

Armeringsspecifikation

Utveckling av en armeringsmodul till AutoCAD

Birgir Karlsson

Sektionen för Väg och vattenbyggnad
Avdelningen för Bärande Konstruktioner
Lunds Tekniska Högskola
Box 118
221 00 LUND

School of Civil Engineering
Department of Structural Engineering
Lund Institute of Technology
Box 118
S-221 00 LUND
Sweden

Utveckling av en armeringsmodul till AutoCAD

Developing reinforcement program for AutoCAD

av
Birgir Karlsson
1997

Abstract: The report includes description of two programs that have been developed for drawing reinforcement in AutoCAD and making reinforcements reports according to Swedish Standard.
(Swedish).

Rapport TVBK-5087
ISSN 0349-4969
ISRN: LUTVDG/TVBK--5087 --SE

Examensarbete
Handledare: Dan Pettersson
Juni 1997

FÖRORD	3
KAPITEL 1	5
BESKRIVNING AV SJÄLVA EXAMENSARBETET OCH HUR ARBETET HAR GÅTT.....	5
<i>Bakgrund</i>	5
<i>Första stegen</i>	5
<i>Examensarbetet, definition</i>	6
<i>Examensarbetets gång</i>	7
<i>Program och komponenter som används</i>	7
<i>Böcker som har använts</i>	8
<i>Definitioner</i>	9
HUR SKALL PROGRAMMEN SLUTLIGEN FUNGERA I FRAMTIDEN?	9
KAPITEL 2	11
ARMSPEC1 PROGRAMMET.....	11
<i>Kort beskrivning av miljön</i>	11
<i>Välja projekt</i>	12
ANVÄNDA MENYER I ARMSPEC1	12
<i>Filer menyn</i>	12
<i>Nytt projekt</i>	13
<i>Importera information om armeringsjärn</i>	14
<i>Redigera</i>	16
<i>Grunddata</i>	16
<i>Ändra information om armeringsjärn i ArmSpec1</i>	16
<i>Data i Logfilen</i>	18
<i>Ändra grunddata</i>	18
<i>Kvalitet</i>	19
<i>Armeringstyp</i>	19
<i>Rapport</i>	20
<i>Skriv ut armeringsspecifikation</i>	20
<i>Reviderings lista</i>	21
<i>Rapporter till skrivare eller på skärmen</i>	21
TILL SLUT I ARMSPEC1	22
KAPITEL 3	23
ARMSERV1 PROGRAMMET.....	23
<i>Kort beskrivning av miljön</i>	23
ANVÄNDA MENYERNA I ARMSERV1	24
<i>Skapa DDE!</i>	24
<i>Rita järn</i>	24
<i>Hitta järn</i>	25
<i>Kopiera!</i>	25
<i>Ändra!</i>	25
<i>Ta bort!</i>	26
<i>Filtrera data!</i>	26
<i>Använda data!</i>	26
<i>Nytt nummer</i>	27
<i>Spara till databas</i>	27

KAPITEL 4	28
DATABASER I ARMSPEC1.....	28
<i>Gemensamma data</i>	28
<i>Tabeller som är unika för varje projekt</i>	29
DATABAS I ARMSERV1.....	30
HJÄLP PROGRAMMET.....	30
RESOURCE-FILEN.....	31
KAPITEL 5	32
HUR PROGRAMMEN KAN UTVECKLAS VIDARE.....	32
SLUTSATSER.....	33
BILAGOR	34
BILAGA A, INSTALLERA PROGRAMMEN.....	34
<i>ArmSpec1</i>	34
<i>ArmServ1</i>	34
<i>AutoLisp</i>	34
BILAGA B, UTSKRIFT FRÅN MÖTE 970121.....	35
BILAGA C, TYPBLAD FÖR BOCKNING.....	38
BILAGA D, VAD ÄR EN ARMERINGSSPECIFIKATION?.....	39
BILAGA E, STRUKTUREN I PROGRAMMEN.....	40
<i>ArmSpec1</i>	40
<i>ArmServ1</i>	41
<i>AutoLisp rutiner</i>	42

Förord.

Detta examensarbete handlar om att utveckla en armeringsmodul till AutoCAD i samarbete med BODAB.

Bakgrunden till examensarbetet är att på BODAB har pågått utvecklingsarbete kring ett projekteringsverktyg i AutoCAD. Verktöget som är utvecklad i AutoLisp, skall hjälpa projektörer att i AutoCAD 13 rita upp och specificera armering. Från ritningen skall komplett armeringsspecifikation genereras. Detta examensarbete är ett försök att lösa problemet helt automatiskt. Lösningen ligger i två program som har skrivits:

- **ArmSpec1:** Är ett Delphi-program som kan importera information om armeringsjärnen via textfil. Användaren kan även definiera armeringsjärn helt manuellt. Programmet kan skriva ut armeringsspecifikation enligt svensk standard och håller reda på alla revideringar efter att armeringsspecifikationen är officiellt utskickad.
- **ArmServ1:** Är ett annat Delphi-program som tar hand om alla databas hanteringar när armeringsjärn ritas eller att om behöver att ändra information om armeringsjärn. ArmServ1 kopplas till AutoCAD via DDE-länkar. ArmServ1 använder egna procedurer och/eller kör igång AutoLisp rutiner i AutoCAD.

Denna rapport är delad i fem kapitel. I kapitel ett beskrivs själva examensarbetet, hur arbetet har gått och sedan hur man skulle kunna tänka sig den slutliga lösningen. I kapitel två beskrivs hur ArmSpec1-programmet är uppbyggt och vad man kan göra i programmet. I kapitel tre beskrivs hur ArmServ1 programmet är uppbyggt och vad man kan göra i programmet. I kapitel fyra beskrivs databaser, hjälpprogrammet och hur resource-filen används. I kapitel fem diskuteras hur programmen kan utvecklas vidare i framtiden tillsammans med sammanfattning och slutsatser.

I bilaga A beskrivs hur man installerar ArmSpec1- och ArmServ1-programmen. Bilaga B är utskrift från möte 970121. I bilaga C är Typblad för bockning av stänger enligt Svensk Armering. I bilaga D visas utskrift av armeringsspecifikation. Bilaga E är en kort beskrivning hur programmen är uppbyggda.

Själva programmen och programkoden till programmen följer med på disketter. Eftersom programmen tillsammans är mer en 6000 programrader, är det meningslöst att skriva ut alla programmen på papper.

I hjälp-programmet finns detaljerad information om hur programmen fungerar. Denna rapport innehåller nästan samma beskrivning om hur programmen fungerar och det finns också bilder av de dialogrutor som finns i programmen. Om man läser hjälpen samtidigt som man använder programmen går det även snabbare att behärska programmen.

Ett programmeringsarbete tar faktiskt aldrig slut. Det finns alltid möjligheter att utveckla ett program vidare. Utvecklingen kan styrs av behov förändringar eller bara lust att lägga till nya funktioner och nytt utseende i programmet. Eftersom examens-

arbetet är tidsbegränsat var det nödvändigt att sätta gränser även om det finns behov för ännu fler funktioner än som redan finns i programmen.

Det fanns ingen möjlighet att skriva färdigt alla de funktioner som man skulle kunna tänka sig att ha i programmen. I stället finns i denna rapport beskrivning hur programmen möjligen kan fungera i framtiden. Denna beskrivning skapar ett underlag till diskussion om hur sådan program verkligen ska fungera och vilka funktioner saknas och vilka som behöver ändras.

Eftersom författaren till programmen och denna rapport bara har varit två år i Sverige kan spåket vara konstigt ibland, men förhoppningsvis kan läsaren förstå det mesta som finns i denna dokumentation.

Kapitel 1.

Beskrivning av själva examensarbetet och hur arbetet har gått.

Bakgrund.

I AutoLisp fanns hos BODAB definierade rutiner för några järntyper. Programmen frågar efter information om respektive armeringsjärn och ritar järnet. Information om järnet lagrades som attribut till symboler. När alla armeringsjärn är ritade används en annan AutoLisp rutin för att samla ihop information om järnen och att spara till en textfil med en post av attributvärden för var armeringsstång.

Textfilen importerades i Microsoft Excel och anpassades via ett antal makron till en komplett armeringsspecifikation. Detta användaravsnitt fungerade men var föga användervänligt.

Huvudmålet med examensarbetet var skriva ett användervänligt program för att hantera data från AutoCAD och att länka ihop ritningen och armeringsspecifikationen dynamiskt och att armeringsspecifikationen genererades så automatiskt som möjligt.

Första stegen.

I början var frågan om Excel verkligen var det bästa alternativet för att generera armeringsspecifikationen. Det finns många möjligheter att lösa problemet och eftersom det handlar ganska mycket om data skulle man kunna tänka sig alternativ som byggde på en underliggande databas. De alternativ som diskuterades var:

1. Fortsätta använda Microsoft Excel och utveckla makron för att hantera data. Fördelen med det är att, det redan finns ett system som fungerar och de flesta kan använda Excel. Nackdelen är att användaren behöver Excel för att köra programmet och om programmet utvecklats vidare i framtiden, och större mängd data behövs hanteras, kan det uppstå problem (gränser) om man kanske ville lägga till en databas för att hålla reda på data.
2. Använda Microsoft Access. Fördelen är att man genast lägger till värden till databasen och det skulle inte vara några gränser i utveckling i framtiden. Nackdelen är att man måste använda ett ganska stort och omfattande program för att hantera data.
3. Använda Delphi för Windows och skriva ett självständigt program. Fördelen är att det är mest flexibelt. Man kan börja med vanliga Pascal filer för att ta hand om data. Senare kan man mycket enkelt lägga till databas om det behövs. Om det finns möjlighet nu, eller det blir möjlighet i framtiden att köra DLL från AutoCAD då blir det enkelt att definiera programmet som DLL.

Efter diskussion valdes alternativ 3, d.v.s. att skriva ett självständigt Windows program i Delphi.

Examensarbetet, definition.

Examensarbetet delades i två delar härnäst definierade som **Del-1** och **Del-2**. Man kan säga att lösning av Del-1 ligger i ArmSpec1-programmet och lösning av Del-2 ligger i ArmServ1-programmet och några AutoLisp rutiner.

Del-1. Program som genererar armeringsspecifikationen.

Skriva ett självständigt program i Delphi för Windows. Programmet använder vanliga Pascal filer för att lagra data.

Indata i programmet blir txt-filer från AutoLisp programmet d.v.s. txt-filer skall importeras automatiskt.

Programmet skall ge användaren möjlighet att hålla reda på alla ändringar av information som kommer automatiskt via txt-filer från ritningen.

Användaren skall kunna lägga till, sortera och undersöka data på skärmen.

Beräkningsmetoder finns i Excel filer som BODAB har utvecklad.

Användaren skall kunna samla ihop alla armeringsjärn på alla ritningar och få summa per projekt, per gjutetapp etc. på skärm eller skrivare.

Programmet skriver ut armeringsspecifikation på skrivare.

- Analysera målgrupp och deras arbetsätt.
- Undersöka Svensk Standard och om det finns andra lösningar som man kan lära sig av eller använda metoder som redan finns.
- Undersöka nuvarande Excel program (nuvarande lösning) och definiera i detalj hur det nya programmet skall fungera. Göra en sammanfattning som kan diskuteras.
- Skriva en modul som laddar in txt-filen, processar data och sparar till en fil.
- Skriva en modul där användaren kan undersöka data på lättillgänglig sätt. Han kan också ändra och lägga till ny information. Programmet tar reda på vem som gör ändringarna, vilken information som blivit ändrad och vilken som är ny.
- Skriva en modul som beräknar allt som behövs för armering specifikationen.
- Skriva en modul som skickar armeringsspecifikationen till skrivare.
- Skriva "Options"-modul. Användaren kan ändra grundinformation för programmet.
- Manual för programmet med exemplar.

Resultatet för Del-1 d.v.s. **ArmSpec1**-programmet beskrivs detalj i kapitel två.

Del 2. Ytterligare järntyper. (5 veckor).

BODAB har redan skrivit AutoLisp program för 5 järntyper (A-E).

Skriva AutoLisp program för de järntyper som finns kvar (F-Y), 18 stycken.

Senare visade det sig att BODAB var mer intresserad av att koppla ihop armeringsjärnet på ritningen och data i databasen. Del-2 var därför ändrad till att skriva en prototyp av ett program som kan rita ett armeringsjärn av B-typ som kopplas till en post i databas. En lösning till det ligger i ArmServ1 programmet. I nuläget finns inget färdigt produkt men en principlösning beskrivs i kapitel tre.

Examensarbetets gång.

Examensarbetet började med att definiera och mer detaljerat utveckla hur ArmSpec1-programmet ska fungera, studera Delphi och undersöka programmeringsspråket. Börja att skriva ArmSpec1 och studera hur man genererar hjälp program, RTF filer i Word, kompilering o.s.v. Studera Resource-filer och hur man använder dessa. Recourse-filen i ArmSpec1 programmet innehåller texten som användaren kan läsa på skärmen. Således kan programmet lättare omarbetas till ett annat språk. Därefter definieras Pascal filer och rutiner för att hantera data från filerna, sorteringsrutiner m.m. Det visade sig senare att det var mycket enklare att använda databaser för datahanteringar, eftersom det finns massor av funktioner färdigskrivna som man kan använda i Delphi.

