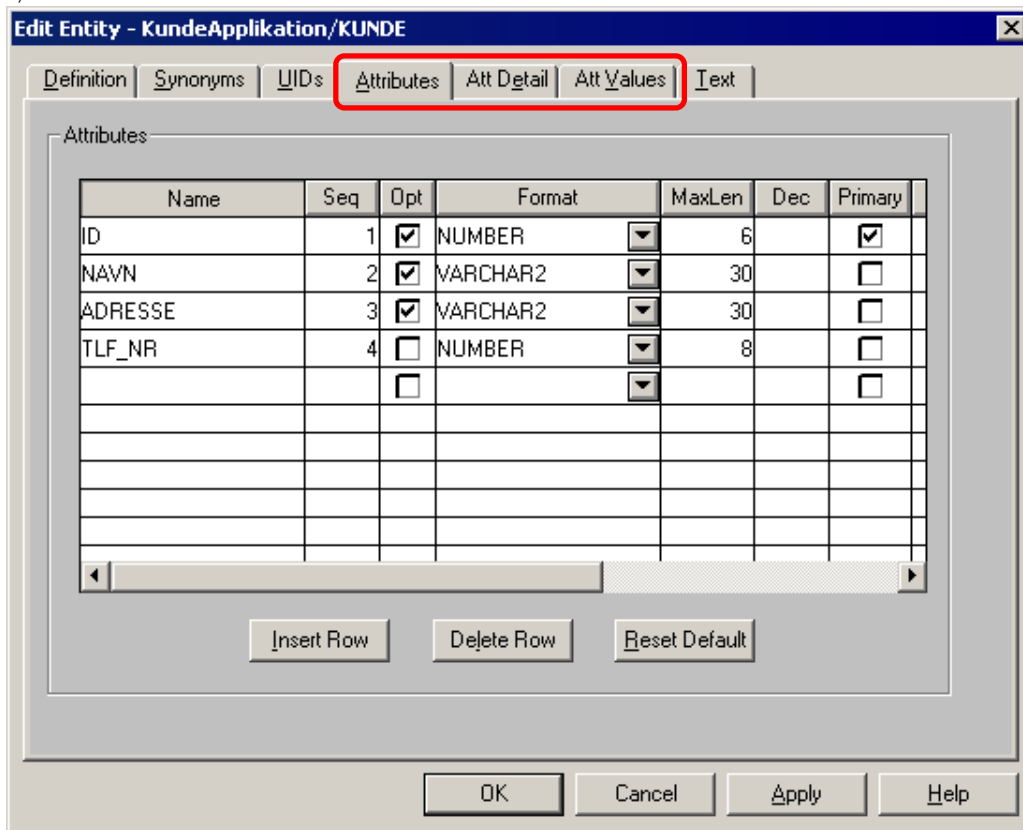


Entiteter skabes vha. den første knap i ramme A, mens relationer mellem entiteter kan laves vha. de andre knapper i ramme A.

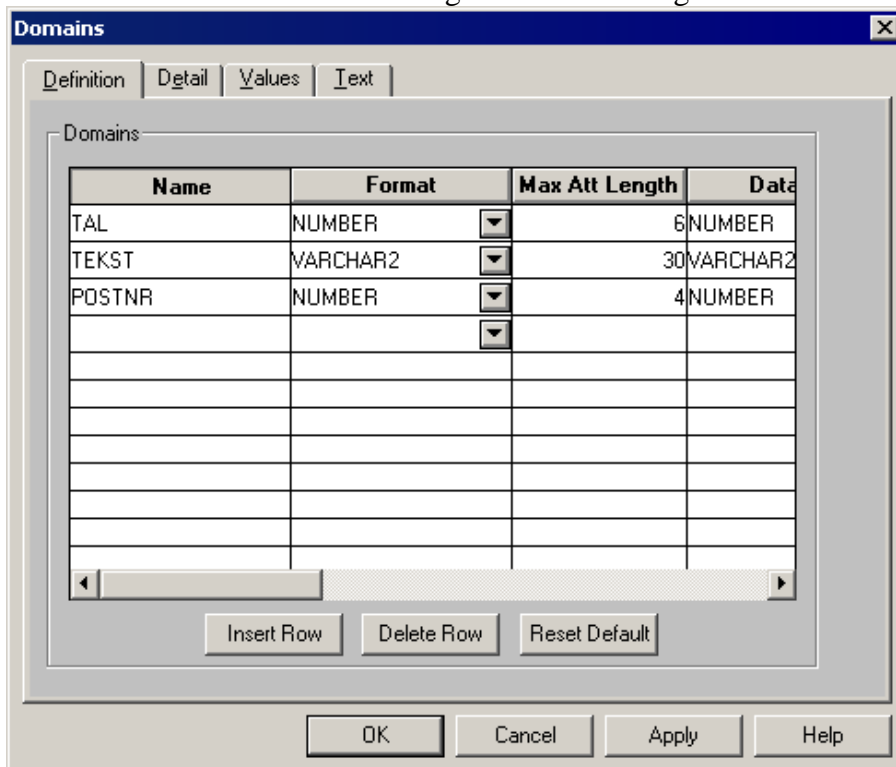
En "Exclusive Arc" dannes ved at markere to eller flere relationer i E/R-diagrammer og trykke på den første knap i ramme B. De øvrige knapper i ramme B anvendes til at tilføje eller slette relationer fra en "Exclusive Arc".

Der kan laves et hierarki i en entitet ved at lade begge en relations ender gå til den samme entitet.

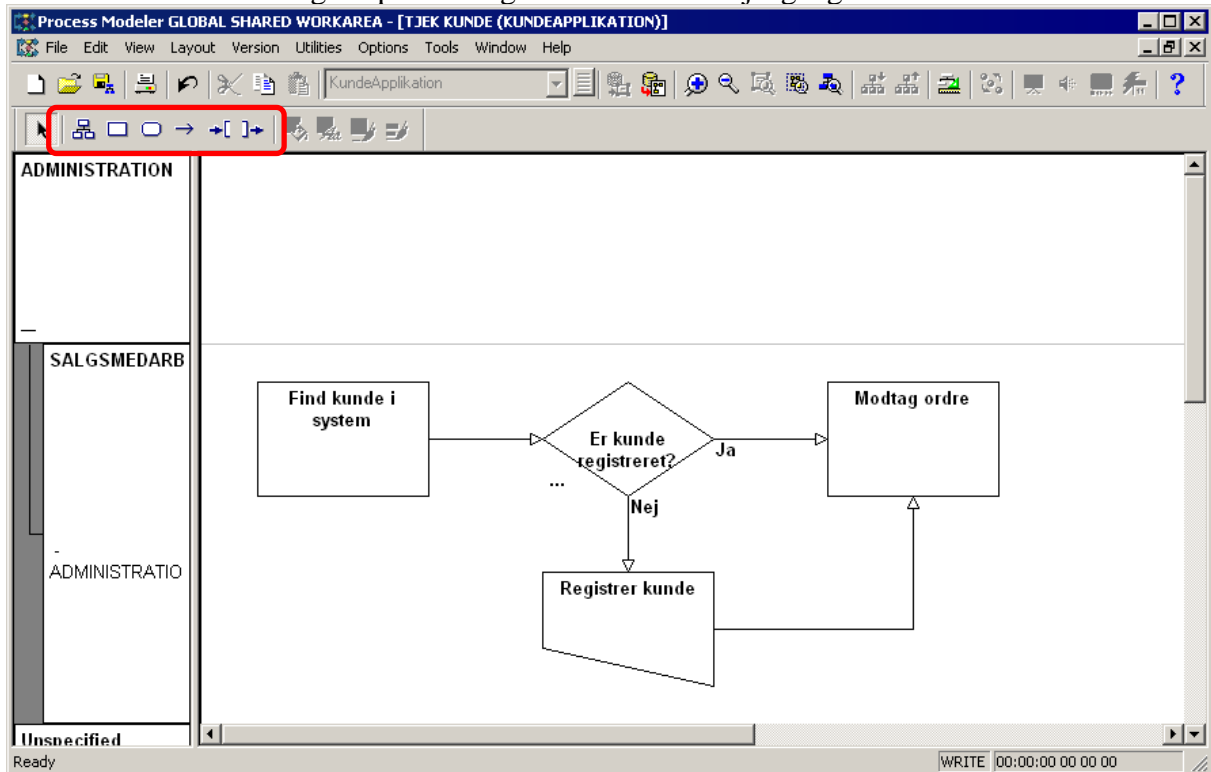
Attributter i entiteterne kan laves ved at dobbeltklikke på en entitet eller højreklikke på en entitet og vælge properties. Informationer til attributterne indtastes i de tre faneblade i den røde ramme.



Domains kan i denne sammenhæng anvendes til brug for ens attributter. (Edit → Domains)



I "Process Modeler" tegnes procesdiagrammer over arbejdsgangene i en virksomhed.



Der skal laves en ny Root Proces første gang Process Modeler tages i brug. Dette er den overordnede proces som de første egentlige processer oprettes i. I ovenstående skærbillede hedder Root Processen "Tjek Kunde", hvilket kan ses i den blå bar.

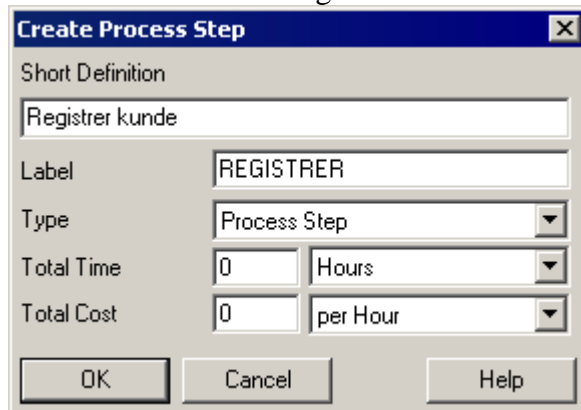
Organisationsenheder/Agents oprettes vha. af den første knap i den røde ramme. Først klikkes på knappen og derefter på kassen "Unspecified" eller andre tidligere oprettede enheder. Hvis man klikker på Unspecified oprettes enheden på "øverste niveau", mens der kan laves et hierarki af enheder, hvis der klikkes på en tidligere oprettet enhed. Hierarkier på organisationsenhederne kan også tilrettes bagefter ved at gå ind i properties for de enkelte enheder.

Organisationsenheder/Agents kan flyttes i forhold til hinanden ved at markere en enhed og derefter flytte den med "Pil op" og "Pil ned". Desuden kan enheder gøres større/mindre ved igen at markere en enhed og derefter holde "Shift" nede, mens man trykker på "Pil op" og "Pil ned".

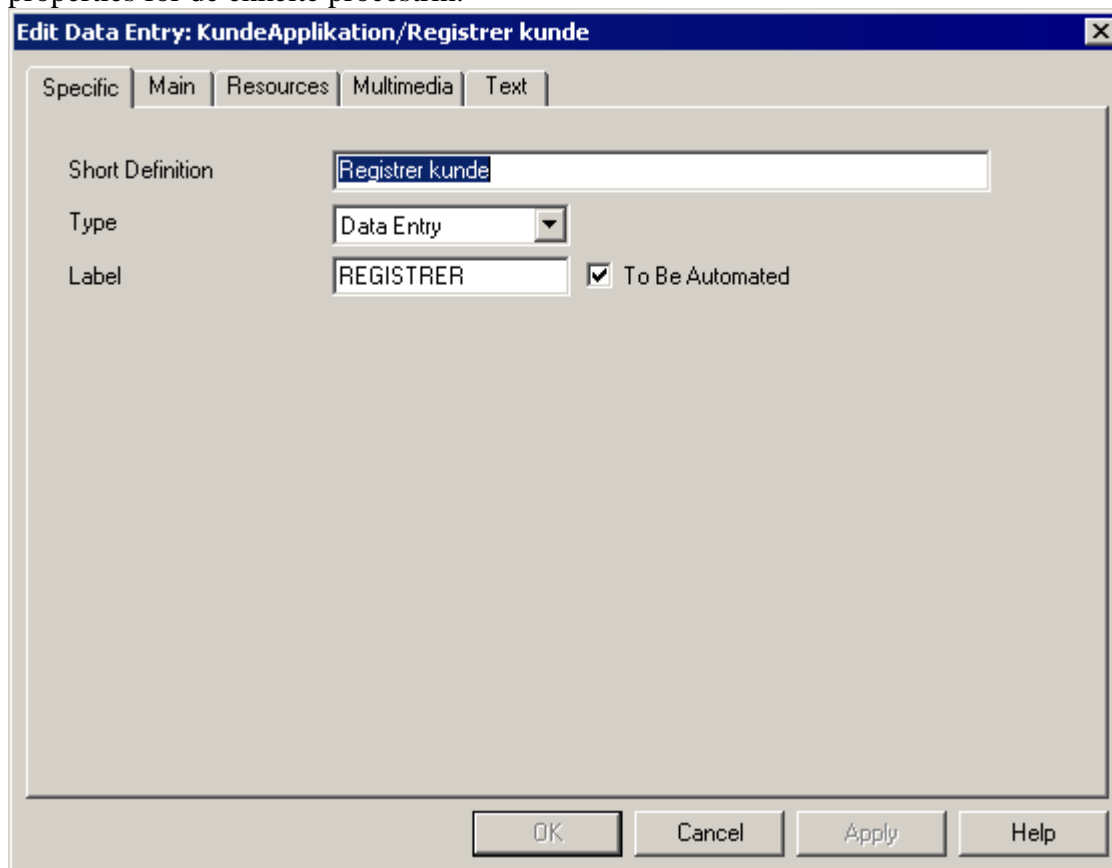
De øvrige knapper i den røde ramme bruges til henholdsvis at oprette Procestrin (gennemgås nærmere nedenfor), Stores, Flows mellem procestrin (og stores) samt Trigger til og Outcome fra procestrin. Det skal her nævnes at Stores er IKKE lig med de entiteter, hvorfra der skal hentes eller indtastes informationer. Stores anvendes kun til at angive væsentlige materielle og ikke-materielle lagre.

Husk at der kan vælges mellem forskellige måder at vise procesdiagrammet på vha. menupunkterne "Symbol" "Enhanced Symbol" og "Iconic" i menuen View. Procestrin laves ved at trykke på den anden knap i den røde ramme

I samme dialogboks som navnet på procestrinet skrives kan man vælge hvilken type, det skal være. De relevante muligheder er "Data Entry", "Decision Point", "Process Step" og "Report"

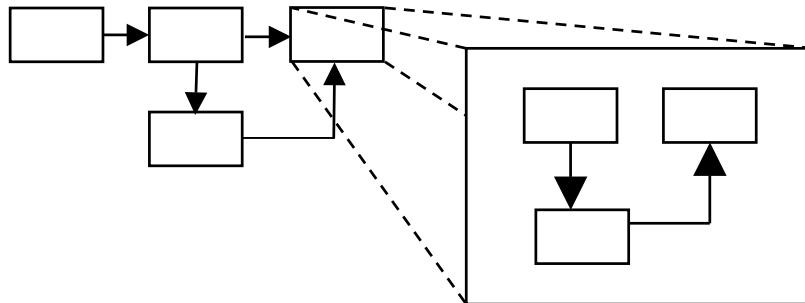


Det kan også lade sig gøre senere at ændre ovennævnte informationer ved at gå ind i properties for de enkelte procestrin.

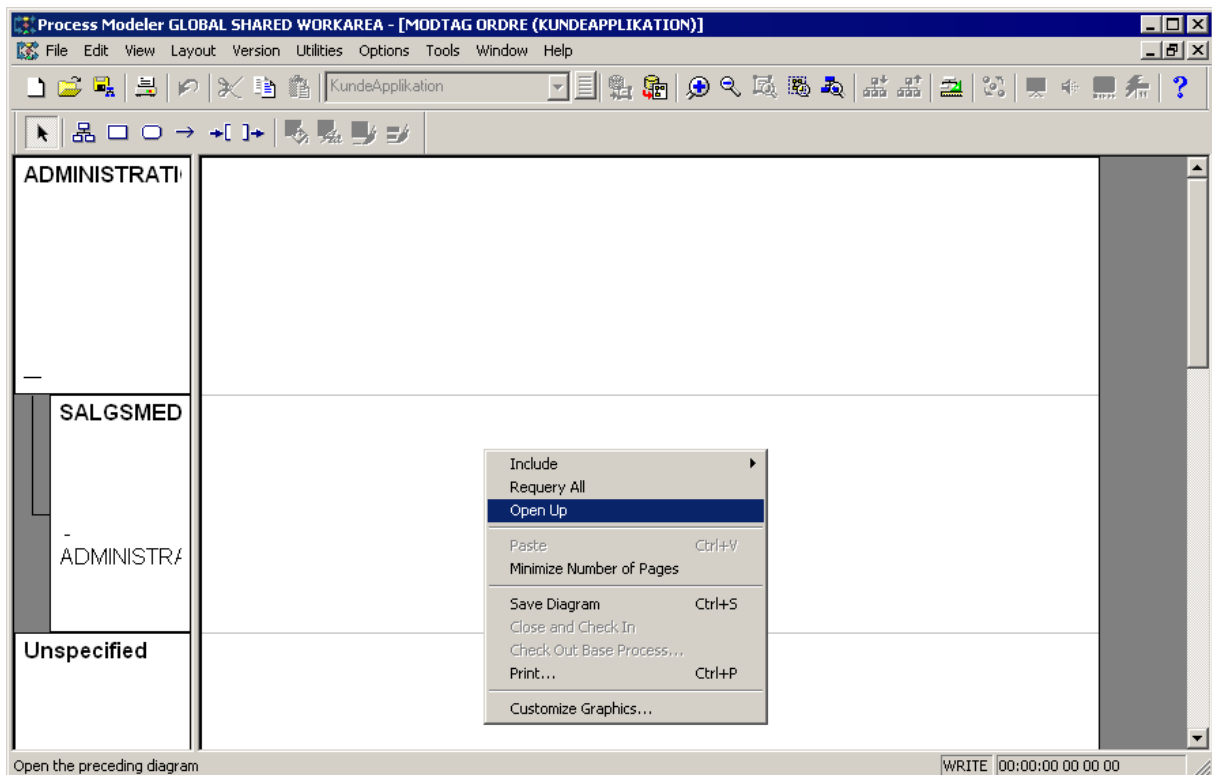


I dette skærmbillede er det muligt at afkrydse hvorvidt et procestrin skal automatiseres eller ej, forstået således om der fra dette proces trin skal hentes, opdateres, indsættes eller slettes data fra ens database. Desuden er det muligt at indtaste detaljerede informationer omkring tids- og ressourceforbrug for hver enkelt procestrin under fanebladene "Main" og "Resources". Endelig kan der tilføjes illustrative ikoner, lyde og video under fanebladet Multimedia.

For ikke at få for uoverskuelige procesdiagrammer anbefales det at opbygge procesdiagrammerne i et "hierarki". Dette gøres vha. af kommandoerne "Open Down" og "Open Up", hvormed man bevæger sig mellem diagrammer. Som det er illustreret i nedenstående figur er der foretaget en "Open Down" ved at højreklikke på processtrinet til højre i den oprindelige model og vælge "Open Down". Nu åbner der et nyt vindue op, hvor man kan arbejde videre indenfor den proces der blev "åbnet op".



I nedenstående skærbillede er der foretaget en "Open Down" på processen "Modtag Ordre". Herfra er det muligt at gå tilbage til det overliggende procesdiagram ved at vælge "Open Up".



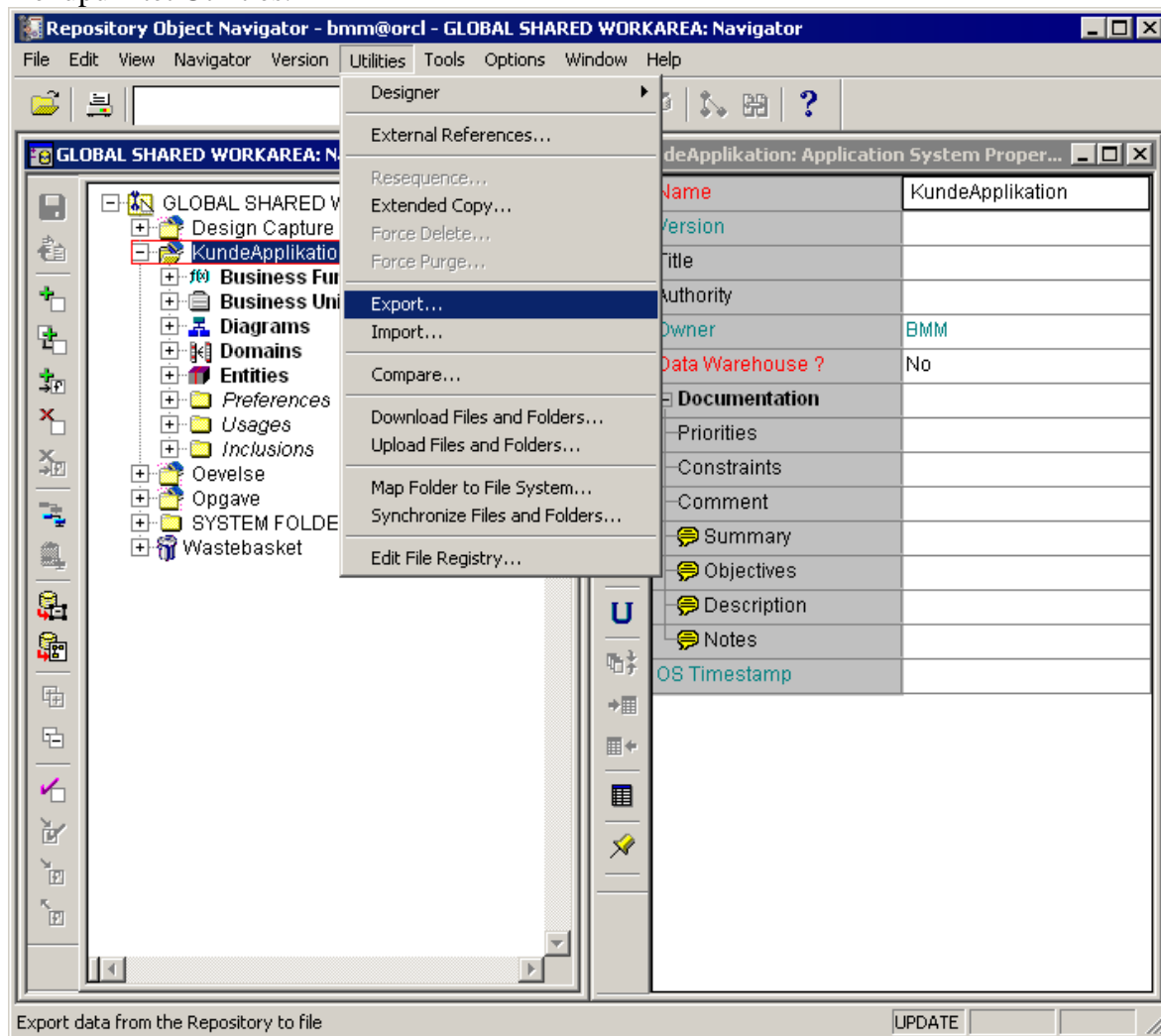
## Backup af applikationen

Når procesdiagrammet er færdigt er det en god ide at tage en backup af ens arbejde ved at lave en ny version af applikationen. Dette foretages i programmet Repository Object Navigator, hvor vi også oprindeligt oprettede applikationen.

En backup kan laves på flere måder. Normalt vil jeg som det nemmeste anbefale at oprette en ny applikation og så foretage en "Extended Copy" (Utilities → Export) af alle elementer fra den originale til den nye, men der er en fejl i den version (9.0.4.4), jeg sidst har undervist ud fra der gør at denne funktionalitet ikke virker korrekt. Almindelig kopiering af elementer fra en applikation til en anden er sjældent en god løsning, da dette ofte medfører afhængigheder mellem applikationerne.

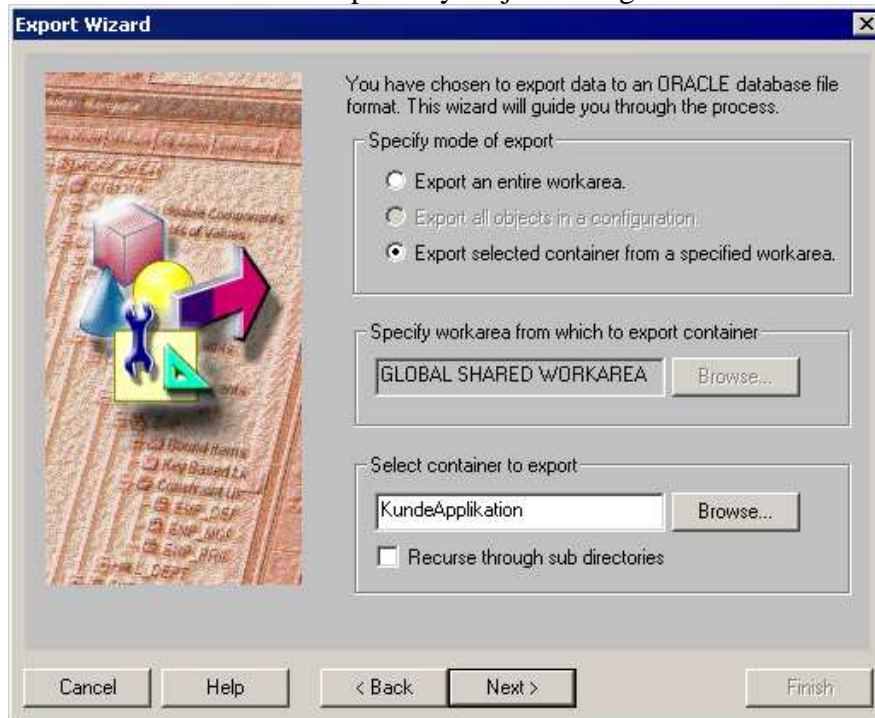
Den måde der derimod virker er at eksportere hele applikationen til en fil, som så på et senere tidspunkt kan importeres igen, hvis man laver fejl. (Importen kan endvidere laves til en ny applikation og behøves altså ikke at hedde det same som den originale application).

Eksport foretages ved at markere den relevante application i træet og vælge Export fra menupunktet Utilities.

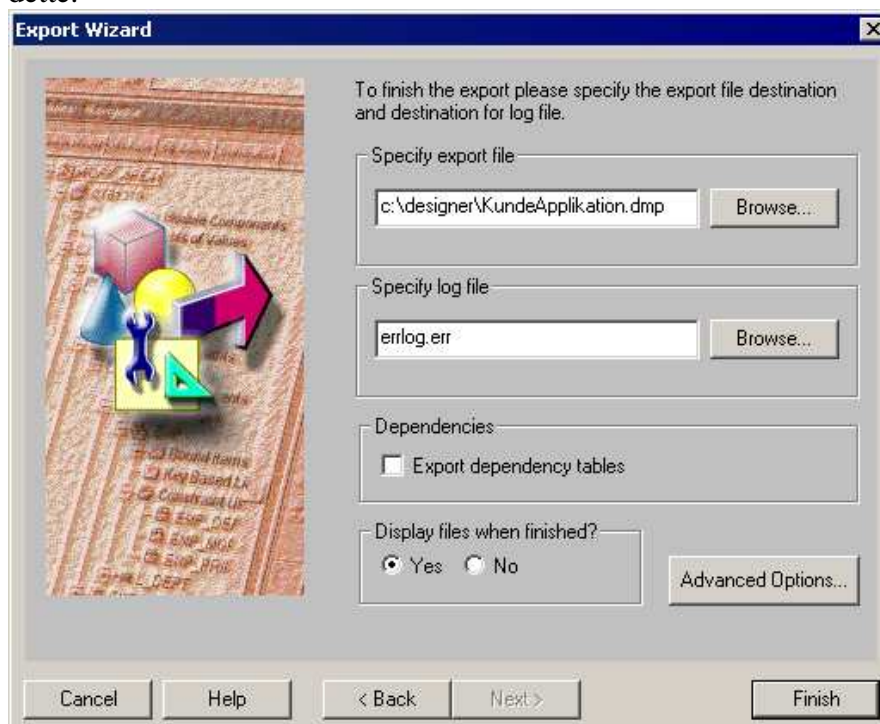




På første side i Export wizarden, vælges at eksportere data til en Oracle database fil (standard). På næste side herunder, vil den relevante applikation allerede være valgt, hvis man havde markeret denne i Repository Object Navigator inden man valgte Export.



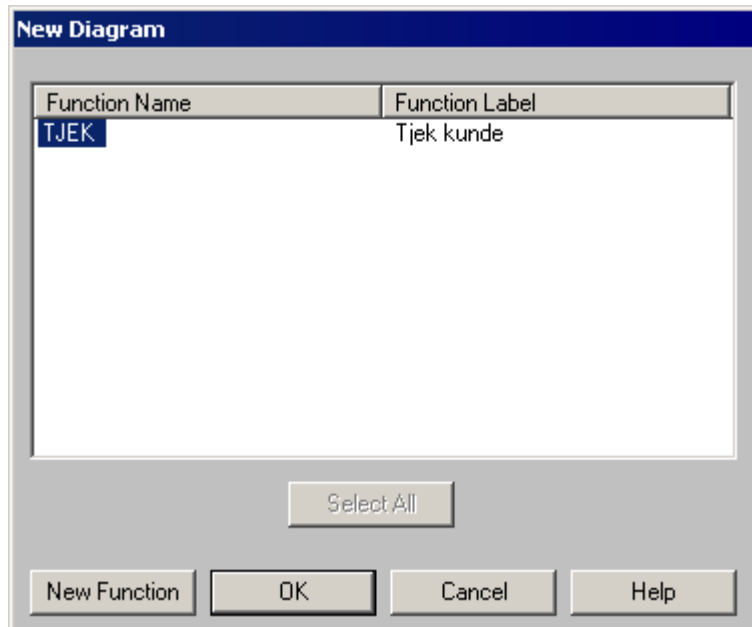
På sidste side skal man tilføje en sti og evt. rette navnet på filen som eksporten gemmes i. Til sidste trykkes på Finish for at udføre eksporten. Hvis man har importeret eller eksporteret tidligere vil der blive vist en advarsel om at nogle tabeller vil blive overskrevet. Sig Ja til dette.