Efter några veckors arbete fanns det en prototyp av ArmSpec1-programmet med layout och vissa funktioner färdiga. I början används Paradox-databaser men senare blev detta ändrat till dBase IV-databaser. Efter redovisning och diskussion specificerades de flesta funktionerna i ArmSpec1 programmet. Rapport från mötet visas i bilaga B.

När ArmSpec1-programmet var nästan färdigt påbörjades ArmServ1-programmet. Som beskrivs senare i detta avsnitt dök det upp ganska många problem som resulterade i ändringar i själva problematiken. Det finns många möjligheter att lösa problemen och innan man har provat de olika alternativen vet man inte vad som är bästa lösningen. Man behöver bara prova och undersöka.

Program och komponenter som används.

Programmen och komponenter som har studeras i detta examensarbete är:

AutoCAD för Windows (r13)

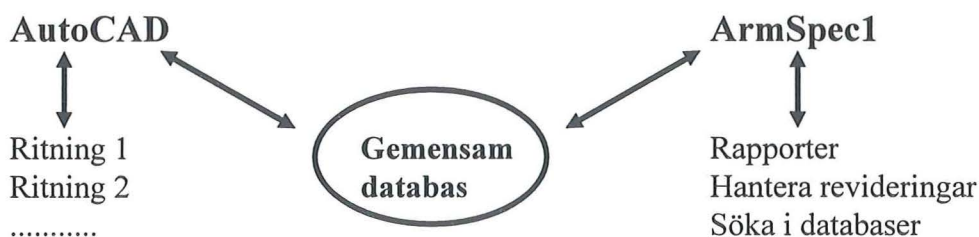
Själva AutoCAD-programmet används för att rita armeringsjärnen som vanligt via ett antal AutoLisprutiner.

AutoLisp

Alla armeringsjärn måste ritas med AutoLispprogram för att kunna skapa och uppdatera länkar mellan ritningen och databasen. Programmen frågar användaren efter nödvändiga informationer för att kunna rita armeringsjärnen. AutoLisp programmen körs från ArmServ1 programmet.

Databas hanteringar i AutoCAD.

Den ursprungliga föreställningen var att använda AutoCAD för att hantera all data som skapas i AutoCAD och använda External Databas-funktionerna för det. AutoCAD och ArmSpec1 skulle då använda samma databas.



Databasrutinerna i AutoCAD visade sig att vara ganska primitiva och långsamma. De klarar bara de enklaste funktionerna som behövs. Det är ganska svårt att använda mer komplicerade databashanteringar och tar längre tid jämfört med t.ex. DBE i Delphi. Det lyckades inte heller att få AutoCAD till att använda dBase IV tabeller. Det finns flera nackdelar med att använda databashanteringarna i AutoCAD, men efter som de inte används i den lösningen som här presenteras, är det meningslöst att behandla det mera här.

DDE länkar i AutoCAD.

Det finns inte så mycket skrivit om detta ämne i handböckerna, man måste prova sig fram. Det visade sig att AutoCAD kan inte koppla sig till ArmServ1-programmet utan att ArmServ1 ligger i samma katalog som AutoCAD. Man vill kunna lägga ArmServ1 programmet i vilken katalog som helst.

Databaser.

Programmen använder dBase IV format som DBE i Delphi tar hand om. Det är ett mycket vanligt format som många andra program kan läsa, t t.ex. Microsoft Excel och Access. Hur databasen är uppbyggd beskrivs i kapitel fyra.

Delphi för Windows.

Mycket kraftfullt programmeringsspråk som använder Pascal programkod. Det finns många tilläggskomponenter som man kan använda för att programmera i Windows.

DBD (databas desktop) för Delphi.

DBD är ett verktyg som följer med Delphi. Man kan definiera databaser och tabeller i olika format, redigera data eller omstrukturera tabeller.

InstallShield Express

Program för att generera installationsdisketter för Delphi program.

Resource-filer och Hjälpfiler.

Dessa beskrivs i kapitel fyra.

Böcker som har använts.

Följande lista innehåller de böcker eller litteratur som har använts i detta examensarbete.

- AutoCAD-handböckerna
- Delphi-handböcker (på CD)
- Using Delphi 2, The Most Complete Reference, QUE.
- AutoLisp Programmer's Reference Manual, AutoDesk.
- Kom igång med AutoCAD - Systemutveckling, Studentlitteratur.

Hjälpprogrammen som följer med AutoCAD och Delphi används också, speciellt hjälpen för Delphi. Där kan man hitta svar till de flesta frågorna som uppkommer när man programmerar i Delphi.

Definitioner.

Följande begrepp definieras som:

Armeringsjärn:

- i databasen: En rad i databasen, d.v.s. all information som finns om ett armeringsjärn i databasen. I kapitel fyra definieras databasen som används av programmen.
- i AutoCAD: Alla objekten som beskriver ett armeringsjärn på ritningen, d.v.s. en polyline, cirkel, texten etc.

Gjutetapp: Projektet är uppdelat i sektioner. Varje sektion kallas gjutetapp.

Bockningsradie: För varje järndiameter och kvalitet finns en bockningsradie. Bockningsradie är den radie som användas vid bockning av armering.

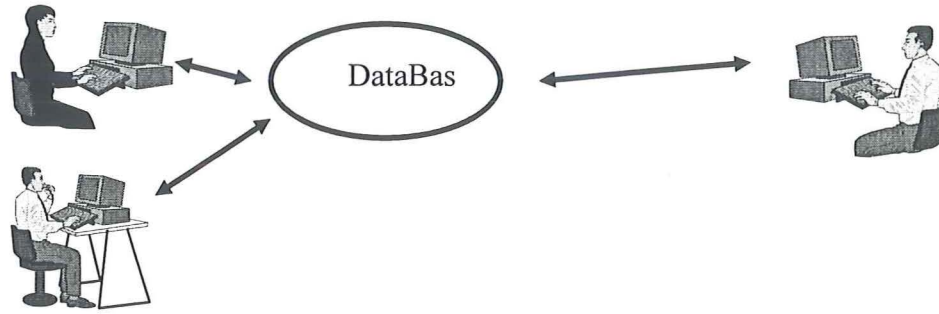
Hur skall programmen slutligen fungera i framtiden?

När allt är klart och allt fungerar på rätt sätt skulle man kunna tänka sig följande:

Användaren startar ArmSpec1-programmet från Startmenyn i Windows. Om han försöker ändra information om armeringsjärn eller texten på ritningen behöver uppdateras, startas AutoCAD automatiskt och respektive ritning laddas. AutoLisp rutiner ändrar sedan texten på ritningen. Om man ändrar information om armeringsjärn i en gjutetapp som är officiellt utskickad håller programmet reda på detta på samma sätt som ArmSpec1-programmet gör i dag.

Om man vill rita ett nytt armeringsjärn måste man välja projekt som vanligt. Därefter väljer man att koppla sig till AutoCAD och då öppnas en dialogruta som ser likadan ut och fungerar som ArmServ1-programmet gör idag. Se kapitel tre.

Om flera datorer är kopplade samman i nätverk, kan databasen ligga på servern. Då kan konstruktörer jobba på samma projekt samtidigt och programmet håller reda på nästa lediga littra eller om det finns armeringsjärn i databasen med samma egenskaper.



Om man sen vill skriva ut rapporter kan man göra det när som helst eftersom databasen alltid uppdateras genast då armeringsjärn ritas.

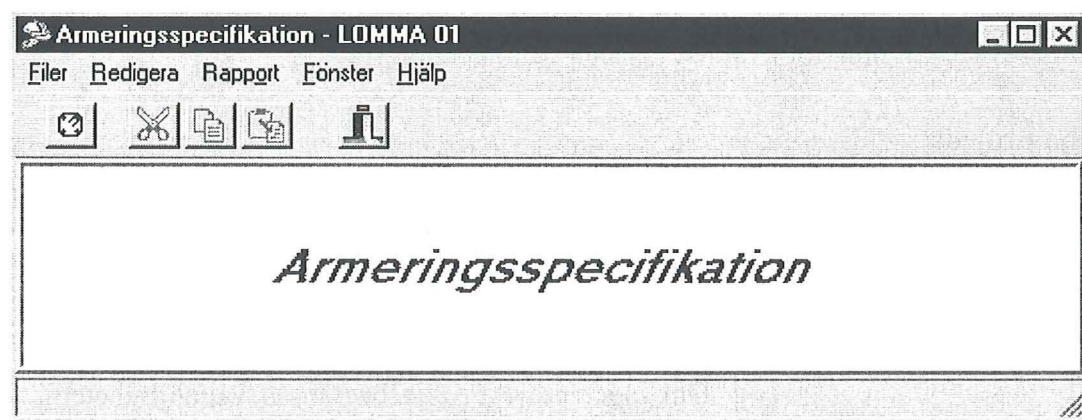
I kapitel fem diskuteras vilka funktioner man vill lägga till i nuläget.

Kapitel 2.

ArmSpec1 programmet.

Kort beskrivning av miljön.

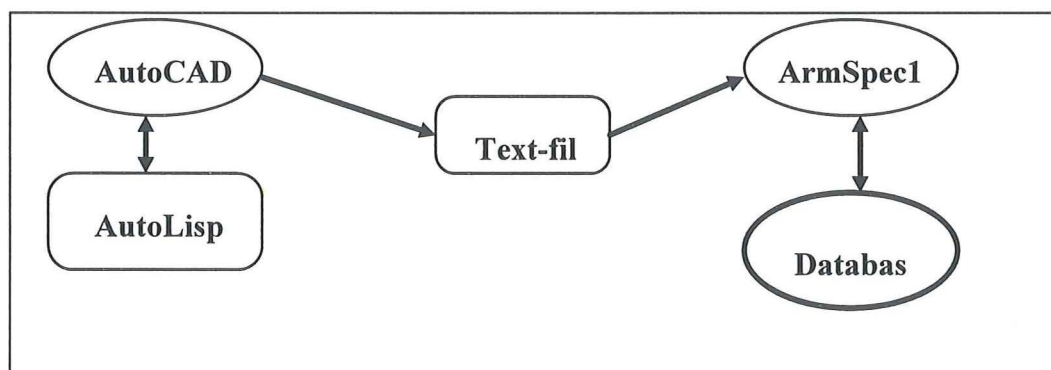
ArmSpec1 håller ordning på, sorterar och genererar listor på skärm eller skrivare. Listorna innehåller information om armeringsjärn i ett visst projekt. Beräkningar görs för korrekationer, längder och vikter. Rapport genereras enligt Svensk Standard. Information om armeringsjärn kan matas in automatiskt eller manuellt. Programmet håller också reda på alla revideringar på järnen. Programmet använder databasformatet dBase IV för att lagra data.



Innan man kan importera eller mata in armeringsjärn behöver man att skapa ett projekt och sedan välja projektet som aktivt.

Det finns tre möjligheter för att definiera armeringsjärn i programmet.

Det första är det ursprungliga sättet, d v s att rita järnet med de AutoLisp-programmen som finns hos BODAB för några järntyper.



Användaren AutoLisp kör rutiner som i sin tur frågar efter information om respektive armeringsjärn och sedan ritar järnet. Information om järnet lagras som attribut till AutoCADs symboler. När alla armeringsjärn ritats används en annan AutoLisp-rutin för att samla ihop information om järnen och att spara den till en textfil med en post

av attributvärden för var armeringsstång. Textfilen importerar sedan i ArmSpec1 som lagrar informationen i databas. På detta sätt finns ingen koppling mellan järnet på ritningen och informationen om järnet i databasen.

En annan möjlighet är att definiera armeringsjärn i programmet ArmSpec1 utan att rita något i AutoCAD. Hur man gör det visas under rubriken att ändra information om armeringsjärn.

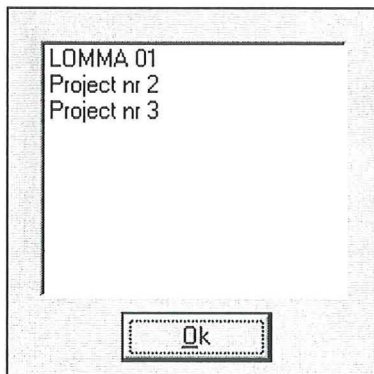
Den tredje möjligheten är att använda programmet ArmServ1 för att rita armeringsjärn och spara information om järnet direkt till databasen. Hur ArmServ1 programmet fungerar beskrivs i kapitel tre.

Om det finns armeringsjärn i ett projekt kan man skriva ut armeringsspecifikation sorterad efter gjutetapp.

Om det finns gjorda revideringar kan man skiva ut en reviderings lista.

Grunddata, dvs data som är gemensam för alla projekt, kan ändras. Med Armeringstyp definieras alla de armering typer som finns i systemet. Med Kvalitet definieras beskrivning för kvalitet, densitet och bockningsradier.

Välja projekt



När programmet har startat finns det en dialogruta som innehåller alla projekt som finns i systemet. Ett projekt kan väljas från listan och när dialogrutan stängs blir projektet aktiverat och dess namn visas på titelraden överst i programfönstret.