## Function Hierarchy Diagrammer

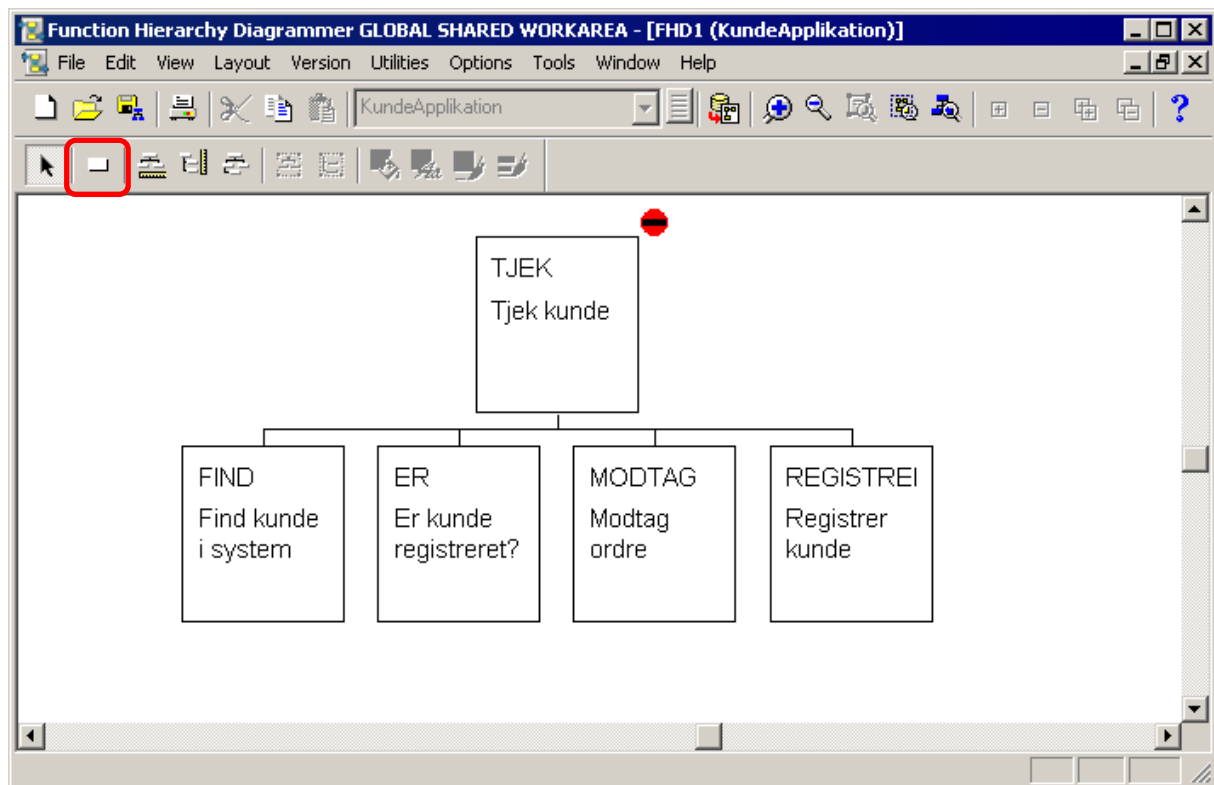
Functionsdiagrammer er en anden synsvinkel på det samme område, som procesdiagrammerne afspejlede. Der ligger dog forskellige informationsværdier i de to modeller, da bl.a. processer (arbejdsgange) ofte vil gå på tværs af funktioner.

Når Function Hierarchy Diagrammer åbnes og der vælges “New Diagram” vil man kunne se øverste niveau i de(n) procesdiagrammer, der blev arbejdet med tidligere. Vælg denne og tryk OK.



Her ses et funktionshierarki på baggrund af det tidligere lavede procesdiagram. Dette er egentlig et proceshierarki og skal ikke ses som værende et egentligt funktionshierarki, men det er muligt at danne et "rigtigt" funktionshierarki på baggrund af proceshierarkiet.

Dette diagram vil jeg gemme inden, man begynder at flytte elementer over i ens applikationshierarki, således at man senere kan genfinde diagrammet. Vær opmærksom på at det kun er selve diagrammet der kan genfindes, mens de faktiske elementer bliver flyttet. Dette vil kunne ses, hvis der foretages en "Requery", når diagrammet åbnes igen efter at have lavet applikationshierarkiet.

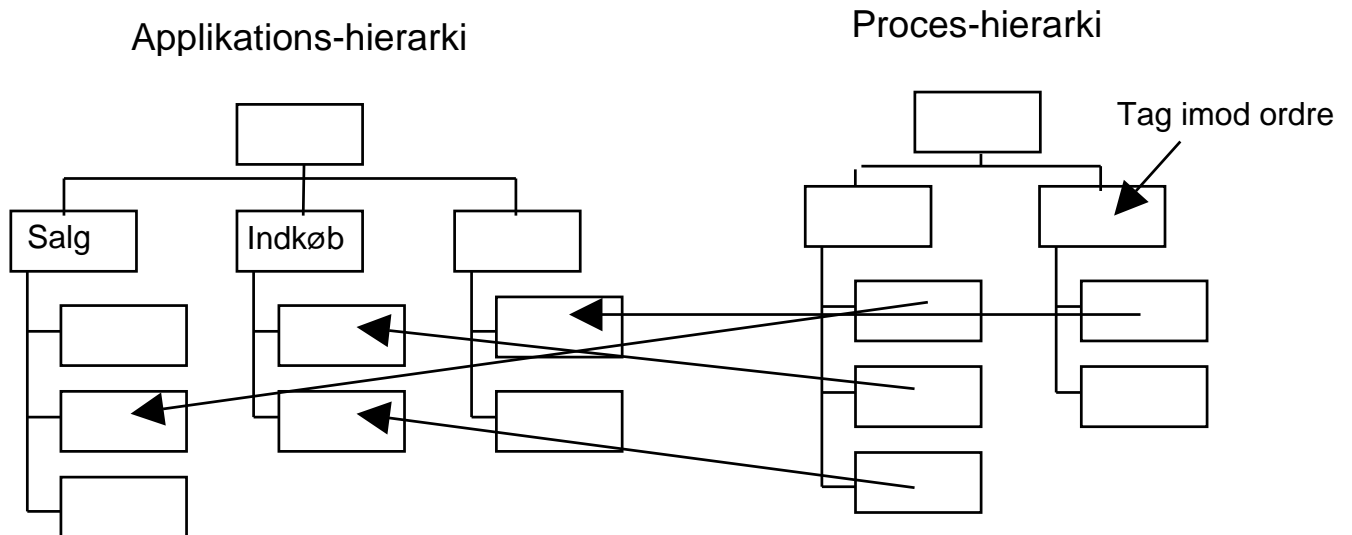


Hvis man selv skal oprette et funktionshierarki fra bunden eller bare tilføje dele til et allerede eksisterende funktionshierarki, så skal man trykke på knappen i den røde ramme og efterfølgende klikke et sted på ens arbejdsområde for at oprette funktioner. Når man vil tilføje en funktion til et allerede eksisterende diagram, så skal man efter at have trykket på funktionsknappen trykke på den funktion, som den nye funktion skal ligge nedenunder. Det er også muligt at flytte på funktioner ved simpelthen at trække dem runde på arbejdsområdet.

Funktionshierarkiet kan opstilles på forskellige måder ved at trykke på de fem knapper til højre for den røde ramme.

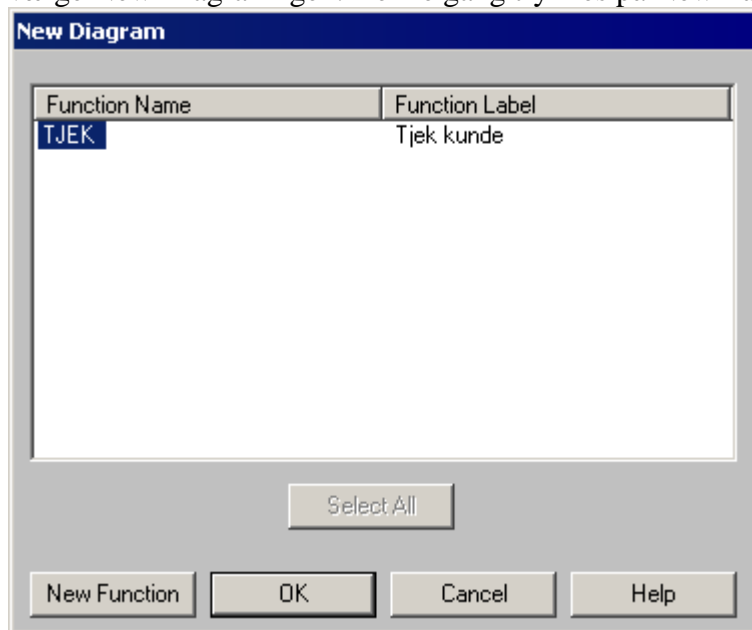
Når man skal lave et funktionshierarki (eller til vores formål et applikationshierarki) vil man først opbygge et overordnet funktionshierarki over de primære funktioner i virksomheden (eks. salg, indkøb, produktion, design), som ikke direkte vil kunne findes i proceshierarkiet. Herefter foretages en re-parenting af funktioner fra proceshierarkiet over til funktionshierarkiet, da processer som sagt før ofte vil foregå på tværs af funktioner i en virksomhed.

Re-parenting foregår ved at man simpelthen trækker funktioner fra proces-hierarkiet over i applikationshierarkiet.

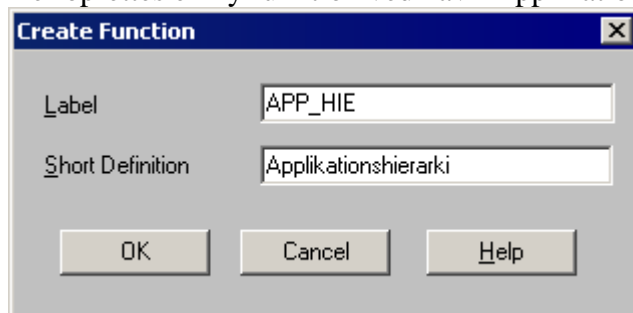


Forskellen på et funktionshierarki og et applikationshierarki er at funktionshierarkiet dækker alle funktioner, der foregår i en virksomhed, mens applikationshierarkiet kun vil indeholde de funktioner, som skal automatiseres i en applikation. Applikationshierarkiet omfatter altså kun de funktioner, som man vil have til at læse, indsætte, opdatere eller slette informationer i ens database.

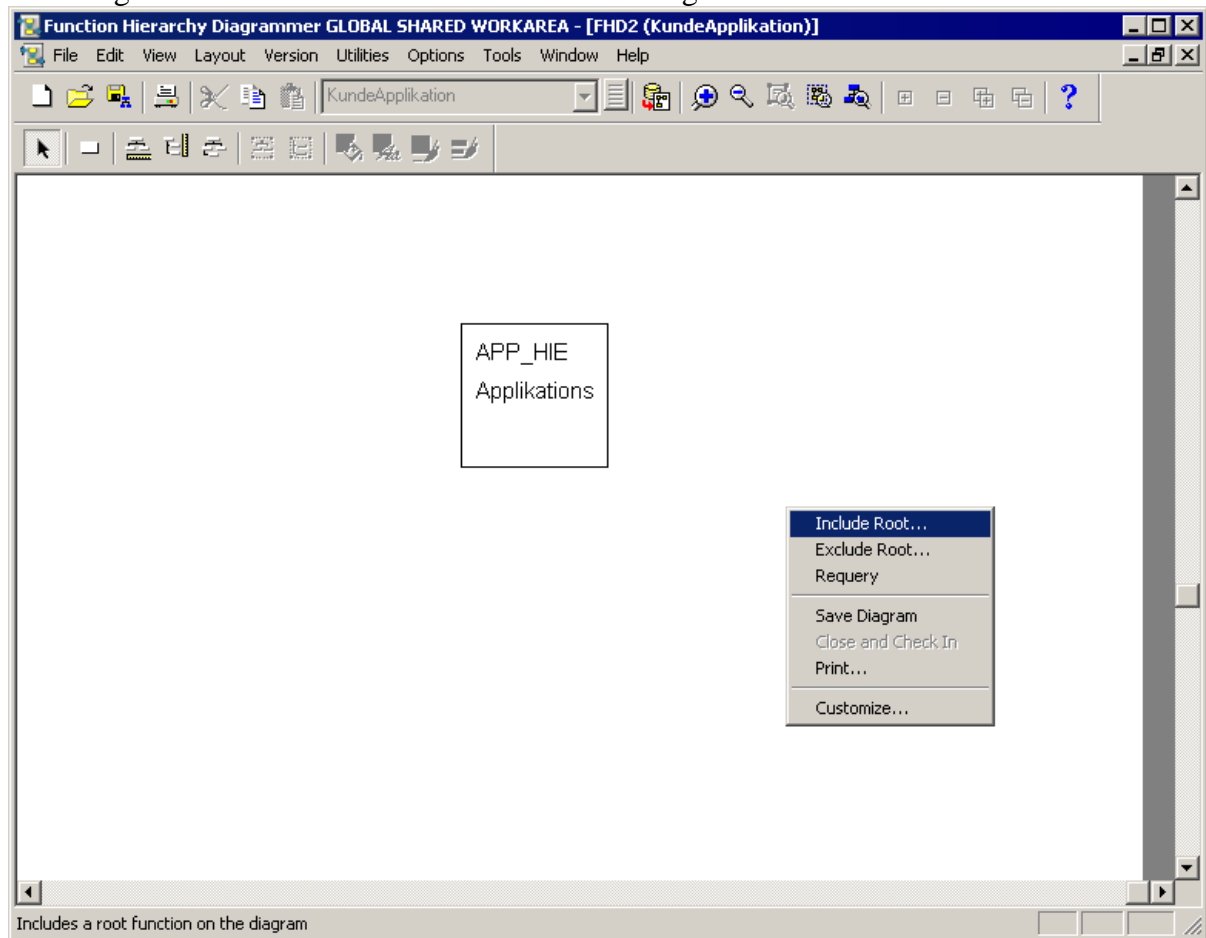
I det praktiske eksempel vil jeg lukke proceshierarkiet efter at have gemt det, og efterfølgende vælge New Diagram igen. Denne gang trykkes på New Function.



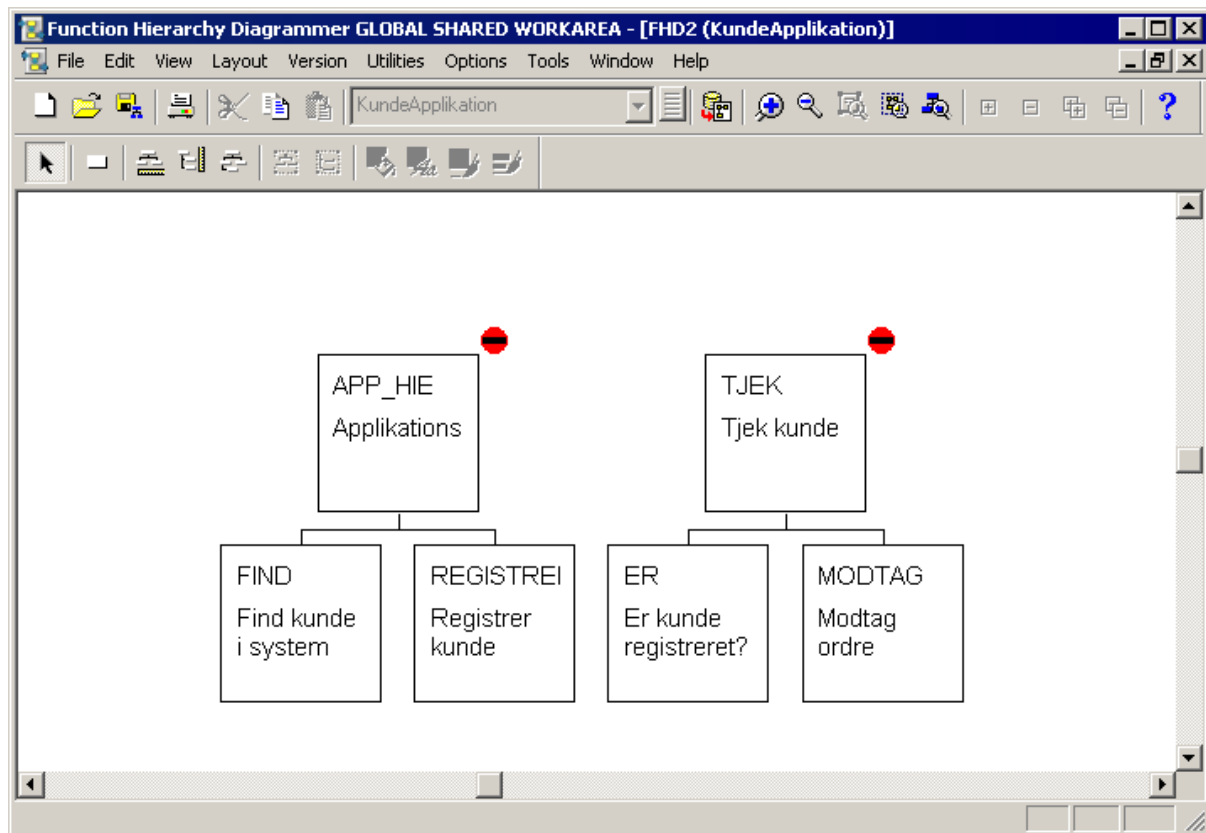
Der oprettes en ny funktion ved navn Applikationshierarki



I det nye funktionsdiagram højreklikkes på den hvide baggrund og der vælges "Include Root". I det vindue, der nu kommer frem, vælges "Root Functionen" for det proceshierarki, der blev vist tidligere. Der vil normalt kun være den ene mulighed.



Efter at have tilføjet proceshierarkiet til samme skema som applikationshierarkiet flyttes (re-parenting) de funktioner, som skal med i applikationen (automatiseres) over i applikationshierarkiet ved at trække de relevante funktioner fra det ene hierarki til det andet.



Til sidst kan man igen højreklikke på den hvide baggrund og denne gang vælge "Exclude Root". Marker "Root Funktionen" for proceshierarkiet og tryk OK. Hermed vil det kun være applikationshierarkiet, der er tilbage i diagrammet.

Inden man kan gå videre til næste punkt med at lave forbindelsen over til E/R-diagrammet er der en sidste indstilling, der skal tjekkes for hver funktion i applikationshierarkiet, da dette har betydning for hvordan Oracle Designer fortolker funktionens opgave.

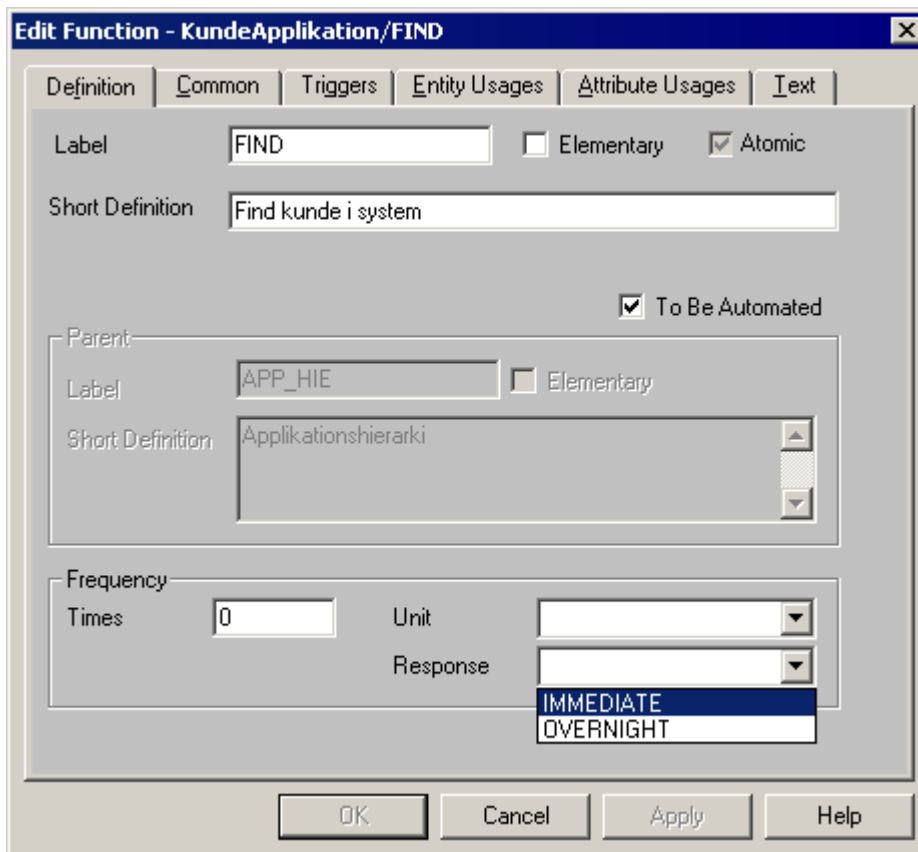
Under properties for hver funktion er der på første faneblad en mulighed for at indstille "Response-tiden", hvilket vil sige, hvornår man vil have svar fra databasen. Mulighederne er "Immediate" og "Overnight".

Hvis funktionens opgave kun er at læse fra databasen (altså ikke noget med at indtaste eller ændre informationer), så vil Designer som standard lave funktionen til en Form, hvis Response er sat til "Immediate", mens funktionen bliver fortolket som en Report, hvis Response er i stedet er "Overnight".

Der er dog stadig mulighed for selv at vælge, om funktionen skal laves til en Form eller en Report på et senere tidspunkt, hvis Designer foretager et forkert valg.

"To Be Automated" skulle gerne være afkrydset for alle funktioner i applikationshierarkiet.

Ikke relevant i forhold til HTMLDDB



Gem diagrammet.

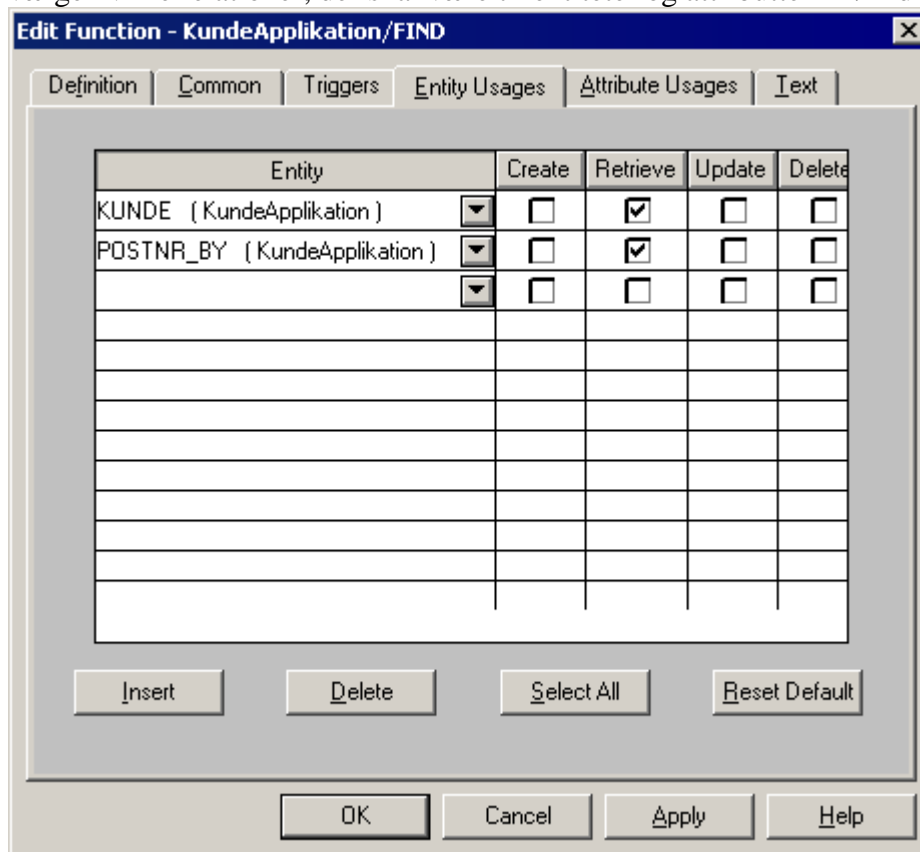
Luk ikke diagrammet endnu.



## Relationer til E/R-diagrammet fra applikationshierarkiet

Efter at have dannet applikationshierarkiet og E/R-diagrammet er det tid til at finde ud af hvilke forbindelser der skal være mellem funktionerne i applikationshierarkiet og entiteter/attributter i E/R-diagrammet.

Ved at gå ind i properties for en af funktionerne i applikationshierarkiet, hvor vi også var inde lige før i forbindelse med "Response-tiden" er der under det 4. og 5. faneblad mulig for at vælge hvilke relationer, der skal være til entiteter og attributter i E/R-diagrammet.

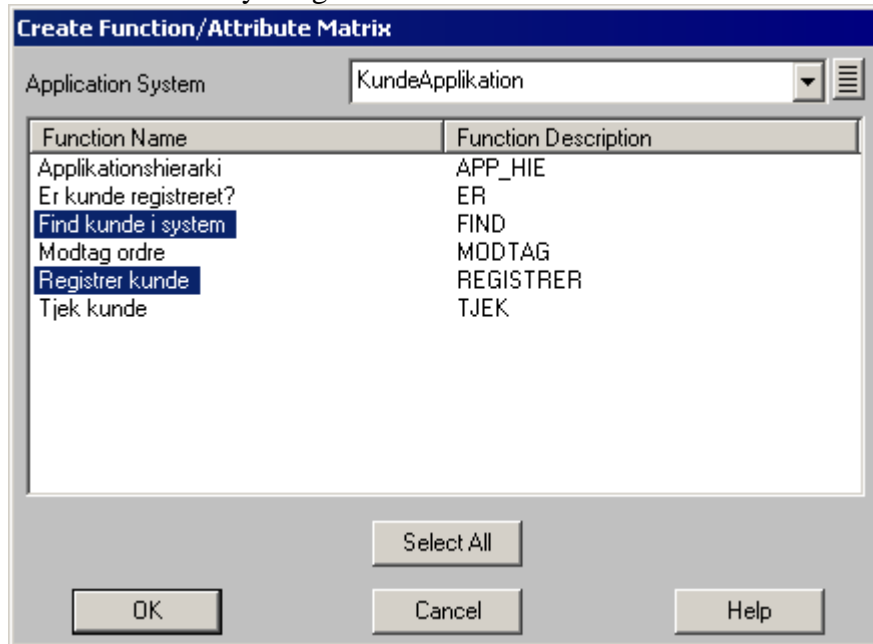


I 4. faneblad laves relationer til entiteterne som helhed. Under kolonnen "Entity" vælges de(n) relevante entitet(er), og derefter sættes der krydser i felterne Create, Retrieve, Update og Delete, som der er behov for.

Det foregår på samme måde for de enkelte attributter i entiteterne under faneblad 5. Umiddelbart anbefaler jeg dog ikke, at relationerne fra funktionerne til attributterne laves her, med mindre at der er behov for forskellig funktionalitet (Create, Retrieve, Update og Delete) for attributter under den samme entitet. Man skal nemlig ind og definere relationerne for samtlige attributter, hvilket kan blive et større arbejde.

## Mindre relevant i forhold til HTML DB

I stedet for at lave relationerne til de enkelte attributter i ovennævnte faneblad, kan man i stedet gå op og vælge menupunktet "Function/Attribute Matrix" under menuen Utilities i Function Hierarchy Diagrammer.



Her vælges de relevante funktioner fra applikationshierarkiet, og der trykkes på OK. Dermed går Designer ind og finder ud af hvilke relationer, man her defineret mellem funktioner og entiteter og overfører disse til også at gælde mellem funktionerne og de enkelte attributter i entiteterne.

Man kan også bare vælge "Select All" i stedet for at skulle udvælge de specifikke funktioner fra applikationshierarkiet. Dette har ingen betydning for funktionaliteten.

Efterfølgende kan man gå ind i ovennævnte faneblad for "Attribute Usages" under properties for funktionerne for at se at relationerne til entiteterne faktisk er blevet lavet. Der kan også laves redigeringer på nuværende tidspunkt, hvis der er behov for dette.