Om inget projekt väljs begränsas valmöjligheterna i menyerna till de funktioner som är allmänna och alltså inte är projekt specifika.

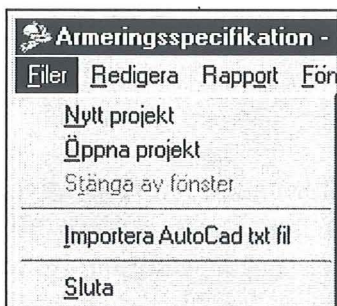
Om ett projekt inte finns på listan är det inte definierat i systemet. Stäng då dialogrutan och välj

Fil | Nytt projekt och lägg till ett nytt projekt. Sedan kan man välja Fil | Öppna projekt. Då öppnas dialogrutan igen och projektet väljs från listan som tidigare.

Det finns bara möjlighet att arbeta på ett projekt i taget.

Använda menyer i ArmSpec1

Möjligheterna i menyerna ändras allt efter vad användaren har valt. Om inget projekt väljs kan man bara arbeta med grunddata, dvs data som används i alla projekt. Efter att projektet valts öppnas flera möjligheter.

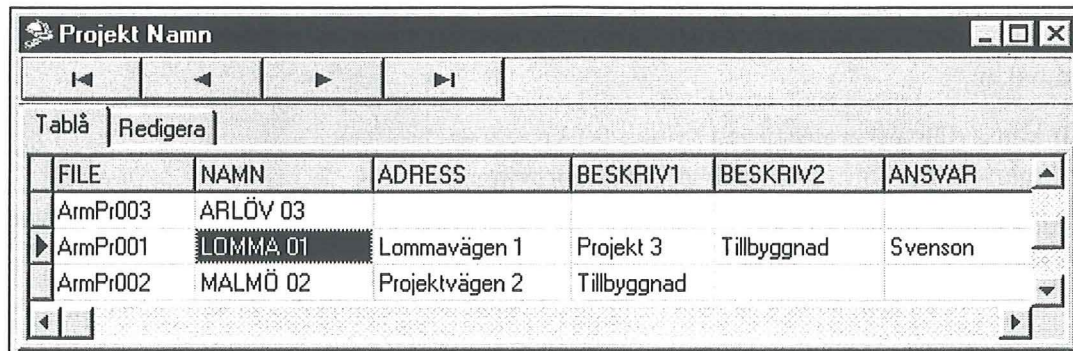


Filer menyn:

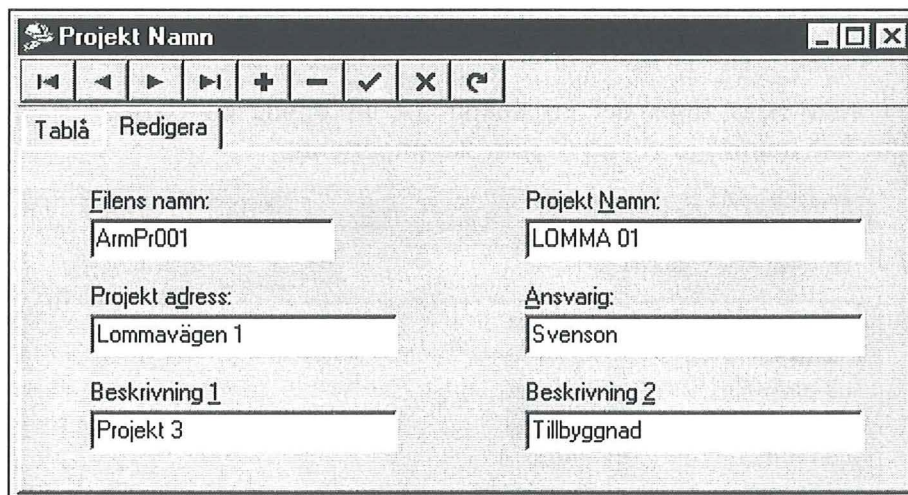
Nytt projekt används om man vill lägga till ett nytt projekt eller ändra nuvarande information om projekten. Öppna projekt används för att välja projekt. Stänga av fönster för att stänga av aktivt fönster. Importera AutoCAD txt fil om man vill importera armeringsjärn automatiskt från en AutoCAD txt-fil. Sluta används sedan för att stänga av

själva programmet.

Nytt projekt



I fönstret finns två flikar, Tablå och Redigera. Överst i fönstret finns knappar som används för att bläddra mellan de projekt som finns i systemet. Om man vill ändra något eller lägga till ett nytt projekt aktiverar man Redigera fliken.



För att lägga till ett nytt projekt, tryck på plus (+) knappen som då finns överst på sidan. I rutorna skriver man information om projektet:

Filens namn: Definiera namn på projektets filer. Använd bara tecken som kan användas i filnamn. Namnets längd kan vara från 1 till 15 bokstäver eller siffror.

Projekt namn: Definiera projektets namn. Det kan vara vanlig text eller siffror (30 tecken).

Alla andra rutor kan man fylla i frivilligt (30 tecken). Det som man skriver i rutorna används bara för att beskriva projektet på rapporter.

När man trycker på check (✓) knappen skapas ett nytt projekt. Två filer skapas, en fil som kommer att innehålla information om armeringsjärnen och en annan som kommer att innehålla information om revideringar. T.ex. om Filens namn är "VOMB-844" då skapas filer med namnen "VOMB-844dat.dbf" "VOMB-844log.dbf"

Om man försöker att skapa ett nytt projekt och det finns filer på hårddisken med samma filnamn får man meddelande om detta och nya projektet kopplas till de gamla filerna.

Om detta inte var meningen d.v.s. man vill starta ett nytt projekt utan något data då måste man skapa ett nytt projekt med filnamn som inte finns på hårddisken.



Man kan avlägsna projekt med minus (-) knappen.

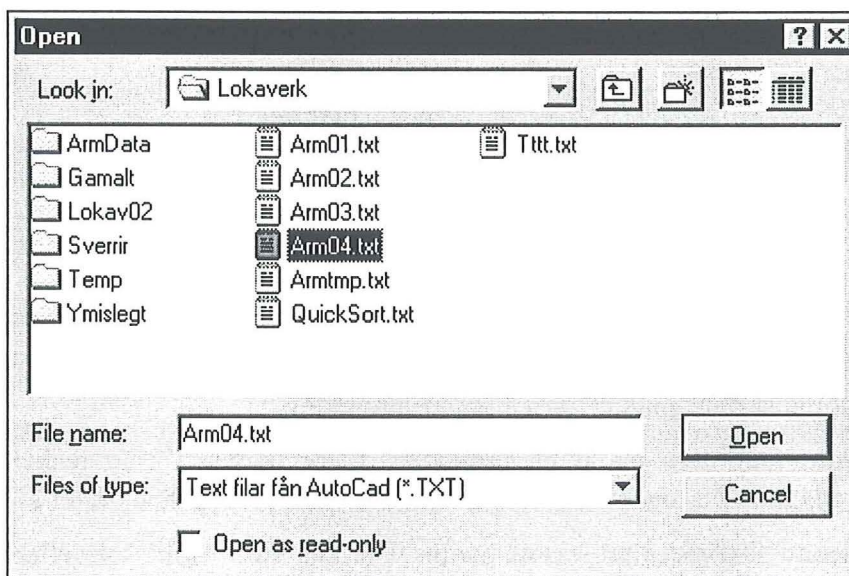
Om man väljer att avlägsna projekt försvinner information om projektet ur systemet men filerna som innehåller allt data finns fortfarande kvar på hårddisken.

Om man väljer att avlägsna aktivt projekt behöver man aktivera ett annat projekt efteråt.

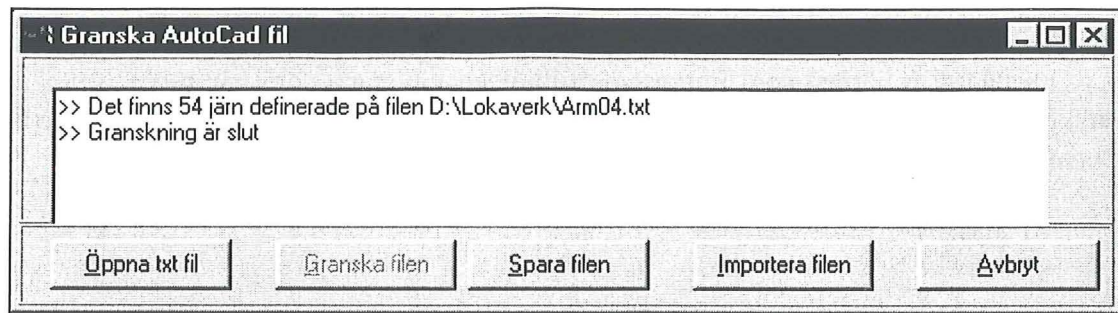
Importera information om armeringsjärn

Armeringsjärn importeras från AutoCAD txt-fil automatiskt. Innan man kan importera armeringsjärnen måste man välja ett projekt.

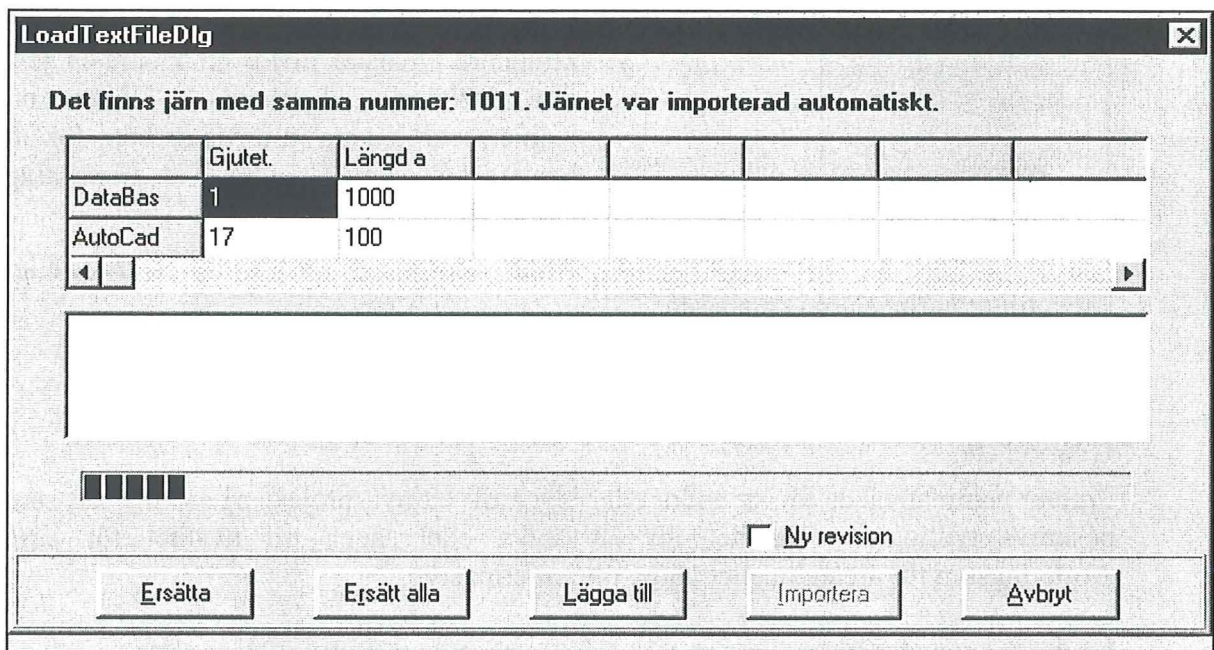
I fönstret som visas finns det en knapp för att öppna en txt-fil. I Öppna Fil-dialogrutan kan man välja vilken fil som ska importeras.



När txt-fil valts aktiveras knappen Granska filen. Om det finns någon fel i filen listas felen i fönstret. Man kan rätta felen i fönstret om man vill. Sedan kan man spara filen, öppna igen och granska. Detta görs tills filen är felfri. Man kan inte importera en fil så länge det finns fel i den.



När granskning är klar och filen är felfri kan den importeras. Ytterligare en dialogruta öppnas och man kan välja att importera eller avbryta. Om man avbryter stängs dialogen och inget information om järnen importeras till databasen.



Om man väljer ny revision ökas revideringsbokstaven, d v s om aktuell revidering är B kommer ny revideringsbokstav att bli C. Om inte ny revision väljs, används fortfarande sista revideringsbokstaven som finns för respektive gjutetapp.

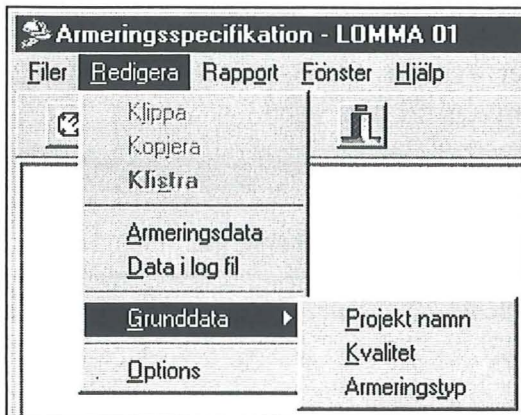
Följande regler gäller vid import av armeringsjärn:

- » Om armeringsjärn med samma nummer ej finns i databasen, läggs informationen till.
- » Om armeringsjärn med samma nummer finns och har samma egenskaper, behöver inget göras men ett meddelande visas på skärmen om att järnet redan fanns i databasen.
- » Armeringsjärn med samma nummer finns men med andra egenskaper, visas hur järnet som finns i databasen importerades (automatiskt, som dublett eller manuellt) och sedan visas skillnaden. Då kan man välja Ersätt eller Lägg till.