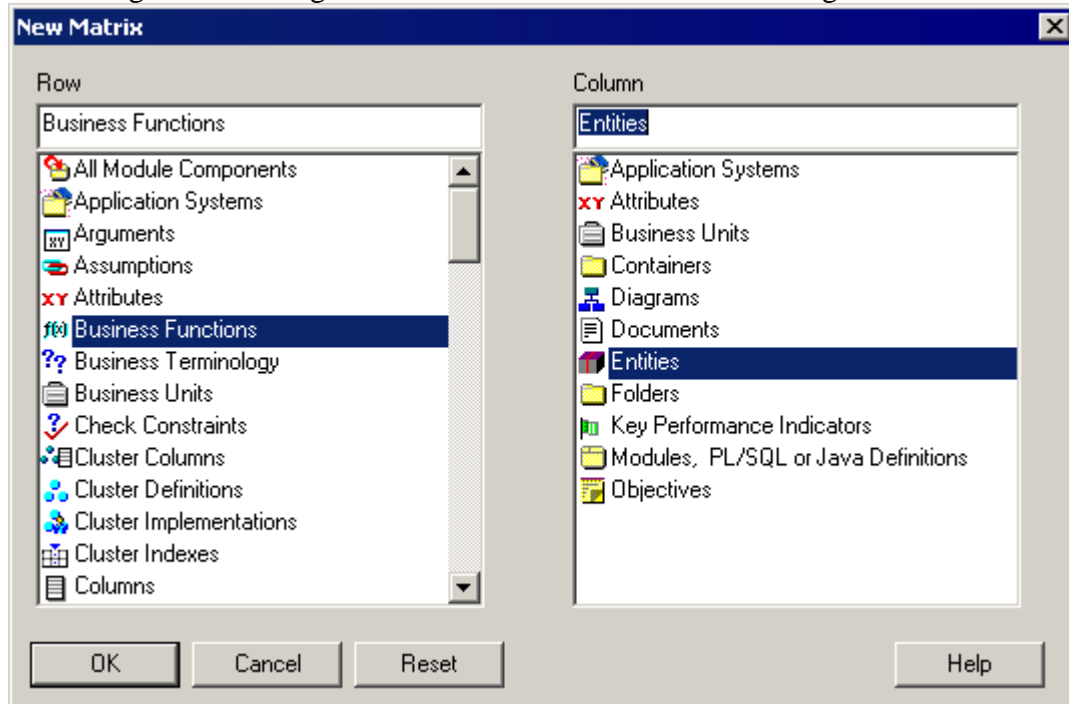
Hvis man foretager ændringer i forholdet mellem funktioner og entiteter på et senere tidspunkt skal "Function/Attribute Matrix" køres igen.

## CRUD-matrix

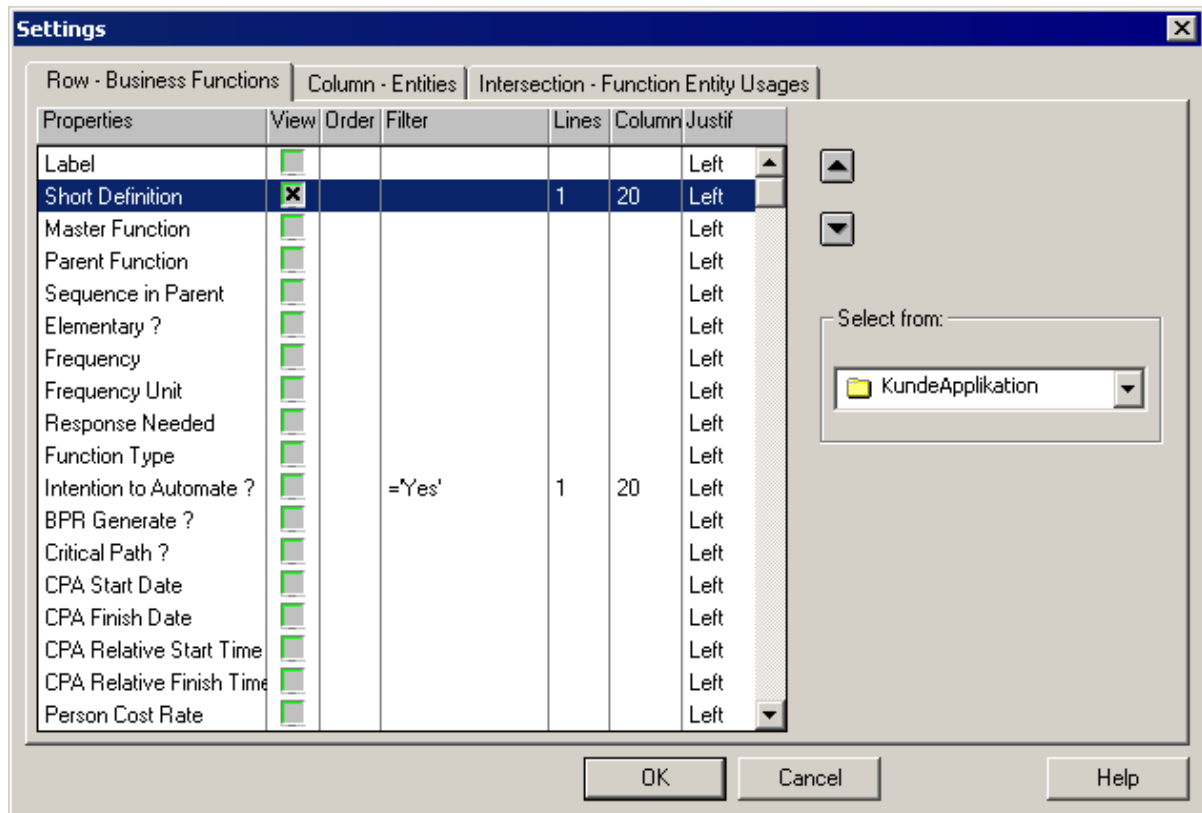
En CRUD-matrix kan også anvendes til at se og lave relationer mellem funktioner og entiteter/attributter.

CRUD-matrixen laves ved at starte programmet Matrix Diagrammer fra Designer Front Panel og indeni programmet trykke på "New Matrix".

I række og kolonne vælges henholdsvis "Business Functions" og "Entities" eller omvendt.



I næste skærmbillede vælges, hvilke informationer, der skal vises i matrixen. For "Business Functions" skal der sættes kryds udfor "Short Definition, og det kan samtidig være nyttigt at tilføje et filter udfor "Intention to Automate?", da vi hermed kun ser de funktioner, som er relevante for applikationen. Der skrives "'Yes'" i filterkolonnen.



På de to andre faneblade skal der under "Entities" sættes kryds udfor "Name", mens der under "Intersection – Function Entity Usages" skal sættes kryds udfor Create, Retrieve Update og Delete.

På de 2 første faneblade skal man endvidere huske at vælge den korrekte Applikation fra listen i højre side, da man ellers vil få alt for mange funktioner og entiteter med i CRUD-matrixen.

Efter at have trykket OK til forrige skærm vises nu CRUD-matrixen. I venstre vindue vises sammenhængene mellem funktionerne og entiteterne illustreret vha. forkortelserne C(rud), R(etrieve), U(pdate) og D(elete). I højre vindue er det muligt at ændre disse indstillinger´.

The screenshot shows the Matrix Diagrammer application interface. The main window displays a CRUD matrix with the following data:

Business Function	KUNDE	POSTNR_BY
Find kunde i	R	R
Registrer kunde	CRUD	R

The right-hand pane, titled 'CRUD:Function Entity Usages', shows configuration for the 'REGISTERER' function:

Function	REGISTERER
Entity	KUNDE
Create ?	Yes
Retrieve ?	Yes
Update ?	Yes
Delete ?	Yes
Archive ?	No
Other ?	No
Comment	
Changed By	BMM
Date Changed	03 November 2004 12:0
Created By	BMM
Date Created	03 November 2004 12:0

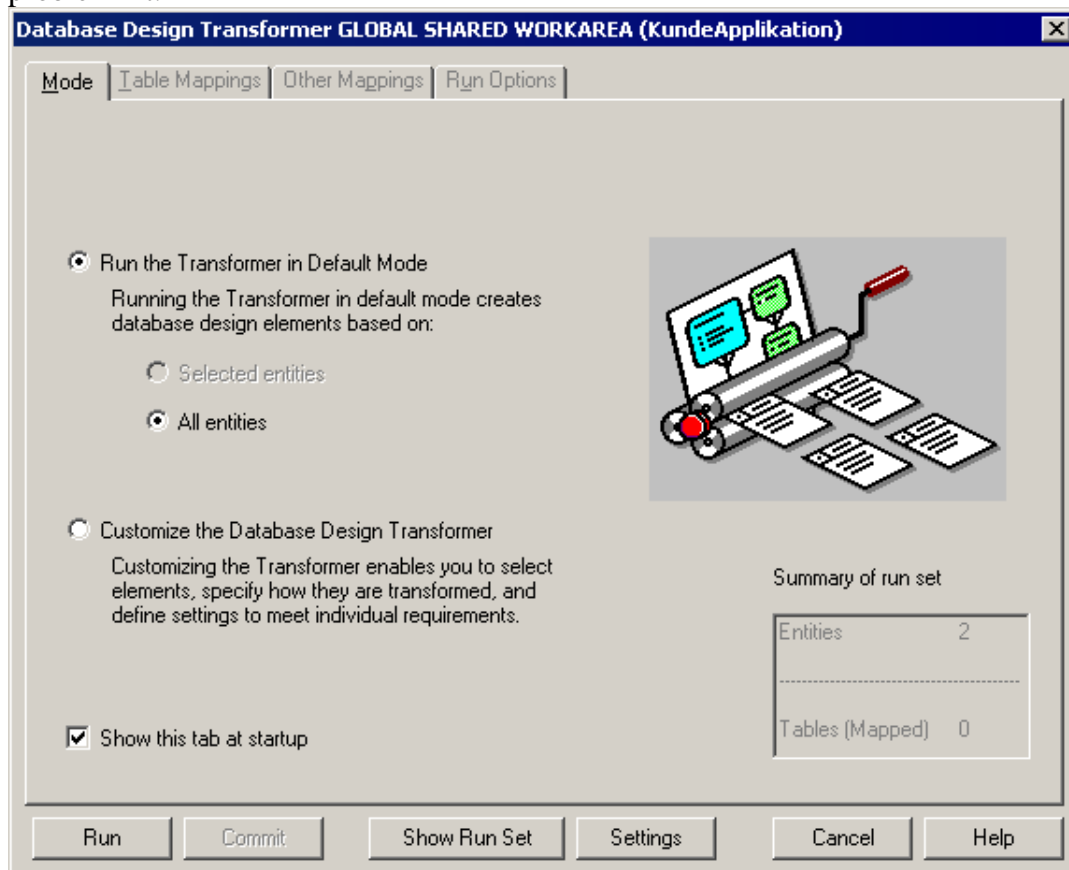
At the bottom of the application window, the status bar indicates: Entities: KUNDE Business Functions: Registrer kunde

Når man er nået til dette punkt er alt det grundlæggende design (de logiske modeller) for henholdsvis database og applikation på plads, og vi kan begynde at generere designs af hvordan den fysiske database og de enkelte moduler i applikationen skal se ud og fungere.

## Database Design Transformer

Med Database Design Transformer, som startes fra Designer Front Panel, dannes et design af den fysiske database på baggrund af E/R-diagrammet.

Når denne startes skal man først vælge den korrekte applikation. Under normale omstændigheder kan man bare køre Database Design Transformer i Default Mode. Man kan vælge Customized mode, hvis man vil ændre på, hvordan super-/subentiteter bliver konverteret til tabeller, eller hvis man vil opdatere tabeller i en eksisterende databasedesign. Nederst i højre side, kan man se hvor mange entiteter man har, og hvor mange tabeller, der er skabt på baggrund af disse. Før kørslen vil der stå 0 udfor tabeller, mens der bagefter gerne skal stå det samme tal udfor entiteter og tabeller, hvis genereringen ellers er forløbet problemfrit.



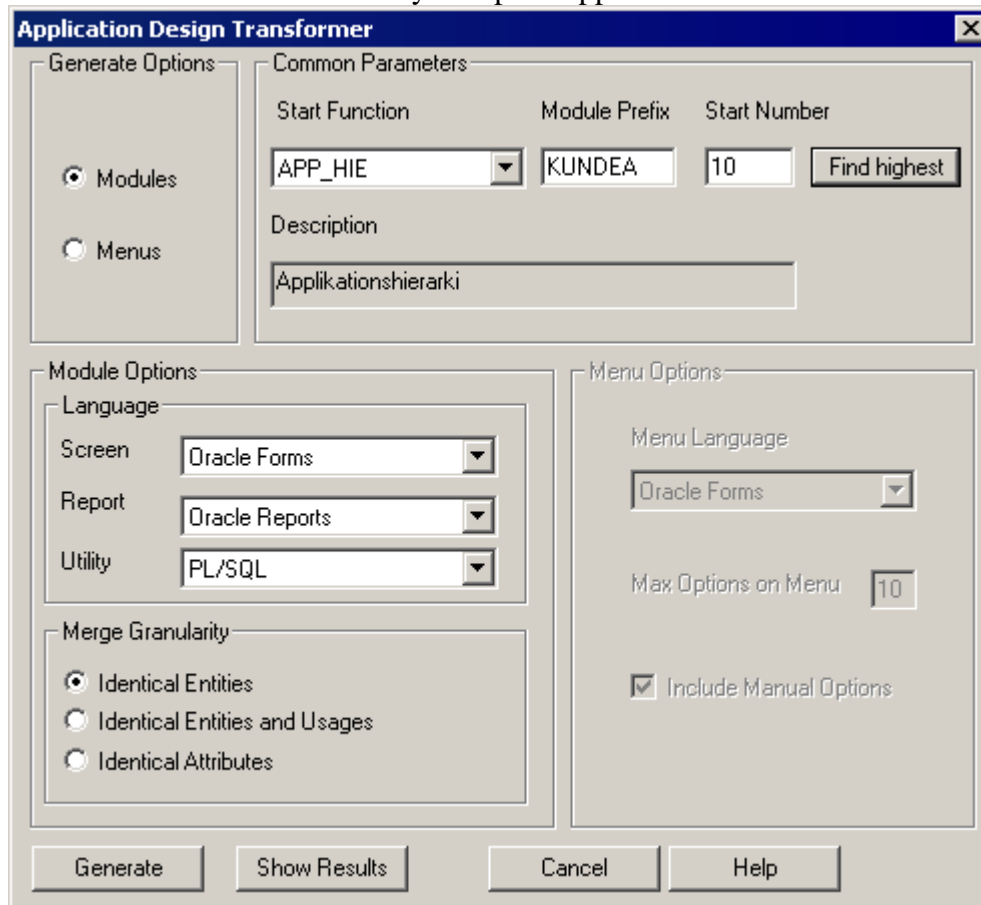
Efter at trykke på run vil der komme et kørselvindue, som man bare skal vente på blive færdigt. Der skal stå "Waiting for Close to be pressed" nederst i vinduet. Hvis der bliver nævnt nogle fejl undervejs, må man prøve at undersøge hvad der er i vejen. Ellers skal man bare trykke på Close og efterfølgende tjekke at antallet af entities og mappede tabeller er det samme, inden man lukker for Database Design Transformer.

Der kan opstå problemer med Database Design Transformer, hvis man har anvendt nogle af Oracles reservede ord i navngivningen af eksempelvis tabeller. Hvis man allerede har lavet et databasedesign/servermodel ud fra et E/R-diagram, og efterfølgende har lavet ændringer i E/R-diagrammet er det ofte mest hensigtsmæssigt at slette resultatet (den gamle servermodel) fra Design Editor, før Database Design Transformer køres igen.

## Application Design Transformer (IKKE RELEVANT i forhold til HTML DB)

Application Design Transformer, som også startes fra Designer Front Panel, laver et design over de faktiske moduler (Forms og Reports - applikationen) på grundlag af ens applikationshierarki og de relationer der er angivet mellem funktioner og entiteter/attributter.

Tilsvarende Database Design Transformer skal der normalt ikke ændres i indstillingerne i dette program. Det eneste man lige skal sikre sig er at "Start Function" bliver sat til den øverste funktion i ens applikationshierarki. Database Design Transformer gør nemlig det, at den laver moduler på baggrund af alle de funktioner, som er placeret nedenunder denne funktion i hierarkiet. Til sidst trykkes på knappen "Generate".



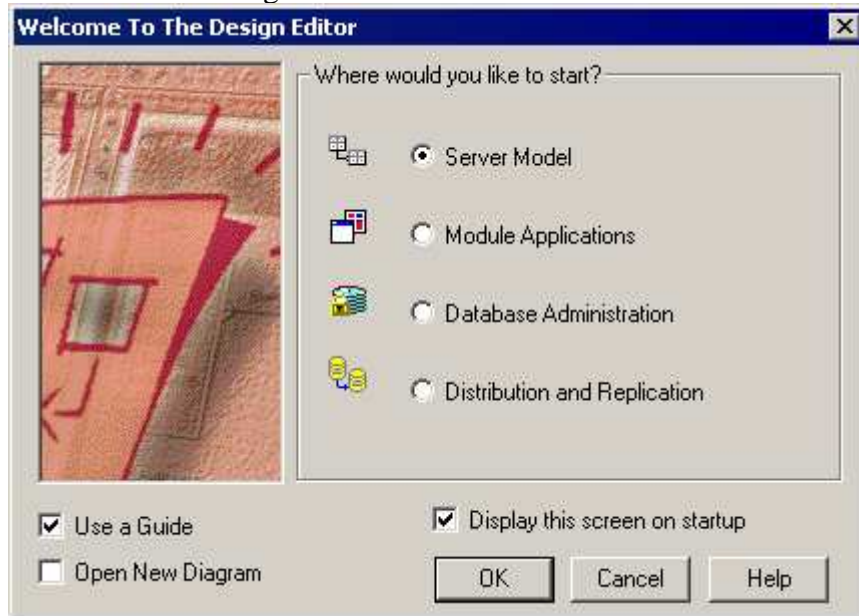
Efter at have fuldført generationen af moduler kan det være nyttigt at trykke på knappen "Show Results", da man her kan få at vide hvilke moduler, der egentlig er blevet lavet og hvorfor. Ellers lukker man efterfølgende bare for programmet.

Det kan nogle gange være hensigtsmæssigt at ændre på "Merge Granularity", hvor det defineres, hvornår Designer skal samle flere funktioner i et modul. Standardindstillingen "Identical Entities" betyder at funktioner samles, hvis de har relationer til de samme entiteter. "Identical Attributes" betyder det samme, men bare på attributniveau i stedet for entitetsniveau. Hvis "Identical Entities and Usages" er valgt skal funktionerne ikke bare vedrører de same entiteter, men der skal være defineret ens sammenhænge (CRUD) mellem funktioner og entiteter, før funktioner samles i et modul.

### Design Editor (Server Model)

Inden den faktiske database og de reelle moduler dannes er det muligt at foretage ændringer i designet til disse dele. Dette foregår for begges vedkommende i programmet "Design Editor", der startes fra Designer Front Panel. Jeg vil her først behandle designet til den fysiske database – også kaldt Server Model.

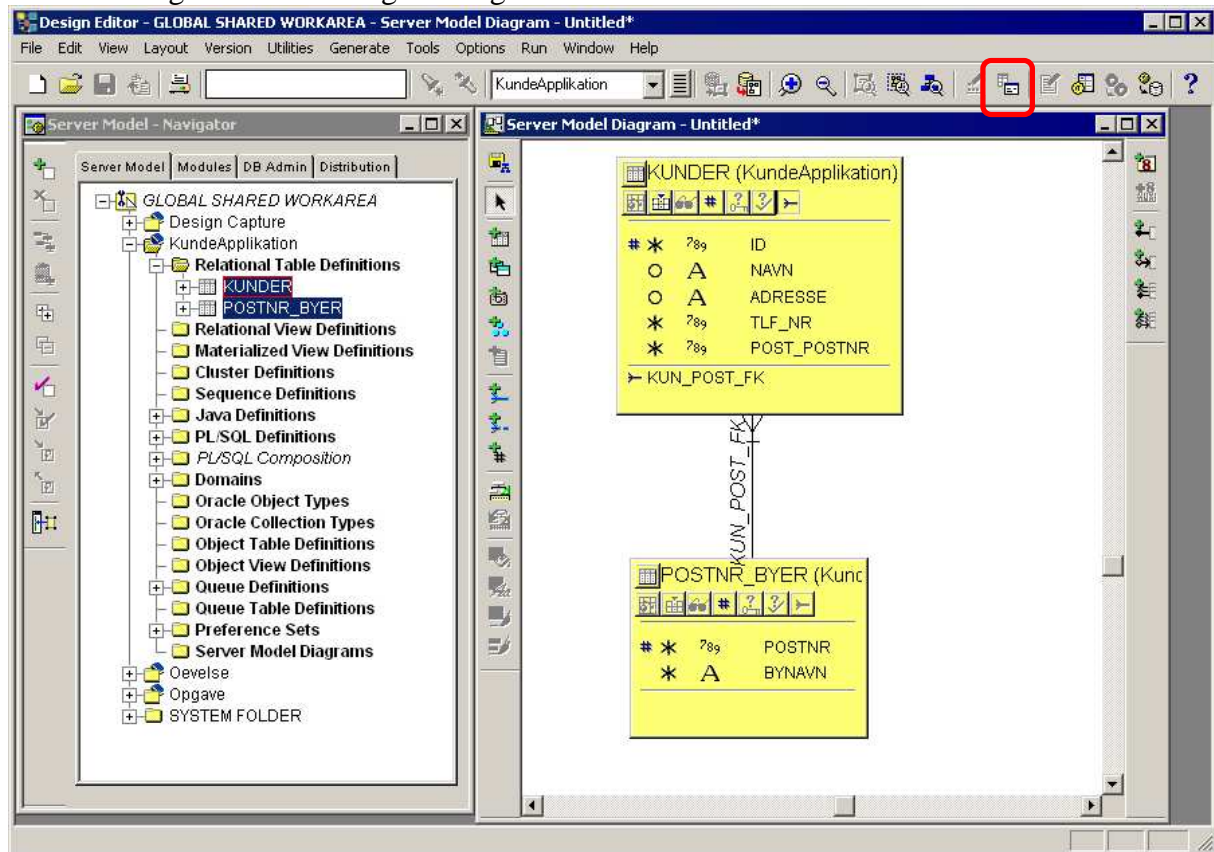
Når Design Editor først startes kan man vælge om man vil arbejde med Server Model eller Module Applications (de to relevante i denne sammenhæng), men det kan også lade sig gøre at skifte mellem alle dele inde i selve programmet. Der er desuden mulighed for at anvende en guide (Sæt kryds udfor "Use a Guide"), som jeg dog ikke vil behandle her. Tryk nu OK, når Server Model er valgt.





Inde i Design Editor vil der til at starte med være en oversigt over ens Server Model i venstre side i Navigatoren. Ved at vælge fanebladet "Modules" her gå man over til at arbejde med modulerne til applikationen.

Hvis man trykker på plusset udfor "Relational Table Definitions" kan man se navnene på de tabeller, som er lavet på baggrund af E/R-diagrammet. Ved at markere disse og trække dem ind på den grå baggrund (Man kan også højreklikke og vælge "Show On New Diagram") laves der en grafisk afbildning af designet til databasen.



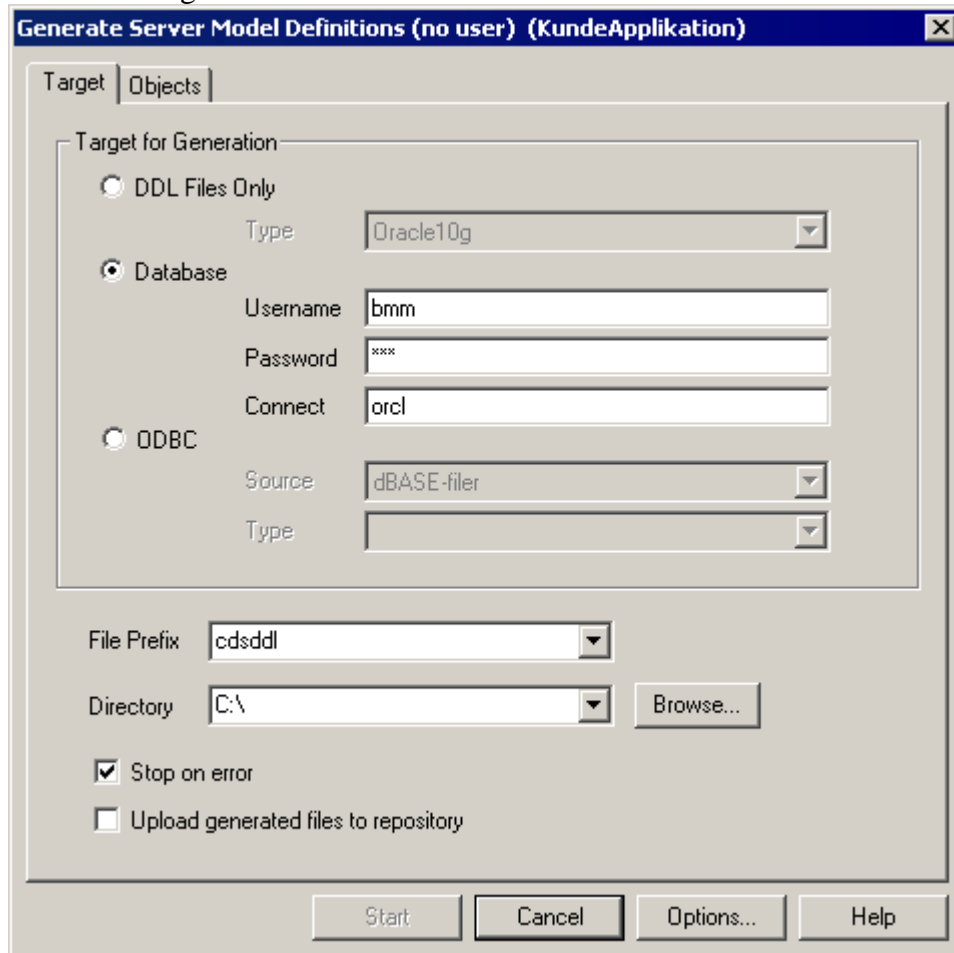
Det skal her nævnes, egenskaberne/properties kan vises på 2 forskellige måder alle elementer i Server Model og Modules. Dette drejer sig om henholdsvis en Property Palette og Property Dialogs. Man skifter mellem disse to måder at vise properties på ved at trykke på knappen i den røde ramme.

Der kan nu laves ændringer til dette design inden selve databasen genereres. Man kan f.eks. lave nye tabeller, attributter, relationer. Hvis man vil lave sådanne ændringer, vil jeg dog anbefale at gå tilbage til E/R-diagrammet og lave ændringerne heri, og derefter generere en ny Server Model.

Det kan også lade sig gøre at tilføje Views til Server Modellen, som dog kræver lidt mere manuelt arbejde, hvis der skal laves views på tværs af tabeller, da man selv skal tilføje joins. Disse funktioner er hovedsagelig tilgængelig i knapperne omkring Server Model diagrammet. Når selve databasen skal bygges, går man op i menuen "Generate" og vælger "Generate Database from Server Model"

Normalt vil man vælge at generere tabellerne direkte nede i databasen ved at vælge "Database", men det kan eksempelvis også lade sig gøre kun at lave en række tekst-filer, hvor SQL-kommandoerne til at generere database lægger i (DDL Files Only).

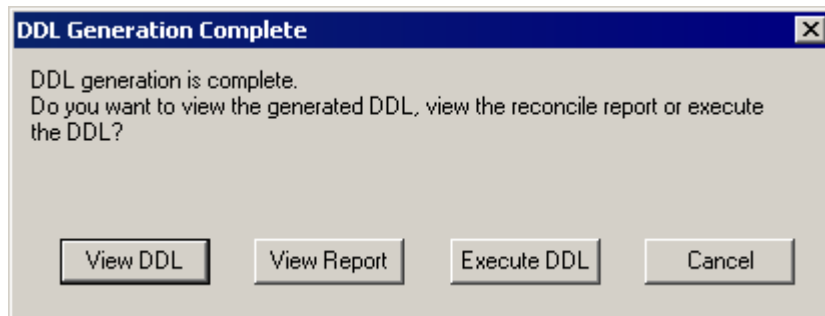
Hvis man vil arbejde direkte ned mod databasen, skrives ens brugernavn, password og connect-streng i boksene udfor "Database".



Under det andet faneblad "Objects" skal man sikre sig at alle de elementer, man vil have genereret er flyttet over i højre side under "Generate"

Til sidst trykkes på knappen "Start"

Efter lidt tid viser dette vindue sig. Man kan vælge at se de SQL-kommandoer, som vil blive sendt ned mod databasen ved at trykke på "View DDL", ellers trykkes til sidst på "Execute DDL" for at lave ens database.



Der popper nu et Message-vindue frem, hvor der skrives om der sker fejl i forløbet, som man så evt. må undersøge.

Det kan anbefales også at gå ud i en SQL-prompt for selv at se at tabellerne bliver lavet korrekt.