Om man väljer att Ersätt, får järnet i databasen samma egenskaper som järnet från AutoCAD filen.

Om man väljer att Lägg till, importeras järnet som dublett och man kan sortera järnen i fönstret för armeringsdata.

- » Om man väljer Ersätt alla, ersätter programmet alla järn utan att fråga.
- » Om något av järnen som importeras tillhör en gjutetapp som har blivit officiellt utskickad öppnas en Log dialog efter att man har tryckt på Avbryt. Den beskrivs senare.



Redigera.

Klippa ut används för att klippa markerad text till Windows klippbord. Kopiera för att kopiera markerad text till Windows klippbord. Klistra in för att klistra in text från Windows klippbord. Klippa ut, Kopiera och Klistra in används bara för att hantera text som genereras vid granskning av textfiler från AutoCAD.

Ändra information om järn för att ändra eller

lägga till armeringsjärn i databasen .

Data i Log Fil används för att ändra, lägga till eller ta bort data i log databasen.

Grunddata.

Projekt namn används för att ändra eller lägga till ett nytt projekt på samma sätt som beskrevs tydligare. Kvalitet för att ändra eller lägga till kvalitet för järn. Armeringstyp för att ändra eller lägga till armeringstyp.

I nuläget används Options inte men säkert när programmet utvecklas vidare.

Ändra information om armeringsjärn i ArmSpec1

ARM_NR	TYP	KVALITET	DIAM	ANTAL	GRUPP	GJ_TAPP	REV	B_TYP	VIN
62	N	Ks 500 ST		8	13	1	1		1
73	N	Ks 500 ST		10	66	1	1	A	1
74	Q	Ks 500 ST		12	136	1	1	B	1
125	A	Ks 500 ST		12	2	1	1		0
133	A	Ks 500 ST		16	6	1	1	B	0

I fönstret finns fyra flikar, Järn per gjutetapp, Mångfaldiga nummerad (Dubletter), Manuellt importerade och Redigera. Överst på sidorna finns det knappar som man använder för att bläddra i databasen. Om man vill ändra något eller lägga till nytt

aktiverar man Redigera fliken eller ändrar direkt i rutorna som finns i fönstret. För att lägga till nytt armeringsjärn, tryck på plus knappen som finns överst i fönstret när man har valt Redigera.

Man kan välja att titta på en gjutetapp i taget eller alla. I vänstra rullgardinslistan finns alla nummer på de gjutetapper som finns i projektet. I högra rullgardinslistan finns alla nummer på armeringsjärn som har använts i projektet. Om man försöker att lägga till ett nytt armeringsjärn med nummer som redan finns visas numret i rullgardinslistan. Om man ändå vill lägga till armeringsjärnet kan man göra det. Båda rullgardinslistorna uppdateras om användaren lägger till ett nytt värde, d v s en ny gjutetapp eller ett nytt littrar.

På sidan Dubletter visas bara armeringsjärn som blivit automatiskt importerade och har ett annat armeringsjärn med samma armeringsnummer i databasen. Om man väljer att ändra armeringsnumret till ett nytt nummer, definieras järnet som manuellt.

Under fliken Manuellt laddad visast bara armeringsjärn som har blivit manuellt definierade. Under fliken Redigera finns:

Redigera indata

1138

Ny revision

Järn per gjutetapp | Mångfaldig nummerad | Manuell inmatning | Redigera

Nummer:	Typ:	Kvalitet:	Diameter:	Antal:	Grupp:	Gjutetapp:	Bygeltyp:
62	N	Ks 500 ST	8	13	1	1	1

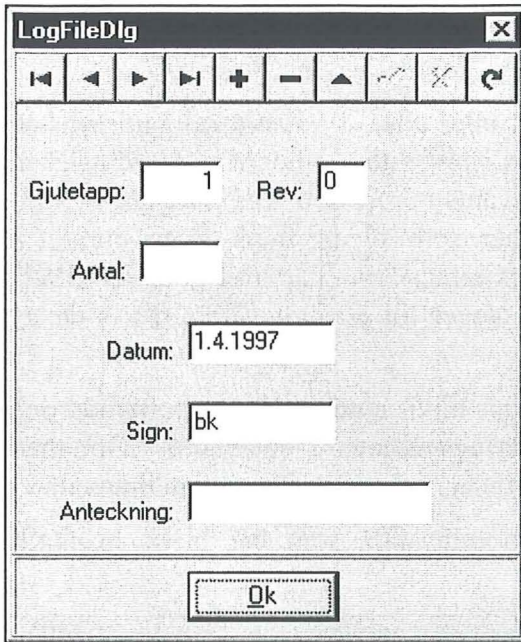
Längd a:	Längd b:	Längd c:	Längd d:	Längd e:	Längd f:	Längd g:
1000	1000	1000	1000	0	0	0

Vinkel v*	Vinkel u*	Längd x:	Längd y:	Åndtyp 1:	Åndtyp 2:
60	120	0	10		

Anteckning:

Samtidigt och programmet försöker spara järnet till databasen kollar programmet datan. Om information saknas får användaren meddelande på skärmen som visar detta. Järnet sparas till databasen men användaren skall då välja järnet igen och lägga till den information som saknas.

Om något av de ändrade järnen tillhör en gjutetapp som har blivit officiellt utskickad öppnas en Log dialog efter att man har stängt fönstret.



Data i Logfilen

I logfilen finns information om vilka gjutetapp är officiellt utskickad. I Logfil dialogen kan man bläddra genom logfilen. Det finns möjlighet att lägga till och avlägsna data i Logfilen.

Efter att man har importerat eller lagt till armeringsjärn i en gjutetapp som är officiellt utskickad måste man lägga till information om revideringen. Man ska lägga till signatur till de revideringar där signatur saknas. Det kan vara på olika gjutetapper efter som järnen som importerats kan vara från många olika gjutetapper.

Det finns plats för åtta revideringar per gjutetapp (A,B,C,D,E,F,G,H). Revidering0

betyder att gjutetappen är officiellt utskickad.

Även om man kan avlägsna revideringar ska man inte göra det efter att gjutetappen är officiellt utskickad. Om man gör det kan man få konflikt mellan Logfilen och databasen som innehåller själva järnen. (Till exempel om armeringsjärn har ändras tre gånger (Rev C) och revideringen tas bort då försvinner Rev B efter som programmet håller inte reda på det)?

Om man avlägsnar revideringar ska man alltid ta de sista bokstäverna. Om man har t.ex. kommit fram till C för gjutetapp 5 och vill avlägsna två revisioner ska man ta bort C och B. Nästa gång armeringsjärn importerats eller ändras blir armeringsjärn i gjutetapp 5 rev A eller B beroende på vad man har valt när man importerar järnen. Oklart!

När man importerar armeringsjärn kan man välja Ersätt eller Ersätta alla. Om man Ersätta alla skriver programmet över de nya järnen som redan finns i databasen. Detta betyder att om gjutetappen är officiellt utskickad definieras alla de nya järnen som reviderade. Om man i stället väljer att ersätta varje armeringsjärn för sig, definieras bara de respektive järnen som reviderade.

Ändra grunddata

I fönstret finns två sidor, Tablå och Redigera. Överst på sidorna finns det knappar som kan används för att bläddra mellan dem. Om man vill ändra något eller lägga till nytt aktiverar man Redigera-sidan. För att lägga till nytt, tryck på plus knappen som då finns ovan på sidorna.

Kvalitet.

I Kvalitet fönstret definierar man järnens kvalitet. Man kan skriva anteckningar om man vill.

Man behöver att definiera densitet. Programmet använder densitet för att beräkna den totala armeringsvikten för varje diameter i varje gjutetapp.

Därefter definierar man bockningsradie för varje järndiameter. Det finns tre kategorier:

B90_xx => Bygel - för alla vinklar - xx är järnets diameter.

AL90_xx => Vanligt järn - vinklar $\leq 90^\circ$ - xx är järnets diameter.

AH90_xx => Vanligt järn - vinklar $> 90^\circ$ - xx är järnets diameter.

Armeringstyp.

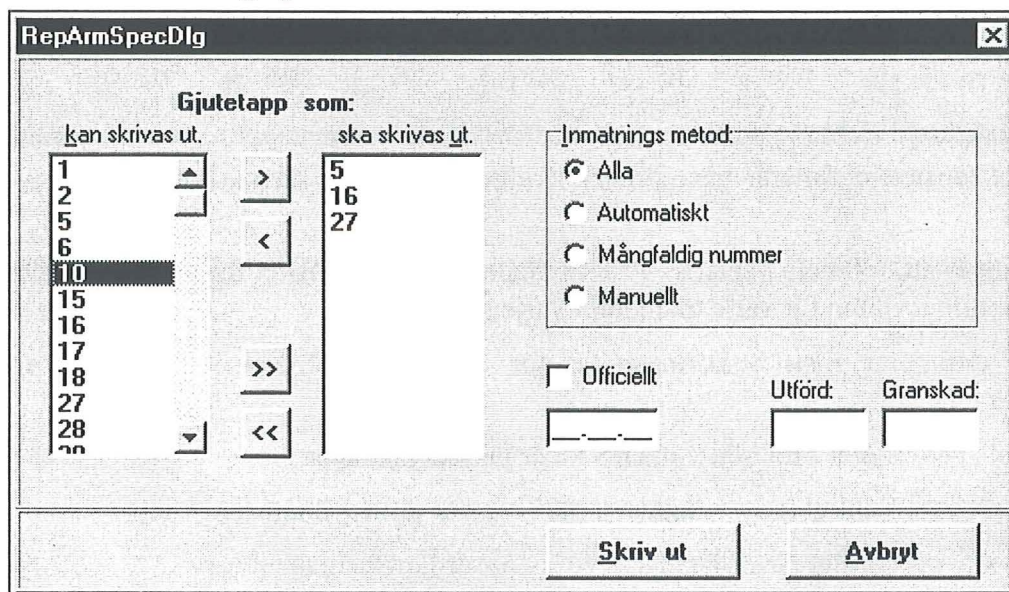
I fönstret för Armeringstyp definierar man typen enligt Svensk Standard. Typen definieras som stor bokstav (nödvändigt). Man kan om man vill lägga till anteckningar för varje armeringstyp.







Rapport

Skriv ut projekt lista används för att skriva ut en lista på de projekt som finns i systemet. Skriv ut kvalitet lista för att skriva ut en lista på de armeringskvalitéer som finns i systemet. Skriv ut armeringstyp lista används för att skriva ut en lista på de armeringstyper som finns i systemet.

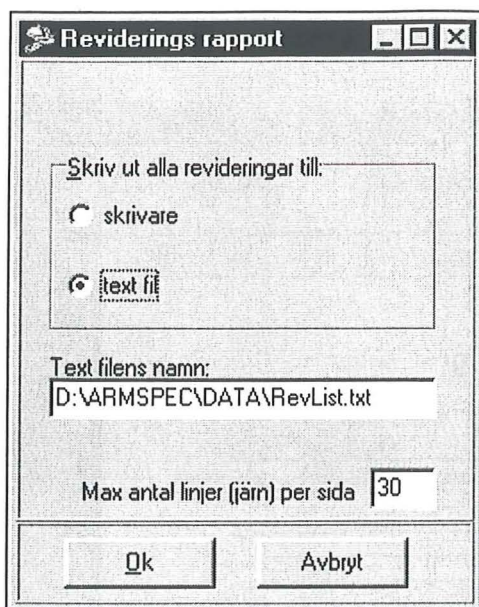
Skriv ut armeringsspecifikation



I dialogrutan finns en lista över alla gjutetapper som finns i projektet. Man väljer från listan och trycker på  knappen för att flytta gjutetappen över till listan som skrivs ut. För att ta bort gjutetapp ur listan som skrivs ut väljer man respektive gjutetapp och trycker på .

Om man vill flytta alla gjutetapp väljer man  respektive  knapparna.

Om man väjer Officiellt skrivs de valda gjutetapperna som är officiellt utskickade ut och information om detta lagras i Log filen. Det spelar ingen roll om någon av de gjutetapperna som ska skrivas ut redan har blivit officiellt utskickad.



Reviderings lista

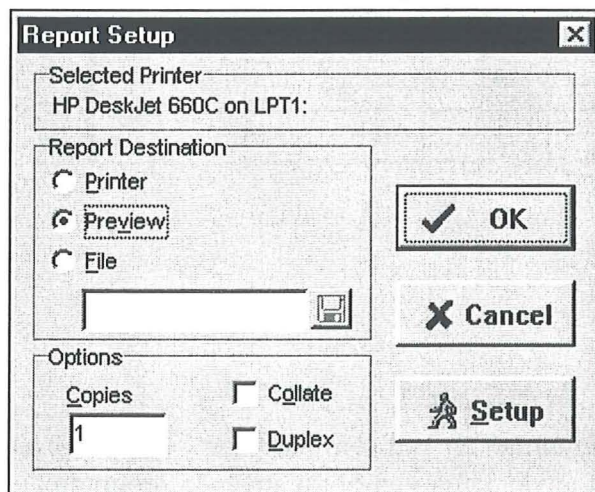
Man kan skriva ut reviderings lista b de p  skrivare och till en text fil.