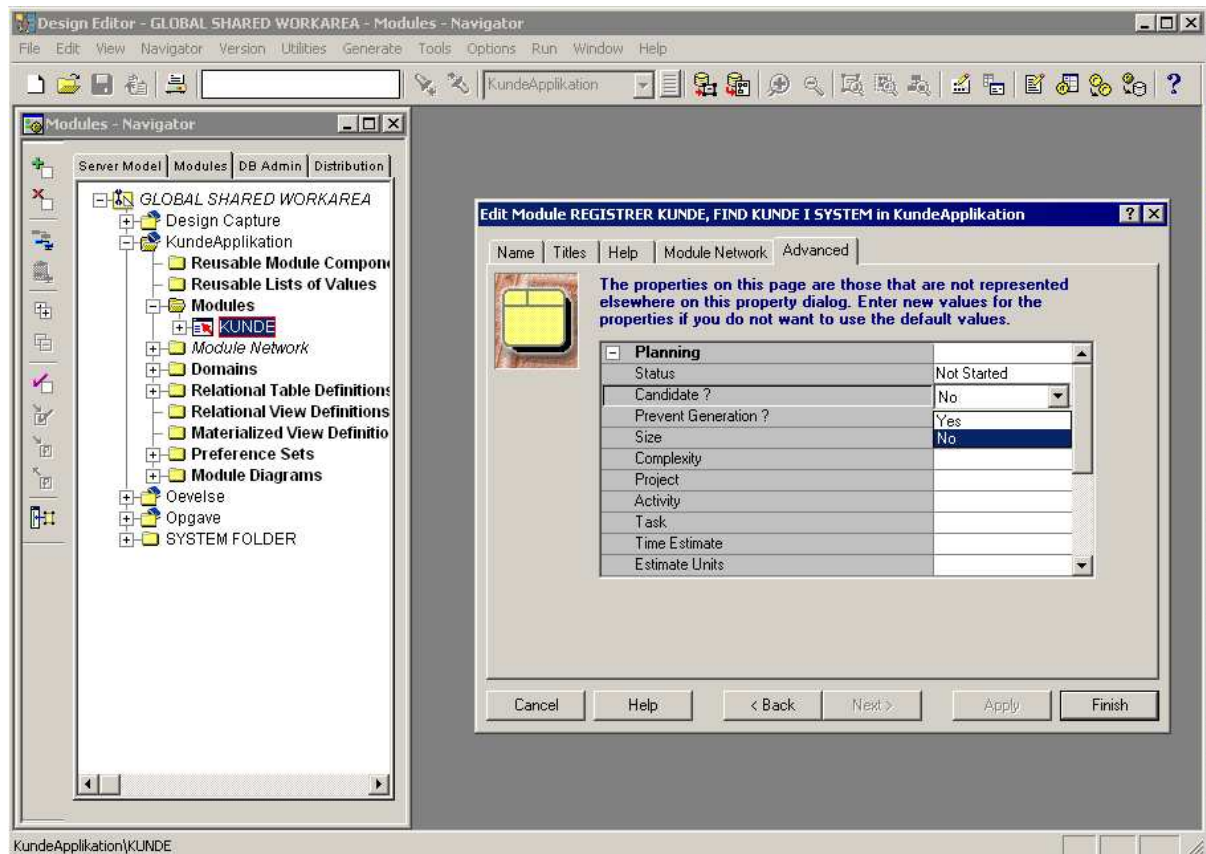
## Design Editor (Modules)

(Resten af dokumentet er IKKE RELEVANT i forhold til HTML DB)

Når man skal til at arbejde med designet til modulerne (applikationen), kan man vælge "Module Applications", når Design Editor startes, eller også skifter man simpelthen over til fanebladet "Modules" i Navigatoren, hvis man allerede har Design Editor kørende i forbindelse med arbejdet på Server Model.

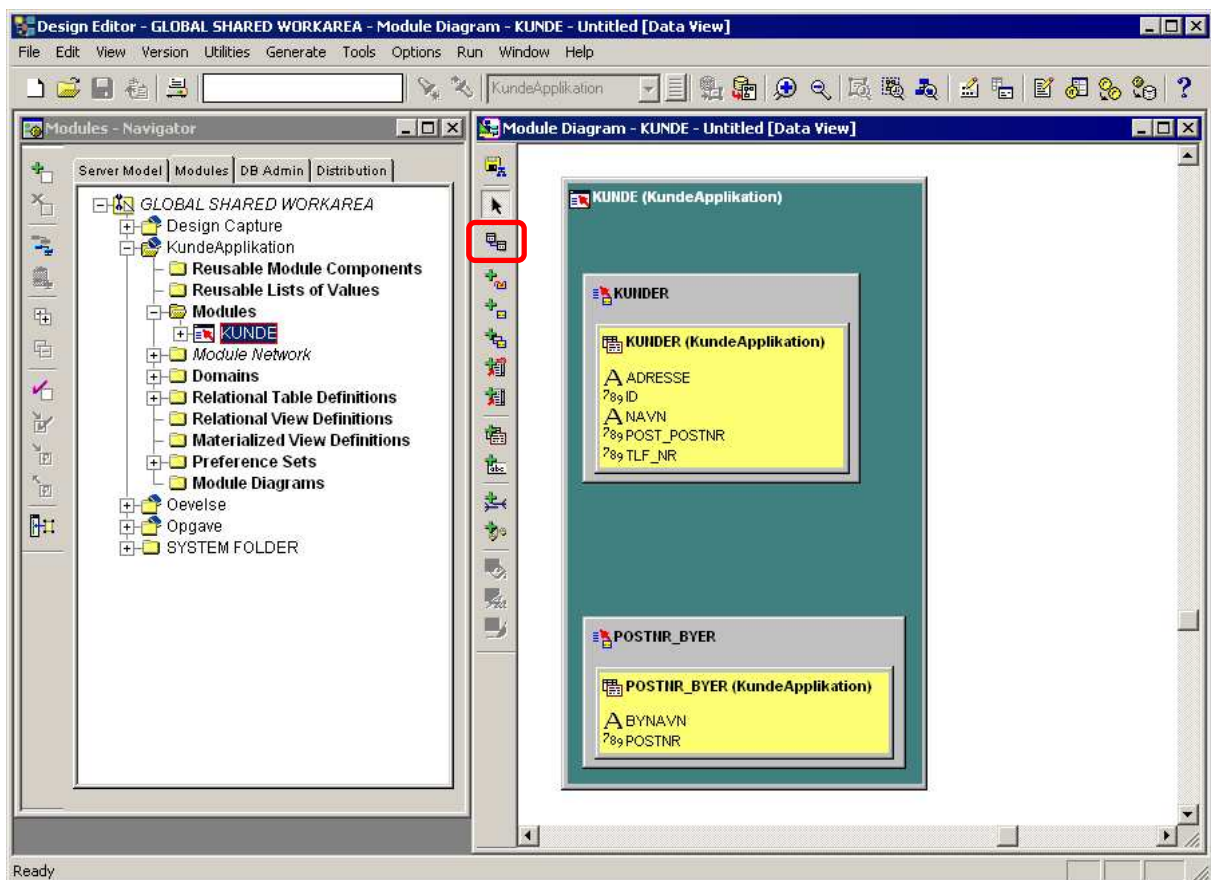
Udgangspunktet er her igen visningen af moduler i Navigatoren. Ved at trykke på plussed udfor Modules ses de forskellige moduler, som Application Design Transformer har lavet. Inden vi begynder at arbejde med indholdet i hvert modul, skal vi lige ændre et par ting i properties. Dobbeltklik på modulet.

Under fanebladet Advanced skal "Candidate?" ændres fra Yes til No, hvilket vil medføre, at der kommer et lille ikon udfor modulet, som viser om det bliver til en form, report eller noget helt tredje. På første faneblad kan navnet på modulet ændres i feltet "Short Name", mens man har mulighed for at bestemme om modulet skal blive til en Forms eller Reports i feltet "Language".



Efter at have fjernet kandidat-betegnelsen fra modulerne er det tid til at arbejde med strukturen og visningen af informationer i modulerne. Der kan laves *mange* forskellige ting i Design Editor omkring struktur og visning af informationer, hvor det kun er nogle få elementer, jeg her vil trække frem.

Først vil man markere et af modulerne og trække det ind på den grå baggrund for at få en grafisk visning af modulets indhold. Den grønne kasse illustrere her et modul, de grå kasser er modulkomponenter, mens de gule (og blå) kasser viser de tabeller, som modulkomponenterne bruger fra databasen. Ved at trykke på knappen i den røde ramme skifter man over til ”Display View”, hvor man redigere udseendet af modulet, mens endnu et tryk på denne knap vil føre tilbage her til ”Data View”, hvor det er strukturen af data, der redigeres.

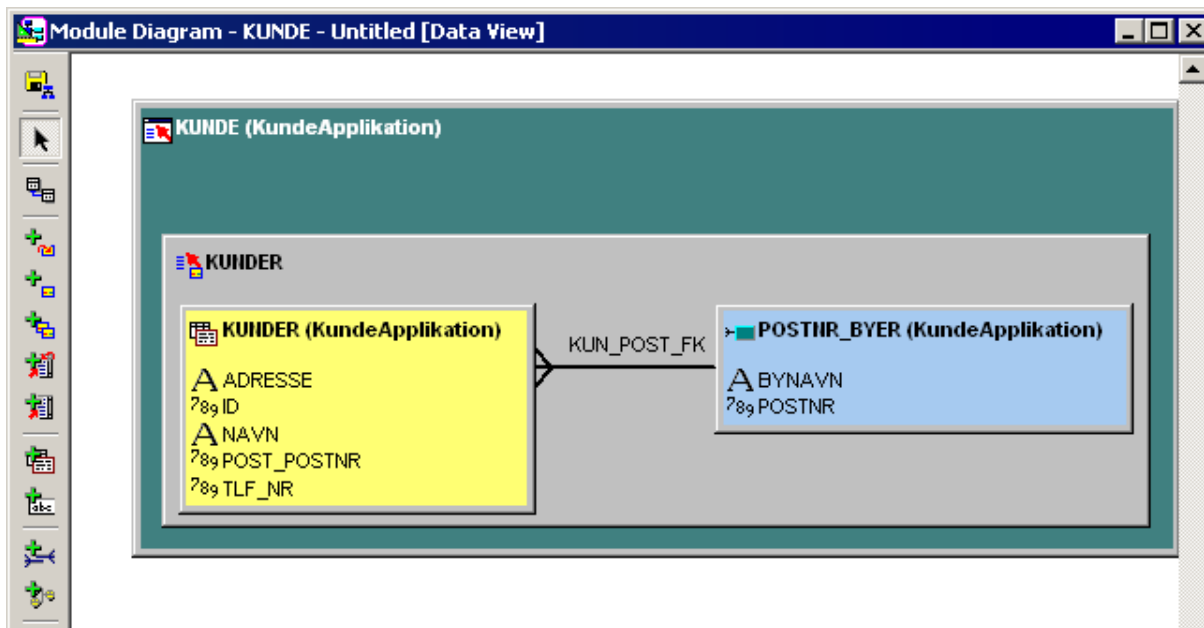


Man kan allerede nu generere et virkende modul, men jeg vil først vise nogle af de ting man kan lave inden vi genererer modulet.

Som det er vist i skærbilledet ovenfor er de to tabeller i modulet uafhængige af hinanden, og man vil kunne læse og skrive uafhængigt til de to tabeller, selv om de er samlet i en Form.

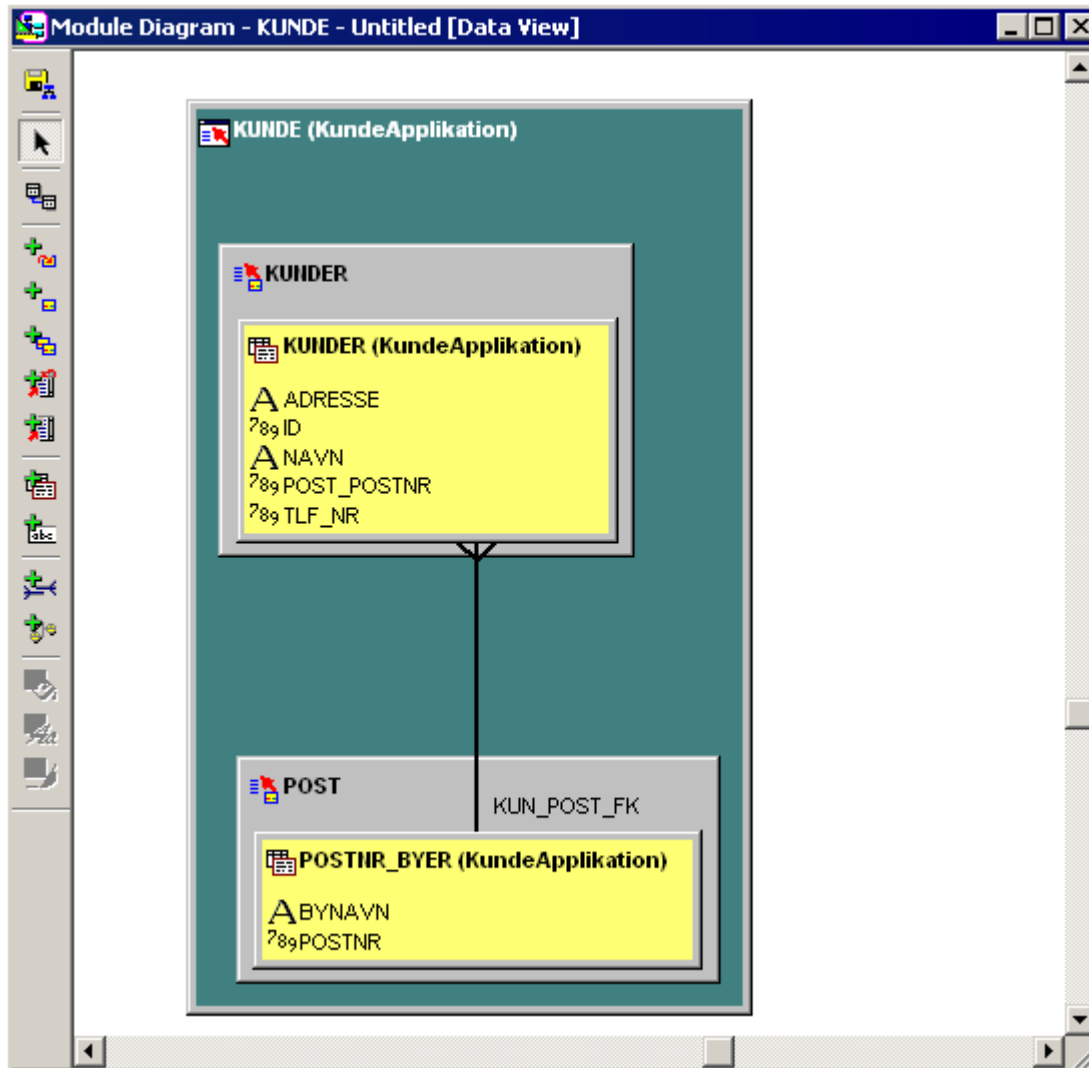
Når der er fremmed-/primærnøgle-relation mellem to tabeller, er det muligt at lave den ene tabel om til en Lookup Table. Hvilket vil sige, at i eksemplet her nedenunder er Postnumre en Lookup Table til Kunder (som er en Base Table). Dermed kan der kun læses fra Postnummer-tabellen, mens der kan læses og skrives i Kunde-tabellen, og når der slås et postnummer op i kundetabellen, vil dette postnummer og tilhørende bydel automatisk blive fundet i postnummer-tabellen.

Man laver en Lookup Table ved at trække den gule kasse, der illustrere den table man vil lave om til en Lookup Table, op på siden af den anden gule kasse (tabel), den skal knyttes til. Der skal være vist en stiptet streg til højre for den gule kasse, som Lookup Tabellen knyttes på, inden man slipper musetasten.