Om man v ljer skrivare f r man som vanligt dialogruta d r som man kan v lja att skriva p  sk rmen eller direkt p  skrivare.

Om man v ljer att skriva till en text fil d  aktiveras textrutan och man beh ver definiera vilken katalog man vill att text filen hamnar.

Nederst m ste man definiera hur m nga armeringsj rn (maximalt) f r plats p  en sida.

Rapporter till skrivare eller p  sk rmen.

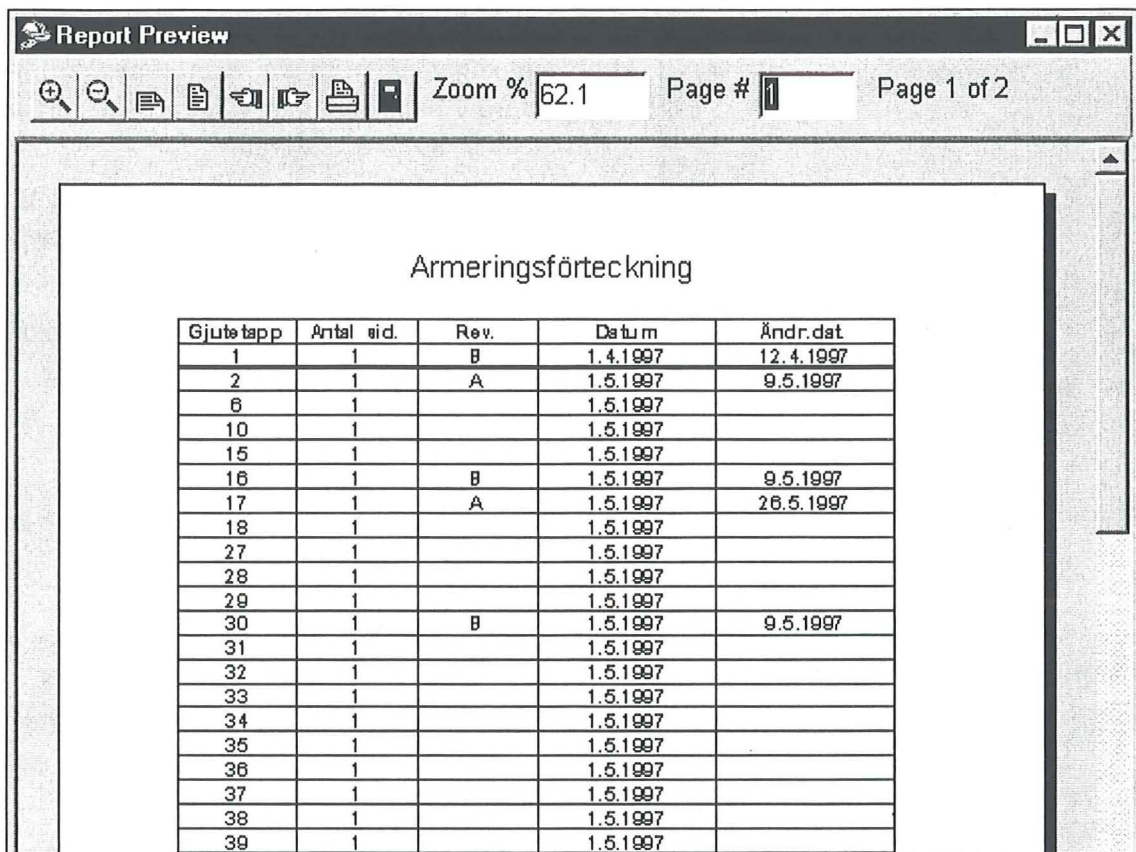


Man kan v lja att skicka alla rapporter direkt till skrivare eller f rst p  sk rmen och sedan till skrivare.

Man kan ocks  v lja Setup d  f r man standard Windows Setup dialogen f r att  ndra installationer eller bytta till en annan skrivare.

Om man v ljer att skriva direkt till skrivare skickas respektive rapport till skrivaren. Exempel p  utskrifter fr n programmet visas i bilagor med denna rapport.

Om man väljer att skicka rapporter till skärm då får man till exempel sådan rapport:



Report Preview

Zoom % 62.1 Page # Page 1 of 2

Armeringsförteckning

Gjute tapp	Antal eid.	Rev.	Datum	Ändr.dat
1	1	B	1.4.1997	12.4.1997
2	1	A	1.5.1997	9.5.1997
6	1		1.5.1997	
10	1		1.5.1997	
15	1		1.5.1997	
16	1	B	1.5.1997	9.5.1997
17	1	A	1.5.1997	26.5.1997
18	1		1.5.1997	
27	1		1.5.1997	
28	1		1.5.1997	
29	1		1.5.1997	
30	1	B	1.5.1997	9.5.1997
31	1		1.5.1997	
32	1		1.5.1997	
33	1		1.5.1997	
34	1		1.5.1997	
35	1		1.5.1997	
36	1		1.5.1997	
37	1		1.5.1997	
38	1		1.5.1997	
39	1		1.5.1997	

Till slut i ArmSpec1.

Det finns flera saker man kan göra i programmet än vad som här visas. Man hoppas att denna korta beskrivning räcker för att användaren kan börja använda programmet och prova de olika funktioner som finns.

Kapitel 3.

ArmServ1 programmet.

Detta kapitel beskriver vad man kan göra i programmet. När man har startat programmet visas följande dialogruta på skärmen.

UNIC_NR	ARM_NR	TYP	KVALITET	DIAM	AVST	ANTAL	GRUPP	GJ_TAPP	B_TYP	VINK_V	VINK_U	L_A	L_B
		B			250	8						1000	1000
3	1	B	Ks 400 s	12	200	15	1	10	0	0	0	1000	20
1	2	B	Ks 400 s	12	200	34	1	10	0	0	0	1500	20
4	2	B	Ks 400 s	12	150	23	1	10	0	0	0	1500	20
6	2	B	Ks 400 s	12	100	26	1	10	0	0	0	1500	20
2	4	B	Ks 400 s	12	300	7	1	10	0	0	0	2000	20
5	5	B	Ks 400 s	16	200	15	1	10	0	0	0	2000	20

Kort beskrivning av miljön.

ArmServ1- programmet är ett program som ligger på samma katalog som AutoCAD. Programmet kopplas till AutoCAD med DDE-länkar och fungerar både som Server och Klient. ArmServ1 använder sedan AutoLisp program för att göra olika saker i AutoCAD, t.ex. rita armeringsjärnen.

Huvudmålet med detta program är att kunna söka i databasen och hjälpa användaren att välja nummer för armeringsjärnen som ritas i AutoCAD.

Överst i dialogrutan finns det tre knappar som vanligt. Dialogrutan är definierad så att den alltid sitter främst på skärmen d.v.s. inget annat fönster kan komma ovanpå det. Man kan minimera fönstret, maximera eller stänga av programmet med de tre knapparna.

Nedanför detta ligger menyerna som fungerar på vanligt sätt och beskrivs mer detaljerad senare i kapitlet.

Nedanför menyerna är rutor där användaren kan skriva in information om armeringsjärn som ritas eller definiera hur han vill filtrera datan i databasen. Programmet använder innehållet i dessa rutor för att spara ett nytt armeringsjärn till databasen eller ändringar som har gjorts. Innan armeringsjärnet sparas till databasen granskas data i dessa rutor och om det saknas data eller att det finns fel data får användaren ett meddelande om detta.

Nedanför kommer ett textfält som programmet använder för att ge användaren meddelande om olika saker beroende på vad användaren vill och kan göra på olika stadium.

Bredvid textfält kan användaren definiera filtertolerans, dvs. när man vill filtrera listan kan man bl.a. definiera längder (för Typ B längd **a** och **b**).

Nederst i fönstret är lista av armeringsjärn som finns i databasen. Listans längd beror på hur användaren har definierad filtren.

I nuläget finns bara möjlighet att rita B-järn med denna metod.

Använda menyerna i ArmServ1.

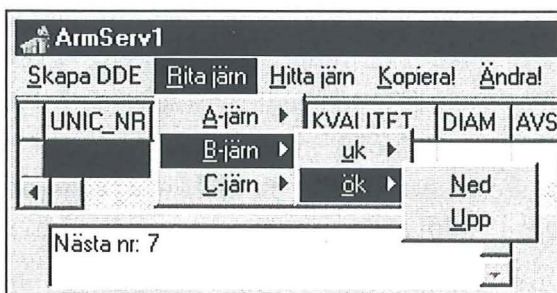
Skapa DDE!

Innan man kan rita armeringsjärn, hitta det eller ändra information om ett armeringsjärn då måste man skapa en DDE länk mellan ArmServ1 och AutoCAD. Det gör man om man väljer att Skapa DDE.

Om AutoCAD inte redan är öppet startas AutoCAD. I AutoCAD ser man till att AutoLisp rutinerna som behövs laddas automatiskt när AutoCAD startas.

I textfältet kommer meddelande om DDE kopplingen lyckades eller inte.

Rita järn



Bredvid S skapa DDE kommer R ita järn. Om man klickar på det kan man välja mellan de järntyper som finns definierade. I nuläget finns bara möjlighet att rita ett armeringsjärn, dvs B-järn, överkant med nedåt riktad längd **b**.

När man väljer det körs ett AutoLispprogram i AutoCAD och programmet frågar efter information om järnet och hur det skall ritas.

Först frågas användaren om armeringsstångens första referenspunkt. För B-järn är det vinkeländan. Användaren kan mata in en relativ förflyttning eller ingen förflyttning.

Sen frågas om armeringsstångens andra referenspunkt. Användaren kan peka med hårkorset men inmatning från tangentbordet är också möjligt. Om inmatning sker från tangentbordet frågas även efter en vinkel tills en jämn kvadrant angivits.

Sen frågar programmet efter områdesmarkeringens första punkt, denna styr även placering i sida. Sedan frågas efter områdesmarkeringens andra punkt, denna styr enbart längd.

Järnet, måttinjer osv ritas sedan automatiskt på lämpligt lager. Där efter visas i rutorna nedanför menyerna den data som skapades i AutoCAD, dvs armeringsjärn av

typ B, avstånd mellan järnen, antal järn och sedan längd a och längd b. Listan nedanför innehåller alla armeringsjärn som finns i databasen av samma typ.

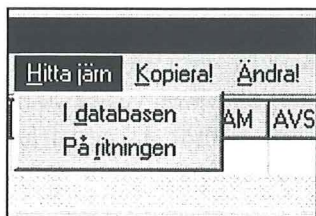
Användaren kan lägga till mer information om järnet, t.ex. diameter och kvalitet. Listan filtreras sedan vidare tills önskade armeringsjärn sållats fram. Dessa kan ha samma egenskaper som järnet som just ritats.

Om det finns armeringsjärn i databasen med samma egenskaper kan användaren helt enkelt välja det och då får nya järnet exakt samma egenskaper och samma littra.

Om det inte finns armeringsjärn i databasen med de egenskaper som nya järnet skall ha kan användaren fråga efter nästa lediga littra och själv definiera alla egenskaper som det nya järnet ska ha.

När alla egenskaperna är definierade sparas informationen om järnet till databasen och en länk mellan järnet på ritningen och nya raden i databasen skapas. Sen skrivs passande text på ritningen bredvid järnet.

Hitta järn



Om man väljer att hitta armeringsjärn i databasen måste man veta var järnet ligger på ritningen. Därför kan man i AutoCAD välja ett eller flera armeringsjärn på ritningen. Sedan visas i textfältet det unika numret för alla armeringsjärn som valdes och de fem första (sorterad efter unika numret) visas i listan nederst i fönstret. Huvudmålet med detta är att välja endast ett armeringsjärn för att kunna

kopiera det eller ändra informationen i databasen.

Om man väljer att hitta armeringsjärn på ritningen då markeras alla armeringsjärn på ritningen som finns i listan. Det är därför viktigt, om man bara vill välja ett eller ett fåtal armeringsjärn, att filtrera listan innan man väljer att hitta armeringsjärn. Sedan körs zoom-kommandot i AutoCAD till lämplig skala på första järnet i listan.

Kopiera!

När man har filtrerat listan till endast ett armeringsjärn aktiveras detta kommandot, eftersom man bara kan kopiera ett armeringsjärn i taget. Det är viktigt att använda detta kommandot om man vill kopiera armeringsjärn eftersom programmet ser till att databasen uppdateras också. Detta kommando fungerar som Copy i AutoCAD, förutom att programmet hämtar nästa lediga unika nummer och frågar sedan efter insättningspunkt, riktning o.s.v. När järnet har kopierats sparas det till databasen. Om man vill ändra järnets utsträckning, centrumavstånd eller antal används kommandot Ändra.

Ändra!