Hvis man fortryder Lookup Tabellen, som blev lavet ovenfor, så kan man trække den nu blå kasse udenfor igen, så den bliver en selvstændig tabel. I første omgang vil der dog automatisk blive lavet en ny sammenhæng mellem tabellerne, som det er vist her nedenunder. Hvis man vil fjerne denne relation, højreklikker man relationen og vælger "Delete from Repository"

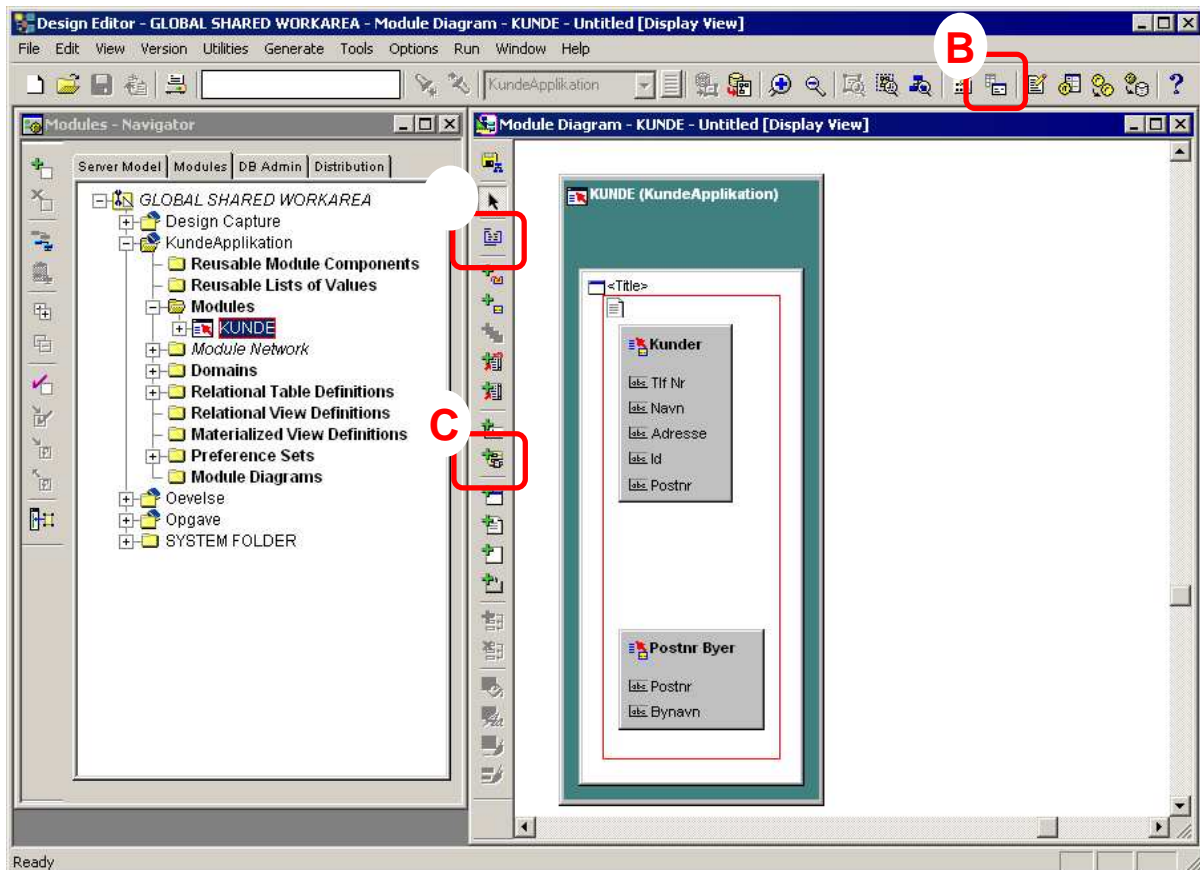
Denne type relation, som er illustreret her kan være nyttig, når man vil lave en sammenhæng som mellem ordre og ordrelinier, hvor der skal indtastes på begge dele. Eksemplet med postnumre og kunder giver slet ikke mening, som det er vist her, men hvis postnumre i stedet var ordre og kunder var ordrelinier, så vil den kunne være hensigtsmæssig.



Når man er færdig med, hvordan strukturen mellem tabeller og modulets andre elementer skal være, mangler man kun at tilpasse selve udseendet på modulet. Først skal man som nævnt tidligere skifte over til "Display View", hvilket sker ved at trykke på knappen i ramme A.

I Display Viewet illustrerer den grønne kasse igen hele modulet, den hvide kasse er et vindue, den røde ramme er en canvas og de grå kasser er modulkomponenter. De samme elementer kan genfindes i Form Builder.

Display Viewet i dette skærbillede er standard når der ikke er rørt ved strukturen mellem tabellerne i Data View – dvs. når tabellerne er uafhængige af hinanden.

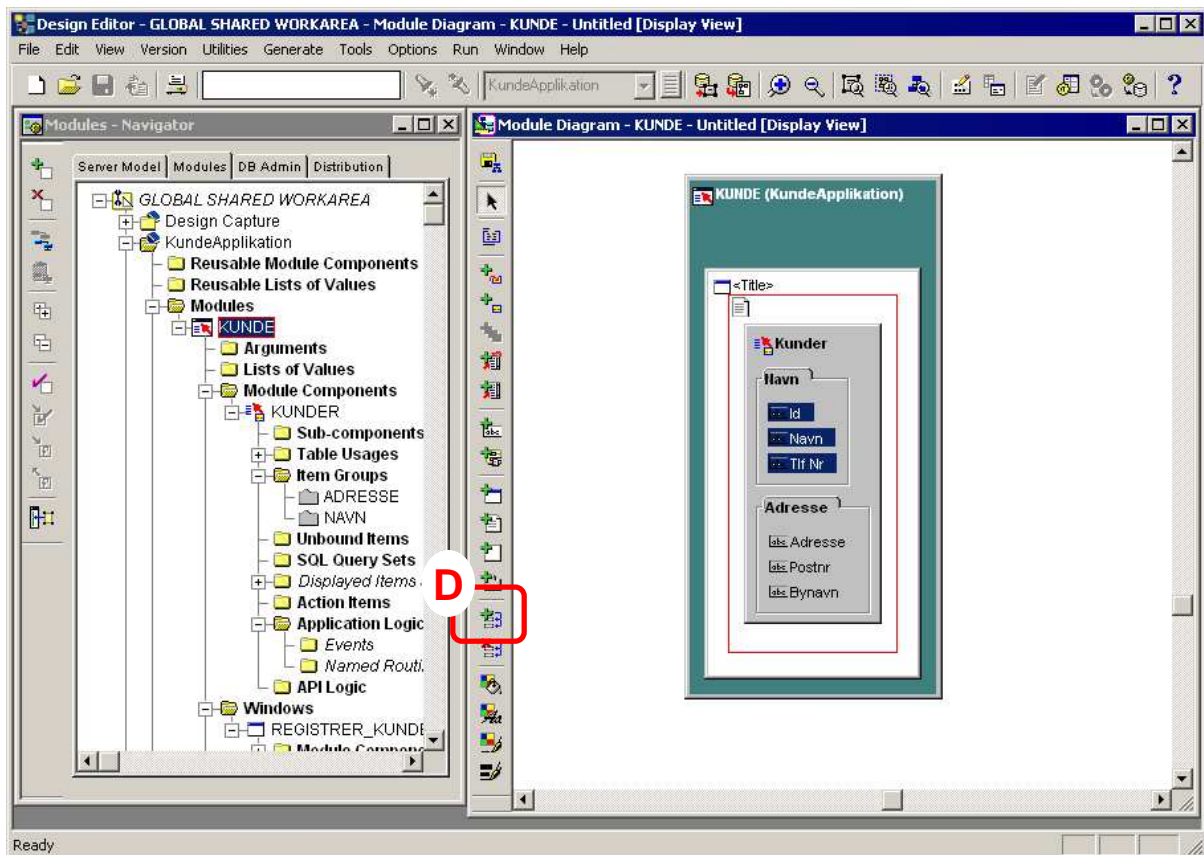


Man kan nu gå ind og ændre på rækkefølgen i hvordan elementerne skal vises i modulet (Formen i dette tilfælde) ved at trække rundt på elementerne. F.eks. kan man trække attributten Postnr op over Tlf Nr, hvis man gerne vil have telefonnummeret til at stå sidst. Der er også mange egenskaber (eksempelvis feltstørrelse, antal rækker der vises, standardværdier etc.) ved de enkelte elementer, som kan ændres ved at se på properties for eksempelvis en tabel eller attribut. Husk igen, at der kan skiftes mellem Property Palette og Property Dialogs ved at trykke på knappen i ramme B.

Der kan laves knapper som henter andre vinduer frem fra samme modul eller aktiverer andre moduler ved at trykke på knappen i ramme C, og derefter markere at knappen skal være i et Vindue (hvide kasser) eller modulkomponent (grå kasser).



I Display Viewet på denne side er postnummer-tabellen tidligere sat til at være en Lookup Table i Data View, hvilket betyder at alle attributter/kolonner som standard er samlet i en modulkomponent. Desuden er attributterne samlet i to faneblade i dette Display View.



Man kan lave faneblade ved at markere en eller flere attributter, og derefter trykke på knappen i ramme D, som angiver at man vil lave en ramme.

Vinduet her nedenunder viser sig, når man trykker på ovennævnte knap til at lave en ramme. "Item Group Name" er det navn, som Designer kender rammen på, mens "Prompt" er det navn, som vil vise sig for brugeren. Jeg finder det praktisk at bruge samme navn i begge felter, hvis det er muligt.

Stacked står normalt til No, hvilket betyder at rammen bare bliver en ramme, som omkranser de attributter man vælger. Når Stacked sættes til Yes, dannes istedet et faneblad, hvilket var meningen her. Almindelige rammer er også nyttige til at adskille elementer på samme skærbillede.

Når der trykkes Next, kan man se hvilke attributter, der skal være med i rammen/fanebladet. Hvis man har valgt korrekt, da man markerede attributterne tidligere, kan man bare trykke på Finish.

**Create Item Group in KundeApplikation : Name**

To create an item group quickly with minimal details, enter the name and click Finish. You may also wish to enter a prompt, specify whether the item group is stacked or unstacked, and enter the name of a template object. Further details may be entered on the following pages.

Item group name:  Prompt:

Template/library object:  Layout Style:

Stacked?  Yes  No

Relative Tab Stops

Tab Stop Scope:  Start:  End:

Cancel Help < Back Next > Finish

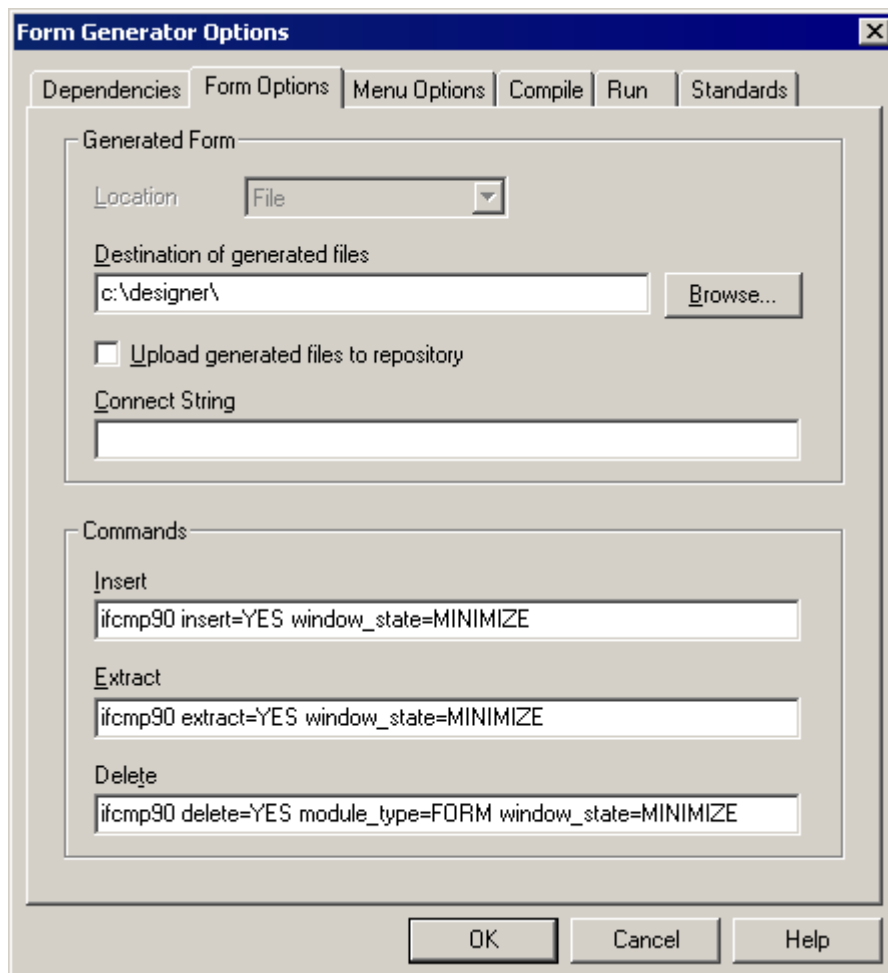
Når man er tilfreds med modulet, så går man op i menuen "Generate" og vælger "Generate Module..." for at danne det konkrete modul, som brugeren skal anvende. Man kan også vælge "Generate Module As ..." for selv at bestemme hvilket type modul, der skal genereres.

I det første skærbillede som vises, når man vil generere en Form (hvis det istedet er en Report er skærbillederne noget simplere), skal der normalt ikke ændres noget. Men der er nogle Options, som kan være hensigtsmæssige at redigere, inden man laver modulet. Options skal kun ændres en gang, da Designer herefter husker på indstillingerne.

The image shows a "Generate Form" dialog box with the following fields and options:

- Module:** KUNDE
- Generate Option:** Form
- Options:**
  - Ignore User Application Logic
  - Include Module Network
  - Generate Associated Menus
  - Generate Attached Libraries
- Preserve Layout:**
  - No
  - Yes
  - Source Form Name: KUNDE.fmb
  - Buttons: Browse, Reset
- Standards:**
  - Object Library Name: ofgwebol.olb
  - Template Name: ofgwebt.fmb
  - Buttons: Browse..., Reset
- Post Generation:**
  - Commit
  - Revert
  - Browse/Edit mode
- Bottom Buttons:** Start, Options..., Cancel, Help

Inde i Options skal man i 3 faneblade (Forms Options, Menu Options og Compile) skrive en destinations-sti for hvor ens færdige moduler skal placeres. Hvis man ikke skriver en sti bliver modulerne placeret i et underbibliotek til Oracle-biblioteket på computeren. Under fanebladet Run fjernes krydset for at køre Formen automatisk, da det for det første ofte er relevant at lave ekstra tilpasninger af Formen i Forms Builder og endvidere nemmere at teste den fra Forms Builder. For en Report er der andre Options, hvor det kun er en sti kaldet "Destination of senerated files", som skal defineres.



Tryk til sidst OK til disse Options og tryk på Start for at danne modulet. Der vil som ved dannelsen af databasen fremkomme et Message Window, hvor evt. fejl ved modulet vil vise sig. Hvis fejlene står med blå tekst, virke modulet stadigvæk, mens røde fejl vil medføre, at modulet ikke virker. Samtidig med, at modulet viser sig, popper en dialogboks frem i Designer, hvor der er mulighed for at gemme ændringer, således at visse fejl ikke viser sig igen. Disse ændringer er dem som Designer selv foretager i modulet, og som står beskrevet med blå tekst i Message Window. Hvis man istedet vælger Revert her, laver Designer ikke nogle ændringer, og de samme fejl vil vise sig næste gang, man genererer det pågældende modul.

Forms og Reports kan efterfølgende finpudses og testes i henholdsvis Forms og Report Builder.