Genom att använda Filtrera och Hitta järn kan man välja armeringsjärn som ska ändras. Man väljer järnet i listan och därefter väljer man kommandot Använda data!. Då skrivs all data i rutorna ovanför och man har tillgång till all informationer om järnet.

Därefter kan man ändra data i rutorna som man vill. Sedan väljer man att Filtrera data för att kolla om det finns armeringsjärn i databasen som har samma egenskaper som det ändrade järnet. Om det finns armeringsjärn i databasen kan man helt enkelt välja det och sedan välja Använda data! och därefter Ändra! för att uppdatera databasen med nya informationen. Järnet har fortfarande samma unika nummer men får samma nummer som järnet som hittades i databasen.

Om det inte finns något armeringsjärn i databasen med samma egenskaper då kan man välja kommandot Nytt nummer! för att få nytt nummer för järnet.

Om ändringen påverkar geometrin (själva ritningen) väljs järnet på ritningen och användaren kan ändra järnet med vanliga AutoCAD-kommandon. Om ändringen påverkar texten på ritningen uppdateras den automatisk genast och raden sparas till databasen.

Ta bort!

Om man vill ta bort armeringsjärn från ritningen då använder man Hitta järn för att hitta de armeringsjärn som ska tas bort. Man kan peka på armeringsjärn på ritningen eller välja en rad i databasen.

När man väljer kommandot då får man meddelande om man verkligen vill ta bort järnen och lista med de unika nummern som järnen har. Om man är säker väljer man Ta bort igen och då försvinner järnen och alla objekten som tillhör järnet från ritningen och raden ur databasen. Därefter uppdateras filtret.

Filtrera data!

Som beskrivs ovan används detta kommandot till att minska antal rader i listan. Filtreringsgraden beror på hur mycket rutorna ovanför är ifyllda.

Det måste finnas en Typ definierad för att kunna filtrera listan över huvud taget. Det är meningslöst att filtrera rader i databasen till en lista med olika typer av armeringsjärn.

Om man definierar typ, kan man filtrera listan med Nummer (Arm_Nr), Diameter, Gjutetapp och längder.

Man kan definiera filtertolerans på längderna (t.ex. för Typ B längd **a** och **b**). Om längd **a** är lika med 1500 och längd **b** är lika med 2000 och filter tolerans är lika med 200, då filtreras listen till armeringsjärn med $1300 < \mathbf{a} < 1700$ och $1800 < \mathbf{b} < 2200$.

Om man vet det unika numret kan man skriva det i rutan och sedan klicka på Filtrera data.

Använda data!

Om man har hittat armeringsjärn i listan som man vill använda (ändra information om armeringsjärnet eller att använda data för ett annat armeringsjärn) väljer man att använda data. Då skrivs information om respektive armeringsjärn i rutorna nedanför menyerna.

Nytt nummer

Om det inte finns något armeringsjärn i databasen som har samma egenskaper som järnet som ritas eller ändras måste man välja ett nytt nummer (littra. Programmet söker efter nästa lediga littra och skriver det i Arm_Nr-rutan nedanför menyerna.

Spara till databas

När man har ritat ett nytt armeringsjärn och definierat alla egenskaperna som armeringsjärnet ska ha, kan man spara armeringsjärnet till databasen och texten som beskriver järnet ritas på ritningen.

Kapitel 4

DataBaser i ArmSpec1.

Det finns två kategorier av tabeller som ArmSpec1 använder.

Gemensamma data.

De första är tabellen som innehåller gemensamma data för alla projekt. De lagras på samma katalog som själva ArmSpec1-programmet. De är tre stycken och innehåller information om projekten, typbokstav för armering och armeringskvalitet m.m.

ProjectName (Databas som innehåller information om de projekt som finns definierade) Dbase IV format.

Nr:	Dbase IV	Förklaring	Typ	Anteckningar.
1	FILE	Tablåns namn	C(20)	Sträng
2	NAMN	Projektets namn	C(30)	Sträng
3	ADRESS	Projektets adress	C(30)	Sträng
4	BESKRIV1	Beskrivning av projektet	C(30)	Sträng
5	BESKRIV2	Beskrivning av projektet	C(30)	Sträng
6	ANSVAR	Vem som är ansvarig	C(30)	Sträng

INDEX på NAMN.

TypName (Databas som innehåller information om de järntyper som finns definierade) Dbase IV format.

Nr:	Dbase IV	Förklaring	Typ	Anteckningar.
1	TECKEN	Bokstav enligt SS	C(2)	A, B, C ...
2	ANTECKN	Beskrivning	C(50)	Frivillig sträng

INDEX på TECKEN och ANTECKN.

KvalName (Databas som innehåller information om armeringsjärns-kvaliteter som finns definierade). Dbase IV format.

Nr:	Dbase IV	Förklaring	Typ	Anteckningar.
1	NR	Nummer	N(11)	Radnummer för kvalitet
2	NAMN	Kvalitetens namn	C(10)	Sträng
3	ANTECKN	Anteckningar	C(50)	Frivillig text.
4	DENSITET	Stålets densitet	N(5)	För att beräkna kg.
5	B90_6	Bygel < 90° 6 mm	N(3)	För att beräkna bockkorr.
6	B90_8	Bygel < 90° 8 mm	N(3)	För att beräkna bockkorr.
7	B90_10	Bygel < 90° 10 mm	N(3)	För att beräkna bockkorr.
8	B90_12	Bygel < 90° 12 mm	N(3)	För att beräkna bockkorr.

9	B90_16	Bygel < 90° 16 mm	N(3)	För att beräkna bockkorr.
10	AL90_6	Järn <= 90° 6 mm	N(3)	För att beräkna bockkorr.
11	AL90_8	Järn <= 90° 8 mm	N(3)	För att beräkna bockkorr.
12	AL90_10	Järn <= 90° 10 mm	N(3)	För att beräkna bockkorr.
13	AL90_12	Järn <= 90° 12 mm	N(3)	För att beräkna bockkorr.
14	AL90_16	Järn <= 90° 16 mm	N(3)	För att beräkna bockkorr.
15	AL90_20	Järn <= 90° 20 mm	N(3)	För att beräkna bockkorr.
16	AL90_25	Järn <= 90° 25 mm	N(3)	För att beräkna bockkorr.
17	AL90_32	Järn <= 90° 32 mm	N(3)	För att beräkna bockkorr.
18	AH90_6	Järn > 90° 6 mm	N(3)	För att beräkna bockkorr.
19	AH90_8	Järn > 90° 8 mm	N(3)	För att beräkna bockkorr.
20	AH90_10	Järn > 90° 10 mm	N(3)	För att beräkna bockkorr.
21	AH90_12	Järn > 90° 12 mm	N(3)	För att beräkna bockkorr.
22	AH90_16	Järn > 90° 16 mm	N(3)	För att beräkna bockkorr.
23	AH90_20	Järn > 90° 20 mm	N(3)	För att beräkna bockkorr.
24	AH90_25	Järn > 90° 25 mm	N(3)	För att beräkna bockkorr.
25	AH90_32	Järn > 90° 32 mm	N(3)	För att beräkna bockkorr.

INDEX på NAMN och NR.

Tabeller som är unika för varje projekt.

Den andra kategorien avtabeller är de som är unika för varje projekt. De lagras på en katalog som definieras när programmet installeras. Det finns en databas för varje projekt och sökvägen till databasen ges automatiskt när användaren har valt projekt .

Tabellen innehåller information om armeringsjärnen som finns i respektive projekt och alla revideringar som har blivit gjorda i projektet.

För varje projekt skapas nya tabeller i databasen med ett namn som användaren definierar själv. Två för information om armeringsjärnen DBF och MDX och sedan två för revideringar DBF och MDX. För detta används ArmDatORG.DBF och ArmLogORG.DBF som är tomma och används som mallar för att skapa tabeller för de projekt som skapas i ArmSpec1 programmet.

ARMDATORG (Databas som innehåller information om armeringsjärnen)
Dbase IV format.

Nr:	Dbase IV	TXT	Rad	Typ	Anteckningar.
1	ARM_NR	Nr:	3	N(8)	Nummer för armeringsjärn.
2	L_MET			N(3)	Hur laddas järnet (auto,man...)
3	TYP	Armeringstyp:	2	C(2)	Sträng
4	KVALITET	Armeringskvalitet	7	C(10)	Sträng
5	DIAM	Stångdiameter:	1	N(3)	Nummer
6	ANTAL	Antal järn:	0	N(8)	Nummer
7	GRUPP			N(3)	Nummer
8	GJ_TAPP	Gjutetapp:	5	N(8)	Nummer
9	REV			C(2)	Sträng
10	B_TYP	Bygeltyp:	6	N(5)	Nummer
11	VINK_V	Vinkel vo:	8	N(5)	Nummer
12	VINK_U	Vinkel uo:	10	N(5)	Nummer
13	L_A	Stånglängd a:	4	N(7)	Nummer

14	L_B	Stånglängd b:	9	N(7)	Nummer
15	L_C	Stånglängd c:	11	N(7)	Nummer
16	L_D	Stånglängd d:	12	N(7)	Nummer
17	L_E	Stånglängd e:	13	N(7)	Nummer
18	L_F		16	N(7)	Nummer
19	L_G		17	N(7)	Nummer
20	L_X		18	N(7)	Nummer
21	L_Y		19	N(7)	Nummer
22	END_1	Ändtyp för ände 1	14	C(3)	L, M eller A
23	END_2	Ändtyp för ände 2	15	C(3)	L, M eller A
24	COMMENT			C(15)	Anteckningar

INDEX on ARM_NR och GJ_TAPP

Kolumnen Rad beskriver i vilken ordning datan ligger i textfilen som genereras i AutoCAD.

ARMLOGORG (Databas som innehåller information om revideringar)
Dbase IV format.

Nr:	Dbase IV	Förklaring	Typ	Anteckningar.
1	GJ_TAPP	Gjutetapp:	N(8)	Nummer
2	REV	Rev. bokstav	C(2)	A, B, C, D, E, F, G, H.
3	ANT	Antal revideringar	N(3)	Nummer
4	DATUM	När reviderad	D	Date format
5	SIGN	Signatur	C(4)	Sträng
6	ANTECKN	Anteckningar	C(20)	Frivillig comments

INDEX on GJ_TAPP och REV.

DataBas i ArmServ1.

ArmServ1-programmet är bara en prototyp och används med fiktivt data. Databasen är uppbyggt på samma sätt som **ARMDATORG** med tillägg av UNIC_NR och AVST fälten. När alla databaserna omstruktureras och ArmSpec1 och ArmServ1 blir ett och samma programmet kan även flera fält behövas.

Hjälp programmet.

Även om de två programmen ArmSpec1 och ArmLisp1 är helt skilda är hjälp programmet gemensamt för båda programmen. Tankegången med detta är att i framtiden kommer de att hänga ihop, dvs ArmLisp1 kommer att vara en del av ArmSpec1-programmet som DDE Server och då kan det vara bra att köra hjälpen från ett ställe.

Det mesta som står i hjälp filen finns i denna rapport och därför finns själva hjälpfilen inte med. Hjälpprogrammet finns på disketterna.

Hjälppilen skrivs i Microsoft Word for Windows på RTF-format. Att skriva hjälpprogram är oberoende av vilket programspråk man använder men med några programmeringsspråk följer kompilator som läser filer på RTF-format och kompilerar dessa till hjälppiler.

Hjälppilen som följer med dessa program är kompilerad i Microsoft Help Workshop.

Hur man definierar hjälppilen i Microsoft Word och hur man använder Microsoft Help Workshop hänvisas till litteratur som följer programmen.

Resource-filen

Tankegången med att använda Resoruse-filer är att kunna redigera programmet utan att behöva bläddra igenom all programmeringskoden. Resource-filen med ArmSpec1-programmet innehåller all text som användaren kan läsa i menyer och hjälpraden som visas nederst i huvudfönstret.

Resource-filen kan man definiera i en vanlig texteditor och så använder man Borland Resoruce Compiler för att kompilera resource-filen.

I själva Delphi programmet definierar man unika nummer för varje text sträng. När Delphi programmet kompileras och länkas, länkas resource-filen till Delphi programmet.

Det visade sig att denne process tar längre tid än om man definierar textsträngen direkt i programmet. Därför finns inte alla textsträngar i programmet i resource-filen. I nästa version av programmet skulle man kunna tänka sig att alla text som finns i programmen definieras i resource-filen och då blir det mycket enkelt att ändra text och även översätta programmet till ett annat språk, t.ex. engelska.

Kapitel 5.

Hur programmen kan utvecklas vidare.

I nuläget vet man ytterligare några funktioner som man vill ha i programmen eller ändringar som man vill utföra:

- Lägga till fler alternativ för att skriva ut armeringsspecifikation. Till exempel att kunna skriva ut summeringen utan att behöva skriva ut de gjutetapper som man vill ha summeringen på.
- Definiera Bygel typ så att:
 - 0 = Vanligt armeringsjärn, bockradie hämtas automatiskt från tabell.
 - 1 = Bygel, bockradie hämtas automatiskt från tabel.
 - > 1 = Användaren bestämmer själv vilken bockradie som ska användas.
- Det visar sig att man inte behöver och faktiskt inte vill använda databas hanteringar i AutoCAD. De är för långsamma och primitiva. Det finns även snabbare databasmotorer för Delphi än DBE som följer med Delphi. Man kan försöka hitta den snabbaste som finns och använda den. Detta skapar läge för att definiera om databaserna och använda relationsdatabaser. Då blir data-hanteringens ännu effektivare.
- Samtidigt som man omstrukturerar databasen ska man fundera på att spara geometriska information om armeringsjärnen i databasen. I stället för att spara längd a för ett armeringsjärn sparas linjens ändpunkter $p1$ och $p2$ (x, y, z). Om man gör det skapas möjlighet att definiera armeringsjärnet i ArmServ1 programmet och sedan frågar programmet efter insättnings punkt och riktning och ritar armeringsjärnet helt automatiskt. Då kan programmet använda relativt avstånd mellan objekten som bildar armeringsjärnet på ritningen för att kolla om ritningen ser ut på samma sätt som armeringsjärnet är definierad i databasen. Då får man interaktiv relation mellan ritningen och databasen.
- Definiera all text som visas i programmen i ressource-filen.
- När man har löst problemet med att ArmServ1 måste ligga i samma katalog som AutoCAD-programmet, dvs att ArmServ1 kan ligga i vilken katalog som helst, kan man baka in ArmServ1-programmet i ArmSpec1-programmet. Därigenom skapas flera möjligheter att automatisera ytterligare.
- När man har lagt ihop ArmSpec1 och ArmServ1 kan man, när man definierar projekt, spara namnen på ritningarna som tillhör projektet. Då vet ArmServ1-programmet på vilken ritning armeringsjärnen befinner sig och kan ladda in rätt ritning i AutoCAD.
- En del av ArmSpec1 programmet är fortfarande på engelska i huvudsak dvs rapport delen. Detta beror på att där används direkt rutiner som finns färdigskrivna i Delphi. Man behöver att översätta dessa till svenska.

Slutsatser

ArmSpec1- programmet klarar allt och även något mer än vad som var definierat som Del-1 i detta examensarbete.

I ArmServ1-programmet finns en lösning till de flesta problemen med att koppla ihop armeringsjärnet på ritningen och i databasen. Enligt definition av Del-2 visar programmet i princip hur man kan göra det.

Som man kan se av denna rapport finns det ganska mycket kvar att göra tills man kan säga att programmen fungerar som de skulle göra när de är färdig. Det är ju en definitionsfråga när man kan säga att något program är färdigt.

I de programmen som har skrivits finns lösning på det mesta som behövs för att utveckla programmen tills en färdig produkt. Det endasom behövs är mer arbete och mer tid.

Eftersom ArmServ1 och AutoCAD länkas ihop via DDE-länkar minskar risken för att ändringar mellan versioner av AutoCAD påverkar funktionaliteten mellan programmen. Risken finns ju alltid och man behöver alltid att kolla om allt fungerar på rätt sätt när man kör igång ny version av AutoCAD.

Nu är det på väg en ny version av AutoCAD, d v s AutoCAD-14. Hur ArmServ1 fungerar med AutoCAD-14 har inte blivit testat.

Man hoppas i alla fall att examensarbetet är ett stort steg framåt i att göra konstruktörens arbete lättare, enklare och att han blir ännu mer effektivare.

Bilagor

Bilaga A, Installera programmen

Programmen finns på disketter som följer denna rapport. Man kör setup-programmet på disk 1 och följer sedan de instruktioner som visas på skärmen.

ArmSpec1

ArmSpec1 programmet kan ligga på vilken katalog som helst. När man installerar programmen definieras de kataloger som behövs. Man följer bara helt enkelt instruktionerna i installationsprogrammet.

ArmServ1

ArmServ1-programmet måste ligga i samma katalog som AutoCAD-programmet för att allt ska fungera på rätt sätt. Databasen som tillhör ArmServ1-programmet kan faktiskt ligga på vilken katalog som helst. Installationsprogrammet sparar ArmServ1-programmet och databasen till samma katalog som ArmSpec1-programmet använder.

Det enda som man behöver att göra när installationen är klar är att kopiera ArmServ.exe och Bjarn.lsp till samma katalog som AutoCAD.EXE ligger i.

DDE kopplingarna görs helt automatiskt så användaren behöver inte bry sig om dessa.

AutoLisp

Alla AutoLisp-rutinerna ligger i Bjarn.lsp filen. Filen måste ligga på AutoCAD katalogen och det är enklast om man alltid laddar programmen när AutoCAD startas.

Bilaga B, Utskrift från möte 970121**Examensarbete: 1-97**

Benämning: Möte nr 1-97
Datum: 970121
Plats: Konferensrum, BODAB
Handl. nr:
Närvarande: Kjell Kristiansson (KKR)
Pär Hagberg
Birgir Carlsson

Distribution: Närvarande

NULÄGET

- P1.01 BCA har kommit igång med Delfi-programmeringen. En prototyp med layout och vissa funktioner finns färdig. Databasen, Paradox, är implementerad i Delfi.
Programmet skall genom databasen hålla ordning på indata från textfilen genererad i AutoCAD av LISP-program. I programmet skall beräkningar göras för korrekationer, längder och vikter. En rapport skall genereras enl SS.
AutoCAD och Delfi-programmet är helt skilda och kommunicerar åt ett håll via en text-fil.

PROGRAMMETS LAYOUT OCH FUNKTION

- P1.02 Programmet startas och automatiskt frågas användaren efter önskat projekt.
Önskat projekt väljes i en listruta genom att man markerar databasfilen. Det är alltså bara intressant att behandla ett projekt åt gången, projekten behöver aldrig hanteras samtidigt.
När projektet är valt visas en tabell över befintlig indata (read only).
Här finns nu tre val: Import av nya data, Manuell inmatning av nya data samt generering av rapport.

IMPORT AV NYA DATA

Önskad textfil väljes. Denna har namn efter ritningsfil. När filen är importerad till databasen raderas den.

Vid import skall man kunna välja Visa manuella järn, järn efter gjutetapp samt visa alla järn

Programmet kontrollerar:

Finns järnet ej -> Lägg till i databas

Finns järnet samma egenskaper -> Lägg till i databas

Finns järnet men med andra egenskaper -> Varningsruta med befintligt järn och nytt järn visade samt en fråga till användaren: Ersätta? Om inte hamnar järnet i en slask som kan skrivas ut men som inte kommer med i den riktiga rapporten. Man vill ju kunna kontrollera mot ritningen i efterhand.

Är järnet officiellt utskickat, dvs har rapporten blivit en gällande handling, skall programmet fråga om revideringsbokstav vid ersättning.

Järn märkta med gjutetapp 0 skall filtreras bort

Data som kommit in automatiskt skall ej gå att editera manuellt?

MANUELL INMATNING AV NYA DATA

Vid manuell inmatning skall man kunna välja Visa manuella järn, järn efter gjutetapp samt visa alla järn.

Inmatning sker via dialogruta, ej i databastabellen.

Programmet kontrollerar:

Finns järnet ej -> Lägg till i databas

Finns järnet automatiskt inmatat -> Får ej ändras

Finns järnet manuellt inmatat -> Får ändras

RAPPORTGENERERING

Rapporter genereras på liggande A4 enl Svensk Standard med plats för ca 20 järn per sida. Var gjutetapp får en egen sida med eget nummer. Innehåller en gjutetapp fler än 20 järn blir numreringen för etapp 1: 1.1, 1.2 osv

Utskrift skall kunna ske av en eller flera gjutetapper.

Rapporten skall dateras manuellt och är från då officiell. Järnen skall märkas med revideringsbokstav om rapporten är daterad, i tidigare skede märks reviderade järn med stjärna. Rapporten skall märkas med senaste revidering och datum.

DATABASEN

Databasen är Paradox men finns stöd för Access är detta att föredra.

Det skall finnas en databas per projekt och sökvägen till databaserna skall kunna ges av användaren om möjligt. Textdatafilerna lagras i ett speciellt underbibliotek till armeringsprogrammet.

ÖVRIGT

- P1.03 Det stora jobbet är nu att tolka beräkningarna som ligger bakom genereringen av rapport från textindatafilern. Kjell hjälper gärna till om det är frågor.

TIDER

- P1.04 Förhoppningsvis träffas vi om en månad för att titta på delar av det ovanstående.
Birgir beräknar vara helt färdig i juli-augusti då han åker hem till Island.

Bilaga C, Typblad för bockning

<p>A</p>	<p>B</p>	<p>C</p>	<p>D</p>	<p>E</p>	<p>EX</p>	
<p>F</p>	<p>G</p>	<p>H</p>	<p>J</p>	<p>K</p>	<p>L</p>	
<p>LX</p>	<p>M</p>	<p>N</p>	<p>NX</p>	<p>O Spiral sammantryckt Antal varv = X Stigning per varv = Y</p>	<p>Q</p> <p>(Yttre radie)</p>	
<p>R</p> <p>$b \approx 3,3 y$</p>	<p>S</p> <p>Klipplängd $\approx a + b + 0,5 y$</p>	<p>SH</p> <p>$d = b + c + 100$ Klipplängd $\approx a + b + c + d + 0,5 y$</p>	<p>SX</p> <p>Perspektiv</p>	<p>T</p>	<p>U</p> <p>Symmetrisk</p>	
<p>V</p> <p>Symmetrisk</p>	<p>W</p>	<p>X XX</p> <p>Perspektiv</p>	<p>Z</p>	<p>AB</p>		
<p>AC</p>			<p>AD</p>		<p>AE</p>	
<p>Mått inom \bigcirc se anvisningar punkt B 10 Ändförankringar: Ändkrok vänd likadant som i typfiguren anges med L och ändkrok vänd åt motsatt håll med M. Ankringsring anges med A. Bockningsmått avser ytterkonturer</p>			<p>AF</p>		<p>AG</p>	
<p>Objekt</p>			<p>Objekt</p>		<p>Objekt</p>	
<p>Utförd av</p>	<p>Granskad av</p>	<p>Datum</p>	<p>Senaste rev. dat.</p>	<p>Arbetsnummer</p>	<p>Blad nr</p>	<p>Av ()</p> <p>Rev</p>

Bilaga D, Vad är en armeringsspecifikation?

Armeringsspecifikationer är listor som innehåller information om armeringsjärn som är definierade i ett projekt. Informationen är uppställd enligt Svensk Standard.. Listorna skrivs ut sorterade efter gjutetapp, en lista för varje gjutetapp.

Följande information finns på listorna för varje gjutetapp:

På nästa sida finns ett exempel på en armeringsspecifikation som genereras i ArmSpec1 programmet.

TYP: Definieras av en bokstav (A-Y). Bokstäverna beskriver typer för armering enligt Svensk Standard.

Nr: Numret är unikt för varje armeringsjärn i respektive gjutetapp. Det finns också möjlighet att definiera numret som unikt i hela projektet.

Kvalitet: Beskriver av vilket kvalitet respektive armeringsjärn skall vara.

Diam: Armeringsjärnets diameter. I systemet kan 6, 8, 10, 12, 16, 20, 25 och 32 mm användas.

Klipplängd: Armeringsjärnets nettolängd. Hänsyn har tagits till korrektion av bockningsradie.

Antal grupp: Antal grupp för respektive armeringsjärn.

Antal stänger: Antalet stänger i varje grupp.

Totalt: Antal grupper multiplicerat med antalet stänger.

Ände: Kan vara L, M eller A

a - g: Längder definierade för varje armeringstyp (se typblad för bockning, Svensk standard).

x - y: Avstånd definierade för några armeringstyper (se typblad för bockning, Svensk standard).

v0 - u0: Bockningsvinklar definierade för vissa armeringstyper.

Bockningsradie: Den minsta radie som kan användas vid bockning av armering.

Korrbock: Korrektionsvärde för att erhålla armeringsjärnets totala längd.

Anmärkn: Användaren kan lägga till anmärkningar för varje armeringsjärn.

Rev: Revideringsbokstav om armeringsjärnet har ändrats på något sätt.

Efter alla armeringsjärn i respektive gjutetapp kommer sammanfattning, information om projektet och revideringar om sådana finns.

T Y P	Nr	Kvalitet	Diam (mm)	Klipp Längd (mm)	Antal			Bockningsmått(yttermått) mm resp. grader												Bock radle (mm)	Korr bock (mm)	Konstr.-del Anmärkn	Re		
					Grupp	Stänger /grupp	Totalt	Ände	a	b	c	d	e	f	g	Ände	x	y	v0					u0	
A	1253	Ks 500 ST	20	2265	1	44	44	L	2040													225		E	
A	1253	Ks 500 ST	20	2265	1	62	62		2040							L							225		E
A	1254	Ks 500 ST	20	2865	1	44	44	L	2640														225		A

Kvalitet	Ks 500 ST									Ss 260 S					Ks 600s					Rev	Ant	Dat	Sigr
Diam. (mm)	6	8	10	12	16	20	25	32	6	8	10	12	16	8	10	12	16	20	A	3	97.05.03	BK	
Längd (m)						366													B	2	97.05.09	BK	
Nom vikt (kg)						931																	
Ant. stänger						31																	
L = Ändkrok vänd likadant som I typfigur						BODAB						LOMMA 01, Lomnavägen 1						Utförd:					
M = Ändkrok vänd vänd åt motsatt håll mot I typfiguren						BOX 64 * 234 22 LOMMA						Projekt 3						Granskad:					
A = Ankringsring						TEL 040-41 61 90, FAX 040-41 31 38						Tillbyggnad											
Utskriftsdatum: 97.06.03												Gjutetapp: 16.1						Lomma, datum: 97.05.01					

TYP	Nr	Kvalitet	Diam (mm)	Klipp Längd (mm)	Antal			Bockningsmått(yttermått) mm resp. grader												Bock radie (mm)	Korr bock (mm)	Konstr.-del Anmärkn	Re					
					Grupp	Stänger /grupp	Totalt	Ände	a	b	c	d	e	f	g	Ände	x	y	v0					u0				
A	4001	Ks 500 ST	12	11780	1	35	35		11780																			
A	4002	Ks 500 ST	12	3400	1	57	57		3400																			
A	4002	Ks 500 ST	12	3400	1	62	62		3400																			
B	4003	Ks 500 ST	12	8030	1	36	36		6000	2100																	125	-70
B	4004	Ks 500 ST	12	8030	1	34	34		6600	1500																	125	-70
B	4005	Ks 500 ST	12	8330	1	18	18		6300	2100																	125	-70
A	4006	Ks 500 ST	12	6300	1	30	30		6300																			
A	4007	Ks 500 ST	12	3970	1	60	60	L	3850																			120
B	4008	Ks 500 ST	12	3900	1	32	32		3570	400																	125	-70
B	4009	Ks 500 ST	12	8930	1	17	17		6300	2700																	125	-70
A	4010	Ks 500 ST	12	3670	1	24	24		3670																			
D	4011	Ks 500 ST	12	3495	1	61	61		3000	500																	125	-5
D	4011	Ks 500 ST	12	3495	1	30	30		3000	500																	125	-5
B	4012	Ks 500 ST	12	4800	1	18	18		3570	1300																	125	-70
B	4013	Ks 500 ST	12	5400	1	18	18		3570	1900																	125	-70
B	4014	Ks 500 ST	12	3100	1	18	18		2670	500																	125	-70
B	4015	Ks 500 ST	12	2500	1	18	18		2070	500																	125	-70
B	4016	Ks 500 ST	12	1530	1	15	15		700	900																	125	-70
B	4017	Ks 500 ST	12	1930	1	15	15		500	1500																	125	-70
C	4018	Ks 500 ST	10	1585	1	57	57		700	230	700																24	-45
C	4018	Ks 500 ST	10	1585	1	30	30		700	230	700																24	-45
C	4018	Ks 500 ST	10	1585	1	15	15		700	230	700																24	-45
B	4019	Ks 500 ST	12	2230	1	35	35		400	1900																	125	-70
B	4020	Ks 500 ST	12	1330	1	15	15		500	900																	125	-70
B	4021	Ks 500 ST	12	2130	1	15	15		700	1500																	125	-70
A	4022	Ks 500 ST	12	3690	1	8	8		3690																			
A	4023	Ks 500 ST	12	3070	1	12	12		3070																			
A	4024	Ks 500 ST	16	4550	1	8	8		4550																			
C	4025	Ks 500 ST	10	1225	1	32	32		500	270	500																24	-45

T Y P	Nr	Kvalitet	Diam (mm)	Klipp Längd (mm)	Antal			Bockningsmått(yttermått) mm resp. grader											Bock radie (mm)	Korr bock (mm)	Konstr.-del Anmärkn	Re		
					Grupp	Stänger /grupp	Totalt	Ände	a	b	c	d	e	f	g	Ände	x	y					v0	u0

Kvalitet	Ks 500 ST									Ss 260 S					Ks 600s					Rev	Ant	Dat	Sigr						
Diam. (mm)	6	8	10	12	16	20	25	32		6	8	10	12	16	8	10	12	16	20										
Längd (m)			201	3171	36																								
Nom vikt (kg)			128	2904	59																								
Ant. stänger			17	265	4																								
L = Ändkrok vänd likadant som i typfigur						BODAB					LOMMA 01, Lomnavägen 1					Utförd:													
M = Ändkrok vänd vänd åt motsatt håll mot i typfiguren						BOX 64 * 234 22 LOMMA					Projekt 3					Granskad:													
A = Ankringsring						TEL 040-41 61 90, FAX 040-41 31 38					Tillbyggnad					Gjutetapp: 50.2					Lomma, datum: 97.05.01								
Utskriftsdatum: 97.06.03																													

Sammanräknad nominell vikt

Gjutetapp	Ks 500 ST							Ss 260 S					Ks 600 S					
Diam. (mm)	6	8	10	12	16	20	25	32	6	8	10	12	16	8	10	12	16	20
16. Vikt (kg)						931												
50. Vikt (kg)			128	2904	59													
			128	2904	59	931												
						BODAB BOX 64 * 234 22 LOMMA TEL 040-41 61 90, FAX 040-41 31 38					LOMMA 01, Lommavägen 1 Projekt 3 Tillbyggnad Utskriftsdatum: 97.06.03							

Armeringsförteckning

Gjutetapp	Antal sid.	Rev.	Datum	Ändr.dat.
1	1	B	1.4.1997	12.4.1997
2	1	A	1.5.1997	9.5.1997
6	1		1.5.1997	
10	1		1.5.1997	
15	1		1.5.1997	
16	1	B	1.5.1997	9.5.1997
17	1	A	1.5.1997	26.5.1997
18	1		1.5.1997	
27	1		1.5.1997	
28	1		1.5.1997	
29	1		1.5.1997	
30	1	B	1.5.1997	9.5.1997
31	1		1.5.1997	
32	1		1.5.1997	
33	1		1.5.1997	
34	1		1.5.1997	
35	1		1.5.1997	
36	1		1.5.1997	
37	1		1.5.1997	
38	1		1.5.1997	
39	1		1.5.1997	
40	1		1.5.1997	
41	1		1.5.1997	
42	1		1.5.1997	
44	1		1.5.1997	
45	1		1.5.1997	
46	1		1.5.1997	
47	2		1.5.1997	
48	1		1.5.1997	
49	1		1.5.1997	
50	2		1.5.1997	
52	1		1.5.1997	
53	1		1.5.1997	
54	2		1.5.1997	
55	2		1.5.1997	
56	1		1.5.1997	
57	1		1.5.1997	
58	1		1.5.1997	
60	1		1.5.1997	
61	1		1.5.1997	
62	1		1.5.1997	
63	1		1.5.1997	
64	1		1.5.1997	
65	1		1.5.1997	
66	1		1.5.1997	
67	1		1.5.1997	
68	1		1.5.1997	
69	1		1.5.1997	
70	1		1.5.1997	

Armeringsförteckning, fortsätt

Gjutetapp	Antal sid.	Rev.	Datum	Ändr.dat.
71	1		1.5.1997	
72	1		1.5.1997	
73	1		1.5.1997	
74	1		1.5.1997	
75	1		1.5.1997	
76	1		1.5.1997	
77	2		1.5.1997	
78	1		1.5.1997	
79	1		1.5.1997	
80	1		1.5.1997	
81	1		1.5.1997	
82	1		1.5.1997	
84	1		1.5.1997	
101	1		1.5.1997	
102	1		1.5.1997	
103	2		1.5.1997	
104	1		1.5.1997	
120	1		1.5.1997	
121	2		1.5.1997	
123	1		1.5.1997	
124	1		1.5.1997	
125	1		1.5.1997	
126	2		1.5.1997	
201	1		1.5.1997	
202	1		1.5.1997	
203	1		1.5.1997	
204	1		1.5.1997	
205	1		1.5.1997	
206	1		1.5.1997	
207	1		1.5.1997	
209	1		1.5.1997	
210	1		1.5.1997	
220	1		1.5.1997	
222	1		1.5.1997	
223	1		1.5.1997	
224	1		1.5.1997	
225	1		1.5.1997	
226	1		1.5.1997	
284	1		1.5.1997	
285	1		1.5.1997	
286	1		1.5.1997	
287	2		1.5.1997	
288	1		1.5.1997	
296	1		1.5.1997	
297	1		1.5.1997	
298	1		1.5.1997	
301	1		1.5.1997	
302	1		1.5.1997	
303	1		1.5.1997	
304	1		1.5.1997	
305	1		1.5.1997	

Bilaga E, Strukturen i programmen.

I själva programmen används långa namn (oftast engelska) som ska beskriva vad proceduren gör. Svenska namn används inte så mycket eftersom man inte vill använda (ääö).

Det finns inte kommentarer till allt som händer i varje programsats men där som det är svårast att förstå finns kommentarer på vad det är som händer.

ArmSpec1

Det finns 18 formulär i programmet av olika storlek och omfattning, femton av dem använder underformulär dvs MDIChild eller DialogBox. Proceduren som dessa formulär innehåller utför de kommando som beskrivs i kapitel två i denna rapport.

Main in 'main.pas' {MainForm}, ~ 600 programrader.

Huvudfönstret som tar hand om menyerna, översätter texten som visas på skärmen med texten i Recourse-filen, öppnar fönster och dialogbox m.m.

childwin in 'childwin.pas' {MDIChild}, ~ 15 programrader.

I Delphi finns det möjlighet att använda objektorienterad programmering. Man kan skapa ett fönster med vissa egenskaper och sedan skapa fler fönster som ärver alla dessa egenskaper och dessutom kan man lägga till flera egenskaper till nya fönstret som då blir unika för detta. I början var det meningen att bygga upp en sådan hierarki i programmet men senare visade det sig att det inte var nödvändigt. Möjligheten finns dock kvar om programmet utvecklas vidare.

About in 'About.pas' {AboutBox}, ~ 40 programrader.

En dialog som innehåller allmän information om programmet.

Strconsts in 'strconsts.pas', ~ 110 programrader.

I detta formulär definieras konstanter för de textsträngar som är definerade i Recoruce-filen. Enligt krav från Delphi måste konstanterna ligga mellan 8000 och 30000. Som beskrivs tidigare finns i nuläget bara texten för menyerna (hjälp texten, tooltips?) definerade i programmet.

ArmIndat in 'ArmIndat.pas' {MDIChild1}, ~ 850 programrader.

Fönster för att manuellt lägga till armeringsjärn i databasen eller redigera armeringsjärn som finns i databasen.

ProjName in 'ProjName.pas' {MDIChild2}, ~ 150 programrader.

Fönster för att skapa projekt eller ändra information om projekten som finns i systemet.

Kvalitet in 'Kvalitet.pas' {MDIChild3}, ~ 130 programrader.

Fönster för att skapa och definiera armeringsjärns kvalitet, densitet och bockningsradie.

ReProjOut in 'ReProjOut.pas' {ReportProjDlg}, ~ 140 programrader.
DialogBox för att skriva ut en lista över de projekt som finns i systemet.

ReTypeOut in 'ReTypeOut.pas' {ReportTypeDlg}, ~ 130 programrader.
DialogBox för att skriva ut en lista över de armeringstyper som finns i systemet.

LoadTextFile in 'LoadTextFile.pas' {LoadTextFileDlg}, ~ 380 programrader.
Sparar data från textfil efter att den har blivit granskad. I DialogBoxen kan användaren bestämma hur armeringsjärnet ska sparas till databasen.

RepArmSpec in 'RepArmSpec.pas' {RepArmSpecDlg}, ~ 1300 programrader.
Skriver ut armeringsspecifikation. Beräknar korrektion, längder, vikt och antal stänger för varje armeringsdiameter. Kollar med revideringar.

ArmerType in 'ArmerType.pas' {MDIChild4}, ~ 80 programrader.
I detta fönster kan man lägga till och ändra de järntyper som finns i systemet.

ReKvalOut in 'ReKvalOut.pas' {ReportKvalDlg}, ~ 250 programrader.
Skriver ut en lista av de armeringskvaliteter som är definerade i systemet.

LogFileUnit in 'LogFileUnit.pas' {LogFileDlg}, ~ 160 programrader.
DialogBox där användaren kan ändra information om revideringar eller lägga till information om revideringar.

CorrUnit in 'CorrUnit.pas', ~ 320 programrader.
Innehåller korrektionsprocedurer för de flesta armeringstyper enligt Typblad för Bockning av Stänger.

GranskaUnit in 'GranskaUnit.pas' {MDIChild5}, ~ 710 programrader.
När man har valt en textfil att importera måste den granskas innan den kan sparas till databasen.

SortUnit in 'SortUnit.pas', ~ 80 programrader.
Innehåller sorteringsrutiner för att sortera bl.a. listorna i listrutorna. Använder Quick Sort.

ReRevOut in 'ReRevOut.pas' {ReRevDlg}; ~ 250 programrader.
Skriver ut revideringslista. Listan innehåller alla Gjutetapper som är officiellt utskickade.

ArmServ1

ArmServ in 'ArmServ.pas' {DDEServ}; ~ 500 programrader.
Se beskrivning i kapitel 3 hur programmet fungerar.

***AutoLisp* rutiner.**

Alla AutoLisp rutinerna ligger i Bjarn.lsp filen.

